THE DL_ANALYSER USER MANUAL

VERSION 2.3, SEPTEMBER 2021

Chin W Yong

Computational Chemistry Group,
Computational Science and Engineering Division,
Scientific Computing Department,
UK Research and Innovation,
Science and Technology Facility Council,
Daresbury Laboratory,
Warrington WA4 4AD,
Cheshire, UK

The Program manual assumes readers possess at least a basic knowledge of molecular simulation and potential force fields. It mainly describes the functionality of the Program. For instance, the manual describes h*ow to use*, but not *when to use*, nor to provide in-depth details of every feature available in the Program.

DL_ANALYSER is a computer program package written in C that primarily serves as a support application software tool for DL_POLY molecular dynamics simulation package. DL_ANALYSER is developed at Daresbury Laboratory by C. W. Yong. The package was developed under the auspices of the Engineering and Physical Sciences Research council (EPSRC) for the EPSRC's Collaborative Computational Project for the Computer Simulation of Condensed Phases (CCP5).

If you use the Program in your work, please include the following reference in your publication:

C.W. Yong and I.T. Todorov, *Molecules* (2018), Vol. *23*, p36 (doi:10.3390/molecules23010036)

Disclaimer

While extensive tests have been made to ensure smooth working of DL_ANALYSER and the accuracy of the result outputs, neither the STFC, EPSRC, CCP5 nor the author of the DL_ANALYSER package or its derivatives guarantee that the software package is free from error. We disclaim any responsibility for any failure, inaccuracy, harm and damage to your projects, theoretical or experimental works as a result of using DL_ANALYSER.

1 Introduction

DL_ANALYSER (the Program) is a computer program written in C. It is intended to serve as a useful utility tool to carry out post-analysis of the output files generated by DL_POLY molecular dynamics (MD) simulation package.

DL_ANALYSER can generate a wide variety of results for a wide range of different kinds of system models, from condensed matters, biological models to surface slabs. All analysis options are made available to different types of system model and DL_ANALYSER will automatically check for any restrictions and limitations of certain analysis options on the system models. However, such checks are not exhaustive and there is no restriction impose whereby an analysis option that is specific to a system model can also be used for other types of models. It is up to users to interpret the usefulness of such result output.

The Program's development concept and file structures are similar to that of DL_FIELD and therefore both programs can work in a synchronous way. This may be useful, for instance, in setting up simulation models.

1.1 File Components

DL_ANALYSER consists of three parts: the main program, the control file and the user's input index file. This is shown as follows:

(1) DL_ANALYSER programs in */source* directory together with the header file, *dl_analyser.h*.

(2) DL_ANALYSER control file (*control*). This is the master control file to select analysis options and control parameters.

(3) DL_ANALYSER input file (*input*). It contains at least one configuration files or a collection of trajectory files.

(4) Special files, *atom_list_A* and *atom_list_B* files for Group A atoms and Group B atoms, respectively.

(5) The *dl_a_path* file that specifies the directory paths of various file components, including the DL_ANALYSER *control* input filenames and their locations, relative to the DL_ANALYSER home directory. The home directory is the directory path where the DL_ANALYSER executable and *dl_f_path* files are located. The directory paths must be redefined in the *dl_a_path* file if they are moved away from the default location.

Upon a successful run, DL_ANALYSER will produce the following output files:

(1) *dl_analyser.output* : Provide general status and time execution of the analysis process.

(2) An optional trajectory output file, according to the user's specifications.

(3) A number of optional results output files, according to user's specifications.

1.2 Program Compilation

Once it is registered, users will be sent an archive file attachment called *dl_a_2.3.tar.gzip*, which contains the Program's source code.

To extract the program, type:

gunzip dl_a_2.3.tar.gzip

And then

tar -xvf dl_a_2.3.tar

A standard C compiler must be pre-installed in your operating system. To compile the Program, in the */source* directory type and enter the following command:

make

After compilation, a *dl_analyser* executable file will be produced in the *workspace/* directory. Type *./dl_analyser* to run the program. Alternative, run the script *run_dla*. The script allows user to explicitly define the maximum number of OpenMP processing threads to carry out analysis and run DL_ANALYSER program.

The */workspace* directory is the folder where you should do the analysis.

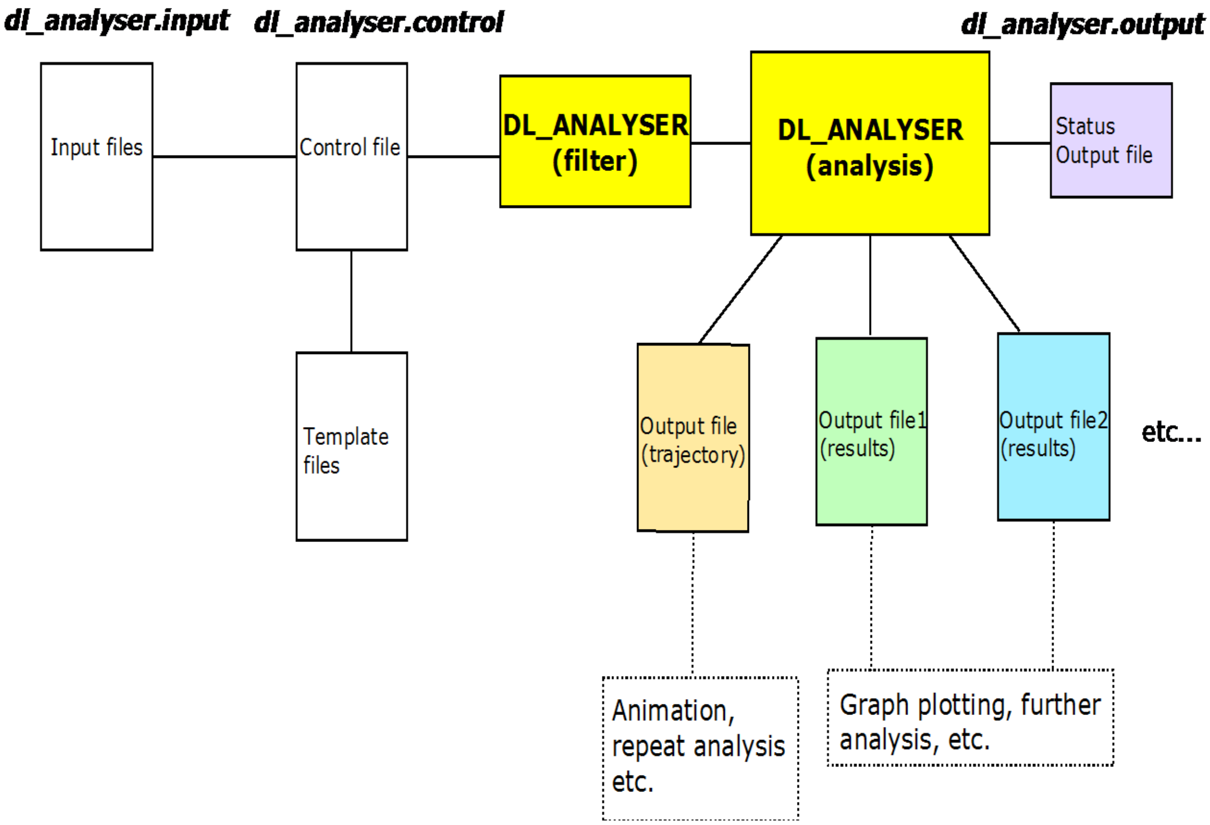
1.3 Functionality

The Program is developed with an aim to make it as simple to use as possible, while at the same time robust enough to handle a range of different molecular models, without having a need to write extra program scripts. The Program itself does not have scripting capability and all possible analysis options are shown in a single control file. To use DL_ANALYSER, user must supply at least one input file specified in the input index file. One of the most common type of input file would be the DL_POLY's HISTORY trajectory file.

When DL_ANALYSER is executed, it will read the *dl_a_path* file and look for the user's configuration files in the designated DL_ANALYSER input file. After that, particles in the user's system model will be selected, based on some filtering processes and then the analysis will be carried out according to the options selected in the *control* file. The analysed results are written out into at least one output file specified by the user.

In addition to the input file, and depending on the analysis option chosen, a supplementary file such as a template file must also provide. The *control* file will indicate the requirement of such file or, indeed, DL_ANALYSER will report an error if such file is absent.

Diagram below illustrates schematically how DL_ANALYSER functions:



Note that the dotted lines indicate involvement of third-party packages where data adjustments may be needed for final results presentations and animation purposes. However, the output trajectory file is fully compatible with DL_ANALYSER, which can be carried out for repeat or further analysis.

The boxes shaded in yellow represent the DL_ANALYSER main program components. The white solid boxes are the information provided by users.

1.4 DL_ANALYSER Features and Release History

DL_ANALYSER version 1.0 was not released but attached as a supplementary file for registered users in previous DL_Software training Workshops.

DL_ANALYSER version 1.1 (released May 2014) contains the following features:

- Reads PDB, xyz, and DL_POLY’s HISTORY, REVCON, CONFIG and STATIS
- Able to extract from HISTORY and convert to PDB and xyz.
- Periodic boundary conditions (cubic and orthorhombic only, apply to most measurements)
- Trajectory production - atoms selection, energy filtering, window size selection.
- Structural Analysis - Molecular matching, center of gravity, radius of gyration, asphericity, density, radial distribution, density profile, group distance.
- Dynamical Analysis - Kinetic energy, specific heat, center of mass velocity, temperature, temperature profiles, velocity profile, cross-correlation displacement.

- Biological Analysis - Phi and Psi dihedral.
- Defect analysis - Defect distribution, defect profiling.
- Sputter analysis
- STATIS data extraction.

DL_ANALYSER 1.2 (released May 2015) contains the following additional features:

- Periodic boundary conditions (triclinic) - on most calculations.
- Structural analysis - block analysis, order parameter, radial distribution, non-bonded dihedral distribution.
- Trajectory production - group atom extraction including its surroundings.
- Misc. Improvement on program stability and flexibility.

DL_ANALYSER 1.3 (released January 2017) contains the following additional features:

- Corrections to periodic boundary effect to ensure correct center of gravity calculations and other related calculations, as a result of groups of atoms straddling across the periodic boundary (only applies to cubic and orthorhombic).
- Improved group atom extraction for Trajectory Production. Introduce options to produce cluster atoms in spherical or cubic shapes.
- Introduce atom selection feature for results analysis as defined in the Atom Range Definition Section. (Introduce user-define files *atom_list_A* and *atom_list_B*). See Section 3.3.

DL_ANALYSER 1.4 (released July 2017) contains the following additional features:

- Introduce periodic boundary features when calculating distance between two groups of atoms.
- Introduce Interaction Analysis - Capable to detect and differentiate local atomistic interactions and describes in DANAI notation, a natural syntax notation to describe atomic interactions. Only work for carboxylic-carboxylic and alcohol-alcohol interactions only.

DL_ANALYSER 2.0 (released December 2017) contains the following additional features:

- Improvement on Interaction Analysis Section: better detection of various modes of carboxylic-carboxylic interactions.
- Hydrophobic alkyl-alkyl detection.
- Cross-correlation calculations between two types of interactions.
- Periodic boundary conditions to distance calculation between two groups (Group A and Group B).
- Mean-square displacement in Dynamical Analysis Section.

DL_ANALYSER 2.1 (released January 2019) contains the following additional features:

- Include molecule-base analysis option (as oppose to atomic-base).

- Detection for π-π benzene aromatic stacking.
- Detection for carboxylic-alcohol interactions (HB_15_20)
- Detection for aniline-carboxylic interactions (HB_20_46)
- Allows intra-interaction and inter-interaction analysis for Group A and Group B atoms.

DL_ANALYSER 2.2 (Released February 2020) contains the following additional features.

- Include molecule-base analysis option for Sputter Analysis Section.
- Detection for alcohol-aniline interactions (HB_15_46)
- Detection for water-water interactions (HB_800_800).
- Detection for ester-water interactions (HB_19_800).
- Detection for phosphate-water interactions (HB_151_800).
- OpenMP implementation on some features.

DL_ANALYSER 2.3 (Released September 2021) contains the following additional features

- Select analysis criteria for atom-based analysis: within molecules, between molecules or all atoms.
- Interaction Analysis Section: Detection for ammonium, carboxylate and water interactions. Detection for haloalkanes interactions.
- Biological Analysis Section: Allow non-amino acid for the phi-psi dihedral calculations.
- Structural Analysis Section: Angular distributions between two non-bonded bond vectors.
- Dynamical Analysis Section: Velocity autocorrelation functions (VACF) and its Fourier Transforms.
- Additional general analysis features: reduced moments.

2 DL_ANALYSER Input File

It is a text file that contains a collection of the location of the user's configuration files, of which analysis will be carried out by the Program. The file format is as follows:

```
Number of files
Path for configuration file 1
Path for configuration file 2
…
…
```

There is no restriction to the number of files can be included for analysis. DL_ANALYSER automatically recognises a number of file formats. They are as follows:

(a) HISTORY - This is the trajectory file produced by the DL_POLY molecular dynamics package. DL_ANALYSER can automatically recognise the versions that are generated by DL_POLY_2.19 onwards or DL_POLY_3 and DL_POLY_4.

(b) CONFIG, REVCON - DL_POLY input configuration files that contain a single molecular configuration.

(c) PDB format - Protein Databank Format. Usually represented for as a single macromolecular structure but can be represented as a number of molecular trajectories, of which each trajectory frame is separated by the 'END' or the 'TERMINATE' statements.

(d) The xyz format - The most compact form that contains only atom labels and the corresponding xyz coordinates.

(e) STATIS - This is the DL_POLY generated file that contains some thermodynamic information of a molecular model.

