SP exam assignment

Group O: Diego Hachez & Roemer Spreij

Q1. Bases and Discrete Transforms | approximating y = sin(x)

Polynomial approximation $y \approx c_0 + c_1 x + c_2 x^2 + c_3 x^3$

$\mathbf{y} \approx \mathbf{B}\tilde{\mathbf{y}}_B$ with: $\tilde{\mathbf{y}}_B = \begin{bmatrix} c_0 & c_1 & c_2 & c_3 \end{bmatrix}^T$ and: $\mathbf{B} = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & x_n^3 \end{bmatrix}$

n_px1 n_px4 4x1

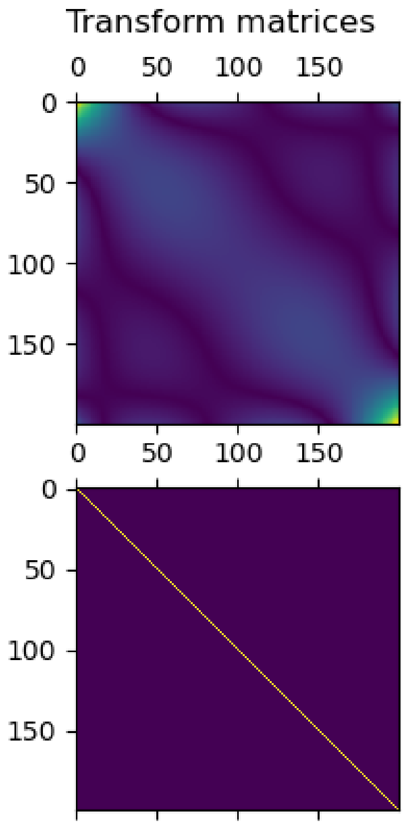
'skinny' representation: $\tilde{\mathbf{y}}_B = (\mathbf{B}^\dagger \mathbf{B})^{-1} \mathbf{B}^\dagger \mathbf{y}$

Columns of **B** (= basis vectors) = **NOT normalised!** → $\tilde{\mathbf{y}}_B$ gives c_i directly. Also: B = **NOT orthogonal!**

Harmonic approximation: DFT: $\tilde{\mathbf{y}} = \Psi^\dagger \mathbf{y}$ with Ψ the Fourier matrix. $\tilde{\mathbf{y}}$ contains the coefficients of the DFT modes. The approximation is found by keeping only the first 4 coefficients in $\tilde{\mathbf{y}}$, and the first 4 modes of Ψ to reconstruct y. The reduced Ψ and $\tilde{\mathbf{y}}$ are here denoted with a hat: $\mathbf{y} \approx \hat{\Psi}\hat{\mathbf{y}}$

Ψ **= ORTHONORMAL.**

n_px1 n_px4 4x1



Polyfit: smaller error for half-period signal
DFT: smaller error for full-period signal

Note: np.polyfit gives exactly the same c_i as the matrix computation! Because it uses the same basis. It is the same transform!

