WaveMon

Sergey Klimenko

University of Florida

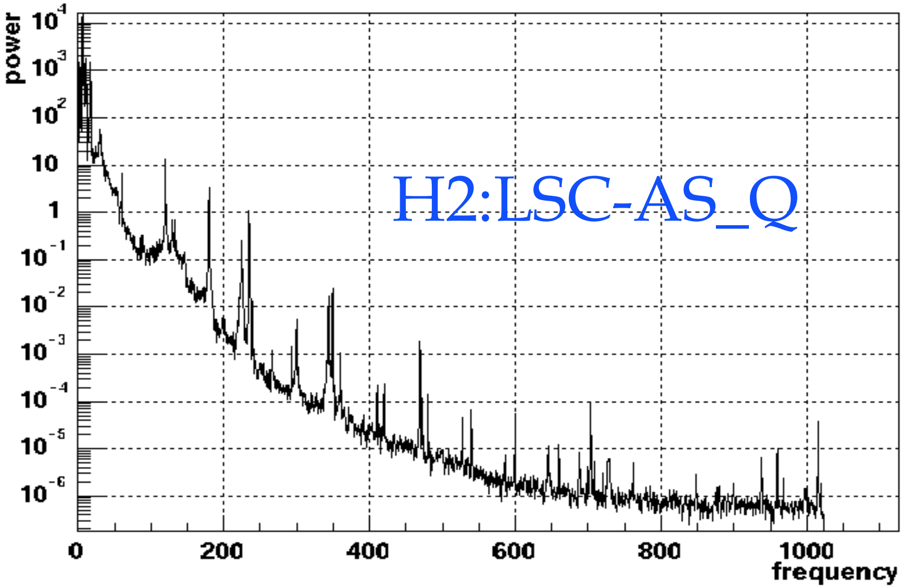
- **What is WaveMon?**
- **WaveMon algorithms**
- **WaveMon results**
- **Summary**

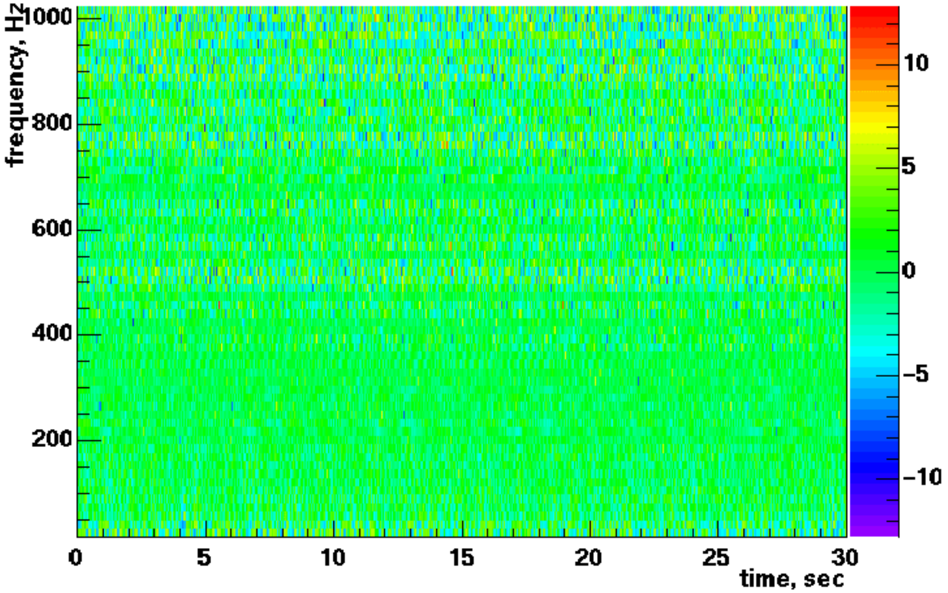
WaveMon

- Purpose
  - monitor non-stationary detector noise (glitches)
  - generate VETO triggers for specified environmental and control channels
- Method
  - Time-frequency analysis in wavelet domain.
- Algorithms
  1. wavelet transform: produce wavelet TF plots for master (GW) and slave (notGW) channels. There could be several slave channels.
  2. percentile transform: select a certain fraction (20%-30%) of TF pixels using the "percentile selection rule".
  3. coincidence: produce a coincidence TF plot using "zero time window coincidence rule" for master and slave channels (for the same interferometer)
  4. cluster analysis: find clusters and apply cluster selection cuts using cluster size, asymmetry and likelihood
- Output
  - VETO triggers (estimate TF phase space available for GW bursts)
  - false alarm triggers (estimate false alarm rate)
  - dmt-viewer & trend data

Wavelet Transform

- interpolating bi-orthogonal wavelet transform
- binary wavelet tree (linear frequency scale)