DL_ANALYSER can automatically recognise and read configuration files in the compressed *.gz* format. Although reading compressed files is slower, this can be desirable for large systems and with long time scale simulations when the trajectory files can take up large storage space.

Example below shows a typical content of the *input* file.

```
4
/usr/johnny/protein/HISTORY1
/usr/johnny/protein/HISTORY2
/usr/johnny/protein/HISTORY3.gz
/usr/johnny/protein/HISTORY4.gz

Remarks, notes, etc (Ignore by DL_ANALYSER).
```

In this example, DL_ANALYSER is expecting four input files and will read the file sequentially and automatically determine the format type for each file. Each file must be specified in a separate line. However, in order to ensure the integrity of your results, all trajectory files must be derived from the same simulation models. That is, the total number of atoms and index sequence are the same.

Since the Program only expects first four lines to contain the input file details, any information after that will be ignored by the Program. This space can be useful for including remarks, notes etc.

3 DL_ANALYSER Control File

The control file is a normal ASCII text file that can contain information up to 120 columns per row.

It contains all possible analysis options that is available in the DL_ANALYSER program. Each option contains a brief description how it is used and possible input values for each option. This is the file that will most frequently access by the users. The general features of the control file is as follows:

(a) The analysis options are categorised into several *Analysis Sections*. Each *Section* contains a separate filename for results output derived from that *Section*.

(b) The *Analysis Sections* do not follow any particular order and can be rearranged according to user's choice. However, the order of the analysis options within an *Analysis Section* and how they are categorised are important and **must not** be changed.

(c) The description after each analysis option can be modified but **must not** be removed.

(d) Information or notes can be inserted between any two *Analysis Sections* and will be ignored by DL_ANALYSER.

(e) With the exception of *Trajectory Production Section*, all analysis will be carried out according to atom range and overall conditions set up by the users, in the *Atom Range Definition*. The *Trajectory Production Section* contains its own atom range settings that are only applicable with that *Section*.

(f) Note that energy conversion feature is not yet available and it is up to the user to decide the energy unit, of which the information can be obtained from the FIELD file. For this reason, the *energy unit* specification in the *control* file will be ignored by the Program.

All *Analysis Sections* are identified by three hash lines '---' that precedes each *Section* label. This is used internally by DL_ANALYSER to indicate the type of analysis and **must not** be changed. The available *Analysis Sections* are as follows: *Trajectory Production*, *Atom Range Definition*, *Interaction Analysis*, *Structural Analysis*, *Dynamical Analysis*, *Biological Analysis*, *Defect Analysis*, *Sputter Analysis* and *STATIS Extraction*.

Each *Section* contains a master switch to either activate or deactivate the whole analysis options in the *Section*. Once it is deactivated, **all** analysis options contained within that *Section* will be ignored.

Note the periodic boundary conditions, if it is switched on, do not apply to some calculations. These are clearly indicated in each option.

3.1 The Trajectory Production Section

This *Section* specifies how to produce a series of configurations and write to a single output file. This can be subsequently used for animation purposes, for instance, by using a third party software such as VMD.

This *Section* also enable to convert DL_POLY's HISTORY and CONFIG files into *PDB* and *xyz* formats.

```
--- Trajectory Production Section
(1) 0     * Produce trajectory? (1=yes 0=no)
(2) t.xyz * Filename for trajectory output. (.xyz or .pdb, .mdcrd)
(3) none  * PDB template (needed for .pdb trajectory output)
(4) none  * Atom label to be excluded for trajectory output. Put 'none' if not require.
(5) 0     * Number of every configuration to skip
(6) 3     * 1 = Static window size, 2= dynamic window size, 3= off
(7) none  * Window size (x,y,z). Put 'none' if not require.
(8) 1 272 * Master atom index range (or put 'none')
(9) none  * Atom range: start end, cut off distance, cluster shape: 1=sphere 2=cube (none to deactivate)
(10)none  * Kinetic energy filter atom index range (or put 'none')
(11)none  * Kinetic energy filter range, put 'none' if not require
(12)0.0 0.0 -40.0  * Translation matrix on output configuration: x, y, z (assume orthorhombic cell)
```

Below list the options available in sequence as shown in a *control* file.

(1) Master switch to activate the *Section*. 1 = on, 0 = off. If it is switched to 0, then all the options and parameters contained within the Section will be ignored.

(2) Filename for trajectory output. The output file can be specified in either the *xyz* or the *PDB* formats which is automatically detected by the Program, provided the filename is ended with the extension .xyz or .pdb.

(3) The PDB template file. This must be supplied, and the Program will look for this file if the output file is specified as a PDB file in Option **(2)**. The template file must contain the atom sequence similar to the input trajectory file. The template file must always start with the first atom index 1 but does not have to contain all the atoms in the molecular system.

One way to obtain a PDB template file is to use the DL_FIELD program to set up the molecular force fields for DL_POLY runs and instruct DL_FIELD to produce the corresponding PDB file (see DL_FIELD user manual for details). In this case, the atom sequence produced from the DL_POLY's HISTORY files will be the same as the PDB template file produced by DL_FIELD.

(4) Atom label to be excluded. If the atom label is matched, this will be skipped, as for instance, one may not interest to show hydrogen atoms. Put 'none' if no exclusion is required.

(5) Frequency to write to the output file. The number every number of configurations to skip before writing to the output file. If the number is set to zero, it means every configuration will be written to file (nothing is skipped).

(6) Window type specification. This specify the type of the rectangular window, centred at the origin of the simulation box, of which atoms that fall within this region ONLY will be written to the output file. There are two types of window: The Static Window and the Dynamic Window.

The Static Window will write any atoms that are located within this window. The number of atoms for each trajectory frame may therefore be different from the other.

The Dynamic Window will only output atoms that are located within the window in the first frame of the input trajectory file and write the locations of these atoms in the subsequent trajectory to the output file. In this case, the number of the atoms and their indices are the same for each frame.

Specify '3' if no such window is required.

(7) Specify the size of the window, size in x, y and z direction centred at the origin of the simulation box. The full range would be -0.5x to +0.5x, -0.5y to +0.5y and -0.5z to +0.5z. The sizes will be ignored if 'none' is specified in Option **(6)**.

(8) Master index atom range. This indicates only the range of the atom specify will be considered. Any atoms that are outside this range will be ignored, EVEN if it is located within the window specified by Option **(6)**. Put 'none' if this is not require. In this case, all atoms in the molecular system will be considered.

(9) Extract a group of atoms and all other atoms that are located within the cutoff distance from the center of mass of the group. This option takes four values: the range of atom index that defines a number of atoms as a group, the cutoff distance and the shape of the cluster atoms: spherical (1) or cubic (2).

To deactivate this feature, put 'none'. Example usage: consider the following parameters for this feature:

18 34 8.0 1

This means atom indices 18 to 34 is considered as the user-defined atom group. The center of gravity of the atom group is first determined. After that, all other neighbouring atoms that are 8.0 Å or less from the center of gravity will be included. The end result is a spherical cluster of atoms, with the pre-selected atom group located in the middle of the cluster. The coordinates of the cluster are written in the output trajectory file (Option **(2)** above).

Information about the selected cluster atoms are written out in the *dl_analyser.output* file with the follow format:

Atom index (atom labels) - position in trajectory files - distance

The example below lists a portion of the cluster atom information:

```
…
…
Time = 1800.002000 ps
Atom index (atom labels) - position in trajectory file - distance
120 (OWS) - 18 - distance = 7.998054
121 (HWS) - 19 - distance = 7.758383
122 (HWS) - 20 - distance = 7.363076
396 (OWS) - 21 - distance = 4.360772
397 (HWS) - 22 - distance = 4.109308
398 (HWS) - 23 - distance = 5.296274
417 (OWS) - 24 - distance = 5.523879
418 (HWS) - 25 - distance = 5.275707
419 (HWS) - 26 - distance = 5.141178
564 (OWS) - 27 - distance = 5.472050
565 (HWS) - 28 - distance = 4.590524
…
…
```

The 'position in trajectory file' starts with 18 since the first 17 atoms (with the actual indices from 18 to 34) are user-selected atom group. So, in this case, atom 18, which the actual atom index in the configuration file of 120, is located at a distance of 7.998 Å from the center of gravity of the user-selected atom group.

Note that the atom list can be extracted as is and copy and paste into a separate file called *atom_list_A* or *atom_list_B* so that further analysis can be carried out on these selected atom cluster of atoms. See *Atom Range Section* for further details.

(10) The range of atom consider with kinetic energy that falls within the range specified in Option **(11)**. This option takes two values: two atom indices that define the range of atoms considered.

(11) Specify kinetic energy range. This option takes two values: the lower and upper limit of the kinetic energy. DL_ANALYSER will select and write out the selected atoms as defined in Option **(10)** provided the kinetic energy falls between the ranges specified.

Option **(10)** and **(11)** work only for HISTORY trajectory files containing forces on atoms.

(12) Translation matrix x, y, z on the selected structure as above before writing out to the output file. This option only works if the system is an open periodic or orthorhombic cell. The Program will option the periodic cell information from the *Atom Range Section*.

3.2 Atom Range Section

This *Section* is the main atom filtering system of the Program. It defines the range of atoms that will be considered for carrying out the analysis. Atom index that falls outside this range will be ignored and will not be considered in the analysis. Each range specified defines a Group of atoms and up to two Groups can be defined: Group A and Group B.

```
--- Atom Range Definition and overall conditions for analysis as below.
(1) 1 8    * Range of atom index (Group A). This must always define.
(2) 9 16   * Range of atom index (Group B), if applicable. Or put 'none'.
(3) 1      * Analysis type: 1=atom-based  2 = molecule-base
(4) 1      * Atom-based analysis criteria: 1=all 2= within molecules 3= between molecules
(5) acid 8 * Molecule-base analysis: name and no of atoms in Group A (MOLECULE A1)
(6) none   * Molecule-base analysis: name and no of atoms in Group A (MOLECULE A2, or 'none')
(7) none   * Molecule-base analysis: name and no of atoms in Group B (MOLECULE B1, or 'none')
(8) none   * For molecule-base analysis: name and no of atoms in Group B (MOLECULE B2, or 'none')
(9)all     * Range of MD time (ps) samples: t1  t2 (put 'all' if all samples to be included).
(10)1      * Assign all atoms with unit mass = 1.0 (1=yes, 0=no)
(11)0.0  0.0  0.0  * Translation marix on coordinates:x y z (assume orthorhombic cell)
(12)auto   * Periodic boundary? 0=no, other number = type of box (DLPOLY), auto = obtain from HISTORY
(13)40.0  0.000  0.0000  * Cell vector a (x, y, z)
    0.000 40.0   0.0000  * Cell vector b (x, y, z)
    0.0   0.000  40.0    * Cell vector c (x, y, z)
(14)2      * Exclude any atoms for analysis? 0=no, or Number of EXCLUDE statements shown below.
EXCLUDE 2000 to 5000
EXCLUDE 5050 to 5070
…
…
```

Below lists the options available in sequence as shown in the *control* file.

(1) Range of atom index for Group A. Must always be defined by default. The minimum value is 1 and up to the maximum value in the molecular system. Alternatively, DL_ANALYSER can select a group of atoms based on the definition listed in a special file called *atom_list_A*. In this case, instead of specifying the actual range of atom index, just insert the filename *atom_list_A*.

(2) Range of atom index for Group B. Definition of this atom group is optional. Put *none* if it is not needed. However, some analysis options may require Group B to be defined as well. The minimum value is 1 and up to the maximum value in the molecular system. Alternatively, DL_ANALYSER can select a group of atoms based on the definition as listed in the enclosed file called *atom_list_B*. In this case, instead of specifying the actual range of atom index, just insert the filename *atom_list_B.*

Examples of *atom_list_A* and *atom_list_B* are included in the *workspace/* directory as references. Basically, the only information that is required is a list of atom indices, one in each line. Any other information after an atom index in each line will be ignored. DL_ANALYSER will also ignore any line containing the hash '#' in the first column.

Note: the *atom_list_A* and *atom_list_B* files can only contain a limited number of atoms up to a maximum limit set by MAX_ATOM_LIST. Currently, this is set to 1000 by default. For fixed model structures such as surface and bulk, they can be easily defined serially. In this case, it is advisable to explicitly state the range of atom index for analysis, rather than listing the indices in a file.

(3) Analysis type. This option takes two possible values: 1 = atom-base analysis and 2 = molecule-based analysis. For value 2, additional information would be needed (see below).

Atom-base analysis treat each atom in a molecule as distinct individual entity, of which all analysis will be based upon. For molecule-based analysis, DL_ANALYSER will attempt to locate and classify all molecules in the system, according to the specification stated in Option **(5)**. After that, the center of mass of each molecule will be calculated and all analysis will be based on these center of masses.

(4) Atom-based analysis criteria. It specifies how DL_ANALYSER should do the analysis on the selected atoms, of whether they should be considered collectively or depending on the molecules to which they belong. This option only applicable if option **(3)** is set to atom-base. The analysis criteria only take three values: 1=all (consider collectively), 2=atoms within molecules, or 3=atoms between the molecules.

(5) Definition for molecule-base analysis (Molecule A1). For this type of analysis, additional information would be needed. This option takes two values: name of the molecule and number of atoms in the molecule. DL_ANALYSER assumes atoms are grouped together in the trajectory files, and the sequence order of atoms are the same for all molecules. Example usage:

benzene 12

The molecule name is '*benzene*' and contains 12 atoms in a molecule. This molecule is designated with a special label called Molecule A1. DL_ANALYSER will first scan the trajectory files and select the atom information based on the Group A definition in Option **(1)** above. It assumes the first 12 atoms in Group A belongs to a *benzene* molecule and the following 12 atoms belongs to a second *benzene* molecules, etc. If Group A consists of atoms belong to two different types of molecules, then Molecule A2 must be defined (see below).

