Preface

Fish for a needle in Galaxy

‘Fish for a needle in the ocean’ is a Chinese proverb, expressing the difficulty in looking for tininess in a grant area. For us, the ‘needle’ is exotic physic, and the ‘ocean’ is the universe. However, other than ‘universe’, I would rather use ‘galaxy’ in the title, because ‘galaxy’ means ‘river of sliver’ in Chinese.(1) So now we are fishing for the signal in this sliver river. I wish this romance could be a relief for us during the long waiting before we were truly stabbed by the detection of physic in the dark.

In this report, I will first share my understanding of data simulation and analysis. Then comes a thorough description of algorithm applied in each section of the work. The report will also include the introduction of realization of the algorithm in Python. Due to the limited period of my work, there would be much to be modified or improved, which will also be explained in details in the report. I hope it could be a framework for the next step.

My e-mail address is zyzoli.mail.ustc.edu.cn. Please contact if any problem or idea with this report. I will keep on updating this report according to feedback. Updates will be recorded in update log in appendix near the end of this report. You can request the up-to-date report or any data, figures, codes shown in the report from me.

Yu

Summer, 2019

in Krakow

(1) This romantic name came from imagination inspired by milky way, the hazy band of light in starry nights.

Chapter 1 Introduction

If you are beginner in simulation and GNOME work, this introduction could be confusing for you. You can just browse on this chapter and read carefully from chapter 2.

**1.1 What is Simulation**

Simulation means imitation[1, wiki]. Here in this report, data simulation is defined as generating data based on the characteristics of a model. Maybe it’s appropriate to use ‘modeling and simulation’. But from my point of view, ‘simulation’ alone has included the procedure of modeling. Let’s waste no time in vocabular game and move on. Imagine that the existence of Laplace’s demon is authentic. In traditional physic experiments, we are trying every best to figure out how this demon calculate the movement of the world. While in simulation, it’s us who make the rules. It seems ridicules that these once rule seekers(us) try to become rule makers. This ridicules misunderstanding is definitely wrong, but actually helps in pushing forward our work. So here comes the question:

**1.2 Why do simulation?**

This is not only a question to be answered, but also a question to be questioned. My first response to this question is, why not do simulation? We used to examine our theory in the laboratory or on paper. Simulation could be the third kind, especially beneficial when we could not push forward lab and paper work.

If Galileo and Newton were given a PC with MATLAB or Mathematica, physic might be mainly developed in virtual lab and discrete math. (Just joking. I think they would first dismantling the PC and then become engineers and develop computer science. )

During this summer, I found three major problems hindering us from the detection of real signal. One is huge background noise. It’s easy to understand the first problem: previous physicists discovered signals above the noise, then the signals beneath the noise are left to us. The second problem is nonlinear relationships in processing. We frequently come into nonlinear transformation or nonlinear equations, which make perfect theoretical derivation impossible. About this point, I would explain it in details later. Please forget this, since it’s quite ambiguous for now. The third barrier is little knowledge about exotic physic, or dark matter and dark energy. We have so little idea about its form of interaction that we have to make many assumptions and then examine these.

Hence, my second response to ‘why do simulation’ is that, we are forced to do it. We have to rely on simulating the real world to test our methods, correct our predictions or assumptions, and finally, have some expectations in mind.

What’s more, I hope such artificial detections help us adapt to the joy of discovery gradually, preventing us from being over-exciting when real signal is detected.

**Game of Probability**

We know two games of probability: gambling and quantum mechanism. In data processing, we are not talking about quantum mechanism (neither gambling). We would focus on the science closely related to gambling: Probability Theory and Statistics.

For signal detection, I think it’s reasonable for us to reach on this: the results should always come with confidence interval, like ‘we are 99% sure that, we find a signal within a certain time range and within a certain energy range’, or ‘false alarm rate estimated to be less than 1 event per 203 000 years, equivalent to a significance greater than 5.1σ’[Observation of Gravitational Waves from a Binary Black Hole Merger]. In other words, what we are searching for should be a probability density distribution, which indicates the confidence that we find the signal in any interval.

As to how-to-do, I recommend Monte Carlo method. We repeat the experiment in simulation, with known input paraments. The result processed with statistics method would show us the probability density distribution.

Take excess power analysis as an example. How do we know the probability that a signal is buried beneath somewhere under the noise? We generate random noise of which characteristics match real noise’s. Then we insert a signal with certain amplitude, FWHM or any other factors. We repeat this simulated experiment, and finally discover how the result is like.

However, Monte Carlo method is blamed for its low convergence speed. Anyone should be careful with this before simulation.

