

Shermo

Software Manual

Version 2.6.1

Last update: 2025-Oct-20

website: <http://sobereva.com/soft/Shermo>

Tian Lu

sobereva@sina.com

Beijing Kein Research Center for Natural Sciences (www.keinsci.com)

1 What is Shermo?

Shermo program is a free, general, easy-to-use and flexible code for calculating molecular thermochemistry data based on ideal gas assumption. Although most quantum chemistry programs have their own codes used to calculate thermochemistry data after performing frequency analysis, they are not as powerful and flexible as Shermo, and usually their outputs are inconvenient to read.

Features of Shermo

- Output file of frequency analysis task of various quantum chemistry programs can be used as input file, including Gaussian, ORCA, GAMESS-US, NWChem, CP2K and xtb. Other programs can also utilize Shermo to compute thermochemistry data as long as they can generate the .shm file defined by Shermo.

- All common thermochemistry quantities can be calculated, including internal energy (U), enthalpy (H), entropy (S), constant volume heat capacity (C_V), constant pressure heat capacity (C_P) and partition function (q).

- Contributions of translation, rotation, vibration, electron to various thermochemistry properties are outputted in a very compact and clear format. Contribution of every vibrational mode can also be printed to understand their influences on the calculated properties.

- Temperature and pressure can be easily scanned.

- Conformation/configuration weights and weighted thermodynamic data can be obtained.

- Thermochemistry properties can be calculated based on the standard rigid-rotor harmonic oscillator (RRHO) model. While in order to much better deal with low frequencies, two kinds of quasi-RRHO treatments can also be used: (1) Raising lower frequencies to a specific value (2) Interpolation between harmonic oscillator and free-rotor approximations (for entropy part, and can also be for internal energy part, corresponding to Grimme's and Minenkov's models, respectively).

- Imaginary frequencies are allowed to be treated as real frequencies.

- Frequency scale factors for ZPE, heating contribution to internal energy, entropy and heat capacity can be individually specified.

- Point group and rotational symmetry number can be automatically assigned.

- Variation of Gibbs free energy due to concentration change from current state to specific state can be taken into account.

- Shermo can not only run in interactive mode but also run in command-line mode, thus it can be easily incorporated into shell script for batch processing.

- Very easy to use. The program can be used without installation, and users do not need to prepare any running environment like Python.

Citation of Shermo

If Shermo is employed in your study, currently **please cite** it in your paper as follows:

Tian Lu, Qinxue Chen, Shermo: A general code for calculating molecular thermodynamic properties, *Comput. Theor. Chem.*, **1200**, 113249 (2021) DOI: 10.1016/j.comptc.2021.113249

Seeking for help

If you encountered any problem in using Shermo, please feel free to contact me via E-mail (sobereva@sina.com). You can also post your question on quantum chemistry board of computational chemistry commune forum (<http://bbs.keinsci.com/forum-103-1.html>), I will always timely reply to you.

2 Usage

This section introduces how to use Sherma. If you do not have basic knowledge about calculation of molecular thermochemistry data, it is strongly suggested to check Appendix of this manual to gain related knowledge first.

2.1 Files in Sherma package

- *Sherma.exe*: Executable file of Sherma in Windows
- *Sherma*: Executable file of Sherma in Linux
- *settings.ini*: The file containing running parameters of Sherma
- *Sherma_manual.pdf*: Manual
- "example" folder: Containing some output files of various quantum chemistry programs, most of them are involved in the examples in Section 3 of this manual.

2.2 Running Sherma

Sherma can run in both interactive mode and command-line mode.

- Interactive mode

In Windows, just double-clicking the *Sherma.exe* icon to boot up it. In Linux, inputting path of Sherma executable file to boot up it. After that, you should input the path of input file, then all thermodynamic quantities will be immediately outputted on screen.

- Command-line mode

In command-line environment, you can run Sherma like this

```
/sob/Sherma_2.0/Sherma /mol1/freq.out -T 250 -P 1.5
```

where */sob/Sherma_2.0/Sherma* is path of Sherma executable file, */mol1/freq.out* is input file, the arguments request that the calculation is conducted at 250 K and 1.5 atm, see next section for detail.

All running parameters and various information about the present system are shown on screen prior to calculation, you should always carefully examine this information to ensure that the calculated data is what you need.

2.3 Running parameters

After booting up, Sherma will first load running parameters from *settings.ini* in current folder. If it does not exist, Sherma will then try to find this file in the directory defined by "Shermopath" environment. If it still cannot be found, default running parameters will be used. In command-line mode, the parameters in *settings.ini* or the default parameters can be overridden by the arguments in the running command.

The parameters in the *settings.ini* are described below

E: Electronic energy in a.u. If it is 0 (default value), then the electronic energy loaded from input file will be finally used in calculation of thermochemistry data

prt vib: Output contribution of each vibrational mode to various thermodynamic data.

1: Printing on screen

-1: Printing to *vibcontri.txt* in current folder

0 (default): Do not print

T: Temperature (K) used in the calculation. Default value is 298.15. If you want to scan

temperature to check variation of various thermodynamic data with respect to temperature, you should specify lower limit, upper limit and stepsize, for example, *50,200,10*

P: Pressure (atm) used in the calculation. Default value is 1. If you want to scan pressure, you should specify lower limit, upper limit and stepsize, for example, *0.5,15,0.1*

scfZPE: Frequency scale factor for calculating zero-point energy (ZPE). Default is 1.0

scfheat: Frequency scale factor for calculating heating contribution to internal energy (U), that is $U(T) - U(0)$ (the same as the scale factor for $H(T) - H(0)$). Default is 1.0

scfS: Frequency scale factor for calculating entropy (S). Default is 1.0

scfCV: Frequency scale factor for calculating heat capacity (C_V or C_P). Default is 1.0

ilowfreq: Consideration of vibrational contribution of low frequencies to thermodynamic data, see Section 4.A for details

0: Harmonic approximation

1: Raising all lower frequencies to a specific value. All quantities related to vibrational frequencies are affected

2 (default): Grimme's entropy interpolation between harmonic and free-rotor models. This treatment only affects entropy. See *Chem. Eur. J.*, **18**, 9955 (2012)

3: The same as 2, but the interpolation is applied to both entropy and internal energy. In *J. Comput. Chem.*, **44**, 1807 (2023), Otlyotov and Minenkov showed that this scheme is better than 2 and thus more recommended to use

ravib: Raising lower frequencies to this value (cm^{-1}) when *ilowfreq*=1. Default is 100

intpvib: Vibrational frequency threshold (cm^{-1}) used in interpolation when *ilowfreq*=2 and 3. Default is 100, which is a good choice for all cases, but for specific systems, decreasing it to lower value such as 50 may result in better data

imagreal: Treat imaginary frequencies with norm smaller than this value (in cm^{-1}) as real frequencies. Only meaningful if used in combination with *ilowfreq* not equal to 0. Usually 50-100 is suitable. See Section A.4 on when this option may be useful.

imode: Mode of evaluating thermodynamic quantities. 0 (default): Consider all terms, namely translation, rotation, vibration, and electron transition contributions. 1: Ignore translation and rotation, thus suitable for solid systems

conc: When this is not the default 0, Shermo will calculate and print variation of Gibbs free energy ($\Delta G_{\text{conc}} = RT \ln(c_B/c_A)$) due to concentration change from current state (c_A) to specific state (c_B), and finally add the ΔG_{conc} to the reported Gibbs free energy. Note that this setting does not work if scanning temperature and/or pressure is requested.

c_A is calculated as P/RT according to ideal gas formula, where pressure P and temperature T are those specified by corresponding parameters in *settings.ini*.

c_B is specified by "conc", the unit could be M (namely mol/L) or atm; for the latter case c_B will be evaluated according to ideal gas formula according to the pressure and present temperature. Example: *conc= 1.5M* and *conc= 2.3atm*.

outshm: Exporting .shm file after successfully loading output file of quantum chemistry program.

0 (default): Do not export

1: Exporting .shm file, with same name and directory as the input file

PGLabel: Point group. If set to ? (default case), then point group will be identified by Shermo according to current geometry. Examples of directly specifying point groups: Cs, Ci, C1, C3, D5,

C3v, C2h, C3h, D2h, D3d, S4, T, F, Td, O, Oh, I, Ih, Civ, Dih.

defmass: When output file of quantum chemistry program is used as input file, this parameter determines how to assign default atomic masses.

1: Mass of element (*e.g.* Cl is 35.45 amu). This is in line with convention of ORCA and xtb

2: Mass of most abundant isotope (*e.g.* Cl correspond to ³⁵Cl with mass of 34.96885 amu).

This is in line with convention of Gaussian, GAMESS-US and NWChem

3 (default): Directly read masses from output file of quantum chemistry program

modmass: Modification on default atomic masses can be specified below this label. For example, at the end of *settings.ini*, if the content is

```
modmass
10 2.01410
14 13.0033548
```

Then mass of atoms 10 and 14 will be modified to 2.01410 and 13.0033548 amu, respectively.

If .shm file is used as input file, the atomic masses are always directly loaded from the .shm file, and thus "defmass" and "modmass" do not affect result.

It is important to note that atomic mass affects thermodynamic data because in principle it influences molecular mass, moments of inertia and vibrational frequencies. If you use "defmass" and "modmass" to modify mass definition or manually modify .shm file, only the former two will be accordingly changed, while vibrational frequencies are not affected since they are loaded from input file. Therefore, the only rigorous way to consider influence of atomic mass is properly defining isotopes in the input file of your quantum chemistry program.

In command line mode, the above parameters (except for "modmass") can be directly specified by arguments, and the parameters specified in this way have higher priority. For example, you can run Shermo like this:

```
Shermo /Aqours/Yoshiko.out -scfZPE 0.98 -T 233 -ilowfreq 2 -prt vib -1
```

There is another argument "-noset", if it is added, such as

```
Shermo C:\love\nico.out -scfZPE 0.9806 -conc 2M -noset
```

Then Shermo will not load any setting from *settings.ini* but only receive settings from argument, and default settings will be employed for parameters that not explicitly specified.

2.4 Input files

Output file of quantum chemistry program

Output files of four well-known quantum chemistry programs are supported by Shermo as input file. If optimization is also involved, the geometry of the final step will be loaded. Example output files of various program can be found in "example" folder of Shermo package.