(6) Definition for molecule-base analysis (Molecule A2). This is an optional definition which is applicable only if Group A atoms contains second type of molecules, in addition to Molecule A1.

Note: DL_ANALYSER can identify different types of molecules automatically and designated as Molecule A1 and Molecule A2 (if defined) accordingly, so long the molecular weights are different. This means it can unpick molecules of different types even if they contained the same number of atoms.

For the same reason, DL_ANALYSER cannot distinguish isomers. For systems contain isomers, then assign an isomer to Group A and another isomer in Group B.

(7) Same as option **(4)**, but applies to Group B atoms. The molecules are designated as Molecule B1.

(8) Same as option **(5)**, but applies to Group B atoms. The molecules are designated as Molecule B2.

(9) Range of simulation time for analysis. This defines the range of simulation time, in unit ps, to carry out the analysis. It takes two values, the minimum time and the maximum time. Any trajectory that falls outside the range specified will be ignored. Put 'all' if all trajectories are to be considered for all times.

(10) Define unit mass for all atoms in the system. Takes two values: 1 (yes) and 0 (no). Usually, the atomic masses are obtained in the HISTORY file. If the user does not wish to use these mass values, or if no mass data available (such as trajectory files in PDB or xyz formats), then switch this option to 1 so that all atoms will be treated as the unit mass of 1.0.

(11) Translation matrix for the coordinates before carry out the analysis.

(12) Periodic boundary condition flag. The value represents the type of periodic boundary condition for the molecular system of interest. Zero means no (open) boundary. If 'auto' is inserted, this instructs DL_ANALYSER to obtain the periodic boundary information from the HISTORY files, or the PDB files that contain the cell parameter information (CRYST statement).

(13) The following three rows of number define the cell vectors **a**, **b**, and **c** according to DL_POLY notation. If the keyword 'auto' or zero value is defined in option **(12)**, then these cell vectors will be ignored. Otherwise, the periodic boundary condition flag defined in the option **(12)** must match with the simulation box type.

(14) Number of EXCLUDE statement require to read. This defines the number of EXCLUDE statements. This will be ignored if it is zero. Otherwise, the Program will look for a number of EXCLUDE statement immediately after option **(14)**. The syntax is as follows:

```
EXCLUDE   a  to  b
EXCLUDE   c  to  d
...
```

The EXCLUDE statements impose additional filtering rules to fine tune atoms to be excluded from analysis. The EXCLUDE statements apply to both Group A and Group B atoms. Any atom index that falls within the range will be excluded from analysis.

For example, suppose the Group A consists of atoms from index number 35 to 4500 and additional EXCLUDE statements are defined as follows:

```
EXCLUDE 20 to 38
EXCLUDE 300 to 305
EXCLUDE 4505 to 5000
```

Then the first EXCLUDE statement (20 to 38 inclusive) will exclude part of the members from Group A, with index number from 35 to 38.

The second EXCLUDE statement will exclude all atoms from index number 300 to 305 from Group A.

The third EXCLUDE statement will not exclude any atoms since the range does not cover the members within Group A.

Note that the EXCLUDE statements also apply to user-selected atoms in the *atom_list_A* and *atom_list_B* files.

3.3 Structural Analysis Section

This *Section* carries out analysis calculations related to the structural aspects of the molecular systems. Species that will be considered for the calculations are obtained from Group A and Group B (if defined) with additional exclusions impose according to the EXCLUDE statements, if these are defined.

3.3.1 Moments of distributions

The distribution of values is usually expressed by means of graphical illustrations. In practice, one often wishes to compare the actual distribution to some theoretical distribution expressions. Comparing the distribution visually in graphical forms provide a crude way to this problem. It would be more useful if some measures can be obtained to represent further the characteristics of a distribution.

Moments of distributions provide a general way for characterising and comparing distributions. In principle, any degree of refinement can be achieved by extending the number of moments to include those of successively higher powers of the variable. The *p*th moment of some observable *X* is defined as

$$< X^p > = \int_0^\infty X^p P(|X|)d(|X|)$$

The first moment (*p* = 1) of *X* is the mean (expectation) itself, and the second moment (*p* = 2) is the variance, third moment is skewness and the fourth moment is the kurtosis (“tailedness”).

Higher moments will result in <*X*^{p}> increasing dramatically. For this reason, the reduced moments will be calculated:

$$\delta_X(p,2) = \frac{< X^p >}{< X^2 >^{p/2}}$$

DL_ANALYSER allows users the option to calculate higher moments (*p*=4, 6 and 8) for some of the observables.

3.3.2 Structural analysis options

Below lists the options available in sequence as shown in the *control* file.

```
--- Structural analysis
(1) 1           * Activate analysis (1=yes 0=no)
(2) test.out    * Output file
(3) 0           * Number of every configuration to skip
(4) 0           * Reduced moments of distributions (p = 4, 6 and 8)
(5) 0           * Block analysis (1= yes, 0=no)
(6) 0   0       * Molecular matching (1=yes, 0=no) and output option (1=yes and 0=no).
(7) none        * Template file for matching (if 'none', first config in input file will be used).
(8) 0           * Center of gravity of Group, or every molecule (1=yes, 0=no)
(9) 0           * Radius of gyration (1=yes, 0=no)
(10)0           * Asphericity (1=yes, 0=no)
(11)0           * System density (1=yes, 0=no)
(12)0  CTL2  CTL2  z * Chain segment orientation order parameter (1=yes, 0=no)…
(13)0  0.05  9.0     * Radial distribution function (RDF) (1=yes, 0 = no), bin width and …
(14)H954N O951       * Atom labels (or molecule labels) for RDF (case sensitive) …
(15)1  0.2  5.0      * Non-bonded Dihedral distribution atoms h i j k (define below), bin width, …
(16)C180 C182 C182 C180   * Atom labels h-i j-k for nonbonded DIHEDRAL distribution above.
(17)1  0.2  5.0      * Non-bonded angular distribution between h-i and j-k (define below), …
(18)C182 C180 C182 C180   * Atom labels i-h j-k for nonbonded ANGLE distribution above.
(19)0  z 0.2 60.0 all * Planar density profile (1=yes, 0=no), direction(x,y,or z), bin width, …
(20)0               * distance between Group A and Group B, or average distance between…
(21)0               * Locate maximum and minimum coordinates (1=yes, 0=no)
(22)0               * Identify closest distance pair (1=yes, 0=no)
…
…
```

Unless otherwise stated, all analysis only applies to atom-base analysis.

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

Once the analysis is finished, the Results Output file contain the following information:

(I) Information about the *Atom Range Definition* as defined in the *dl_analyser.control* file.

(II) Descriptions for each of the activated analysis option.

(III) The output format for each of the activated analysis option.

(iv) List of analysed results with the output formats in accordance to section (III).

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

(4) Reduced moments of distributions. See Section 3.3.1.

(5) Block analysis. This option calculates the averages of a successive blocks of values. In MD simulations, the successive values measured are highly correlated since there is only little change to the configuration from one timestep to the next. If averages are evaluated over blocks of successive values, then as the block size increases the block averages will become less correlated.

This means that, provided the simulation timescale is greater than the longest correlation time, the successive variance of the block averages will increase until a plateau is reached. The standard deviation over all the blocks of a given size *n* is defined as

$$SD = \sqrt{\frac{1}{(N-1)}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Where *N* is the total number of blocks, x_i, an average of a block and μ is the average over all blocks. The estimated standard error is therefore

$$SE_n = \frac{SD_n}{\sqrt{N}}$$

When *n* is small, then the consecutive blocks are highly correlated, and *SE* will be underestimated and gradually increases as *n* increases until a plateau is reached. When this happens, the block size at the onset of the plateau gives an indication of the extent of sample correlations.

DL_ANALYSER only carries out the averages on the radius of gyration of a given group of atoms and therefore to use the block analysis, Option **(9)** must also switch on.

(6) Molecular matching. This takes two values: *a* *b*

This option calculates the best molecular fit based on the supplied template (as defined in option **(7)**).

The *a* is the activation switch 1 (ON) or 0 (OFF) for this option. If this is switched on, then matching coefficient will be calculated.

The value *b* indicates the option to output the matched atomic coordinates, which also takes the value either 1 (ON) or 0 (OFF).

If *b* is 1, then the file *dl_analyser.matched_A* will be created and the matched coordinates from the Group A atoms will be written to the file. Similarly, the *dl_analyser.matched_B* file will be created if Group B atoms are also defined.

(7) Template file for molecular matching, option **(6)**. The file can be either in *PDB* or *xyz* format and the atom sequence must match with those in the user's input trajectory files.

If 'none' is specified, then the first configuration in the input trajectory file will be used as the template. All other subsequent trajectories will be matched against this template.

(8) Centre of gravity, *cg*. (Also applies to molecule-base analysis)

This is defined as

$$cg = \frac{1}{M}\sum_{i}^{n} m_i r_i$$

where *M* is the total mass in a Group and *m* and *r* are the mass and position vector of an atom *i*. The Program displays *cg* in terms of x, y and z components.

For atom-base analysis: The results are shown in the *x*, *y* and *z* components of the *cg* of Group A and Group B (if defined) atoms.

For molecule-base analysis: The results are shown in the *x*, *y*, and *z* components of every molecules (Molecule A1 and Molecule A2) in Group A atoms; and the *x*, *y*, and *z* components of every molecules (Molecule B1 and Molecule B2) in Group B atoms (if defined).

(9) Radius of gyration, *s*

This is defined as

$$s = \sqrt{\frac{\sum_i m_i c_i^2}{M}} \qquad c_i = r_i - cg$$

(10) Shape quantity (asphericity), *A*s

This measures the overall shape of a Group, in terms of the total square differences of the three eigenvalues, λ^2, of the radius of gyration tensors.

$$A_s = \frac{\sum_{i>j}^{3}\left(\lambda_i^2 - \lambda_j^2\right)^2}{2\left(\sum_{i=1}^{3}\lambda_i^2\right)^2}$$

If A_s = 1.0 means a perfect rod; and A_s = 0.0 means a perfect sphere.

(11) System density, ρ

This is defined, in unit *kgm*$^{-3}$, as

$$\rho = \frac{\sum_i m_i}{V}$$

Where *V* is the system volume, calculated from the cell vectors of the simulation box.

(12) Chain segment order parameter. This is defined as the second order Legendre polynomial as

$$< P_2 > = \left\langle \frac{3}{2}\cos^2\theta - \frac{1}{2} \right\rangle$$

This option takes three parameters: the two atom labels that made up a bond and a director, which can be either the *x*-, *y*- or *z*-axis. The θ is the angle between a bond segment and a director.

(13) Radial distribution function, *g(r)*. (Also applies to molecule-base analysis)

Describes local organisation around any given atom. The *g(r)* is proportional to the probability of finding an atom (or molecule), *i*, in the *dr* at a distance *r* from a given atom (or molecule), *j*.

$$g(r) = \frac{n(r)}{N_d 4\pi r_n^2 \Delta r} \qquad r_n = (n - \frac{1}{2})\Delta r$$

The quantity *N*$_d$ is the number density, which is number of atoms per unit volume of the simulation box. The option also requires two additional parameters: the bin width, Δr, and the cut off radius, beyond which no atom will be considered. Both quantities dictate the size of the data for g(r) graph plotting.

Bonded *i*-*j* pairs will always be ignored.

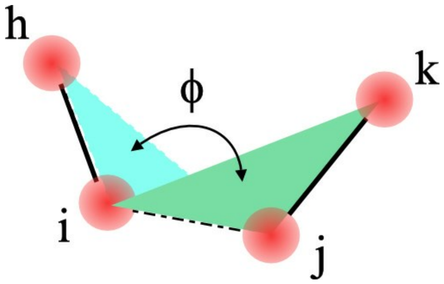
(14) Pair of atom labels to which *g(r)* will be determined. The Program will ignore atom pairs that are immediately bonded to each other but will still consider non-neighbour atom pairs (1-3 onwards) even if they belong to the same molecule.

For atom-base analysis, specify two atom pairs. The atom labels must be the same as shown in the user's input files.

For molecule-base analysis, specify the molecule labels. For instance, A1 A2. This instructs DL_ANALYSER to determine *g(r)* for molecule pairs Molecule A1 and Molecule A2.

(15) Non-bonded dihedral distribution between two bond vectors, *d*(ϕ).

This option measures the dihedral angle between two bond vectors, h-i and j-k define in the following line. Note that the criteria is that i and j are not bonded. The cutoff value is the distance between i and j atoms within which the dihedral angle (view along i-j vector)

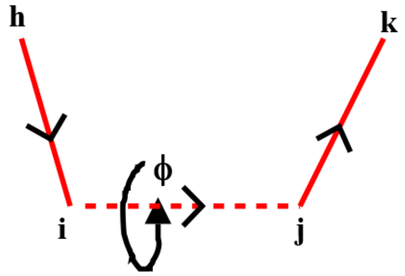


$$d(\phi) = \frac{n(\phi)}{N . r_n}$$

$$r_n = (n - \frac{1}{2})\Delta r$$

between the two bond vectors will be considered. In other words, the dihedral angle is a measure of twist between two bond vectors h-i and j-k. The dihedral angle takes the value clockwise, from 0 to 360 degrees.

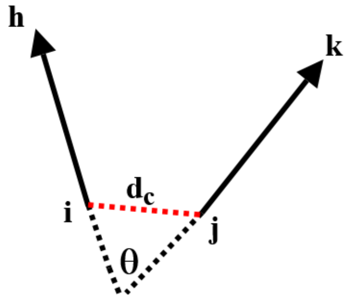
Where N is the total number of sample and *Δr*, the bin width. Diagram below shows the vectors for the dihedral calculations.



