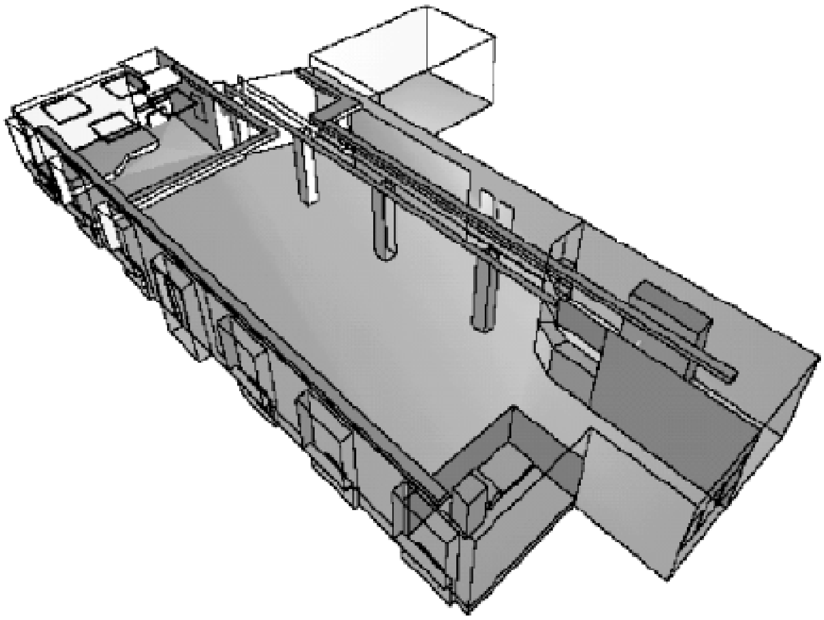
CHALMERS



Acoustical Improvements in Kulturhuset Oceanen Using Slitted Low Frequency Absorbers

Master's Thesis in the Master of Science Programme in Electrical Engineering

MATTIAS DAVIDSSON

Department of Civil and Environmental Engineering
Division of Applied Acoustics
Room Acoustics Group
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2006

Master's Thesis 2006:18

Acoustical Improvements in Kulturhuset Oceanen Using Slitted Low
Frequency Absorbers

Master's Thesis 2006:18

Department of Civil and Environmental Engineering
Division of Applied Acoustics
Room Acoustics Group
Chalmers University of Technology
SE-41296 Göteborg
Sweden

Tel. +46-(0)31 772 1000

Cover:
Computer model of the concert hall of Kulturhuset Oceanen, the way it appears in the simulation software CATT-Acoustic.

Reproservice / Department of Civil and Environmental Engineering
Göteborg, Sweden 2006

Acoustical Improvements in Kulturhuset Oceanen Using Slitted Low Frequency Absorbers
Mattias Davidsson
Department of Civil and Environmental Engineering
Division of Applied Acoustics
Room Acoustics Group
Chalmers University of Technology

Abstract

The task was to improve the acoustic properties in the concert hall at Kulturhuset Oceanen. Measurements showed that the main problem was related to resonance frequencies around 70 and 110 Hz. There was also a peak at 40 Hz, but since the P.A. system does not excite such low frequencies, it was left without any consideration. The other two strong peaks were treated with slitted Helmholtz absorbers. The actual project did not include mounting of the full-scale absorbers; instead a prototype was constructed and built for experimentation and measurements in a reverberation chamber at Chalmers. The measurements showed that the absorber worked very well within the targeted frequency range, that is around 70 to 110 Hz. Since the absorber is mainly consisting of wood, it will also have higher absorption coefficients in the high frequency region, compared to the concrete walls that now are exposed in the concert hall. This implies that there will be a reduction of the reverberation times in the upper frequencies as well. Measurements on the prototype, together with simulations in CATT-Acoustic show a possible reduction of the reverberation time at the 125 Hz octave band with about 0.2 – 0.3 seconds.

Key words: resonance frequencies**,** slitted Helmholtz absorbers, octave bands, absorption coefficient, CATT-Acoustic, Schroeder backward integration, room acoustics, least squares method, MLSSA

Acknowledgements

Special thanks to the Division of Applied Acoustics at Chalmers University of Technology for backing up and funding this project, Georgios Natsiopoulos for excellent supervision and guidance, Börje Wijk for help with various computer issues, Wolfgang Kropp for authorization and allowance, Bengt-Inge Dalenbäck for the outstanding CATT-Acoustic and Mats Öst, chairman of Oceanen, for the opportunity of making the improvements possible.

Contents

1 Background and Project Development..1

2 Measurements..4

2.1 Initial Measurements...4
2.2 Testing the Prototype...6

3 Data Analysis...8

3.1 Fast Fourier Transform 8
3.2 Reverberation Times..9
3.3 Computer Model..9
3.4 Schroeder Backward Integration...11

4 Absorber Design...13

4.1 Calculating the Coefficient...14
4.2 Choosing the Damping Material...15

5 Results...17

5.1 Initial Measurements..17
5.2 Absorber Design Parameters..18
5.3 Absorber Measurements...19
5.4 Theory vs. Reality 19
5.5 Simulation Results...20

6 Conclusions and Discussion...21

Appendix A...22

Appendix B...29

Bibliography...52

1 Background and Project Development

Kulturföreningen Oceanen is a cultural association that has existed since 1996 and which managed to acquire a real estate located at Stigbergstorget 8 in Göteborg. The building was previously the Oceanographic Institute, thereby the name. One curiosity about this house, is the deep sea tank located in the centre of the building stretching from the basement to the attic. Unfortunately, they could never make use of it, since the concrete support of the tank was insufficient to hold the pressure from the water.

The object of interest in this house is the concert hall, where live musical performances are held. The hall's dimensions are 21.0 x 8.00 x 2.72 m and it is almost rectangular except for a wardrobe, a bar and a backstage room with an angled wall. These irregularities make it difficult to use simplifying formulas for predicting room resonances.

Figure 1.1: *Exterior and interior view of Kulturhuset Oceanen, Göteborg.*

Measurements were executed to examine the acoustical properties of the room. The equipment consisted of a portable computer with the software MLSSA installed and an omni-directional microphone with a microphone amplifier. It was considered to be appropriate to use the permanently installed P.A. system for the stimuli sound, since the acoustics should be adjusted to suit the sound that is emitted through it and it is not likely that the system will be exchanged in the immediate future.

Several measurements were done in various microphone positions in the hall, in order to get the MLSSA to calculate the different room impulse responses (RIR:s). The computer transmitted the unique repetitive stimulus via the locally installed P.A. system and simultaneously recorded the responses measured by the microphone.

The most important microphone positions were of course the ones directly on the dance floor in front of the stage, since that was where it was crucial for the sound quality to be optimised.

Other positions were also examined, but were later discarded, partly due to lack of time, but also because of practical and economical reasons.

At the Division of Applied Acoustics at Chalmers University of Technology, a model of the concert hall was constructed in the computer program CATT-Acoustic v8.0b (see figure 2.1), where the estimated absorption coefficients were inserted. The model was then adjusted to fit the calculated reverberation times from the actual measured values in MLSSA. The modelled reverberation times were obtained through simulations in CATT through a special mapping technique, where the entire room was mapped in the height of the microphone positions. It was then very easy to compare the model with the measurements. In order to adjust the modelled reverberation times, small increments or decrements, respectively, were made (by logic estimation) in suitable absorption coefficients. Suitable in this case means in the right frequency octave band and the right material (that covers a sufficiently large area to effect the reverberation time) at the right coordinates.

When all adjustments in CATT were finished and the calculated reverberation times from MLSSA were nearly identical to the ones in the model, possibilities to improve the acoustical condition of the hall were examined. However, because of fundamental limitations in the underlying theory of CATT-Acoustic, the validity of the model is questionable for frequencies as low as 125 Hz. For instance, the ray tracing method that the program uses does not take interference of the sound waves in account. This interference becomes more significant in the lower frequency region, while it plays a lesser role in the upper.

The idea behind the low frequency absorber, was that it should take care of both the problem peaks at 70 and 110 Hz. That means that their centre frequency will turn out between those peaks and that is where the absorbers will have maximum absorption. From a logarithmic point of view, the absorption curve will decrease symmetrically on both sides. This also implies that the centre frequency will end up at about 88 Hz. The philosophy behind the project was simply to give birth to an absorber that had an arbitrarily high absorption coefficient at its centre frequency, but as high as possible at their points of interest (70 and 110 Hz), preferably as high as 0.8.

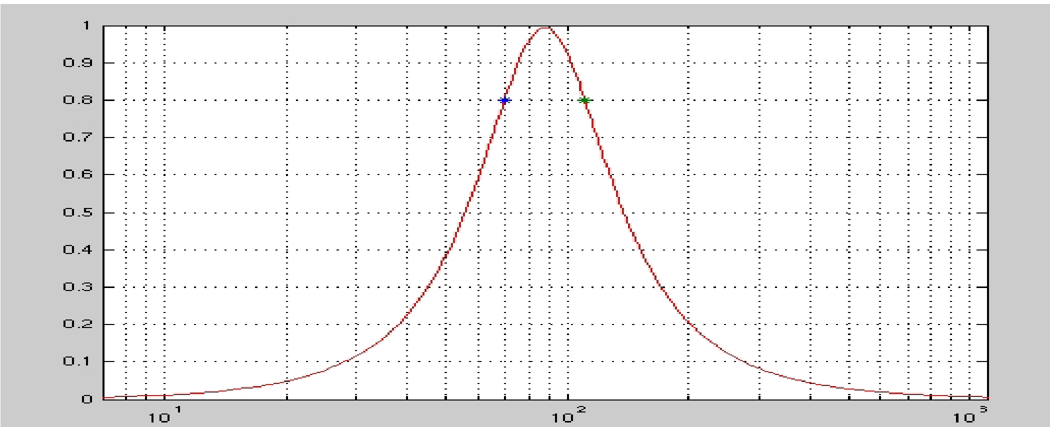


Figure 1.2: *The theoretical absorption coefficient of a slitted low frequency absorber with the two points at 70 and 110 Hz inserted.*

The next step was to examine which surfaces in the hall that could be covered with absorbers and how much the reverberation times would be affected. The outcome of the simulations would be the foundation of a design proposition that the board of Oceanen would use to make the final decisions concerning the project. (However, the task for this master's thesis would be completed when the final measurements on a prototype had been analysed and therefore independent of Oceanen´s decision.)

The next challenge faced, was how to construct a slitted low frequency absorber with the desired specifications. It had some aesthetical demands on it, as well as functional. In the design proposition, it was assumed that the theoretical absorption coefficient at 70 and 110 Hz would exceed 60% and therefore would a total covering of 69 square meters suffice for satisfying decrease of the reverberation times. In this case, that means a reduction of nearly a half second at the most and the times would then be lower than one second in all octave bands from 125 to 4000 Hz. The absorbers would preferably be mounted in the ceiling.

Since the ceiling height in the hall measured 2.72 meters, any reduction of it did not appeal to the board of Oceanen, and a second approach to treat the acoustics needed to be considered.

The absorber was eventually designed using Matlab. A program was written that calculated the unknown dimensions of the absorber, such as the slit width, thickness of the damping material and the absorption coefficient as a function of frequency. The program adjusted the absorbers according to the locked design parameters, which were the depth, panel width, density of the damping material and desired Q-value (i.e. the real input parameters in this case were the peak frequencies and desired absorption coefficients at those frequencies).

It turned out that it would be possible to design an absorber with higher absorption around the frequency range of interest than first assumed, partly due to the ability to make it deeper when it is mounted on the walls instead of the ceiling. A smaller Q-value also seemed a lot easier to obtain than what was previously assumed. After the concluding calculations on this new design, the depth could be established to 40 cm and the resulting absorption coefficient around 70 and 110 Hz could theoretically reach almost 90%. This meant that only about 45 square meters needed to be covered to yield the same attenuation as in the case with the absorbers mounted in the ceiling.

The new proposition was approved by Oceanen and the project was basically good to go. However, there was one more task to perform before the real construction. A prototype was to be built and tested in a reverberant laboratory at Chalmers, so that the theoretical model could be verified.

The slitted low frequency Helmholtz absorber is mainly made of wood, except for the damping material inside. The prototype was built inside the laboratory, with the necessary parts delivered to the door. After completion, the relevant measurements took place in order to reveal any deviations from the theoretical values.

The main purpose of the prototype experimentation was twofold:

(1) to conclude the project within a reasonable time frame without having to wait for the real absorbers to be mounted, and
(2) to adjust the CATT-model to make it suit the real world to a higher extent and thereby present an even more precise prediction of the result for the chair board of Oceanen.

2 Measurements

2.1 Initial Measurements

Due to a number of advantages, it was decided that the software MLSSA would be suitable for the measurements at Oceanen. The room impulse responses (RIR:s) are easily calculated as well as the reverberation times. MLSSA transmits a specially designed pseudorandom stimulus, maximum-length sequence (MLS) that strongly reminds of white noise (in fact, it has the same spectrum), but with the difference of appearing as an audible repeating pattern. The MLS has a number of advantages over commonly used test signals, such as low crest factor, it is exactly repeatable, it is periodic and permits efficient computation of long impulse responses. [1]

The equipment used in the measurement set-up also consisted of an omni-directional Larson – Davis microphone, model 2520, connected to a microphone amplifier, Larson – Davis 2200C, which in turn was directly connected to the portable computer, Toshiba T5200. The computer was equipped with an appropriate sound card and the MLSSA software. The output stimulus was transmitted to the permanently installed P.A. system at Oceanen. This was motivated by the fact that the acoustics should be optimised to fit this particular sound system, especially since it is not likely that the system will be exchanged for at least another ten years.

The complete P.A. system at Oceanen consists of an Allen – Heath GL3 24 channel mixer console, a Peavey VSX active crossover, adjusted to a crossover frequency of 250 Hz, a Peavey PV1200 power amplifier for the high frequencies, producing 2 x 600 Watts at 8 Ω and a LAB SS1000B power amplifier for the low frequencies, producing 2 x 500 Watts at 8 Ω. The subwoofers are two Yamaha S250B cabinets, each equipped with an 18 inch radiator of 600 Watts RMS maximum power. The top cabinets are two Krafft Hiller, each containing a 12 inch speaker with a maximum power handling of 300 Watts RMS.

The measurements were carried out in ten various microphone positions with the microphone mounted vertically on a microphone stand at a height of 1.25 meters above the floor. This might seem to be an odd height, but as the audience usually is both sitting and standing, this height is a compromise between the two situations. This simplified the measurements and kept down the amount of data in comparison to two different height set-ups. The microphone positions, shown in figure 2.1, were evenly spread out on the dance floor and at some auxiliary spots on the stage, near the mixer console, at the entrance, at the side of the dance floor and behind the bar. These complimentary positions were merely used for additional control measurements and were not included later in the analysis. The reason for this was to reduce the processing time, in benefit for the five most important spots at the dance floor, where the main part of the audience will be located during musical performances.