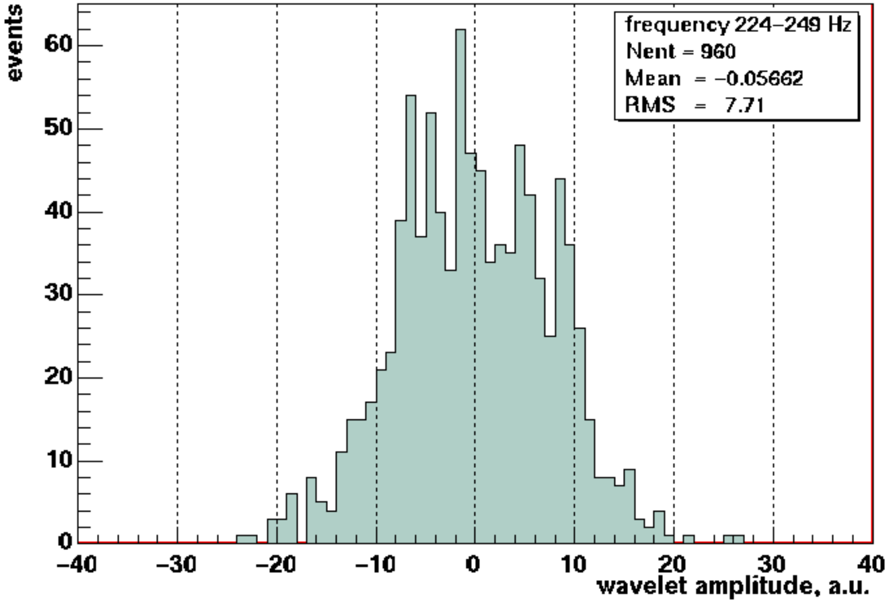


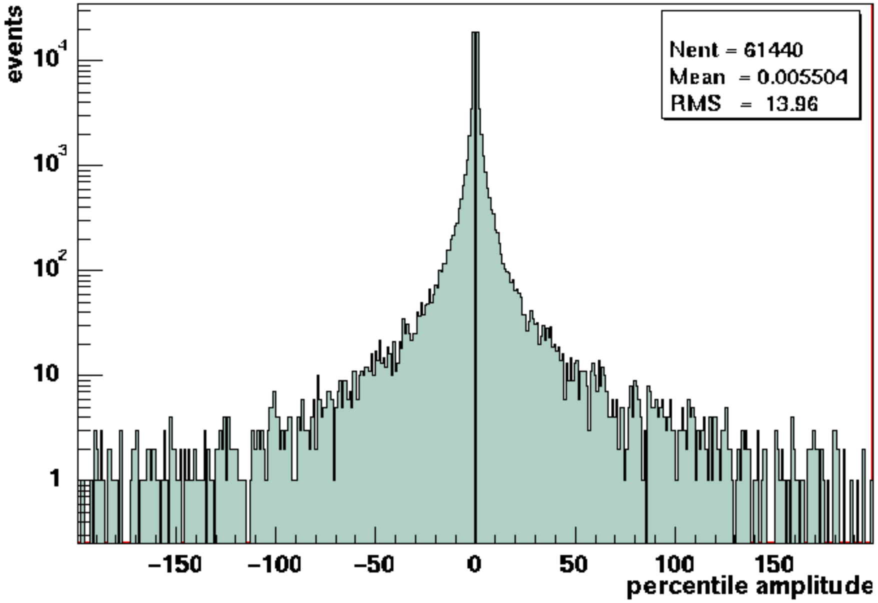


- What we do to handle large dynamic range and non-Gaussian detector noise?

Percentile Transform

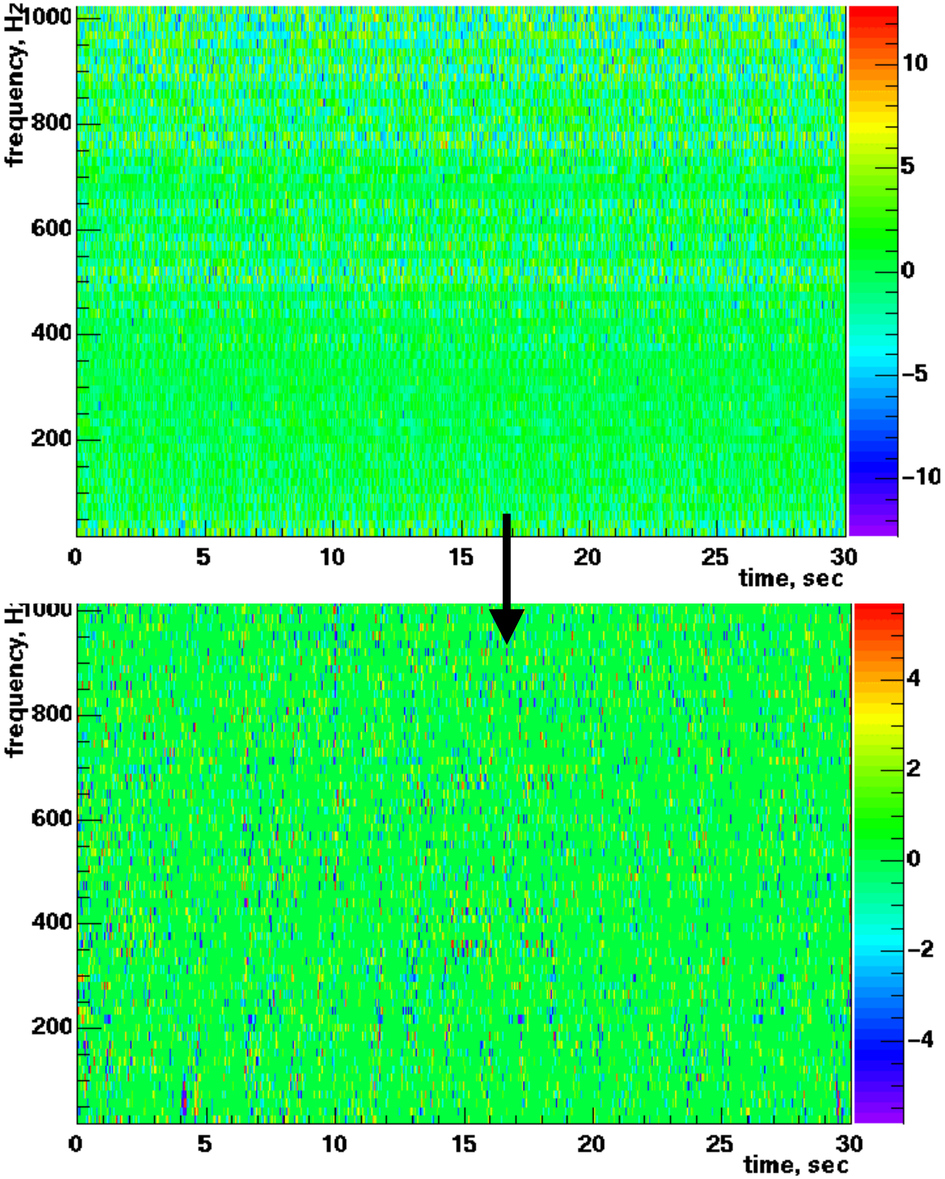
- given a sample with amplitude *A*, define its *percentile amplitude* as $a=1/f$, where *f* is fraction of samples with absolute value of amplitude greater then |*A*|
- percentile amplitude distribution function: $P(a)=1/a^2$
  - the same for all wavelet layers



