Directivity measurements applied to DML (how to do it?)

ChristianV (Homeswinghome@diyAudio)

21 Nov 2024

Abstract

This document details how to use REW to build a directivity diagram (spinorama) of a DML.

Contents

This document was exported from a markdown file with pandoc. Done : November 21, 2024.

1 Introduction

2 years ago, I posted a paper (see link in post #1 or the pdf in post #6900 DML Material chart) that makes a synthesis of the main relations from the physics that are supposed to describe the behavior of a DML.

One important conclusion of the paper is the efficiency of a DML, its high frequency cutoff and its coincidence frequency are linked and can't be adjusted independently. Those figures change according to the plate characteristics not all in the desired way so that only a compromise is possible.

This is purely theoretical and if there are clues about what plays a role in the efficiency, there are almost no evidence in our DIYer world about the coincidence frequency and the high frequency cut off.

The directivity aspect and its relation to the coincidence frequency were probably not explained enough in the paper even if it was shared in other posts of this thread.

The coincidence frequency is the frequency at which the speed of the waves in the panel are equal to the speed of the sound in the air.

According to the theory:

- Below this frequency, the panel is supposed to have enough lobes to be considered omnidirectional.
- At the coincidence frequency, the emission is supposed to jump to 90°.
- Above, as the frequency increase, the main lobe angle decreases.

A directivity diagram, a spinorama, of the DML should show that.

The few directivity diagrams shared in diyAudio thread (see for exemple post #8160 from Eric) are the one from Tectonic or those from M Zenker's papers.

Some months ago, a discussion in the diyAudio thread about the content of the 1st reflections seen in IRs and the fact I have a panel with such a behavior that sounds not good let me to think again about the directivity and the possibility of some HF emissions at high angles (ie post #11896).

More recently, I was reading documents about VituixCAD where how to make directivity measurements with REW is explained… and it is so simple, at least for the measurements that I am still wondering not thinking about that before!

2 Tips for testing

Before going to details, I would like to share 2 tips from Steve (thank you Steve!) that make for me the DML tests much easier and relialable (compared to what I did before):

- The exciters are glued to the panel with PVA. With PVA it is strong enough to have a good HF transmission, there is no risk to brake the exciter when changing for a new test. The central area remains in addition free of any tape (what I used before).
- A panel without suspension is suspended with a double ribbon of large tape at the center of the upper side, each end of the ribbon sticked on one face of the plate. A sponge is at the top of the panel to damp the possible tape noise. It is a dish washing sponge aged by 2 or 3 dish washer cycles ;-)

3 Gated measurements with REW

So directivity measurement is no more than a collection of REW measurements where the direction of the loudspeaker is changed at each sweep. I experimented from -90° to +90° by 10° steps. Think to make also at least one at 180° on the rear side to check about a possible exciter noise (emission in a specific frequency band not visible from the front side).

The first key point is to adjust the "IR window" of REW so that only the signal before the 1st reflexion is kept in order to make a quasi-anechoic measurement.

When the mic is at 1m, the first reflection occurs in the range 3 to 4ms. Note that with a large panel, due to the proximity with the floor, it can be lower like 2ms.

Here is an example with a Visaton FRS8 full range speaker(with a DML which has less directivity than a full range, the amplitude of the 1st reflection is higher: **See figure 1**.

