



---

# Two Phase Thermodynamics (2PT) Program for entropy and Free Energies of Complex Systems

---

## Funding

---



行政院國家科學委員會  
National Science Council



## **i. Copyright notice**

The Two-Phase Thermodynamics (2PT) program is distributed open-source and free of charge by the authors to any academic, commercial or government entity for calculating thermodynamic properties with the supported Molecular Dynamics (MD) simulation packages. If you were charged for this code or received any solicitation for monies from any source, please inform the authors. You, the user, are free to modify the source as needed. We only ask that you not redistribute the source to a third-party. If you add a feature to the code that you deem to be of broad interest to the rest of the community, we encourage you to send us your modifications for inclusion in the distributed source. If you find the code useful for your work, please cite the method as shown in part iii and send us an email so that we can add your paper to the list.

2PT program, Copyright 2010, Caltech, Pasadena, CA USA

Copyright © 2011  
Materials and Process Simulation Center (MSC),  
California Institute of Technology,  
Pasadena, CA 91125 USA  
All rights Reserved

## ii. Authorship

The 2PT code methodology was developed by Dr. Shiang-Tai Lin, Dr. Mario Blanco, and Prof. William A. Goddard in the Materials and Process Simulation Center (MSC), at the California Institute of Technology, in Pasadena CA 91125, USA in 2002-2003 and published in J. Chemical Physics, 119: 11792-11805 (2003). The 2PT code implementing this methodology was written by Dr. Shiang-Tai Lin during his post-doctoral appointment in the lab of William A. Goddard III at Caltech, 2000 – 2003. Currently, it is also being actively developed and supported by Tod A. Pascal at Caltech/KAIST, Prabal K. Maiti at IISC.

### ii.a. Affiliations & Contact info

- Shiang-Tai Lin, Department of Chemical Engineering, National Taiwan University, Taipei, Taiwan. email: [stlin@ntu.edu.tw](mailto:stlin@ntu.edu.tw) web: [http://www.che.ntu.edu.tw/ntuche/eng/prof\\_detail.php?id=20](http://www.che.ntu.edu.tw/ntuche/eng/prof_detail.php?id=20)
- Tod A. Pascal, Materials and Process Simulations Center (MSC), California Institute of Technology (Caltech), Pasadena, CA USA 91106 and Center for Materials Simulations and Design (CMSD), Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea. email: [tpascal@wag.caltech.edu](mailto:tpascal@wag.caltech.edu). web: <http://www.wag.caltech.edu/home/tpascal/>
- Prabal K. Maiti, Department of Physics, Indian Institute of Science, Bangalore, India 560012. email: [maiti@physics.iisc.ernet.in](mailto:maiti@physics.iisc.ernet.in). web: <http://www.physics.iisc.ernet.in/~maiti/>
- William A. Goddard III, MSC Caltech and CMSD Kaist. email: [wag@wag.caltech.edu](mailto:wag@wag.caltech.edu). web: <http://www.wag.caltech.edu>

### ii.b. Version information

- Version 0.1 June 1<sup>st</sup> 2002 – Original Fortran code by Georgios Zamanakos and Jianwe Che at Caltech. Based on 1PT method of Ken Wilson at U. San Diego: J. Chem. Phys. 79(5), 2375, 1983
- Version 0.2 January 23<sup>rd</sup> 2003 – A mixture of C and FORTRAN code. Revision by Shiang-Tai Lin at Caltech to apply 2PT correction to system thermodynamics
- Version 0.3 April 24<sup>th</sup> 2007 – Code rewritten entirely in C++. Revision by Shiang-Tai Lin at NTU to generalize for molecular systems
- Version 0.3.1 December 12<sup>th</sup> 2009 – Support for various trajectories added by Tod A. Pascal at Caltech/KAIST. Code optimized for readability and efficiency. Memory management simplified
- Version 0.3.2 February 7<sup>th</sup> 2010 – Support for AMBER trajectories added by Prabal K Maiti at IISC
- Version 0.3.2.1. May 14<sup>th</sup> 2010 – Minor bug fixes and enhancements. Simplified thermo output

### iii. Citations & relevant papers

#### Please include the following when citing 2PT

- S.T. Lin, M. Blanco and William A. Goddard, *J. Chem. Phys.*, 2003, 119, 11792-11805
- S.T. Lin, P.K. Maiti and W.A. Goddard, *J. Phys. Chem. B.*, 2010, 114, 8191-8198
- T.A. Pascal, S.T. Lin and W.A. Goddard, *Phys Chem Chem Phys*, 2011, 13, 169-181

#### The following lists other papers that made use of 2PT

- Lin, S. T.; Maiti, P. K.; Goddard, W. A. *J. Phys. Chem. B* **2005**, 109, 8663.
- Jana, B.; Pal, S.; Maiti, P. K.; Lin, S. T.; Hynes, J. T.; Bagchi, B. *J. Phys. Chem. B* **2006**, 110, 19611
- V. Vasumathi and P.K. Maiti, *Macromolecules*, **2010** 43 (19), 8264-8274
- T. A. Pascal, R. Abrol, R. Mittal, Y. Wang, N. V. Prasadarao and W. A. Goddard, *J. Biol. Chem.*, 2010, DOI: 10.1074/jbc.M110.122804.
- A. Debnath, B. Mukherjee, K. G. Ayappa, Prabal K. Maiti and S. T. Lin *J. Chem. Phys.*, 133, 174704 (2010)

#### The following lists some relevant papers and alternative Free Energy Methods

- W. C. Swope, H. C. Andersen, P. H. Berens and K. R. Wilson, *J. Chem. Phys.*, 1982, 76, 637-649.
- P.H. Berens, D.H.J. Mackay, G.M. White, and K.R. Wilson, *J. Chem. Phys.*, 1983, 79, 2375-2389.
- Young, T.; Abel, R.; Kim, B.; Berne, B. J.; Friesner, R. A. *Proc. Natl. Acad. Sci. U. S. A.* 2007, 104, 808.
- White, R. P.; Meirovitch, H. *Proc. Natl. Acad. Sci. U. S. A.* 2004, 101, 9235.
- White, R. P.; Meirovitch, H. *J. Chem. Phys.* 2004, 121, 10889.
- Lazaridis, T.; Karplus, M. *J. Chem. Phys.* 1996, 105, 4294.
- Wang, L.; Abel, R.; Friesner, R. A.; Berne, B. J. *J. Chem. Theory Comput.* 2009, 5, 1462.
- Sharma, R.; Agarwal, M.; Chakravarty, C. *Mol. Phys.* 2008, 106, 1925.
- Tyka, M. D.; Sessions, R. B.; Clarke, A. R. *J. Phys. Chem. B* 2007, 111, 9571.
- Henchman, R. H. *J. Chem. Phys.* 2007, 126.
- Andricioaei, I.; Karplus, M. *J. Chem. Phys.* 2001, 115, 6289.
- Zielkiewicz, J. *J. Chem. Phys.* 2008, 128.

