

KAGRAデータ解析スクール

バースト解析

端山和大

National Astronomical Observatory of Japan



- Octaveはmatlabと互換性が高い数値演算用のインタプリタ言語。
- プログラムを走らせるのに明示的にコンパイルする必要が無い。
- コードを書く労力が減り、新しいアイデアを気軽に試せる。
- そのかわりoctaveだけで大きなコードを書くことには不向き。
- 関数はC++で書くことも可能で、その際C/C++のライブラリを用いることができる。そのためC/C++ライブラリを使うインターフェースとして用いるケースも多い。
- 重力波業界でも、LVCで開発が行われているLALという解析ライブラリ（主にC言語で書かれている）を用いるインターフェースとして用いられる。（ただし、Pythonの方が利用者は多い。）
- マルチメッセンジャー観測で、他の望遠鏡データの解析を行うためのプログラムがoctaveで行われている。



% データを保存するファイルを開く

```
fid = fopen('SNparam_bLCGT.dat','w');
```

octaveマニュアルp219

ファイルを閉じるときは

```
fclose(fid)
```

とする。

% ファイルに収められているデータをlcgtという名前をつけてロードする。

```
lcgt=load('data1/LCGT/bLCGT_v1.dat');
```

octaveマニュアルp211

% signal_dataディレクトリ下にある重力波信号データの名前を変数LISTに入れる。

```
LIST=dir('data1/signal_data/*.dat');
```

octaveマニュアルp648

% LISTにあるデータファイル(~.dat)の個数をfnumに入れる。

```
[fnum tmpval]=size(LIST);
```



% LISTにあるデータすべてについて、forループでhrssなどの量を計算する。

```
for i = 1:fnum
```

[octaveマニュアルp150](#)

% LISTにあるデータファイルのデータをsigという名前をつけてロードする。

```
sig = load(fullfile('data1/signal_data',LIST(i).name));
```

[octaveマニュアルp636](#)

...

% 重力波データを補間して、サンプリングレートfsのデータにする。

```
Vt = 1/fs*[0:sig(end,1)*1e-3*fs]; % 1 second data[s]
```

```
Vsig = interp1(sig(:,1)*1e-3,sig(:,2),Vt,'linear');
```

[octaveマニュアルp547](#)

% 重力波のroot-sum-squareを計算する。

```
hrss = sqrt(sum(Vsig.^2).*dt);
```



% 重力波データに0を補間して長さ1秒分のデータにする。

```
Vt = 1/fs*[0:fs-1];
```

```
Vsig = [zeros(1,ceil(fs/2)) zeros(1,ceil((fs-length(Vsig))/2)) Vsig zeros(1,(fs-length(Vsig))-ceil((fs-length(Vsig))/2)) zeros(1,ceil(fs/2))];
```

[octaveマニュアルp360](#)

%データの長さを求める。

```
len = length(Vsig);
```

```
len2 = ceil((len-1)/2);
```

%周波数ベクトルと、解像度を計算する。

```
Vfre = fs*1/len*[1:len2];
```

```
df = Vfre(3)-Vfre(2);
```

% LCGT(KAGRA)のスペクトルを求める。

```
Vlcgt = interp1(lcgt(3:end,1),lcgt(3:end,2),Vfre(3:end),'linear');
```

```
Vlcgt = [Vlcgt(1) Vlcgt(1) Vlcgt];
```

% 重力波をフーリエ変換する。

```
fftVsig = fft(Vsig);
```

```
Vhf = fftVsig(1:ceil(length(fftVsig)/2)+1);
```

```
Vhf = Vhf(2:end);
```

[octaveマニュアルp569](#)

% 重力波の感度曲線に依存する特徴周波数fcを計算する。

```
fc = sum(Vhf*dt.*conj(Vhf*dt)./Vlcgt.^2.*Vfre)*df./(sum(Vhf*dt.*conj(Vhf*dt)./Vlcgt.^2)*df);
```



```
% 求めたhrssとfcをファイルに保存する。
fprintf(fid,'%7.2f %e ¥n',fc,hrss);
octaveマニュアルp222
end

% ファイルを閉じる。
fclose(fid);

% パラメータファイルをparamという名前で保存する。
param = load ('SNparam_bLCGT.dat');

% 計算したデータをグラフに出す。
loglog(Vfre,Vlclgt,'b');octaveマニュアルp241
hold on;octaveマニュアルp283
plot(param(:,1),param(:,2),'r*');octaveマニュアルp238
hold off;
xlabel('frequency[Hz]');octaveマニュアルp278
ylabel('amplitude spectral density[Hz{-1/2}]');
```



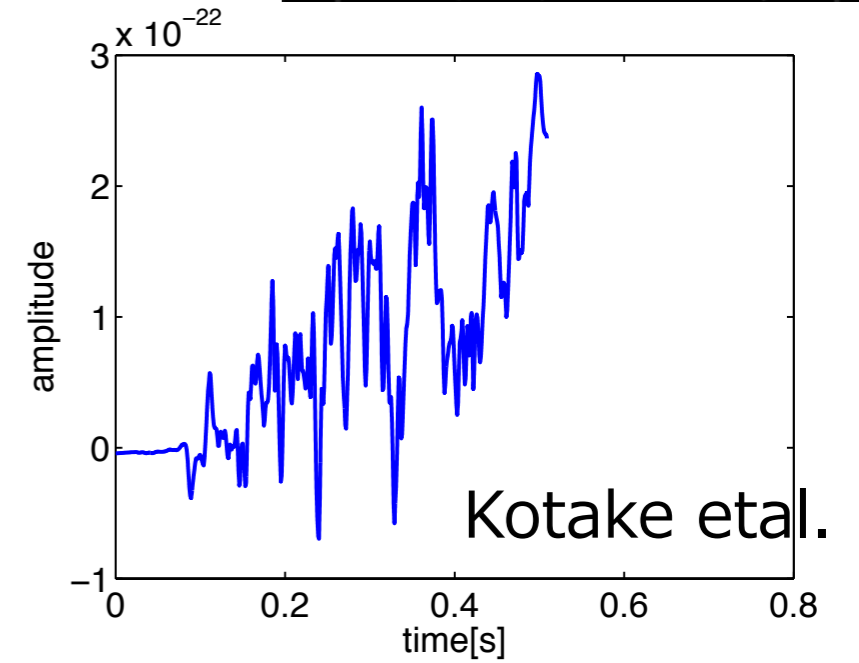
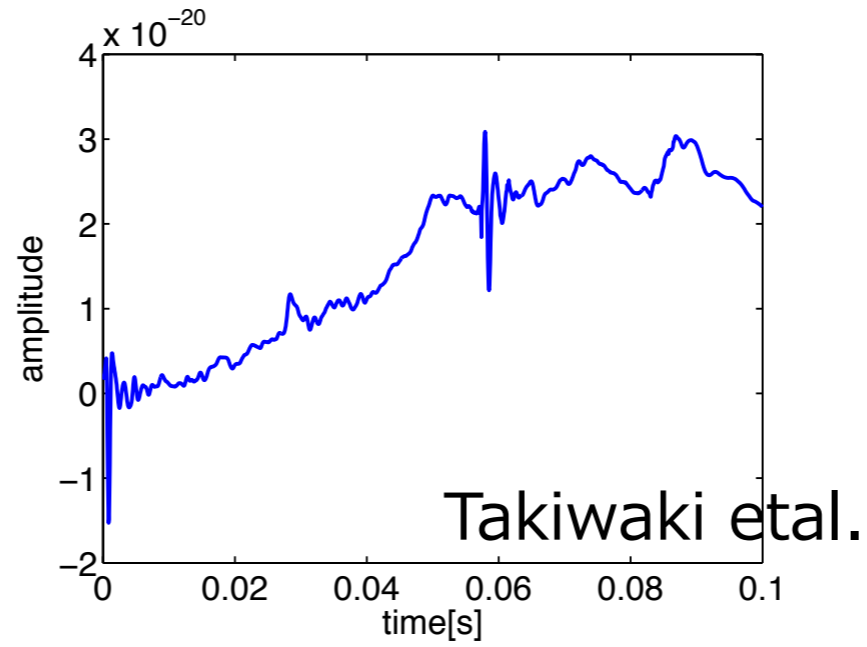
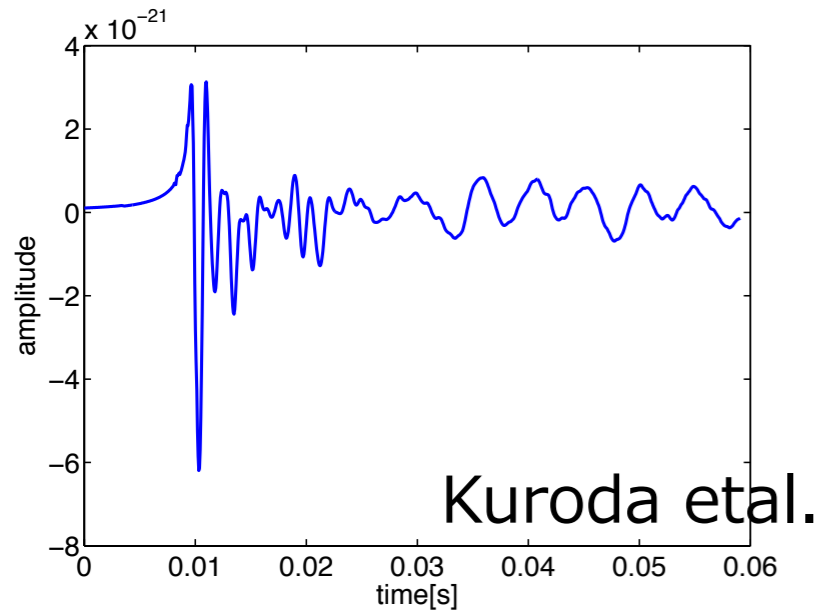


- パルサーからの連続波
- コンパクト連星合体からの重力波
- 背景重力波
- 波形不定性が強いバースト性重力波
 - 超新星爆発
 - パルサーグリッチ
 - 中性子星の星震(accreting mass等による)
 - ガンマ線バースト

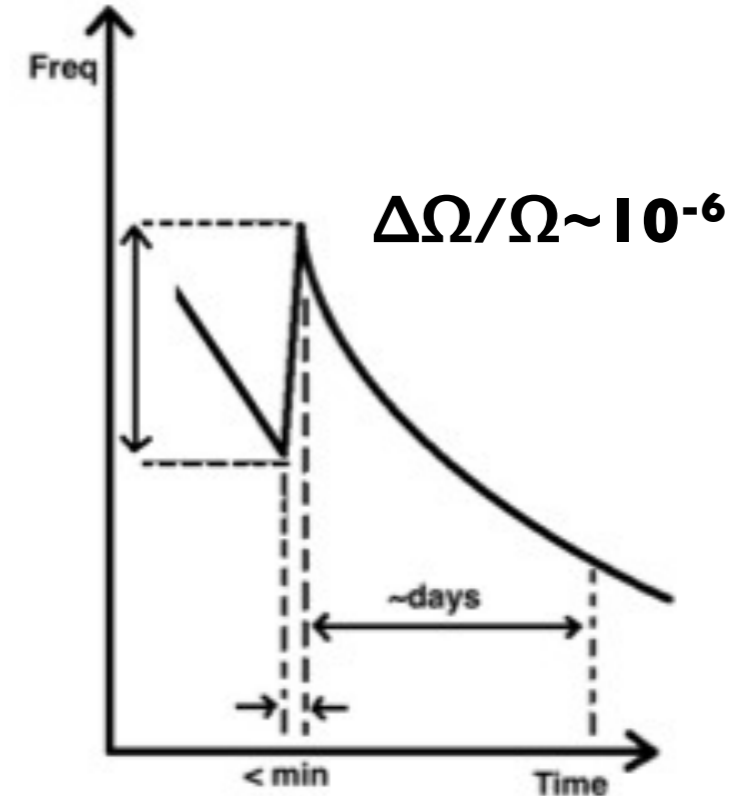
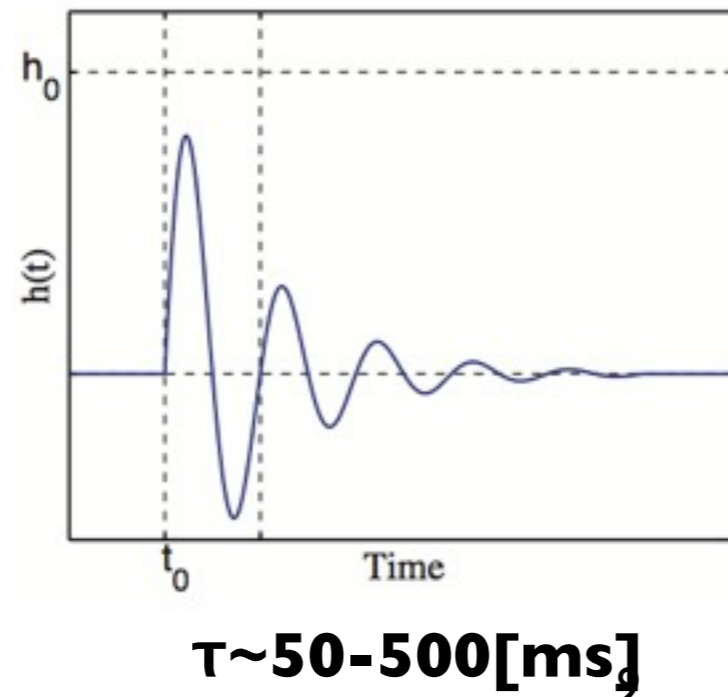
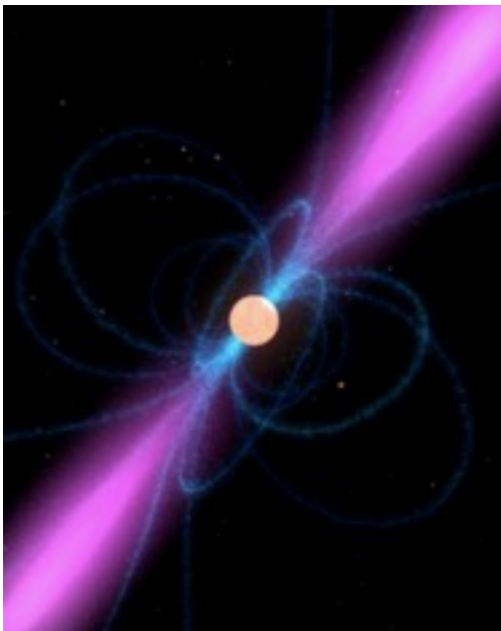
バースト性重力波源



