

**HOCHSCHULE
MITTWEIDA**
UNIVERSITY OF
APPLIED SCIENCES



HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN
MÜNCHEN

Ermittlung der Raumimpulsantwort eines Raumes mit Hilfe der Kreuzkorrelation

im Studiengang Ingenieurakustik der
Hochschule Mittweida & Hochschule München

**Belegarbeit im Modul:
akustische Messtechnik und Signalverarbeitung**

vorgelegt von

Daniel Steinshorn
geboren am 24.11.1995 in Würzburg

Abgabedatum am 21. Juli 2020

Professor: Prof. Dr-Ing. Jörn Hübelt

Inhaltsverzeichnis

Aufgabenstellung	iii
Abkürzungsverzeichnis	vi
Symbolverzeichnis	vii
1 Einleitung	1
2 Grundlagen	2
2.1 Impulsantwort	2
2.2 Auto- und Kreuzkorrelation	3
2.3 Nachhallzeit	4
2.4 Schröder-Rückwärtsintegration [Sch65]	5
2.4.1 Überlegung zur Rückwärtsintegration ($ETC \rightarrow EDC$)	5
2.5 lineare Regression / Regressionsmodell	7
2.6 Zeitsynchrone Mittelung	8
3 Messung und Test des Skriptes	9
3.1 Messdurchführung und Equipment	9
3.2 Untersuchung zum Zeitversatz der Soundkarte	10
3.3 Entwicklung des Skriptes	11
3.3.1 Parametereingabe	11
3.3.2 Messung und Datenspeicherung	12
3.3.3 Kreuzkorrelation und Impulsantwort	15
3.3.4 ETC und Integrationsbeginn ermitteln	16
3.3.5 Rückwärtsintegration	17
3.3.6 lineare Regression und Nachhallzeitbestimmung	17
3.3.7 Frequenzspektrum der Impulsantwort	19
3.3.8 <i>Room EQ Wizard</i>	20
3.4 Diskussion der Ergebnisse	21
4 Zusammenfassung und Ausblick	23
A Anhang	24
A.1 vollständiges Skript	24
A.2 Room EQ Wizard	27
A.3 Latenz der Soundkarte bestimmen	28
A.4 exponentieller Sweep erstellen	29
A.5 Messanordnung	30
A.6 zufällige Artefakte	30
Abbildungsverzeichnis	32
Literaturverzeichnis	33

Aufgabenstellung der Belegarbeit

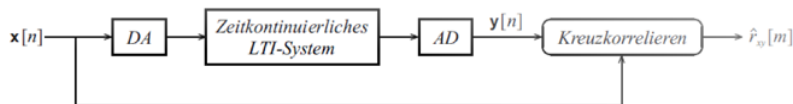
1. Termine/Vorlagen/Umfang

Abgabe: 30.9.20
Umfang: ca. 20 Seiten ohne Anhang
Vorlage für Dokument: <https://www.me.hs-mittweida.de/studienangebote/informationen-fuer-studierende-der-master-studiengaenge/ingenieurakustik/masterarbeit.html>
und Hinweise zur formalen Gestaltung: <https://www.me.hs-mittweida.de/webs/huebelt/abschlussarbeiten-praktika/formale-gestaltung-der-berichte.html?Size=xantqmhovxza%3Fprint%3D1%3Fprint%3D1>

2. Aufgabenstellung

2.1. Impulsantwort

Bestimmen Sie die Impulsantwort eines Raumes mit Hilfe der Kreuzkorrelation zwischen Empfangs- (Mikrofon) $y[n]$ und Sendesignal (Lautsprecher) $x[n]$.



Hinweis: Herkömmliche Soundkarten weisen einen nicht reproduzierbaren zeitlichen Versatz zwischen Eingang und Ausgang auf. Wenn Sie als Eingang an der Soundkarte z.B. den rechten Stereokanal gewählt haben, so können Sie die zeitliche Synchronisierung zwischen Eingang und Ausgang durch Rückführung des Ausgangssignals in den linken Stereokanal realisieren. Die angestrebte Korrelation findet dann zwischen dem rechten und linken Kanal statt.

Anforderung an das Matlab-Skript

- Länge des Zeitfensters einstellbar
- wenn möglich: Abtastfrequenz einstellbar
- Anzahl der Mittlungen einstellbar, dazu Overlap einstellbar
- 1. Plot: Kreuzkorrelationsfunktion
- 2. Plot: Impulsantwort

2.2. Frequenzspektrum

Stellen Sie die in Aufgabe 1.1 gemessene Impulsantwort spektral als Funktion der Zeit („Campell-Diagramm“, „Waterfall plot“) dar. Wählen Sie dazu eine sinnvolle Zeit- und Frequenzauflösung.

Anforderung an das Matlab-Skript

- f_o , f_u und Δf einstellbar
- 1. Plot: Betragsspektrum mit physikalisch richtigen Werten
- 2. Plot: Waterfall plot

2.3. Nachhallzeit

Berechnen Sie aus der in Aufgabe 1.1 gemessene Impulsantwort die Nachhallzeit durch Anwendung der „Schröder Rückwärtsintegration“ [Hübel 2018].

Anforderung an das Matlab-Skript

- Berechnung „Schröder Rückwärtsintegration“ aus der Impulsantwort
- Auswahl des für die Nachhallzeit sinnvollen Zeitbereich für den Pegel-Zeit-Verlauf (Pegelabfall, z.B. 30 dB)

- Lineare Regression des Pegel-Zeit-Verlaufs (sinnvollen Zeitbereich) für die Berechnung der Nachhallzeit
- Bestimmung der Nachhallzeit
- 1. Plot: Ergebnis „Schröder Rückwärtsintegration“
- 2. Plot: Pegel-Zeit-Verlaufs (sinnvollen Zeitbereich)
- Ergebnis Nachhallzeit

3. Inhalt Bericht

An dieser Stelle steht ein Vorschlag für eine Gliederung des Berichts.

Alle Darstellungen und Ergebnisse sollen kurz beschrieben und anschließend diskutiert werden (z.B. Ist das Ergebnis plausibel? Auf welche Zusammenhänge sind die Abweichungen zwischen Messergebnis und Theorie zurückzuführen? Beschreibung der Messtechnik. ...)

