1 Q-scan on coherent network SNR

Q transform time-frequency spectrogram can show the gravitational wave signal track clearly. Here the template filtering is applied before the time-frequency decomposition using Q transform technique. The gravitational wave should have coherent property through multi-detector signals.This will enhance the data power by 1.4 times.

1.1 Template filtering and signal-to-noise ratio (SNR)

The gravitational wave search method is mainly based on matched-filtering. When gravitational wave was detected, the output of detector will be a time series of strain $s(t)$.

$$s(t) = n(t) + h(t) \tag{1}$$

Where $n(t)$ is the noise of detector, and the one-side power spectral density (PSD) can be expressed by the following expectation,

$$E(\tilde{n}(f)\tilde{n}(f')) = \frac{1}{2}S_h(f)\delta(f - f') \tag{2}$$

where $\tilde{n}$ means the fourier transformation of detector noise $n(t)$.

The matched filtering signal to noise ratio(SNR) is defined as:

$$\rho = \frac{< s|h >}{\sqrt{< h|h >}} \tag{3}$$

The inner product $< \cdot|\cdot >$ is defined by,

$$< A|B >= \int_{-\infty}^{\infty} \frac{\tilde{A}(f)\tilde{B}^*(f) + \tilde{A}^*(f)\tilde{B}(f)}{S_h(|f|)} df \tag{4}$$

and * means the complex conjugate.

1.2 Coherent network SNR

The recieved signal by detector can be written by the antenna pattern F_p, F_c of the detector d:

$$\begin{align} h_d(t) &= F_{p,d}h_p(t) + F_{c,d}h_c(t) \tag{5} \\ &= [F_{p,d}\ F_{c,d}] \begin{bmatrix} h_p \\ h_c \end{bmatrix} \tag{6} \\ &= [G_{p,d}\ G_{c,d}]\, \mathbf{A}_d \begin{bmatrix} h_p \\ h_c \end{bmatrix} \tag{7} \\ &= GA_p h_p + GA_c h_c \tag{8} \end{align}$$

The template h_p, h_c contains the intrinsic parameters $\theta = (m1, m2, \mathbf{s}_1, \mathbf{s}_2)$.The antenna pattern F_p, F_c are functions of extrinsic parameters(the polarization ψ, inclination ι, initial orbit phase ϕ_c, arrival time t_a, distance D, and sky location α, δ). It can be split into a matrix $\mathbf{A}$ which contains extrinsic parameters except arrival time t_a and sky location (α, δ), and G_p, G_c.

The likelihood ratio for single detector d is

$$ln\mathcal{L}_d =< s_d|h_d > -\frac{1}{2} < h_d|h_d > \tag{9}$$

Similarly, the joint network likelihood ratio is

$$ln\mathcal{L} =< \mathbf{s}|\mathbf{h} > -\frac{1}{2} < \mathbf{h}|\mathbf{h} > \tag{10}$$

where $\mathbf{s}, \mathbf{h}$ are the combination of all detectors:

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \\ ... \end{bmatrix}, \mathbf{h} = \begin{bmatrix} h_1 \\ h_2 \\ ... \end{bmatrix} \mathbf{G} = \begin{bmatrix} G_{p,1} & G_{c,1} \\ G_{p,2} & G_{c,2} \\ ... & ... \end{bmatrix} \tag{11}$$

And the inner product rules for the list of data is similar to matrix multiplication.

