

# Density of States Analysis Example

- Given sample and empty-can data, compute phonon DOS
- To use this notebook, first click jupyter menu File->Make a copy
- Click the title of the copied jupyter notebook and change it to a new title
- Start executing cells

## Summary of processing steps

- Gather experimental information and experimental raw data
- Reduce raw data to  $S(Q,E)$
- Convert  $S(Q,E)$  to DOS

## Preparation

Create a new working directory and change into it.

Please modify the following path to suit your need!

```
In [1]: workdir = '/SNS/users/lj7/reduction/ARCS/getdos-demo-test'
!mkdir -p {workdir}
%cd {workdir}

/SNS/users/lj7/reduction/ARCS/getdos-demo-test
```

Get tools ready

```
In [2]: import os, numpy as np
import histogram.hdf as hh, histogram as H
from matplotlib import pyplot as plt
%matplotlib notebook
# %matplotlib inline

from multiphonon.sqe import plot as plot_sqe
from multiphonon.ui.getdos import Context, NxsWizardStart, QEGridWizardStart, GetDOSWizardStart
```

Create a context for getdos

```
In [5]: context=Context()
```

If you want to reuse a previously-saved context, please uncomment the following cell and execute

```
In [7]: context.from_yaml('./getdos2-context.yaml')
```

## Experimental data and condition

Phonon Density of States (DOS) can be obtained from inelastic neutron scattering (INS) spectrum. This notebook allows for extracting DOS from INS spectrum measured at the ARCS instrument at SNS. To start, we need data files measured for the sample and the empty can, as well as experimental conditions such as incident energy and sample temperature. The following wizard help you go through these steps.

```
In [4]: NxsWizardStart(context).show()
```

× Please choose the sample nxs or nxspe file:

Select a file

Select

En

Obtain S

S(Q,E) spectr

Here we defin

ARCS_83912_event.nxs	Mon Aug 1 19:21:16 2016
ARCS_83912_histo.nxs	Mon Aug 1 19:18:25 2016
ARCS_83913_event.nxs	Mon Aug 1 19:54:34 2016
ARCS_83913_histo.nxs	Mon Aug 1 19:51:46 2016
ARCS_83914_event.nxs	Mon Aug 1 20:24:06 2016
ARCS_83914_histo.nxs	Mon Aug 1 20:21:15 2016
ARCS_83915_event.nxs	Mon Aug 1 22:30:57 2016
ARCS_83915_histo.nxs	Mon Aug 1 22:28:06 2016
ARCS_83916_event.nxs	Tue Aug 2 01:03:00 2016
ARCS_83916_histo.nxs	Tue Aug 2 01:00:09 2016
ARCS_83917_event.nxs	Tue Aug 2 01:05:00 2016

Example datasets:

- samplnxs = "/SNS/ARCS/2014\_1\_18\_CAL/0/47435/NeXus/ARCS\_47435\_event.nxs"
- mtnxs = Skip
- Ei=80
- T=300

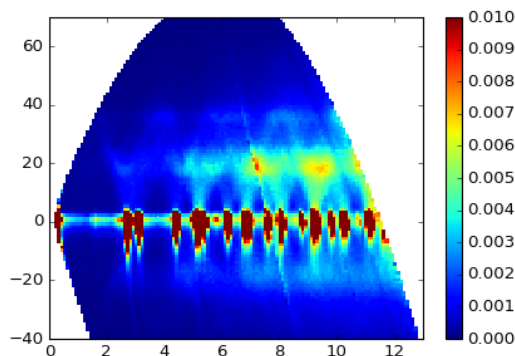
```
In [8]: NxsWizardStart(context).show()
```

Save configuration so you can reuse it

```
In [ ]: context.to_yaml('./getdos2-context.yaml')
```

## Obtain $S(Q,E)$

$S(Q,E)$  spectra for both the sample and the empty can is the starting point for getdos processing. Here is an example:



Run the following wizard to define the E and Q axes so that  $S(Q,E)$  spectra can be obtained the INS raw data.

```
In [9]: QEGridWizardStart(context).show()
```

Parameters are saved in the work dir. Uncomment the script below to see.