超新星爆発からの重力波



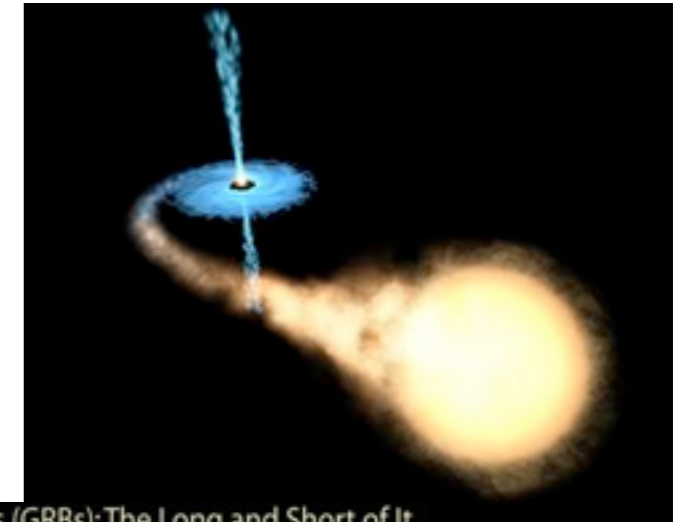
パルサーグリッチ





- 中性子星の星震(accreting mass等による)
- 降着ガスが降り積もり外殻が重力でつぶれ、エネルギーが重力波、X線などで放出。

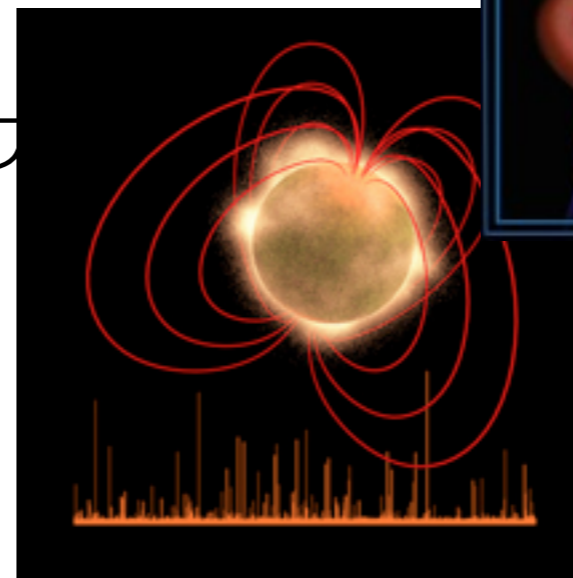
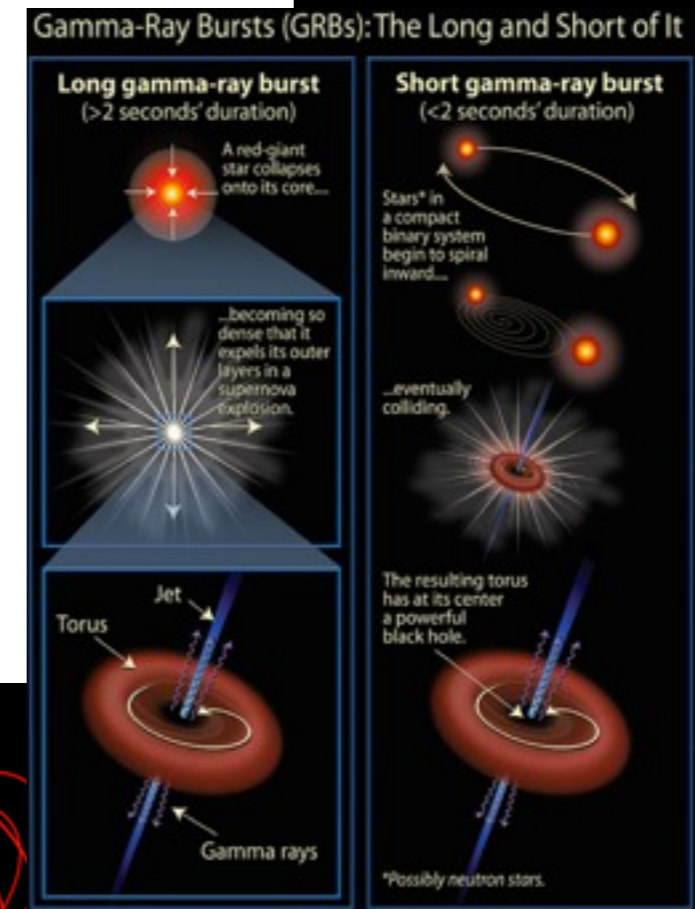
E. Cocchia et al.(2004)



- ガンマ線バースト
- 連星合体
- 高速回転大質量星によるhypernova

- SGR(Soft Gamma Repeater)
強磁場を持つ中性子星からのフレア

-



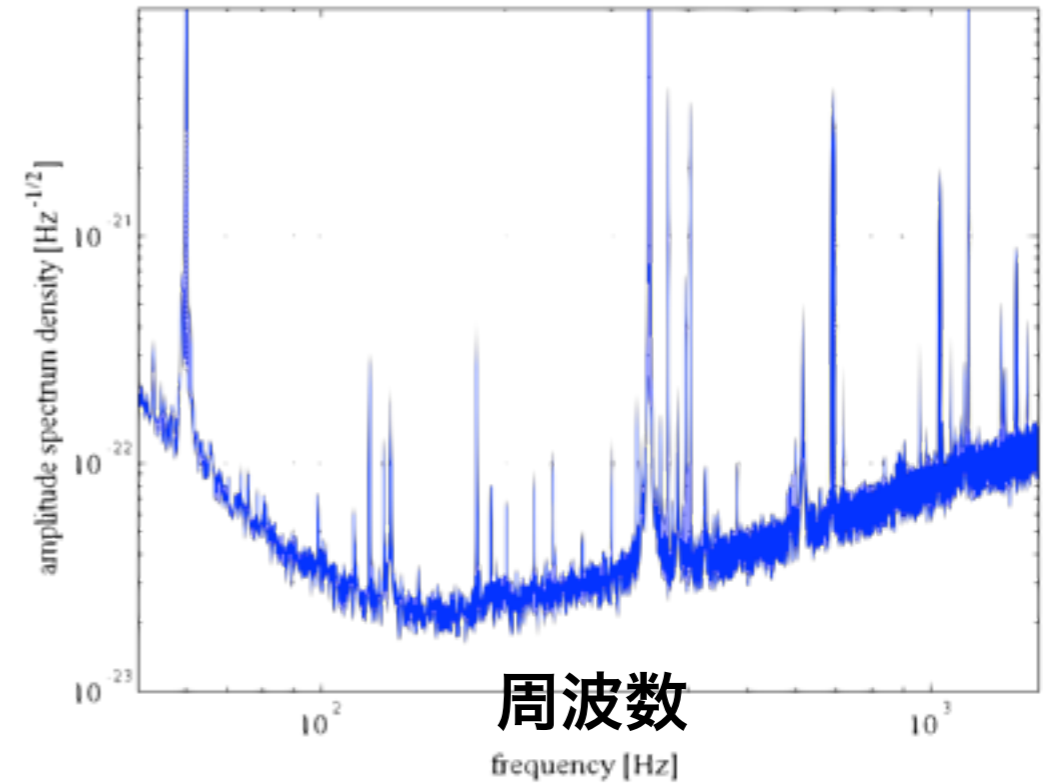


- 波形をパラメタ化できないバースト性重力波の探査。
- パルサーや連星合体からの重力波探査で用いられるようなフルテンプレートのサーチと比較すると計算コストが低いいため、クイックサーチでイベント候補をテンプレートサーチグループに渡す。
- 望遠鏡の出すバースト性雑音 (Glitch) を検出して、その雑音源となる望遠鏡の部分を特定する。(detector のcharacterization)
- データの質を評価して、サイエンスを行うのに不適當なデータ (glitchが多く含まれてしまっているデータ) にフラグを立てるなどする。

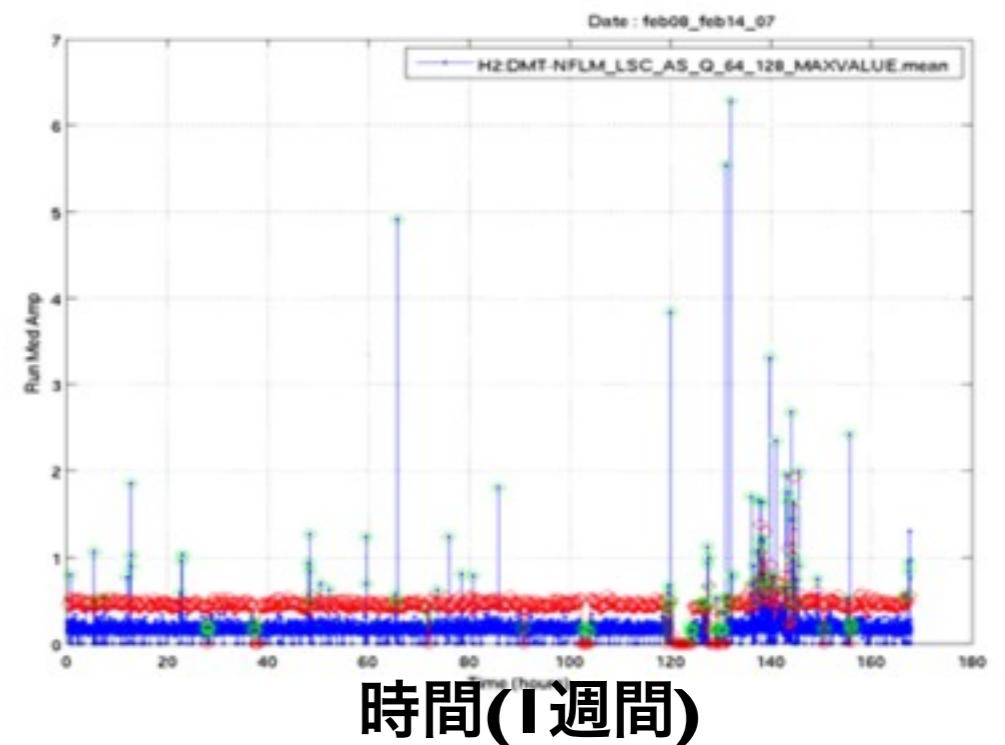


- 感度の周波数特性
- 狭帯域信号(ライン)
- データの非定常性
- 短いタイムスケールで感度、非定常雑音の統計が変化
- バースト性雑音(Glitch)
- 重力波との区別

