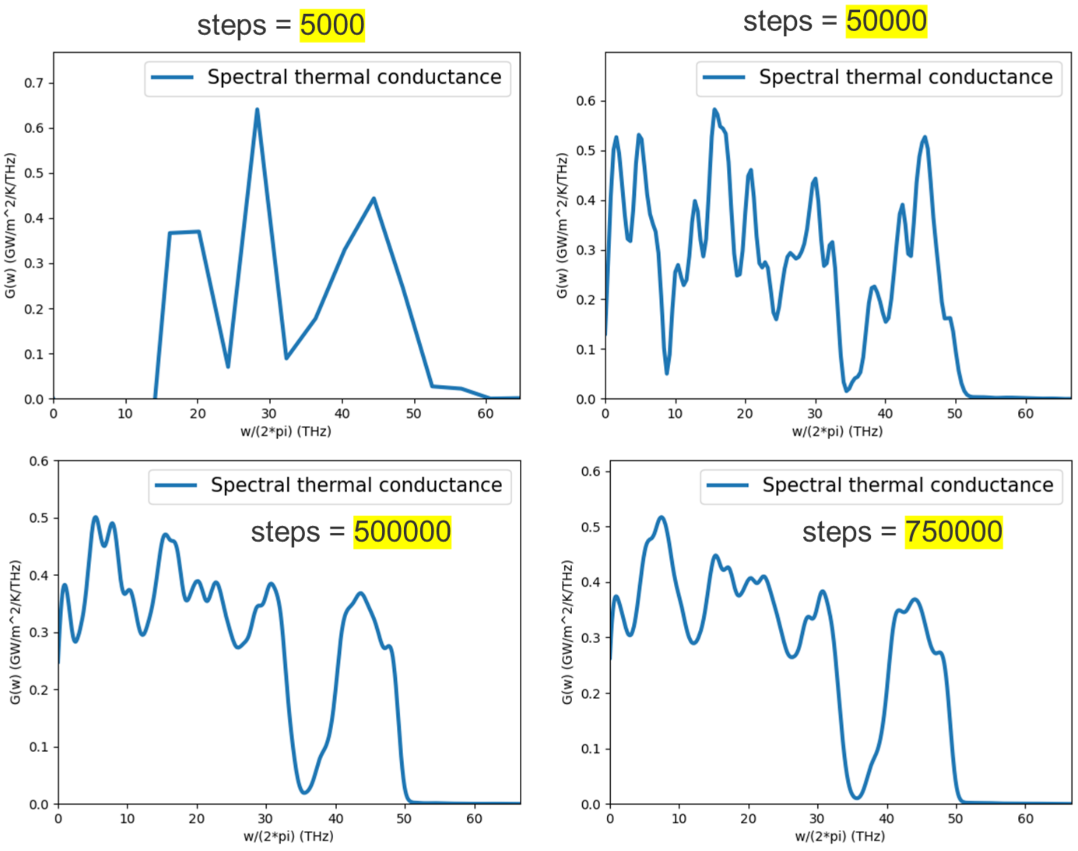
Here I provide a detailed test for several key parameters in the SHC calculation process.

We provide the following input parameter:

simulation_step, NChunks, frequency_window_width (for smooth).

(1) We first test simulation steps (The default parameter is 500000.):

simulation_step = 5000~750000, NChunks = 10, frequency_window_width = 4e12



steps = 5000

steps = 50000

steps = 500000

steps = 750000

Tips:

Since we have the rules following pseudo-code:

```
self.chunkSize = int(self.steps/self.dn/self.NChunks)
self.oms_fft=np.fft.rfftfreq(self.chunkSize,d=self.sampleTimestep) * 2 * np.pi
Nfreqs = np.size(self.oms_fft)
```

So more simulation time will result in a larger chunkSize, a larger chunkSize will have more frequency points throughout the frequency range, and the smoother the image will be. Of course, this is while maintaining the size of the NChunks.

