

19.11.13 Fujikawa meeting

ON-SITE WORK REPORT

Overview

1

期間

10/28 (月) ~ 11/8 (金)

主な作業内容

- ROOTによるヒストグラムの作成
- ロックロスの調査
- fujimopy
- Public_html

ROOT

- ・CERNで開発されたツール
- ・ビッグデータの処理に活用される
- ・統計的・視覚的に解析できる
- ・柏サーバで使える

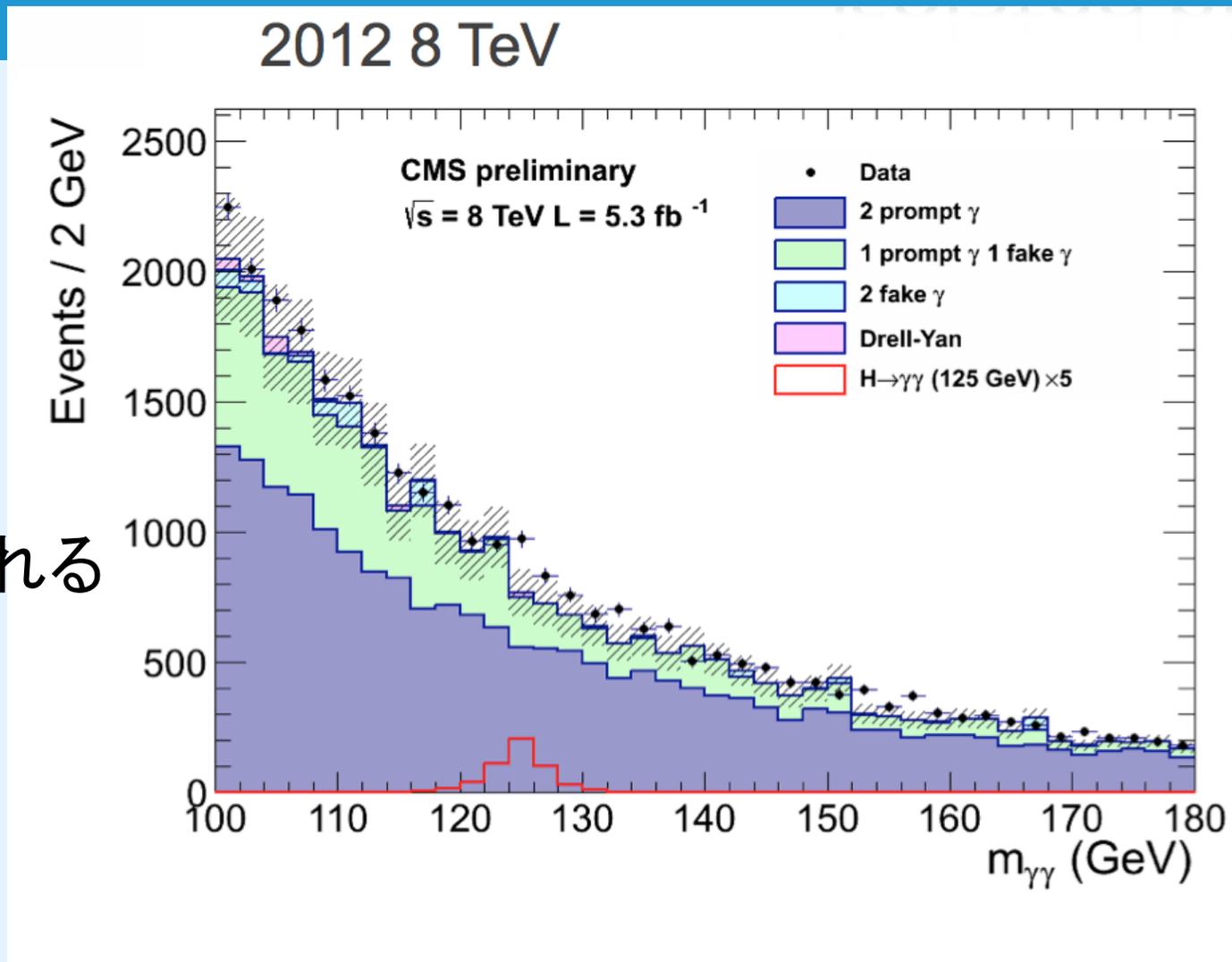


図: ヒストグラムの例 <https://root.cern.ch/gallery>

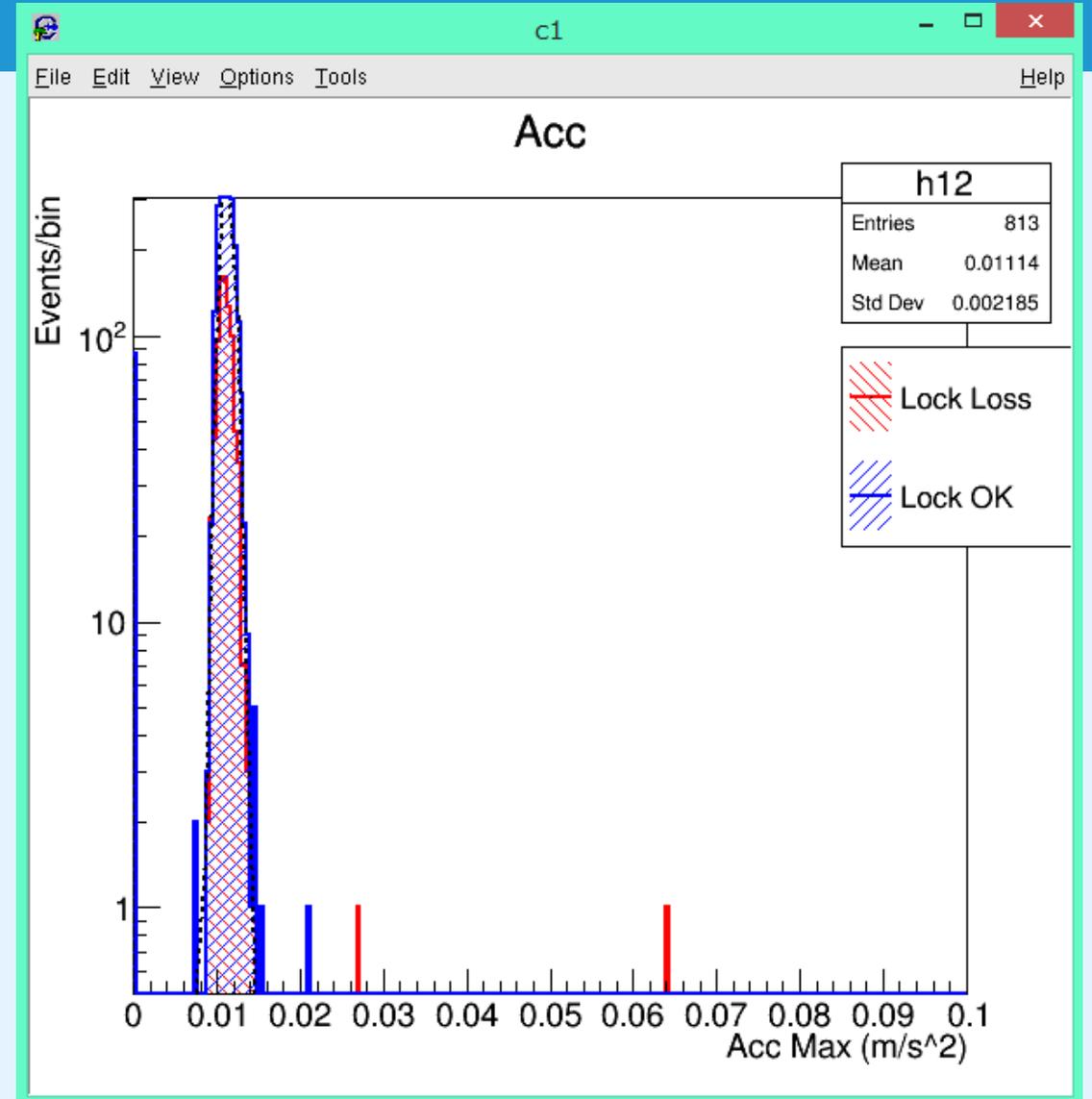
柏サーバーでROOT

ROOT.TNtuple('nt', 'nt', 'time:lock:mic')