3.1. Theorie

- Def. Nachhallzeit
- Kreuzkorrelation [Grünigen 2014]
- Fouriertransformation
- „Schröder Rückwärtsintegration“ [Hübelt 2018]
- Lineare Regression

3.2. Bestimmung der Impulsantwort durch Korrelation

- Plot der Kreuzkorrelation
- Plot der Impulsantwort

3.3. Spektrale Darstellung der Impulsantwort

- Frequenzspektrum
- Waterfall Plot

3.4. Nachhallzeit

- Messung in einem Raum
 - Plot Impulsantwort
 - Plot: Ergebnis „Schröder Rückwärtsintegration“
 - Lineare Regression des Pegel-Zeit-Verlaufs (sinnvollen Zeitbereich) für die Berechnung der Nachhallzeit
 - Bestimmung der Nachhallzeit
- Diskussion der Ergebnisse für die Nachhallzeit, Abschätzung mit der Nachhallzeitgleichung nach Sabine [Fasold 2003]

3.5. Zusammenfassung

4. Literatur

- [Grünigen 2014] Daniel von Grünigen: Digitale Signalverarbeitung: mit einer Einführung in die kontinuierlichen Signale und Systeme. Carl Hanser Verlag GmbH & Co. KG; Auflage: 5 (3. Juli 2014)
- [Fasold 2003] Fasold, W.: Schallschutz und Raumakustik in der Praxis: Planungsbeispiele und konstruktive Lösungen. Huss-Medien, 2. Auflage, bearb. Aufl., 2003
- [Hübelt 2018] Skript „Bau- und Raumakustik, Architectural Acoustics“, Hochschule Mittweida, 2018 <https://www.me.hs-mittweida.de/webs/huebelt/bachelor-media-and-acoustical-engineering/architectural-acoustics.html>

Viel Erfolg

Jörn Hübelt

Eigenständigkeitserklärung

Ich erkläre hiermit, die vorliegende Arbeit selbständig verfasst zu haben. Es wurden keine unerlaubten Hilfsmittel und nur die im Literaturverzeichnis angegebenen Quellen verwendet. Ich versichere, keine unzulässige fremde Hilfe in Anspruch genommen zu haben.

Wittighausen, den

Abkürzungsverzeichnis

RMS	quadratischer Mittelwert (<i>engl. root-mean-square</i>)
LZI	Linear-Zeitinvariant (<i>linear-time-invariant</i>)
SNR	Signal-Rausch-Abstand (<i>engl. signal-noise-ratio</i>)
AKF	Autokorrelationsfunktion r_{xx}
KKF	Kreuzkorrelationsfunktion r_{xy}
ETC	(<i>engl. energy-time-curve</i>)
EDC	(<i>engl. energy-decay-curve</i>)
REW	<i>Room-EQ-Wizard</i> → kostenlose Messsoftware für Raumakustik
XLR	symmetrische Verbindungsstecker von Mikrofonen und Lautsprechern

Symbolverzeichnis

Allgemeine Symbole

$x(\bullet)$	kontinuierliche Werte
$x[\bullet]$	diskrete Werte
$\delta[n]$	Dirac-Impuls
σ_{xy}	Kovarianz
σ_x^2	Varianz
f_G	Grenzfrequenz eines Filter
$H(z)$	Impulsantwort im Bildbereich
$h[n] = \{x_1, x_2, x_3, \dots\}$	diskrete Wertreihenfolge
$h[n]$	Impulsantwort im Zeitbereich (diskret)
r_{xx}	Autokorrelationsfunktion
S_{xx}	Leistungsdichtespektrum

Konventionen

j	imaginäre Einheit
$z = e^{j\Omega}$	Drehung auf dem Einheitskreis (z-Transformation)

Raumakustische Größen

α	Absorptionsgrad
ΔL	Pegeldifferenz
Ω	normierte Kreisfrequenz $\Omega = 2\pi \frac{f}{f_s}$
ω_d	gedämpfte Kreisfrequenz $\omega = 2\pi f$
A_W	äquivalente Absorptionsfläche
f	Frequenz
f_s	Abtastfrequenz
L_{-5dB}	Messwertindex, an der die EDC um 5dB abgenommen hat

L_{10dB}	Messwertindex, an der die EDC 10dB über dem Umgebungsgeräusch liegt
S	Wandfläche
$s^2(t)$	Schröder-Rückwärtsintegrierte Funktion (EDC)
t	Zeit
T_E	Nachhallzeit nach Eyring
T_S	Nachhallzeit nach Sabine
T_{60}	Nachhallzeit bei 60 dB Pegelabfall
V	Raumvolumen des Messraum

Operatoren

$\bar{\bullet}$	arithmetischer Mittelwert
$\bullet * \bullet$	Faltungsoperator
\bullet^*	konjugiert komplex
$\hat{\bullet}$	Schätzwert einer Variable
$\circ \text{---} \bullet$	Laplace Transformations-Symbol
$\underline{\bullet}$	komplexe Zahl
$E(\bullet)$	Erwartungswert einer Variablen

1 Einleitung

Kann jede interessierte Person mit einfachen oder kostengünstigen Mitteln und einer Matlab-Lizenz, raumakustische Untersuchungen durchführen?

Jeder Mensch verbringt einen Großteil seiner Zeit in Räumen und ist somit der schlechten Raumakustik und Sprachverständlichkeit ausgesetzt. Die zunehmende *do-it-yourself* Mentalität der Menschen rückt dabei in den Vordergrund und wirft die Frage auf, ob eine kostengünstige Untersuchung in Eigenregie möglich ist.

Die vorliegende Belegarbeit setzt sich mit der Erstellung eines Matlab-Skriptes unter Verwendung von einfacher Audiotechnik und Hardware auseinander. Grundlage der Arbeit stellt hierbei die Ermittlung der Raumimpulsantwort mit Hilfe der Kreuzkorrelationsmethode dar. Weiterführend wird die Umsetzung des Frequenzspektrum, sowie die Nachhallzeitbestimmung durch die Schröder-Rückwärtsintegration in Matlab untersucht.