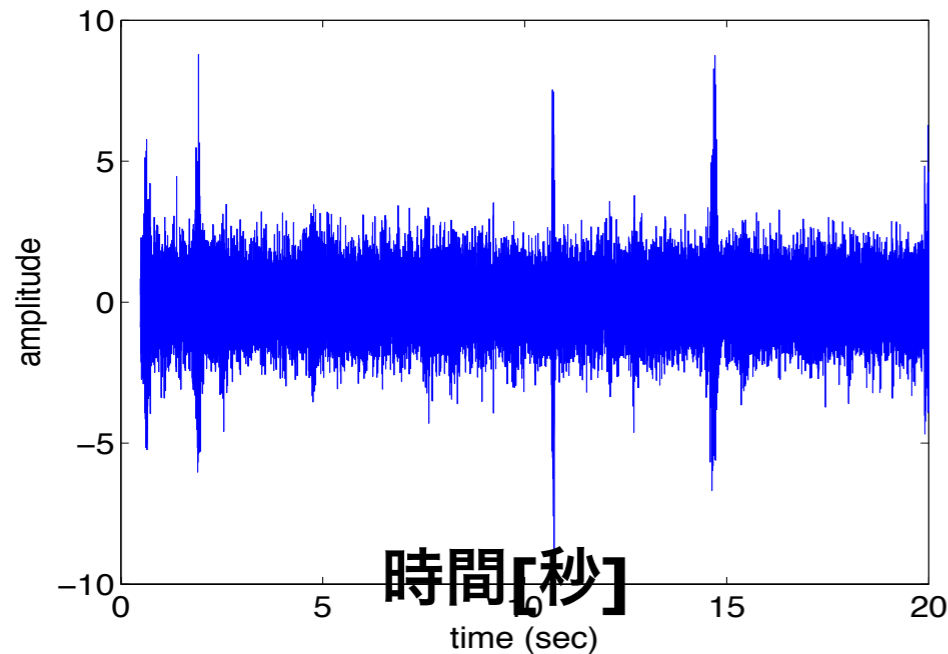
感度曲線



Noise Floor



Glitch

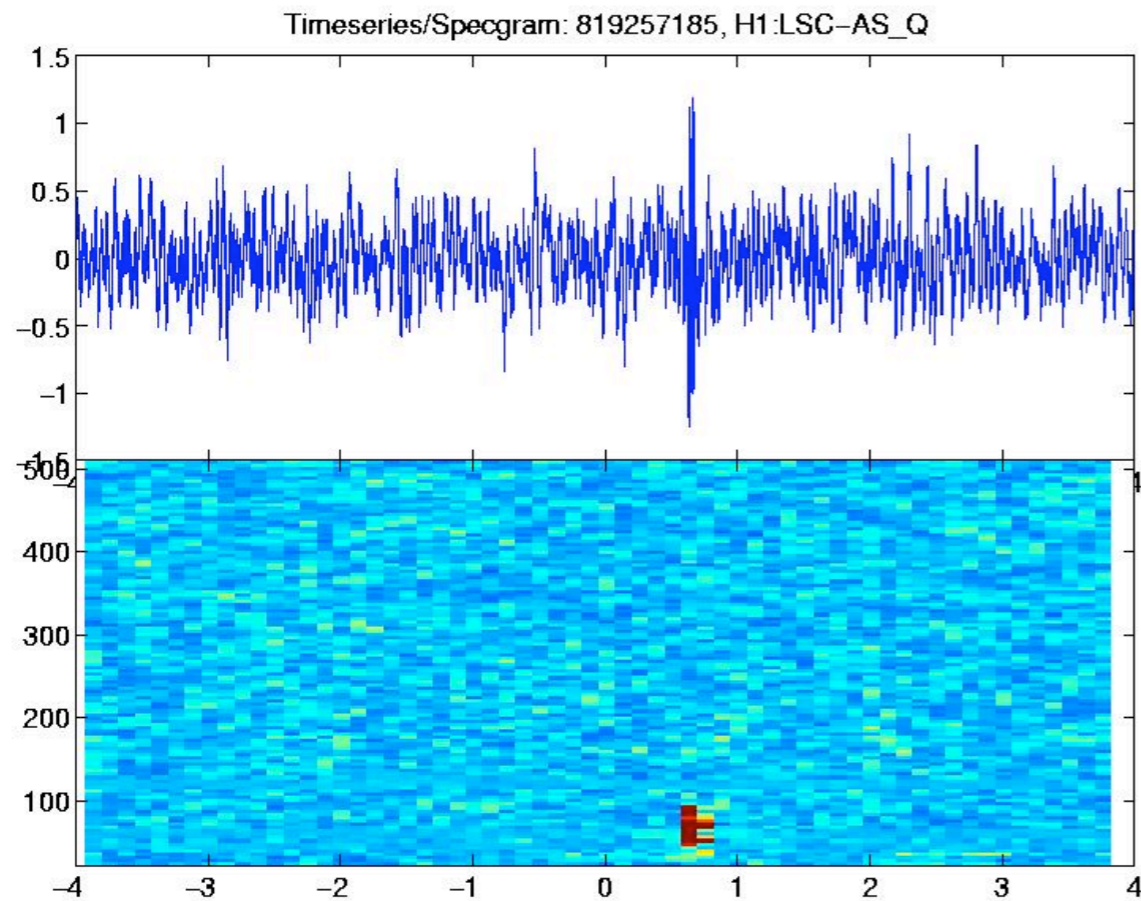


バースト性重力波の検出



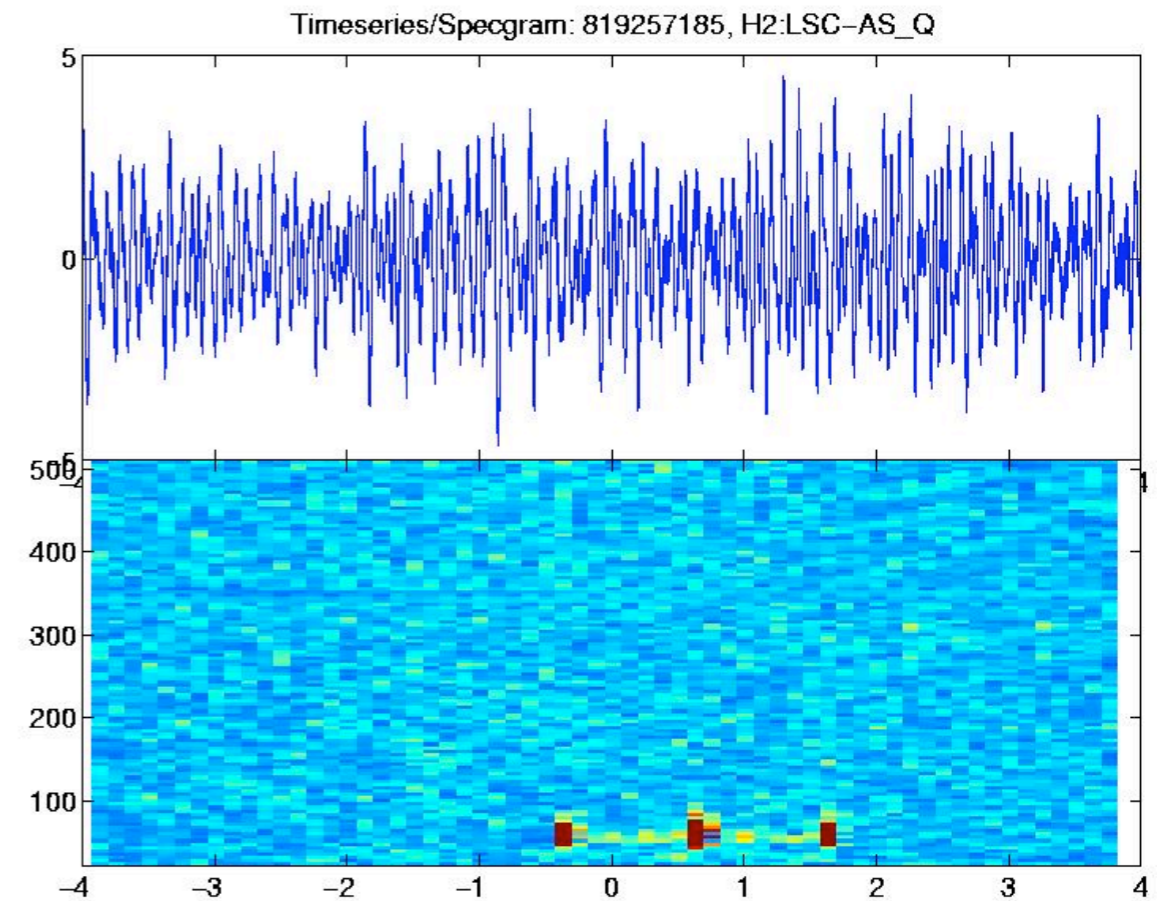
- 望遠鏡が多く、バースト性雑音 (Glitch) を含み、複数の望遠鏡を用いてさえコインシデンスイベントが多い。
- さまざまな情報を駆使して雑音と重力波を区別することが必要。

H1



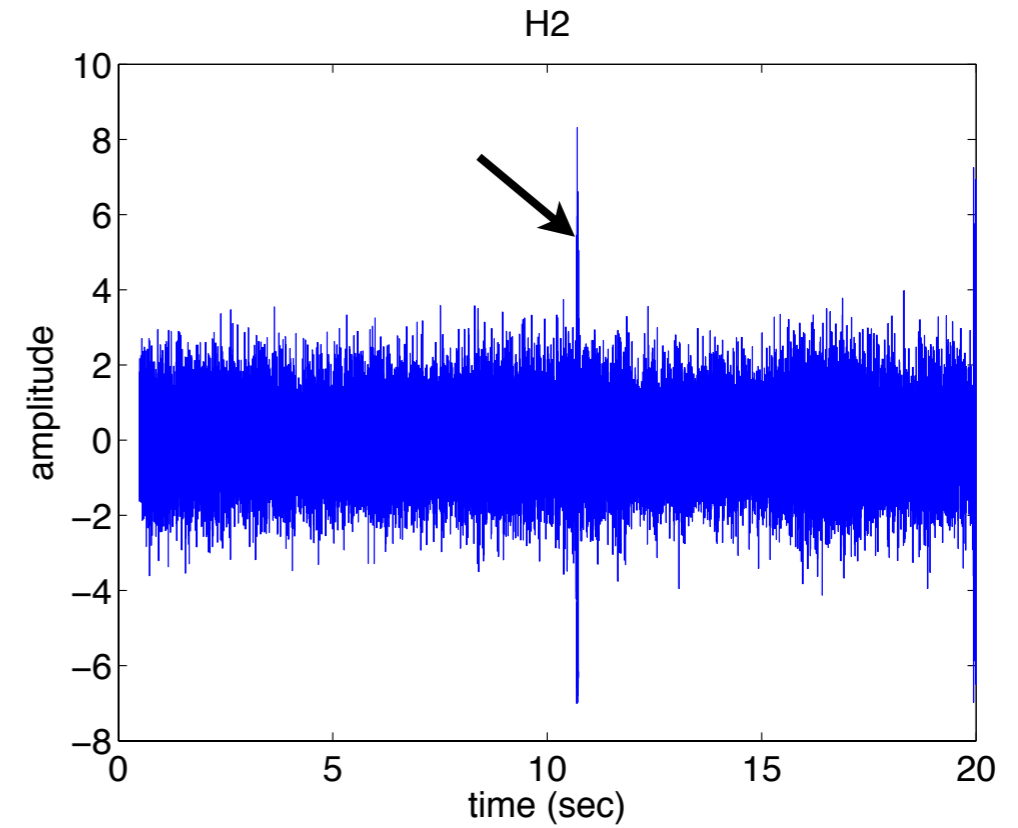
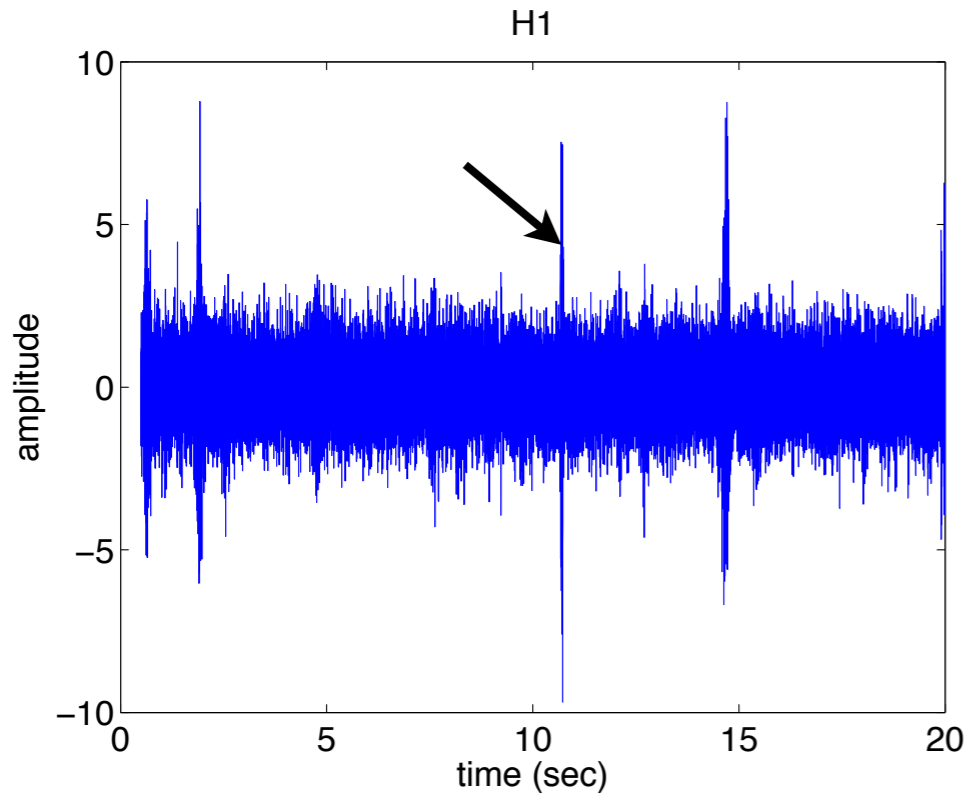
This plot made 16-May-2007 21:36:47

H2



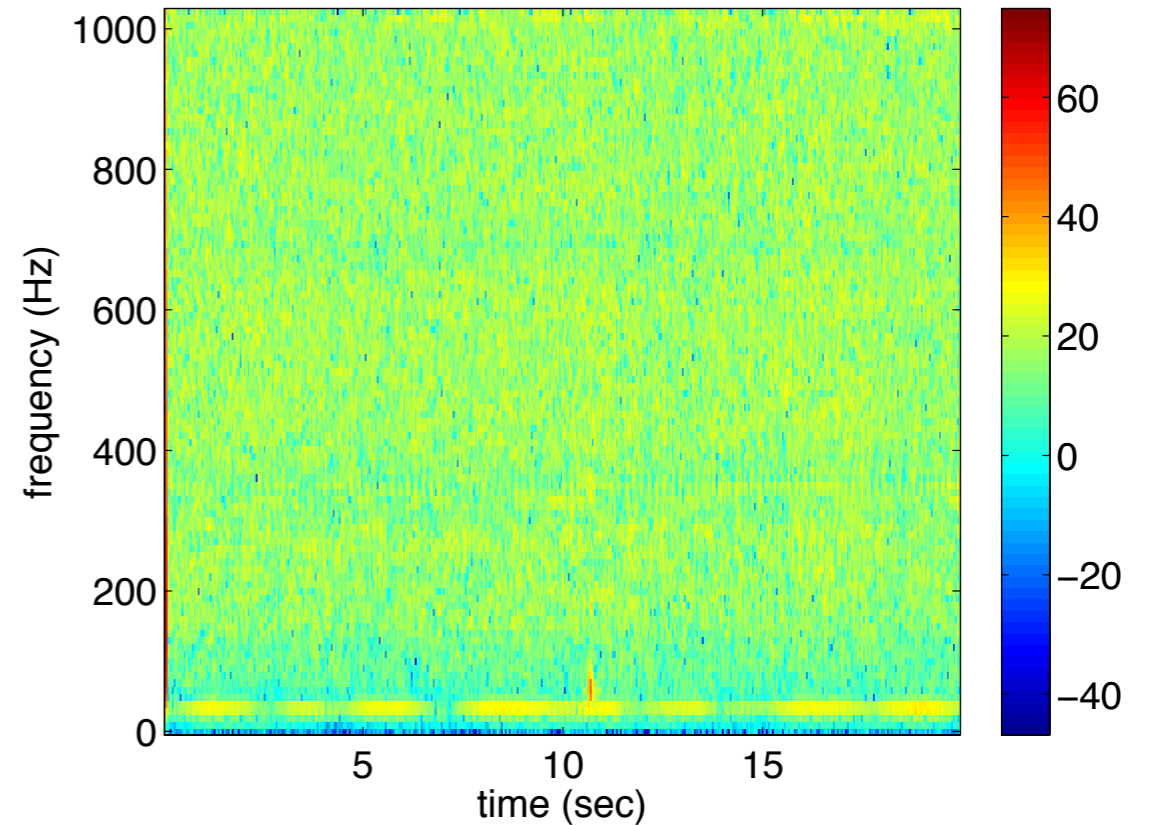
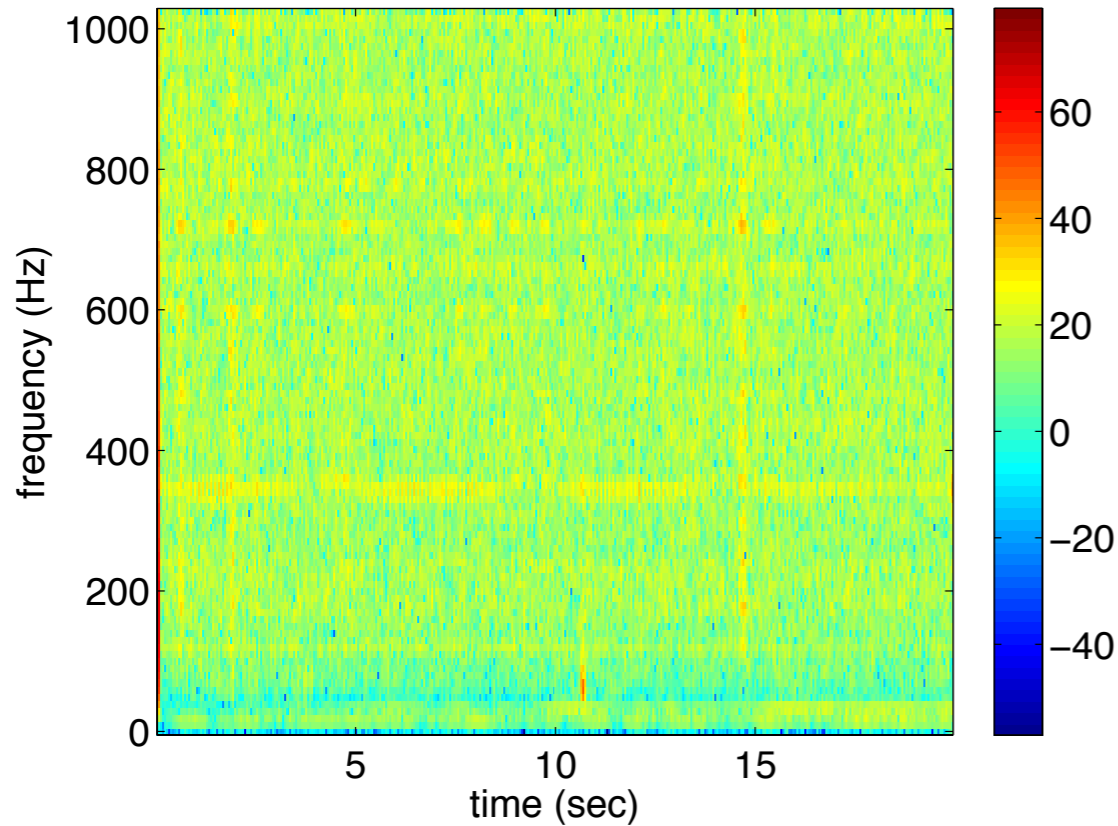
This plot made 16-May-2007 21:38:53