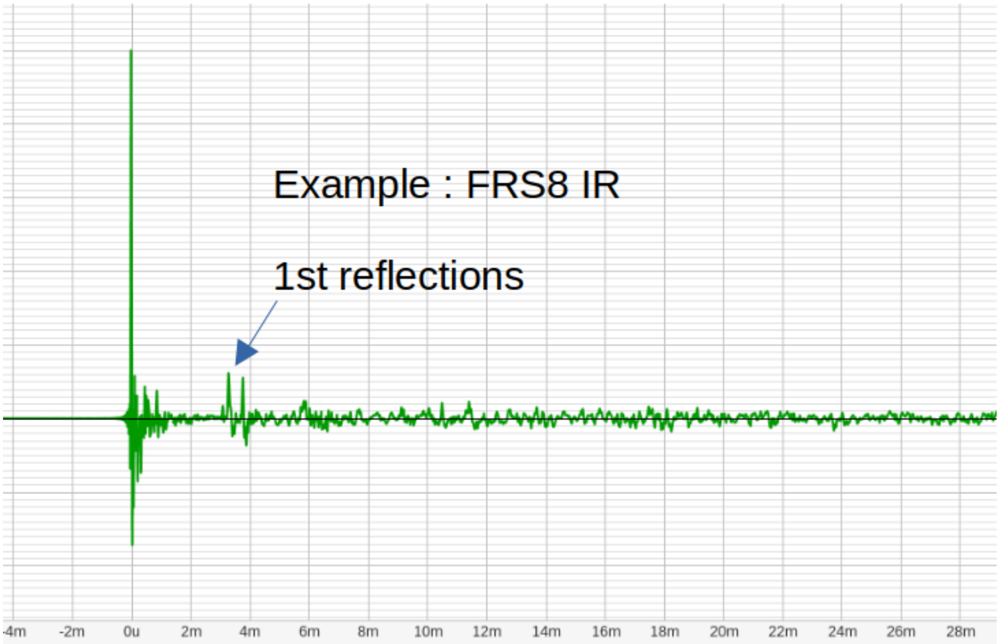

Figure 1: FRS8 IR

The IR going to be gated let say at 2 to 3ms, that means only the frequencies above 330Hz or even 500Hz are measured. In addition, even if REW shows more, only the multiple of these frequencies are "real", the other points are interpolated. This makes a kind of smoothing.

To adjust the IR window, check the IR for 0°.

The IR window can be set :

- After measurements, click on "IR windows" then fill the Left and Right values with the desired value. **See figure 2.**
- Before measurements, for all with the same values in "Preferences" > "Preferences" > "Analysis" tab > unchek the "Set window width automaticaly" box and enter the values just below. It supposes a first measurement was done. **See figure 3.**

In addition in the measurement window, choose for Timing "Set t=0 at IR start"... (I didn't do it in my first tests). Adjust also the file name options to add automatically some angle info or index to the measurement name. The angle value will have to be in the file name. VituixCAD ask also for the information "hor" or "ver" for the plane of the rotation. **See figure 4**

When all the measurements are collected, they are visible in the "All SPL" graph but not easy to examine from there. **See figure 5.**

Check each IR to be sure the window is correctly adjusted. When the angle is high, depending of the environment, with a USB mic, it is possible that REW does't apply the window on the right peak if the option t=0 at start above was not selected. A manual adjustment is needed (see in the "Tools" > "IR windows" "ref time" or IR window icon). It may happen also that at high angle some reflections come earlier. **See figure 6**

4 Export the measurements from REW in txt files

Then is the second key point which is to export them as a set of txt files to a tool that will build the "spinorama", the pseudo 3D view with the frequency and the angle as axis and the SPL as a color. **See figure 7**

Here are the settings I used for the export: **see figure 8.**

It is a functionality of VituixCAD to build a spinorama. VituixCAD has some requirements in the naming of the files to operate.

Being under Linux, it is not the way I have followed so I can't explain it fully. You can refer to the VituixCAD site or to the following ASR thread How to make quasi-anechoic speaker measurements/spinoramas with REW and VituixCAD

It was quicker for me in a first time to write a Python script that gathers the measurements. The advantage is the possibility to add on the graph some information specific to DML like an hypothesis of coincidence frequency or a possible frequency of exciter noise seen on the rear side. I intend to dive in VituixCAD installation under Linux later.

For standard loudspeaker, the gated FR are extended by a near field measurement (the mic is very close to the membrane). For a DML because of modes below the lower frequency of the gated response, it is not possible to proceed in the same way.

Note that VituixCAD recommend not adding smoothing to FR before export, it applies it after. For my script which is much less sophisticated, I applied a 1/6 octave smoothing before export.

Some correction of the names of the txt files may be needed to show clearly the angle and the plan.

