

Noise Subtraction in Gravitational-Wave Detectors

Gabriele Vajente

LIGO Laboratory – Caltech





Gabriele Vajente



Scientist, Engineer and Educator | Leading R&D and Detector Improvements for gravitational-wave detection | Communicating science to the public

About

I'm a physicist, engineer and educator. I manage the LIGO Laboratory research and development for new technologies that allow us to reach further into the universe with gravitational waves.

I've been involved for two decades in the design, construction and operations of two of the large-scale gravitational-wave detectors in the world, LIGO and Virgo.

My main expertise is in instrumentation and experimental physics, and in management of large research infrastructures and teams, with a focus on R&D and instrumentation development.

I am passionate about communicating science, and enabling all people from all background to understand the importance of science and to pursue a career in science and engineering.

Looking for new opportunities to work on future technologies with applications in fundamental physics, astronomy and engineering. All while ensuring that everyone can benefit from our progress and we all have a better life.

www.linkedin.com/in/gabriele-vajente

Gabriele Vajente
*Deputy Head of Systems
Science and Engineering*

LIGO Laboratory – Caltech
Pasadena, CA (USA)

vajente@caltech.edu

What kind of noise?

Focus on removing noise from the main detector output

Focus on persistent noise (even though not necessarily stationary)

We won't talk about glitches

What techniques do we have to discover and subtract noise?

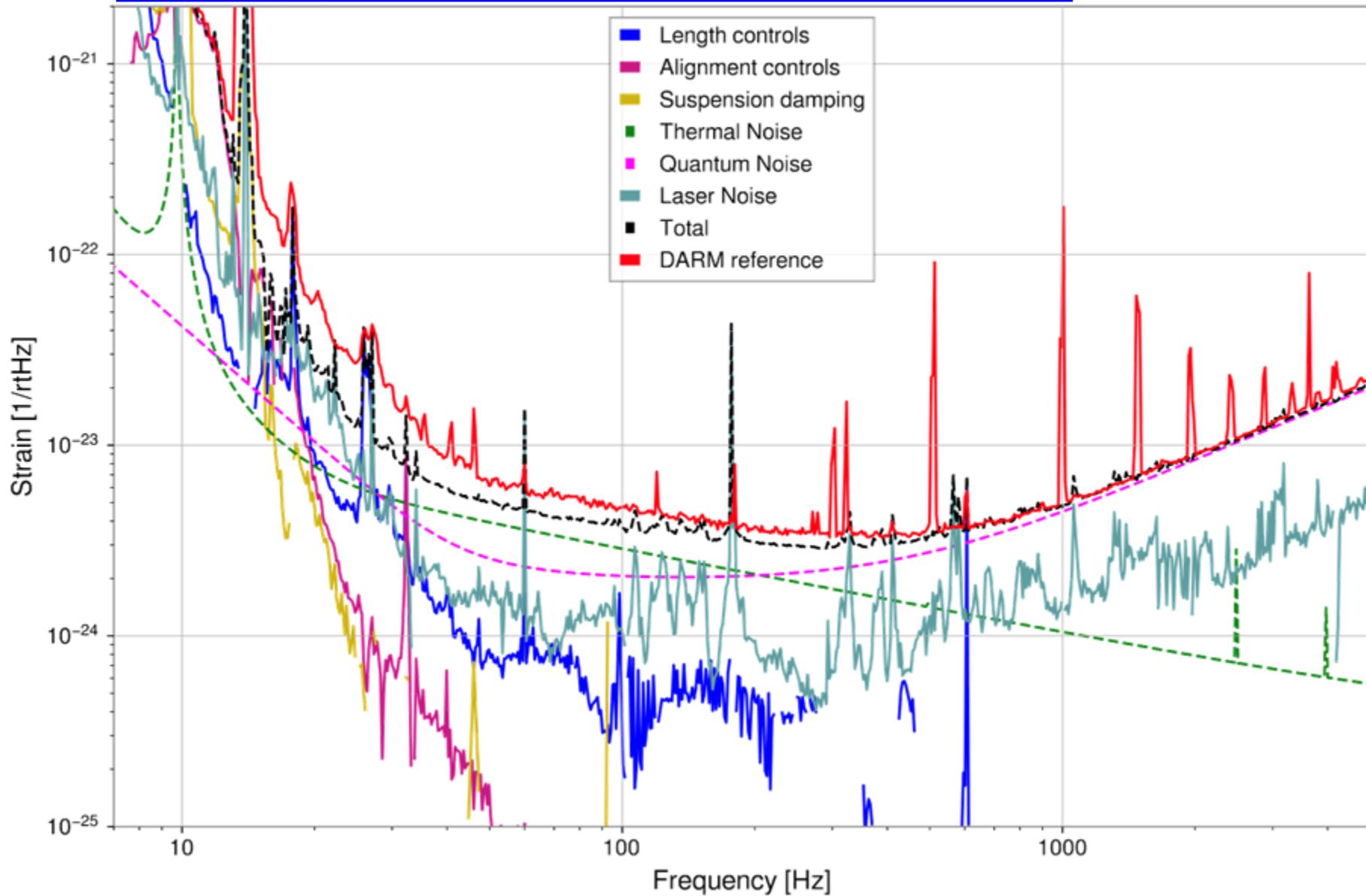
Discover

analyze all recorded data and find noise sources and couplings

Subtract

devise a way to remove the noise from the main detector output

<https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=86555>



Scattered light

Unexpected
noise sources

Acoustic vibrations

Frequency noise

Not understood
noise couplings

Laser beam jitter

Scattered light

Non-linear noise
couplings

Coupling modulation

Can't measure beam everywhere

Can't put seismometers everywhere

Not enough
monitoring channels

Too many monitoring
channels

4,000+ fast channels

200,000+ slow channels

28 MB/s of potentially interesting data

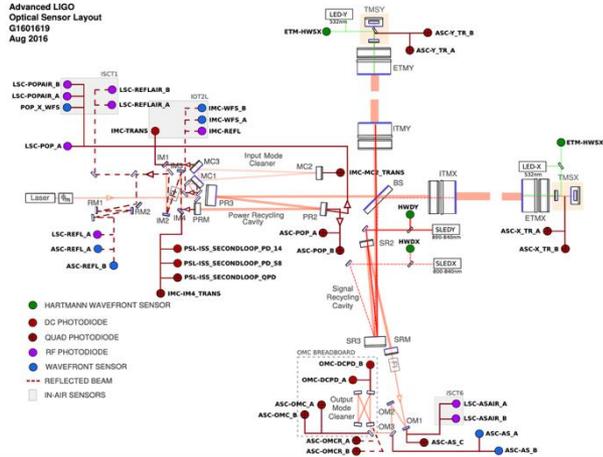
Non-linear readout and drive

Noise characterization and subtraction

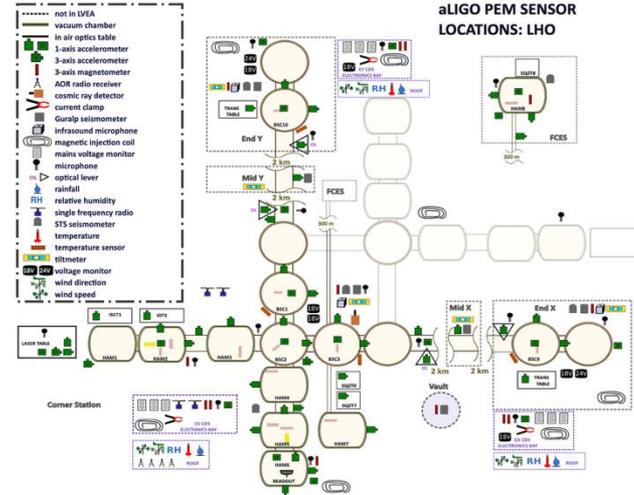
Goals

1. **Find** as many noise sources as possible, as witnessed by auxiliary channels
2. **Characterize** the way the noise couples to the strain channel
3. Devise a strategy to **subtract** the noise
4. Possibly **understand** the physical origin of the noise or noise coupling and engineer it out

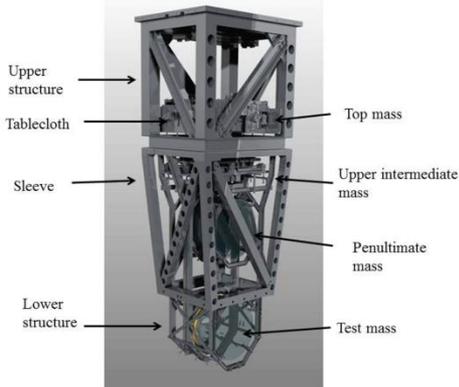
Interferometer Sensing and Control (ISC)



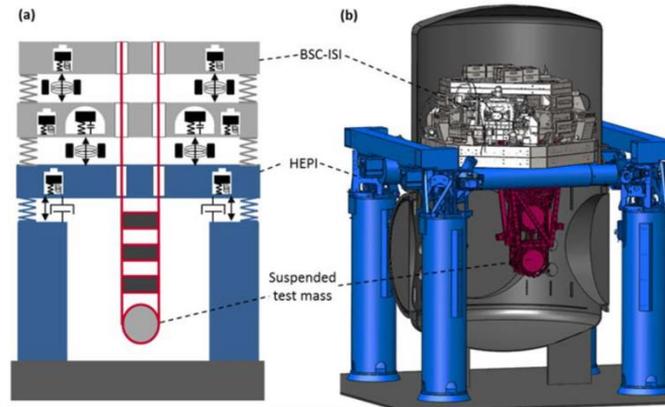
Physical Environment Monitoring (PEM)



Suspension Control (SUS)



Seismic Isolation Systems (SEI)



And a lot more

- 4,000+ fast channels
- 200,000+ slow channels
- 28 MB/s of potentially interesting data

$$h(t) = \mathcal{F}[x_1(t), x_2(t), \dots]$$

Detector output
(GW strain)

Auxiliary witness channels

Any general functional
coupling function

Linear coupling

$$h(t) = \int_{-\infty}^0 x(t - \tau) f(\tau) d\tau$$

Linear, causal and time invariant

Non-Linear coupling



Time domain

Frequency domain

$$h(t) = \int_{-\infty}^0 x(t - \tau) f(\tau) d\tau \quad \longleftrightarrow \quad \tilde{h}(\omega) = \tilde{f}(\omega) \cdot \tilde{x}(\omega)$$

Transfer function \nearrow

Transfer function can be estimated from sampled signals, using ratios of power and cross spectral densities.

$$\tilde{f}(\omega) \sim \frac{S_{h,x}^*(\omega)}{S_{x,x}(\omega)}$$

Coherence: frequency-domain equivalent of frequency-by-frequency correlation

Between 0 and 1:

0 = no linear correlation at that frequency,

1 = maximal linear correlation

$$C(\omega) \sim \frac{|S_{h,x}^*(\omega)|^2}{S_{h,h}(\omega) S_{x,x}(\omega)}$$

```
import scipy.signal

def tfe(x, y, *args, **kwargs):
    """
    tfe(x, y, *args, **kwargs)

    Estimate transfer function from x to y,
    see scipy.signals.csd for calling convention

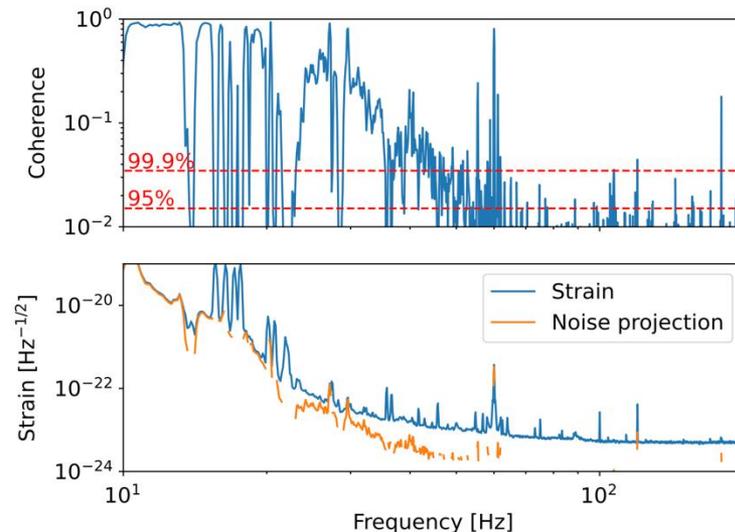
    Returns tf, fr, cohe
    """
    fr,cxy = scipy.signal.csd(x, y, *args, **kwargs)
    fr,px = scipy.signal.welch(x, *args, **kwargs)
    fr,py = scipy.signal.welch(y, *args, **kwargs)
    return cxy / px, fr, abs(cxy)**2/px/py
```

