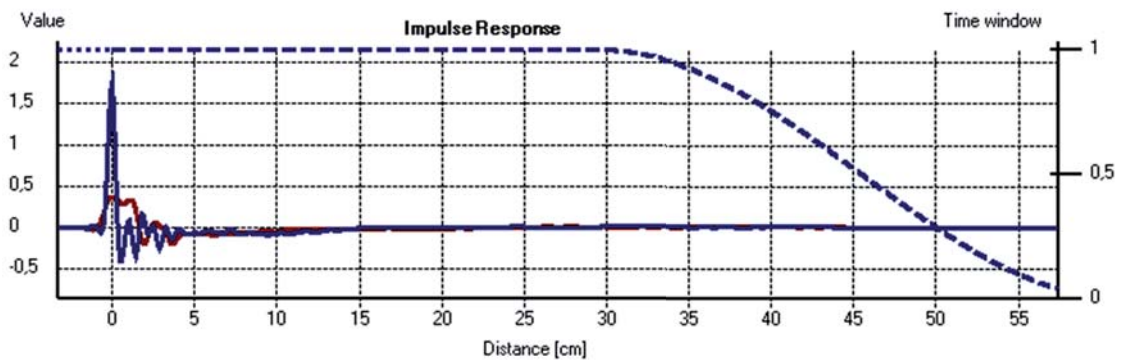
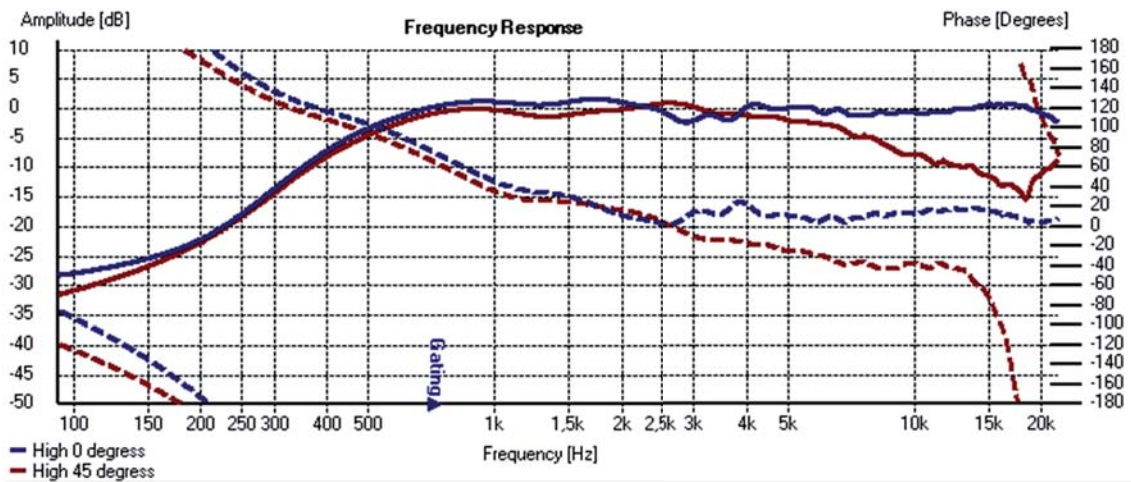


# HOLMImpulse User Guide



Document version	0.0.8 (October 23, 2009)
Compatible HOLMImpulse versions	1.4.x.x

# Contents

<b>Contents</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
1.0.1 HOLMImpulse and DSPreLab? . . . . .	1
1.0.2 HOLMImpulse @ diyAudio.com . . . . .	1
<b>2 Navigating the Response Graphs</b>	<b>2</b>
2.1 Adjusting the heights of the areas . . . . .	2
2.2 Coordinates for the cursor position . . . . .	3
2.3 Adjusting the Ranges . . . . .	3
2.3.1 Lower limit [dB] for the frequency response . . . . .	3
2.3.2 'Auto Zoom' & 'Zoom Out' for the Impulse Response . . . . .	4
2.3.3 Zoom in & out with the cursor . . . . .	5
2.4 Impulse response with normalized dB-scale . . . . .	6
<b>3 Exporting a Response</b>	<b>7</b>
3.1 Text-file options . . . . .	8
3.2 Exporting the Frequency Response . . . . .	8
3.3 Exporting the Impulse Response . . . . .	9
<b>4 Importing a response</b>	<b>11</b>
4.1 Importing a frequency response . . . . .	11
4.2 Importing a impulse response . . . . .	12
4.3 Importing a Signal & Recording pair . . . . .	12
<b>5 Measurement Options</b>	<b>13</b>
5.1 Normalization . . . . .	14
5.2 Time zero . . . . .	15
5.2.1 Auto detect time-zero . . . . .	16
5.3 Time-window . . . . .	17
5.4 Frequency response - Amplitude Smoothing . . . . .	18
5.5 Frequency response - Frequency dependent time-window. . . . .	19
5.6 Highpass and Lowpass . . . . .	20
<b>6 Manipulation</b>	<b>22</b>
6.1 Frequency domain arithmetic . . . . .	22

---

<b>A Mathematics</b>	<b>24</b>
A.1 Definitions and Fourier transforms . . . . .	24
A.2 Filtering in the frequency domain . . . . .	26
A.2.1 Constant time-window . . . . .	26
A.2.1.1 Example: Hann window . . . . .	27
A.2.1.2 Example: Blackman window . . . . .	27
A.2.1.3 Example: Gaussian smoothing - time window . . . . .	28
A.3 $1/N$ Octave Smoothing . . . . .	29
A.3.1 The intervals . . . . .	29
A.3.2 Complex smoothing of discrete frequency values . . . . .	29
A.3.2.1 Filter shape . . . . .	30
A.3.2.2 Time-windowing . . . . .	30
A.3.2.3 Visualizing with graphs . . . . .	31
A.3.3 Absolute smoothing of discrete frequency values . . . . .	32
<b>Index</b>	<b>33</b>

# Chapter 1

## Introduction

The goals for the application is to

- Measure Impulse-response
- Measure Phase-response
- Measure Frequency-response
- Measure Harmonic distortion (THD)
- Make bandwidth limited measurements with no phase-errors
- Compare measurements
- Import Frequency- & Impulse-response
- Export Frequency- & Impulse-response
- Using open format for saved files (zip, wav/flac, txt)

HOLMImpulse is freeware (Non-Commercial) and works with any PC soundcard with no limitations.

### 1.0.1 HOLMImpulse and DSPreLab?

The main product for HOLM Acoustics is the digital preamplifier [DSPre 1](#), which is configured using the [DSPreLab](#) software. DSPreLab has the same measurement techniques as HOLMImpulse, but with automatic integrated corrections for the DSPre 1 based upon the measurements. The core software libraries for HOLMImpulse and DSPreLab are the same - Hence no extra effort making the core of HOLMImpulse.

### 1.0.2 HOLMImpulse @ diyAudio.com

The people at the diyAudio.com forum have been very help full making this application better - Thanks!

**Discussing technicalities and mathematics - And reporting bugs:**

<http://www.diyaudio.com/forums/showthread.php?threadid=144984>

**Measurements in practice - Feedback & help:**

<http://www.diyaudio.com/forums/showthread.php?threadid=145662>

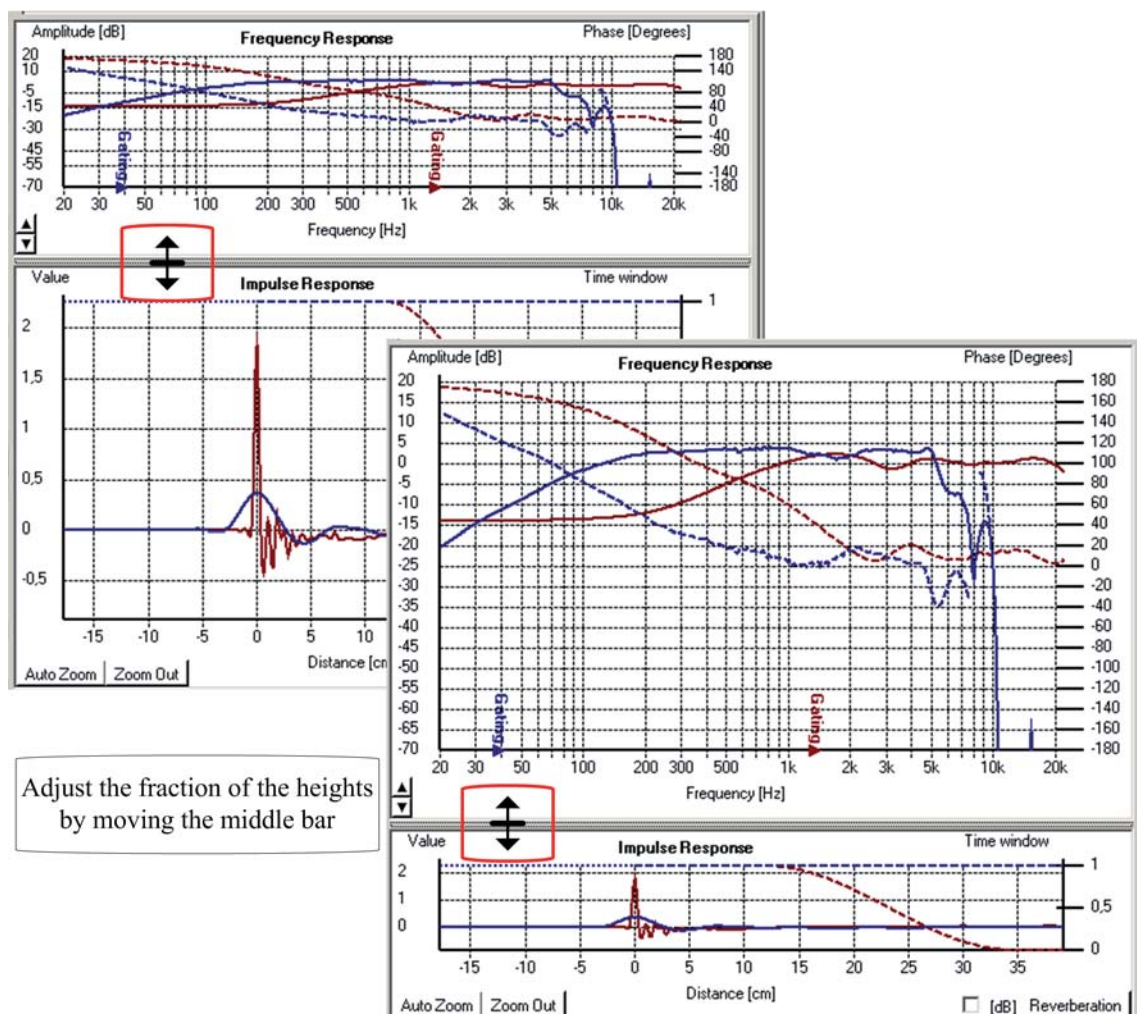
## Chapter 2

# Navigating the Response Graphs

The frequency and impulse response graphs are the cornerstone in HOLMImpulse. There are several tips and tricks for changing the visualization of these graphs. This chapter will also reveal some very nice features when dealing with these graphs.

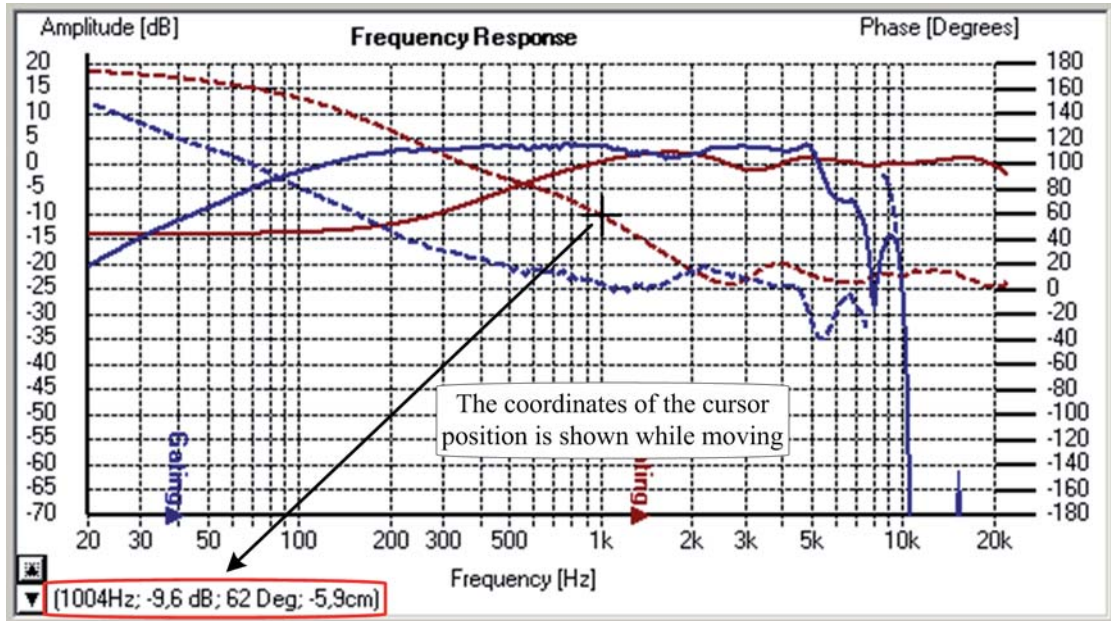
### 2.1 Adjusting the heights of the areas

