TTT4250 - Acoustical Measurement Techniques

Sound insulation

performed by

Erlend Kristiansen Berg
Silje Toftevåg Urke
Stephanie Evers
Hugo Suleyman

Report by

Erlend Kristiansen Berg

Department of Electronic Systems
Norwegian University of Science and Technology

Summary

In building acoustics, the choice of sound absorbent materials has distinctive characteristics and work better for certain frequencies. This lab will therefore find the characteristics of a test material consisting of plywood and rubber, by measuring the sound pressure levels in a source and receiver room. Through measurements and calculations, the normalized level difference, standard level difference as well as the apparent sound reduction index was found and plotted and show a clear correlation with the pressure plot for both rooms. The test material shows a significantly higher damping for the higher frequencies, and the weighted sound reduction index, R_w, was found to be 33dB with the reference curve being shifted 19dB down. The sum of the unfavorable deviation was found to be 23.3dB.

Contents

1 Introduction

This report will guide you through how to find the sound pressure level for the two rooms, the sound reduction index, the various kinds of level differences as well as supply the necessary theory for the lab. The lab will be performed in two reverberate rooms separated by a test material. The object is to find to find the test materials effect on the sound radiating from the source room to the receiving room. The report will show in detail how to set up the equipment, how the measurements were performed, as well as give the necessary theory and equation to be able to reproduce the results.

2 Theory

This chapter will present the necessary theory to understand the abbreviations used, the necessary equations to reproduce the lab, as well as general theory about level difference, sound reduction and the pressure method used to calculate the specific values.

2.1 Background Noise and Correction

The background noise in the receiving room must be sufficiently low as to ensure quality measurements. If the sound power coming through the sound absorbent material from the source room is lower than 10dB higher than the background noise in the receiver room, correction is needed. This argument must be valid in any frequency band. If this is not fulfilled, correction is needed. The correction is given as

$$L = 10\log\left(10^{L_{sb}/10} - 10^{L_b/10}\right) \tag{2.1}$$

where L is the correct sound power, L_{sb} is the sound power of the signal and L_b is the sound power of the background noise. If the difference in sound power between the signal and background noise is less than 6dB, use the correction of 1.3dB corresponding to a difference of 6dB for the bands that have the difference less than 6dB.

Since the measurements are performed for only one speaker at the time, the location of the speakers influences the result. The positioning of the speakers in the source room should be placed in the corners than a central point, so that as many wave modes as possible is formed. The microphones are placed so that they all are in the diffuse field, meaning the sound energy density is uniform throughout the space. Diffuse sound fields do not exist in real box-shaped enclosures with stationary objects inside it, so to get a solid average for the sound power, the sound power level is averaged over five microphones' positions and two speaker positions for a total of 10 measurements. The average sound pressure level in a room is given as

$$\overline{L} = 10\log_{10}\left(\frac{1}{n}\sum_{j=1}^{n} 10^{L_j/10}\right) \tag{2.2}$$

where L_j is the sound pressure level at microphone position j in the room. Further, if $\overline{L_1}$ and $\overline{L_2}$ are respectively the average sound pressure levels in the source-and receiving room, the level difference D can be found as

$$D = \overline{L_1} - \overline{L_2} \tag{2.3}$$

The level difference D is the difference between the sound pressure level in the source room and the receiving room, measured and averaged over all the microphone positions.

2.2 Level Difference, Reverberation Time and Sound Reduction Index

In an enclosed space, if a sound source stop emitting energy, it will take time for the sound to become inaudible. This prolongation of the sound in the enclosed space caused by reflections from the walls in the room is called reverberation. The time it takes for the energy from the sound source to drop 60dB is called the reverberation time and is denoted as T_{60}. When the difference between the sound source and the background noise is less than 60dB, the T_{20} and T_{60} may be used. These two values are used to estimate the reverberation time, by extrapolating the observations of decays over 20dB(T_{20}) or 30dB(T_{30}). This notation may seem to note the time the sound use to decay with 20dB or 30dB, but this is not the case, it is however an estimation of T_{60} from a fragment of the total decay.

2.3 Standardized Level Difference

With the level difference, D, and the reverberation time, T, the standardized level difference D_{nT} can be calculated as

$$D_{nT} = D + 10 \cdot \log_{10} \frac{T}{T_0} \tag{2.4}$$

where T is the reverberation time T_{60} and $T_0 = 0.5$s is the reference reverberation time. The standardized level difference takes account of all possible sound transmission paths between the two rooms and gives a correlation to the measured impression of the airborne sound insulation.

2.4 Normalized Level Difference

The normalized level difference, D_n, is calculated as

$$D_n = D - 10 \cdot \log_{10} \frac{A}{A_0} \tag{2.5}$$

where A is the equivalent sound absorption area of the receiving room, as seen in Equation (2.6), and $A_0 = 10m^2$ is the reference sound absorption area. The level difference measured *on situ* vary vastly due to different room sizes encountered. By normalizing the results to a reference absorption, these differences are minimized. The equivalent sound absorption area is given by Sabine‘s formula

$$A = 0.16 \cdot \frac{V}{T} \tag{2.6}$$

where V is the volume in the receiving room and T is the reverberation time T_{60}.

2.5 Apparent Sound Reduction Index

The apparent sound reduction index, R', is the field measurements of the sound reduction index. It is calculated as

$$R' = D + 10 \cdot \log_{10} \frac{S}{A} \tag{2.7}$$

where S is the area of the separating element under test and A is as mentioned the equivalent sound absorption area. From the apparent sound reduction index, the weighted sound reduction index can be found. A reference curve defined in ISO 717-1[1] is used with the apparent sound reduction index to find the unfavorable deviation. To find the reference curve suited for the measurements done, the sum of the unfavorable deviation must be no larger than 32dB, but as high as possible. the unfavorable deviation is calculated by checking for the difference from the reference curve and the apparent sound reduction index. For every positive deviation value, this value is listed as the unfavorable deviation for the respective frequency band, but for every negative difference, the value is set to zero. The sum of all the unfavorable deviation values is then summed, and if the value is above 32dB, the reference curve is shifted 1dB down for all values, and the procedure is repeated. This is repeated until the sum of unfavorable deviation is as high as possible without exceeding 32dB. When this is satisfied, the weighted sound reduction index, R'_w, is found as the value from the reference curve at 500Hz.