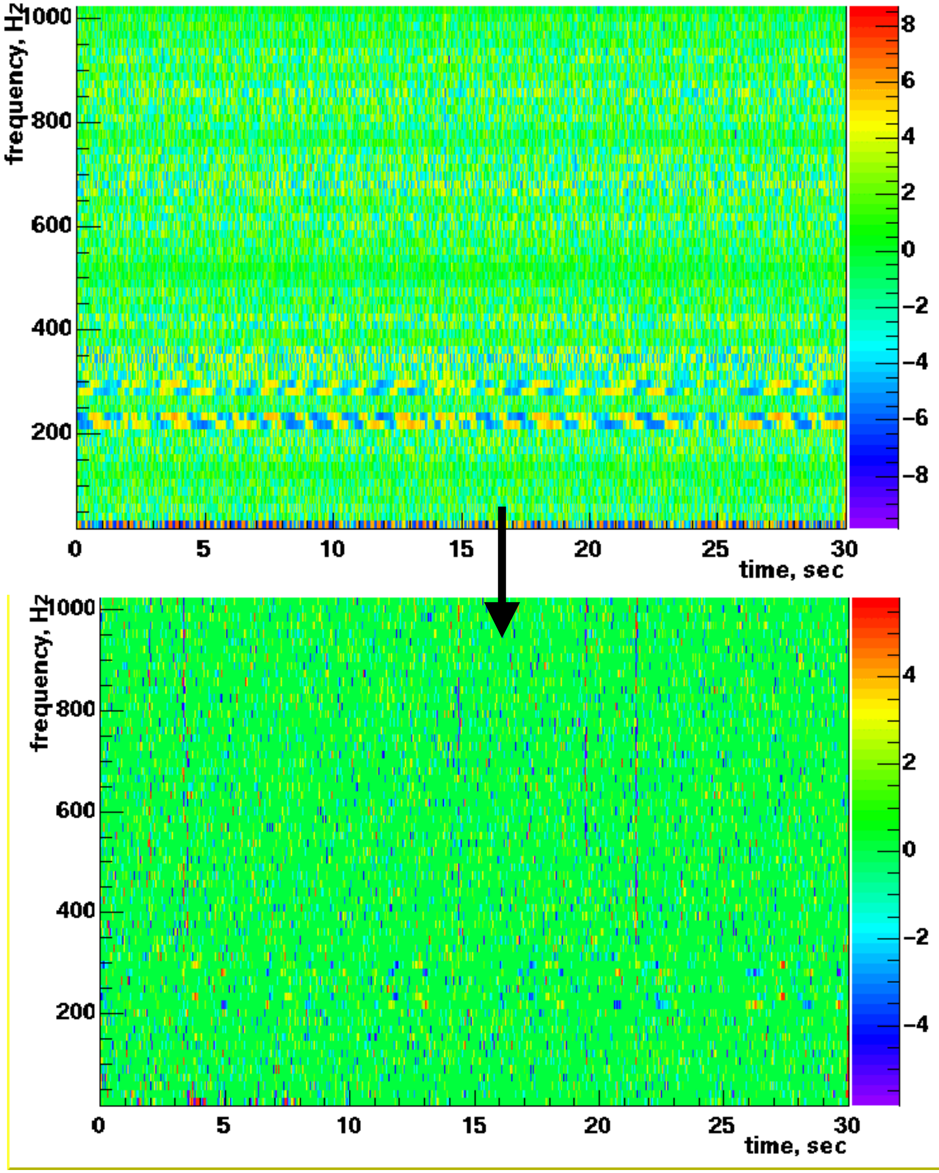


raw → percentile TF plots

H2:LSC-AS_Q

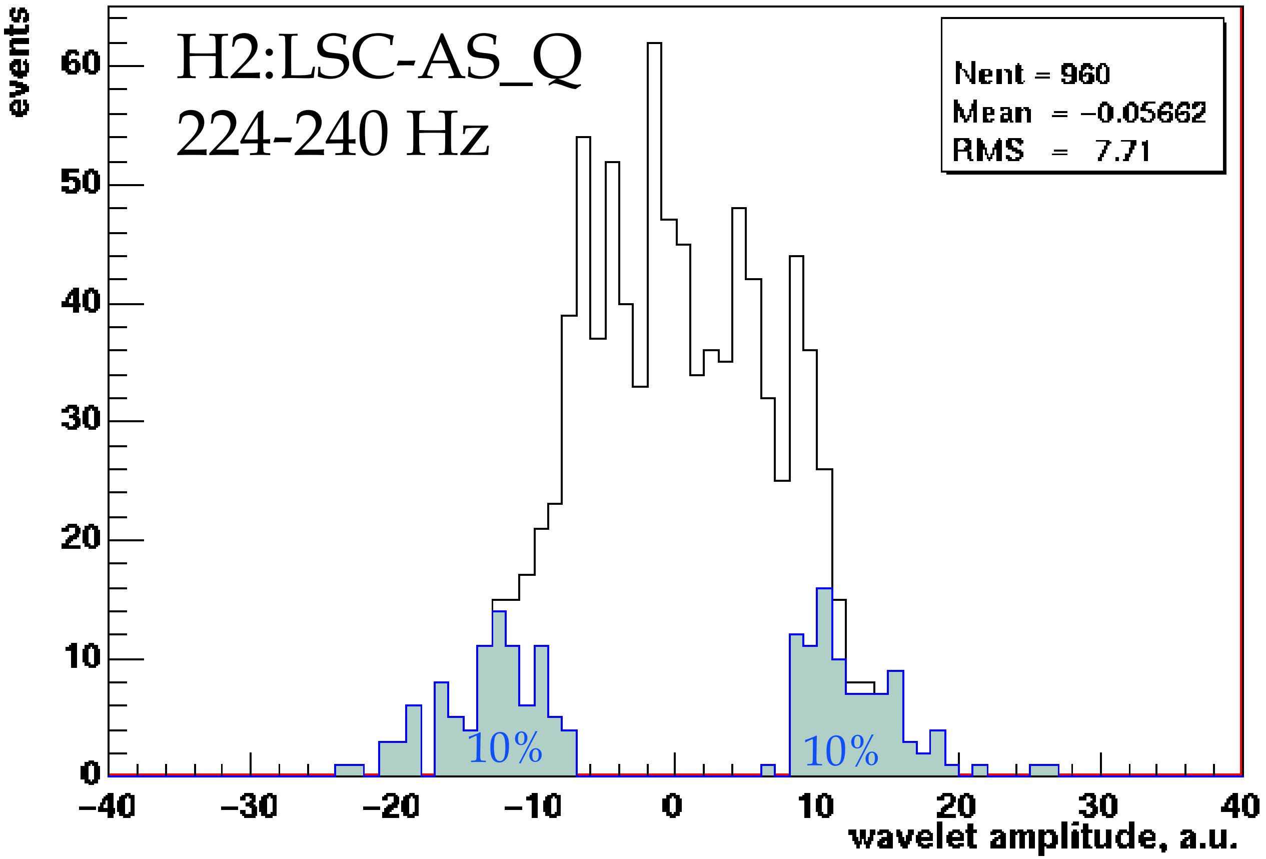


H2:IOO-MC_F



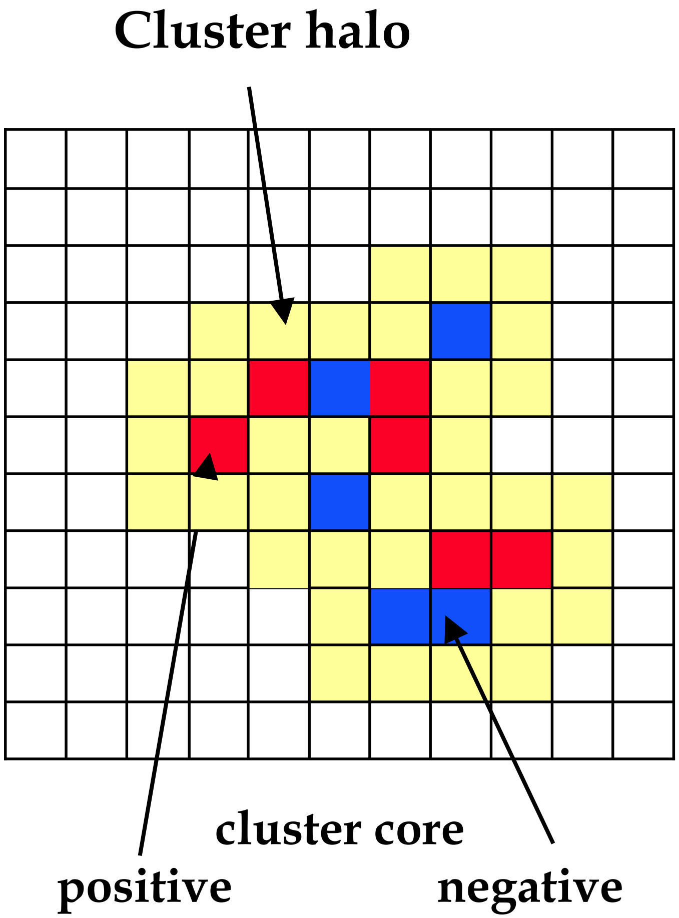
Percentile Thresholding Rule

- threshold on percentile amplitude
- select a certain fraction of samples (reduce T-F occupancy)
- don’t care about the data distribution function
- handle the data dynamic range



Cluster Analysis

cluster – T-F plot area with high occupancy



Cluster halo

cluster core

positive

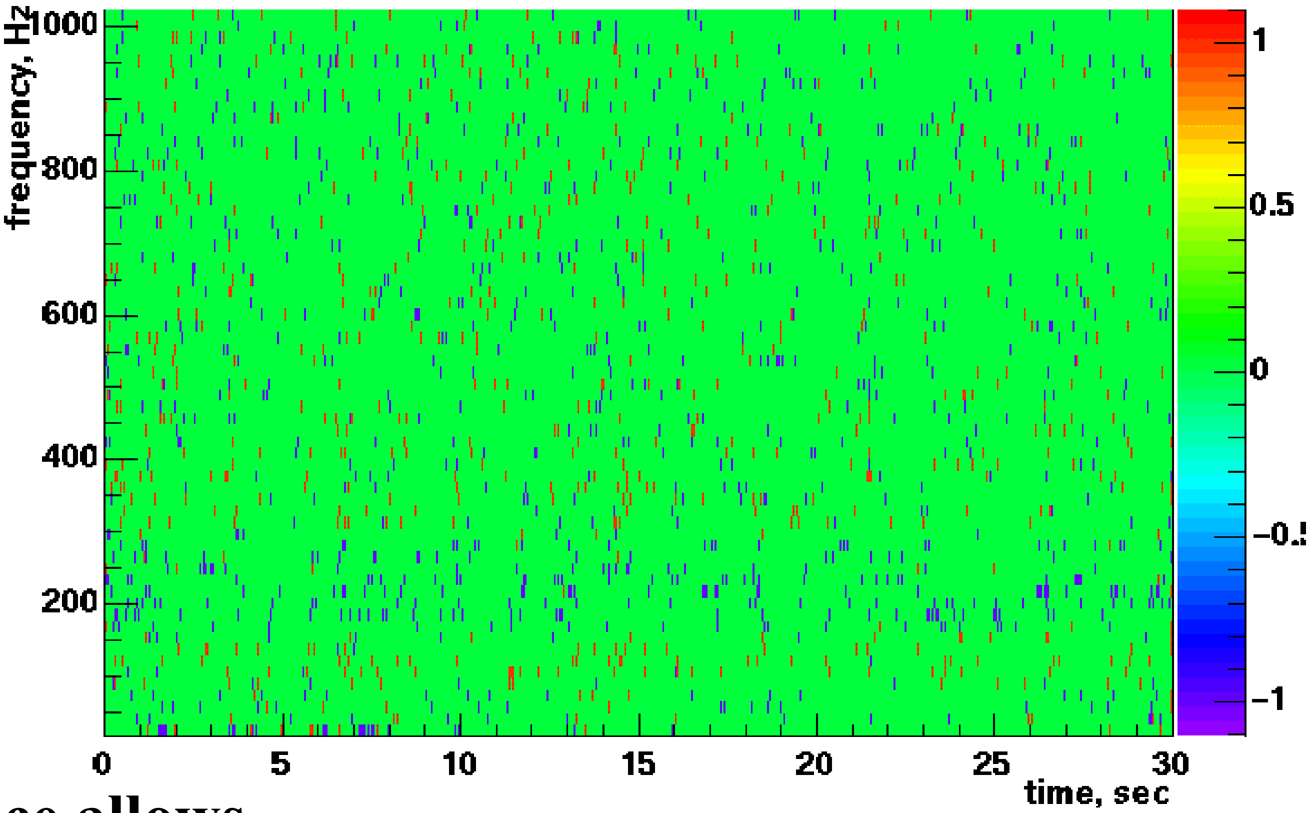
negative

Cluster Parameters

size	– number of pixels in the core
volume	– total number of pixels
density	– size/volume
amplitude	– maximum amplitude
energy	- total energy
asymmetry	– (#positive - #negative)/size
likelihood	– $\log(\Pi A_i)$
neighbors	– total number of neighbors
frequency	- core minimal frequency [Hz]
band	- frequency band of the core [Hz]
time	- GPS time of the core beginning
interval	- core duration in time [sec]

coincidence

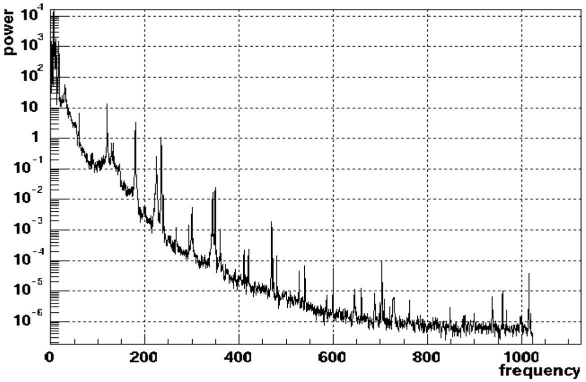
- coincidence TF plot for LSC-AS_Q && IOO-MC_F