- **Coherence** is the tool of the trade
- **Scalable** to large number of channels for discovery of noise couplings
- **Limited** by availability of witness channels
- **Routinely used** in LIGO: can scan all fast channels for coherence with strain in matter of tens of minutes
- Allowed identification of many noise sources

Top 20 coherences of DCS-CALIB_STRAIN with auxiliary channels

GPS 1242441180 + 1200 s [2019/05/21 02:32:42 UTC]

Frequency [Hz]	Top channels									
9.12	OAF-CAL_XARM _DQ (0.40)	CAL-CS_XARM _ERR_DQ (0.40)	CAL-CS_XARM _ERR_DQ (0.40)	LSC_XARM_IN1 _DQ (0.40)	LSC_POP_A_RF45 _T_ERR _DQ (0.40)	LSC_POP_A_RF45 _T_ERR _DQ (0.40)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.38)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.38)	LSC_SRCLFF1 _OUT_DQ (0.36)	CAL-CS_SRCL _ERR_DQ (0.36)
9.25	LSC_XARM_IN1 _DQ (0.29)	LSC_POP_A_RF45 _T_ERR _DQ (0.29)	CAL-CS_XARM _ERR_DQ (0.29)	CAL-CS_XARM _ERR_DQ (0.29)	OAF-CAL_XARM _DQ (0.29)	LSC_POP_A_RF45 _T_ERR _DQ (0.29)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.29)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.27)	CAL-CS_SRCL _ERR_DQ (0.27)	CAL-CS_SRCL _ERR_DQ (0.27)
9.38	ASC-AS_C_NSUM _OUT_DQ (0.41)	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.39)	ASC-AS_A_RF45 _I_SUM_OUT _DQ (0.30)	ASC-OMC_A_PIT _OUT_DQ (0.30)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.30)	SUS-ITMY_L3 _JSCINF_P _DQ (0.19)	ASC-CSOFT_P _OUT_DQ (0.17)	ASC-CSOFT_P _SM_DQ (0.17)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.16)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.16)
9.50	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.73)	ASC-AS_C_NSUM _OUT_DQ (0.70)	ASC-AS_A_RF45 _I_SUM_OUT _DQ (0.58)	PEM-EX_MAG _EBAY_SUSRACK _Y_DQ (0.40)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.39)	ASC-OMC_A_PIT _OUT_DQ (0.39)	ASC-X_TR_A _PIT_OUT_DQ (0.31)	ASC-X_TR_B _PIT_OUT_DQ (0.29)	SUS-SR2_M3 _JSCINF_Y_IN1 _DQ (0.25)	ASC-AS_C_NSUM _OUT_DQ (0.25)
9.62	ASC-AS_C_NSUM _OUT_DQ (0.67)	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.67)	ASC-AS_A_RF45 _I_SUM_OUT _DQ (0.51)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.37)	ASC-OMC_A_PIT _OUT_DQ (0.37)	ASC-X_TR_A _PIT_OUT_DQ (0.29)	PEM-EX_MAG _EBAY_SUSRACK _Y_DQ (0.29)	ASC-CHARD_P _SM_DQ (0.18)	ASC-CHARD_P _OUT_DQ (0.18)	ASC-AS_C_NSUM _OUT_DQ (0.18)
9.75	ASC-AS_C_NSUM _OUT_DQ (0.72)	ASC-OMC_A_PIT _OUT_DQ (0.55)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.55)	ASC-AS_A_RF45 _I_SUM_OUT _DQ (0.55)	ASC-AS_B_RF45 _I_PIT_OUT _DQ (0.51)	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.51)	ASC-AS_A_DC _NSUM_OUT _DQ (0.17)	ASC-OMC_B_YAW _OUT_DQ (0.16)	OMC-ASC_QPD _B_YAW_OUT _DQ (0.16)	ASC-AS_C_NSUM _OUT_DQ (0.16)
9.88	ASC-AS_C_NSUM _OUT_DQ (0.69)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.55)	ASC-OMC_A_PIT _OUT_DQ (0.55)	ASC-AS_A_RF45 _I_SUM_OUT _DQ (0.50)	ASC-AS_B_RF45 _I_PIT_OUT _DQ (0.54)	OMC-ASC_QPD _B_PIT_OUT _DQ (0.21)	ASC-OMC_B_PIT _OUT_DQ (0.21)	OMC-ASC_QPD _B_YAW_OUT _DQ (0.17)	ASC-OMC_B_YAW _OUT_DQ (0.17)	ASC-AS_C_NSUM _OUT_DQ (0.17)
10.00	ASC-OMC_A_PIT _OUT_DQ (0.42)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.39)	ASC-AS_C_NSUM _OUT_DQ (0.39)	ASC-Y_TR_A _PIT_OUT_DQ (0.39)	ASC-CHARD_P _OUT_DQ (0.28)	ASC-CHARD_P _OUT_DQ (0.28)	ASC-CHARD_P _SM_DQ (0.27)	ASC-OMC_B_PIT _OUT_DQ (0.23)	OMC-ASC_QPD _B_PIT_OUT _DQ (0.23)	ASC-AS_C_NSUM _OUT_DQ (0.23)
10.12	ASC-AS_C_NSUM _OUT_DQ (0.24)	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.20)	OMC-ASC_QPD _A_PIT_OUT _DQ (0.17)	ASC-OMC_A_PIT _OUT_DQ (0.17)	ASC-Y_TR_A _PIT_OUT_DQ (0.16)	ASC-CSOFT_P _OUT_DQ (0.15)	ASC-CSOFT_P _SM_DQ (0.15)	ASC-CSOFT_P _INI_DQ (0.13)	ASC-CSOFT_P _OUT_DQ (0.13)	ASC-AS_C_NSUM _OUT_DQ (0.13)
10.25	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.71)	ASC-AS_C_NSUM _OUT_DQ (0.71)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.71)	LSC_POPAIR _A_RF45_T_ERR _DQ (0.71)	ASC-DHARD_P _SM_DQ (0.71)	ASC-DHARD_P _OUT_DQ (0.71)	ASC-DHARD_P _INI_DQ (0.71)	CAL-CS_XARM _ERR_DQ (0.71)	OAF-CAL_XARM _ERR_DQ (0.71)	ASC-AS_B_RF45 _Q_PIT_OUT _DQ (0.71)



Vajente Phys. Rev. D 105, 102005 (2022)

<https://git.ligo.org/gabriele-vajente/bruco>



Offline subtraction

Do the subtraction offline, when data has been already recorded.

No limitation on computational time (within reason), no causality limits.

Well known approaches: Wiener filter (frequency domain or FIR filters)

Novel approaches: IIR filters in time domain

Online in-loop subtraction

Do the subtraction on-line, in real time, as part of the feedback control loop.

Feed-forward technique, needs to account for control loop response.

Online out-of-loop subtraction

Do the subtraction on-line, in close-to-real time, as part of the calibration pipeline, outside the control loops.

More flexible, need time-domain processing.

Davis *et al* 2019 *Class. Quantum Grav.* **36** 055011, Driggers *et al* 2019 *Phys. Rev. D* **99**, 042001

- Frequency-domain estimation of coupling transfer functions (for multiple witness channels)
- Frequency-domain subtraction

$$\tilde{h}(f) = \tilde{h}_0(f) + \sum_i \tilde{f}_i(\omega) \tilde{x}_i(\omega)$$

Multiple witness signals x_i (potentially cross-correlated) contribute to the strain

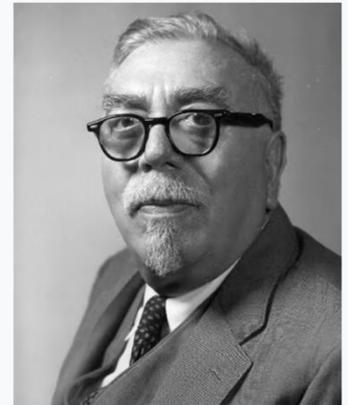
$$\tilde{c}_{ij}(f) = \tilde{Y}_i(f) \tilde{Y}_j^*(f)$$

Cross-spectral density estimated from data, where $Y_i = \{h, x_1, x_2, \dots\}$

We can invert the estimated transfer function matrix separately for each frequency bin and use the estimated couplings in the first equation above to do the subtraction

$$\begin{bmatrix} \tilde{c}'_{01}(f_b) \\ \tilde{c}'_{02}(f_b) \\ \vdots \\ \tilde{c}'_{0N}(f_b) \end{bmatrix} = \begin{bmatrix} \tilde{c}_{11}(f_b) & \dots & \tilde{c}_{N3}(f_b) \\ \tilde{c}_{12}(f_b) & \dots & \tilde{c}_{N2}(f_b) \\ \vdots & \ddots & \vdots \\ \tilde{c}_{1N}(f_b) & \dots & \tilde{c}_{NN}(f_b) \end{bmatrix}^{-1} \begin{bmatrix} \tilde{c}_{01}(f_b) \\ \tilde{c}_{02}(f_b) \\ \vdots \\ \tilde{c}_{0N}(f_b) \end{bmatrix}$$

Norbert Wiener



Born	November 26, 1894 Columbia, Missouri, U.S.
Died	March 18, 1964 (aged 69) Stockholm, Sweden
Education	Tufts College (BA) Cornell University (MA) Harvard University (PhD)

- Implementation details

- Need to produce time-domain signal
- Subtraction is done in frequency domain, need to transform back
- Overlapping windows to get rid of transients at segment boundaries

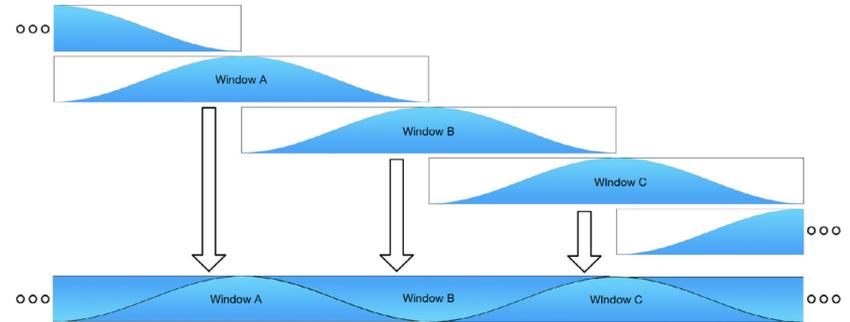


Figure 2. Visualization of how transfer function measurements are tiled in time. Transfer functions are measured in time windows (typically 1024 s) with 50% overlap. For a given time, the transfer function between the witness sensor and $h(t)$ is measured and the witness data are filtered to generate their projected contributions to $h(t)$. A Hann window is applied to each section of projected data before adding them together, resulting in a single projected $h(t)$ time series that has incorporated the time dependence of the transfer functions.

