Table of Contents

[Analyzing Single-component 'histogram\_complexes\_time.dat' File 1](#_Toc1990763102)

[General histogram function: 1](#_Toc443011772)

[Function for plotting maximum number of monomers in single complex over time: 2](#_Toc69338224)

[Function for plotting mean number of monomers in single complex over time: 3](#_Toc1449516575)

[Function for converting single-species ‘histogram\_complexes\_time.dat’ file to a pandas.DataFrame: 4](#_Toc1345850513)

[Function for converting single-species ‘histogram\_complexes\_time.dat’ file to a .csv file: 5](#_Toc1003885338)

[Function for creating 3D histogram (histogram with time axis): 5](#_Toc515178859)

[Function for creating 2D heatmap showing distribution of N-mers vs. time (low contrast option): 6](#_Toc360326826)

[Function for creating 2D heatmap showing number of total count of monomers inside N-mers vs. time: 8](#_Toc574373952)

[Function for creating 2D heatmap showing fractions of monomers forming N-mers vs. time: 9](#_Toc1291488721)

[Function for converting .xyz file to a .csv file: 11](#_Toc1834837693)

[Function for converting .xyz file to a pandas.DataFrame 11](#_Toc1615041956)

[Function for tracking the trajectory of specific molecule(s) 12](#_Toc1777083932)

[Analyzing 'transition\_matrix\_time.dat' File 12](#_Toc1981601694)

[Function for creating line plot of free energy among different size of complexes: 12](#_Toc2052693610)

[Function for creating line plot of symmetric associate probability: 13](#_Toc166213457)

[Function for creating line plot of asymmetric associate probability: 15](#_Toc2049247315)

[Function for creating line plot of symmetric dissociate probability: 16](#_Toc1549630847)

[Function for creating line plot of asymmetric dissociate probability: 18](#_Toc1588255305)

[Function for creating line plot of growth probability: 19](#_Toc282303550)

[Function for creating line plot of complexes lifetime: 20](#_Toc1200038952)

[Analyzing Multi-component 'histogram\_complexes\_time.dat' File 21](#_Toc163587121)

[Function for converting multi-species ‘histogram\_complexes\_time.dat’ file to a pandas.DataFrame: 21](#_Toc895844884)

[Function for converting multi-species ‘histogram\_complexes\_time.dat’ file to a .csv file: 22](#_Toc882712246)

[Function for generating general multi-species histogram: 22](#_Toc1797213274)

[Function for generating stacked multi-species histogram: 23](#_Toc1839314712)

[Function for generating line plot of maximum size of complex: 25](#_Toc2082368553)

[Function for generating line plot of mean size of complex: 26](#_Toc1911426284)

[Function for generating heatmap for multi-species system: 28](#_Toc862771042)

[Function for generating 3D histogram for multi-species system: 29](#_Toc577800256)

[Locating position for certain size of complexes by PDB/restart file 31](#_Toc2053239477)

[By PDB file: 31](#_Toc287565771)

[By ‘restart.dat’ file: 32](#_Toc1056909761)

[Platonic Solid Self-assembly Input File Writing 32](#_Toc1667391721)

[Reading Real PDB file and Generating Inputs for NERDSS (UI version) 33](#_Toc1188566597)

[Reading Real PDB file and Generating Inputs for NERDSS (Separated functions) 35](#_Toc2121188774)

[Function for returning COM and binding information of chains 35](#_Toc363633852)

[Function for taking specific chains from tuple of COMs and binding information 35](#_Toc2109992264)

[Function for changing distance between 2 binding sites 35](#_Toc160266481)

[Function for calculating the angles between pairs of interfaces (???) 36](#_Toc1300736494)

[Function for normalizing the COM of each chain 36](#_Toc1218684323)

[Function for writing new NERDSS input files based on input chain info 37](#_Toc1426317162)

[Function for creating new PDB file based on input chain info 37](#_Toc51904702)

[Function for creating a 3D plot of each inputted chain 37](#_Toc2093707551)

**Package Analyzing Tools**

## Analyzing Single-component 'histogram\_complexes\_time.dat' File

### General histogram function:

hist (FileName, FileNum, InitialTime, FinalTime, SpeciesName, BarSize, ShowFig, SaveFig)

Description: This function enables users to obtain a histogram during a certain time period of selected species.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* BarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for plotting maximum number of monomers in single complex over time:

max\_complex (FileName, FileNum, InitialTime, FinalTime, SpeciesName, ShowFig, SaveFig)

Description: This function enables users to obtain a plot indicating maximum number of monomers in single complex molecule during a certain time period.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for plotting mean number of monomers in single complex over time:

mean\_complex (FileName, FileNum, InitialTime, FinalTime, SpeciesName, ExcludeSize, ShowFig, SaveFig)

Description: This function enables users to obtain a plot indicating mean number of monomers in single complex molecule during a certain time period.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* ExcludeSize

Acceptable value: Int (Optional)

Default: 0

Description: In the generated plot, the number of monomers in the complex that are no larger than this number will be excluded and will not be considered into the average calculation.

