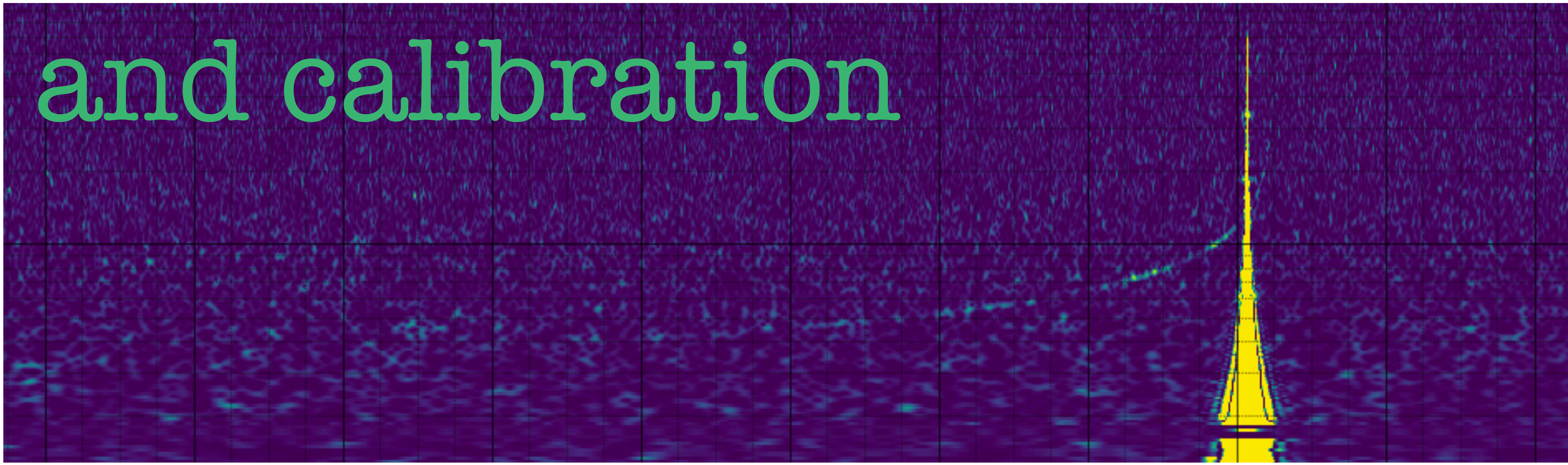
GW detector characterization and calibration



Made with GWpy: gwpy.github.io

Dr. Jess McIver

ISAPP summer school
June 10, 2021
LIGO DCC

LIGO VIRGO LSC THE UNIVERSITY OF BRITISH COLUMBIA

Outline

GW detector characterization

The challenge of non-stationary GW detector data

- Search sensitivity and PE accuracy
- The “worst offender” glitches
- Uptime

Techniques for identification and characterization

- Data visualization: into the frequency domain!
- GWOSC tools: GWpy and Omegascans

Mitigation: Data quality products and glitch subtraction

An overview of GW detector calibration

Outline

GW detector characterization

The challenge of non-stationary GW detector data

- Search sensitivity and PE accuracy
- The “worst offender” glitches
- Uptime

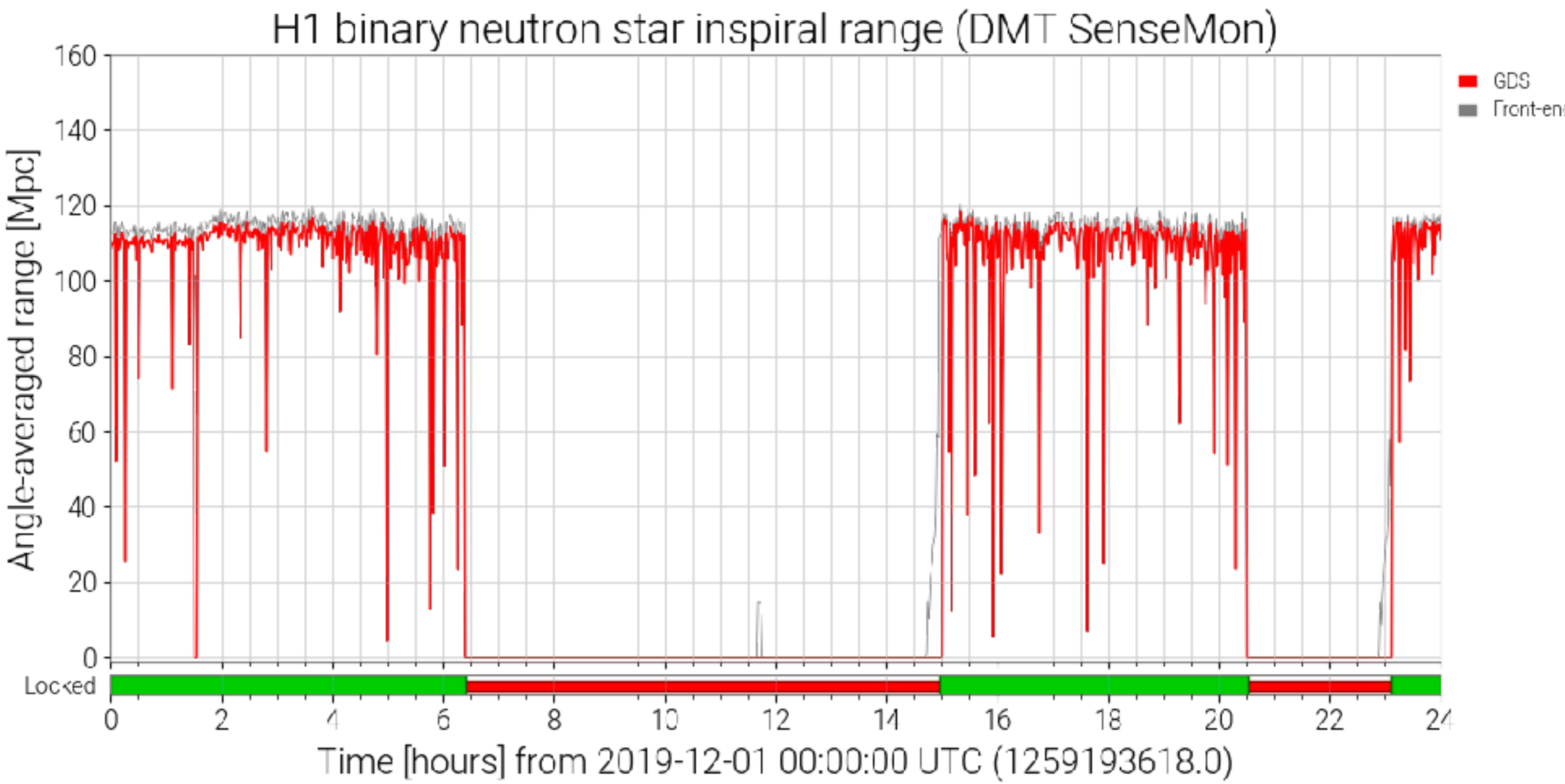
Techniques for identification and characterization

- Data visualization: into the frequency domain!
- GWOSC tools: GWpy and Omegascans

Mitigation: Data quality products and glitch subtraction

An overview of GW detector calibration

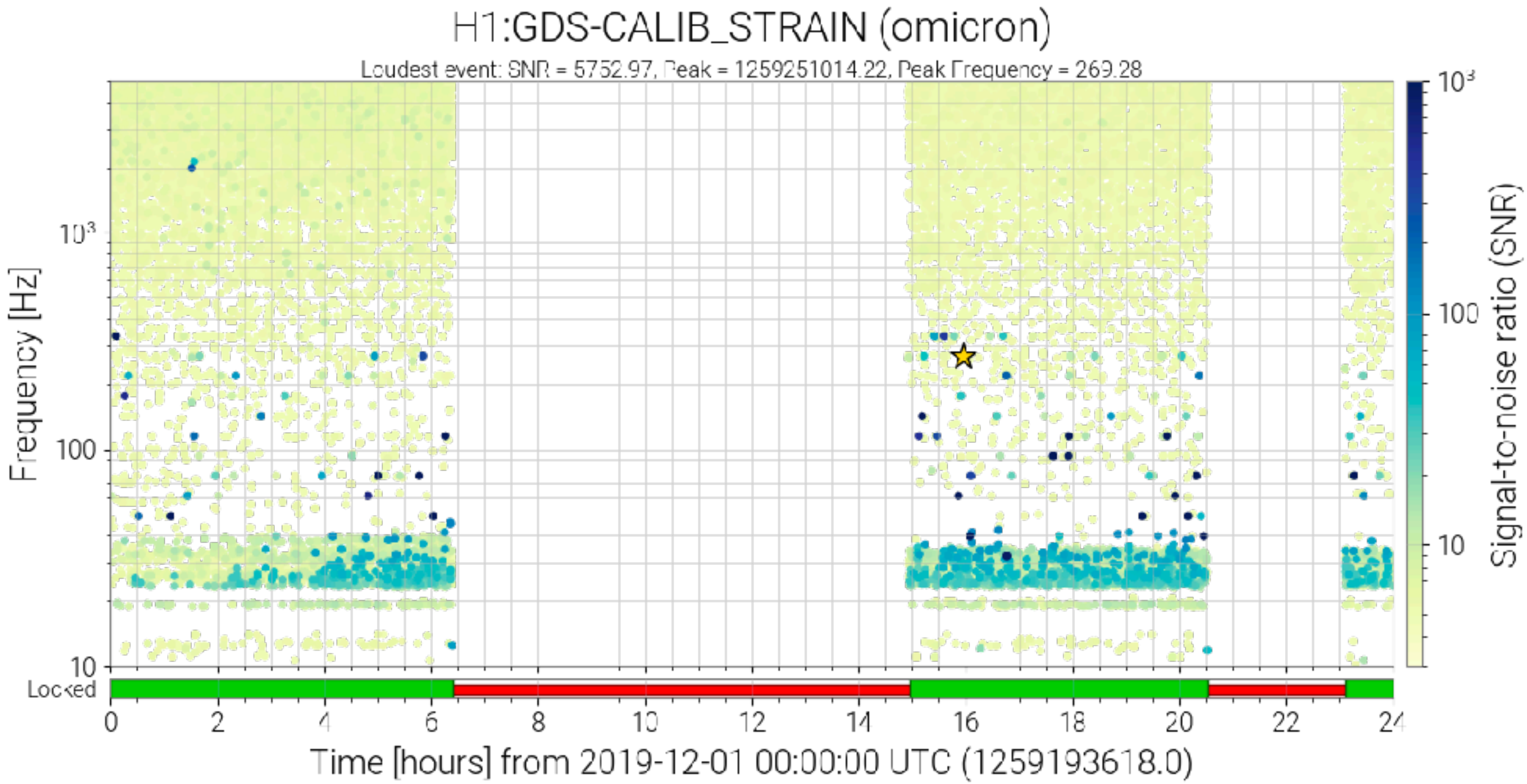
What do we mean by "non-stationary"?



BNS inspiral range = given the PSD of the (average) noise, the distance at which we'd detect a 1.4-1.4 M_sol BNS with an SNR of 8, averaged over orientation and sky position angles

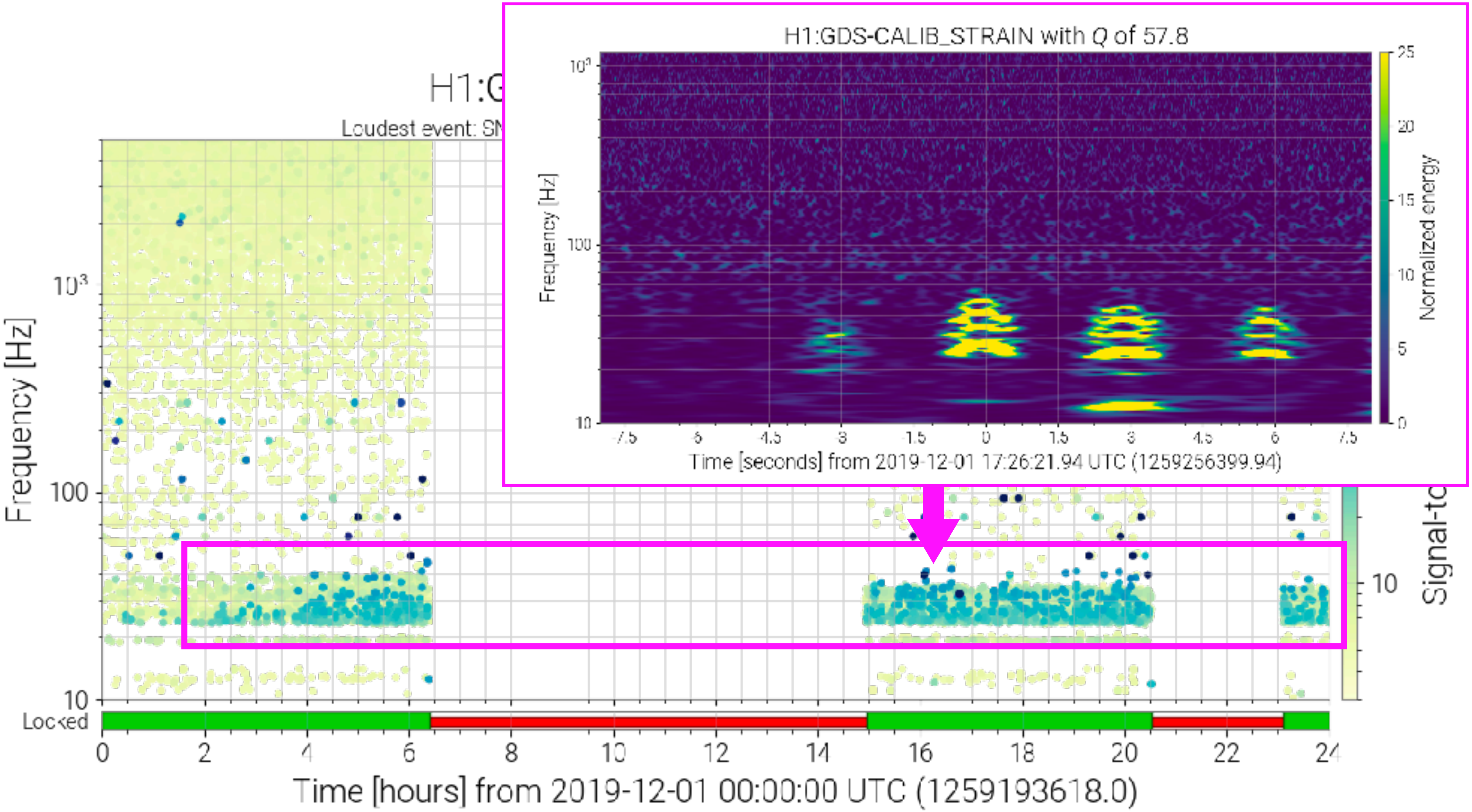
The LIGO summary pages

What do we mean by "non-stationary"?



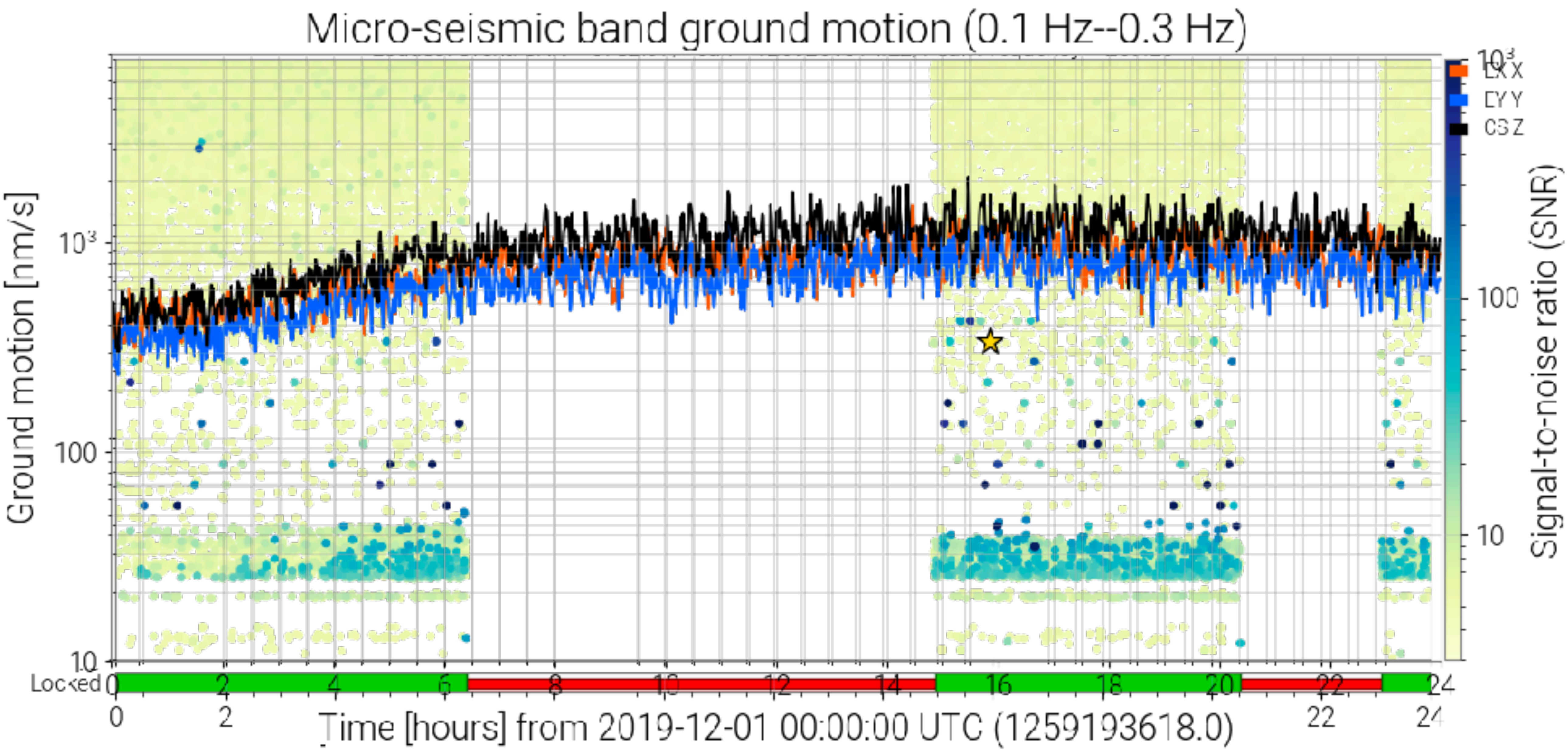
The LIGO summary pages

What do we mean by "non-stationary"?



The LIGO summary pages

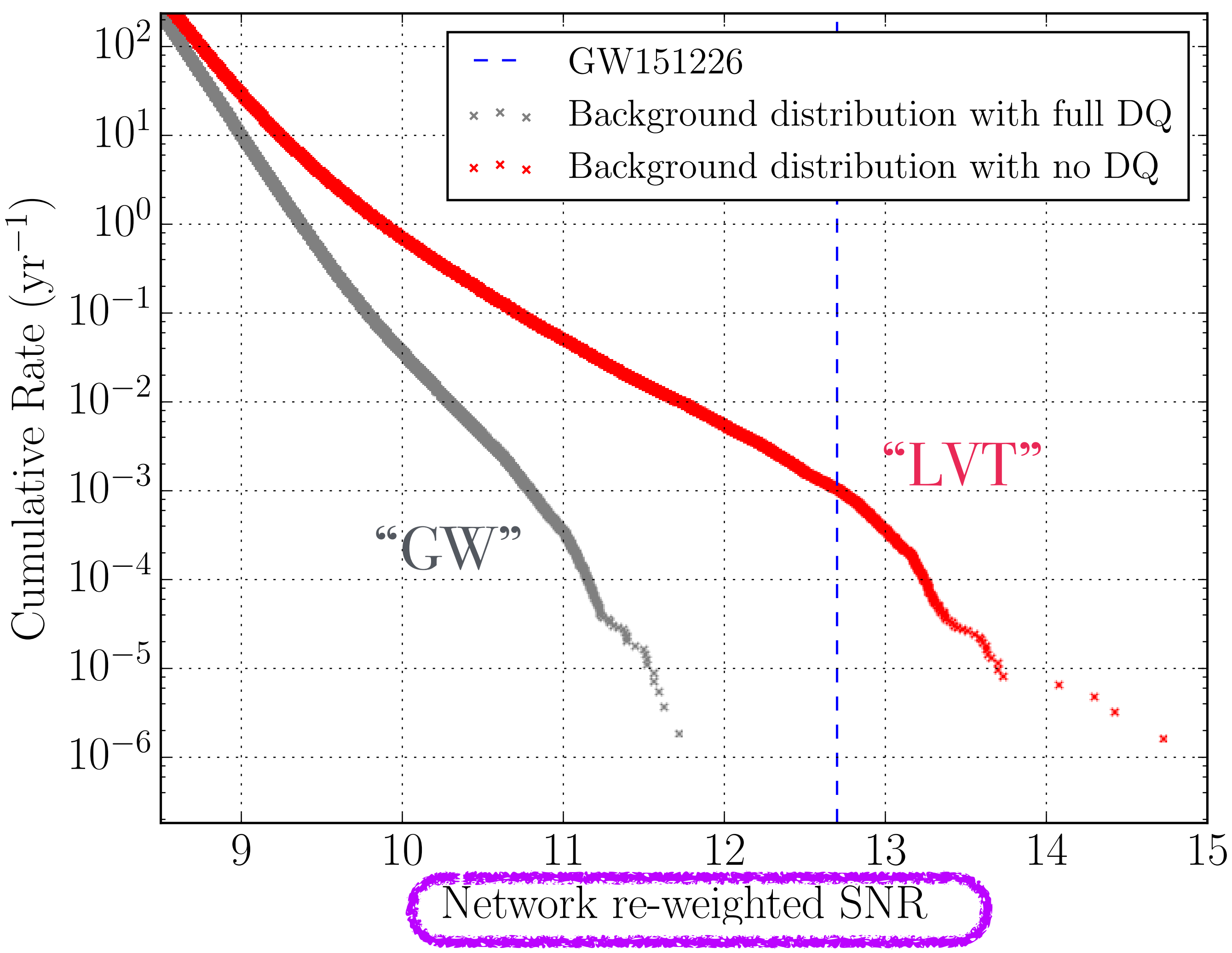
Key question: what drives non-stationarity?



The LIGO summary pages

Glitches adversely impact search sensitivity

GW151226 analysis



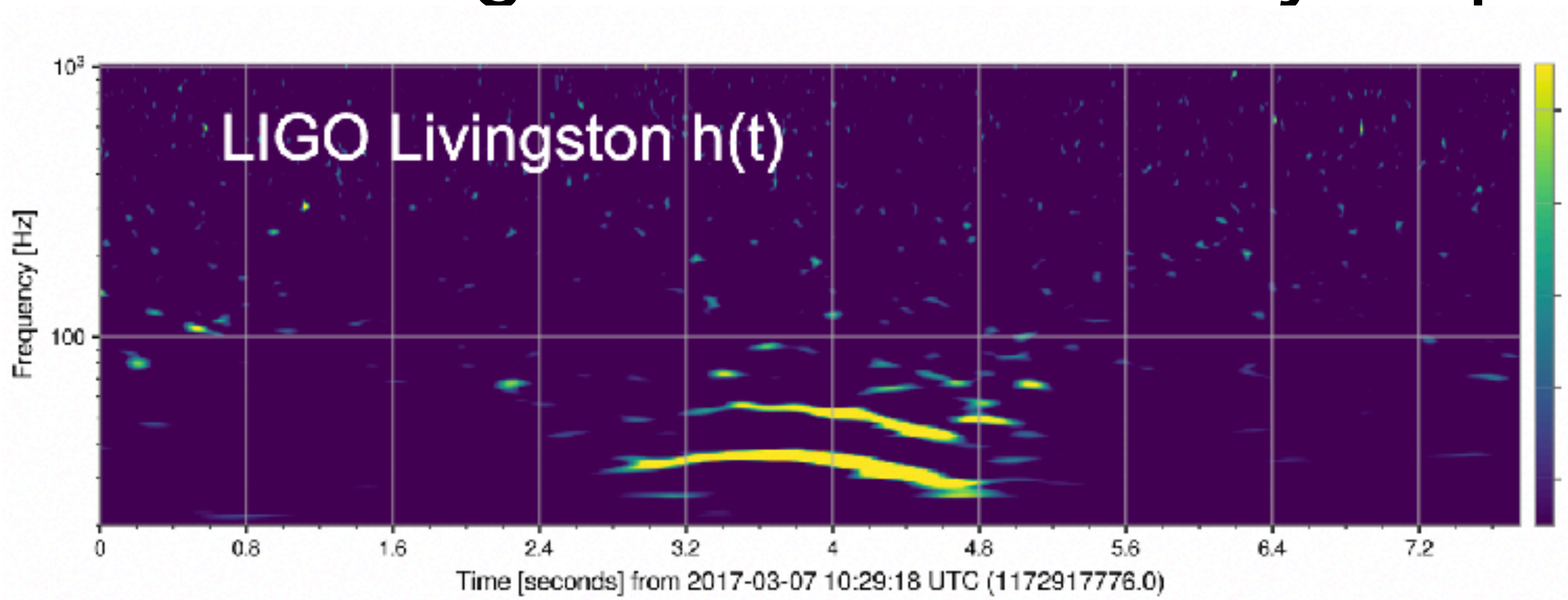
The false alarm rate of GW151226 **improves by a factor of 567**, from 1 in 320 years to 1 in 183000 years, **with interferometer data quality information!**

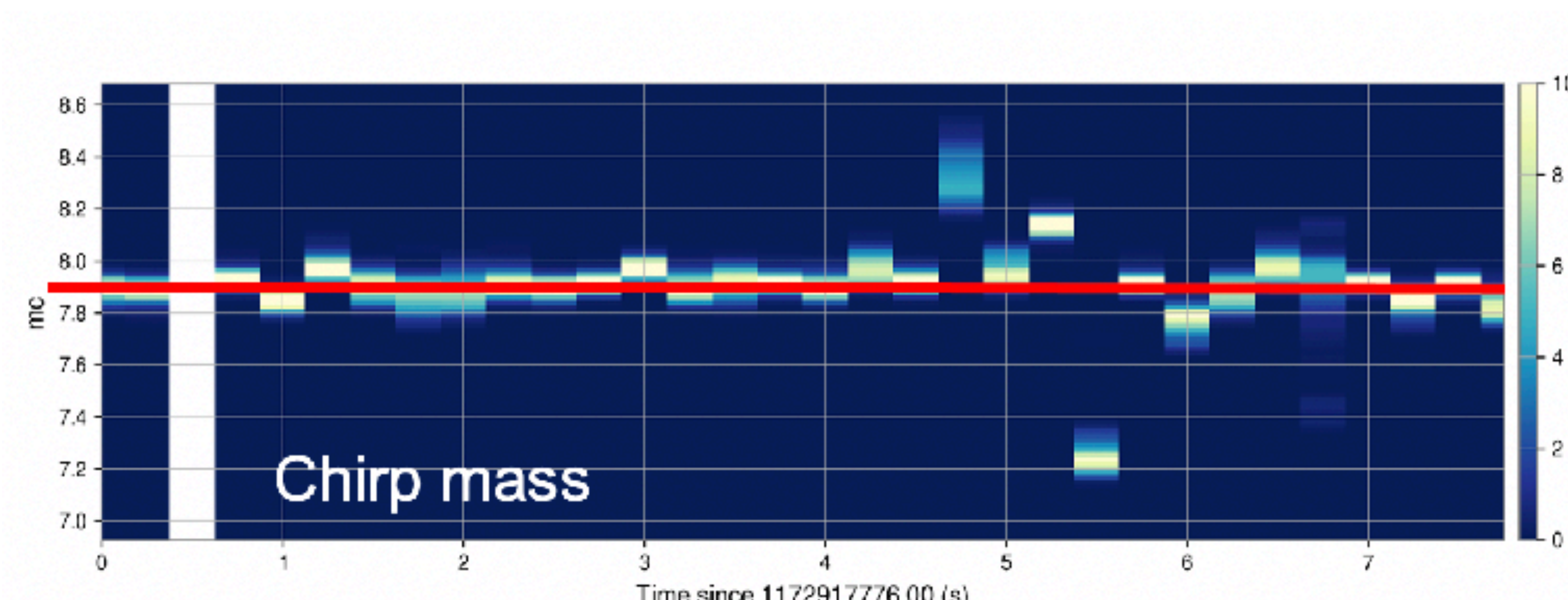
See Eric's talk for more on calculating SNR, event significance, and analysis-specific re-weighting methods like the chi-squared veto.

Using the *network SNR*, the quadrature sum of SNR across detectors, requires coincident triggers and greatly reduces false alarms.

LIGO-Virgo collaboration (2017) - arXiv 1710.02185

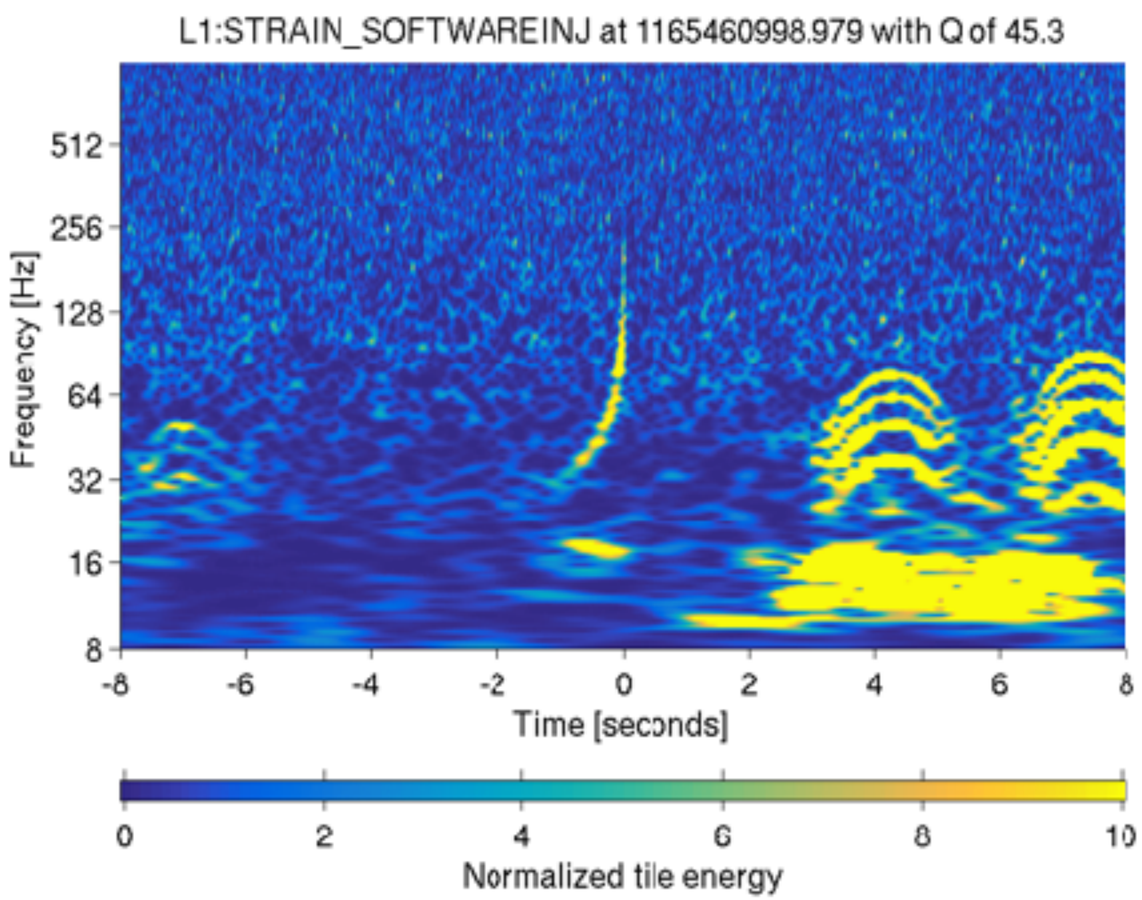
Glitches near signals can adversely impact PE

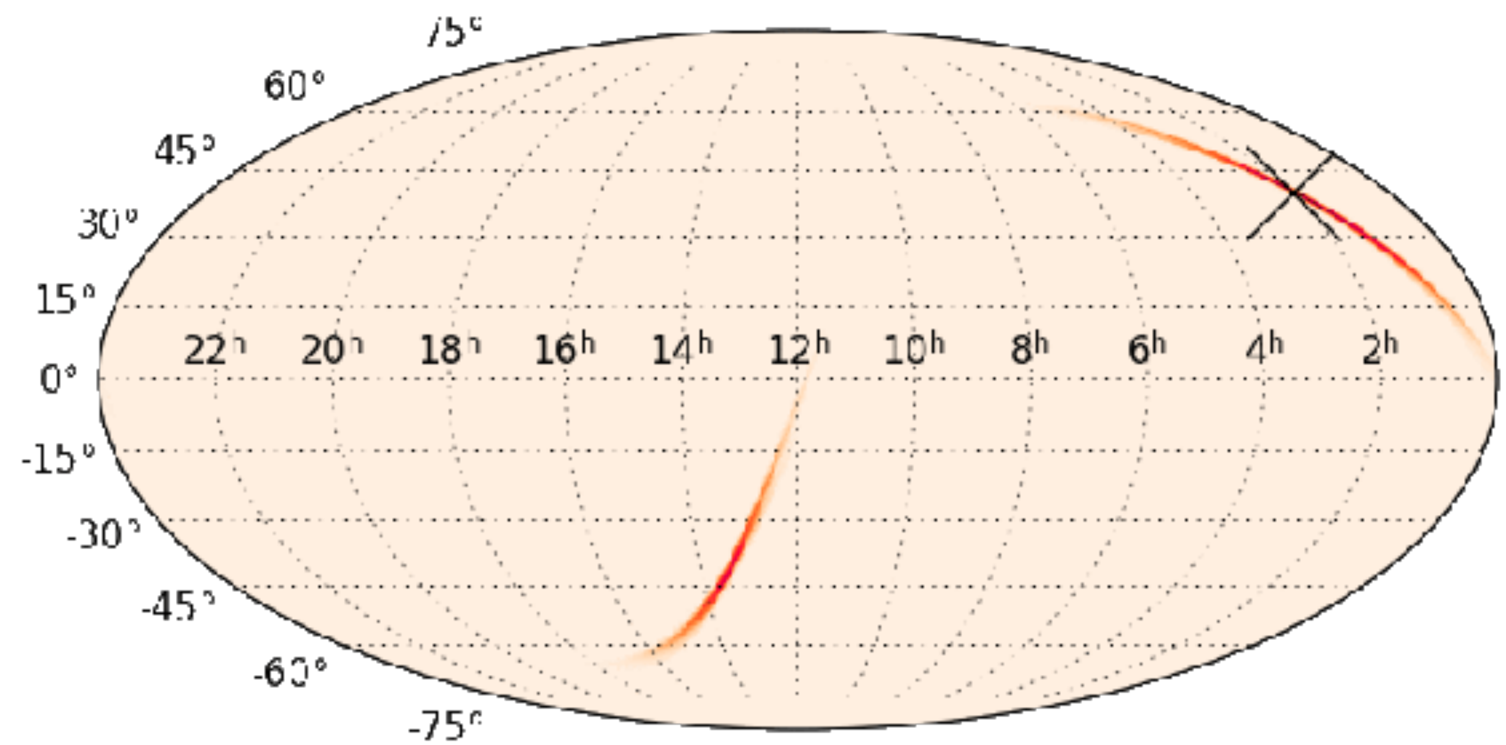




Parameter estimation produced with the lalinference pipeline: arXiv 1409.7215

Example: Impact of light scattering on sky localization





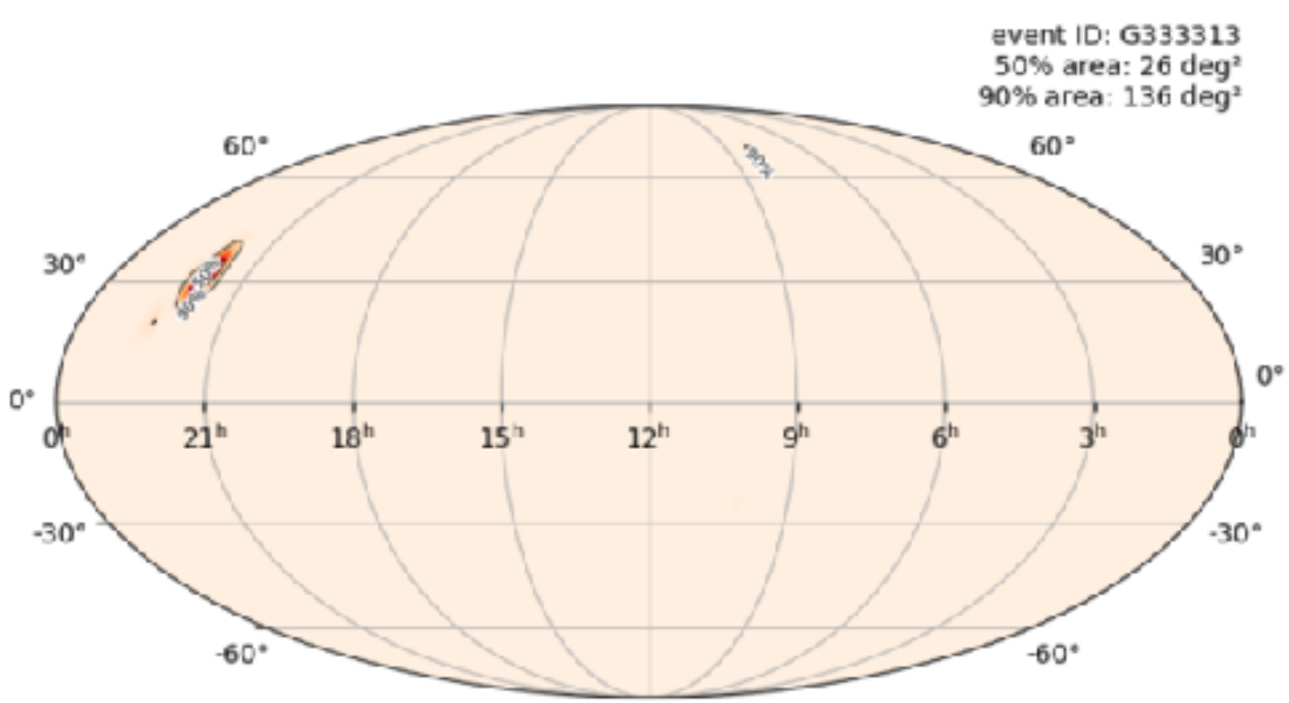
McIver et al. In prep.