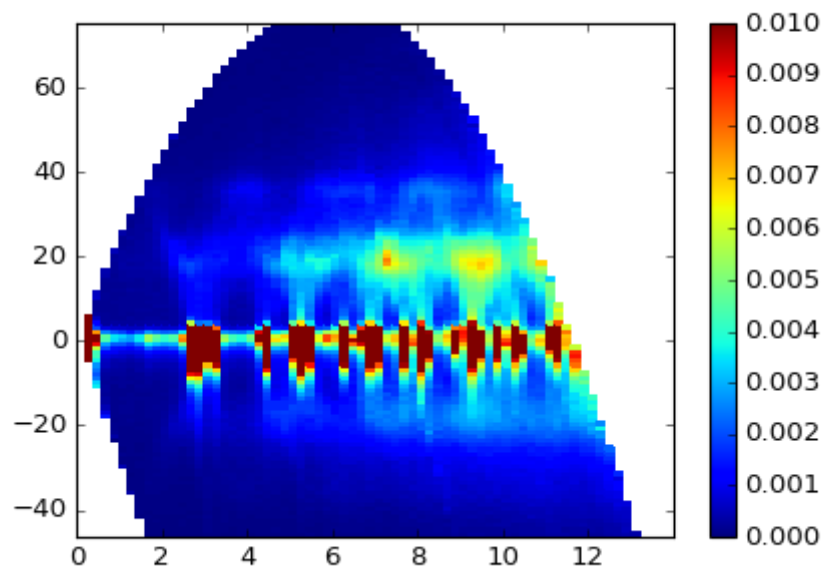
```
In [ ]: %%script bash
        # ls work/
        # cat work/raw2iqe-sample.params
```

Plot sample IQE

```
In [10]: iqe = hh.load('work/iqe.h5')

/home/lj7/miniconda2/envs/dev-mph/lib/python2.7/site-packages/histogram/hdf/Loader.py:129: FutureWarning: elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison
    if 'storage' in list(dataGroup): # this uses the 'storage' convention
```

```
In [11]: plt.figure(figsize=(6,4))
plot_sqe(iqe)
# plt.xlim(0, 11)
plt.clim(0, 1e-2)
```

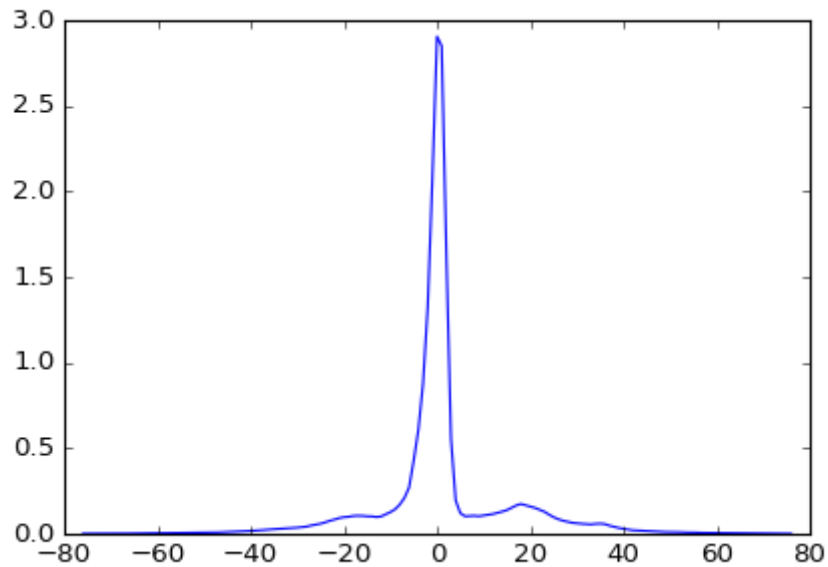


You can improve the Q,E grid parameters if you like, by re-executing the above cell of

```
QEGridWizardStart(context).show()
```

