IRELE

Measurement and Data Driven Modelling

Lab report

Authors :

Arico Amaury

Colot Emmeran

Professor :

J. Lataire

Academic year :

2024 - 2025

Contents

1
Lab 1

1.1 DFT

Task 1.1.1.

Prove that

$$\omega_k = \frac{2\pi}{T} k$$

where $T = nT_s$ is the window length.

Starting from the general definition of the iDFT and the one in case of a discrete signal:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{\frac{j2\pi kn}{N}}$$
$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\omega_k n T_s}$$

By simply comparing both equations, $\omega_k = \frac{2\pi}{nT_s} k$. Replacing the sampling period T_s multiplied with the number of samples n by the total sampling time, the result is obtained.

$$\omega_k = \frac{2\pi}{T} k$$

Task 1.1.2.

Prove that

$$\omega_1 = \frac{2\pi}{T} = 2\pi \frac{f_s}{N}$$

The first equality is proven using the result of the previous task. By replacing T the window length by nT_s and then defining the sampling frequency $f_s = \frac{1}{T_s}$, the result is obtained.

Task 1.1.3.

Prove that

$$X(N-k) = X(-k) = X^*(k)$$

Hint: Use $e^{j2\pi n} = 1$ ***for n*** $\in \mathbb{N}$ ***and*** $x(n) \in \mathbb{R}$**.**

By using the definition of the DFT, the following equation is obtained:

$$\begin{aligned}
X(k) &= \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi kn}{N}} \\
X(N-k) &= \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi Nn}{N} - \frac{-j2\pi kn}{N}} \\
X(N-k) &= \sum_{n=0}^{N-1} x(n) e^{\frac{j2\pi kn}{N}}
\end{aligned}$$

Where the last equality is obtained by removing the term $e^{-j2\pi n}$ which is equal to 1. By finally comparing the first and the third equation, one can see that only a minus sign is missing. Either k is replaced by $-k$ giving:

$$X(N-k) = X(-k)$$

Either the conjugate of the whole equation is taken (needing the assumption that $x(n) \in \mathbb{R}$):

$$X(N-k) = X^*(k)$$

1.2 DFT of a (co)sine

Task 1.2.1. DFT of a 3 periods cosine

Generate a cosine sequence in Matlab with a randomly selected phase, and with a period that fits exactly 3 times in a data sequence of N = 1000 samples. Make a plot of the DFT of this sequence (amplitude and phase).

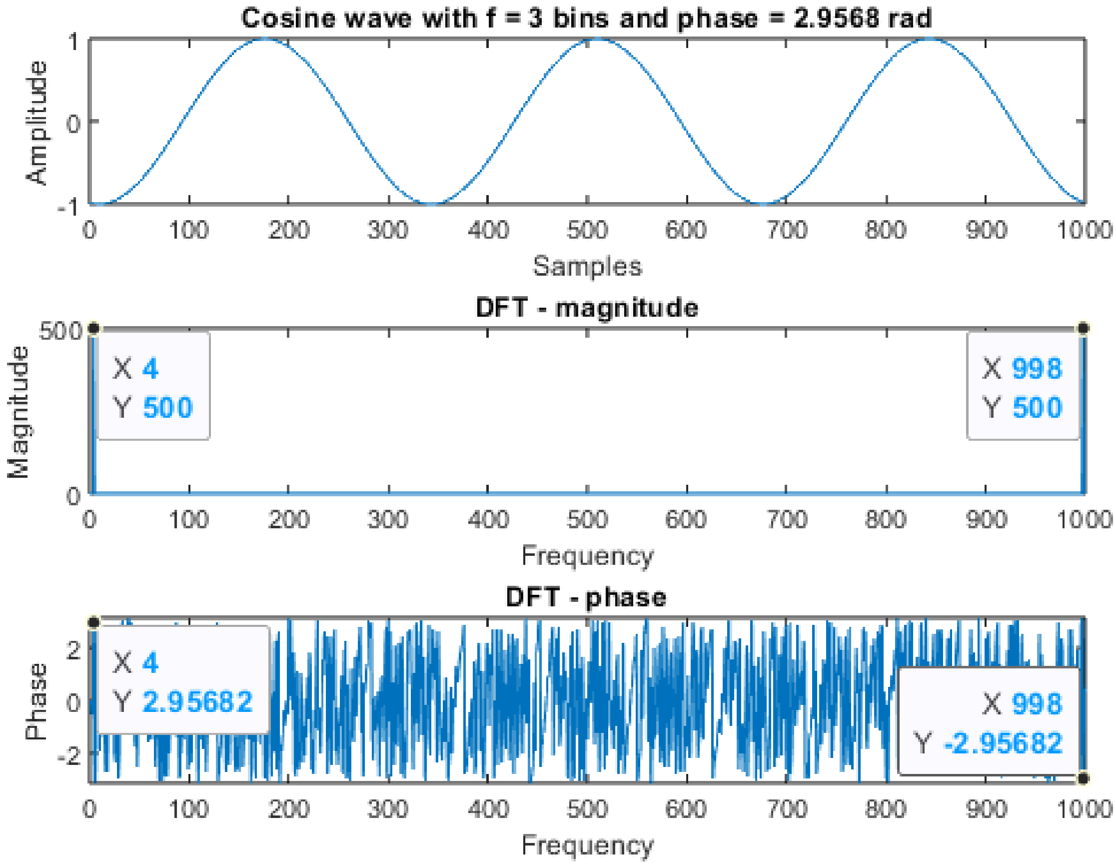

Figure 1.1: DFT of a 3 periods cosine

Remarks: from the previous section, it is known that the DFT of a cosine at a frequency of 3 bins would create a peak at the third bin and one at the N - 3 bin as $X(N-k) = X^*(k)$. There is a shift of 1 bin as matlab indices starts at 1. concerning the phase plot, the phase at the third bin is indeed the one chosen randomly. The phases at the other bins are not relevant as the cosine is not present at these frequencies. The phase at the N - 3 bin is the opposite of the one at the third bin as $X(N-k) = X^*(k)$.

Task 1.2.2. Perfect reconstruction

From the DFT plot, check that the condition for perfect reconstruction is satisfied. Is there any leakage visible?

Hint: use a logarithmic amplitude axis to distinguish small (but non-zero) values.

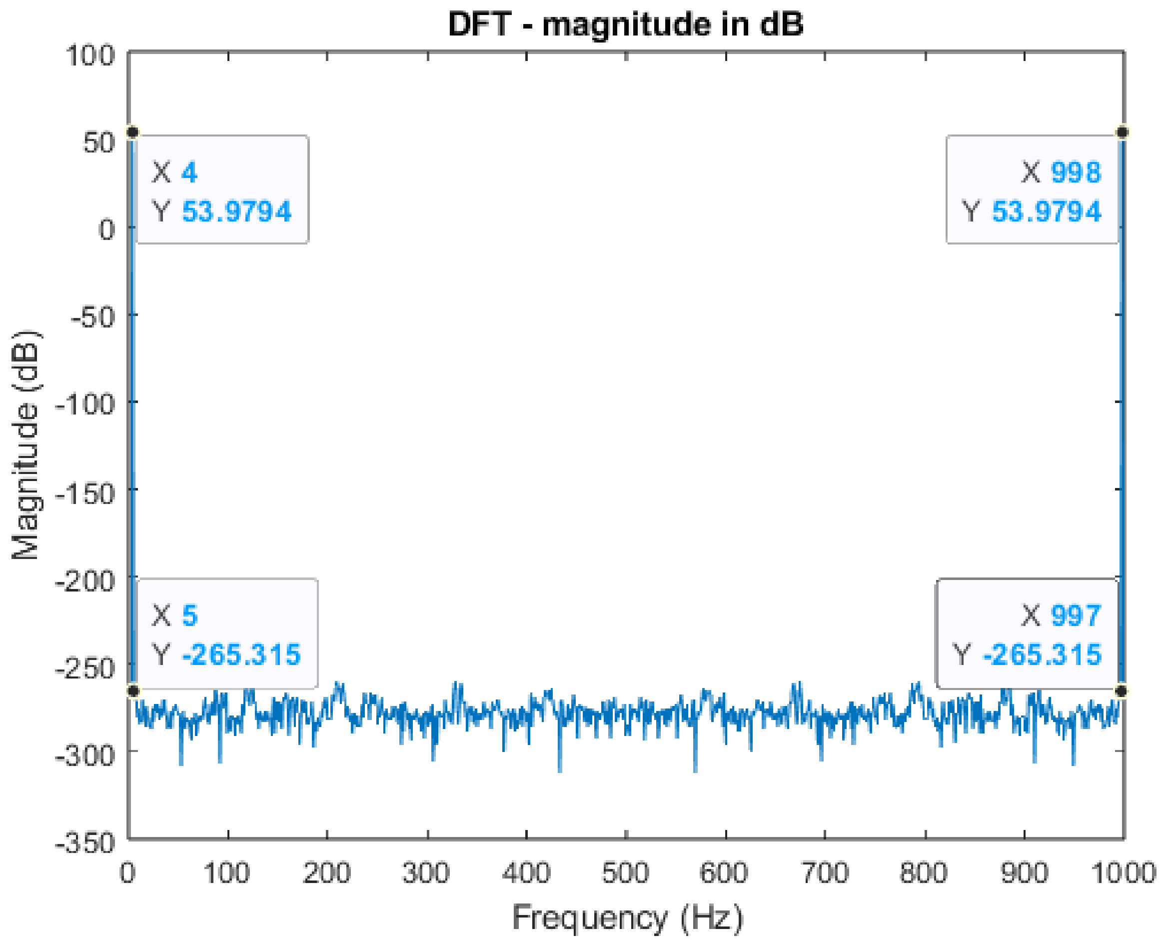

Figure 1.2: Perfect reconstruction

There is a difference in amplitude of more than 300 dB between the excited bin and its neighbors. This shows that the condition for perfect reconstruction is satisfied.

Task 1.2.3. Intrpretation of the frequency axis

At which indices of the DFT do you obtain non-zero values? Explain. (Keep in mind that Matlab indices start at 1, not at 0.)

As already discussed in task 1.2.1, the DFT of a cosine at a frequency of 3 bins would create a peak at the third bin and one at the N - 3 bin as $X(N-k) = X^*(k)$. There is a shift of 1 bin as matlab indices starts at 1. The other bins are (close to) zero as there is no excitations at these frequencies.

Task 1.2.4. Frequency axis in bins

Construct the frequency axis for the plots, expressed in bins.

It is already done in the matlab script as the x-axis of the plots is expressed in "DFT samples number", which are in fact bins.

Task 1.2.5. Frequency axis in Hz

Consider that the sample frequency is fs = 100 Hz. Construct the frequency axis for the plots, expressed in Hz.

(Hint: use the results from Task 1.1.1.)