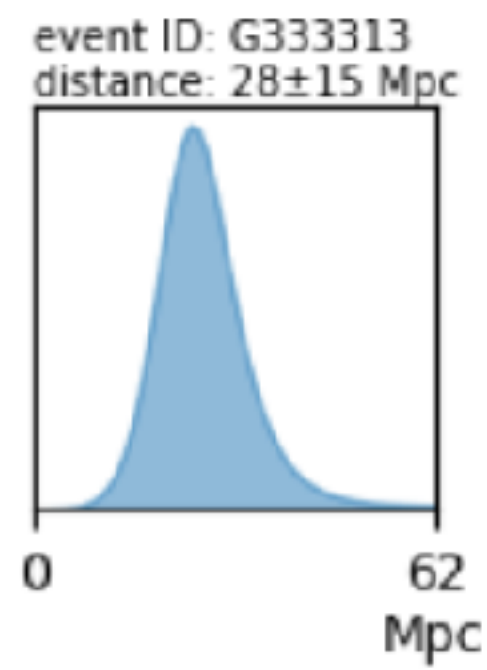
Parameter estimation produced with the lalinference pipeline: arXiv 1409.7215

LIGO-Virgo OPA retractions: S190518bb

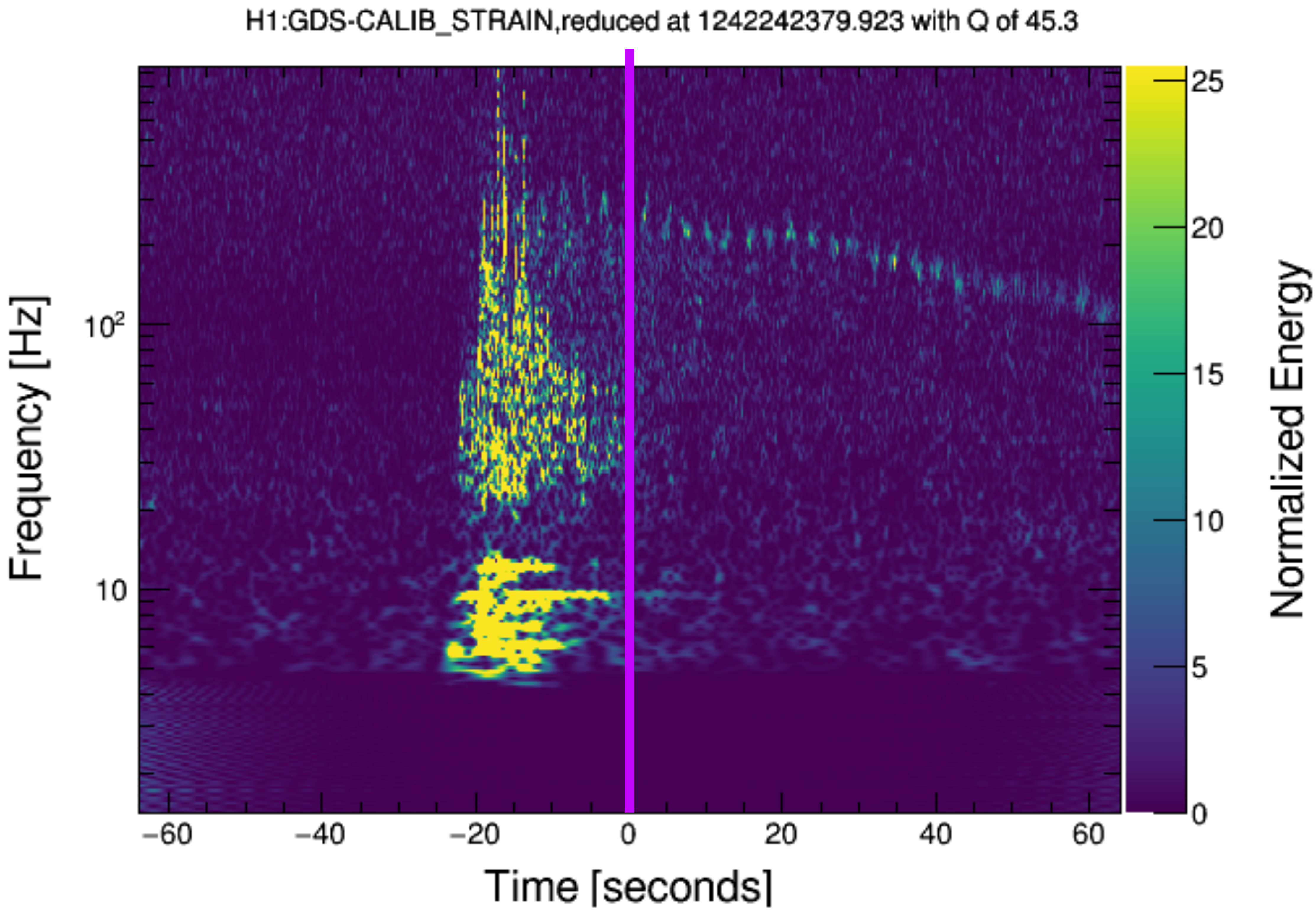
Automatic Preliminary Notice sent ~6 minutes after the event:

FAR: 1.004e-08 [Hz] (one per ~3 years)
PROB_NS: 1.00 [range is 0.0-1.0]
PROB_REMNANT: 1.00 [range is 0.0-1.0]
PROB_BNS: 0.75 [range is 0.0-1.0]
PROB_TERRES: 0.24 [range is 0.0-1.0]

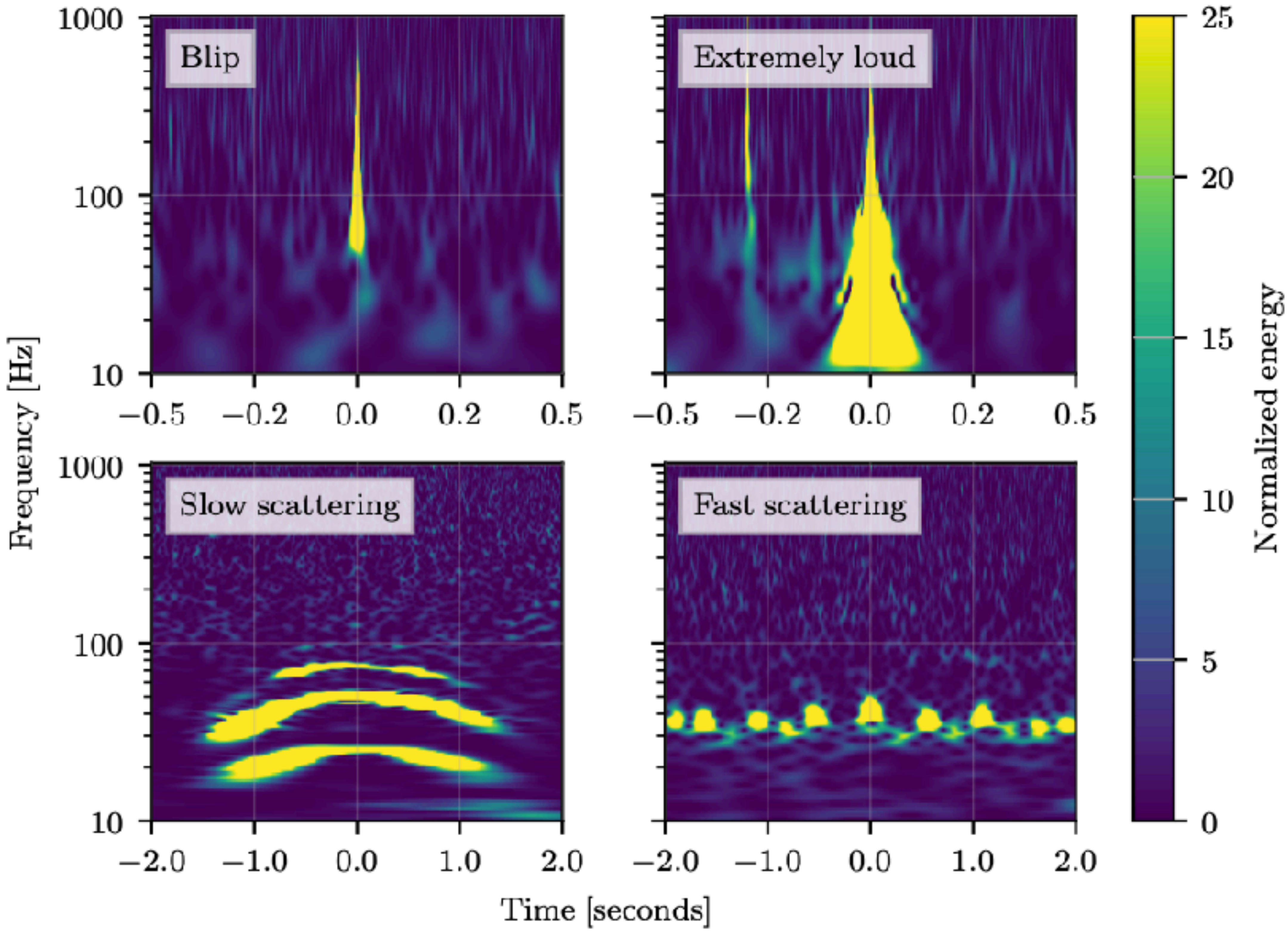




S190518bb: what's in the data?

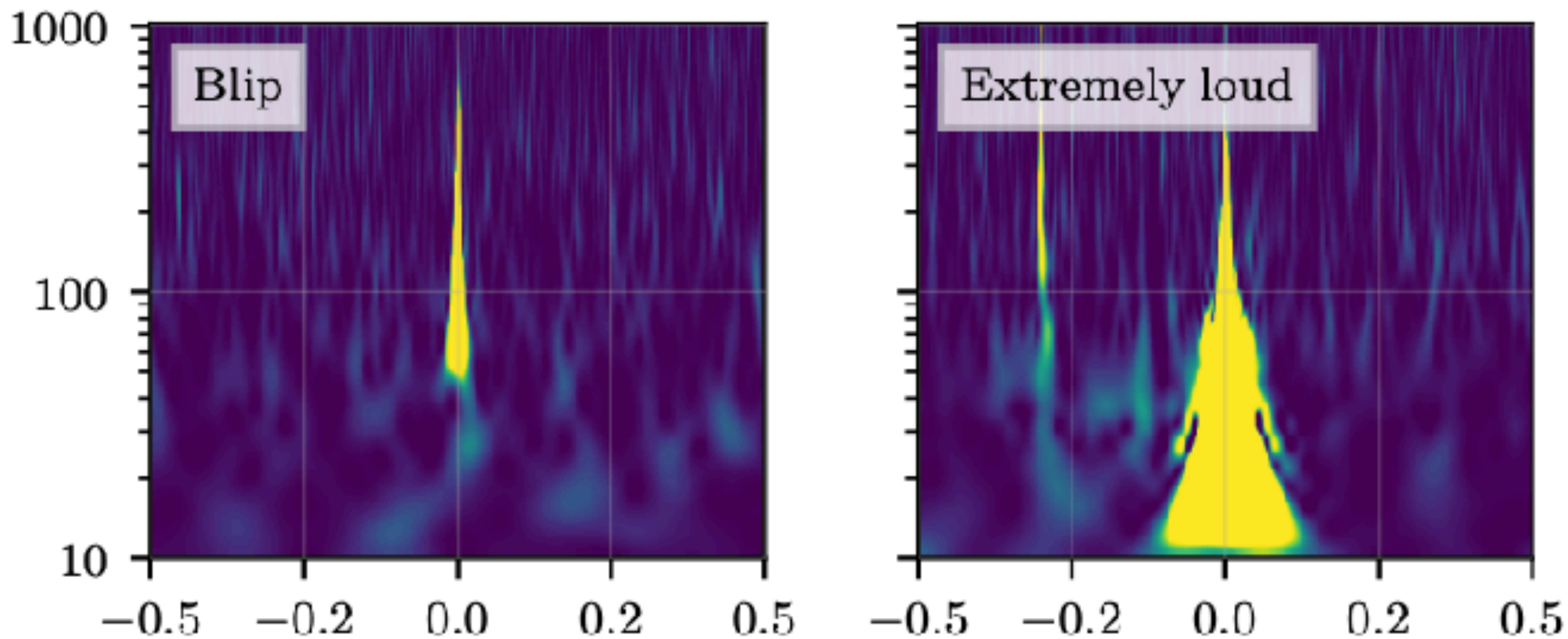


"Worst offender" glitches



Davis et al. CQG 2021

"Worst offender" glitches



Blips

- Few known witnesses
- Shared time-frequency morphology with high mass CBC signals

Extremely loud

- Much more common in O3 than in O2 for LIGO detectors
- Few clear witnesses
- Pollute PSD estimation

Davis et al. CQG 2021

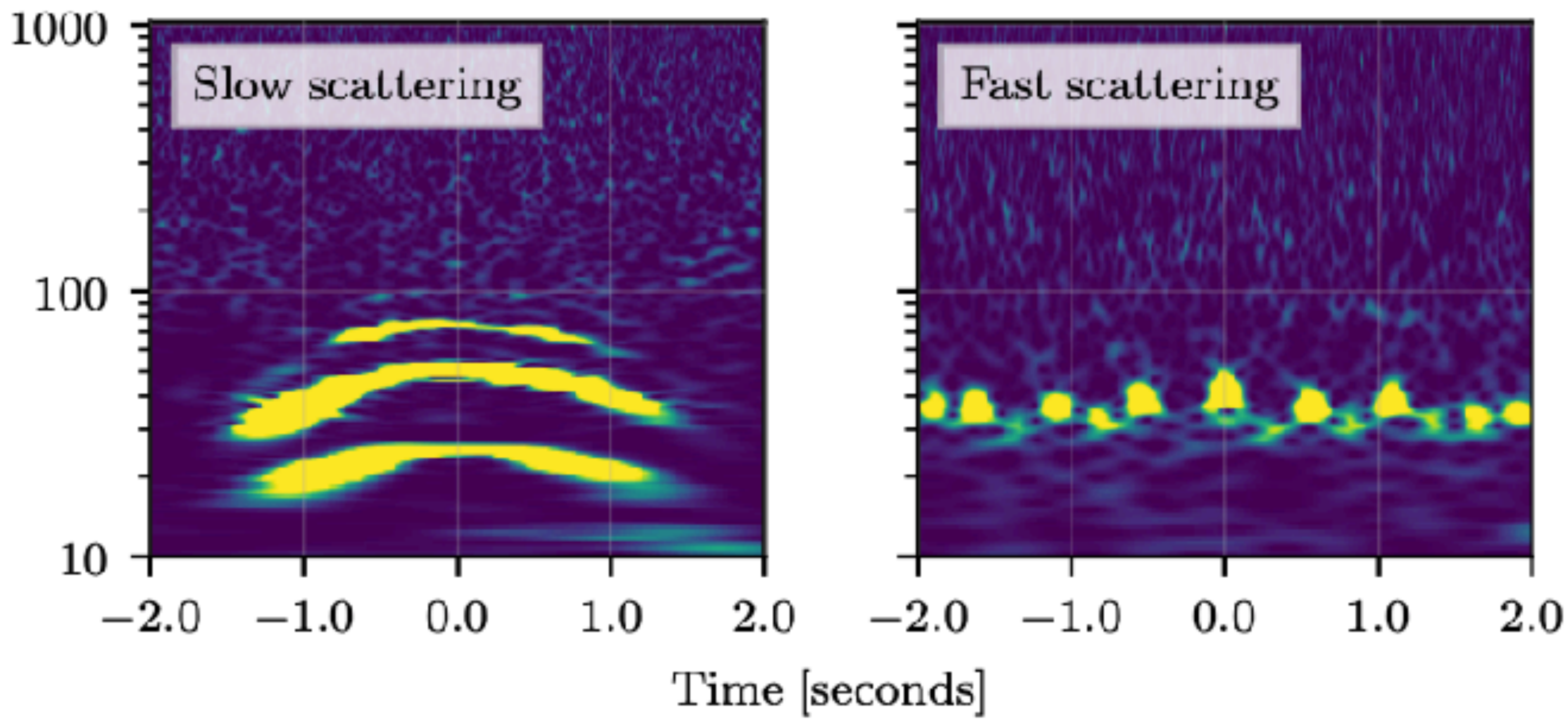
"Worst offender" glitches

Slow scattering

- Well understood witnesses and coupling
- Still difficult/impractical to veto because they are at times so frequent

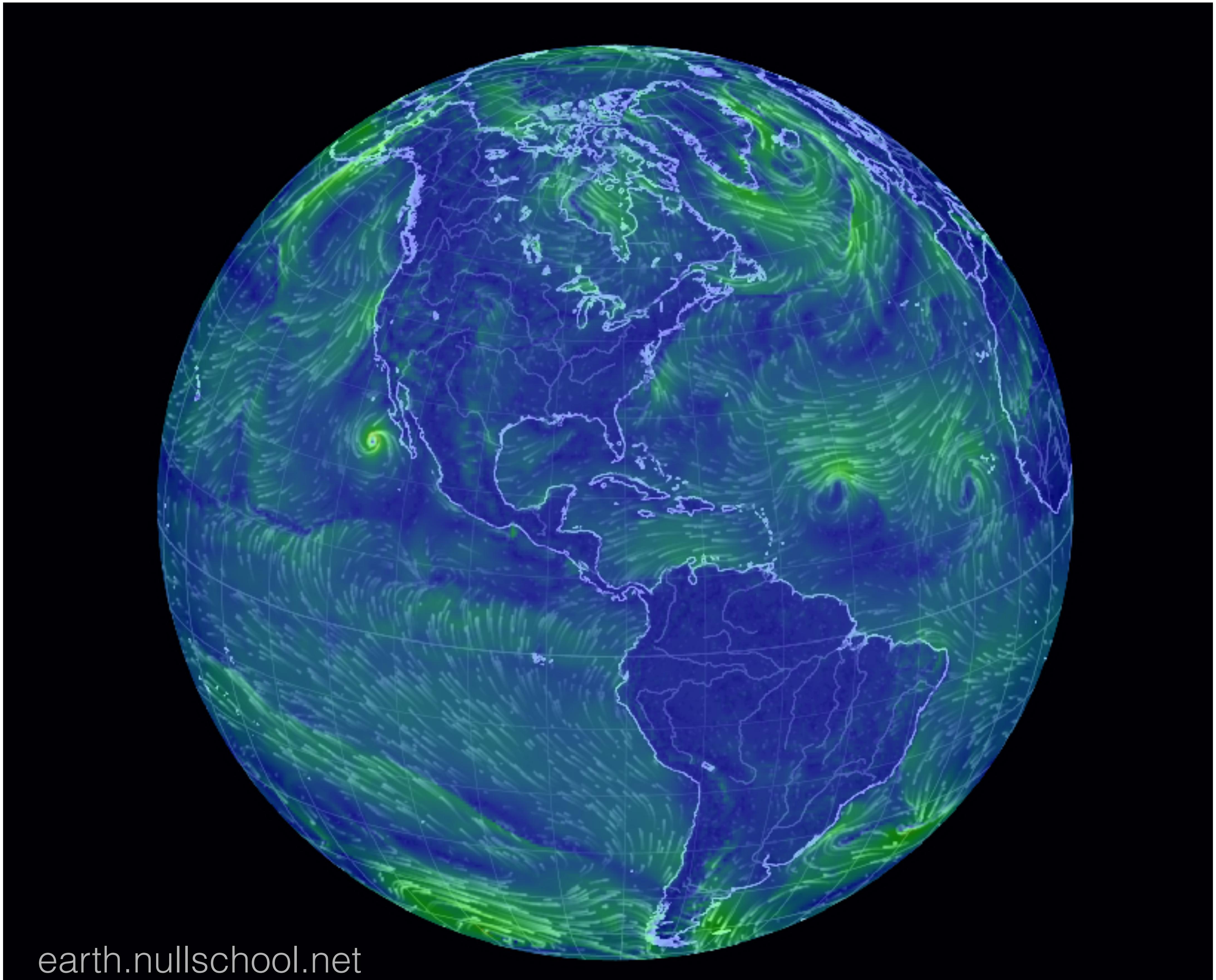
Fast scattering

- Modulated fast scattering
- Troublesome for lower mass CBC templates
- Most comment LIGO-Livingston: thought to be understood for O3

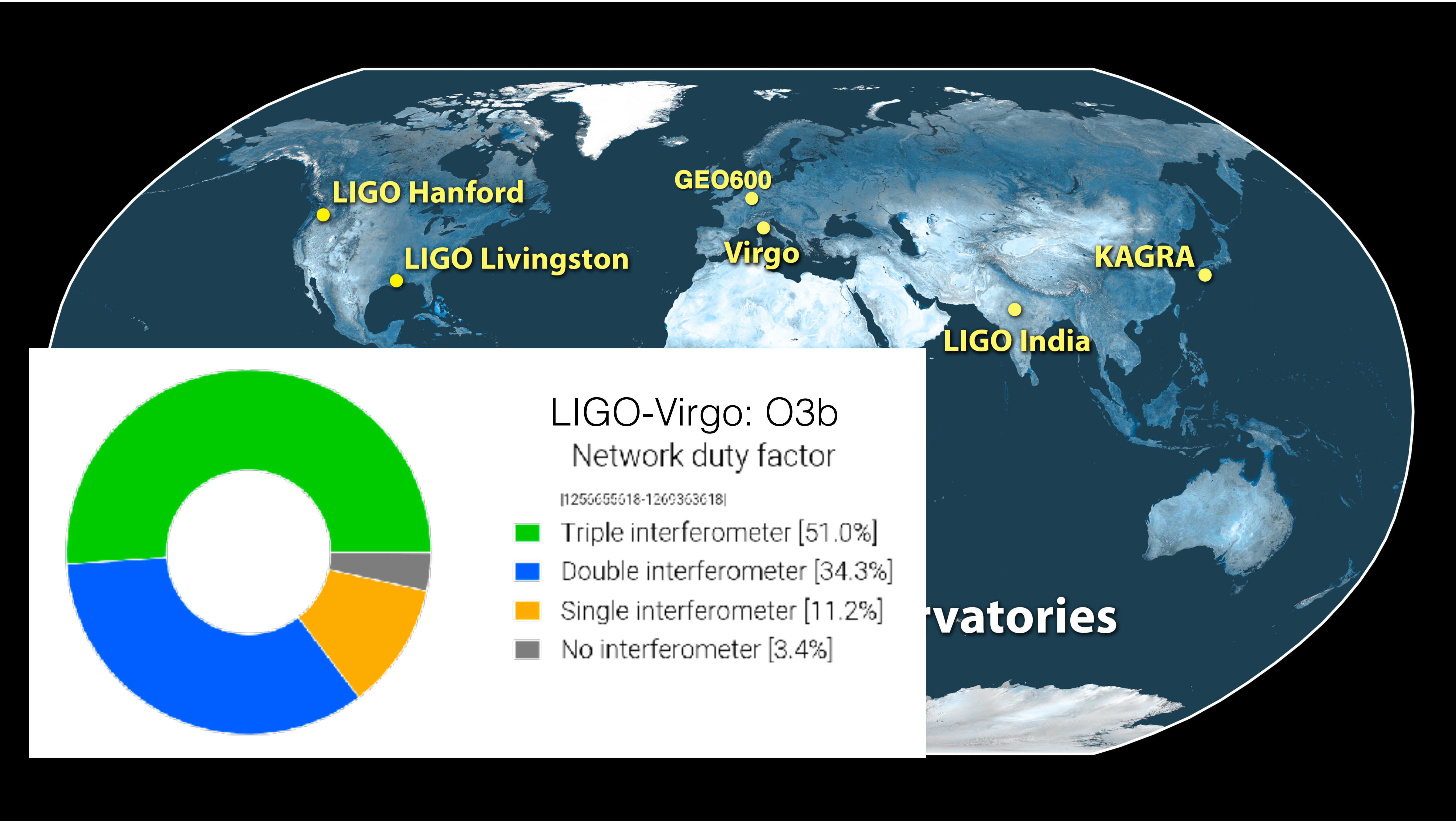


Davis et al. CQG 2021

GW detector duty cycle



Uptime and the global network of detectors



LIGO/Caltech

Outline

GW detector characterization

The challenge of non-stationary GW detector data

- Uptime
- Search sensitivity and PE accuracy
- The “worst offender” glitches

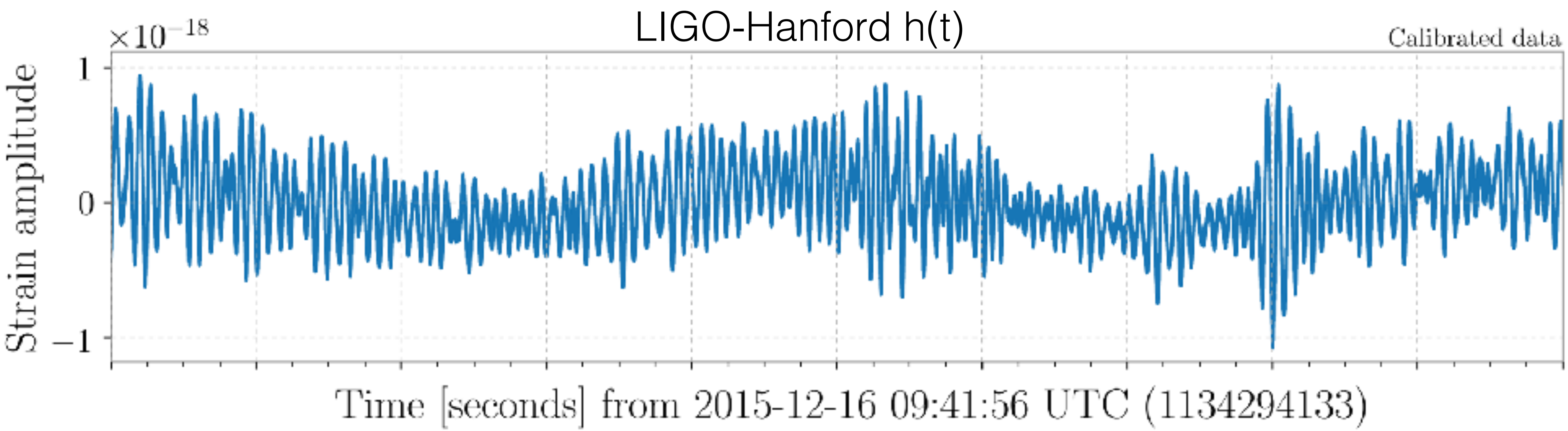
Techniques for identification and characterization

- Data visualization: into the frequency domain!
- GWOSC tools: GWpy and Omegascans

Mitigation: Data quality products and glitch subtraction

An overview of GW detector calibration

What does GW detector data look like?



Made with GWpy!

What's in a GW data file?

meta: Meta-data for the file. This is **basic information** such as the GPS times covered, which instrument, etc.

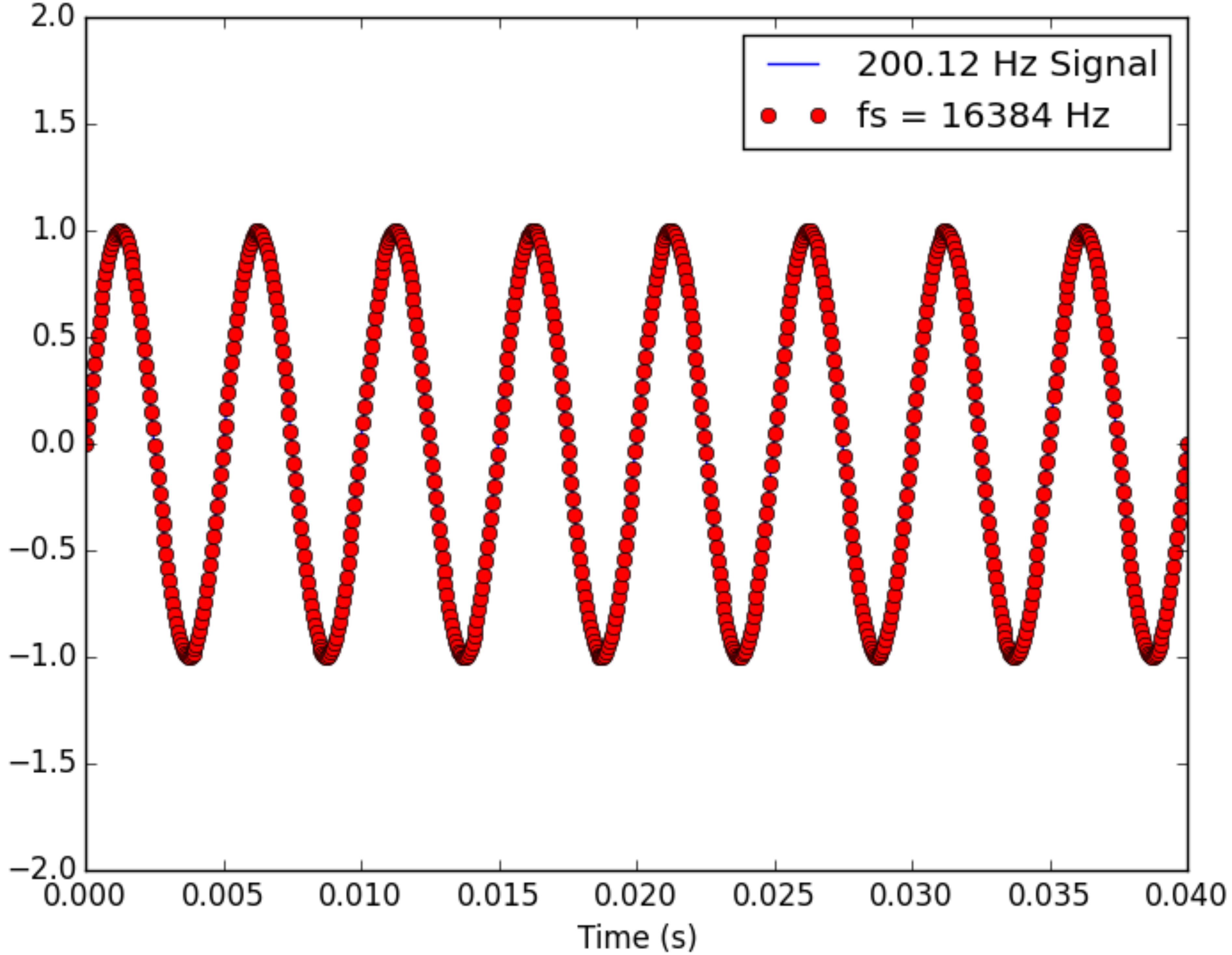
strain: Strain data from the interferometer. This is "the data", the **main measurement of spacetime strain** recorded by the LIGO detectors.

quality: A 1 Hz time series describing the **data quality** for each second of data.

h(t) **sampling rate** for LIGO detectors: 16384 Hz
Open data: 4096 Hz and 16384 Hz

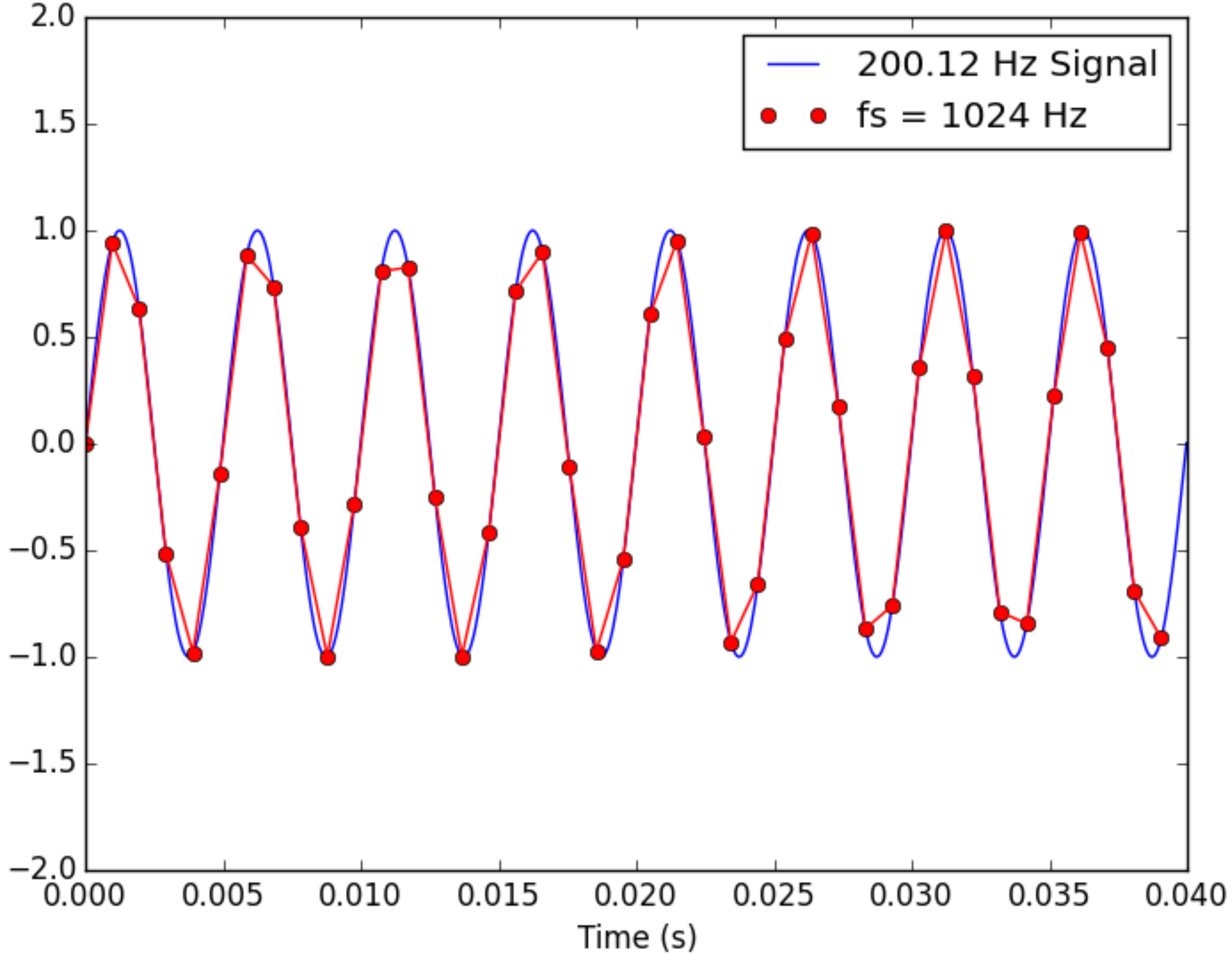
Why do we care about sampling rate, f_S ?