## Table of Contents

i. Copyright notice.....	i
ii. Authorship.....	ii
ii.a. Affiliations & Contact info .....	ii
ii.b. Version information .....	ii
iii. Citations & relevant papers.....	iii
Chapter 1. Compiling & Running.....	3
1.a. Quick Start.....	3
1.a.i. General Recommendations for trajectories .....	3
1.b. Prerequisites .....	3
1.c. The Makefile.....	4
Chapter 2. Background & Introduction.....	5
2.a. Program capabilities & scope of use.....	5
2.a.i. Appropriate tasks .....	5
2.a.ii. Supported Trajectory formats.....	5
2.a.iii. Supported input structure/topology files .....	6
2.a.iv. Operating systems & compilers .....	6
Chapter 3. Input & Output Files.....	7
3.a. The Control file (.ctl) .....	7
3.a.i. Specifying the input structure.....	7
3.a.ii. Specifying the input trajectory.....	8
3.a.iii. Specifying the output file .....	10
3.a.iv. Trajectory options .....	10
3.a.v. ANALYSIS_VACMEMORYMB .....	11
3.a.vi. Molecule and degrees of freedom.....	11
3.a.vii. Specifying thermodynamic inputs.....	12
3.b. The Group definition file.....	13
3.b.i. Specifying atoms .....	14
3.b.ii. Specifying Constraints.....	14
3.b.iii. Specifying Linear molecules.....	14
3.b.iv. Specifying external rotational numbers.....	15
3.b.v. Specifying group energies.....	15

3.c. The Output files.....	15
3.c.i. Thermodynamic output file (xxx.thermo).....	15
3.c.ii. Velocity autocorrelation file (xxx.vac).....	16
3.c.iv. Degree of freedom file (xxx.3n).....	16
3.c.v. Screen dump file (xxx.log) .....	16
Chapter 4. Theory .....	17
5.a. The density of state function .....	17
5.b. Thermodynamic properties of systems undergo harmonic motions.....	18
5.c. Thermodynamic properties of Mono-atomic fluids.....	19
5.d. Thermodynamic properties of polyatomic fluids .....	20
References & Notes .....	23

2PT program, Copyright 2010, Caltech, Pasadena, CA USA

## Chapter 1. Compiling & Running

### 1.a. Quick Start

Assuming that you are running on a supported platform with an appropriate compiler, you can compile and use the code using the following 5 steps:

- a. Extract the contents of the compressed file into your favorite directory:  
`tar -zxof 2pt.tar.gz` (on linux)
- b. Navigate to the 2pt/src directory and edit the Makefile to point to the location of the FFTW3 library
- c. Type 'make install' in this directory. If all goes well you should have a binary called 2pt\_analysis in the ../bin directory
- d. Create your control file
- e. Run the analysis using "{install\_directory}/2pt/bin/2pt\_analysis {control\_file}"

### 1.a.i. General Recommendations for trajectories

1. We suggest that the trajectory files are saved at a frequency that 1/5 the fastest molecular vibration. A general rule of thumb is to save the trajectory at twice the simulation's timestep. Thus, for biological systems or for flexible water molecules, this implies a save frequency of  $\sim 2$ fs. If SHAKE/RATTLE constraints are employed, the trajectory can be saved less frequently at 4fs. Simulations involving bonds breaking and formations usually have much smaller (0.5 fs) timesteps.
2. We suggest that at least 5000 frames of the trajectory is saved for condensed phase systems at STP. Thus at a save frequency of 2fs, this implies 10ps trajectories. Systems that are more gas-like should be saved for longer periods to allow sufficient collisions between atoms. Thus in simulating water vapor, we found that trajectories  $\sim 100$  ps are required to obtain converged results. The user is advised to examine the system thermodynamics using successively larger trajectory lengths to ensure convergence.

### 1.b. Prerequisites

The only requirement is the FFTW3 library for doing Fourier Transforms. This may already be installed on some Linux systems. Alternatively, you can get a copy here <http://www.fftw.org/download.html>. The code has been successfully compiled using version 3.2.2.

## 1.c. The Makefile

```
# RedHat Linux 64 bit machine, Intel icpc 64 bit version 11.1, FFTW 3.2.2

SHELL = /bin/sh

# -----
# compiler/linker settings
# edit this section if additional compiler/linker flags or
# libraries are needed for your machine

CC = icpc ← ①
FFTW = /home/tpascal/codes/fftw/3.2.2 ← ②
OPTFLAGS = -fast -xSSE4.2 -I${FFTW}/include
LIBFLAGS = -lm -lfftw3 -lz -L${FFTW}/lib
CDEBUG =

# generally no need to edit the following sections

# Compile file list

OBJ5 = driver.o model.o structure.o control.o analysis.o utility.o \
      trajectory.o timing.o memory.o compute_vac.o statistics.o
```

Figure 1. Example of 2PT Makefile

There should only be 2 places that you need to change here: the compiler locations (item 1 in Fig. 1) and your FFTW3 locations (item 2 in Fig. 1). Note, the FFTW3 libraries may already be installed in /usr or /usr/local (check with your system administrator).



## Chapter 2. Background & Introduction

The Two-Phase Thermodynamics (2PT) program is designed to calculate equilibrium thermodynamic properties of condensed phase systems accurately from short MD trajectories. The original code was written to compute the phase diagram of a Lennard Jones fluid with the same accuracy as exhaustive Monte-Carlo calculations, while being order of magnitude more efficient. It has since been extended to molecular systems, and is able to handle pure liquids as well as heterogeneous systems such as solvated bio-molecules.

All code is provided “as is”, with no guarantee of accuracy or completeness. The user is urged to check the results against others in literature or with other methods for verification. Further, the released version has a number of experimental features that can at best be described as “alpha” code (see list below). No support is provided for these parts of the code and the user uses them at their own risks. While every effort was made to eliminate any bugs, there is always that possibility as with any computer code. Users are advised to carefully examine their input files for errors, as 99% of the times the source of any problem lies here. For the other 1%, users are kindly asked to submit a bug report to the authors.

We have endeavored to optimize the code for versatility and ease of use and not speed (hence the use of C++), however due to the inherent efficiency of the method, you will similar speeds to the newer approximate approaches of calculating free energies (Cell Theory, Pair-correlation functions etc) while being orders of magnitude faster than the more common Thermodynamic Integration and Umbrella Sampling schemes.