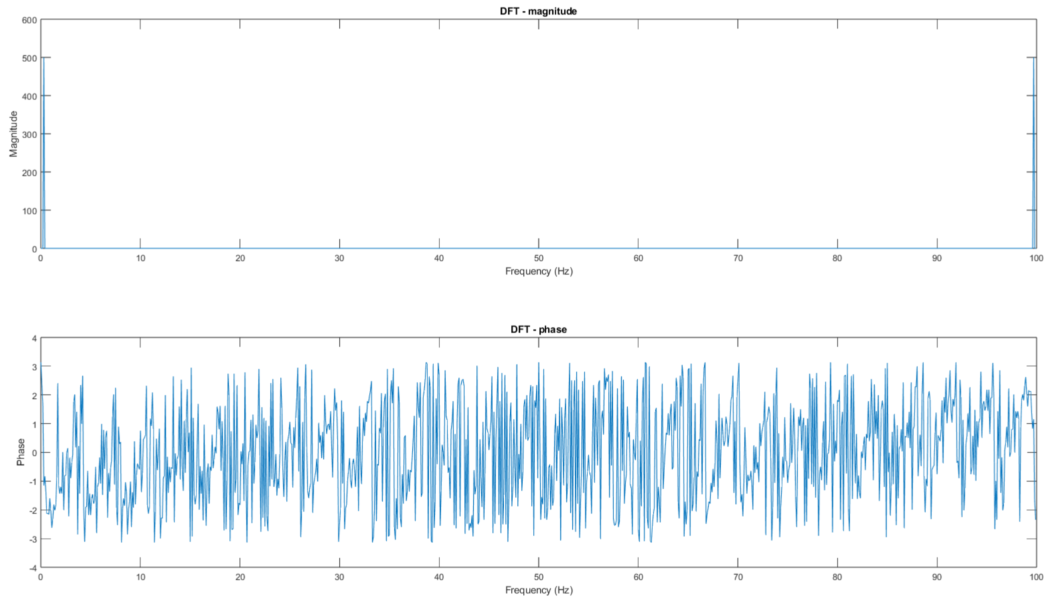

Figure 1.3: Frequency axis in Hz

To change the frequency axis from bins to Hz, it was simply multiplied by $\omega_1 = \frac{f_s}{N}$.

1.3 Time domain construction of a multisine

Task 1.3.1. Time domain random phase multisine

Generate a multisine in the time domain, by implementing (1.6), with $N = 1000$ samples and $K = 10$ excited frequencies. Set the amplitudes $A_m = 1$, and choose the phases φ_m randomly between 0 and 2π (i.e. a random phase multisine). Check that this multisine satisfies the condition for perfect reconstruction by plotting its DFT. (Use a logarithmic amplitude axis). Include the frequency axis, expressed in bin.

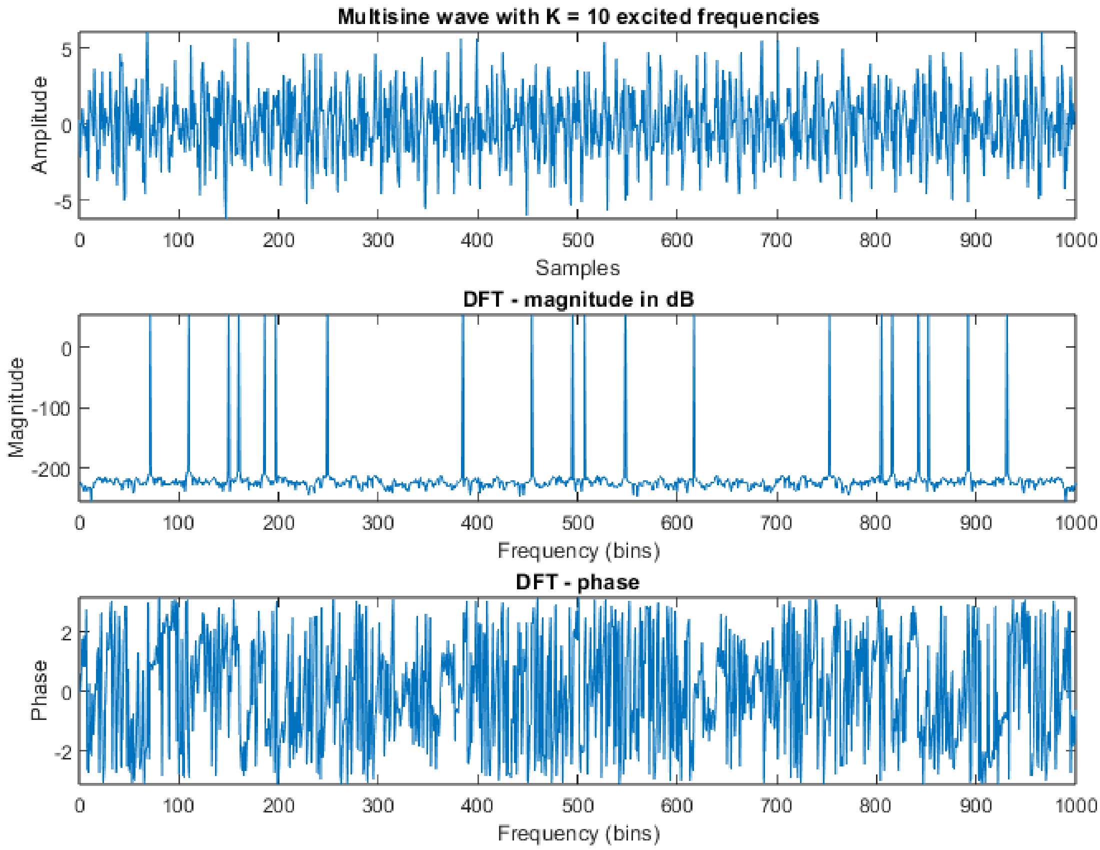

Figure 1.4: Time domain random phase multisine

The condition for perfect reconstruction is satisfied as the difference in amplitude between the excited bins and their neighbors is more than 250 dB.

Task 1.3.2. Frequency axis in Hz

For the multisine generated in Task 1.3.1, consider that the sampling frequency is $f_s = 100Hz$. Include the frequency axis expressed in Hz in the DFT plot, and the time axis expressed in seconds for the time domain plot.

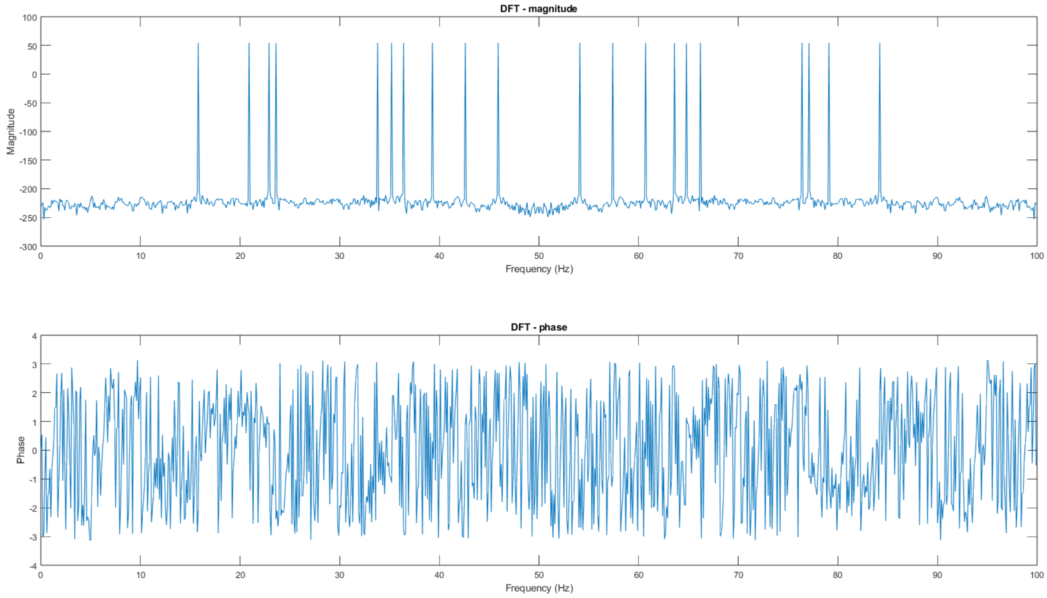

Figure 1.5: Frequency axis in Hz

As in task 1.2.5, the frequency axis was converted from bins to Hz by multiplying it by $\omega_1 = \frac{f_s}{N}$.

Task 1.3.3. Excite specific frequencies

Generate a random phase multisine with a sampling frequency of $200Hz$, with excited frequencies

$$[4, 8, 12, 16, 20, 24]\,Hz$$

Plot the time and frequency domain results, with appropriate axes.

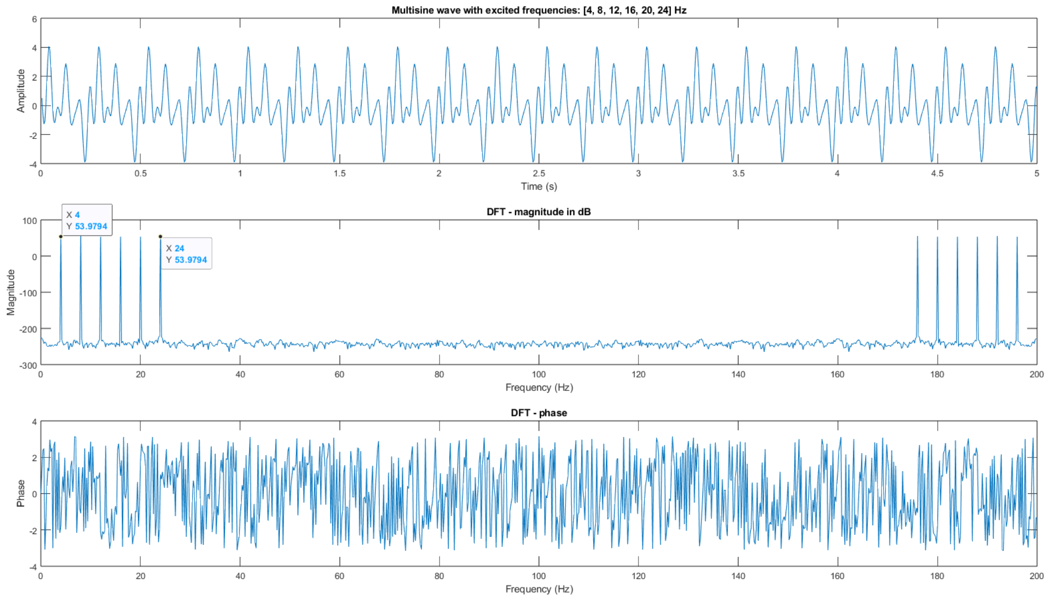

Figure 1.6: Excite specific frequencies

1.4 Frequency domain construction of a multisine

Task 1.4.1. Trick for frequency domain multisine construction

Consider the vector $\tilde{X}(k)$, such that

$$\tilde{X}(k) = A_k e^{j\varphi_k} \qquad \textbf{for } 1 \leq k \leq K$$

$$\tilde{X}(k) = 0 \qquad \textbf{otherwise}$$

Prove that

$$x(n) = N\Re\left\{\textbf{iDFT}(\tilde{X}(k))\right\} = \sum_{k=1}^{K} A_k \cos(\frac{2\pi k}{N} + \varphi_k)$$

Where $\Re$ denotes the real part.

Hint: use the definition of the iDFT

Starting from the definition of the iDFT:

$$\begin{aligned} x(n) &= \frac{1}{N}\sum_{k=0}^{N-1} X(k) e^{\frac{j2\pi kn}{N}} \\ &= \frac{1}{N}\sum_{k=1}^{K} A_k e^{j\varphi_k} e^{\frac{j2\pi kn}{N}} \\ &= \frac{1}{N}\sum_{k=1}^{K} A_k e^{\frac{2\pi k}{N}+\varphi_k} \end{aligned}$$

By then taking the real part of $x(n)$ and multiplying it by N:

$$x(n) = \frac{N}{N}\sum_{k=1}^{K} A_k \Re\left(e^{\frac{2\pi k}{N}+\varphi_k}\right)$$
$$= \sum_{k=1}^{K} A_k \cos\left(\frac{2\pi k}{N}+\varphi_k\right)$$

Task 1.4.2. Frequency domain multisine

Use the frequency domain approach to construct a random phase multisine, by using the trick from Task 1.4.1. Let $N = 1000$, and excite the first 30 bins. Make time and frequency domain plots (frequency axis expressed in bins).