- Gaussian: Output file of "freq" or "opt freq" task. Gaussian 09 and 16 are formally supported.
- ORCA: Output file of "freq" or "opt freq" task.
- GAMESS-US: Output file of "RUNTYP=hessian", or "RUNTYP=optimize" with "\$STATPT hssend=.T. \$END"
- NWChem: Output file "TASK xxx freq", or "TASK xxx optimize" with "TASK xxx freq", where xxx is type of theoretical method.
- xtb: The *g98.out* file outputted by vibrational analysis task (--hess or --ohess) of xtb program is supported. However, because this file does not contain electronic energy, if you did not provide the energy by explicitly setting "E" in *settings.ini*, then during running Shermo, Shermo will ask you to input the path of output file of xtb (for example, the *test.out* generated by running *xtb test.xyz -ohess > test.out*), Shermo will try to load "total energy" printed last time in this file (while if you

do not input the path but press ENTER button directly, electronic energy will be regarded as zero).

- CP2K: Output file of "RUN_TYPE VIBRATIONAL_ANALYSIS" task with "PRINT_LEVEL MEDIUM". **IMPORTANT NOTICE:** If the calculated system is a crystal or adsorbate, you should set "imode" in *settings.ini* to 1 to remove translation and rotation contributions, since they should not occur in this situation; in addition, in this case point group will be automatically set to C1, since determining point group is meaningless when rotation contribution is not taken into account.

Shermo loads all necessary information that needed by calculating thermochemistry quantities from the output file, including electronic energy, spin multiplicity, elements, and coordinates of all atoms, as well as harmonic frequencies (therefore, anharmonic frequencies will be ignored even if they are calculated).

The actually used electronic energy (E) can be directly specified by "E" parameter, however if it keeps default value (0), then the E loaded from output file will be used. For Gaussian and ORCA, the loaded E always corresponds to current calculation level, while for GAMESS-US and NWChem, DFT energy will be loaded if the calculation was conducted at DFT level (except for double-hybrid functionals).

For GAMESS-US and NWChem, if the theoretical method you employed is not DFT of normal functionals, the situation is more complicated, because in this case it is difficult to determine the energy of current level from the output file. Briefly speaking, when post-HF or double-hybrid functionals is employed, energy of reference state is loaded by Sherma. Of course this is not what you want, therefore you should simply ignore the sum of E and various thermal corrections printed by Sherma; alternatively, you can let Sherma output .shm file, then manually fill the actual post-HF energy into this file, and then use this file as input file. The situation of double-hybrid functional is similar to post-HF.

.shm file

There is a private format (.shm) defined by Sherma. This file contains everything that needed by Sherma calculation, thus it can also be employed as input file. The shm format is very concise and clear, thus you can easily edit it. An example of shm file is shown below, the blue texts are comments

```
*E Electronic energy (a.u.)
-91.6752090000

*wavenum Wavenumbers (cm-1). Negative value means imaginary frequency
952.5433
952.5433
2541.2397
3915.9410

*atoms Information of all atoms: Name, mass (amu), X, Y, Z (Angstrom)
H 1.007830 0.000000 0.000000 -1.569030
C 12.000000 0.000000 0.000000 -0.498995
N 14.003070 0.000000 0.000000 0.654010

*elevel Relative energy w.r.t. ground state (eV) and level degeneracy
0.000000 1
```

All data in the shm file are in free format. All frequencies, at least the real frequencies, must be presented. Information of all atoms must be given.

The *elevel field is particularly worth noting. The energy levels recorded in this field are with respect to ground state. Spin multiplicity should be equivalently represented by this field. For

example, if a system is triplet, content of this field should be

```
0.0 3
```

or

```
0.0
```

```
0.0
```

```
0.0
```

Note that the default degeneracy is 1. Another example, a singlet molecule has a low-lying doubly degenerate excited states with excitation energy of 0.15 eV, then the *elevel field should be

```
0.0
```

```
0.15 2
```

or

```
0.0
```

```
0.15
```

```
0.15
```

The shm file can be directly generated by Shermo after loading output file of quantum chemistry program, namely setting "outshm" parameter in *settings.ini* to 1.

List file (.txt)

Shermo is able to calculate various thermodynamic quantities for a batch of conformations (or configurations); at the same time, Boltzmann weights of the conformers as well as conformation weighted data can be outputted. To realize this, you should prepare a file with .txt suffix, which contains paths of output files of quantum chemistry programs or .shm files. For example:

```
C:\mol\conf1.out
```

```
C:\mol\conf2.out
```

```
C:\mol\conf3.out
```

You can also directly specify the electronic energies (in a.u.) for each system used in the actual calculation of thermodynamic quantities. For example

```
C:\mol\conf1.out;233.3333
```

```
C:\mol\conf2.out;233.4333
```

```
C:\mol\conf3.out;233.3733
```

In this case the electronic energies loaded from the three .out files will not be used.

See example 3.5 for illustration of the batch calculation. In this running mode, contribution of individual vibrational modes cannot be printed via the "prtvib" parameter, and scan of temperature and pressure is not supported.

3 Examples

In this section several examples are given, by following the examples one by one you can quickly understand how to utilize various features of Sherma to solve practical problems.

3.1 Calculating common thermodynamic quantities

In this section, I will take output file of frequency analysis for H₂CO by Gaussian 16 as example to describe how to use Sherma very easily to compute common thermodynamic quantities. This file is *example\G16_H2CO_freq.out*.

We will calculate data at 350 K, and we will employ frequency scale factors to eliminate systematical error. As can be seen from the Gaussian output file, the frequency analysis was conducted at B3LYP/6-31G* level (the geometry has been optimized at the same level before). According to Table 10 of *J. Phys. Chem.*, **100**, 16502 (1996), at this level the scale factor for ZPE is 0.9806, and the factors for entropy and heating contribution to internal energy are 1.0015 and 0.9989, respectively. Since the latter two are very close to 1, we simply approximate them as 1.0.

Open the *settings.ini* file in Sherma folder by text editor, change the value after "T=" to 350, change the value after "sclZPE=" to 0.9806, and change the value after "ilowfreq=" to 0 (using RRHO model in this example, which is the same as most quantum chemistry codes like Gaussian, but not as good as quasi-RRHO when there are low-frequencies, see next section). Note that there must be a space between the value and the equal sign. Finally, save the file.

I assume that you are using Windows system, double-clicking the "Sherma.exe" icon to boot up it, then you will see the parameters for current calculation:

```
Running parameters:
Printing individual contribution of vibration modes: No
Temperature:      350.000 K
Pressure:         1.000 atm
Scale factor of vibrational frequencies for ZPE:          0.9806
Scale factor of vibrational frequencies for U(T)-U(0):    1.0000
Scale factor of vibrational frequencies for S(T):         1.0000
Scale factor of vibrational frequencies for CV:           1.0000
Low frequencies treatment: Harmonic approximation
```

Then input the path of input file, namely *example\G16_H2CO_freq.out*. You can also directly drag icon of this file to the window of Sherma, the path will be automatically generated. After pressing ENTER button, Sherma will start to carry out calculation.

Sherma first prints molecular information, as shown below. I suggest you always check this part to guarantee that the running is meaningful. As you can see, the molecular information is fully correct. The electronic energy, spin multiplicity, atom masses and vibrational frequencies are directly loaded from Gaussian output file, the point group is determined by Sherma according to the geometry; the moments of inertia, rotational constants and temperatures are calculated based on geometry and atomic masses.

```
===== Molecular information =====
Electronic energy:      -114.54925400 a.u.
Spin multiplicity:     1
Atom    1 (C )   Mass:   12.000000 amu
```

```

Atom    2 (H )   Mass:    1.007830 amu
Atom    3 (O )   Mass:    15.994910 amu
Atom    4 (H )   Mass:    1.007830 amu
Total mass:      30.010570 amu

Point group: C2v
Rotational symmetry number: 2
Principal moments of inertia (amu*Bohr^2):
    6.337337    46.220300    52.557637
Rotational constants relative to principal axes (GHz):
    284.779127    39.046506    34.338325
Rotational temperatures (K):  13.667227    1.873935    1.647978
This is not a linear molecule

```

```

There are      6 frequencies (cm^-1):
1210.2 1273.5 1544.3 1819.4 2887.7 2945.7

```

Next, you can find contribution to various thermodynamic data by translation. The result is given in two common units. The "NA" denotes Avogadro constant.

```

===== Translation =====
Translational q:  5.810322E+30    q/NA:  9.648265E+06
Translational U:    4.365 kJ/mol    1.043 kcal/mol
Translational H:    7.275 kJ/mol    1.739 kcal/mol
Translational S:   154.502 J/mol/K   36.927 cal/mol/K   -TS:   12.92 kcal/mol
Translational CV:   12.472 J/mol/K   2.981 cal/mol/K
Translational CP:   20.786 J/mol/K   4.968 cal/mol/K

```

Then you can see rotation contribution. Since only for translation, the U differs to H and C_v differs to C_p , therefore contributions to H and C_p from rotation are not explicitly shown.

```

===== Rotation =====
Rotational q:      8.932107E+02
Rotational U:      4.365 kJ/mol    1.043 kcal/mol   =H
Rotational S:      68.967 J/mol/K   16.484 cal/mol/K   -TS:   5.769 kcal/mol
Rotational CV:     12.472 J/mol/K   2.981 cal/mol/K   =CP

```

Next, you can find vibration contribution. As described in Section A.4, there are two forms of vibrational partition function, both of them are shown. The $U(T)-U(0)$ term corresponds to heating contribution to internal energy (or enthalpy) from 0 to 350 K.

```

===== Vibration =====
Vibrational q(V=0):  1.014709E+00
Vibrational q(bot):  3.798025E-11
Vibrational U(T)-U(0):  0.227 kJ/mol    0.054 kcal/mol   =H(T)-H(0)
Vibrational U:      68.738 kJ/mol    16.429 kcal/mol   =H
Vibrational S:      0.771 J/mol/K    0.184 cal/mol/K   -TS:   0.065 kcal/mol
Vibrational CV:      3.534 J/mol/K    0.845 cal/mol/K   =CP
Zero-point energy (ZPE):  68.51 kJ/mol,    16.37 kcal/mol    0.026094 a.u.