- Need to use numerically stable ways to invert coupling matrix
- Frequency-by-frequency inversion
 - No constraint on causality or continuity of coupling functions
 - Could lead to over-subtraction or noise increase
- Computationally demanding at subtraction time
- There are ways to enforce causality in Wiener filters, not explored in LIGO (FIR filters for example)

Davis *et al* 2019 *Class. Quantum Grav.* **36** 055011, Driggers *et al* 2019 *Phys. Rev. D* **99**, 042001

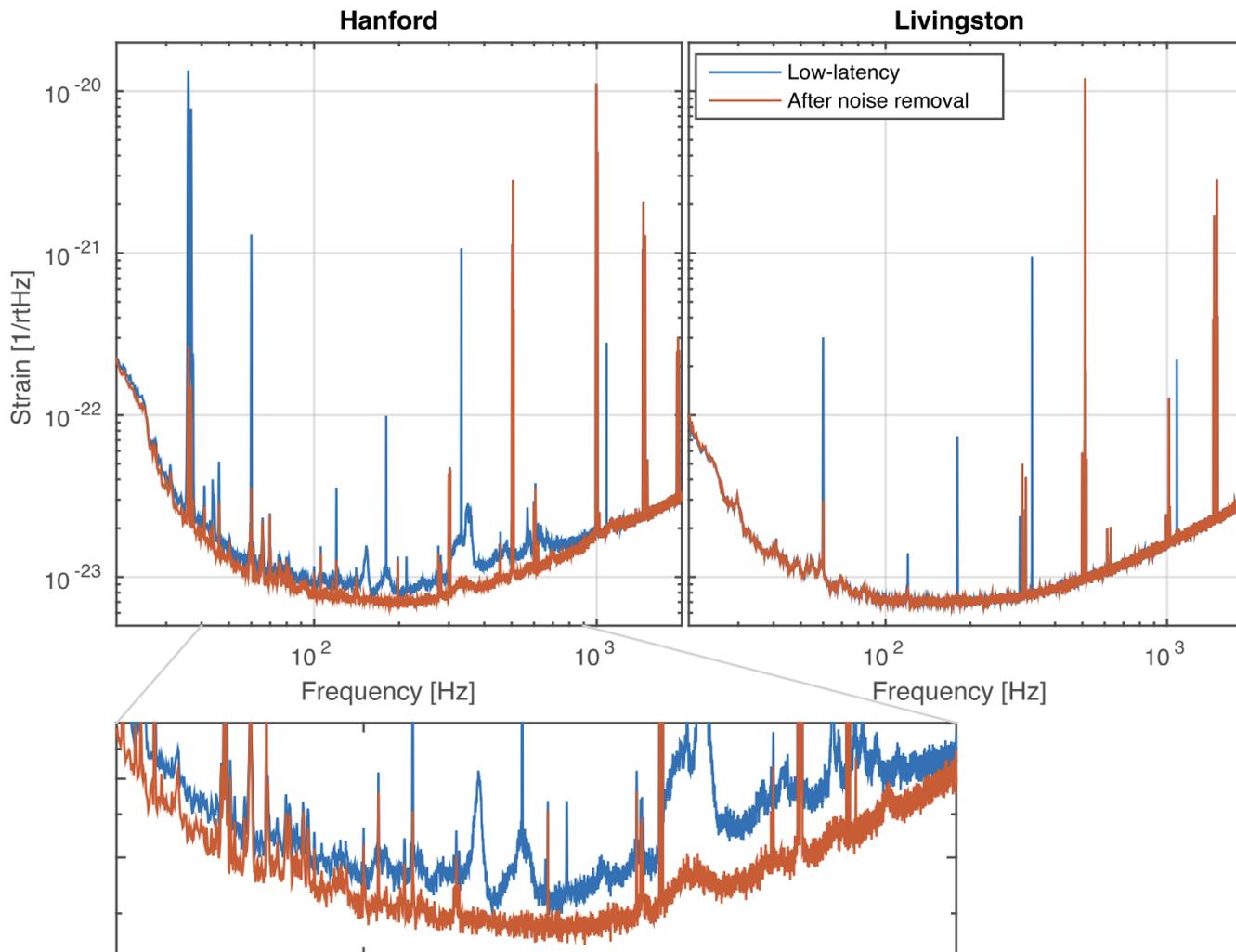
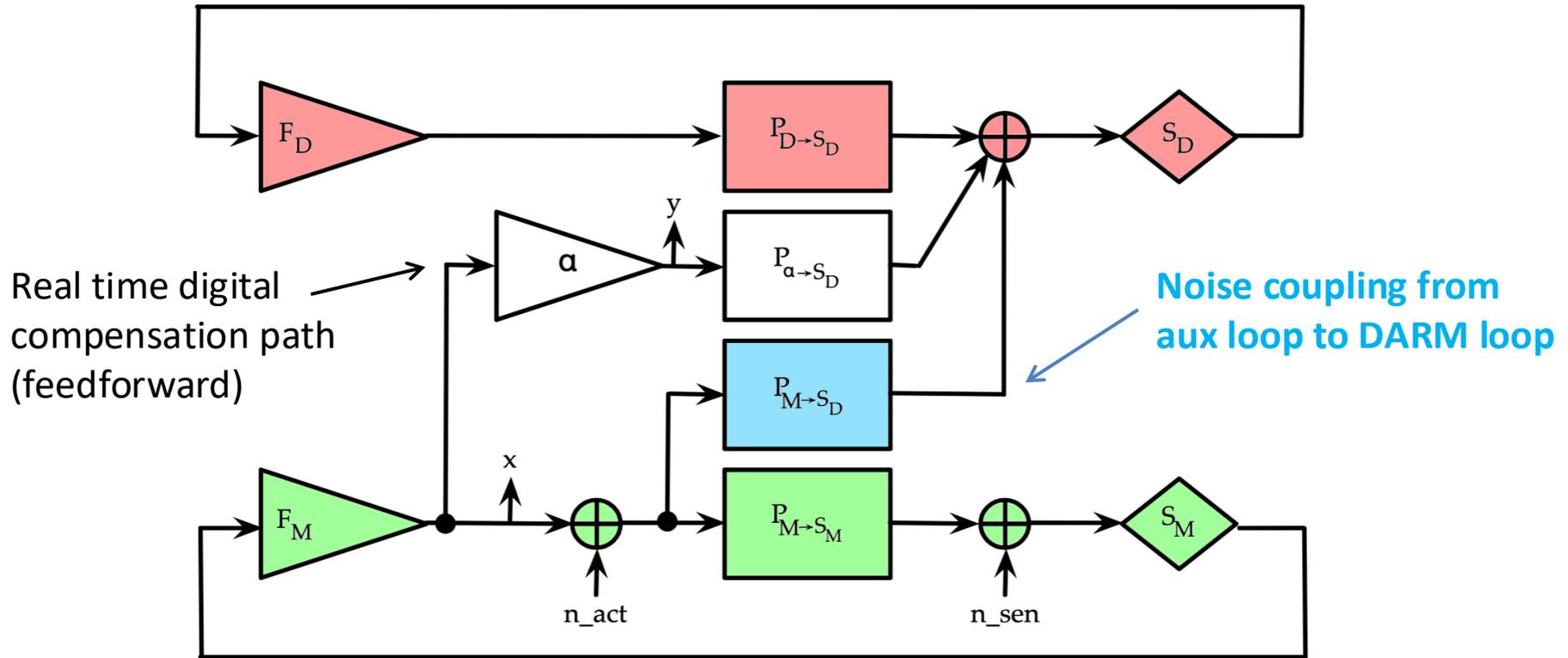
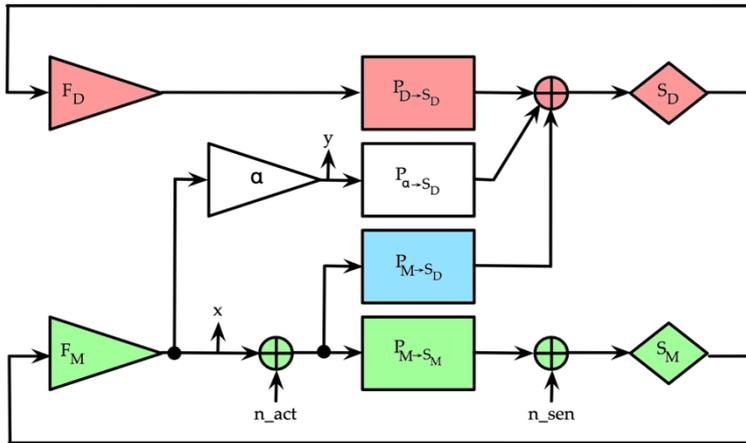


FIG. 2. Noise amplitude spectral density improvement of the LIGO detectors, Hanford in the left panel and Livingston in the right panel. Low-latency data used to identify GW candidates and determine their significance are shown in blue traces. Interferometer noise ASD after postprocessing noise removal is shown in red traces. The inset is a zoom of Hanford data. These spectra were estimated using 1024 s of data starting on 25 June 2017 at 08:00:00 UTC, a time when both LIGO interferometers were online and in an observation-ready state.

Differential arm (DARM) length feedback control loop

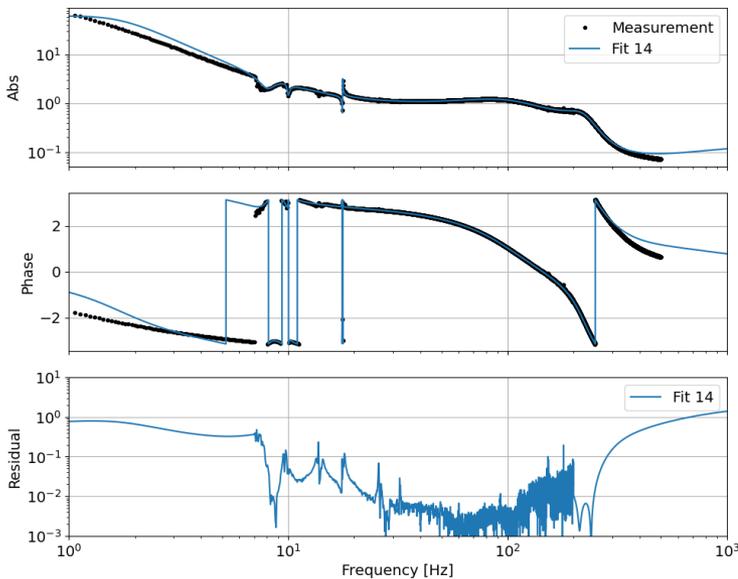


Auxiliary degree of freedom length feedback control loop

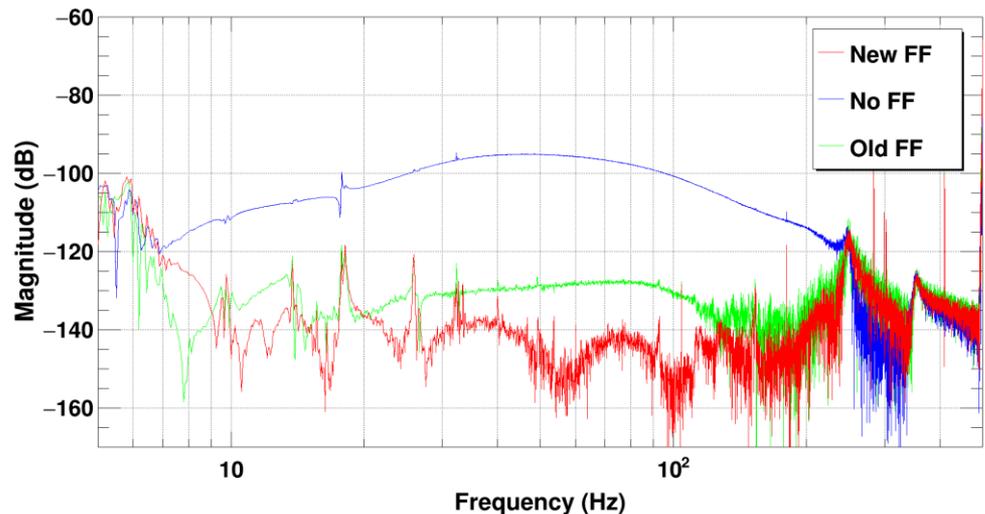


- α is a transfer function that needs to be estimated from measurements
- α is implemented as an infinite impulse response (IIR) filter in the real time control system
- Need a way to fit the estimated transfer function (measurement sampled in frequency domain) into zeros, poles and gain for the IIR implementation

$$\alpha = -\frac{P_{(M \rightarrow S_D)}}{P_{(\alpha \rightarrow S_D)}} \quad \text{Those two transfer functions can be measured with active noise injections}$$

