Molecular Dynamics FF, Manual , version 0.2.2

December 8, 2011

Contents

1 Aim

Le but de ce document est de repertorier les differents algorithmes utilisés par MDFF. Notamment, definir les commandes (TAG) et leurs valeurs par défaut. Mais également, les réferences aux codes, articles originaux, livres dont sont extraits les différents algorithmes, ou preciser dans le cas contraire (i.e FMV!) ce que fait la routine point par point. Ce travail de documentation permettra de faciliter la découverte de bug qui independamment de ma volonté sont (et seront) forcément présents.

2 Theory

2.1 Molecular Dynamics

2.2 Optimisation

2.3 Phonons

2.4 Electric-Field Gradient

2.5 Radial distribution function

2.6 Binary Mixture Lennar-Jones

2.7 Kob-Andersen model

2.8 Verlet list

2.9 Units

3 Usage

MDFFonly needs one input file usually called control.F [1]

3.1 Minimum settings: the input file

The input file is composed of different sections. Each section is defined by &SECTIONNAME and &END. Everything outside the sections is ignored.

In MDFF, five sections are always required :

- &controltag ... &end → General control parameters for calculation
- &mdtag ... &end → General Molecular dynamics parameters.
- &configtag ... &end → Information on the configuration
- &fieldtag ... &end → Set the Force-Field.
- &proptag ... &end → Calculation of properties

Using this minimum settings (5 empty sections) the code runs with default values. It correspond to a molecular dynamics trajectory of 10 unit of times of a fcc lennard-jones crystal (256 atoms) at ρ =1.0 at T =1.0. In the next sextion we show the standard output of this minimum setting.

[1] the .F extension permits to get nice colors with vim

3.2 Minimum setting: standart output

```
                          \\|//
                         -(o o)-
=======================oOO==(_)==OOo=======================

          ____    ____  ______   ________  ________
         |_   \  /   _||_   _ `.|_   __  ||_   __  |
           |   \/   |    | | `. \ | |_ \_|  | |_ \_|
           | |\  /| |    | |  | | |  _|     |  _|
          _| |_\/_| |_  _| |_.' /_| |_     _| |_
         |_____||_____||______.'|_____|   |_____|

============================================================

MOLECULAR DYNAMICS ...for fun
mdff.0.2.0
parallel version
filipe.manuel.vasconcelos@gmail.com
Running on   1 nodes
time     : 2011/11/05    15:50:14
calc     : md

============================================================

periodic boundary conditions in cubic cell
verlet list used
units reduced by the number of atom
dynamic calculation
NVE ensemble --- velocity verlet integrator
with equilibration:
berendsen scaling ( is not canonical ...and so NVE)
number of steps                      =         10
timestep                             = .10000E-02
time range                           = .10000E-01
temperature                          =    1.00000
number of equilibration steps        =         10
equilibration period                 =          1
Berendsen thermo time scale          = .10000E-02
tauberendsen = dt -> simple rescale
print thermo  periodicity            =          1

============================================================

structure generated from the code
Face-centered cubic structure :  4x  4x  4
system                               : fcc
natm                                 =        256
cell parameter                       =     6.3496
volume                               =   256.0000
```

```
density                                =     1.0000
distance check subroutine
smallest distance =   1.12

============================================================

force field information :
no masses are implemented
LENNARD-JONES

       eps    /    / sigma* \ q       / sigma*  \ p \
 V = ------- |  p | ------- |    - q | -------- |   |
      q - p   \    \   r    /         \    r    /   /

USER DEFINED MODEL
cutoff  =    2.50000

A-A interactions:
sigmaAA =    1.00000
epsAA   =    1.00000
qAA     =   12.00000
pAA     =    6.00000
long range correction :         -0.535433102

paralelisation - atom decomposition
rank    0 atom    1 to  256
============================================================
... MD output
============================================================

                 TOTAL :  cpu time     0.15
                    MD :  cpu time     0.01

main subroutines:
MD:
    engforce_bmlj      :  cpu time     0.01
    engforce_coul_DS   :  cpu time     0.00
    vnlistcheck        :  cpu time     0.00
============================================================
```

3.3 Miniumim setting : Output files

For this minimum setting, the code generates 4 files:

- OUTFF : close to the standard output
- OSZIFF : thermodynamic parameters
- TRAJFF : trajectory file (empty in this case)
- CONTFF : final configuration

theses files and other output files are described in a following section.