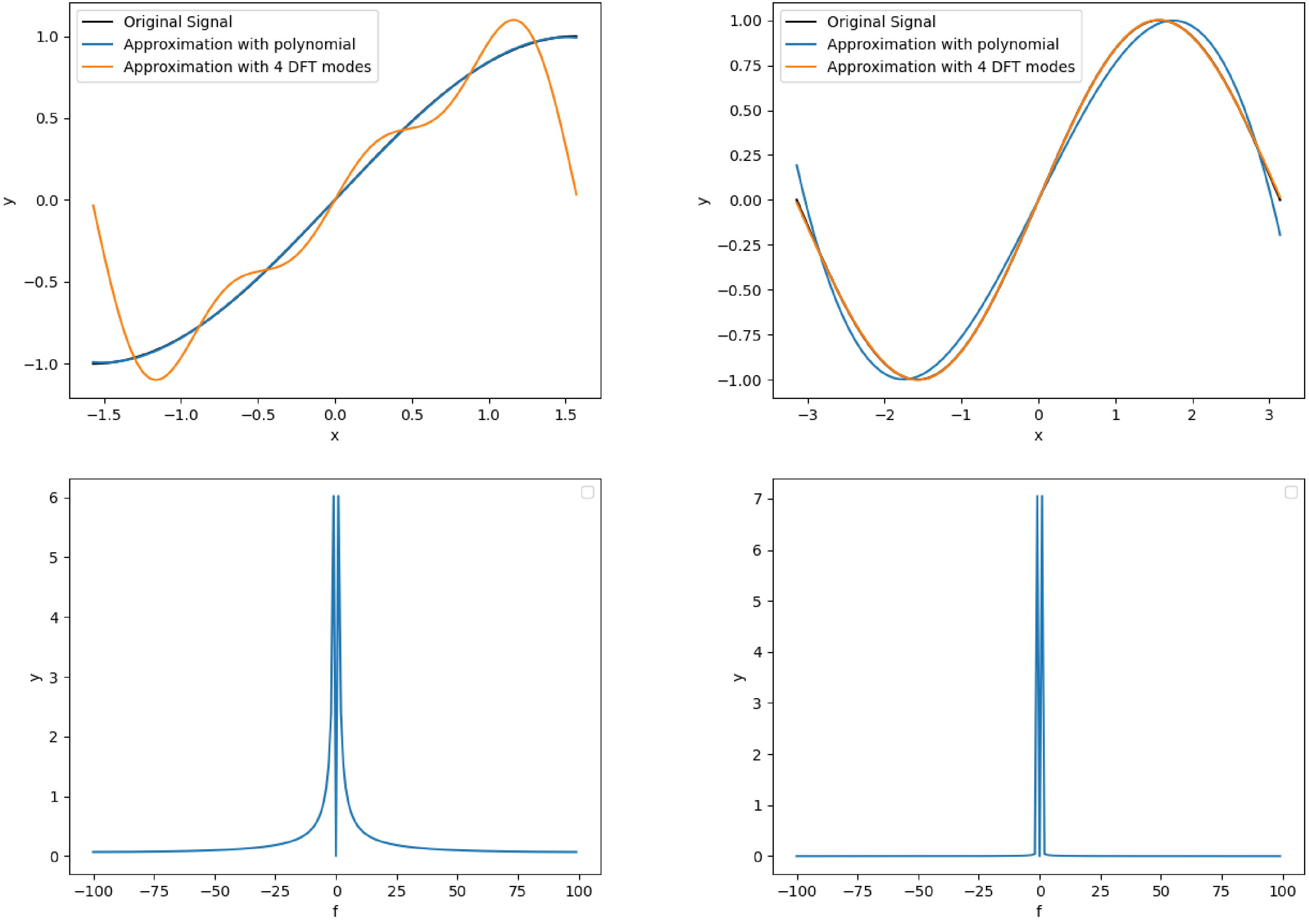
Error is: $E = \frac{||\mathbf{y} - \tilde{\mathbf{y}}||}{||\mathbf{y}||}$

	Method	C0	C1	C2	C3	Error [%]
Half period	Matrix	0	9.89e-1	0	-1.45e-1	**0.40**
	DFT	-	-	-	-	**34.04**
Full period	Matrix	0	8.55e-1	0	-9.28e-2	**9.59**
	DFT	-	-	-	-	**0.55**

Q1. Bases and Discrete Transforms

Because of the discontinuity in the half-period signal, the DFT spectrum has strong **sidelobes**. These contain a big part of the representation of y, that we throw away when we only take 4 modes. This explains the poor approximation.

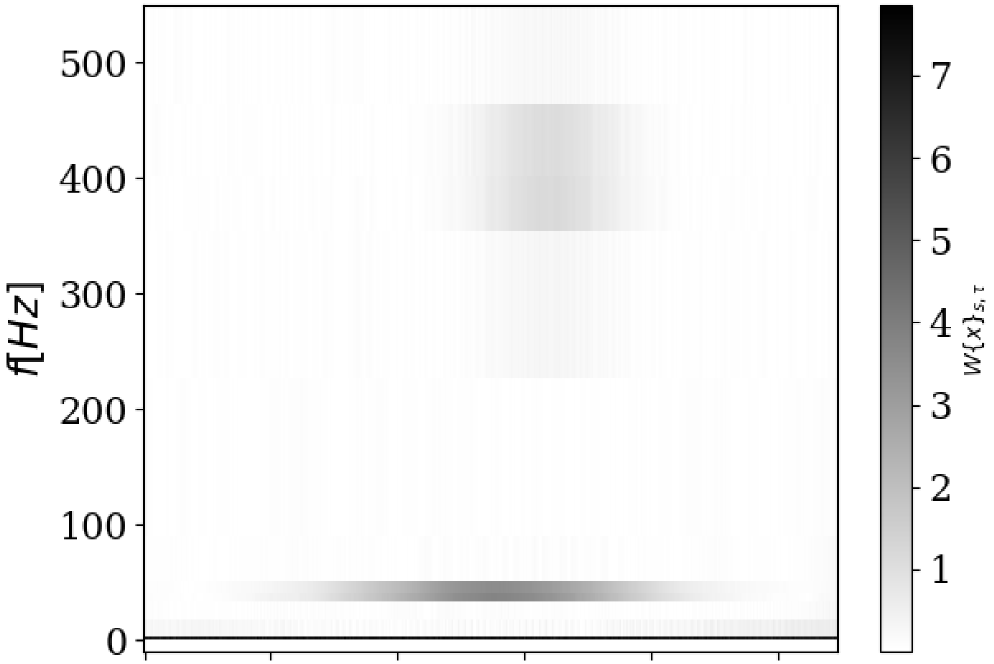
The full-period signal does not have this problem.