Sample: ExcludeNum = 1 (which means the monomer will be excluded)

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for converting single-species ‘histogram\_complexes\_time.dat’ file to a pandas.DataFrame:

single\_hist\_to\_df (FileName, SaveCsv)

Description: This function enables users to convert the raw .dat file to a data frame in python pandas package for single-species system. Each column in the data frame includes the simulation time and selected occurrences of species during the simulation; each row is separated by a different simulation time.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* SaveCsv

Acceptable value: Bool (Optional)

Default: True

Description: If True, the corresponding .csv file will also be saved as ‘histogram.csv’ under the same directory; if False, the generated .csv file will be deleted.

### Function for converting single-species ‘histogram\_complexes\_time.dat’ file to a .csv file:

single\_hist\_to\_csv (FileName)

Description: This function enables users to convert the raw .dat file to a .csv file for single-species system. Each column in the data frame includes the simulation time and selected occurrences of species during the simulation; each row is separated by a different simulation time.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

### Function for creating 3D histogram (histogram with time axis):

hist\_3d\_time (FileName, FileNum, InitialTime, FinalTime, SpeciesName, TimeBins, xBarSize, ShowFig, SaveFig)

Description: This function enables users to generate 3D histogram representing the number of monomers in single complex as simulation time develops. The x-axis is the number of monomers, y-axis is the averaged time and z-axis is the relative occurrence probabilities.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* TimeBins

Acceptable value: Int (Must provide)

Description: It is the number of bins that users want to divide the selected time period into. The value should be positive integer.

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating 2D heatmap showing distribution of N-mers vs. time (low contrast option):

hist\_time\_heatmap (FileName, FileNum, InitialTime, FinalTime, SpeciesName, TimeBins, xBarSize, ShowFig, ShowMean, ShowStd, SaveFig)

Description: This function enables users to generate 2D histogram of numerical distribution of different N-mers vs. time. The x-axis is the distribution of number of monomers in single complex and y-axis is the time period. The color in each box indicates the number of corresponding N-mers when corresponding time period is reached.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* TimeBins

Acceptable value: Int (Must provide)

Description: It is the number of bins that users want to divide the selected time period into. The value should be positive integer.

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* ShowMean

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding mean value will be shown in the center of each box; if false, there will be no number shown.

* ShowStd

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding std value will be shown in the center of each box; if false, there will be no number shown.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating 2D heatmap showing number of total count of monomers inside N-mers vs. time:

hist\_time\_heatmap\_mono\_count (FileName, FileNum, InitialTime, FinalTime, SpeciesName, TimeBins, xBarSize, ShowFig, ShowMean, ShowStd, SaveFig)

Description: This function enables users to generate 2D histogram of total count of monomers in different N-mers vs. time. The x-axis is the number of monomers in single complex and y-axis is the time period. The color in each box indicates the total number of corresponding monomers in N-mers when corresponding time period is reached.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* TimeBins

Acceptable value: Int (Must provide)

Description: It is the number of bins that users want to divide the selected time period into. The value should be positive integer.

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* ShowMean

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding mean value will be shown in the center of each box; if false, there will be no number shown.

* ShowStd

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding std value will be shown in the center of each box; if false, there will be no number shown.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating 2D heatmap showing fractions of monomers forming N-mers vs. time:

hist\_time\_heatmap\_fraction (FileName, FileNum, InitialTime, FinalTime, SpeciesName, TimeBins, xBarSize, ShowFig, ShowMean, ShowStd, SaveFig)

Description: This function enables users to generate 2D histogram of fractions of monomers forming different N-mers vs. time. The x-axis is the number of monomers in single complex and y-axis is the time period. The color in each box indicates the fraction of monomers forming corresponding N-mers when corresponding time period is reached.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* TimeBins

Acceptable value: Int (Must provide)

Description: It is the number of bins that users want to divide the selected time period into. The value should be positive integer.

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* ShowMean

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding mean value will be shown in the center of each box; if false, there will be no number shown.

* ShowStd

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding std value will be shown in the center of each box; if false, there will be no number shown.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for converting .xyz file to a .csv file:

xyz\_to\_csv (FileName, LitNum):

Description: This function enables users to convert the output .xyz file by NERDSS simulation into a .csv file of a specific or entire time frame. The generated csv file will contain 5 columns, including number of literation, species name, x, y, and z coordinates.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the .xyz file, which is usually names as ‘trajectory.xyz’.