3.4 Input file : Description

3.4.1 &controltag

Where : `SUBROUTINE control_init` in `control.f90`

Parameter	Type	Short Description	Default value
`lbmlj`	(LOGICAL)	set the binary mixture lennard-jones potential (see section 2.6)	`.TRUE.`
`lcoulomb`	(LOGICAL)	set coulombic potentials (to be tested)	`.FALSE.`
`lvnlist`	(LOGICAL)	use verlet neighbour list (see section 2.8)	`.TRUE.`
`lstatic`	(LOGICAL)	static calculation	`.FALSE.`
`lpbc`	(LOGICAL)	use periodic boundary conditions (if .FALSE. should be tested)	`.TRUE.`
`lreduced`	(LOGICAL)	reduced the output quantities by the total number of particules Comment: this has nothing to do with the units of the calculated quantities Units are discussed in section 2.9	`.TRUE.`
`lshiftpot`	(LOGICAL)	use a shifted potential (maybe redundant with ctrunc)	`.TRUE.`
`ltest`	(LOGICAL)	get round of the main code (for test purpose only)	`.FALSE.`
`lrestart`	(LOGICAL)	restart calculation read previous positions and velocities	`.FALSE.`

`calc`	(CHARACTER)	`'md'` : molecular dynamics simulation (see section 2.1) `'opt'` : unconstrained optimization of configuation saved in TRAJFF file (see section 2.2) `'vib'` : 'vib+fvib', 'vib+gmod', 'vib+band', 'vib+dos' : phonons related (see section 2.3) `'efg'` : Electric Field Gradient calculation from configuration in TRAJFF (see section 2.4) `'efg+acf'` : EFG auto-correlation function from EFGALL file (see section 2.4) `'gr'` : radial distribution function of configurations in TRAJFF (see section 2.5)	`'md'`
`dgauss`	(CHARACTER)	algorithm for gaussian distribution (no fondamental differences test purpose only) (see section **??**) `'boxmuller_basic'` : also known as the cartesian form `'boxmuller_polar'` : supposed to be faster than the basic method because it is simpler to compute. `'knuth'` : knuth algorithm cited by [1]	`'boxmuller_basic'`
`longrange`	(CHARACTER)	algorithm for long-range interactions. It will be used for coulombic potential calculation and Electric-Field Gradient calculation, if both are set together. `'direct'` : Direct summation (can be used for EFG but difficult to converge for columbic potential) `'ewald'` : Ewald summation technique	`'ewald'`
`cutoff`	(DBLE)	real-space cut-off for lennard-jones potentials	`2.5`
`skindiff`	(DBLE)	verlet-list cutoff (cutoff + skindiff)	`0.1`

3.4.2 &mdtag

Where : `SUBROUTINE md_init` in `md.f90`

Parameter	Type	Short Description	Default value
`ltraj`	(LOGICAL)	save trajectory in file TRAJFF	`.FALSE.`
`integrator`	(CHARACTER)	algorithm for dynamic integration (see section **??**) `'nve-vv'` : NVE velocity-verlet `'nve-lf'` : NVE leap-frog `'nve-be'` : NVE beeman `'nvt-and'`: NVT Andersen `'nvt-nh'` : NVT Nosé-Hoover `'nvt-nhc2'` : NVT Nosé-Hoover two chains `'nve-lfq'` : NVE leap-frog + equilibration	`'nve-vv'`
`setvel`	(CHARACTER)	velocity distribution `'MaxwBoltz'` : Maxwell-Boltzmann distribution `'uniform'` : Uniform distribution (test purpose)	`'MaxwBoltz'`
`npas`	(INTEGER)	total number of time steps	`10`
`nequil`	(INTEGER)	total number of equilibration steps	`10`
`nequil_period`	(INTEGER)	equilibration period	`1`

nprint	(INTEGER)	print thermo info to standard output and OUTFF (period)	1
fprint	(INTEGER)	print thermo info to file OSZIFF (period)	1
itraj_start	(INTEGER)	write trajectory from step itraj_start	1
itraj_period	(INTEGER)	write trajectory each itraj_period steps	10000
spas	(INTEGER)	save configuration each spas steps in file CONTFF	1000
dt	(DBLE)	time step for integration	0.001
temp	(DBLE)	temperature	1.0
nuandersen	(DBLE)	characteristic frequency in andersen thermostat	NULL
tauberendsen	(DBLE)	characteristic time in berendsen thermostat (simple rescale if tauberendsen = dt)	dt
Qnosehoover	(DBLE)	Q parameter in Nose-Hoover two chains	NULL