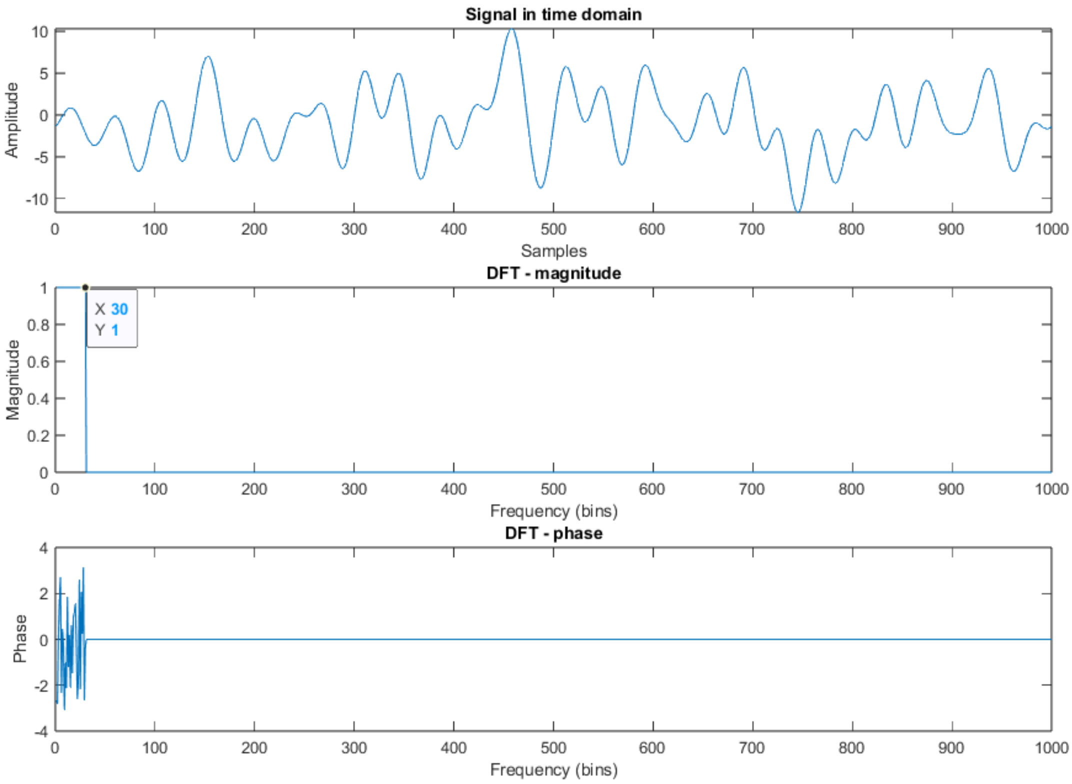

Figure 1.7: Frequency domain multisine

Task 1.4.3. Specified excited frequency band and frequency resolution

Construct a random phase multisine in the frequency domain, which excites the frequency band [5, 15] Hz at 30 equidistantly spaced frequencies. Choose an appropriate sampling frequency. Make time domain and frequency domain plots (time axis in seconds, frequency axis in Hz). How long is one period of this multisine (expressed in seconds)?

A frequency band of 10 Hz with 30 equidistantly spaced frequencies means that a bin must be equal to

$1/3Hz$ (or a divider of it). Using the formula of $\Delta_f = \frac{f_s}{N}$ (where Δ_f is the frequency resolution) and using a sampling frequency f_s larger than twice the maximum signal frequency:

$$f_s = 50Hz \qquad N = 3f_s = 150$$

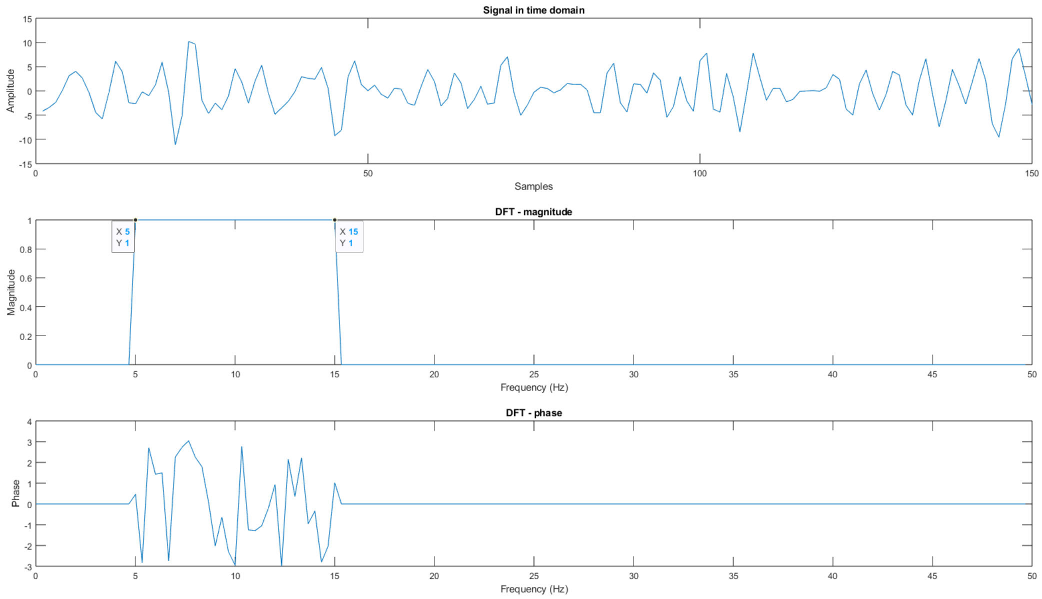

Figure 1.8: Specified excited frequency band and frequency resolution

Because it was built with the perfect reconstruction condition, the period of the multisine is equal to the period of a sine at the frequency resolution, which is $1/3Hz$ so the period is 3 seconds.

1.5 Influence of the phase of the multisine

Task 1.5.1. Crest Factor

Construct a multisine, with N = 500, with the first K = 60 bins excited, and with the following phases:

- **random phase: chosen randomly in** $[0, 2\pi]$ **(uniform distribution)**
- **Schroeder phase:** $\varphi_k = \frac{m(m+1)\pi}{K}$
- **Linear phase:** $\varphi_m = m\pi$

Make time and frequency domain plots (in samples and bins), and compute the Crest Factors. Describe, qualitatively, the relationship between the time domain plot and the crest factor. What is the advantage of a low/high crest factor?

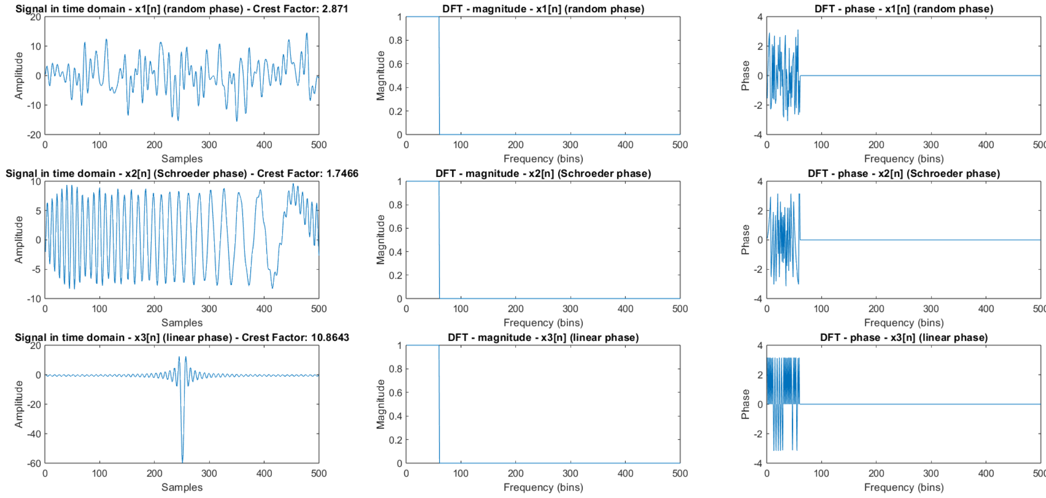

Figure 1.9: Crest Factor

Based on the figure, it is quite clear that a larger Crest Factor corresponds to a signal that has peaks of larger amplitude ($\max|x_3| \approx 6 \times \max|x_2|$). This is easily understandable as the Crest Factor is defined as the ratio between the peak value and the RMS value of the signal. The advantage of a low Crest Factor is that the signal has no big peaks, which could damage an unprotected electronic system receiving it.

1.6 Random noise signals

Task 1.6.1. White Gaussian random noise

Generate a normally distributed (Gaussian), random, white noise sequence of N = 1000 samples, by using the Matlab function randn. Make time and frequency domain plots (axes in samples and bins). Observe that all the bins are excited, with random amplitudes and phases.

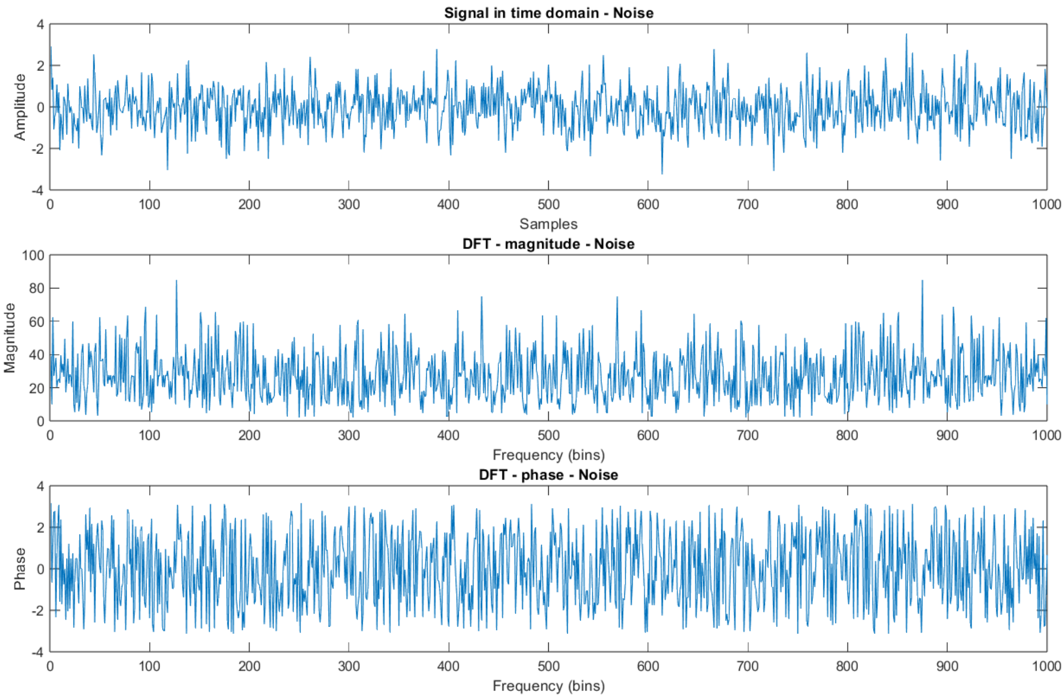

Figure 1.10: White Gaussian random noise

Task 1.6.2. Filtered random noise

Generate a filtered random noise sequence with N = 1000, sampling frequency 100 Hz, from a Gaussian white noise sequence (use randn). Do this by using the function cheby1 to create a low-pass digital Chebyshev filter of order 5, ripple 2 dB, and such that the passband edge lies at 5 Hz. Filter the sequence by using the function filter. Make time and frequency domain plots (axes in seconds and Hz), and check that the excited frequency band is as expected. What do you observe in the stop-band of the filter? Is it equal to 0? Explain.