例：電源が生み出す磁場によって生じるGlitch



Power Glitch H1

Power Glitch H2





- 波形をパラメタ化できないバースト性重力波の探査。
- パルサーや連星合体からの重力波探査で用いられるようなフルテンプレートのサーチと比較すると計算コストが低いいため、クイックサーチでイベント候補をテンプレートサーチグループに渡すことができる。
- 望遠鏡(や、環境モニター)の出すバースト性雑音 (Glitch) を検出して、その雑音源となる望遠鏡の部分を特定する。(detector の characterization)
- データの質を評価して、サイエンスを行うのに不適當なデータ (glitchが多く含まれてしまっているデータ) にフラグを立てるなどする。



- データコンディショニング
望遠鏡データの周波数特性をフラットにする。
- データから重力波イベント候補を検出する
- 検出したイベントから偽のイベントを除去する。
 - コインシデンス解析
 - 波形の整合性のチェック
 - アンテナパターンによる信号振幅のチェック



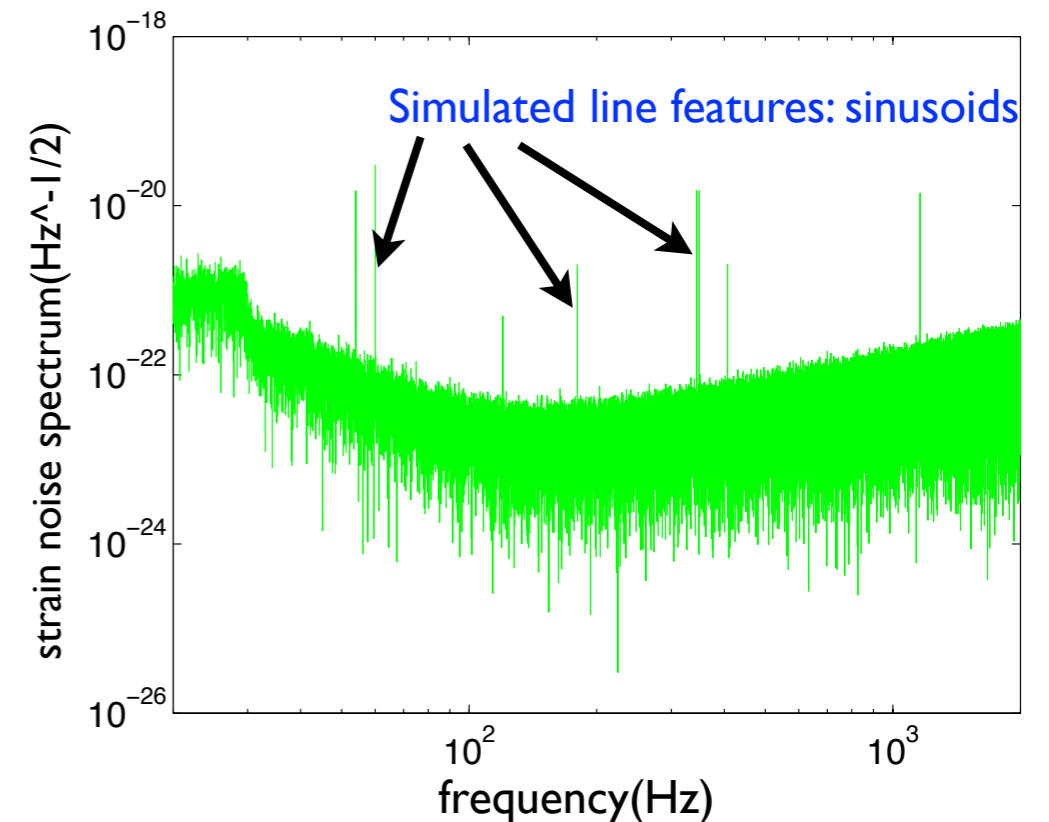
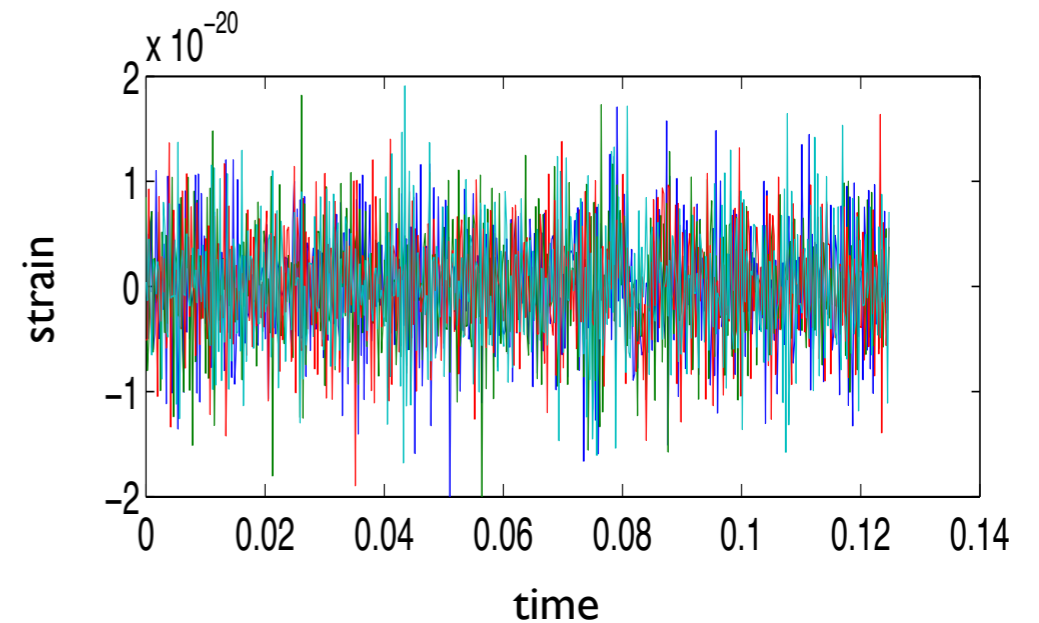
● ホワイトニング

- どの周波数帯も同様に探査したい。
- データの周波数特性をフラットにしてイベント探査。
- あとで周波数特性を戻してやる。

● ラインの除去

- 狭帯域ではあるが、非常に強いピーク。
- ラインの強度変化等は検出効率を悪化させる。
- 重みをかけて影響を無くす等。

Band-pass filtered(64-2000Hz) signals





フーリエ変換について

観測データは、時系列データそのものを解析するとともに、離散フーリエ変換などを行ってその周波数特性を調べることが多い。

\mathbf{x} の離散フーリエ変換は以下のように定義される。

$$\tilde{\mathbf{x}}(f_k) = \Delta t \sum_{j=0}^{n-1} \mathbf{x}(t_j) e^{-2\pi i f_k t_j}$$

$$\mathbf{x}(t_j) = \Delta f \sum_{k=0}^{n-1} \tilde{\mathbf{x}}(f_k) e^{2\pi i f_k t_j}$$

$$t_j = j\Delta t, f_k = k\Delta f, \Delta f = 1/T, T = N\Delta t$$



データの周波数特性を調べる際に頻繁に用いられるのがパワースペクトル密度という量で、詳しくはスペクトル解析の講義で出てくるが、データ長を**T**とすると、パワースペクトル密度**P(|f|)**は、データ**n(t)**のフーリエ変換 $\tilde{n}(f)$ を用いて、

$$\langle \tilde{n}(f_k) \tilde{n}^*(f_l) \rangle = \frac{T}{2} P(|f_k|) \delta_{kl}$$

と定義する。 $\langle \cdot \rangle$ はアンサンブル平均を表す。重力波データは非定常的ではあるが、数時間程度のデータであれば定常的と見なせるのでエルゴード的と見なし、アンサンブル平均は時間平均に置き換えられる。

ちなみにここで定義したパワースペクトルは片側パワースペクトル密度(One-sided power spectral density)と呼ばれるものである。

(前回の解析スクールのノートも参照)

<http://vega.ess.sci.osaka-u.ac.jp/~tagoshi/KAGRAschool/>



パワースペクトル密度

$$P(f_k) = \frac{2}{T} \langle \tilde{n}(f_k) \tilde{n}^*(f_k) \rangle$$

をoctaveで計算する。octaveのフーリエ変換はfftコマンドを用いて行う。(マニュアルのp569参照)

fftコマンドが行うフーリエ変換は

$$FFT[n](f_k) = \sum_{j=0}^{n-1} n(t_j) e^{-2\pi i f_k t_j}$$

$$IFFT[\tilde{n}](t_j) = \frac{1}{N} \sum_{k=0}^{n-1} \tilde{n}(f_k) e^{2\pi i f_k t_j}$$

$$t_j = j\Delta t, f_k = k\Delta f, \Delta f = 1/T, T = N\Delta t$$

なので、 $n(t)$ のフーリエ変換は

$$\tilde{n}(f) = \Delta t \cdot FFT[n(t)]$$

と計算する。(GRASPマニュアルp53も参照)



以上から、パワースペクトル密度は

$$\begin{aligned}
 P(f) &= \frac{2}{T} \langle \tilde{n}(f_k) \tilde{n}^*(f_k) \rangle \\
 &= \frac{2}{N\Delta t} (\Delta t \cdot |FFT[n(t)]|)^2 \\
 &= \frac{2\Delta t}{N} |FFT[n(t)]|^2
 \end{aligned}$$

と計算できる。アンサンブル平均はM個の同じデータ長のデータを用いて

$$\langle \tilde{n}(k) \tilde{n}^*(k) \rangle = \frac{1}{M} \sum_{m=1}^{m=M} \tilde{n}_m(k) \tilde{n}_m^*(k)$$

として得る。



- データをフーリエ変換する。
- 入力データが偶数の時、 $n(f_1)$ がDC成分で、 $n(f_2)$ から $n(f_{(N-1)/2})$ を用いて $n \cdot n^*$ を計算する。
- 得られた数列に $2\Delta t/N$ をかけてやるとパワースペクトル密度が得られる。
- それぞれの要素に対応する周波数と合わせて、パワースペクトル密度をプロットしてみる。



● Linear Prediction Part:

n番目のサンプルはその前M個のサンプルの線形和で表されると仮定する。

$$\hat{x}[n] = \sum_{m=1}^M c[m]x[n-m]$$

● Prediction Error Part:

実データと比較する。

$$e[n] = x[n] - \hat{x}[n]$$

Prediction Errorの自乗平均が最小になるようにc[m]を決める。

$$\sigma_e^2 = \frac{1}{N} \sum_{n=1}^N e[n]^2$$

そのために満たすべき条件は

$$\frac{\partial \sigma_e^2}{\partial c[k]} = 0, \quad 1 \leq k \leq M$$

となる。この方程式は、Yule-Walker方程式

$$\sum_{m=1}^M r[m-k]c[m] = r[k], \quad 1 \leq k \leq M$$

$$r[k] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=-N/2}^{N/2} x[n]x[n-k]$$