3.4.3 &configtag

Where : SUBROUTINE config_init in config.f90

Parameter	Type	Short Description	Default value

lgenconf	(LOGICAL)	if .TRUE. generate the configuration if .FALSE. read configuration from file POSFF if calc='md' TRAJFF if calc='opt'or 'efg' ISCFF if calc='vib'	.TRUE.
lfcc	(LOGICAL)	for lgenconf=.TRUE. generate face-centered cubic lattice	.TRUE.
lsc	(LOGICAL)	for lgenconf=.TRUE. generate simple cubic lattice (primitive)	.FALSE.
lbcc	(LOGICAL)	for lgenconf=.TRUE. generate body-centered cubic lattice	.FALSE.
system	(CHARACTER)	system name	'UNKNOWN'
struct	(CHARACTER)	crystal structure for fcc (only make sense for ntype=2) 'NaCl' 'random'	'random'
ntype	(INTEGER)	number of types =1 or =2	1
natm	(INTEGER)	total number of atoms	NULL
ncell	(INTEGER)	number of cell in 1D (with lfcc,lsc,lbcc)	4
xna	(DBLE)	relative composition of atom of type A	1.0

xnb	(DBLE)	relative composition of atom of type B	0.0
rho	(DBLE)	density number (if box=0.0)	NULL
box	(DBLE)	lattice parameter (if rho=0.0)	NULL

3.4.4 &fieldtag

Where : SUBROUTINE field_init in field.f90

Parameter	Type	Short Description	Default value
lKA	(LOGICAL)	use the Kob-Andersen model for BMLJ2.7	.FALSE.
ctrunc	(CHARACTER)	truncation and shift of the potential 'linear': linear correction (potential discontinuity at $r = r_{cut}$) 'quadratic': quadratic correction (potential and force discontinuity at $r = r_{cut}$)	'linear'
ncelldirect	(INTEGER)	number of neighboring cells in the direct summation (total number of cells = $ncelldirect^3$ for lcoulomb=.TRUE. and longrange='direct' not efficient for coulombic interaction	2
ncellewald	(INTEGER)	number of kpoints in the reciprocal contribution of the Ewald summation (total number of cells = $ncellewald^3$ for lcoulomb=.TRUE. and longrange='ewald'	10

alphaES	(DBLE)	Ewald screening parameter	1.0
epsXY	(DBLE)	depth of the potential well of the Lennard jones potential. XY = AA, BB or AB	1.0
sigmaXY	(DBLE)	finite distance at which the inter-particle potential is zero. XY = AA, BB or AB	1.0
qljXY	(DBLE)	exponent of the repulsive part. XY = AA, BB or AB	12
pljXY	(DBLE)	exponent of the attratcive part. XY = AA, BB or AB	6
mX	(DBLE)	mass of the particule. X= A or B (not used)	1.0
qX	(DBLE)	charge of the particule. X= A or B	qA = 1.0 qB =-1.0

3.4.5 &proptag

Where : SUBROUTINE prop_init in prop.f90

Parameter	Type	Short Description	Default value
lefg	(LOGICAL)	calculate electric field gradient on the fly during the md	.FALSE.
lgr	(LOGICAL)	calculate radial distribution function on the fly during the md	.FALSE
lstrfac	(LOGICAL)	calculate static structural factor on the fly during the md	.FALSE
lmsd	(LOGICAL)	calculate mean square displacment on the fly during the md	.FALSE

`lvacf`	(LOGICAL)	calculate velocity autocorrelation function on the fly during the md	`.FALSE`
`nprop_start`	(INTEGER)	calculate properties from step `nprop_start`	`0`
`nprop`	(INTEGER)	calculate properties each `nprop` step (for lgr and efg not sure anymore)	`1`
`nprop_print`	(INTEGER)	print properties each `nprop_print`	`nprop`

