

mm_agl
V.1.1.0

Generated by Doxygen 1.8.5

Mon Sep 2 2013 17:07:24

Contents

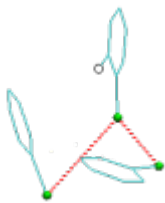
1	mm_agl	1
1.1	Introduction	1
1.2	How to use	1
2	Install	3
2.1	Requirements	3
2.2	How to install	3
2.2.1	Linux	3
2.2.2	Windows	3
3	Changelog	5
4	File Index	7
4.1	File List	7
5	File Documentation	9
5.1	src/add_main.c File Reference	9
5.1.1	Function Documentation	9
5.1.1.1	error_checking	9
5.1.1.2	print_message	10
5.1.1.3	set_defaults	10
5.2	src/coords.c File Reference	10
5.2.1	Function Documentation	11
5.2.1.1	reading_coords	11
5.3	src/main.c File Reference	11
5.3.1	Function Documentation	12
5.3.1.1	main	12
5.4	src/messages.c File Reference	12
5.4.1	Function Documentation	12
5.4.1.1	message	12
5.5	src/print_struct.c File Reference	12
5.5.1	Function Documentation	13
5.5.1.1	print_structure	13

5.6	src/read_agl.c File Reference	13
5.6.1	Function Documentation	13
5.6.1.1	reading_agl	13
5.7	src/select_mol.c File Reference	14
5.7.1	Function Documentation	14
5.7.1.1	select_molecule	14
5.8	src/set_center.c File Reference	14
5.8.1	Function Documentation	15
5.8.1.1	set_center	15

Index	16
--------------	-----------

Chapter 1

mm_agl



1.1 Introduction

About this program:

- Program that creates PDB file with chosen agglomerate

Developer:

- Evgeniy Alekseev aka arcanis

<esalexeev (at) gmail (dot) com>

License:

- GPL

1.2 How to use

Usage:

```
mm_agl -a AGL_INP -i INPUT -c X,Y,Z -o OUTPUT [ -l LOGFILE ] [ -q ] [ -h ]
```

Parameters:

-a	- input file with agglomerates (in format statgen)
-i	- input file with coordinates
-c	- cell size (float), Å
-o	- output file name
-l	- log enable
-q	- quiet enable
-h	- show this help and exit

Chapter 2

Install

2.1 Requirements

The application mm_agl requires the following external stuff:

- cmake \geq 2.8
- gcc \geq 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.1.0.3 (2013-08-30)

- Bug fixes

V.1.0.1 (2013-07-27)

- initial release

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

src/ add_main.c	9
src/ coords.c	10
src/ main.c	11
src/ messages.c	12
src/ print_struct.c	12
src/ read_agl.c	13
src/ select_mol.c	14
src/ set_center.c	14

Chapter 5

File Documentation

5.1 src/add_main.c File Reference

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

Functions

- int [error_checking](#) (const char *aglinp, const float *cell, const char *input, const char *output)
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 *aglinp, float *cell, char *input, int *log, char *output, int *quiet)
function that sets default values of variables

5.1.1 Function Documentation

5.1.1.1 int error_checking (const char * *aglinp*, const float * *cell*, const char * *input*, const char * *output*)

function that checks errors in input variables

```
* error\_checking (aglinp, cell, input, output);
*
```

Parameters

<i>aglinp</i>	agglomerate file name
<i>cell</i>	massive of cell size
<i>input</i>	input file name
<i>output</i>	output file name

Returns

- 11 - error in 'cell'
- 12 - error in 'input'
- 13 - error in 'output'
- 14 - error in 'aglinp'
- 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

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

Returns

0 - exit without errors

5.1.1.3 `int set_defaults (char * aglinp, float * cell, char * input, int * log, char * output, int * quiet)`

function that sets default values of variables

```
* set_defaults (aglinp, cell, input, &log, output, &quiet);
*
```

Parameters

<i>aglinp</i>	agglomerate file name
<i>cell</i>	massive of cell size
<i>input</i>	mask of trajectory files
<i>log</i>	status of log-mode
<i>output</i>	output file name
<i>quiet</i>	status of quiet-mode

Returns

0 - exit without errors

5.2 `src/coords.c` File Reference

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

Functions

- `int reading_coords` (const int mode, const char *filename, const int type_inter, const int *label_atom, const float *cell, int *num_mol, int *num_atoms, int *true_label_mol, int *label_mol, int *type_atoms, float *coords, char *ch_type_atoms)

function that reads coordinates from special file format

5.2.1 Function Documentation

5.2.1.1 `int reading_coords (const int mode, const char * filename, const int type_inter, const int * label_atom, const float * cell, int * num_mol, int * num_atoms, int * true_label_mol, int * label_mol, int * type_atoms, float * coords, char * ch_type_atoms)`

function that reads coordinates from special file format

```
* reading_coords (0, filename, type_inter, label_atom, cell, &num_mol, &num_atoms,
*                 true_label_mol, label_mol, type_atoms, coords, ch_type_atoms);
*
```

Parameters

<i>mode</i>	mode of reading; '1' is statgen, '2' is enviro or frad, '3' is agl
<i>filename</i>	input file name
<i>type_inter</i>	number of needed atoms (number of needed molecules)
<i>label_atom</i>	massive of needed atom types (massive of needed molecules)
<i>cell</i>	massive of cell size
<i>num_mol</i>	number of molecules
<i>num_atoms</i>	number of atoms
<i>true_label_mol</i>	massive of true numbers of molecule for atoms
<i>label_mol</i>	massive of numbers of molecule for atoms
<i>type_atoms</i>	massive of atom types
<i>coords</i>	massive of coordinates
<i>ch_type_atoms</i>	massive of char atom types