A standard windows graphical user-interface option is to adjust the heights of windows by moving the middle bar:



## 2.2 Coordinates for the cursor position

When moving the cursor around on the graph, the coordinates in the selected units are shown



In this example the coordinates are: (1004Hz; -9,6dB; 62 Deg; -5,9cm)  
 The first three values (1004Hz; -9,6dB; 62 Deg) are obvious, the -5,9cm is the phase delay calculated as distance:

$$\text{Phase delay [sec]} = -\frac{\text{Phase [Deg]}}{360 \times \text{Frequency [Hz]}} = -\frac{62}{360 \times 1004\text{Hz}} = -0,1715\text{ms}$$

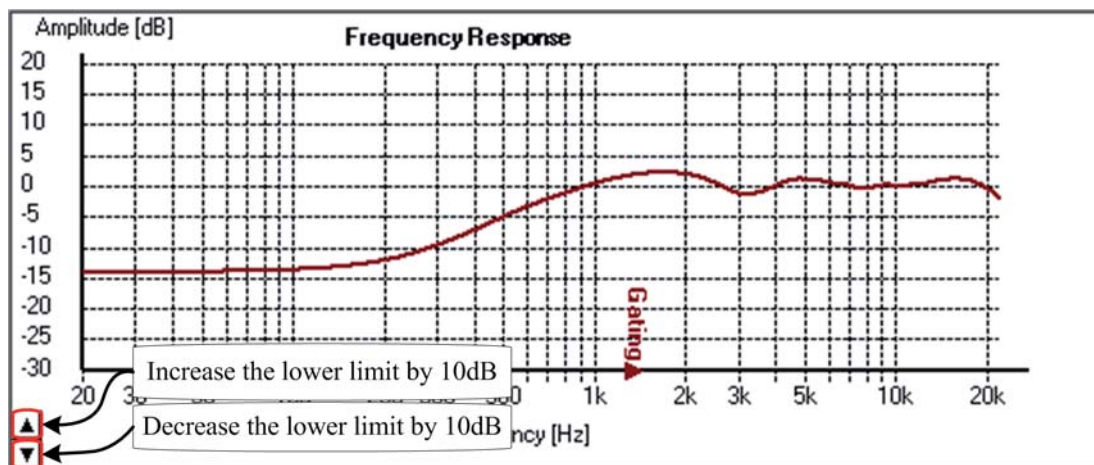
$$\text{Phase distance [m]} = 344\text{m/s} \times \text{Phase delay [sec]} = -0,059\text{m} = -5,9\text{cm}$$

## 2.3 Adjusting the Ranges

The possibility to adjust the ranges of a graph is essential when visualizing data.

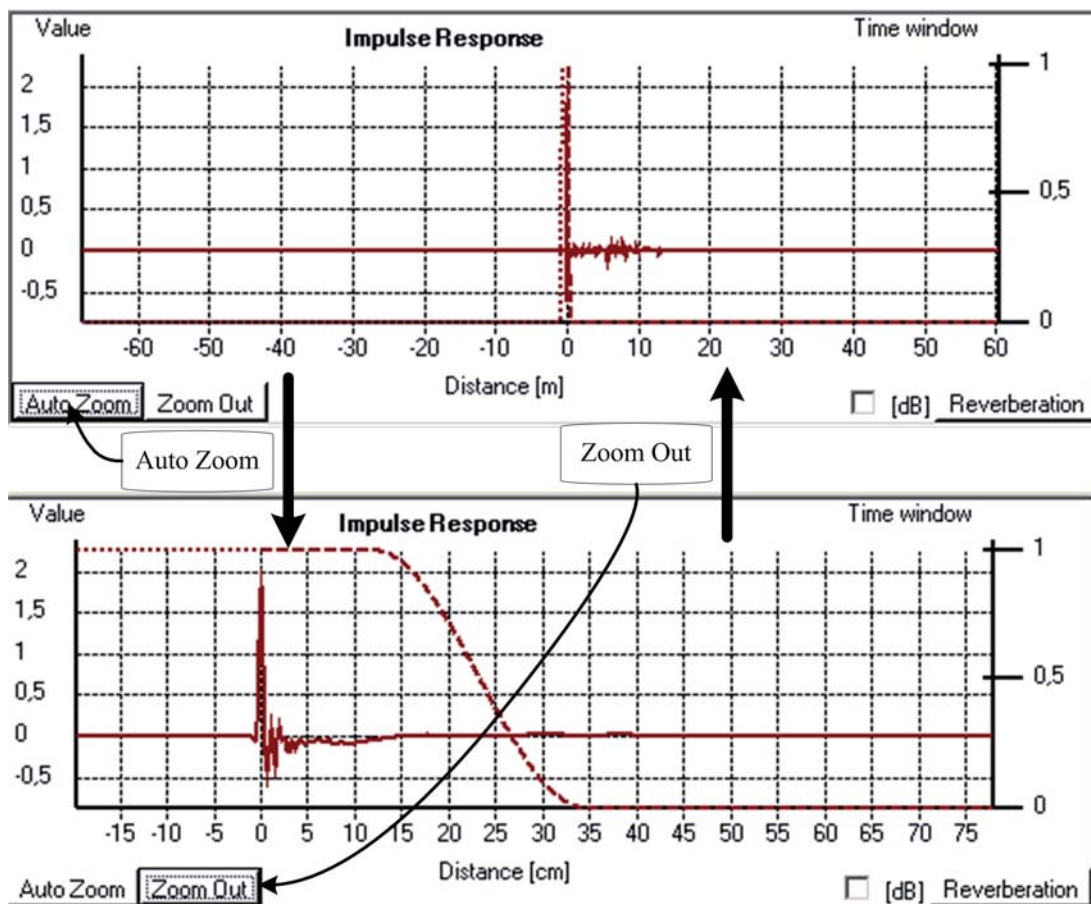
### 2.3.1 Lower limit [dB] for the frequency response

The lower dB-limit can be adjusted easily by the two arrows in the lower left corner:



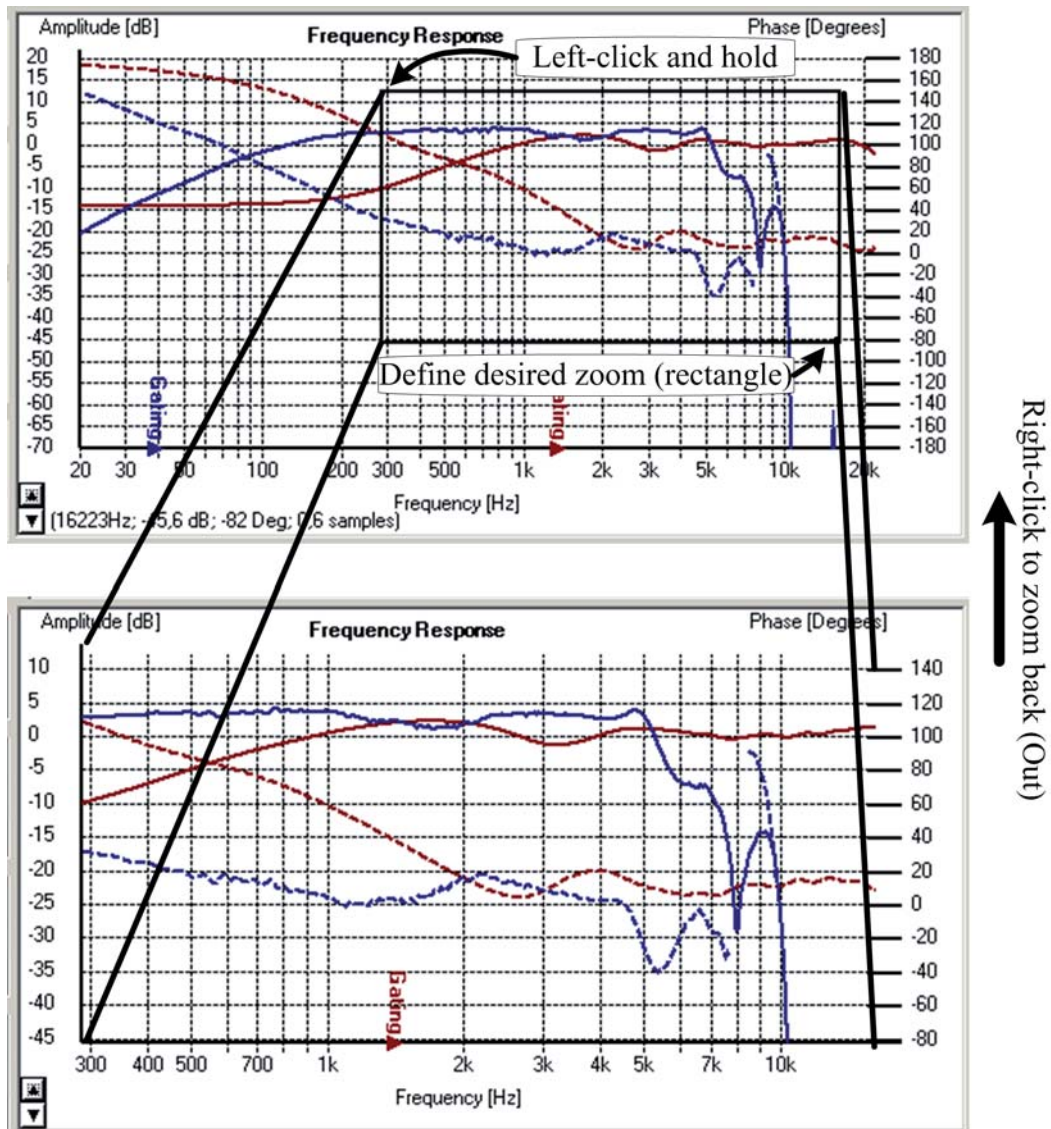
### 2.3.2 'Auto Zoom' & 'Zoom Out' for the Impulse Response

The 'Auto Zoom' and the 'Zoom Out' button in the lower left corner will adjust the ranges for the shown impulse response(s)



### 2.3.3 Zoom in & out with the cursor

Navigating in graphs with respect to zooming in and out is easy using the pointing device:

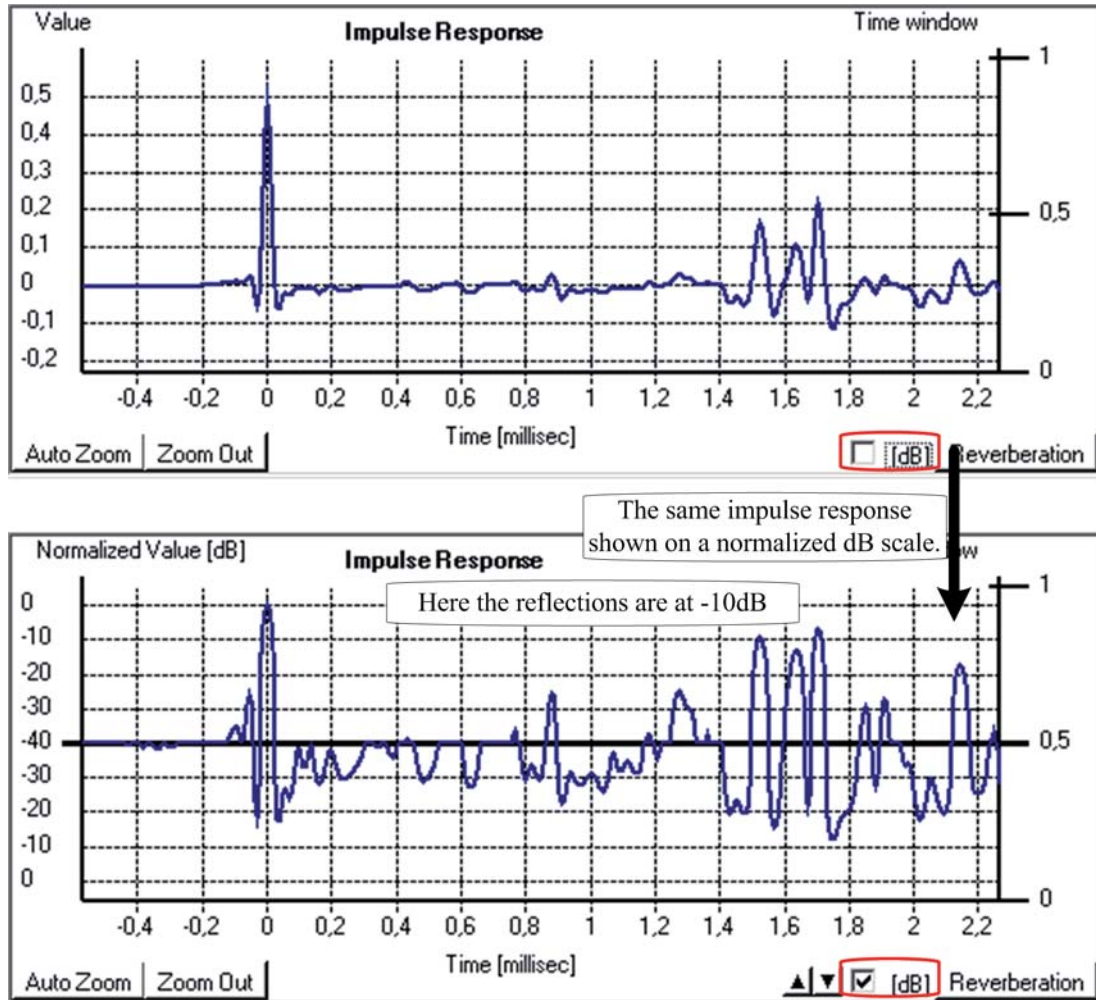


When you have zoomed in you can zoom in again, and right-clicking will follow your zoom history back.