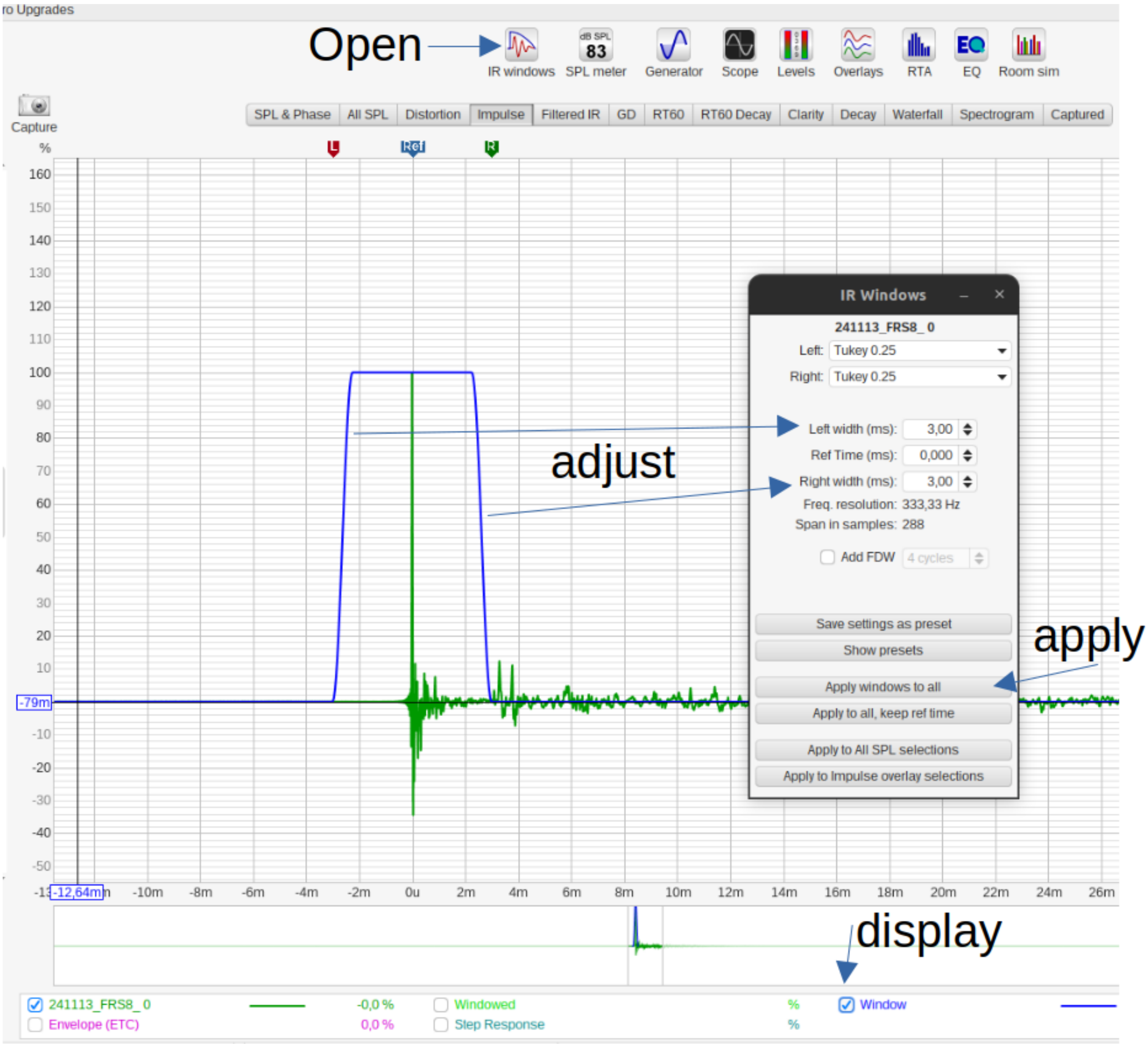

Figure 2: IR window setting

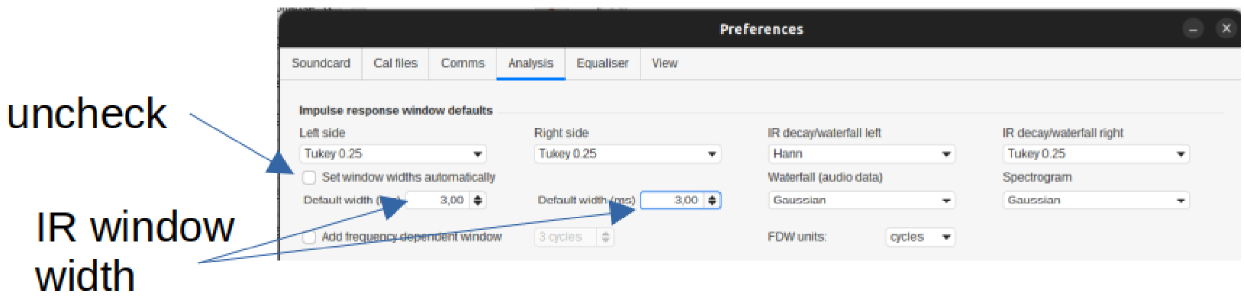

Figure 3: REW preferences

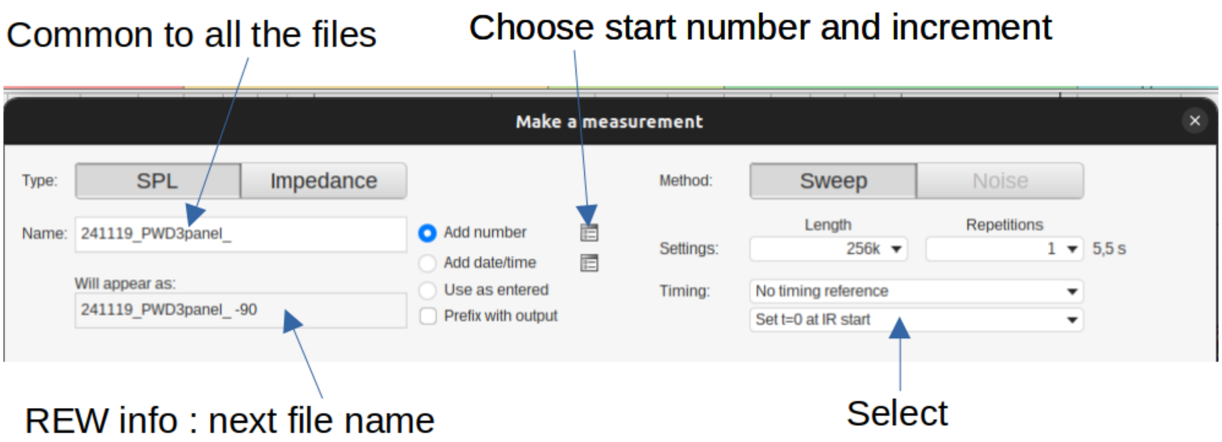

Figure 4: REW measurement options