(16) Specify atom labels for non-bonded dihedral distributions. This option takes four values, or two bond vectors h-i and j-k. The labels must match with those in the input files.

(17) Angular distribution between two bond vectors, *d(θ)*

Measures the angular orientation between two bond vectors i-h and j-k, as shown below. The criteria are that the two bond vectors are not directly bonded to each other at atom i and atom j. Furthermore, the distance between these atoms (i and j) must be less than the specified cutoff value d$_c$. Otherwise, DL_ANALYSER will not carry out the calculation.



$$d(\phi) = \frac{n(\phi)}{N.r_n}$$

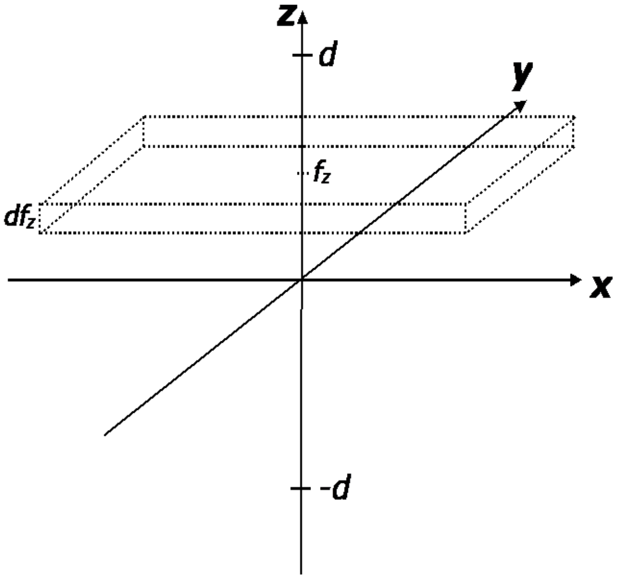
$$r_n = (n - \frac{1}{2})\Delta r$$

Where N is the total number of sample and *Δr*, the bin width. The angle takes the value between 0 (complete alignment of bond vectors) and 180 degrees (bond vector completely opposite to each other).

(18) Specify atom labels for non-bonded angular distributions, for Option **(17)**. This option takes four values, or two bond vectors i-h and j-k. The labels must match with those in the input files.

(19) Planar density profile, $\rho(f_c)$. (Also applies to molecule-based analysis).

$$\rho(f_c) = \frac{n_f}{Ndf_c}$$



The value N is the total atom and n_f is the number of atom or center of mass of molecule within an element df_c. The Program essentially sweep across along an axis specified from d to +d and produce a set of density profile plot along the axis. Atoms with the position vector component outside the region d ≤ f_c ≤ +d will be ignored.

(20) Distance between Group A and Group B. (Also applies to molecule-base analysis)

This measures the centre of mass distance between two atom groups Group A and Group B. To use this option for atom-base analysis, atoms for Group B also must be defined. For molecule-base analysis, definition for Group B atoms is an option.

For atom-base analysis: Distance between the centre of mass of Group A and Group B.

For molecule-base analysis: Average distances between molecules of the same and different types. For example, the average distance between Molecule A1 – A1, A1 – A2, etc.

(21) Maximum and minimum coordinate range.

This determines the maximum and minimum of the coordinate range in a Group and all three x, y and z component ranges will be displayed.

(22) Identify closest atom pair distance.

The Program identifies atom pairs that are closest to each other within a Group together with the distance. If Group is also defined, then closes inter-pair distance between Group A and Group B will also be identified.

3.4 Dynamical Analysis Section

This *Section* carries out analysis calculations related to the dynamical aspects of the molecular system. Atoms that will be considered for the calculations are obtained from

Group A and Group B (if defined) with additional exclusions impose according to the EXCLUDE statements, if it is defined.

Below list the options available in sequence as shown in *control* file.

```
--- Dynamical analysis
(1) 0               * Activate analysis (1=yes, 0=no)
(2) temp.out        * Output file
(3) 0               * Number of every configuration to skip
(4) bulk            * Surface or bulk?
(5) -1.425y 62.1y    * Surface definition (top and bottom threshold)
(6) 0                * kinetic energy
(7) 0  C             * Mean square displacement (MSD) (1=yes, 0=no). Atom label
(8) 1  C20           * Velocity auto-correlation (VACF) (1=yes, 0=no). Atom label.
(9) 0  100.0         * Specific heat at constant volume, temperature
(10)0                * Center of mass velocity (1=yes, 0 = no)
(11)0  967           * temperature (1=yes, 0=no) and constrain number, Nc.
(12)0  13.0          * temperature profile (1=yes, 0=no) and sphere concentric radius increment
(13)0  13.0          * velocity profile (1=yes, 0=no) and sphere concentric radius increment
(14)0.0 -1.425 0.0 * Center point of the concentric spheres (for temperature and velocity profiles).
(15)0                * Cross correlation displacement coefficient, C(ij) (1= yes, 2=0)…
(16)some_file         * File that lists atom index for cross correlation displacement coefficient
```

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

Once the analysis is finished, the Results Output file contain the following information:

(I) Information about the *Atom Range Definition* as defined in the *control* file.

(II) Descriptions for each of the activated analysis option.

(III) The output format for each of the activated analysis option.

(iv) List of analysed results with the output formats in accordance to section (III).

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

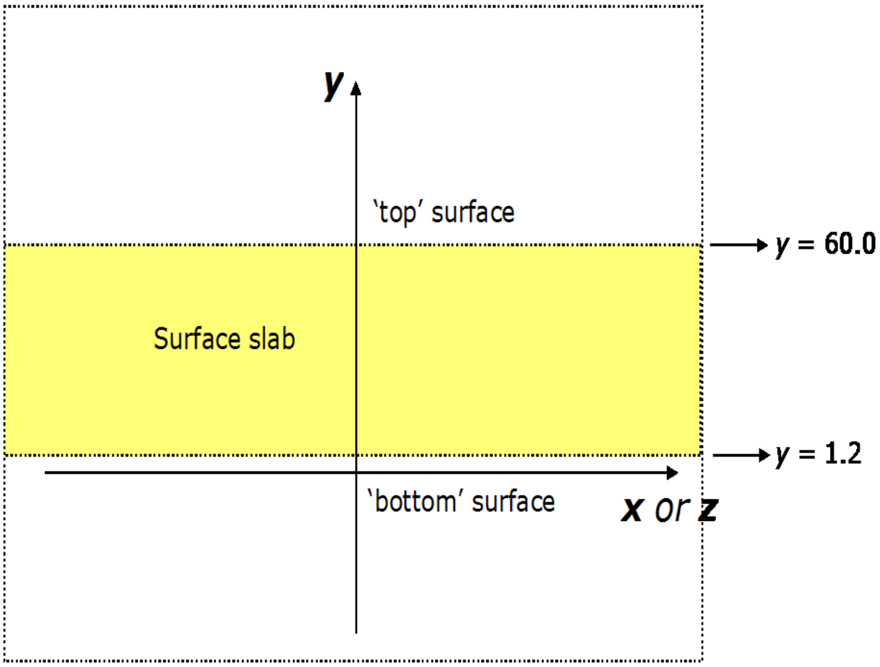
(4) Indicate whether the molecular system is a bulk or a surface. This information is only relevant to some analysis options shown below.

(5) Define surface normal. In 3D box, a periodic surface slab is defined with a periodic gap in between. This information is only relevant to some analysis option shown below. This will be ignored if the molecular system is defined as a bulk.

The surface normal must be orientated in one of the three x, y or z directions. For example, suppose the surface plane is orientated ***normal*** to y-direction, then the surface normal is defined as follows:

60.0y 1.2y

This means the surface plane is along x-z and the 'top' surface plane threshold is y = 60.0 and the 'bottom' surface plane threshold is y = 1.2. Diagram below shows a 2D representation of a surface slab.



(6) Kinetic energy, Ke

$$K_e = \frac{1}{2}\sum m_i v_i^2$$

The velocity, *v*, is obtained from the HISTORY file, provided such information is available.

(7) Mean-square displacement, msd. (Also applies to molecule-base analysis)

Measures the mobility of the atoms or molecules, by calculating the displacement of an atom over time t, averaged over all time intervals for all atoms with the label specified by the users.

$$<\Delta r(t)^2> = \frac{1}{N}\sum_{i=1}^{N}(r_i(t) - r_i(0))^2$$

The Program assumes the time interval between the successive trajectory frames is always equal. Note that for large t ~ the total MD time, the msd results may not be accurate since it is averaged over a few time-interval samples. This is especially true if the total number of the specified atoms is small in the molecular system. For molecule-base analysis, the msd refers to the distances between the center of masses of the molecules.

(8) Velocity auto-correlation function (VACF)

This provides a measure for expressing the extent to which the velocities (dynamical properties) are correlated over a sequence of values. The correlation function contains all the necessary information to define the relaxation of the system. The decay of the VACF indicates the decay in the correlations in atomic motion along the trajectories of the atoms.

The VACF is defined as follows

$$G(\tau) = \langle v(t_0) \cdot v(t_0 + \tau)\rangle_{i,t_0}$$

Which is averaged over all the starting points t_0 for all atoms *i* and normalised to 1 at τ = 0.

The vibrational spectrum for the system can be calculated through the Fourier transformation of the VACF that converts the information along the atomic trajectories from time to frequency frame of reference:

$$I(\nu) = \int_{-\infty}^{\infty} \exp(-2\pi i \nu\tau)\, G(\tau) d\tau$$

The output will be shown as the magnitude of the complex results, $\sqrt{Re^2 + Im^2}$.

(9) Specific heat at constant volume, C_v.

$$C_v = \frac{N}{T^2} < \Delta^2 >$$

But in the NVT ensemble, scaling temperature in MD results in $<\Delta^2>$ = 0. However, it is shown* that fluctuation of the energy component such as kinetic energy, E_k, is to be considered instead.

$$C_v = \frac{3}{2}\left[1 - \frac{2N < \Delta_k^2 >}{3T^2}\right]^{-1} \qquad < \Delta_k^2 > = < E_k^2 > - < E_k >^2$$

Where T is the temperature and N is the number of atoms.

(10) Centre of mass velocity, v_m (No periodic boundary condition apply)

The net velocity at the centre of mass of a group of atoms. This is equal to the sum of momentum divided by the total mass.

$$v_m = \frac{\sum m_i v_i}{\sum m_i}$$

The velocity information is obtained from the HISTORY trajectory file.

(11) Temperature, T

This option also requires the constrain number, N_c, which must be deducted from the total degrees of freedom, 3*N*, where *N* is the number of particles.

For non-periodic system, N_c = 0; periodic system, N_c = 3 if linear momentum is conserved, but not the angular momentum. If both linear and angular momentum are conserved, then N_c = 6. The actual value of N_c can be found in the OUTPUT file of the DL_POLY program.

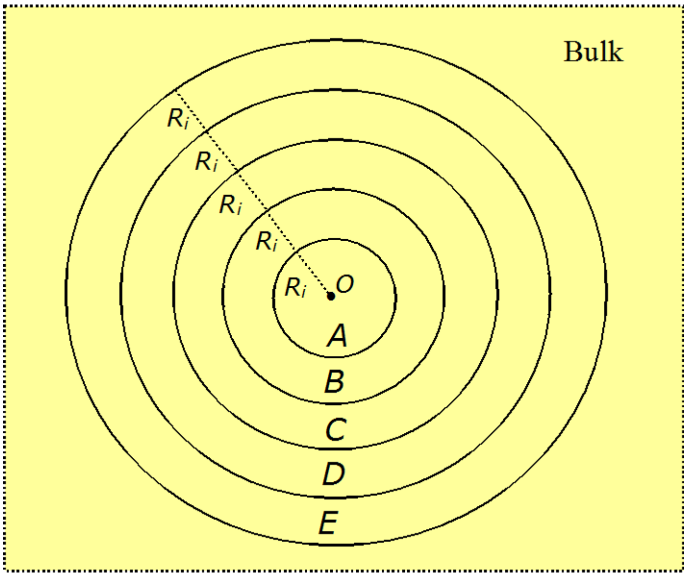
$$T = \frac{\sum m_i v_i^2}{(3N - N_c)k_B}$$

(12) Radial temperature profile. (No periodic boundary condition apply)

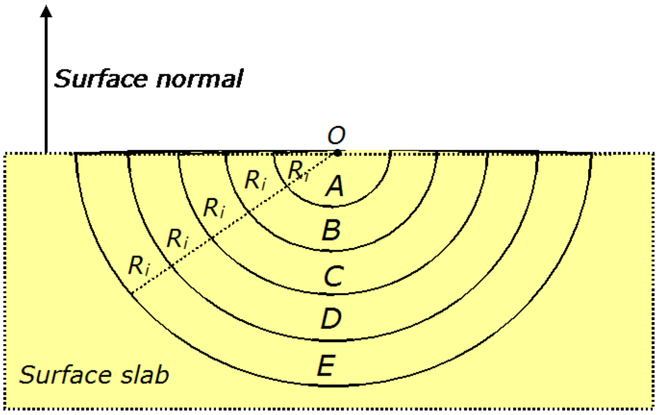
This measures the temperature independently over five circular regions of increasingly larger radius centred at a point specify by the user in Option **(14)**. To use this option, Option **(11)** mentioned above must also be activated.

This option is useful to track energy propagation originated from a certain point. For instance, in sputtering, particle bombardment, radiation damage, etc.

Two parameters are required for this option: the on/off switch and radius increment, R_i. The latter quantity determines the width and size of each circular region, within which the temperature is measured. The way how each circular region is mapped is shown below, depending on whether the system model is specified as a 'bulk' or a 'surface' in the Option **(4)**:



A 2D projection of a set of overlapping growing spheres with a common centre point, *O*, as defined in Option **(14)**. Temperature values are calculated for atoms that are in various Regions *A, B, C, D and E*.