Q2. Time-Frequency Analysis

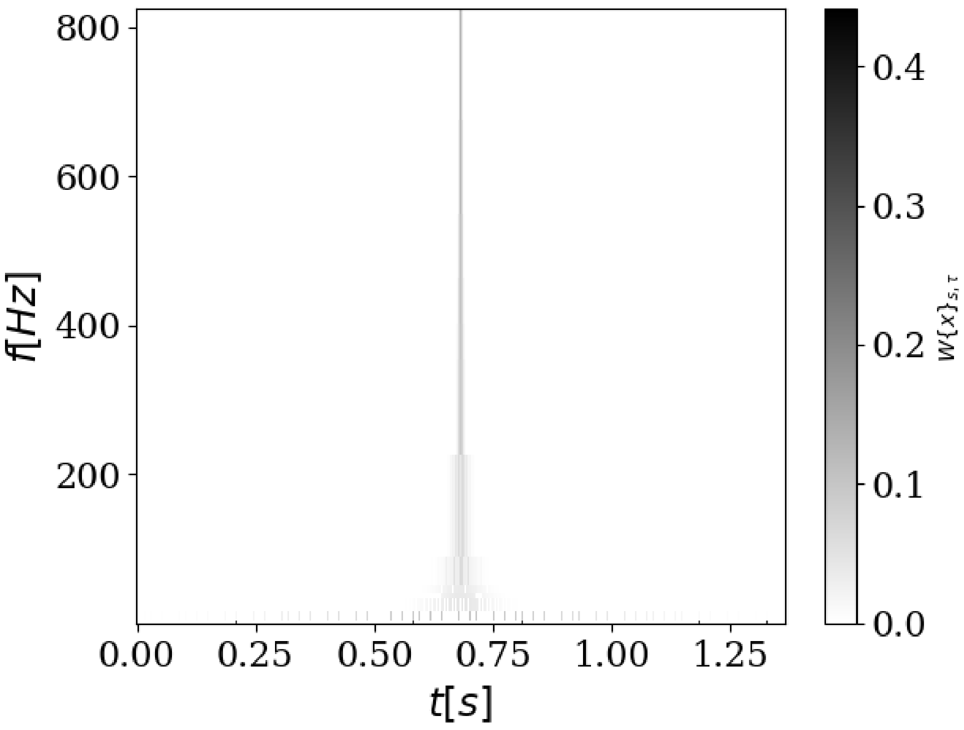
Show the time-frequency contour plots. Which of the introduced harmonics is harder to localize in time? Why?

Due to the time-frequency area constraint, the signal at low frequency that is also compact in time (40Hz) is most difficult to localize in time, if we want to determine its frequency with a reasonable relative error.
At the lowest frequencies, a black bar spans the whole time-axis. This bar corresponds to the 0,5 Hz part of the signal and is due to the locally high frequency resolution that leads to a small time resolution (again, due to the Heisenberg Uncertainly Principle). In this case the lack of time resolution is not a problem since the low-frequency signal has a considerable magnitude over the whole time-span.



Plot the time-frequency contour of a delta function. What do you see and why?