Das Ziel dieser Arbeit ist die Erstellung eines funktionierenden (idealerweise automatisierten) Matlab-Skriptes zur Ermittlung von einfachen raumakustischen Kenngrößen wie der Nachhallzeit. Augenmerk liegt hier auf der Glaubwürdigkeit der Daten infolge der minderwertigen Messkette und ihren Einflussfaktoren.

Zunächst werden in [Kapitel 2](#) die theoretischen Grundlagen und ihr Zusammenhang zu dieser Arbeit dargestellt und erläutert. In [Kapitel 3](#) wird die verwendete Messtechnik, der Messaufbau sowie die Skripterstellung in chronologischer Reihenfolge anhand der ermittelten Daten dargestellt und erklärt. Anschließend erfolgt die [Diskussion der Ergebnisse](#) sowie deren Zuverlässigkeit. Die Arbeit endet im [Kapitel 4](#) mit der Zusammenfassung und einem möglichen weiteren Vorgehen dieser Untersuchung. Im [Anhang](#) sind zusätzliche Informationen, Skripte und Grafiken abgelegt.

2 Grundlagen

2.1 Impulsantwort

Die Impulsantwort ist eine klassische Beschreibungsmöglichkeit von linear-zeitinvarianten Systemen (LZI-System). Dabei wird ein idealer Dirac-Impuls $\delta[n]$ als Eingangssignal $x[n]$ für das System verwendet und die Ausgangssignal des Systems $y[n]$ aufgezeichnet.

Da nach der Überführung in den Bildbereich der ideale Impuls alle Frequenzen gleichphasig mit der Amplitude 1 anregt, ergibt sich somit direkt aus der Systemantwort eines Impulses, das Systemverhalten $h[n]$ in Form der Übertragungsfunktion. [Grü14, Kapitel 4.2.2] Somit enthält das Ausgangssignal das unverfälschte Systemverhalten $Y(z) = H(z) \cdot 1$.

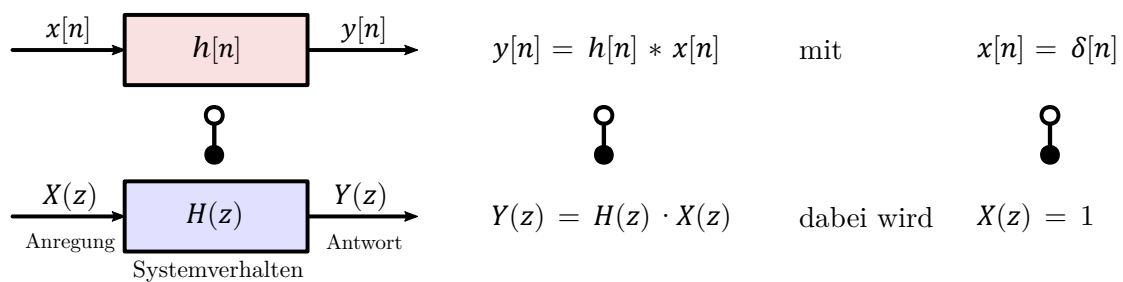


Abb. 2.1: Impulsantwort im diskreten Zeit- und Bildbereich [Grü14, Bild 4.12]

In der Praxis ist die Erzeugung eines idealen Einheitsimpuls in der Audiotechnik leider Wunschenken. Aus diesem Grund muss das Verhalten für nicht ideale Signale betrachtet werden. Hierfür wird in [Abbildung 2.2](#) der Amplitudengang von Systemen mit ansteigender Impulsbreite dargestellt.

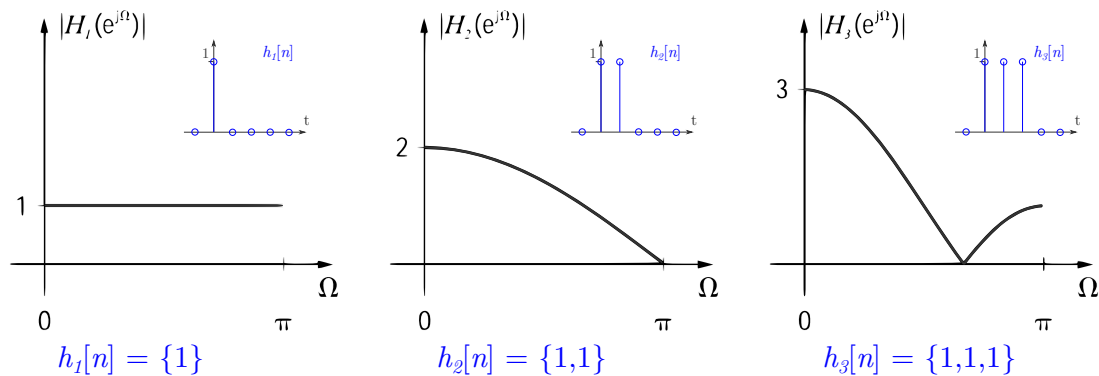


Abb. 2.2: Amplitudengang bei Abweichung vom Einheitsimpuls [Grü14, Bild 4.13]

Wie sich zeigt, haben schon kleinste Veränderungen am Eingangsimpuls eine starke Filterwirkung auf die Systemantwort. Aufgrund dieser Tatsache ist die Bestimmung des Raumverhaltens über eine Impulserregung (Luftballon) bestenfalls für eine grobe Abschätzung geeignet. Zur messtechnischen Erfassung der Impulsantwort wird häufig ein breitbandiges Rauschen oder ein Sweep und das Verfahren der Kreuzkorrelation verwendet.

2.2 Auto- und Kreuzkorrelation

Die Korrelation quantifiziert die *lineare Abhängigkeit zwischen zwei Variablen* und ist somit ein Maß für den Zusammenhang von Signalen. Dabei beschreibt die Autokorrelationsfunktion (AKF) r_{xx} die Ähnlichkeit eines Signals zu sich selbst und die Kreuzkorrelation (KKF) r_{xy} die Ähnlichkeit zu unterschiedlichen Signalen. Die Korrelationsberechnung ist dabei sehr nahe mit der Faltungsoperation verwandt.

$$r_{xy}(\tau) = \lim_{T \rightarrow \infty} \int_{-T}^T x(t) \cdot y(t + \tau) d\tau \quad (2.1)$$

Die Korrelationsfunktion wird durch die Bildung des eingeschlossenen Flächenintegrals der beiden Funktionen, bei variierender Zeitverschiebung τ ermittelt. Das Integral ist somit groß, wenn große Ähnlichkeit zwischen den Signalen besteht.