Discrete Time Samples



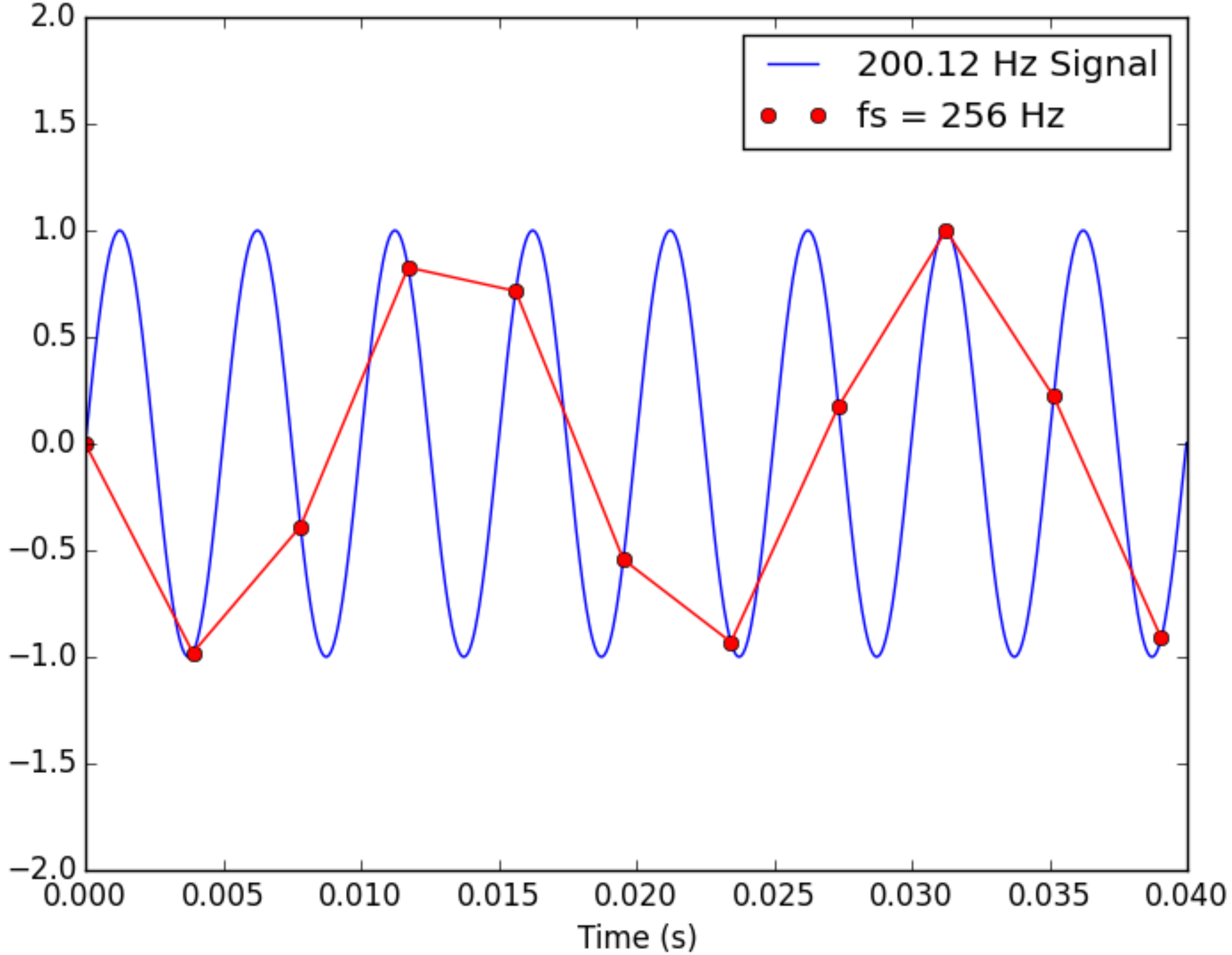
Slide from J. Kanner

Discrete Time Samples



Slide from J. Kanner

Discrete Time Samples



Slide from J. Kanner

Nyquist Frequency

- Nyquist Frequency = $\frac{f_s}{2}$
- Data can only accurately represent frequency content below the *Nyquist frequency*
- Higher frequency signals will be lost or “aliased” to lower frequencies

Introduction to GWpy

A python package for gravitational-wave astrophysics

https://gwpy.github.io

Heavily dependent on numpy, scipy, astropy, matplotlib

Provides intuitive object-orientated methods to access GW detector data, process, and visualize them

Not specific to GW data other than data access routines

Slide from D. Macleod

GWpy Quickstart

Import the class that represents the data you want to study

```
>>> from gwpy.timeseries import TimeSeries
```

Fetch some open data from the OSC

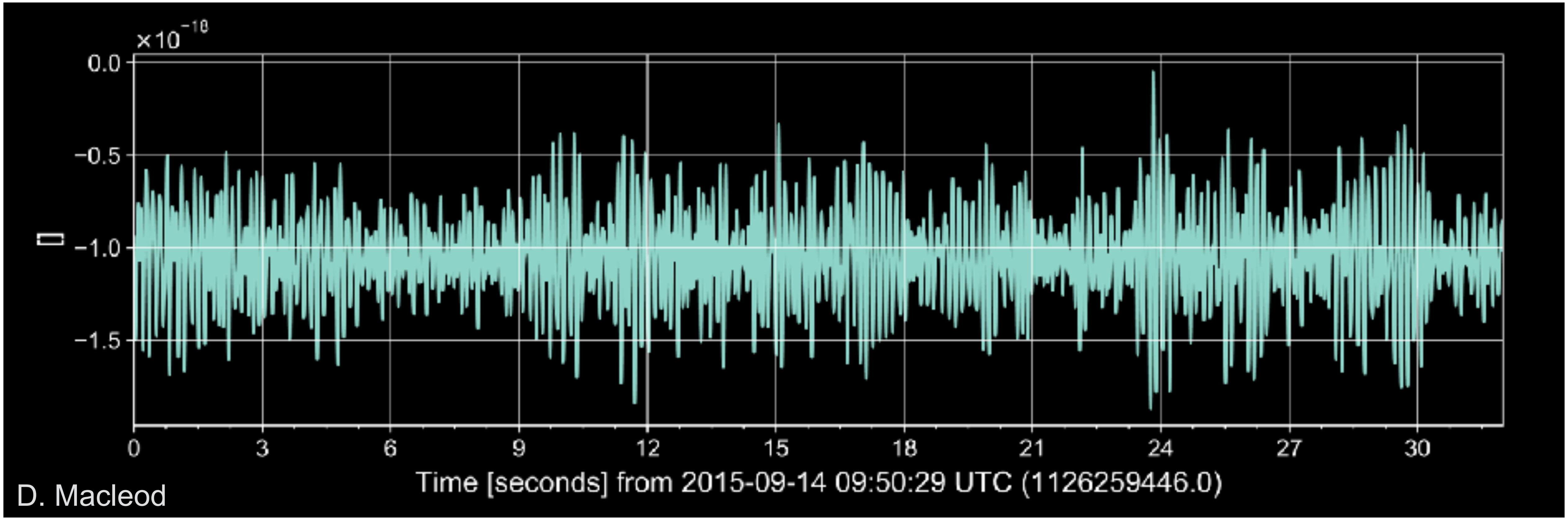
```
>>> data = TimeSeries.fetch_open_data('L1', 'Sep 14 2015 09:50:29', 'Sep 14 2015 09:51:01')
```

Make a plot

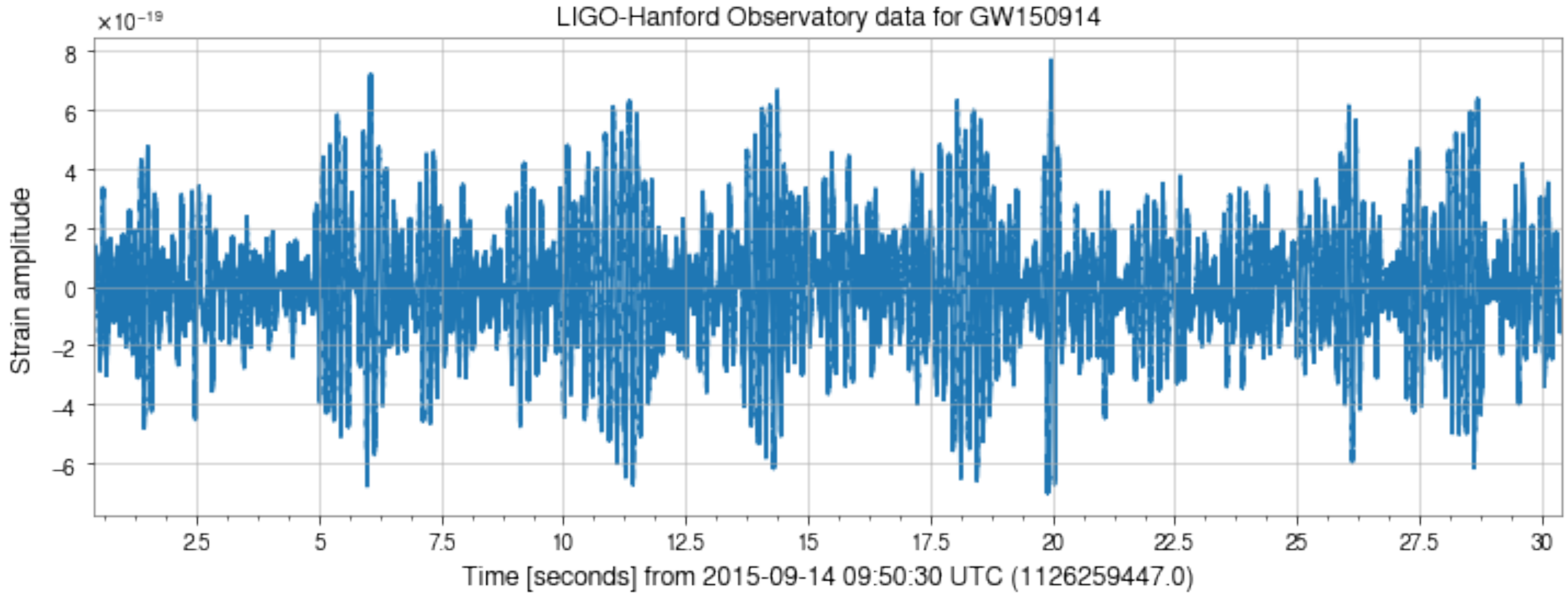
```
>>> plot = data.plot()
```

Display the plot

```
>>> plot.show()
```



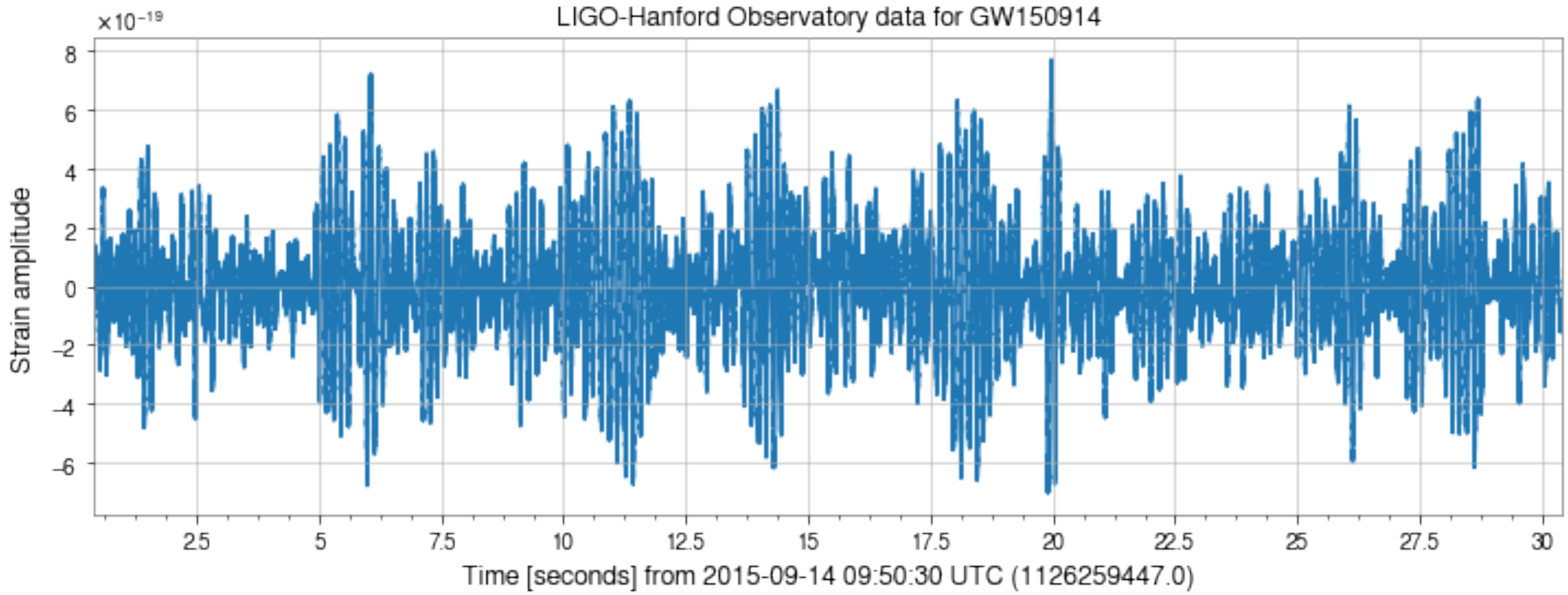
Time domain



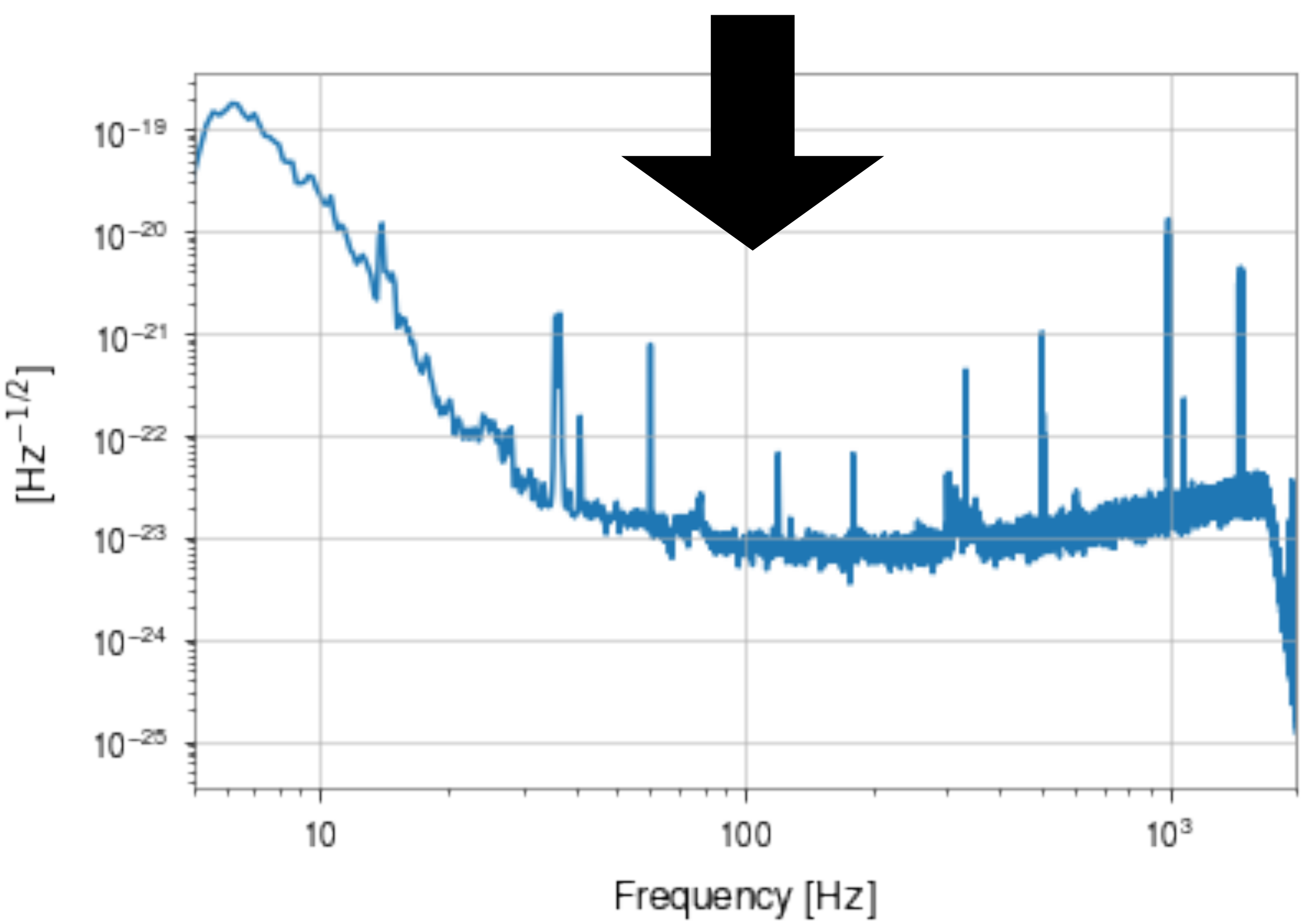
LIGO-Hanford Observatory data for GW150914
×10⁻¹⁹
8
6
4
2
0
-2
-4
-6
Strain amplitude
2.5
5
7.5
10
12.5
15
17.5
20
22.5
25
27.5
30
Time [seconds] from 2015-09-14 09:50:30 UTC (1126259447.0)

Time domain ➔ Frequency domain

Time series (strain vs time)



Amplitude spectral density ($Hz^{-\frac{1}{2}}$ vs. frequency)



The Fourier Series

Any function can be represented as a sum of sines and cosines (with some coefficients that can also be functions).

$$f(x) = \sum_{n=0}^{\infty} A_n \cos\left(\frac{n\pi x}{L}\right) + \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right)$$

The Fourier Transform

When we transform our function or time (or space) into the "frequency domain", we are **projecting f(x) onto an orthogonal basis of sines and cosines**.

Fourier transform

$$\tilde{x}(f) = \int_{-\infty}^{\infty} dt\, x(t) e^{-i2\pi f t}$$

The Fourier Transform

When we transform our function or time (or space) into the "frequency domain", we are **projecting f(x) onto an orthogonal basis of sines and cosines**.

Fourier transform

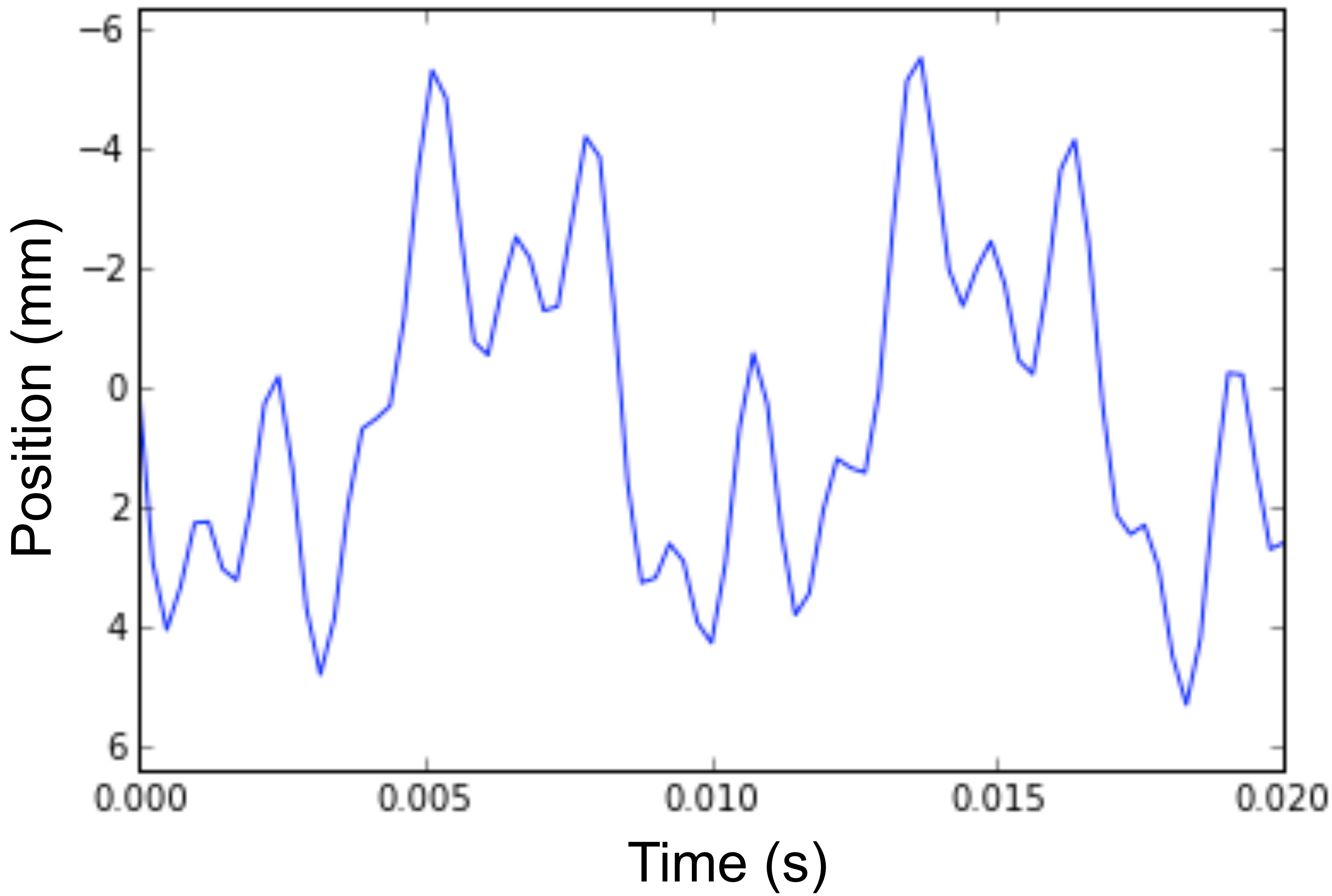
$$\tilde{x}(f) = \int_{-\infty}^{\infty} dt\, x(t) e^{-i2\pi f t}$$

Inverse Fourier transform

$$x(t) = \int_{-\infty}^{\infty} df\, \tilde{x}(f) e^{i2\pi f t}$$

Another way to think about it: when we take a Fourier transform **we are decomposing the function into its component frequencies**.

How would you describe this function?



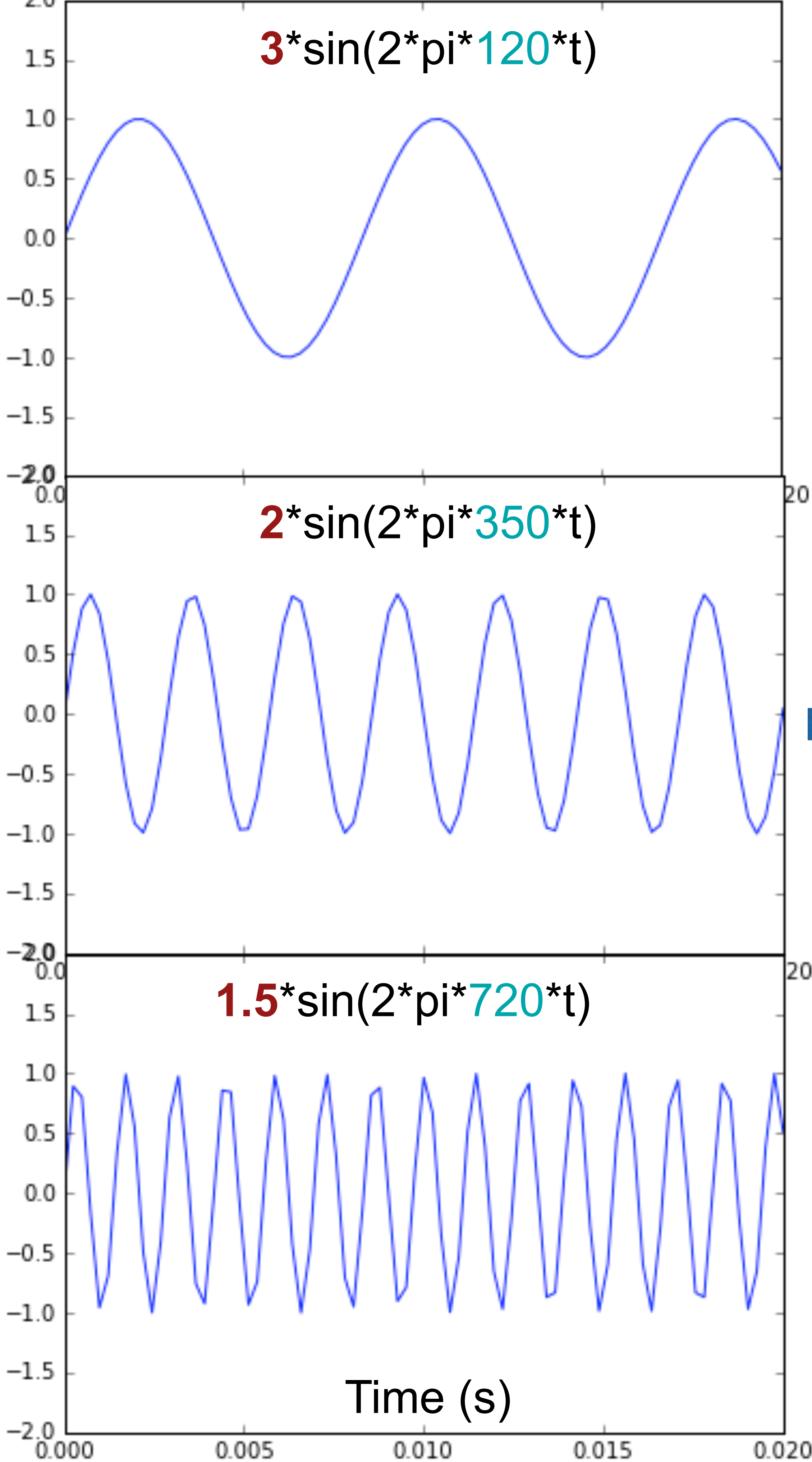
2 sinusoids

3 sinusoids

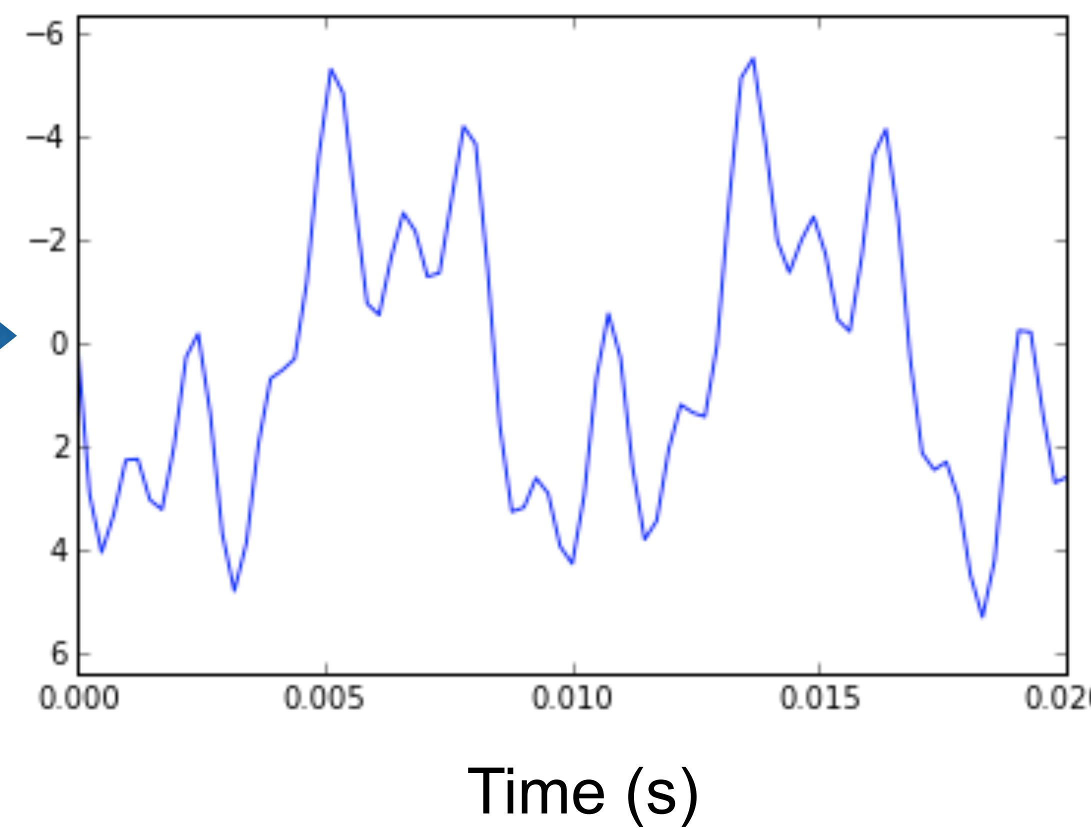
4 sinusoids

5 sinusoids

Slide from J. Kanner



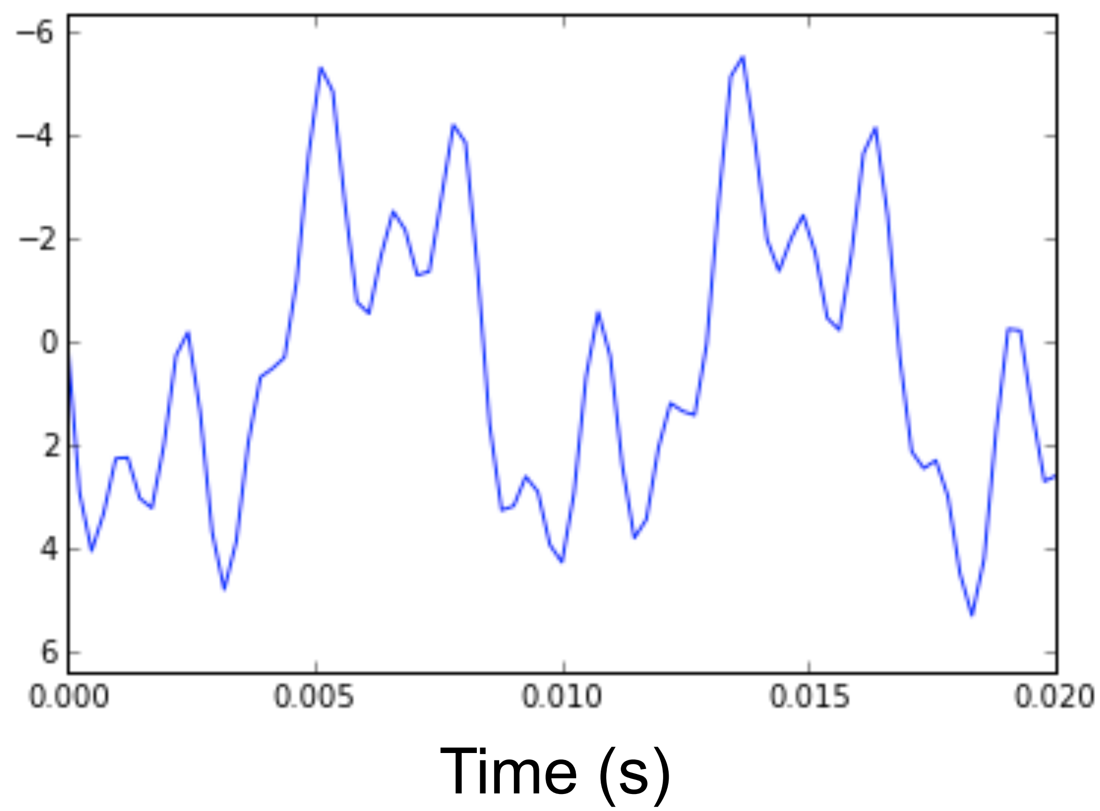
Our original function

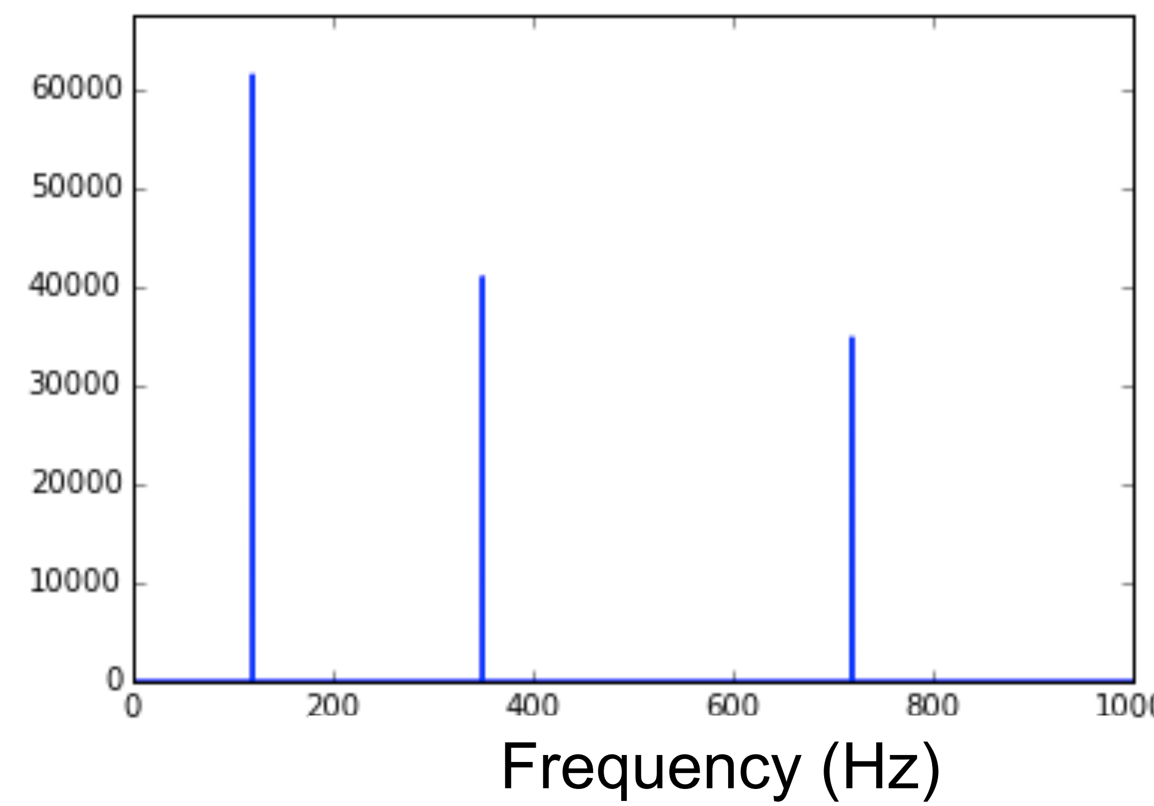


Slide from J. Kanner

Time Domain

Frequency Domain





h(t) – Position as a function of time

H(f) – Amplitude as a function of frequency

h(t) = 3 * sin(2*pi*120*t) +
2 * sin(2*pi*350*t) +
1.5* sin(2*pi*720*t)

|H(120 Hz)| = 3
|H(350 Hz)| = 2
|H(720 Hz)| = 1.5
H(f) = 0 otherwise

Fourier Transform

Slide from J. Kanner

Power Spectral Density

Parseval's theorem:

$$\int_{-\infty}^{\infty} dt\, |x(t)|^2 = \int_{-\infty}^{\infty} df\, |\tilde{x}(f)|^2$$