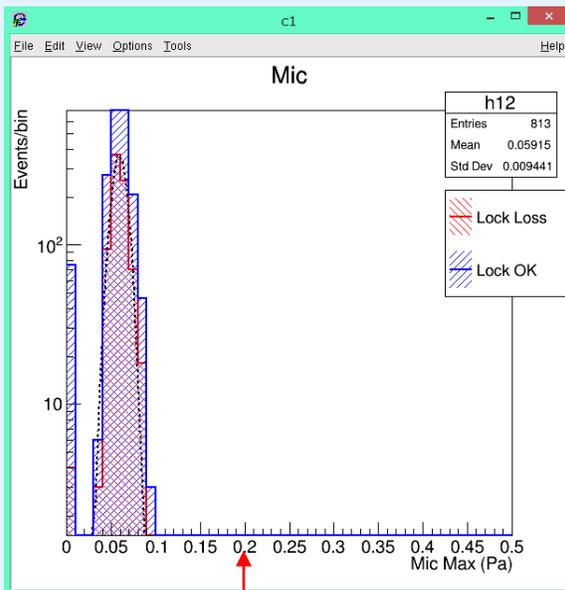
ロックの状態

PEMセンサの値

ロックロスする閾値が求められる

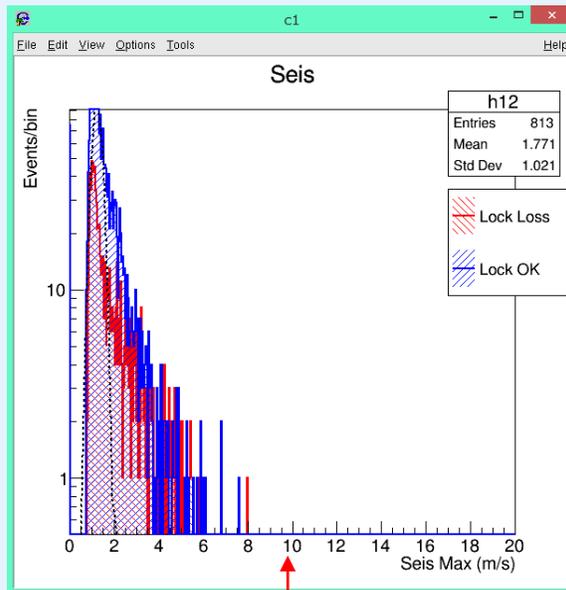


PEMの閾値



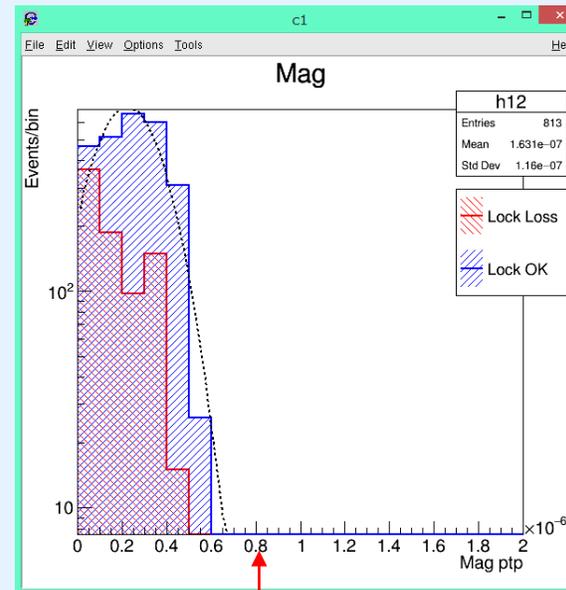
Mean = 0.059
Sigma = 0.0083

閾値: 0.2



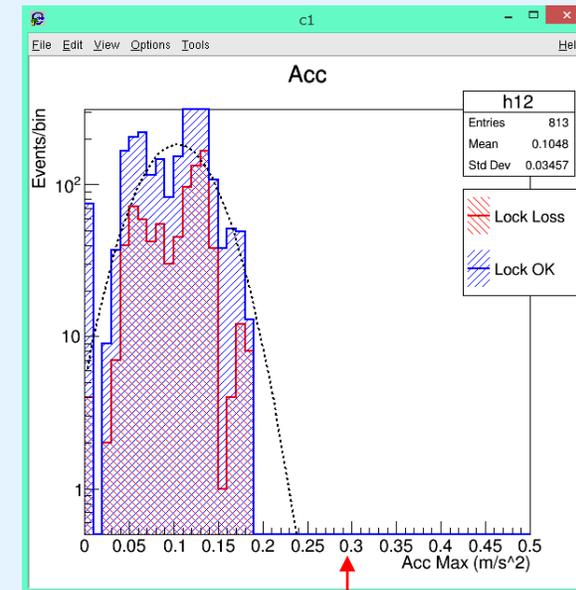
Mean = 1.2
Sigma = 0.21

閾値: 10.0



Mean = 2.2×10^{-7}