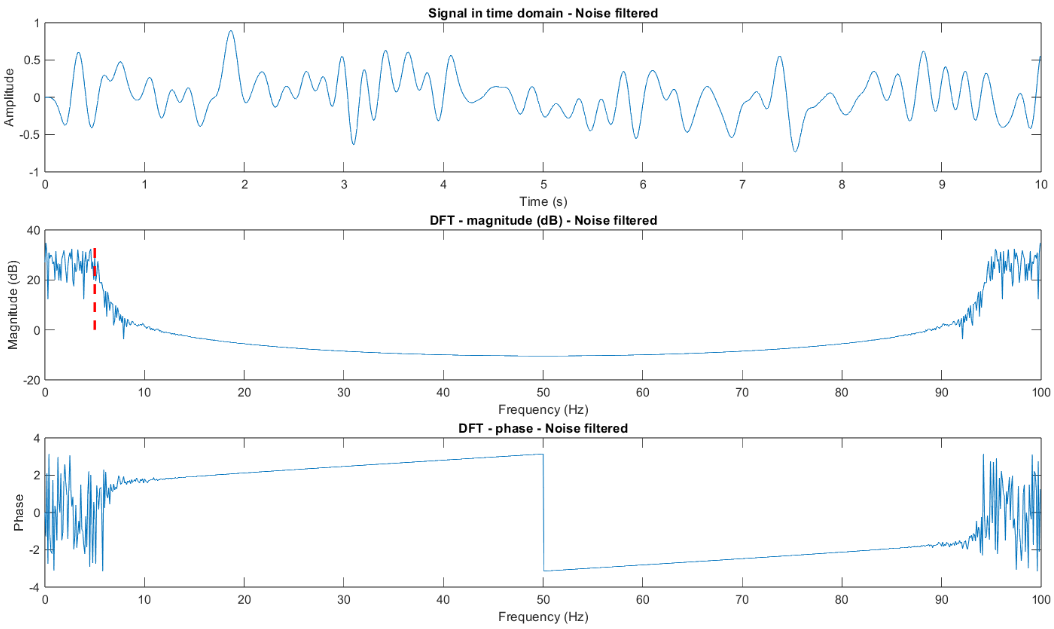

Figure 1.11: Filtered random noise

The excited frequency band is indeed reduced to 5 Hz as expected. In the stop-band of the filter, the amplitude is not equal to 0. This is due to the fact that the filter is not an ideal filter, and thus does not completely remove the frequencies above 5 Hz. The amplitude after $10Hz$ is already reduced by $30dB$ and the increase for higher bins is due to the symmetry of the DFT proved in task 1.1.3.

Task 1.6.3. Periodic band-limited random noise

Generate a Gaussian random noise sequence (use randn), with N = 1000 and sampling frequency 100 Hz. Compute the DFT, and set the DFT at all frequencies beyond 5 Hz to zero:

$$\tilde{X}(k) = 0 \qquad \textbf{for } \omega_k \geq 10\pi \textbf{ rad/s}$$
$$\tilde{X}(k) = X(k) \qquad \textbf{otherwise}$$

and use the expression

$$x(n) = 2\Re\left\{\textbf{iDFT}(\tilde{X}(k))\right\}$$

to obtain the time sequence. Make time and frequency domain plots (axes in seconds and Hz). If you repeat this time domain sequence x(n) (by putting multiple copies of the sequence after each other), and computing the DFT of the result, no leakage should occur. Check this, and explain why this is the case.

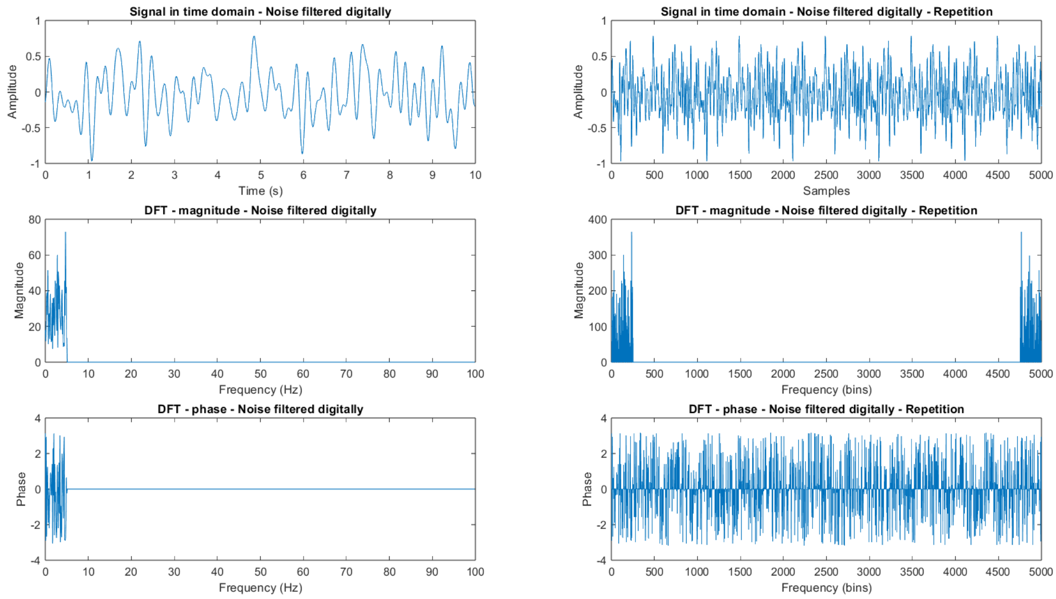

Figure 1.12: Periodic band-limited random noise

There is no leakage when repeating the **idft** of the filtered noise as the condition on perfect reconstruction is fulfilled. As the signal that is repeated is built from the frequential domain, it will have an integer number of periods in the time domain, and thus when replicated no leakage will occur as it will still have an integer number of periods inside the window of the **dft**.

1.7 Set the Root-Mean-Square of the signal

Task 1.7.1. RMS value

Set the RMS value of your favourite signal from the previous tasks to $RMS_{des} = 3$, as follows:

$$x_{des}(n) = x(n)\frac{RMS_{des}}{RMS(x)}$$

Prove that thr RMS of $x_{des}(n)$ is indeed RMS_{des}.

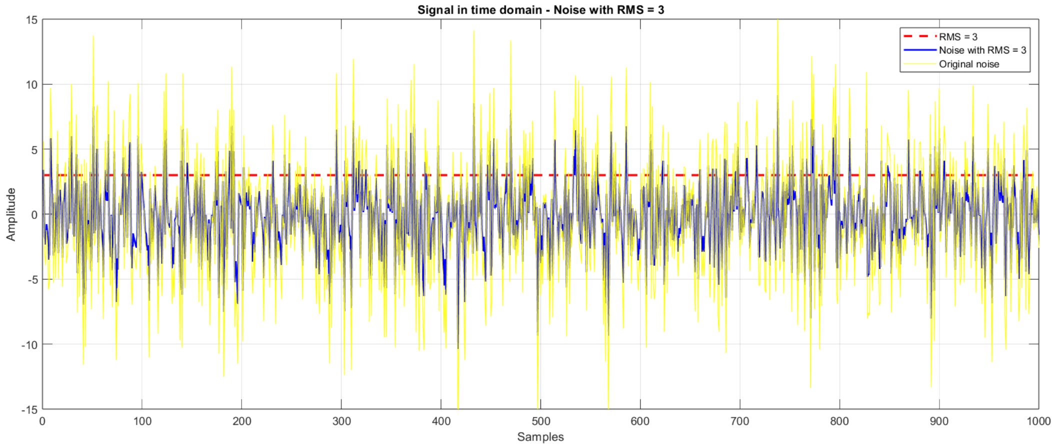

Figure 1.13: RMS value

Note that the title of the plot contains the RMS value of the modified signal and it indeed reaches 3.

2

Lab 2

2.1 Acquisition and Generator

Task 2.1.

Synchronization If the clocks of the generator and the acquisition are not synchronized, which effect can you expect? Is this equally important for low and high frequencies?

The potential concerns are of 2 kind :

- The first one is the aliasing if the acquisiation clock is slower than the generator clock
- The second one is a phase shift between the input and the ouptut which may lead to misleading results for the FRF.

For low frequency signal, those 2 factors are less significant than the case of high frequency signals.

Task 2.2.

Dynamic range What is the dynamic range of the ADC used in the ELVIS II board?

Considering the ADC got a resolution of 10 bit, the maximum quantization is calculated as follow :

$$\text{Quantization} = 2^{10} = 1024 \text{ levels}$$

The LSB is the minimal voltage difference between 2 folliwing voltage measurements.

$$\text{LSB} = \frac{V_{\max}}{1024}$$

This reults as the dynamic range defined as

$$\text{Dynamic range} = 20 * log(\frac{V_{\max}}{\frac{V_{\max}}{1024}}) = 60 \text{ dB}$$

2.2 Sampling

Task 2.3.

What is the minimum sampling frequency required to perfectly reconstruct a sinusoidal signal with a frequency of 10 Hz?

Taking the Nyquist limit into account, the sampling frequency should be set at a minimum of 20 Hz. By security and considering intermodulations/harmonics, the limit is set at 10 times the signal frequency : 100 Hz.

2.3 Phase and impact on signal behaviour

Task 2.4.

Explain why the accuracy of the FRF is affected by the phase spectrum.

By only considering the power spectrum of the FRF, we can realize that the time delay / shift introduced by the system between the input and the ouptut cannot be taken into account which could result to a mis-reconstruction of the output time signal.

Task 2.5.

Generation of multisine signals Use MATLAB to generate the data sequence of a multisine that consists of 4096 time samples and contains 100 excited spectral lines, located at the low end of the band (from line 1 to 100). Generate the multisine with

- **(a) a constant phase spectrum**
- **(b) a random phase spectrum, with phases uniformly distributed between [0, 2 pi]**
- **(c) a Schroeder phase spectrum. Choose the phases according to**

$$\phi_k = \frac{k(k+1)\pi}{K}$$

where k is the line number (= the frequency of the line expressed in bins) and K is the number of excited lines in the signal.