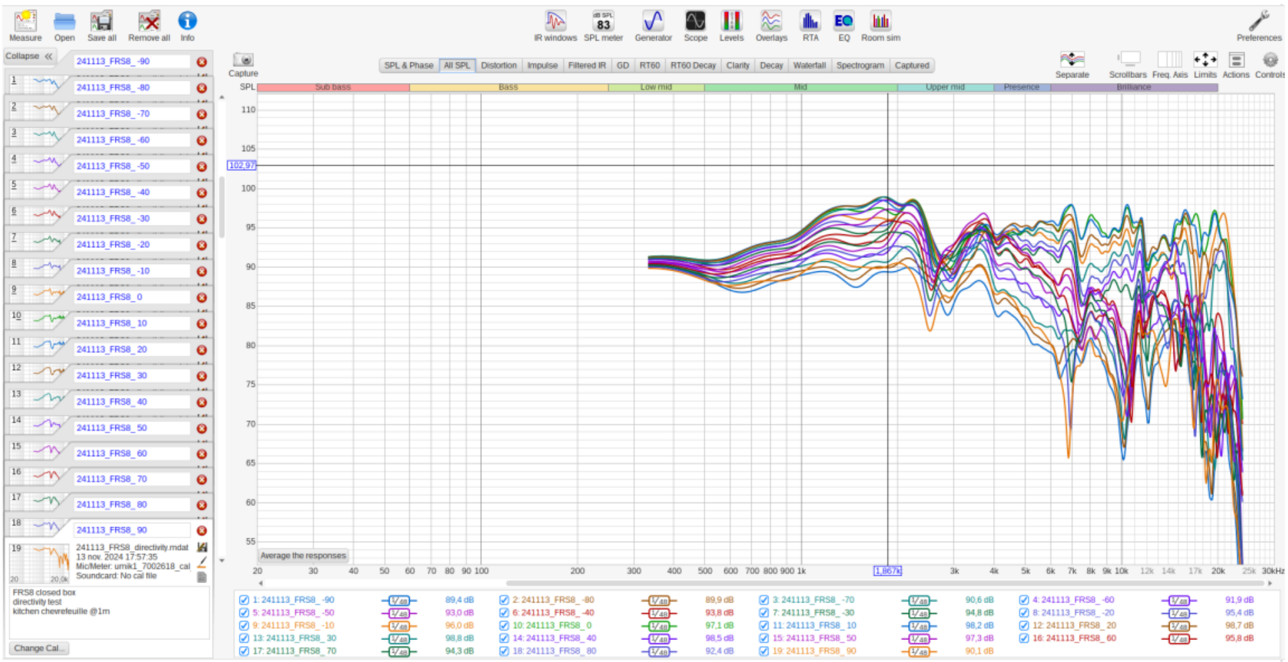

Figure 5: REW All SPL screen

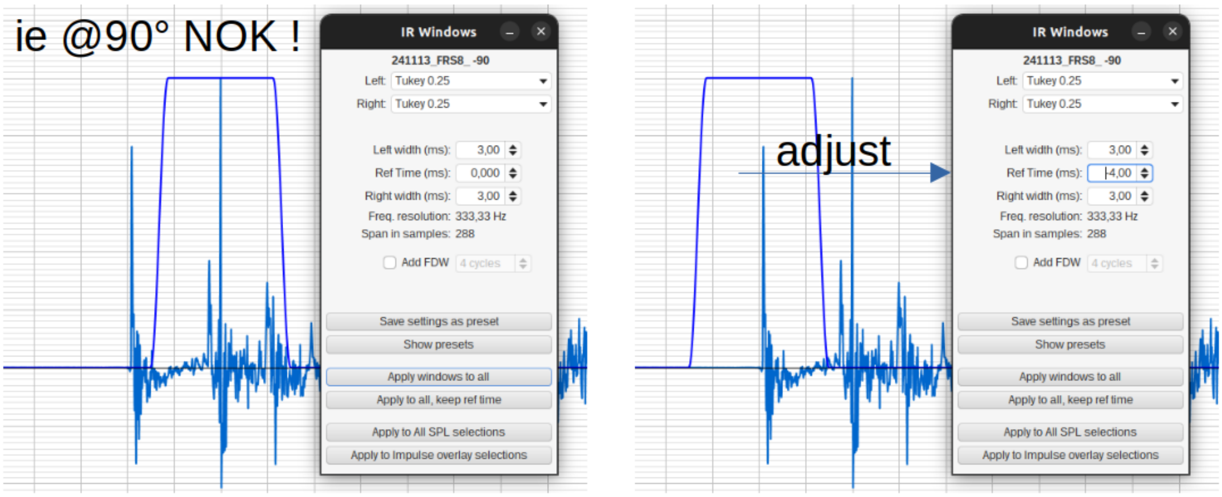

Figure 6: IR window reference time adjustment

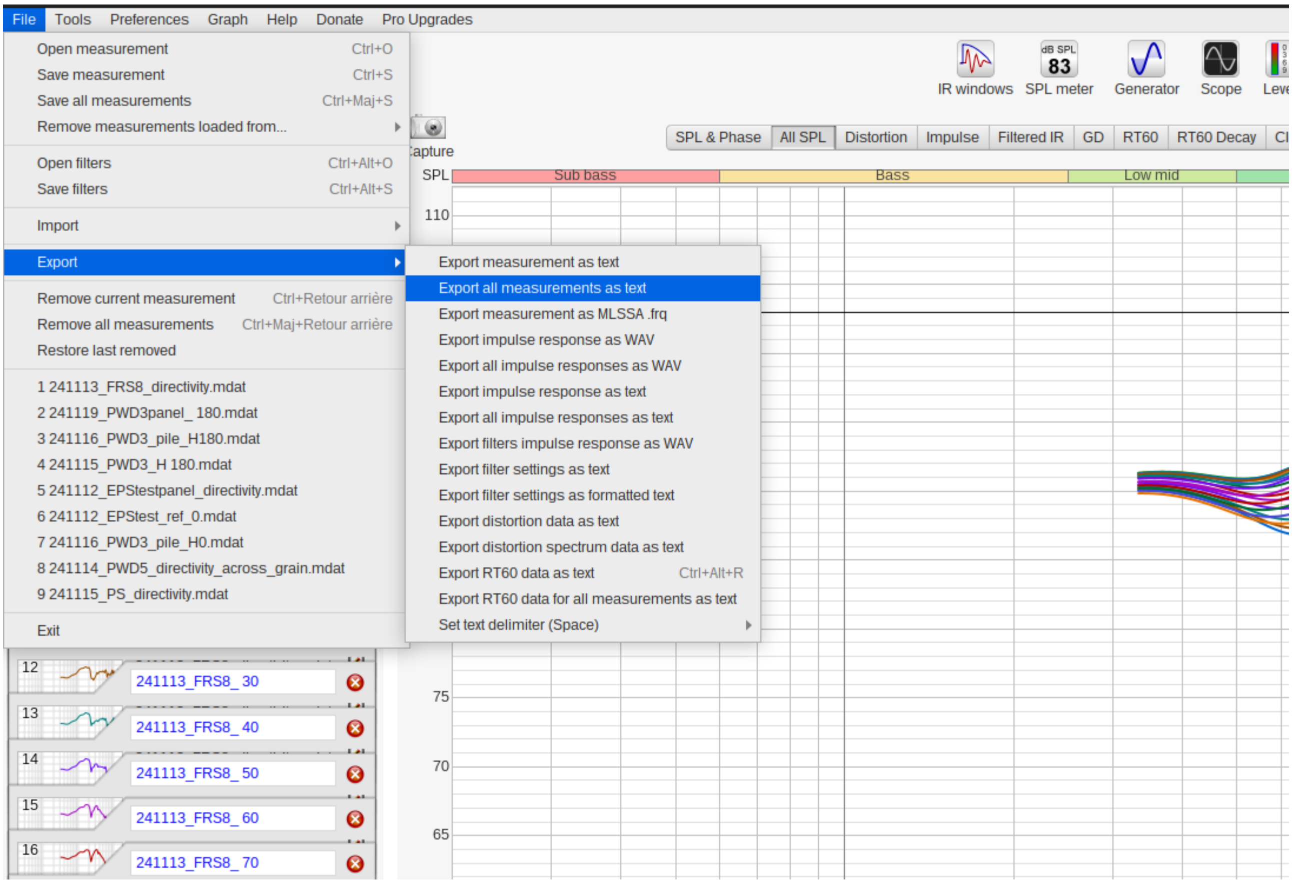

Figure 7: REW export menu