Sample: FileName = ‘/Users/UserName/Documents/ trajectory.xyz’

* LitNum

Acceptable value: Int (Must provide)

Description: It is the number of literation user desire to examine. If the input is -1, the function will extract the entire literation.

Sample: LitNum = 100000000 or LitNum = -1

### Function for converting .xyz file to a pandas.DataFrame

xyz\_to\_df (FileName, LitNum, SaveCsv):

Description: This function enables users to convert the output .xyz file by NERDSS simulation into a pandas.DataFrame of a specific or entire time frame. The generated csv file will contain 5 columns, including number of literation, species name, x, y and z coordinates.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the .xyz file, which is usually names as ‘trajectory.xyz’.

Sample: FileName = ‘/Users/UserName/Documents/ trajectory.xyz’

* LitNum

Acceptable value: int (Must provide)

Description: It is the number of literation user desire to examine. If the input if -1, the function will extract the entire literation.

Sample: LitNum = 100000000 or LitNum = -1

* SaveCsv

Acceptable value: Bool (Optional)

Default: True

Description: If True, the corresponding .csv file will also be saved under the same directory; if False, the generated .csv file will be deleted.

### Function for tracking the trajectory of specific molecule(s)

traj\_track (FileName, SiteNum, MolIndex)

Description: This function enables users to track the COM coordinate changing of one or more molecule. The return will be a 2D matrix with the size of the number of literation times the number of desired molecules.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the .xyz file, which is usually names as ‘trajectory.xyz’.

Sample: FileName = ‘/Users/UserName/Documents/ trajectory.xyz’

* SiteNum

Acceptable value: Int (Must provide)

Description: This is the total number of COM and interfaces of a single molecule. For example, if a molecule possesses 1 COM and 5 interfaces, the SiteNum value should be 6.

Sample: SiteNum = 6

* MolIndex

Acceptable value: List with Int elements (Must provide)

Description: This is the index of molecule users desired to track. The number in the list should be no smaller than 1.

Sample: MolIndex = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

## Analyzing 'transition\_matrix\_time.dat' File

### Function for creating line plot of free energy among different size of complexes:

free\_energy (FileName, FileNum, InitialTime, FinalTime, SpeciesName, ShowFig, SaveFig)

Description: The plot indicates the change in free energy in selected time period among different size of complexes. The x-axis is the size of complex and the y-axis is the free energy calculated as in the unit of , where the refers to the probability of occurrence of the number of times N-mer is counted (including association and dissociation). If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of symmetric associate probability:

associate\_prob\_symmetric(FileName, FileNum, InitialTime, FinalTime, SpeciesName, DivideSize, ShowFig, SaveFig)

Description: This line plot represents the probability of association between complexes of different sizes and other complexes of different sizes. The x-axis is the size of the complex and y-axis is the associate probability. Three lines will exist in the line graph, representing associating to complexes of sizes less than, equal to, or greater than the specified size, respectively. 'Symmetric' in the function name means that for the associate reaction, both sizes of complexes are counted as associating events symmetrically, for example, if an associate event occurs where a trimer associates to a tetramer as a heptamer, then this event is counted twice, which are trimer associates to tetramer and tetramer associates to trimer. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* DivideSize

Acceptable Value: int (Optional)

Default: DivideSize = 2

Description: This is the value that distinguishes the size of the associate complex, for example, if DivideSize = 2, that means the associate events are classified as ‘associate size < 2’, ‘associate size = 2’ and ‘associate size > 2’.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of asymmetric associate probability:

associate\_prob\_asymmetric(FileName, FileNum, InitialTime, FinalTime, SpeciesName, DivideSize, ShowFig, SaveFig)

Description: This line plot represents the probability of association between complexes of different sizes and other complexes of different sizes. The x-axis is the size of the complex and y-axis is the associate probability. Three lines will exist in the line graph, representing associating to complexes of sizes less than, equal to, or greater than the specified size, respectively. 'Asymmetric' in the function name means that for the associate reaction, only the complexes of smaller size associating to the larger one is counted as associate event asymmetrically, for example, if an associating event occurs where a trimer associates to a tetramer as a heptamer, then this event is counted only once, which is a trimer associates to tetramer. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine and plot on the plot. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* DivideSize

Acceptable Value: int (Optional)

Default: DivideSize = 2

Description: This is the value that distinguishes the size of the associate complex, for example, if DivideSize = 2, that means the associate events are classified as ‘associate size < 2’, ‘associate size = 2’ and ‘associate size > 2’.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of symmetric dissociate probability:

dissociate\_prob\_symmetric (FileName, FileNum, InitialTime, FinalTime, SpeciesName, DivideSize, ShowFig, SaveFig)