Für die AKF stellt das weiße Rauschen (*engl. white noise*) eine der wichtigsten stochastischen Signale dar. Durch das unkorrelierte Verhalten des weißen Rauschen, lässt sich die AKF mit einem Dirac-Impuls der Zeitverschiebung $\tau = 0$ ausdrücken. Das Rauschen ist sich somit nur zum gleichen Zeitpunkt ähnlich. Dieses Verhalten trifft im weitesten Sinne ebenfalls auf Multisinus und Sweep zu.

$$r_{xx}[n] = \sigma_x^2 \cdot \delta[n] \quad (2.2)$$

Die Variable σ_x^2 beschreibt die Varianz des Rauschens und entspricht somit gleich der mittleren Leistung des Signals. Aus der Standardabweichung des weißen Rauschens $\sqrt{\sigma_x^2}$ ergibt sich somit der Effektivwert (RMS) des Rauschsignals. Für die AKF mit diskreten Werten ergibt sich außerdem eine Fourier-Transformationspaar mit dem Leistungsdichtespektrum (*engl. power-spectral-density*) $S_{xx}(z)$.

$$r_{xx}[n] \circ \bullet S_{xx}(z) \rightarrow r_{xx}[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_{xx}(e^{j\Omega}) * e^{j\Omega n} d\Omega \quad (2.3)$$

Dabei beschreibt $z = e^{j\Omega}$ die Auswertung auf dem Einheitskreis und $\Omega = 2\pi f/f_s$ die normierte Kreisfrequenz.

Wie wird mit der Kreuzkorrelation die Raumimpulsantwort bestimmt?

Wird der Raum mit einem Signal $\underline{X}(z)$ angeregt, verändert sich das Signal durch die Raumimpulsantwort/Übertragungsfunktion $\underline{H}(z)$ und die Mikrofone messen das überlagerte Signal von Impulsantwort und Anregung $\underline{Y}(z)$.

$$\underline{Y}(z) = \underline{H}(z) \cdot \underline{X}(z) \quad (2.4)$$

Durch die Multiplikation des konjugiert komplexen Eingangssignals $\underline{X}^*(z)$ ergibt sich zum Einen das reale Leistungsdichtespektrum des Eingangssignals $S_{xx}(z)$ und zum Anderen das komplexe Kreuzleistungsdichtespektrum $S_{xy}(z)$ nach [Gleichung \(2.5b\)](#).

$$\underline{Y}(z) \cdot \underline{X}^*(z) = \underline{H}(z) \cdot \underline{X}(z) \cdot \underline{X}^*(z) \quad (2.5a)$$

$$\underline{S}_{xy}(z) = \underline{H}(z) \cdot S_{xx}(z) \quad (2.5b)$$

Für den Spezialfall von weißem Rauschen ist somit das Kreuzleistungsdichtespektrum proportional zum Frequenzgang, skaliert mit der mittleren Leistung des Eingangssignals. Durch die Rücktransformation in den Zeitbereich ist der Zusammenhang der KKF mit der Raumimpulsantwort nach Gleichung (2.6) zu erkennen.

$$\underline{S}_{xy}(z) = \underline{H}(z) \cdot \sigma_x^2 \quad \bullet \text{---} \circ \quad r_{xy}[n] = h[n] \cdot \sigma_x^2 \quad (2.6)$$

2.3 Nachhallzeit - (*reverb*ation time T_{60})

Nach *Wallace C. Sabine (1868 -1919)* beschreibt die Nachhallzeit den Zeitraum, in dem der Schalldruck auf den tausendsten, und die Energiedichte auf den millionsten Teil vom Anfangswert abgefallen ist. Dies entspricht einem Pegelabfall von $\Delta L = 60$ dB.

Hierzu definierte er die empirisch ermittelte Gleichung zur Bestimmung der Nachhallzeit T_S . Diese hängt vom Raumvolumen $[V] = \text{m}^3$ und der äquivalenten Absorptionsfläche $[A_W] = \text{m}^2$ ab, welche von der Raumbofläche $[S] = \text{m}^2$ und ihrem mittleren Absorptionskoeffizienten α abhängen.

$$T_S = 0,163 \frac{V}{A_W} \quad \text{wobei:} \quad A_W = \alpha \cdot S \quad (2.7)$$

Heute wird üblicherweise die Nachhallzeit nach *Eyering* berechnet, da diese hochwertigere Ergebnisse für Räume mit hoher Absorption liefert.

$$T_E = 0,163 \frac{V}{-S \cdot \ln(1 - \alpha)} \quad (2.8)$$

Da die Nachhallzeit eine mittlere Eigenschaft des gesamten Raumes darstellt, wird bei der messtechnischen Erfassung der Nachhallzeit eine räumliche und zeitliche Mittelung angestrebt. Somit werden mehrere Durchläufe an verschiedenen Mess- und Anregungspunkten durchgeführt. Zudem ist Sie eine stark frequenzabhängige Größe und wird bei genauerer Untersuchung für die mittleren Terz- oder Oktavfrequenzen angegeben. Die Bestimmung von T_{60} erfolgt aus der Abklingkurve des gemessenen Mikrofonsignals. Häufig wird der technisch praktikable Lautsprecher keinen Signal-Rauschabstand (SNR) von 60 dB ermöglichen. Erschwerend sind noch die 5 dB Zeit für den Abfall nach Abschalten der Rauschquelle zusätzlich aufzubringen. Daher wird in der Praxis eine nutzbare Pegeldifferenz von 20 – 30dB angestrebt und deren Abklingzeit auf 60 dB Pegelabfall extrapoliert. Diese Abklingzeit wird dann mit T_{20}/T_{30} angegeben, was gleichzeitig eine Maßgabe für die Qualität der Messung darstellt.

