mm_trj

V.1.0.2

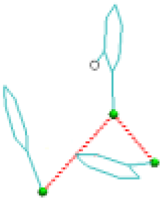
Generated by Doxygen 1.8.5

Fri Aug 30 2013 15:16:51

Contents

Chapter 1

mm_trj



1.1 Introduction

About this program:

- Program that generates trajectory files

Developer:

- Evgeniy Alekseev aka arcanis

  `<esalexeev (at) gmail (dot) com>`

License:

- GPL

1.2 How to use

Usage:

```
mm_trj -i INPUT_TRJ -t INPUT_TYPE -s NUMBER -a INPUT_ATOMS -o OUTPUT [ -tt TOTAL_TYPES ]
                                                            [ -l LOGFILE ] [ -q ] [ -h ]
Parametrs:
   -i           - input file name
   -t           - type of trajectory. Supported formats: gmx, puma
   -s           - number of trajectory steps (integer)
   -a           - input file with atom types. See file format in manual
   -o           - mask of output files
   -tt          - number of different atom types. Default is 1024
   -l           - log enable
   -q           - quiet enable
   -h           - show this help and exit
```

Chapter 2

Install

2.1 Requirements

The application mm_trj requires the following external stuff:

- cmake >= 2.8
- gcc >= 4.8

2.2 How to install

2.2.1 Linux

```
* mkdir build && cd build
* cmake -DCMAKE_INSTALL_PREFIX=/usr -DCMAKE_BUILD_TYPE=Release ../
* make
* make install
*
```

2.2.2 Windows

```
* create project file using 'cmake'
* compile project
*
```

You may also download compiled executable file for Win_x86.

Chapter 3

Changelog

V.1.0.2 (2013-07-27)

- initial release

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

File Documentation

5.1 src/add_main.c File Reference

```
#include <stdio.h>
#include "messages.h"
```

Functions

- int error_checking (const char ∗input, const char ∗input_at, const char ∗output, const int step, const int type)

  function that checks errors in input variables
- int print_message (const int quiet, FILE ∗std_output, const int log, FILE ∗f_log, const int mode, const char ∗str)

  function that prints message in log and stdout
- int set_defaults (char ∗input, char ∗input_at, int ∗log, char ∗output, int ∗step, int ∗total_types, int ∗type, int ∗quiet)

  function that sets default values of variables

5.1.1 Function Documentation

5.1.1.1 int error_checking (const char ∗ *input,* const char ∗ *input_at,* const char ∗ *output,* const int *step,* const int *type*)

function that checks errors in input variables

```
* error_checking (cell, from, input, num_needed_at, needed_at, output, to);
*
```

Parameters

input	input file name
input_at	input file name with atom types
output	output file name
step	number of trajectory steps
type	type of trajectory

Returns

11 - error in ’input_at’
12 - error in ’input’
13 - error in ’output’
14 - error in ’step’
15 - error in ’type’
0 - exit without errors

5.1.1.2 int print_message (const int *quiet,* FILE ∗ *std_output,* const int *log,* FILE ∗ *f_log,* const int *mode,* const char ∗ *str*)

function that prints message in log and stdout

```
* print_message (quiet, stdout, log, f_log, 0, str);
*
```

Parameters

quiet	status of quiet-mode
std_output	stdout
log	status of log-mode
f_log	log file
mode	number of message in "messages.c"
str	additional text in message

Returns

0 - exit without errors

5.1.1.3 int set_defaults (char ∗ *input,* char ∗ *input_at,* int ∗ *log,* char ∗ *output,* int ∗ *step,* int ∗ *total_types,* int ∗ *type,* int ∗ *quiet*)

function that sets default values of variables

```
* set_defaults (input, input_at, &log, output, &step, &type, &quiet);
*
```

Parameters

input	input file name
input_at	input file name with atom types
log	status of log-mode
output	output file name
step	number of trajectory steps
total_types	number of different atom types
type	type of trajectory
quiet	status of quiet-mode

Returns

0 - exit without errors

5.2 src/atom_types.c File Reference

```
#include <stdio.h>
```

Functions

- int reading_atoms (const char ∗input_at, int ∗num_types, int ∗num_mol, int ∗num_atoms, char ∗ch_atom_types, int ∗atom_types, const int total_types)

    function that reads atom types from input file

5.2.1 Function Documentation

5.2.1.1 int reading_atoms (const char ∗ *input_at,* int ∗ *num_types,* int ∗ *num_mol,* int ∗ *num_atoms,* char ∗ *ch_atom_types,* int ∗ *atom_types,* const int *total_types*)

function that reads atom types from input file

```
* reading_atoms (input_at, &num_types, num_mol, num_atoms, ch_atom_types, atom_types,
*                total_types);
*
```

Parameters

input_at	input file name with atom types
num_types	number of molecule types
num_mol	massive of number of molecules of selected type
num_atoms	massive of number of atoms of selected molecule
ch_atom_types	massive of char atom types
atom_types	massive of atom types
total_types	number of different atom types

Returns

1 - error in opening file
2 - error in file format
3 - memory error
0 - exit without errors