A 2D projection of a set of overlapping growing hemispheres with a common centre point, *O*, as defined in Option **(14)**. Temperature values are calculated for atoms that are located in various Regions *A, B, C, D and E*.

If the 'surface' option is selected, then, only atoms that are located between the two surface planes (top and bottom surfaces) will be considered. In this case, usually, the centre point of Option **(13)** is located along one of the surfaces.

(13) Radial velocity profile. (No periodic boundary condition apply)

This is similar to option **(12)**, it measures the resultant velocities independently over 5 circular regions of increasingly larger radius centred at a point specify by the user in option **(14)**. To use this option, option **(10)** mentioned above must also be activated.

In this case, the calculated velocities are displayed in components for each region.

(14) Define centre point of the concentric spheres, *O*

For use with Option **(12)** and Option **(13)**. Describes in x, y, and z components.

(15) Cross correlation displacement coefficient, C_{ij}. (No periodic boundary condition apply)

The coefficient is a useful for measuring the influence of an atom with respect to another atom, by measuring the correlation motions of atom pairs. The correlation coefficient of an atom pair *i-j* is defined as

$$C_{ij} = \frac{< \Delta r_i \cdot \Delta r_j >}{\sqrt{< \Delta r_i^2 >< \Delta r_j^2 >}}$$

where Δr is the distance from the mean position of an atom. C_{ij} is normalised so that when C_{ij} = 1.0 it means the atom pairs are completely correlated, whereas, C_{ij} = 0.0 means the atoms are completely uncorrelated. A negative C_{ij} means motional influence of the atom pair is opposite to each other.

A filename that contains a list of atom index, of which the correlations are to be calculated, must also be supplied. This file is indicated in Option **(16)**.

The Program will go through two passes over the trajectory input files. The first pass is to determine the average positions of the atoms and then the second pass will determine the correlation displacements of the atoms. All possible combinations of the atom pairs and the corresponding C_{ij} values will be shown in the results output file.

(16) Atom index file for cross correlation displacement

The filename that contains a list of atom index number for the correlation calculations of the Option **(15)**. The only parameter that is needed is the atom index number on the first column and one atom for each line. Any other information after the index number will be ignored.

3.5 Defect Analysis Section

This *Section* carries out analysis calculations related to the structural aspects of the molecular system that relates to the arrangements and dislocations of the atoms with respect to the reference crystal lattice (template). Atoms that will be considered for the calculations are obtained from Group A and Group B (if defined) with additional exclusions impose according to the EXCLUDE statements, if it is defined.

Below list the options available in sequence as shown in *control* file.

```
--- Defect analysis
(1) 0                * Activate analysis (1=yes, 0=no)
(2) r.out            * Output file
(3) 10               * Number of every configuration to skip
(4) surface          * surface or bulk?
(5) -1.425y 62.1y    * surface definition (top and bottom).
(6) 1.425            * cutoff radius around an original site.
(7) 1                * Defect distribution profile scan.(1=yes, 0=no)
(8) y                * Profile direction. Scan along x, y or z direction.
(9) 0.2              * Bin width for defect distribution profiles.
(10)1                * Defect profiles according to (1)template, or (2)current sites
```

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

Once the analysis is finished, the Results Output file contain the following information:

(I) Information about the *Atom Range Definition* as defined in the *control* file.

(II) Descriptions for each activated analysis option.

(III) The output format for each activated analysis option.

(iv) List of analysed results with the output format in accordance to section (III).

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

(4) Indicate whether the molecular system is a bulk or a surface. This information is only relevant to some analysis options shown below.

(5) Define surface normal axis. In 3D periodic simulation box, a periodic surface slab is defined with a periodic gap in between. This information is only relevant to some analysis option shown below and will be ignored if the molecular system is defined as a bulk. Please see Option **(5)** of the *Dynamical Analysis Section* for more details.

(6) Cutoff radius of sites, r_{nn}

This value determines how the state of an atom within a molecular system is defined when assessing the damage state of the system. Typically, this would be the half of the nearest-neighbour between two atoms in a lattice site. See option **(7)** below for the definition of various atom states.

(7) Defect (atom site states) distribution profile scan

When this option is switched on, DL_ANALYSER will scan along the axis specified in option **(8)** and determine the type of states for each atom site contained within a layer of thickness as defines in option **(9)**. A set of data will be printed in the output file and user can plot the data to show the profile scan for each type of atom sites.

The various types of atom sites as determined by the Program are shown below:

(a) Original occupancy site, $N_{original}$: A lattice site that is occupied by the original atom. In other words, there is only one, **same** atom contains within a sphere of radius r_{nn} centred at the lattice site.

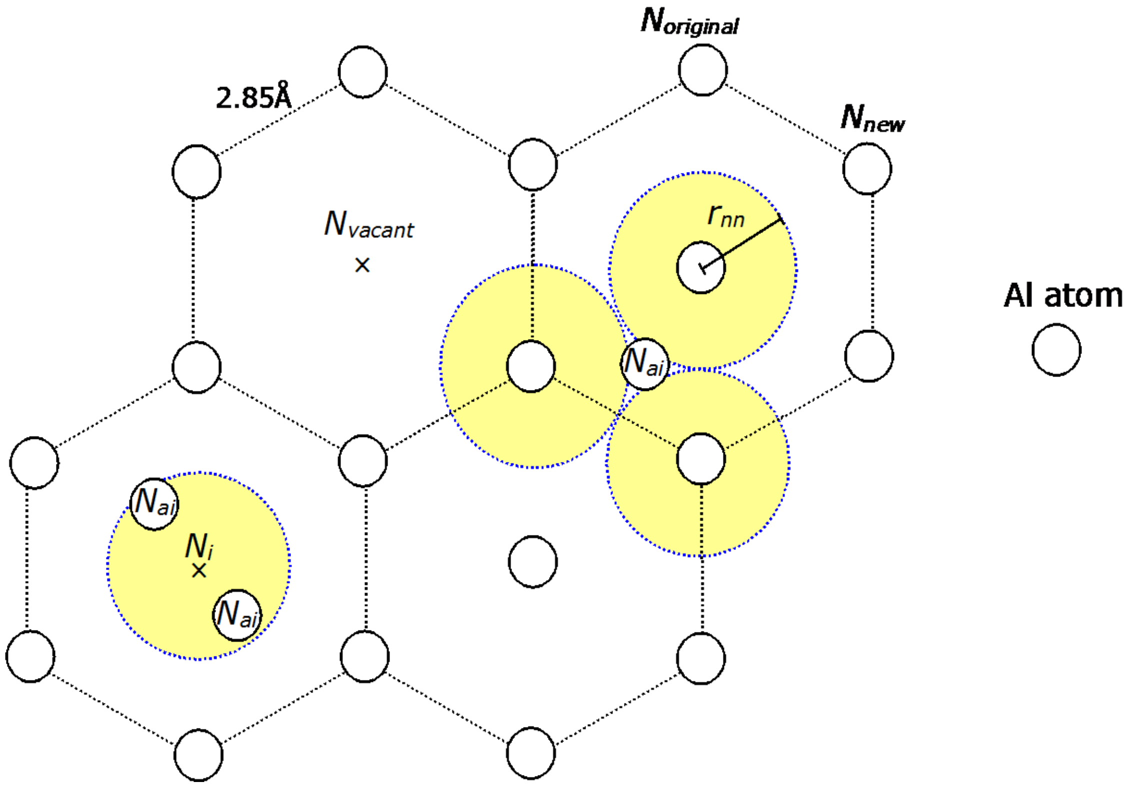
(b) Vacant site, N_{vacant}: A spherical region, of radius r_{nn} centred at a lattice site, that contains no atoms.

(c) New occupancy site, N_{new}: A vacant site that is re-occupied by another atom of the same type.

(d) Interstitial site, N_i: If a spherical region of radius r_{nn} is centred at a site contains more than one atoms, then the site is considered as an interstitial site. The atoms are considered as the interstitial atoms, N_{ai}. In addition, all other atoms which are found on locations outside all the sphere sites within the system bulk will also be considered as interstitials. For this reason, N_{ai} is always greater than N_i.

(e) N_{surf}: If an atom is located at a normal distance of more than r_{nn} away from the surface (as defined from option **(4)** and option **(5)**), it is considered as an 'above-surface' atom, N_{surf}. To detect whether the atom is completely left the surface or reabsorbed on the surface as an adatom can be detected from the *Sputter Analysis Section.*

Take for example, consider the Al(111) surface which has a hexagonal close packed configuration with the nearest neighbour distance of 2.85 Å. Then by defining r_{nn} = 2.85/2.0 = 1.425Å, diagram below shows, by viewing at the Al(111) plane, the various types of atom sites:



Once DL_ANALYSER has defined all atom states, at every number of time step (depending on option (3)), the **total number** of various atom states will be written, in a sequence of columns. This is followed by the profile scan of various sites across the system, according to option (8). Below shows the display format of the results in columns:

Once DL_ANALYSER has defined all atom states, at every number of time step (depending on option **(3)**), the **total number** of various atom states will be written, in a sequence of columns. This is followed by the profile scan of various sites across the system, according to option **(8)**. Below shows the display format of the results in columns:

Defect Group A: MD_time $N_{original}$ N_{vacant} N_i N_{ai} N_{new} N_{surf} Frame_no
Defect profile Group A: Distance1 $N_{original}$ N_{vacant} N_i N_{ai} N_{new} N_{surf}
Defect profile Group A: Distance2 $N_{original}$ N_{vacant} N_i N_{ai} N_{new} N_{surf}
Defect profile Group A: Distance3 $N_{original}$ N_{vacant} N_i N_{ai} N_{new} N_{surf}
…
…

In other words, the first row indicates the total number of various site states, whereas, the following list of data shows the locations of various site states along the axis as defined in option **(8)** below.

(8) Profile sampling direction. Can only be one of the x, y or z axis. For instance, if the surface normal is along y direction, then specifying profile sampling direction along the y direction will cause the Program to perform the scan from the top to the bottom of the surface.

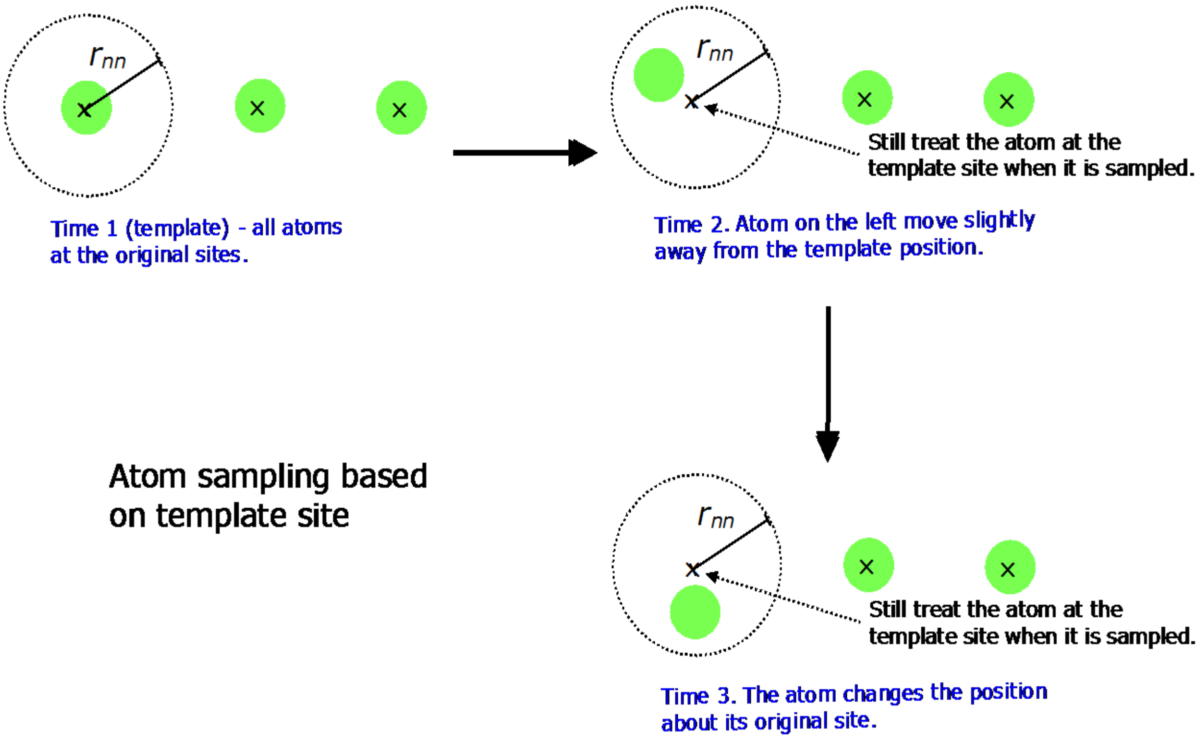
(9) Profile sampling width. Define the sampling width along the axis as the Program scan across the molecular system.

(10) The way how sites are sampled along the sampling direction. Basically, there are two ways to determine the profile: (a) the template site base, or (b) the current atom site base.

These are explained in more details as follows:

(a) Template site base (choose 1 for this option). This means that the first frame in the trajectory file will be used as a template for all other subsequent trajectory frames. Recall that an atom is still considered to locate at its original site provided the distance is not greater than r_{nn} (option **(6)**). The corresponding data profile is then collected base on the location of the template site, ***not*** the location of the atom at the current trajectory frame.

Example below illustrates how the data profile is collected. From time 1 to time 3, the atom (in green) is moved away from the template site but is still located near to the site. It is still regarded as an original atom occupying the same site as that of the template. Then, as the distance is scanned along the axis, the atom is sampled base on the location of the original site, rather than the current atom site.