Sigma = 1.4×10^{-7}

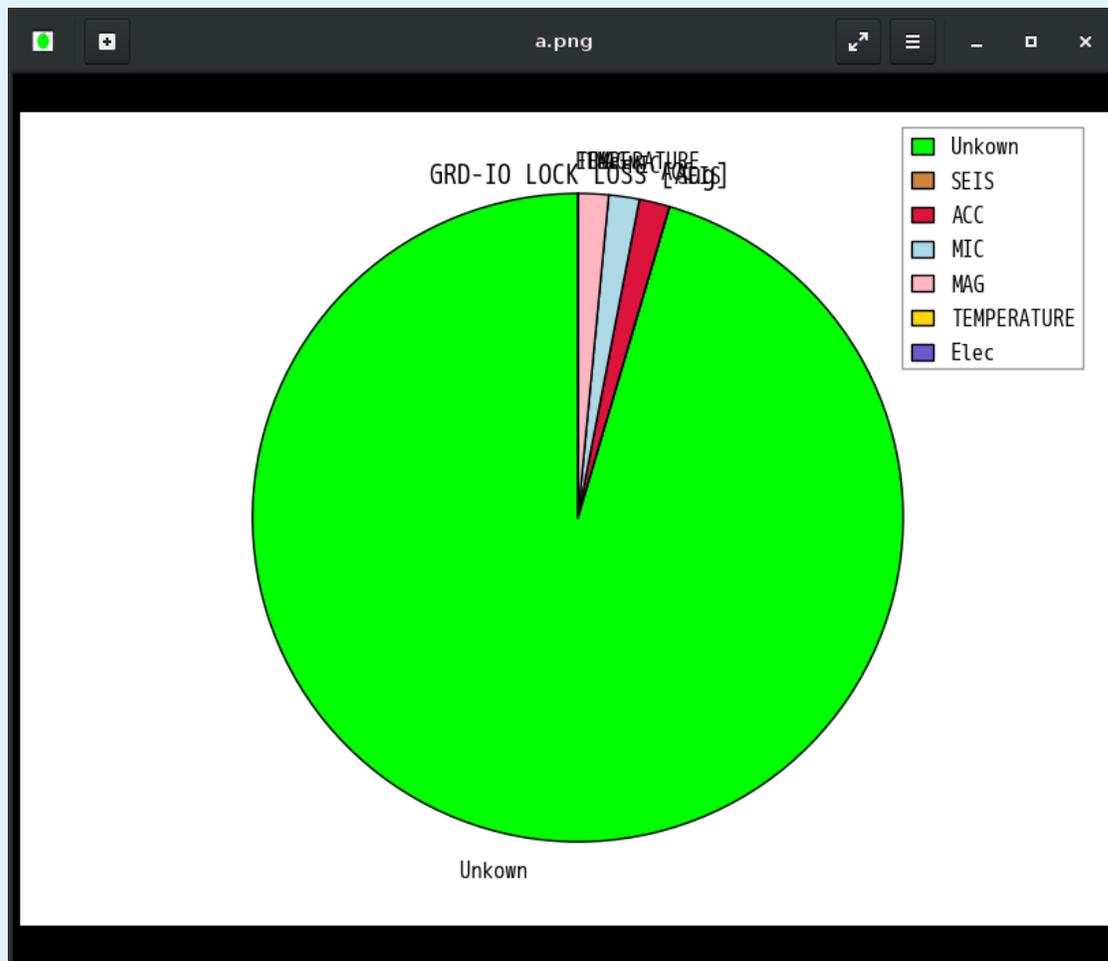
閾値: 8.0×10^{-7}



Mean = 0.10
Sigma = 0.039

閾値: 0.3

ロックロスの原因解析



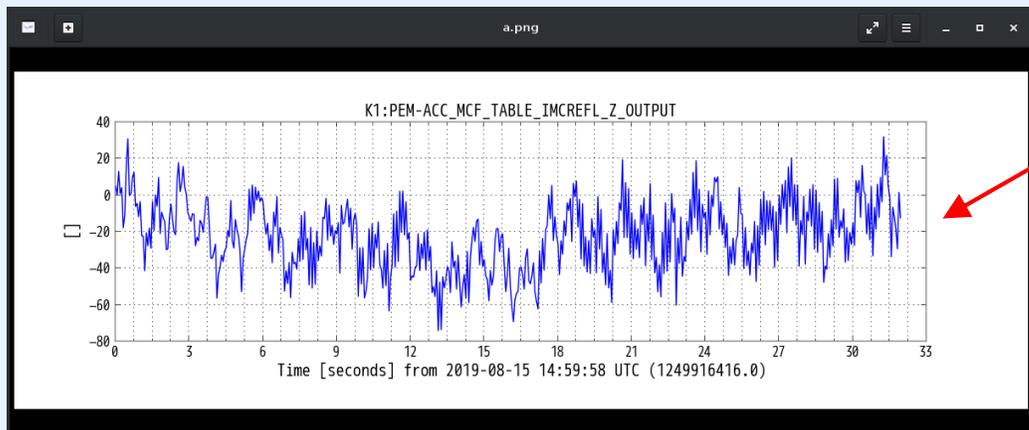
8月のIOガーディアンロックロス66個を原因ごとに分類

- ・PEMセンサ波形の peak to peak や平均値を利用
- ・**閾値を超えたPEMにフラグ**を立てる
- ・PEM以外はまだunknown
- ・PEMも時系列波形しか見れていない