Now, we rewrite the network likelihood ratio as

$$ln\mathcal{L} = \sum_{x=p,c} ln\mathcal{L}_x \tag{12}$$

$$= \sum_{x=p,c} (< \mathbf{s}^T|\mathbf{G}\mathbf{A}_x h_x > -\frac{1}{2} < \mathbf{A}_x^T \mathbf{G} h_x|\mathbf{G}\mathbf{A}_x h_x >) \tag{13}$$

Follow [?], maximization over $\mathbf{A}_x$ by $\partial ln\mathcal{L}/\partial \mathbf{A}_x = 0$, we will get maximal network likelihood ratio:

$$ln\mathcal{L} =|\mathbf{I}\mathbf{U}^T < \bar{\mathbf{H}}_c + i\bar{\mathbf{H}}_p|\mathbf{d} > |^2 \tag{14}$$

$$\mathbf{I} = diag(1, 1, 0, 0...) \tag{15}$$

Finally we get the coherent SNR $\rho_c(t, \alpha, \delta)$ by

$$\rho_c(t, \alpha, \delta) = \max_{\mathbf{A},\theta} ln\mathcal{L} \tag{16}$$

1.3 Q transform

Here is the equation for short time Fourier transform (STFT):

$$STFT\{x(t)\}(\tau, \omega) = \int_{-\infty}^{\infty} x(t)\Psi(t - \tau)e^{-jwt}dt \tag{17}$$

The uncertainty principle stats that $\delta t \delta f >?$.

The Q transform(QT) is a modified STFT that the window function varies inversely with the frequency. Can be expressed as,

$$QT\{x(t)\}(\tau, f) = \tag{18}$$

$$\int_{-\infty}^{\infty} df' \int_{-\infty}^{\infty} dt' x(t') e^{i2\pi f'(\tau - t')} \Psi(f'; f, Q) \tag{19}$$

We use the bi-square window for the Q transform window function:

$$\Psi(x; f, Q) \tag{20}$$

$$= \begin{cases} -\frac{Q^2}{N_Q}(x - f_l)(x - f_r),\ f_l < x < f_r \\ 0,\ else \end{cases} \tag{21}$$

where $f_l = (1 - \frac{\sqrt{N_Q}}{Q})f$, $f_r = (1 + \frac{\sqrt{N_Q}}{Q})f$ and N_Q is a normalization constant, we choose $N_Q = 11$.

Now we define the Q transform filtering SNR $\rho_Q(t, f)$ by:

$$\rho_Q(t, f) = \tag{22}$$

$$\int_{-\infty}^{\infty} df' \frac{\tilde{s}(f')\tilde{h}^*(f') + \tilde{s}^*(f')\tilde{h}(f')}{S_h(f')} e^{i2\pi t f'} \Psi(f'; f, Q) \tag{23}$$

From which we can define the frequency-segmented template:

$$\tilde{h}_{f,Q}(f') = \tilde{h}(f')\Psi(f'; f, Q) \tag{24}$$

And the likelihood ratio for this frequency in detector d:

$$ln\mathcal{L}_{f,Q,d} =< s_d|h_{f,Q,d} > -\frac{1}{2} < h_{f,Q,d}|h_{f,Q,d} > \tag{25}$$

Then, similarly, we can follow section II to construct joint likelihood for frequency f.

$$ln\mathcal{L}_{f,Q} =< \mathbf{s}|\mathbf{h}_{f,Q} > -\frac{1}{2} < \mathbf{h}_{f,Q}|\mathbf{h}_{f,Q} > \tag{26}$$

The definition method of bold symbol is consistent with section II.2. And follow the same method in section II.2, using $\mathbf{h}_{f,Q}$ to replace $\mathbf{h}$, we will get joint frequency-segmented likelihood ratio:

$$ln\mathcal{L}_{f,Q} =|\mathbf{I}\mathbf{U}^T < \bar{\mathbf{H}}_{f,Q,c} + i\bar{\mathbf{H}}_{f,Q,p}|\mathbf{d} > |^2 \tag{27}$$

$$\mathbf{I} = diag(1, 1, 0, 0...) \tag{28}$$

The method to construct $\bar{\mathbf{H}}_{f,Q,x}$ is replacing $\mathbf{h}$ with $\mathbf{h}_{f,Q}$. And the coherent network SNR spectrum $\rho_c(t, f, Q)$

$$\rho_c(t, f, \alpha, \delta, Q) = \max_{\mathbf{A},\theta} ln\mathcal{L}_{f,Q} \tag{29}$$