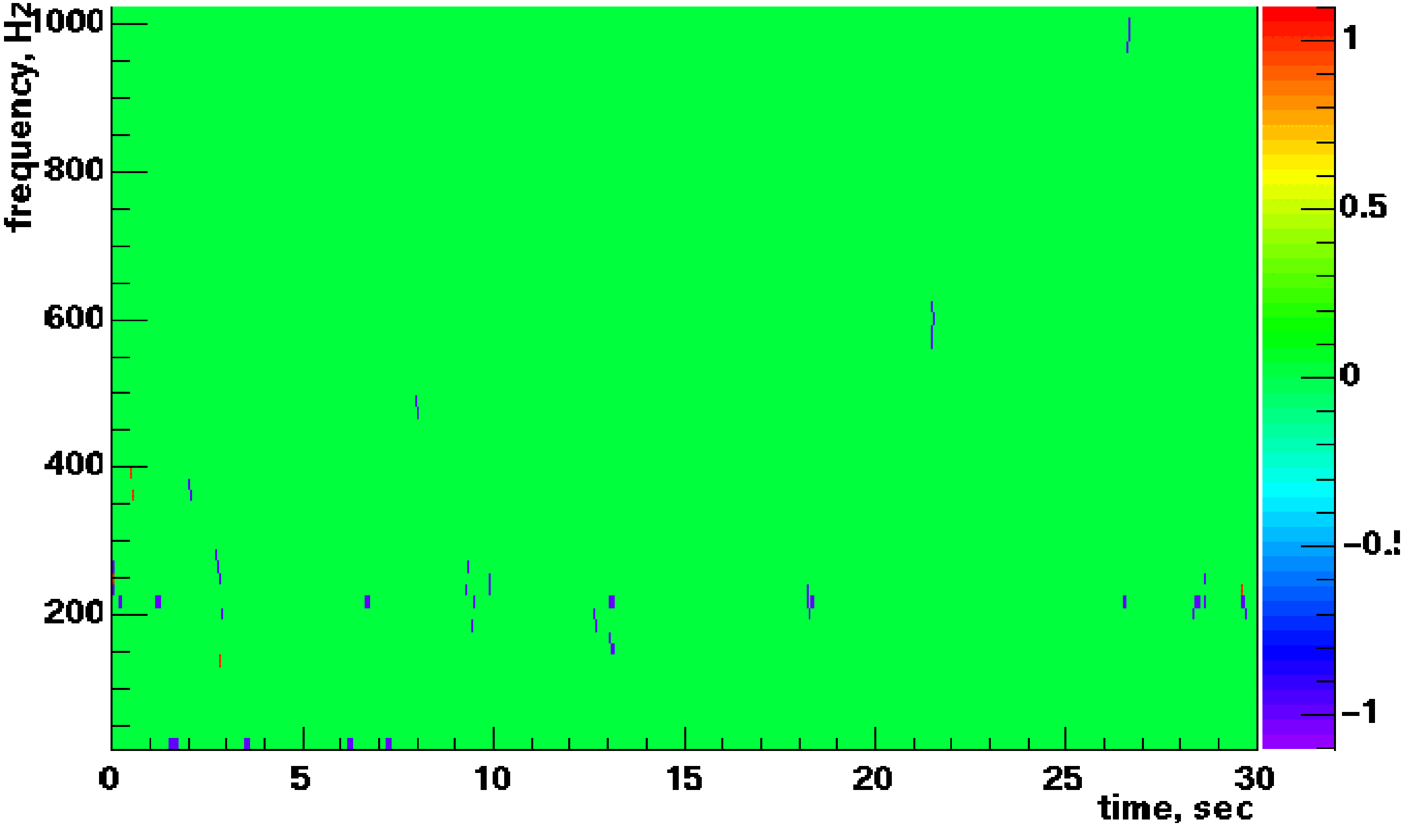


- coincidence allows
  - to set a low threshold at the previous (thresholding) step
  - to “build” clusters with known false alarm rate.
- WaveMon allows different (or non) coincidence policies

cluster selection

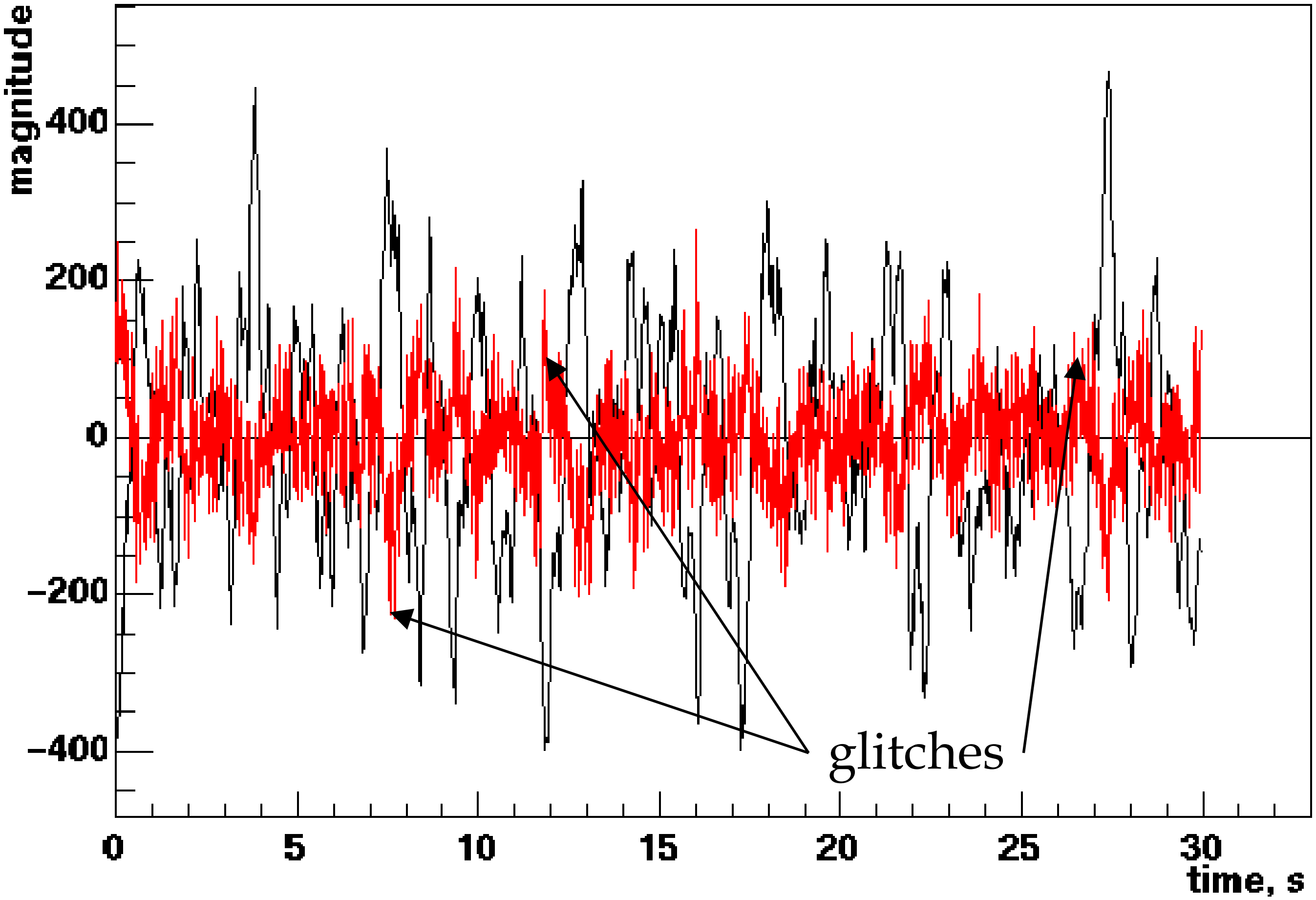
- size selection cut (>3)
- asymmetry selection cut