```

Electron excitation contribution is also outputted. However, for present system this part does not make sense, since only ground state is taken into account and this state is not degenerate.

```

===== Electron excitation =====
Electronic q:      1.000000E+00
Electronic U:      0.000 kJ/mol      0.000 kcal/mol      =H
Electronic S:      0.000 J/mol/K      0.000 cal/mol/K      -TS:  0.000 kcal/mol
Electronic CV:      0.000 J/mol/K      0.000 cal/mol/K      =CP

```

At the end of the entire output you can find total result, see below. The partition function is product of all aforementioned contributions, all other quantities are sum of all aforementioned contributions.

```

=====
===== Total =====
=====
Total q(V=0):      5.266176E+33
Total q(bot):      1.971115E+23
Total q(V=0)/NA:    8.744690E+09
Total q(bot)/NA:    3.273112E-01
Total CV:          28.478 J/mol/K      6.806 cal/mol/K
Total CP:          36.792 J/mol/K      8.794 cal/mol/K
Total S:           224.240 J/mol/K      53.595 cal/mol/K      -TS:  18.758 kcal/mol
Zero point energy (ZPE):  68.511 kJ/mol      16.375 kcal/mol      0.026094 a.u.
Thermal correction to U:  77.469 kJ/mol      18.515 kcal/mol      0.029506 a.u.
Thermal correction to H:  80.379 kJ/mol      19.211 kcal/mol      0.030615 a.u.
Thermal correction to G:   1.895 kJ/mol      0.453 kcal/mol      0.000722 a.u.
Electronic energy:      -114.5492540 a.u.
Sum of electronic energy and ZPE, namely U/H/G at 0 K:      -114.5231595 a.u.
Sum of electronic energy and thermal correction to U:      -114.5197478 a.u.
Sum of electronic energy and thermal correction to H:      -114.5186394 a.u.
Sum of electronic energy and thermal correction to G:      -114.5485323 a.u.

```

As can be seen from above information, Shermo also outputs sum of electronic energy and various thermal corrections. For example, the "Sum of electronic energy and thermal correction to H" corresponds to the enthalpy of present molecule at current condition (350 K and 1 atm). However, this value is not accurate since electronic energy calculated at B3LYP/6-31G* level is poor. You can manually add the thermal corrections to a more accurate single point energy (e.g. CCSD(T)/cc-pVTZ) at current geometry to obtain better result.

Running under Linux

The way of running this example in Linux platform is mentioned below.

You can edit the *settings.ini* in aforementioned way, then enter the Shermo folder and run *./Shermo* to boot up it, then input the path of input file. You can also specify the path of input file as argument, for example, *./Shermo example/G16_H2CO_freq.out*.

If you hope Shermo can run in arbitrary directory, you should add below two lines into your *~/.bashrc* file and re-enter the shell (the Shermo folder is assumed to be /sob/Shermo_2.0)

```
export Shermopath=/sob/Shermo_2.0
```

```
export PATH=$PATH:/sob/Shermo_2.0
```

In this case, if *settings.ini* cannot be found in current folder, Sherma will try to locate it from /sob/Shermo_2.0 folder. If it still cannot be found, default parameters will be used.

Modification on *settings.ini* in fact can be avoided, since parameters can also be specified by arguments. To reproduce the present example, you can run below command

```
Sherma example/G16_H2CO_freq.out -T 350 -sclZPE 0.9806
```

In this case temperature and scale factor for ZPE are explicitly specified, all other parameters will use the values defined in *settings.ini*; while if *settings.ini* cannot be found, default parameters will be used.

If you want, you can use > operator to redirect all outputted information from screen to a plain text file.

3.2 Printing contribution of individual vibrational modes

In this example, we check contribution of individual vibrational modes to thermodynamic quantities, this is useful to understand the role played by various modes. The input file is *example\ORCA_ethanol_optfreq.out*, which is output file of optimization with frequency analysis of ORCA 4.2 program for ethanol, the B3LYP/6-31G* level was employed. In the *settings.ini*, we set "sclZPE" to 0.9806 as the last example, set "T" to 298.15, and set "prt vib" to -1.

Now, use Sherma to analyze the ORCA output file, then not only thermodynamic data are printed on screen, but also a file named *vibcontri.txt* is generated in current folder, its content is shown below:

Note: The wavenumbers shown below are unscaled ones

Mode	Wavenumber	Freq	Vib. Temp.	q(V=0)	q(bot)
	cm ⁻¹	GHz	K		
1	247.04	0.74061E+04	355.44	1.43589955	0.79114346
2	297.95	0.89323E+04	428.68	1.31138455	0.63901869
3	417.19	0.12507E+05	600.24	1.15414458	0.42178802
...ignored					
19	3127.85	0.93771E+05	4500.27	1.00000028	0.00052769
20	3130.25	0.93843E+05	4503.73	1.00000028	0.00052465
21	3746.34	0.11231E+06	5390.14	1.00000001	0.00011865

Mode	Wavenumber	ZPE	U(T)-U(0)	U(T)	CV(T)	S(T)
	cm ⁻¹	kcal/mol	kcal/mol	kcal/mol	cal/mol/K	cal/mol/K
1	247.04	0.34631	0.30789	0.65420	1.76768	1.75161
2	297.95	0.41768	0.26526	0.68294	1.67754	1.42839
3	417.19	0.58483	0.18386	0.76870	1.43290	0.90157
...ignored						
19	3127.85	4.38474	0.00000	4.38474	0.00013	0.00001
20	3130.25	4.38810	0.00000	4.38811	0.00012	0.00001
21	3746.34	5.25176	0.00000	5.25176	0.00001	0.00000

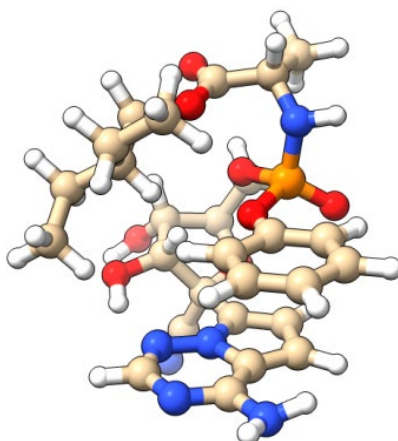
In the output, contribution to various thermodynamic quantities due to vibration has been clearly decomposed as contributions from various modes. Note that the values under "Wavenumber" and

"Freq" labels are the original frequencies (*i.e.* without applying scale factor). From the data it can be seen that the lower the frequency of a mode, the higher its contribution to $U(T)-U(0)$, C_V and S . In contrast, the higher the frequency of a mode, the larger its contribution to ZPE and $U(T)$.

If you prefer to use Shermo in command-line environment, you can also use `-prt vib -l` argument to request Shermo output the `vibcontri.txt`.

3.3 Using quasi-RRHO treatment for low-frequencies

As described in Section A.4, if your system contains many low-frequency modes, it is highly suggested to deal with them using quasi-RRHO model instead of RRHO (*i.e.* harmonic approximation) to obtain more reasonable entropy, $U(T)-U(0)$ and C_V . In this example, we will calculate thermodynamic data for remdesivir, which is a specific medicine against COVID-19 virus. Its most stable conformation determined by the powerful conformation and configuration search program Molclus (<http://www.keinsci.com/research/molclus.html>) is shown below. The output file of optimization with frequency analysis task at B3LYP-D3(BJ)/6-31G* level of Gaussian 16 is `example\G16_remdesivir_optfreq.out`.



We first conduct analysis at standard condition with RRHO treatment like previous examples (using "ilowfreq= 0"). Notice that from output you can find that there are many low frequencies, as shown below. The contribution to S , $U(T)-U(0)$ and C_V from the modes lower than 150 cm^{-1} is usually unable to be well represented by RRHO model.

There are 225 frequencies (cm^{-1}):

8.7	20.0	32.1	33.3	38.1	41.0	45.9	53.2	65.1
70.3	74.4	75.6	79.9	87.4	92.8	104.4	108.3	117.6
123.9	131.4	139.1	150.3	155.5	165.6	175.7	184.9	190.8

...ignored

The output of vibration part is

```

===== Vibration =====
Vibrational q(V=0):      2.077261E+18
Vibrational q(bot):      7.935674-270
Vibrational U(T)-U(0):   100.764 kJ/mol    24.083 kcal/mol    =H(T)-H(0)
Vibrational U:          1709.523 kJ/mol    408.586 kcal/mol    =H
Vibrational S:           688.648 J/mol/K    164.591 cal/mol/K    -TS:  49.073 kcal/mol
Vibrational CV:          632.893 J/mol/K    151.265 cal/mol/K    =CP

```

Zero-point energy (ZPE): 1608.76 kJ/mol, 384.50 kcal/mol 0.612744 a.u.

Then, we change the "ilowfreq" in *settings.ini* to 1, namely raising frequencies lower than a specific threshold to this value. The threshold is controlled by "ravib" parameter, which is 100 by default. The output in this case is

```
===== Vibration =====
Note: 15 low frequencies are raised to 100.0 cm^-1 during calculating S, U(T)
-U(0), CV and q
Vibrational q(V=0): 1.163912E+14
Vibrational q(bot): 8.570305-275
Vibrational U(T)-U(0): 97.144 kJ/mol 23.218 kcal/mol =H(T)-H(0)
Vibrational U: 1705.902 kJ/mol 407.720 kcal/mol =H
Vibrational S: 595.110 J/mol/K 142.235 cal/mol/K -TS: 42.407 kcal/mol
Vibrational CV: 631.362 J/mol/K 150.899 cal/mol/K =CP
Zero-point energy (ZPE): 1608.76 kJ/mol, 384.50 kcal/mol 0.612744 a.u.
```

Next, we use Grimme's quasi-RRHO treatment of entropy for this system. Change the "ilowfreq" to 2 and redo the calculation, the result is

```
===== Vibration =====
Vibrational q(V=0): 2.077261E+18
Vibrational q(bot): 7.935674-270
Vibrational U(T)-U(0): 100.764 kJ/mol 24.083 kcal/mol =H(T)-H(0)
Vibrational U: 1709.523 kJ/mol 408.586 kcal/mol =H
Vibrational S: 608.552 J/mol/K 145.447 cal/mol/K -TS: 43.365 kcal/mol
Vibrational CV: 632.893 J/mol/K 151.265 cal/mol/K =CP
Zero-point energy (ZPE): 1608.76 kJ/mol, 384.50 kcal/mol 0.612744 a.u.
```

Some data are summarized in the following table. Note that the Grimme's quasi-RRHO is only applied to entropy and thus $U(T)-U(0)$ and C_V are not affected.

Model	$U(T)-U(0)$ (kJ/mol)	-TS (kJ/mol)	C_V (J/mol/K)
RRHO	24.1	49.1	151.3
Raising low freq.	23.2	42.4	150.9
Grimme	24.1	43.4	151.3

From the table it can be seen that the $U(T)-U(0)$ and C_V calculated under RRHO and quasi-RRHO (with ilowfreq=1) do not differ with each other significantly, however, both the two quasi-RRHO models altered the -TS term of RRHO by more than 10%, rendering importance of applying quasi-RRHO model for large and flexible systems like this. Since the two quasi-RRHO models result in similar result, both of them may be used in practice. In my opinion, using Grimme's model (ilowfreq=2) is more ideal, since it has relatively more rigorous physical basis.

In command-line environment, you can also activate the quasi-RRHO treatment in calculation by adding *-lowfreq 1* or *-ilowfreq 2* argument.