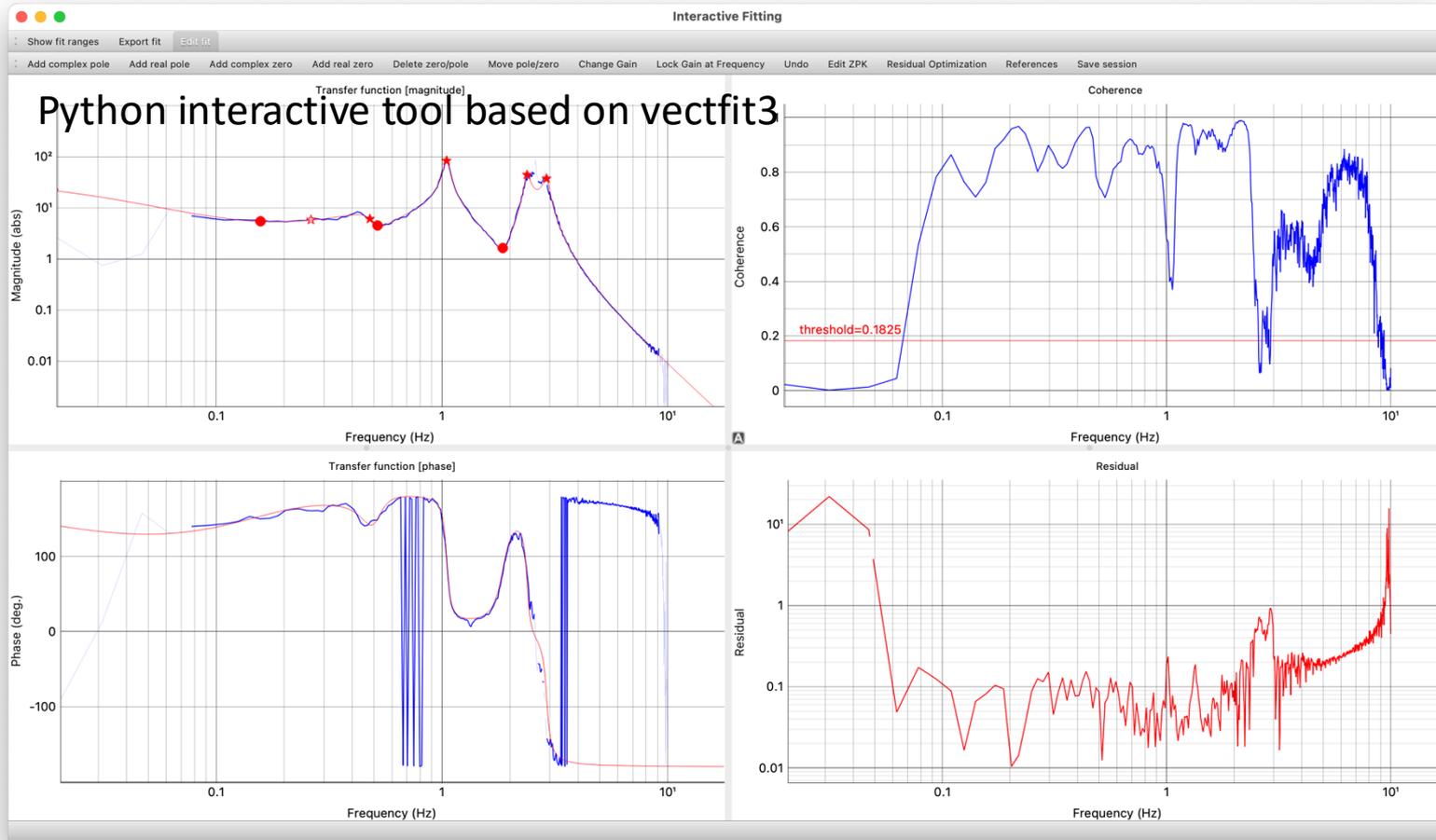


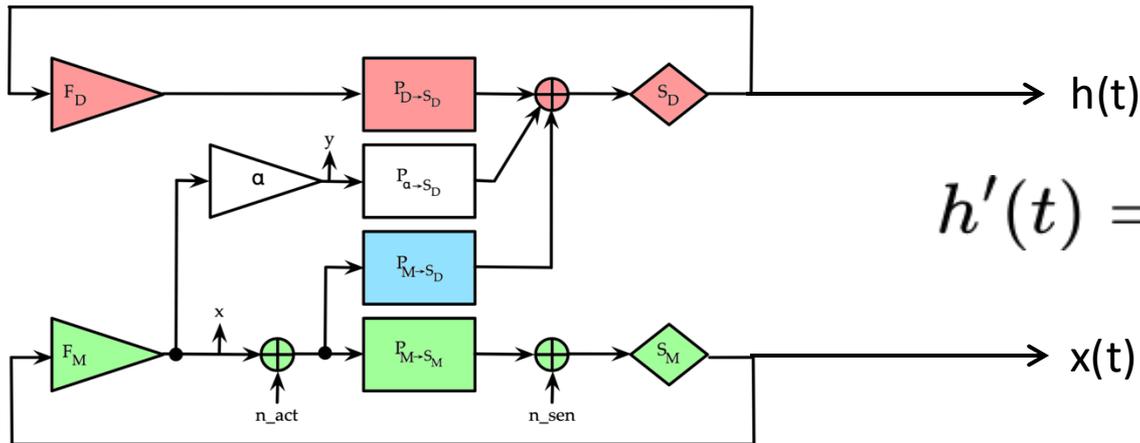
DARM / MICH



- From frequency-sampled measured transfer function to zeros, poles, gain model (that can be implemented online as IIR filter)
- Many tools available (IIRational, vectfit3, MATLAB sysid)

$$\tilde{T}(f) = k \frac{\prod_{i=0}^N (2j\pi f - z_i)}{\prod_{i=0}^M (2j\pi f - p_i)}$$





$$h'(t) = h(t) - IIR[x(t)]$$

- Digital subtraction, with time series already recorded, out-of-loop
- By implementing real-time digital filters
- No need to compensate for feedback loop suppression
- Still need to measure and fit transfer functions (more later when discussing NonSENS)
- In principle any physical coupling can always be approximated with a digital IIR filter
 - However, there might be delays in the data acquisition and real-time processing that can introduce non-physical time advances

Non-linear means anything but linear: no universal model

What about DNN?

They are “universal approximators to any function”

Input layer Deep neural network Output layer

Multiple hidden layers

Use your physics knowledge

To build a “reduced order” model

$$\epsilon_{NL}(t) = \sum_{i=1}^N \int_0^{+\infty} \alpha_i(\tau) n_i(t - \tau) d\tau.$$

Strain (m/√Hz)

Frequency [Hz]

- Original
- Modularity
- Non-Stationarity

Unknown complex non-linear processes

Give up?

Spectrogram: H1-GDS-CALIB_STRAIN

15 Hz resonance 15 Hz resonance

DARM

Frequency [Hz]

Time (minutes) from 2023-11-08 23:38:00 (1313521898.0)

Spectrogram: H1-PEM-CS_ACC_BEAMTUBE_MCTUBE_Y_DO

Accelerometer

Frequency [Hz]

Time (minutes) from 2023-11-08 23:38:00 (1383521898.0)

Stepped amplitude injection showing Q in the low freq.

Swept frequency injection

An increase in vibration by only a factor of 2 produces noise in-band in DARM.

- Use a Volterra series expansion of the non-linear coupling
- Stop at the second order for now
- What can we do with that?

$$\begin{aligned}
 h(t) &= h_0(t) + \overset{\text{Linear term}}{L[x(t)]} + \overset{\text{Non-Linear term}}{NL[x(t)]} \\
 &= h_0(t) + \int_{-\infty}^0 \alpha_L(\tau) x(t - \tau) d\tau \\
 &+ \int_{-\infty}^0 \int_{-\infty}^0 \alpha_{NL}(\tau_1, \tau_2) x(t - \tau_1) x(t - \tau_2) d\tau_1 d\tau_2 \\
 &+ \dots \quad \text{higher order terms}
 \end{aligned}$$

Second order
Volterra term

Vito Volterra



Born	3 May 1860 Ancona, Papal States
Died	11 October 1940 (aged 80) Rome, Kingdom of Italy
Alma mater	University of Pisa Scuola Normale Superiore di Pisa

$$h(t) = h_0(t) + \int_{-\infty}^0 \int_{-\infty}^0 \alpha_{NL}(\tau_1, \tau_2) x(t - \tau_1) x(t - \tau_2) d\tau_1 d\tau_2$$

From time domain \uparrow to frequency domain \downarrow

$$\tilde{h}(\omega) = \tilde{h}_0(\omega) + \int \int \tilde{\alpha}_{NL}(\omega_1, \omega_2) \tilde{x}(\omega_1) \tilde{x}(\omega_2) \delta(\omega - \omega_1 - \omega_2) d\omega_1 d\omega_2$$


Second order Volterra kernel,
depends on two frequencies

- This quadratic kernel is the equivalent to the linear transfer function
- Depends on two frequencies
- Increased complexity
- Extension of coherence to quadratic coupling is possible but not straightforward

$$x = \sin(7 \text{ Hz} \cdot 2\pi t) + \cos(3 \text{ Hz} \cdot 2\pi t)$$

$$y = x^2$$

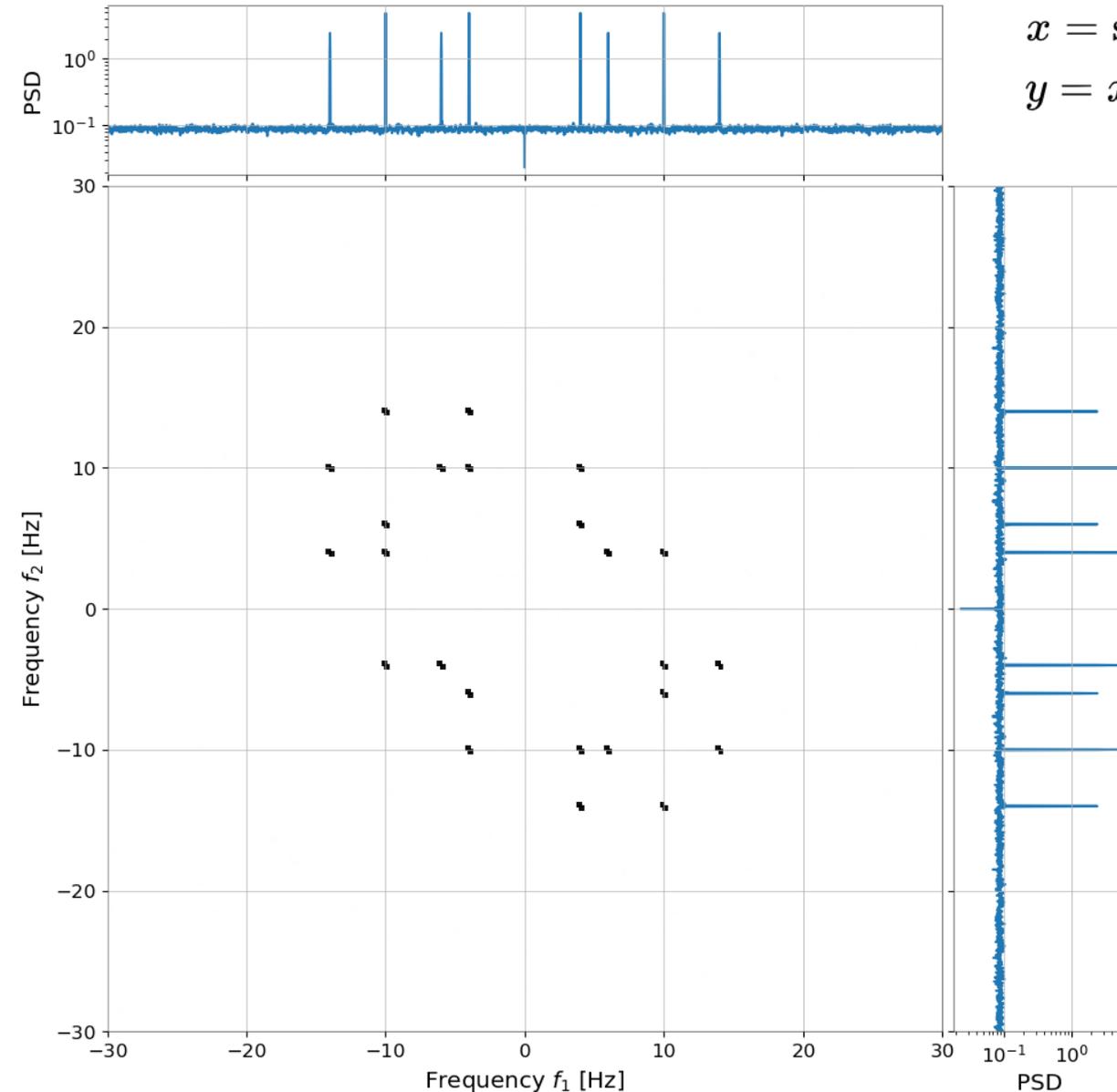
bicoherence(y, y, y)