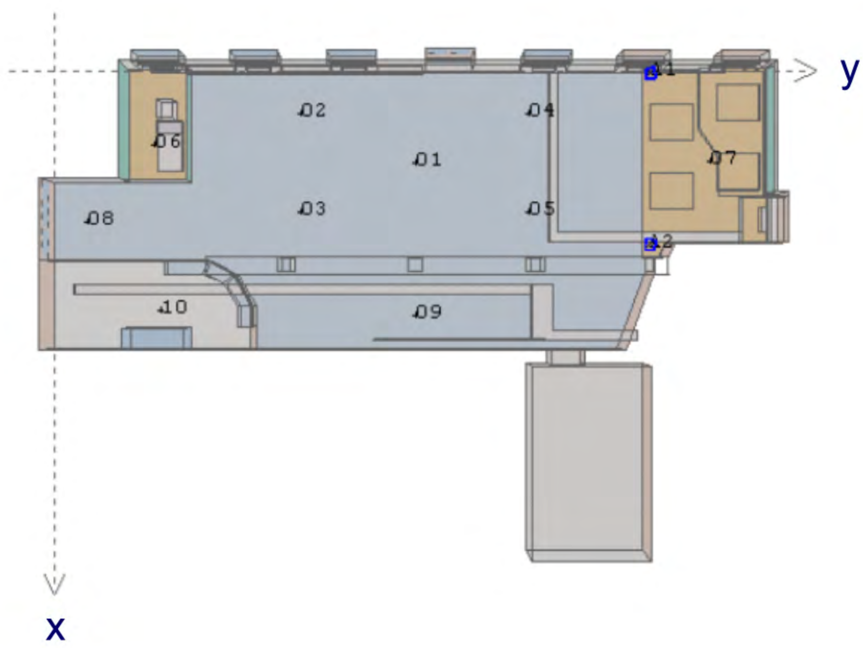


Figure 2.1: *The figure is taken from CATT-Acoustic and shows the computer model of Oceanen's concert hall. The microphone positions are marked with points and numbers, with the five most important on the dance floor. The stage is to the far right with microphone position 07.*

When standing in the middle of the dance floor facing the stage, one has a wall with six windows and one emergency escape on the left. The two windows closest to the stage are sealed and covered with fibreboard. The windows can be covered with foldable mirrors that stretches from the floor to the ceiling. This is usually done during dance classes. Furthermore, these mirrors can be covered with heavy drapes. Or, if preferred, the mirrors can be folded in and the windows covered by the curtains. This contributes to some various acoustical conditions. Due to this, measurements were carried out in two different manners. First, with the curtains opened and the mirrors folded in, and secondly, with the drapes covering the unfolded mirrors. The latter condition is the usual state of the mirrors and curtains during concerts and it is presumed that they will remain in that state for optimal acoustical conditions.

Under the premises mentioned above, the individual measurements summed up to a total of twenty; one at each of the ten microphone positions with the curtains covering the left wall and one at each positions with the windows exposed.

Before the measurements started, four parameters had to be adjusted in the software. Those parameters were:

(1) the input signal sensitivity, which also was adjusted on the microphone amplifier,
(2) the number of measurements of which to calculate an average from to get a more reliable result, which was set to four,
(3) the sampling frequency, which was set to 40 kHz to produce sufficient bandwidth, and
(4) the number of sample points, which determines the frequency resolution, given a fixed sampling frequency. In this case it was set to a length of 65535 samples, which gives a resolution Δf of 1.22 Hz. This is calculated through equation (2.1):

$$\Delta f = \frac{1}{T} = \frac{f_s}{2^{N-1}} \tag{2.1}$$

where T is the period, f_s is the sampling frequency and N is the order of the MLS sequence (which is 16 with 65535 points). [2]

The output signal was directly connected to the mixer console of the local P.A. system and was left with no equalization filtering whatsoever and with adequate sound level (to get sufficient resolution in the readings on the computer). The data from the sessions was then stored in the computer's hard drive.

2.2 Testing the Prototype

After the initial simulations in CATT-Acoustic, as well as structuring the design program that would deliver the proper dimensions of the low frequency absorber modules, time had come for the realisation of a manageable prototype. A wooden absorber with a surface of three square meters was easily built directly inside a reverberation chamber at Chalmers. The proceeding measurement session on the prototype inside the chamber was almost into detail a repeated process of the measurements at Oceanen.

First, the RIR:s of the chamber without the absorber were measured in four different microphone positions. The settings in the MLSSA were identical to the ones in the initial measurements at Oceanen. The stimulus was emitted via a NAD 3020 amplifier and a dual speaker setting with a top speaker and a subwoofer. All the measurements were done both through the top speaker and the subwoofer, one at the time, to make sure that the whole audible spectrum was fully covered.

After the measurements in the empty chamber were finished, the absorber was placed in the chamber at a certain calculated position. The absorber was tuned to have its maximum absorption coefficient at 88 Hz. It would therefore be especially beneficial to place the prototype where the dominant room modes would have maximum particle velocity. Results for an idealised room is readily calculated (see equation (3.1)). Since the room was not an empty, perfectly rectangular room with perfectly reflecting walls, the absorber was tested in three different orientation angles (see figure 2.2). That was a precaution, in case that some important aspect of the sound field in the room would have been missed in the calculations.

For simplicity, the microphone positions were the same as in the case with the empty chamber. All in all, the total amount of measurements summed up to 32 individual tests - two times four in the first case, two times four microphone positions times three absorber positions in the latter case.

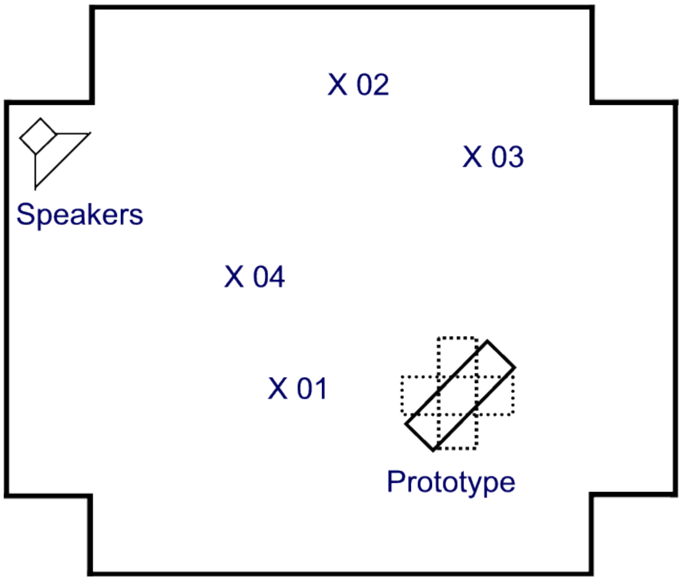


Figure 2.2: *Schematic drawing of the measurements on the absorber prototype in three various angles and four randomly chosen microphone positions (marked with X´s).*

All the measurement data were collected, analysed and processed to yield reverberation times. These were in turn used to calculate the absorption coefficients for the prototype in six octave bands (125 Hz through 4 kHz).

3 Data Analysis

As mentioned in the previous chapter, the reason for choosing 65535 sample points with MLS order of 16 in the MLSSA, is to obtain a suitable resolution (calculated with equation (2.1)). To get the appropriate bandwidth, the sample frequency was set to 40 kHz. Though it is only necessary to examine frequencies up to 4 kHz, a higher bandwidth is needed because of the filters MLSSA uses. In this case it would be interesting to measure impulse responses within the octave bands 125 through 4000 Hz, since the model in CATT-Acoustic hopefully would be valid down to 125 Hz. Frequencies below 125 Hz are included in the analysis by examining the fast Fourier transforms of the impulse responses from the initial measurements.

Intuitively, the main problems with the room acoustics at Oceanen felt to be located in the lower frequency region. Frequencies above 4 kHz were not considered as problematic and therefore excluded in favour for lesser processing times in CATT-Acoustic. But, like the lower frequencies, the data contained higher frequencies which could be examined with Fourier analysis in order to get a complete overall picture.

3.1 Fast Fourier Transform

The first simple analysis made on all the room impulse responses was the Fast Fourier Transform (FFT), which enabled a total view of the frequency response situation of the concert hall. It was easy to isolate the distinct problems in the lower part of the frequency spectra. These peaks in the responses from the measurements were recorded in a table in order to distinguish the specific problem frequencies – resonance frequencies dependent of the hall geometry and the P.A. system. The table turned out to reveal three major peaks located at 40, 70 and 110 Hz which are the 221, 340 and 043 modes according to equation (3.1). The coordinate system is oriented according to figure 2.1 (the z-axis running vertically, non-visible).

$$f_{n_x n_y n_z} = \frac{c}{2}\left[\left(\frac{n_x}{L_x}\right)^2 + \left(\frac{n_y}{L_y}\right)^2 + \left(\frac{n_z}{L_z}\right)^2\right]^{1/2} \tag{3.1}$$

c is the speed of sound (340 m/s), *n* is an integer which gives the actual modes, *L* is the length of each dimension. [3]

It was easy to conclude that the lowest peak at 40 Hz could be left alone without any consideration since it is not likely that such low frequencies are excited by the P.A. system. In fact, special cut-off filters are installed to prevent any frequencies at 40 Hz and below to leave the speakers. So all attention were directed towards the peaks at 70 and 110 Hz.

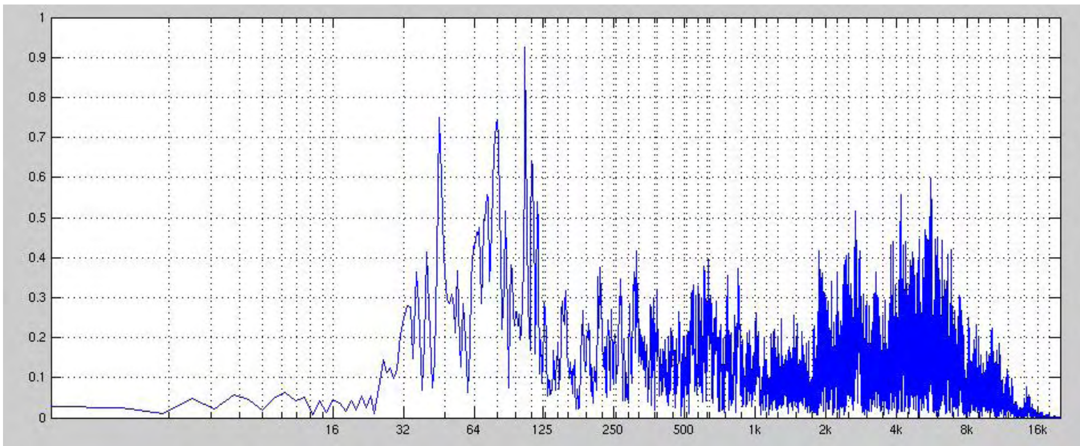


Figure 3.1: *Example of a FFT of the room impulse response from microphone position 1 at the middle of the dance floor with the drapes fully covering the left wall. The sharp peaks at 40, 70 and 110 Hz are clearly visible.*

The frequency response at the upper domain appeared to be less problematic, especially with the curtains covering the unfolded mirrors. Higher frequencies are also less complicated to damp with ordinary high frequency absorbers, such as mineral wool.

3.2 Reverberation Times

In order to adjust the various absorption coefficients in the CATT–model, it would be necessary to study the reverberation times in the octave bands between 125 through 4000 Hz. These reverberation times are automatically calculated by MLSSA. The idea was to study and record all the surfaces in the concert hall, look up the absorption coefficients in different tables and insert these into the computer model. If the simulated reverberation times in CATT-Acoustic would differ from the ones calculated by MLSSA, the absorption coefficients in CATT would be slightly adjusted to better fit the data from MLSSA.

3.3 Computer Model

CATT-Acoustic is a simulation software where it is possible to make predictions of acoustical behaviour in geometrical spaces with given absorbing surfaces. CATT uses different variants of ray tracing as prediction methods and thus it can not predict modal patterns. Moreover, scattering and diffraction are not handled exactly. Therefore it has its limitations at both low and high frequencies.

The room or hall that one wishes to study is built in the software by giving the corner points of the surfaces and defining the plane within the corner points. After constructing a detailed three dimensional image of the enclosure under study, it is necessary to give the absorption coefficients for all the surfaces and octave bands between 125 Hz and 4 kHz (it is possible to use octave bands up to 16 kHz if one wishes, but in this case 4 kHz was sufficient). Fortunately, there are plenty of tables to get input data from. To measure absorption coefficients on multiple surfaces in all octave bands would be a too complicated process and take far too much time to be justified in a project like this (and unnecessary too, since there are useful tables). Like said earlier, these values would only need a slight adjustment to fit the measurement data and then one has a quite realistic model that can give satisfying predictions.

To further refine the model, the speakers' directivities should be simulated. This can be hard to do exactly, especially when accurate specifications are not known. However, it is possible to make own measurements in an anechoic chamber or one can make approximations that the speakers directivity follows simple models. In this work, the latter was chosen with good results. No technical descriptions of the speakers were found and they were securely bolted to the walls at the location. These things, plus the fact that a thorough measurement is a time consuming process, justified the choice.

The speaker system at Oceanen consists of two high frequency speakers and two subwoofers. The crossover frequency is set to 250 Hz, which means that the directivity of the subwoofers practically are omni-directional. The high frequency speakers on the other hand, is assumed to act more and more like a piston in a circular tube, the higher in frequency one goes. [4] This assumption is based on the physical appearance of the speakers, which basically have minimized baffles.

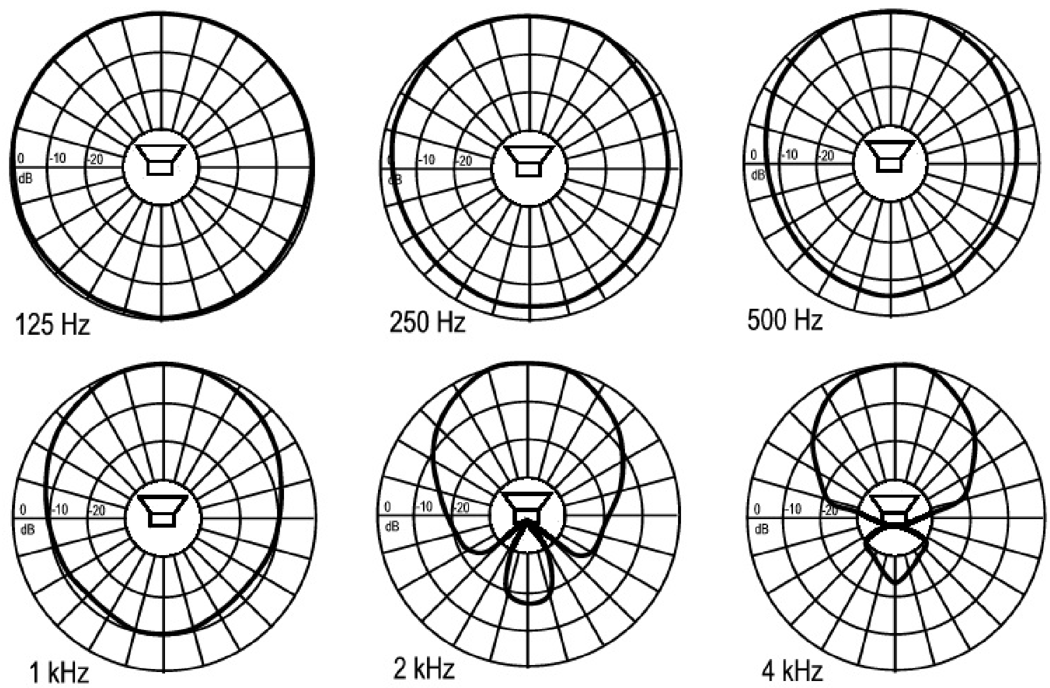


Figure 3.2: *The simulated speaker directivities for the six octave bands. Each circle represents a 10 dB attenuation. Since the directivity is symmetrical, only one view is necessary.*

With all the parameters set in CATT-Acoustic, the most convenient way to proceed was to map the entire dance floor at a height of 1.25 meters; the same height as the microphone had during the measurements. By using this particular method it was possible to see the effect of a pair of speakers simultaneously. The mapping procedure revealed the different reverberation times at the chosen octave bands. Since the crossover cuts all frequencies below 250 Hz that is transmitted through the top speakers, only the octave bands above 250 Hz can be simulated at the same time. The octave band below 250 Hz, that is 125 Hz, was simulated after switching to the simulated subwoofers which had different directivity and location than the top speakers. The model was adjusted by this means as mentioned above, until the reverberation times seemed as close to the ones calculated by MLSSA as possible.