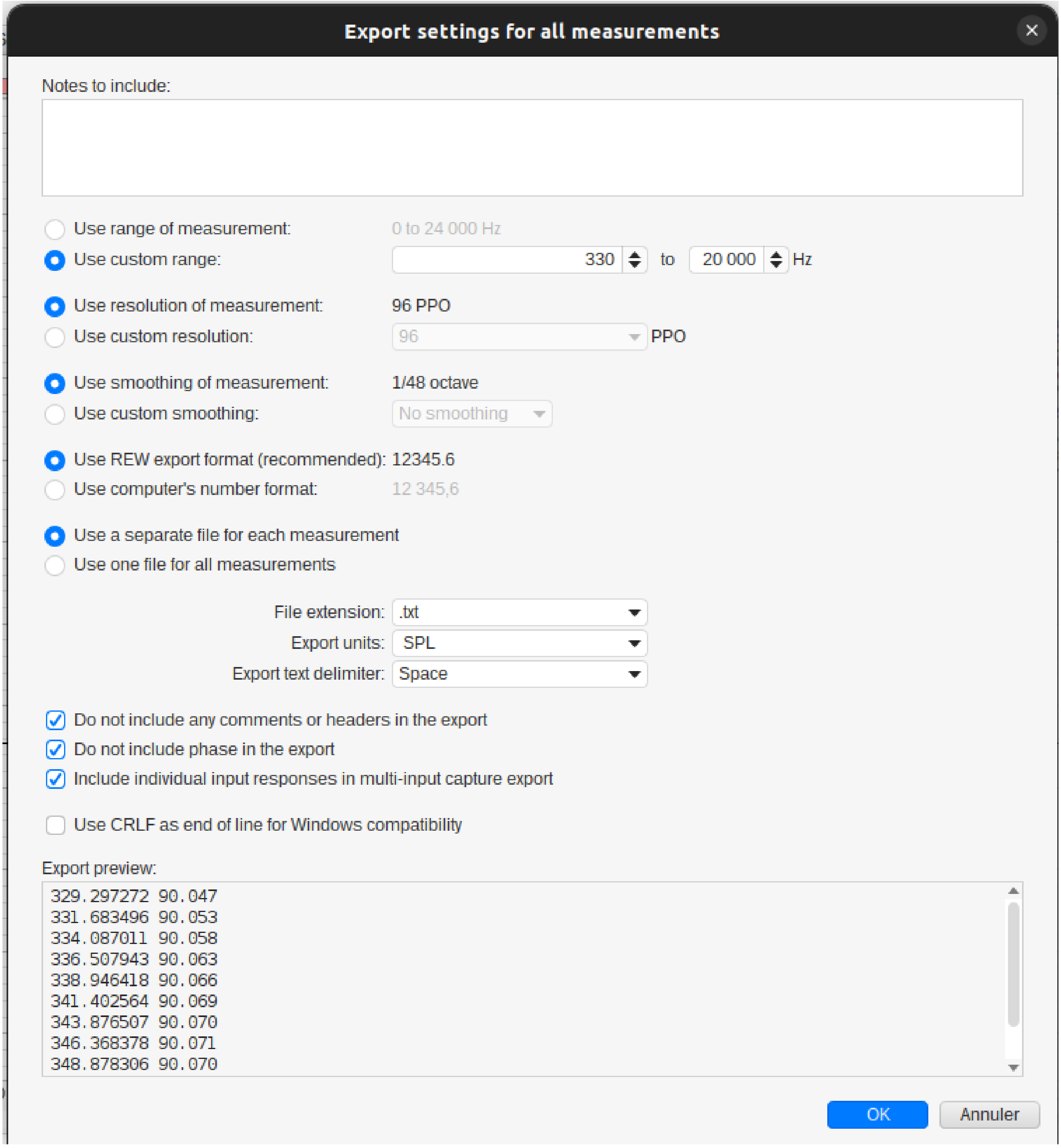

Figure 8: REW export as txt settings

5 Examples of directivity diagrams

Here is the directivity diagram for the FRS8 with this method: **see figure 9**.

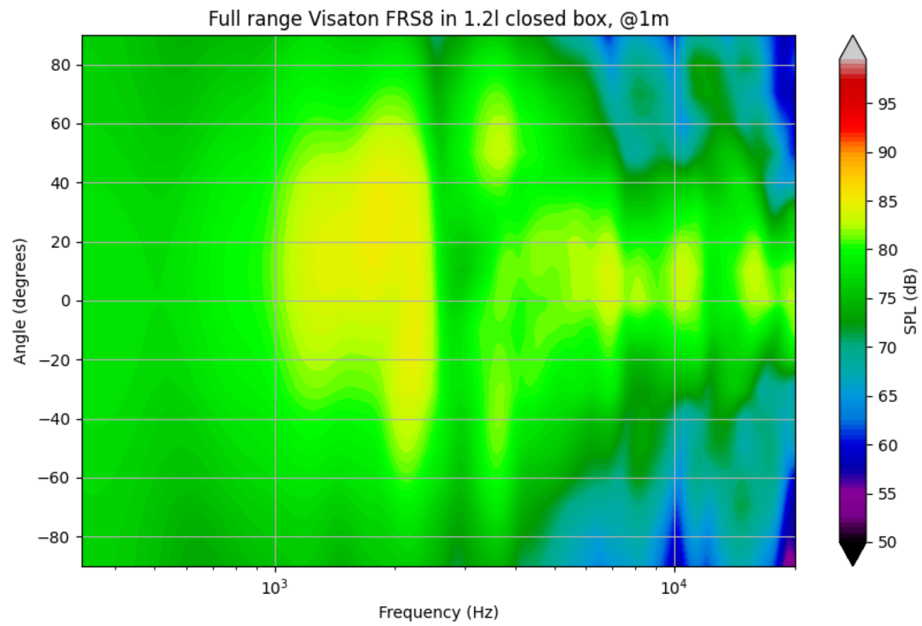