Wavelet coefficients (t)

- **Frequency band 224-240 Hz (wavelet layer 14)**
  - red – H2:LSC-AS_Q, black – H2:IOO-MC_F

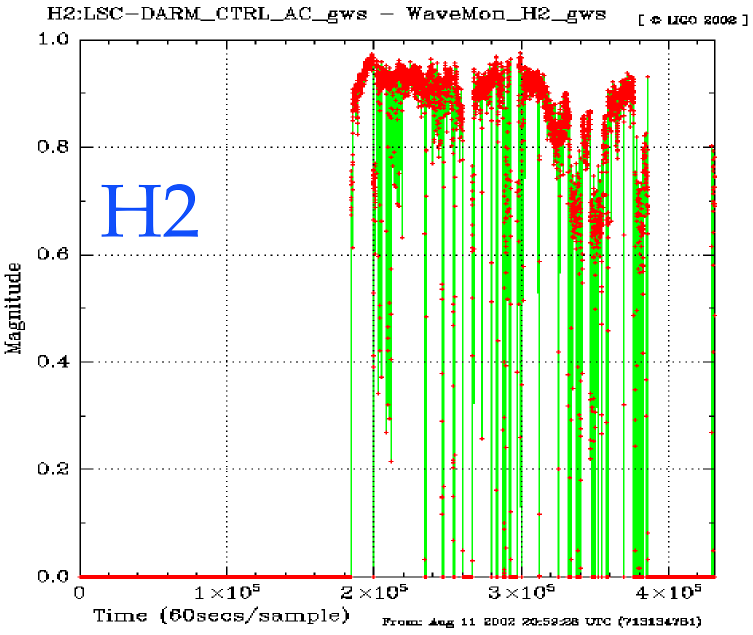


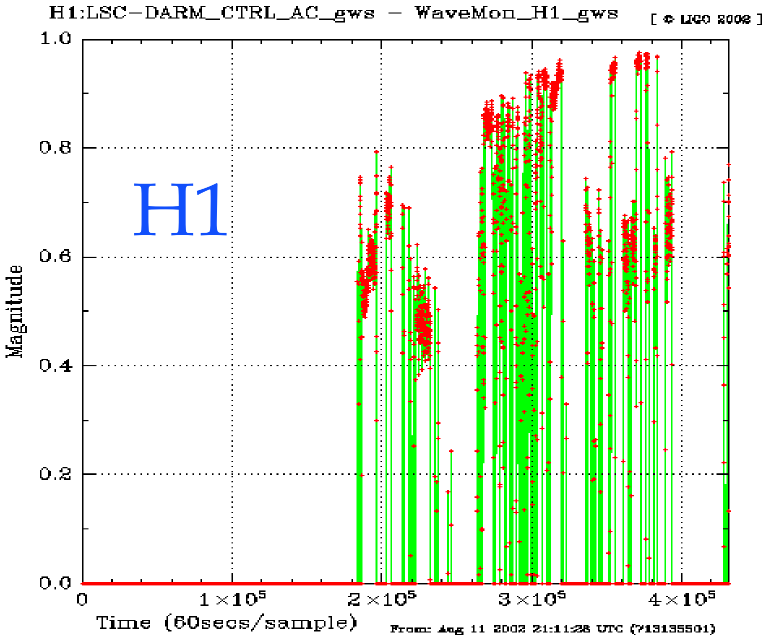
Available T-F Space

- **Glitches and GW bursts compete for the same fraction of the T-F space, defined by the percentile threshold *P*.**
  - **acceptance *A* – fraction of *PV* available for GW bursts**

$$A = 1 - \frac{1}{P \cdot V} \sum V_i$$

V_i – cluster volume
V - total T-F volume





- **excellent measure of the detector performance in terms of glitches**

False Alarm

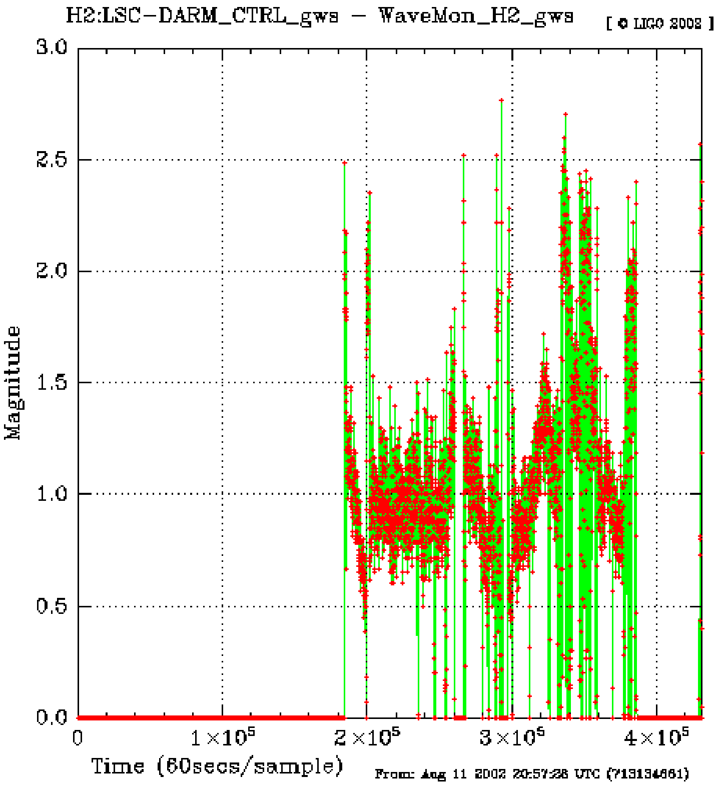
- due to random clustering of T-F pixels
  - since the percentile transform is used → don’t care about noise PDF