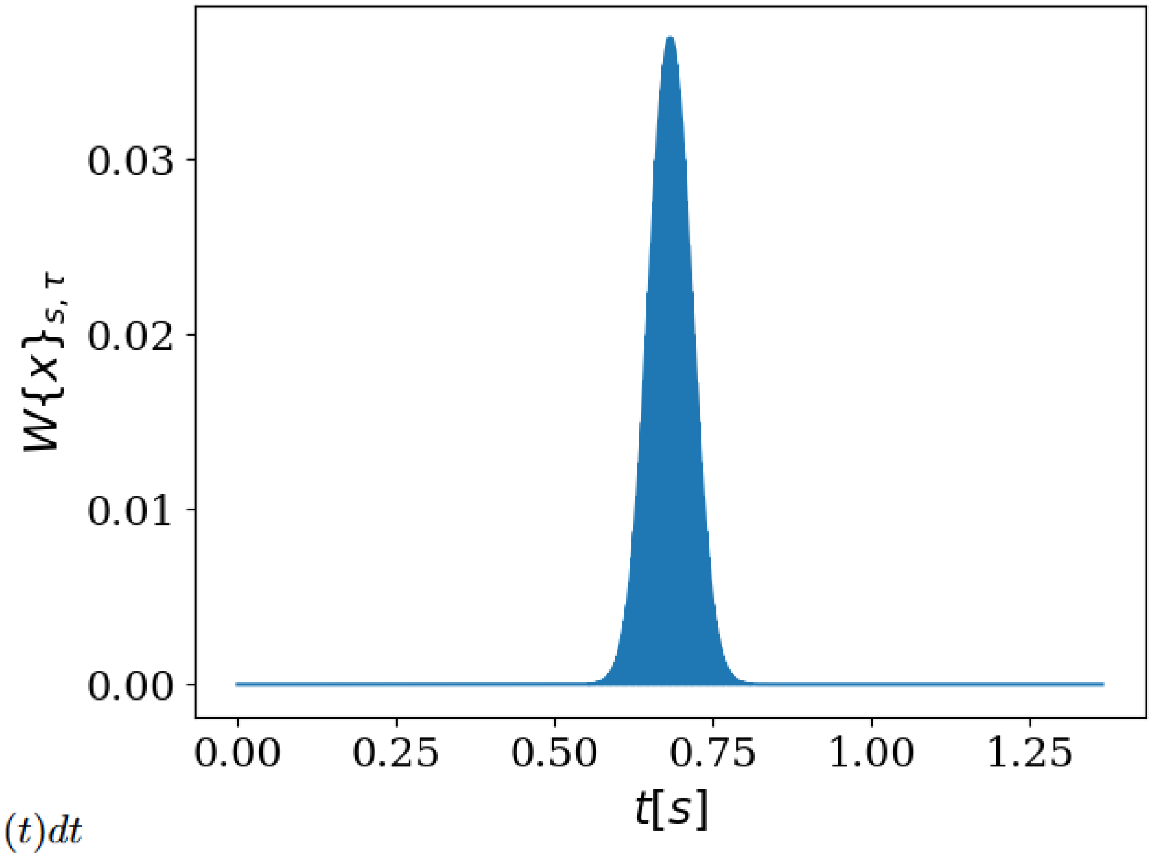
The time-frequency contour of the delta-function has the shape of an impulse that gets wider at low frequencies.
Theoretically, a delta-function has Fourier-transform that is constant along all frequencies. Here it is visibly constant along high frequencies, at the time location corresponding to the moment when the delta-function occurs. Toward lower frequencies, the line gets diffused in time due to the Heisenberg Uncertainty Principle



Q2. Time-Frequency Analysis

Next, take a slice of this plot at a given frequency and plot it as a function of time. What is the curve you see?

As expected, at given frequency the line is spread in time. The shape is gaussian.



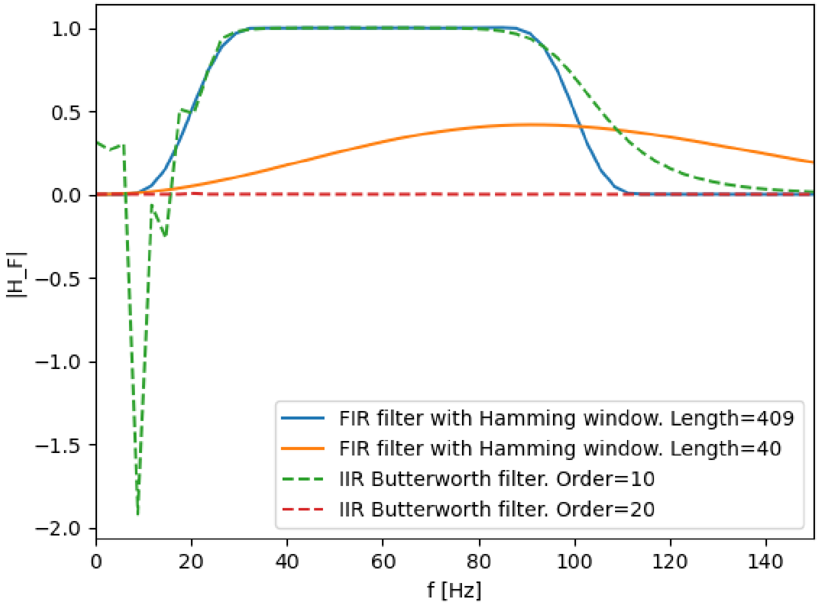
Slice at 42.8 Hz

The wavelet transformation is $\mathcal{W}\{x(t)\}(s,\tau) = \langle x(t), \psi_{s,\tau}(t)\rangle = \int_{-\infty}^{\infty} x(t)\bar{\psi}_{s,\tau}(t)dt$ If x is a delta-pulse at $t = n_t/2 * dt = t_p$, then the integral reduces to the complex conjugate of the wavelet $\bar{\psi}_{s,\tau}(t_p) = \frac{1}{\sqrt{s}}\bar{\psi}\left(\frac{t_p-\tau}{s}\right)$ We picked the complex Morlet wavelet defined as $\psi(t) = \frac{1}{\sqrt{\pi B}}\exp^{-\frac{t^2}{B}}\exp^{\mathrm{j}2\pi C t}$ where B is the bandwidth and C is the center frequency (defined in python as "cmorB-C"). So at constant frequency, the result of the wavelet operation has a magnitude proportional to $\frac{1}{\sqrt{s}}\exp^{-\frac{(t_p-\tau)^2}{s^2}}$, which can be seen as a sort of Gaussian distribution.