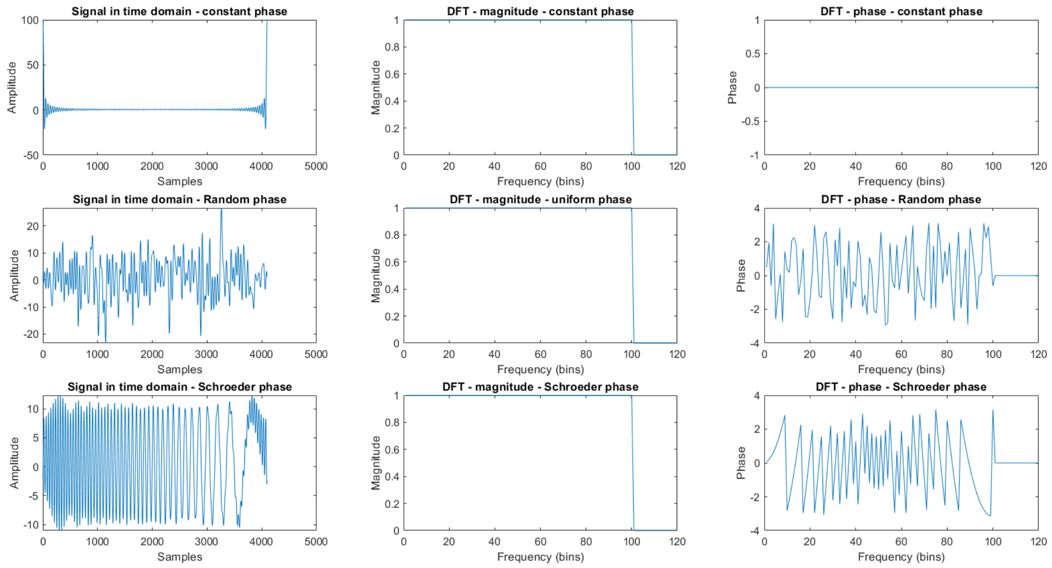

Figure 2.1: Multisine generation with different types of phase definition

Task 2.6.

Crest factor Calculate the crest factor of these signals and explain the differences.

Please find below the crest factor calculated for each signal generation:

Crest factor - constant phase :14.072

Crest factor - random phase :3.7526

Crest factor - Schroeder phase :1.7367

We can see that the random phase signal and the Schroeader phase signal have the lowest crest factor. This comes from :

- (a) Sub-signal phases being desconstructive for the random phase signal resulting to no peak generation in temporal.
- (b) Optimized power destribution in temporal domain due to the Schroeder signal definition

In contrary, the constant phase signal leads to constructive sub-signals which results to a peak in temporal domain (constructive sub-signals).

Task 2.7.

Plotting and interpretation Visualize the signals both in the time and the frequency domain then discuss differences and similarities.

Regarding the plot of the signal in temporal and frequency domain, this can be found in the point

2.1 The interpretation is quite straightforward, the constant phase is genrating sub-signal with the same phase leading the constructive sub-signal reuslting to a peak. In contrast, the random phase signal and the Schroeder signal are exciting the system with multiple phase signal (please find the DFT plot) leading to a most suitable optmized power distribution.

2.4 Transient

Task 2.8.

Differences between periods Compare the measurements of the different periods of the signal in the time domain and calculate the FRF for each period of the signals separately. Explain what you see and decide which periods to select or leave out and why.

The important component to consider while taking the measurement is the response transient. We should be sure to have enough repetition to minimyze the transient component of the signal. To ensure it, we have to compare the variantion between the current ouput and the previous one. By comparing the varaition, we filter the output data to keep in order to compute the FRF of the signal.

On the figure below 2.2 , we can obeserve the FRF computed at the first repetition and the FRF computed at the 8th one.

Important note The varaition of output signal on the figure 2.3 is highlighted in red. We can obeserve the variation is more or less the same at the first repetition and the last one. This may be due to the high reaction speed of the evaluated system meaning that the transient is very small even at the first repetition (to be confirmed with the TA).

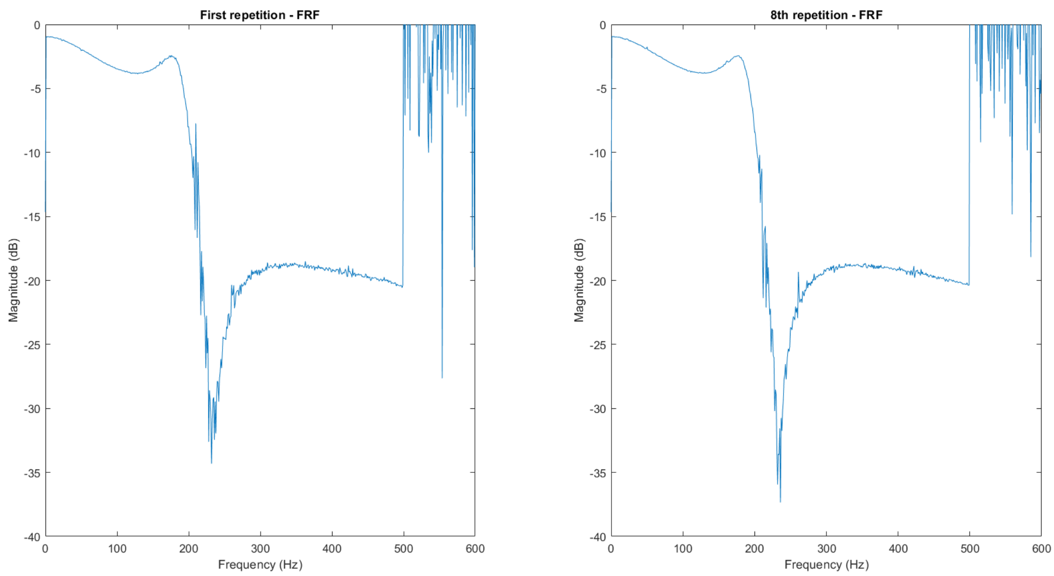

Figure 2.2: FRF at the first and 8th repetition

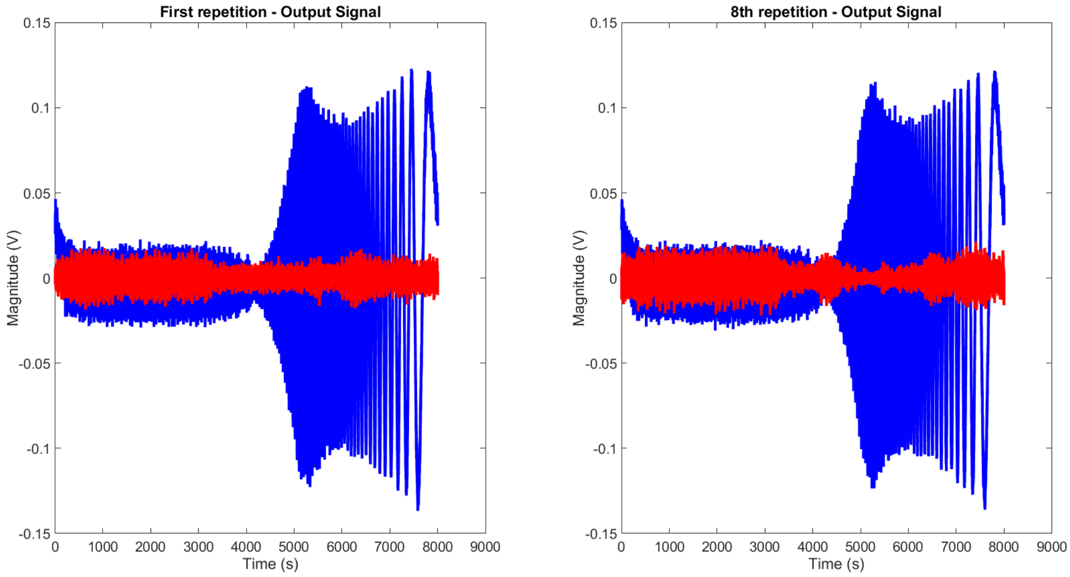

Figure 2.3: Output signal at the first and 8th repetition

Task 2.9.

Frequency resolution Visualize the spectra of input and output signals. What does this measurement show? Focus on the bandwidth where all the important features of the system are included and modify the frequency resolution to improve the representation of these features. Fix the frequency resolution for the remainder of this lab.

The figure 2.4 shows the power spectrum of the input and the output signal. We can obeserv easily that the excited band gives us information about the system behaviour at this specific frequency band. At low frequency, either an very small amplification of the signal or a constant signal power can be observed. In contrary, at higher frequency, above 150 Hz, we can see a degradation of the signal. The system behaves like a low pass filter with a cut off at 150 Hz.

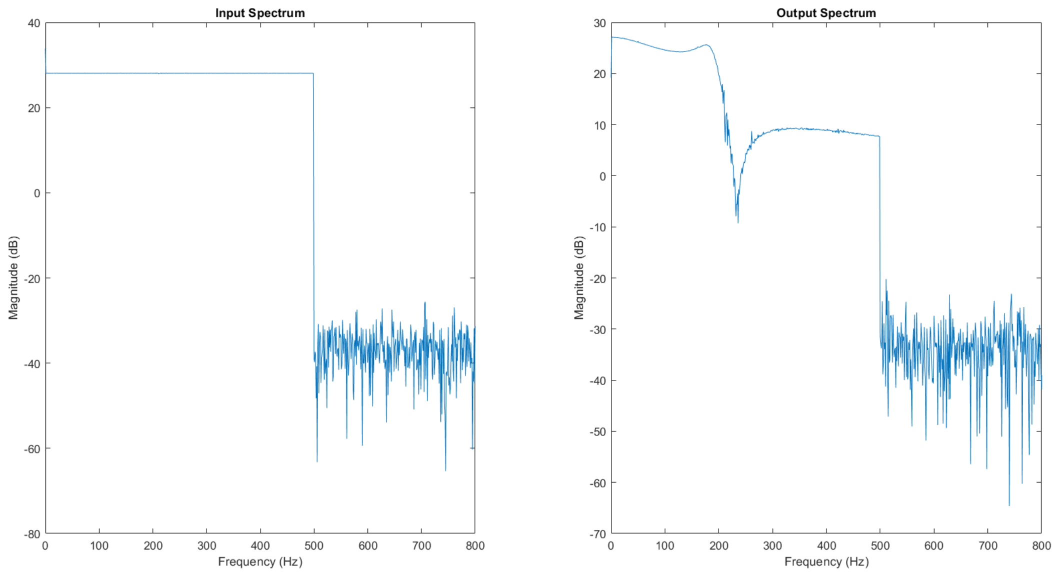

Figure 2.4: Power spectrum of the input and output signals

Task 2.10.

FRF computation For each measurement compute the FRF and provide a plot in the report. Which differences do you observe? Explain. Periodic

- **(a) The Schroeder multisine designed in the previous section.**
- **(b) A multisine with constant phase and amplitude in the specified analysis band only.**
- **(c) A multisine with an arbitrary (random) phase and constant amplitude in the specified analysis band only.**
- **(d) A periodic noise signal.**

Aperiodic

- **(e) An aperiodic noise signal. Remember that here measuring P periods requires to load P different signals in the AWG.**
- **(f) Same as (e), but multiply the measured aperiodic input and output signals by a Hann window and observe the results. How are the results in comparison to the results with a rectangular window?**

Periodic

For periodic signals, we can obeserve differences depending on if the signals have constant phase, random phase or schroeder phase.

- (a) The constant phase signals lead to a very clean FRF power spectrum. The cleaniless of the FRF is due to the constructive sub-signals (same phase) leading to a high signal power. The drawback of this excitation is the non-linearity / harmonics distorsion if the system is non-linear (which is not the case of this TP).

- (b) The random phase signals lead to a more noisy FRF power spectrum (less sharp). Nevertheless, this excitation is more robust once we deal with system introducing noise or non-linearity (cf Crest factor).