2.4 Schröder-Rückwärtsintegration [Sch65]

Die Schröder-Rückwärtsintegration ermöglicht es, aus der Impulsantwort des Raumes die energetische Abklingvorgang (EDC *engl. energy-decay-curve*) als Funktion der Zeit zu ermitteln. Dabei wird die quadrierte Impulsantwort in umgekehrter Zeitrichtung integriert. Diese Berechnungsmethode ermöglicht den linearen Pegelverlauf in einem Diagramm darzustellen und daraus die **Nachhallzeit** zu bestimmen.

$$s^2(t) = \int_{t_{10}-T_S}^{t_{10}} h^2(t) dt \quad (2.9)$$

Um den Einfluss des Störpegels zu minimieren wird die Integration zum Zeitpunkt t_{10} begonnen. Der Zeitpunkt t_0 stellt den Schnittpunkt des Störpegel (*horizontal*) und der Geraden der quadrierten Impulsantwort (*abfallend*) dar. Um zuverlässige Resultate zu erzielen wird ein Offset von 10 dB zum Umgebungsrauschen verlangt. Somit beschreibt t_{10} den Zeitpunkt an dem die Gleichung ($L_{t_{10}} = L_{t_0} + 10 \text{ dB}$) das Erste mal erfüllt ist.

Die Genauigkeit der Nachhallzeit wird nach [Deu09, Abschnitt 7.2] als gut beschrieben, da keine weitere zeitliche Mittelung (in Form von Messwiederholungen) notwendig ist.

2.4.1 Überlegung zur Rückwärtsintegration (ETC \rightarrow EDC)

Die Berechnung der Nachhallzeit ist in [Sch65] klar beschrieben. Aus meinem Bauchgefühl heraus war mir nicht schlüssig, wie die Steigung des Energieabfalls (ETC) nach der Rückwärtsintegration erhalten bleibt. Dieser Sachverhalt wird hier an einer gedämpften harmonischen Schwingung bzw. deren Hüllkurve näher untersucht.

$$\begin{aligned} p(t) &= \text{Impulsantwort} \\ p^2(t) &= \text{energy-time-curve (ETC)} \\ \int p^2(t) &= \text{energy-decay-curve (EDC)} \end{aligned}$$

Die Impulsantwort wird hier durch eine harmonisch gedämpfte Schwingung $p(t) = p_x \cdot e^{-\alpha t} \cdot \sin(\omega_d \cdot t)$ dargestellt. Da für die diese Betrachtung die eigentliche Schwingung nicht von Nöten ist, wird weiterhin lediglich deren Hüllkurve $p(t) = p_x \cdot e^{-\alpha t}$ betrachtet. Dabei stellt p_x eine beliebige Amplitude des Schalldrucks und α die Bedämpfung des Raumes dar.

Durch Quadrieren der Impulsantwort ergibt sich die ETC, die zu jedem Zeitpunkt die aktuell einfallende Schallenergiemenge darstellt.

$$ETC \Rightarrow p^2(t) = (p_x \cdot e^{-\alpha t})^2 = p_x^2 \cdot e^{-2\alpha t} \quad (2.10)$$

Wird diese ETC nun über eine logarithmische Achsenskalierung dargestellt ergibt sich:

$$\lg(p^2(t)) = \lg(p_x^2) + \lg(e^{-2\alpha t}) \quad (2.11)$$

$$= 2 \cdot \lg(p_x) + (-2\alpha t \cdot \lg(e)) \quad \text{mit: } \lg(e) = \frac{1}{\ln(10)} \quad (2.12)$$

$$= \left(\frac{-2\alpha}{\ln(10)} \right) \cdot t + 2 \cdot \lg(p_x) \quad (2.13)$$

$$y = m_{ETC} \cdot x + b_{ETC} \quad (2.14)$$

Bei geeigneter Darstellung lässt sich in Gl. (2.13) direkt der Zusammenhang einer einfachen Geradengleichung $y = mx + b$ nach Gl. (2.14) erkennen. Wie sich zeigt, hat die Amplitude des Signals keine Auswirkung auf die Steigung der Kurve und lediglich eine y-Achsenverschiebung zur Folge. Die Steigung m_{ETC} hängt somit lediglich von der Dämpfung des Raumes und weiteren Konstanten ab.

Diese beiden Sachverhalte sind logisch und nachvollziehbar, denn das Abklingverhalten nimmt stetig um ein Verhältnis und keinen Absolutwert ab.

Wird nun die ETC nach Schröder rückwärtsintegriert, ergibt sich die EDC.

$$EDC = \int p^2(t) = \int (p_x^2 \cdot e^{-2\alpha t} dt) \quad (2.15)$$

$$= p_x^2 \cdot \int e^{-2\alpha t} dt \quad (2.16)$$

$$= p_x^2 \cdot \left(-\frac{e^{-2\alpha t}}{(-2\alpha)} + C \right) \quad (2.17)$$

In Gleichung Gl. (2.17) muss besonders auf die Vorzeichen geachtet werden und woher diese stammen. Das grüne Minuszeichen stammt aus der üblichen inneren Ableitung des Exponenten. Das rote Minuszeichen entsteht durch die Rückwärtsintegration. Nach dem Vertauschungsgesetz der Integrationsregeln muss beim Vertauschen der Integrationsgrenzen das Vorzeichen umgekehrt werden. Somit heben sich beide negativen Vorzeichen auf. Die Integrationskonstante wird in der weiteren Betrachtung durch $C = 0$ vernachlässigt.

In der logarithmischen Darstellung der EDC entsteht:

$$\lg \left(\int p^2(t) \right) = \lg \left(\frac{p_x^2}{2\alpha} \right) + \lg(e^{-2\alpha t}) \quad (2.18)$$

$$= \left(\frac{-2\alpha}{\lg(10)} \right) \cdot t + 2 \cdot \lg(p_x) - \lg(2\alpha) \quad (2.19)$$

$$y = m_{EDC} \cdot x + b_{EDC} \quad (2.20)$$

Betrachten wir nun die beiden relevanten Koeffizienten der ETC und EDC, so wird ersichtlich, dass $b_{ETC} \neq b_{EDC}$ und somit beide Geraden unterschiedliche Schnittpunkte mit der y-Achse aufweisen. Jedoch ist $m_{ETC} = m_{EDC}$ und damit der Pegelabfall über die Zeit identisch.