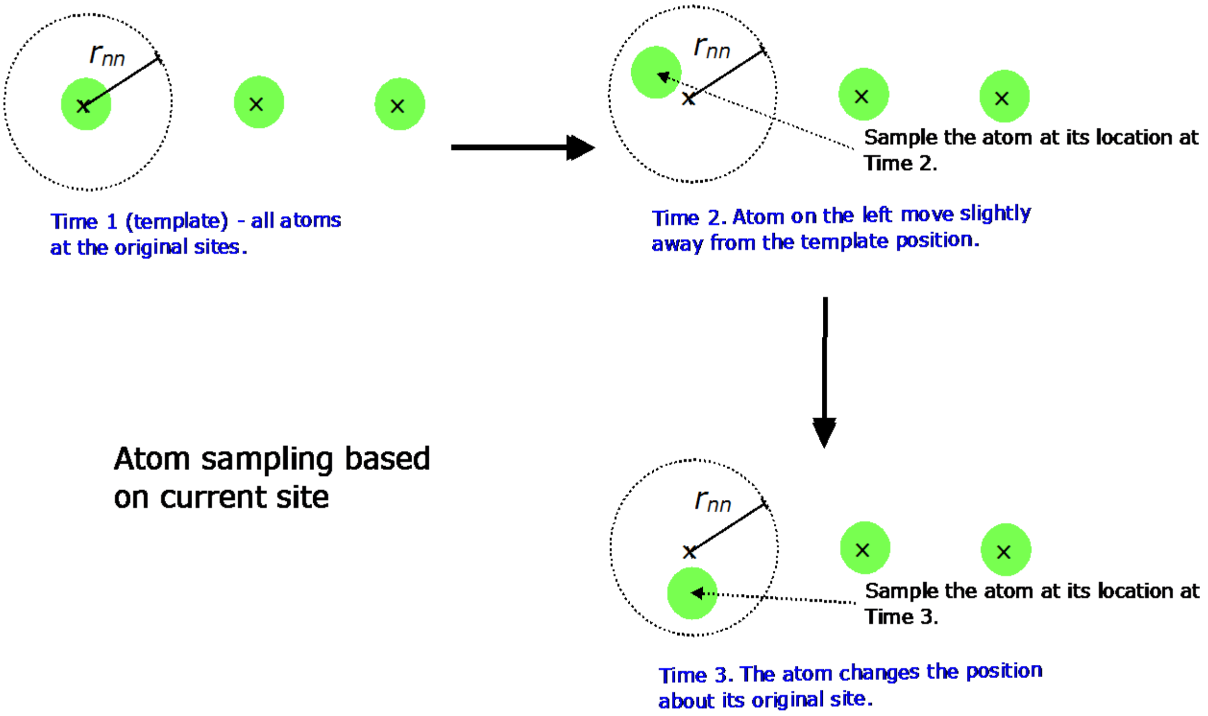

r_{nn}
Time 1 (template) - all atoms at the original sites.
r_{nn}
Still treat the atom at the template site when it is sampled.
Time 2. Atom on the left move slightly away from the template position.
Atom sampling based on template site
r_{nn}
Still treat the atom at the template site when it is sampled.
Time 3. The atom changes the position about its original site.

This type of sampling style also applies to the vacant sites (N_{vacant}), interstitial sites (N_i), and the newly occupied sites (N_{new}). The **only exception** is the location of the interstitial atoms, N_{ai}, which will be sampled according to the location of the atoms.

(b) Current atom site base (choose 2 for this option). The first frame will still be used as the template for all other subsequent trajectory frames, to determine the states of the atoms. Recall that an atom is still considered to locate at its original site provided the distance is not greater than r_{nn} (option **(6)**). The corresponding data profile is then collected base on the current location of the atom, ***not*** the original location of the atom in the template frame.

Example below illustrates how the data profile is collected. In time 1 to time 3, the atom (in green) is moved away from the template site but is still located near to the site. It is still regarded as an original atom occupying the same site as that of the template. Then, as the distance is scanned along the axis, the atom is sampled base on the location of the atom at the current time.

This type of sampling style also applies to the interstitial atoms (N_{ai}), and the newly occupied sites (N_{new}). The **exceptions** are the locations of the vacant sites (N_{vacant}) and the interstitial sites (N_i), which will be sampled according to the template.


r_{nn}
Time 1 (template) - all atoms at the original sites.
r_{nn}
Sample the atom at its location at Time 2.
Time 2. Atom on the left move slightly away from the template position.
Atom sampling based on current site
r_{nn}
Sample the atom at its location at Time 3.
Time 3. The atom changes the position about its original site.

3.6 Sputter Analysis Section

Note: This *Section* is not necessarily applied only to sputtering, or radiation damage simulations. The Analysis can also apply to other situations such as surface desorption process and crystal dissolution process. In this case, the 'sputtered' atoms would refer loosely to those that crossed the threshold from the surface into bulk (or vacuum).

This *Section* works for both atom-base and molecule-base analysis. In the latter case, molecules in the system will be identified and then center of masses of each molecule will be determined, which represent the location of each molecule and assumes there is no breakage of bonds. In other words, DL_ANALYSER cannot determine fragmented molecules during sputtering. For the sake of illustrations, whether it is atom-base or molecule base analysis, the word 'atom' will be used below to indicate the particles in the system.

This *Section* carries out analysis calculations relate to the atoms that are above the surface. Atoms that will be considered for the calculations are obtained from Group A and Group B (if defined) with additional exclusions impose according to the EXCLUDE statements, if it is defined. This Analysis only works for the surface model and the axis that is normal to the surface plane must be defined.

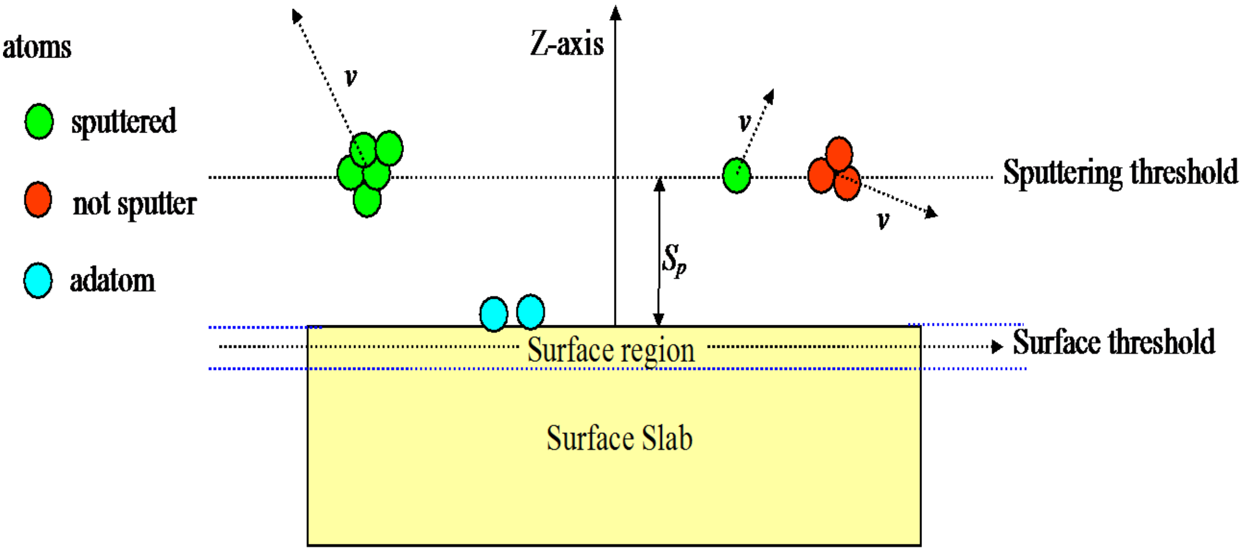
In general, when the atoms are located above the surface, the Program will classify them as one of the two types: either the atom is an adatom or it is a sputtered atom. Note that, the Program distinguishes atoms at the 'top' surface from the 'bottom' surface.

Adatoms - Atoms or groups of atoms that locate **beyond** the surface threshold that are adsorbed on the surface as determined by the nearest neighbour cut off distance measure from the surface threshold.

Sputtered - Atoms or groups of atoms of which the centre of mass passes the sputtering threshold and, if velocity data is available, with a resultant velocity that is directed away from the surface.

Interface – Atoms locate between the surface threshold and sputtering threshold.

Diagram below illustrates the sputter analysis model, assuming the z-axis is normal to the surface plane. The S_p also defines the extent of the interfacial region.



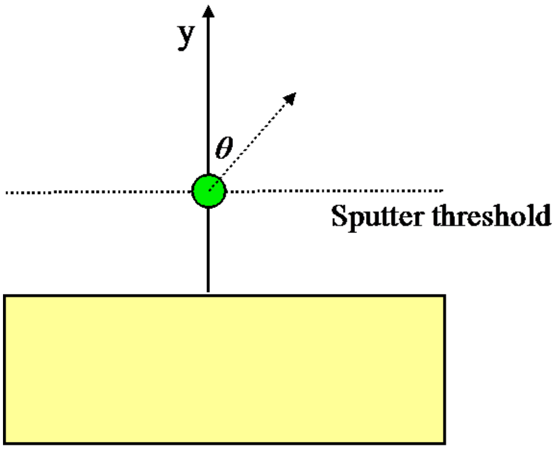
Below lists the options available in sequence as shown in the *control* file.

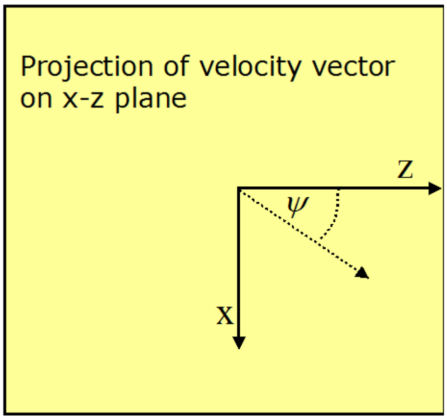
```
--- Sputter analysis
(1) 0                 * Sputter analysis (1=yes, 0=no)
(2) s.out             * Output file
(3) 0                 * Number of every configuration to skip
(4) 0                 * Detailed output (1=yes, 0=no)
(5) -47.0z 1.5z       * surface definition (top and bottom threshold).
(6) 5.0               * surface region (thickness, centred around surface threshold)
(7) 8.0               * sputtering threshold.
(8) 4.50              * Nearest neighbour cutoff distance.
```

(1) Master switch to activate (1) or deactivate the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

When an atom or group of atoms are sputtered, the directions of the sputtering angles will also be reported. These are the sputtering angle, θ, with respect to the surface normal and the spreading angle, ψ. These angles are illustrated in the diagram as shown below. Consider a surface slab with the surface normal aligns along the y direction:





Note that the right-hand notation is used for the axis orientation. If x or y is the surface normal, then the spreading angle will be measured from the z-axis, anti-clockwise. If z is the surface normal, then the y axis would be the reference axis for the spreading angle.

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

(4) Detail output option. This gives user option to produce detail output for sputtered atom, for instance, the atom index of every sputtered atom.

(5) Define surface threshold along the surface normal axis. Please see Option **(5)** of the *Dynamical Analysis Section* for more details.

(6) Define surface region (thickness). This value defines the region spans within which surface atoms are located. The region is extended along the surface normal (above and below) centred around the surface threshold (see diagram above).

(7) Sputtering threshold, S_p. The distance along the surface normal axis, measures from just outside the surface region and extended outward. The threshold is the critical distance above which an atom or a group of atoms is considered sputtered and completely leave the surface provided the force is directed away from the surface.

Once the atom or group of atoms are sputtered will always consider sputter. The sputtering threshold therefore must be chosen with care that it is not too close to the surface.

The region between sputtering threshold and surface region (extended by S_p) is considered as the interfacial region and atoms that are located within this region is call interfacial atoms.

(8) Nearest neighbour cut off distance. This is the distance along the surface normal axis, measures from above the surface threshold, within which an atom or group of atoms are considered as adatoms. The same distance is also used, measured from below the surface threshold, within which an atom or group of atoms are considered as surface atoms.

3.7 Biological Analysis Section

This *Section* carries out analysis calculations specific to biological molecules. This Section requires the trajectories file in PDB format (See Chapter 4). Atoms that will be considered for the calculations are obtained from Group A and Group B (if defined) with additional exclusions impose according to the EXCLUDE statements, if it is defined.

Below list the options available in sequence as shown in *control* file.

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

Once the analysis is finished, the Results Output file contain the following information:

(I) Information about the *Atom Range Definition* as defined in the *control* file.

(II) Descriptions for each activated analysis option.

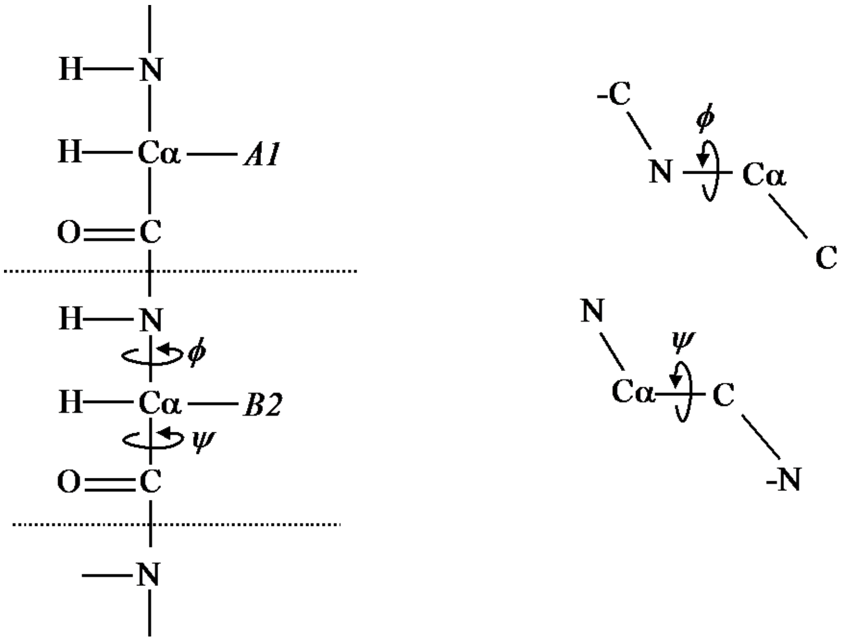
(III) The output format for each activated analysis option.