$\Rightarrow$ Total energy in the data can be calculated in either time domain or frequency domain

Units:

$|\tilde{x}(f)|^2$ Energy spectral density

(normalize by 1/T to get power)

Signal energy per unit frequency (per Hz)

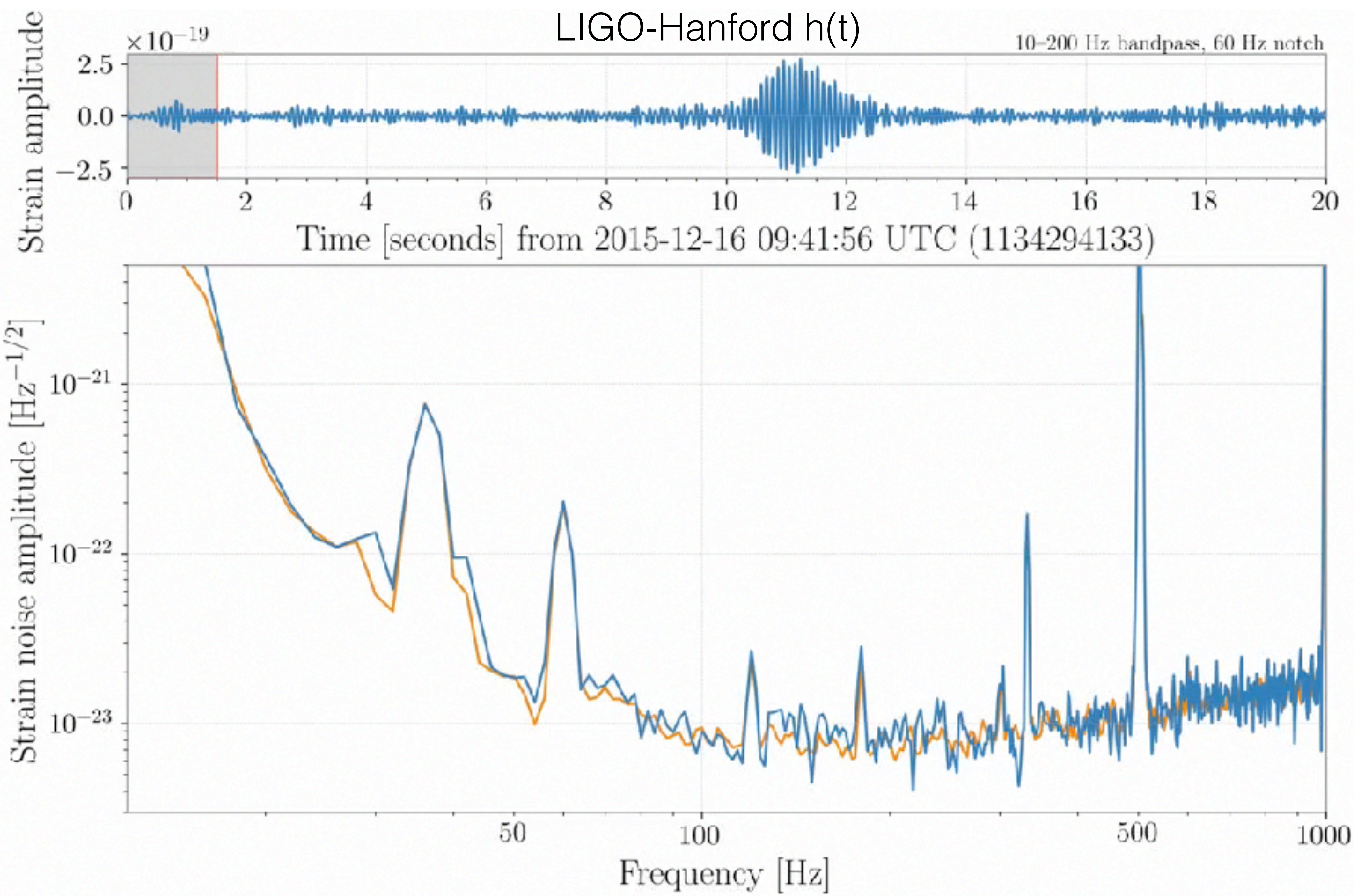
$|\tilde{x}(f)|$ $\propto$ Amplitude spectral density

(sqrt of power for each discrete frequency)

Signal amplitude per unit frequency (per *sqrt Hz*)

Slide adapted from P. Shawhan

LIGO data in time and frequency

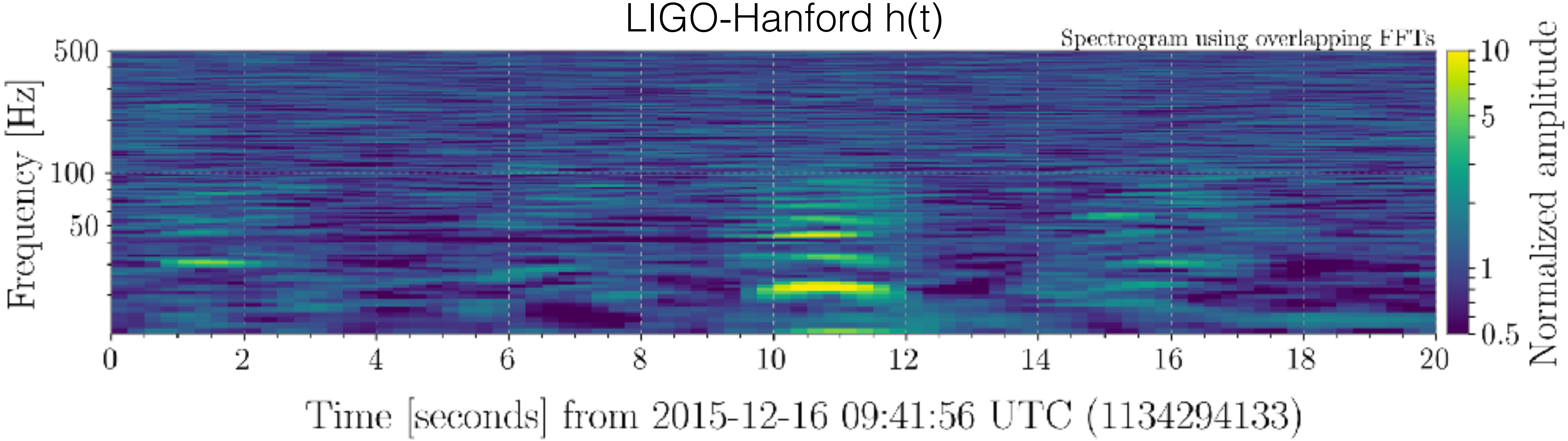


Made with GWpy by Duncan Macleod. Code: https://git.io/gwpy-ligo-scattering-animation

0.5 second FFT; 5 averages covering 1.5 seconds; 50% overlap

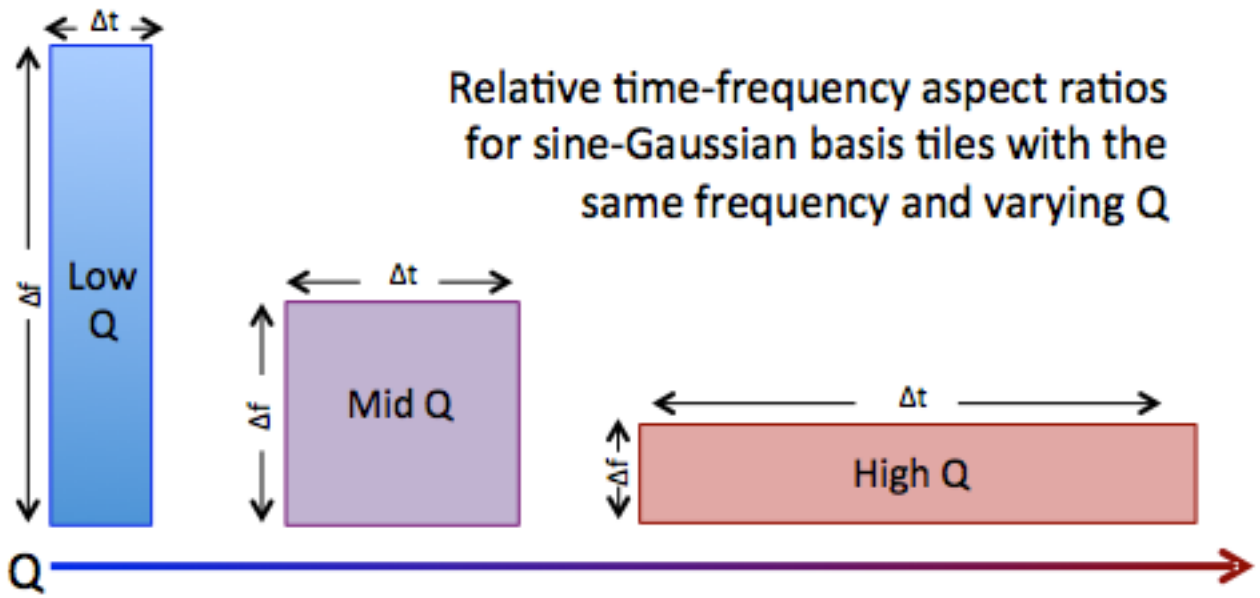
Time-frequency spectrogram

LIGO-Hanford h(t)



Made with GWpy by Duncan Macleod

The Q transform

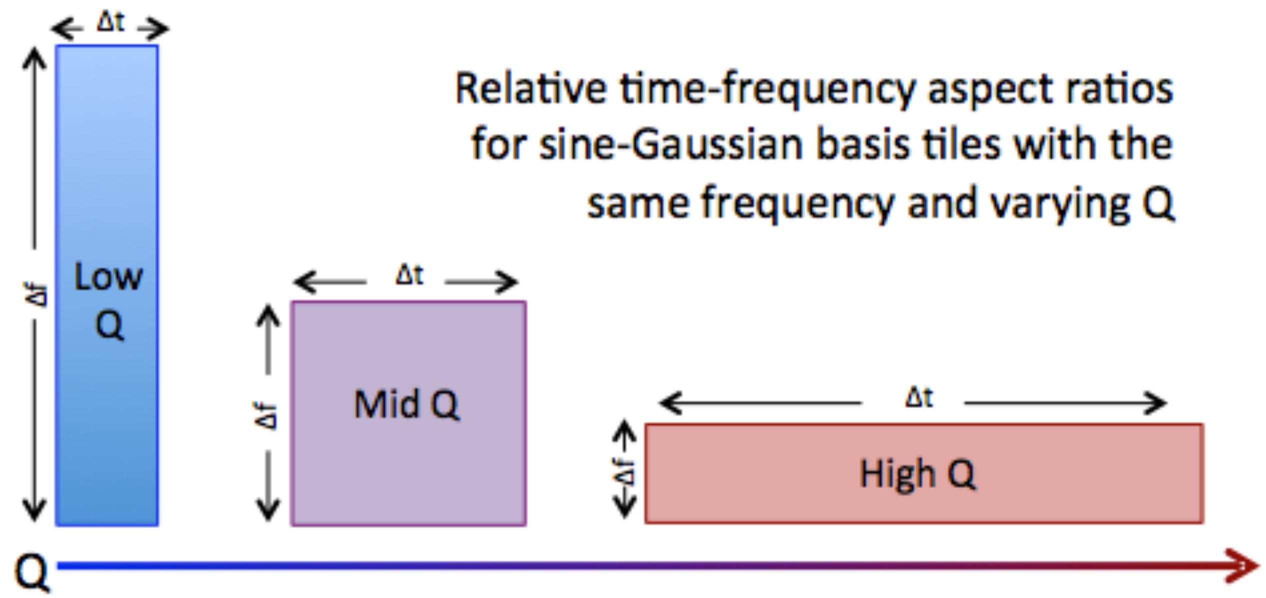


S. Chatterji et al. CQG (2010)
Images: McIver

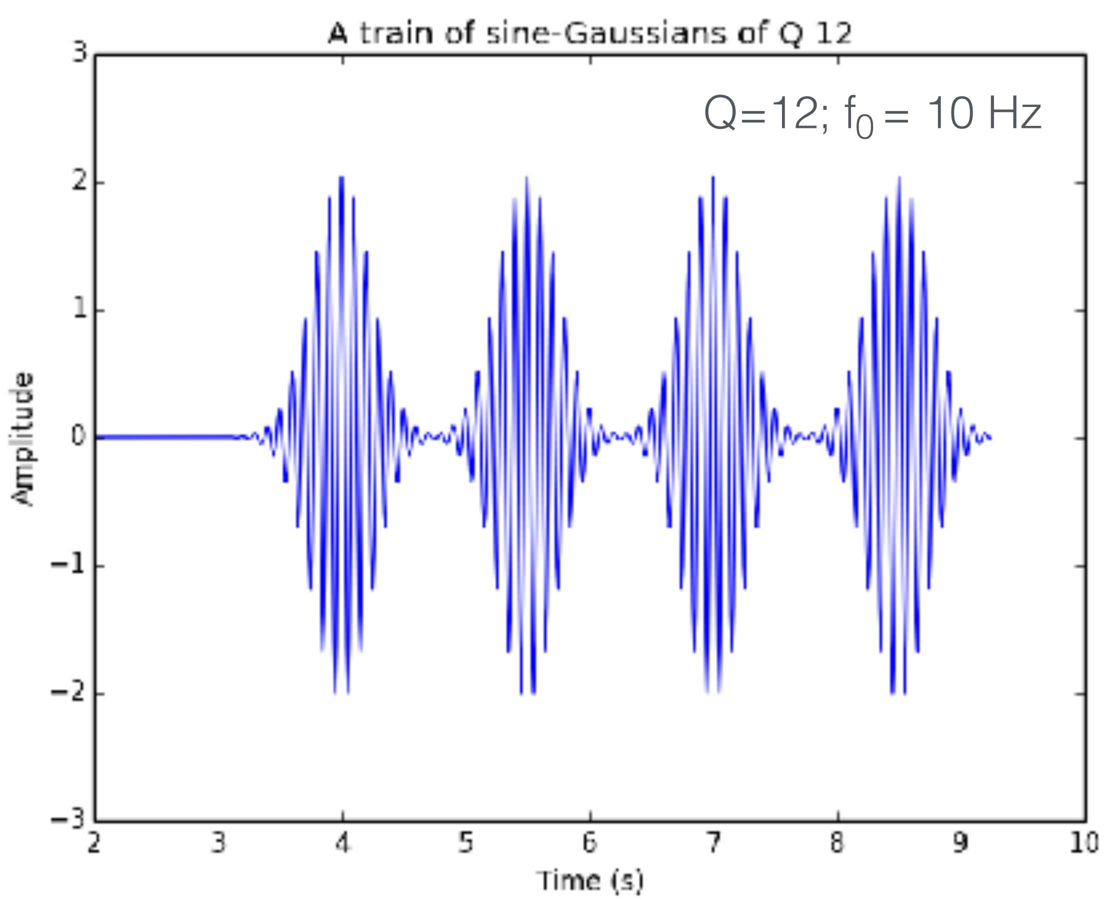
$$Q = \frac{f_0}{\Delta f}$$

The Q transform

S. Chatterji et al. CQG (2010)
Images: McIver

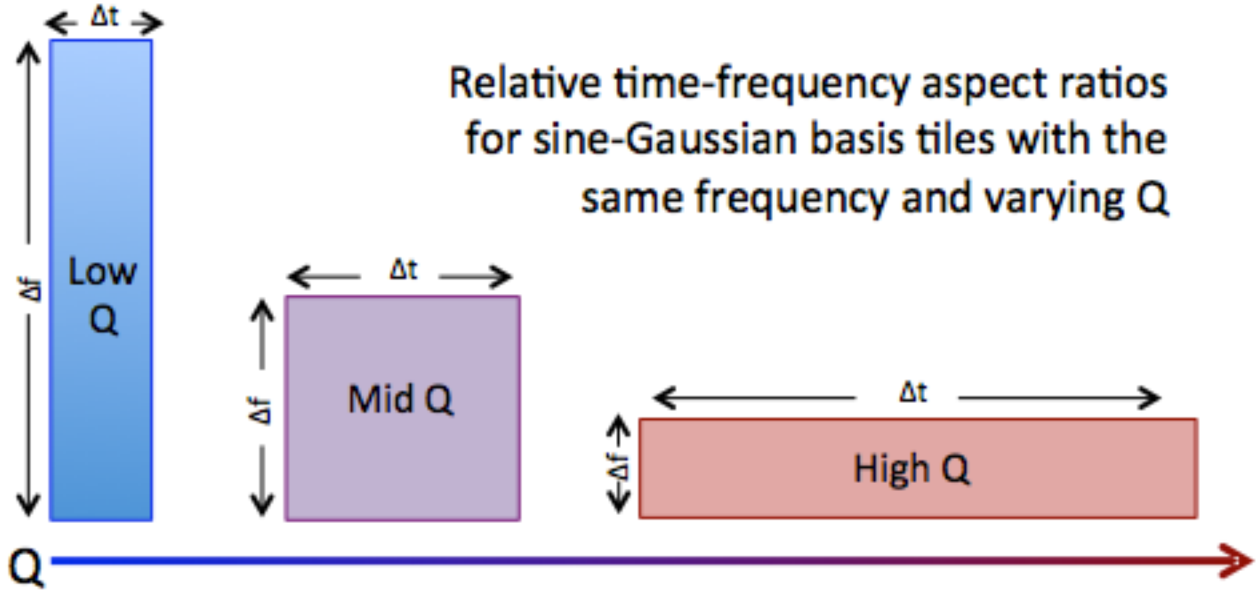


$$Q = \frac{f_0}{\Delta f}$$

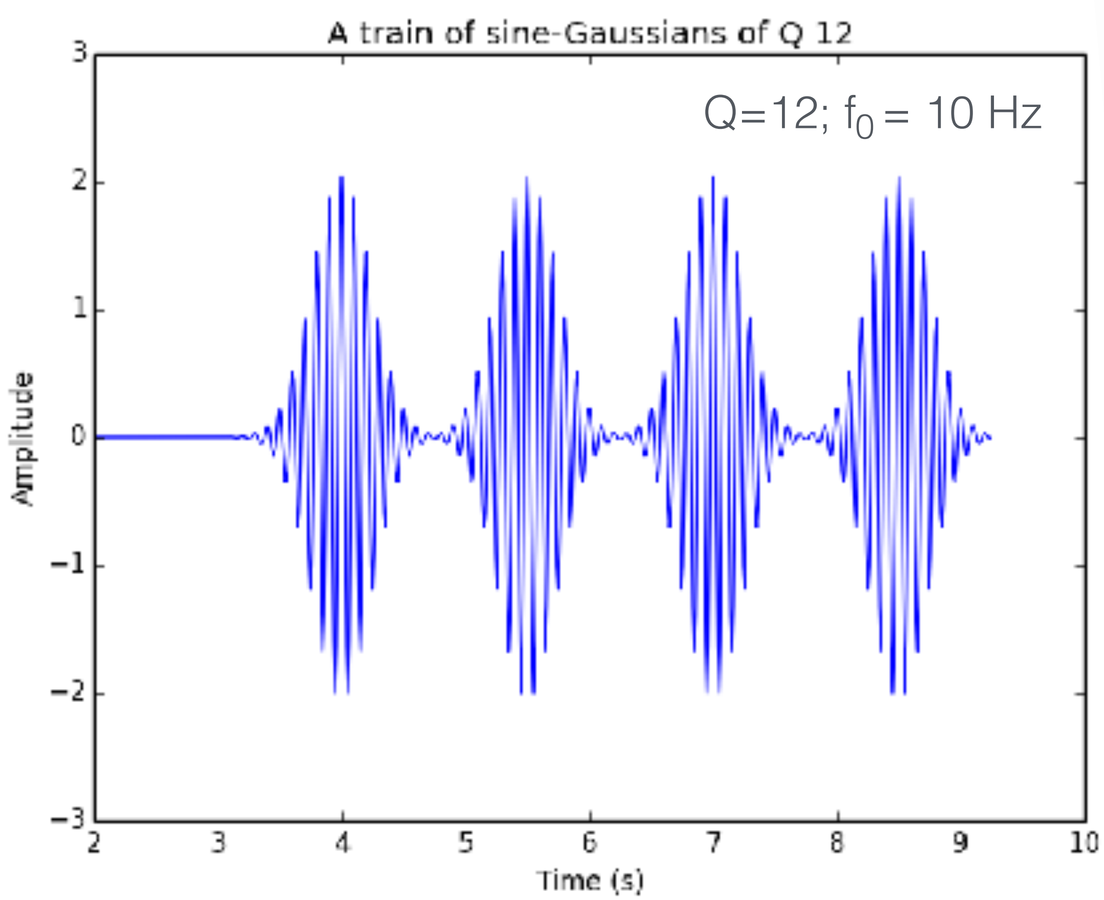


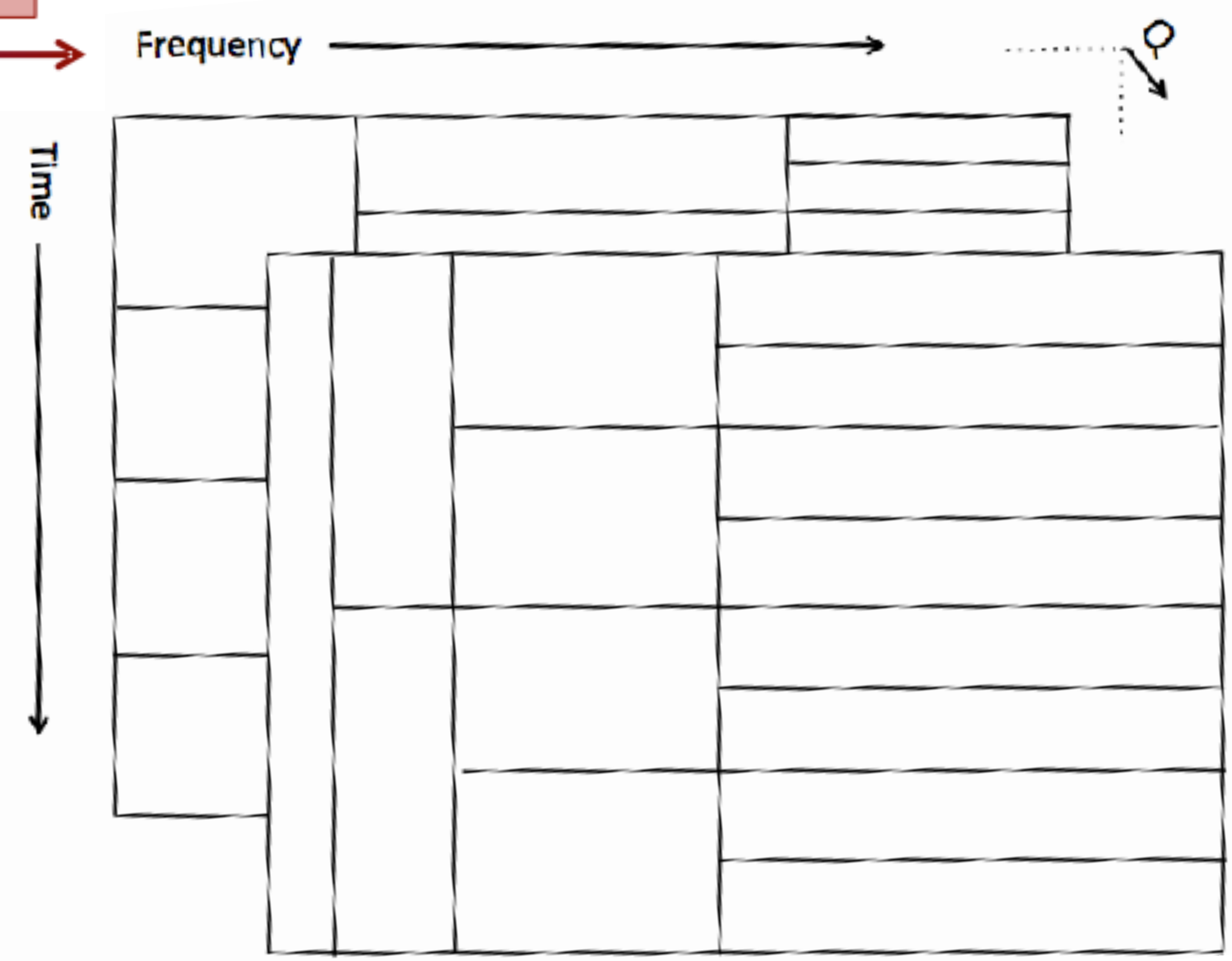
The Q transform

S. Chatterji et al. CQG (2010)
Images: McIver



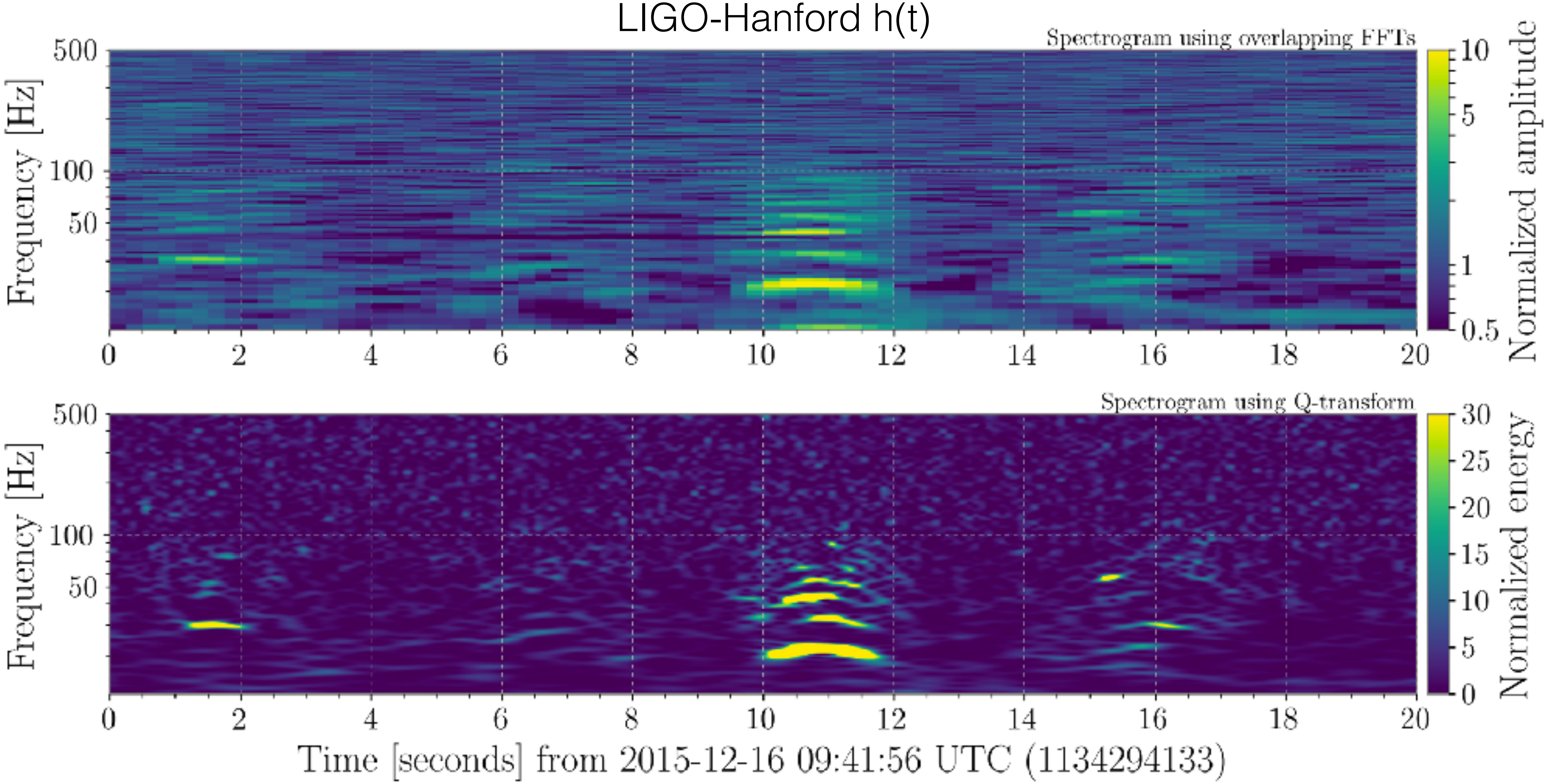
$$Q = \frac{f_0}{\Delta f}$$





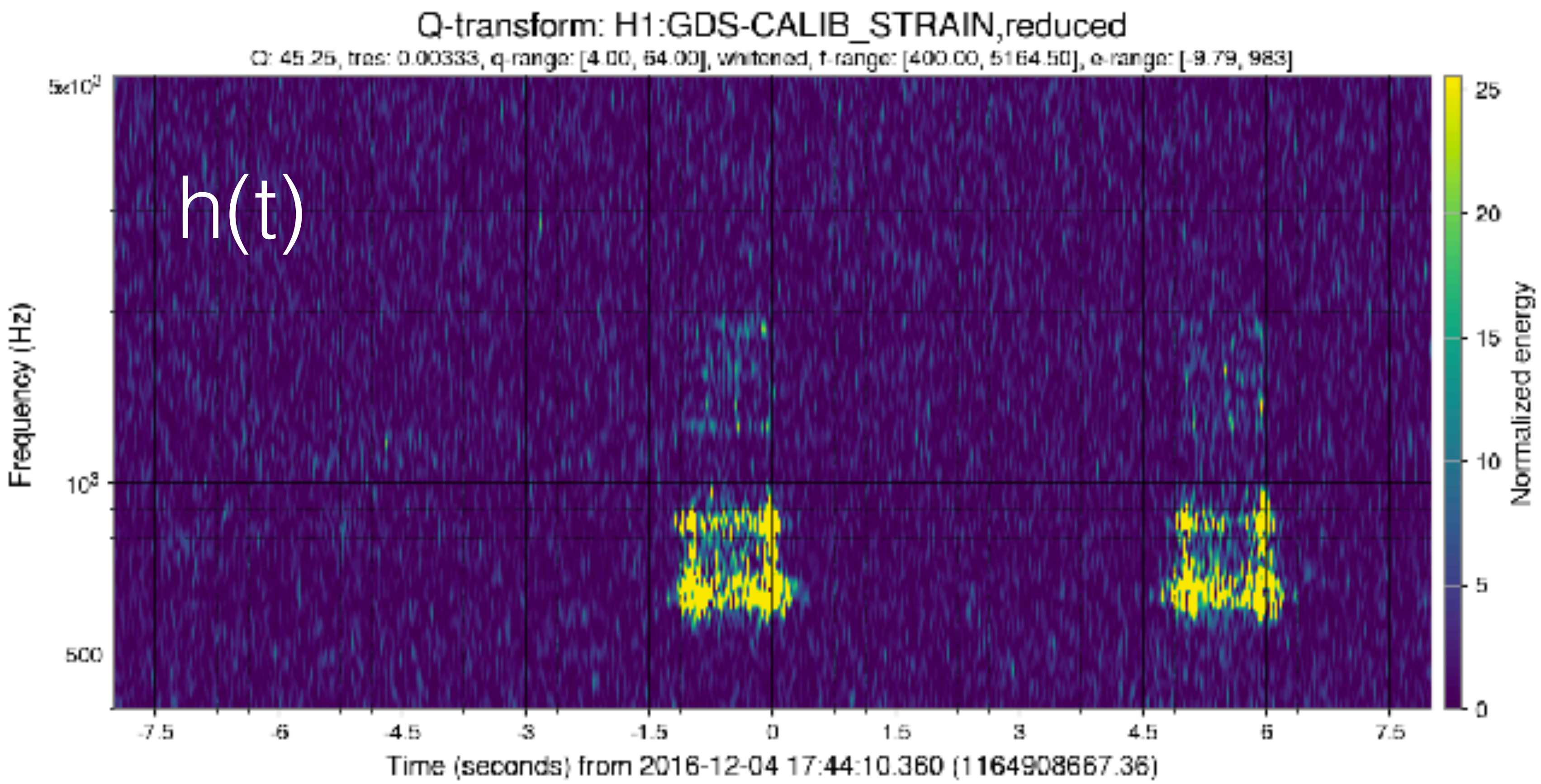
Time-frequency spectrograms

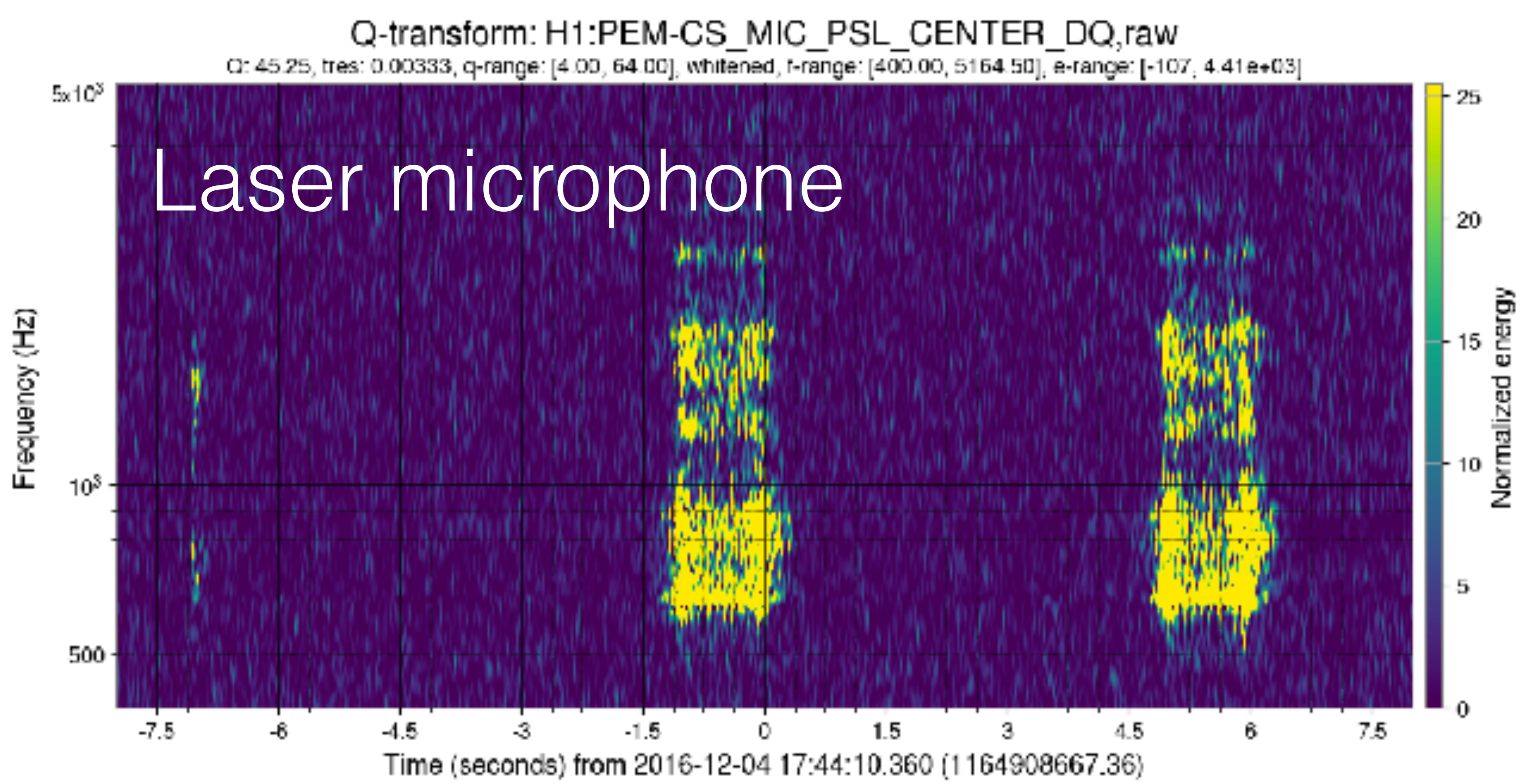
LIGO-Hanford h(t)