2.6 Airborne Sound Reduction Index and the Incidence Mass Law

For an infinite wall made of a single layer of homogeneous material, the sound reduction index, R, for a structure can in theory be predicted by the *Mass Law* given as

$$R(\theta) \approx 10 \cdot \log_{10} \left(1 + \left(\frac{\omega \rho_s}{2\rho c} \cos\theta \right)^2 \right) \tag{2.8}$$

where ρ_s and ρ is the surface density for respectively the test material and for air given in kg/m^2. ω is the angular frequency, θ is the angle of incidence and c is the speed of sound in air. For $\theta = 0$, R_0 is called the normal incidence mass law. The rooms in the lab does not have an infinite wall between them, so a more realistic case is where a wall of finite size separate the two rooms. The sound signal from the source room will radiate through the separating wall and into the receiving room. When sound comes from all different directions, the random incidence mass law, R_{random}, or alternatively the field-incidence mass law, R_{field}, is used to estimate the sound reduction index for the separating wall. They are given as

$$R_{random} \approx 10 \cdot \log_{10} (0.23 R_0) \tag{2.9}$$

and

$$R_{field} \approx R_0 - 5 \tag{2.10}$$

The difference between the two, is for what angles of incidence they are valid for. R_{random} is valid for all incidence angles from 0° to 90°, while R_{field} is valid for incidence angles between 0° to 78°. These values are only valid if R_0 is above 15dB.

3 Method and Equipment

This section will cover in detail how the lab were performed. It will include a detailed list of equipment as well as figures describing the setup and positioning of the used equipment.

3.1 Equipment and Setup

The used equipment for this lab can be seen in Table 3.1, and a figure of the source- and receiving room can be seen in Figure 3.1. The source room is in room D0016 and receiving room is in room D0017, both found at NTNU, Gløshaugen, Elektro D+B2. The two rooms are connected by a hole, as illustrated with a orange line in Figure 3.1. In this separating hole, a sound absorbing material is fixed, consisting of rubber and plywood. This material will be the test material for the lab. Before any measurements were performed, the microphone was calibrated using the automatic calibration function on the Nor150 with the help of a calibrator sending insert here . For the source room, speakers were placed in the two locations shown in Figure 3.1, marked as $S1$ and $S2$. $S2$ is placed in the corner intersecting the roof-plane and the two walls next to it, while $S1$ is placed in the corner intersecting the floor and the two walls. To produce the noise, a noise generator were connected to an amplifier that further sent the noise signal to the speaker of choice. To ensure no interference with the measurements, the subjects performing the lab were found outside the reverberate rooms, with the doors almost close.

Table 3.1: Equipment list.

Equipment	Model number/type	Serial number
Sound and Vibration Analyser	norsonic Nor150	15030749
Pressure Calibration	Norsonic Nor1256	
Foam windscreen for microphone		
Microphone stand	K&M	
Cables	Norsonic Nor 1408A 5 meter	
Microphone/Measurement probe	Norsonic Type 1201/30490	
Loudspeakers source room	Made in house	
Loudspeaker receiving room	Made in house	
NAD Electronics LTD	Model 208 Stereo Power Amplifier	
Noise Generator	Type 1405 AN 2005 B037	
Ruler	1 meter	
Two layered isolation sample	50%Plywood & 50% Rubber	

3.2 Measuring the Sound Pressure

To ensure solid results, five different microphone positions were used. To ensure that the microphones are in the diffuse field, a minimum distance of 0.7m between each microphone, 0.5m between any microphone and room boundary, and 1.0m between any microphone and the sound sources had to be supported for all microphones. Each microphone position for both the receiving- and source room was chosen, and marked using tape, before any measurement were done. The microphone was then placed in the receiving room at the first microphone position, and checked if