- (c) Schroeder phase signals are the most optimiwed signal to compute a clean FRF due to the optimized energy distribution across the frequency band. The orthogonal characteristic of the frequency components allow to have no interference between frequencies which would result to distorsion of the FRF.

- (d) the periodic noise signal results to the noisiest FRF as excpected due to his random nature. Nevertheless, by repetiting the signal, the FRF computed is still leading to relevant result due to the minimization of the transient after each period.

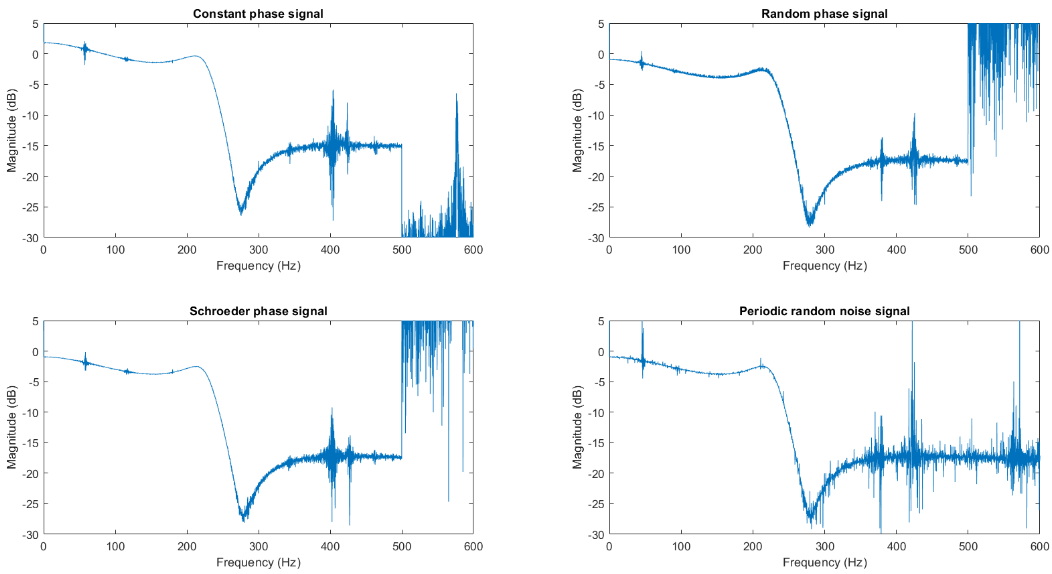

Figure 2.5: Power spectrum of FRF

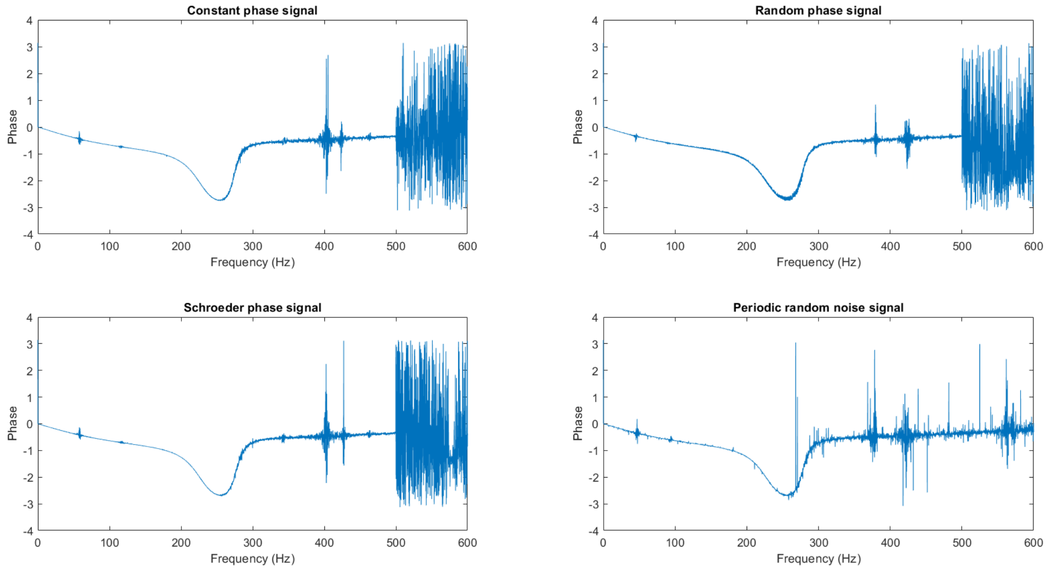

Figure 2.6: Phase spectrum of FRF

Aperiodic

Regarding aperiodic signals, the wrong manipulation leads to invalid results. Those results will be redone during week 28.

3

Lab 3

Question 3.1. Length data records

The FFT algorithm works faster when the length of the data records N is a power of 2. Do you know why?

To numerically implement a DFT, the DFT algorithm is used. It works by dividing the input signal into two parts, each of which is transformed separately, and then the results are combined and this is done recursively. It means that a number of samples that is a power of 2 will be perfectly divided at each step, leading to a faster computation.

Question 3.2. Trigger signal

What is the purpose of a trigger signal? A number of processing methods were discussed in the theory to improve the SNR of the FRF by averaging of measurements. Which methods require a trigger signal to operate properly? Explain what happens if the trigger is absent while needed.

For the averaging in the time domain and the averaging in the DFT spectra, the input and output signals need to be identical:

$$H_{\text{time}}(j\omega_k) = \frac{\mathbf{DFT}\left(\frac{1}{M}\sum_{i=1}^{M} y_i(n)\right)}{\mathbf{DFT}\left(\frac{1}{M}\sum_{i=1}^{M} u_i(n)\right)}$$
$$H_{\text{DFT}}(j\omega_k) = \frac{\frac{1}{M}\sum_{i=1}^{M} \mathbf{DFT}\left(y_i(n)\right)}{\frac{1}{M}\sum_{i=1}^{M} \mathbf{DFT}\left(u_i(n)\right)}$$

For the measurements to be identical between two repetitions (except for the noise of course), they should always start at the same point in the period of the excitation and the trigger signal make sure it is the case.

Question 3.3. Frequency domain multisine

In your report, show the Matlab code you used to construct the multisine signal (with random or Schroeder phase) in the frequency domain. Make sure that the code is sufficiently commented to improve its readability.

```
fs = 16000;                  % Sampling frequency
excFreq = [4 1000];          % Excited frequencies in Hz
rmsVal = 0.5;                % RMS value
N = 4000;                    % Number of samples

m = (excFreq(1)*N/fs:excFreq(2)*N/fs);
                             % Excited frequencies in bins
K = length(m);               % Number of excited frequencies

shroedPhase = m.*(m+1) * pi / K;
signalFreq = zeros(1, N);

signalFreq(m + 1) = exp(1j*shroedPhase);
                             % m + 1 because the index starts from 1 in MATLAB

                             % Generate the signal
signalTime = 2*N*real(ifft(signalFreq));
                             % Normalize the signal
signalTime = signalTime*rmsVal/rms(signalTime);
```

Question 3.4. Averaging the measurements

Which averaging techniques are applicable to which excitation signals? Explain.

The time averaging and the frequency averaging can be applied with the periodic excitation only. The issue with a random input is that averaging it could still result in some bins with a close to zero amplitude, wich would create spikes in the FRF. It is worth noting that these two methods are equivalent (at least theoretically, in practice there are some differences due to the way they are implemented).

Computing the FRF of each repetition separately and then averaging them could also lead to a division by zero, making it unusable for the random excitation.

Finally, using the average of the auto-power of the input/output is a method that can be used for any input signal as it divides by a power that is not zero. It is preferred to use the average of the auto-power of the input signal as it is less noisy than the output signal. This is because de DUT is a filter and the output signal is lower in amplitude than the input signal, leading to a smaller SNR. It must be noted that these methods introduce a bias. This bias however is null if the additive noise is circular complex.

Question 3.5. Plots

Provide relevant plots of the estimated FRF, obtained with the different methods, with the different excitation signals and with a different number of averaged records.

On the 3 first figures, the aperiodic FRF is "too clean". It should have more spikes as explained in question 3.4.