### ***2.a. Program capabilities & scope of use***

#### **2.a.i. Appropriate tasks**

- a. Calculate accurate thermodynamic properties of equilibrated systems
- b. Calculate accurate thermodynamic properties of meta-stable systems, so long as the system is in quasi-equilibrium over the sampling window (typically 10 – 20 ps)
- c. Calculate absolute entropies using quantum mechanical weighting functions
- d. Calculate quantum effects (zero-point energy corrections to the internal energy) and thermal effects (specific heat corrections to the internal energy)
- e. Calculate quantum specific heat capacities

#### **2.a.ii. Supported Trajectory formats**

- a. CERIUS2/MPSIM/Polygraf/Material Studios from MSI/Accelrys
- b. AMBER (.trj)
- c. LAMMPS (.lammmps)
- d. XYZ (.xyz)
- e. CHARMM (**experimental** – not fully supported or implemented)

### 2.a.iii. Supported input structure/topology files

- a. MSI Biograf format (.bgf)
- b. LAMMPS data/topology files (xxx.data or data.xxx)
- c. AMBER .prmtop files from their leap code for example

**TIP:** A variety of other file formats can be used by first converting them to the BGF format using Openbabel ([http://openbabel.org/wiki/Main\\_Page](http://openbabel.org/wiki/Main_Page)). Simply use the `-obgf` option to create a bgf file.

### 2.a.iv. Operating systems & compilers

#### Architectures

The code has been extensively tested on the following architectures

- a. Linux (x86 and 64 bit machines)
- b. SGI Altix 4700
- c. IBM PowerPC G4 and G5

#### Compilers

The code has been compiled using the following compilers

- a. GNU g++ (version 3.3.2)
- b. Intel C/C++ compilers icpc (versions 9 thru 11)
- c. AMD Open64

2PT cannot and does not pretend to run MD simulations or determine whether a system is in thermal equilibrium. These tasks are left up to the user. As with any post-trajectory code, 2PT operates on the GIGO philosophy, so users are urged to check carefully the output logs to ensure their calculations gives reasonable results.

## Chapter 3.

### Input & Output Files

This section describes the various input and output files you will encounter to run a 2PT program. The control (.ctl) and group definition (.grp) files are the main drivers of the 2PT program. The output consists of a thermodynamics file (xxx.thermo), velocity autocorrelation file (xxx.vac), power spectrum file (xxx.pwr), degrees of freedom file (xxx.3n) and a log file (xxx.out.log). Each is detailed below

#### 3.a. The Control file (.ctl)

```
#structure input options
IN_AMBERPRMTP spcewbox.prmtp
IN_GROUPFILE spcewbox.2pt.grps
#2pt options
ANALYSIS_FRAME_INITIAL 1
ANALYSIS_FRAME_FINAL 0
ANALYSIS_FRAME_STEP 1
ANALYSIS_VAC_CORLENGTH 0.5
ANALYSIS_VAC_MEMORYMB 10500
ANALYSIS_MOLECULE_FLAG 1
ANALYSIS_OUT spcewbox.amber.prmtp
#md options
MD_TSTEP 0.002
MOL_LINEAR_FLAG 0
MOL_ROTN_SYMMETRY 2
MD_AVGTEMPERATURE 300.217
MD_AVGENERGY -1706.22
#trajectory options
IN_AMBERTRJ spcewbox.2pt.mdvel
IN_AMBER_COORD_FILE spcewbox.2pt.mdcrd
TRAJ_DUMPFREQ 2
```

Figure 2: Example of 2PT control file

#### 3.a.i. Specifying the input structure

##### \*Required

Structures can be specified as either a LAMMPS data/topology, MSI BGF or AMBER .prmtp topology files. As mentioned before, using OpenBabel, one can convert numerous other formats (PDB, MOL2 etc) into BGF format for use with the 2PT program.

KEYWORDS:

- IN\_LMPDATA – LAMMPS data file
- IN\_BGF – MSI BGF format
- IN\_AMBERPRMTOPTOP – AMBER .prmtop file
- IN\_GROUPFILE – Group file for heterogeneous systems

### 3.a.ii. Specifying the input trajectory

#### \*Required

The 2PT program expects atomic velocities that correspond to the input structure. If the system is molecular, then atomic coordinates (saved at the same frequency as the velocities) can also be specified.

#### 3.a.ii.A. CERIUS2/MPSIM/Polygraf/Material Studios Trajectories

KEYWORD: IN\_C2TRJ

NOTES: The code can read either binary or ASCII trajectories. When saving the trajectory, save both the atomic coordinates and velocities in the same file.

Trajectory format:

xx

#### 3.a.ii.B. AMBER

KEYWORD: IN\_AMBERVEL

NOTES: If the molecular option is specified, then an atomic coordinate file saved at the same frequency as the input file is required. This is specified with the IN\_AMBERCOORD keyword, i.e.,

IN\_AMBERVEL xxx.mdvel

IN\_AMBERCOORD xxx.mdcrd

Trajectory/velocity format:

atom\_1x atom\_1y atom\_1z atom\_2x atom\_2y atom\_2z atom\_3x atom\_3y atom\_3z atom\_4x  
atom\_4y atom\_4z atom\_5x atom\_5y atom\_5z atom\_6x atom\_6y atom\_6z atom\_7x atom\_7y

...

atom\_nx atom\_ny atom\_nz

[box: a b c alpha beta gamma] (optional)

#### 3.a.ii.C. CHARMM (experimental)

KEYWORD: IN\_CHARMMVEL

NOTES: If the molecular option is specified, then an atomic coordinate file is required. This is specified with the IN\_CHARMMCOORD keyword, i.e.,

IN\_CHARMMVEL xxx.vel

IN\_CHARMMCOORD xxx.trj

Trajectory/velocity format:

xx xx

#### 3.a.ii.D. XYZ

KEYWORD: IN\_XYZVEL

NOTES: If the molecular option is specified, then an atomic coordinate file, saved at the same frequency as the input file is required. This is specified with the *IN\_XYZ\_COORD\_FILE* keyword, i.e.,

*IN\_XYZTRJ xxx.vel*

*IN\_XYZ\_COORD\_FILE xxx.trj*

Trajectory/velocity xyz format:

#number of atom

STEP xxx

atom\_sym\_1 x y z

atom\_sym\_2 x y z

....

atom\_sym\_n x y z

Additional information needed to be included:

MD\_AVGENERGY

MD\_AVGVOLUME

MD\_AVGTEMPERATURE

IN\_BGF This should be from the first snapshot of the MD. Can be obtained by using the obabel START.xyz -O xxx.bgf

TRAJ\_DUMPFREQ

Using VASP output:

Obtaining xyz coordinate and velocity format from XDATCAR can be done by running the script xdat2xyz.pl and VASP\_XDATCAR2XYZVEL.py respectively.