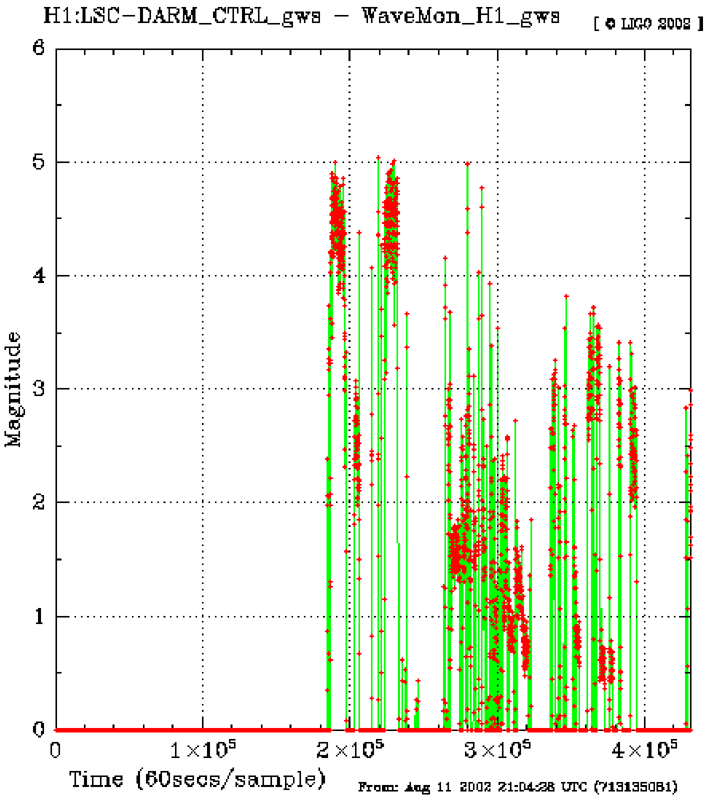
    calculate false alarm for coincidence of two random channels

    (no correlation between pixels in both channels)
  - WaveMon false alarm is set to be ~5mHz (16-1024Hz band)
- due to random coincidence of glitches in master channel with random slave channel
  - takes into account correlation between pixels in master channel
  - depends on glitch rate in master channel
  - comparable with *random false alarm* at low (<1Hz) glitch rate
- both types of false alarm are constantly monitored by WaveMon
  - false alarm triggers are added to output trigger table

VETO Triggers Rate

- For significant clusters the WaveMon puts an entry in the trigger table
- The rate of triggers is monitored