Description: This line plot represents the probability of dissociation of complexes of different sizes into other complexes of different sizes. The x-axis is the size of the complex and y-axis is the dissociate probability. Three lines will exist in the line graph, representing dissociating to complexes of sizes less than, equal to, or greater than the specified size, respectively. 'Symmetric' in the function name means that for the dissociate reaction, both sizes of complexes are counted as dissociating events symmetrically, for example, if an dissociate event occurs where a heptamer dissociates into a tetramer and a trimer, then this event is counted twice, which are heptamer dissociates to tetramer and heptamer dissociates to trimer. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* DivideSize

Acceptable Value: int (Optional)

Default: DivideSize = 2

Description: This is the value that distinguishes the size of the dissociate complex, for example, if DivideSize = 2, that means the dissociate events are classified as ‘dissociate size < 2’, ‘dissociate size = 2’ and ‘dissociate size > 2’.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of asymmetric dissociate probability:

dissociate\_prob\_asymmetric (FileName, FileNum, InitialTime, FinalTime, SpeciesName, DivideSize, ShowFig, SaveFig)

Description: This line plot represents the probability of dissociation of complexes of different sizes into other complexes of different sizes. The x-axis is the size of the complex and y-axis is the dissociate probability. Three lines will exist in the line graph, representing dissociating to complexes of sizes less than, equal to, or greater than the specified size, respectively. 'Asymmetric' in the function name means that for the dissociate reaction, only the complexes of smaller size dissociating from the original one is counted as dissociate event asymmetrically, for example, if an dissociate event occurs where a heptamer dissociates into a tetramer and a trimer, then this event is counted only once, which is heptamer dissociates to trimer. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* DivideSize

Acceptable Value: int (Optional)

Default: DivideSize = 2

Description: This is the value that separates the size of the dissociate complex, for example, if DivideSize = 2, that means the dissociate events are classified as ‘dissociate size < 2’, ‘dissociate size = 2’ and ‘dissociate size > 2’.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of growth probability:

growth\_prob (FileName, FileNum, InitialTime, FinalTime, SpeciesName, ShowFig, SaveFig)

Description: This line plot indicates the probability of growth in size for different sizes of complexes. The x-axis is the size of complexes, and the y-axis is the growth probability. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for creating line plot of complexes lifetime:

complex\_lifetime(FileName, FileNum, InitialTime, FinalTime, SpeciesName, ShowFig, SaveFig)

Description: This line plot indicates the lifetime for different sizes of complexes. The x-axis is the size of complexes, and the y-axis is the corresponding lifetime in unit of second. If multiple input files are given, the output plot will be the average value of all files and an error bar will also be included.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/transition\_matrix\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘transition\_matrix\_time.dat’); if multiple file is provided, the name of input file should also include serial number as ‘transition\_matrix\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.5

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesName

Acceptable value: String (Must provide)

Description: It is the name of species that users want to examine, which should also be identical with the name written in the input (.inp and .mol) files.

Sample: SpeciesName = ‘clat’

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

## Analyzing Multi-component 'histogram\_complexes\_time.dat' File

### Function for converting multi-species ‘histogram\_complexes\_time.dat’ file to a pandas.DataFrame:

multi\_hist\_to\_df (FileName, SaveCsv)

Description: This function enables users to convert the raw .dat file to a data frame in python pandas package for multi-species system. Each column in the data frame includes the simulation time and selected occurrences of species during the simulation; each row is separated by a different simulation time.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* SaveCsv

Acceptable value: Bool (Optional)

Default: True

Description: If True, the corresponding .csv file will also be saved as ‘histogram.csv’ under the same directory; if False, the generated .csv file will be deleted.

### Function for converting multi-species ‘histogram\_complexes\_time.dat’ file to a .csv file:

multi\_hist\_to\_csv (FileName)

Description: This function enables users to convert the raw .dat file to a .csv file for multi-species system. Each column in the data frame includes the simulation time and selected occurrences of species during the simulation; each row is separated by a different simulation time.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

### Function for generating general multi-species histogram:

multi\_hist (FileName, FileNum, InitialTime, FinalTime, SpeciesList, xAxis, BarSize, ExcludeSize, ShowFig, SaveFig)

Description: This function enables users to plot general histogram of total size of complex or selected species inside each complex for a multi-species system. It will also analyze multiple input files and show the result along with error bar. The x-axis is the size of selected species or total number of monomers, and the y-axis is the average number of counts for corresponding size.