### 3.a.ii.E. LAMMPS

KEYWORD: *IN\_LMPTRJ*

NOTES: The LAMMPS trajectory format allows tremendous flexibility. We suggest trajectories that are saved using the following options:

dump atomtrj all custom 4 xxx.lammps id type vx vy vz or

dump atomtrj all custom 4 xxx.lammps id type xu yu zu vx vy vz for molecular systems.

The atoms coordinates should be saved in absolute coordinates (not reimaged into the unit cell and not fractional coordinates).

Trajectory/velocity format:

ITEM: TIMESTEP

x

ITEM: TOTAL ATOMS

n

ITEM: BOX BOUNDS

xx xx xlo xhi

yy yy ylo yhi

zz zz zlo zhi

ITEM: ATOMS id type xu yu zu vx vy vz

atom\_1 type\_1 x y z vx vy vz

atom\_2 type\_2 x y z vx vy vz

...

atom\_n type\_n x y z vx vy vz

### 3.a.iii. Specifying the output file

\*Optional

**KEYWORD:** OUT\_PREFIX xxx

This will create 5 output files:

1. xxx.3n – the number of degrees of freedom as a function of frequency
2. xxx.vac – the velocity autocorrelation file used to obtain the power spectrum
3. xxx.pwr – the power spectrum/spectral velocity used to obtain the thermodynamics
4. xxx.thermo – contains the thermodynamic output from the 2PT program
5. xxx.log – screen dump

**Accepted input:** any string

**Default:** 2pt\_results

### 3.a.iv. Analysis options

\*Optional

#### 3.a.iv.A. Frame selection

1. **ANALYSIS\_FRAME\_INITIAL:** start frame.

**Accepted input:** any integer. Start frame must be less than final frame unless final frame is 0

**Default** 1

2. **ANALYSIS\_FRAME\_FINAL:** final frame. Set to 0 to use the last frame found in the trajectory.

**Accepted input:** any integer.

**Default** 0.

3. **ANALYSIS\_FRAME\_STEP:** frame increment. Will select from FRAME\_INITIAL to FRAME\_FINAL every FRAME\_STEP.

**Accepted input:** any integer

**Default** 1.

#### 3.a.iv.B. MD\_TSTEP

The simulation timestep in ps.

**Accepted input:** any decimal value

**Default** 0.001

**Units:** picoseconds (ps)

#### 3.a.iv.C. ANALYSIS\_VAC\_CORLENGTH

The correlation length when computing the velocity autocorrelation function.

**Accepted input:** any decimal value

**Default:** 0.5

**Units:** none

#### 3.a.iv.D. TRAJ\_DUMPFREQ

How often (steps) during the MD the trajectory file is saved.

**Accepted input:** any positive integer

**Default:** 1

**Units:** steps

### 3.a.v. ANALYSIS\_VACMEMORYMB

\* **Optional**

The amount of memory (in megabytes) to allocate for the 2PT program. In order to calculate the autocorrelation function. The 2PT program is memory intensive due to the calculation of the atomic velocity autocorrelation functions (VAC). This value should be set as high as is physically possible, as 2PT will determine the amount of memory needed to store the velocities of an atom/molecule over the entire trajectory. If the total required exceeds the allocation, the trajectory will be rescanned until the VAC of all atoms/molecules is obtained.

**Accepted input:** Any number – must be smaller or equal to the available system memory or a memory violation may result

**Default:** 500

**Units:** Megabytes

### 3.a.vi. Molecule and degrees of freedom

\* **Optional**

#### 3.a.vi.A. ANALYSIS\_MOLECULE\_FLAG

Flag specifying whether system is mono-atomic fluid (e.g. LJ liquid) or molecular. If molecular, then the librational (non-diffusive) modes of the system will be analyzed in terms of its rotational, solid-like translational and internal vibrational components.

**Accepted input:** 1 or 0

**Default:** 0

**Units:** none

#### 3.a.vi.B. MD\_FIXED\_DF

The number of the 3N degrees of freedom (DOF) which are removed/constrained during dynamics. This is usually the results of using SHAKE/RATTLE constrained in biomolecular simulations. For example, if running simulations with the rigid SPC/TIPx water models, this will be 3 degrees of freedom removed for each water molecule (2 frozen hydrogens and 1 frozen HOH angle). For typical AMBER and CHARMM simulations, the motions of the hydrogens are typically frozen. Thus the total number of frozen hydrogens, bond angles should be recorded here. If incorrectly specified, the code will issue a warning.

Instead of an integer, if the letter g is specified, then the constrained degrees of freedom for each group is read from the bottom of the group file.

**Accepted input:** any integer or g

**Default:** 0

**Units:** none

### 3.a.vi.C. MOL\_ROT\_N\_SYMMETRY

This specifies the external symmetry number  $\sigma$  for molecules. If you are familiar enough with group theory to identify the molecule's point group, then  $\sigma$  can be determined from Table I. Without identifying the point group, one can count manually the number of orientations of the rigid molecule that interchange only identical atoms.

Group	$\sigma$	Group	$\Sigma$	Group	$\sigma$	Group	$\sigma$
$C_1, C_i, C_s,$ $C_{\infty v}$	1	$D_{\infty h}$	2	$T, T_d$	12	$O_h$	24
$C_n, C_{nv},$ $C_{nh}$	n	$D_n, D_{nh},$ $D_{nd}$	2n	$S_n$	n/2	$I_h$	60

Table 1. External symmetry numbers corresponding to symmetry point groups

For example, the benzene molecule ( $C_6H_6$ ) belongs to the  $D_{6h}$  point group. From Table I,  $\sigma = 12$ . Alternatively, one can draw the molecule as a hexagon with numbered vertices. Rotating the drawing by  $n \times 60^\circ$ , where n runs from 0 to 5, generates six different orientations that are distinguished only by the artificial numbering of the vertices. Each of these six orientations can be flipped over to generate another orientation, for a total of 12 unique orientations,  $\sigma = 12$ . Instead of an integer, if the letter g is specified, then the external symmetry number for every molecule in a particular group is read from the bottom of the group file. If the group file is specified by the g option is not activated, the code assumes that each molecule has the same rotational symmetry.

**Accepted input:** any integer or g

**Default:** 0

**Units:** none

### 3.a.vi.D. MOL\_LINEAR\_FLAG

Linear molecule flag. Linear molecules (e.g. rigid  $CO_2$ ) are described by a single rotational constant, B, and a single moment of inertia, I. Hence a special rotational partition function has to be used in these cases. Details may be found in textbooks of molecular spectroscopy. Instead of an integer, if the letter g is specified, then the linear molecule flag for every molecule in a particular group is read from the bottom of the group file. If a group file is specified but the g option is not present, the code assumes that each molecule has the specified linear molecule flag.