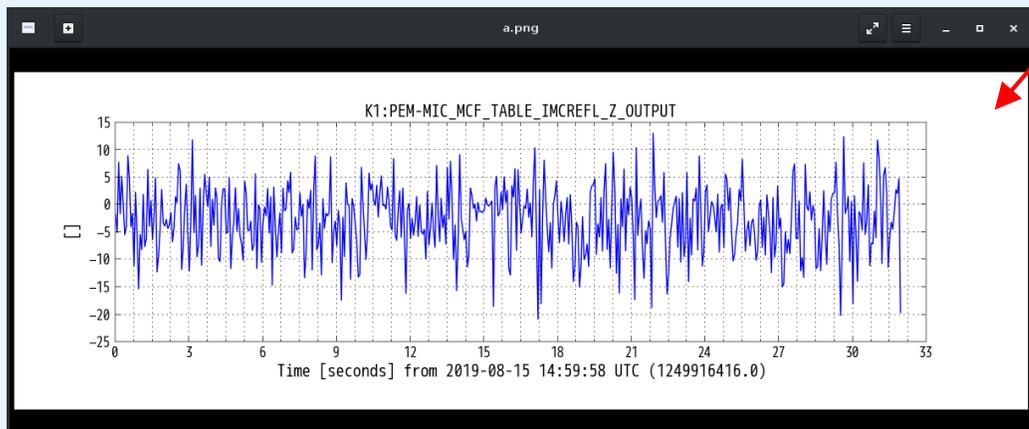
PEMの波形

```
Successfully finished!!  
====< SEIS >=====  
  
====< ACC >=====  
1249916416.0  
  
====< MIC >=====  
1249916416.0  
  
====< MAG >=====  
1249916416.0  
  
====< TEMP >=====  
  
[yuta.fujikawa@m31-01 analyze]$
```