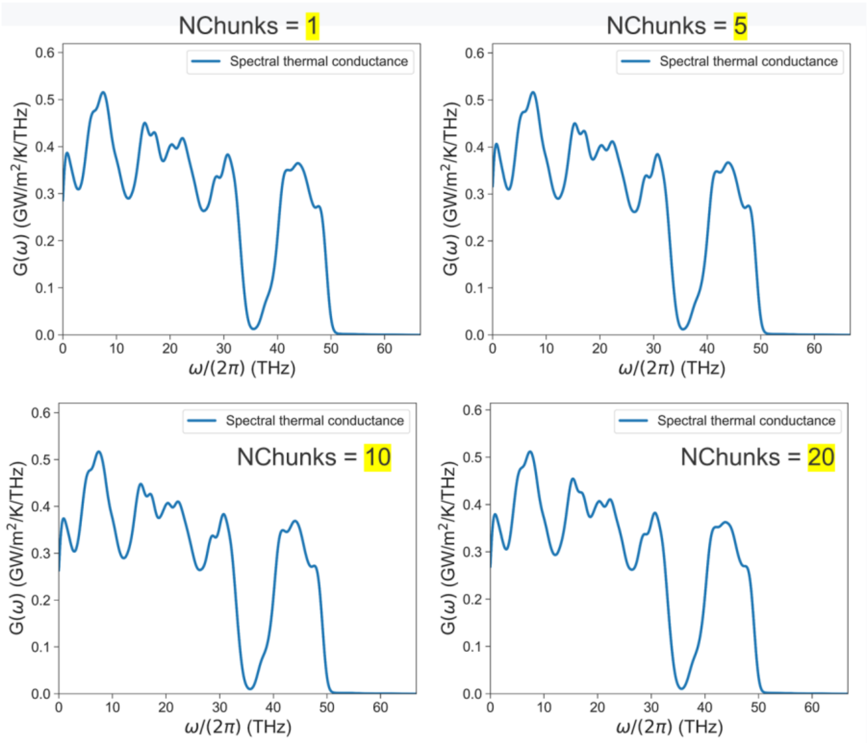
Sampling frequencies = Nfreqs = Step/dn/NChunks/2 + 1.

(2) Then, we test the NChunks:

self.steps = 750000, self.NChunks = 1~20,
frequency_window_width = 4e12

Since the total number of frames is already large enough, we divide the total number of frames into multiple blocks, and the different blocks have little effect on the SHC.

If your total number of frames is small, then you need to be careful not to divide the blocks too large. The default parameter is 10.

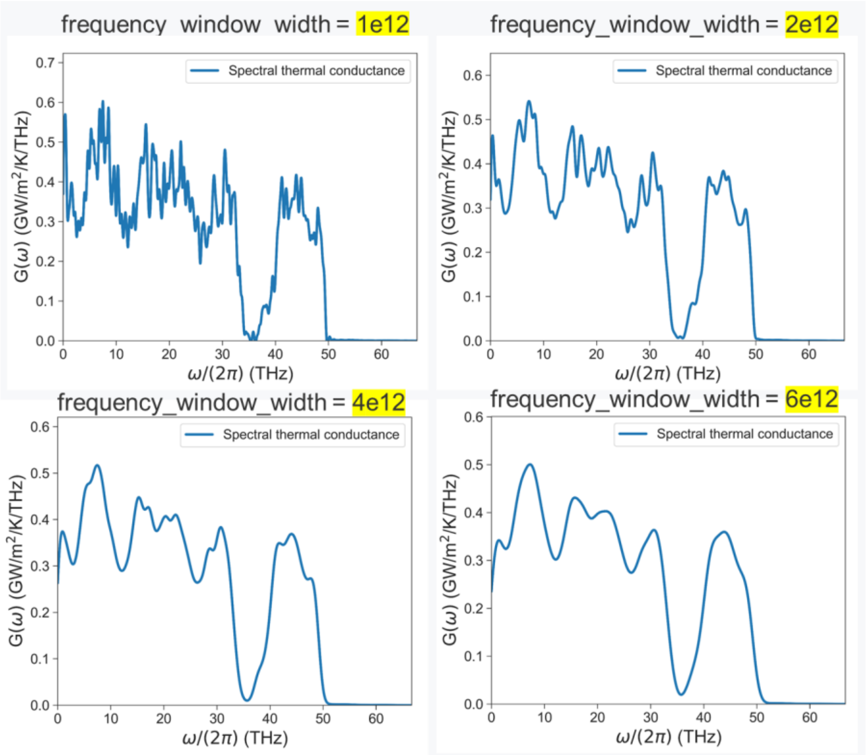


(3) Then, we test frequency_window_width:

self.steps = 750000, self.NChunks = 10,

frequency_window_width = 1e12~6e12.

The larger the width of the window function, the smoother the curve, the lower the resolution, and the less detail. The default parameter is 6e12.



For the generating force constant parameter hstep, the default is 0.1. For the system tested, a change from 0.01 to 0.1 does not affect the SHC. However, when the interface thermal conductance is small, the interaction is weak, and the dispersion force is included, this parameter needs to be increased.