**Accepted input:** 0,1 or g

**Default:** 0

**Units:** none

## 3.a.vii. Specifying thermodynamic inputs

### 3.a.vii.A. MD\_AVGVOLUME

\*Required if not present in trajectory



This is the average cell volume ( $\text{\AA}^3$ ) during the entire 2PT trajectory. Certain trajectories (such as LAMMPS and CERIUS2), stores this information in the trajectory file and will be determined dynamically. Otherwise this is required.

**Accepted input:** any decimal

**Units:** cubic angstroms ( $\text{\AA}^3$ )

### 3.a.vii.B. MD\_AVGENERGY

**\*Required**

This is the average strain energy (KE+PE in kJ/mol) of the system during the entire 2PT trajectory. This is required in order to calculate the internal energy. Note, that this is the strain energy ( $E - E^0$ ) and so the 0K energy of the system should be subtracted.

If specified as g, then the average total energy of each group is read from the bottom of the group file. If a group file is included, but the g option is not specified, then the total energy of the system will be distributed evenly to each degree of freedom.

**Accepted input:** any decimal or g

**Units:** kilo-Joules per mole (kJ/mol)

### 3.a.vii.C. MD\_AVGPRESSURE

**\*Optional**

The average system pressures (GPa) during the entire 2PT trajectory. This option will be used in a future version of the program.

**Accepted input:** any decimal

**Units:** Giga-pascal (GPa)

## 3.b. The Group definition file

This sections describes the format of the group definition file. This is used to partitions a heterogeneous system into logical components (eg. DNA, Ions and water for a solvated DNA structure, or water and ethanol for a water – ethanol mixture). Although not strictly enforced, it is recommended that each group consists of only 1 type of molecule, as some of the options are molecule specific for molecular systems. Additionally, we recommend a group file for every all heterogeneous systems.

```

Total Groups: 3
Group 1 Atoms 23968
2 - 6 8 9 11 12 14 1517 18 20 21 23 24 26 27 29 3032
33 35 36 38 39 41 42 44 4547 48 50 51 53 54 56 57 59
...
Group 2 Atoms 23
2352 - 2360 3982 33421
Group 3 Atoms 39982
1 3942 4984 983 3984 9233 4534 232 556 672 892 7799
...
Constraints
1100 0 39982
LinearMoleculeFlag
0 0 0
RotationalSymmetryNumber
1 1 2
GroupEnergy
-238327 -382 -8712837

```

Figure 3: Example of 2PT group file

### 3.b.i. Specifying atoms

The first line of a group file must be:

Total Group: x

The heading for each subsequent group should be

Group x Atoms y

This is followed by a list of atoms ids. Ranges can also be specified (i.e. xx – yy).

**Note:** An extra group containing the entire system is always created by default and so need not be specified.

### 3.b.ii. Specifying Constraints

The heading for the constraints line must be:

Constraints:

Followed by a list of constraints for each of the groups specified.

**Note:** This option is only activated if the MD\_FIXED\_DF flag in the control file is set to g.

### 3.b.iii. Specifying Linear molecules

The heading for the linear molecule line must be:

LinearMoleculeFlag:

Followed by list of 1 or 0 for the molecules in each group. Note that the linear molecular flag is applied to each molecule in that particular group, hence it is incorrect to use this if the more than one type of molecule is specified for the group. In that case, creating a separate group for each type of molecule is advised.

**Note:** This option is only activated if the MD\_LINEAR\_FLAG in the control file is set to g.

### 3.b.iv. Specifying external rotational numbers

The heading for the rotational symmetry number lines must be:

RotationalSymmetryNumber:

Followed by a line of integers specifying the rotational symmetry number of each molecule in that particular group.

Note: This option is only activated if the MOL\_ROT\_N\_SYMMETRY flag in the control file is set to g.

### 3.b.v. Specifying group energies

The heading for the group energy lines must be

GroupEnergy:

Followed by energies for each of the group specified.

Note: This option is only activated if the MD\_AVGENERGY flag in the control file is set to g.

## 3.c. The Output files

### 3.c.i. Thermodynamic output file (xxx.thermo)

This is the main output file from the program. It contains all the calculated system thermodynamics, separated by groups if the group file is specified. The various options are listed in table 2.

Calculation of Thermodynamic Properties				
property	Trns[G001]	Rot[G001]	Ivib[G001]	Tot[G001]
nmolecules	216	216	216	216
natom	648	648	648	648
dof	646.00	646.00	644.86	1936.86
temperature____(K)	272.63	273.34	356.07	300.68
pressure____(GPa)	0.00	0.00	0.00	0.00
ZPE_(kJ/mol/SimBox)	564.57	2278.51	13715.01	16558.09
Emd_(kJ/mol/SimBox)	-2379.61	-2379.61	-2379.61	-7138.82
Eq_(kJ/mol/SimBox)	-2262.41	-1281.94	9427.06	5882.71
Sq_(J/mol_K/SimBox)	8946.98	1587.90	2.99	10537.87
Aq_(kJ/mol/SimBox)	-4701.66	-1715.98	9426.00	3008.36
Cvq(J/mol_K/SimBox)	4674.95	2554.43	13.71	7243.09
S(0)(cm/mol/SimBox)	0.42	0.04	0.00	0.46
Diffus(cm <sup>2</sup> /s, _1/s)	6.8550e-06	6.2104e-07	1.0794e-09	8.2443e-06
fluidicity_____	0.12273107	0.03234645	0.00000000	0.00000000

Figure 4: Example of 2PT thermodynamics output file

<b>[Gxxx]</b>	Group xxx	Always shown
<b>Trns</b>	Translations	Only for molecular systems
<b>Rot</b>	Rotations	Only for molecular systems
<b>Ivib</b>	Internal vibrations	Only for molecular systems
<b>Tot</b>	Sum of all components	
<b>dof</b>	Degrees of freedom	
<b>ZPE</b>	Zero point energy	
<b>Emd</b>	Total system energy from MD	
<b>Eq</b>	Quantum internal energy	$E_q = E_{md} + ZPE + \text{heat capacity corrections}$
<b>Sq</b>	Quantum entropy	
<b>Aq</b>	Quantum Helmholtz free energy	$A_q = E_q - TS_q$
<b>Cvq</b>	Quantum heat capacity	
<b>S(0)</b>	Value of DoS at $\nu = 0$ (zero frequency)	
<b>Diffuse</b>	Diffusion constant	
<b>fluidicity</b>	% of 3N modes in the gas phase	

Table 2: Description of 2PT thermo file

### 3.c.ii. Velocity autocorrelation file (xxx.vac)

This file contains the velocity autocorrelation file, separated by groups if the group file is specified. The magnitude of the VAC as a function of time is reported and can be plotted to ensure that it converges to zero within the timespan of the 2PT calculation.

### 3.c.iii. Power spectrum file (xxx.pwr)

This file contains the density of states of the system, i.e. the population and energy (cm<sup>-1</sup>) of normal modes in the system.