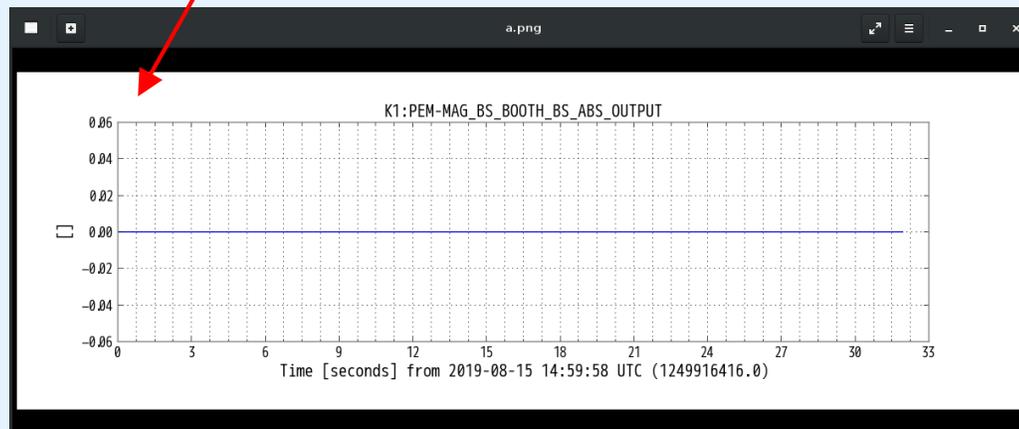
3つとも同じ時刻



とくに特徴はない

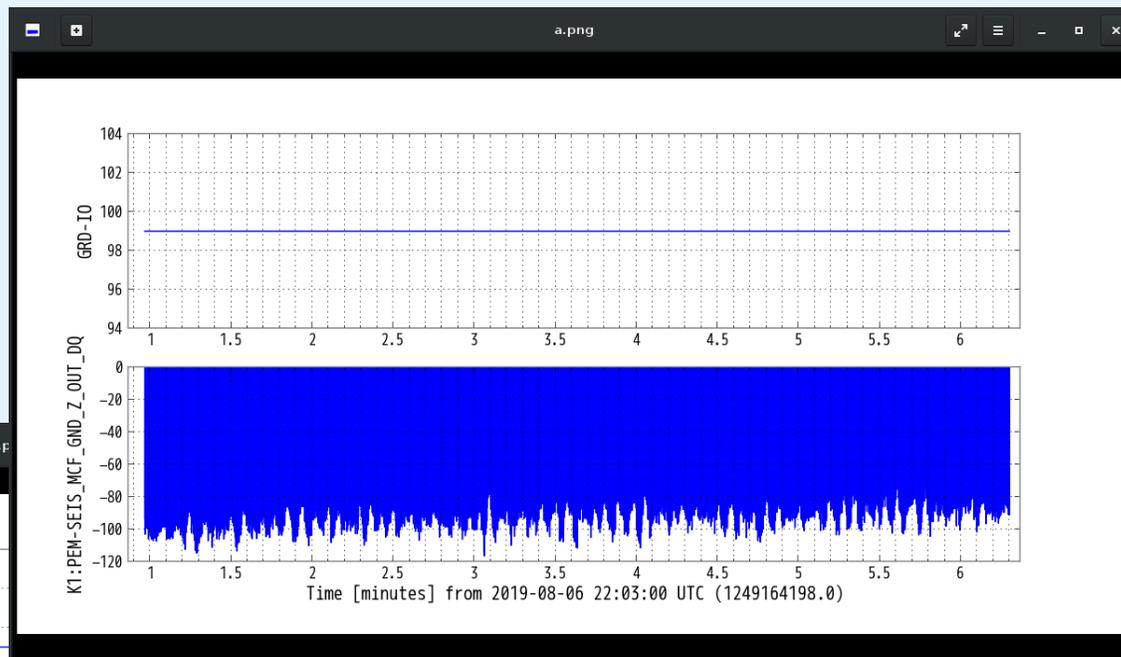


むしろ振幅0



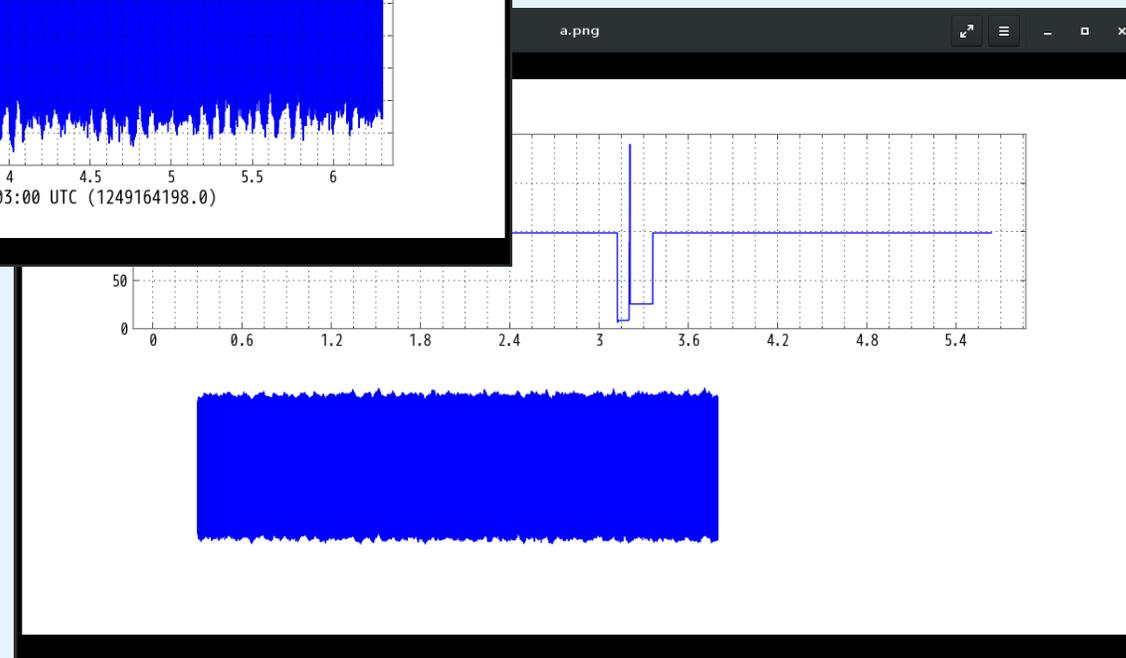
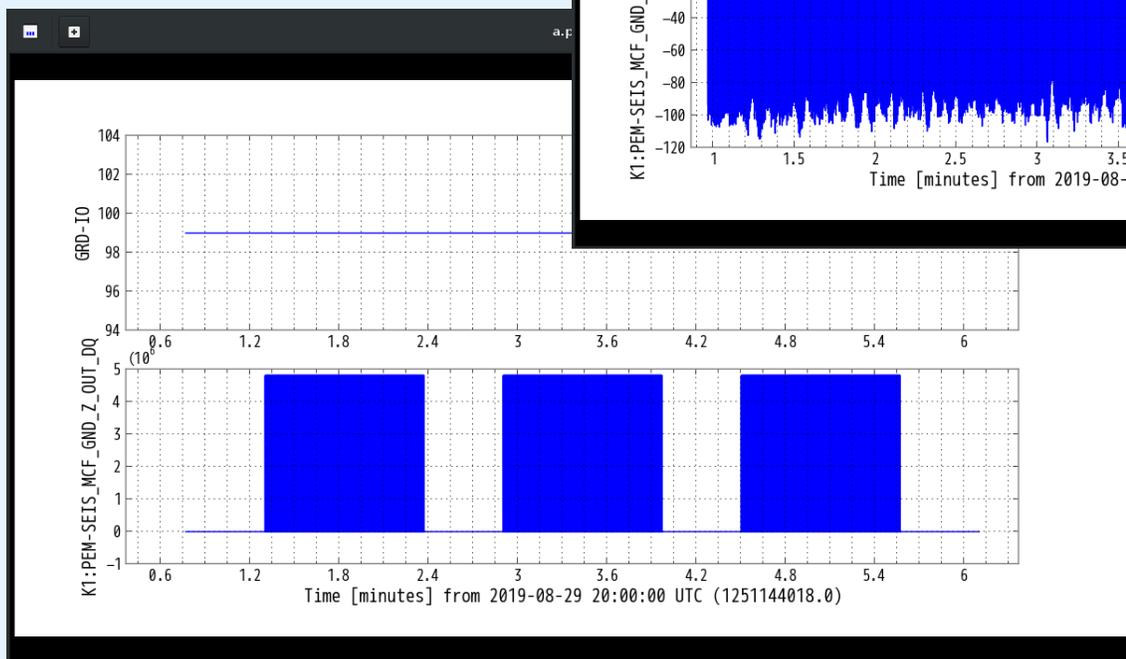
サチュレーションの例