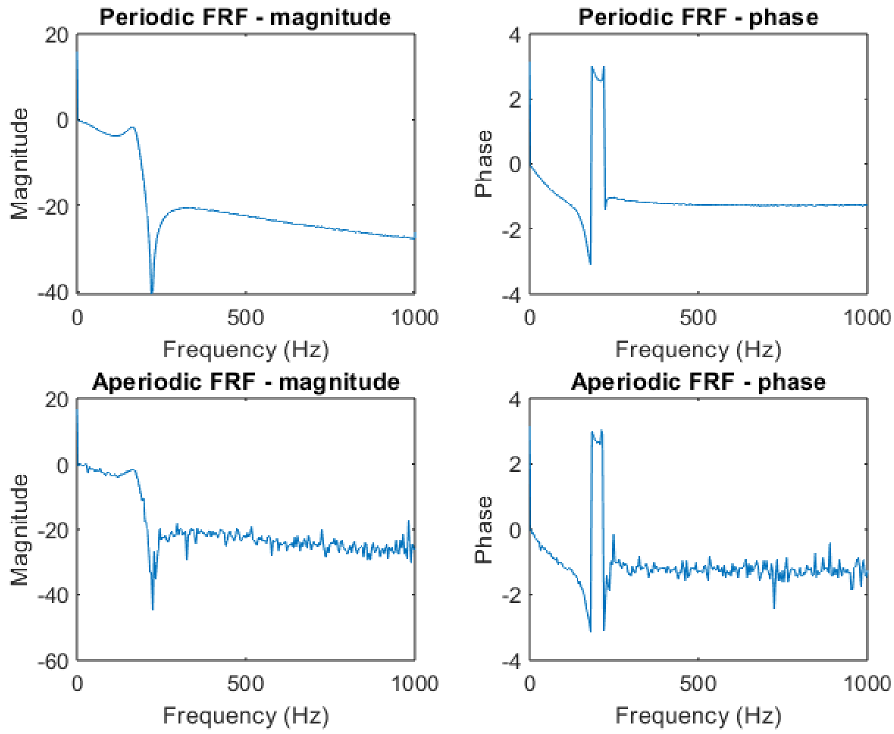

Figure 3.1: Estimated FRF with time averaging

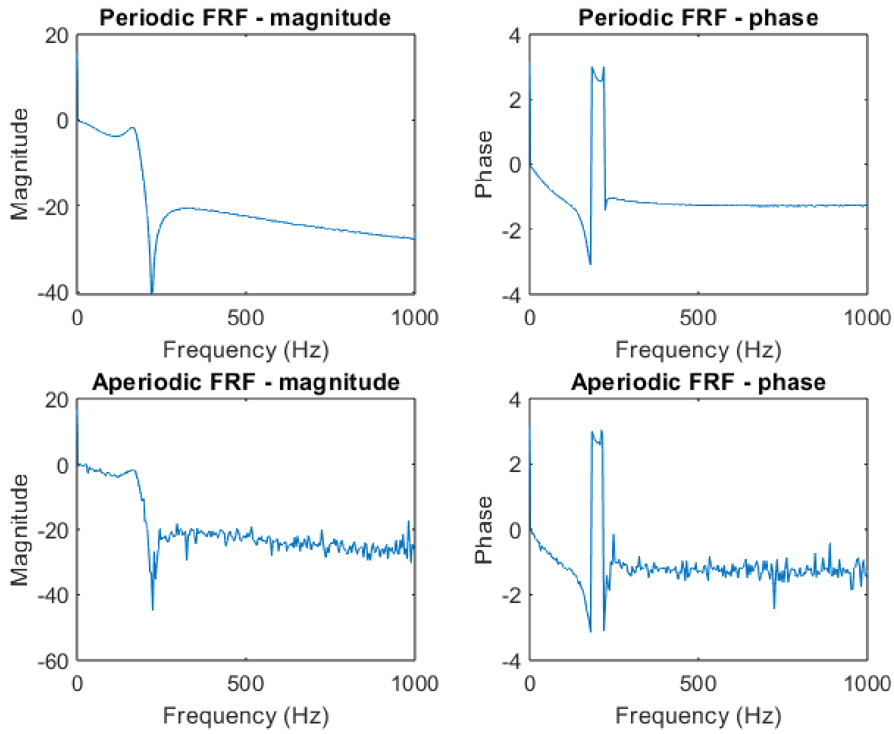

Figure 3.2: Estimated FRF with frequency averaging

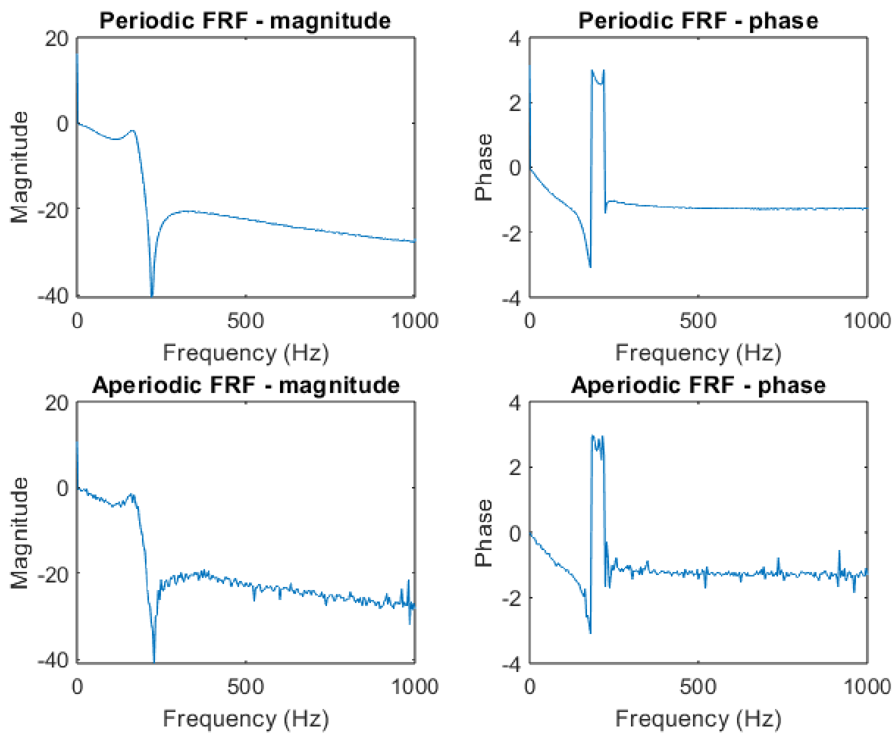

Figure 3.3: Estimated FRF by averaging the FRF

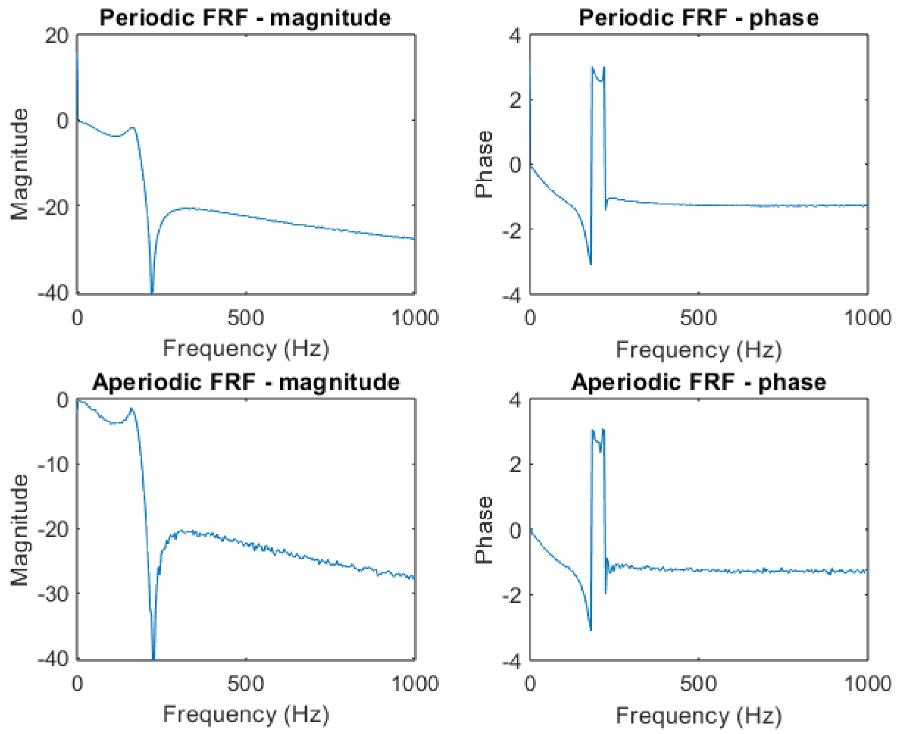

Figure 3.4: Estimated FRF by averaging the auto-power of the input signal

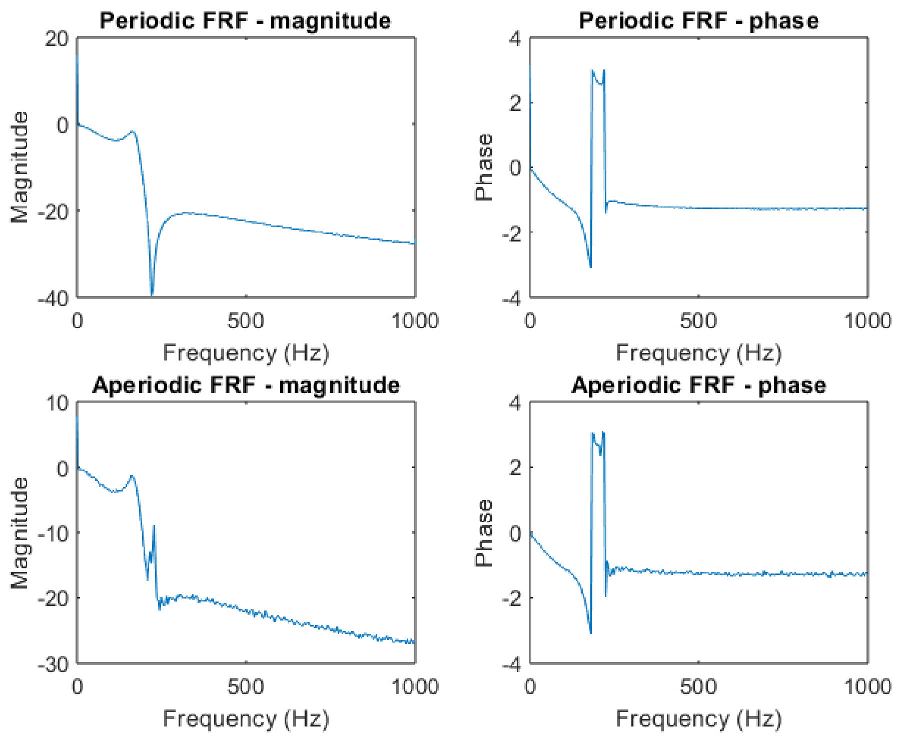

Figure 3.5: Estimated FRF by averaging the auto-power of the output signal

Here is the effect of the number of averaged records for the time averaging method:

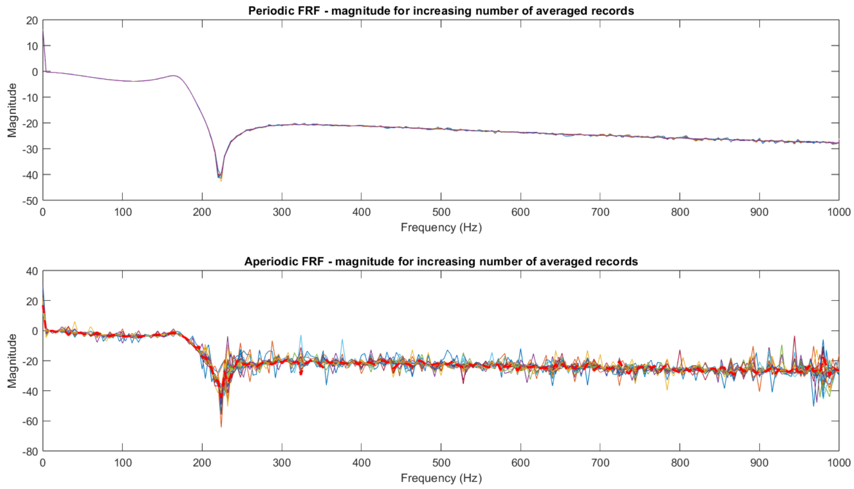

Figure 3.6: Estimated FRF for different number of averaged records

Because the transient has been removed by discarding the first 8 periods, the estimated FRF based on the periodic signal does not change. This indicates that the noise power was extremely small compared to the signal power (we might have accidentally saved the response of a noiseless test). The estimated FRF based on the random input is however more interesting: the last estimation (the red dotted line) has way less spikes than all the previous ones. It really shows the benefit of averaging the measurements.

Question 3.6. Impact of repetition number

Determine the effect of the number of averaged records on the variability of the averaged result. Compare the standard deviation.

By computing the different std's for different number of averaged records, we see that there is no major difference. This is very surprising and it proves even more that we might have kept a noiseless response. What should have happened is that the std should have decreased as the number of averaged records increased.