It's a function of two frequencies

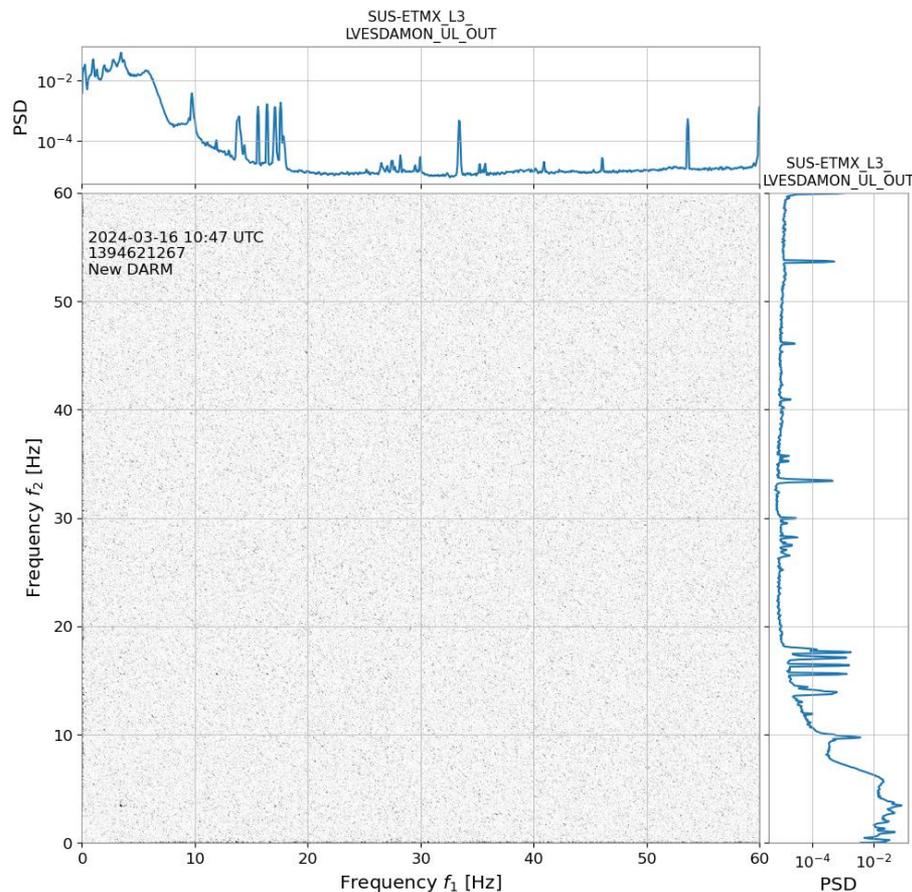
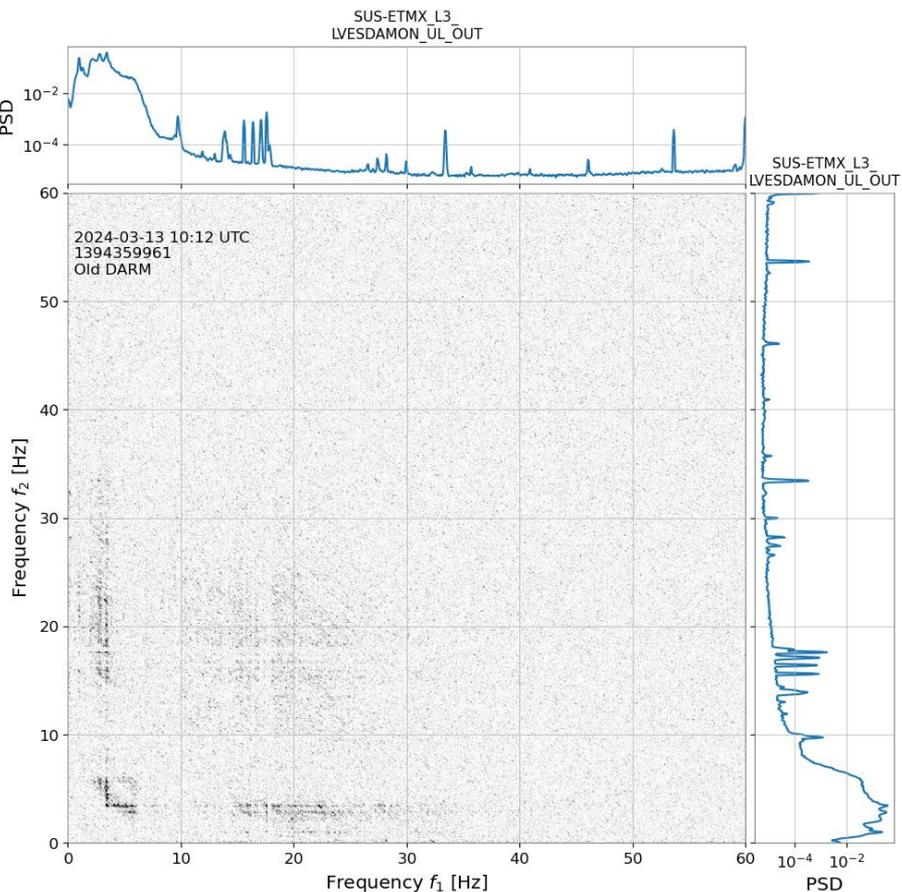
High values when there is phase correlation between the two frequencies f_1 and f_2 and combinations like f_1+f_2 and f_1-f_2

Lot of redundancy in a plot like this

There are different definitions of bicoherence, not all limited between 0 and 1, not all well behaved

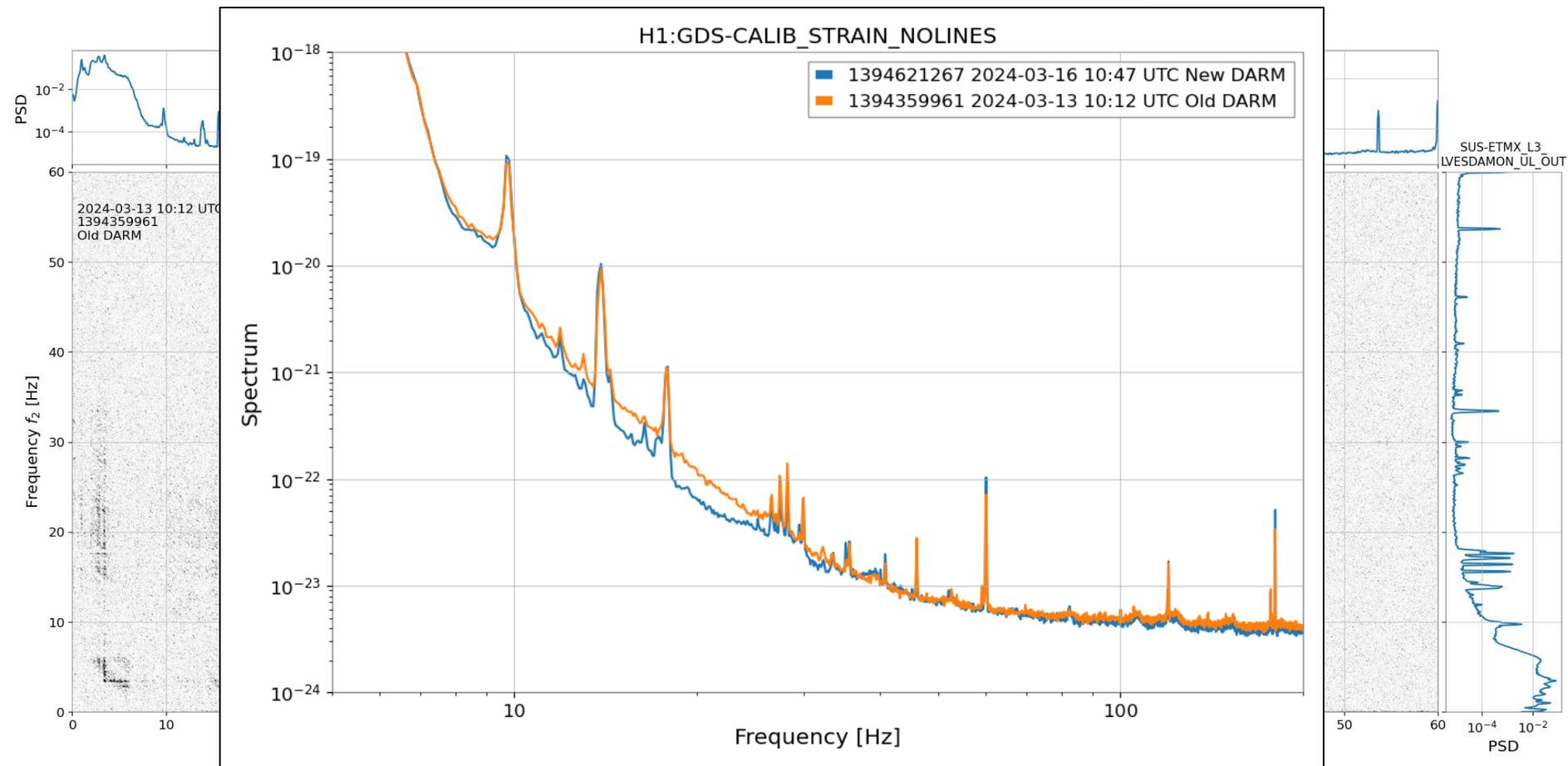


<https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=76459>



Helped with identifying broadband noise in DARM (10-40 Hz) that was modulated by low frequency RMS (~ 2 Hz). Control improvements reduced the low frequency motion and improved the noise. **We didn't try to subtract this non-linear noise coupling!**

<https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=76459>



Helped with identifying broadband noise in DARM (10-40 Hz) that was modulated by low frequency RMS (~2 Hz). Control improvements reduced the low frequency motion and improved the noise. **We didn't try to subtract this non-linear noise coupling!**

- Can we use bicoherence for discovery? **Not really**
 - Takes pairs of signals: 4000 x 4000 combinations
 - Computational demanding
 - One bicoherence takes 5 minutes on one computer (with 1000 seconds of data and two channels sampled at 512 Hz)
 - 150 years to blindly explore all combinations
 - Not well conditioned between 0 and 1, biased estimator or poor convergence
 - Difficult to have an automatic threshold to select significant channel candidates

<https://github.com/synergetics/spectrum>

[Sigl 1994] Sigl, J.C. and N.G. Chamoun. 1994. An introduction to bispectral analysis for the electroencephalogram. *Journal of Clinical Monitoring* 10:392-404

[Nagashima 2006] Nagashima, Y., Itoh, K., Itoh, S.-I., Hoshino, K., Fujisawa, A., Ejiri, A., Takase, Y., et al. (2006). Observation of coherent bicoherence and biphasic in potential fluctuations around geodesic acoustic mode frequency on JFT-2M. *Plasma Physics and Controlled Fusion*, 48(5A), A377-A386. **[Hagihira 2001]** Hagihira, S., Takashina, M., Mori, T., Mashimo, T., & Yoshiya, I. (2001). Practical Issues in Bispectral Analysis of Electroencephalographic Signals. *Anesthesia & Analgesia*, 93(4), 966-970.