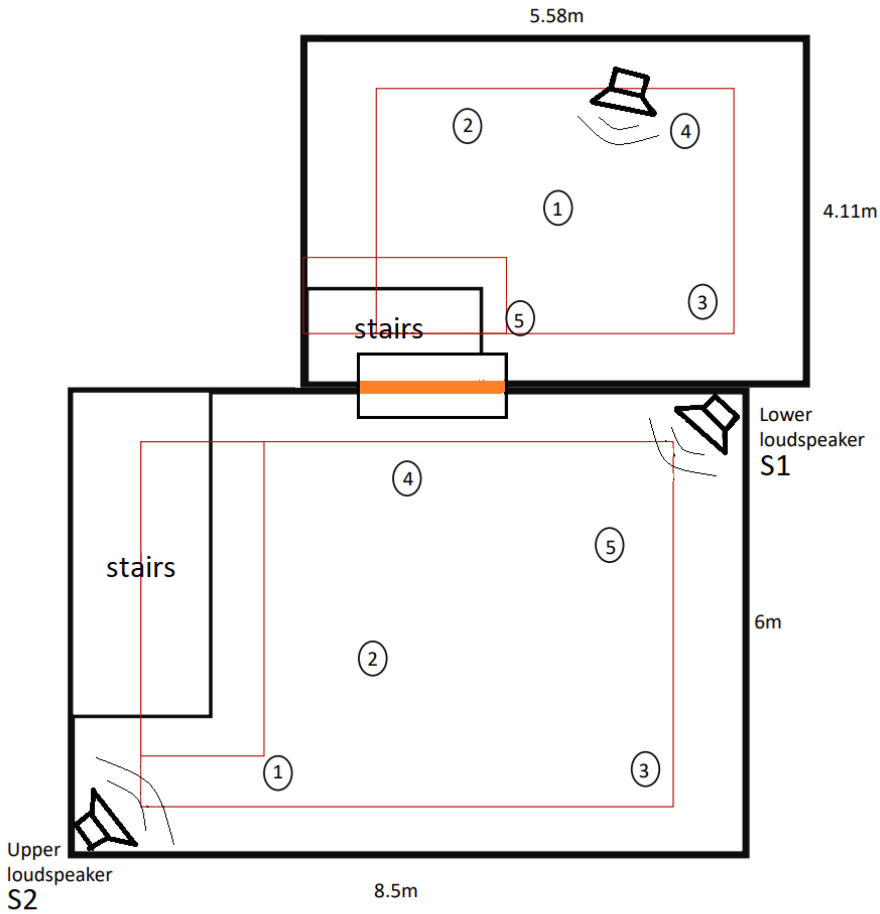

Figure 3.1: Illustration of the two rooms with the positioning of sound source $S1$ and $S2$, microphone positions and where the separating hole, marked in orange, is placed relative to the two rooms. The upper and smaller box is the receiving room while the lower box is the source room. The enumerated circles marks the microphone position used for the two rooms. The speaker icon in the receiving room mark the location for the speaker used when measuring the reverberation time. The height for the two rooms are not listed due to it being a 2D drawing. The two boxes are illustrated by Stephanie Evers.

all the distance parameters, as mentioned above, were satisfied. The lower speaker, $S1$, was then connected to the amplifier, and after everyone had left the receiving room and closed the door as much as possible without applying pressure on the microphone cable, the speaker were excited with the noise signal. The noise producer is actively producing random sound, so to not damage the amplifier, the following procedure must be done when using the noise generator. First, the button that stop the noise generator to send sound must be pushed and hold. While the button is pushed, the amplifier can be turned on. When the amplifier has finished its startup and the indicator light turns green, the button on the noise generator can be released, and the speaker will be excited with the random noise signal.

The noise inside the room, excited by speaker, $S1$, was then recorded for 10 seconds. then, the noise signal was turned off by reversing the process of how to turn it on, and the speaker connected to the amplifier was changed from $S1$ to $S2$. The recording was then repeated, and then, the microphone position was changed to the second position. This was then repeated for all 5 microphones, giving a total amount of 10 recordings for the source chamber.

After the measurements were performed in the source chamber, the same was done for the receiving chamber. The noise signal was still excited through speaker $S1$ and $S2$, but now, a sound absorbing material was separating the two rooms. The measurements for the receiving room was done in the exact same manner as for the source chamber. After all pressure measurements were performed, the background noise in the receiving room was calculated by recording the background noise at two different microphone positions. The used positions was microphone position 3 and 5.

3.3 Measuring the Reverberation Time

The next step is to measure the reverberation time for the receiving chamber. This is done by exciting the speaker inside the receiving room with the noise signal in the same manner as for the earlier method. The Nor150 sound and vibration analyzer was then set to measuring the reverberation time, and after letting the noise signal play in the room for a while, the button stopping the sound inside the room was pressed and held down, so that the reverberation time could be measured. This was done two times in the same microphone position. The fifth Microphone position were used for this.

4 Results

This section will cover the calculated results from all the measurements. It will include graphs and tables holding the key values for this lab. All post processing were done using Python[6].

4.1 Sound Pressure Levels and Correction

The SPL(Sound Pressure Levels) values were calculated for both rooms and plotted as can be seen in Figure 4.1. The difference between the background noise and the signal are also plotted in Figure 4.1. For the receiving room, the signal has need for correction, this is because the signal is not above the 10dB line for all frequencies in the third octave intervals, as can be seen by the value dropping below the red line at 5KHz.

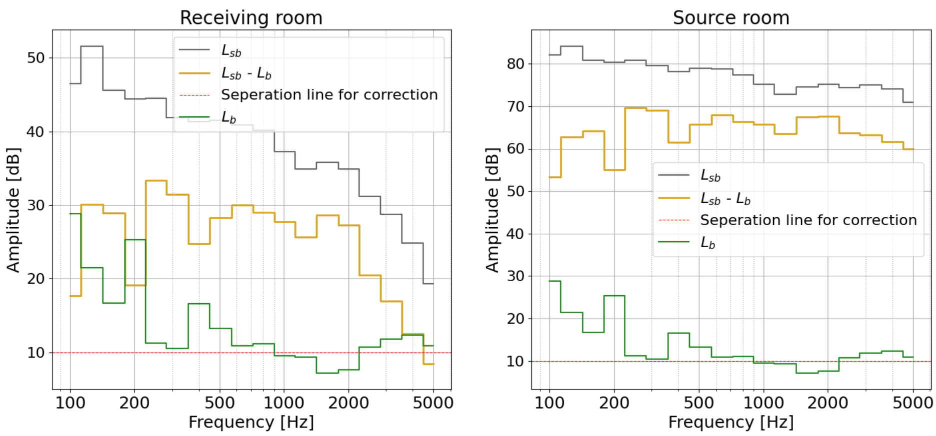

Figure 4.1: Sound Pressure Levels for the source-and receiving room. L_{sb} is the SPL value for the noise signal and L_b is the SPL value for the background noise. The goldenrod-colored line, marked as $L_{sb} - L_b$, is the difference between the signal and background noise, and if this line goes below the striped, red line, correction is needed.