に帰着される。ここで $r[k]$ は自己相関係数である。実際にはトレーニングデータで係数 $c[m]$ を決めるので、 N は有限で、

$$r[k] = \frac{1}{N} \sum_{n=k+1}^N x[n]x[n-k]$$

を用いる。



このYule-Walker方程式を解いて係数**c[m]**を求める手法がLevinson-Durbin再帰法である。

<http://www.mathworks.co.jp/jp/help/signal/ref/levinson.html>

に簡単な説明がある。

octaveでは、自己相関**r[m]**が与えられたら、

```
c = levinson(r,n)
```

で**c[m]**が求められる。

$$\hat{x}[n] = \sum_{m=1}^M c[m]x[n-m]$$

というフィルタを掛けるoctaveコマンドの一つはfiltfiltで、

```
X_filtered = filtfilt(c,1,x);
```

のように用いる。

octave-forgeマニュアル

<http://octave.sourceforge.net/signal/function/filtfilt.html>

matlabマニュアル

<http://www.mathworks.co.jp/jp/help/signal/ref/filtfilt.html>



- トレーニングデータで自己相関関数を求める。
- パワースペクトルを計算する。
- パワースペクトルを逆フーリエ変換して自己相関関数を求める。

パワースペクトル $P(\omega)$ と自己相関関数 $R(\tau)$ の関係

$$P(\omega) = \int_{-\infty}^{\infty} R(\tau) e^{-i\omega\tau} d\tau$$

- 自己相関関数と、係数の数 M からLevinson-Durbin再帰法を用いてホワイトニングフィルタの係数 $c[m]$ を求める



Linear prediction では フィルタ長分のデータのprediction errorを最小自乗法の意味で最小にする。そのためフィルタ長の矩形窓をかけていることと同じとみなせるから、周波数解像度 Δf は

$$\Delta f \sim \frac{f_s}{M}$$

と見積もることができる。故に、解析するデータの時間幅をTとした時、

$$M \geq f_s T$$

を満たすように取ってやれば良い。

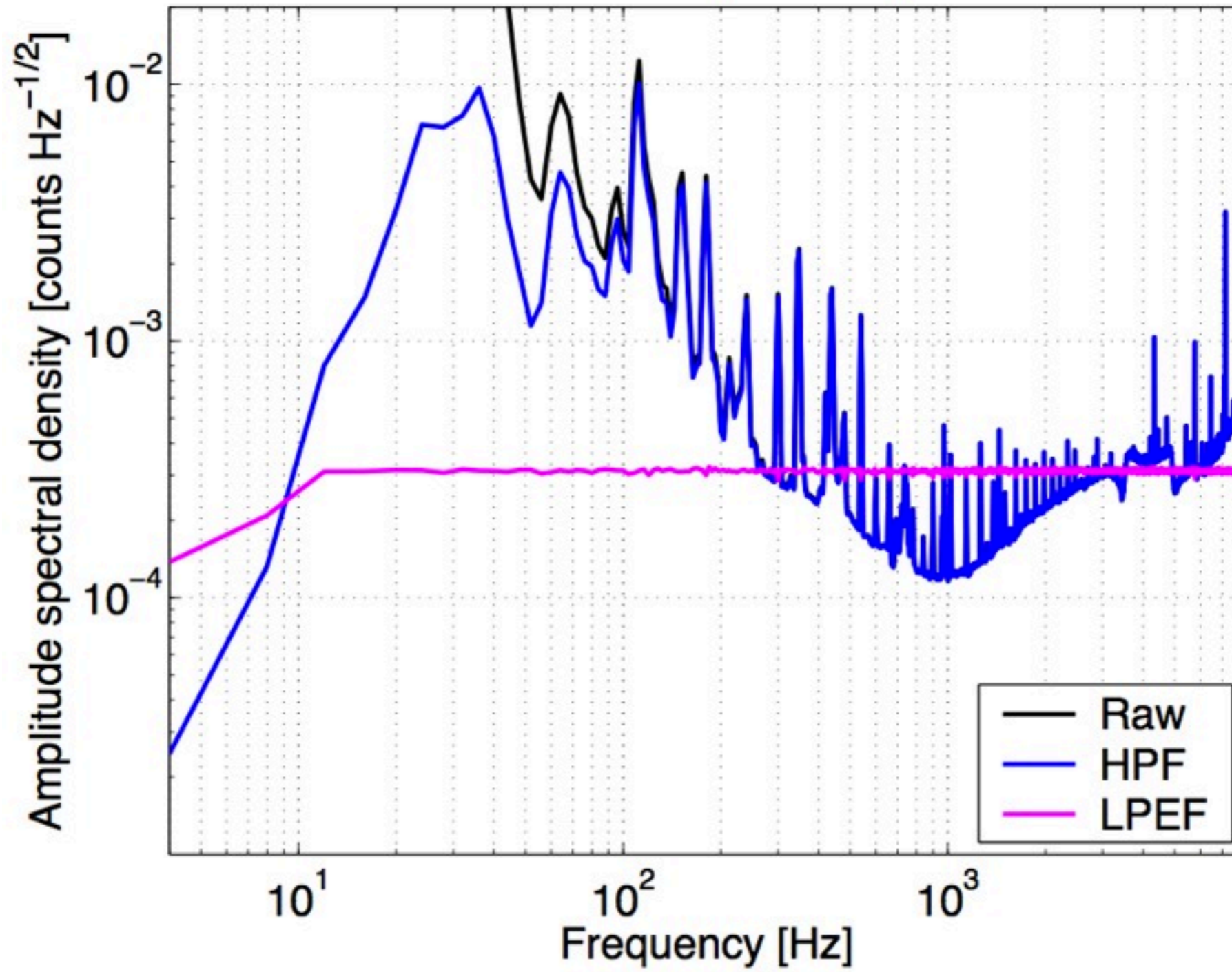


- データをフーリエ変換した後、その周波数特性 $S(f)$ の逆数 $1/S(f)$ を持った係数をコンボリューションし、逆フーリエ変換で時系列にする。
- この処理はLinear prediction error filterを周波数空間で行っていることと同等となる。

ホワイトニング例 (スペクトル表示)



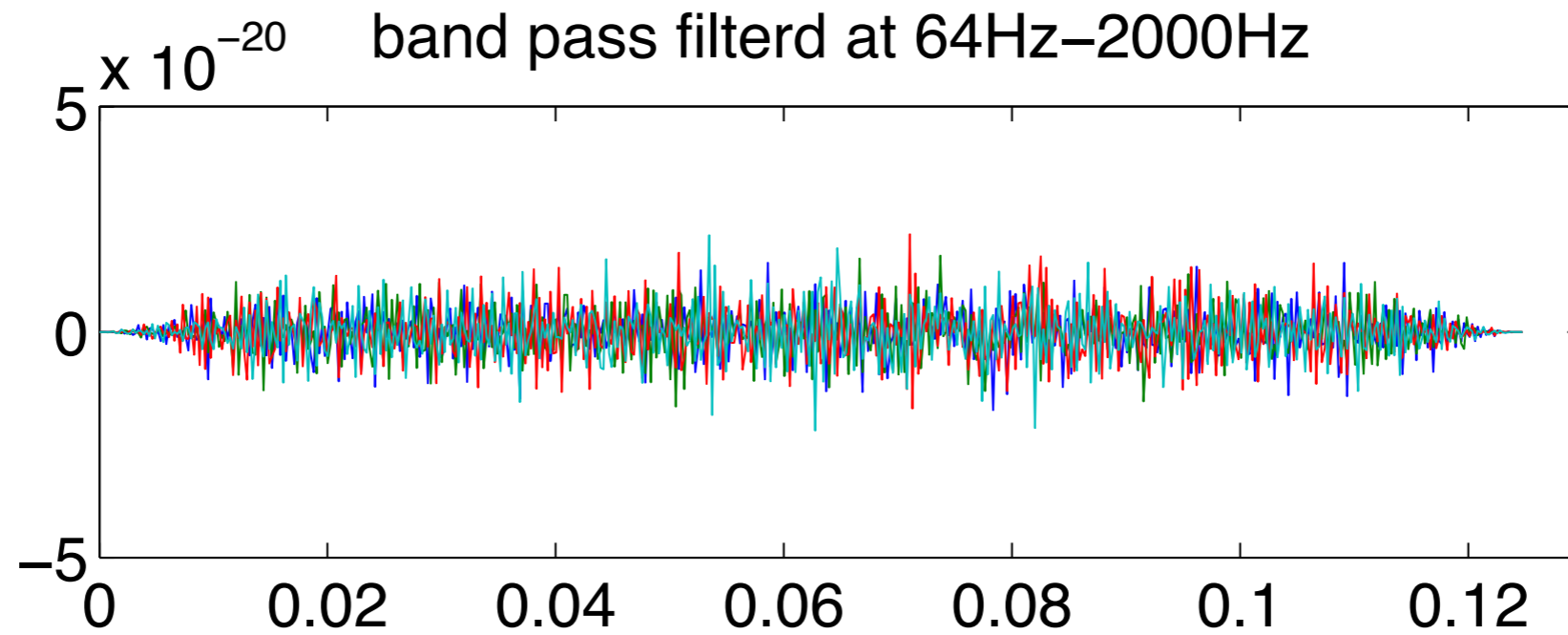
Uncalibrated amplitude spectra



ホワイトニング (時系列表示)

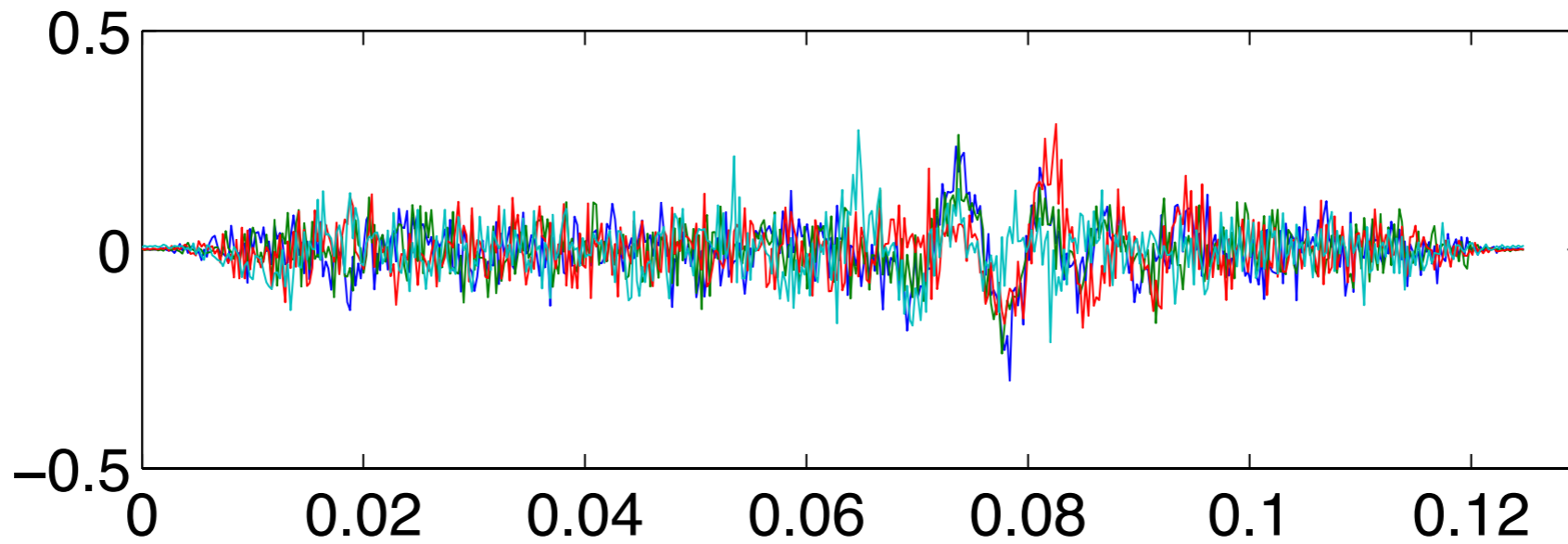


前



after conditioning

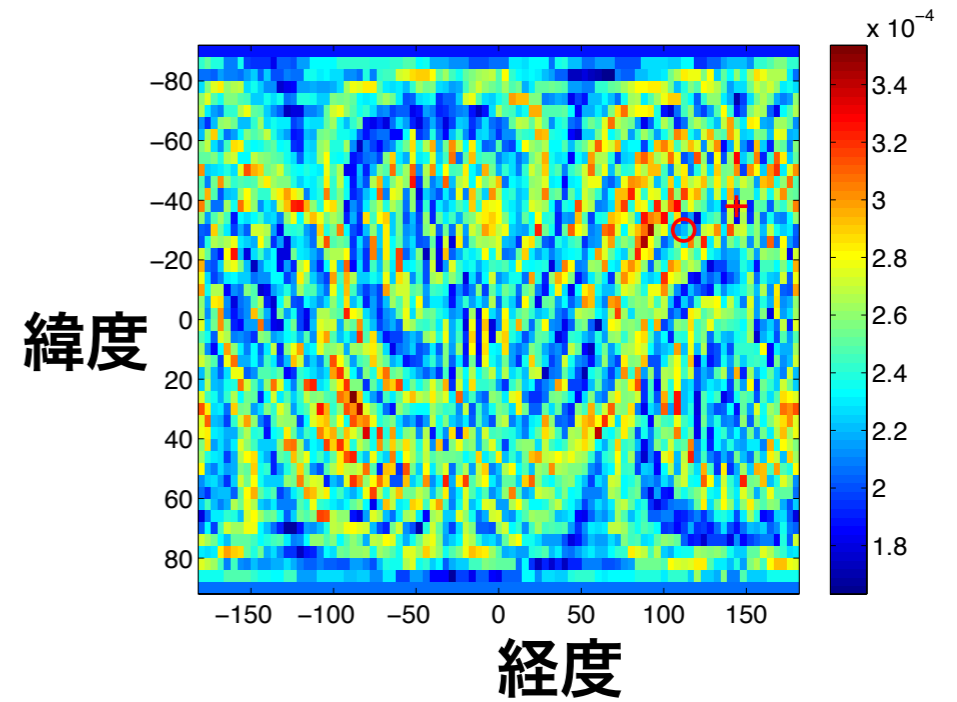
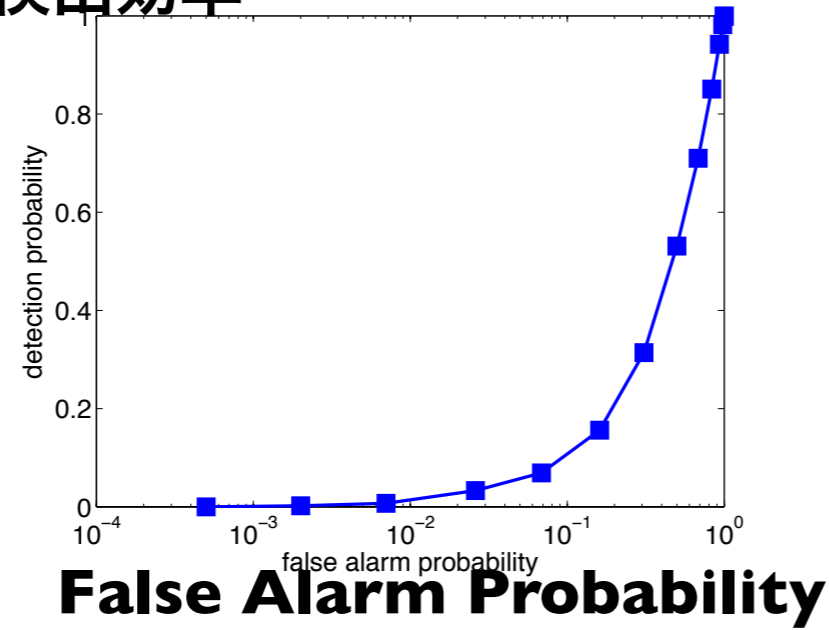
後