[Kim 1979] Young C. Kim; Edward J. Powers, *Digital Bispectral Analysis and Its Applications to Nonlinear Wave Interactions*, IEEE Transactions on Plasma Science (Volume: 7, Issue: 2, June 1979) 120

Simplest non-linearity:
quadratic



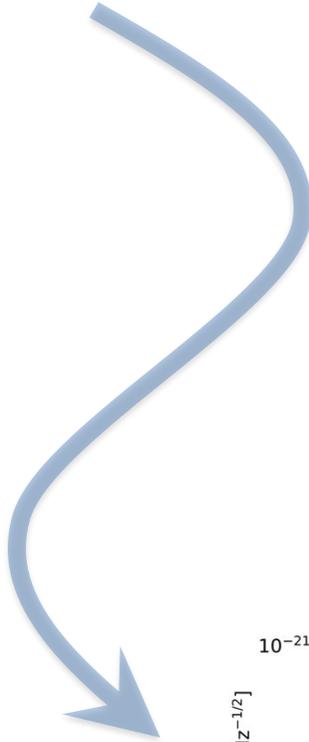
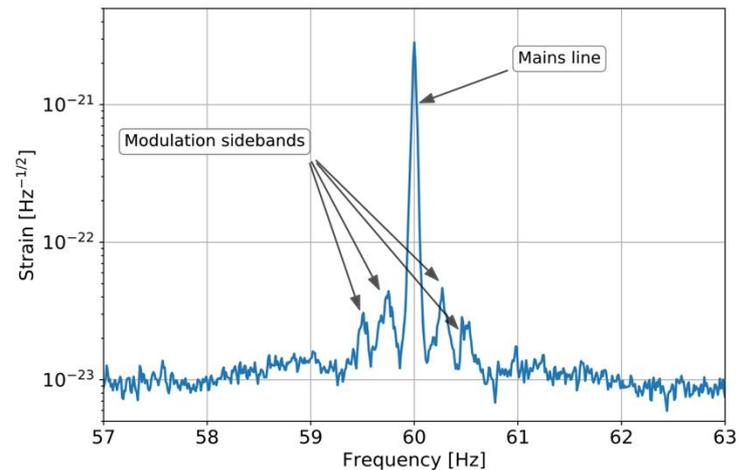
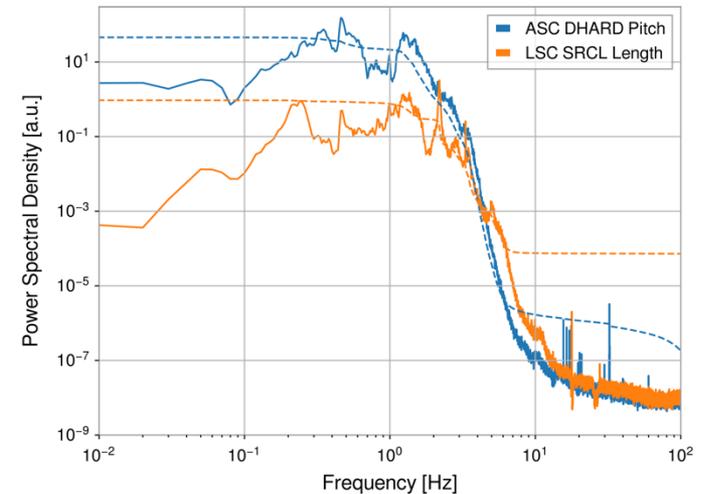
Most signals are
dominated by low
frequency (<3-4 Hz)



Non-stationary linear couplings

$$\varepsilon_{\text{NL}}(t) = \sum_{i=1}^N \int_0^{+\infty} \alpha_i(\tau) n_i(t - \tau) d\tau,$$

$$\sum_{i,j=1}^N \iint_0^{+\infty} \alpha_{i,j}(\tau_1, \tau_2) w_i(t - \tau_1) w_j(t - \tau_2) d\tau_1 d\tau_2$$



$$h(t) = H[s(t)] + \sum_{i=1}^N \alpha_i [x_i(t)s(t)]$$

- **$h(t)$** : target signal, what we want to clean (example: GW strain)
- **$s(t)$** : noise witness signal, i.e. a measurement of the noise that couples into $h(t)$, through modulated transfer functions (example: SRCL control signal)
- **$x_i(t)$** : a set of auxiliary signals that witness the coupling modulation (example: angular degree of freedom fluctuations)
- Assuming that $x_i(t)$ varies on time scales much slower than the noise $s(t)$
- The **stationary coupling** is modeled with a transfer function H
- The **non-stationary couplings** are modeled by assuming the noise couples through (many) stationary transfer functions, each one modulated by one of the witness signals

- Frequency domain approach
 - Find the optimal solution, independently for each frequency bin (frequency-domain a-causal **Wiener filter** [1])

Modulated signals

$$s_i(t) = \begin{cases} s(t) & \text{if } i = 0 \\ x_i(t)s(t) & \text{if } i = 1, \dots, N \end{cases}$$

Noise prediction

$$y(t) = \sum_{i=0}^N \alpha_i [s_i(t)]$$

Residual after subtraction

$$\tilde{r}(\omega) = \tilde{h}(\omega) - \tilde{y}(\omega) = \tilde{h}(\omega) - \sum_{i=0}^N \tilde{\alpha}_i(\omega) \tilde{s}_i(\omega)$$

Minimize PSD of residual

$$S[r, r] = S[h, h] - (\alpha^+ \mathbf{H} + \mathbf{H}^+ \alpha) + \alpha^+ \mathbf{S} \alpha$$

Optimal solution

$$\alpha = \mathbf{S}^{-1} \mathbf{H}$$

- Describe each transfer function using a **parametrized form**

- Laplace domain: $\alpha_h(s) = \frac{\sum_{i=0}^{N_N} b_i s^i}{\sum_{j=0}^{N_D} a_j s^j}$

- Define a frequency-integrated cost function

$$\mathcal{C}(\theta_{h,i}) = \frac{1}{\omega_2 - \omega_1} \int_{\omega_1}^{\omega_2} W(\omega) S_{rr}(\omega) d\omega \quad W(\omega) = \frac{1}{S_{hh}(\omega)}$$

- Use gradient-based optimization algorithms

$$\frac{\partial \mathcal{C}}{\partial \theta_{h,n}} = \frac{1}{\omega_2 - \omega_1} \int_{\omega_1}^{\omega_2} W(\omega) \left[\frac{\partial S_{rr}}{\partial \alpha_h} \cdot \frac{\partial \alpha_h}{\partial \theta_{h,n}} + \frac{\partial S_{rr}}{\partial \alpha_h^*} \cdot \frac{\partial \alpha_h^*}{\partial \theta_{h,n}} \right] d\omega$$

- How to enforce stability and improve the coefficient scaling

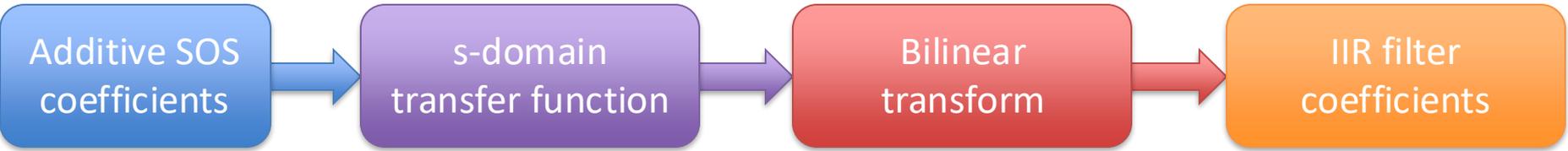
$$\alpha_h(s) = \frac{\sum_{i=0}^{N_N} b_s s^i}{\sum_{j=0}^{N_D} b_s s^j}$$

$$\alpha(s) = k + \sum_{i=1}^{N_D} \frac{\rho_i}{s - p_i}$$

$$\alpha(s) = k + \sum_i \left[\frac{\rho_i}{s - p_i} + \frac{\rho_i^*}{s - p_i^*} \right] + \sum_j \frac{r_j}{s - f_j}$$

$$\alpha_h(s) = k_h + \sum_{i=1}^{N_D/2} \frac{b_{h,1}s + b_{h,0}}{s^2 + a_{h,1}s + a_{h,0}}$$

Poles are **stable**
 $\mathcal{R}(p_i) < 0$ and $f_i < 0$
 if and only if
 $a_{h,i} > 0$



- Or parametrize directly in z-domain using starred transform

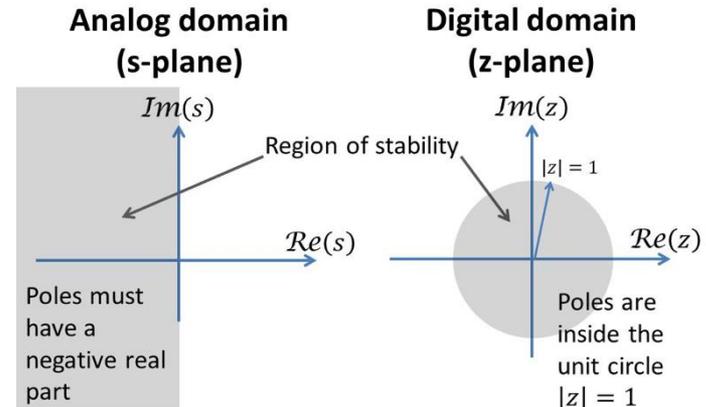
$$z = e^{sT} \quad T = f_s^{-1}$$

$$\alpha_h(z) = k_h + \sum_{i=1}^{N_{SOS}} \frac{b_{h,1}z + b_{h,0}}{z^2 + a_{h,1}z + a_{h,0}}$$

- Poles are stable (inside the unit circle in z-plane) if and only if

$$\begin{aligned} -1 < a_0 < 1 \\ -1 - a_0 < a_1 < 1 + a_0 \end{aligned}$$

Filter stability in the analog and digital domain



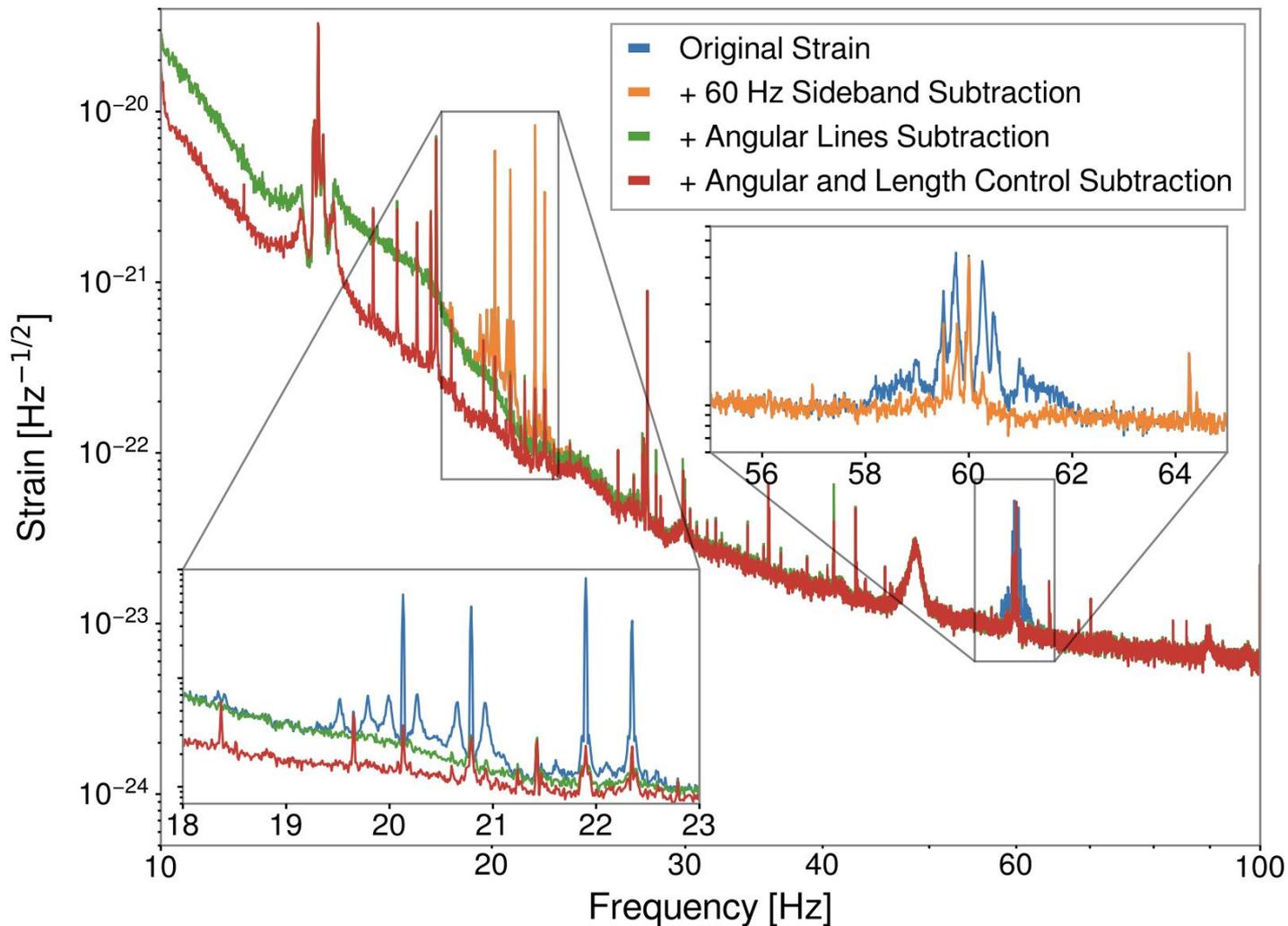
LIGO-G1100863 Matone: An Overview of Control Theory and Digital Signal Processing (5) 21 21