Made with GWpy by Duncan Macleod

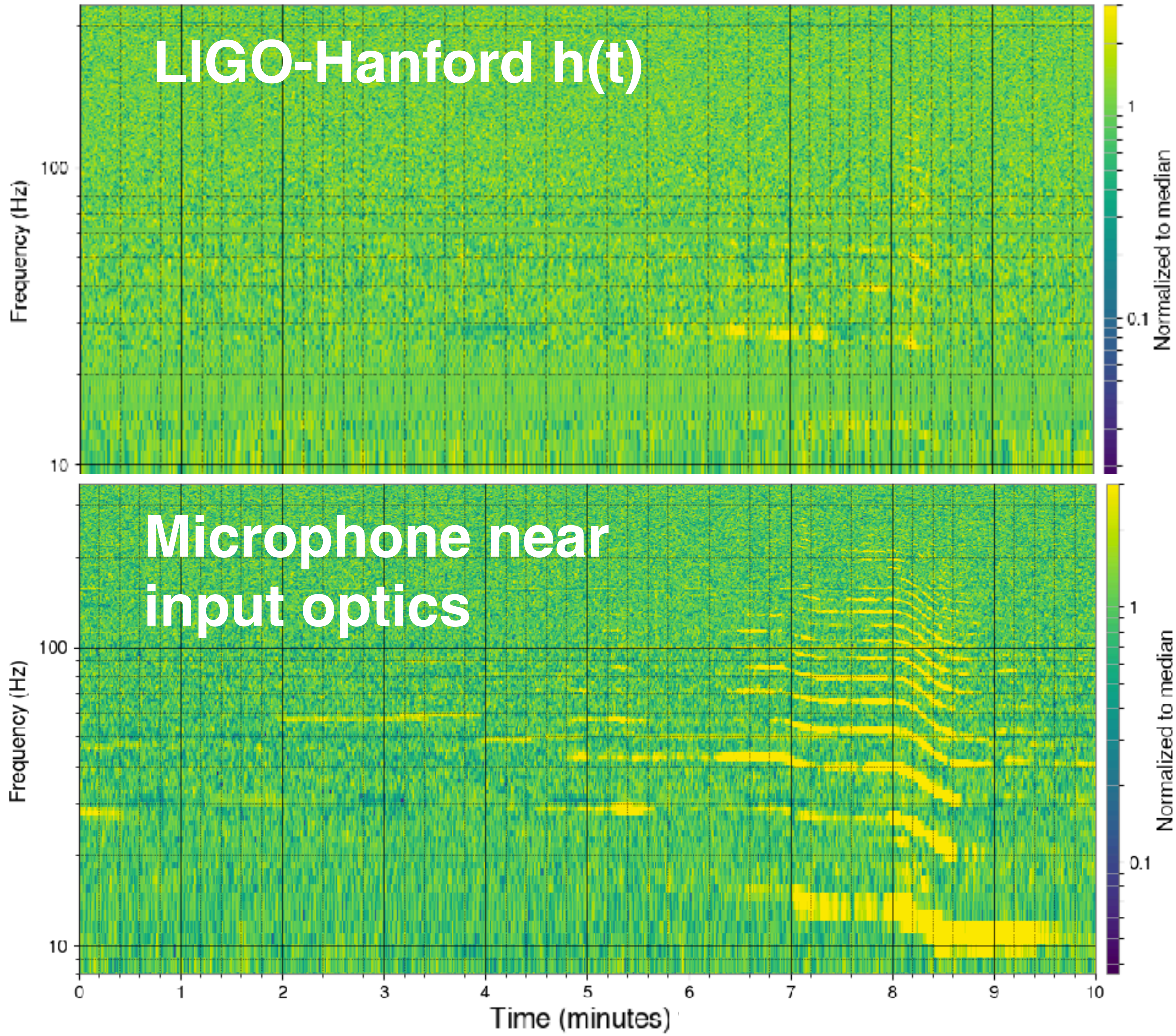
Sleuthing with data visualization: *what is this!?*





LHO alog #32503: https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=32503

Sleuthing with data visualization: *what is this!?*



GWpy data visualization: Try it for yourself

Get some practice with GWpy - access GW detector data and make an Omegascan

Navigate to tutorials from the 2020 GWOSC workshop:
https://github.com/gw-odw/odw-2020/tree/master/Day_1

Work through tutorials 1.1-1.3

Support will be available in ***office hours:***
16:00 June 11th and 15th

Note: this material is also available in the form of a 2021 web course on the GWOSC

Outline

GW detector characterization

The challenge of non-stationary GW detector data

- Uptime
- Search sensitivity and PE accuracy
- The “worst offender” glitches

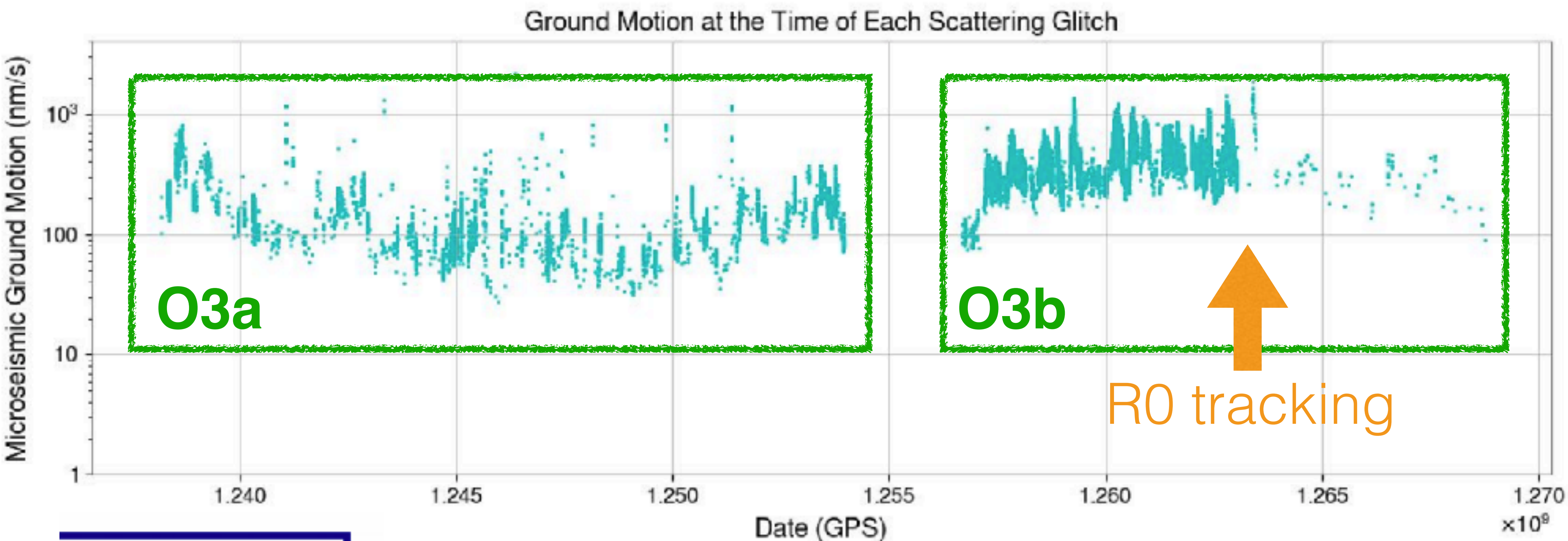
Techniques for identification and characterization

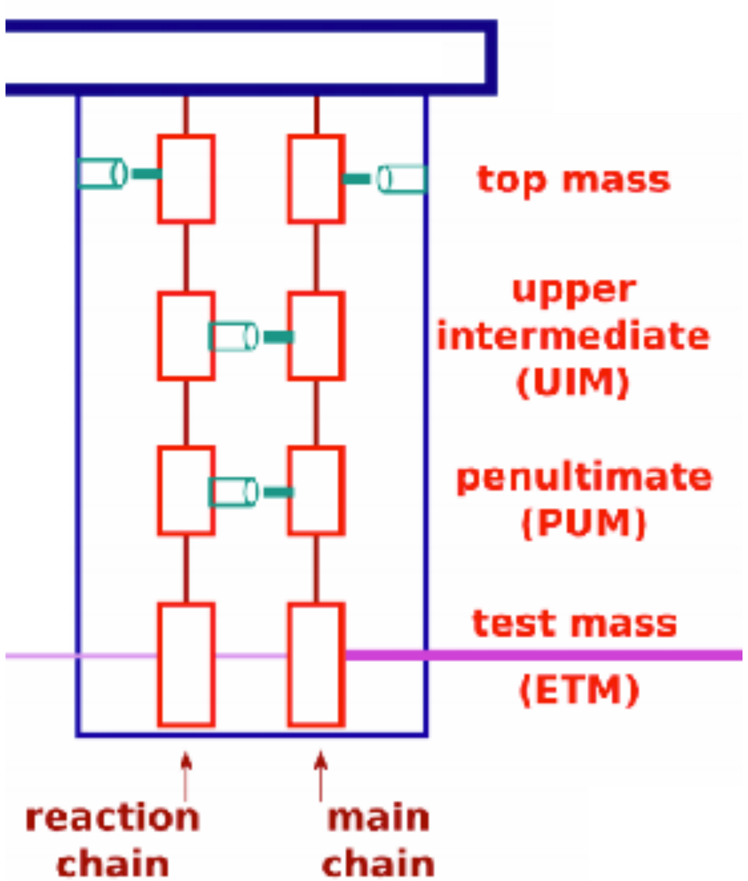
- Data visualization: into the frequency domain!
- GWOSC tools: GWpy and Omegascans

Mitigation: Data quality products and glitch subtraction

An overview of GW detector calibration

The ideal: fix it in the detector!





- Above: **LIGO-Hanford** scattering glitches and microseism
- R0 tracking was implemented in January 2020 at the LIGO detectors
- *O3b LIGO-Virgo OPA retraction rate dropped 50% afterward*

K. Rink, LHO alog 56080; S. Soni et al 2020

When we can't fix it: data quality products

Available via the GWOSC

Note: These definitions are slightly different between LIGO and Virgo data

Bit	Short Name	Description
Data Quality Bits		
0	DATA	data present
1	CBC_CAT1	passes the cbc CAT1 test
2	CBC_CAT2	passes cbc CAT2 test
3	CBC_CAT3	passes cbc CAT3 test
4	BURST_CAT1	passes burst CAT1 test
5	BURST_CAT2	passes burst CAT2 test
6	BURST_CAT3	passes burst CAT3 test
Injection Bits		
0	NO_CBC_HW_INJ	no cbc injection
1	NO_BURST_HW_INJ	no burst injections
2	NO_DETCHAR_HW_INJ	no detchar injections
3	NO_CW_HW_INJ	no continuous wave injections
4	NO_STOCH_HW_INJ	no stoch injections

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

CAT1 (Category 1): Failing a data quality check at this category indicates **a critical issue with a key detector component not operating in its nominal configuration**.

- These times are identical for each data analysis group.
- *Times that fail CAT1 flags are not available as LIGO open data.*

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

CAT1 (Category 1): Failing a data quality check at this category indicates **a critical issue with a key detector component not operating in its nominal configuration**.

- These times are identical for each data analysis group.
- *Times that fail CAT1 flags are not available as LIGO open data.*

CAT2 (Category 2): Failing a data quality check at this category indicates times when there is a **known, understood physical coupling to the gravitational wave channel**. For example, high seismic activity.

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

CAT1 (Category 1): Failing a data quality check at this category indicates **a critical issue with a key detector component not operating in its nominal configuration**.

- These times are identical for each data analysis group.
- *Times that fail CAT1 flags are not available as LIGO open data.*

CAT2 (Category 2): Failing a data quality check at this category indicates times when there is a **known, understood physical coupling to the gravitational wave channel**. For example, high seismic activity.

CAT3 (Category 3): Failing a data quality check at this category indicates times when there is **statistical coupling to the gravitational wave channel** which is not fully understood.

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

CAT1 (Category 1): Failing a data quality check at this category indicates **a critical issue with a key detector component not operating in its nominal configuration**.

- These times are identical for each data analysis group.
- *Times that fail CAT1 flags are not available as LIGO open data.*

CAT2 (Category 2): Failing a data quality check at this category indicates times when there is a **known, understood physical coupling to the gravitational wave channel**. For example, high seismic activity.

CAT3 (Category 3): Failing a data quality check at this category indicates times when there is **statistical coupling to the gravitational wave channel** which is not fully understood.

Data quality levels are defined in a cumulative way: a time which fails a given category automatically fails all higher categories.

Data quality categories are defined independently for different analysis groups: if something fails at CAT2_BURST, it could pass CAT2_CBC.

How are data quality segments defined?

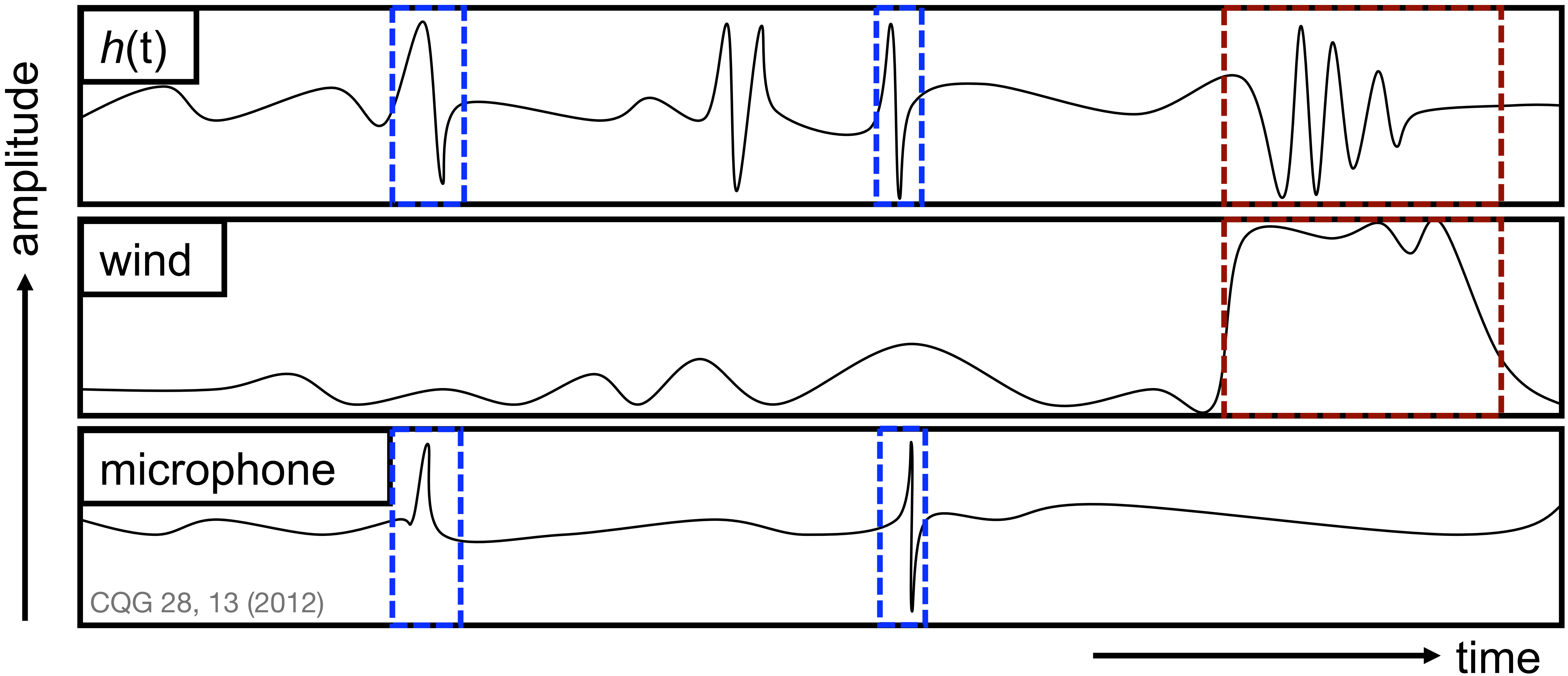
For O1-O3 and prior analyses:

- Data quality vetoes required an *auxiliary witness*
- That auxiliary witness was required to be ***safe***; to not be sensitive to changes in spacetime strain
    - This is tested empirically with **hardware injections** - inject a signal into h(t) and conduct statistical correlation tests with auxiliary channels of interest
- Veto segments were defined based on noise sources known to couple to h(t)
- Veto categories were determined for each type of search independently depending on noise contributors to that search's background
    - Note: There are differences between CAT2 and CAT3 definitions between the burst and CBC searches

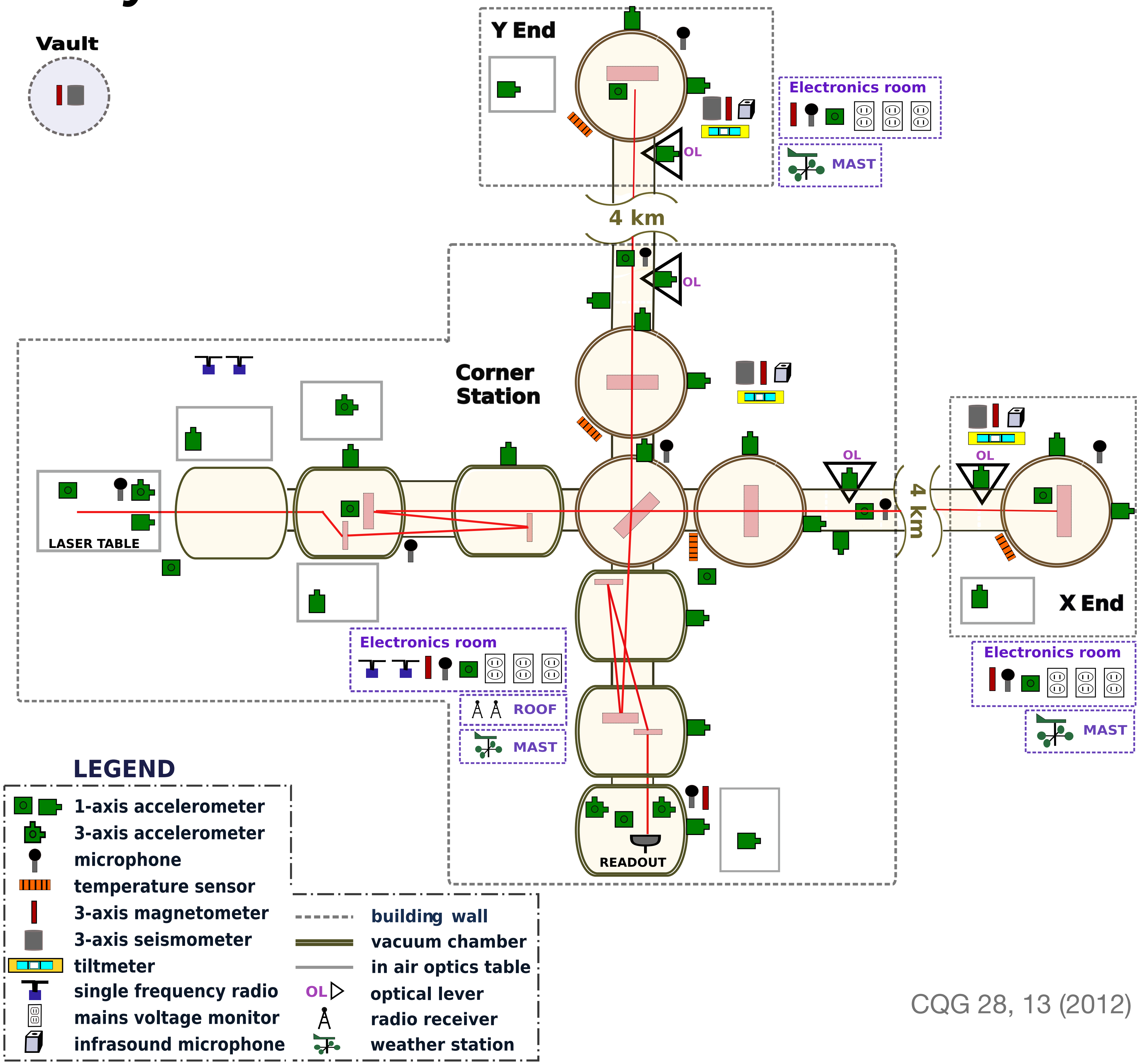
Auxiliary channels

We record **over 200,000 channels per detector** that monitor the environment and detector behavior.

We can use these to **help trace the instrumental causes of glitches** that pollute the search backgrounds.

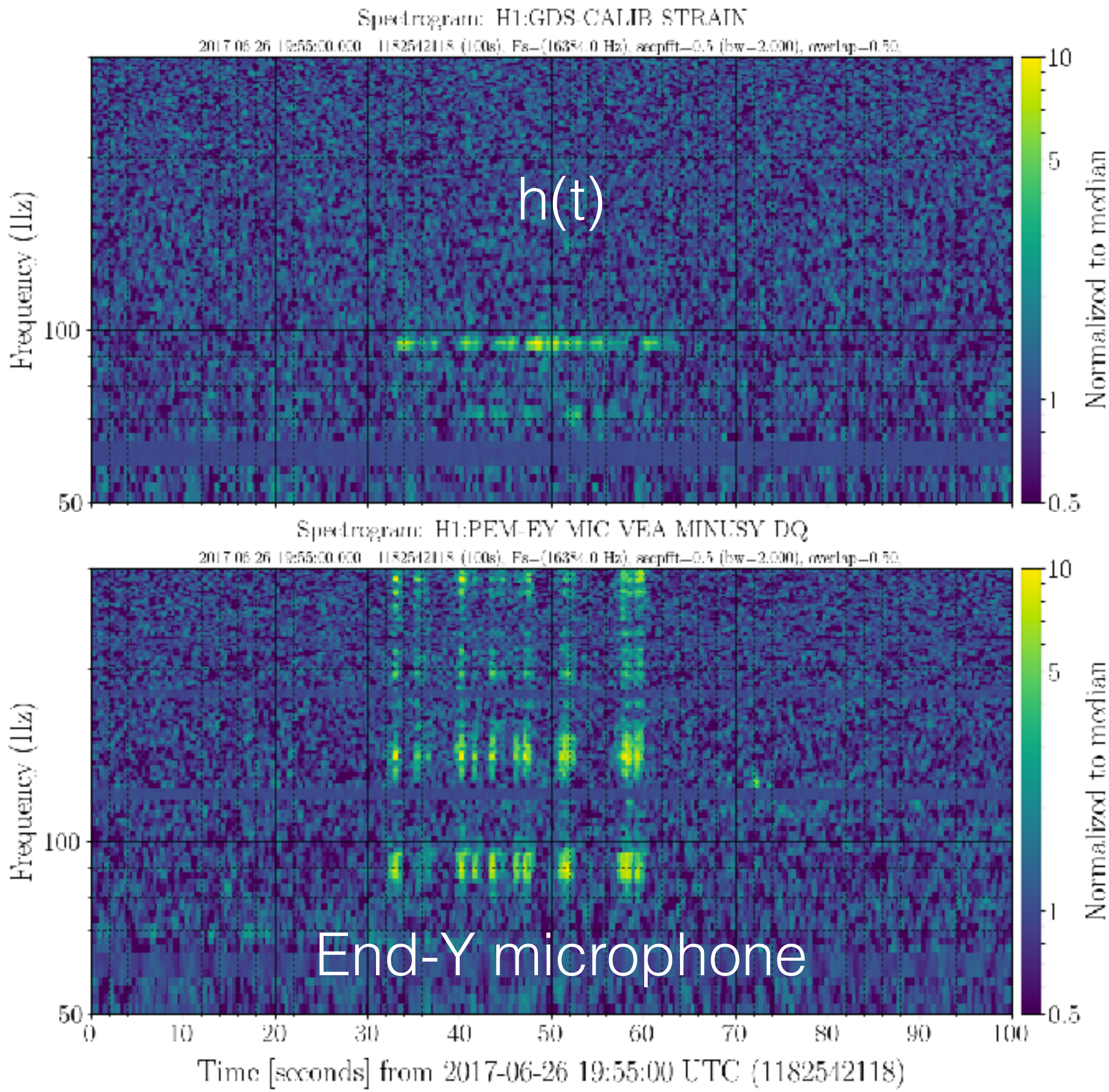


Physical environment channels

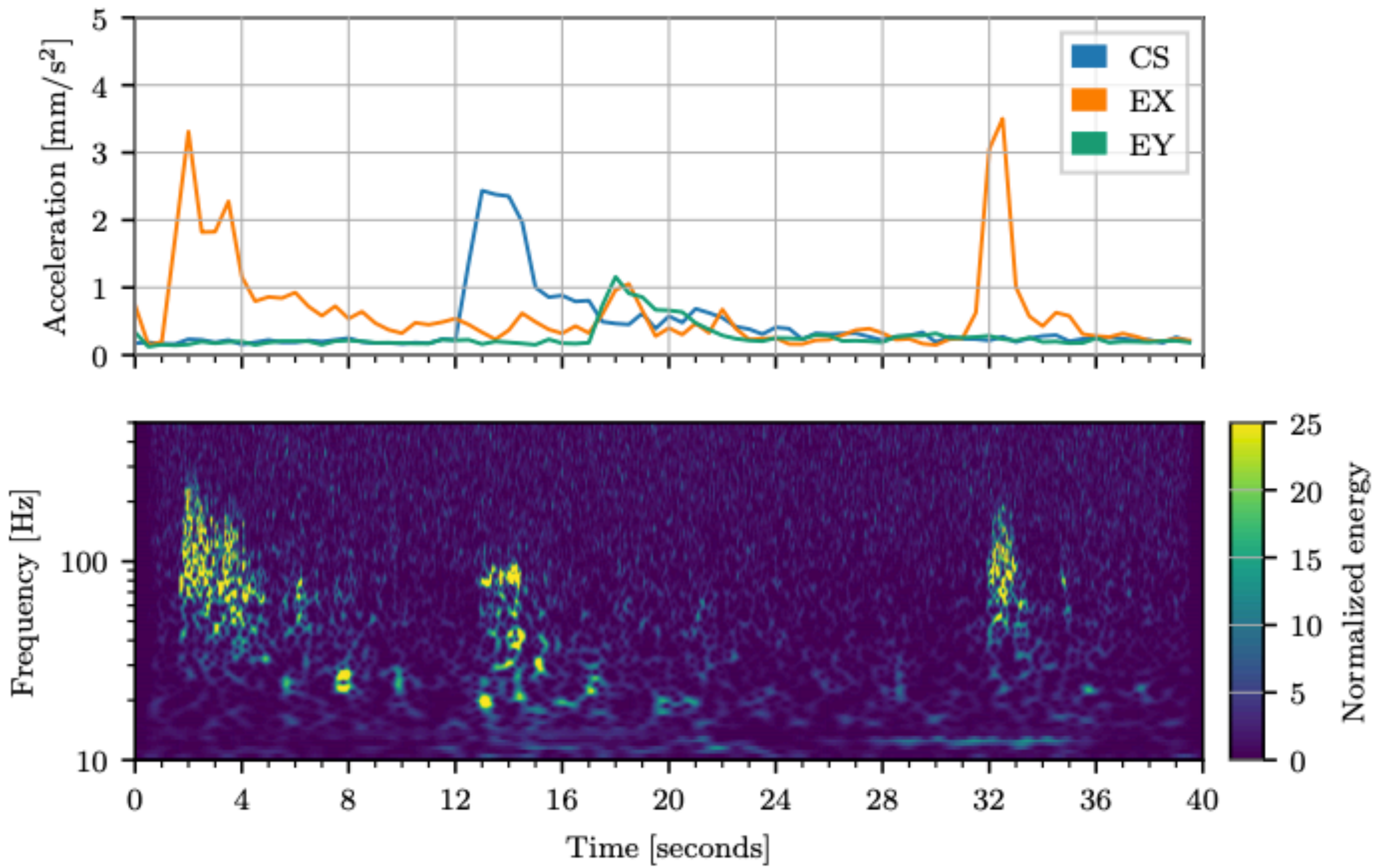


CQG 28, 13 (2012)

Example: glitches witnessed by microphone

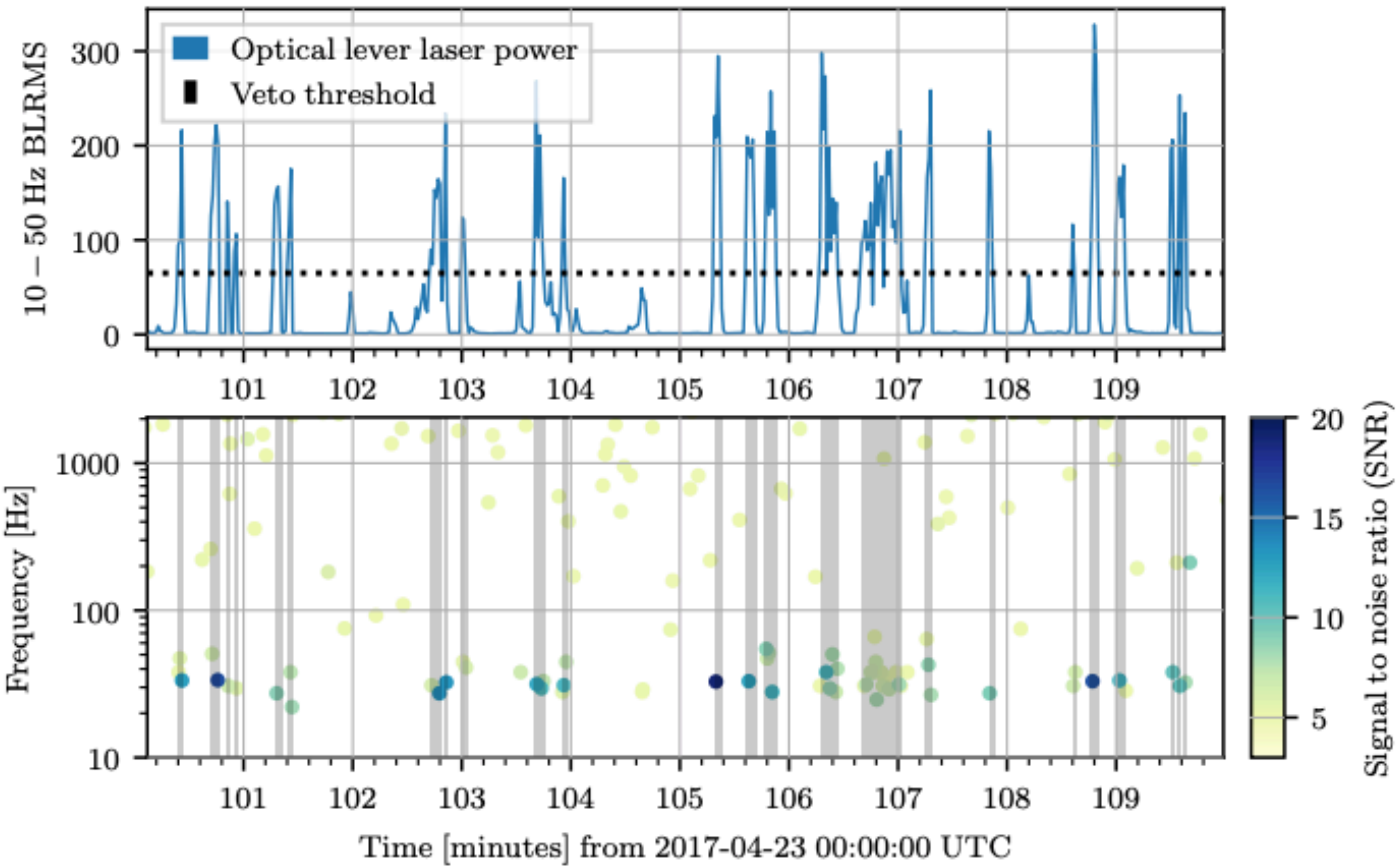


Example: glitches witnessed by accelerometers



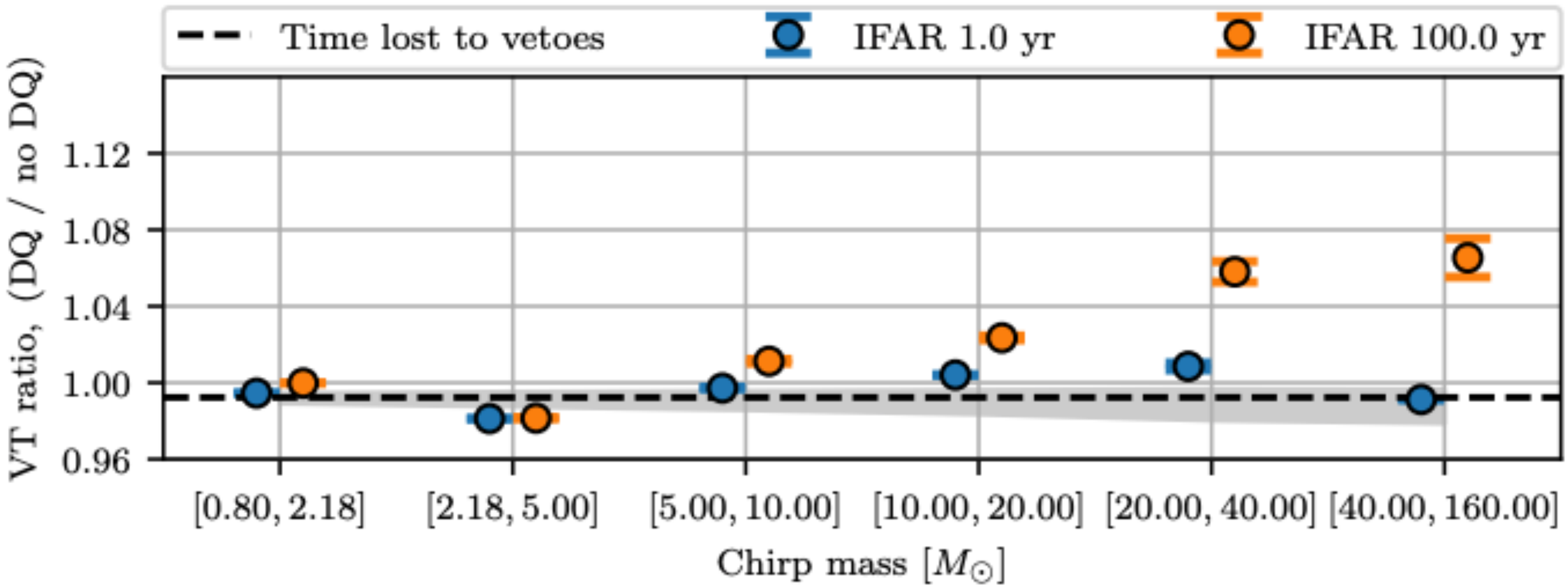
Davis et al. CQG 2021

Example of a data quality veto



Davis et al. CQG 2021

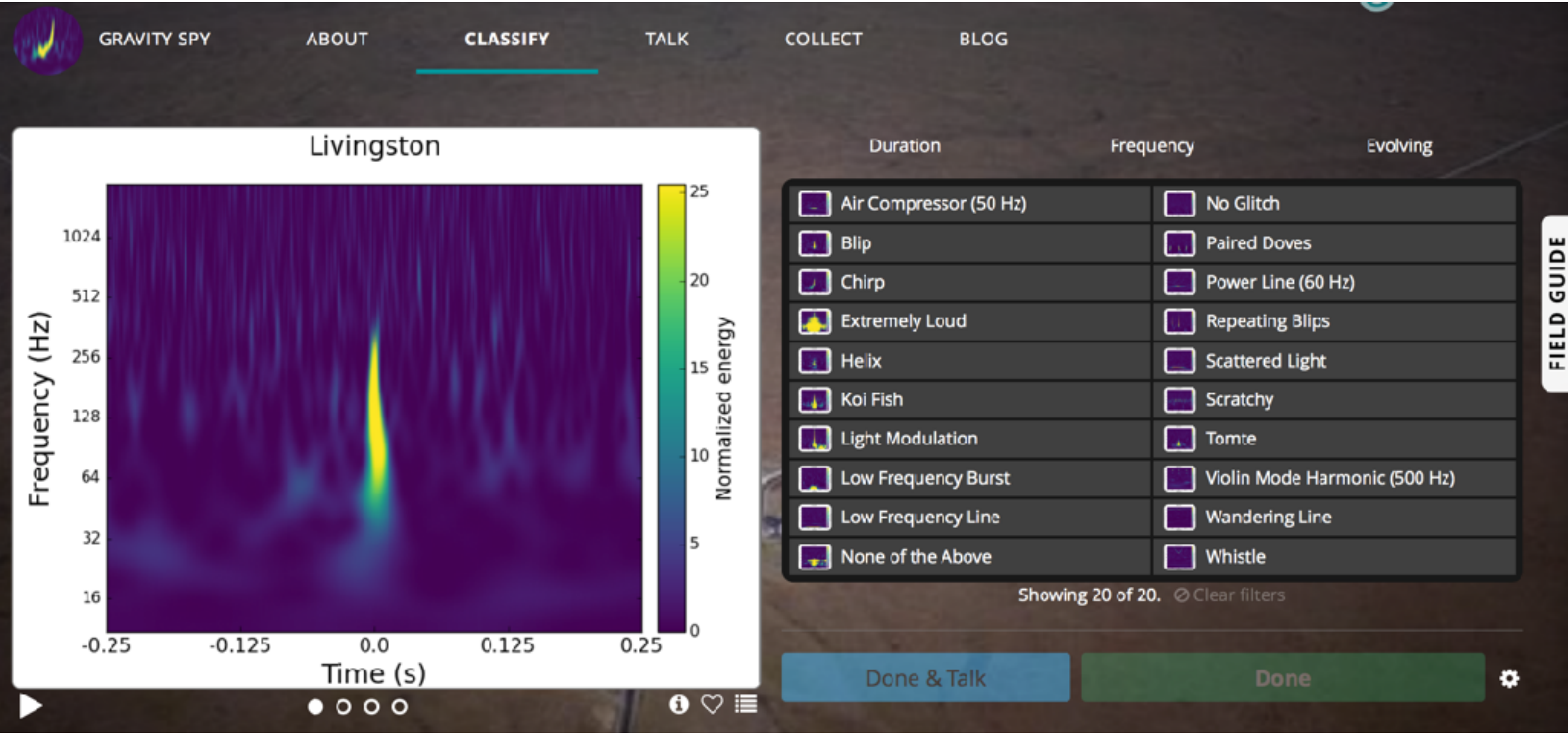
Measuring the impact of DQ vetoes: volume-time