4.2 Level Differences and Sound Reduction Index

The normalized level difference, D_n, standardized level difference, D_{nT} and the apparent sound reduction index, R', was found and plotted as seen in Figure 4.2. From R', the shifted reference curve and the unfavorable deviation is found and can be seen in Figure 4.2. The reference curve had to be shifted down 19dB to follow the arguments told in the Theory chapter for the unfavorable deviation and the reference curve. The sum for the unfavorable deviation is then 23.3dB. The weighted sound reduction index, R_w, was then found to be 33dB by looking at the reference curve at 500Hz.

From the normal incidence mass law, the random incidence mass law and field-incidence mass law was found and plotted in Figure 4.3 with the apparent sound reduction index.

Frequency[Hz]	Unfavorable Deviation[dB]
100	0
125	0
160	0
200	0
250	0
315	0
400	0.6
500	1.1
640	1.8
800	3.2
1000	3.8
1250	5.0
1600	4.4
2000	3.2
2500	0.2
3150	0
4000	0
5000	0

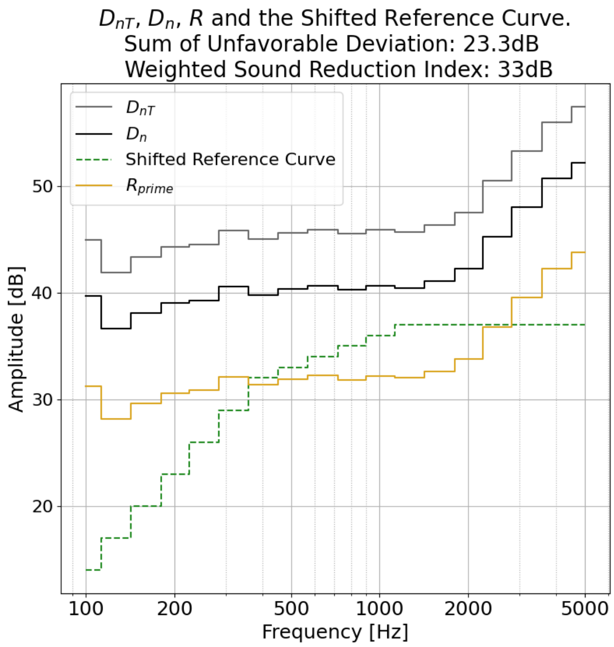

Figure 4.2: Plot for the normalized level difference, D_n, standardize level difference, D_{nT}, the apparent sound reduction index, R', and the shifted reference curve. The reference curve is shifted down 19dB and the unfavorable deviation can be seen in the table on the left with a sum of 23.3dB. The weighted sound reduction index is 33dB.

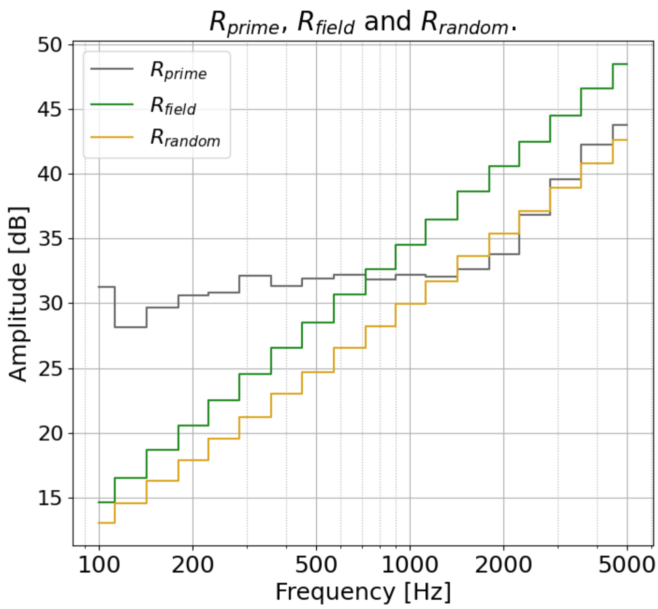

Figure 4.3: Plot for the apparent sound reduction index, R', relative to the random incidence mass law, R_{field}, and field-incidence mass law, R_{random}.

5 Discussions

From the sound pressure levels for the receiving room correction was needed. This was only due to a minor break on the separation line at 10dB in the 5KHz band. For R_{random} and R_{field} the latter has a steeper curve. Comparing them to the apparent sound reduction index, it is clear that for the lower frequencies the curve shows that both R_{random} and R_{field} have a lower sound reduction value than for the measured R.

For this lab, there exist small room for error. The reverberant chambers are large in size, and the only stationary objects were inside both rooms during recording. Possible errors can be related to change in background noise between the measurements and that the doors were not closed properly so to not damage the microphone wire. From looking at the level differences and the sound pressure levels for both rooms, it is visible that the higher frequencies have been severely dampened by the separating wall. Also, a smaller damping is visible for the 125Hz signal, as is seen by the higher spike in the plot for the receiving room and in the level difference. It is also clear that for frequencies above 1KHz, the estimate for R_{random} follow the curve for R with only minor differences. The separating wall is therefore highly efficient for higher frequencies, and less for the lower frequencies. Since the material was fastened to the hole in the wall, any possibilities of checking the thickness or adjusting the thickness was impossible.

6 Conclusions

By measurements and calculations, for the higher frequencies, the material under test is highly more efficient than that of the lower frequencies. The normalized level difference, standard level difference as well as the apparent sound reduction index was found and plotted and show a clear correlation with the pressure plot for both rooms. The weighted sound reduction index, R_w, was found to be 33dB with the reference curve being shifted 19dB down. This resulted in a sum for the unfavorable deviation at 23.3dB.