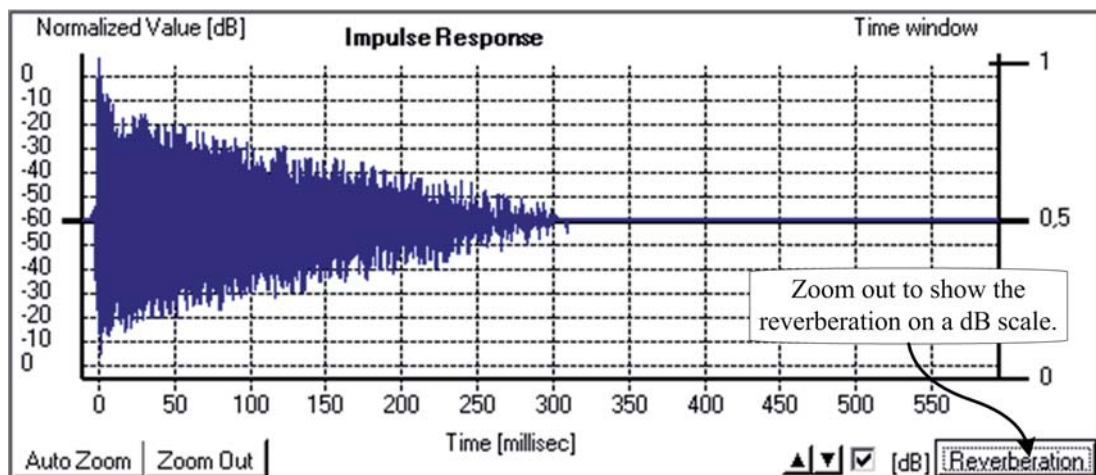


## 2.4 Impulse response with normalized dB-scale

The value for the impulse response can be shown on a dB-scale to eg. visualize reflections on a dB-scale:



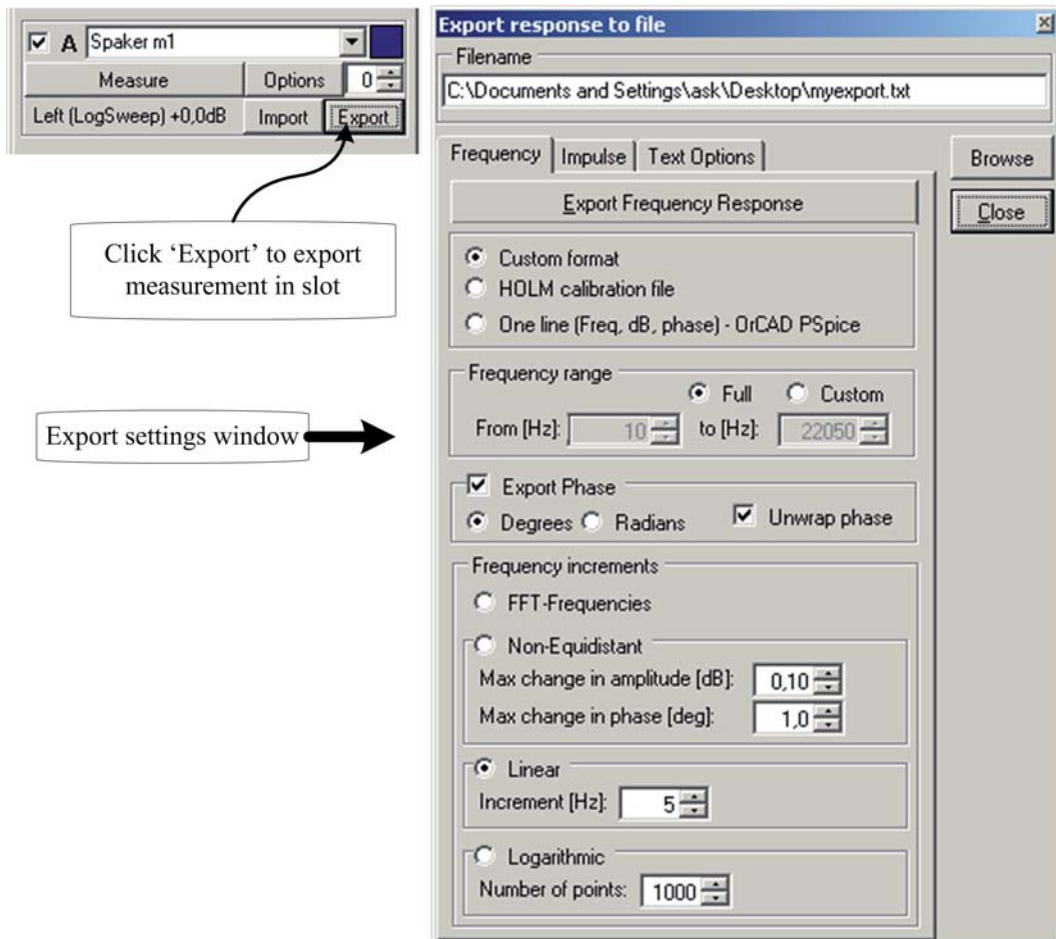
The 'Reverberation' button will enable the dB-scale and adjust the range so that the reverberation is shown:



## Chapter 3

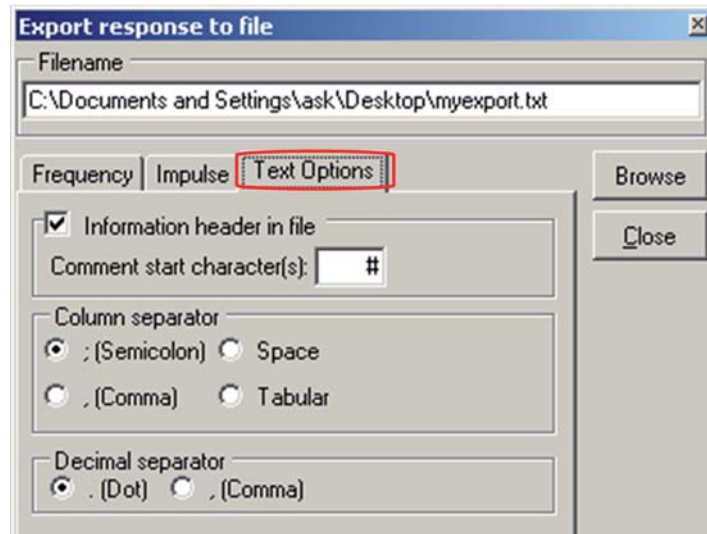
# Exporting a Response

A response can be exported as a frequency response or as an impulse response.



### 3.1 Text-file options

When exporting to a text-file you can adjust some self-explanatory preferences:



- Information header in file is a comment section in the top of the text-file, where the comment start character(s) can be configured
- The column separator is the character which is used to separate to columns
- The decimal separator is a . (Dot) or a , (Comma) such that eg.  
 $\pi = 3.1416\dots$  or  $\pi = 3,1416\dots$

Note: The text-file options are remembered - also after a restart of the application

### 3.2 Exporting the Frequency Response

The frequency response is exported as an ASCII text file with the above text-file options. The procedure for exporting is:

1. Find the measurement, which you want to export and click the 'Export' button
2. Select file-name (Browse) and verify settings in the Frequency tab
3. Click the 'Export Frequency Response' button

Exporting phase is optional and so is the phase unwrapping.

The discrete frequencies can be exported using following settings:

**FFT-Frequencies** will export all the discrete frequencies (current FFT-size)

**Non-Equidistant** will only export frequencies, when the response changes

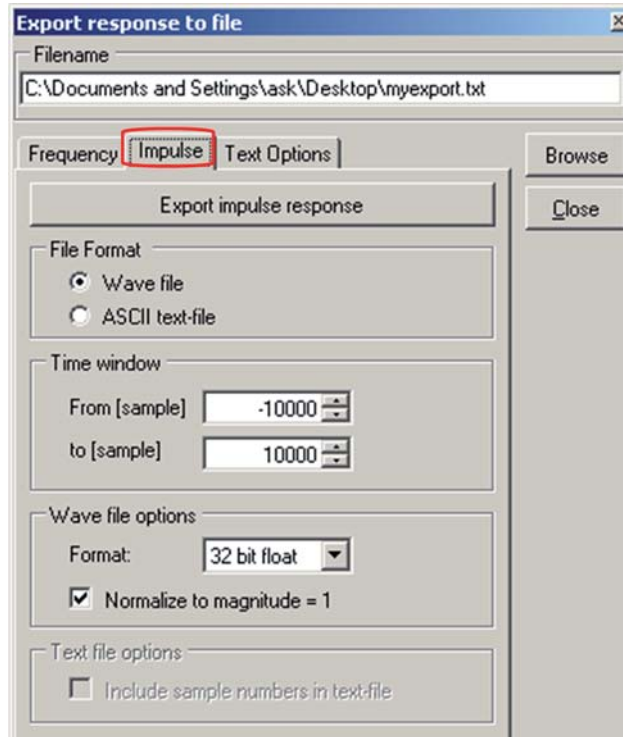
**Linear** will export frequencies at a selected step

**Logarithmic** will export frequencies with a step increasing logarithmic

Note: The frequency export options are remembered - also after a restart of the application

### 3.3 Exporting the Impulse Response

Exporting the impulse response can either be done as text-file or as a wave-file (<http://en.wikipedia.org/wiki/WAV>)



The procedure for exporting is:

1. Find the measurement, which you want to export and click the 'Export' button
2. Select file-name (Browse) and verify settings in the Impulse tab
3. Click the 'Export impulse response' button

Note: The frequency export options are remembered - also after a restart of the application

#### Time window

The time window lets you define the range of samples that should be exported

#### Wave file options

The format of the wave-file can be:

- 32 bit float
- 16 bit PCM (integer)
- 24 bit PCM (integer)
- 32 bit PCM (integer)

When exporting to wave-files it is convenient to normalize the impulse response, so that the highest magnitude equals 1.

### **Text file options**

Exporting an impulse as a text-file can result in rather large files. The impulse sample number is trivial when using this file in other programs - therefore is the sample numbers in the text-file optional.

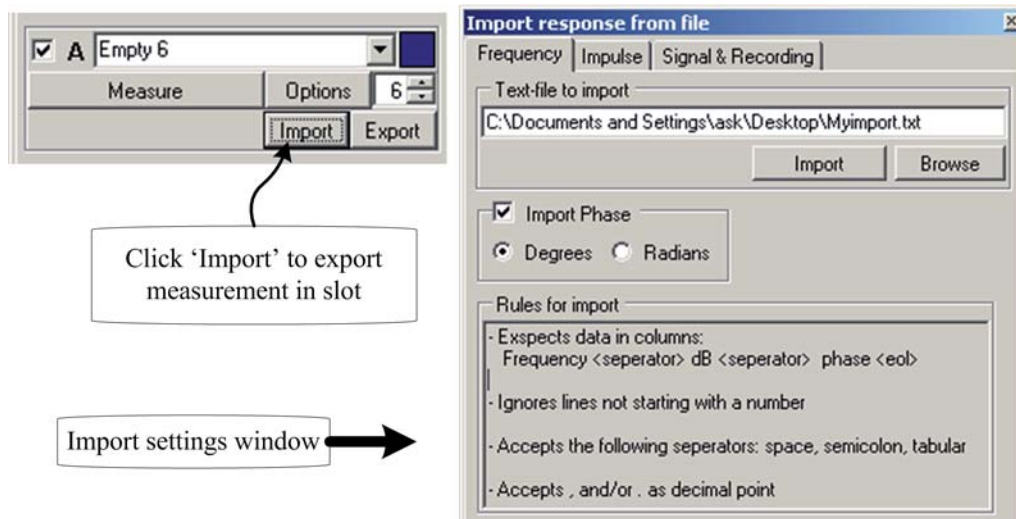
Note: The general settings in the 'Text Options' tab as described in Section [3.1](#) are used when exporting the impulse response as ASCII text-file

## Chapter 4

# Importing a response

A response can be imported as a frequency response, impulse response or a recorded/manipulated response to a known signal.