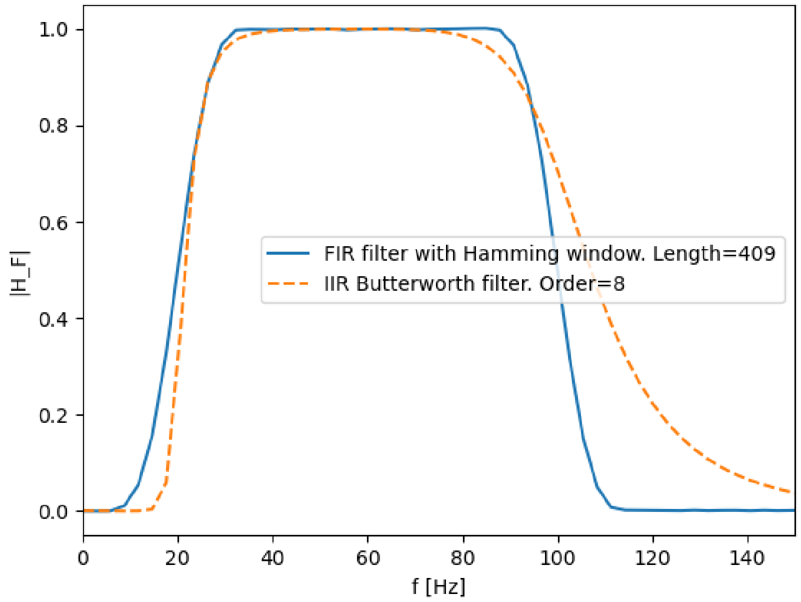
Q3. FIR and IIR Filters

From the frequency response, we see some problems with the given parameters:

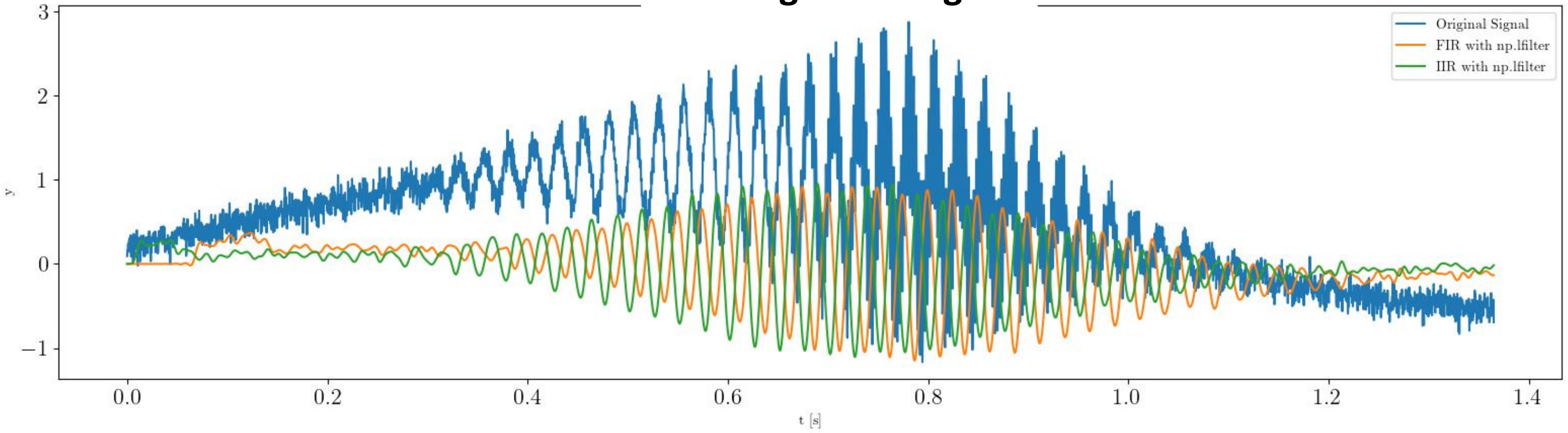
Filter	Problem	Cause
FIR | nt/10 (=40)	None.	-
FIR | nt/100 (=409)	Amplitude does not reach 1. Δf around crossover frequencies is too large.	Length of filter too short.
IIR | 10 poles	Irregularity at 20Hz crossover frequency.	10 poles = too high for such a low frequency.
IIR | 20 poles	Zero transfer function!	Same as above, but even more so. Unstable filter.