これらの値も調べて
排除する必要がある

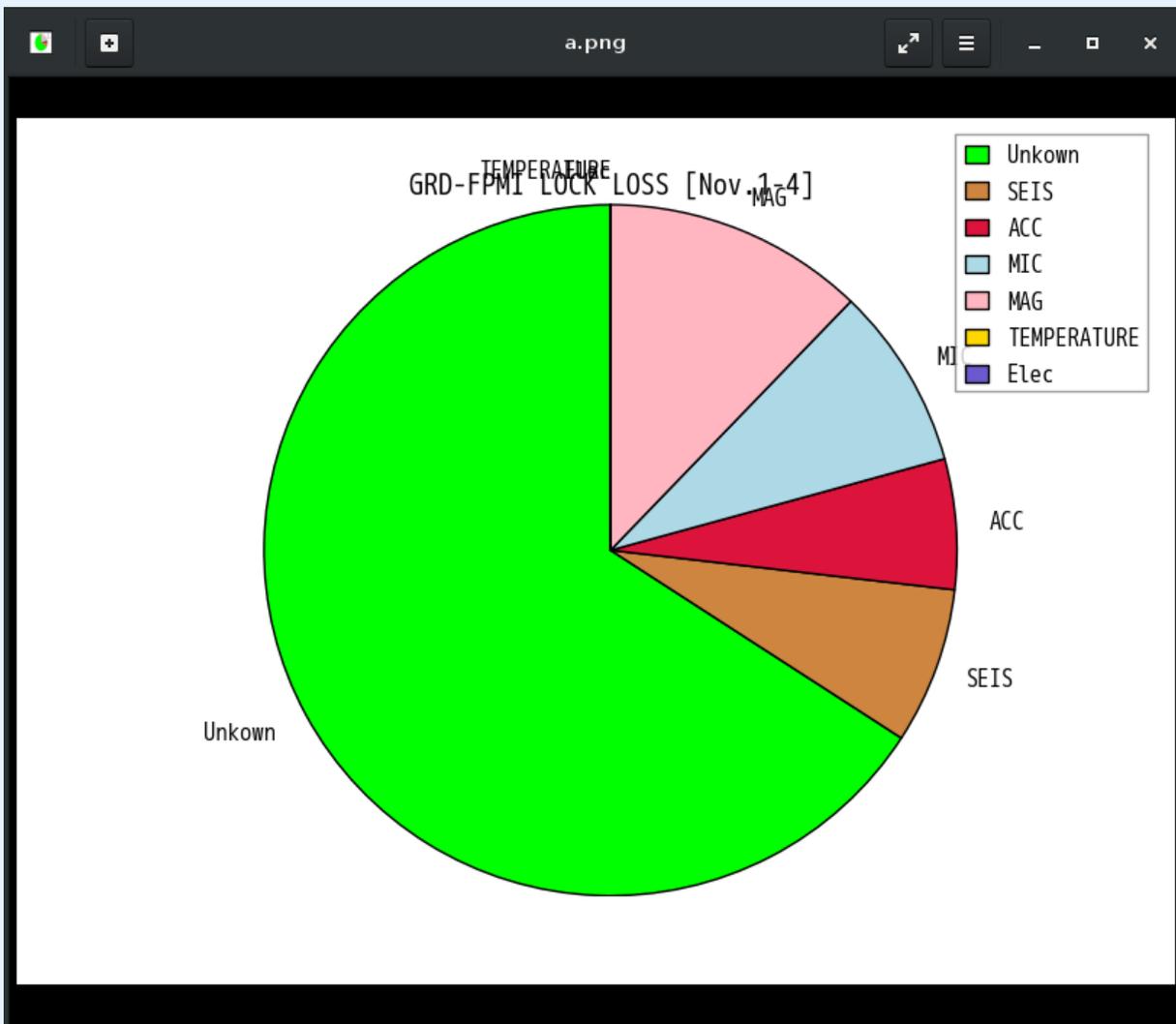


0より上がない

途中までしかない&
軸やタイトルが出ない



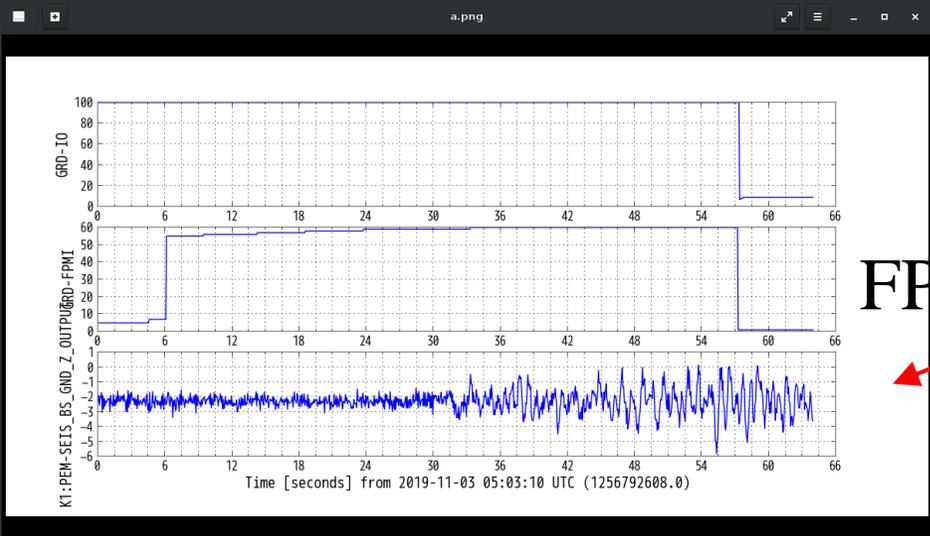
FPMIのロックロス



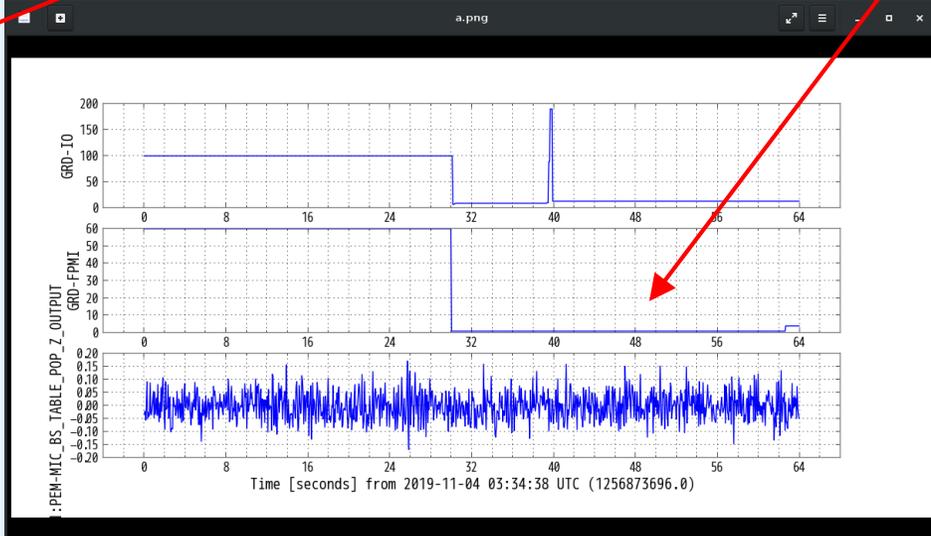
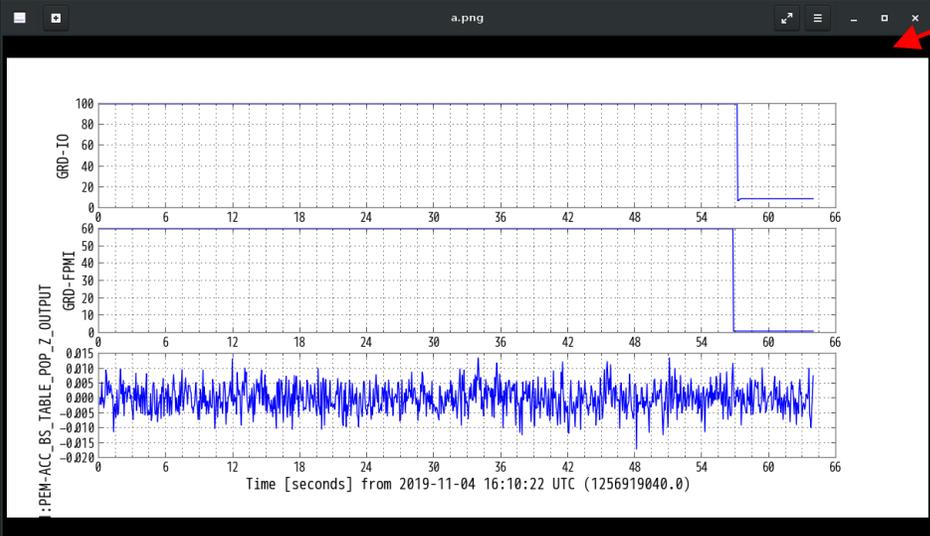
BSに置かれたセンサしか見ていない

```
yuta.fujikawa@m31-01:~/analyze/root/fpmi
[yuta.fujikawa@m31-01 fpmi]$
[yuta.fujikawa@m31-01 fpmi]$ emacs -nw lockloss.py
[yuta.fujikawa@m31-01 fpmi]$ python lockloss.py
100%
Successfully finished!!
exit
====< SEIS >====
1256672288.38
1256674801.19
1256677344.19
1256758471.62
1256792665.25
1256857449.75
====< ACC >====
1256702071.44
1256739223.56
1256776909.56
1256784816.25
1256919096.81
====< MIC >====
1256683200.62
1256709721.62
1256791035.94
1256812769.06
1256865354.31
1256871972.69
1256873726.0
====< MAG >====
1256666991.25
1256667977.75
1256672288.38
1256674801.19
1256677344.19
1256678777.81
1256680968.44