Click 'Import' in a measurement slot which can be overwritten:



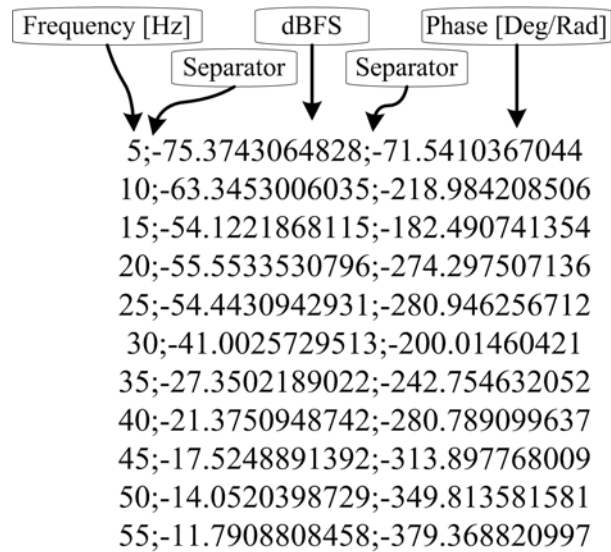
### 4.1 Importing a frequency response

A frequency response can be imported from a text-file with either two or three columns if the phase is to be imported. The rules of import are:

- Expects data in columns: Frequency <separator> dB <separator> phase <eol>
- Ignores lines not starting with a number
- Accepts the following column-separators: space, semicolon, tabular
- Accepts , (Comma) and/or . (Dot) as decimal point

Note: The samplerate will be the current set under the 'Device & Signal' tab

An example of a valid frequency response text-file:



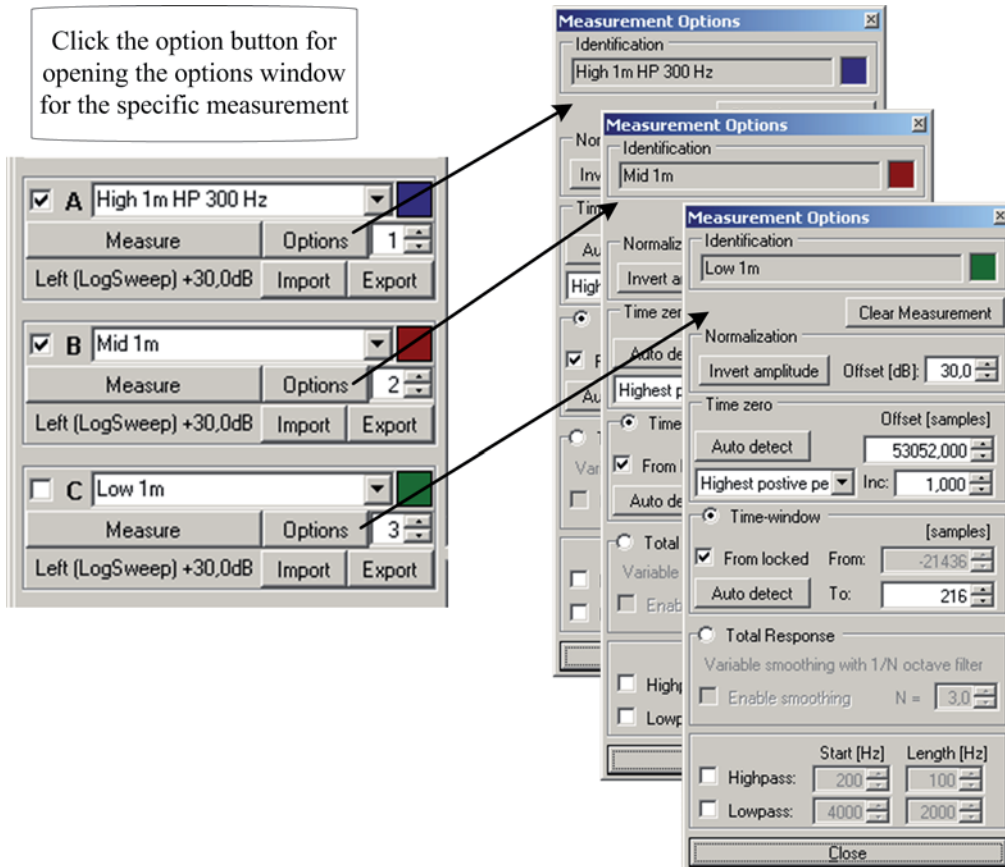
## 4.2 Importing a impulse response

## 4.3 Importing a Signal & Recording pair

## Chapter 5

# Measurement Options

The measurement options allows you to change the representation of the measurement. Changing the options for the measurements does not affect the measurement data - only the representation.



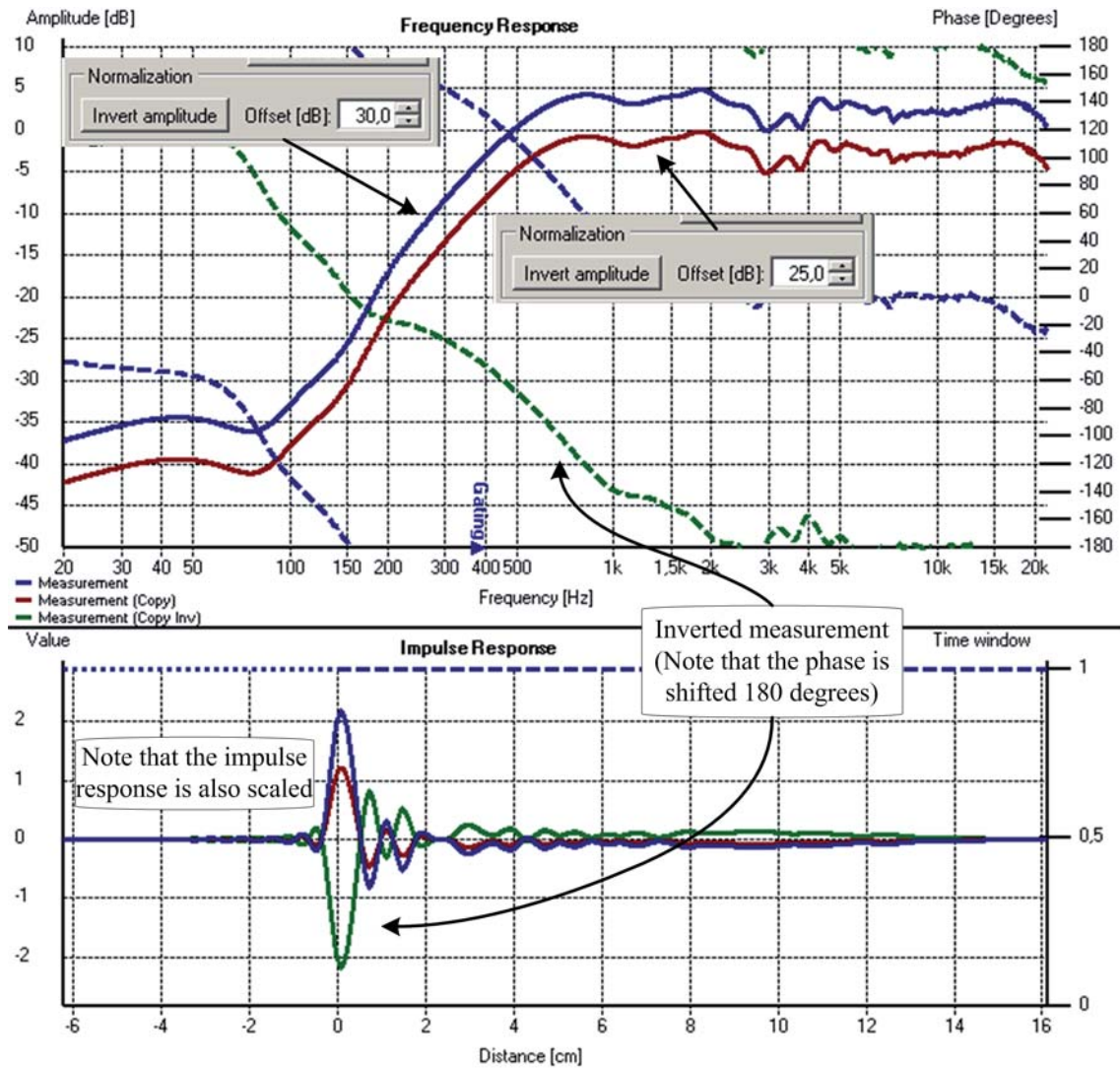


### 5.1 Normalization

When comparing measurements the input gain often differs, therefore the normalization allows you to offset measurements.

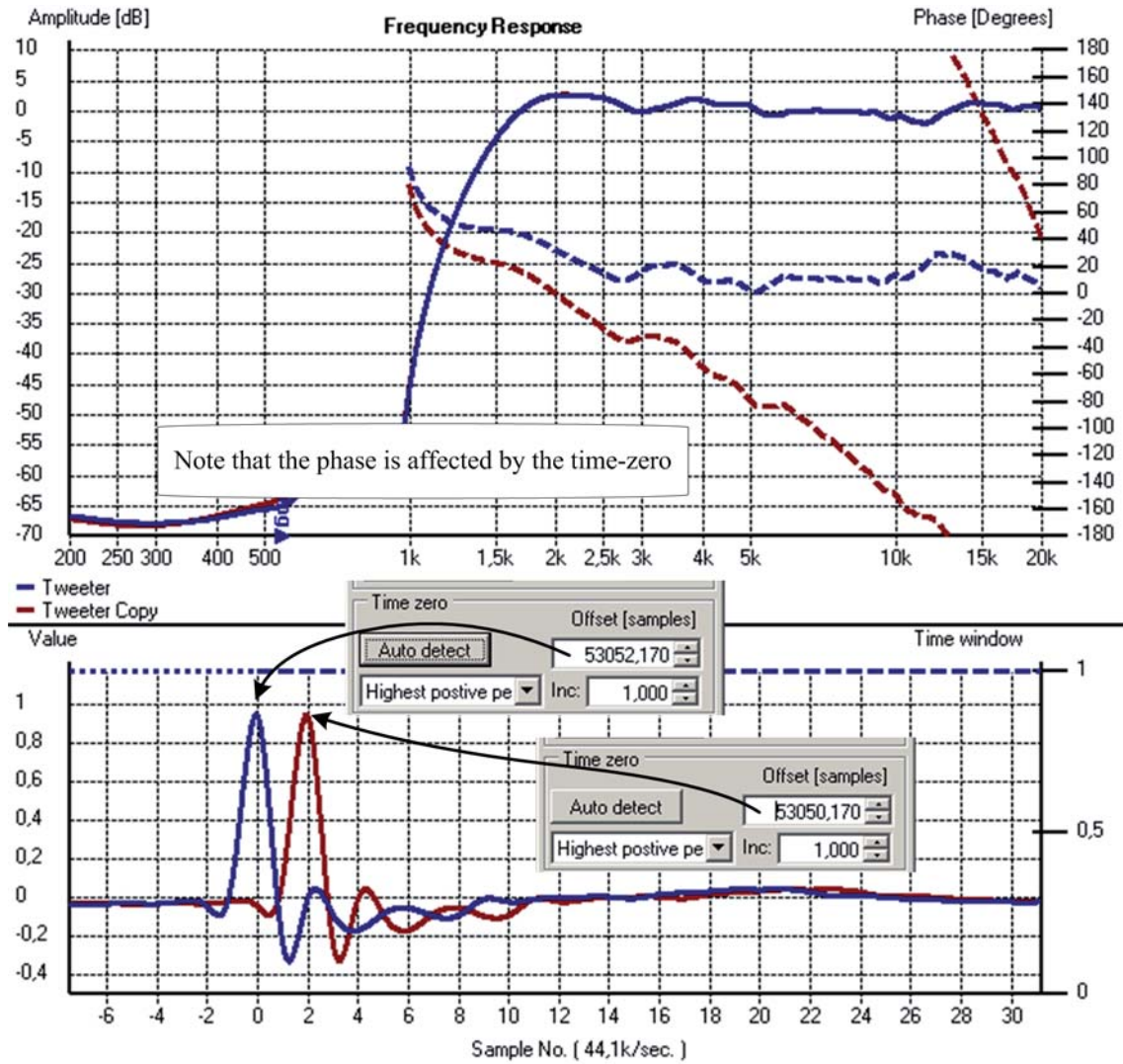
Below shows the same measurement with two different normalization offset.