Therefore, the filters used are **FIR: nt/10** and **IIR: 8 poles.**
The IIR filters can't be applied by convolution with their impulse response h[k], because h[k] has infinite length. So, both lfilter & filtfilt are based on **difference equations.** They differ in that filtfilt applies the filter twice: once forward and once backward. lfilter introduces a **phase lag** because it starts from t=0 and moves only forward. filtfilt does not have this lag.



Filtering of full signal



Q3. FIR and IIR Filters

Comparison FIR vs. IIR, using lfilter

Note: with filtfilt, the differences between IIR and FIR are smaller (not shown here)

40Hz component

Inside filter band. Is untouched in filtered signal besides lfilter phase lag.

400Hz component

Well outside filter band. Fully filtered out.

0.5Hz component

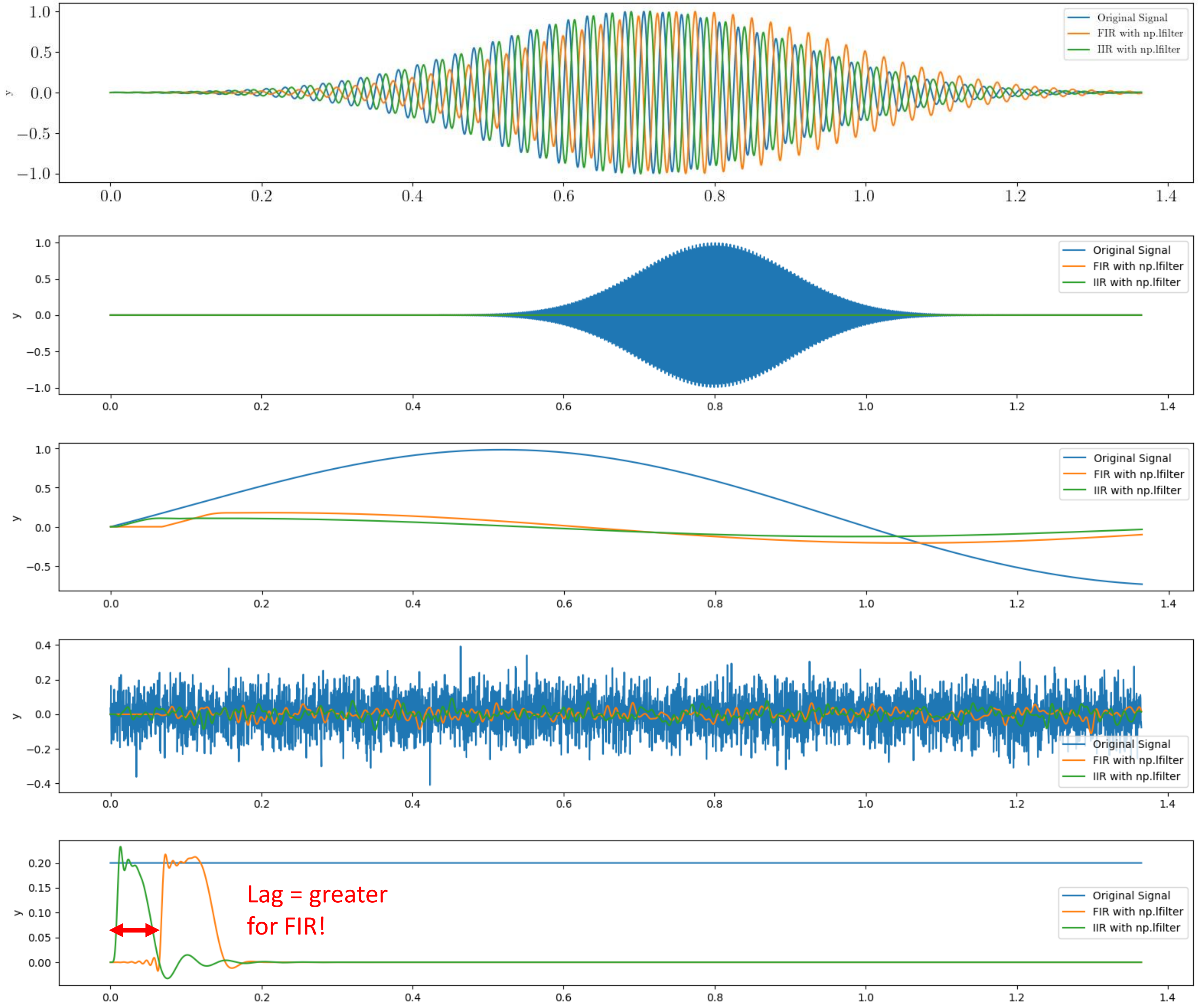
Not fully filtered out in both cases! 0.5Hz is close to 20Hz. IIR performs slightly better (see transfer function)

Noise component

Noise is present at all frequencies. Component inside band remains in filtered signal.