<https://slideplayer.com/slide/4469125/>

<https://git.ligo.org/gabriele-vajente/nonsens>

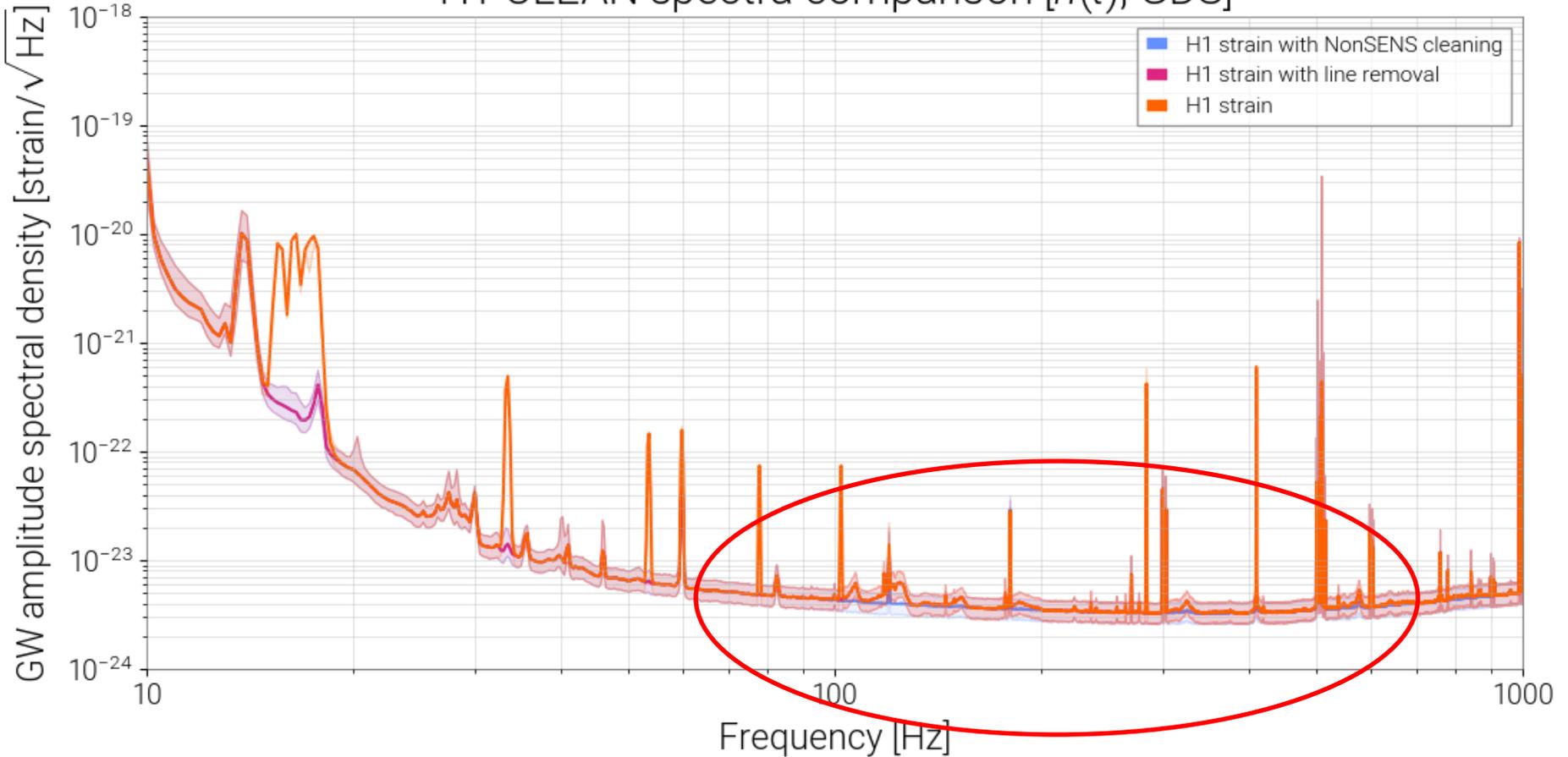
Vajente et al. *Phys. Rev. D* **101**, 042003 (2020)

Vajente *Phys. Rev. D* **105**, 102005 (2022)



[1396828818-1396915218, state: Locked]

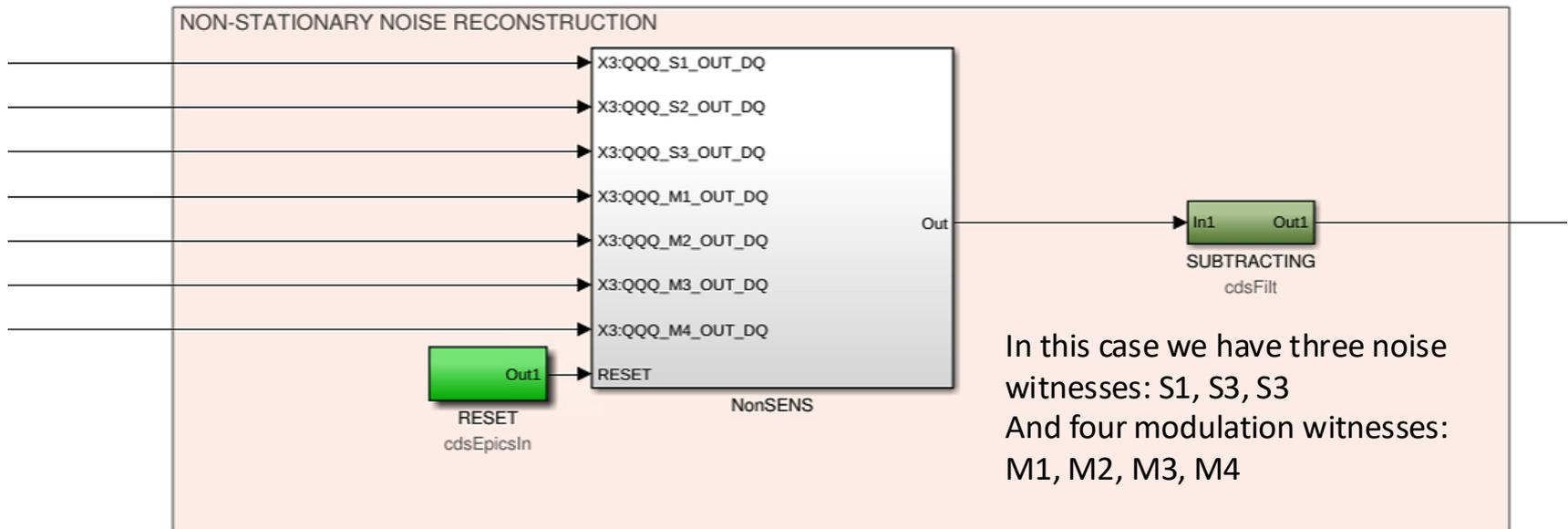
H1 CLEAN spectra comparison $[h(t), \text{GDS}]$



Linear subtraction of input laser beam jitter noise

The Simulink block contains a C-code block. Both are generated automatically, so implementing the subtraction is as easy as copy and paste (one needs only to provide signals with the correct sampling frequency)

<https://git.ligo.org/gabriele-vajente/nonsens/-/blob/master/frontend/README.md>



Subtraction coefficients are accessible via **EPICS channels**

A python script to initialize the coefficient is generated automatically.

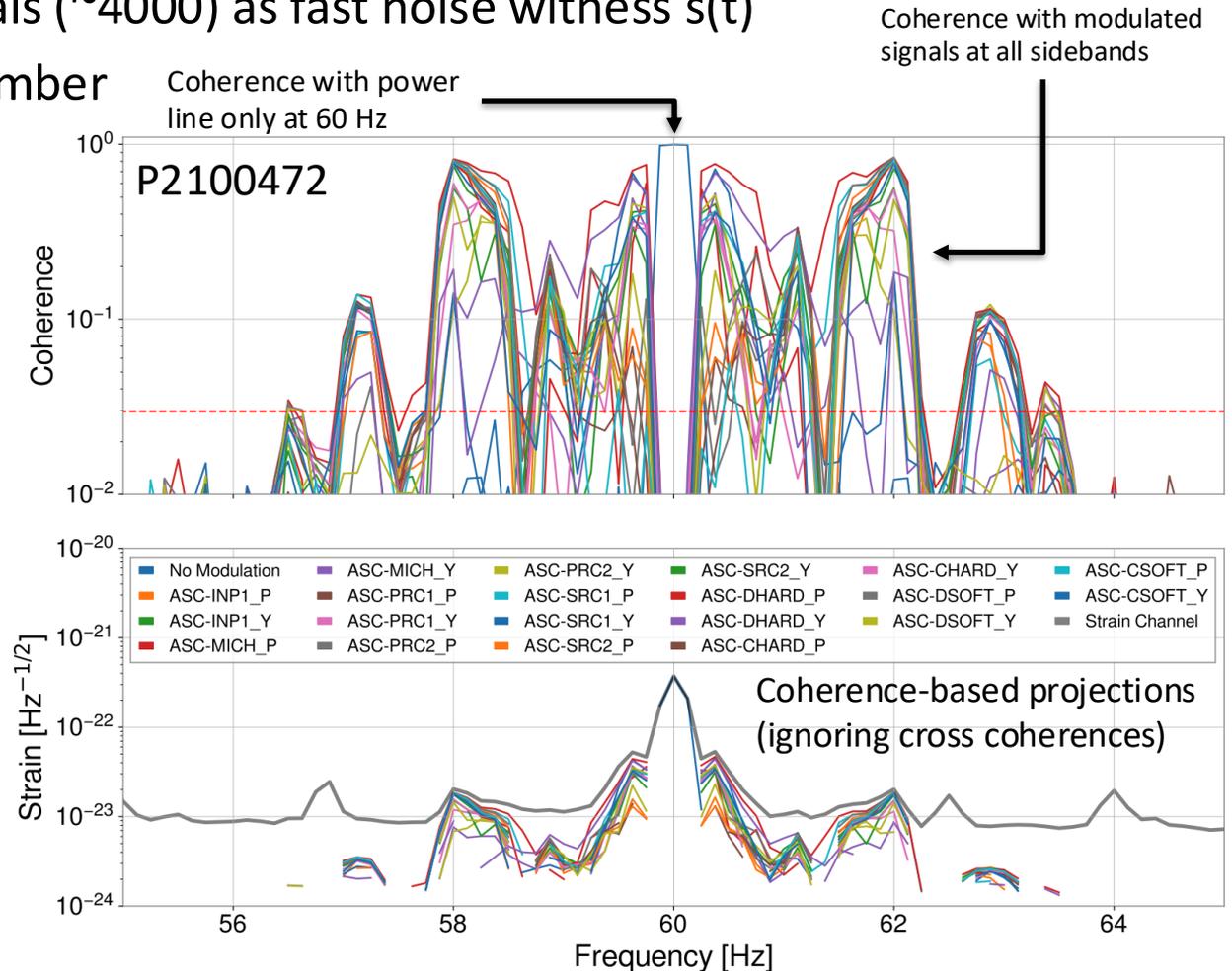
Coefficients can be updated on the fly, without the need to recompile the frontend model (unless one changes the subtraction architecture, such as the number of channels)

The infrastructure is ready for **adaptive filtering**.