3.4 Scanning temperature and pressure

In this example, we study variation of various thermodynamic quantities with respect to temperature and pressure. The input file is *examples\NWChem_NH2COH_optfreq.out*, which is optimization with frequency analysis task at B3LYP/6-31G* level by NWChem for NH₂COH.

Again, we employ scale factor of 0.9806 for ZPE by modifying "sclZPE" parameter in *settings.ini*. All other parameters keep default value.

Initially, we scan temperature from 50 to 1000 K with stepsize of 10 K, therefore we modify the line defining temperature in *settings.ini* as "T= 50,1000,10". Start Shermo calculation as usual, then you will find *scan_SCq.txt* has been generated in current folder, which contains S , C_V , C_P , $q^{(V=0)}/N_A$ and $q^{(bot)}/N_A$. Its content is

Unit of S , C_V and C_P is cal/mol/K, $q(V=0)/N_A$ and $q(bot)/N_A$ are dimensionless

T(K)	P(atm)	S	CV	CP	q(V=0)/NA	q(bot)/NA
50.000	1.000	45.106	7.286	9.273	1.012978E+008	7.711220E-117
60.000	1.000	46.813	7.456	9.443	2.219362E+008	8.206613E-096
70.000	1.000	48.278	7.574	9.561	4.353700E+008	9.611775E-081
80.000	1.000	49.561	7.661	9.648	7.865970E+008	2.097511E-069
90.000	1.000	50.701	7.735	9.722	1.333255E+009	1.479802E-060

...ignored

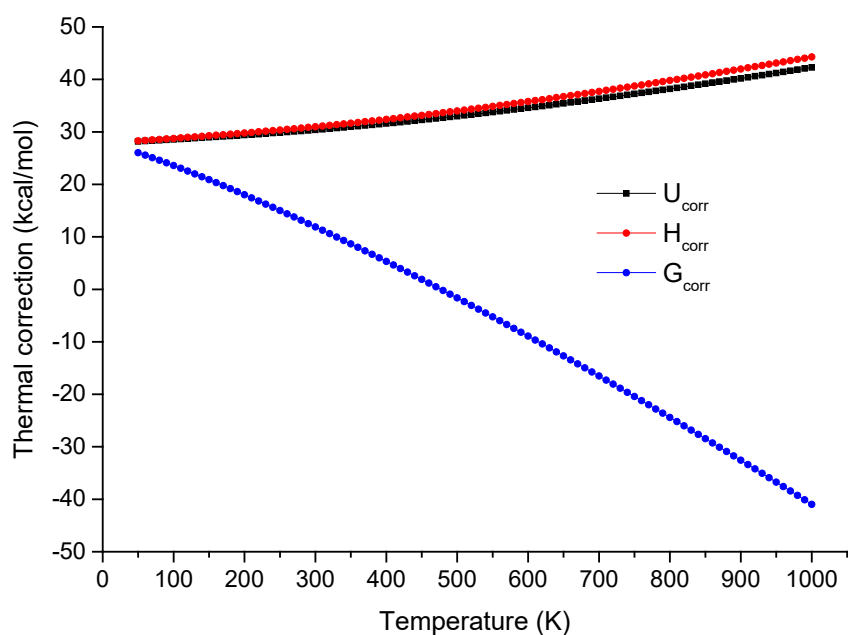
Another file generated in current folder is *scan_UHG.txt*, which contains thermal corrections to U , H and G , as well as sum of each of them and electronic energy:

Unit of U_{corr} , H_{corr} and G_{corr} is kcal/mol, unit of U , H and G is a.u.

T(K)	P(atm)	Ucorr	Hcorr	Gcorr	U	H	G
50.000	1.000	28.170	28.269	26.014	-169.843950	-169.843791	-169.847386
60.000	1.000	28.244	28.363	25.554	-169.843832	-169.843642	-169.848118
70.000	1.000	28.319	28.458	25.079	-169.843712	-169.843491	-169.848876
80.000	1.000	28.395	28.554	24.589	-169.843591	-169.843338	-169.849656

...ignored

Based on the two files, you can easily plot variation curve of various quantities with respect to temperature. For example, the below map was plotted by Origin:



The G_{corr} decreases with increase of temperature due to the $-TS$ term and the fact that S is a positive value.

Similarly, you can set the "P" parameter in *settings.ini* to e.g. "P= 0.5,20,0.1" to scan pressure from 0.5 to 20 atm with stepsize of 0.1 atm.

If you specify lower and upper limits as well as stepsize for both "T" and "P", then Shermo will do two-dimension scan. For example, if you set

T= 250,300,10

P= 0.6,1.0,0.2

then the resulting *scan_SCq.txt* will be

T(K)	P(atm)	S	CV	CP	q(V=0)/N	q(bot)/N
250.000	0.600	62.407	9.953	11.940	2.790475E+11	4.187769E-14
250.000	0.800	61.835	9.953	11.940	2.092856E+11	3.140827E-14
250.000	1.000	61.392	9.953	11.940	1.674285E+11	2.512662E-14
260.000	0.600	62.879	10.149	12.136	3.404227E+11	4.603420E-13
260.000	0.800	62.307	10.149	12.136	2.553170E+11	3.452565E-13
260.000	1.000	61.864	10.149	12.136	2.042536E+11	2.762052E-13
270.000	0.600	63.341	10.349	12.336	4.128023E+11	4.274044E-12
270.000	0.800	62.769	10.349	12.336	3.096017E+11	3.205533E-12
270.000	1.000	62.326	10.349	12.336	2.476814E+11	2.564426E-12

...ignored

3.5 Using electronic energy at better level

In the previous examples, the electronic energy loaded from input file was employed. However, in practical calculations of thermodynamic data, commonly electronic energy evaluated at a better level via an additional single point task should be used, since U , H and G are dominated by electronic energy, while accuracy of electronic energy is significantly more sensitive to quality of calculation level than optimization and frequency analysis tasks. In this section, the ethyl radical will be taken as example to illustrate how to use electronic energy evaluated at a better level in thermochemistry calculation.

This time we perform optimization and frequency analysis at a more robust level than earlier examples, namely B2PLYP-D3(BJ)/cc-pVTZ. The output file is *example\C2H5_optfreq.out*. The optimal ZPE scale factor for this level is reported to be 0.9830 in *J. Chem. Theory Comput.*, **12**, 3774 (2016), while the scale factor for S and $U(T)-U(0)$ can be still approximated to be 1.0. Based on the optimized geometry a single point task was conducted, the electronic energy calculated at the very accurate CCSD(T)/cc-pVQZ level was found to be -79.0257037 a.u.

In the *settings.ini*, we change the "scfZPE" parameter to 0.983, change the "ilowfreq" to 2 to use Grimme's quasi-RRHO treatment since this system has a low-frequency mode (rotation of methyl group), and set "E" to -79.0257037.

Now boot up Shermo and use the *C2H5_optfreq.out* as input file, the result is

```
Electronic energy:          -79.0257037 a.u.
Sum of electronic energy and ZPE, namely U/H/G at 0 K:      -78.9670179 a.u.
Sum of electronic energy and thermal correction to U:        -78.9630711 a.u.
Sum of electronic energy and thermal correction to H:        -78.9621269 a.u.
Sum of electronic energy and thermal correction to G:        -78.9910754 a.u.
```


Since the current electronic energy is highly accurate, the ZPE scale factor has been taken into account, and quasi-RRHO treatment is employed, the resulting U , H and G shown above should be fairly reliable.

If you prefer to use command-line mode to run Shermo, you can input

```
Shermo example/C2H5_optfreq.out -E -79.0257037 -sclZPE 0.983 -ilowfreq 2
```

Customizing electronic energy via shm file

In Shermo there is another way of employing electronic energy calculated at other level than frequency analysis, namely utilizing .shm file as input file, as illustrated below.

We set the "outshm" in *settings.ini* to 1, in this case Shermo will generate a .shm file when output file of quantum chemistry program is used as input file. The shm file contains everything that needed by Shermo to compute thermodynamic data and thus can then be used as input file. You can manually make various modifications on .shm file.

Now, boot up Shermo and use the *example\C2H5_optfreq.out* as input file, then *example\C2H5_optfreq.shm* will be generated. Open this file by text editor, change the value under *E label as -79.0257037. Now the file content should be

```
*E //Electronic energy (a.u.)
-79.0257037

*wavenum //Wavenumbers (cm-1). Negative value means imaginary frequency
120.0311
479.5592
818.5287
990.8014
1073.5208
1207.5756
1413.2669
1483.9190
1497.2870
1498.1094
2985.2923
3070.3466
3114.8558
3174.7147
3277.7883

*atoms //Information of all atoms: Name, mass (amu), X, Y, Z (Angstrom)
C 12.000000 0.055460 -0.706009 0.000000
H 1.007830 -0.452385 -1.093686 0.882849
H 1.007830 1.060514 -1.146711 0.000000
H 1.007830 -0.452385 -1.093686 -0.882849
C 12.000000 0.103111 0.777922 0.000000
H 1.007830 0.193594 1.327564 -0.923232
H 1.007830 0.193594 1.327564 0.923232

*elevel //Energy (eV) and degeneracy of electronic energy levels
0.000000 2
```

Note that the data under *elevel label represents electronic energy levels. Since only ground state is

considered, the excitation energy is 0.000000 eV. The current system is doubly degenerate due to spin multiplicity of 2, therefore there is a "2" in this line. If you use this .shm file as input file of Shermo, you will obtain exactly the same result as before.

3.6 Consideration of degeneracy of ground state and spin-orbit coupling splitting

When ground state has spatial degeneracy, its influence on entropy and free energy must be properly taken into account. In this section I make a few remarks about this point.

The *elevel field in .shm file records all electronic levels that considered in the calculation. Usually you do not need to manually modify this field, because:

(1) Separations between ground state and excited states are commonly as large as a few electronvolts, thus electron excitations do not detectably contribute to thermodynamic data.

(2) Commonly degeneracy of ground state is only due to electron spin multiplicity, which is directly loaded by Shermo from input file, thus this point is automatically taken into account.

However, for some single atom systems and diatom systems, degeneracy also comes from spatial part of wavefunction. In addition, spin-orbit coupling (SOC) effect can split the degenerate ground states and result in very low-lying excited states. Below I give two instances to illustrate how to consider this point.