Figure 3.3: *Example of a simulation result from CATT-Acoustic. The concert hall is mapped at a height of 1.25 meters above the floor and the different shades represents different reverberation times. The sound sources are marked A1 and A2 and the dotted lines are their pointed directions. This particular case is a simulation of the reverberation times at the 1 kHz octave band with all the low frequency absorbers in place.*

The goal was to damp the lower peak frequencies and if necessary, the higher frequencies as well. The computer model is very helpful and easy to use in this design phase. It is simple to add new surfaces with desired absorption coefficients at various places and immediately see the new predicted reverberation times, as demonstrated in figure 3.3. The idea behind the low frequency absorbers was to make one specific absorber that would damp both the peaks at 70 and 110 Hz. That means that it would have its centre frequency between those two, at about 88 Hz (logarithmic scale) and its maximum absorption would be at this frequency. The challenge was to get an absorption as high as possible at 70 and 110 Hz.

A typical absorption coefficient graph for a low frequency absorber has the shape of a gaussian curve (see figure 1.2) and with a low Q-value, it is possible to maintain a high coefficient through out the relevant terse band. Under these premises, different surfaces were tested in the model as possible absorption spaces in order to examine the effects on the reverberation.

3.4 Schroeder Backward Integration

In 1964, M.R. Schroeder came up with the brilliant idea that it was possible to estimate reverberation times by using a tape recorder. The method of playing the tape with the decaying echo from a sound burst backwards, gave birth to a new approach in dealing with reverberation time calculation - the backward integration. [5] This is fairly simple to carry out when the room impulse responses already are nicely packaged in vectors as digital samples. The Matlab algorithm that was specially designed for this project, first reverses the concerned vector and makes a new one with all the elements as accumulated sums of their squared values. When this vector in turn is reversed, there will be a surprisingly smooth decaying curve, which usually reveals two specific slopes. In most cases it would be appropriate to calculate a standard reverberation time by considering the later slope, where it is most likely that the sound field is diffuse. In this case, however, the first slope turned out to be a better choice, since the initial slope is closely related to the average damping constant of all excited normal modes [6].

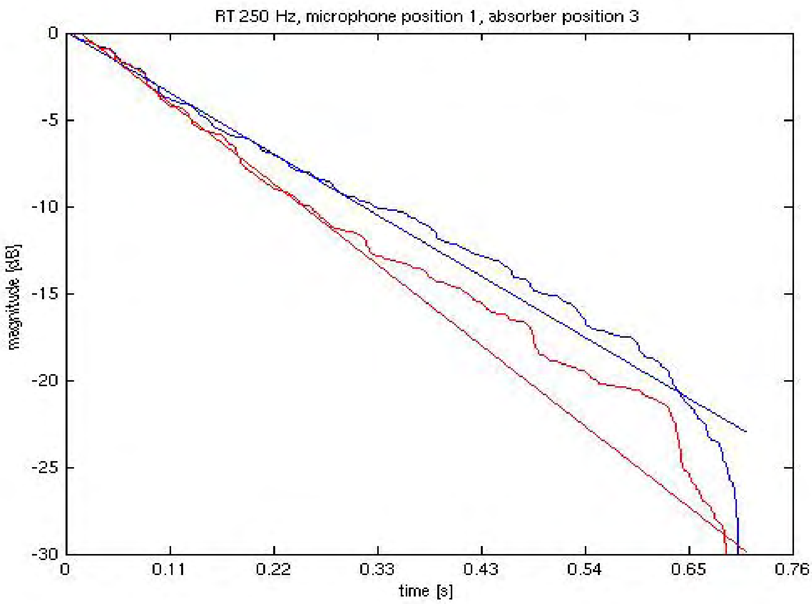


Figure 3.4: *This graph shows the backward integration curves from measurements before and after the prototype absorber was installed in the chamber. They are compared to each other along with their least squares line fittings. The difference in early decay time is very obvious.*

The smoothness of the curves is also an important aspect. If you are unlucky, you might end up with a value that is not representative for the overall decay. It is therefore advisable to use a least squares line fitting and proceed with the mathematical manipulations on the line instead. Once the lines are fitted, the reverberation time estimations are very easy. If the lines span over less than the -60 dB that is customary, one can use a smaller interval, like –15 dB for instance, and multiply it by four.

The absorption coefficient is computed by comparing the times before and after the absorber is installed in the chamber. Since the absorption is a function of the reverberation time, and the original amount of the damping surface in the room does not change, as well as the dimensions, it is a straightforward operation. All the room impulse responses from the measurements were treated according to the procedures described above. Plots from the backward integrations and line fittings can be found in appendix A.

4 Absorber Design

One simple design is the Helmholtz slitted absorber, which was suitable for Oceanen, not only because it is easy to build, but also because it is aesthetically pleasing. The basic construction consists of an airtight chamber with a thin layer of damping material inside, attached at a small distance behind a slitted panel that covers the front. Except for the damping material, the whole absorber can be made out of wood.

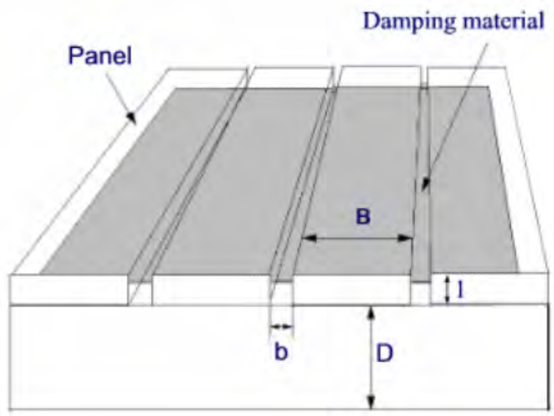


Figure 4.1: *A principle drawing of the slitted Helmholtz absorber, with the design parameters panel width B, slit depth l, slit width b and absorber depth D.*

To construct an absorber that is going to be mounted in a given space, immediately puts some restrictions on it. In this case for instance, the first approach was to put 60 square meters of absorber in the ceiling. That would have limited the total depth of the damper to 20 cm, and all the other parameters had to be adjusted to yield the desired absorption coefficient. Before the computer program was ready, it was assumed that a coefficient about 0.6 was realistic to achieve. Fortunately, the calculations made it possible to optimise the design for a far better result (in fact, up to a theoretical value of about 0.9 at the frequencies of interest).

When the chair board of Oceanen disapproved of the proposition of covering the ceiling with absorbers, considerably less surfaces were available to mount on. This in turn, put further demands on the design. On the other hand, it was possible to make use of a greater depth. The design program showed that an absorber with a depth of 40 cm could give about the same result with less total surface covered than the previous design. With only 46 square meters covered, the reverberation times in all octave bands would be reduced to less than a second and the approval from Oceanen gave even further motivation to build a prototype of the absorber.

Figure 4.2: *The Helmholtz slitted absorber prototype with an absorbing surface of three square meters.*

An interesting aspect of using wooden low frequency absorbers is the properties of the wood, which actually damps higher frequencies as well. It turned out that the 46 square meters of low frequency absorbers would make any additional high frequency damping excessive, at least from a theoretical point of view. The wooden panel is in any case a far better damper than the concrete walls that originally were exposed at the surfaces in question. For every one's convenience, this type of low frequency absorber takes care of the total acoustical issue of Oceanen.

4.1 Calculating the Coefficient

Two specially designed calculation programs were used to determine the geometrical parameters of the absorber, where the two programs worked from two directions - one controlling the other. After the first program had calculated the unknown variables, they were inserted in the second to keep track of the absorption coefficient. This also made it easy to examine the impact of the different input parameters on the absorption coefficient. The absorption coefficient is calculated as

$$\alpha = \frac{4R/Z}{(R/Z+1)^2 + \beta^2\left(\frac{f}{f_0} - \frac{f_0}{f}\right)^2} \qquad (4.1)$$

where R is the airflow resistance, Z is the air impedance, β is a parameter mainly depending on the porosity factor σ of the absorber (equation (4.2)), f is the frequency variable and f_0 is the absorber's resonance frequency (equation (4.3)).

$$\beta = \sqrt{\frac{l}{\sigma D}} \tag{4.2}$$

$$f_0 = \frac{c}{2\pi}\sqrt{\frac{1}{l'D(1+1/\sigma)}} \tag{4.3}$$

All design parameters of the absorber (see figure 4.1) are hidden inside β and f_0 and consists of the porosity factor σ (which is the ratio between the slit width b and the panel width B, and actually also the ratio between the particle velocities outside and inside the slit), depth of the absorber D (i.e. distance from the panel to a rigid wall), l' is the efficient slit depth, which consists of the physical slit depth l, plus the end corrections. As seen in equation (4.3), l' has a significant impact on the resonance frequency. [7] c is the speed of sound. The total participating slit depth is computed according to:

$$l' = l + 0.5b + \frac{2b}{\pi}\ln\frac{\lambda_0}{\pi b} \tag{4.4}$$

Since λ_0 is the wavelength of the resonance frequency f_0, equation (4.3) can only be solved iteratively. It is also interesting how the end correction factor depends on the slit width b. Note that the end corrections are not included in equation (4.2). That is because β is depending on the relations between the real physical distances and not the actual vibrating air volumes.

The parameters seem all to be delicately interlocked with each other and need special care in the algorithm design. Two separate programs are recommended, one that calculates the slit width with the resonance frequency as input, and the other calculating the resonance frequency with the slit width as input. Of course this is a bit more complicated than the thought suggests. However, all code that was used in Matlab for this project is gathered in appendix B.

4.2 Choosing the Damping Material

It is in the damping layer that the actual damping of the sound wave occurs. Through friction, the ordered acoustic motion of the air molecules is transformed into inaudible disordered motion, i.e. heat. The thickness of the material is crucial; if the damper has a high density (which means a high airflow resistance), the greater the risk of the sound waves being reflected. On the other hand, if the flow resistance is low, a thicker layer is required.

As equation (4.1) implies, it is desirable to obtain an airflow resistance *R* as close to the air impedance *Z* as possible. This is not so hard to achieve, since the effectiveness of the absorber is depending on the specific flow resistivity *r* of the damping material. For optimum performance, the damping material should be attached at a small distance behind the front panel, mainly because of the end correction phenomenon. This will give the absorber a total airflow resistance that will equal the flow resistance of the damping material without interference from the construction. [8] If the specific flow resistivity is known (or the density, since the resistivity is a function thereof) the thickness of the layer is easy to calculate, because of the equality *R*=*Z,* where the airflow resistance *R* simply is the specific flow resistivity multiplied with the thickness. [9]

The rock wool brand Paroc is commonly used as building insulation and has suitable properties for sound absorbing. The specific type used in the prototype had a density of 150 kg/m^3 and a specific resistivity *r* of 10500 Ns/m^4. To get an airflow resistance that equalled the air impedance according to the previous reasoning, a suitable thickness turned out to be 0.037 m, when the values were inserted in equation (4.5):

$$t = \frac{R}{r} = \frac{Z}{r} = \frac{\rho_0 c}{r} \tag{4.5}$$

R is the airflow resistance, which should equal the characteristic air impedance *Z*. *r* is the specific airflow resistivity of the damping material. The density of air ρ_0 is 1.125 kg/m^3 and the speed of sound *c* is approximately 340 m/s.

The available thickness of Paroc rock wool closest to 0.037 m is 0.045 m, which gives about 22% higher airflow resistance *R* than the characteristic air impedance *Z*. But considering equation (4.1), the difference has very little influence on the absorption coefficient. This is easy to see at the resonance frequency, when the right term in the denominator is zero. Then the total coefficient will still equal 0.99 with the figures above inserted.

5 Results

5.1 Initial Measurements

The initial measurements at Oceanen were carried out in two different manners. Both with the heavy drapes fully extended and withdrawn, in order to investigate the impact of the drapes on the reverberation times. The calculated reverberation times from the measurements are shown in the tables below, with each table handling its specific octave band, ranging from 125 Hz to 4 kHz. The calculations are computed by MLSSA and the unit is seconds.

125 Hz

Microphone position	1	2	3	4	5
With drapes	1.22	1.33	1.36	1.20	1.51
Without drapes	1.25	1.35	1.29	1.36	1.54

250 Hz

Microphone position	1	2	3	4	5
With drapes	0.92	1.52	1.06	1.03	0.91
Without drapes	1.00	1.45	0.97	0.91	1.04

500 Hz

Microphone position	1	2	3	4	5
With drapes	0.89	0.91	0.91	0.91	0.88
Without drapes	0.96	0.95	0.93	0.96	0.95

1 kHz

Microphone position	1	2	3	4	5
With drapes	0.80	0.80	0.83	0.85	0.84
Without drapes	0.92	0.94	0.93	0.97	0.96

2 kHz

Microphone position	1	2	3	4	5
With drapes	0.81	0.79	0.83	0.84	0.83
Without drapes	0.91	0.91	0.92	0.92	0.90

4 kHz

Microphone position	1	2	3	4	5
With drapes	0.69	0.68	0.70	0.69	0.69
Without drapes	0.76	0.77	0.80	0.77	0.77

As the tables show, the greatest differences are found above 500 Hz where the impact of the drapes is significant. Since the times may differ more than 100 ms and are therefore audible in the region between 500 Hz to 4 kHz, it was concluded that the acoustic properties of the drapes were far too important to be discarded. Only the case with the drapes fully extended would be considered, as they appeared to be an extensive part of the high frequency damping.

5.2 Absorber Design Parameters

The first proposition, where absorbers were to be mounted in the ceiling (due to its large available area), was rejected by the association. After all, a reduction of the height to the ceiling may improve the acoustics but not the feel of space. This led to a second suggestion, which would be to use as much of the wall area as possible. When the maximum absorber area is known, the other design parameters can be calculated to obtain an absorption coefficient as high as possible, according to equations (4.1), (4.2), (4.3), (4.4) and (4.5). For convenience, the actual available dimensions of the appropriate material would also be considered. Oceanen agreed to an absorber depth of 0.4 meters. The other parameters were calculated in Matlab and presented in the table below.