The third curve shows an inverted measurement



## 5.2 Time zero

The application uses a strategy to locate the measured impulse response. The located time offset (Time zero) can be adjusted manually



The phase is depending on the time-zero - In the above example the two graphs represents the exact same measurement, but with a different time-zero. It is good to remember that

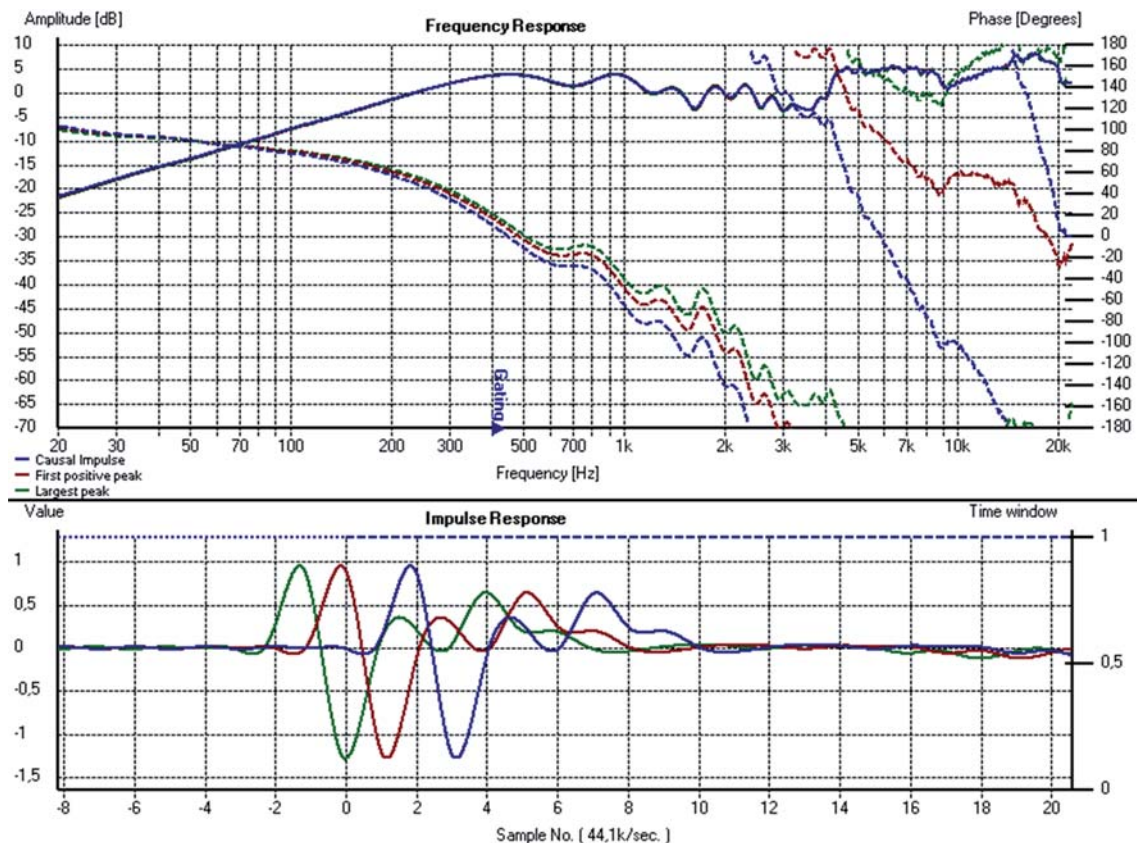
$$1 \text{ sample} = 180 \text{ Degrees @ Nyquist frequency, } \quad (\text{Nyquist frequency} = \text{samplerate}/2)$$

In above example the one impulse is delayed 2 samples, which gives a phase shift of 360 Degrees at 22.5 kHz.

### 5.2.1 Auto detect time-zero

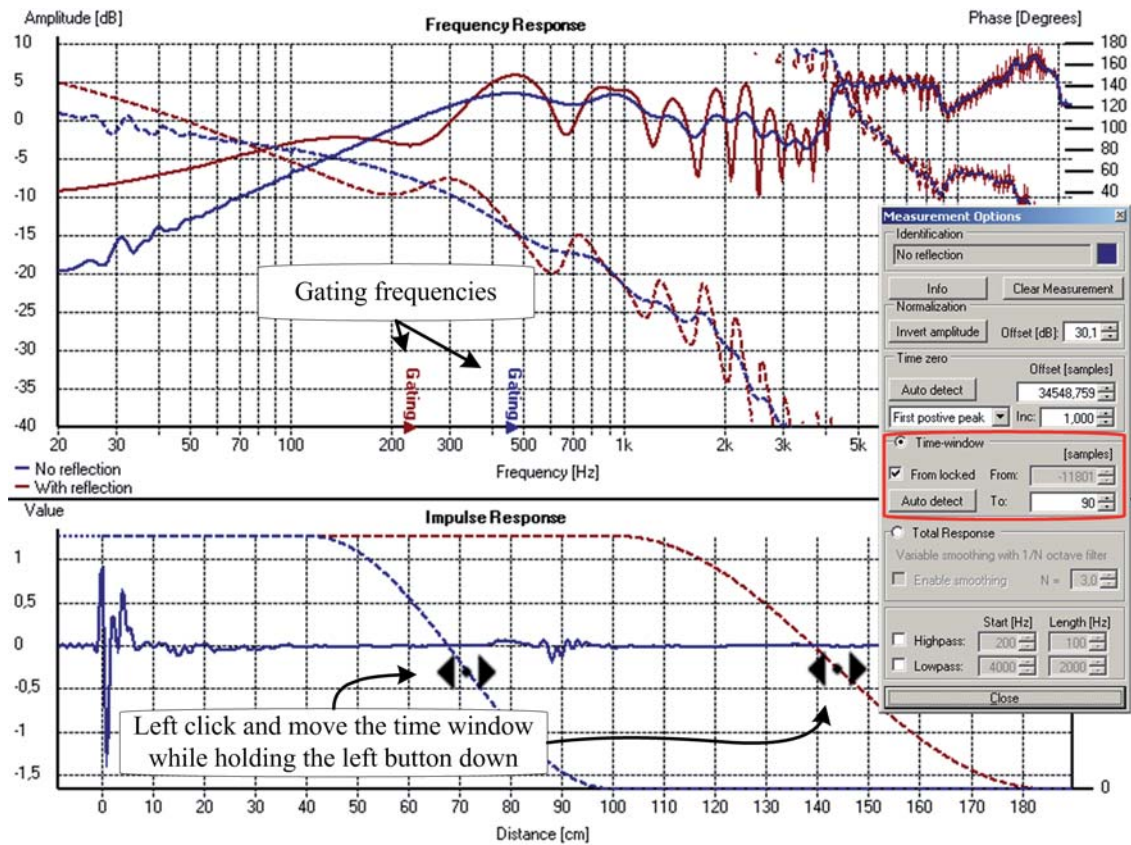
As seen the time-zero is crucial when inspecting the phase. Different strategies can be chosen. In below graph three different methods are shown - all five are:

- First positive peak
  - Disregard the pre-rings from DAC/ADC
  - Assume correct polarity
- Highest positive peak
  - Disregard every information except the highest positive peak
  - Assume correct polarity
- First peak
  - Disregard the pre-rings from DAC/ADC
- Largest peak
  - Disregard every information except the largest peak (Safest method)
- Causal impulse
  - Don't allow any pre-rings



### 5.3 Time-window

The time is used for gated measurements. Reflections/echoes can be neglected. The time-window can be dynamically adjusted using the pointing device (The frequency response is instantly updated)



#### Gating frequency

When the wavelength becomes longer than the time window the frequency response is no longer reliable (rule of thumb).

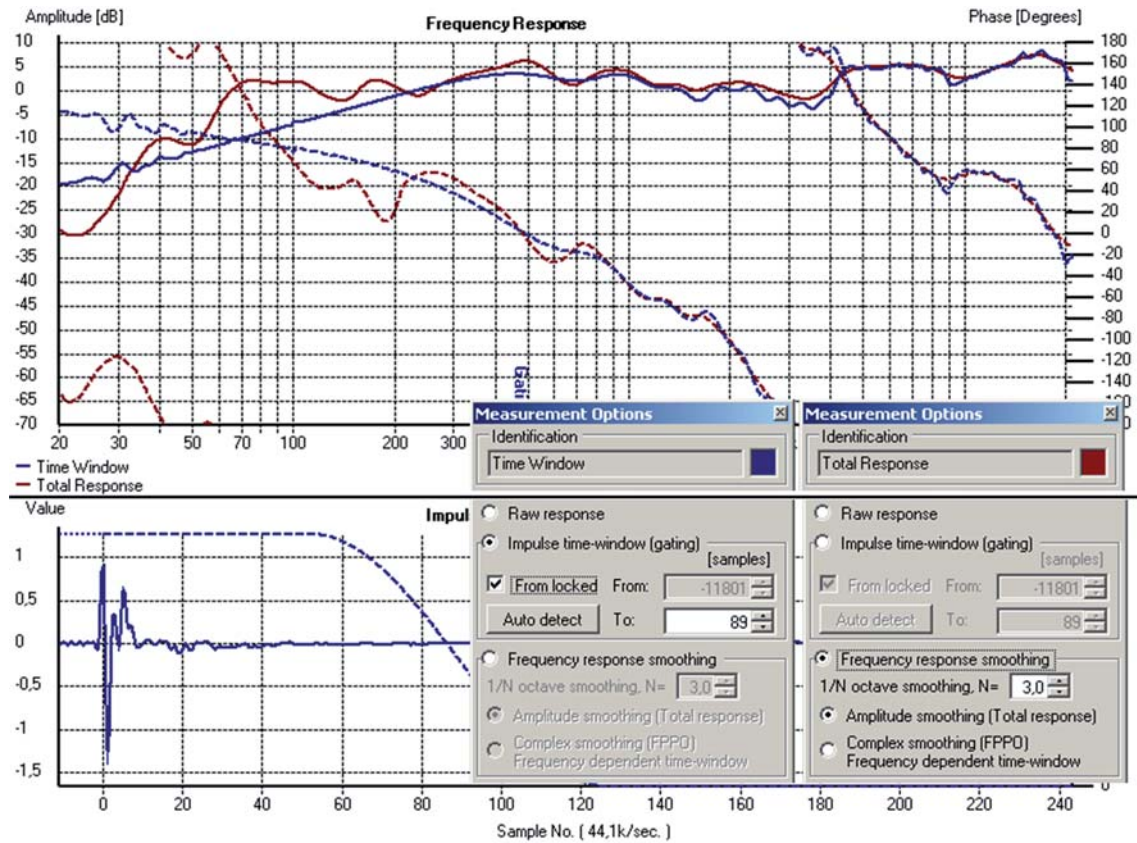
$$\text{Gating frequency [Hz]} = \frac{1}{\text{Time window [s]}} = \frac{344\text{m/s}}{\text{Time window [m]}}$$

#### Auto detect

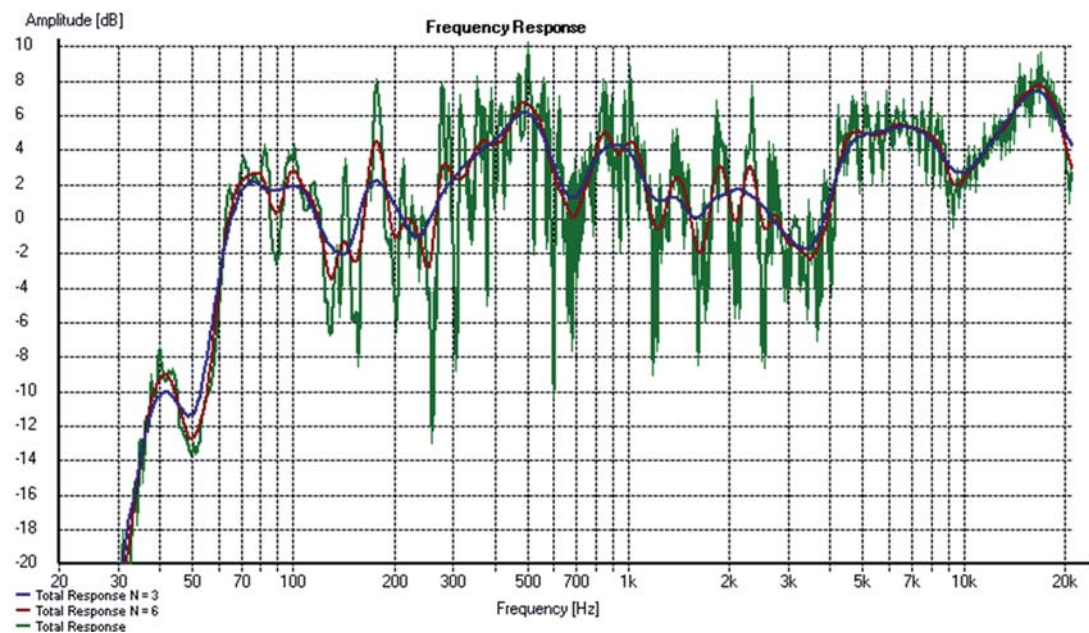
The 'Auto Detect' button will locate reflections from the main-impulse and set the time-window just before these reflections.