Parameters:

* FileName:

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* xAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If xAxis is included inside SpeciesList, the x-axis will only show the number of selected components. The other input is ‘tot’, which represents the x\_axis will count all species in a single complex.

Sample: xAxis = ‘tot’ or xAxis = ‘A’

* BarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ExcludeSize

Acceptable value: Int (Optional)

Default: 0

Description: In the generated plot, the number of monomers in the complex that are no larger than this number will be excluded and will not be considered into the average calculation.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for generating stacked multi-species histogram:

multi\_hist\_stacked (FileName, FileNum, InitialTime, FinalTime, SpeciesList, xAxis, DivideSpecies, DivideSize, BarSize, ExcludeSize, ShowFig, SaveFig)

Description: This function enables users to plot general histogram of total size of complex or selected species for a multi-species system. Each bar is split by three stacked bars which represent the size distribution of another selected species compared to a desired input. It will also analyze multiple input files and show the result along with error bar. The x-axis is the size of selected species or total number of monomers, and the y-axis is the average number of counts for corresponding size.

Parameters:

* FileName:

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* xAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If xAxis is included inside SpeciesList, the x-axis will only show the number of selected components. The other input is ‘tot’, which represents the x\_axis will count all species in a single complex.

Sample: xAxis = ‘tot’ or xAxis = ‘A’

* DivideSpecies

Acceptable value: String (Must provide)

Description: It indicates the name of the species that users want to separate by size.

Sample: DivideSpecies = ‘A’

* DivideSize

Acceptable value: Int (Must provide)

* Description: This is the value that separates the size of the dissociate complex, for example, if DivideSize = 5, that means the dissociate events are classified as ‘DivideSpecies size < 5’, ‘DivideSpecies size = 5’ and ‘DivideSpecies size > 5’.

Sample: DivideSize = 5

* BarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ExcludeSize

Acceptable value: Int (Optional)

Default: 0

Description: In the generated plot, the number of monomers in the complex that are no larger than this number will be excluded and will not be considered into the average calculation.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for generating line plot of maximum size of complex:

multi\_max\_complex (FileName, FileNum, InitialTime, FinalTime, SpeciesList, SpeciesName, ShowFig, SaveFig)

Description: This function enables users to obtain a plot indicating maximum number of selected monomers in single complex molecule during a certain time period.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* SpeciesName

Acceptable value: String (Must provide)

Description: It indicates the species that user want to examine. If SpeciesName is included inside SpeciesList, the plot will only show the number of selected components. The other input is ‘tot’, which represents the plot will count all species in a single complex.

Sample: SpeciesName = ‘A’ or SpeciesName = ‘tot’

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for generating line plot of mean size of complex:

multi\_mean\_complex (FileName, FileNum, InitialTime, FinalTime, SpeciesList, SpeciesName, ExcludeSize, ShowFig, SaveFig)

Description: This function enables users to obtain a plot indicating mean number of selected monomers in single complex molecule during a certain time period.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* SpeciesName

Acceptable value: String (Must provide)

Description: It indicates the species that user want to examine. If SpeciesName is included inside SpeciesList, the plot will only show the number of selected components. The other input is ‘tot’, which represents the plot will count all species in a single complex.

Sample: SpeciesName = ‘A’ or SpeciesName = ‘tot’

* ExcludeSize

Acceptable value: Int (Optional)

Default: 0

Description: In the generated plot, the number of monomers in the complex that are no larger than this number will be excluded and will not be considered into the average calculation.

Sample: ExcludeNum = 1 (which means the monomer will be excluded)

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for generating heatmap for multi-species system:

multi\_heatmap (FileName, FileNum, InitialTime, FinalTime, SpeciesList, xAxis, yAxis, xBarSize, yBarSize, ShowFig, ShowMean, ShowStd, SaveFig)

Description: This function enables users to generate a heatmap during a certain time period representing the distribution of size of selected species. The x and y axis are both desired individual components and the color of each square represents the relative occurrence probability of complex of corresponding size.

Parameters:

* FileName:

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* xAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If xAxis is included inside SpeciesList, the x-axis will only show the number of selected components.

Sample: xAxis = ‘A’

* yAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If yAxis is included inside SpeciesList, the x-axis will only show the number of selected components.

Sample: yAxis = ‘B’

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* yBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in y-dimension. The y-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* ShowMean

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding mean value will be shown in the center of each box; if false, there will be no number shown.

* ShowStd

Acceptable value: Bool (Optional)

Default: False

Description: If True, the corresponding std value will be shown in the center of each box; if false, there will be no number shown.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

### Function for generating 3D histogram for multi-species system:

multi\_3D\_hist(FileName, FileNum, InitialTime, FinalTime, SpeciesList, xAxis, yAxis, xBarSize, yBarSize, ShowFig, SaveFig)

Description: This function enables users to generate a 3D histogram during a certain time period representing the distribution of size of selected species. The x and y axis are both desired individual components and the height of each column represents the relative occurrence probability of complex of corresponding size.