Plot I(E)

```
In [12]: iqe2 = iqe.copy()
I = iqe2.I; I[I!=I] = 0 # remove NaNs
IE = iqe2.sum('Q') # sum over Q
plt.figure(figsize=(6,4))
plt.plot(IE.energy, IE.I)
```



```
Out[12]: [<matplotlib.lines.Line2D at 0x7ff983c08a90>]
```

The plots above provide clues to selecting parameters for the getdos procedure

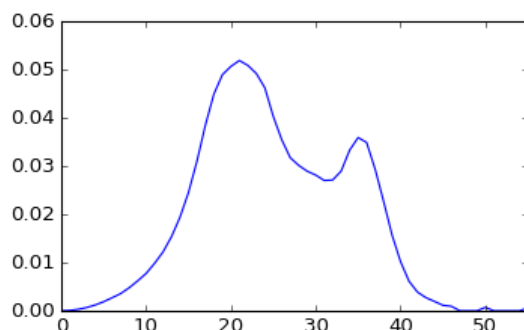
Save configuration so you can reuse it

```
In [ ]: context.to_yaml('./getdos2-context.yaml')
```

## Run GetDOS

DOS will be obtained from SQE by an iterative procedure where multiphonon and multiple scattering corrections are applied to the measured SQE spectrum, assuming incoherent approximation, and the corrected spectrum is then converted to DOS.

An example DOS plot:



```
In [13]: GetDOSWizStart(context).show()
```

Save context

```
In [ ]: context.to_yaml('./getdos2-context.yaml')
```

Print context

```
In [ ]: print context
```

## Check output

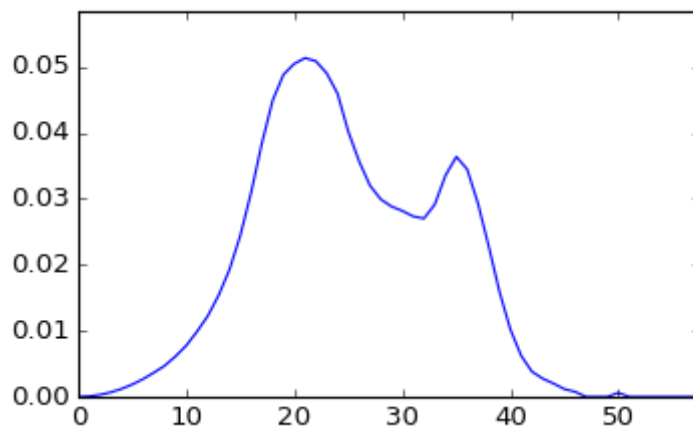
Results are saved in "work" directory

```
In [14]: ls work/
```

<b>dirddydos/</b>	iqe.nxs	raw2iqe-MT.params	<b>rou</b>
<b>nd-2/</b>			
final-dos.h5	mt-iqe.h5	raw2iqe-sample.params	<b>rou</b>
<b>nd-3/</b>			
getdos-kargs.yaml	mt-iqe.nxs	residual_E-posE.h5	<b>rou</b>
<b>nd-4/</b>			
I_E-exp-posE.h5	<b>plot_dos_iteration.py*</b>	<b>round-0/</b>	<b>rou</b>
<b>nd-5/</b>			
iqe.h5	<b>plot_residual.py*</b>	<b>round-1/</b>	

Plot the final result for DOS

```
In [15]: dos = hh.load('work/final-dos.h5')
plt.figure(figsize=(5,3))
plt.plot(dos.E, dos.I)
plt.xlabel('Energy (meV)')
# plt.xlim(0, 30)
```