### 3.c.iv. Degree of freedom file (xxx.3n)

This file contains the 3n degrees of freedom as a function of frequency, separated by group if the group file is specified.

### 3.c.v. Screen dump file (xxx.log)

Contains all of the previous files, as well as various diagnostic messages if the debug option is activated in the source.

## Chapter 4. Theory

There have been growing interests in the evaluation of absolute entropy of water and several methods are proposed to determine the entropy of water from the results of molecular dynamic simulations. For example, White and Meirovitch (White and Meirovitch 2004; White and Meirovitch 2004) proposed a hypothetical scanning method which determines the absolute entropy and free energy from the Boltzmann probability distribution. Lizaridis and Karplus (Lazaridis and Karplus 1996) showed that the entropy can be obtained from truncated expansion of molecular pair correlation functions. (Wang, Abel et al. 2009) More recently, Sharma et al. (Sharma, Agarwal et al. 2008) report the accuracy of entropy determined from the two-body pair correlation function determined from atom-atom radial distribution functions. Tyka et al. (Tyka, Sessions et al. 2007) proposed a confinement method that determines the absolute entropy using thermodynamic integration from a hypothetical harmonic state to the liquid state. Henschman (Henschman 2007) demonstrated estimation of entropy from cell theory, in which the entropy is calculated within the harmonic approximation in the potential surface.

These methods have been applied to water, but only under limited conditions (e.g. at 1 atm and 25°C). Furthermore, the harmonic approximations implicit in many of these methods make the results suspect, since anharmonic effects are known to be important in diffusive systems. (Andricioaei and Karplus 2001; Lin, Blanco et al. 2003) It is thus questionable whether these methods would be applicable to study the properties of water under various physical and chemical environments around biological molecules. (Zielkiewicz 2008)

### 5.a. The density of state function

The density of state function,  $S(\nu)$ , is defined as the mass weighted sum of the atomic spectral densities (Berens, Mackay et al. 1983; Lin, Blanco et al. 2003)

$$S(\nu) = \frac{2}{kT} \sum_{j=1}^N \sum_{k=1}^3 m_j s_j^k(\nu) \quad (1)$$

where  $m_j$  is the mass of atom  $j$ , and  $N$  is total number of atoms of the system. The spectral density  $s_j^k(\nu)$  of atom  $j$  in the  $k$  direction ( $k=x, y$ , and  $z$  in the Cartesian coordinate) is determined from the square of the Fourier transform of the velocities as

$$s_j^k(\nu) = \lim_{\tau \rightarrow \infty} \frac{1}{2\tau} \left| \int_{-\tau}^{\tau} v_j^k(t) e^{-i2\pi\nu t} dt \right|^2 \quad (2)$$

where  $v_j^k(t)$  is the  $k$ -component of velocity vector of atom  $j$  at time  $t$ . The atom spectral density and the density of state function may also be obtained from the Fourier transform of the velocity autocorrelation function. (Berens, Mackay et al. 1983; Lin, Blanco et al. 2003) The physical significance of  $S(\nu)$  is that it represents the density of normal modes of the system at frequency  $\nu$

$$S(\nu) = \sum_{i=1}^{3N} [\delta(\nu - \nu_i^n) + \delta(\nu + \nu_i^n)] \quad (3)$$

where  $\nu_i^n$  represents the  $i$ -th of the  $3N$  normal mode frequencies of the system. In other words,  $S(\nu)d\nu$  is the number of modes of a system moving within the frequency range of  $\nu$  to  $\nu+d\nu$ . Therefore, the integration of  $S(\nu)$  gives the total degrees of freedom of the system, i.e.,

$$\int_0^{\infty} S(\nu)d\nu = 3N \quad (4)$$

Furthermore, the intensity of  $S(\nu)$  at zero frequency is associated with the diffusivity  $D$  of the particles (Berens, Mackay et al. 1983; Lin, Blanco et al. 2003)

$$S(0) = \frac{12mND}{kT} \quad (5)$$

### 5.b. Thermodynamic properties of systems undergo harmonic motions

For solid state systems (e.g., crystals), where all the normal modes are essentially harmonic, the canonical partition function,  $Q$ , can be determined as the product of those of harmonic oscillators ( $q_{HO}(\nu)$ ) having the same vibration frequencies, i.e.,

$$\ln Q = \int_0^{\infty} d\nu S(\nu) \ln q_{HO}(\nu) \quad (6)$$

where  $q_{HO}(\nu) = \frac{\exp(-\beta h\nu/2)}{1 - \exp(-\beta h\nu/2)}$  with  $\beta=1/kT$  and  $h$  being the Plank constant. The

thermodynamic properties of such systems can be expressed in terms of the integration of the density of state function weighted by the corresponding property weighting functions (Berens, Mackay et al. 1983; Lin, Blanco et al. 2003)

$$E = E_0 + T\beta^{-1} \left( \frac{\partial \ln Q}{\partial T} \right)_{N,V} = E_0 + \beta^{-1} \int_0^{\infty} d\nu S(\nu) W_E(\nu) \quad (7a)$$

$$S = k \ln Q + \beta^{-1} \left( \frac{\partial \ln Q}{\partial T} \right)_{N,V} = k \int_0^{\infty} d\nu S(\nu) W_S(\nu) \quad (7b)$$

$$A = E_0 - \beta^{-1} \ln Q = V_0 + \beta^{-1} \int_0^{\infty} d\nu S(\nu) W_A(\nu) \quad (7c)$$

where the energy, entropy, and Helmholtz free energy weighting functions are

$$W_E(\nu) = \frac{\beta h\nu}{2} + \frac{\beta h\nu}{\exp(\beta h\nu) - 1} \quad (8a)$$

$$W_S(\nu) = \frac{\beta h\nu}{\exp(\beta h\nu) - 1} - \ln[1 - \exp(-\beta h\nu)] \quad (8b)$$

$$W_A(\nu) = \ln \frac{1 - \exp(\beta h\nu)}{\exp(-\beta h\nu/2)} \quad (8c)$$

The reference energy  $E_0$  is the potential energy of the system at 0 K. For convenience we choose to subtract the kinetic energy from the total energy (e.g.  $E^{MD}$  in a molecular dynamic simulation) as

$$E_0 = E^{MD} - \beta^{-1}3N \quad (9)$$

Equation (9) ensures that the total energy  $E$  (eqn. 7a) determined from a classical MD simulation is the same as the sum of energies of classical harmonic oscillators ( $q_{HO}^c(\nu) = \frac{1}{\omega \nu}$ ) at standing still. (Lin, Blanco et al. 2003)

The set of equations presented in this section are also referred to as the one-phase thermodynamic (1PT) model.

