TJW Stretched Polymer Monte Carlo Simulation: User Guide

1 Author contact details

If you have any enquiries or issues regarding use of this code, please contact me via *tom.wicks1988@gmail.com*.

2 Installation and compilation

This code is entirely written in C and can be compiled using either *icpc* or *gcc*, with OpenMP enabled. Follow the steps below to install the code on your system.

1. Download the zip file containing all files and extract them to your desired working directory.
2. Compile the code, using *make*. By default, the code compiles with *icpc*. To compile with *gcc*, comment and uncomment the appropriate lines in the makefile, following the instructions in the file itself.
3. On successful compilation, the executable *singleChainOMP* is generated.

3 Running the code

There are two components to running the code successfully:

1. Enter the desired inputs into an input file. An exemplar input file can be found in the root directory (inputFileEXAMPLE.dat). This file contains some typical values for the parameters to get you started.
2. Create the output directories for storing the data generated by the code.

Details are provided in the following subsections for how to perform these tasks.

3.1 Setting up the input file

The input file contains all of the information needed by the code to perform a simulation. The following table provides a description of each input variable to be set by the user, along with its type.

Variable	Type	Description
N	int	Number of particles in the chain.
LAMBDA	double	Square-well interaction range.
TEMP_INTEREST	int	The indicial value of the temperature of interest (where the counter starts from 0).
NUM_TEMPS	int	The number of temperatures to be simulated. This is also the number of parallel threads to be initiated.
TEMPS_FILE	string	File containing the temperature profile.
fileInput	int	Toggles whether to use an input file for initial configurations or to generate a planar zigzag.
CONFIGS_FILE	string	File containing the initial configurations of each chain.
MSIZE_DIR	string	Directory containing files with initial move sizes for each type of move.
PHI_MAX	double	The initial maximum size of a bond-torsion move.
ALPHA	double	The relative influence of the Lennard-Jones interaction on the move acceptance probability.
BETA	double	The relative influence of the bond-torsion energy on the move acceptance probability.
GAMMA	double	The relative influence of the bond-angle energy on the move acceptance probability.
sigma	double	Stretching coefficient.
stretchStart stretchFinish	int int	Stretching occurs between the particles with indices stretchStart and stretchFinish.
Nmin Nmax	int int	Energy states between Nmin and Nmax will be sampled.
kappaLHS kappaRHS	double double	These variables define the steepness of the quadratic arm on the left and right-hand side of the sampling window.
UPDATE	int	Toggles the automated feedback mechanism for the biasing function.

Variable	Type	Description
tolerance	double	The tolerance (0.0-1.0) for the automated checks for uniform sampling in the biased distribution.
PRINT_INTERVAL	double	Number of moves between updates of the output files.
PHI_UPDATE_INTERVAL	double	Number of moves between updates of PHI_MAX.
UPDATE_INTERVAL	double	Number of SWAP_INTERVALs between updates of the biasing function.
SWAP_INTERVAL	double	Number of moves between swap cycles.
NUM_SWAPS	int	Number of configuration swaps in each swap cycle.
NUM_SWAP_INTERVALS	double	Number of swap intervals to complete over the whole simulation.
RESET_INTERVAL	double	Number of moves between resetting the counters and initiating a new sampling block.
SMALL_ANGLE_MOVES REPTATION_MOVES CRANK_MOVES END_ROT_MOVES END_BRIDGE_MOVES	double double double double double	The probability of proposing each move-type.
OUTPUT_DIR	string	The directory where all the output files are to be sent.
weightFile	string	The file containing the biasing function.
VMDdir	string	The directory to send files for generating 3D images of configurations using Visual Molecular Dynamics.
initCoords	string	File containing initial coordinates to be copied to all configurations if fileInput is set to 0.

Examples of the input files referred to in the table are provided in the *EXAMPLES* folder.

3.2 Creating the output directories

Use the executable *createDirs.sh* to create the output directories for storing the output data from the code. You will be asked to enter a name for your desired directory. The following subdirectories are generated:

- outputFiles/MyOutputDirectory/
  - outputFiles/MyOutputDirectory/occupancies/

  - outputFiles/`MyOutputDirectory`/restartInfo/

- VMD/`MyOutputDirectory`/

Note: generate the output directory *EXAMPLE* to use the exemplar input file *inputFileEXAMPLE.dat*.

3.2.1 Deleting output data

To delete an output directory, use the executable *deleteDirs.sh*. WARNING: this will remove the directory and all of its contents, so use with caution!

4 Output files

The following files are outputted by the code into the directory outputFiles/`MyOutputDirectory`/:

- *crossings_chainX.dat*: These files contain information regarding the number of "zero crossings" chain X has experienced. This is an indication of the number of times the chain has completed N (forward or backward) reptation moves and so has replenished its configuration.

- *EtraceX.dat*: At each swap interval, these files update with the current temperature of chain X, where each integer refers to the index in the temperature profile.These temperature traces are used to determine how well-mixed the simulation is.

- *FE_chainX.dat*: The latest estimate of the free energy of chain X as a function of energy state.

- *occW_chainX.dat*: The latest occupancy of each energy state by chain X.

- *upwardMoves_chainX.dat*: The latest information regarding number of attempted and accepted upward moves from each energy state visited by chain X.

- *occupancies/block_K.dat*: These file contain the occupancies of each chain in sampling block K. The nth entry in line m of the file refers to chain n's occupancy of state m.

- *restartInfo/*: The files in this subdirectory are used to restart a simulation where the current one finishes. This is particularly useful to avoid the need to repeat a long equilibration process.

  - *configs.dat*: This file contains the latest configurations of all chains.
  - *crankMax.dat, endMax.dat, phiMax.dat*: These files contain the latest optimal move sizes for crank-shaft, end-rotation and torsional moves respectively.

If the Visual Molecular Dynamics output is enabled, the following files are outputted to the directory VMD/`MyOutputDirectory`/:

- *coordinatesX_Y.pdb*: A file that can be interpreted by VMD containing information for the Yth realisation of a configuration in energy state X.
- *bondsX_Y.psf*: A file containing the corresponding bond information.
- *StateX_Y.vmd*: An executable that can be run in the command line of VMD to generate the visualisation corresponding with the coordinates and bonds files with given values of X and Y.