3.4.6 &opttag

Where : `SUBROUTINE opt_init` in `opt.f90`

Parameter	Type	Short Description	Default value
`optalgo`	(CHARACTER)	choose unconstrained optimization algorithm `'sastry'` : Linear minimisation via cubic extrapolation of potential Original author S. Sastry JNCASR `'lbfgs'` : Limited memory BFGS method for large scale optimisation Author: J. Nocedal [ref] `'m1qn3'` : M1QN3, Version 3.3, October 2009 Authors: Jean Charles Gilbert, Claude Lemarechal, INRIA. [ref]	`step`
`ncopt`	(INTEGER)	number of configurations in TRAJFF	`0`
`nskipopt`	(INTEGER)	number of configurations skipped in the beginning	`0`
`nmaxopt`	(INTEGER)	number of configurations to be optimize	`1`

3.4.7 &vibtag

Where : `SUBROUTINE vib_init` in `vib.f90`

Parameter	Type	Short Description	Default value
`lwrite_vectff`	(LOGICAL)	write the vector field (i.e eigenvectors) in VECTFF file	`.FALSE.`
`ncvib`	(INTEGER)	number of configurations in ISCFF to be analysed	`0`
`ngconf`	(INTEGER)	number of configurations generated with `calc='vib+gmod'`	`0`
`imod`	(INTEGER)	if `calc='vib+gmod'` generate the mode `imod`	`4`
`nkphon`	(INTEGER)	number of kpoint between ks and kf used when calc = 'vib+band'	`NULL`
`resdos`	(DBLE)	resolution in vibrational density of states distribution	`1.0`
`omegamax`	(DBLE)	maximum value in dos	`100.0`
`ksx,ksy,ksz`	(DBLE)	first kpoint	`NULL`
`kfx,kfy,kfz`	(DBLE)	last kpoint	`NULL`

3.4.8 &efgtag

Where : `SUBROUTINE efg_init` in `efg.f90`

Parameter	Type	Short Description	Default value

`lefgprintall`	(LOGICAL)	whether or not we want to print all the efg for each atoms and configurations in file EFGALL	`.FALSE.`
`lefg_it_contrib`	(LOGICAL)	if one wants to get the contribution to EFG separated in types (only `'direct'` and test purpose)	`.FALSE.`
`ncefg`	(INTEGER)	number of configurations read in TRAJFF for EFG calculation (`calc='efg'`)	`0`
`ntcor`	(INTEGER)	maximum number of steps for the efg auto-correlation function(`calc='efg+acf'`)	`10`
`ncellewald`	(INTEGER)	number of kpoints in the reciprocal contribution of the Ewald summation (total number of cells $= ncellewald^3$ for `longrange='ewald'`	`0`
`ncelldirect`	(INTEGER)	number of neighboring cells in the direct summation (total number of cells $= ncelldirect^3$ for `longrange='direct'` only	`0`
`cutefg`	(DBLE)	cut-off distance for efg calculation in for direct summation (`longrange='direct'`)	`30.0`
`alphaES`	(DBLE)	Ewald sum screening parameter	`1.0`
`resvzz`	(DBLE)	resolution in vzz distribution	`0.1`
`reseta`	(DBLE)	resolution in eta distribution	`0.1`
`resu`	(DBLE)	resolution in Ui ditribution	`0.1`

vzzmin	(DBLE)	minimum value of vzz distribution which is then between [vzzmin, -vzzmin]	-4.0
umin	(DBLE)	minimum value of umin distribution which is then between [umin, -umin]	-4.0

3.4.9 &msdtag

Where : SUBROUTINE msd_init in msd.f90 BE CAREFUL !!!

Parameter	Type	Short Description	Default value
nblock	(INTEGER)	???????? I forgot ;)	10
ibmax	(INTEGER)	???????? I forgot ;)	20
tdifmax	(DBLE)	???????? I forgot ;)	100.0

3.4.10 &grtag

Where : SUBROUTINE control_init in radial_distribution.f90

Parameter	Type	Short Description	Default value
resg	(DBLE)	resolution in the radial distribution function	

3.4.11 &vacftag

Where : SUBROUTINE vacf_init in vacf.f90 BE CAREFUL

Parameter	Type	Short Description	Default value

`it0`	(INTEGER)	???????? I forgot ;)	`1`
`tdifmax`	(DBLE)	???????? I forgot ;)	`100.0`

4 Output files description

References

[1] M. P. Allen and D. J. Tildesley, *Computer simulation of liquids*, Oxford University Press, Oxford UK, 1987.