(iv) List of analysed results with the output format in accordance to section (III).

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

(4) Protein backbone dihedral angles, ϕ (phi) and ψ (psi)

These are the dihedral angles involving amide atoms and α-carbon (Cα) that made up the protein backbone. Consider two consecutive amino acid residues of side groups *A1* and *B2* that formed part of a protein molecule as shown below.



In other words, ϕ is the dihedral angle about the N-Cα bond with respect to two neighbouring bonds involving carbonyl carbon backbone from the current residue (C) and the previous residue (-C). Whereas, ψ is the dihedral angle about the Cα–C bond with respect to two neighbouring bonds involving amide nitrogen backbone from the current residue (N) and the previous residue (-N).

The results can be plotted with respect to each angle as a Ramachandra Plot, which describes the overall backbone structure of the proteins.

3.8 Interaction Analysis Section

This *Section* carries out analysis calculations on the atomic interaction in the system. Atoms that will be considered for the calculations are obtained from Group A (or B), with additional exclusions impose according to the EXCLUDE statements, if this is defined.

Note: The following restriction applies to the Analysis Section:

Atom labels must be expressed in DL_F Notation*. This is the universal atom typing expression that is independent of type of force field schemes used in molecular simulations. The DL_F Notation can be produced during force field model setup by using DL_FIELD.

*C. W. Yong, 'Descriptions and Implementations of DL_F Notation: A Natural Chemical Expression System of Atom Types for Molecular Simulations' *J. Chem. Inf. Model.* (2016) **56**, 1405-1409

When scanning through the atoms in the system, DL_ANALYSER will automatically identify various modes of interaction at atomistic scales. These different modes of interactions are expressed in the DANAI notation, which is a universal notation to annotate the atomistic interactions without resolving to the use of pictorial illustrations (See Section 5).

Below list the options available in sequence as shown in the *control* file.

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

Once the analysis is finished, the Results Output file contain the following information:

(I) Information about the *Atom Range Definition* as defined in the *control* file.

(II) Descriptions for each of the activated analysis option.

(III) The output format for each of the activated analysis option.

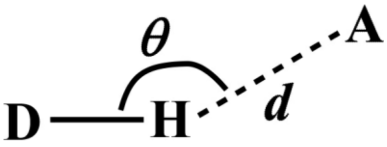
(iv) List of analysed results with the output formats in accordance to section (III).

(3) Frequency to carry out the analysis in this *Section*. If the number is set to zero, it means every configuration from the input trajectory file will be analysed. Otherwise, any number will refer to the number of configurations that will be skipped before carrying out the analysis.

(4) Switch to detect atomistic interactions *within* a Group (intra-interactions). This option instructs DL_ANALYSER to carry out atomic interaction analysis within Group A atoms and Group B atoms.

(5) Switch to detect atomistic interactions *between* Group A and Group B atoms. This option only works if Group B atoms are defined.

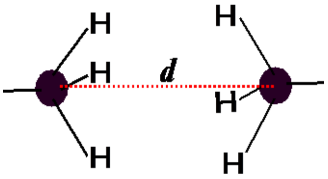
(6) Criteria to identify a hydrogen bond (HB) interaction. This takes two values, distance, *d*, and angle, θ. The distance is between a hydrogen atom and an acceptor atom (**A**), to which the HB forms. The *angle* is defined with the hydrogen being the apex atom. Diagram below illustrates a hydrogen bond definition.



Here, **D** is a donor atom. For example, for carboxylic and alcohol groups, both **D** and **A** are oxygen atoms. DL_ANALYSER will only identify a hydrogen bond if it satisfies both criteria: if the H- - -A distance is smaller than or equal to *d* and the angle D-H-A is larger or equal to θ.

(7) Criterion to identify a general dipole-dipole (DD) interaction, excluding the hydrogen-bond interactions (see above). This is simply the distance between two atoms, usually one with negative partial charge and the other a positive partial charge.

(8) Criterion to identify a hydrophobic contact. This applies to specific van-der-Waals (vdw) types of interactions between non-bonded alkyl groups. The criterion is simply the distance between two alkyl carbon atoms. Below shows a sketch of two methyl groups in hydrophobic contact with each other.



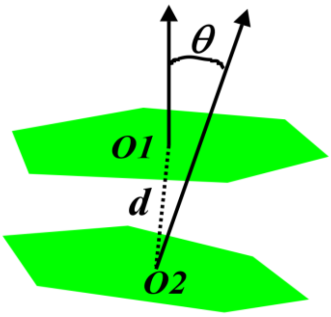
The hydrophobic contact distance, *d*, is shown in red dotted line, which is the distance between two alkyl carbon atoms. Although hydrophobic contact distance can be large compare with a typical bond length, usually the value of *d* (~4.5 Å) is selected to ensure no other atoms can be located along the distance between two alkyl carbon atoms.

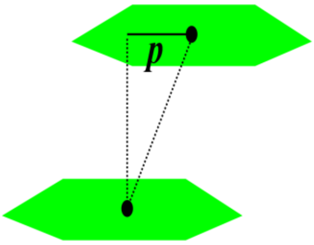
Note that bonded (1-2) and (1-3) non-bonded alkyl carbon atoms will not be considered. Only the 1-4 (alkyl carbon atom pairs separated by two connecting atoms) onwards and intermolecular interactions will be considered.

(9) Criterion to identify general dispersive induced-dipole (ID) interactions excluding the alkyl groups (see above). This is simply the distance between two atoms, usually of small partial charges. For example, ID interactions between two haloalkyl groups are identified based on the distance between two haloalkyl carbon centers.

Note that bonded (1-2) and (1-3) non-bonded atoms will not be considered. Only the 1-4 (atom pairs separated by two connecting atoms) onwards and intermolecular interactions will be considered.

(10) Criterion to identify π-π stacking in benzene-containing systems. Requires two parameters: the distance *d* between two benzene rings and the acute angle *θ* between the planes of two benzene rings, as highlighted in green (diagram on the left below):





The value *d* is the distance between the center of the ring planes, *O1* and *O2*. It is so chosen to represent the closest distance between two benzene rings, typically around 4.5 Å. A much larger value may run into the risk of wrongly identifying π-π stacking between two benzene rings sandwiched with some other atomic species.

The value θ = 0 degree means the ring planes are perfectly parallel, whereas, θ = 90 degrees means the ring planes are orthogonal to each other. The Program will identify two stacking rings if the distance is smaller than *d* and the angle is smaller than θ.

In addition, when both *d* and θ criteria are fulfilled, the Program will proceed to decide the stacking configuration type, by measuring the lateral displacement value *p* (see diagram on the right above). If *p* is less than or equal to 1.4 Å, then it is a sandwich stacking configuration (PS). If the value is 1.4 Å < *p* ≤ 2.8 Å, then it is a parallel-displaced (PD) configuration. If p > 2.8 Å, then the Program will assume no significant π-π interaction between the benzene rings and the whole configuration will be disregarded.

(11) Cross-correlation calculations between two different types of interactions. This only works if two interactions are selected (see below). This is simply defined as follows:

$$C_{x-y} = \frac{<\Delta C_x \Delta C_y>}{\sqrt{<\Delta C_x^2><\Delta C_y^2>}}$$

$$\Delta C_i = C_i - \mu_i$$

Where *x* and y are the two different interaction types, *C* is the number of counts for a certain mode of interaction and μ is the average count of that interaction. This Option enable the Program to calculation the correlation coefficient (relationship) of one mode of interaction with respect to the other modes of interaction.

For example, by switching to on (1) to Option **(12)** and Option **(19)** and the molecular system contains both the alkyl groups and the carboxylic groups, then cross-correlation calculation will be carried out between the hydrophobic contacts between the alkyl groups and the HB interactions between the carboxylic group.

Note: Correlation calculations will always carry out among the various modes of an interaction type.

(12) Option to detect interaction among the alkyl groups (HP_1_1). The Program will detect various modes of hydrophobic interactions (HP) according to the criterion set in the Option **(7)** and express these in the DANAI notation.

Note that the HP interactions refer to non-bonded alkyl groups from different molecules. It the alkyl groups belong to the same molecule (as in an aliphatic chain), then interactions between 1-2 (bonded) and 1-3 (angular) will be omitted. The 1-4 is considered as a non-bonded interaction and therefore will be considered.

(13) Option to detect interactions among the haloalkyl groups, including the trihaloalkyl ($-CX_3$), dihaloalkyl ($-CX_2$) and monohaloalkyl (-CX) groups. It has a general macro-interaction ID_18X_18Y, where X and Y can be 0 to 3.

(14) Option to detect interactions between benzene rings, specifically those of π-π stacking interactions (PS_6_6 and PD_6_6). Once a stacking is identified, according to the criteria set in Option **(8)**, the Program will decide the stacking types: either it is a parallel sandwich (PS) stacking, or the parallel displaced (PD) stacking configuration.

(15) Option to detect interactions among the alcohol groups (-OH) (HB_15_15). Since the only interaction type among the alcohol group is that of HB, the Program will use the Option **(6)** as criteria to detect and classify various modes of HB interactions and express these interactions in the DANAI notation.

(16) Detect HB interactions between the alcohol and carboxylic groups (HB_15_20). The Program will use Option **(6)** as criteria to detect and classify various modes of such interactions.

(17) Detection HB interactions between alcohol and aniline groups (HB_15_46), using Option **(6)** as the detection criteria. Note that, for this option, DL_ANALYSER will ignore non-aromatic amino groups such as amines.

(18) Detect HB and DD interactions between ammonium and carboxylate groups, HB_21_47 and DD_21_47. Note that DL_ANALYSER can detect various types of carboxylate and ammonium groups, including the amino acid zwitterions, such as the COO-terminus (950) and NH3-terminus (954). In any case, these groups will be annotated as 21 for carboxylate and 47 for ammonium, respectively.

(19) Option to detect interactions among the carboxylic groups (-COOH) (HB_20_20). The Program will use Option **(6)** as criteria to detect and classify various modes of HB interactions according to the DANAI notation.

(20) Option to detect HB interactions between carboxylic and aniline groups (HB_20_46), using Option **(6)** as the criteria to detect and classify various modes of such interactions.

(21) Option to detect HB interactions among the water molecules (HB_800_800), using Option **(6)** as the criteria to detect and classify various modes of such interactions.

(22) Option to detect HB interactions between the water molecules and ester group (HB_19_800), using Option **(6)** as the criteria to detect and classify various modes of such interactions.

(23) Option to detect water-carboxylate interactions, HB_X_800 and DD_X_800, where X can be either a general carboxylate (21), or carboxylate as in COO-terminus (950) for amino acids and proteins.

(24) Option to detect water-ammonium interactions, HB_X_800 and DD_X_800, where X can be either a general ammonium (47), or ammonium as in NH3-terminus (954) for amino acids and proteins.

(25) Option to detect HB interactions between the water molecules and phosphate group (HB_151_800), using Option **(6)** as the criteria to detect and classify various modes of such interactions.

3.9 STATIS Analysis Section

This *Section* basically extracts data from the STATIS output file generated by DL_POLY. The STATIS file or a filename that contains the word 'STATIS' must be included in the *dl_analyser.input* file.

(1) Master switch to activate (1) or deactivate (0) the analysis *Section*. All analysis in this *Section* will be ignore if it is deactivated even if some of the options are activated.

(2) Results Output filename, specify by the user. Make sure this filename is unique will not be shared by other *Sections*.

(3) - (27) Data from STATIS. See the *control* file for details.

4 File Conversions

The DL_ANALYSER program can read several file formats as described in Chapter 2. The most common form of the input file is the HISTORY trajectory file produced by the DL_POLY program. However, the *Trajectory Production Section* can be used to extract all or some of the trajectories in the HISTORY file and convert them into the PDB or the xyz format. These translated files can be either used for further analysis or used for animation purposes using other third party packages.

4.1 CONFIG and REVCON Files

These are single-configuration files for the DL_POLY program. The CONFIG file is the input file for DL_POLY program and contains the user's initial system configuration. The REVCON file is the file produced during the course and at the end of MD simulations. It contains the user's system configuration.

They can be translated into the *PDB* or the *xyz* format, but not the other way round. However, DL_FIELD program can be used to convert a *PDB* structure into the corresponding CONFIG file, together with the FIELD file (the force field model file).

4.2 The xyz Format File

This file format is the least informative that only contains the atom labels and the corresponding x, y and z coordinates. It takes the least disk space. The format is as follows:

```
No_of_atoms          time= 3.403 ps          step = 0.002
[Space, or some title text]
C    -168.977917  0.083913  -172.855802
Al_  -166.101003  0.011045  -172.847280
Al_  -163.233789  0.058050  -172.845844
Al_  -160.365398  0.091437  -172.887635
Al_  -157.509796  0.036537  -172.905057
...
...
No_of_atoms          time= 3.503 ps          step = 0.002
[Space, or some title text]
C    -168.947712  0.073760  -172.905155
Al_  -166.111531  0.142515  -172.867775
Al_  -163.193092  0.076674  -172.869657
Al_  -160.299232  0.134099  -172.840889
Al_  -157.412480  0.106132  -172.815243
...
...
```

The first line of every trajectory frame begins with the status line: The *no_of_atoms* is the total number of atoms contains in a trajectory frame. The *time* is the MD time in ps and the *step* is the MD timestep.