OH radical

The ground state of OH radical ($^2\Pi$ state) has spin multiplicity of 2 and spatial degeneracy of 2, therefore the total degeneracy of this state is $2 \times 2 = 4$. So, the content under the *elevel label should be

```
0.0 4
```

The result of electron excitation part of output is

```
Electronic q: 4.000000E+000
Electronic U: 0.000 kJ/mol      0.000 kcal/mol      =H
Electronic S: 11.526 J/mol/K    2.755 cal/mol/K    -TS: 0.821 kcal/mol
Electronic CV: 0.000 J/mol/K    0.000 cal/mol/K    =CP
```

While if you hope to take SOC effect into account to obtain more rigorous result, the four degenerate states will split into two doubly degenerate states with energy separation of 139.2 cm^{-1} (0.0172 eV), in this case the content of the *elevel field should be changed to

```
0.0 2
```

```
0.0172 2
```

Now the result is

```
Electronic q: 3.023977E+000
Electronic U: 561.954 kJ/mol    134.310 kcal/mol    =H
Electronic S: 11.085 J/mol/K    2.649 cal/mol/K    -TS: 0.790 kcal/mol
Electronic CV: 0.835 J/mol/K    0.199 cal/mol/K    =CP
```

The comparison shows that SOC effect only marginally affects thermodynamic data, the $-TS$ only changed by 0.03 kcal/mol.

Fluorine atom

Ground state of fluorine atom is 2P , its spin multiplicity is 2 while spatial degeneracy is 3 because fluorine atom has three 2p orbitals, the unpaired electron may occupy any of them. To represent this fact, the content of the *elevel field should be

0.0 6

Due to SOC effect, the six degenerate states are split to a $J=3/2$ state of degeneracy of four (ground state) and a $J=1/2$ state of degeneracy of two, whose excitation energy is 404.141 cm^{-1} (0.050107 eV). To account for this situation, the content of *elevel should be

0.0 4

0.050107 2

Comparison of different considerations of ground state degeneracy is given below, where $T=298.15\text{ K}$ and n stands for degeneracy

	Spin degeneracy ($n=2$)	Spin + spatial degeneracy ($n=6$)	with SOC ($n=4+2$)
q	2	6	4.3
S (J/mol/K)	5.76	14.90	13.17
$-TS$ (kJ/mol)	0.41	1.06	0.94

As can be seen, including spatial degeneracy has crucial influence on Gibbs free energy, which reflects by the $-TS$ term. Clearly, spatial degeneracy should never be ignored if accuracy of the result is important. The SOC effect only slightly affects the $-TS$ term ($0.94 - 1.06 = -0.12\text{ kJ/mol}$), thus ignoring SOC effect is usually an acceptable approximation in the calculation of thermal correction to Gibbs free energy.

3.7 Calculate a batch of conformations and obtain conformation weighted thermodynamic data

The remdesivir molecule has been studied in Section 3.3. In fact, conformation search of the remdesivir by Molclus code yielded a batch of low energy conformations. In this example we will use Shermo to calculate thermodynamic data of all conformers at once and meantime obtain Boltzmann weighted result. For the three lowest conformers, the output files of optimization with frequency analysis task of Gaussian as well as output files of single point calculation of ORCA have been provided in "example\ensemble" folder. Since the frequency analyses were conducted at B3LYP-D3(BJ)/6-31G* level, we will use ZPE scale factor of 0.9806. The single point calculation was conducted at PWPB95-D3(BJ)/def2-QZVPP level with SMD solvation model, which is fairly robust. Again, since the remdesivir is quite flexible, we will use Grimme's quasi-RRHO treatment for entropy.

In the *settings.ini* file, we set "sclZPE" to 0.9806, set "ilowfreq" to 2, all other parameters keep default values. Then we create a file named *list.txt* with following content

```
example\ensemble\gau00001.out; -2321.367718855799
example\ensemble\gau00002.out; -2321.363532347723
example\ensemble\gau00003.out; -2321.369898466825
```

Each line contains path of an output file, the value after semicolon is the single point energy in a.u. evaluated at PWPB95-D3(BJ)/def2-QZVPP with SMD (if you do not explicitly specify the electronic energies, then the electronic energies loaded from the .out files will be used, which are poor).

Now run Shermo using the *list.txt* as input file, you will see below output information

```
Processing example\ensemble\gau00001.out... ( 1 of 3 )
Processing example\ensemble\gau00002.out... ( 2 of 3 )
Processing example\ensemble\gau00003.out... ( 3 of 3 )
```

#System	U	H	G	S	CV
	a.u.	a.u.	a.u.	J/mol/K	J/mol/K
1	-2320.713764	-2320.712819	-2320.821265	954.971	657.837
2	-2320.709745	-2320.708801	-2320.817499	957.191	657.295
3	-2320.715999	-2320.715054	-2320.822180	943.341	654.902

System	1	Relative G=	2.401 kJ/mol	Boltzmann weight=	27.380 %
System	2	Relative G=	12.289 kJ/mol	Boltzmann weight=	0.507 %
System	3	Relative G=	0.000 kJ/mol	Boltzmann weight=	72.113 %

Conformation weighted data:

Electronic energy: -2321.369269 a.u.

U: -2320.715355 a.u.

H: -2320.714411 a.u.

G: -2320.822488 a.u.

S: 951.727 J/mol/K Conformation entropy: 5.132 J/mol/K

CV: 655.718 J/mol/K

CP: 664.032 J/mol/K

It can be seen that various thermodynamic quantities for the three conformations are separately outputted, then the Gibbs free energies of the three systems relative the lowest one are shown, and meantime the Boltzmann weights estimated according to relative free energies are also given. Evidently conformer 3 plays a dominant role, while conformer 2 can be safely ignored (at least in the present condition, namely 298.15 K and 1 atm). Finally, conformation weighted thermodynamic data as well as conformation entropy are shown, see Section A.6 on how they are calculated. Note that the conformation entropy has already been considered in the value printed after "S".

Note that there is an another way of realizing the present example. We create *list.txt* with below content:

```
example\ensemble\gau00001.out
example\ensemble\gau00002.out
example\ensemble\gau00003.out
```

Then change "outshm" in *settings.ini* to 1, then boot up Shermo and use the *list.txt* as input file, you will see

```
Processing example\ensemble\gau00001.out... ( 1 of 3 )
Outputting data to example\ensemble\gau00001.shm
example\ensemble\gau00001.shm has been successfully generated
Processing example\ensemble\gau00002.out... ( 2 of 3 )
Outputting data to example\ensemble\gau00002.shm
example\ensemble\gau00002.shm has been successfully generated
Processing example\ensemble\gau00003.out... ( 3 of 3 )
Outputting data to example\ensemble\gau00003.shm
example\ensemble\gau00003.shm has been successfully generated
```

Now we have three .shm files corresponding to the three conformers. In order to use more accurate electronic energy, we copy the PWPB95-D3(BJ)/def2-QZVPP single point energies from

the three ORCA output files to the corresponding .shm files. For example, the *example\ensemble\orcaSP00001.out* contains the energy of conformer 1, thus we copy the energy "-2321.367718855799" from it to the second line of *example\ensemble\gau00001.shm*. After modifying the three .shm files, we replace all ".out" in the *list.txt* with ".shm". Finally, use the *list.txt* as input file to run Shermo, you will obtain exactly the same result as before.

3.8 Consideration of variation of Gibbs free energy due to concentration change

Variation of Gibbs free energy caused by change of concentration at temperature T is

$$\Delta G_{\text{conc}} = RT \ln \left(\frac{c_B}{c_A} \right)$$

where c_A and c_B are concentrations of initial state and final state. respectively.

This change of G must be properly taken into account. In particular, when G of a molecule in solution is studied, it is usually given at concentration of 1 M, which is standard state concentration of solution phase. Via Shermo it is straightforward to convert the G at present concentration (namely $c=P/RT$ according to ideal gas formula) to 1 M or other expected concentration.

As an example, the G obtained in the example of last section corresponds to concentration at gas phase standard state (since T and P are set to 298.15 K and 1 atm in *settings.ini*, respectively). To obtain the G corresponding to 1 M concentration (still at 298.15 K), we should change value of "conc" parameter in *settings.ini* from the default 0 to 1M. Then after running the example in last section again, at the end of output you will see

```
[ignored...]
G:      -2320.822488 a.u.
S:      951.727 J/mol/K      Conformation entropy:      5.132 J/mol/K
CV:      655.718 J/mol/K
CP:      664.032 J/mol/K

Present concentration (estimated by ideal gas model):  0.040874 mol/L
Concentration specified by "conc" parameter:      1.000000 mol/L
delta-G of conc. change:      7.926 kJ/mol      1.894 kcal/mol      0.003019 a.u.
Weighted Gibbs free energy at specified concentration:      -2320.8194695 a.u.
```

It shows that the ΔG_{conc} corresponding to changing from present state (0.040874 M) to the concentration specified by "conc" parameter (1 M) is 0.003019 a.u., adding it to the G in present state (-2320.822488 a.u., the same as the last example) results in the G in 1M ($G_{1\text{M}} = -2320.8194695$ a.u.) Usually, when G in solution is involved in study, such as calculating ΔG of chemical reaction that occurs in solution, you should use $G_{1\text{M}}$. Of course, you can also for example set "conc" to 2.4M if G corresponding to other concentration is needed.

PS: The 0.040874 M (mol/L) is calculated $P/RT=101325/(8.314*298.15)/1000$. Note that 1 atm=101325 Pa, $R=8.314$ J/mol/K, the 1000 is used to convert mol/m³ to mol/L.

Skill: Using shell script to invoke Shermo to deal with a batch of files

If you have a lot of files for which thermodynamic data are to be calculated, you can easily write a shell script to automatically invoke Shermo to calculate the data. For example, assume that

you are using Linux platform (or under Windows platform but using WSL, cmd or Cygwin to simulate Bash shell), there are a batch of .out files, you want to calculate Gibbs free energy at standard condition for all of them, ZPE scale factor of 0.975 and Grimme's QRRHO treatment will be used, you can create a text file (for example *getGall.sh*) with the following content

```
#!/bin/bash
outname=Gall.txt
rm -f $outname
for inf in `ls -v *.out`
do
echo Processing ${inf} ...
echo ${inf} >> $outname
./Shermo ${inf} -sclZPE 0.975 -ilowfreq 2 | grep "Sum of electronic energy and
thermal correction to G:" | cut -d: -f 2 >> $outname
done
echo "Data have been exported to $outname in current folder"
```

The content of this script is very easy to comprehend if you have basic knowledge about shell scripting. Put this shell script and Sherma executable file in the current folder, then run command *./getGall.sh*, then all .out files will be processed by Sherma in turn, finally from the *Gall.txt* in current folder you can find the result, for example

```
Cs-atom-in.out
-705.4867512 a.u.
Cs-ion-in.out
-705.2690829 a.u.
Cs-ion-out.out
-705.2490426 a.u.
K-atom-in.out
-1285.2091502 a.u.
K-ion-in.out
-1284.9920128 a.u.
K-ion-out.out
-1284.9852452 a.u.
```

Please write your own scripts based on the above example to facilitate your practical researches. The scripts for extracting electronic energy, *G*, *H* and *-TS* terms for all .out files in current folder have already been provided as *getEall.sh*, *getGall.sh*, *getHall.sh* and *get-TSall.sh* in “examples” folder, respectively.