Bibliography

[1] NS-EN ISO 717-1:2013, Akustikk Vurdering av lydisolasjon i bygninger og av bygningsdeler, Del 1: Luftlydisolasjon (ISO 717-1:2013), published: 2013.06.01

[2] Acoustic Glossary, Sound Insulation Terms and Definitions, `SoundInsulationTermsandDefinitions`

[3] Airborne Sound Insulation, MEASUREMENT WITH THE XL2 SOUND LEVEL METER, `http://www.nti-audio.com/Portals/0/data/en/NTi-Audio-AppNote-Airborne-Sound-Insulation-Index-with-XL2.pdf`

[4] TTT4250 Acoustical Measurement Techniques, Laboratory Exercise Compendium, Robin André Rørstadbotnen

[5] NS-EN ISO 16283-1:2014, Akustikk, Feltmåling av lydisolasjon i bygninger og av bygningsdeler, Del1: Luftlydisolasjon(ISO 16283-1:2014), ICS 91.120.20

[6] Python, Version 3.9.0, Python Software Foundation, `https://www.python.org/downloads/release/python-390/`

A Appendix

Table A.1: Table containing all the raw data from the pressure methods done in the source and receiving room. The first 10 measurement(meas) are for the source room, while the rest are from the receiving room. The description containing the microphone- and speaker position can be seen in Table B.1.

Meas	Duration	Unit	$Lfeq$ 100 Hz	$Lfeq$ 125 Hz	$Lfeq$ 160 Hz	$Lfeq$ 200 Hz
1	00:00:10.000	dB	83.2	88.0	82.5	82.5
2	00:00:10.000	dB	83.2	85.9	81.3	79.5
3	00:00:10.000	dB	78.7	79.6	78.0	79.1
4	00:00:10.000	dB	82.8	86.4	81.0	81.2
5	00:00:10.000	dB	83.2	83.2	81.7	80.7
6	00:00:10.000	dB	81.1	80.8	80.3	78.6
7	00:00:10.000	dB	79.2	82.3	78.8	78.9
8	00:00:10.000	dB	80.4	84.1	80.7	81.9
9	00:00:10.000	dB	84.4	83.0	83.3	80.0
10	00:00:10.000	dB	80.7	80.0	78.8	77.6
11	00:00:10.000	dB	28.9	20.3	14.5	22.6
12	00:00:10.000	dB	44.8	49.1	45.0	43.7
13	00:00:10.000	dB	46.7	49.5	45.4	43.4
14	00:00:10.000	dB	28.7	22.5	18.2	26.9
15	00:00:10.000	dB	49.5	53.5	46.0	45.0
16	00:00:10.000	dB	43.7	50.3	43.6	42.7
17	00:00:10.000	dB	43.6	49.1	45.5	44.8
18	00:00:10.000	dB	44.5	52.7	47.2	44.6
19	00:00:10.000	dB	46.2	52.0	45.5	44.5
20	00:00:10.000	dB	41.9	46.7	44.3	42.4
21	00:00:10.000	dB	47.5	52.2	44.7	44.0
22	00:00:10.000	dB	49.8	55.0	47.1	46.8

Meas	$Lfeq$ 250 Hz	$Lfeq$ 315 Hz	$Lfeq$ 400 Hz	$Lfeq$ 500 Hz	$Lfeq$ 630 Hz	$Lfeq$ 800 Hz
1	83.3	81.6	79.6	80.5	81.3	79.6
2	80.5	80.0	79.5	79.5	79.1	77.6
3	81.0	77.5	78.3	79.1	78.9	76.5
4	81.5	79.3	77.2	77.3	77.9	77.2
5	81.1	80.7	77.4	78.6	79.1	77.2
6	80.0	80.1	79.1	80.0	78.8	77.2
7	79.7	78.3	78.0	79.7	78.5	76.6
8	79.7	78.8	76.9	76.7	77.4	77.1
9	81.8	79.6	76.9	78.5	77.9	77.2
10	79.0	77.6	77.2	78.3	77.6	76.4
11	9.6	10.2	18.6	6.7	7.8	10.9
12	43.2	42.7	42.1	41.8	42.5	40.2
13	43.3	40.1	39.9	41.0	40.1	40.2
14	12.4	10.7	12.9	15.7	12.7	11.2

15	45.6	42.0	39.6	41.0	40.1	40.4
16	46.1	42.0	42.9	42.0	40.7	39.3
17	43.0	42.9	41.3	42.1	40.8	39.8
18	42.0	40.8	39.4	41.1	40.4	40.1
19	45.9	40.9	39.9	39.5	40.5	40.4
20	42.9	40.6	42.2	41.4	40.7	39.2
21	44.0	43.0	42.0	42.5	41.1	39.6
22	46.0	42.5	41.4	41.5	41.5	41.5

Meas	$Lfeq$ 1.25 kHz	$Lfeq$ 1.6 kHz	$Lfeq$ 2 kHz	$Lfeq$ 2.5 kHz	$Lfeq$ 3.15 kHz	$Lfeq$ 4 kHz
1	75.6	77.4	77.9	77.2	77.4	76.2
2	72.2	73.6	74.9	73.6	74.7	73.8
3	71.3	73.2	73.7	73.4	74.6	73.1
4	73.4	75.2	75.3	73.9	74.0	73.5
5	72.7	75.4	75.8	75.2	75.0	74.2
6	72.2	72.9	73.6	73.6	74.4	73.3
7	70.8	72.1	73.9	73.2	74.2	72.8
8	72.9	75.3	75.5	73.6	74.2	74.4
9	73.7	75.4	75.8	74.8	75.8	75.3
10	70.4	72.4	73.4	73.4	74.2	72.1
11	9.0	6.0	6.6	11.0	11.0	12.4
12	34.5	35.3	35.3	31.9	29.8	25.7
13	35.3	36.4	35.3	31.9	28.7	24.6
14	9.6	8.2	8.4	10.3	12.5	12.2
15	35.5	36.8	35.1	31.1	28.5	24.2
16	33.6	34.5	34.2	30.7	28.9	24.8
17	33.5	34.3	33.5	30.3	28.2	24.6
18	35.6	36.7	34.7	31.0	27.5	23.7
19	35.8	36.7	35.3	31.3	28.2	24.1
20	34.1	34.6	34.3	31.0	28.2	24.3
21	34.4	34.3	34.3	30.7	29.4	26.2
22	35.9	36.6	35.6	31.4	28.8	25.6