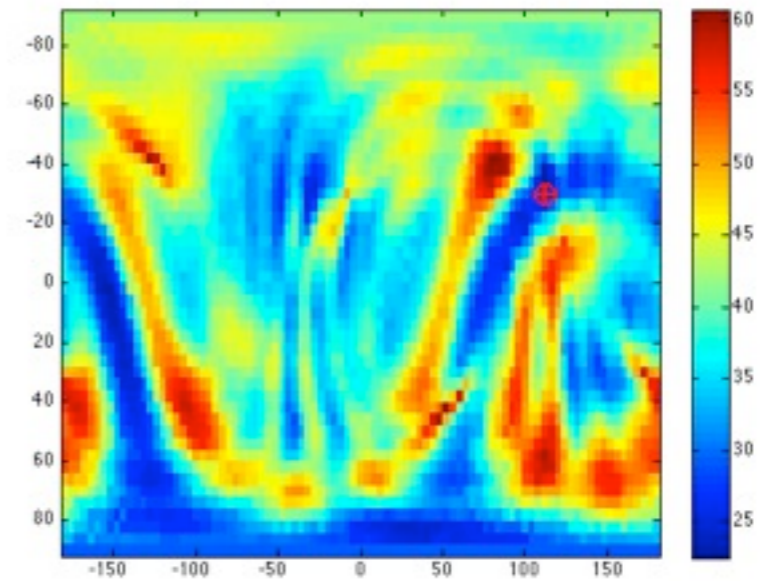
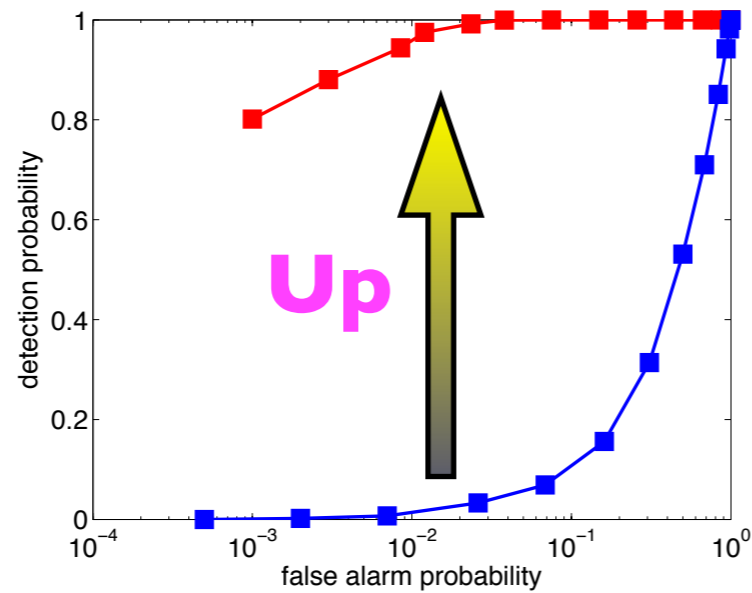


**Without
DC**

検出効率



**With
DC**





- データからバースト性信号を抽出する。
- 波形不明のバーストサーチの場合、エネルギーが時間的に局在した信号を抽出する。
 - 単独望遠鏡の場合
 - Excess power method
時間周波数空間上で強いエネルギーを持つイベントを探す。
 - 複数望遠鏡の場合
 - コインシデンス解析
複数の望遠鏡データで検出されたイベントのコインシデンスを取る。
 - コヒーレントネットワーク解析
最初の段階で、望遠鏡データ全てを重力波の到達時間差を考慮して結合。



重力波望遠鏡の観測データ h は重力波信号 s と望遠鏡雑音 n で

$$h = s + n$$

$$s = F_+ s^+ + F_- s^-$$

と書け、我々の目的は観測データ h から重力波 s があるか無いかを判断することである。その判断を統計的仮説検定で行う。重力波でよく用いられる方法がネイマン・ピアソンの尤度比検定で、

- $p[h | s]$: 重力波信号が入っているときに、観測されたデータ h が得られる確率
 $p[h | 0]$: 重力波信号が入っていないときに観測されたデータ h が得られる確率

の比

$$\Lambda[h] \equiv \frac{p[h | s]}{p[h | 0]}$$

と、設定した閾値とを比較して検定を行う。

望遠鏡の雑音 n が白色でガウス分布していると仮定すると、

$$p[n] = C \exp\left[-\frac{n \cdot n}{2}\right]$$

と書ける。すると、 $p[h|s]$ と $p[h|0]$ は、 n は重力波が観測データに含まれる場合と含まれない場合で

$$n = h - s$$

$$n = h$$

のように表せるより、

$$\frac{p[h|s]}{p[h|0]} = \exp\left[(s \cdot h) - \frac{1}{2}(s \cdot s)\right]$$

と書ける。

$$\frac{p[h | s]}{p[h | 0]} = \exp \left[(s \cdot h) - \frac{1}{2} (s \cdot s) \right]$$

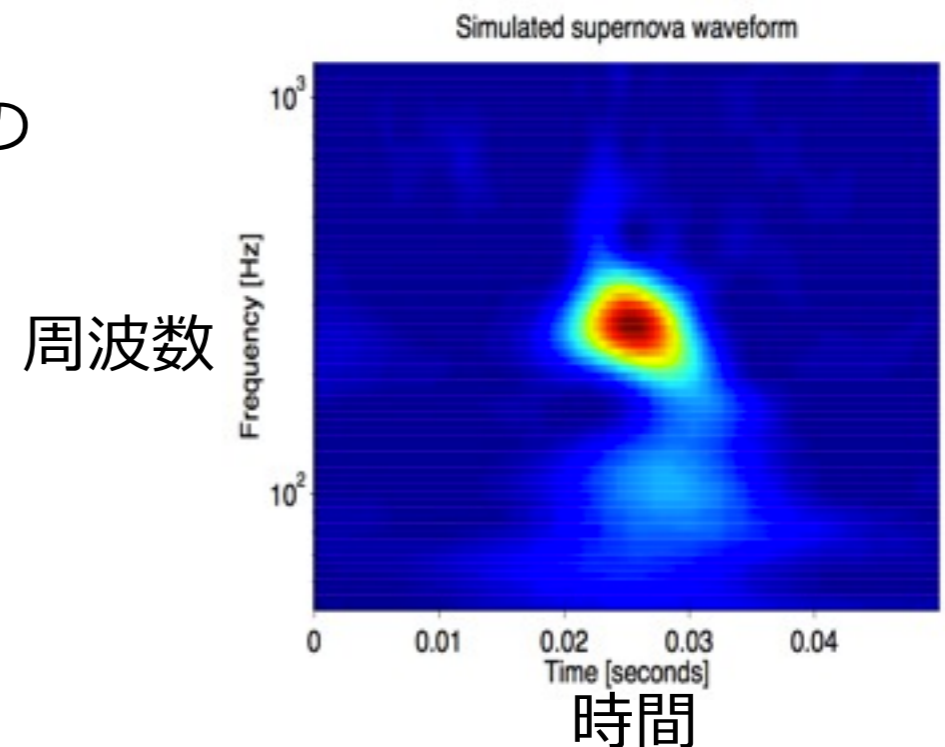
sの持つパラメタを走査して尤度比 (likelihood ratio) が最大となるパラメタを探し、閾値を越えるものを探していく。連星合体からの重力波のように波形がパラメタ化されているとき (s(p1,p2,...)) はパラメタ空間を高々 10 数個に制限できる。このような探査方法を **Matched filter method** という。

超新星爆発からの重力波のように波形をパラメタ化することが困難な重力波を探査する場合は、このままでは例えば時系列での探査であればサンプル数分のパラメータを走査することになり、困難となる。

そこでBH-BH合体からの重力波がある時間-周波数の限られた領域で表すことができることから、時間周波数空間でパワーが雑音に比べて高い領域を検出する方法が提案された

(Flanagan&Hughes(1998))。

これが **Excess power method** という手法で、今回用いる方法である。



重力波信号が時間周波数空間のある領域 $\mathbf{T} = \{t_s, \delta t, f_s, \delta f\}$ に集中すると仮定する。すると、その領域のフーリエ要素は

$$\tilde{H}_K = \sum_{j=0}^{N_t-1} e^{-2\pi i(t_j - t_s) f_k} h(t_j)$$

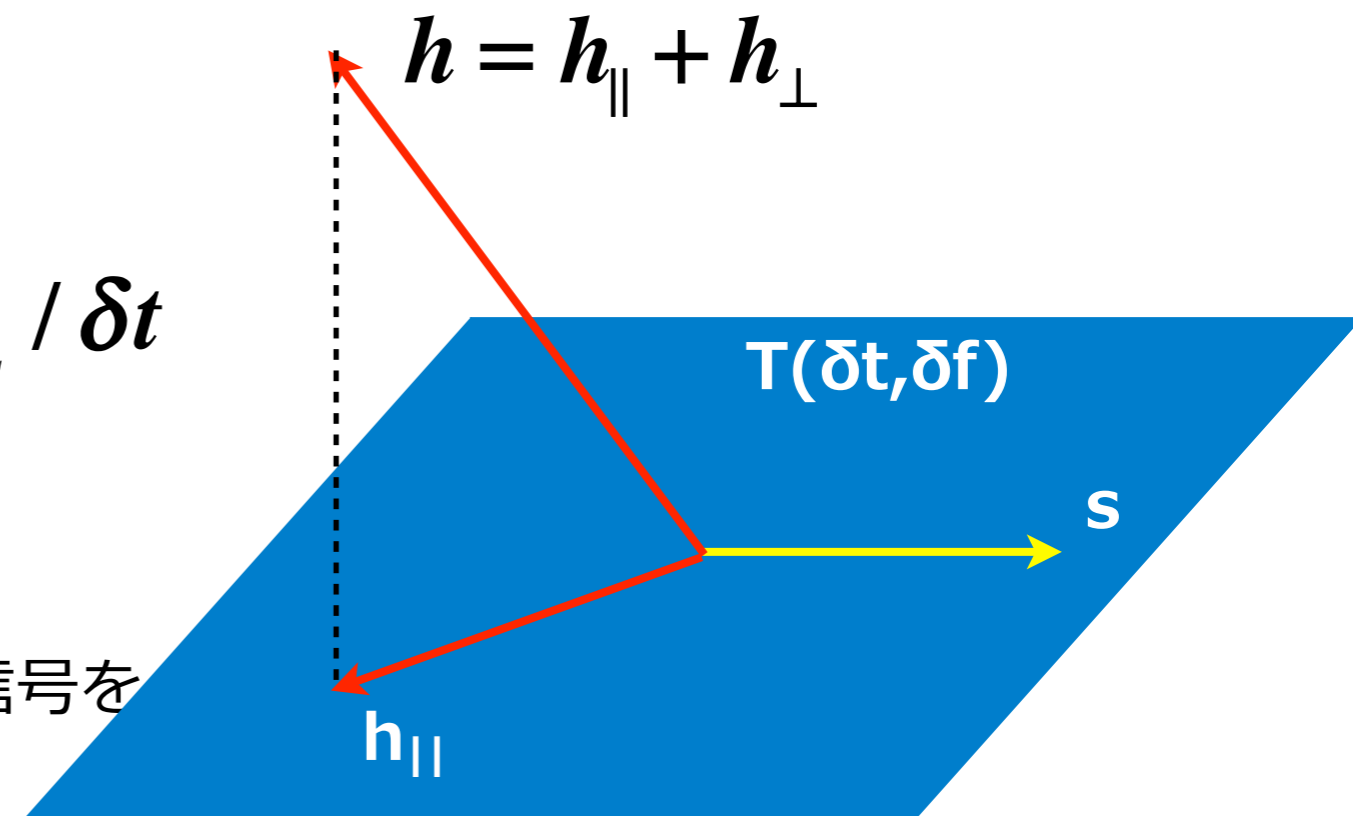
$$t_j = t_s + j\Delta t, \quad N_t = \delta t / \Delta t, \quad f_k = \frac{1}{\Delta t} \frac{k}{N_t}$$

となる。

$$E \equiv h_{\parallel} \cdot h_{\parallel} = 4 \sum_{k=k_1}^{k_2} \frac{|\tilde{H}_K|^2}{\sigma^2}$$