### 5.4 Frequency response - Amplitude Smoothing

This total response is the response including all reflections and echoes from the room (Speaker & Room-response)

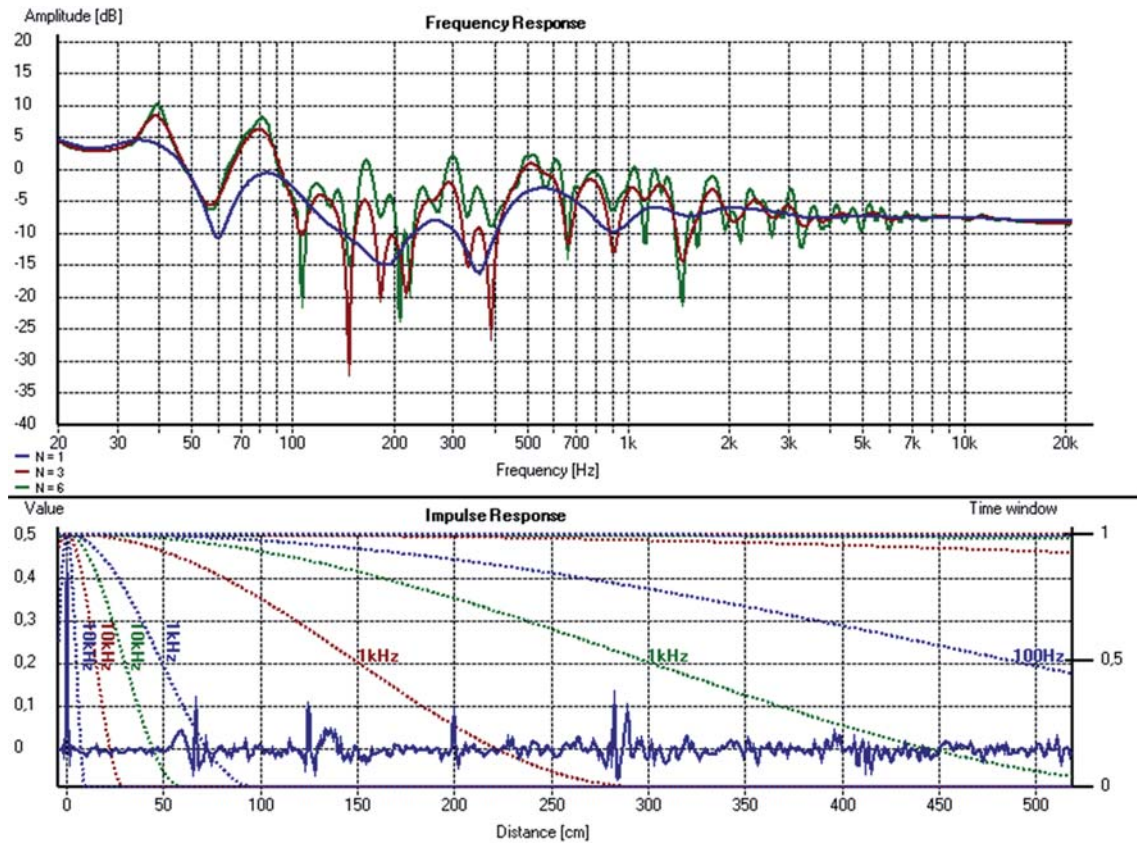


The smoothing is optional and can be adjusted. In below graph the green curve is without smoothing and the red and blue has different amount of smoothing

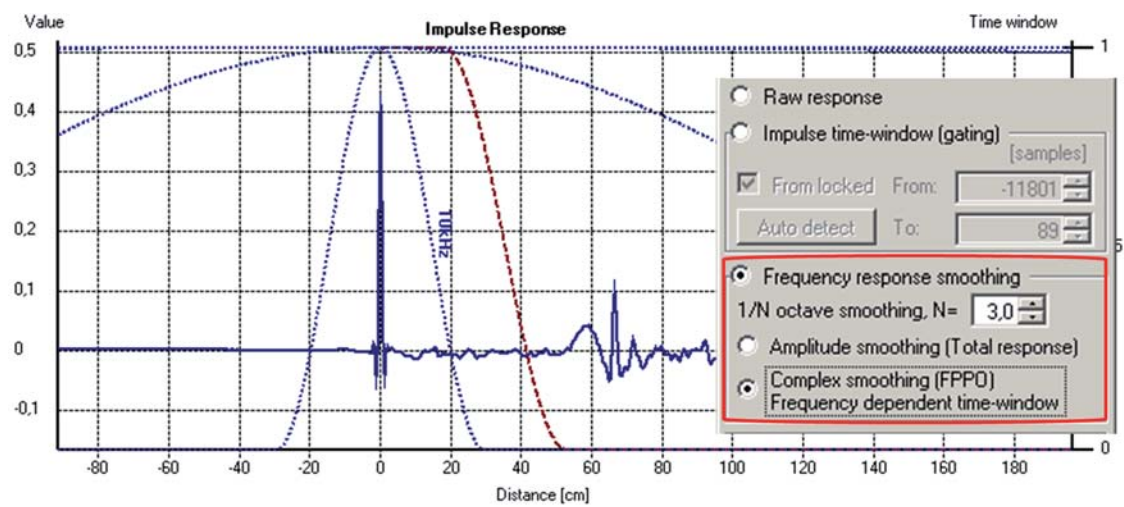


### 5.5 Frequency response - Frequency dependent time-window.

Making a complex logarithmic average in the frequency domain corresponds to a frequency dependent time-window. The frequency dependent window is shown for each decade (10Hz, 100Hz, 1kHz, 10kHz)



There are advantages and disadvantages compared to a constant (normal) time window

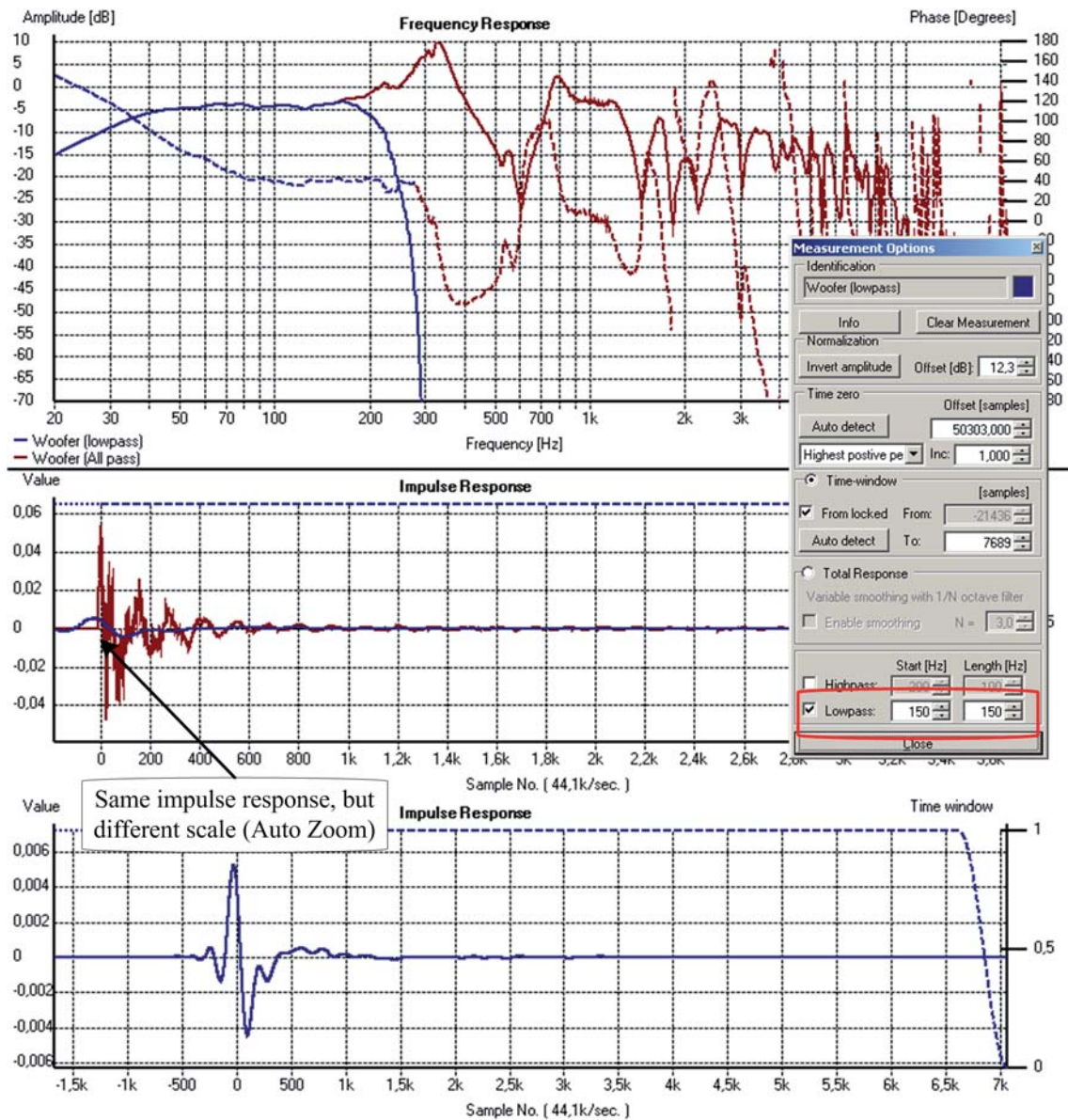


## 5.6 Highpass and Lowpass

Deploying highpass and/or lowpass to a measurement will reveal the impulse response in the desired frequency range.

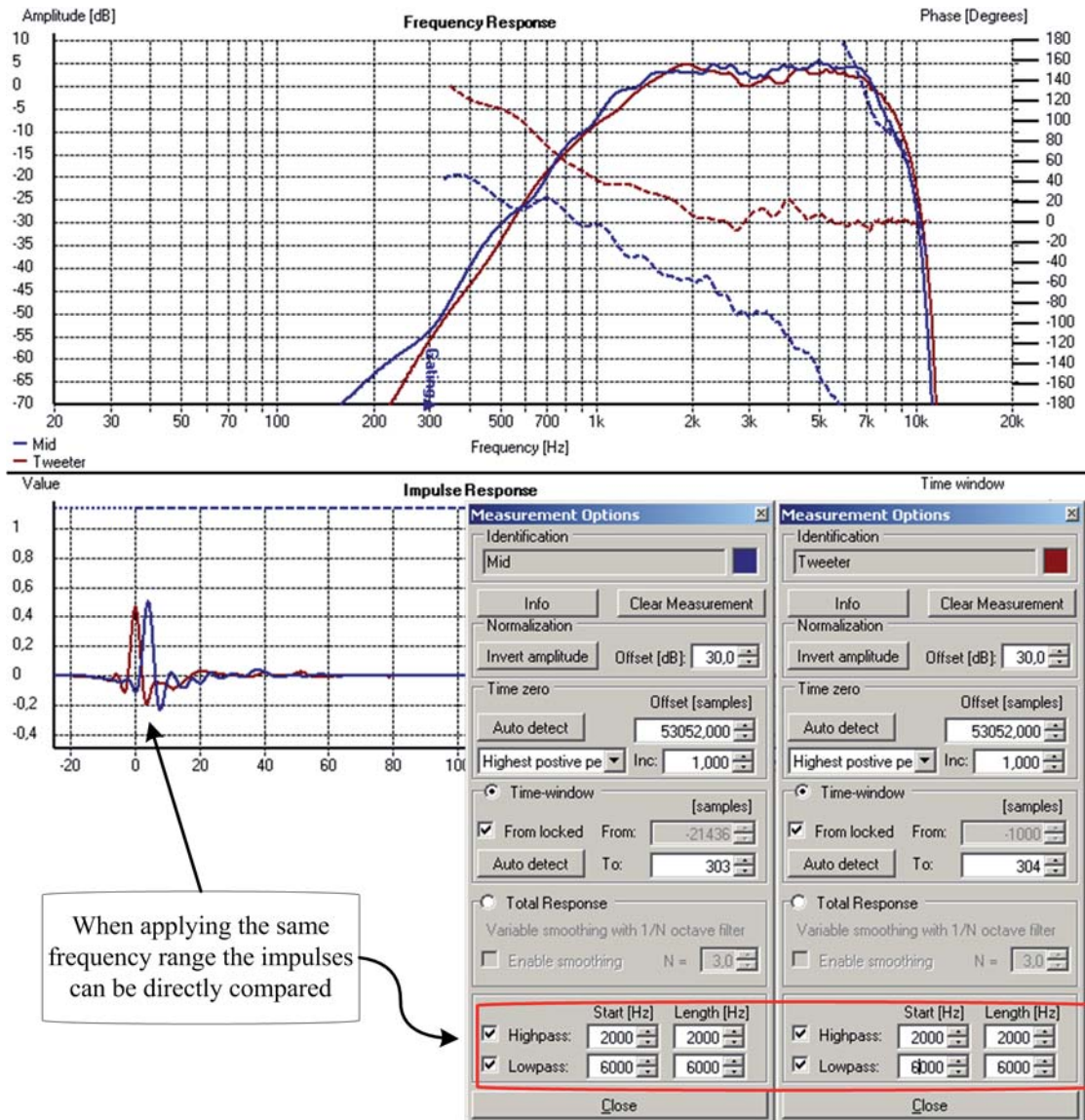
### Lowpass a woofer measurement

In below measurement a woofer has been measured, but the impulse response is only interpretable after the lowpass



### Lowpass and Highpass (Bandpass) two measurements

If a eg. tweeter and a mid is measured, and we want to compare the time alignment of these two measurements we can bandpass the measurements for directly comparison in the time domain

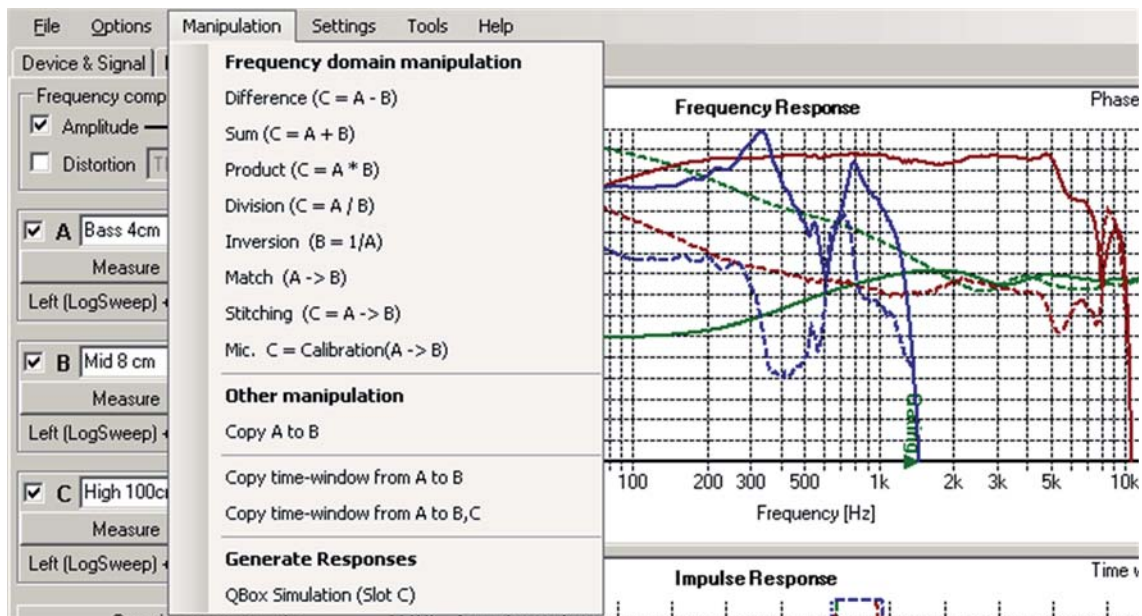




## Chapter 6

# Manipulation

The manipulation menu allows you to make certain operations on existing measurements and make new simulated data.