Appendix 1: An introduction of thermodynamic data calculation

In order to make users understand how Shermo works, an overview of calculation of thermochemistry quantities is given in this section.

A.1 Basic concept

The thermochemistry quantities of common interest include internal energy (U), enthalpy (H), entropy (S), Gibbs free energy (G), constant volume heat capacity (C_V) and constant pressure heat capacity (C_P). The partition function (q), which plays a key role in statistical thermodynamics, is also frequently investigated.

Below equations can be found in any physical chemistry text book

$$\begin{aligned}H(T) &= U(T) + P \times V \\G(T) &= H(T) - T \times S(T)\end{aligned}$$

where T is temperature, P is pressure, V is volume.

In Shermo, all calculations are based on ideal gas assumption and are reported for 1 mole substance. In this case, $PV=RT$ and $C_P=C_V+R$, where $R=8.3144648 \text{ J}\cdot(\text{mol}\cdot\text{K})^{-1}$ is ideal gas constant.

The U , H and G can be written as sum of electronic energy (ε_{ele}) and corresponding thermal corrections:

$$\begin{aligned}U(0) &= H(0) = G(0) = \varepsilon_{\text{ele}} + \text{ZPE} \\U(T) &= \varepsilon_{\text{ele}} + U_{\text{corr}}(T) \\H(T) &= \varepsilon_{\text{ele}} + H_{\text{corr}}(T) \\G(T) &= \varepsilon_{\text{ele}} + G_{\text{corr}}(T)\end{aligned}$$

where ZPE is acronym of zero-point energy, which is sometimes also referred to as ZPVE (zero-point vibrational energy).

At finite temperature, the thermal corrections can also be written as sum of ZPE and contribution by heating the system from 0 K to current temperature

$$\begin{aligned}U_{\text{corr}}(T) &= \text{ZPE} + U_{0 \rightarrow T} = \text{ZPE} + [U(T) - U(0)] = \text{ZPE} + \int C_V(T) dT \\H_{\text{corr}}(T) &= \text{ZPE} + H_{0 \rightarrow T} = \text{ZPE} + [H(T) - H(0)] = \text{ZPE} + \int C_P(T) dT \\G_{\text{corr}}(T) &= \text{ZPE} + G_{0 \rightarrow T} = \text{ZPE} + [G(T) - G(0)]\end{aligned}$$

The thermochemistry quantities can be approximated as sum of contributions from translation (trans), rotation (rot), vibration (vib) and electron (ele) by ignoring their coupling

$$\begin{aligned}U_{\text{corr}} &= U_{\text{trans}} + U_{\text{rot}} + U_{\text{vib}} + U_{\text{ele}} \\S &= S_{\text{trans}} + S_{\text{rot}} + S_{\text{vib}} + S_{\text{ele}} \\C_V &= C_{V,\text{trans}} + C_{V,\text{rot}} + C_{V,\text{vib}} + C_{V,\text{ele}}\end{aligned}$$

Similarly for $H_{\text{corr}}(T)$ and $G_{\text{corr}}(T)$.

The H_{corr} and U_{corr} , as well as C_P and C_V , only differ by their translation parts

$$H_{\text{trans}} = U_{\text{trans}} + RT$$

$$C_{\text{P,trans}} = C_{\text{V,trans}} + R$$

Partition function (q) is a dimensionless quantity playing crucial role in statistical thermodynamic theory, it represents average number of thermal accessible quantum states. Formally, all aforementioned quantities can be derived based on q of a molecule, for example

$$U = kT^2 \left(\frac{\partial \ln Q}{\partial T} \right)_V$$

$$S = k \ln Q + kT \left(\frac{\partial \ln Q}{\partial T} \right)_V$$

where Q is ensemble partition function. Under ideal gas assumption, for an ensemble consisting of N molecules, the Q relates to molecular q with

$$Q = \frac{q^N}{N!}$$

When coupling between translation, rotation, vibrations and electron is ignored, q can be expressed as

$$q_{\text{tot}} = q_{\text{trans}} q_{\text{rot}} q_{\text{vib}} q_{\text{ele}}$$

$$= \sum_a^{\text{trans}} \exp \left(\frac{-\varepsilon_{\text{trans}}^a}{kT} \right) \sum_b^{\text{rot}} \exp \left(\frac{-\varepsilon_{\text{rot}}^b}{kT} \right) \sum_c^{\text{vib}} \exp \left(\frac{-\varepsilon_{\text{vib}}^c}{kT} \right) \sum_d^{\text{ele}} \exp \left(\frac{-\varepsilon_{\text{ele}}^d}{kT} \right)$$

where ε denotes energy of a molecular level. For example, $\varepsilon_{\text{rot}}^b$ denotes molecular rotation state b . The k is Boltzmann constant $1.3806503 \times 10^{-23} \text{ J} \cdot \text{K}^{-1}$.

Next, let us see how the aforementioned quantities are calculated, all working equations are given. Unless otherwise specified, unit of U , H and G is $\text{J} \cdot \text{mol}^{-1}$, unit of S , C_V and C_P is $\text{J} \cdot (\text{mol} \cdot \text{K})^{-1}$.

A.2 Translation contribution

Contribution of molecular overall translation motion to various thermodynamic data can be well calculated as follows if temperature is not extremely low ($< 5 \text{ K}$ for most molecules)

$$q_{\text{trans}} = \left(\frac{2\pi mkT}{h^2} \right)^{3/2} \frac{RT}{P}$$

$$U_{\text{trans}}(T) = (3/2)RT$$

$$H_{\text{trans}}(T) = U_{\text{trans}}(T) + RT$$

$$C_{\text{V,trans}} = (3/2)R$$

$$C_{\text{P,trans}} = C_{\text{V,trans}} + R$$

$$S_{\text{trans}}(T) = R[\ln(q_{\text{trans}} / N_A) + 5/2]$$

where h is Planck constant, $6.6260696 \times 10^{-34} \text{ J} \cdot \text{s}$, N_A is Avogadro constant $6.02214179 \times 10^{23}$, m is mass of current molecule. It is noteworthy that in all thermochemistry quantities, pressure only affects translation contribution to entropy (S_{trans}).

A.3 Rotation contribution

Molecular rotation also has contribution to various thermodynamic data. Of course, single atom does not have rotation contribution.

Linear molecule:

$$q_{\text{rot}} = \frac{8\pi^2 IkT}{\sigma h^2}$$

$$U_{\text{rot}}(T) = RT$$

$$C_{V,\text{rot}}(T) = R$$

$$S_{\text{rot}}(T) = R(\ln q_{\text{rot}} + 1)$$

Nonlinear molecule:

$$q_{\text{rot}} = \frac{8\pi^2}{\sigma h^3} (2\pi kT)^{3/2} \sqrt{I_A I_B I_C}$$

$$U_{\text{rot}}(T) = (3/2)RT$$

$$C_{V,\text{rot}}(T) = (3/2)R$$

$$S_{\text{rot}}(T) = R(\ln q_{\text{rot}} + 3/2)$$

where σ is rotational symmetry number, $\{I\}$ are principal moments of inertia.

The σ is determined according to point group:

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

It is worth to note that in Shermo, the point group is detected based on current geometry based on my adapted SYVA code. The original SYVA code was described in *Comput. Phys. Commun.*, **215**, 156 (2017).

The principal moments of inertia are the three eigenvalues of the moments of inertia tensor:

$$\mathbf{I} = \begin{bmatrix} \sum_i m_i (y_i^2 + z_i^2) & -\sum_i m_i x_i y_i & -\sum_i m_i x_i z_i \\ -\sum_i m_i y_i x_i & \sum_i m_i (x_i^2 + z_i^2) & -\sum_i m_i y_i z_i \\ -\sum_i m_i z_i x_i & -\sum_i m_i z_i y_i & \sum_i m_i (x_i^2 + y_i^2) \end{bmatrix}$$

where x_i, y_i, z_i are Cartesian coordinates of atom i , the m_i is mass of atom i .

For a linear molecule, two of its principal moments of inertia are identical, and the remaining one is zero. Shermo determines whether or not a molecule is linear by checking its lowest principal moment of inertia. If it is less than $0.001 \text{ kg}\cdot\text{m}^2$, then the molecule will be regarded as linear.

The rotation constants are also outputted by Shermo, they are closely related to principal moments of inertia:

$$A = \frac{h}{8\pi^2 I_A} \quad B = \frac{h}{8\pi^2 I_B} \quad C = \frac{h}{8\pi^2 I_C}$$

If the unit of I is $\text{kg}\cdot\text{m}^2$, then the unit of the corresponding rotation constants will be Hz.

Shermo also outputs rotation temperatures:

$$T_A^{\text{rot}} = \frac{hA}{k} \quad T_B^{\text{rot}} = \frac{hB}{k} \quad T_C^{\text{rot}} = \frac{hC}{k}$$

A.4 Vibration contribution

Every real frequency has contribution to partition function and various thermodynamic quantities, while imaginary frequency modes are ignored during calculation. Harmonic approximation is assumed below; this is also known as rigid-rotor harmonic oscillator (RRHO) model.

Total vibration partition function is product of contributions of individual vibrational modes:

$$q_{\text{vib}}^{(\text{BOT})} = \prod_i \frac{e^{-h\nu_i/(2kT)}}{1 - e^{-h\nu_i/(kT)}}$$

$$q_{\text{vib}}^{(\text{V}=0)} = \prod_i \frac{1}{1 - e^{-h\nu_i/(kT)}}$$

where ν_i is frequency of vibration mode i in Hz. The $q_{\text{vib}}^{(\text{BOT})}$ is the partition function calculated by taking bottom of potential energy surface as zero point, while $q_{\text{vib}}^{(\text{V}=0)}$ is the partition function calculated by viewing vibrational ground state as zero point, and it reflects thermal excitation of vibrational states. $q_{\text{vib}}^{(\text{V}=0)}$ is more commonly used since its magnitude is not so small as $q_{\text{vib}}^{(\text{BOT})}$.