Parameter	**Dimension**
Desired absorption coefficient at peaks, α	0.8
Front panel width, *B*	0.120 m
Front panel thickness (slit depth), *L*	0.022 m
Absorber depth, *D*	0.400 m
Damper resistivity, *r*	10500 $\mathrm{Ns/m^4}$
Slit with, *b*	**0.0017 m**
Damper thickness, *t*	**0.037 m**

By choosing the fixed parameters wisely and in regards of available material, the design program calculated and returned the two unknown values, the slit width *b* and the damper thickness *t*. With the help of a control program, the obtained parameters could easily be verified by calculating them in reverse order. Complete Matlab code is found in appendix B.

5.3 Absorber Measurements

After the prototype was built directly at the laboratory, measurements concerning room impulse responses were carried out by using MLSSA. The RIR:s were then imported to Matlab where extensive filtering and calculations of the reverberation times were done. Finally the data could be transformed into absorption coefficients to be used in CATT-Acoustic. The coefficients are as follows:

Octave band	125 Hz	250 Hz	500 Hz	1 kHz	2 kHz	4 kHz
Absorption coefficient	0.44	0.37	0.18	0.14	0.18	0.23

Diagrams of the Schroeder backward integrations obtained from all the RIR:s can be found in appendix A.

5.4 Theory vs. Reality

Additional calculations were also carried out in order to get a glimpse of the absorber's function below 125 Hz, where the real problem actually starts. Even if the measurements at these low frequency might not be valid due to the Schroeder frequency, it would still give some indication on how well tuned the prototype would be, for instance. To apprehend more data points, the filter was slightly narrowed and the frequency range was extended down to 63 Hz. In the graph below, the curve from the design program is compared with the one calculated from the measurement data.

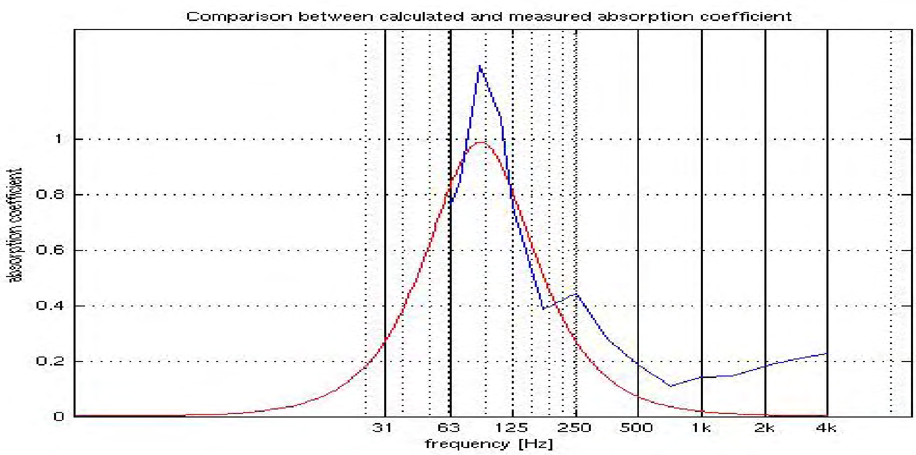


Figure 5.1: *The graph shows the calculated absorption coefficient together with the processed measured data. The measured room impulse responses have been band pass filtered, backward integrated and least squares line fitted in order to calculate the reverberation times and finally the absorption coefficients.*

The peaks from both curves appears to align well indeed. The coefficient from the measurements exceeds the value 1, which in reality would not be possible. The reason it does so, is probably because of diffraction effects around the edges of the construction. The sides, as well as the back, are also exposed to the sound and will have substantial absorption. The odd values will dominate in some cases and result in doubtful figures. In any case, the diagram still shows, for the purpose, a fully functional absorber.

5.5 Simulation Results

After the calculated absorption coefficients were inserted in CATT-Acoustic and yet another simulation was completed, the final reverberation times were obtained which also were the last important results that could be compared with the initial measurements. The tables below shows the simulated influence of the mounted absorbers.

Simulation without absorbers (with drapes)

Microphone position	125 Hz	250 Hz	500 Hz	1 kHz	2 kHz	4 kHz
1	1.3	1.0	1.0	0.9	0.9	0.7
2	1.4	1.0	0.9	0.8	0.8	0.6
3	1.4	1.1	0.9	0.9	0.8	0.7
4	1.2	0.9	0.8	0.7	0.8	0.7
5	1.2	0.9	0.8	0.8	0.8	0.7
Mean	1.30	0.98	0.88	0.82	0.82	0.68
Mean MLSSA	1.32	1.01	0.90	0.82	0.82	0.69

Simulation with absorbers (and drapes)

Microphone position	125 Hz	250 Hz	500 Hz	1 kHz	2 kHz	4 kHz
1	1.1	0.9	1.0	0.9	0.9	0.7
2	1.1	0.9	0.9	0.9	0.8	0.6
3	1.1	1.0	0.9	0.9	0.8	0.7
4	1.0	0.8	0.8	0.8	0.8	0.6
5	0.9	0.8	0.8	0.8	0.8	0.6
Mean	1.04	0.88	0.88	0.86	0.82	0.64
ΔRT CATT	0.26	0.10	0.00	-0.04	0.00	0.04

The computer model was very well adjusted to fit the values from the initial measurements at Oceanen, as shown in the first table. In the second table, the comparisons between the simulation with and without absorbers show quite high impact on the low frequencies, up to a difference as large as 0.26 s, and practically no difference at 500 Hz and above.

6 Conclusions and Discussion

The goal of the project was to give verified suggestions on how to improve the room acoustics of the concert hall in Kulturhuset Oceanen. The measurements in the hall showed that the main problems consisted of three low resonance frequencies. It was decided that two of those, 70 and 110 Hz, would be treated with Helmholtz absorbers. In order to deliver validated strategic solutions, a prototype was built. Measurements on the prototype made it possible to calculate its absorption coefficients in six octave bands, which in turn were inserted in CATT-Acoustic. The subsequent detailed simulations could finally give the wanted predictions.

The final results show that it is possible to reduce the reverberation times substantially, especially in the targeted frequency range, even with the lesser covering area. This depends on how much the total absorbing area is increased. If, for instance, the new added absorbers will double the original Sabine surface, the reverberation time will be reduced by half. This becomes obvious when one considers Sabine's formula:

$$T = \frac{0.163V}{S\overline{\alpha}} \tag{6.1}$$

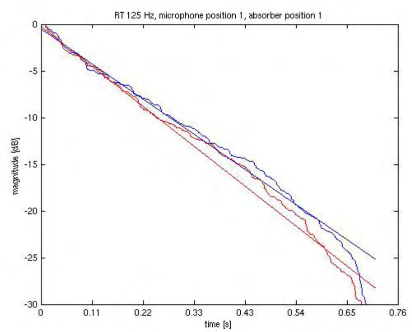
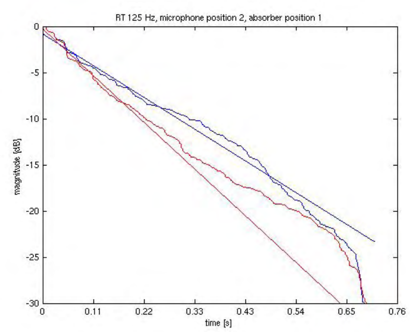
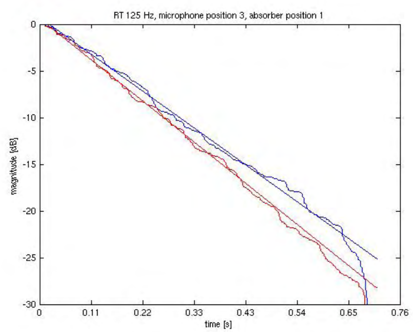
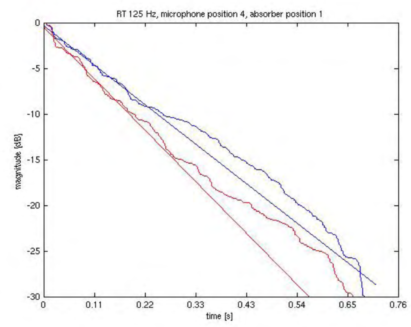
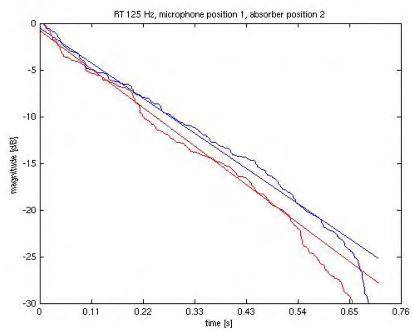
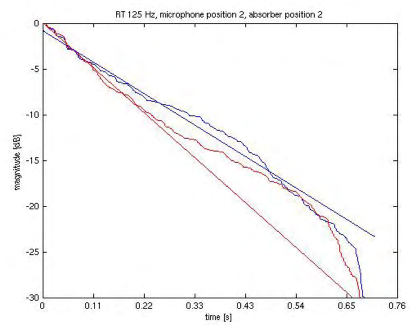
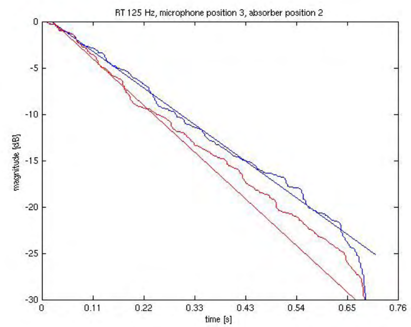
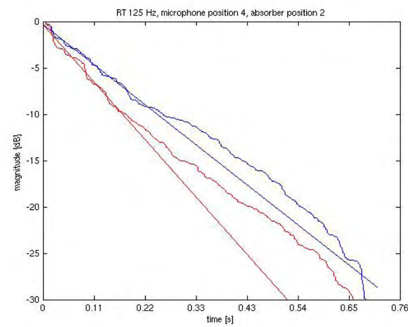
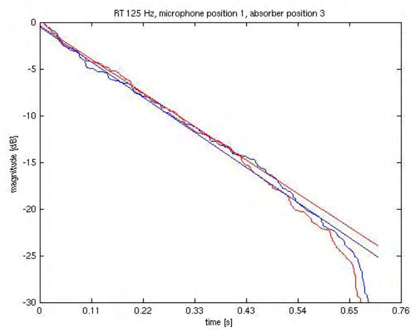
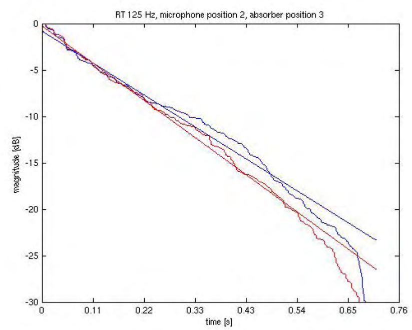
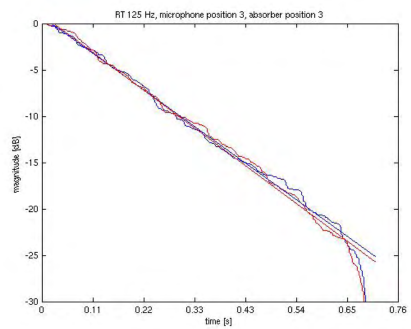
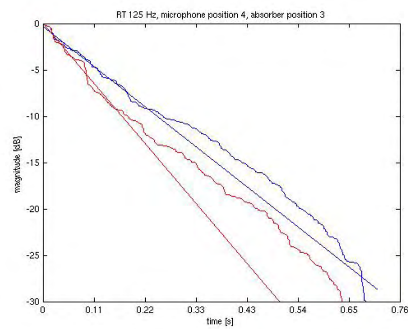
where S is the absorbing surface , $\overline{\alpha}$ is the average absorption coefficient and V is the volume of the enclosure.

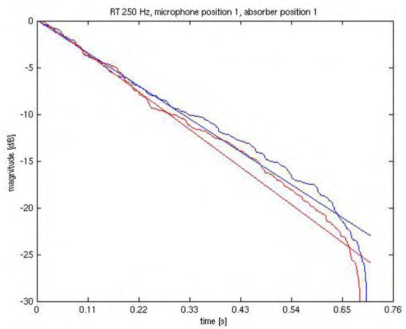
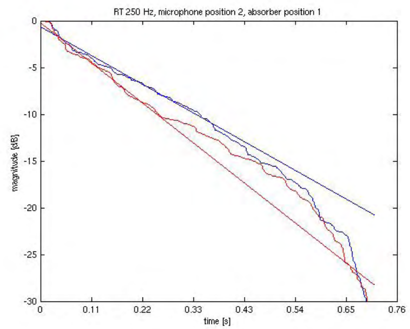
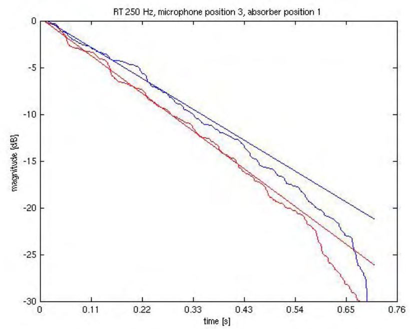
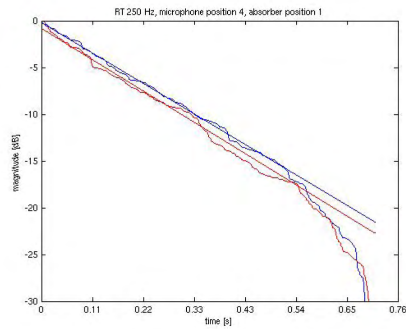
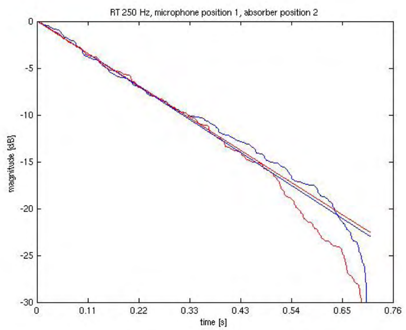
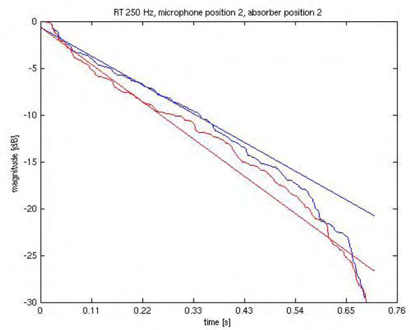
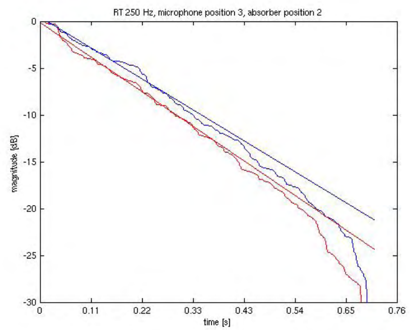
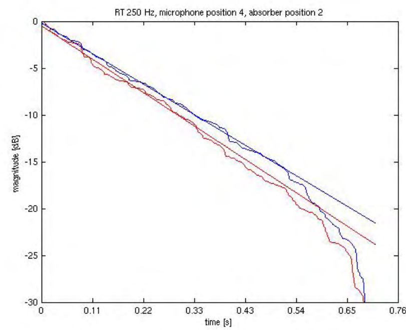
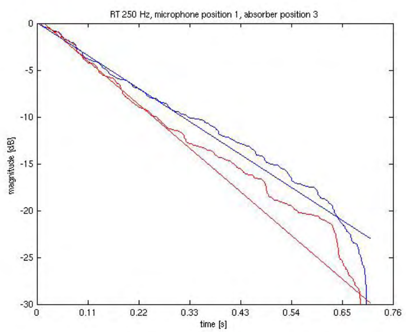
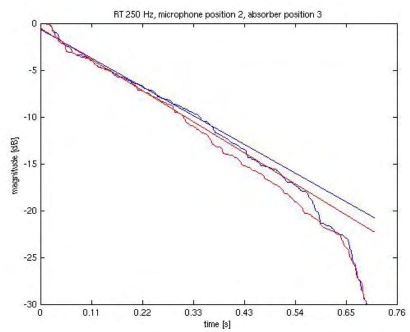
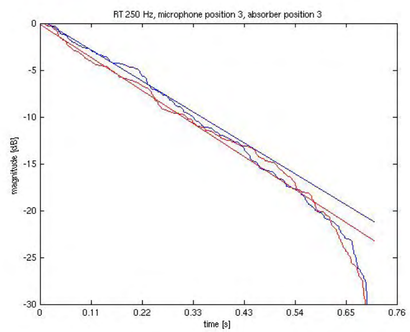
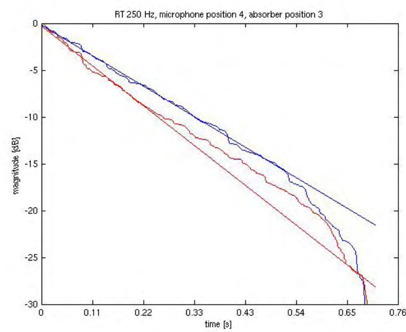
Concerning this project, it is clear how important a computer simulation model is in predicting approximate behaviour of altered acoustical circumstances. The total amount of Sabine surface for the targeted frequencies would increase with almost thirty percent, which is easy to see since the relation between decreased reverberation and increased absorption area is inversely proportional, according to equation (6.1). If the Sabine area is increased by thirty percent, the reverberation time is reduced by a third.