Figure 9: FRS8 spinorama

A standard approach is to normalize the diagram to the on axis values which gives: **see figure 10**.

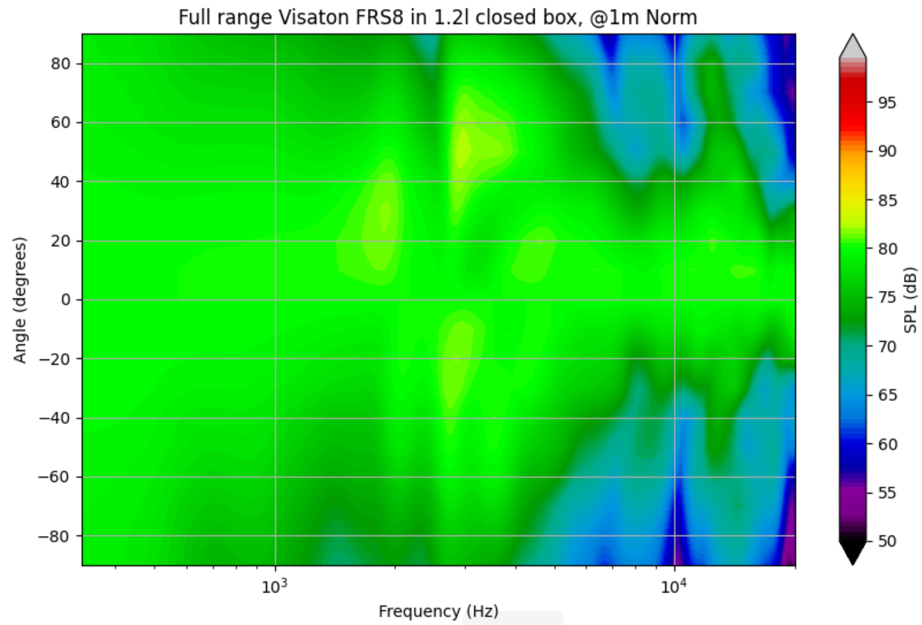

Figure 10: FRS8 normalized spinorama

6 A word of conclusion

If those measurements were made in my kitchen or in my living room, it will work from a garage or any other room. Joke aside, any location or arrangement that increases the time for the 1st reflection is better.