Sie sind somit parallele Linien und liefern die gleiche Dämpfungskonstante α und damit die gleiche Nachhallzeit.

Einfach gesagt, kommt dies aus der Eigenschaft der e-Funktion. Da die Ableitung und Integration der e-Funktion stets sich selbst ergibt, bleibt in logarithmischer Darstellung die Steigung der Kurven gleich.

2.5 lineare Regression / Regressionsmodell

Die wahre Funktion kann durch ein Regressionsmodell nie gefunden werden, denn durch systematische und zufällige Fehler werden unsere erhobenen Daten verfälscht.

Jedoch ermöglicht ein Regressionsmodell eine lineare Funktion zu ermitteln, welche die Beziehung zwischen diesen Variablen *am Besten* beschreibt. Der Zusammenhang ist eben nicht deterministisch, und somit nicht vorher festgelegt. [Fro18]

- deterministisch
 - gleicher Eingabewert \rightarrow gleicher Ausgabewert
 - basiert auf bekannten Gesetzmäßigkeiten
- stochastisch
 - gleicher Eingabewert \rightarrow stets unterschiedlicher Ausgabewert
 - Überlagerung von zufälligen Fehlereinflüssen

Die Grundlage des linearen Regressionsmodells beschreibt die [Gleichung \(2.21\)](#). Wobei die Variable u_i alle unsystematischen und zufälligen Fehler beinhaltet.

$$y_i = a + bx_i + u_i, \quad a, b \in \mathbb{R}, \quad i = 1, 2, \dots, n \quad (2.21)$$

Die beste Näherung der wahren Funktion ist die, wenn ihre Koeffizienten (a, b) den Fehler nach der Methode der kleinsten Quadrate minimieren.

$$\sum_n^{i=1} u_i^2 = \sum_n^{i=1} (y_i - a - bx_i)^2 \rightarrow \min_{a,b} \quad (2.22)$$

Die Bestimmung der geschätzten Koeffizienten liefern:

$$\hat{b} = \frac{s_{xy}}{s_x^2} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2} \quad \text{und} \quad \hat{a} = \bar{y} - \hat{b}\bar{x} \quad (2.23)$$

Dabei beschreibt \hat{b} den Schätzwert, für den unbekannt *wahren* Modellparameter b . Aus beiden Koeffizienten lässt sich die Regressionsgerade \hat{y} und ihren Regressionswerten \hat{y}_i bilden. Da es sich bei Regressionswerten um Schätzwerte handelt, ergibt sich die Gleichung

$$E(y_i) = \hat{a} + \hat{b}x_i. \quad (2.24)$$

Diese Regressionsgerade verläuft dabei stets durch den Schwerpunkt (\bar{x}, \bar{y}) .

Da es sich bei \hat{a}, \hat{b} um Erwartungswerttreue und unverzerrte Schätzwerte handelt, sind diese weder über- noch unterschätzt. Und es lässt sich somit schreiben:

$$\widehat{E(\hat{b})} = \hat{y}_i = b \quad \text{bzw.} \quad E(\hat{a}) = a \quad (2.25)$$

Jedoch kann ein Modell niemals die Realität vollständig abbilden. Hierbei sei auf die Güte der für das Modell notwendigen Modellvoraussetzungen verwiesen, die in [Fro18, Kap.4] näher beschrieben werden. In dieser Arbeit dient die lineare Regression zur verbesserten Annäherung der abfallenden Energiekurve und somit stochastisch geeigneteren Nachhallzeitermittlung.

2.6 Zeitsynchrone Mittelung

Die zeitsynchrone Mittelung (*engl. time-synchronous-averaging*) wird zur Minderung von Störgeräuschen verwendet. Die Idee dahinter ist denkbar einfach. Die Wiederholung einer Messung mit dem gleichen Nutzsignal liefert jedes Mal *fast* das gleiche Ergebnis. Der Raum prägt dem Signal seine Eigenschaften immer gleich auf, jedoch wird das Störgeräusch immer zeitlich unterschiedlich auftreten.

Somit tritt eine konstruktive Überlagerung des Nutzsignal und eine destruktive Überlagerung des Störsignals ein. Mit dieser zeitlichen Mittelung wird es somit möglich eine Steigerung der SNR zu erreichen. Theoretisch könnten mit einer langen Mittelungszeit alle Störgeräusche eliminiert werden [RCI11, Abschnitt 2.3].

$$\bar{x}[n] = \frac{1}{N} \sum_{r=0}^{N-1} x[n + rT_0] \quad (2.26)$$

$\bar{x}[n]$ = gemittelttes Signal
 N = Anzahl Messungen
 T_0 = Signallänge

Dieser Ansatz findet besonders bei Modalanalysen mit einer Hammer-Erregung Verwendung. Die Mittelung bei dieser speziellen Anwendung führt neben einer Erhöhung des SNR auch zu einer Mittelung der Anregungsrichtung, welche immer ein paar Grad Verschiebung zueinander aufweist. Dies ermöglicht den *Störeinfluss Mensch* zu minimieren und liefert objektivere Ergebnisse.

3 Messung und Test des Skriptes

3.1 Messdurchführung und Equipment

Zur Testung des Skriptes wurde ein großer Raum gesucht, welcher zudem noch eine relativ lange Nachhallzeit aufweisen sollte. Hierzu fiel der Gewölbekeller unseres Hauses in den Mittelpunkt. Er weist allseitig schallharte Oberflächen auf und sollte durch seine ungleichmäßig gewölbte Deckenform eine ordentliche Diffusität des Schallfeldes erzeugen. Vor dem Messaufbau klatschte ich einmal in die Hände und war zunächst verblüfft, denn die Nachhallzeit schien erheblich kürzer als ich erwartet hätte. Die [Abbildung 3.1](#) zeigt den Raum mit seinem Inhalt und groben Form. Es ist von Autoreifen, Wasserkästen, Kartoffeln, alten Schlitten, Fahrrädern und Holz & Kartonplatten alles dabei (eine typische, volle, unaufgeräumte Abstellkammer). Der Raum hat die Abmessungen (H:2,3 m – B:5,4 m – T:4,3 m) und ein in SpaceClaim geschätztes Gesamtvolumen von $V = 50,7 \text{ m}^3$. Die Mikrofon- und Lausprecheranordnung im Raum ist in [Abbildung A.4](#) skizziert.