Parameters:

* FileName:

Acceptable value: String (Must provide)

Description: It is the path to the ‘.dat’ file, which is usually named as ‘histogram\_complexes\_time.dat’, representing the histogram data to be analyzed.

Sample: FileName = ‘/Users/UserName/Documents/histogram\_complexes\_time.dat’

* FileNum

Acceptable value: Int (Must Provide)

Description: It is the number of the total input file. If multiple files are provided, their names should obey the naming rule listed below.

Sample: FileNum = 5

Note (naming rule): If single file is provided, the input file should be named as its original name (‘histogram\_complexes\_time.dat’); if multiple files are provided, the name of input file should also include serial number as ‘histogram\_complexes\_time\_X.dat’ where X = 1,2,3,4,5…

* InitialTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the initial time that users desire to examine. The acceptable range should not smaller than the starting time or exceed the ending time of simulation.

Sample: InitialTime = 0.0

* FinalTime

Acceptable value: Float (Must provide)

Unit: s

Description: It is the final time that users desire to examine. The acceptable range should not smaller than the value of InitialTime or exceed the ending time of simulation.

Sample: FinalTime = 1.0

* SpeciesList

Acceptable value: List with String Elements (Must provide)

Description: It contains all the name of species inside the simulation system.

Sample: SpeciesList = [‘A’, ‘B’]

* xAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If xAxis is included inside SpeciesList, the x-axis will only show the number of selected components.

Sample: xAxis = ‘A’

* yAxis

Acceptable value: String (Must provide)

Description: It indicates the species shown on the x-axis. If yAxis is included inside SpeciesList, the x-axis will only show the number of selected components.

Sample: yAxis = ‘B’

* xBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in x-dimension. The x-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* yBarSize

Acceptable value: Int (Optional)

Default: 1

Description: It is size of each data bar in y-dimension. The y-axis will be separated evenly according to this number and the count of each size range will be sum up and shown together.

* ShowFig

Acceptable value: Bool (Optional)

Default: True

Description: If True, the plot will be shown; if False, the plot will not be shown. No matter the plot is shown or not, the returns will remain the same.

* SaveFig

Acceptable value: Bool (Optional)

Default: False

Description: If True, the plot will be saved as a ‘.png’ file in the current directory; if False, the figure will not be saved.

## Locating position for certain size of complexes by PDB/restart file

### By PDB file:

locate\_position\_PDB (FileNamePdb, NumList, FileNameInp, BufferRatio)

Description: This function enables users to locate specific complexes of certain size from a PDB file after simulation. The result will be output as a separated file named “output\_file.pdb” containing only the desired complex.

Note: Reading only the PDB file slows down the function compared to reading the 'restart.dat' file, because the function needs to calculate the distance between all reactive atoms that can be reacted based on the reaction information to determine whether they are bound or not. Therefore, this function is universal but runs slowly in time.

Parameters:

* FileNamePdb

Acceptable value: String (Must provide)

Description: It is the path to the PDB file, which is usually the last frame of simulation.

Sample: FileNamePdb = ‘/Users/UserName/Documents/999999.pdb’

* Numlist

Acceptable value: List with Int elements (Must provide)

Description: It represents the number of individual monomers in the complex that the user needs to locate, and the order of the monomers in the list is determined by the order in which they appear in the PDB file.

Sample: NumList = [12] (This represents the complexes is consist of only one kind of monomer and the desired complexes is a dodecamer, also known as 12-mer)

* FileNameInp

Acceptable value: String (Must provide)

Description: It is the path to the ‘.inp’ file, which usually stores the reaction information.

Sample: FileNameInp = ‘/Users/UserName/Documents/parms.inp’

* BufferRatio

Acceptable value: Float (Optional)

Default: 0.01

Description: When the sigma value is within the interval of 1.0 plus or minus the ratio, the two reaction interfaces can be considered as bonded.

### By ‘restart.dat’ file:

locate\_position\_restart (FileNamePdb, NumList, FileNameRestart)

Description: This function enables users to locate specific complexes of certain size from a PDB file along with ‘restart.dat’ file after simulation. The result will be output as a separated file named “output\_file.pdb” containing only the desired complex.

Note: The advantage of reading the 'restart.dat' file is that the file directly stores the binding information of each complex in the system and can be used directly, so the function runs faster; however, the function is not universal, if the 'restart.dat ' file's write logic changes, then this function will no longer work.