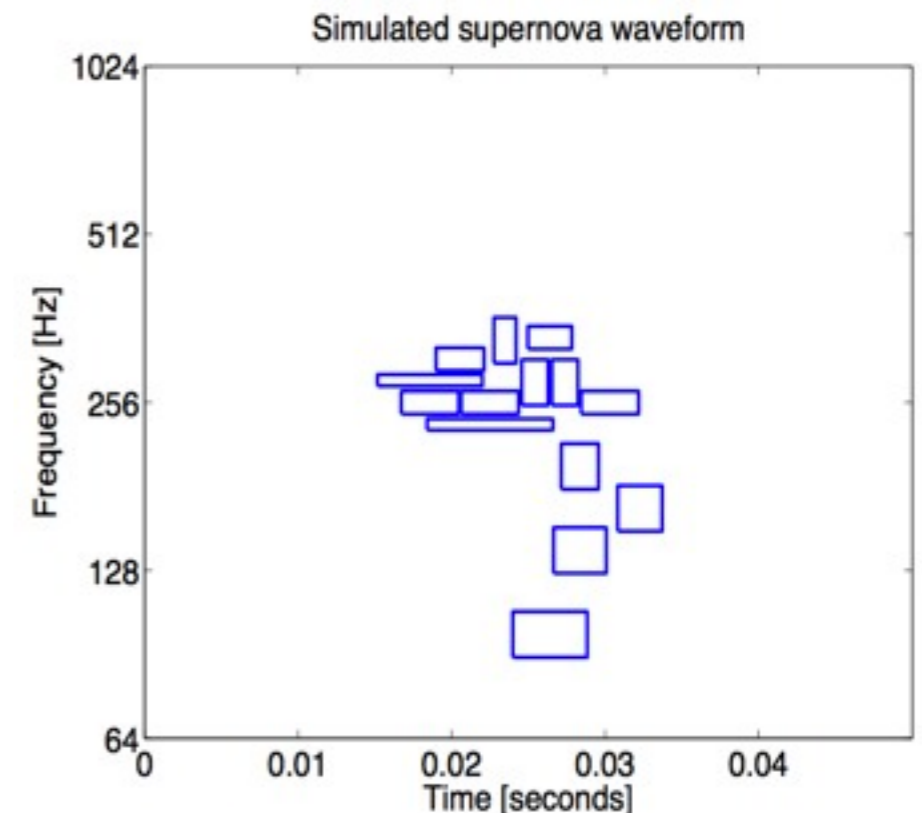
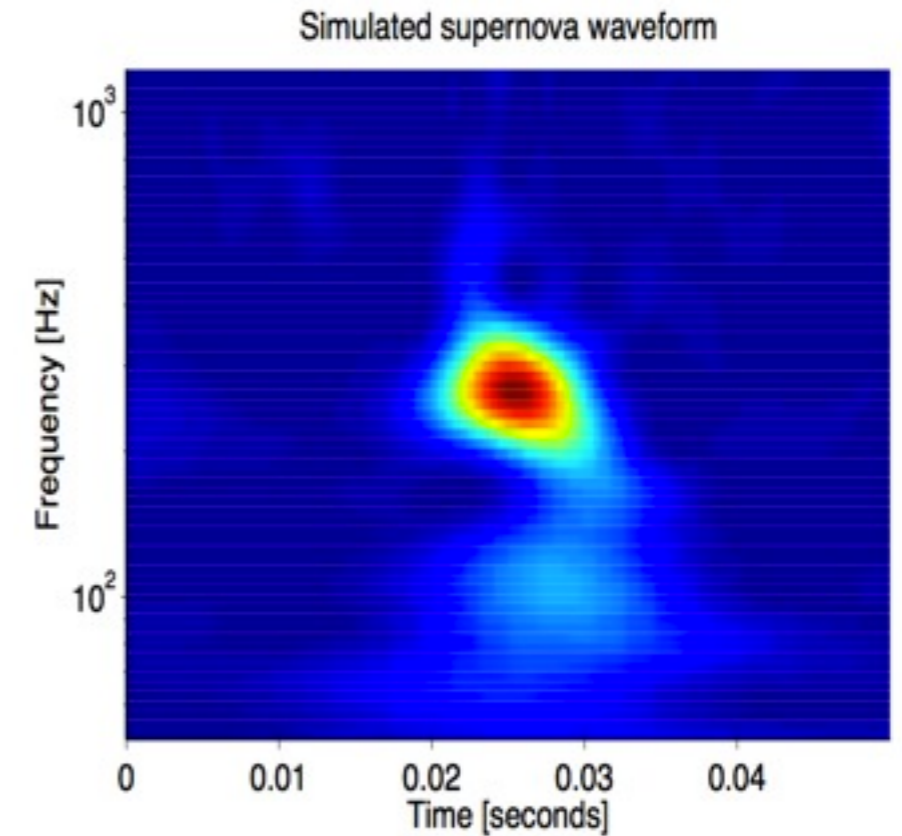
$$\delta f = f_2 - f_1, \quad f_1 = k_1 / \delta t, \quad f_2 = k_2 / \delta t$$

E は、雑音のみのとき、 $\mathbf{V}_T = 2\delta t \delta f$ の自由度をもつカイ自乗分布をするので、カイ自乗分布からのずれを測って、重力波信号を検出する。





- 時系列データをFourier変換により時間周波数空間に射影する。
- 時間周波数窓の形を変えながら、Excess powerを計算する。
- EPが閾値を越える時間周波数ピクセルを抽出する。
- 時間的に連続しているものを一つの信号としてクラスタリングする。
- イベントの再構成完了。





- 時間領域 δt を決めて、そのデータ長をフーリエ変換する。
- 調べる周波数領域 δf を決めて、その領域のexcess powerを計算する。
- excess powerは、時間周波数領域 $T(\delta t, \delta f)$ 中にあるタイル数の2倍の自由度を持つカイ自乗分布をしている。
- excess powerが、対応する $T(\delta t, \delta f)$ の雑音のみの時のexcess powerと比較して、設定した閾値を越える場合、検出候補としてリストする。

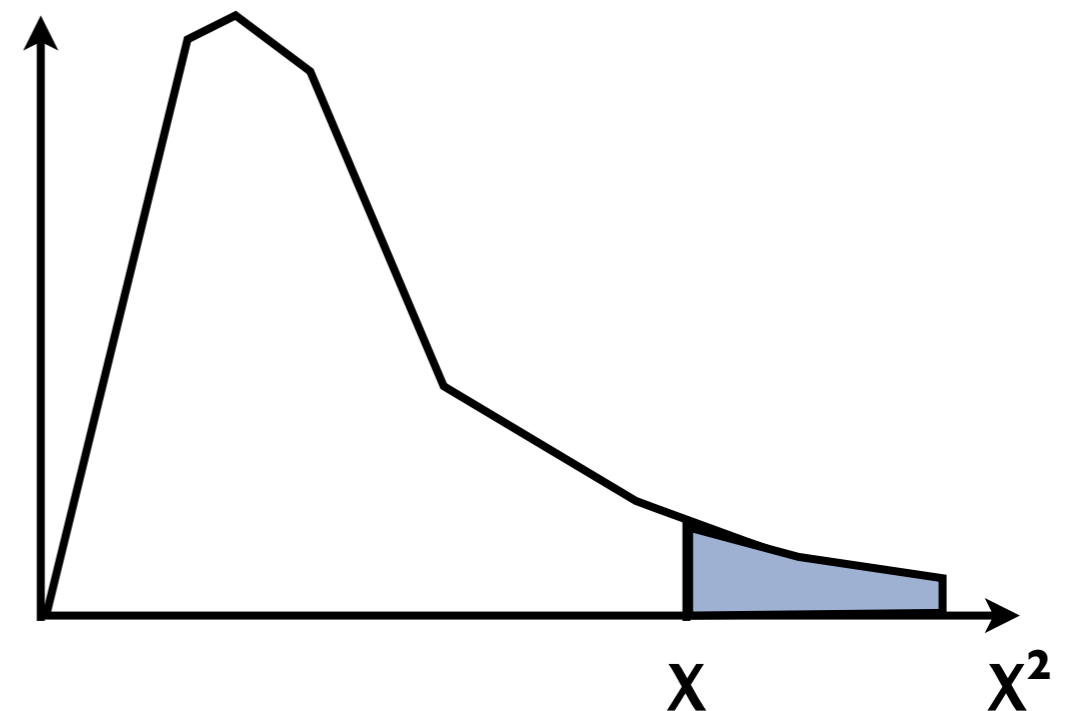
ガウス分布をしている雑音 $\mathbf{x}(\mathbf{t})$ は、そのフーリエ成分 $\mathbf{X}(\mathbf{f})$ もガウス分布になる。そのためpower $|\mathbf{X}(\mathbf{f})|^2$ はカイ二乗分布に従う。そこで、時間周波数ピクセル数 N を含む時間周波数領域 $\mathbf{T}(\delta\mathbf{t}, \delta\mathbf{f})$ のパワーは、自由度 N のカイ二乗分布に従っているかどうかを検定することで、 $\mathbf{T}(\delta\mathbf{t}, \delta\mathbf{f})$ が雑音由来のものではないかどうかを統計的に判断する。このような検定をカイ二乗検定という。

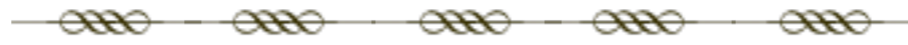
ある時間周波数領域 $\mathbf{T}(\delta\mathbf{t}, \delta\mathbf{f})$ 内の i 番目のピクセルのpower E_i が対応する雑音のpower E_n と同じカイ二乗分布をするかどうかは、

$$\chi^2 = \sum_{i=1}^N \frac{(E_i - E_n)^2}{E_n}$$

が自由度 N のカイ二乗分布で見いだされる確率 (**p-値**という) が、有意水準(0.05など)を越えるかどうかで検定を行う。

p-値が有意水準よりも小さな値である場合、 $\mathbf{T}(\delta\mathbf{t}, \delta\mathbf{f})$ が雑音であることが棄却される。





各望遠鏡のデータから、Excess powerで重力波候補信号を抽出したら、次はコインシデンス解析である。重力波信号は、その天球位置に依存した到達時間にずれを生じながらほぼ同時に各望遠鏡データに記録される。そこで各望遠鏡データに記録されたイベント候補が、到達時間の差を考慮した時間窓の中に入っているかどうかをテストする。

時間窓を w 、望遠鏡-i、-jで検出されたイベント候補の（ピークの）時間を t_i 、 t_j 、継続時間を Δt_i 、 Δt_j ととすると、 t_i 、 t_j が満たすべき条件は

$$|t_i - t_j| \leq w + \frac{1}{2}(\Delta t_i + \Delta t_j)$$

となる。 w は、通常望遠鏡間の最大距離を指定する。

[演習の資料] coincidence analysisの資料を参照



各望遠鏡で検出されたイベント候補で、たまたまコインシデンスしているものも実際多く存在する。その際、その候補を除去するための指標が、波形の相関解析である。ここでは、ピアソンの積率相関係数(Pearson product-moment correlation coefficient)を用いて評価する。

$$R = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}}$$

ただし、各望遠鏡のアンテナパターンが両偏極モードともに一致していないと完全なテストが行えないことに注意。

<http://adsabs.harvard.edu/abs/2004CQGra..21S1695C>



各望遠鏡に記録されていたコインシデンスイベントは、それが重力波信号だとすると時間差を調べることによって到来方向が推定できる。その情報を利用して、信号を雑音を区別することもできる。

重力波の到来方向から各望遠鏡のアンテナパターンが分かるため、信号はアンテナパターンに対応した振幅を持っている必要がある。

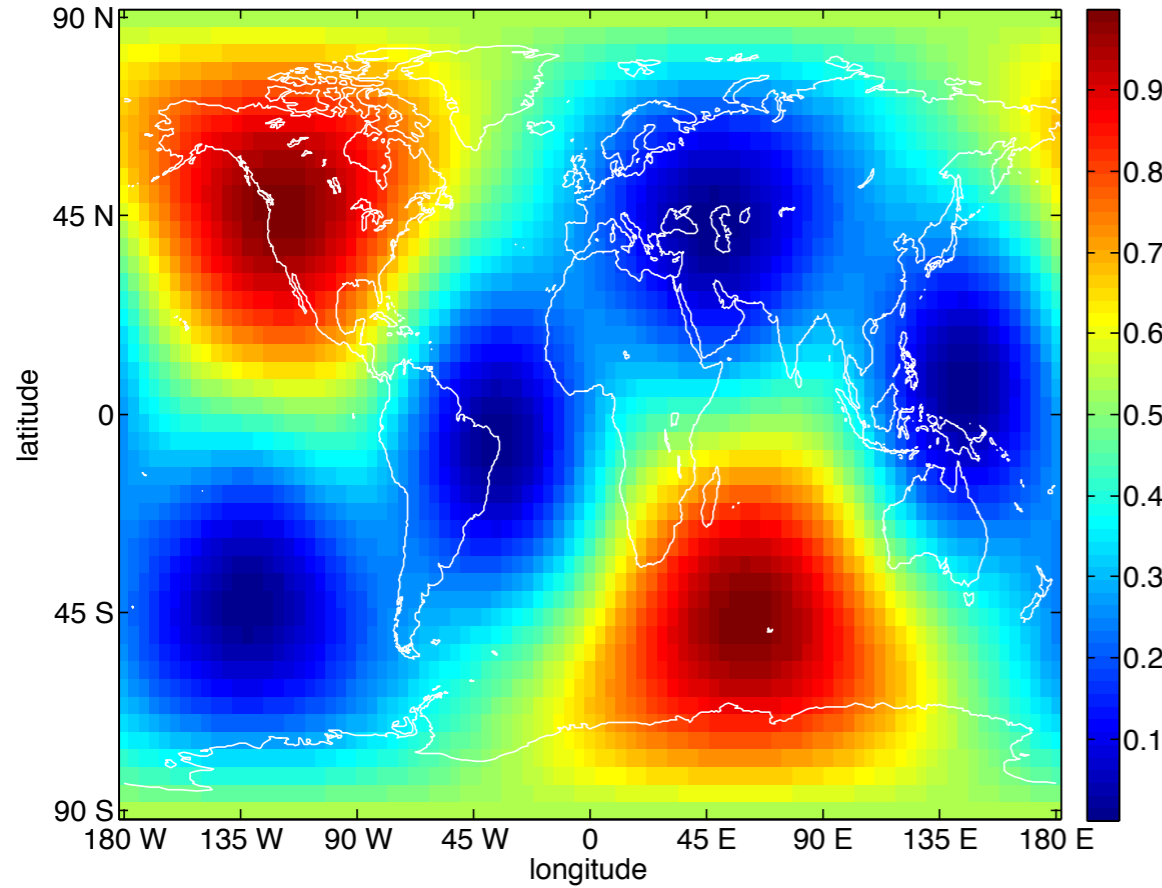
つまり、アンテナパターンが悪い望遠鏡に記録されているコインシデンスイベントの振幅が、アンテナパターンが良い望遠鏡に記録されているコインシデンスイベントの振幅よりも大きくなってはいけない。