### 5.c. Thermodynamic properties of Mono-atomic fluids

The DoS of mono-atomic gases and liquids represents the translational motions of the molecules. The low frequencies modes (diffusive motions and libration), which are highly anharmonic, dominate the properties of the system; therefore, eqns 7 to 9 no longer valid. Lin et al. resolved this problem by treating the density of states of a fluidic system as the sum of a gas-like ( $S^g(\nu)$ ) and a solid-like ( $S^s(\nu)$ ) contribution (Lin, Blanco et al. 2003)

$$S(\nu) = S^g(\nu) + S^s(\nu) \quad (10)$$

where  $S^g(\nu)$  accounts for all the diffusive component (i.e.,  $S(\nu=0) = S^g(\nu=0) = s_{trn}^0$ ) and the solid component is non-diffusive ( $S^s(0)=0$ ). The density of states of the gas-like component is assumed to be that of hard spheres (Lin, Blanco et al. 2003)

$$S^g(\nu) = \frac{s_{trn}^0}{1 + \left[ \frac{\pi s_{trn}^0 \nu}{6 f_{trn} N} \right]^2} \quad (11)$$

where  $s_{trn}^0 = S(0)$  is the intensity of the density of state of the real system at zero frequency. The variable  $f_{trn}$  indicates the translational fluidicity of the system (total degrees of freedom of the gas-like component is  $3Nf_{trn}$ ). Its value is determined from the following universal function, (Lin, Blanco et al. 2003)

$$2\Delta^{-9/2} f^{15/2} - 6\Delta^{-3} f^5 - \Delta^{-3/2} f^{7/2} + 6\Delta^{-3/2} f^{5/2} + 2f - 2 = 0 \quad (12)$$

where  $f=f_{trn}$  for the translational contributions, and the dimensionless diffusivity constant  $\Delta$  is a function of material properties

$$\Delta(T, V, N, m, s^0) = \frac{2s^0}{9N} \left( \frac{\pi kT}{m} \right)^{1/2} \left( \frac{N}{V} \right)^{1/3} \left( \frac{6}{\pi} \right)^{2/3} \quad (13)$$

where  $V$  is the volume of the system. [Note that  $f=f_{trn}$  (in eq. 12) and  $s^0=s_{trn}^0$  (in eq. 13) for translational motions.] Therefore, the thermodynamic state of the system uniquely determines the value of  $\Delta$  (eqn. 13) and thus the fluidicity  $f_{trn}$  (eqn. 12) and  $S^g(\nu)$ .

With this 2PT decomposition of the density of states, the thermodynamic properties of the system are determined as the sum of the gas and solid contributions

$$E = E_0 + \beta^{-1} \left[ \int_0^{\infty} d\nu S^s(\nu) W_E^s(\nu) + \int_0^{\infty} d\nu S^g(\nu) W_E^g(\nu) \right] \quad (14a)$$

$$S = k \left[ \int_0^{\infty} d\nu S^s(\nu) W_S^s(\nu) + \int_0^{\infty} d\nu S^g(\nu) W_S^g(\nu) \right] \quad (14b)$$

$$A = E_0 + \beta^{-1} \left[ \int_0^{\infty} d\nu S^s(\nu) W_A^s(\nu) + \int_0^{\infty} d\nu S^g(\nu) W_A^g(\nu) \right] \quad (14c)$$

where the weighting functions for the solid-like component ( $W_E^s(\nu)$ ,  $W_S^s(\nu)$ , and  $W_A^s(\nu)$ ) are the same as those in eqn. 8, and those for the gas-like component are

$$W_E^g(\nu) = W_E^{HS}(\nu) = 0.5 \quad (15a)$$

$$W_S^g(\nu) = W_S^{HS}(\nu) = \frac{1}{3} \frac{S^{HS}}{k} \quad (15b)$$

$$W_A^g(\nu) = W_A^{HS}(\nu) = W_E^{HS}(\nu) - W_S^{HS}(\nu) \quad (15c)$$

where  $S^{HS}$  is the hard sphere entropy

$$\frac{S^{HS}(f_{im}N, V, T)}{k} = \frac{5}{2} + \ln \left[ \left( \frac{2\pi mkT}{h^2} \right)^{3/2} \frac{V}{f_{im}N} z(y) \right] + \frac{y(3y-4)}{(1-y)^2} \quad (16)$$

and  $y = \frac{f_{im}^{5/2}}{\Delta^{3/2}}$ , and  $z(y)$  is the compressibility factor from the Carnahan-Starling equation of state of hard sphere gases

$$z(y) = \frac{1 + y + y^2 - y^3}{(1-y)^3} \quad (17)$$

The reference energy now becomes

$$E_0 = E^{MD} - \beta^{-1} 3N(1 - 0.5f_{im}) \quad (18)$$

Equations 10 to 18 constitute the 2PT method for obtaining thermodynamic properties of mono-atomic fluids (e.g., argons) based on the density of state distribution. (Lin, Blanco et al. 2003; Li, Lin et al. 2004; Lin, Maiti et al. 2005; Jana, Pal et al. 2006)

### 5.d. Thermodynamic properties of polyatomic fluids

For polyatomic species (e.g., water), it is possible to decompose the total density of state functions into contributions from molecular translation ( $S_{trn}(\nu)$ ), rotation ( $S_{rot}(\nu)$ ), and vibration ( $S_{vib}(\nu)$ )

$$S(\nu) = S_{trn}(\nu) + S_{rot}(\nu) + S_{vib}(\nu) \quad (19)$$



where the  $S_{trn}(\nu)$  is determined from the center of mass velocities of all the molecules, and  $S_{vib}(\nu)$  is determined from intramolecular vibration velocities. The rotational density of state is determined from the angular velocity (analogous to eqn. 1 and 2)

$$S_{rot}(\nu) = \beta \sum_{l=1}^M \sum_{k=1}^3 \lim_{\tau \rightarrow \infty} \frac{I_l^k}{\tau} \left| \int_{-\tau}^{\tau} \omega_l^k(t) e^{-i2\pi\nu t} dt \right|^2 \quad (20)$$

where  $I_l^k$  is the  $k$ -th principle moment of inertia of molecule  $l$ ,  $\omega_l^k$  is the angular velocity along the  $k$  principle axis, and  $M$  is total number of molecules in the system.

In analogy to the gas-solid decomposition for the translational spectrum (eqn. 10), the anharmonic effect in rotational motions can be treated by decomposing the rotational density of state into a gas and a solid-like component

$$S_{rot}(\nu) = S_{rot}^g(\nu) + S_{rot}^s(\nu) \quad (21)$$

with the gas-component being determined as

$$S_{rot}^g(\nu) = \frac{s_{rot}^0}{1 + \left[ \frac{\pi s_{rot}^0 \nu}{6 f_{rot} N} \right]^2} \quad (22)$$

with  $s_{rot}^0 = S_{rot}(\nu=0)$  so that all the rotational diffusional behavior is considered in the gas-component, and the rotational fluidicity  $f_{rot}$  is determined from eqns. 12 and 13 with  $s^0 = s_{rot}^0$ .