Abb. 3.1: Vermessene Raum mit Inhalt und Messtechnik

Wiedergabe- und Aufnahmegeräte

- Als Wiedergabegerät wurde eine günstige PA-Aktivbox von „Pronomics-Fox12A“ verwendet. Mit einem Frequenzbereich von 50 Hz bis 18 000 Hz und einer 12"Membran sollten sie einen zwar nicht linearen, aber ausreichenden Schalldruckpegel erzielen. Mit einem 6,3 mm auf 3,5 mm-Klinkenadapter im Mic-Eingang der Box, konnte das Messsignal direkt an die Box geleitet werden und über den *Level-Regler* angepasst werden.
- Als Aufnahmegerät wurde das günstige Messmikrofon „Behringer-ECM8000“ verwendet. Das Elektret-Kondensatormikrofon hat einen sehr linearen Frequenzgang und eine omnidirektionale-Richtcharakteristik und ist somit ideal für eine *Hobby-Messung*. Die Übertragung erfolgt über ein symmetrisches XLR-Kabel an das Mischpult.
- Das digitale Mischpult „Behringer XR12“ versorgt das Mikrofon mit der notwendigen Phantomspeisung und ermöglicht die Ausgabe über den Kopfhörerausgang. Ein erheblicher Nachteil stellt die Lautstärkeregelung des Kanals dar. **Somit ist der Kalibrierwert des**

Mikrofons hinfällig und alle absoluten Pegelwerte der Messung sind inkorrekt!

Da es jedoch ein systematischer Fehler ist, sind die zur Nachhallzeitbestimmung notwendigen Pegeldifferenzen korrekt. Die Einstellung des Mischpults sind neutral, lediglich ein Hochpassfilter mit Grenzfrequenz von $f_G = 30$ Hz ist zugeschaltet.

- Die interne Soundkarte eines Laptop „Lenovo Thinkpad E595“ dient zur Aufzeichnung und Wiedergabe sämtlicher Signale in Matlab.



(a) digitales Mischpult - Behringer XR12



(b) Lautsprecher Eingang

Abb. 3.2: Messequipment - Eingänge und Ausgänge

Als Problem der Soundkarte eines Laptops ist direkt die Anschlussart zu nennen. Der 3,5 mm-Klinkenstecker ist zwar im Consumerbereich sehr üblich, jedoch ist das Kabel unsymmetrisch und damit anfällig für Störeinflüsse. Neben ungeeigneter Abstrahlcharakteristik und nicht-Linearität des Lautsprechers sind besonders Artefakte der Aufnahmekette ein großes Problem. Eine Kontrolle der Daten ist immer notwendig, da stets neue Fehlereinflüsse auftauchten die unberechenbar und zufällig eintraten. Ebenfalls problembehaftet ist die Aussteuerung von Lautsprecher und Mikrofon.

3.2 Untersuchung zum Zeitversatz der Soundkarte

In der Aufgabenstellung wird auf einen nicht reproduzierbaren zeitlichen Versatz von herkömmlichen Soundkarten hingewiesen. Hierbei wird zur Untersuchung ein weißes Rauschen am Kopfhörerausgang angelegt und über ein 3,5 mm-Klinkenkabel direkt wieder in den Mikrofon-eingang geleitet. Somit entsteht eine *ideale Verbindung*, wodurch eine Abweichung nur auf die Soundkarte oder die Software zurückzuführen ist.

Zur Untersuchung wurde das Skript in [Abschnitt A.3](#) erstellt. Da das weiße Rauschen lediglich zur Zeit $t = 0$ sich selbst ähnlich ist, entsteht bei identischem Ein- und Ausgangssignal ein Peak in der Mitte der Kreuzkorrelierten aus beiden Signalen.

Liegt dieser Peak nun außerhalb der Mitte, weist die Soundkarte einen zeitlichen Versatz auf. Um herauszufinden ob dieser wirklich nicht reproduzierbar ist, wird in einer Schleife ein weißes Rauschen erzeugt, abgespielt und wieder aufgezeichnet. Sind diese zeitlichen Abstände bei jedem Schleifendurchgang unterschiedlich, sind diese zufällig.

Wie sich im Ausgabefenster ([Listing A.3](#)) zeigte, sind diese zeitlichen Abweichungen stets unterschiedlich. Dies konnte für die interne und eine weitere externe USB-Soundkarte getestet und bestätigt werden. Zusätzlich wurde noch die Varianz des erzeugten weißen Rauschen untersucht, da $\sigma^2 \neq 1$ nach [\[Grü14, Gleichung 5.66\]](#) eine Skalierung der Impulsantwort zur Folge hätte. Die Werte schwanken leicht um die den Wert 1 und ihre Amplituden-Verteilung folgt wie erwartet einer Gauss-Verteilung. Diese sind als hinreichend genau angenommen.

3.3 Entwicklung des Skriptes

Zur Programmierung wurde Matlab-R2020a (9.8.0.1323502) verwendet. Zur Verwendung ist die Toolbox *Communications Toolbox* und *Signal Processing Toolbox* notwendig.

Die Beschreibung des Skriptes erfolgt in der chronologischen Reihenfolge bei Skriptausführung. Das vollständige Skript ist im [Abschnitt A.1](#) vollständig enthalten. In den weiteren Abschnitten sind die zugehörigen Code-Zeilen hinzugefügt.