Davis et al. CQG 2021

Classifying LIGO-Virgo glitches with no witness

gravityspy.org



Zevin et al, 2017, CQG

LigoDV-web and Glitch DB

https://ldvw.ligo.caltech.edu/ldvw/view

LIGO

LigoDV-Web — *v0.1.07 Glitch DB*

Welcome Jessica McIver (admin)

Home | History ▾ | Status ▾ | Glitch DB ▾ | Batch ▾ | Help ▾ | Admin ▾

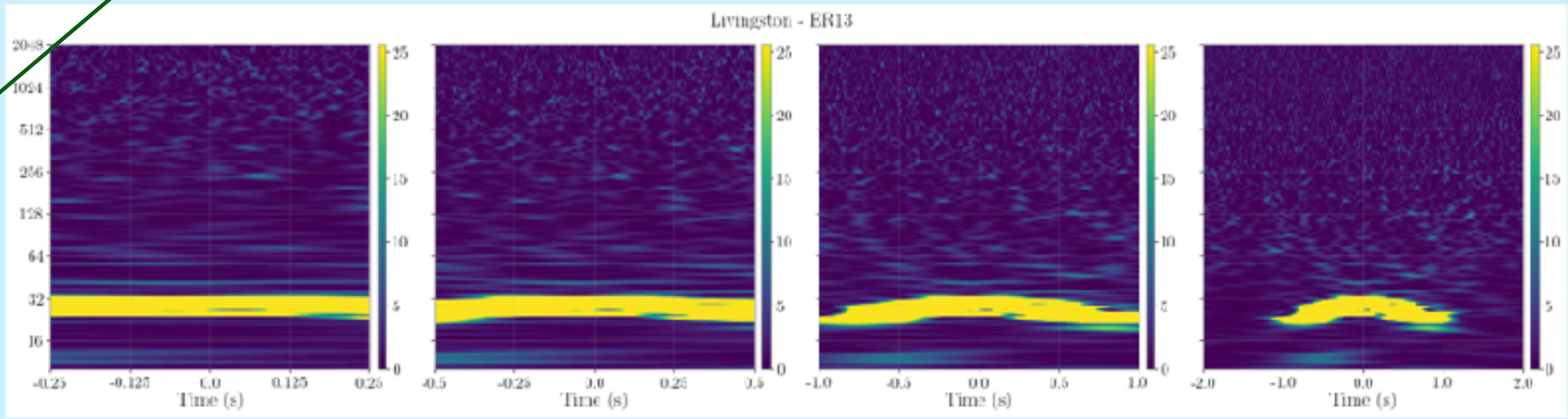
Link to this search

Searchable database of CNN-classified glitches

< | Page: 1 of 3 | Go | > | Save comments

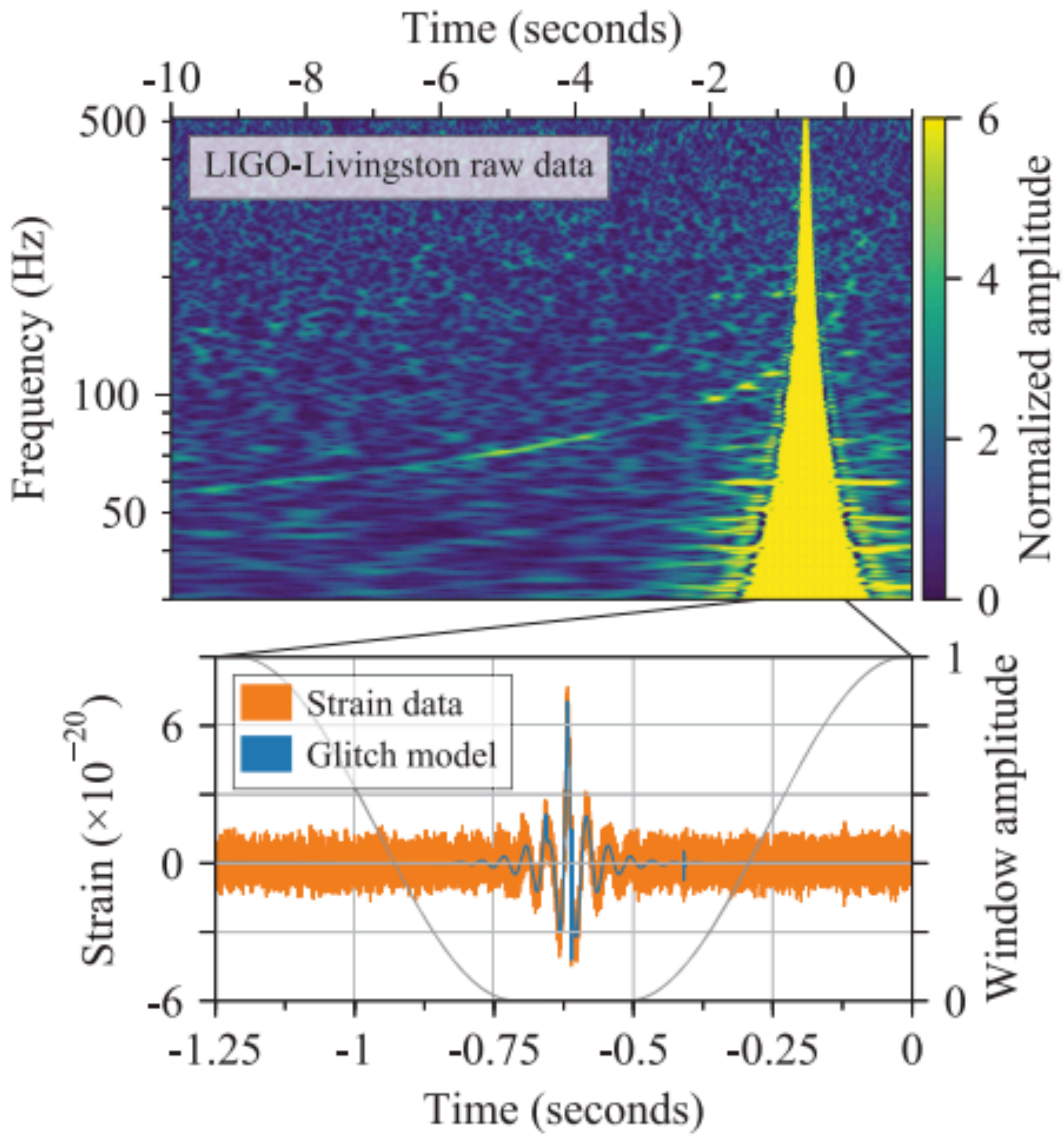
249 matches found.

Unique Id	tihveWX1dQ
GPS Time	1229057821.313
UTC Time	2018-12-17 04:56:43
Pipeline	GravitySpy
Class	Scattered_Light
Status	testing
Ifo	L1
SNR	80.946
Amplitude	1.89e-21
Peak freq	28.117
Central freq	3977.720



J. Areeda et al. Astronomy and Computing (2017), S. Coughlin et al (2019)

Mitigating glitches near signals for PE



LIGO-Virgo collaboration PRL (2017)

Glitch subtraction with BayesWave

$$\mathbf{s} = \mathbf{R} \star \mathbf{h} + \mathbf{g} + \mathbf{n}_G$$

Data → $\mathbf{s}$

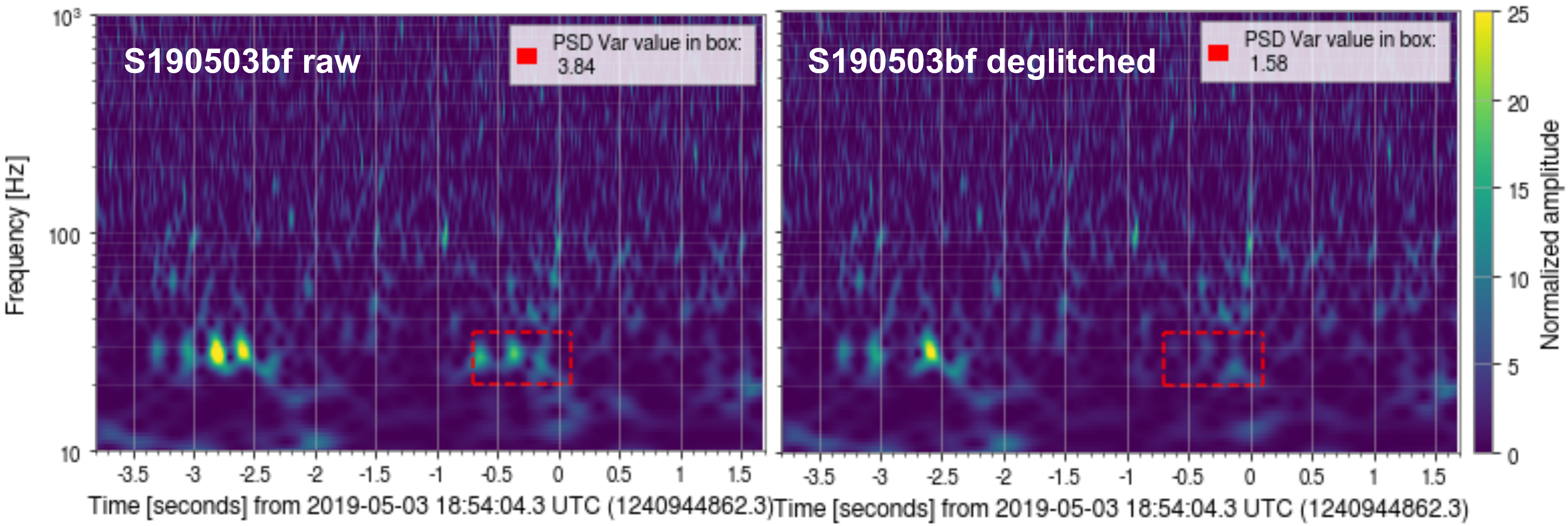
Glitches → $\mathbf{g}$

Gaussian noise described by PSD → $\mathbf{n}_G$

Detector response * GW signal → $\mathbf{R} \star \mathbf{h}$

Signal: coherent wavelets
Glitches: incoherent wavelets

What is "Gaussian enough" for PE? Use PSD non-stationarity test



Plots by D. Davis. BayesWave paper: Cornish and Littenberg CQG 2014

Outline

GW detector characterization

The challenge of non-stationary GW detector data

- Uptime
- Search sensitivity and PE accuracy
- The “worst offender” glitches

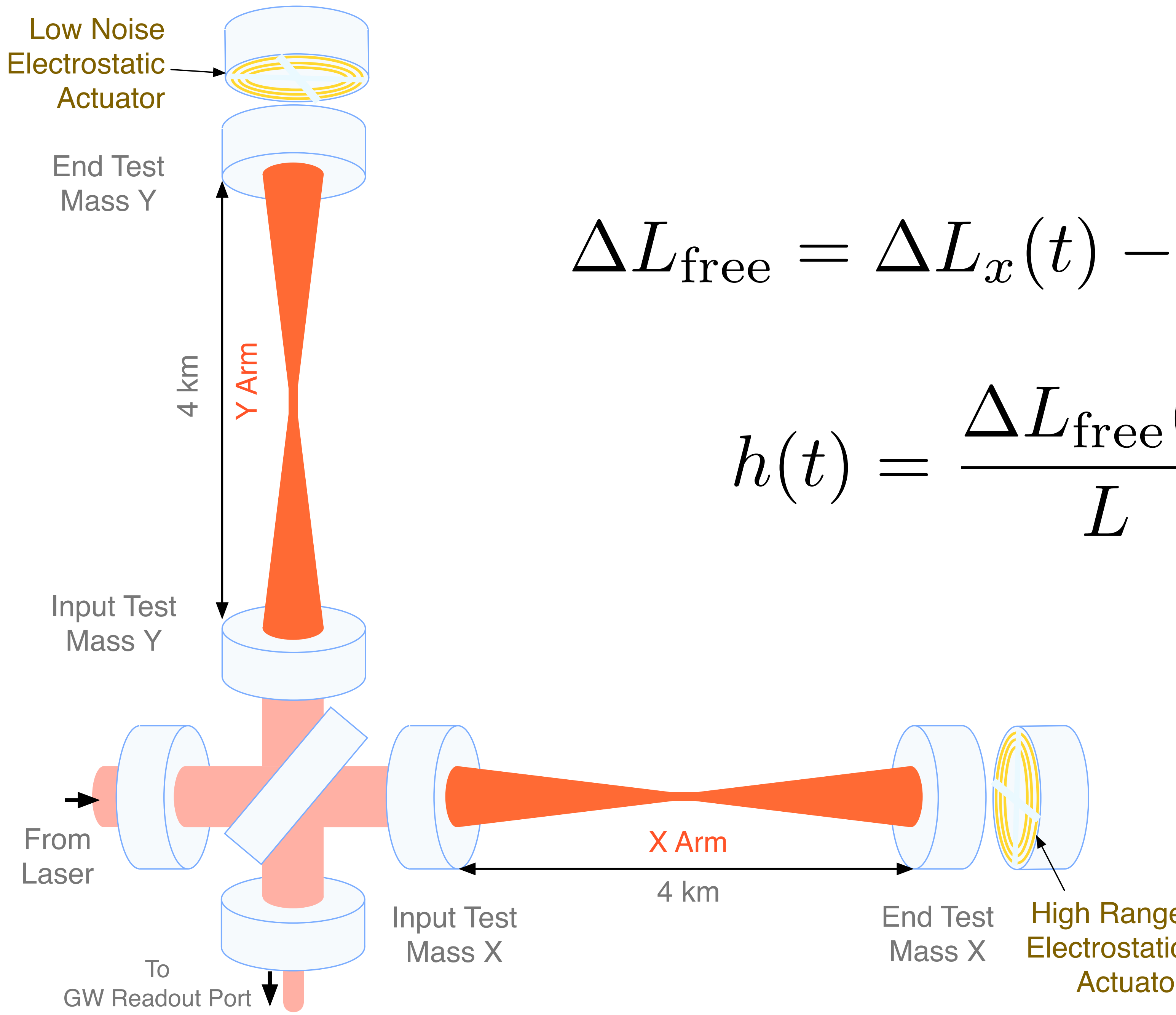
Techniques for identification and characterization

- Data visualization: into the frequency domain!
- GWOSC tools: GWpy and Omegascans

Mitigation: Data quality products and glitch subtraction

An overview of GW detector calibration

Calibrating GW detector data

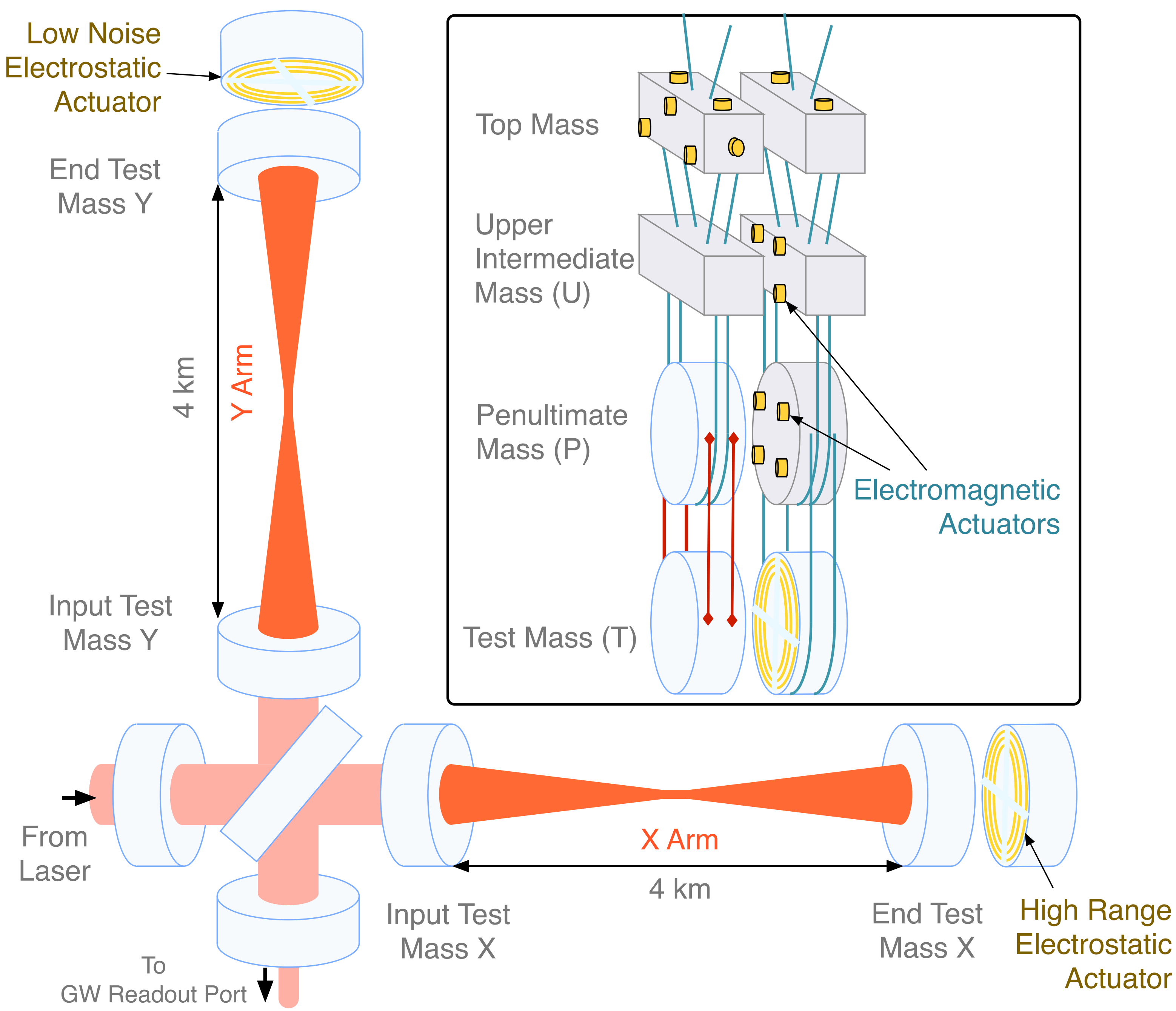


$$\Delta L_{\text{free}} = \Delta L_x(t) - \Delta L_y(t)$$

$$h(t) = \frac{\Delta L_{\text{free}}(t)}{L}$$

[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

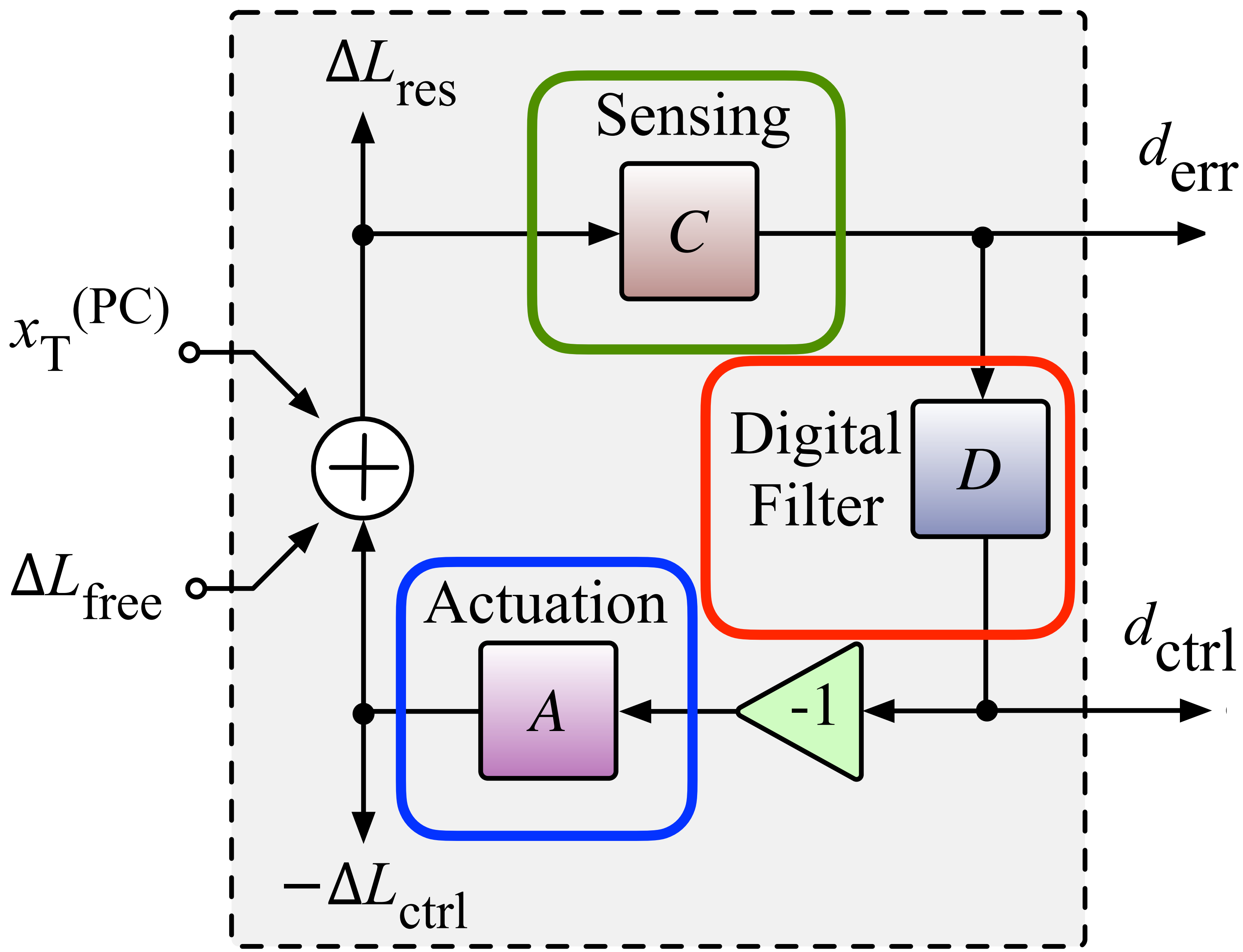
Basic GW detector interferometer schematic



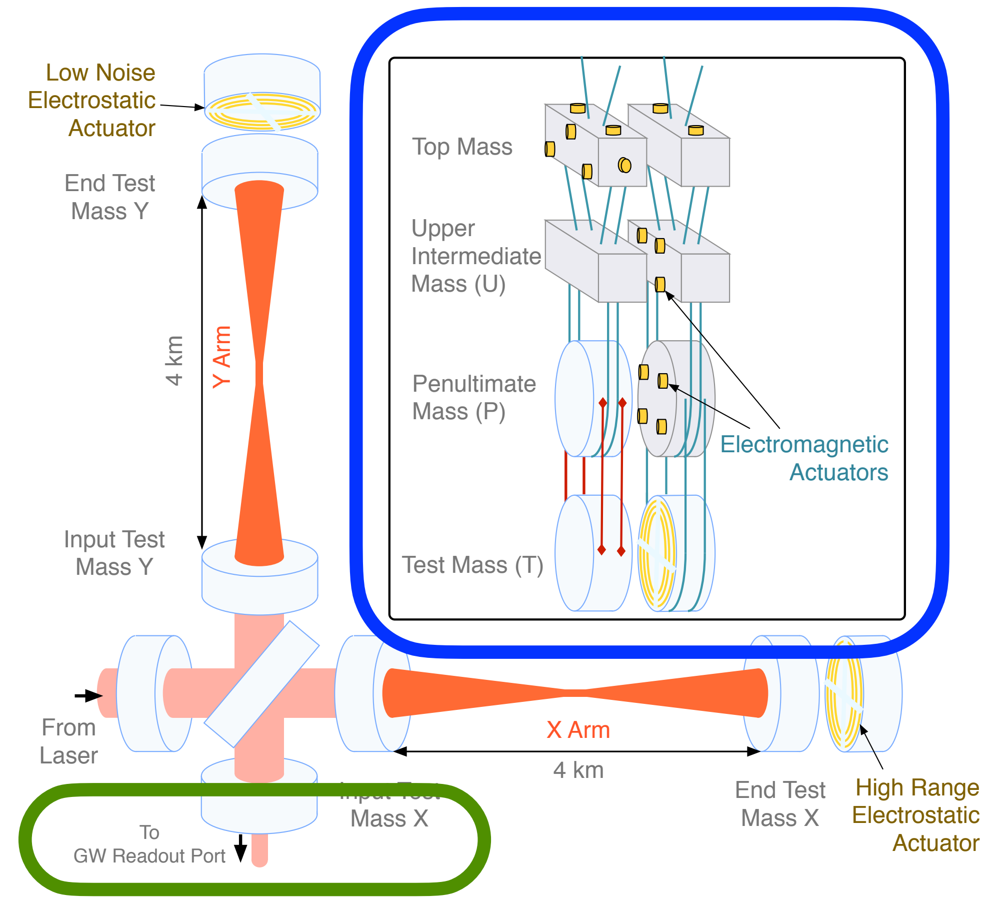
- Keeping the detector in the null range involves:
    - Lots of active seismic isolation (including quad. pendulum system)
    - AND control loop applied to readout signal