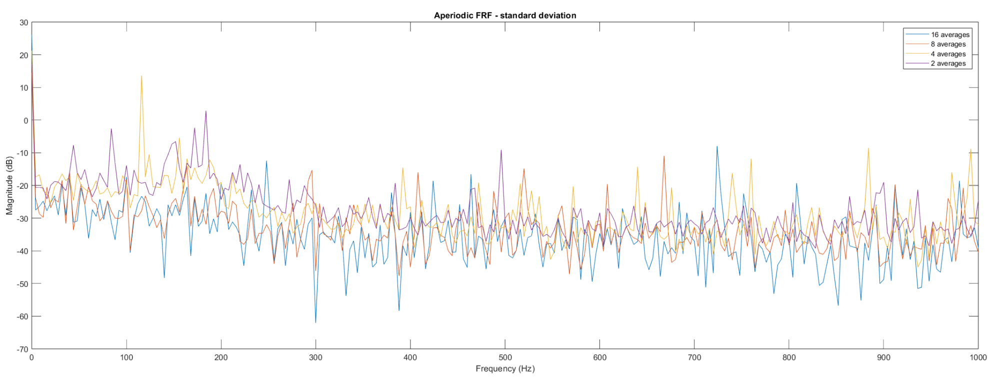

Figure 3.7: Standard deviation of the estimated FRF for different number of averaged records using the time averaging method and the aperiodic excitation signal

> *Question 3.7. Discussion*
>
> **Discuss the differences and explain according to you, which will deliver the best result. Discuss the pros and cons of each excitation signal.**

This question has already been partially answered in Question 3.4. The first 3 methods should be used only with known excitations as they do not support close to zero excitations. The last two methods can be used with any excitation signal and are unbiased only when the noise is circular complex. The main advantage of using a random excitation (so one of the power averaging methods) is that they give an FRF of the whole spectrum whereas using a deterministic excitation gives result only for the excited frequencies. It can be seen by comparing the whole measured FRF with the input power method:

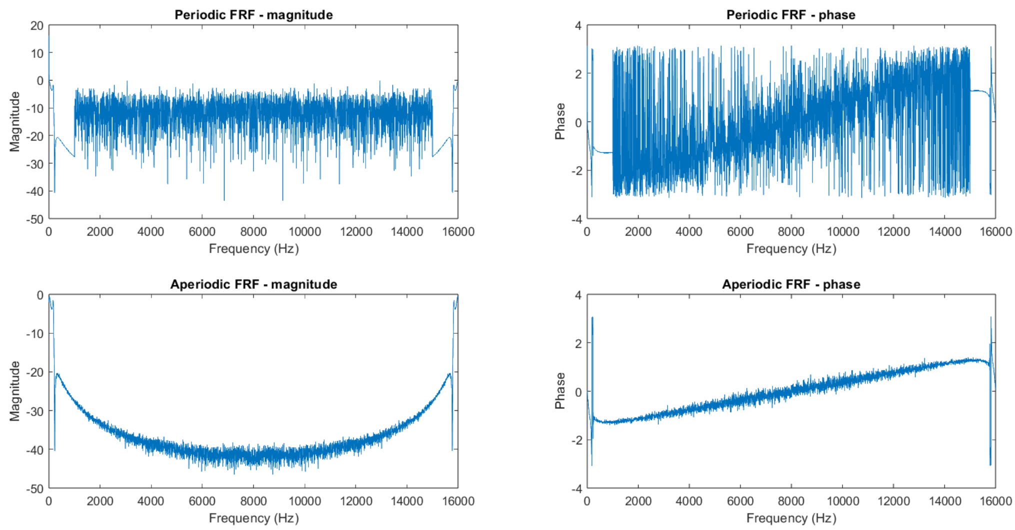

Figure 3.8: Estimated FRF by averaging the auto-power of the input signal

The FRF is indeed unusable for the periodic excitation signal above 1000 Hz, the upper limit of the excited frequencies.

4 Lab 4

Question 4.1. harmonic and intermodulation distortion

Consider a static nonlinear system whose response is given by:

$$y(t) = u(t) - \frac{1}{2}u^3(t) - \frac{1}{4}u^4(t) \tag{4.1}$$

At which frequencies will energy appear at the output when the input is excited by the sum of 2 sinewaves, one at frequency 4Hz and one at frequency 11Hz?

using matlab, each excited frequency has been plotted on figure 4.1. The different amplitudes are not correct, they simply give the order of magnitude of each frequency component. A more exact simulation with the fft of the output signal is shown in figure 4.2.

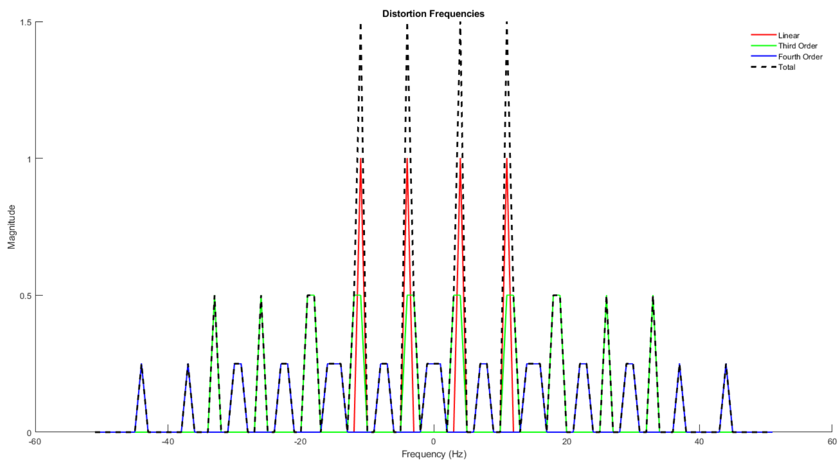

Figure 4.1: Excited frequencies - Approximation

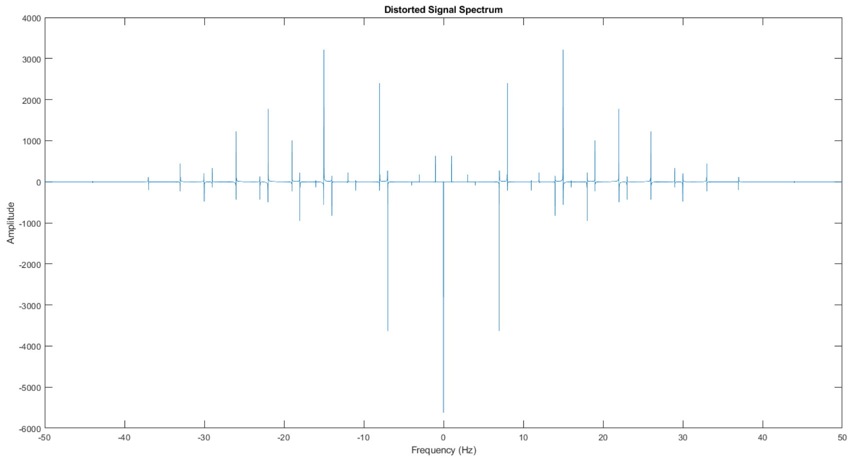

Figure 4.2: Excited frequencies - Exact

Question 4.2.

Design an excitation sinewave with a frequency of 100 Hz and an amplitude of 1 V. Plot the input signal and the output signal of the DUT in the frequency domain. What do you observe? Give a list of all possible solutions to get rid of this behaviour. Design experiments to show that the proposed solutions do indeed work.

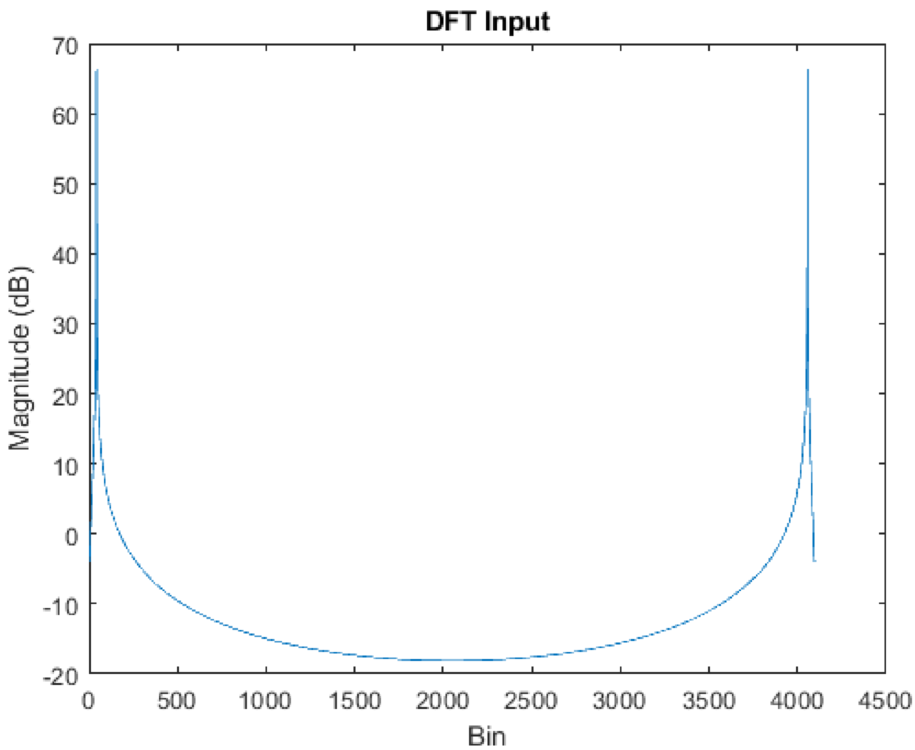

Figure 4.3: Initial sine wave DFT

The DFT of the sinusoidal input signal is clearly not a Dirac pulse as the chosen frequency is not at an integer bin index.

The size of a bin being the inverse of the signal duration N/f_s, we want

$$f_{\text{signal}} = k \cdot \frac{f_s}{N} \quad \text{with} \quad k \in \mathbb{N}$$

This is visible as there is leakage in the DFT plot. There are 3 ways of solving this:

- Changing the sampling frequency $\rightarrow f_s = 9990.25\text{Hz}$
- Changing the number of samples $\rightarrow N = 4100$
- Changing the sine frequency $\rightarrow f_{\text{signal}} = 100.1\text{Hz}$

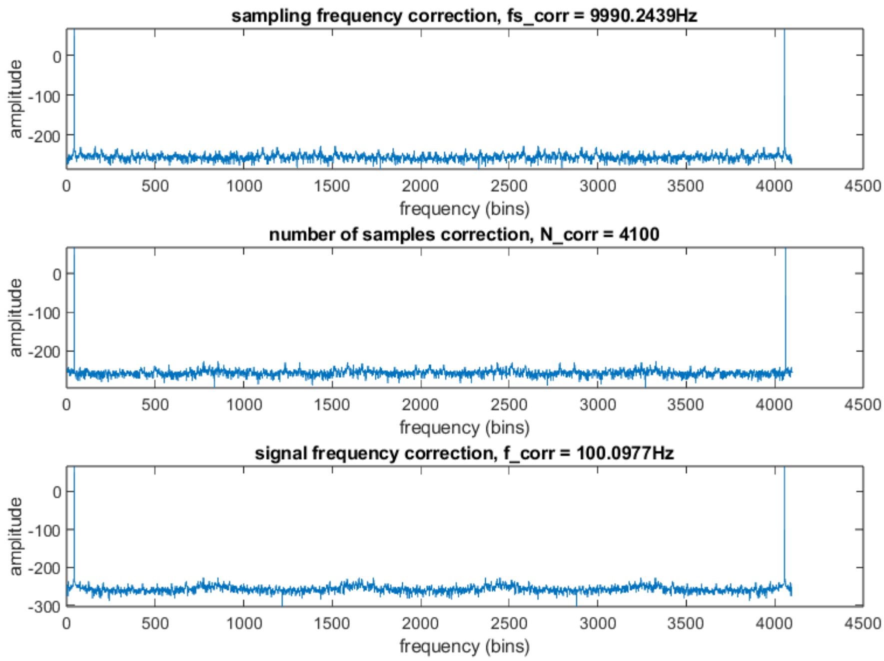

Figure 4.4: Corrected sine wave DFT

Question 4.3.

Change the excitation frequency to a frequency in the neighborhood of 100 Hz in order to solve the problem in the previous step. Use 10 different amplitudes that are spaced logarithmicaly from 100 mV to 1.1 V (logspace command in MATLAB]. Plot the DFT of the output for each amplitude. What are the frequencies at which you expect distortion to appear? Are they all present in reality? Explain.

Question 4.4.

Plot the measurements from the previous question on an input power – output power plot. Do you see compression or expansion? From this plot, estimate the 1 dB compression or expansion point. Generate a single sine wave with the corresponding input amplitude to see how accurate your estimate is.