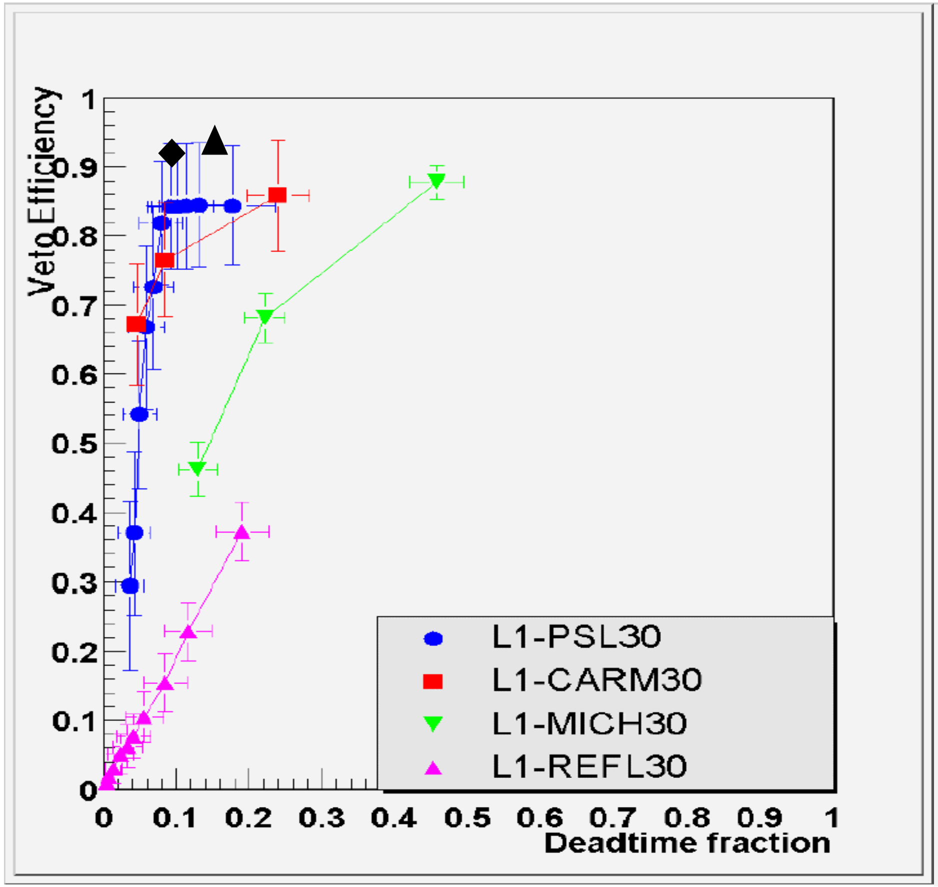
efficiency vs. deadtime

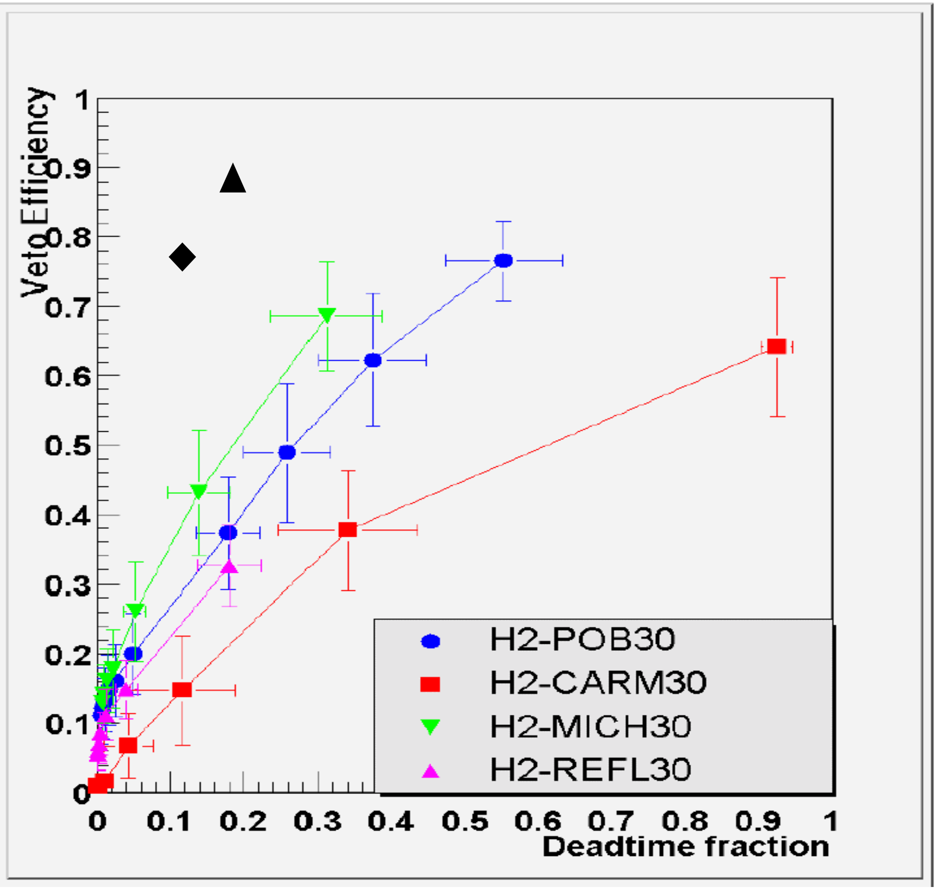
Use WaveMon veto to suppress Tfcluster events

▲ Combined veto: MICH + POB+CARM+MCF+REFL, gap=0

◆ Veto: H2:LSC-MICH_CTRL, gap=0

VETO efficiency may be limited by Tfcluster false alarm rate.

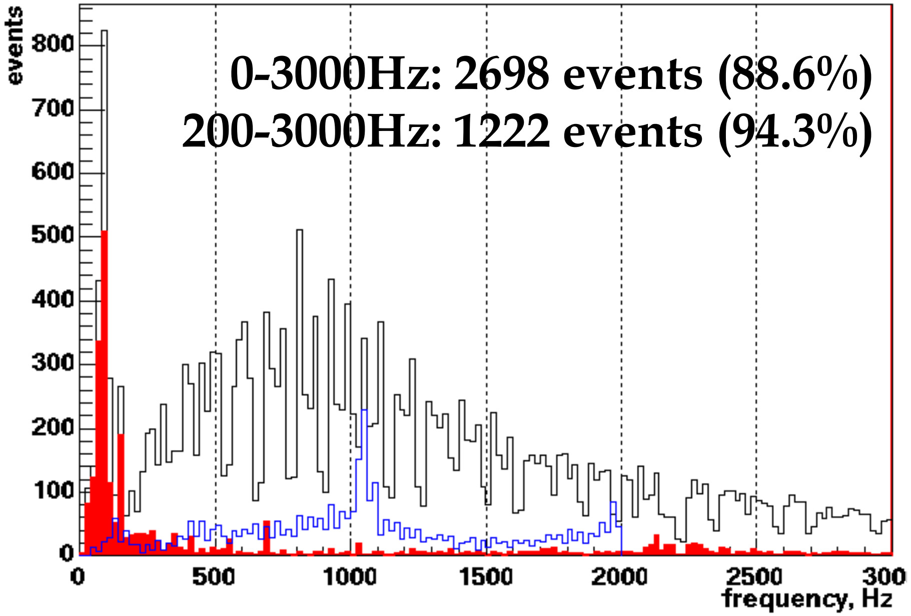




E7 playground Data

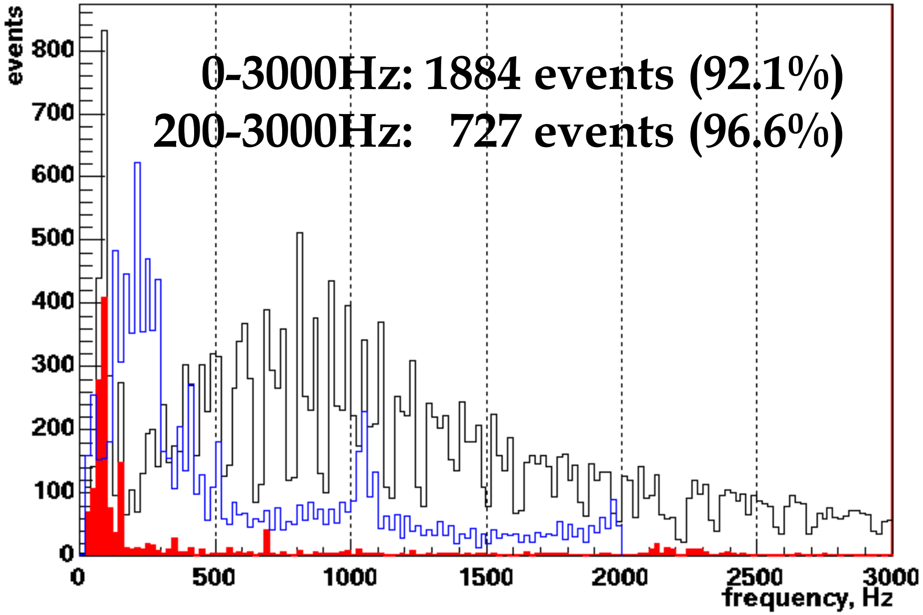
- black – TFCLUSTER
- blue - WaveMon (only 16-2046Hz frequency band is used)
- red -TFCLUSTR x VETO

L1:LSC-AS_Q x L1:PSL-ISS_ACTM_F



0-3000Hz: 2698 events (88.6%)
200-3000Hz: 1222 events (94.3%)

L1:LSC-AS_Q x ALL



0-3000Hz: 1884 events (92.1%)
200-3000Hz: 727 events (96.6%)

Conclusion

- WaveMon is a tool to monitor the detector performance in terms of glitches.
  - Can help identify the source of glitches in the GW channel by looking at frequency content and correlation with auxiliary channels
  - Estimates dead fraction of T-F space due to glitches
  - Has predictable false alarm rate
- Produce veto triggers
- Fast: can monitor up to 20 16kHz channels in real time