Returns

- 1 - file \$filename does not exist
- 2 - unknown mode
- 0 - exit without errors

Work blocks

reading file

translation

free memory

5.3 src/main.c File Reference

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "add_main.h"
#include "coords.h"
#include "messages.h"
#include "print_struct.h"
#include "read_agl.h"
#include "select_mol.h"
#include "set_center.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

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

Returns

- 1 - unknown mode
- 0 - exit without errors

5.5 src/print_struct.c File Reference

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


Functions

- int [print_structure](#) (const char *output, const int num_needed_mol, const int *needed_mol, const int num_atoms, const int *label_mol, const char *ch_type_atoms, const float *coords)
function that prints structure to pdb file

5.5.1 Function Documentation

5.5.1.1 int [print_structure](#) (const char * *output*, const int *num_needed_mol*, const int * *needed_mol*, const int *num_atoms*, const int * *label_mol*, const char * *ch_type_atoms*, const float * *coords*)

function that prints structure to pdb file

```
* print\_structure (output, num_needed_mol, needed_mol, num_atoms, label_mol,
*                  char_type_atoms, coords);
*
```

Parameters

<i>output</i>	output file name
<i>num_needed_mol</i>	number of needed molecules
<i>needed_mol</i>	massive of number of needed molecules
<i>num_atoms</i>	number of atoms
<i>label_mol</i>	massive of numbers of molecule for atoms
<i>ch_type_atoms</i>	massive of char atom types
<i>coords</i>	massive of coordinates

Returns

0 - exit without errors

5.6 src/read_agl.c File Reference

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

Functions

- int [reading_agl](#) (const char *aglinp, int *num_needed_mol, char *agl_class, int *needed_mol)
function that reads agglomerate from statgen-formated file

5.6.1 Function Documentation

5.6.1.1 int [reading_agl](#) (const char * *aglinp*, int * *num_needed_mol*, char * *agl_class*, int * *needed_mol*)

function that reads agglomerate from statgen-formated file

```
* reading\_agl (aglinput, &num_needed_mol, agl_class, needed_mol);
*
```

Parameters

<i>aglinp</i>	agglomerate file name
<i>num_needed_mol</i>	number of needed molecules
<i>agl_class</i>	agglomerate class
<i>needed_mol</i>	massive of numbed of needed molecules

Returns

0 - exit without errors

5.7 src/select_mol.c File Reference

```
#include <math.h>
```

Functions

- `int select_molecule (const float *centr_coords, const int num_needed_mol, int *needed_mol)`
function that selects molecules from array of translated molecules

5.7.1 Function Documentation

5.7.1.1 `int select_molecule (const float * centr_coords, const int num_needed_mol, int * needed_mol)`

function that selects molecules from array of translated molecules

```
* select_molecule (centr_coords, num_needed_mol, needed_mol);
*
```

Parameters

<i>centr_coords</i>	massive of centered coordinates
<i>num_needed_mol</i>	number of needed molecules
<i>needed_mol</i>	massive of number of needed molecules

Returns

0 - exit without errors

5.8 src/set_center.c File Reference

Functions

- `int set_center (const int num_atoms, const int num_mol, const int *label_mol, const float *coords, float *centr_coords)`
function that searchs center mass of molecules

5.8.1 Function Documentation

5.8.1.1 `int set_center (const int num_atoms, const int num_mol, const int * label_mol, const float * coords, float * centr_coords)`

function that searches center mass of molecules

```
* set_center (num_of_atoms, num_of_molecules, label_molecules, coords, centr_coords);  
*
```

Parameters

<i>num_atoms</i>	number of atoms
<i>num_mol</i>	number of molecules
<i>label_mol</i>	massive of numbers of molecule for atoms
<i>coords</i>	massive of coordinates
<i>centr_coords</i>	massive of centered coordinates

Returns

0 - exit without errors

Index

- add_main.c
 - error_checking, [9](#)
 - print_message, [9](#)
 - set_defaults, [10](#)
- coords.c
 - reading_coords, [11](#)
- error_checking
 - add_main.c, [9](#)
- main
 - main.c, [12](#)
- main.c
 - main, [12](#)
- message
 - messages.c, [12](#)
- messages.c
 - message, [12](#)
- print_message
 - add_main.c, [9](#)
- print_struct.c
 - print_structure, [13](#)
- print_structure
 - print_struct.c, [13](#)
- read_agl.c
 - reading_agl, [13](#)
- reading_agl
 - read_agl.c, [13](#)
- reading_coords
 - coords.c, [11](#)
- select_mol.c
 - select_molecule, [14](#)
- select_molecule
 - select_mol.c, [14](#)
- set_center
 - set_center.c, [15](#)
- set_center.c
 - set_center, [15](#)
- set_defaults
 - add_main.c, [10](#)
- src/add_main.c, [9](#)
- src/coords.c, [10](#)
- src/main.c, [11](#)
- src/messages.c, [12](#)
- src/print_struct.c, [12](#)
- src/read_agl.c, [13](#)
- src/select_mol.c, [14](#)
- src/set_center.c, [14](#)