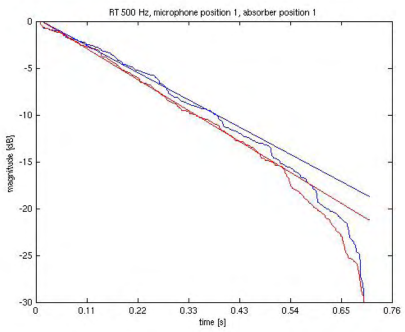
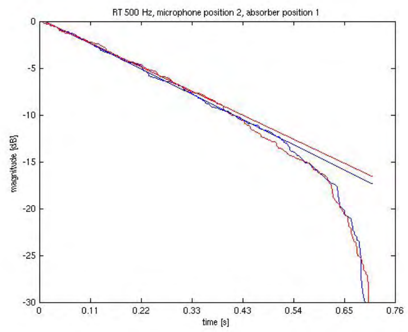
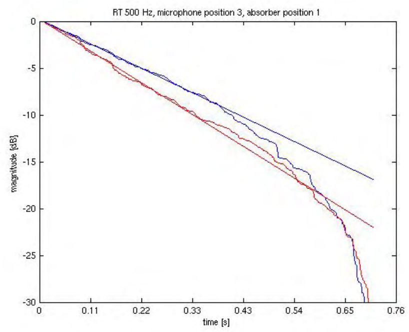
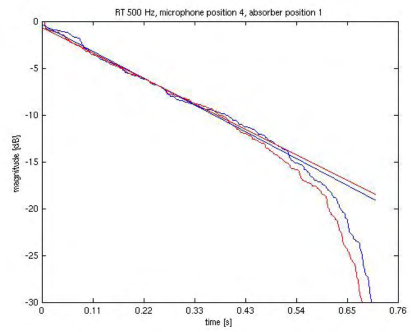
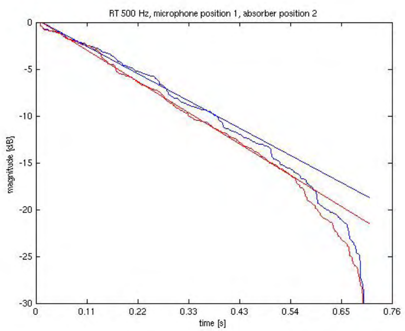
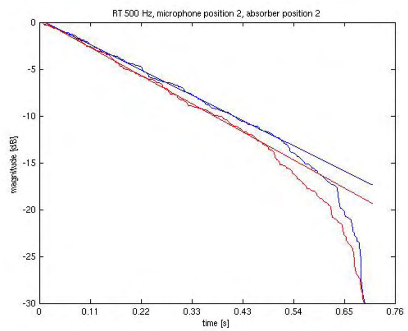
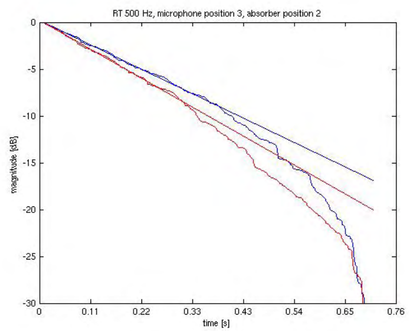
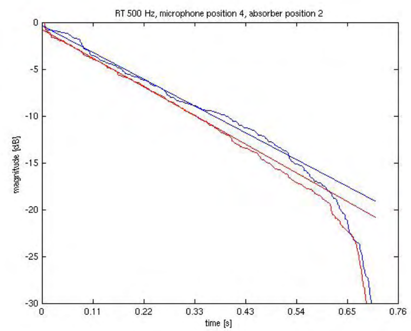
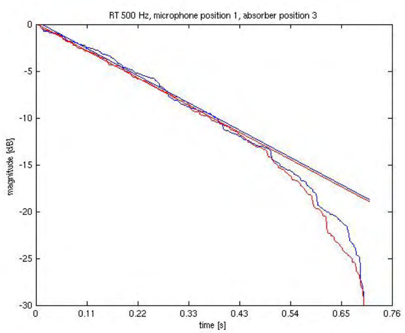
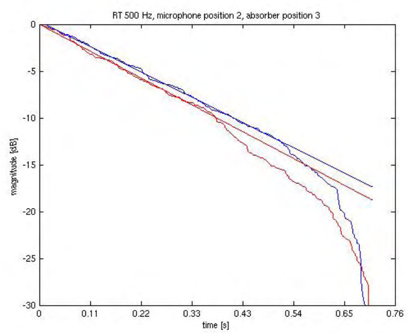
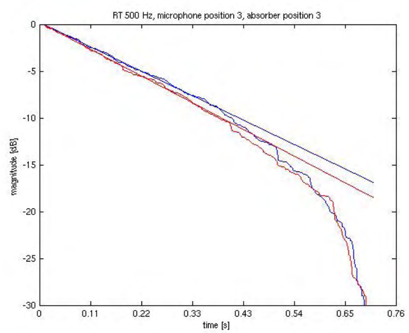
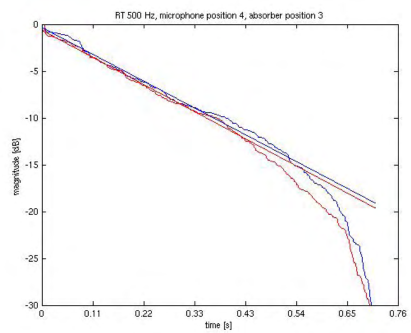
The simulation results show no change of reverberation times in the octave bands above 500 Hz. One probable cause of that, is the instability of the measurement data, which has been averaged in several steps. Some of the data gives either too high contributions or too low or even negative. This in turn, may be due to certain circumstances during the measurement set-up. When lower frequencies were examined, like the ones below 500 Hz, there is reason to suspect that the sound field was no longer diffuse and that the direct sound from the source may have played an unwanted but important roll, distorting the measurements. The philosophy behind the analysis of these somewhat distorted data, is to let the values that are too high cancel out the ones that are too low or negative and thus leave a result that represents a mean. From a statistical point of view, these odd values should be about the same in number. Though this seems like a rough estimate, it still gives some valuable information concerning the absorber system's functionality, tuning and overall damping.

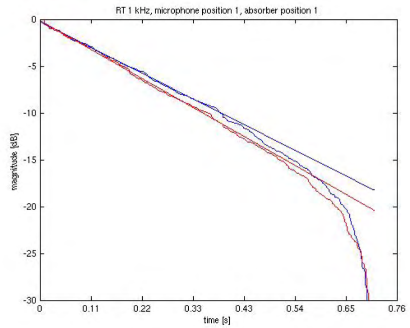
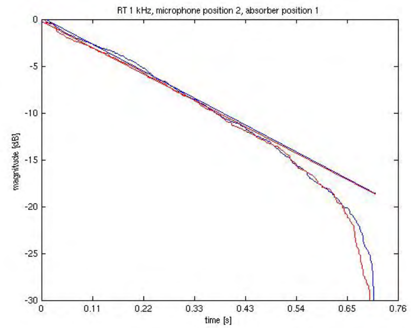
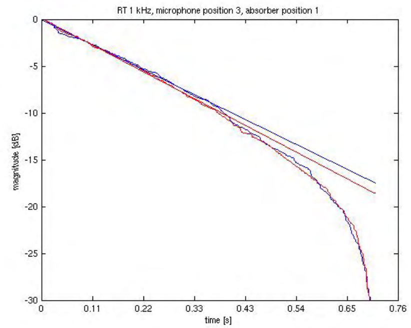
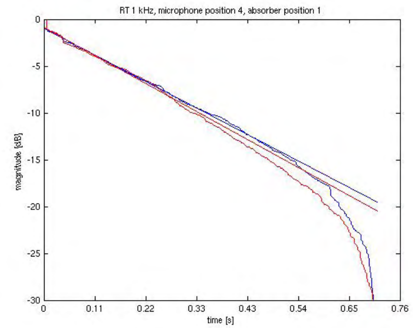
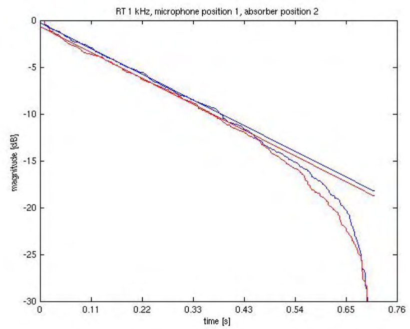
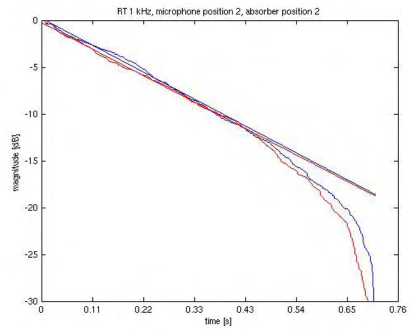
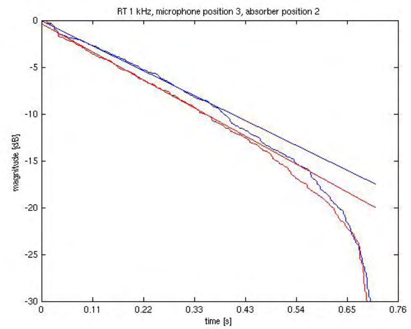
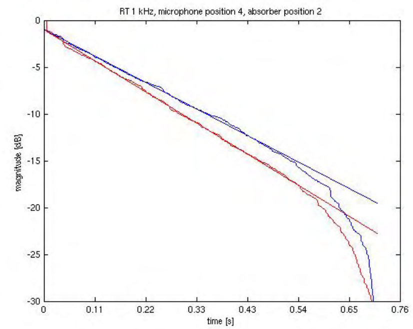
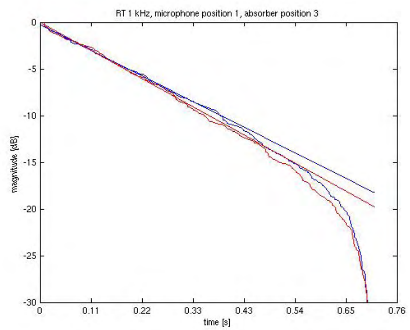
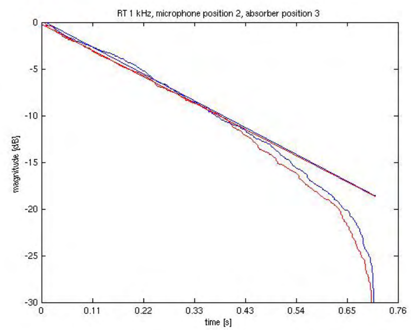
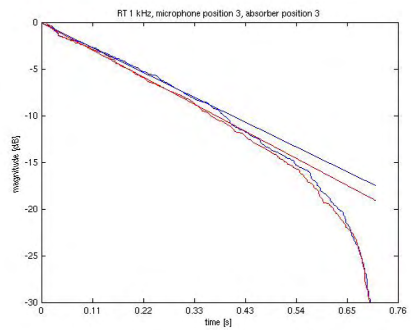
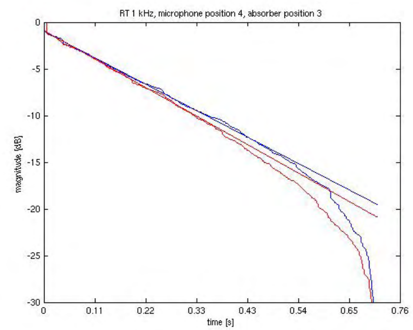
Appendix A