1256683200.62
1256683681.62
1256708070.44
====< TEMP >====
```

PEMの波形



FPMIは正しく検出できていそう。



```
Successfully finished!!
exit
====< SEIS >=====
1256672288.38
1256674801.19
1256677344.19
1256758471.62
1256792665.25
1256857449.75
====< ACC >=====
1256702071.44
1256739223.56
1256776909.56
1256784816.25
1256919096.81
====< MIC >=====
1256683200.62
1256709721.62
1256791035.94
1256812769.06
1256865354.31
1256871972.69
1256873726.0
====< MAG >=====
1256666991.25
1256667977.75
1256672288.38
1256674801.19
1256677344.19
1256678777.81
1256680968.44
1256683200.62
1256683681.62
1256708070.44
====< TEMP >=====
```

原因ではない
気がしますが

Overview

期間

10/28 (月) ~ 11/8 (金)

主な作業内容

- ROOTによるヒストグラムの作成
- ロックロスの調査
- fujimopy
- Public_html

About fujimopy

10

- Fujikawa と Mori さんで作成
- スペクトルのピークを検出して表示する
- 様々な指定をしてピークを検出できる
- バージョン1:ローカルでピーク検出
- バージョン2:柏サーバのデータのピーク検出
- バージョン3:ピークの時間推移