5.3 src/main.c File Reference

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "add_main.h"
#include "atom_types.h"
#include "messages.h"
#include "print_trj.h"
#include "read_gmx.h"
#include "read_puma.h"
```

Functions

- int main (int argc, char ∗argv[])

5.3.1 Function Documentation

5.3.1.1 int main (int *argc,* char ∗ *argv[]*)

Returns

1 - error in error_checking
2 - input file does not exist
3 - memory error
4 - unknown flag
0 - exit without errors

5.4 src/messages.c File Reference

```
#include <stdio.h>
#include <time.h>
```

Functions

- int message (const int log, const int mode, const char ∗text, FILE ∗output)

  function that prints messages to output

5.4.1 Function Documentation

5.4.1.1 int message (const int *log,* const int *mode,* const char ∗ *text,* FILE ∗ *output*)

function that prints messages to output

```
* message (log, mode, text, output);
*
```

Parameters

log	equal to 1 if print to logfile
mode	number of message
text	additional text
output	output file (may be stdout)

Returns

1 - unknown mode
0 - exit without errors

5.5 src/print_trj.c File Reference

```
#include <stdio.h>
```

Functions

- int printing_trj (const char ∗filename, const int atoms, const int num_types, const int ∗num_mol, const int ∗num_atoms, const char ∗ch_atom_types, const int ∗atom_types, const float ∗coords)

  function that prints trajectory snapshots

5.5.1 Function Documentation

5.5.1.1 int printing_trj (const char ∗ *filename*, const int *atoms*, const int *num_types*, const int ∗ *num_mol*, const int ∗ *num_atoms*, const char ∗ *ch_atom_types*, const int ∗ *atom_types*, const float ∗ *coords*)

function that prints trajectory snapshots

```
* printing_trj (filename, atoms, num_types, num_mol, num_atoms, ch_atom_types,
*               atom_types, coords);
*
```

Parameters

filename	output file name
atoms	number of atoms in system
num_types	number of molecule types
num_mol	massive of number of molecule of selected type
num_atoms	massive of number of atoms of selected molecule
ch_atom_types	massive of char atom types
atom_types	massive of atom types
coords	massive of coordinates

Returns

0 - exit without errors

5.6 src/read_gmx.c File Reference

```
#include <math.h>
#include <stdio.h>
#include "print_trj.h"
```

Functions

- int translate_coords (const float coords, const float cell, float ∗trans)

  funtion that translates coordinate
- int rw_gmx (const char ∗input, const int step, const char ∗output, const int num_types, const int ∗num_mol, const int ∗num_atoms, const char ∗ch_atom_types, const int ∗atom_types, float ∗coords)

  function that read GROMACS trajectory file and write to output

5.6.1 Function Documentation

5.6.1.1 int rw_gmx (const char ∗ *input*, const int *step*, const char ∗ *output*, const int *num_types*, const int ∗ *num_mol*, const int ∗ *num_atoms*, const char ∗ *ch_atom_types*, const int ∗ *atom_types*, float ∗ *coords*)

function that read GROMACS trajectory file and write to output

```
* rw_gmx (input, step, output, num_types, num_mol, num_atoms, ch_atom_types,
*         atom_types, coords);
*
```

Parameters

input	input file name
step	number of trajectory steps
output	mask of output files
num_types	number of molecule types
num_mol	massive of number of molecule of selected type
num_atoms	massive of number of atoms of selected molecule
ch_atom_types	massive of char atom types
atom_types	massive of atom types
coords	massive of coordinates

Returns

1 - file does not exist
0 - exit without errors

5.6.1.2 int translate_coords (const float *coords,* const float *cell,* float ∗ *trans*)

funtion that translates coordinate

```
* translate_coords (coords[3*i+j], cell[j], trans);
*
```

Parameters

coords	coordinate
cell	cell size
trans	massive of translated coordinates

Returns

0 - exit without errors

5.7 src/read_puma.c File Reference

```
#include <stdio.h>
#include "print_trj.h"
```

Functions

- int rw_puma (const char ∗input, const int step, const char ∗output, const int num_types, const int ∗num_mol, const int ∗num_atoms, const char ∗ch_atom_types, const int ∗atom_types, float ∗coords)
  function that read PUMA trajectory file and write to output

5.7.1 Function Documentation

5.7.1.1 int rw_puma (const char ∗ *input,* const int *step,* const char ∗ *output,* const int *num_types,* const int ∗ *num_mol,* const int ∗ *num_atoms,* const char ∗ *ch_atom_types,* const int ∗ *atom_types,* float ∗ *coords*)

function that read PUMA trajectory file and write to output

```
* rw_puma (input, step, output, num_types, num_mol, num_atoms, ch_atom_types,
*          atom_types, coords);
*
```

Parameters

input	input file name
step	number of trajectory steps
output	mask of output files
num_types	number of molecule types
num_mol	massive of number of molecule of selected type
num_atoms	massive of number of atoms of selected molecule
ch_atom_types	massive of char atom types
atom_types	massive of atom types
coords	massive of coordinates

Returns

1 - file does not exist
0 - exit without errors

Index