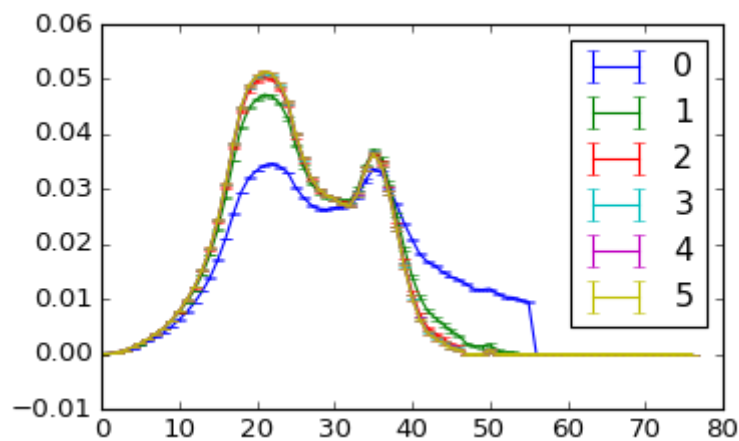


Out[15]: <matplotlib.text.Text at 0x7ff983ba7410>

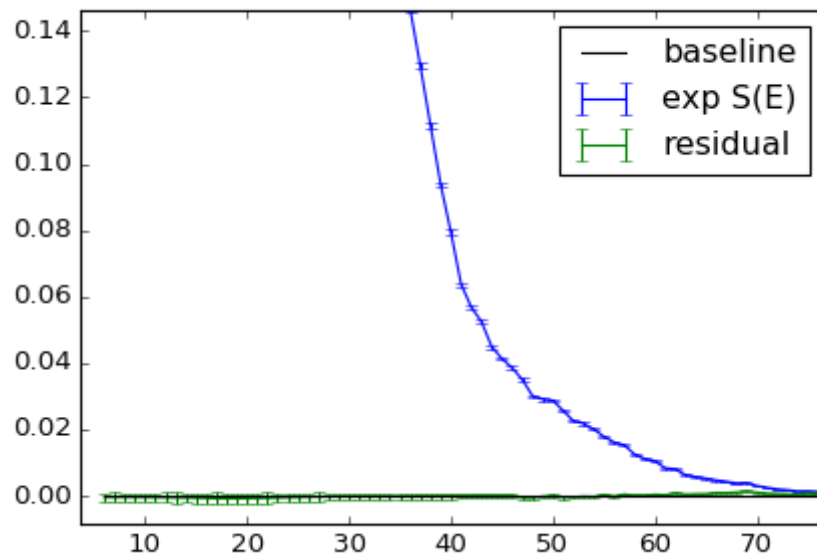
More plotting utils are available

```
In [16]: from multiphonon.backward import plotutils as pu
```

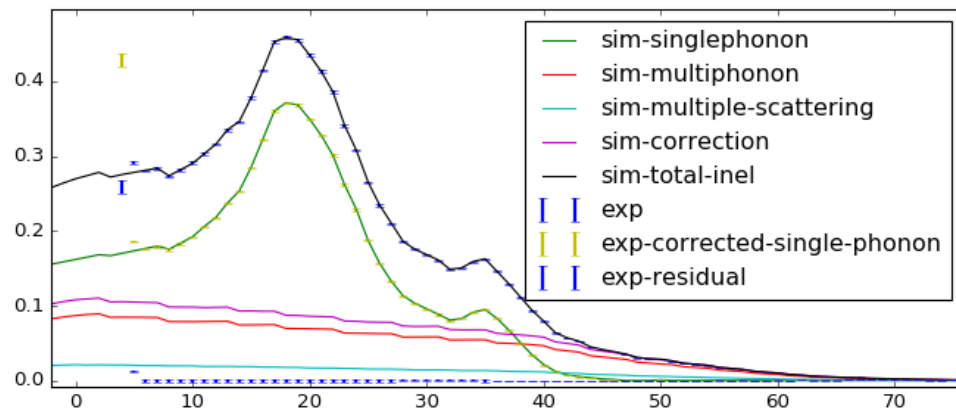
```
In [17]: plt.figure(figsize=(5,3))
pu.plot_dos_iteration('work/')
```



```
In [18]: plt.figure(figsize=(6,4))
         pu.plot_residual('work/')
```



```
In [19]: plt.figure(figsize=(10, 4))
         pu.plot_intermediate_result_se('work/round-3')
```



```
In [ ]:
```