B Appendix

Table B.1: The measurement number(meas) and the description defining the setup of what microphone, M, in the selected room as well as which of the two speaker in the source room that are used. The first 10 measurements are recorded in the source room, while the rest of the measurements are done in the receiving room. This description are related to the raw data in Table A.1.

meas	Description
1	M1 - S1
2	M1 - S2
3	M2 - S2
4	M2 - S1
5	M3 - S1
6	M3 - S2
7	M4 - S2
8	M4 - S1
9	M5 - S1
10	M5 - S2
11	Background noise M1
12	M1 - S2
13	M1 - S1
14	Background noise M2
15	M2 - S2
16	M2 - S1
17	M3 - S1
18	M3 - S2
19	M4 - S2
20	M4 - S1
21	M5 - S1
22	M5 - S2

C Appendix

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.table as tbl

###### GLOBAL ############################################


third_octave = [100, 125, 160, 200, 250, 315, 400, 500, 640, 800, 1000, 1250, 1600,
     2000, 2500, 3150, 4000, 5000]

x_ticks_third_octave = [100, 200, 500, 1000, 2000, 5000]#[100, 125, 160, 200, 250,
    315, 400, 500, 640, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000]
x_ticks_third_octave_Bands = ["100", "200", "500", "1000", "2000", "5000"]#['100',
    '125', '160', '200', '250', '315', '400', '500', '640', '800', '1000', '1250',
    '1600', '2000', '2500', '3150', '4000', '5000']

third_octave_entire_range = [50, 63, 80,100, 125, 160, 200, 250, 315, 400, 500,
    640, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000]
x_ticks_entire_range = [63, 125, 250, 500, 1000, 2000, 4000]
x_ticks_entire_range_label = ["63", "125", "250", "500", "1000", "2000", "4000"]



Background = [10, 13]
Source_room_S1 = [0,2,4,6,8]
Source_room_S2 = [1,3,5,7,9]
Receiver_room_S1 = [12,15,17,19,21]
Receiver_room_S2 = [11,14,16,18,20]

#### Font details ########################################
SMALL_SIZE = 16
MEDIUM_SIZE = 18
BIGGER_SIZE = 20

plt.rc('font', size=MEDIUM_SIZE)
plt.rc('axes', titlesize=BIGGER_SIZE)
plt.rc('axes', labelsize=MEDIUM_SIZE)
plt.rc('xtick', labelsize=MEDIUM_SIZE)
plt.rc('ytick', labelsize=SMALL_SIZE)
plt.rc('legend', fontsize=SMALL_SIZE)
plt.rc('figure', titlesize=BIGGER_SIZE)
#########################################################

room_size_source = [8.501, 6.042, 5.174]
room_size_receiver = [4.11, 5.58, 4.58]

######## Define what to plot ############################



#########################################################

def _calculate_unfavorable(array):
    steps = [3,3,3,3,3,3,1,1,1,1,1,0,0,0,0,0,0]
    start = 33
```

```
    ref_curve = [33]
    for i in range(len(steps)):
        start += steps[i]
        ref_curve.append(start)


    unfav_dist = 100
    ones = [1]*len(ref_curve)
    cnt = 0

    while(unfav_dist > 32):
        unfav = []
        for i in range(len(array)):
            if ref_curve[i] > array[i]: unfav.append(round(ref_curve[i] - array[i],1))
            else: unfav.append(0)
        new_unfav = sum(unfav)
        if new_unfav < unfav_dist: unfav_dist =  new_unfav

        if unfav_dist > 32 :
            for i, val in enumerate(ref_curve):
                ref_curve[i] = val - 1
            cnt += 1

    print("R --- Reference curve --- Unfavorable Deviation")
    for i in range(len(array)): print("{0} \t {1} \t {2}".format(array[i], ref_curve[i], unfav[i]))
    print("################################################")
    print(unfav_dist, "\t", cnt)
    return unfav, cnt, ref_curve

def _split_array(array):
    """

    :param array:
    :return: R_S1, R_S2, S_S1, S_S2, b
    """
    R_S1 = []
    R_S2 = []
    S_S1 = []
    S_S2 = []
    b = []
    for i in Source_room_S1 : S_S1.append(array[i])
    for i in Source_room_S2 : S_S2.append(array[i])
    for i in Receiver_room_S1 : R_S1.append(array[i])
    for i in Receiver_room_S2: R_S2.append(array[i])
    for i in Background : b.append(array[i])
    return np.array(R_S1), np.array(R_S2), np.array(S_S1), np.array(S_S2), np.array(b)

def _LeqArray_Lab4(file):
    df = pd.read_csv(file, sep=";")
    array = df.to_numpy()
    Data_Array = array[:,28:46].astype(np.float)
    return Data_Array

def _calculate_log_mean(lst):
    avg = 0
    for i in lst:
        avg += 10**(i / 10)
    return round(10*np.log10(avg / len(lst)),1)

def _create_L_together(array):
    lst = np.transpose(array)
    temp = []
    for i in lst : temp.append(_calculate_log_mean(i))
    return np.array(temp)
```