Constant component

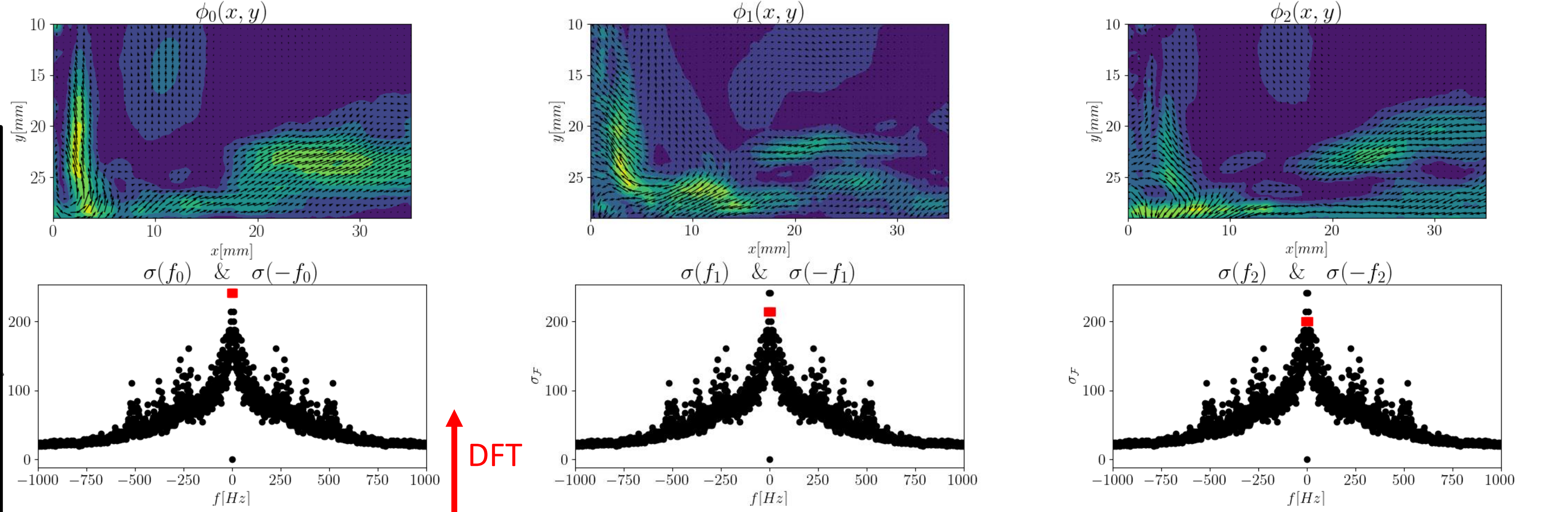
Boundary effect of lfilter present at start. Greater lag of FIR very clear.