This is followed by a blank line.

After this is a list of atom labels with the corresponding *x*, *y* and *z* coordinates. The number of rows of atom labels must match with the *no_of_atoms.*

The *xyz* file can contain a number of trajectory frames which can be used to render the animation in some visualisation package such as the VMD.

The *xyz* file can also be read as an input trajectory file in the.

4.3 The PDB Format

DL_ANALYSER adheres to strict PDB standard format and different information must be contained within the appropriate columns in a PDB file. However, not all information will be required by DL_ANALYSER.

Each atom definition that precedes with 'ATOM' and/or 'HETATM' will be read by DL_ANALYSER (case-insensitive). Table below lists data that will be collected for each atom.

Note that the Program will process a PDB file until the END statement or the TERMINATE statement is encountered. Any atoms after these statements will be ignored.

Column range (inclusive)	**Data**	**Remark**
13-16	*Atom label*	If *element symbol* is not defined, then this data will be treated as the element symbol for the atom. Any numerical characters will be ignored.
18-20	*Residue names*	Residue names such as those of amino acids.
21	*Reserves for residue names*	Some residue names such as those of carbohydrates may contain more than three characters. The fourth one will be located here.
23-26	*Residue sequence*	Numerical sequence for molecules such as amino acids or carbohydrates.
29-56	*X,y,z coordinates*	Location of the atom in x,y,z coordinates.
77-79	*Element symbol*	The element symbol of the atom. If this is not defined, then *Atom label* will be used to determine the element symbol of the atom.

Note that, unlike the HISTORY file, the PDB file does not contain the atomic mass information. The atomic mass will be determined from the element symbol. If this information is not available, then the Program will try to determine the mass from the atom labels. If this fails, the Program will stop and report the error. In this case, select the Option **(4)** to assign unit mass for all atoms from the *Atom Range Definition*.

5 DANAI Notation

DANAI* is a standard notation to describe non-bonded atomistic interactions between two functional groups. Up till now most of the atomistic interactions are described by some arbitrary diagrammatic illustrations, frequently on an ad-hoc basis. While such approach conveys clear pictorial information to the reader, they are difficult to annotate and not directly retrievable by means of data query. The use of DANAI notation enable the Program to identify and quantify the various modes of atomistic interactions for a given type of non-bonded interaction. From such, statistical analysis can be carried out.

* C.W. Yong and I.T. Todorov, *Molecules* (2018), Vol. *23*, p36

DANAI – A name in Shona language given to a Zimbabwean girl, means *love each other*, aptly referring to the embrace of molecules as a result of their functional group interactions!

Following the flavours of the standard DL_F Notation (C. W. Yong, *J. Chem. Inf. Model.* **2016**, *56*, 1405-1409) for atom typing in molecular simulations, the DANAI expression provides a universal scheme that can be easily interpreted by modeller, experimentalist as well as computational means. It contains the actual chemical information and precisely annotate a given atomic interaction configuration that can be accessed by means of data analytics.

In the DANAI notation, the full description of any given interactions must always be expressed in terms of the *macro-interactions* and the corresponding *micro-interactions*.

5.1 Macro-interactions

Molecular non-bonded interactions between two functional groups in a general sense.

General format: *A_CGI1_CGI2*

Where *A* is the interaction types and *CGI* is the *Chemical Group Index* which is the unique numerical value for a given *Chemical Group* (CG) in the DL_F Notation. The latest version 2.3 can detect the following interactions:

DD – dipole-dipole interactions.
HB – hydrogen bonding. This is a special case of the *DD* interaction.
ID – induced dipole interactions.
HP – hydrophobic interactions, a special case of *ID* interactions between two alkyl groups.
PS – parallel π-π stacking interactions between benzene rings.
PD – parallel displaced π-π stacking between benzene rings.

For more information about the criteria used to decide the interactions, please refer to *Interaction Analysis Section*.

5.2 Micro-interactions

A set of particular interaction configuration mode between two Chemical Groups as defined in the macro-interactions.

General format: [***S****a*]*atomic_interaction*

Where ***S*** is the general description of the topological structure of the micro-interaction and, *a*, the number of CGs involve in an interaction that form such structure. Below lists some examples of ***S***.

J – A junction or a network intersection.
R – ring
L – Linear
C - Complex structure contains some or all of the above mentioned topological structures.

For instance, ***L****3* means a micro-interaction that involves *three* Chemical Groups that orientated in some linear structure. **R***2* means a micro-interaction that involves *two* Chemical Groups in a ring enclosure.

The *atomic_interaction* is a line of text that annotates the atomic species that involve in the interaction. The text consists of atomic species expressed in the DL_F Notation and can have the following symbols:

- **:** Represent non-bonded interactions, of which the type is described according to the macro-interactions. Every *atomic_interaction* expression must always include at least one non-bonded interaction.
- – A chemical bond between two atoms within a functional group.
- # The rest of the part of the same functional group that may or may not participate in any non-bonded interaction. This is usually used when two interactions occur at different parts of a functional group, with the inactive part represents as an '#'. Atoms that are collectively represented by the '#' symbol are covalently bonded and ***will not*** be used as part of criteria to identify an interaction. In other words, instead of using the '#' symbols, these atoms can be explicitly expressed of which all the element symbols are in lower-case.
- @ The rest of the part of the same functional group that do not participate in any non-bonded interaction. This is different from '#' whereby, atoms that are collectively represented by the '@' symbol ***will be*** used as part of the criteria to identify an interaction. In other words, instead of using the '@' symbol, all these atoms can be explicitly expressed with all the element symbols are in uppercase.
- (X) The bracket is used to indicate an atom X that forms a branch or part of a molecule, which may or may not belong to a Chemical Group.

In general, the atomic species are described in the DL_F Notation. However, in DANAI, the atomic symbols can be expressed in either uppercase or lowercase, which indicates the extent of the atomic interactions on the atom for DL_ANALYSER to identify. If the atom is specified in the uppercase, then such atom that contains precisely the number of interactions, as defined in the *atomic_interaction*, will be considered. If the atom is specified in the lowercase, then such atom will be considered irrespective of the number of different interactions involve with this atom. However, one of such interactions must include that as described in the *atom_description*. In other words, the selection criteria for a given set of atoms that are described in an *atomic_description* is dependent upon the letter case of the atomic symbols.

5.3 Examples – HB Interactions

Consider the macro-interaction between two carboxylic groups, HB_20_20. The value of CGI = 20 refers to the carboxylic group in the DL_F Notation. The micro-interactions between two hydroxyl groups (as contained within the carboxylic group) can be described in a variety way (but not limited to):

Macro-interaction HB_20_20
Micro-interaction [L2]**O20L**–H20O:O20L–**H20O**

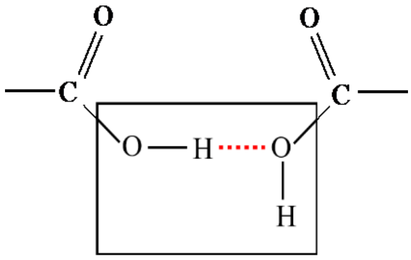
If the macro-interaction is specified involving the same CG, then the corresponding micro-interaction can be abbreviated by removing the CGI value as follows:

[L2]**OL**–H:OL–**H** (1)

Note that the letter 'L' is retained to distinguish the linked oxygen atom from the terminal carbonyl oxygen atom (OE) according to the DL_F Notation.

In this example, DL_ANALYSER will only identify the set of atoms that participates precisely the interaction as indicated by the *atomic_interaction* expression (1). In this case, the hydrogen atom (H) from one carboxylic group and the oxygen atom (OL) from the other carboxylic group forms the hydrogen bond and the two connecting atoms **OL** and **H** from the two carboxylic groups, respectively.

The uppercase letters stipulate that *all* atoms must not involve in any other hydrogen bonding interactions other then what is specified, that is, between the H and the OL. In other words, expression (1) is in fact an annotation of the hydrogen bond formation between two hydroxyl parts of the carboxylic groups in isolation.

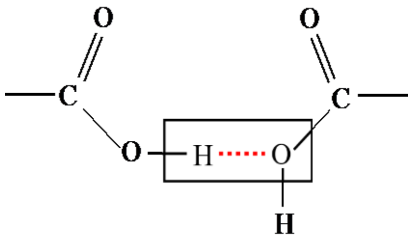


[L2]OL-H:OL-H

However, if one were to concentrate only on the immediate HB between H and OL, the the DANAI expression can be written as

[L2]oL–H:OL–h (2)

In this case, the Program will select the participating atoms H and OL, *if and only if* one hydrogen bond is formed between them and there is no other hydrogen bond interaction with any other atoms in the system. Since the corresponding connected atoms, oL and h are now expressed in lowercase. This will instruct the Program to bypass the task of interaction detection on these atoms. Consequently, Expression (2) can be equivalently shown as simply [L2]H:OL.



[L2]oL-H:OL-h

Obviously, the least strict criteria would be to express the atomic symbols in the lowercase as follows:

[L2]h:oL

In this case, DL_ANALYSER will select the interaction set wherever and whenever it is identified, irrespective whether they form the interaction in isolation or as part of the interaction network in the system.

Finally, combination of uppercase and lowercase can also be allowed. For instance,

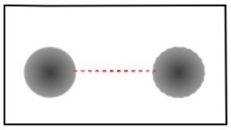
[L2]H:oL

In this case, DL_ANALYSER will identify such interaction if and only if the H atom forms only one hydrogen bond with the oL atom, but irrespective of how many any other hydrogen bonds form with the oL atom.

5.4 Examples – HP Interactions

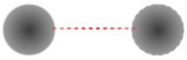
This example illustrates various possible interactions among the various types of alkyl groups, as occur in some aliphatic chains. Each set of interaction can be annotated using the DANAI Notation. The letters p, s and t refers to the primary (**C**H_3), secondary (**C**H_2) and tertiary (**C**H) alkyl carbon, respectively. They are represented collectively as black, red and green spheres which refer to **C**H_3, **C**H_2 and CH, respectively. The solid lines are the covalent bonds between two alkyl carbon atoms. The dotted lines refer to the non-bonded hydrophobic contacts among the spheres.

1



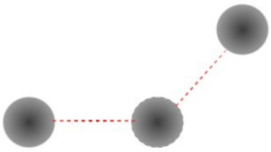
[L2]C1p:C1p

2



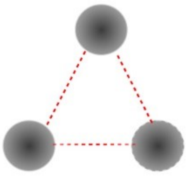
[L2]c1p:c1p

3

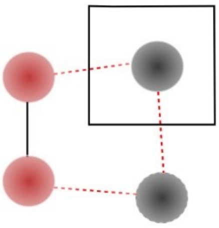


[L3]c1p:c1p:c1p

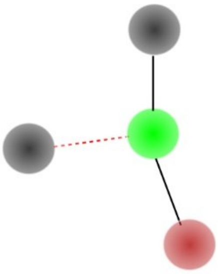
4



[R3]c1p:c1p:1p:c1p

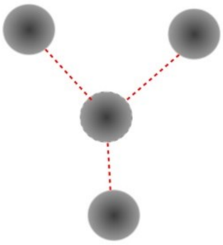


[R3]c1p:c1s– c1s:C1p:c1p



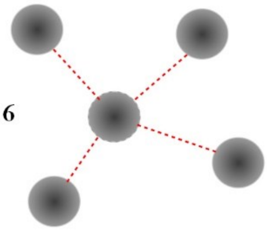
[L2]c1p–c1t(:c1p)–c1s

5



[J4]c1p:c1p(:c1p):c1p

6



[J5]c1p:(c1p:)c1p(:c1p):c1p

6 Example Trajectory File

An example of the HISTORY trajectory file is enclosed with the Program suite. The filename is called *HISTORY_acetic.gz*. It was produced using DL_POLY_4 and the system consists of 674 ethanoic (acetic) acid molecules which gives a total of 5392 atoms.

The file is indicated in the *dl_analyser.input* file and the option to detect HB interactions among the carboxylic groups was switched on (1) in the Interaction Analysis Section of the *dl_analyser.control* file.

There are two ways to run the analysis. First of all, go to the folder *workspace/*:

(1) type *./dl_analyser*
(2) run the script *./run_dla*

The *run_dla* allows users to adjust the number of threads to use in the OMP shared memory computation. Otherwise, by running *./dl_analyser* directly will use the number of threads according to the default set by your computer.

Running DL_ANALYSER should read the file and carry out the analysis as instructed. If there is a problem in reading the compressed file, uncompress it and try again. Remember to rename the file accordingly in the *dl_analyser.input* file.

Please note the following:

(1) The simulation time shown (6440.002 ps onwards) is what was shown in the HISTORY file. This is so because the example file was extracted as a part of trajectories frames from the original, larger HISTORY file.

(2) Once the analysis is completed, inspect the *dl_analyser.output* and *test1.output* files.

(3) The results produce are meant for illustrative purposes only and are likely not to be accurate. Bear in mind that these were produced by averaging over just a few (up to 15) frames.

(4) The sequence of atoms within a molecule is identical and the atoms are grouped together. For example, since there are eight atoms in an ethanoic acid molecule, the indices 1 to 8 refer to a molecule, the following indices 9 to 16 refer to the other molecule, and so forth.

Try different analysis option and see what happens. Due to the nature of the system, the example file is suited only for the following *Analysis Sections*:

Interaction Analysis

Structural Analysis

Dynamical Analysis

DL_ANALYSER version 2.3 User Manual

THE END

C W Yong (September 2021)