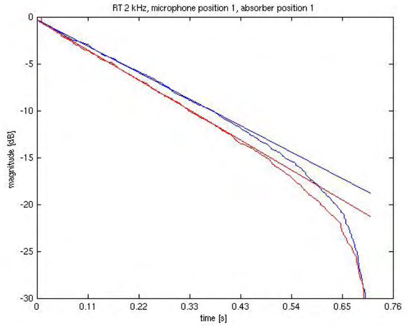
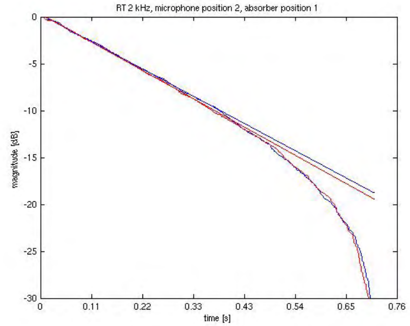
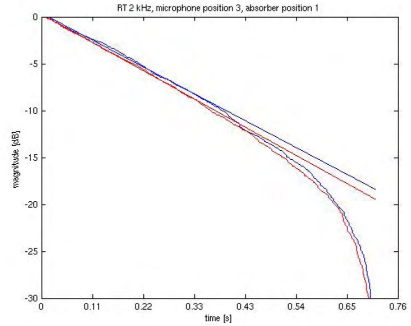
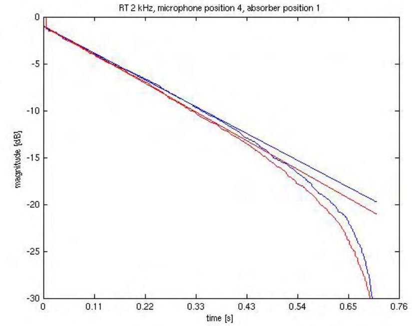
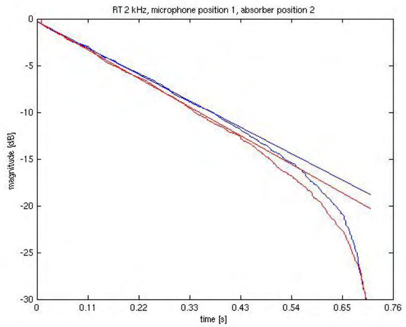
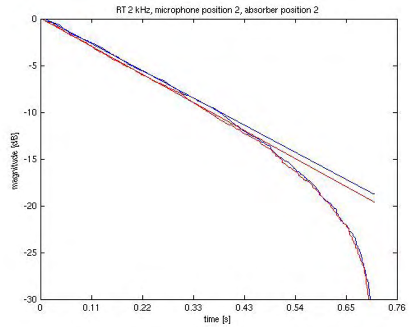
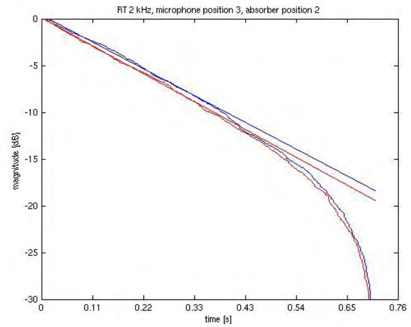
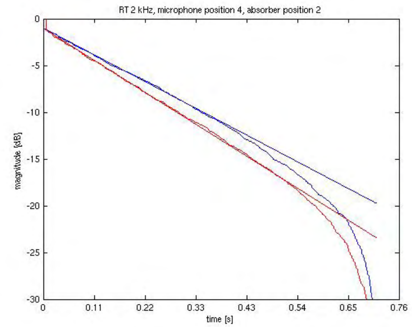
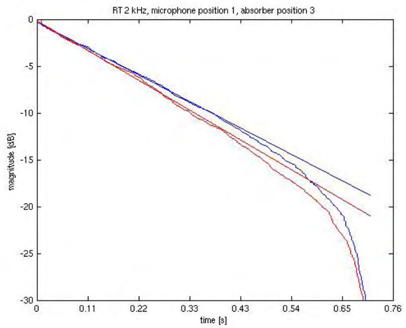
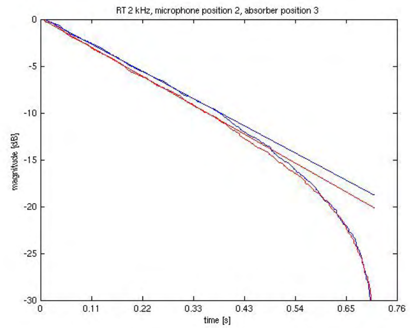
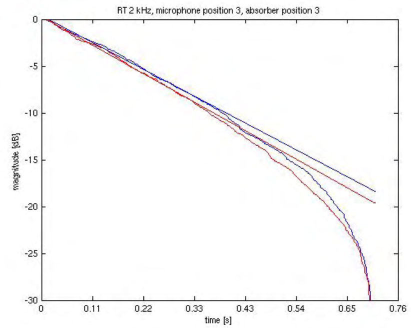
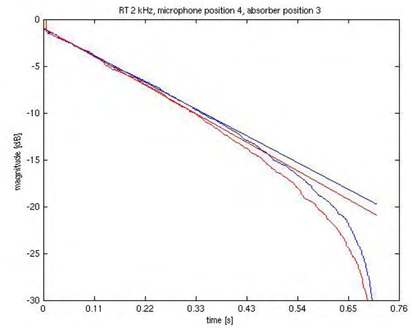
Schroeder Backward Integration results

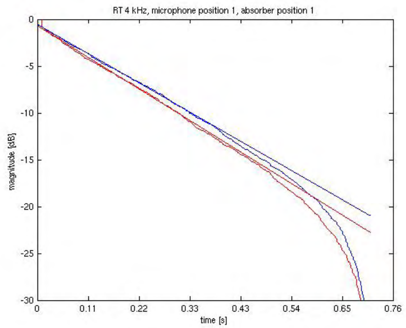
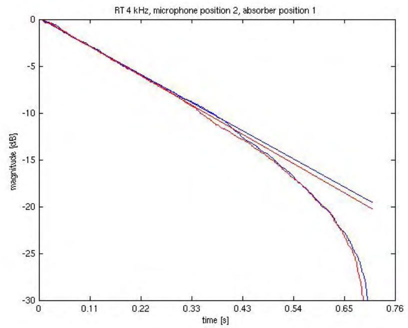
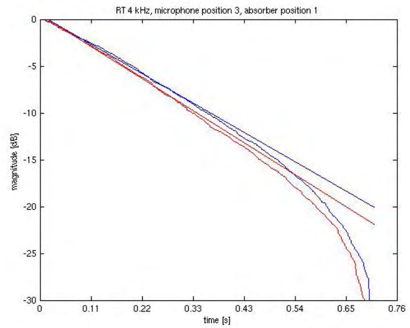
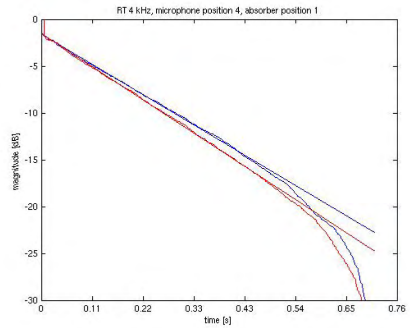
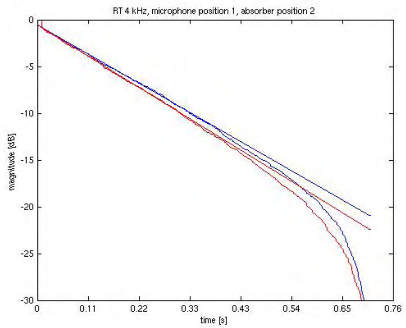
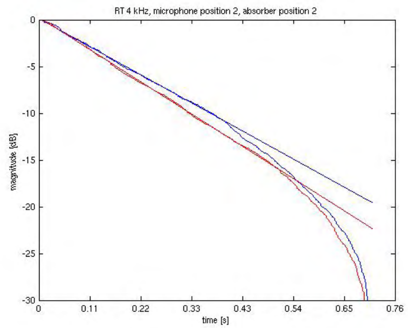
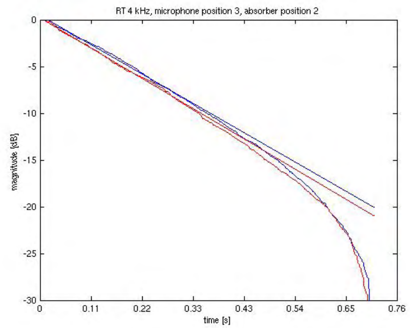
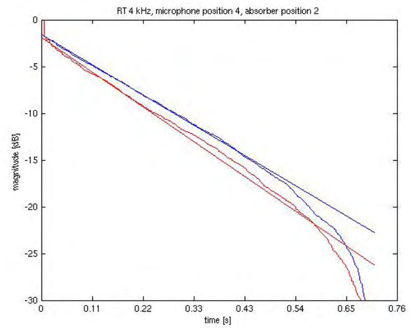
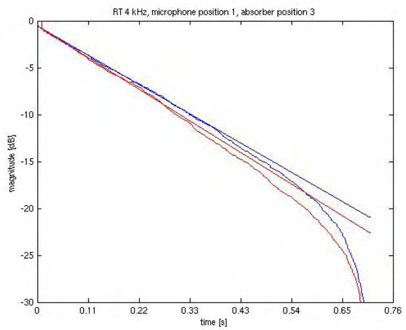
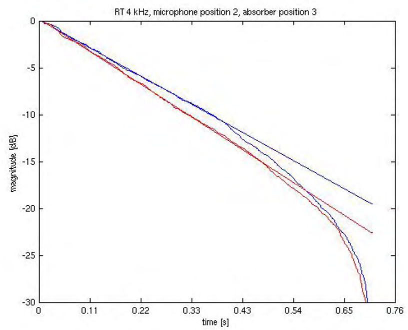
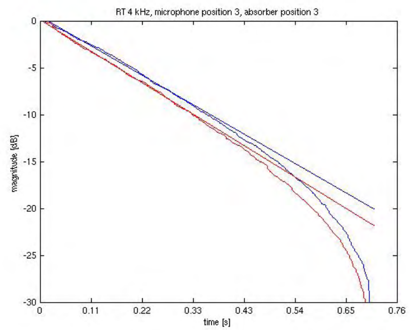
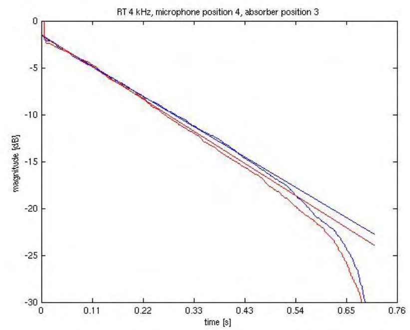
125 Hz	Microphone position 1	Microphone position 2	Microphone position 3	Microphone position 4
Absorber position 1				
Absorber position 2				
Absorber position 3				

<table>
<tr><th>250 Hz</th><th>Microphone position 1</th><th>Microphone position 2</th><th>Microphone position 3</th><th>Microphone position 4</th></tr>
<tr><td>Absorber position 1</td>
<td></td>
<td></td>
<td></td>
<td></td></tr>
<tr><td>Absorber position 2</td>
<td></td>
<td></td>
<td></td>
<td></td></tr>
<tr><td>Absorber position 3</td>
<td></td>
<td></td>
<td></td>
<td></td></tr>
</table>

500 Hz	Microphone position 1	Microphone position 2	Microphone position 3	Microphone position 4
Absorber position 1				
Absorber position 2				
Absorber position 3				

1 kHz	Microphone position 1	Microphone position 2	Microphone position 3	Microphone position 4
Absorber position 1				
Absorber position 2				
Absorber position 3				

2 kHz	Microphone position 1	Microphone position 2	Microphone position 3	Microphone position 4
Absorber position 1				
Absorber position 2				
Absorber position 3				

<table>
<tr><th>4 kHz</th><th>Microphone position 1</th><th>Microphone position 2</th><th>Microphone position 3</th><th>Microphone position 4</th></tr>
<tr><td>Absorber position 1</td>
<td>
</td>
<td>
</td>
<td>
</td>
<td>
</td></tr>
<tr><td>Absorber position 2</td>
<td>
</td>
<td>
</td>
<td>
</td>
<td>
</td></tr>
<tr><td>Absorber position 3</td>
<td>
</td>
<td>
</td>
<td>
</td>
<td>
</td></tr>
</table>