[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

The Differential Arm (DARM) control loop



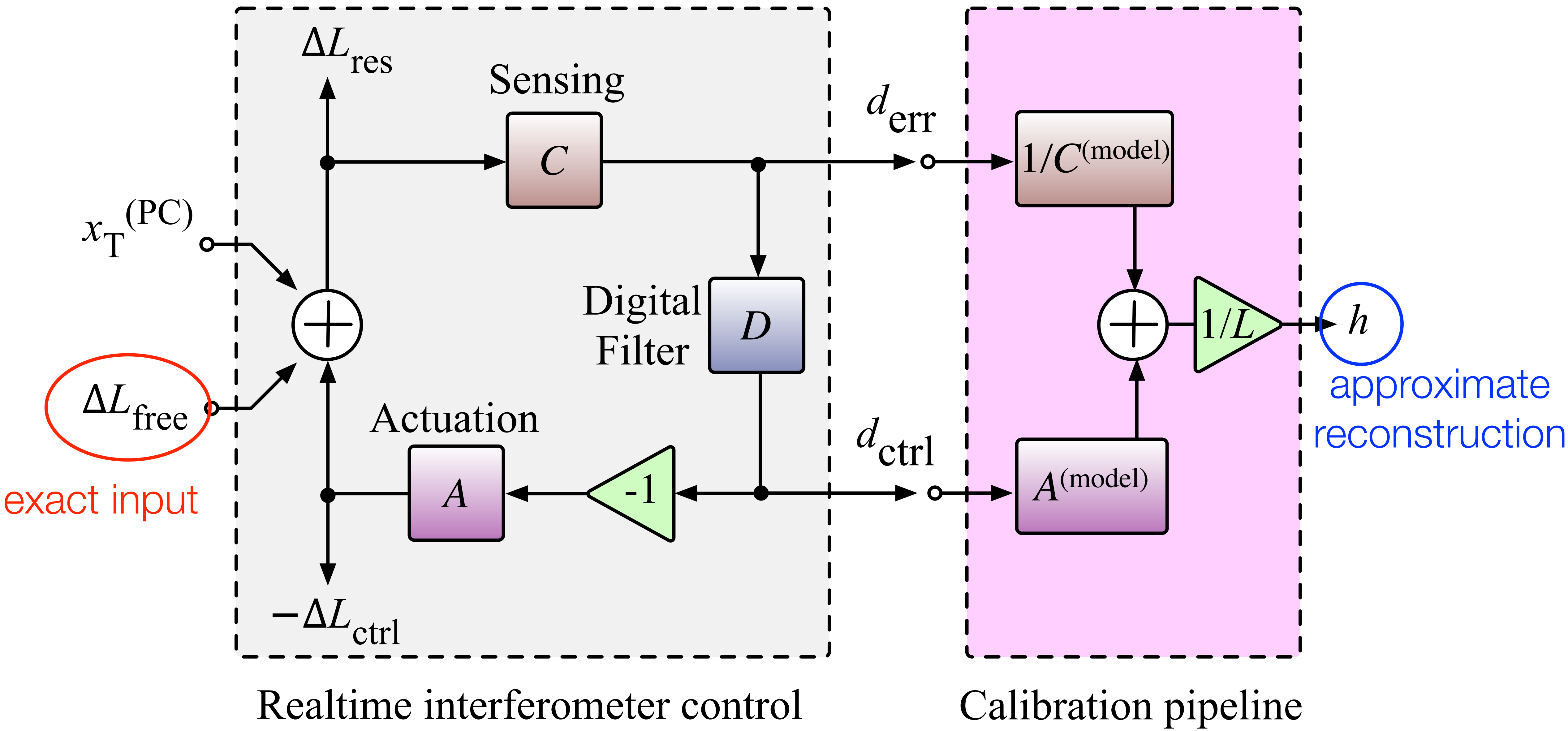
Realtime interferometer control



Digital filters include:
notch filters
low-pass filter

[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

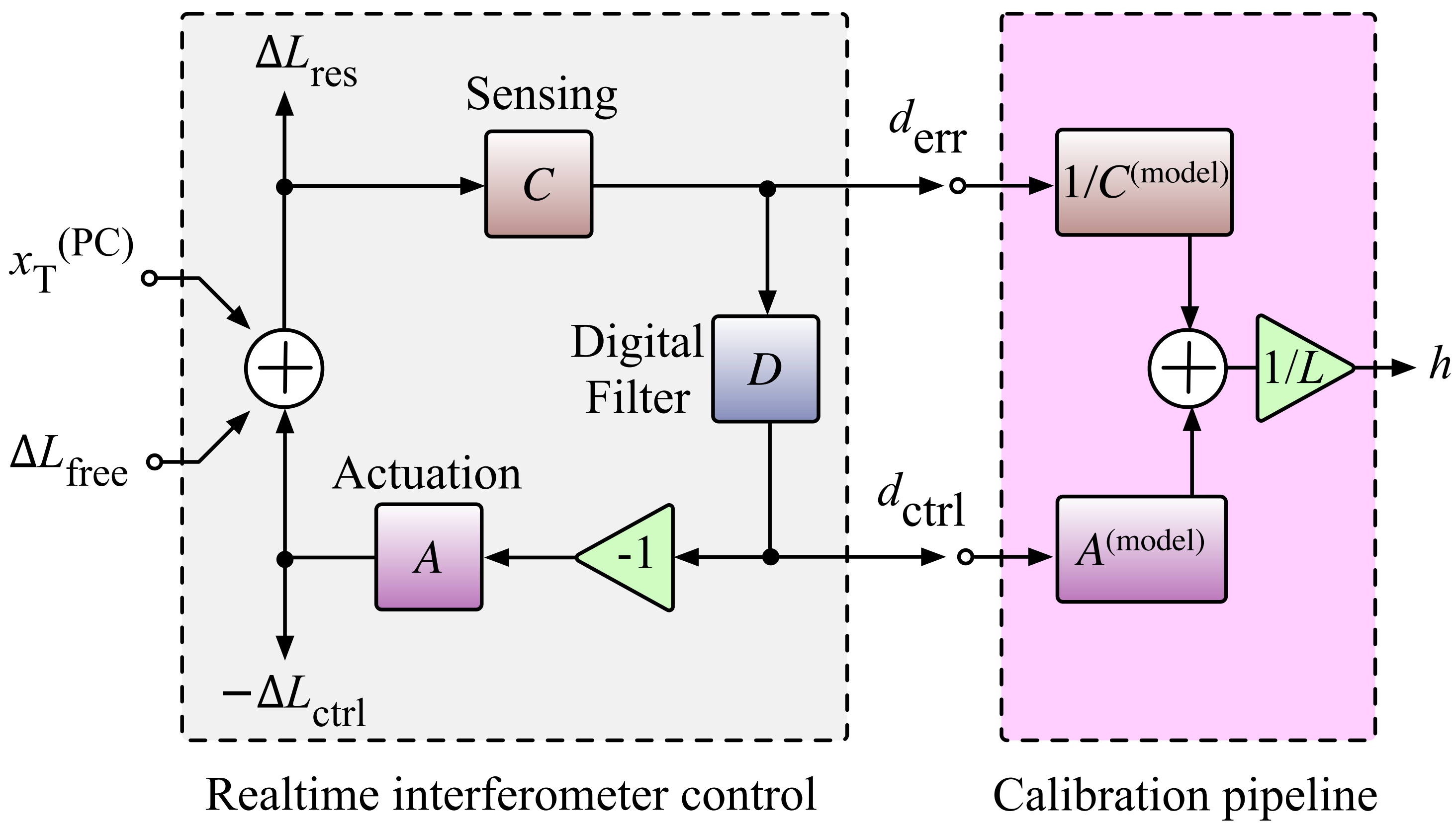
Reconstructing h(t) from the DARM loop



[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

Slide by M. Wade

Reconstructing h(t) from the DARM loop



$$h(t) = R * d_{\rm err}(t)$$

where

$$R = \frac{1}{L}\left(\frac{1+G}{C}\right)$$

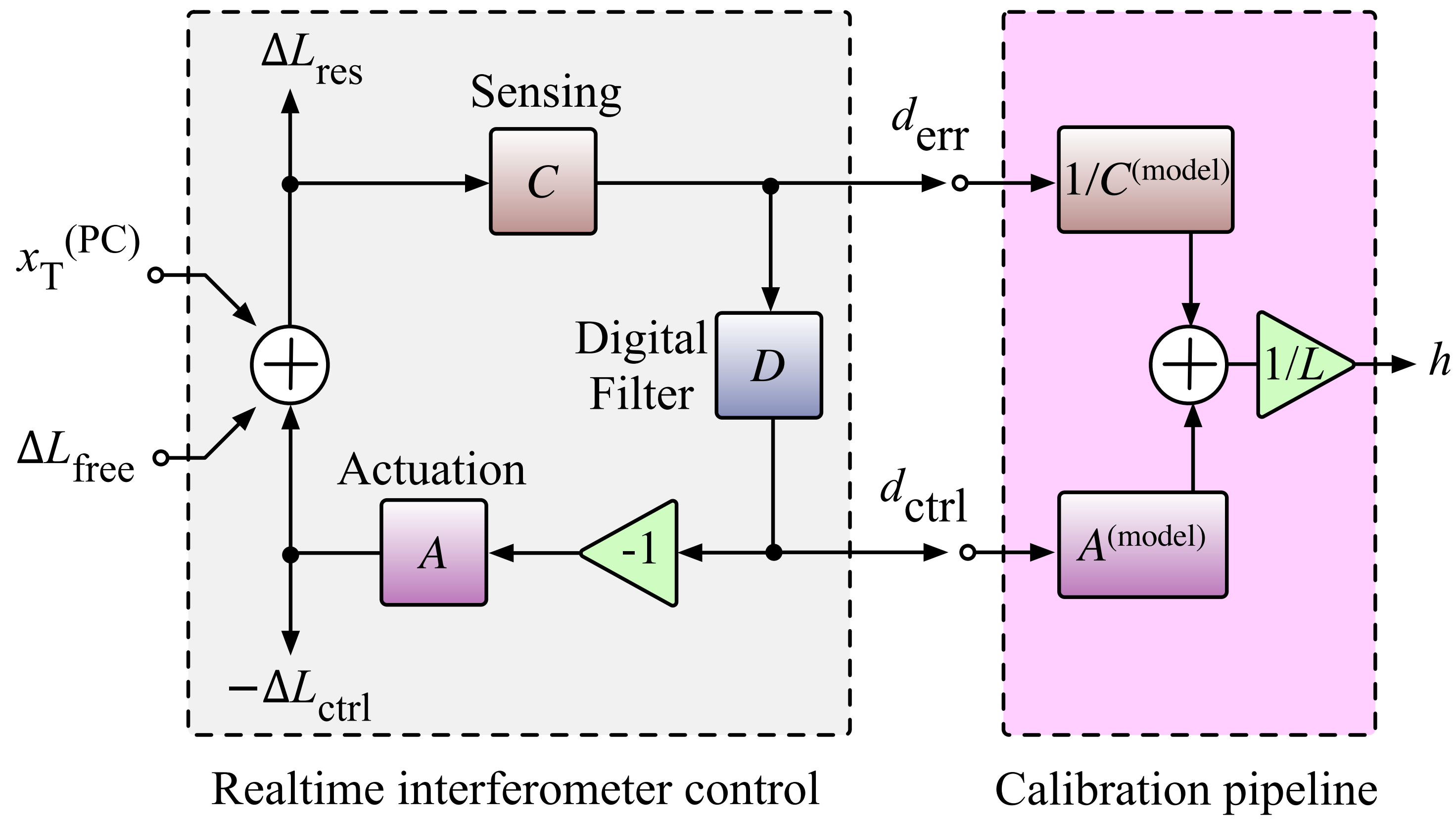
and

$$G = C\ D\ A$$

[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

Slide by M. Wade

Reconstructing h(t) from the DARM loop



$$h(t)\ L = \frac{1}{C} * d_{\mathrm{err}} + A * d_{\mathrm{ctrl}}$$

This method historically allowed us to bypass tracking the digital filters *D*

[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

Sensing function

$$C(f,t) = \kappa_C(t) H_C \left(\frac{1}{1 + if/f_{cc}(t)} \right) \left(\frac{f^2}{f^2 + (f_s(t))^2 - iff_s(t)/Q_s(t)} \right) C_R(f) e^{-2\pi i f \tau_C}$$

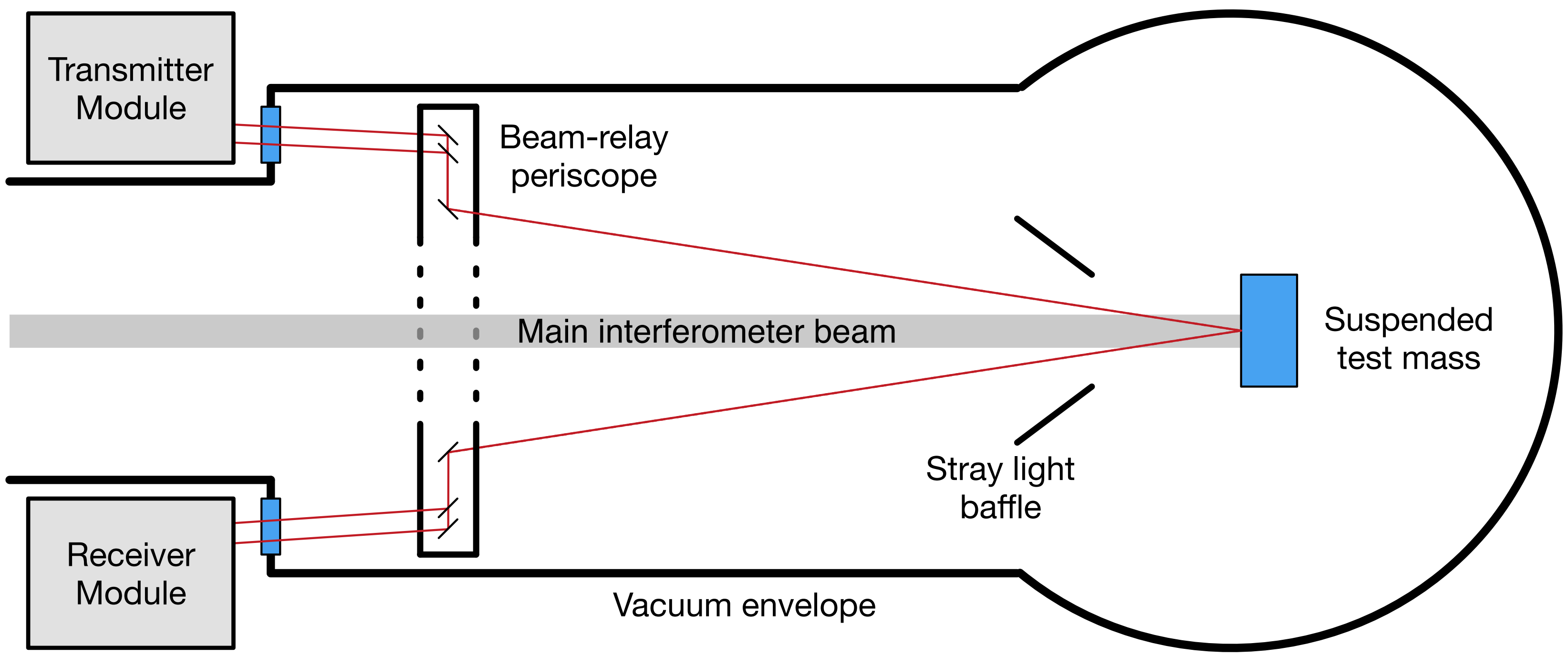
Overall gain — Fabry-Perot cavity response — Optical anti-spring from signal recycling cavity — Electronics response and time delay

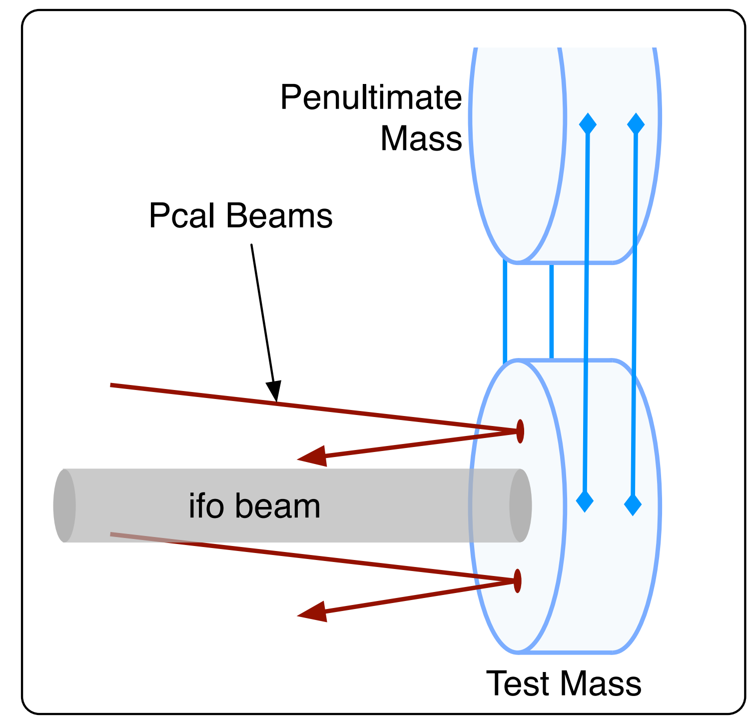
- Blue color indicates parameters that change in time
- Time-dependent parameters are tracked using narrow-band sinusoidal injections

[7] Tuyenbayev et al., Class. Quantum Grav. Vol. 34 Num. 1 (2016)

Slide by M. Wade

Absolute calibration: The photon calibrator





Modulated auxiliary laser (photon calibrator) is used to set absolute scale factor for calibration

[6] Karki et al. Review of Scientific Instruments 87, 114503 (2016)

Actuation function

$$A(f,t) = \left[\kappa_U(t)A_U(f) + \kappa_P(t)A_P(f) + \kappa_T(t)A_T(f)\right] e^{-2\pi i f \tau_A}$$

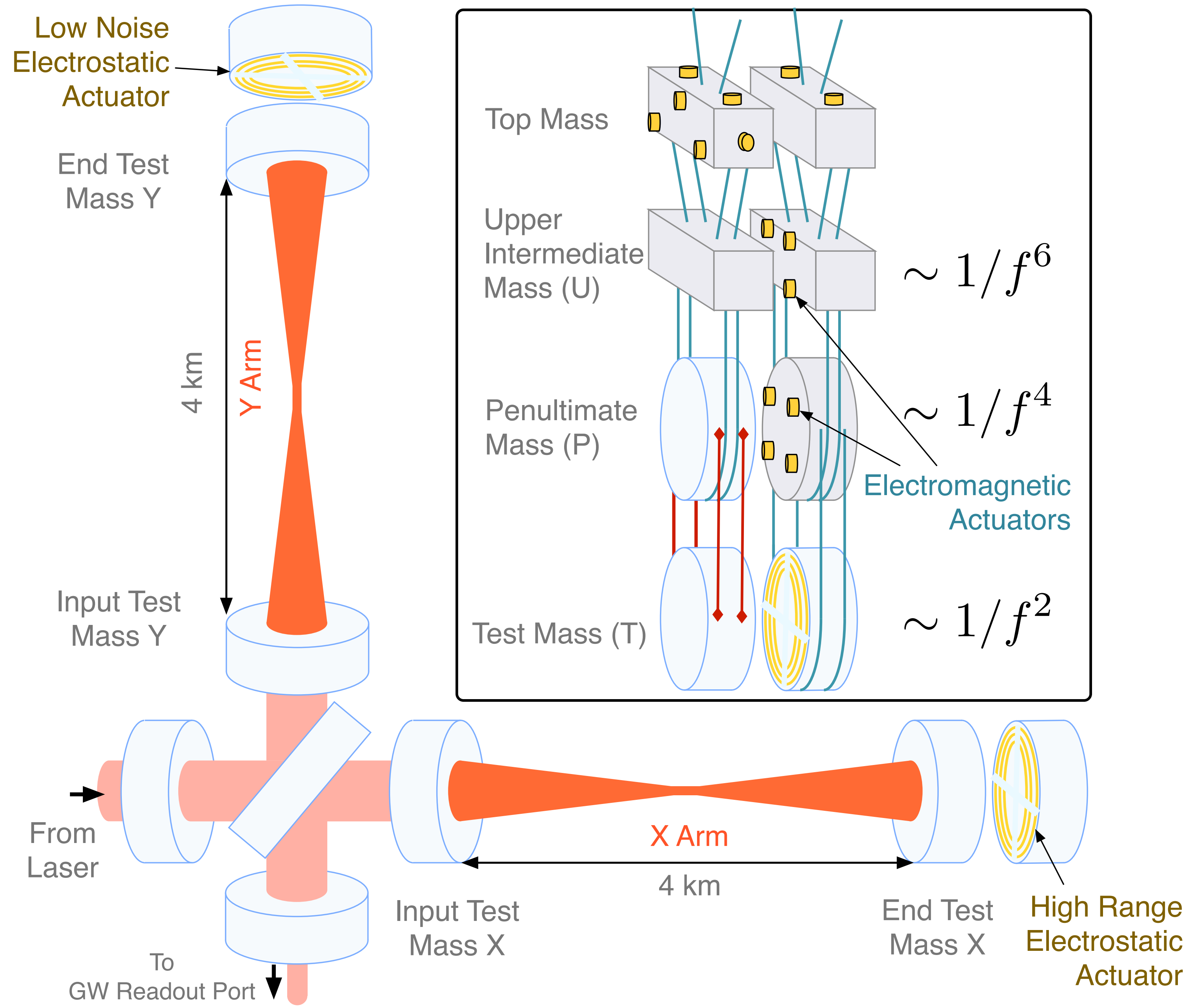
- $\kappa_U(t)A_U(f)$: Upper intermediate mass actuation response
- $\kappa_P(t)A_P(f)$: Penultimate mass actuation response
- $\kappa_T(t)A_T(f)$: Test mass actuation response
- $e^{-2\pi i f \tau_A}$: Time delay

- Blue color indicates parameters that change in time
- Time-varying gains of each stage are currently compensated for in calibration software

[7] Tuyenbayev et al., Class. Quantum Grav. Vol. 34 Num. 1 (2016)

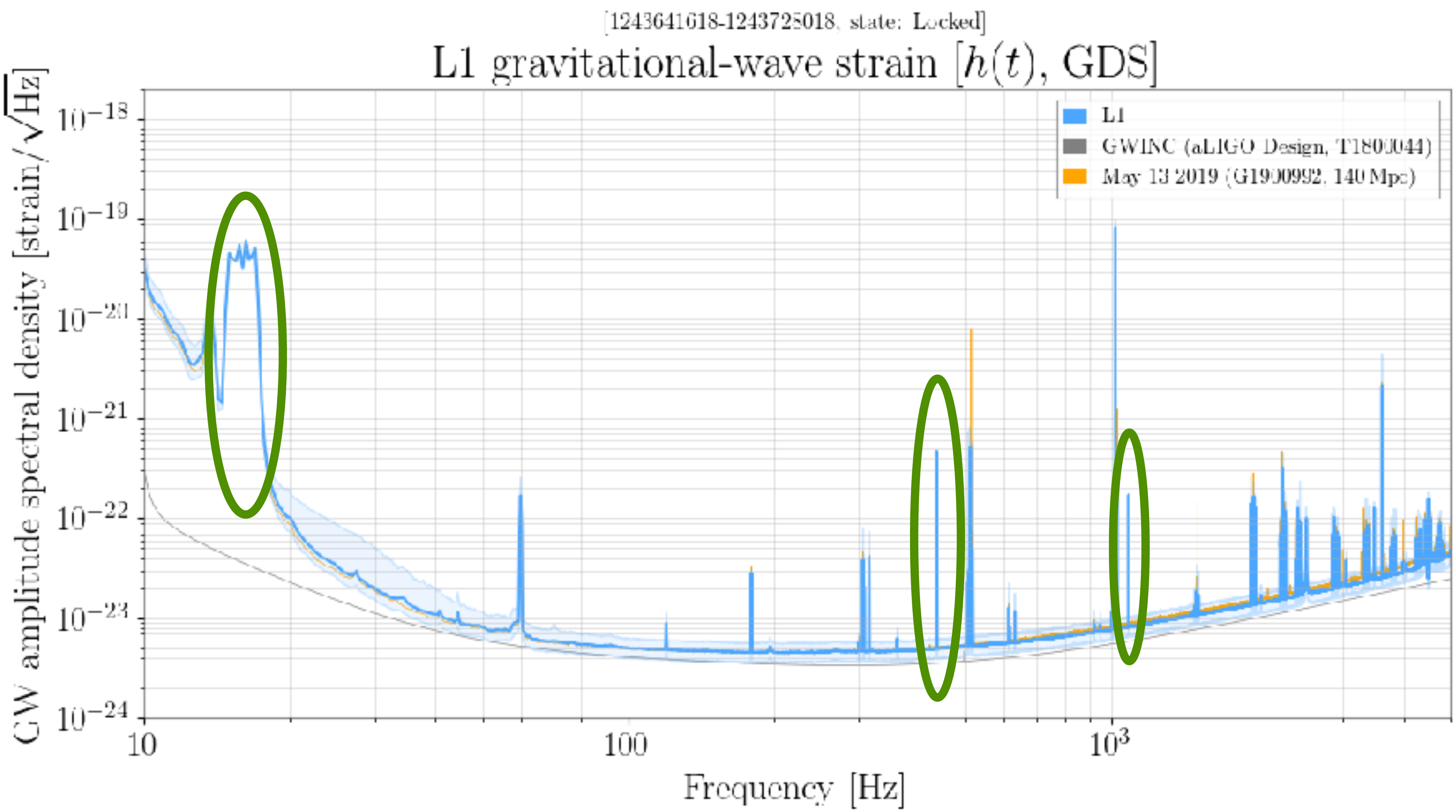
Actuation function

- Mechanical response of each stage of pendulum



[3] Cahillane et al., Phys. Rev. D 96, 102001 (2017)

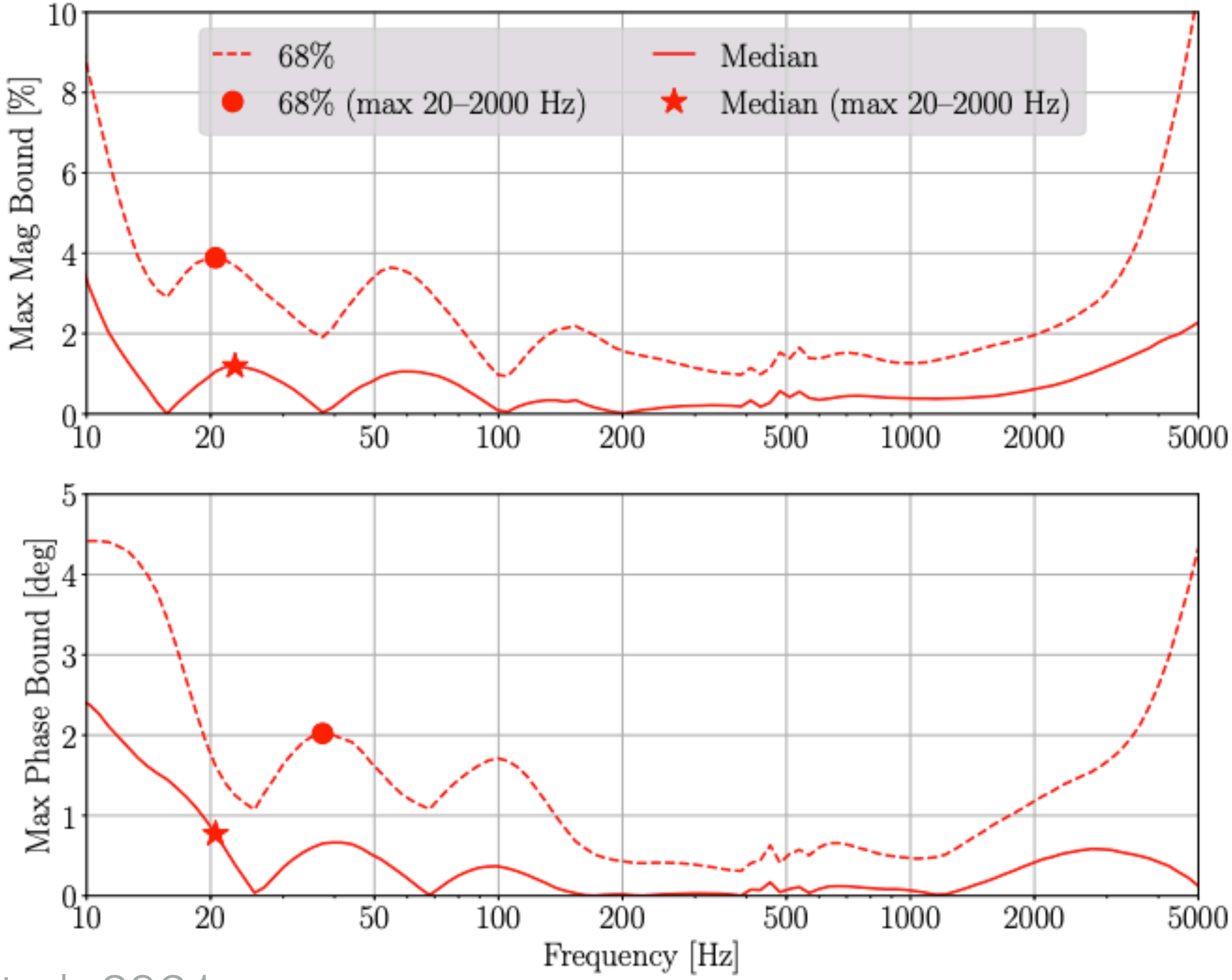
“Calibration lines”



Slide by M. Wade

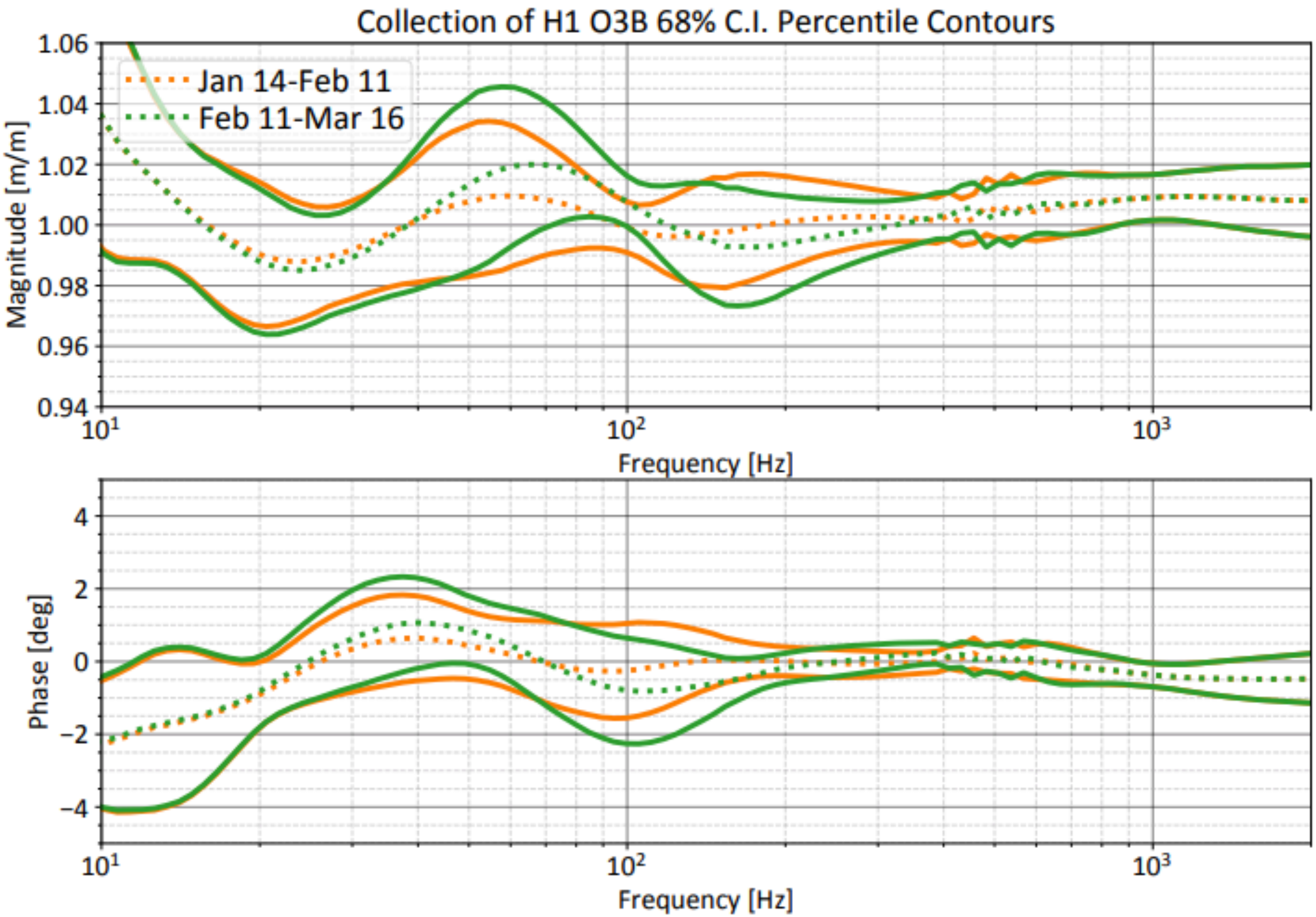
Calibration uncertainty

LIGO Hanford O3a calibration uncertainty vs. frequency



Betzweiser et al. 2021

Calibration uncertainty evolves over time!



Example: change in electronics cabling in TST actuation function on February 11, 2020

Betzweiser et al. 2021

Useful resources

GWpy examples: https://gwpy.github.io/docs/latest/examples/index.html

Virgo DetChar paper: https://arxiv.org/abs/1203.5613
O2-O3 LIGO DetChar paper: https://arxiv.org/abs/2101.11673
LIGO Calibration uncertainty: https://arxiv.org/abs/2005.02531
LSC Calibration bootcamp: https://dcc.ligo.org/LIGO-G2100709

The Gravitational Wave Open Science Center (GWOSC): https://www.gw-openscience.org/
GWOSC tutorials: https://www.gw-openscience.org/tutorials/

Public interferometer status monitoring: https://www.gw-openscience.org/detector_status/

LIGO-Virgo alerts guide: https://www.gw-openscience.org/alerts/

Gravity Spy: gravityspy.org

Extra slides:

PSD estimation

Intro to GWpy digital filters

Estimating the PSD

Step 0: Take a **Fast Fourier Transform (FFT)**, which is any algorithm useful for quickly estimating the **Discrete Fourier Transform** that describes a **discrete time series**.

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$
$$= \sum_{n=0}^{N-1} x_n \cdot [\cos(2\pi kn/N) - i \cdot \sin(2\pi kn/N)],$$

Need to shift our thinking to discretized data; **frequency bins instead of continuous smooth sinusoids**

Estimating the PSD

Step 1: Apply a window to your data (if it's linear! as a time series is) to prevent **spectral leakage** from the assumption the signal is periodic.

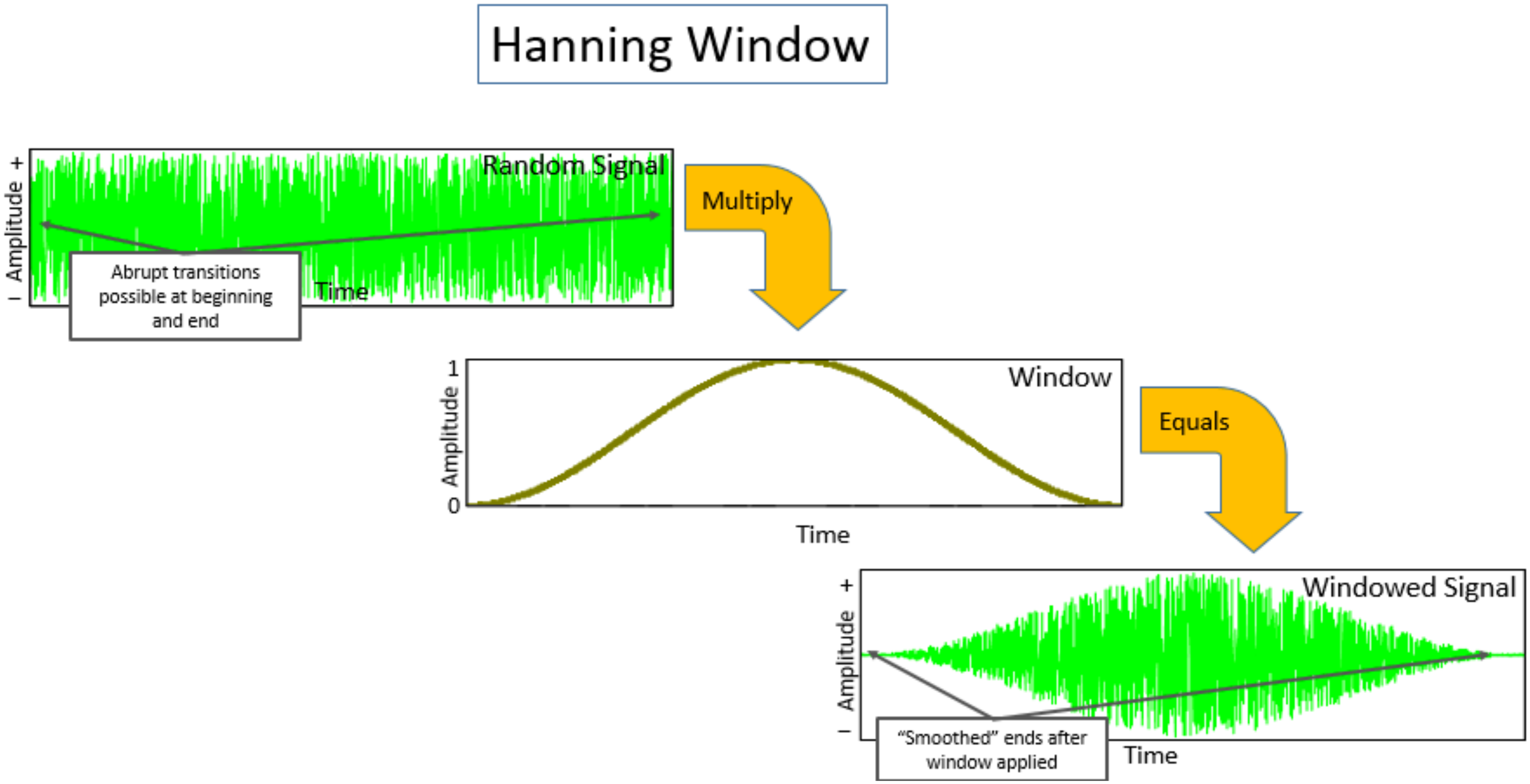


Image: Siemens.

Estimating the PSD

A single windowed FFT is unbiased (i.e. will give the correct mean PSD), but has **high variance**.

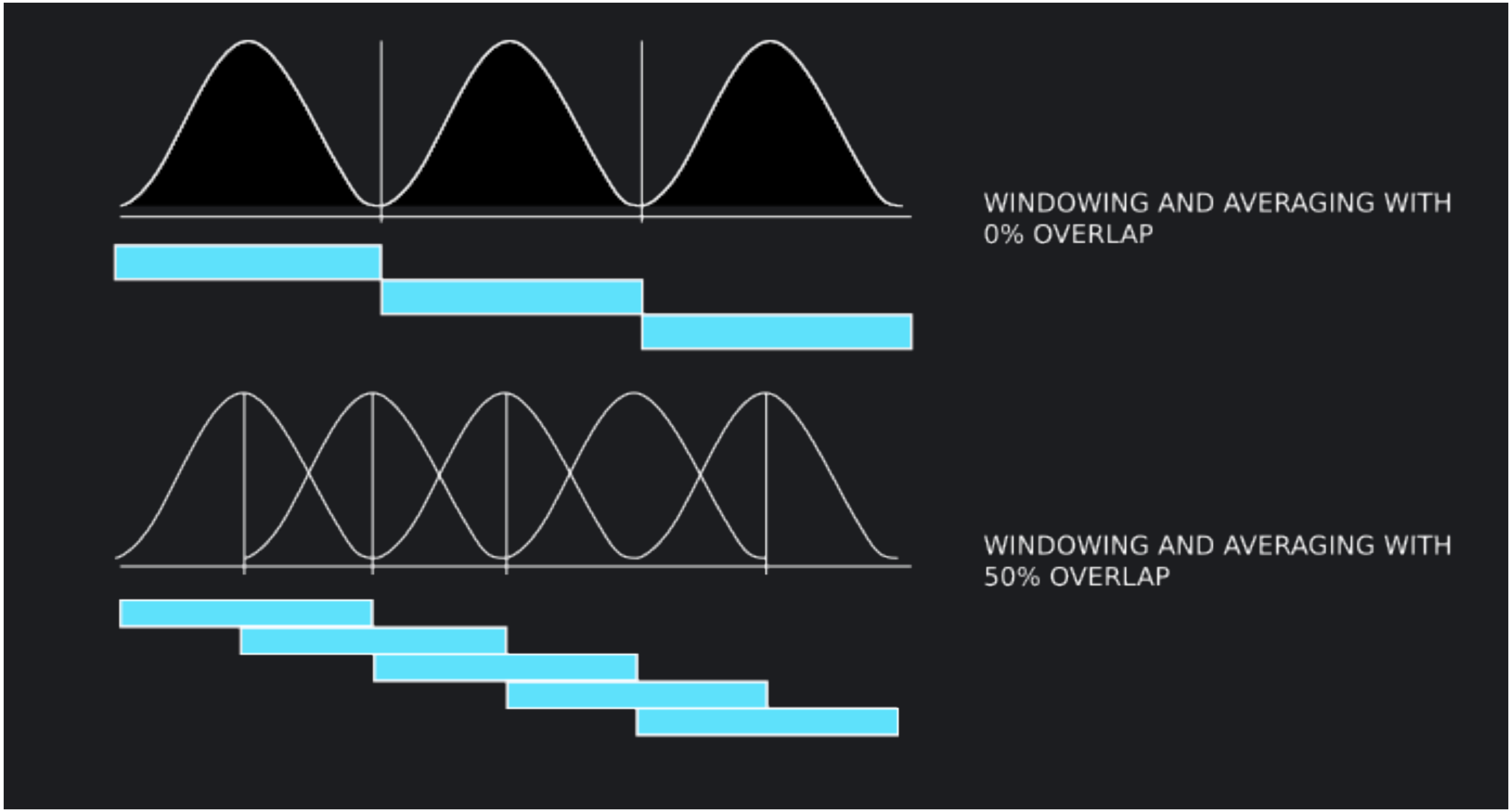
Solution: average several FFTs!