対して望遠鏡起源の雑音によるアクシデンタルイベントの場合はそのような振幅に対する相関がない。

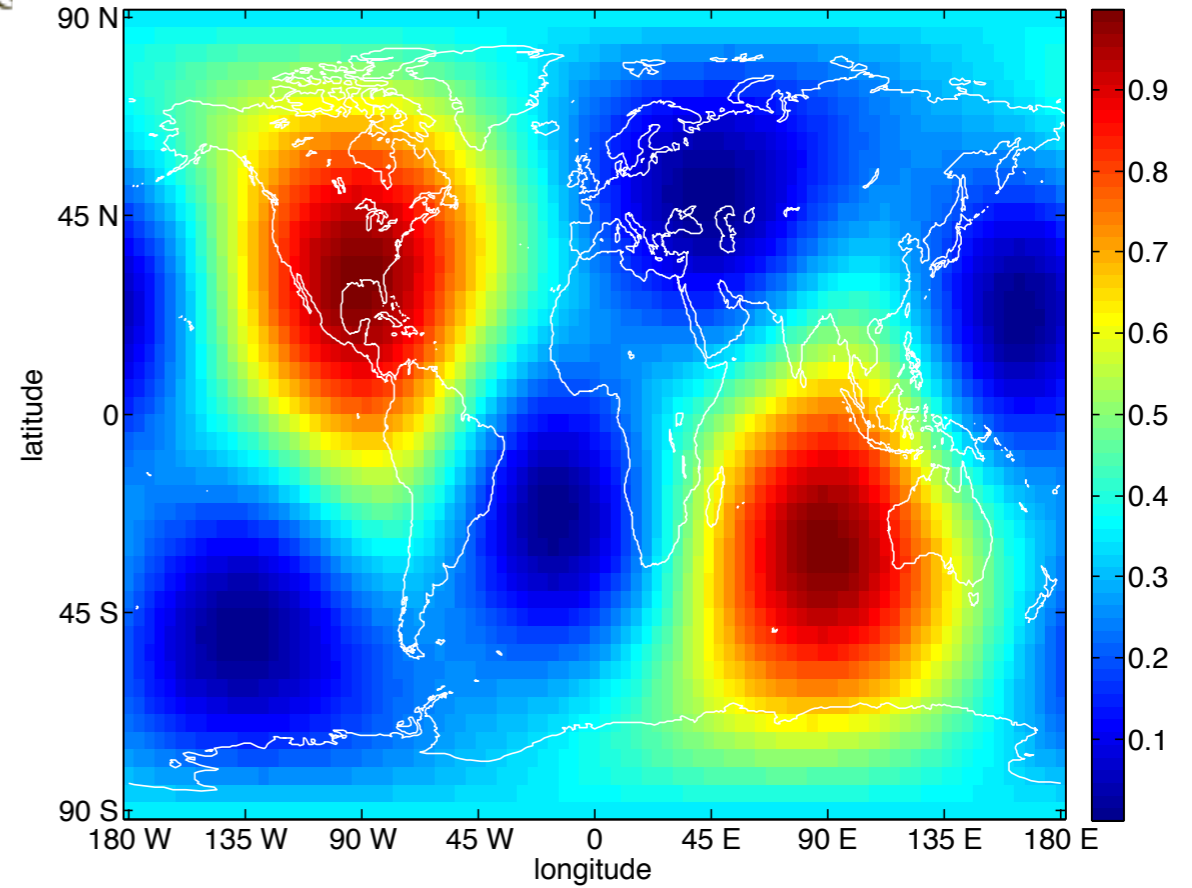
こうした情報を用いて重力波と雑音を区別していく。

アンテナパターン($F_+^2 + F_x^2$)

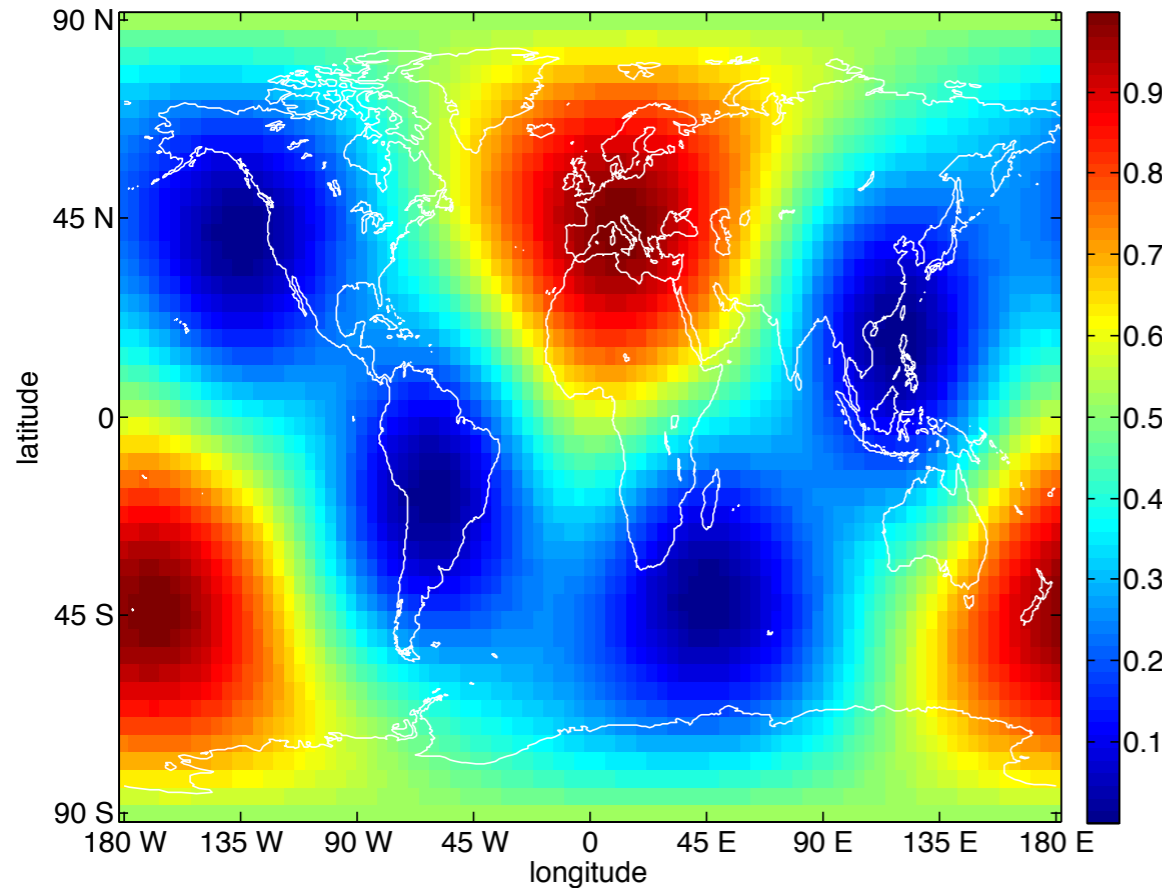
Antenna pattern for H1



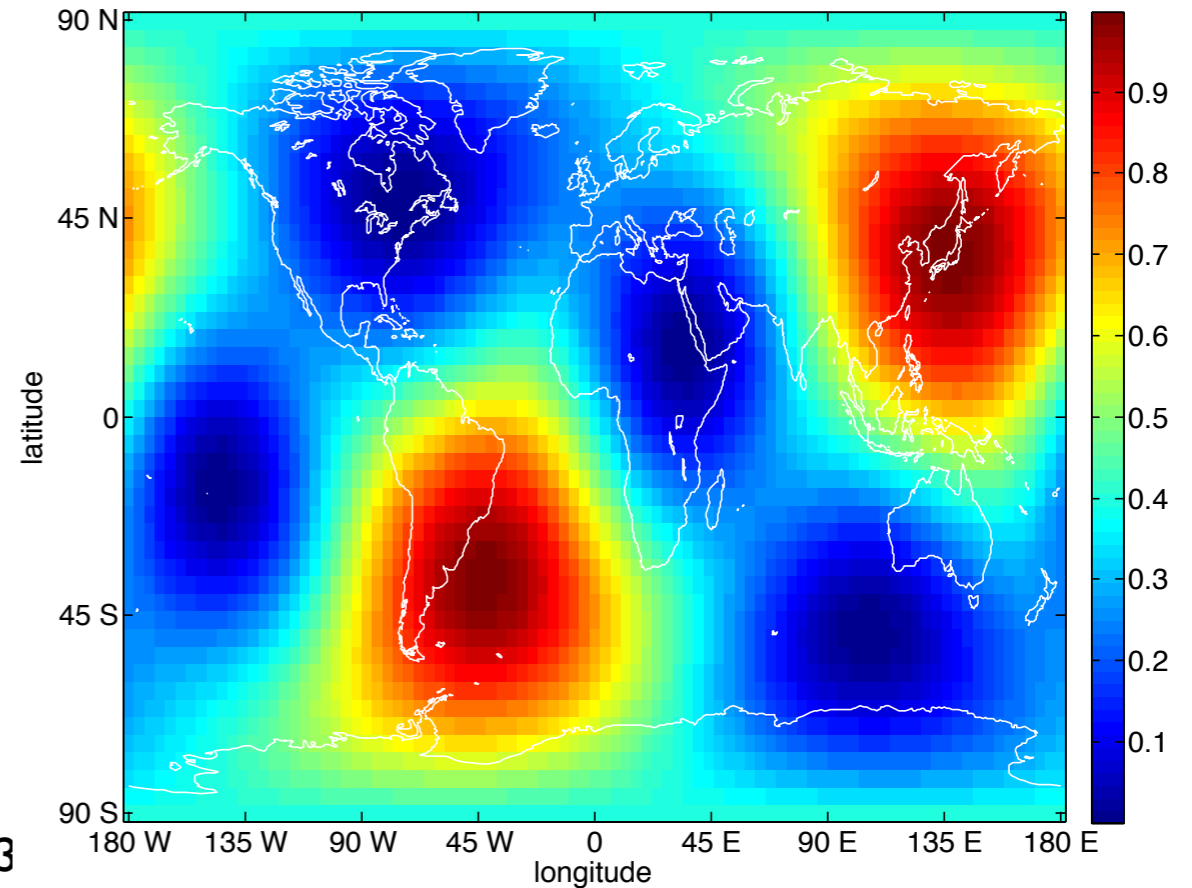
Antenna pattern for L1



Antenna pattern for V1



Antenna pattern for KAGRA





● enshu_dataにはLIGO Hanford(H1)、LIGO Livingston(L1)、Virgo(V1)、KAGRA (LCGT)のデザイン感度を持ったデータが含まれている。

● 各データは次のような構造体になって格納されている

H1-H1.detector : 望遠鏡の名前

-H1.GPSinfo : 保存されているデータのGPS時間

-H1.fs : サンプルング周波数

-H1.data : 時系列データ

GPSinfo(i)-GPSinfo.detector : i番目の望遠鏡の名前

-GPSinfo.GPSinfo.start_sec: データ開始のGPS時間(秒)

-GPSinfo.GPSinfo.nstart_nsec: データ開始のGPS時間(秒の小数点以下)

-GPSinfo.GPSinfo.stop_sec: データ終了のGPS時間(秒)

-GPSinfo.GPSinfo.nstop_nsec: データ終了のGPS時間(秒の小数点以下)



Characteristic strain amplitude

$$h_c = \left(3 \int_0^\infty \frac{S_h(f_c)}{S_h(f)} h(f) h(f)^* f df \right)^{1/2}$$

Characteristic frequency

$$f_c = \left(\int_0^\infty \frac{h(f) h(f)^*}{S_h(f)} f df \right) \left(\int_0^\infty \frac{h(f) h(f)^*}{S_h(f)} df \right)^{-1}$$

Thorne, K. S., "gravitational wave" in 300 Years of Gravitation, (Eds.) S. W. Hawking and W. Israel, 1987, (Chicago: Univ. of Chicago Press)

H1



$$r = [-2.161414928e6, -3.834695183e6, 4.600350224e6];$$

$$a = [-0.223891216, 0.799830697, 0.556905359];$$

$$b = [-0.913978490, 0.026095321, -0.404922650];$$

L1

$$r = [-7.427604192e4, -5.496283721e6, 3.224257016e6];$$

$$a = [-0.954574615, -0.141579994, -0.262187738];$$

$$b = [0.297740169, -0.487910627, -0.820544948];$$

V1

$$r = [4.5463741e6, 8.429897e5, 4.378577e6];$$

$$a = [-0.700458309, 0.208487795, 0.682562083];$$

$$b = [-0.053791331, -0.969082169, 0.240803326];$$

KAGRA

$$r = [-3776899.062, 3483900.163, 3766657.585];$$

$$a = [-0.4300, -0.8363, 0.3400];$$

$$b = [0.6821, -0.0542, 0.7292];$$



※マークの項目は早くできた人用で、オプションの演習

2月20日

講義+演習1 14:00-14:45

○ 解析環境の確認

- Octave はインストールできたか？
- 必要なパッケージはインストールされているか？

いくつかのコマンドがあるかどうかチェックする

○ Octaveの基本的な文法

- octaveの文法を学ぶ

2月21日

講義+演習3 10:00-12:00

○ パワースペクトル密度の定義とデータのスペクトルを求める

- FFTを用いてパワースペクトル密度を計算せよ

○ データコンディショニング：カラーノイズをホワイトニングする。

- linear predictor error filterを用いてデータをホワイトニングし、埋め込まれている重力波信号を確認せよ。



講義+演習4 14:30-17:30

単一望遠鏡でのバーストサーチ

o アルゴリズムの解説と実装

- 以下のアルゴリズムを記述したコードを完成させよ

-- Excess power

o 実装したアルゴリズムでサーチ

- 実装したアルゴリズムを用いて、与えられたデータを解析し、埋め込まれたバースト重力波信号を検出せよ

-- データをホワイトニングせよ

-- ホワイトニングされたデータに対し、バーストサーチせよ

※ -- 検出したバースト重力波の特徴パラメータ（演習2を参照）を計算せよ。

2月22日

講義+演習5 9:00-12:00

LIGO、Virgo、KAGRAのシミュレーションデータを用いたコインシデンス解析

o 各望遠鏡のデータをコンディショニング+バーストサーチ

- それぞれの望遠鏡からのデータをホワイトニングせよ

- ホワイトニングしたデータに対しバーストサーチをせよ

※ - イベントリストと、各イベントの特徴パラメータを計算せよ

o 時間窓を用いたコインシデンス解析

- 各望遠鏡の座標系を利用して時間窓を求めよ

※ - 与えられたアンテナパターンマップや、時間窓、抽出した特徴パラメータからフェイクイベントを除去せよ。(Veto解析)



答え合わせ 12:00-12:50

- いくつ真のイベントを検出できたから
 - いくつ偽のイベントを検出してしまったか
 - 特徴パラメータは正しかったか
- をチェックせよ。