### 6.1 Frequency domain arithmetic

There are three measurement slots: A, B and C, that makes it possible to make the most common algebraic operations.

NB: Be aware that the measurement on the left-hand-side of the equation will be overwritten.

For simplicity - Assume that the samperates for A, B are same.

First define some symbols:

$$\begin{aligned}
 k & : \text{Frequency number} \\
 A[k], B[k], C[k] & : \text{Complex sequences representing the three slots A,B,C}
 \end{aligned}$$

Thereafter the frequency domain manipulations are straight forward

**Difference (C = A - B)**

Example: A and B are two drivers played where B has inverted polarity - then C is the resulting response.

$$C[k] = A[k] - B[k], \quad \text{for all } k$$

**Sum (C = A + B)**

Example: A and B are two drivers played - then C is the resulting response.

$$C[k] = A[k] + B[k], \quad \text{for all } k$$

**Product (C = A \* B)**

Example: A represents a driver and B represents a filter - then C is the result you will get deploying the filter to the driver.

$$C[k] = A[k]B[k], \quad \text{for all } k$$

**Division (C = A / B)**

Example: A represents a measurement and B represents a microphone response - then C is the measurement after microphone calibration.

$$C[k] = \frac{A[k]}{B[k]} = A[k] \frac{B[k]^*}{|B[k]|^2}, \quad \text{for all } k$$

**Inversion (B = 1 / A)**

Example: A represents a measurement - then B is the filter you will need to deploy to the speaker to get a flat response.

$$B[k] = \frac{1}{A[k]} = \frac{A[k]^*}{|A[k]|^2}, \quad \text{for all } k$$

## Appendix A

# Mathematics

Below definitions and derivations can be found in many books. I have chosen to write down my own derivations, because we have special cases when dealing with computational Fourier transforms:

- The frequency- and time-domain are always discrete
- The frequency- and time-domain are always finite
- The Fourier-size (FFT-size) is always a power of two (even)
- The values in time-domain are always real

### A.1 Definitions and Fourier transforms

Symbol	Definition
$L = L_{FFT}$	FFT length, $L$ is a power of 2: $L = 2^M$ , $M = 1, 2 \dots$
$N = L/2 = N_{FFT}$	Half FFT length
$R$	Samplerate [Hz]
$i$	Complex constant: $e^{i\theta} = i \cos \theta + \sin \theta$ and $i^2 = -1$
$\delta[n]$	Kronecker delta function, $\delta[n] = \begin{cases} 1 & , n = 0 \\ 0 & , n \neq 0 \end{cases}$
$\delta[n, m] = \delta[n - m]$	Kronecker delta function, $\delta[n, m] = \begin{cases} 1 & , n = m \\ 0 & , n \neq m \end{cases}$
$*$	Complex conjugation $z = a + ib$ , $z^* = a - ib$
$\arg(z) \in [-\pi, \pi[$	Phase of the complex number $z$

Impulse responses are represented by real sequences (lowercase):

$$x[n] \in \mathbb{R}, \quad n = -N, \dots, N - 1$$

Frequency responses are represented by complex sequences (uppercase):

$$X[k] \in \mathbb{C}, \quad k = 0, \dots, L - 1$$

The Fourier basis functions and properties:

$$\begin{aligned} e[nk] &= \exp\left(i\frac{2\pi}{L}nk\right) \\ e[nk]^* &= e[-nk] \\ e[(n \pm L)k] &= e[nk] \\ e[n(k \pm N)] &= (-1)^n \\ e[nk] + e[nk]^* &= 2 \cos\left(\frac{2\pi}{L}nk\right) \end{aligned}$$

Kronecker delta function represented by the basis functions:

$$\begin{aligned} \delta[k] &= \frac{1}{L} \sum_{n=-N}^N e[nk] \\ \delta[n] &= \frac{1}{L} \sum_{k=0}^L e[nk] \end{aligned}$$

The Fourier transforms:

$$X[k] = \sum_{n=-N}^{N-1} x[n]e[nk] \quad (\text{A.1})$$

$$x[n] = \frac{1}{L} \sum_{k=0}^{L-1} X[k]e[nk]^* \quad (\text{A.2})$$

Properties of these transforms:

$$\begin{aligned} X[L-k] &= \sum_{n=-N}^{N-1} x[n]e[n(L-k)] = X[k]^* \\ X[0] &= \sum_{n=-N}^{N-1} x[n], \quad X[0] \in \mathbb{R} \\ X[N] &= \sum_{n=-N}^{N-1} (-1)^n x[n], \quad X[N] \in \mathbb{R} \end{aligned}$$

These properties can be used to see, that there is in fact only  $N$  frequencies with  $2N = L$  degrees of freedom

$$\begin{aligned} x[n] &= \frac{1}{L} \sum_{k=0}^{L-1} X[k]e[nk]^* \\ &= \frac{1}{L}X[0] + \frac{1}{L} \sum_{k=1}^{N-1} (X[k]e[nk]^* + X[k]^*e[nk]) + \frac{1}{L}X[N](-1)^n \\ &= \frac{1}{L}X[0] + \frac{2}{L} \sum_{k=1}^{N-1} |X[k]| \cos\left(\frac{\pi}{N}nk + \theta[k]\right) + \frac{1}{L}X[N](-1)^n \quad (\text{A.3}) \end{aligned}$$

$$\text{where } X[k] = |X[k]|e^{i\theta[k]}, \quad \theta[k] = \arg(X[k])$$

## A.2 Filtering in the frequency domain

Let  $W[k, j]$  be a general filtering function defined in the frequency domain such that the filtered frequency sequence  $X_W[k]$  is a result of the unfiltered  $X[k]$

$$\begin{aligned} X_W[k] &= \frac{1}{N+1} \sum_{j=0}^N W[k, j] X[j] \\ &= \frac{1}{L} \sum_{j=0}^{L-1} W[k, j] X[j] \end{aligned} \quad (\text{A.4})$$

This expression is the most general filtering. The relation to the impulse response  $x[n]$  is

$$\begin{aligned} x_w[n] &= \frac{1}{L} \sum_{k=0}^{L-1} X_W[k] e[nk]^* \\ &= \frac{1}{L} \sum_{k=0}^{L-1} \frac{1}{L} \sum_{j=0}^{L-1} W[k, j] X[j] e[nk]^* \\ &= \frac{1}{L} \sum_{k=0}^{L-1} \frac{1}{L} \sum_{j=0}^L W[k, j] \sum_{m=-N}^N x[m] e[mj] e[nk]^* \\ &= \sum_{m=-N}^N x[m] \left( \frac{1}{L} \sum_{k=0}^{L-1} \frac{1}{L} \sum_{j=0}^{L-1} W[k, j] e[mj] e[nk]^* \right) \\ &= \sum_{m=-N}^N x[m] w[n, m] \end{aligned}$$

So that filtering in the frequency domain corresponds to another operation in the impulse domain where:

$$\begin{aligned} w[n, m] &= \frac{1}{L} \sum_{k=0}^{L-1} \frac{1}{L} \sum_{j=0}^{L-1} W[k, j] e[mj] e[nk]^* \\ W[k, j] &= \sum_{n=-N}^N \sum_{m=-N}^N w[n, m] e[mj]^* e[nk] \end{aligned}$$

### A.2.1 Constant time-window

A constant time-window in the impulse domain corresponds to a translation invariant frequency domain filter.

Assume that  $w[n, m]$  is a constant time-window:

$$\begin{aligned} w[n, m] &= w[n] \delta[n, m] \\ x_w[n] &= x[n] w[n], \quad w[n] \in \mathbb{R} \end{aligned}$$

The filter in frequency domain becomes:

$$\begin{aligned} W[k, j] &= \sum_{n=-N}^N \sum_{m=-N}^N w[n, m] e[mj]^* e[nk] \\ &= \sum_{n=-N}^N w[n] e[n(k-j)] \\ &= L \times W[k-j] \end{aligned}$$

Utilizing the translation invariance gives:

$$\begin{aligned} X_W[k] &= \frac{1}{L} \sum_{j=0}^{L-1} W[k, j] X[j] \\ &= \sum_{j=0}^{L-1} W[k - j] X[j] \end{aligned}$$

The Fourier transforms becomes:

$$\begin{aligned} W[k] &= \frac{1}{L} \sum_{n=-N}^N w[n] e[nk] \\ w[n] &= \sum_{k=0}^{L-1} W[k] e[nk]^* \\ &= W[0] + 2 \sum_{k=1}^{N-1} |W[k]| \cos\left(\frac{\pi}{N} nk + \theta[k]\right) + W[N] (-1)^n \quad (\text{A.5}) \end{aligned}$$

$$\text{where } W[k] = |W[k]| e^{i\theta[k]}, \quad \theta[k] = \arg(W[k])$$

#### A.2.1.1 Example: Hann window

A running average (smoothing) in frequency domain:

$$W[k] = \frac{1}{2} \delta[k] + \frac{1}{4} \delta[k + 1] + \frac{1}{4} \delta[k - 1]$$

or in table form:

$$\begin{aligned} W[0] &= \frac{1}{2} \\ W[-1] = W[1] &= \frac{1}{4} \end{aligned}$$

In time domain using (A.5):

$$w[n] = \frac{1}{2} + \frac{1}{2} \cos\left(\frac{\pi}{N} n\right)$$

This is the well-known Hann time-window.

#### A.2.1.2 Example: Blackman window

A running average (smoothing) in frequency domain:

$$W[k] = 0.42 \delta[k] + 0.25 \delta[k + 1] + 0.25 \delta[k - 1] + 0.04 \delta[k + 2] + 0.04 \delta[k - 2]$$

or in table form:

$$\begin{aligned} W[0] &= 0.42 \\ W[-1] = W[1] &= 0.25 \\ W[-2] = W[2] &= 0.04 \end{aligned}$$

In time domain using (A.5):

$$w[n] = 0.42 + \frac{1}{2} \cos\left(\frac{\pi}{N}n\right) + 0.08 \cos\left(2\frac{\pi}{N}n\right)$$

This is the well-known Blackman time-window.

### A.2.1.3 Example: Gaussian smoothing - time window

Now assume a Gaussian frequency smoothing:

$$W[k] = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-k^2}{2\sigma^2}\right)$$

$$\sigma = \text{FWHM (width of the smoothing)}$$

The Fourier transform of a Gaussian distribution is again a Gaussian distribution:

$$w[n] = \frac{\sigma}{\sqrt{2\pi}} \exp\left(\frac{-n^2\sigma^2}{2}\right)$$

$$\frac{1}{\sigma} = \text{FWHM (time-window-width)}$$

So as a rule of thumb:

$$\begin{aligned} \sigma &= \text{Smoothing width in frequency domain [Hz]} && \text{(A.6)} \\ \Downarrow & && \\ \frac{1}{\sigma} &= \text{Time-window-width in time domain [s]} \end{aligned}$$

Note: When using a Gaussian time-window one actually needs to apply an additional window to avoid sidelobes in the frequency domain due to the non-zero values at  $w[-N]$  and  $w[N-1]$ . Further more one need to use the so called [discrete gaussian kernel](#) for correct normalization.

### A.3 1/N Octave Smoothing

The 1/N octave smoothing originates from the analog world, with limited number of bandwidth bins. But the concept can be applied to the digital world, where we have a huge number of equidistant frequencies - typical  $2^{15} = 32768$  or more.