Thermodynamic quantities due to vibrations can be written as sum of individual contribution of each vibration mode:

$$U(0)_{\text{vib}} = \text{ZPE} = \frac{1}{2} \sum_i h\nu_i$$

$$U_{0 \rightarrow T}^{\text{vib}} \equiv U(T)_{\text{vib}} - U(0)_{\text{vib}} = RT \sum_i \left(\frac{h\nu_i}{kT} \right) \frac{e^{-h\nu_i/(kT)}}{1 - e^{-h\nu_i/(kT)}}$$

$$C_{\text{V,vib}}(T) = R \sum_i \left(\frac{h\nu_i}{kT} \right)^2 \frac{e^{-h\nu_i/(kT)}}{[1 - e^{-h\nu_i/(kT)}]^2}$$

$$S_{\text{vib}}(T) = R \sum_i \left\{ \frac{h\nu_i}{kT} \frac{e^{-h\nu_i/(kT)}}{1 - e^{-h\nu_i/(kT)}} - \ln[1 - e^{-h\nu_i/(kT)}] \right\}$$

A vibration mode with higher frequency has larger contribution to ZPE, while a mode with lower frequency has larger contribution to the other three terms.

Frequency scale factor

The error of thermodynamic quantities calculated in above way has various sources:

- (1) Representation of potential energy surface is inaccurate. This is inevitable for any practical calculation level.
- (2) Anharmonic effect is ignored
- (3) Torsional motions of some groups or fragments should be treated as free or hindered rotor

rather than (an)harmonic oscillator.

(4) Some torsional motions may be strongly coupled with others

The errors (1) and (2) are most prominent and relatively systematical, many studies found that they can be largely eliminated by applying scale factors for vibrational frequencies. Specifically, before calculating a thermodynamic quantity, the raw harmonic frequencies are multiplied by a pre-fitted factor (λ). The λ is dependent of both theoretical method and basis set, in other words, different calculation level has different optimal λ . Many papers fitted λ for ZPE by taking experimental data as reference, such as *J. Chem. Theory Comput.*, **6**, 2872 (2010), while some other papers also present λ for S and $U_{0 \rightarrow T}^{\text{vib}}$, such as *J. Phys. Chem.*, **100**, 16502 (1996). Different λ should be applied to frequencies prior to calculating different thermodynamic quantities, thus Shermo allows user to simultaneously specify different λ . Usually, the λ for ZPE deviates 1.0 notably (except for pure DFT functionals), therefore it should always be considered if accuracy of the result is important. In contrast, the λ for S and $U_{0 \rightarrow T}^{\text{vib}}$ is usually close to 1.0 and thus often be ignored.

To maximally eliminate the aforementioned error source (3), hindered rotor treatment should be employed, such as using the `freq=hinderedrotor` keyword in Gaussian, however this is beyond scope of Shermo. Using the black box quasi-RRHO treatment as shown below can also partially relieve this error.

The way to deal with error source (4) is more involved and sophisticated method is needed, such as the MS-T method, see *J. Chem. Theory Comput.*, **9**, 1356 (2013).

Quasi-RRHO treatment for low frequencies

The breakdown of harmonic approximation of low-frequency modes is well known, especially for Gibbs free energy. The S and heating contribution to U due to mode i can be written as

$$S^i(T) = R \left\{ \frac{h\nu_i}{kT} \frac{1}{e^{h\nu_i/(kT)} - 1} - \ln[1 - e^{-h\nu_i/(kT)}] \right\}$$

$$U_{0 \rightarrow T}^i(T) = \left(\frac{h\nu_i}{kT} \right) \frac{RT}{e^{h\nu_i/(kT)} - 1}$$

Clearly, as the vibration frequency ν_i approaches zero, the above two quantities diverge. There are some methods to avoid this issue and they are usually referred to as quasi-RRHO (QRRHO) or modified RRHO (mRRHO), and usually frequency scaled factor is simultaneously used, in this case they are known as modified scaled RRHO (msRRHO).

Truhlar's QRRHO model: The first QRRHO treatment is the one suggested by Truhlar, see *J. Phys. Chem. B*, **115**, 14556 (2011) for example. This method artificially raises all (unscaled) frequencies that lower than 100 cm^{-1} to 100 cm^{-1} , this threshold was chosen empirically by the authors. In Shermo, this treatment is enabled in the calculation of S , C_V/C_P and $U/H/G_{0 \rightarrow T}$ when "ilowfreq" parameter is set to 1, and the threshold can be adjusted by "ravib" parameter. In *J. Phys. Chem. A*, **123**, 7430 (2019), it was shown that for small molecules, the accuracy of result is insensitive to the choice of the threshold, while for molecular complexes, the threshold of 100 cm^{-1} performs best.

Grimme's QRRHO model: Another very popular QRRHO treatment is the one proposed by Grimme in *Chem. Eur. J.*, **18**, 9955 (2012), it only applies to entropy. Essentially, this method performs interpolation between the vibrational entropy calculated under harmonic approximation (S_{RRHO}) and the one of free-rotor model (S_{FR}):

$$S = w(\nu)S_{\text{RRHO}} + [1 - w(\nu)]S_{\text{FR}}$$

$$w(\nu) = \frac{1}{1 + (\nu_0 / \nu)^\alpha} \quad \alpha = 4$$

where ν is the unscaled frequency of the mode under consideration, while ν_0 is usually taken as 100 cm^{-1} (determined by “intpvib” in *settings.ini*), other values within 50~150 cm^{-1} can also give similar result. The S_{FR} is calculated as follows

$$S_{\text{FR}} = R \left(\frac{1}{2} + \ln \sqrt{\frac{8\pi^3 \mu' kT}{h^2}} \right)$$

with

$$\mu' = \frac{\mu B_{\text{av}}}{\mu + B_{\text{av}}} \quad \mu = \frac{h}{8\pi^2 \nu}$$

where μ is moment of inertia of a free-rotor with the same frequency. μ' is effective moment of inertia. The B_{av} corresponds to the average molecular moment of inertia and was empirically chosen to be $10^{-44} \text{ kg}\cdot\text{m}^2$.

The Grimme's QRRHO can be enabled by setting "ilowfreq=2". It is evidently more elegant than artificially raising low frequencies and thus should be more accurate and robust. Therefore, I always recommend setting "ilowfreq" to 2 if the system is quite flexible (e.g. a large molecule with many rotatable bonds or molecular complexes), since it is always accompanied by lots of low-frequency modes. It was found that for a flexible organic molecule consisting of several hundreds of atoms, the difference of G computed based on RRHO and QRRHO can reach as high as several kcal/mol, clearly in this case employing QRRHO is compulsory if quantitative accuracy of the result is important. For rigid molecules, the two QRRHO methods do not bring additional benefits since there is no low frequency.

Minenkov's QRRHO model: In *J. Comput. Chem.*, **44**, 1807 (2023), Otlyotov and Minenkov showed that if the interpolation between harmonic approximation and free-rotor model is not only applied to entropy as above, but also applied to vibrational contribution to internal energy, then the free energy will be slightly more accurate, and quality of internal energy will also be improved. This consideration also makes the Grimme's QRRHO model more consistent. This model corresponds to "ilowfreq=3". The interpolation equation for the internal energy part is as follows

$$U_{\text{vib}} = w(\nu)U_{\text{RRHO}} + [1 - w(\nu)]U_{\text{FR}}$$

where definition of w is exactly the same as that in Grimme's QRRHO model, $U_{\text{FR}} = RT/2$ is free-rotor contribution to internal energy, and U_{RRHO} is vibrational internal energy estimated by harmonic approximation. Notice that when using this model, ZPE and $U_{0 \rightarrow T}$ cannot be separated, and thus they are not outputted by Shermo.

Note that the S_{RRHO} and U_{RRHO} in above formula are calculated based on scaled frequencies.

Consideration of small imaginary frequencies

Note: See my blog article “On the calculation of thermodynamic quantities with small imaginary frequencies” (<http://sobereva.com/699>, in Chinese) for more comprehensive discussion about this topic.

For systems such as flexible macromolecules and weakly interacting complexes, where certain dimensions of potential energy surface are very smooth, after geometric optimization and vibration analysis, it is often found that there are small imaginary frequencies, such as 9.8 and 27.3 cm^{-1} , which are generally less than 50-100 cm^{-1} . This is usually caused by the issue that geometric optimization was not converged accurately enough, or the integration grid employed is not fine

enough (this problem is particularly significant for functionals that are sensitive to the integration grid such as M06-2X). In this case, the imaginary frequency modes usually correspond to the motion such as methyl rotation, flexible skeleton deformation, and intermolecular relative translation or rotation. Note that the imaginary frequency (usually above 100 cm^{-1}) that exists due to the transition state structure or the symmetry of the initial structure is higher than the actual one does not belong to the situation discussed here.

When encountering such small imaginary frequencies, the most ideal approach is to use tighter geometry optimization convergence threshold or use better integration grid to redo the optimization and vibration analysis so that the imaginary frequencies are completely eliminated before calculating thermodynamic quantities. However, for large systems, eliminating all imaginary frequencies often requires very high computational time and thus is difficult to achieve. In fact, it is not unacceptable to calculate thermodynamic quantities in the presence of such small imaginary frequencies. These frequencies actually do not cause obvious problems in the calculation accuracy of electronic energy, because gradients of potential energy surface in the direction of normal vectors of these imaginary frequencies are fairly low, so perturbing structure in these directions does not affect electronic energy notably. In addition, according to the equation of ZPE, the contribution of these small imaginary frequencies to ZPE is also negligible. Therefore, the existence of small imaginary frequencies will not cause evident inaccuracy in the calculation of electronic energy and $U(0)$.

When calculating thermodynamic corrections to free energy, internal energy and enthalpy at finite temperature, these small imaginary frequencies need special consideration, because they have a significant contribution to entropy, heat capacity and the change in internal energy caused by heating from 0 K, so they cannot be simply ignored. In Section 3.2 of *Angew. Chem. Int. Ed.*, **2022**, e202205735, Grimme et al. suggest directly treating these small imaginary frequencies as real frequencies when using the aforementioned QRRHO model. Although this approach is obviously not strict, it is indeed reasonable to a certain extent. As shown in Figure 2 of *Chem. Eur. J.*, **18**, 9955 (2012), contributions of small frequencies to entropy are close. After eliminating the small imaginary frequencies by refining structure, one will definitely get same number of real frequencies with small wave numbers, so directly replacing the small imaginary frequencies with real frequencies of same magnitude may be an acceptable approximation, perhaps much better than completely ignoring their contributions. In addition, as shown in Figure 1 of *J. Phys. Chem.*, **100**, 16502 (1996), in the low-frequency range, the change in internal energy caused by heating from 0 K to expected temperature is insensitive to frequency, which also somewhat justified the treatment of setting the small imaginary frequencies as real ones. Note that this treatment is never suitable for the case when using standard RRHO model ($\text{ilowfreq} = 0$), because entropy in low-frequency range is very sensitive to frequency value in this case.