Therefore, the value of  $S_{rot}(0)$  and  $f_{rot}$  uniquely determines  $S_{rot}^g(\nu)$  and thus  $S_{rot}^s(\nu)$ .

The thermodynamic properties of molecular fluids are determined as sum of contributions from the corresponding translation, rotation, and vibration motions

$$E = E_0 + E_{tm} + E_{rot} + E_{vib} \quad (23a)$$

$$S = S_{tm} + S_{rot} + S_{vib} \quad (23b)$$

$$A = E_0 + A_{tm} + A_{rot} + A_{vib} \quad (23c)$$

and the properties associated with each type of motions determined from the corresponding density of state functions

$$E_m = \beta^{-1} \left[ \int_0^{\infty} d\nu S_m^s(\nu) W_{m,E}^s(\nu) + \int_0^{\infty} d\nu S_m^g(\nu) W_{m,E}^g(\nu) \right] \quad (24a)$$

$$S_m = k \left[ \int_0^{\infty} d\nu S_m^s(\nu) W_{m,S}^s(\nu) + \int_0^{\infty} d\nu S_m^g(\nu) W_{m,S}^g(\nu) \right] \quad (24b)$$

$$A_m = \beta^{-1} \left[ \int_0^{\infty} d\nu S_m^s(\nu) W_{m,A}^s(\nu) + \int_0^{\infty} d\nu S_m^g(\nu) W_{m,A}^g(\nu) \right] \quad (24c)$$

where  $m=trn$  (center of mass translation),  $rot$  (rotation), or  $vib$  (intramolecular vibration). For vibration motions ( $m=vib$ ), the gas component is always zero,  $S_{vib}^g(\nu)=0$ , i.e., all vibrational motions are taken as harmonic,  $S_{vib}(\nu) = S_{vib}^s(\nu)$ .

The harmonic weighting functions (eqn. 8) are used for all solid-components ( $W_{m,E}^s(\nu)$ ,  $W_{m,S}^s(\nu)$ , and  $W_{m,A}^s(\nu)$ ). The hard sphere weighting functions (eqn. 15) are used for the gas-like component of the translational motions ( $W_{tm,E}^g(\nu)$ ,  $W_{tm,S}^g(\nu)$ , and  $W_{tm,A}^g(\nu)$ ). The ideal gas weighting functions are used for the gas-like component of rotational motions

$$W_{rot,E}^g(\nu) = 0.5 \quad (25a)$$

$$W_{rot,S}^g(\nu) = \frac{1}{3} \frac{S^R}{k} \quad (25b)$$

$$W_{rot,A}^g(\nu) = W_{rot,E}^g(\nu) - W_{rot,S}^g(\nu) \quad (25c)$$

where  $\frac{S^R}{k} = \ln \left[ \frac{\pi^{1/2} e^{3/2}}{\sigma} \left( \frac{T^3}{\Theta_A \Theta_B \Theta_C} \right)^{1/2} \right]$  is the rotational entropy of a rigid body with rotational

temperatures  $\Theta_A = \frac{h^2}{8\pi^2 I_A k}$  and  $\sigma$  is the rotational symmetry. The reference potential energy is taken to be

$$E_0 = E^{MD} - \beta^{-1} 3N(1 - 0.5f_{tm} - 0.5f_{rot}) \quad (26)$$

It should be noted that the 2PT method provides the correct thermodynamic properties in the limits of (1) crystalline solid where the molecular diffusivity is zero (and hence  $f_{trn}=f_{rot}=0$ , and the system possesses only solid-like component), and (2) ideal gas where the fluidicity  $f_{trn}$  and  $f_{rot}$  are unity.

## References & Notes

- Andricioaei, I. and M. Karplus (2001). "On the calculation of entropy from covariance matrices of the atomic fluctuations." Journal of Chemical Physics **115**(14): 6289-6292.
- Berens, P. H., D. H. J. Mackay, et al. (1983). "Thermodynamics and Quantum Corrections from Molecular-Dynamics for Liquid Water." Journal of Chemical Physics **79**(5): 2375-2389.
- Henchman, R. H. (2007). "Free energy of liquid water from a computer simulation via cell theory." Journal of Chemical Physics **126**(6).
- Jana, B., S. Pal, et al. (2006). "Entropy of water in the hydration layer of major and minor grooves of DNA." Journal of Physical Chemistry B **110**(39): 19611-19618.
- Lazaridis, T. and M. Karplus (1996). "Orientational correlations and entropy in liquid water." Journal Of Chemical Physics **105**(10): 4294-4316.
- Li, Y. Y., S. T. Lin, et al. (2004). "Efficiency of various lattices from hard ball to soft ball: Theoretical study of thermodynamic properties of dendrimer liquid crystal from atomistic simulation." Journal of the American Chemical Society **126**(6): 1872-1885.
- Lin, S. T., M. Blanco, et al. (2003). "The two-phase model for calculating thermodynamic properties of liquids from molecular dynamics: Validation for the phase diagram of Lennard-Jones fluids." Journal of Chemical Physics **119**(22): 11792-11805.
- Lin, S. T., P. K. Maiti, et al. (2005). "Dynamics and thermodynamics of water in PAMAM dendrimers at subnanosecond time scales." Journal of Physical Chemistry B **109**(18): 8663-8672.
- Sharma, R., M. Agarwal, et al. (2008). "Estimating the entropy of liquids from atom-atom radial distribution functions: silica, beryllium fluoride and water." Molecular Physics **106**(15): 1925-1938.
- Tyka, M. D., R. B. Sessions, et al. (2007). "Absolute free-energy calculations of liquids using a harmonic reference state." Journal Of Physical Chemistry B **111**(32): 9571-9580.
- Wang, L., R. Abel, et al. (2009). "Thermodynamic Properties of Liquid Water: An Application of a Nonparametric Approach to Computing the Entropy of a Neat Fluid." Journal of Chemical Theory and Computation **5**(6): 1462-1473.
- White, R. P. and H. Meirovitch (2004). "Lower and upper bounds for the absolute free energy by the hypothetical scanning Monte Carlo method: Application to liquid argon and water." Journal Of Chemical Physics **121**(22): 10889-10904.
- White, R. P. and H. Meirovitch (2004). "A simulation method for calculating the absolute entropy and free energy of fluids: Application to liquid argon and water." Proceedings of the National Academy of Sciences of the United States of America **101**(25): 9235-9240.
- Zielkiewicz, J. (2008). "Entropy of water calculated from harmonic approximation: Estimation of the accuracy of method." Journal Of Chemical Physics **128**(19).