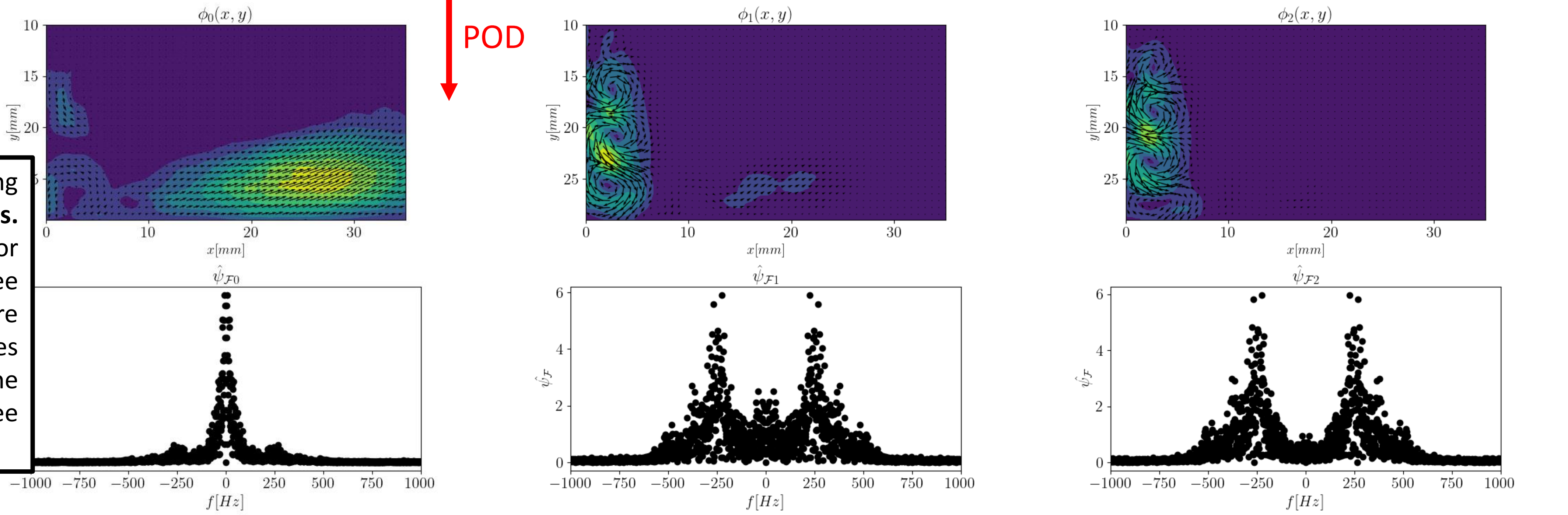


Q4. Data-Driven Modal Analysis

DFT: each mode is changing at a **single frequency:** indicated with **red dot.** The frequency spectrum for a single mode is a **dirac impulse** at the frequency of that mode (here: all these are combined into 1 graph)

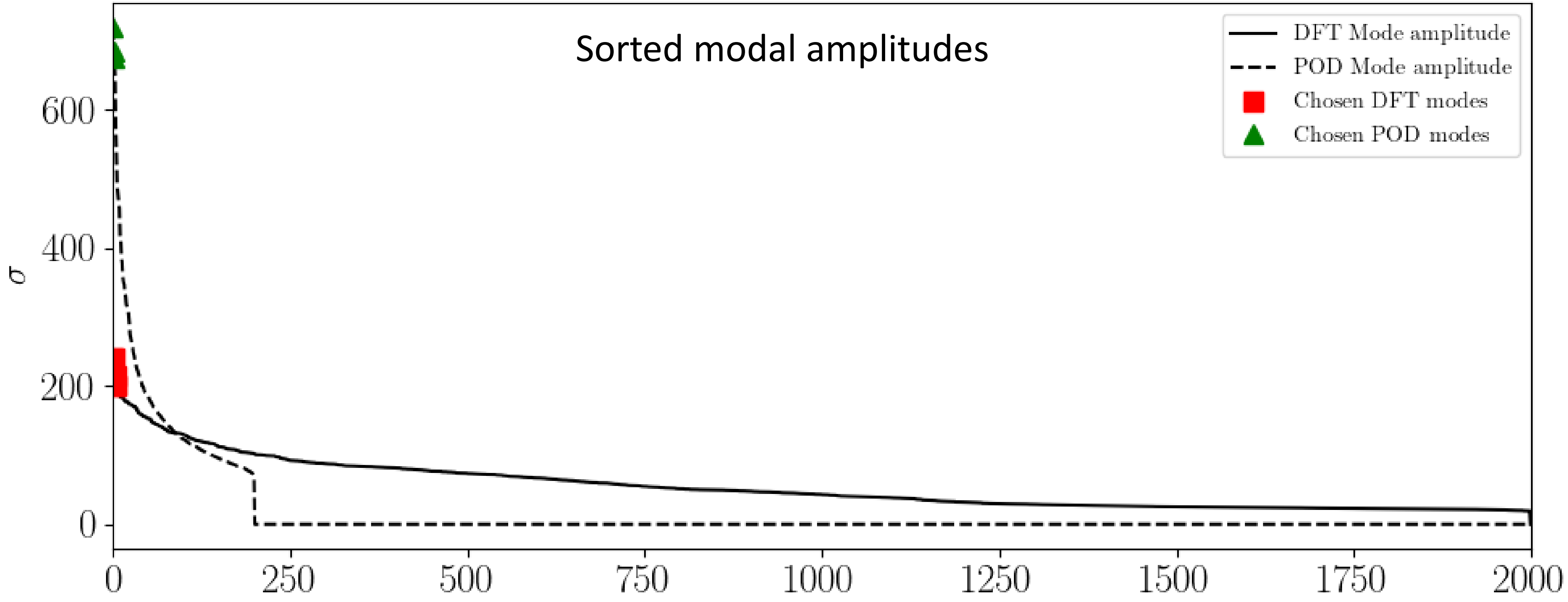
Note: to get the DFT spatial structures we are adding the conjugate modes at +/- f_i





POD: each mode is changing at a **range of frequencies.** The frequency spectrum for each mode is **different** (see graphs). Spatial structure show coherent structures more clearly, because of the sorting by energy. (see vortices in mode 1 & 2)

Q4. Data-Driven Modal Analysis



In DFT, the amplitude is more 'spread out' across the modes. The energy (~modal amplitude) is more distributed across the spectrum.
In POD the energy is more concentrated in the leading modes (by design!).

Because of this, **spatial structures** of the leading modes cannot be compared, since the decomposition is entirely different.

In this case, the leading (highest amplitude) DFT modes ***happen to be*** the lowest frequency modes, but this is not necessarily so.