parameter setting

File name: data/acc-BS-1.dat

graph start: 0

graph end: 5000

freq start: 10

freq end: 100

order: 10

peak number: 0

peak cut: 0.0001

fcut

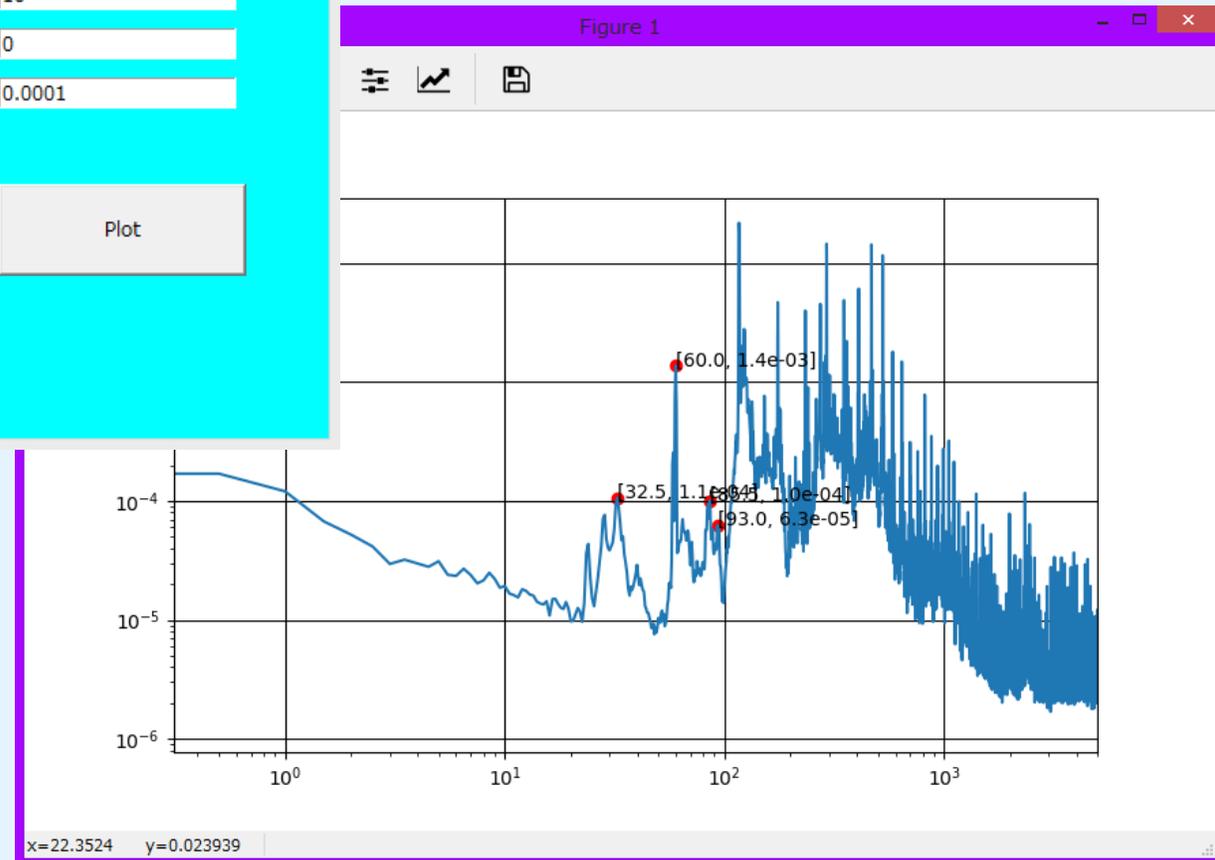
pcut

peakn

xlog

Plot

バージョン1.7

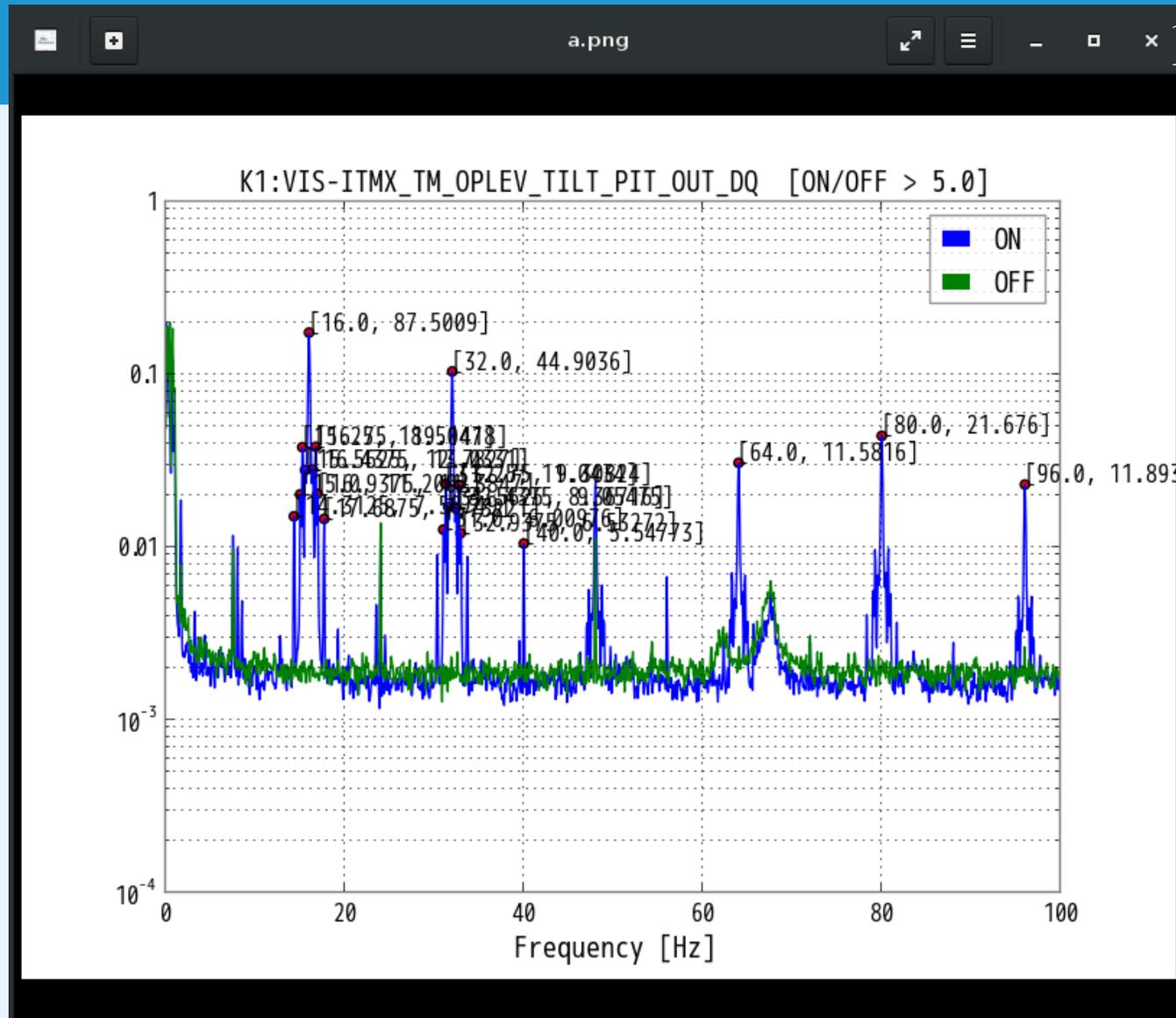


ON/OFFujmopy

ON/OFFでn倍になった
ピークだけを検出

k1sumxで使えるようにする
htmlの方も進めていく

10回の平均を使用



- 分類の精度を高める & Unknownの割合を減らす
- 時系列波形以外も見る
- グリッチなどの見つけ方を考える
- 制御信号などによる原因も組み込む
- fujimopy, htmlもアップデートする
- 次回は11月25日～

Appendix