The whole process is, firstly, follow the steps in section II.2 to get the coherent SNR $\rho_c(t, \alpha, \delta)$.Then, maximizing over sky location to get the maximal coherent SNR $\rho_{max} = \max_{\alpha,\delta} \rho_c(t, \alpha, \delta)$ and get the most-likely sky location (α_c, δ_c). Finally, calculating coherent SNR spectrum $\rho_c(t, f, \alpha_c, \delta_c, Q)$

1.4 Track integration significance

The coherent SNR spectrum will amplify the correlation of a specific gravitational wave signal. When a real gravitational wave signal come, the coherent spectrum will show the time-frequency track $f_{track}(t)$ clearly. Now we construct a numerical value $\rho_{track}(\tau)$ to describe the significance of this track.

$$\rho_{track}(\tau) = \frac{\int_{f_{track}(t-\tau)} \rho_c(t, f, Q) df dt}{\int_{f_{track}(t-\tau)} df dt} \tag{30}$$

Which means the average SNR alone the gravitational wave signal track. Where τ is the end time of the track. For a gravitational event, we use $\tau = t_c$, which is the event trigger end time of the event. We call $\rho_{track}(tc)$ the forground of the event. For comparison we will also do multiple sampling for different τ, which we call background.

In the program, we calculate the background by making time shift several times, in each time shift, we calculate the background track integration randomly.

2 Program options description

2.1 Installation

github repository: https://github.com/Shallyn/pygwcoh.git
The format of the program is a python module. Firstly git clone the repository:

```
git clone https://github.com/Shallyn/pygwcoh.git
```

Then, go to Cextension directory:

```
cd pygwcoh/Cextension
```

Install Cextension module:

```
./mkconf.sh
./configure
make
```

Before configuring, make sure you have add your gsl, python and numpy head file path to C_INCLUDE_PATH or CFLAGS.

The whole module relies on several python basic module, and two LIGO module: gwpy and glue. You can use such .py file to call the main function:

```
import re
import sys
from pygwcoh.exe import main

if __name__ == '__main__':
    sys.exit(main())
```

also, you'd better add the path of pygwcoh to PYTHONPATH.

2.2 Options

–graceid=GRACEID GraceDB event ID, if added, will load source parameters from such event.
–Sgraceid=SGRACEID GraceDB superevent ID, if added, will load source parameters from the preferred event of such superevent.

–sample-rate=FS sample rate[4096]
If you have not specified gracedb options, you can use following options to generate template.
–gps=GPS gps time of event.
–m1=M1 component mass1 for template generation
–m2=M2 component mass2 for template generation
–s1z=S1Z component spin_z for template generation
–s2z=S2Z component spin_z for template generation
–fini=FINI initial frequency for template generation
–approx=APPROX waveform for template generation

The program has several tools to load data.
–cache=CACHE Data cache for data loading(Not sure this can work)
or using gwpy, if so, just specify how much time duration you need.
–stepforward=STEPFORWARD [30]
–stepback=STEPBACK [5]
These two options decide the data segment that you will get by gwpy.timeseries.Timeseries.

–channel=CHANNEL You can specify the channel you need, you can choose GATED or CALIB, the program will load C00 data by using gwpy.[GATED]

–ref-psd=REF-PSD set reference psd for matched filtering.
–gaussian if added, will generating gaussian noise from given reference psd.

–injection if added, will make injection at gps-inj
–gps-inj=GPS-INJ the gps time for injection.
–m1-inj=M1-INJ
–m2-inj=M2-INJ
–s1z-INJ=S1Z-INJ
–s2z-INJ=S2Z-INJ
if you have not specified such parameters, the program will use the parameters of event.
–ra=RA right ascension of the injection event.
–de=DE declination of the injection event.

And some options for Q transformation.
–Q=Q Q value

–minf=MINF [20]
–maxf=MAXF [1200] frequency range.
–mismatch=MISMATCH [0.2] frequency window mismatch.