Appendix B

Matlab Code

```
% This matlab program calculates the slit width, absorber depth,
% thickness of the damping material thickness and draws a graph of
% the absorption coefficient for slitted low frequency absorbers with
% the damping material consisting of mineral wool attached at a small
% distance behind the slits.

function [b, d, t] = absorber(f_u,f_o,alfa_u,B,l,rho)

% f_u    is the lower frequency that will be damped
% f_o    is the upper frequency that will be damped
% alfa_u is the desired absorption coefficient at the frequencies
% B     is the width of the panel
% l     is the depth of the slits
% rho   is the density of the damping material

format long;

% The resonance frequency
f0 = f_u*sqrt(f_o/f_u);

% The desired beta-value
beta_desired = sqrt((4/alfa_u-4)/(sqrt(f_u/f_o)-sqrt(f_o/f_u))^2);

% Initial default depth
d=0.5;

% A help function is engaged
[b,beta_calc,t,f,alfa] = abscalc(f_o,f0,B,l,d,rho);

while abs(beta_calc-beta_desired) > 0.01
   if beta_calc > beta_desired
     d=d+0.01;
     [b,beta_calc,t,f,alfa] = abscalc(f_o,f0,B,l,d,rho);
   end
   if beta_calc < beta_desired
     d=d-0.01;
     [b,beta_calc,t,f,alfa] = abscalc(f_o,f0,B,l,d,rho);
   end
end

figure(1)
semilogx(f,alfa,f_u,alfa_u,'*',f_o,alfa_u,'*');
xlim([f_u/10 f_o*10]);
ylim([0 1]);
grid on;
```

```
% This is a help function that is called repeatedly from absorber.m
% and is used iteratively to give new values of beta to compare with
% along with the other updated parameters

function [b,beta_calc,t,f,alfa] = abscalc(f_o,f0,B,l,d,rho)

% The speed of sound
c = 343;

% Calculation of slit width
b = 0.1;
btemp = 0.01;

while abs(b-btemp) > 1e-5;
 b = btemp;
 btemp=B/(1/(d*(l+0.5*b+2*b/pi*log(c/(b*pi*f0))))*(c/(2*pi*f0))^2-1);
end

% The end correction
lprim = l+0.5*b+2*b/pi*log(c/(b*pi*f0));

% The density of air
rho0 = 1.125;

% The air impedance
Z = rho0*c;
R = Z;

% The airflow resistivity (this formula is derived from a graph)
r = 0.051.*rho.^1.52;

% The porosity factor of the absorber
sigma = (b/B);

% The beta factor
beta_calc = sqrt(lprim/(sigma*d));

% The thickness of damping
t = Z/r;

% The frequency vector
f = [1:1:f_o*10];

% The absorption coefficient
alfa = (4.*R./Z)./((R./Z+1).^2+beta_calc.^2.*(f./f0-f0./f).^2);
```

```
% This program will verify the results from absorber.m by inserting
% the parameters and calculating everything backwards and drawing the
% alpha graph again.

function a2 = absorber2(f_u,f_o,alfa_u,B,l,b,d,rho,t)

% The actual values in the specific design…
% absorber2(70,110,0.8,0.12,0.022,0.0017,0.40,150,0.045)

% The centre frequency
f0 = f_u*sqrt(f_o/f_u);

c = 343;
rho0 = 1.125;
Z = rho0*c;

lprim = l+b/2+2*b/pi*log(c/b/pi/f0);
beta = sqrt(lprim*B/b/d);
sigma = (b/B);

r = 10500; % 51.*rho.^1.52;
%R1=t*r/sigma; % This is only valid if the damping material is
               % attached immediately behind the panel and is very
               % thin.
R2=t*r;

f = [1:1:f_o*10];

%alfa1 = (4.*R1./Z)./((R1./Z+1).^2+beta.^2.*(f./f0-f0./f).^2);
alfa2 = (4.*R2./Z)./((R2./Z+1).^2+beta.^2.*(f./f0-f0./f).^2);

a0 = 1;
a1 = 10;
a2 = 100;
a3 = 1000;

figure(1)
semilogx(f,alfa2,'r',f_u,alfa_u,'*',f_o,alfa_u,'*');
xlim([f_u/10 f_o*10]);
ylim([0 1]);
set(gca,'XTickLabel',{a1;a2;a3})
title('Calculated absorption coefficient')
xlabel('Frequency [Hz]')
ylabel('Absorption coefficient')
grid on;
```

```
% This function loads room impulse responses using readmls.m and
% filter them in given octave bands and performs Schroeder backward
% integration. Besides calculating some statistical data on the
% measurements, the main purpose is to yield absorption coefficients
% in the chosen frequency bands. They are obtained through comparison
% of the reverberation times with and without absorber prototype,
% calculated from least squares line fitting on the backward
% integrations. The additional statistical parameters are the
% complete T60 matrix (reverberation time at -60 dB), complete alpha
% matrix (all absorption coefficients for all measurements) as well
% as the average, standard deviation for the absorption coefficients,
% the difference in reverberation times with and without absorber, as
% well as average and standard deviation.

function p = prototype

% Load all subwoofer measurement data from the four microphone
% positions

[sub_pos1_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SUB.TIM');
[sub_pos1_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABS.TIM');
[sub_pos1_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABT.TIM');
[sub_pos1_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABU.TIM');

[sub_pos2_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SUB.TIM');
[sub_pos2_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABS.TIM');
[sub_pos2_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABT.TIM');
[sub_pos2_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABU.TIM');

[sub_pos3_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SUB.TIM');
[sub_pos3_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABS.TIM');
[sub_pos3_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABT.TIM');
[sub_pos3_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABU.TIM');

[sub_pos4_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SUB.TIM');
[sub_pos4_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABS.TIM');
[sub_pos4_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABT.TIM');
[sub_pos4_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABU.TIM');

% Load all top speaker measurement data from the four microphone
% positions

[top_pos1_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TOP.TIM');
[top_pos1_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABS.TIM');
[top_pos1_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABT.TIM');
[top_pos1_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABU.TIM');

[top_pos2_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TOP.TIM');
```

```
[top_pos2_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABS.TIM');
[top_pos2_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABT.TIM');
[top_pos2_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABU.TIM');

[top_pos3_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TOP.TIM');
[top_pos3_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABS.TIM');
[top_pos3_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABT.TIM');
[top_pos3_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABU.TIM');

[top_pos4_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TOP.TIM');
[top_pos4_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABS.TIM');
[top_pos4_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABT.TIM');
[top_pos4_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABU.TIM');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Frequency responses

%figure(1)
%semilogx(abs(fft(sub_pos1_noabs)))
%xlim([20 5000])
%title('FFT, subwoofer, microphone position 1, no absorber')

%figure(2)
%semilogx(abs(fft(sub_pos2_noabs)))
%xlim([20 5000])
%title('FFT, subwoofer, microphone position 2, no absorber')

%figure(3)
%semilogx(abs(fft(sub_pos3_noabs)))
%xlim([20 5000])
%title('FFT, subwoofer, microphone position 3, no absorber')

%figure(4)
%semilogx(abs(fft(sub_pos4_noabs)))
%xlim([20 5000])
%title('FFT, subwoofer, microphone position 4, no absorber')

%figure(5)
%semilogx(abs(fft(top_pos1_noabs)))
%xlim([20 5000])
%title('FFT, top speaker, microphone position 1, no absorber')

%figure(6)
%semilogx(abs(fft(top_pos2_noabs)))
%xlim([20 5000])
%title('FFT, top speaker, microphone position 2, no absorber')

%figure(7)
%semilogx(abs(fft(top_pos3_noabs)))
%xlim([20 5000])
%title('FFT, top speaker, microphone position 3, no absorber')

%figure(8)
```

```
%semilogx(abs(fft(top_pos4_noabs)))
%xlim([20 5000])
%title('FFT, top speaker, microphone position 4, no absorber')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculate the differences in reverberation times (requires help
% function protoype2.m)

%[T60_125, deltaT60_125] = prototype2(125, mlsfs, sub_pos1_noabs,
sub_pos2_noabs, sub_pos3_noabs, sub_pos4_noabs, sub_pos1_abs1,
sub_pos2_abs1, sub_pos3_abs1, sub_pos4_abs1, sub_pos1_abs2,
sub_pos2_abs2, sub_pos3_abs2, sub_pos4_abs2, sub_pos1_abs3,
sub_pos2_abs3, sub_pos3_abs3, sub_pos4_abs3);

%[T60_250, deltaT60_250] = prototype2(250, mlsfs, sub_pos1_noabs,
sub_pos2_noabs, sub_pos3_noabs, sub_pos4_noabs, sub_pos1_abs1,
sub_pos2_abs1, sub_pos3_abs1, sub_pos4_abs1, sub_pos1_abs2,
sub_pos2_abs2, sub_pos3_abs2, sub_pos4_abs2, sub_pos1_abs3,
sub_pos2_abs3, sub_pos3_abs3, sub_pos4_abs3);

%[T60_500, deltaT60_500] = prototype2(500, mlsfs, top_pos1_noabs,
top_pos2_noabs, top_pos3_noabs, top_pos4_noabs, top_pos1_abs1,
top_pos2_abs1, top_pos3_abs1, top_pos4_abs1, top_pos1_abs2,
top_pos2_abs2, top_pos3_abs2, top_pos4_abs2, top_pos1_abs3,
top_pos2_abs3, top_pos3_abs3, top_pos4_abs3);

%[T60_1k, deltaT60_1k] = prototype2(1000, mlsfs, top_pos1_noabs,
top_pos2_noabs, top_pos3_noabs, top_pos4_noabs, top_pos1_abs1,
top_pos2_abs1, top_pos3_abs1, top_pos4_abs1, top_pos1_abs2,
top_pos2_abs2, top_pos3_abs2, top_pos4_abs2, top_pos1_abs3,
top_pos2_abs3, top_pos3_abs3, top_pos4_abs3);

%[T60_2k, deltaT60_2k] = prototype2(2000, mlsfs, top_pos1_noabs,
top_pos2_noabs, top_pos3_noabs, top_pos4_noabs, top_pos1_abs1,
top_pos2_abs1, top_pos3_abs1, top_pos4_abs1, top_pos1_abs2,
top_pos2_abs2, top_pos3_abs2, top_pos4_abs2, top_pos1_abs3,
top_pos2_abs3, top_pos3_abs3, top_pos4_abs3);

%[T60_4k, deltaT60_4k] = prototype2(4000, mlsfs, top_pos1_noabs,
top_pos2_noabs, top_pos3_noabs, top_pos4_noabs, top_pos1_abs1,
top_pos2_abs1, top_pos3_abs1, top_pos4_abs1, top_pos1_abs2,
top_pos2_abs2, top_pos3_abs2, top_pos4_abs2, top_pos1_abs3,
top_pos2_abs3, top_pos3_abs3, top_pos4_abs3);

% Average times and standard deviations (requires help function
% statistics.m)

%[dT60_average_125, sigma_dT_125, T60_average_125_noabs,
T60_average_125_abs] = statistics(deltaT60_125, T60_125);
%[dT60_average_250, sigma_dT_250, T60_average_250_noabs,
T60_average_250_abs] = statistics(deltaT60_250, T60_250);
%[dT60_average_500, sigma_dT_500, T60_average_500_noabs,
T60_average_500_abs] = statistics(deltaT60_500, T60_500);
```

```
%[dT60_average_1k, sigma_dT_1k, T60_average_1k_noabs,
T60_average_1k_abs] = statistics(deltaT60_1k, T60_1k);
%[dT60_average_2k, sigma_dT_2k, T60_average_2k_noabs,
T60_average_2k_abs] = statistics(deltaT60_2k, T60_2k);
%[dT60_average_4k, sigma_dT_4k, T60_average_4k_noabs,
T60_average_4k_abs] = statistics(deltaT60_4k, T60_4k);

% Absorption coefficients (requires help function absorption.m)

% Room volume
V=96.6;
% Absorber surface
S=3;

%[alfa_125, alfa_average_125, sigma_alfa_125] = absorption(T60_125,
V, S);
%[alfa_250, alfa_average_250, sigma_alfa_250] = absorption(T60_250,
V, S);
%[alfa_500, alfa_average_500, sigma_alfa_500] = absorption(T60_500,
V, S);
%[alfa_1k, alfa_average_1k, sigma_alfa_1k] = absorption(T60_1k, V,
S);
%[alfa_2k, alfa_average_2k, sigma_alfa_2k] = absorption(T60_2k, V,
S);
%[alfa_4k, alfa_average_4k, sigma_alfa_4k] = absorption(T60_4k, V,
S);

% Results (the end number, _XX, means the actual frequency band and
% can be changed respectively)

%T60_125
%alfa_125
%alfa_average_125
%sigma_alfa_125
%deltaT60_125
%dT60_average_125
%sigma_dT_125

%alfa_average = [alfa_125_average, alfa_250_average,
alfa_500_average, alfa_1k_average, alfa_2k_average, alfa_4k_average];
```

```
% This is a help program that filters a given set of room impulse
% vectors, then creates reverberation time vectors which are
% normalized and converted to dB-scale. Next they will undergo least
% squares line fitting and the program draws additional control
% plots. From the line fits, reverberation T60-matrices are formed as
% well as the delta T60´s.

function [T60, deltaT60] = prototype2(CF, mlsfs, POS1_0, POS2_0,
POS3_0, POS4_0, POS1_1, POS2_1, POS3_1, POS4_1, POS1_2, POS2_2,
POS3_2, POS4_2, POS1_3, POS2_3, POS3_3, POS4_3)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 1. Filter (uses the help function butterfilter.m)

n = 2;
[b,a] = butterfilter(CF,mlsfs,n);

POS1_0_filt = filtfilt(b,a,POS1_0);
POS2_0_filt = filtfilt(b,a,POS2_0);
POS3_0_filt = filtfilt(b,a,POS3_0);
POS4_0_filt = filtfilt(b,a,POS4_0);

POS1_1_filt = filtfilt(b,a,POS1_1);
POS2_1_filt = filtfilt(b,a,POS2_1);
POS3_1_filt = filtfilt(b,a,POS3_1);
POS4_1_filt = filtfilt(b,a,POS4_1);

POS1_2_filt = filtfilt(b,a,POS1_2);
POS2_2_filt = filtfilt(b,a,POS2_2);
POS3_2_filt = filtfilt(b,a,POS3_2);
POS4_2_filt = filtfilt(b,a,POS4_2);

POS1_3_filt = filtfilt(b,a,POS1_3);
POS2_3_filt = filtfilt(b,a,POS2_3);
POS3_3_filt = filtfilt(b,a,POS3_3);
POS4_3_filt = filtfilt(b,a,POS4_3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 2. Create reverberation time vectors (with help function
% reverberation.m)

[POS1_0_rev] = reverberation(POS1_0_filt);
[POS2_0_rev] = reverberation(POS2_0_filt);
[POS3_0_rev] = reverberation(POS3_0_filt);
[POS4_0_rev] = reverberation(POS4_0_filt);

[POS1_1_rev] = reverberation(POS1_1_filt);
[POS2_1_rev] = reverberation(POS2_1_filt);
```

```
[POS3_1_rev] = reverberation(POS3_1_filt);
[POS4_1_rev] = reverberation(POS4_1_filt);

[POS1_2_rev] = reverberation(POS1_2_filt);
[POS2_2_rev] = reverberation(POS2_2_filt);
[POS3_2_rev] = reverberation(POS3_2_filt);
[POS4_2_rev] = reverberation(POS4_2_filt);

[POS1_3_rev] = reverberation(POS1_3_filt);
[POS2_3_rev] = reverberation(POS2_3_filt);
[POS3_3_rev] = reverberation(POS3_3_filt);
[POS4_3_rev] = reverberation(POS4_3_filt);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 3. Normalize and convert to dB

POS1_0_norm = 10*log10(POS1_0_rev/POS1_0_rev(1));
POS2_0_norm = 10*log10(POS2_0_rev/POS2_0_rev(1));
POS3_0_norm = 10*log10(POS3_0_rev/POS3_0_rev(1));
POS4_0_norm = 10*log10(POS4_0_rev/POS4_0_rev(1));

POS1_1_norm = 10*log10(POS1_1_rev/POS1_1_rev(1));
POS2_1_norm = 10*log10(POS2_1_rev/POS2_1_rev(1));
POS3_1_norm = 10*log10(POS3_1_rev/POS3_1_rev(1));
POS4_1_norm = 10*log10(POS4_1_rev/POS4_1_rev(1));

POS1_2_norm = 10*log10(POS1_2_rev/POS1_2_rev(1));
POS2_2_norm = 10*log10(POS2_2_rev/POS2_2_rev(1));
POS3_2_norm = 10*log10(POS3_2_rev/POS3_2_rev(1));
POS4_2_norm = 10*log10(POS4_2_rev/POS4_2_rev(1));

POS1_3_norm = 10*log10(POS1_3_rev/POS1_3_rev(1));
POS2_3_norm = 10*log10(POS2_3_rev/POS2_3_rev(1));
POS3_3_norm = 10*log10(POS3_3_rev/POS3_3_rev(1));
POS4_3_norm = 10*log10(POS4_3_rev/POS4_3_rev(1));


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 4. Least square line fitting with help function linemaker.m

start_value = 0;
stop_value = -10;


[x1_0, y1_0] = linemaker(POS1_0_norm, start_value, stop_value);
[x2_0, y2_0] = linemaker(POS2_0_norm, start_value, stop_value);
[x3_0, y3_0] = linemaker(POS3_0_norm, start_value, stop_value);
[x4_0, y4_0] = linemaker(POS4_0_norm, start_value, stop_value);

[x1_1, y1_1] = linemaker(POS1_1_norm, start_value, stop_value);
[x2_1, y2_1] = linemaker(POS2_1_norm, start_value, stop_value);
[x3_1, y3_1] = linemaker(POS3_1_norm, start_value, stop_value);
[x4_1, y4_1] = linemaker(POS4_1_norm, start_value, stop_value);
```

```
[x1_2, y1_2] = linemaker(POS1_2_norm, start_value, stop_value);
[x2_2, y2_2] = linemaker(POS2_2_norm, start_value, stop_value);
[x3_2, y3_2] = linemaker(POS3_2_norm, start_value, stop_value);
[x4_2, y4_2] = linemaker(POS4_2_norm, start_value, stop_value);

[x1_3, y1_3] = linemaker(POS1_3_norm, start_value, stop_value);
[x2_3, y2_3] = linemaker(POS2_3_norm, start_value, stop_value);
[x3_3, y3_3] = linemaker(POS3_3_norm, start_value, stop_value);
[x4_3, y4_3] = linemaker(POS4_3_norm, start_value, stop_value);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 5. Control plots (it is possible to create 12 plots, but only one
% is issued for demonstration purpose in order to save space)

a0 = 0;
a1 = round(5000/mlsfs*100)/100;
a2 = round(10000/mlsfs*100)/100;
a3 = round(15000/mlsfs*100)/100;
a4 = round(20000/mlsfs*100)/100;
a5 = round(25000/mlsfs*100)/100;
a6 = round(30000/mlsfs*100)/100;
a7 = round(35000/mlsfs*100)/100;


figure(1)
plot(POS1_0_norm,'b')
hold on;
plot(POS1_1_norm,'r')
plot(x1_0, y1_0,'b')
plot(x1_1, y1_1,'r')
hold off;
xlim([0 35000])
ylim([-30 0])
set(gca,'XTickLabel',{a0;a1;a2;a3;a4;a5;a6;a7})
title('RT 125 Hz, microphone position 1, absorber position 1')
xlabel('time [s]')
ylabel('magnitude [dB]')


figure(2)… etc.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 6. Calculate T60 (uses help program deltatime.m)

T60(1,1) = deltatime(y1_0, mlsfs);
T60(2,1) = deltatime(y2_0, mlsfs);
T60(3,1) = deltatime(y3_0, mlsfs);
T60(4,1) = deltatime(y4_0, mlsfs);

T60(1,2) = deltatime(y1_1, mlsfs);
T60(2,2) = deltatime(y2_1, mlsfs);
T60(3,2) = deltatime(y3_1, mlsfs);
```

```
T60(4,2) = deltatime(y4_1, mlsfs);

T60(1,3) = deltatime(y1_2, mlsfs);
T60(2,3) = deltatime(y2_2, mlsfs);
T60(3,3) = deltatime(y3_2, mlsfs);
T60(4,3) = deltatime(y4_2, mlsfs);

T60(1,4) = deltatime(y1_3, mlsfs);
T60(2,4) = deltatime(y2_3, mlsfs);
T60(3,4) = deltatime(y3_3, mlsfs);
T60(4,4) = deltatime(y4_3, mlsfs);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 7. Calculate deltaT60

deltaT60(1,1) = T60(1,1) - T60(1,2);
deltaT60(2,1) = T60(2,1) - T60(2,2);
deltaT60(3,1) = T60(3,1) - T60(3,2);
deltaT60(4,1) = T60(4,1) - T60(4,2);


deltaT60(1,2) = T60(1,1) - T60(1,3);
deltaT60(2,2) = T60(2,1) - T60(2,3);
deltaT60(3,2) = T60(3,1) - T60(3,3);
deltaT60(4,2) = T60(4,1) - T60(4,3);


deltaT60(1,3) = T60(1,1) - T60(1,4);
deltaT60(2,3) = T60(2,1) - T60(2,4);
deltaT60(3,3) = T60(3,1) - T60(3,4);
deltaT60(4,3) = T60(4,1) - T60(4,4);
```

```
% This help function is repeatedly called in protytype2.m and
% calculates the filter coefficients for a n-order butterworth
% band pass filter for given octave bands.

function [b,a] = butterfilter(CF,mlsfs,n);

% CF is centre frequency of the band, mlsfs is the MLSSA sample rate
and n is the desired order of the filter.

N = n/2;
F1 = CF/sqrt(2);
F2 = CF*sqrt(2);
W1 = F1*2/mlsfs;
W2 = F2*2/mlsfs;
Wn = [W1 W2];  % The cut off frequency Wn must be 0.0 < Wn < 1.0,
               % with 1.0 corresponding to half the sample rate.

[b,a] = butter(N,Wn);
```

```
% The function takes a vector, reverses it and sum up the squared
% elements and reverses it again, according to the Schroeder backward
% integration method. The  program is frequently called from
% protoype2.m.

function [reverberation_time_vector] = reverberation(RIR)

n = length(RIR);

for i = 1:length(RIR);
  reverse(n) = RIR(i);
   n = n-1;
end

sum = 0;

for j = 1:length(reverse);
   sum = sum + reverse(j)*reverse(j);
   squaresum(j) = sum;
end

l = length(squaresum);

for k = 1:length(squaresum);
  reverberation_time_vector(l) = squaresum(k);
  l = l-1;
end
```

```
% This frequently called subroutine to prototype2.m performs least
% squares line fitting with given start and stop points expressed in
% dB. It returns the x and y coordinates for the line. The input
% function vectors need to be normalized.

function [x, y] = linemaker(normfunction, start_value, stop_value)

[Y1, n] = min(abs(normfunction-(start_value)));
[Y2, p] = min(abs(normfunction-(stop_value)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A

A = 0;
for i = n:p
  A = A + (-i^2);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% B

B = 0;
for i = n:p
  B = B + (-i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% C

C = 0;
for i = n:p
  C = C + (i)*normfunction(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% D

D = B;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% E

E = 0;
for i = n:p
  E = E + (-1);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% F

F = 0;
for i = n:p
  F = F + normfunction(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

k = -1/(A*E - D*B)*(E*C - B*F);
m = -1/(A*E - D*B)*(-D*C + A*F);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x = [1:1:length(normfunction)];
y = k*x + m;
```

```
% This is a help function used in prototype2.m which computes the T60
% reverberation time given a straight descending line.

function [T60] = deltatime(REVvector, mlsfs)

% Calculating the T60 reverberation time

[Y1,I1] = min(abs(REVvector-(-5)));
[Y2,I2] = min(abs(REVvector-(-15)));
T60 = (I2-I1)*6/mlsfs;
```

```
% This is a help function that is called numerous times from
% prototype.m in order to contribute with important statistics, as
% averages and standard deviations.

function [dT60_average, sigma, T60_average_noabs, T60_average_abs] =
statistics(deltaT60, T60)

dT60 = sum(sum(deltaT60))/12;

for j = 1:3
  for i = 1:4
     res(i,j) = (deltaT60(i,j) - dT60).^2;
   end
end

T60_sum_noabs = 0;

for k = 1:4
  T60_sum_noabs = T60_sum_noabs + T60(k,1);
end

T60_sum_abs = 0;

for l = 2:4
  for m = 1:4
    T60_sum_abs = T60_sum_abs + T60(l,m);
  end
end

T60_average_noabs = T60_sum_noabs/4;
T60_average_abs = T60_sum_abs/12;
sigma = sqrt(sum(sum(res))/12);
dT60_average = dT60;
```

```
% This function is a help program to prototype.m and calculates
% absorption coefficients from reverberation times by comparing them
% with and without absorber. It also calculates standard deviations
% as well as averages.

function [alfa, alfa_average, sigma] = absorption(T60,V,S)

T60_noabs(1) = T60(1,1);
T60_noabs(2) = T60(2,1);
T60_noabs(3) = T60(3,1);
T60_noabs(4) = T60(4,1);

n = 1;

for j = 2:4;
  for i = 1:4;
    alfa(n) = 0.163*V/S*(1/T60(i,j) - 1/T60_noabs(i));
    n = n+1;
  end
end

alfa_average = sum(alfa)/12;

for k = 1:12
  alfa_res(k) = (alfa(k)-alfa_average).^2;
end

sigma = sqrt(sum(alfa_res)/12);
```

```
% This matlab program draws a detailed graph of absorption
% coefficient as function of frequency from measured impulse
% responses. This is done by backward integration and least squares
% line fitting and comparison between reverberation times with and
% without absorber.

function a = alfaplot

[sub_pos1_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SUB.TIM');
[sub_pos1_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABS.TIM');
[sub_pos1_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABT.TIM');
[sub_pos1_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS1SABU.TIM');

[sub_pos2_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SUB.TIM');
[sub_pos2_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABS.TIM');
[sub_pos2_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABT.TIM');
[sub_pos2_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS2SABU.TIM');

[sub_pos3_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SUB.TIM');
[sub_pos3_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABS.TIM');
[sub_pos3_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABT.TIM');
[sub_pos3_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS3SABU.TIM');

[sub_pos4_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SUB.TIM');
[sub_pos4_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABS.TIM');
[sub_pos4_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABT.TIM');
[sub_pos4_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS4SABU.TIM');

[top_pos1_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TOP.TIM');
[top_pos1_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABS.TIM');
[top_pos1_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABT.TIM');
[top_pos1_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS1TABU.TIM');

[top_pos2_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TOP.TIM');
[top_pos2_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABS.TIM');
[top_pos2_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABT.TIM');
[top_pos2_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS2TABU.TIM');

[top_pos3_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TOP.TIM');
[top_pos3_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABS.TIM');
[top_pos3_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABT.TIM');
[top_pos3_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS3TABU.TIM');

[top_pos4_noabs,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TOP.TIM');
[top_pos4_abs1,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABS.TIM');
[top_pos4_abs2,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABT.TIM');
[top_pos4_abs3,mlsfs,stimulus_amp,mlsdf] = readmls('POS4TABU.TIM');
```

```
%f= [100:50:1000];
% The frequency vector should be kept as short as possible to reduce
% computation time. It is a trade off between time and resolution.
f= [63 70 88 110 125 176 250 353 500 707 1000 1414 2000 2828 4000];

% The program should use top speaker measurements above 250 and
% subwoofer measurements below. The index breakpoint is set manually,
% counting from left in the frequency vector (250 is the 7:th
% element).
breakpoint = 7;

% The calculations uses a subroutine

[vec1] = alfaplotvector(sub_pos1_noabs, sub_pos1_abs1,
top_pos1_noabs, top_pos1_abs1, f, breakpoint, mlsfs)
[vec2] = alfaplotvector(sub_pos1_noabs, sub_pos1_abs2,
top_pos1_noabs, top_pos1_abs2, f, breakpoint, mlsfs)
[vec3] = alfaplotvector(sub_pos1_noabs, sub_pos1_abs3,
top_pos1_noabs, top_pos1_abs3, f, breakpoint, mlsfs)
[vec4] = alfaplotvector(sub_pos2_noabs, sub_pos2_abs1,
top_pos2_noabs, top_pos2_abs1, f, breakpoint, mlsfs)
[vec5] = alfaplotvector(sub_pos2_noabs, sub_pos2_abs2,
top_pos2_noabs, top_pos2_abs2, f, breakpoint, mlsfs)
[vec6] = alfaplotvector(sub_pos2_noabs, sub_pos2_abs3,
top_pos2_noabs, top_pos2_abs3, f, breakpoint, mlsfs)
[vec7] = alfaplotvector(sub_pos3_noabs, sub_pos3_abs1,
top_pos3_noabs, top_pos3_abs1, f, breakpoint, mlsfs)
[vec8] = alfaplotvector(sub_pos3_noabs, sub_pos3_abs2,
top_pos3_noabs, top_pos3_abs2, f, breakpoint, mlsfs)
[vec9] = alfaplotvector(sub_pos3_noabs, sub_pos3_abs3,
top_pos3_noabs, top_pos3_abs3, f, breakpoint, mlsfs)
[vec10] = alfaplotvector(sub_pos4_noabs, sub_pos4_abs1,
top_pos4_noabs, top_pos4_abs1, f, breakpoint, mlsfs)
[vec11] = alfaplotvector(sub_pos4_noabs, sub_pos4_abs2,
top_pos4_noabs, top_pos4_abs2, f, breakpoint, mlsfs)
[vec12] = alfaplotvector(sub_pos4_noabs, sub_pos4_abs3,
top_pos4_noabs, top_pos4_abs3, f, breakpoint, mlsfs)

alfa_matrix(1,:) = vec1;
alfa_matrix(2,:) = vec2;
alfa_matrix(3,:) = vec3;
alfa_matrix(4,:) = vec4;
alfa_matrix(5,:) = vec5;
alfa_matrix(6,:) = vec6;
alfa_matrix(7,:) = vec7;
alfa_matrix(8,:) = vec8;
alfa_matrix(9,:) = vec9;
alfa_matrix(10,:) = vec10;
alfa_matrix(11,:) = vec11;
alfa_matrix(12,:) = vec12;

for i = 1:length(vec1)
```

```
  vecsum = 0;
  for j = 1:12
     vecsum = vecsum + alfa_matrix(j, i);
  end
  alfa_matrix(13,i) = vecsum/12;
end

figure(1)
semilogx(f, alfa_matrix(13,:))
set(gca,'XTickLabel',{a0;a1;a2;a3;a4})
title('Measured absorption coefficient')
xlabel('frequency [Hz]')
ylabel('absorption coefficient')
grid on;
```

```
% A help function for alfaplot.m which calculates the absorption
% coefficients and returns them in a vector. This is a somewhat time
% consuming algorithm. The program uses a set of subfunctions.

function [alfa_vector] = alfaplotvector(unfiltvecsub_noabs,
unfiltvecsub_abs, unfiltvectop_noabs, unfiltvectop_abs, f,
breakpoint, mlsfs)

for i = 1:breakpoint
  [b,a] = butterfilter(f(i), mlsfs, 2);
  filtvec_noabs = filtfilt(b,a,unfiltvecsub_noabs);
  filtvec_abs = filtfilt(b,a,unfiltvecsub_abs);
  revvec_noabs = reverberation(filtvec_noabs);
  revvec_abs = reverberation(filtvec_abs);
  normvec_noabs = 10*log10(revvec_noabs/revvec_noabs(1));
  normvec_abs = 10*log10(revvec_abs/revvec_abs(1));
  [x1, y1] = linemaker(normvec_noabs,0,-10);
  [x2, y2] = linemaker(normvec_abs,0,-10);
  T60_noabs = deltatime(y1, mlsfs);
  T60_abs = deltatime(y2, mlsfs);
  alfa = 0.163*96.6/3*(1/T60_abs - 1/T60_noabs);
  alfa_vector(i) = alfa;
end

for i = breakpoint+1:length(f)
  [b,a] = butterfilter(f(i), mlsfs, 2);
  filtvec_noabs = filtfilt(b,a,unfiltvectop_noabs);
  filtvec_abs = filtfilt(b,a,unfiltvectop_abs);
  revvec_noabs = reverberation(filtvec_noabs);
  revvec_abs = reverberation(filtvec_abs);
  normvec_noabs = 10*log10(revvec_noabs/revvec_noabs(1));
  normvec_abs = 10*log10(revvec_abs/revvec_abs(1));
  [x1, y1] = linemaker(normvec_noabs,0,-10);
  [x2, y2] = linemaker(normvec_abs,0,-10);
  T60_noabs = deltatime(y1, mlsfs);
  T60_abs = deltatime(y2, mlsfs);
  alfa = 0.163*96.6/3*(1/T60_abs - 1/T60_noabs);
  alfa_vector(i) = alfa;
end
```