**What is ‘zero result’ indeed?**

Let’s think about one more question. You may heard about this, ‘zero result is also a result’. When I heard about this first in high school, I thought this is nothing more than a relief for people like us. Does ‘zero result’ mean zero or blank? Of course NOT. By announcing ‘zero result’, the scientists are actually answering such a question: at which level of confidence, it’s impossible to detect signal with some certain characteristic. ‘Zero result’ not only sets the upper limit for detecting, but also implies the orientation of the next-step research.

**1.2 What’s a Good Simulation for GNOME**

The procedure......

Finally we are at the most exciting section. However, we should contain our emotion when doing simulation. It never matters whether we have any discovery in simulation. It only matters whether the simulation is rational. Details are explained as follows.

Firstly, we have to bare it in mind that what we are looking for is a probability distribution, but not a discovery. We don’t need to be happy when we find the injected signal, neither be depressed when the injected signal disappears after processing. The first thing we need to do is to repeat the experiment with random noise, and see the statistical result.

Secondly, it’s crucial to do rational simulation.

**1.3 For Developer**

This section is written for program developers.

The algorithms are all realized in Python 3.7. When it comes to large amount of calculations, I will display the running time of the program. My laptop is HP ENVY x360 Convertible 13-ag0xx, coming in AMD Ryzen 5 2500U with Radeon Vega Mobile Gfx 2.00GHz, 8GB RAM, 256GB SSD, Windows 10 Version 1903. All tests are run in Best Performance Mode when charging. My Python IDE is PyCharm, setting heap size to 890 MB.

I always trying to avoid using some uncommon Python packages for three reasons. First of all, I tried every method to install some LIGO packages but failed anyway. I am afraid that this could happen on others’ PC. The next reason is that some packages requires Python 2.7, which would not be maintained soon. The last reason is that I found no package catering to GNOME data processing. It’s better for comprehension to write the package on our own. However, LIGO packages are still important reference for us. I am jealous that they have so many experts in data processing.

As a student major in physics, I am far from professional in programing, not to mention that it has been just 2 months since I started with Python. I am deeply dependent on my little experience with C, C++, MATLAB and Mathematica. Codes are written in the most simple way without Python skill. If you are a skillful programmer or even professional, you can skip this section. If you are a non-professional like me, please read the following tips.

1. Always turn to common or well-known packages for help

When it comes to calculation, try to find a wheel in Python packages, instead of building the wheel on your own. Recommendation: NumPy, Matplotlib, Pandas.

2. Be careful with program running time

When your program consumes extremely long time (1 minute is extreme for one section in GNOME data processing), program would probably end up in failure. Don’t blame your PC first. Try to use functions from common packages. Try to use vectorization or parallel operations instead of loop structure.

3. 10^8 could be a threshold

For my laptop and Python IDE, 10^8 is a useful standard. For any calculation or amount of data, things are completely different in 10^7 and 10^8. When it reaches 10^8, it’s usual that we go into memory error and cooling problem.

The following figure shows the exponential growth of running time with linear growth of calculations.

[figure running time]

4. Python IDE

If you have just started with Python, I recommend PyCharm for your Python IDE choice. Its Community Edition (Free) is adequate for us. It’s comfortable and efficient to develop in a good IDE.

Chapter : Single-station Signal Identification

3.1 Introduction

Finally we are at detection stage. To distinguish signal detection in single and multi stations, ‘Identification’ is applied here to emphasize that we are considering whether a signal is authentic, while in multi-station detection, ‘Locating and Examination’ is used to fully describe two unique steps in detection. In practice, single-station signal identification constructs the basis for locating and examining signals in multi-station network.

Let’s focus on a single station first.

The problem at this stage is simple: We know ‘power’ distribution over time and frequency from excess power analysis. How much confidence do we have to conclude the existence of a signal? (Please always bare it in mind that, we are playing a game of probability. )

3.2 Algorithm

\subsection Signal Identification in Practice

[noise hist and noise+signal hist]

The only remaining problem is the criteria in the last step. In this report, we are handling signals slightly above or even beneath noise, where SNR is usually set to 1.0 or lower (Otherwise way too appearant signals would kill the fun in detection). We should have the least expectation for obvious signal in practice. Instead, we have to analyze the results in statistical way, for instance, in histogram.

\subsection Creating Background Reference under a certain set of parameter

Since we are able to simulate noise and signal responses, it’s natural to have the following algorithm. We first generate the noise series and then inject a signal into it. After that we do excess power analysis. Usually, we would find excess power cause by the injected signal. Such excessive power indicates the reality of something exotic.

There are two key points in the algorithm. One is the choices of parameters for excess power analysis. %The other is setting the criteria for ‘excessive’ or ‘exotic’ power.