Step 2: Divide your data into shorter time segments; take a windowed FFT of each, and average these together.

Note you lose some frequency resolution this way.

Welch's method averages the mean value for each frequency bin across FFTs, with some overlap in the data analyzed.

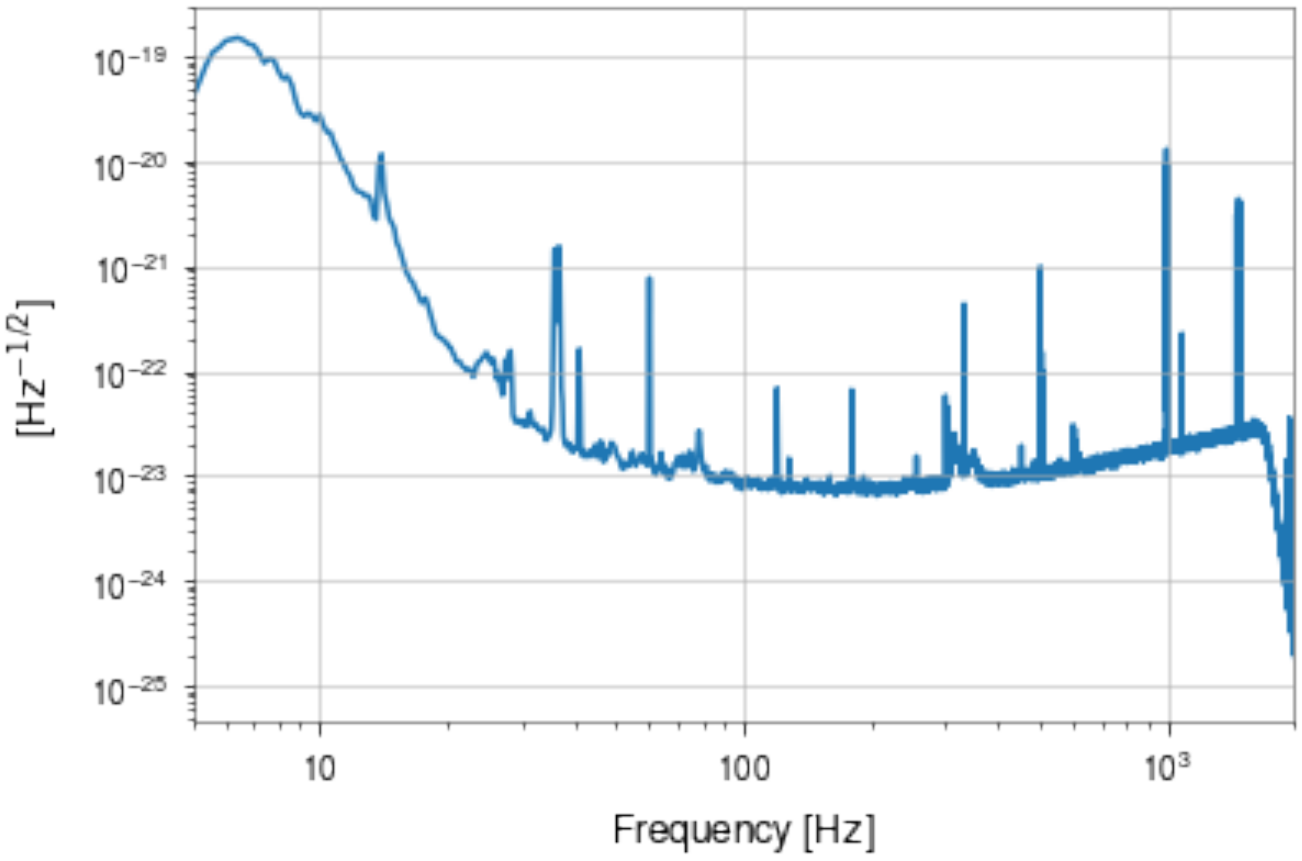
Averaging FFTs



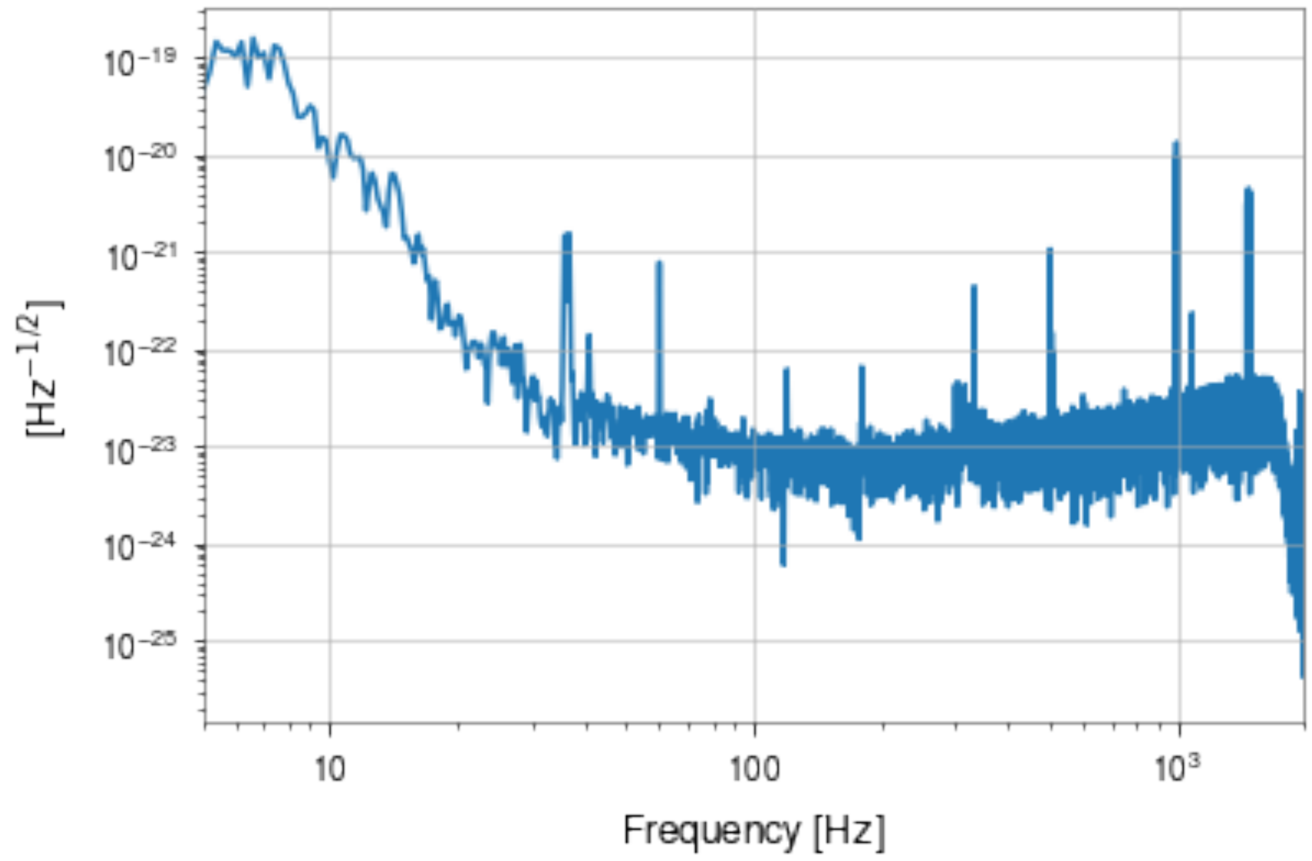
https://www.dewesoft.com/pro/course/spectral-analysis-using-the-fft-29?page=10

Example: Averaging FFTs

FFT length = 5 seconds
Overlap = 2 seconds
120 averages

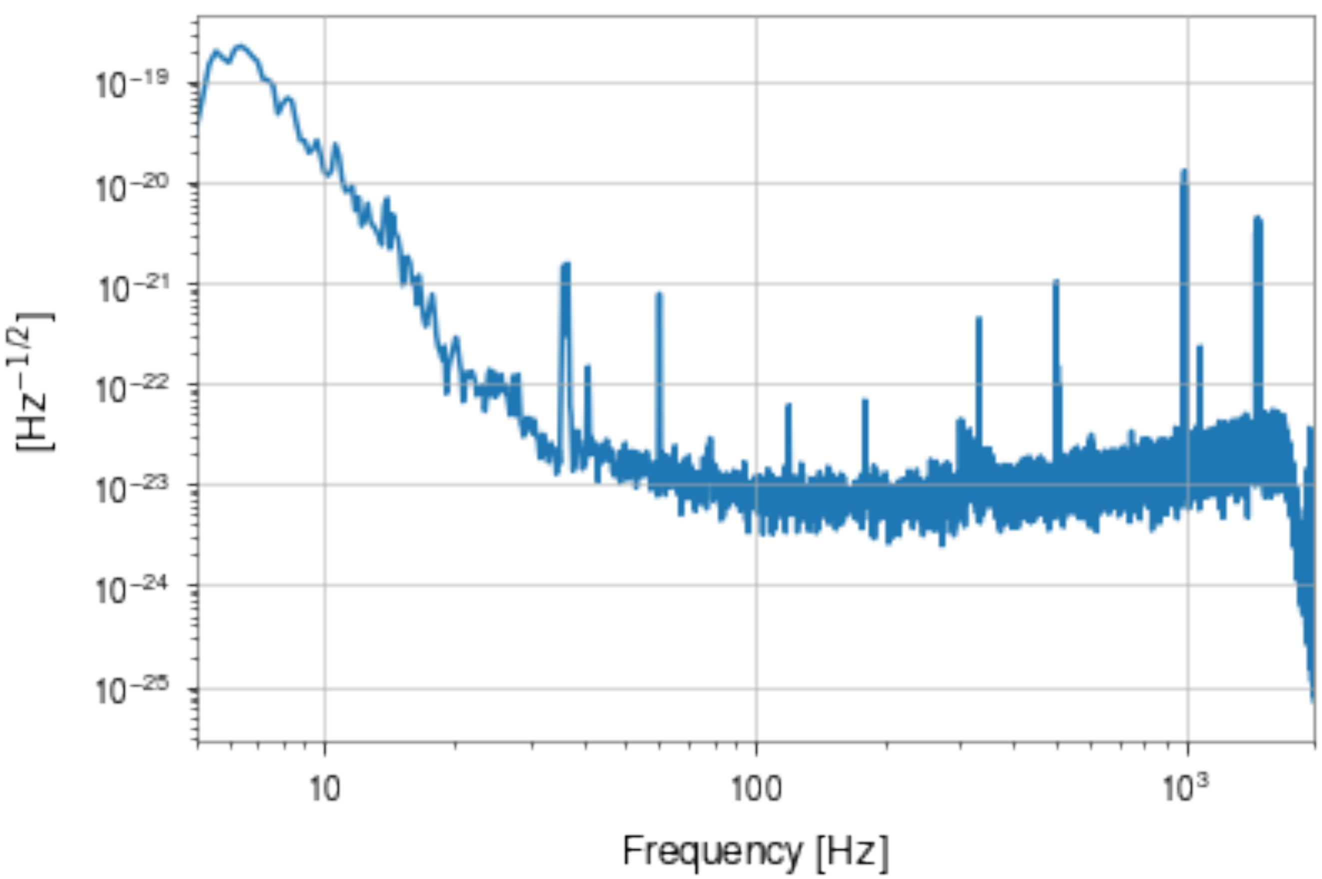


FFT length = 5 seconds
Overlap = 2 seconds
4 averages

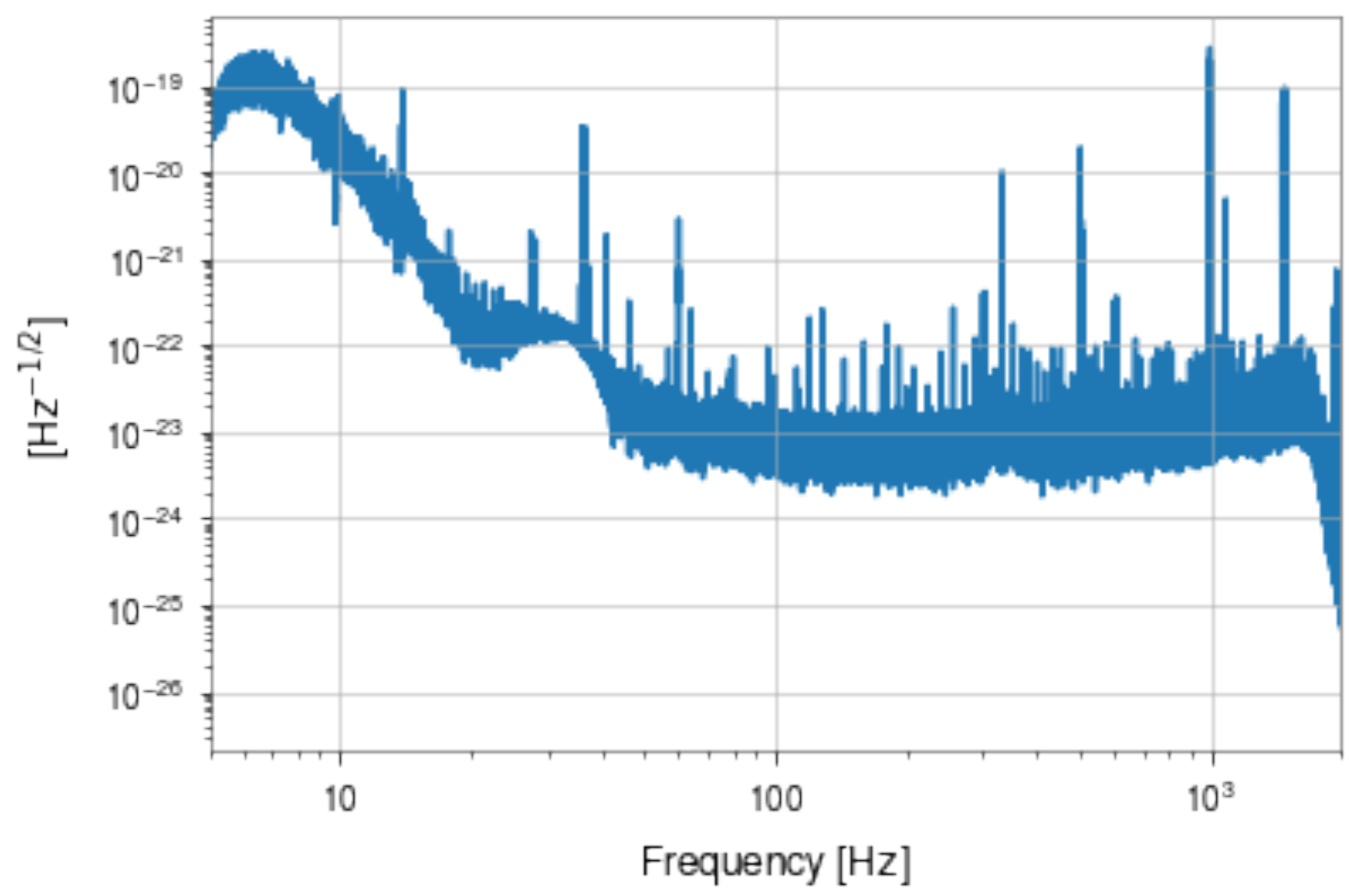


Example: FFT length

FFT length = 5 seconds
Overlap = 2 seconds
6 averages



FFT length = 2048 seconds
Overlap = 1024 seconds
6 averages

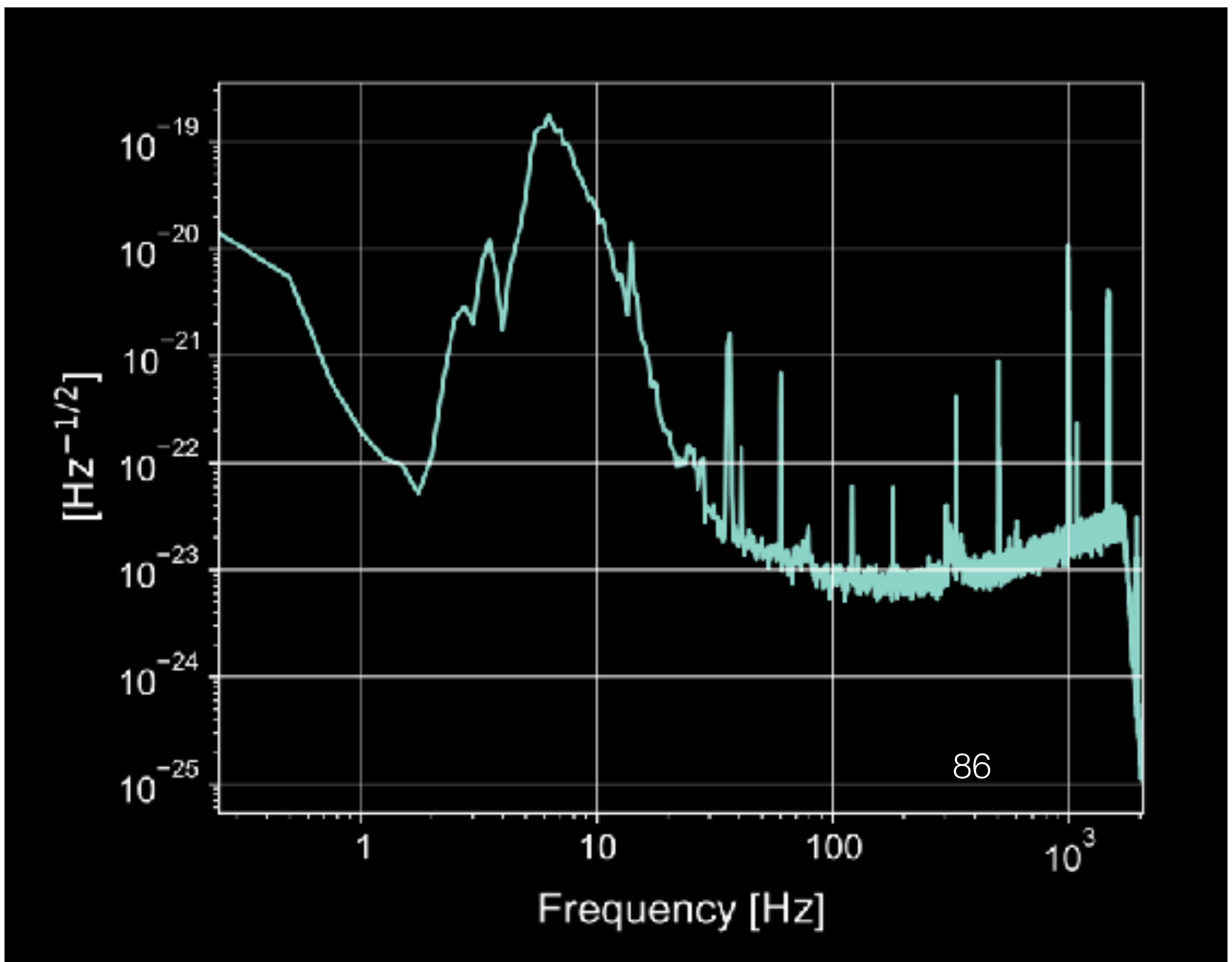


Signal processing with GWpy

GWpy provides **FFT wrappers** to estimate frequency-domain content of data:

FFT length (s) Overlap between averages (s)

```
>>> asd = data.asd(4, 2)
```



Can also specify:

Time window
(default = Hanning)

Averaging method
(default = Welch)

Signal processing with GWpy

GWpy provides simple signal-processing methods to filter data

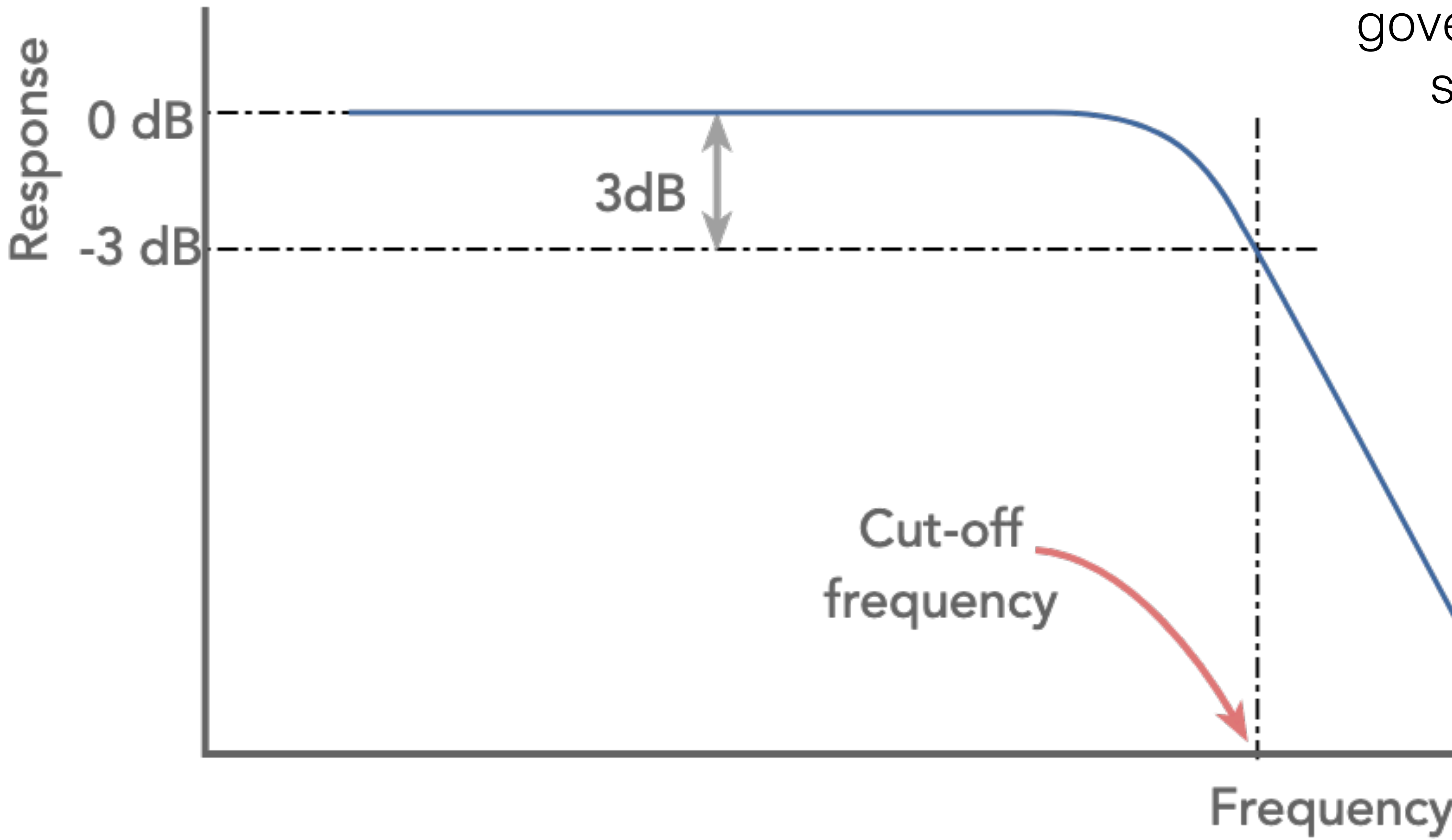
- lowpass, highpass
- bandpass
- notch
- whitening

You will use these tools to design frequency domain filters later!

High pass and low pass filters

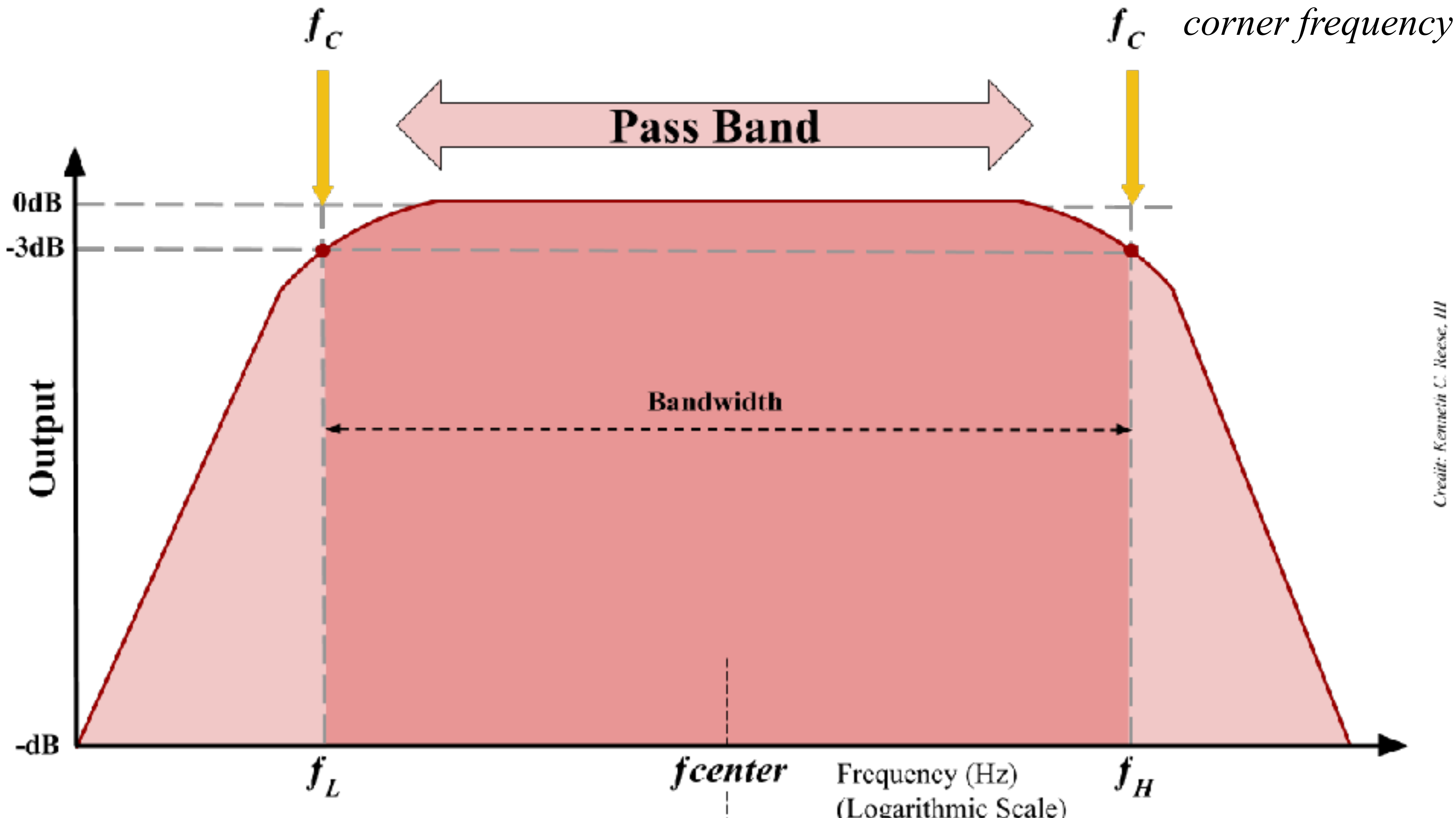
A "low pass" filter

Filter "order" governs the slope



Credit: electronics-notes

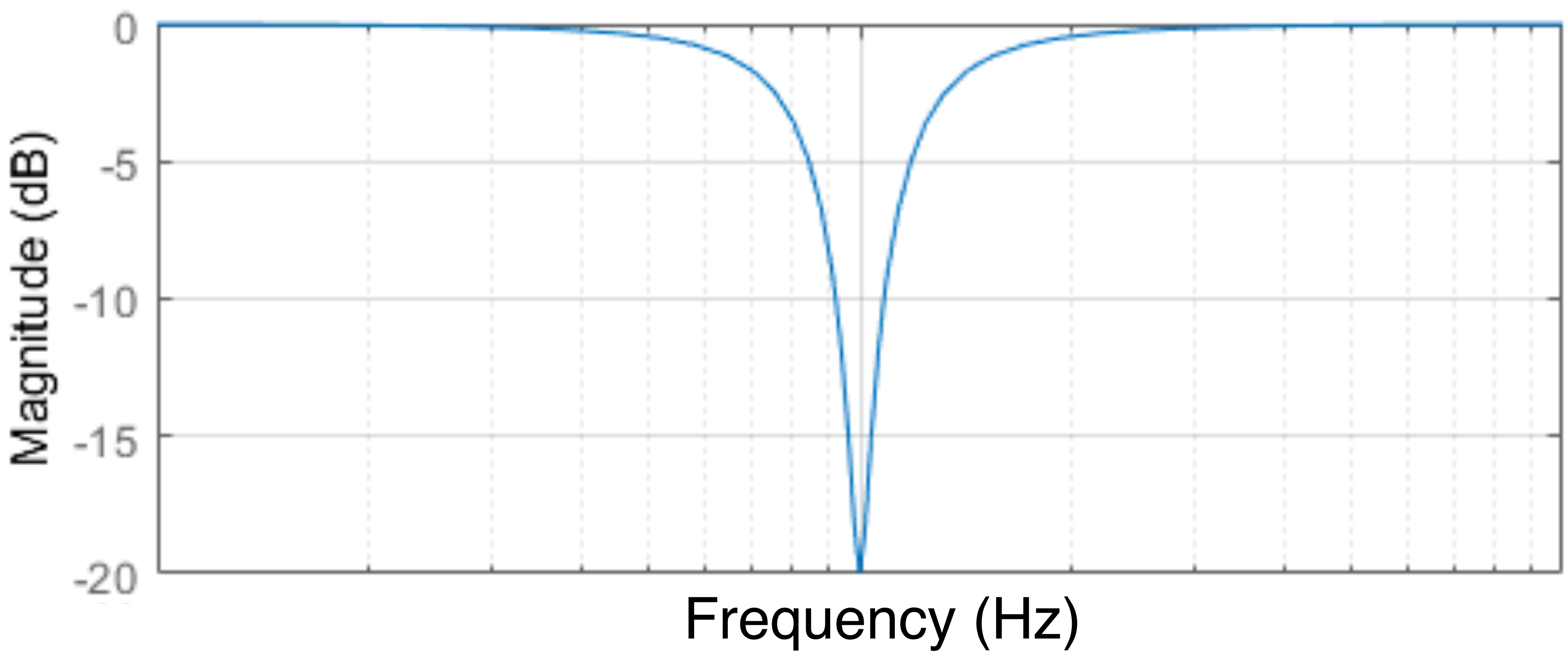
Bandpass filter



Notch filter

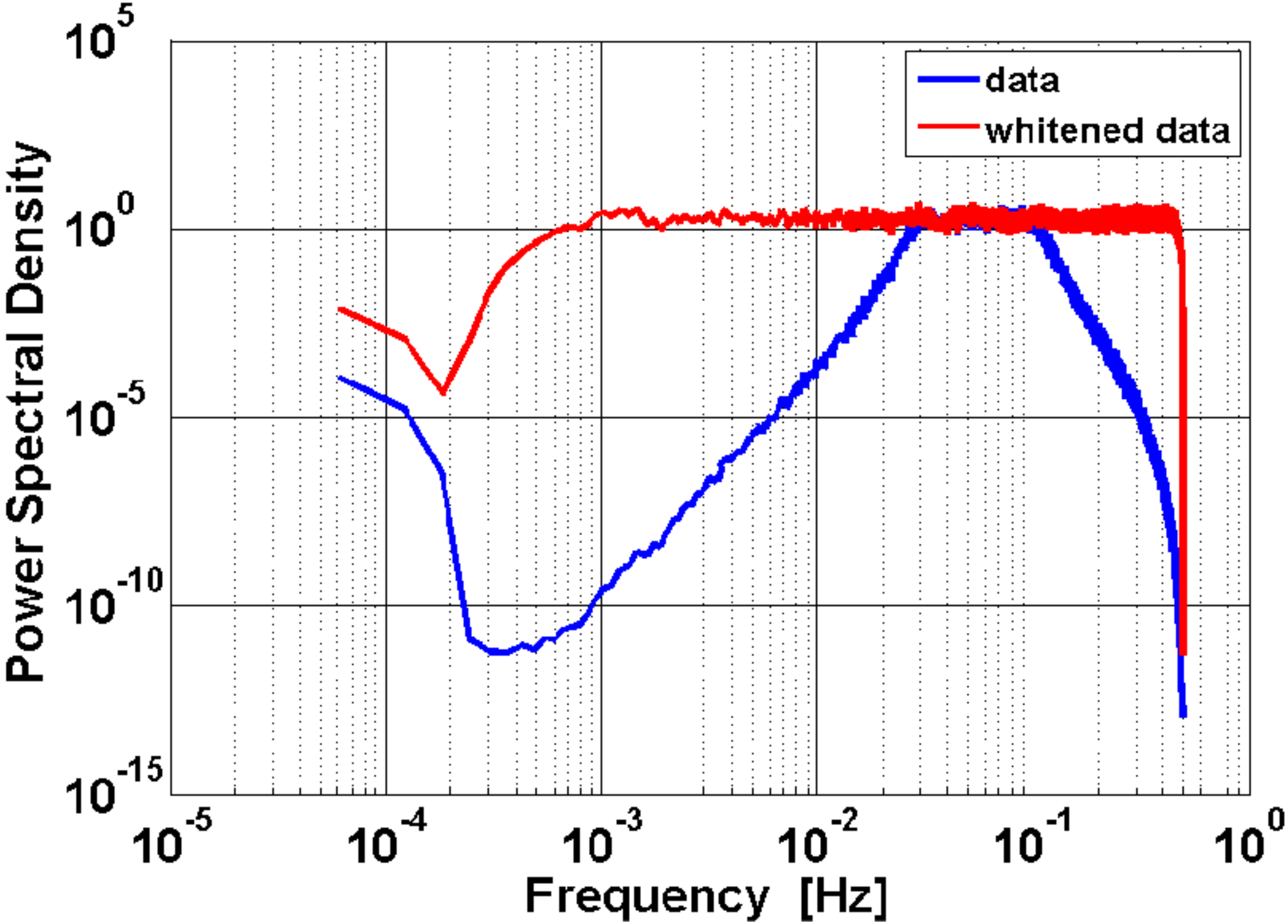
Not quite the inverse of the bandpass filter

Only described by one frequency (and the filter order)



Credit: electronics-notes

Whitening



Credit: LISA data tools