Bibliography

[1] Rife, Douglas D. *MLSSA Reference Manual Version 10WI-4,* p. 161 (2001)

[2] Rife, Douglas D. *MLSSA Reference Manual Version 10WI-4,* pp. 162-164 (2001)

[3] Kuttruff, Heinrich. *Room Acoustics. Third Edition*. Elsevier Science Publishers LTD, pp. 56-58 (1991)

[4] Beranek, Leo L. *Acoustics*. Acoustical Society of America., chapter 4 (1996)

[5] Schroeder, M.R. *New Method of Measuring Reverberation Time*, Journal of the Acoustical Society of America, vol. 37, no. 3, pp. 409-412 (1965)

[6] Kuttruff, Heinrich. *Room Acoustics. Fourth Edition*. Elsevier Science Publishers LTD, chapters 3.5, 5.1, 8.5-8.7 (2004)

[7] Andersson, K. and Nyström, Å. *Optimization of a Slitted Absorber in the Low Frequency Range,* Chalmers University of Technology, pp. 5 – 6, 8 (1989)

[8] Andersson, K. and Nyström, Å. *Optimization of a Slitted Absorber in the Low Frequency Range,* Chalmers University of Technology, p. 7 (1989)

[9] Vigran, Tor Erik. *Bygningsakustikk*. Tapir Akademisk Forlag, p. 166 (2002)