```python

def _create_b(b):
    b = np.transpose(np.array(b))
    temp = []
    for i in b: temp.append(_calculate_log_mean(i))
    b = np.array(temp)
    #LpiBLP = _calculate_log_mean(noise)
    #print("########",b)
    return b

def _calculate_SPL(array):
    temp = 0
    for i in array:
        temp = temp + 10**(0.1*i)
    return round((10*np.log10(temp)),1)

def _two_into_one_array(arr1, arr2):
    arr3 = np.column_stack((arr1, arr2))
    temp = []
    for i in arr3: temp.append(_calculate_log_mean(i))
    return np.array(temp)

def _plot_Semilogx(R_S1, R_S2, S_S1, S_S2, background, title1, title2, S1andS2 = False):
    fig = plt.figure(figsize=(17,7))

    ax1 = fig.add_subplot(121)
    ax2 = fig.add_subplot(122)

    receiver = _two_into_one_array(R_S1, R_S2)
    source = _two_into_one_array(S_S1, S_S2)

    back = np.array(background)


    if S1andS2:
        ax1.step(third_octave, R_S1, where="mid", color="dimgray", label="$L_{sb}$ for S1")
        ax1.step(third_octave, R_S1 - back,where="mid", color="blue", label="$L_{sb}$ - $L_b$  for S1")
        ax1.axline((100, 10), (5000, 10), linestyle="--", linewidth=0.8, color="r", label="Seperation line for correction")
        ax1.step(third_octave, R_S2,where="mid", color="goldenrod", label="$L_{sb}$ for S2")
        ax1.step(third_octave, R_S2 - back,where="mid", color="blueviolet", label="$L_{sb}$ - $L_b$  for S2")

        ax2.step(third_octave, S_S1,where="mid", color="dimgray", label="$L_{sb}$ for S1")
        ax2.step(third_octave, S_S1 - back,where="mid", color="blue", label="$L_{sb}$ - $L_b$  for S1")
        ax2.step(third_octave, S_S2,where="mid", color="goldenrod", label="$L_{sb}$ for S2")
        ax2.step(third_octave, S_S2 - back,where="mid", color="blueviolet", label="$L_{sb}$ - $L_b$  for S2")
        ax2.axline((100, 10), (5000, 10), linestyle="--", linewidth=0.8, color="r", label="Seperation line for correction")

    else:
        ax1.step(third_octave, receiver,where="mid", color="dimgray", label="$L_{sb}$")
        ax1.step(third_octave, receiver - back,where="mid", color="goldenrod", linewidth=2, label="$L_{sb}$ - $L_b$")
        ax1.axline((100, 10), (5000, 10), linestyle="--", linewidth=0.8, color="r", label="Seperation line for correction")

        ax2.step(third_octave, source,where="mid", color="dimgray", label="$L_{sb}$
```

```
")
        ax2.step(third_octave, source - back,where="mid", color="goldenrod", linewidth=2, label="$L_{sb}$ - $L_b$")
        ax2.axline((100, 10), (5000, 10), linestyle="--", linewidth=0.8, color="r", label="Seperation line for correction")

        ax1.step(third_octave, back, where="mid", color="forestgreen", label="$L_b$")
        ax2.step(third_octave, back, where="mid", color="forestgreen", label="$L_b$")

    ax1.set_xscale('log')
    ax1.grid(which="major")
    ax1.grid(which="minor", linestyle=":")
    ax1.set_xlabel("Frequency [Hz]")
    ax1.set_ylabel("Amplitude [dB]")
    ax1.set_xticks(x_ticks_third_octave)
    ax1.set_xticklabels(x_ticks_third_octave_Bands)
    ax1.set_title(title1)

    ax2.set_xscale('log')
    ax2.grid(which="major")
    ax2.grid(which="minor", linestyle=":")
    ax2.set_xlabel("Frequency [Hz]")
    ax2.set_ylabel("Amplitude [dB]")
    ax2.set_xticks(x_ticks_third_octave)
    ax2.set_xticklabels(x_ticks_third_octave_Bands)
    ax2.set_title(title2)

    ax1.legend()
    ax2.legend()
    fig.savefig("Lab4Pressureplot.png")
    plt.show()

def _plot_DnT_ref(DnT, Dn, ref_curve, R_prime, tab, unfav_dist_sum):
    fig = plt.figure(figsize=(20, 8.5))
    fig.tight_layout()
    ax1 = fig.add_subplot(121)
    ax2 = fig.add_subplot(122)

    Header = ["Frequency [Hz]", "$R'$ [dB]", "Shifted Reference\nCurve [dB]", "Unfavorable \nDeviation [dB]"]
    Header = ["$\\bf{Frequency [Hz]}$","$\\bf{Unfavorable}$ \n $\\bf{Deviation [dB]}$" ]
    table = tbl.table(ax1, cellText=tab, colLabels=Header, cellLoc="center",loc="center")
    table.auto_set_font_size(False)
    table.set_fontsize(13)
    #table = ax1.add_table(cellText=table, cellLoc="center", loc="center",colLabels=Header,)
    ax1.add_table(table)


    table.scale(0.7,2.5)
    ax1.set_axis_off()

    ax2.step(third_octave, DnT,where="mid", color="dimgray", label="$D_{nT}$")
    ax2.step(third_octave, Dn,where="mid", color="black", label="$D_{n}$")
    ax2.step(third_octave, ref_curve,where="mid",linestyle="--", color="forestgreen", label="Shifted Reference Curve")
    #ax.axline((100,0), (100, 80), linestyle="..", linewidth=0.8, color="black", label="Frequency range according to the lab")
    #ax.axline((5000, 0), (5000, 80), linestyle="..", linewidth=0.8, color="black")
    ax2.step(third_octave, R_prime,where="mid", color="goldenrod", label="$R_{prime}$")

    ax2.set_xscale('log')
```

```
    ax2.grid(which="major")
    ax2.grid(which="minor", linestyle=":")
    ax2.set_xlabel("Frequency [Hz]")
    ax2.set_ylabel("Amplitude [dB]")
    ax2.set_xticks(x_ticks_third_octave)
    ax2.set_xticklabels(x_ticks_third_octave_Bands)

    title = str("$D_{nT}$, $D_n$, $R‘$ and the Shifted Reference Curve. \n") + str("Sum of Unfavorable Deviation: {0}dB \n Weighted Sound Reduction Index: {1}dB".format(round(unfav_dist_sum,1),ref_curve[7]))

