Terphenyl folding Documentation

Release 0.0

Garrett A. Meek
Theodore L. Fobe
Benjamin J. Coscia

Research group of Professor Michael R. Shirts

Dept. of Chemical and Biological Engineering
University of Colorado Boulder

Sep 22, 2019

CONTENTS

This documentation is generated automatically using Sphinx, which reads all docstring-formatted comments from Python functions in the 'terphenyl_folding' repository. (See terphenyl_folding/doc for Sphinx source files.)

CHAPTER

ONE

TERPHENYL FOLDING SIMULATION UTILITIES

Shown below are tools that allow simulation of terphenyl oligomers.

simulation.**build_directories**(*polymer_name*, *polymer_length*, *fresh_run=False*)

Given a set of input strings, this function builds the directories that are needed to perform GROMACS simulations with terphenyl oligomers.

Parameters

- **polymer_name** (*str*) – The name of the polymer
- **polymer_length** (*str*) – The length of the polymer we are modeling (in monomer units)
- **run_directory** (*str*) – The directory where simulations will be run
- **fresh_run** (*Logical*) – A logical variable determining whether old run files should be removed.

Returns

- run_directory (str) - The path to a directory where simulations will be run.
- pdb_file (str) - The path to pdb file that will be used for simulations.
- solvent_file (str) - The path to a file containing a box of solvent molecules
- topology_file (str) - The path to a file containing the topology for the pdb_file.

simulation.**compress_large_files**(*directory*, *size_threshold=100000000.0*)

simulation.**equilibrate**()

simulation.**minimize**(*mdrun_file*, *topology*, *input_structure*)

simulation.**parameterize**(*param_directory*, *pdb_file*, *topology_file*)

Given a directory path, PDB file, and a topology file, this function parameterizes the structure with GAFF.

Parameters

- **param_directory** (*str*) – The path to a directory where intermediate and output parameterization files will be written.
- **pdb_file** (*str*) – The path to a PDB file containing data for the structure that will be parameterized.
- **topology_file** (*str*) – The path to a file containing the topology for the structure that will be parameterized.

simulation.**replace**(*file*, *original_text*, *replacement_text*)

Given a file, a target search string, and a replacement string, this function replaces the text in 'file'.

Parameters

- **file** (*file*) – A file containing the text that will be replaced.
- **original_text** (*str*) – Text that will be replaced.
- **replacement_text** (*str*) – Text that will be used to replace the original text.

simulation.**simulate**()

simulation.**solvate**(*polymer_name*, *polymer_length*, *polymer_code*, *input_pdb*, *run_directory*, *solvent_density=0.5*)

CHAPTER

TWO

TERPHENYL FOLDING ANALYSIS TOOLS

Shown below are functions/tools that allow analysis of terphenyl oligomer simulation results.

analysis.**construct_selector**(*atom_list*, *res_dict*, *n_residues*)
Since our molecule is all in one residue, we need some interesting ways to extract residues, instead of a simple select string

atom_list [list] String name of base atoms in the first residue

res_dict [dict] Dictionary telling how many atoms of a given element are in each residue

analysis.**get_internal_coordinate_definitions**(*structure*, *polymer_name*, *polymer_length*)
structure: pdb file path

polymer_name: string (ie: “o-terphenyl”)

polymer_length: string (ie: “monomer”)

analysis.**get_phenyl_carbon_indices**(*polymer_name*, *polymer_length*)
polymer_name: string (ie: “o-terphenyl”)

polymer_length: string (ie: “monomer”)

phenyl_indices_list: List(dict(‘1’: [phenyl-1 carbon indices], ‘2’: [phenyl-2 carbon indices], ‘3’: [phenyl-3 carbon indices]))

analysis.**get_phenyl_centers_of_mass**(*structure*, *polymer_name*, *polymer_length*)
structure: pdb file path

polymer_name: string (ie: “o-terphenyl”)

polymer_length: string (ie: “monomer”)

analysis.**read_trajectory**(*structure*, *trajectory*)
structure: pdb file path

trajectory: xtc file path

PYTHON MODULE INDEX

INDEX