In Shermo, imaginary frequencies with magnitude smaller than “imagfreq” parameter (cm^{-1}) in *settings.ini* will be treated as real frequencies. For example, if it is set to 50, then 34.9i cm^{-1} will be automatically replaced with 34.9 cm^{-1} after loading data from input file. Obviously, if “imagfreq” is set to 0, then no imaginary frequency will be set to real.

Unfortunately, according to my brief test in <http://sobereva.com/699> (a molecular complex with imaginary frequency of 6.4i cm^{-1}) with $\text{ilowfreq} = 2$ or 3, the treatment of setting the small imaginary frequencies to real ones does not reduce the difference in $G_{\text{corr}}(298.15\text{ K})$ between the case with and without the imaginary frequency, but increases notably the difference! So, setting

small imaginary frequencies to real ones can never be regarded as a robust treatment. It should be aware that the aforementioned of justification of this treatment is evidently too optimistic, actually during refining the structure from that with imaginary frequencies to exact minimum, not only the imaginary frequencies change, but also values of many real low frequencies will change, which makes situation fairly complicated. However, according to theoretical analysis and my test, setting the small imaginary frequencies to real ones is beneficial for calculating U_{corr} when there are small imaginary frequencies. See <http://sobereva.com/699> for detailed discussion.

A.5 Electron contribution

Contribution to thermodynamic data from electrons comes from two parts, the electronic energy and electron excitation. The contribution due to the latter one is given below, where ε_i is electronic energy level i with respect to ground state, and g_i is degeneracy of level i .

$$q_{\text{ele}} = \sum_{i=0}^{\infty} g_i e^{-\varepsilon_i/(kT)}$$

$$U(T)_{\text{ele}} - U(0)_{\text{ele}} = RT \frac{\sum_i g_i [\varepsilon_i / (kT)] e^{-\varepsilon_i/(kT)}}{\sum_i g_i e^{-\varepsilon_i/(kT)}}$$

$$S_{\text{ele}} = R \ln \left(\sum_i g_i e^{-\varepsilon_i/(kT)} \right) + R \frac{\sum_i g_i [\varepsilon_i / (kT)] e^{-\varepsilon_i/(kT)}}{\sum_i g_i e^{-\varepsilon_i/(kT)}}$$

$$C_{V,\text{ele}} = R \frac{\sum_i g_i [\varepsilon_i / (kT)]^2 e^{-\varepsilon_i/(kT)}}{\sum_i g_i e^{-\varepsilon_i/(kT)}} - R \left\{ \frac{\sum_i g_i [\varepsilon_i / (kT)] e^{-\varepsilon_i/(kT)}}{\sum_i g_i e^{-\varepsilon_i/(kT)}} \right\}^2$$

At room temperature, only the states with excitation energy less than a few tenths of an eV may have notable contribution to thermodynamic data, therefore, contribution of electronic excitations is usually not explicitly taken into account since the excitation energies are commonly one order of magnitude higher (except for the cases that energy splitting is caused by spin-orbit coupling, the magnitude of lowest excitation energy in this situation is usually about 0.01 eV).

If ground state is degenerate, electronic contribution to entropy will be significant and never be ignored. If the degeneracy of ground state is g_0 , then the above expressions can be reduced to

$$q_{\text{ele}} = g_0$$

$$S_{\text{ele}} = R \ln g_0$$

As an instance, a doublet radical has spin multiplicity of 2, thus electronic contribution to entropy should be $R \ln 2 = 5.76 \text{ J} \cdot (\text{mol} \cdot \text{K})^{-1}$. Note that ground state degeneracy does not contribute to U and C_V .

A.6 Conformation weighting

Flexible molecules usually have many thermal accessible conformations (*i.e.* their relative free energies with respect to the lowest one is about the order of magnitude of RT , which corresponds to 2.48 kJ/mol at standard condition). Similarly, molecular complexes usually have many thermal accessible configurations. Appropriate consideration of the conformers/configurations is crucial for

obtaining reliable and meaningful thermodynamic quantities for these kinds of systems.

The conformation (or configuration, similarly hereinafter) weighted U , H and C_V/C_P can be easily obtained via the following way

$$U = \sum_i^{\text{conf}} p_i U_i \quad H = \sum_i^{\text{conf}} p_i H_i \quad C = \sum_i^{\text{conf}} p_i C_i$$

where p_i stands for Boltzmann weight of conformer i .

The way of calculating conformation weighted entropy is different to other thermodynamic quantities, because an additional term, the conformation entropy (S^{conf}), must be taken into account

$$S = S^{\text{conf}} + \sum_i^{\text{conf}} p_i S_i$$

$$S^{\text{conf}} = -R \sum_i^{\text{conf}} p_i \ln p_i$$

The S^{conf} comes from the interconversion between various conformations.

The conformation weighted Gibbs free energy can be obtained in usual way based on conformation weighted H and S , namely $G=H-TS$.

The Boltzmann weights can be calculated straightforwardly via relative Gibbs free energy

$$p_i = \frac{e^{-\Delta G_i/RT}}{\sum_j e^{-\Delta G_j/RT}} \times 100\%$$

$$\Delta G_i = G_i - G_{\text{lowest}}$$

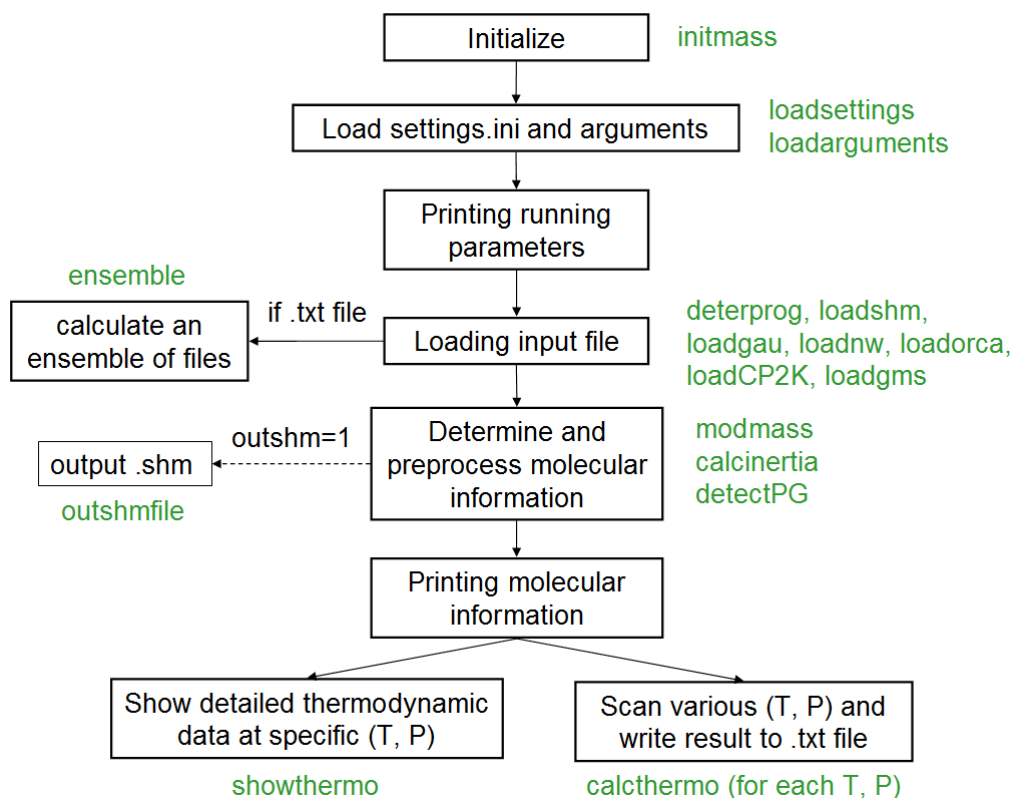
where G_{lowest} is the lowest Gibbs free energy of all considered conformers.

Appendix 2: Structure and subroutines of Shermo

In order to facilitate professional users to easily extend the functionality of Shermo, framework and subroutines of source code of Shermo are described in this appendix.

The source code files involved in Shermo are described as follows

Shermo.f90: Containing main program "Shermo", as well as the module "defvar" defining global variables and various constants. The flow chart of "Shermo" is shown below. The green texts indicate the subroutines involved in the corresponding stage



calc.f90: Containing various subroutines of calculating thermodynamic quantities

- *showthermo*: Calculate and print detailed thermodynamic data at present (T , P)
- *calcthermo*: Calculate and return thermodynamic data for given (T , P), invoked by both Shermo main program and "ensemble"
- *ensemble*: Load an ensemble of input files, calculate and print thermodynamic data for each of them and conformational weighted data at present (T , P)
- *elecontri*: Calculate thermodynamic contribution of electron levels. Employed by "showthermo" and "calcthermo"
- *getvibcontri*: Get contribution of a vibration mode to ZPE, $U(T)-U(0)$, C_V , S . Employed by "showthermo" and "calcthermo"
- *calcinertia*: Calculate moment of inertia matrix and its eigenvalues
- *getGconc*: Calculate present concentration, return concentration specified by "conc" parameter, and calculate variation of G due to concentration change.

atommass.f90: Defining the subroutine "initmass", which sets isotope masses, compositions,

and finally evaluate element masses

loadfile.f90: Containing subroutines to load output file frequency analysis task of various programs:

- loadCP2K: Load CP2K output file
- loadgau: Load Gaussian output file
- loadnw: Load NWChem output file
- loadgms: Load GAMESS-US output file
- loadorca: Load ORCA output file
- loadxtb: Load *g98.out* generated by xtb
- loadshm: Load .shm file

sub.f90: Containing a variety of subroutines involved in Shermo code

- loadarguments: Load arguments from command-line
- loadsettings: Load settings.ini
- modmass: Load modification on default atomic masses from settings.ini
- setatmmass: Set mass for all atom
- deterprog: Detect what program generated the current input file

symmetry.f90: Containing subroutines that deal with symmetry and point group

- detectPG: Detecting point group, will call PG_eqvatm
- PG_eqvatm: Get point group by invoking adapted subroutines of SYVA code
- PGlabel2rotsym: Convert point group label to rotational symmetry number

SYVA.F: Modified SYVA code. SYVA was originally released as a stand-alone program for determining all symmetry elements and symmetry operations of rigid molecules

util.f90: Utility subroutines shared by Shermo subroutines

- diagmat: Use Jacobi method to diagonalize matrix
- skiplines: Skip specific number of lines in specific file ID
- readaftersign: Read float data after the last specific sign from inputted string
- readaftersign_int: The same as "readaftersign", but for reading integer
- loclabel: Locate the line where the specific label first appears in the given file ID
- loclabelfinal: Locate to the final specific label, and meantime returns the number of matches
- sortr8: Sort values (double precision float) from small to large by bubble method