    #title = str("$D_{nT}$, $R‘$ and the Shifted Reference Curve \n Sum of Unfavorable Deviation: ", str(unfav_dist_sum) ,str("dB \n Weighted Sound Reduction Index: ", str(ref_curve[7]),"dB"))

    ax2.set_title(title)

    ax2.legend()
    fig.savefig("Lab4DnT.png")
    plt.show()

def _calculate_D(receiver, source):
    return source - receiver

def _calculate_Dn(D, A):
    temp = []
    A0 = 10
    for i in range(len(D)) : temp.append(D[i] - 10*np.log10(A[i] / A0))
    return np.array(temp)

def _calculate_Dnt(D, T):
    temp = []
    T0 = 0.5
    for i in range(len(D)) : temp.append(D[i] + 10*np.log10(T[i] / T0))
    return np.array(temp)

def _surface_seperate():
    return round(1.18 * 1.21,2)

def _calculate_R_prime(D, A):
    temp = []
    S = _surface_seperate()
    for i in range(len(D)): temp.append(D[i] + 10*np.log10(S / A[i]))
    return np.array(temp)

def _calculate_T(file):
    df = pd.read_csv(file, sep=";")
    array = df.to_numpy()
    data = array[4:22, 4].astype(np.float)
    for i in range(len(data)) : data[i] = data[i]*2
    return data

def _calculate_A(T, V):
    return 0.16 * (V / T)

def _calculate_V(vol_array):
    x = vol_array[0]
    y = vol_array[1]
    z = vol_array[2]
    return round(x * y * z, 2)

def _create_table_data(R, ref_curve, unfav_dist):
    #temp = [["Frequency [Hz]","R‘ [dB]","Shifted Reference Curve [dB]","Unfavorable Deviation [dB]"]]
    temp = []
    for i in range(len(R)):
        #temp1 = [third_octave[i],round(R[i],1),round(ref_curve[i],1),round(
```

```
    unfav_dist[i],1)]
        temp1 = [third_octave[i], round(unfav_dist[i],1)]
        temp.append(temp1)
    return temp

def _create_R0():
    w = []
    R0 = []
    rhos = (15.33 + 10.08) / 2
    rho  = 1.225 * 343.2
    for i in third_octave : w.append(i*2*np.pi)
    for i in w : R0.append(10*np.log10(1 + ((i * rhos) / (2 * rho))**2 ))

    return R0


def _create_R_random(R0):
    R_random = []
    for i in R0 : R_random.append(i - 10*np.log10(0.23 * i))
    return R_random

def _create_R_field(R0):
    R_field = []
    for i in R0 : R_field.append(i - 5)
    return R_field


def _plot_R_R_field_R_random(R_prime, R_field, R_random):
    fig, ax = plt.subplots(figsize=(8,7))

    ax.step(third_octave, R_prime,where="mid", color="dimgray", label="$R_{prime}$"
    )
    ax.step(third_octave, R_field,where="mid", color="forestgreen", label="$R_{
    field}$")
    ax.step(third_octave, R_random,where="mid", color="goldenrod", label="$R_{
    random}$")
    ax.set_xscale('log')
    ax.grid(which="major")
    ax.grid(which="minor", linestyle=":")
    ax.set_xlabel("Frequency [Hz]")
    ax.set_ylabel("Amplitude [dB]")
    ax.set_xticks(x_ticks_third_octave)
    ax.set_xticklabels(x_ticks_third_octave_Bands)

    title = "$R_{prime}$, $R_{field}$ and $R_{random}$."
    ax.set_title(title)

    ax.legend()
    fig.savefig("LabRs.png")
    plt.show()

def _calculate_Corrected(arr, b):
    temp = []
    for i in range(len(arr)) : temp.append(10*np.log10(10**(0.1 * arr[i]) -
    10**(0.1 * b[i])))
    return temp

############# CODE RUNS FROM HERE ###############################

array = _LeqArray_Lab4("lab4.csv")

R_S1, R_S2, S_S1, S_S2, b = _split_array(array)

###################################
R_S1_avg = _create_L_together(R_S1)
R_S2_avg = _create_L_together(R_S2)
S_S1_avg = _create_L_together(S_S1)
```

```python
S_S2_avg = _create_L_together(S_S2)
##################################

b = _create_b(b)
R_S1 = _create_L_together(R_S1)

title1 = str("Receiving room")
title2 = str("Source room")

_plot_Semilogx(R_S1_avg, R_S2_avg, S_S1_avg, S_S2_avg ,b, title1, title2)

print(_calculate_SPL(R_S1_avg))
print(_calculate_SPL(R_S2_avg))

receiver = _two_into_one_array(R_S1_avg, R_S2_avg)
source = _two_into_one_array(S_S1_avg, S_S2_avg)

source = _calculate_Corrected(source, b)

####### Values #####################
T = _calculate_T("reverb_time.csv")
V = _calculate_V(room_size_receiver)
A = _calculate_A(T, V)
D = _calculate_D(receiver,source)
R_prime = _calculate_R_prime(D,A)
DnT = _calculate_Dnt(D,T)
Dn = _calculate_Dn(D, A)
####################################

unfav_dist, cnt, ref_curve = _calculate_unfavorable(R_prime)

table = _create_table_data(R_prime,ref_curve,unfav_dist)

_plot_DnT_ref(DnT, Dn, ref_curve, R_prime, table, sum(unfav_dist))

R0 = _create_R0()
R_random = _create_R_random(R0)
R_field = _create_R_field(R0)

_plot_R_R_field_R_random(R_prime,R_field, R_random)
```

Listing C.1: Python code for the post processing