- To discover modulated noise couplings:

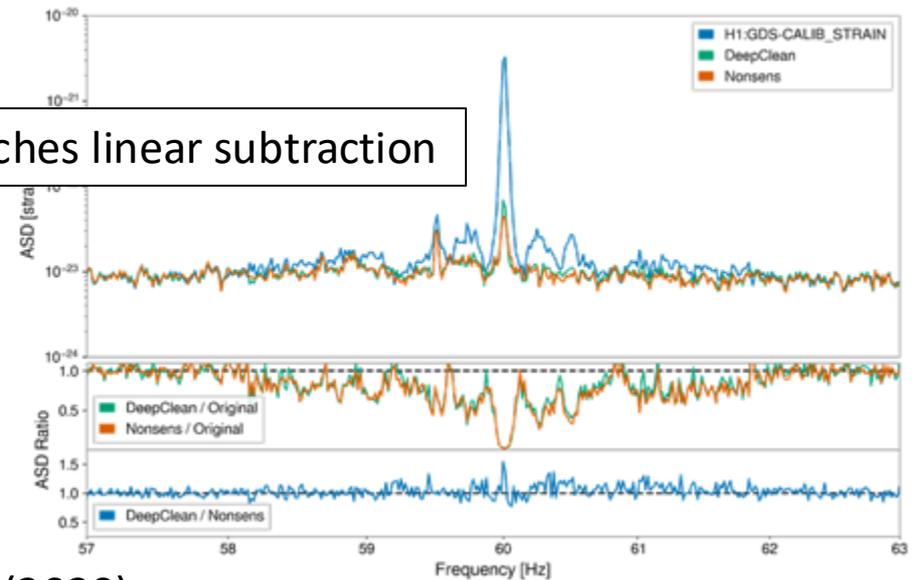
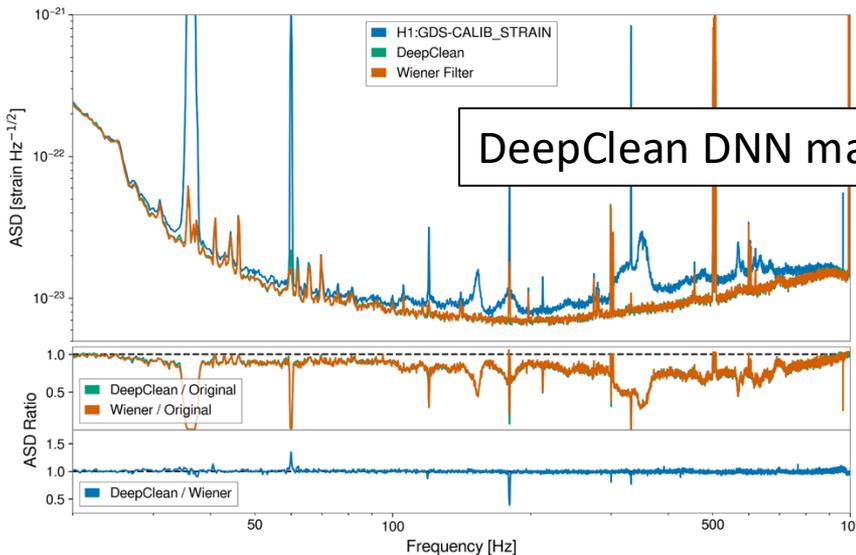
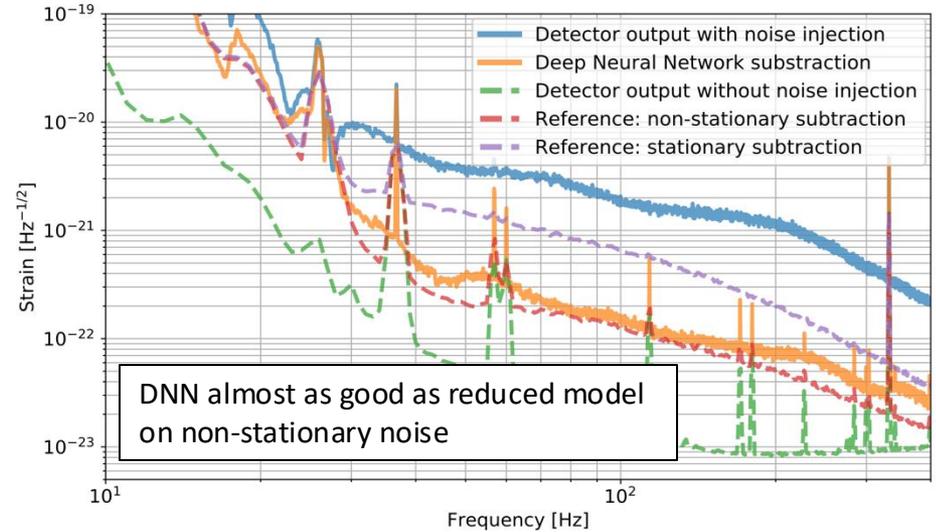
$$h(t) = H[s(t)] + \sum_{i=1}^N \alpha_i [x_i(t)s(t)]$$

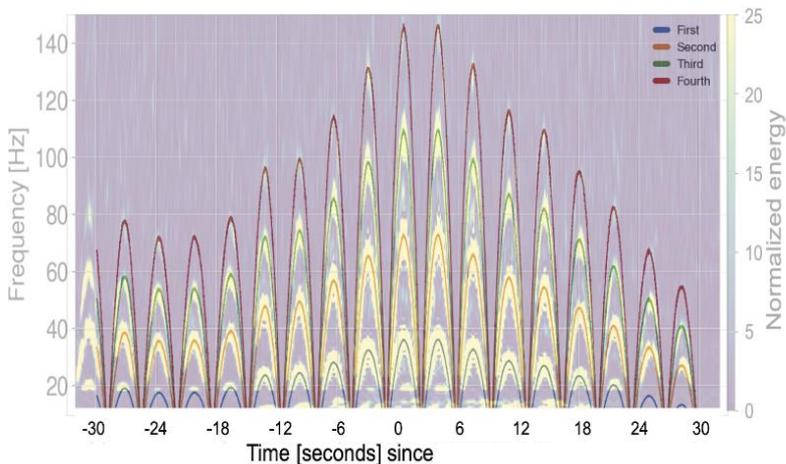
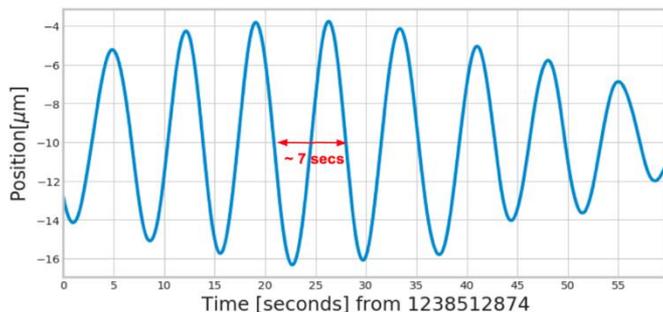
- Use all available signals (~4000) as fast noise witness $s(t)$
- Restrict to a small number (10-20) modulation witness signals (for example: all ASC error signals)
- This tool would have discovered the origin of the 60 Hz power line sidebands



- Can we use **Deep Neural Networks** to approximate the non-linear / non-stationary noise couplings?
 - in principle yes
 - in practice it's hard to train and harder to understand

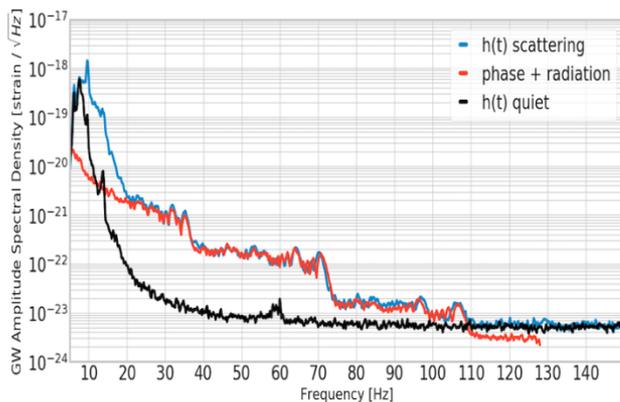
Vajente et al. *Phys. Rev. D* **101**, 042003 (2020)





$$h_{ph}(f) = A \frac{\lambda}{8\pi L} \mathcal{F}[\sin \delta\phi(t)]$$

$$\phi(t) = \phi_0 + \delta\phi_{sc}(t) = \frac{4\pi}{\lambda} |x_0 + \delta x_{sc}(t)|$$



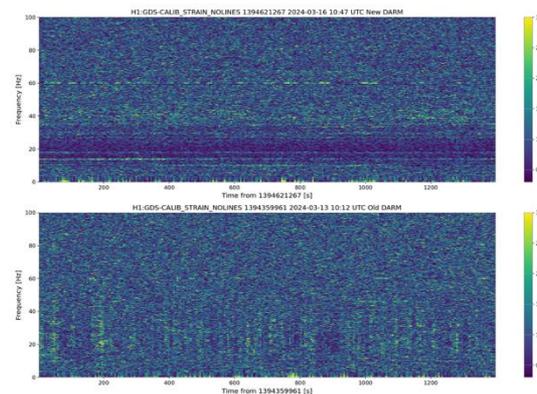
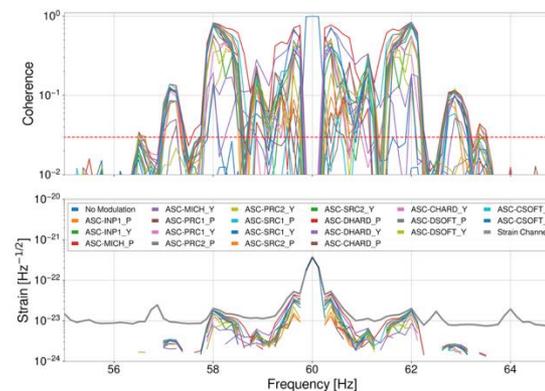
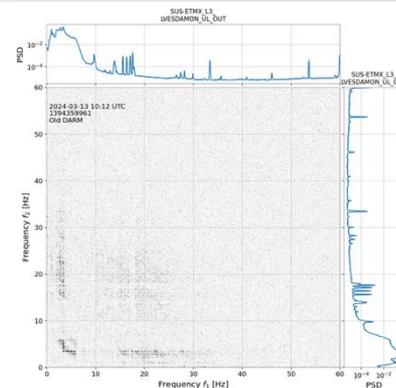
Highly non-linear process

We don't observe all inputs

S Soni *et al* 2021
Class. Quantum Grav. **38** 025016

- Many ad-hoc solutions, based on human intuition and guesses
 - **Bicoherence** with limited number of channels, need physical knowledge of the detector to interpret
 - **Modulated coherence**, need good guesses of modulation sources
 - Look for non-stationarity in spectrograms
- Can we leverage some data mining or artificial intelligence techniques?

4,000+ fast channels
 200,000+ slow channels
 28 MB/s of potentially interesting data





- Linear noise subtraction has been successful
 - Brute-force discovery tools work well
 - Offline Wiener filtering
 - Offline time-domain NonSENS
 - Online time-domain NonSENS



- Non-linear noise subtraction much harder
 - Success with bicoherence and quadratic couplings
 - Success with modulated couplings: NonSENS
 - Can we do more with ML / AI / DNN?