Parameters:

* FileNamePdb

Acceptable value: String (Must provide)

Description: It is the path to the PDB file, which is usually the last frame of simulation.

Sample: FileNamePdb = ‘/Users/UserName/Documents/999999.pdb’

* Numlist

Acceptable value: List with Int elements (Must provide)

Description: It represents the number of individual monomers in the complex that the user needs to locate, and the order of the monomers in the list is determined by the order in which they appear in the PDB file.

Sample: NumList = [12] (This represents the complexes is consist of only one kind of monomer and the desired complexes is a dodecamer, also known as 12-mer)

* FileNameRestart

Acceptable value: String (Optional)

Default: 'restart.dat'

Description: It is the path to the ‘restart.dat’ file.

## Platonic Solid Self-assembly Input File Writing

Platonic solid self-assembly include 10 models, so that 10 separate functions are needed, which are shown in the following table:

|  |  |  |
| --- | --- | --- |
| Platonic Solid | Center-of-Mass Position | Name of Function |
| Tetrahedron (4-face) | Each Face | tetr\_face (radius, sigma) |
| Tetrahedron (4-face) | Each Vertex | tetr\_vert (radius, sigma) |
| Cube (6-face) | Each Face | cube\_face (radius, sigma) |
| Cube (6-face) | Each Vertex | cube\_vert (radius, sigma) |
| Octahedron (8-face) | Each Face | octa\_face (radius, sigma) |
| Octahedron (8-face) | Each Vertex | octa\_vert (radius, sigma) |
| Dodecahedron (12-face) | Each Face | dode\_face (radius, sigma) |
| Dodecahedron (12-face) | Each Vertex | dode\_vert (radius, sigma) |
| Icosahedron (20-face) | Each Face | icos\_face (radius, sigma) |
| Icosahedron (20-face) | Each Vertex | icos\_vert (radius, sigma) |

Description: This function enable users to generate NERDSS input files (.inp and .mol files) for Platonic solid self-assembly system.

Parameters:

* radius

Acceptable value: Float (Must provide)

Unit: nm

Description: It is the radius of the Platonic solid, which is defined by the distance from the center of Platonic solid to each vertex.

Sample: radius = 40.0

* sigma

Acceptable value: Float (Must provide)

Unit: nm

Description: It is the distance of each interface when reaction takes place.

Sample: sigma = 1.0

## Reading Real PDB file and Generating Inputs for NERDSS (UI version)

real\_PDB\_UI ()

Description: This function enables users to read a real PDB file and extract the coordinate information stored in it, and generate NERDSS-readable input files (.mol and .inp files). This function possesses a user interface so it should be run in an interactive IDEs instead of Jupyter Notebook.

Tutorial:

First, store a real PDB file in the same path with the python code and call this function in propriate IDE (here take VSCode for example). The interface will require user to input the name of the desired PDB file. Type in the full name of the file and press return to continue. See screenshot below for details.



Once the file name is input, the code will read the desired information inside this PDB file and show some basic parameters on the interface (this will take for a while), including the name of each chain, size of each chain, the coordinate of each COM and each pair of interfaces.

A picture containing table

Description automatically generated

Among all pairs of interfaces, users then are required to change the distance between interfaces (sigma), if type in ‘yes’, users can change any of the distance shown above or change all sigma into a same value; if type in ‘no’, the distance will not be changed

Graphical user interface, text

Description automatically generated with medium confidence

Then, users are asked if they intend to use the default vector (0,0,1) as normal vector. If type in ‘yes’, normal vectors for all interfaces will be set as (0,0,1); If type in ‘no’, users are able to manually input desired normal vectors in a format of ‘1,1,1’ (without parentheses). If a colinear issue takes place (which is an extremely rare situation), the algorithm will automatically detect it and for the default vector, it will take (0,1,0) instead and for manually input vector, users are required to input the vector again until the colinear issue is resolved.

Text

Description automatically generated

At last, users are asked if they want each chain to be centered at COM. If type in ‘yes’, the COM coordinate will be normalized as (0,0,0) and the corresponding coordinates of all interfaced will be all changed accordingly in the final output; if type in ‘no’, the coordinate for all COM and interfaces will stay the same as the original ones.



The code will then automatically quit and the corresponding input (multiple .mol files and single .inp file) will be found in the same directory as the Python file.

## Reading Real PDB file and Generating Inputs for NERDSS (Separated functions)

### Function for returning COM and binding information of chains

real\_PDB\_separate\_read (FileName,ChainsIncluded)

Description:

This function will extract the coordinate information stored inside a real PDB file and calculate the COM of each unique chain, as well as recognize the binding information between each pair of chains (all atoms of different unique chains that are closer that 3.0 angstroms are considered as binded), including whether two chains are binded and the coordinates of each binding interface. All the information will be printed on the screen and the returns will contain all the information for further analysis.