\subsection 3.2.1 Optimal Parameters for Excess Power Analysis

There can’t be a more stupid idea than mine: try all sets of parameters and pick up the best one. However, I don’t recommend any clever way. There’s no conclusion in deciding or calculating the best choice. Also, I don’t recommend you look for any conclusion or formula in choosing parameters. I tried to find some ‘trends’ with simulation in deciding the optimal parameters. I did find some limits for parameters which are shown at the end of this subsection. But generally speaking, there’s no definite conclusion. Considering the quantity of all these parameters, it’s certainly time-consuming to seek a easy conclusion. By the way, I feel doubtful about existentiality of such ‘conclusion’.

Therefore, simulation is essential.

Let’s review on the input parameters in data processing.

segment length

segment stride

welch method number per segment

average method: default to ‘Exponential Moving Average(EMA)’

EMA factor

EMA window length

Here we introduce two more parameters.

frequency low-pass

frequency high-pass

In the last section, we kept all frequency bands on spectrum. However, real signals often respond in a narrow frequency band. We can focus on only the sensitive band of the excess power analysis result.

Let me illustrate the simulation with an example.

1. Generate Signal series

Considering the random noise, we can generate, say 100 siganal series, each with random noise and same injected signal. Of course, the random noise and injected signal should match the characteristics of the specific sensor.

2. Choose parameters:

segment length[s]: 2, 4, ...., 20 (10 different choices)

segment stride: as half of segment length

welch method number per segment: default 512

average method: default ‘Exponential Moving Average(EMA)’

EMA factors[s^-1]: 0.04 0.06, ..., 0.2 (10 different choices)

EMA window length: always 10\*segment length

low-pass[Hz]: 0

high-pass[Hz]: 200

So we have 10\*10=100 sets of parameters.

3. Simulation test

We test each set of parameters with each signal series. So we have to do 100\*100=10000 tests.

4. Select the optimal set of parameters

Finally, under some criteria, we pick up the best set of parameters for the identification in a specific sensor.

I believe the procedures of simulation are easy to grasp.

Even though there’s no definite conclusion in deciding parameters, there are some tips or precautions for reference. At last, the optimal parameters are decided by simualtion results.

1. Segment length

Large segment length could be a trouble in locating the signal in time domain. Imagine that we set length to 1000s, then it’s impossible for locating the signal at 10s scale. Additionally, we would get one more problem. More ‘power’ would accmulate in a larger segment, which makes the signal less appearant. What’s worse, the sensors may not be stable over a long time.

Little segment length will result in bad resolution. With a certain sampling rate, a smaller segment contains less data point, which limits the resolution in frequency. We can set the lower limit for segment length according to the desired resolution.

2. EMA factor

Greater factor would decrease the influence of more previous segments. Imagine that we set factor to 100, and segment stide to 2s. The weights(not normalized) of the previous segments would be 1, exp(-100), exp(-200)... so that only the closest one segment contributes to the whitening, which would probably increase the varance (just ‘probably’).

3. EMA window length

We have less confidence to choose a small window to whiten the signal series.

Large window usually leads to heavy burden for computing. Also, considering the stability of the sensors, it’s not reasonable to chosing an extremly large window.

4. Frequency filter

It’s often the case that only a narrow frequency band is essential in analysis. It should alse be taken into simulation that the high-pass and low-pass. Figure x.x would

[pure Lz and Lz with f noise]

[ringing signal and mixed ringing]

\subsection Reconstruct Orignal Signals from Result

\subsection Superiority of Simulation

to be written

%In traditional research, we should aim at figuring out the relationship

Chapter : Multi-station Signal Locating and Examination

\subscetion Intro

Thanks to the construction of global network, we have the confidence in discovering or denying exotic physics. LIGO earned a Nobel Prize with only two stations. So we can expect three or even more prizes if it’s a linear relation.

Still, we start simple. I made the following bizarre assumptions in order to address the problem in the most simple way. Of course we can modify these assumptions when we are clear about the basis of multi-station signal locating and examination.

Assumption1 When we take Earth as stationary reference system, the perturbation of exotic can be descibed as a plane wave.

Assumption2 the speed of perturbation is constant.

Assumption1 and Assumption2 is related to the time interval of signals between stations. We aren’t sure if the perturbation can be viewed as constant-speed plane wave from a human’s or Earth’s perspective. Things become more complicated when the perturbation is not a perfect plane wave. Anyway, it’s obvious that we can modify the model later on the basis of plane model.

Assumption3 The amplitude of the signal in a sensor is proportional to the projection of the plane wave on sensitive axis.

Time interval and amplitude are two factors in locating and examining signals. Assumption for signal amplitude is needed.

From my perspective, these reasonable assumptions are not likely to be true since we are looking for ‘exotic’. We just carry on research under these assumptions for now.

\subsection Locating signals in time domain