#### A.3.1 The intervals

At a frequency  $f$  we want to apply smoothing in an interval from  $f^-$  to  $f^+$ . The Interval must be  $N_{Oct} = 1/N$  octaves wide. This starting point gives the following constraints:

$$\begin{aligned} f^- < f < f^+ & \text{ or } f_{\log}^- < f_{\log} < f_{\log}^+ & (\text{The interval}) \\ f^- = \frac{f}{\Delta} & \text{ or } f_{\log}^- = f_{\log} - \Delta_{\log} & (\text{The interval start}) \\ f^+ = f \times \Delta & \text{ or } f_{\log}^+ = f_{\log} + \Delta_{\log} & (\text{The interval end}) \\ \frac{f^+}{f^-} = 2^{N_{Oct}} & \text{ or } f_{\log}^+ - f_{\log}^- = \log_{10}(2) \times N_{Oct} & (N_{Oct} \text{ octaves}) \end{aligned}$$

In above the log-subscript denotes the logarithmic value of the number such that:

$$\begin{aligned} f_{\log} &= \log_{10}(f) \\ f_{\log}^- &= \log_{10}(f^-) \\ f_{\log}^+ &= \log_{10}(f^+) \\ \Delta_{\log} &= \log_{10}(\Delta) \end{aligned}$$

Solving these equations for  $\Delta$  gives:

$$\Delta = \sqrt{2^{N_{Oct}}} = 2^{0.5 \times N_{Oct}} \quad (\text{A.7})$$

#### A.3.2 Complex smoothing of discrete frequency values

Assume the discrete complex frequencies in the frequency domain are:

$$F[k], \quad k = 0, \dots, N_{FFT}$$

The complex average is calculated without taking the norm of the complex frequencies:

$$F_{C,N}[k] = \frac{1}{S} \times \sum_{j=1}^N \frac{1}{j} F[j] W[k, j] \quad (\text{A.8})$$

$$\text{where } S = \sum_{j=1}^N \frac{1}{j} W[k, j] \quad (\text{Normalization constant})$$

$$F_{C,N}[0] = F[0] \quad (\text{Special case for } k = 0)$$

$W[k, j]$  is a symmetric logarithmic filter with center at  $k$  and bandwidth from  $k_0$  to  $k_1$ :

$$\begin{aligned} 1/N &= 1/N \text{ Octave smoothing} \\ \Delta_N &= \sqrt{2^{(1/N)}} = 2^{(0.5/N)} \\ k_0 &= k/\Delta_N, \quad f_0 = f_k/\Delta_N \\ k_1 &= k \times \Delta_N, \quad f_1 = f_k \times \Delta_N \end{aligned}$$



Example at  $f_k = 1000\text{Hz}$  with  $1/N = 1/3$

$$\begin{aligned}\Delta_N &= \sqrt[2]{2^{(1/3)}} = 2^{(0.5/3)} = 1.12246\dots \\ f_0 &= 1000\text{Hz}/\Delta_N = 891\text{Hz} \\ f_1 &= 1000\text{Hz} \times \Delta_N = 1122\text{Hz}\end{aligned}$$

So that at 1000Hz there is an average from 891Hz to 1122Hz.

Such an average is equivalent to a FPPO (Fixed Point Per Octave) calculation.

### A.3.2.1 Filter shape

There is no standard filter shape for the  $W[k, j]$  filter in (A.8). The filter shape used in this implementation is the Blackman filter:

$$W[k, j] = \begin{cases} 0.42 + 0.5 \cos(\pi x) + 0.08 \cos(2\pi x) & , k_0/2 \geq k \geq 2k_1 \\ 0 & , \text{otherwise} \end{cases} \quad (\text{A.9})$$

where:

$$\begin{aligned}x &= \frac{2}{W_{\log}} |j_{\log} - k_{\log}| \\ j_{\log} &= \log_{10}(j) \\ k_{\log} &= \log_{10}(k) \\ x_{0.5} &= 0.40547865\dots \\ W_{\log} &= \frac{1}{x_{0.5}} (\log_{10}(k_1) - \log_{10}(k_0)) = \frac{1}{x_{0.5}} 2\Delta_{\log} \\ \text{so that } x &= x_{0.5} \times \frac{|j_{\log} - k_{\log}|}{\Delta_{\log}}\end{aligned}$$

the bandwidth of the filter has been adjusted for FWHM using  $x_{0.5}$ :

$$0.42 + 0.5 \cos(\pi x_{0.5}) + 0.08 \cos(2\pi x_{0.5}) = \frac{1}{2}$$

thereby  $W[k, j]$  has the following properties:

$$\begin{aligned}W[k, k] &= 1 \\ W[k, k_0] = W[k, k_1] &= \frac{1}{2} \quad (\text{bandwidth})\end{aligned}$$

### A.3.2.2 Time-windowing

The smoothing in (A.8) can be written in the general form (A.4):

$$F_{C,N}[k] = \frac{1}{N+1} \sum_{j=0}^N W[k, j] F[j] = \sum_{j=1}^N \frac{1}{j} F[j] \frac{1}{S} W[k, j]$$

At a frequency  $f$  we have a smoothing width:

$$\text{Smoothing-width}(f) = f_1 - f_0 = f \left( \Delta - \frac{1}{\Delta} \right)$$

Using (A.6) gives the the time-window-width (TWW) becomes:

$$\begin{aligned} \text{TWW}(f) &= \text{Time-window-width}(f) \\ &= (\text{Smoothing-width}(f))^{-1} \end{aligned} \tag{A.10}$$

$$\begin{aligned} &= \frac{1}{f} \times \frac{1}{\left(\Delta - \frac{1}{\Delta}\right)} \\ &= T \times \frac{1}{2^{(0.5/N)} - 2^{-(0.5/N)}} \end{aligned} \tag{A.11}$$

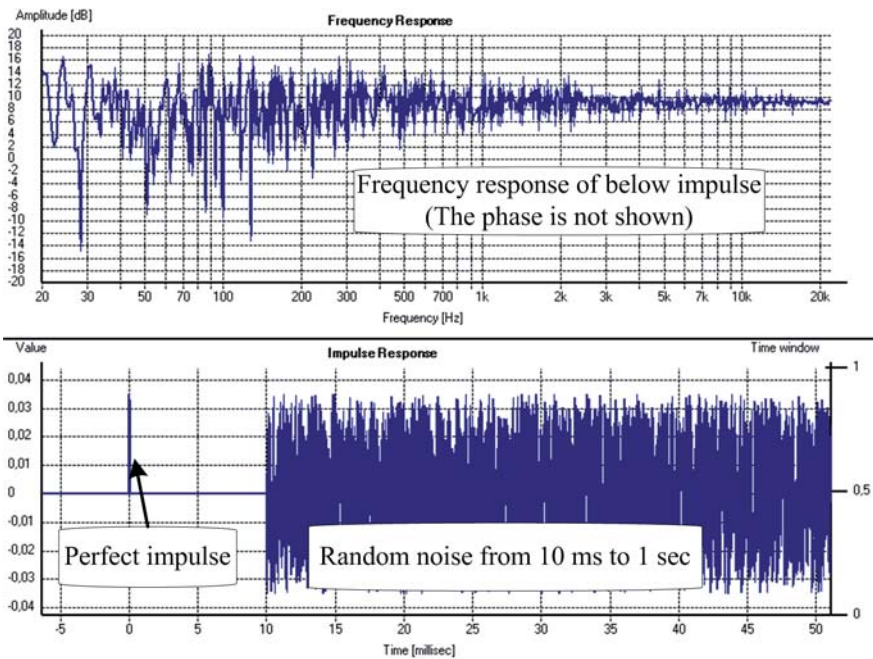
where  $T = \frac{1}{f} = \text{period of frequency [s]}$

The exact form of this time-window depends on the logarithmic window function  $W[k, j]$ , but this expression (A.11) shows that the width of the time-window is proportional to the time period. When windowing the impulse response using a symmetric window the half window-width is what determines the time-gating frequency. Eg. for  $N = 1, 3, 6$

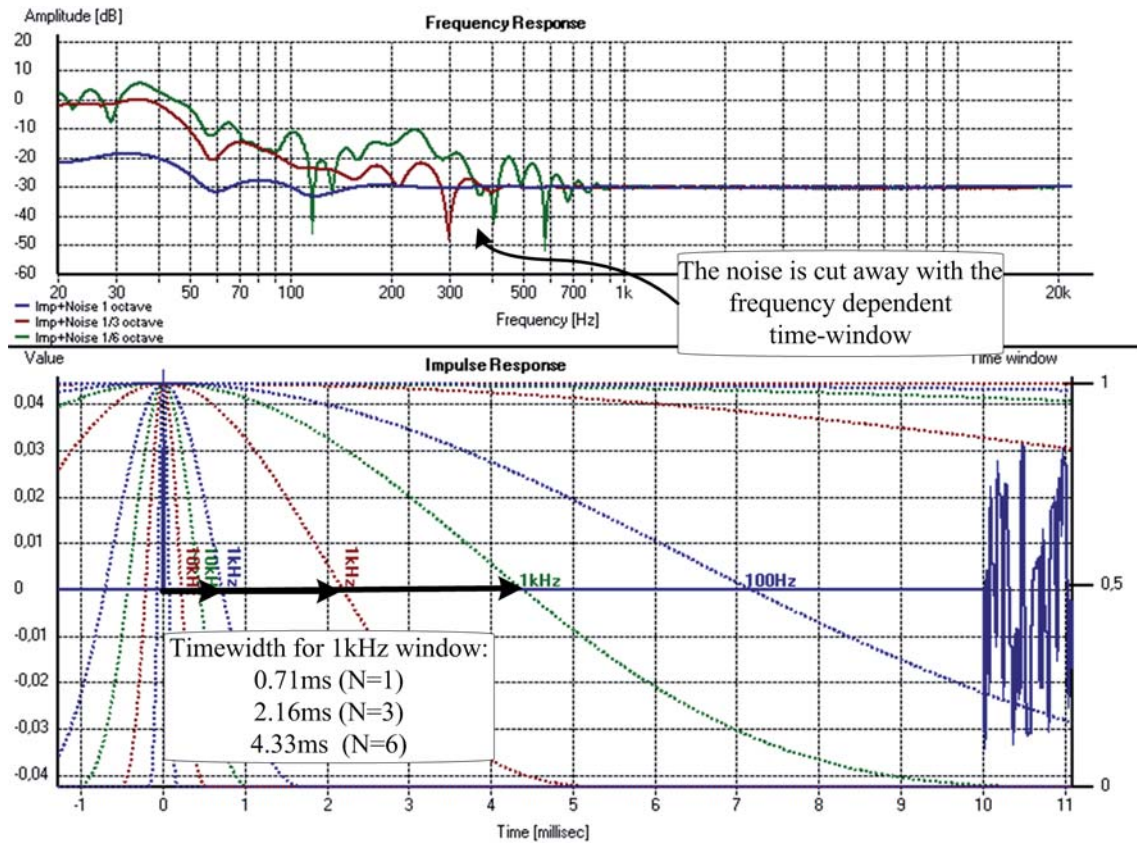
$$\begin{aligned} \frac{1}{2} \text{TWW}_{N=1} &= \frac{1}{2} \frac{1}{2^{(0.5/1)} - 2^{-(0.5/1)}} \times T \simeq 0.71 \times T \\ \frac{1}{2} \text{TWW}_{N=3} &= \frac{1}{2} \frac{1}{2^{(0.5/3)} - 2^{-(0.5/3)}} \times T \simeq 2.16 \times T \\ \frac{1}{2} \text{TWW}_{N=6} &= \frac{1}{2} \frac{1}{2^{(0.5/6)} - 2^{-(0.5/6)}} \times T \simeq 4.33 \times T \end{aligned}$$

### A.3.2.3 Visualizing with graphs

The smoothing concept can be visualized with graphs. As a test impulse a artificial perfect impulse followed by extreme noise can be used



Applying different smoothing cuts the noise away for different frequencies:



Here it has been chosen to display the time-window for each decade (1Hz, 10Hz, 100Hz, 1kHz, 10kHz)

### A.3.3 Absolute smoothing of discrete frequency values

The total response is calculated by making a running average in the frequency domain (smoothing). Assume the discrete complex frequencies in the frequency domain are:

$$F[k], \quad k = 0, \dots, N_{FFT}$$

Then the logarithmic absolute average is calculated by:

$$F_{A,N}[k] = \frac{1}{S} \times \sum_{j=1}^N \frac{1}{j} |F[j]| W[k, j] \tag{A.12}$$

where  $S = \sum_{j=1}^N \frac{1}{j} W[j]$  (Normalization constant)

$$F_{A,N}[0] = F[0] \quad (\text{Special case for } k = 0)$$

$W[k, j]$  is a symmetric logarithmic window function with center at  $k$ .

Such a smoothing CAN NOT be written as a convolution like in (A.4), since the function  $F[k] \mapsto |F[k]|$  is in play. A absolute smoothing can therefore not be seen as time-windowing.

# Index

Auto detect time-zero, [16](#)

Bandpass, [21](#)

Causal impulse, [16](#)

Exporting the Frequency Response, [8](#)

Exporting the Impulse Response, [9](#)

FFT-Frequencies, [8](#)

First peak, [16](#)

First positive peak, [16](#)

FPPO, [30](#)

Highest positive peak, [16](#)

Largest peak, [16](#)

Lowpass, [20](#)

Non-Equidistant, [8](#)

Normalization, [14](#)

Room-response, [18](#)

smoothing, [18](#)

Text-file options, [8](#)

Time zero, [15](#)

Time-window, [17](#)

wave-file, [9](#)

woofer measurement, [20](#)