Parameters:

* FileName

Acceptable value: String (Must provide)

Description: The full path of the desired PDB file or name of the file if in same directory.

Sample: FileName = ‘/Users/UserName/Documents/1utc.pdb’

* ChainsIncluded  
  Acceptable value: list (optional)  
  Description: A list of which chains you want to be included. Must be more than 2.  
  Sample: ChainsIncluded = [‘A’,’B’]

### Function for taking specific chains from tuple of COMs and binding information

real\_PDB\_separate\_filter (Result, ChainList)

Description:

This function will filter the desired chain according to the input list of chain and exclude all the unnecessary coordinate information for future analysis.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_read (FileName)’.

Sample: Result = result1

* ChainList

Acceptable value: List with String elements

Description: The desired name of chains that users intend to examine.

Sample: ChainList = [‘A’, ‘B’]

### Function for changing distance between 2 binding sites

real\_PDB\_separate\_sigma (Result, ChangeSigma, SiteList, NewSigma)

Description:

This function allows users to change the value of sigma (the distance between two binding interfaces). The new sigma value and the corresponding coordinates of interfaces will be shown on the screen and the returns will contain all the information for further analysis.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_read(FileName)’.

Sample: Result = result1

* ChangeSigma

Acceptable value: Bool (Optional)

Default: False

Description: If True, the users are capable of changing the sigma value; if False, the sigma will remain as the original ones.

* SiteList

Acceptable value: List with Int elements (Optional)

Default: []

Description: It consists of the serial numbers of the pair of interfaces for which the user needs to modify the sigma value. The serial number is determined by the pairing sequence shown by the function ‘real\_PDB\_separate\_read’. The serial number should be no greater than the total number of interface pairs and no smaller than 0. If the serial number is 0, it means to change all pairs of interfaces into a same sigma value.

* NewSigma

Acceptable value: List with Float elements (Optional)

Default: []

Description: It consists of the serial numbers of the pair of interfaces for which the user needs to modify the sigma value. The serial number is determined by the pairing sequence shown by the function ‘real\_PDB\_separate\_read’.

Description: It consists of the actual sigma value that users desire to change, according to the sequence of input ‘SiteList’.

### Function for calculating the angles between pairs of interfaces (???)

real\_PDB\_separate\_angle (Result)

Description:

This function will calculate the 5 associating angles of each pair of interfaces. The default normal vector will be assigned as (0, 0, 1). If the co-linear issue occurs, the system will use (0, 1, 0) instead to resolve co-linear issue. The calculated 5 angles will be shown on the screen automatically. If user intends to manually input the normal vector, please refer to function ‘real\_PDB\_UI’, the separated function does not support manual inputs. The returns will contain all the information for further analysis.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_sigma (Result, ChangeSigma, SiteList, NewSigma)’.

Sample: Result = result2

### Function for normalizing the COM of each chain

real\_PDB\_separate\_COM (Result)

Description:

This function will normalize the COM of each chain as (0, 0, 0). The interface of each chain will be subtracted by the COM coordinates accordingly. Once the calculation is completed, there will be message shown on the screen. The returns will contain all the information for further analysis.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_angle (Result)’.

Sample: Result = result3

### Function for writing new NERDSS input files based on input chain info

real\_PDB\_separate\_write (Result)

Description:

This function will write ‘.inp’ and ‘.mol’ files according to all the calculations and modifications above. Multiple ‘.mol’ file and a ‘.inp’ file can be found in the same directory as the Jupyter Notebook file once the function finish running.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_COM (Result)’.

Sample: Result = result4

### Function for creating new PDB file based on input chain info

real\_PDB\_show\_PDB (Result)

Description:

This function will generate a PDB file that only contains the calculated COMs and reaction interfaces for visualization and comparison with the original PDB file. The input will be the returns of the previous function. Besides, the unit for the coordinates in PDB file is in Angstrom but not nm, so the value will be 10 times larger than that in NERDSS input files.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_read (FileName)’.

Sample: Result = result1

### Function for creating a 3D plot of each inputted chain

real\_PDB\_show\_3D (Result)

Description:

This function will generate a 3D plot indicaiting the spacial geometry of each simplified chain. The solid lines of different colors are connecting the COM with interfaces within each chain; the black dotted line are connecting each pair of interfaces and the COMs are shown as solid points with their names above. To interact with the plot, other IDEs rather than Jupyter Notebook (such as VSCode) are recommended.

Parameters:

* Result

Acceptable value: Tuple (Must provide)

Description: The output result of function ‘real\_PDB\_separate\_read (FileName)’.

Sample: Result = result1