7 Annexe : Python script source code

spinorama.py

```
import os
import math
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import cm

# Function to read data from a file
def read_data(file_path):
    data = pd.read_csv(file_path, sep='\s+', header=None, names=['Frequency', 'SPL'])
    return data

# Function to extract angle from filename
def extract_angle(file_name):
    angle_str = file_name.split('_')[-1].split('.')[0]  # Get the angle part from the filename
    return int(angle_str)

EQ = ""
colorscale = 'nipy_spectral'
# colorscale = 'turbo'
# /!\ -------EQ option-----------
EQoption = 'Norm' # Can be 'without', 'EQed', 'Norm'
fc=0. ; frear=0

# Directory containing the text files ; plot title; fc, frear
directory = './PWD3_DAEX25_on_piles'; title = 'Plywood 3mm H25xV125cm test panel (raw), V axis=grai

# Initialize lists to store frequency, angle, and SPL data
frequencies = []
angles_raw = []
spl_values = []

# Read all text files in the directory
for file_name in os.listdir(directory):
    if file_name.endswith('.txt'):
        file_path = os.path.join(directory, file_name)
        data = read_data(file_path)
        print(file_name)
        # Extract angle and append data
        angle = extract_angle(file_name)
        # print(angle)
        frequencies.append(data['Frequency'].values)
        spl_values.append(data['SPL'].values)
        angles_raw.append(angle)

# Convert lists to numpy arrays
frequencies = np.unique(np.concatenate(frequencies))
# print(frequencies)
angles = np.sort(np.unique(angles_raw))
# print(angles_raw)
# print(angles)
spl_values = np.array(spl_values)
# spl_values = spl_values - np.mean(spl_values) + 80. # set mean to 80dB
# print(spl_values.ndim)


# Create a meshgrid for frequencies and angles
```

```python
F, A = np.meshgrid(frequencies, angles)

# Create a grid
SPL_grid = np.zeros((len(angles), len(frequencies)))

for i, angle in enumerate(angles):
    # Find the index of the angle in the angles list
    index = np.where(angles_raw == angle)[0][0]
    # print(i, angle, index)
    SPL_grid[i, :] = spl_values[index]
    # SPL_grid[i, :] = np.interp(frequencies, frequencies[index], spl_values[index])

# Adjust gain to have a 80dB mean value in the -20° to 20° range
gain = 80 - np.mean(SPL_grid[7:10,:])
SPL_grid[:,:] = SPL_grid[:,:] + gain

if EQoption == 'EQed':
    EQ=' EQed'
    for j, freq in enumerate(frequencies):
        gain = 80 - np.mean(SPL_grid[7:12,j])
        SPL_grid[:,j] = SPL_grid[:,j] + gain

if EQoption == 'Norm':
    EQ=' Norm'
    for j, freq in enumerate(frequencies):
        gain = 80 - np.mean(SPL_grid[9,j])
        SPL_grid[:,j] = SPL_grid[:,j] + gain

# Define the levels for the contour plot
levels = np.linspace(50, 99.5, 100)  # 100 levels from 50 dB to 100 dB
#print(levels)



# Plotting the spinorama
plt.figure(figsize=(10, 6))
plt.contourf(F, A, SPL_grid, levels=levels, cmap=colorscale, vmin=50., vmax=100, extend='both')
plt.colorbar(label='SPL (dB)')
plt.title(title + EQ)
plt.xlabel('Frequency (Hz)')
plt.ylabel('Angle (degrees)')
plt.xscale('log')  # Optional: log scale for frequency
plt.grid()
if EQoption != 'none':
    if fc != 0:
        for alpha in range(-90, 0, 5):
            f = fc / math.sin(alpha*math.pi/180)**2
            if f < 20000:
                plt.plot(f,alpha,'k+')
        for alpha in range(90, 0, -5):
            f = fc / math.sin(alpha*math.pi/180)**2
            if f < 20000:
                plt.plot(f,alpha,'k+')
        plt.plot([fc,fc], [-80,80], color='black', linestyle='dotted')
    if frear != 0:
        plt.plot([frear,frear], [-90,90], color='black', linestyle='dotted')
        # plt.plot(frear,0,'ko')
plt.show()
```