3.3.1 Parametereingabe

Zu Beginn müssen zunächst die Eingabeparameter durch den Nutzer eingegeben werden. Hierzu zählt die Zuweisung des richtigen Abspiel- und Aufnahme Gerätes. Eine Abtastrate von 48 000 kHz bei einer Auflösung von 24 bit sorgt für eine ausreichende Aufnahmegenaugigkeit. Ein Einzahl-Kalibrierwert des Mikrofons ermöglicht (zumindest theoretisch) eine Amplitudenrichtige Messung. Die untere Grenze für die Messdauer wird durch die Nachhallzeit des Raumes, sowie die tiefste Frequenz festgelegt. Letztere muss mindestens eine volle Schwingung in diesem Zeitfenster absolvieren. Bei einer Messzeit von 30 Sekunden sind diese Anforderungen zu genüge erfüllt. Die extreme Länge wäre physikalisch nicht nötig. Jedoch habe ich diese so lange gewählt um die zufälligen Artefakte der Soundkarte später ausschneiden zu können und lange Messabschnitte mit brauchbaren Messwerten zu erhalten. Zur zeitlichen Mittelung wurden 5 Messwiederholungen durchgeführt. Als wählbarer Parameter wurde die Pegeldifferenz zum Umgebungsrauschen eingefügt. Nach der ISO 3382 [Deu09] soll dieser Pegelabstand $\Delta L = 10$ dB betragen, um einen systematischen Messfehler auf 5 % zu beschränken.

Des weiteren werden die Aufnahme und Abspielgerät für Audiosignale erstellt. Als Messsignal wurde ein in der Frequenz exponentiell ansteigender Sweep ausgewählt, dessen Matlab-Funktion zur Erstellung in [Abschnitt A.4](#) ist.

```

1  %—Eingabe der Parameter
2  %%% Auswahl der richtigen Ein/Ausgänge
3  audio = audiodevinfo;           %Informationen über Audiodevice auslesen
4  audioIN = 1;                   %Mikrofon
5  audioOUT= 4;                   %Lautsprecher

7  %%% Parameter der Messdaten
8  fs = 48000;%Hz                 % Abtastrate [Hz]
9  nBits = 24; nChannels = 1;     % Bitrate % Mono Eingang
10 kalibrierwert = 0.0103;%V/Pa   % 10,3mV/Pa @1kHz (Behringer ECM8000)

12 %%% Messdauer und Mittelungsanzahl
13 messzeit = 30;%s               % Messzeit > Nachhallzeit & > 1/fu
14 Naverage = 5;                  % Anzahl von zeitlichen Mittelungsvorgängen
15 umgebungsabstand_db = 10;%dB   % Pegelabstand zu Umgebungsgeräuschen

17 %%% Erstellen von Messgerät/Player/Sweep
18 messgeraet=audiorecorder(fs, nBits, nChannels, audioIN);
19 noise=expsweep(30,6000, messzeit, fs);%expsweep(fu, fo, t, fs)
20 player = audioplayer(noise, fs, nBits, audioOUT);

```

3.3.2 Messung und Datenspeicherung

Der Pegelabstand zum Umgebungsrauschen wird zur Bestimmung der Nachhallzeit notwendig. Zur Anwendung hierfür wird eine Messung der Umgebungsgeräusche von 20s vorgenommen. Das Messsignal wird aus dem Aufnahmegerät geladen und mit dem Kalibrierwert des Mikrofons in Schalldruck umgerechnet. Um auftretende Artefakte der Soundkarte zu umgehen, wurde die erste Sekunde nicht berücksichtigt. Aus diesem Vektor wurde der Effektivwert gebildet und in den Schalldruckpegel umgerechnet. Dieser *noise-floor* dient später zur Festlegung des Integrationsbeginn bei der Rückwärtsintegration.

Bei der Messung trat ein Umgebungsschall von 43 dB auf. Wie in [Abschnitt 3.1](#) beschrieben sind die absoluten Pegelwerte durch die variable Lautstärkeregelung des Kopfhörerausgang nicht korrekt.

```

1 %%% Bestimmung des Umgebungspegel (noise Floor) mittels Effektivwert
2 pause(0.2)%s %Pause zur Vermeidung von Tastaturanschlägen
3 recordblocking(messgeraet,20); %20 Sekunden Umgebungsgeräusche aufnehmen
4 umgebung = getaudiodata(messgeraet); %Daten in Variable ablegen
5 umgebung_kal = (umgebung(1*fs:end))./kalibrierwert;
6 umgebung_rms = rms(umgebung_kal); %Pa %Effektivwert Umgebungsschall
7 noise_floor = 20*log10(umgebung_rms/(2*10^-5)); %dB
8 %Schalldruckpegel der Umgebung als noise_floor

```

Das Abspielen und Aufzeichnen, des durch den Raum veränderten Nutzsignals wird in einer *for*-Schleife umgesetzt. Nach Ablauf der Messzeit wird eine kurze Zeit zum Abklingen des Raumes gewartet (muss für sehr hallige Räume angepasst werden). Die aufgezeichneten Mikrofondaten werden in ein leeres Array Spaltenweise nebeneinander abgelegt. Diese Methode ist ziemlich elegant, da die Anzahl der Messungen problemlos verändert werden kann. Zu Erwähnen ist, dass stets das gleiche Anregungssignal genutzt werden muss (auch bei Rauschsignalen) um eine zeitliche Mittelung wie in [Abschnitt 2.6](#) durchzuführen.

```

1 %%% Aufzeichnung, Wiederholung, Speicherung des Mikrofondaten
2 %c-te Aufnahmedaten in c-te Spalte schreiben
3 c=1; mikrofonsignal=[];
4 for c=c:Naverage % Wiederholung
5 play(player); % Sweep abspielen
6 recordblocking(messgeraet,messzeit);% Aufnahme starten
7 pause(2); % Ausklingen des Raums
8 messdaten_messzeit = getaudiodata(messgeraet);
9 mikrofonsignal=[mikrofonsignal messdaten_messzeit];
10 end %Messdaten nach für jeden Durchgang in Spalte speichern

```

Wie in [Abschnitt 3.2](#) gezeigt wurde, weisen die Mikrofonssignale einen nicht reproduzierbaren zeitlichen Versatz auf und können nicht direkt gemittelt werden. Wie kann dennoch eine Mittelung durchgeführt werden?

Dieser Teil befindet sich nicht im automatisierten Skript, da eine einfache Implementierung nicht möglich war. Der Ansatz hierbei ist den Versatz der Messdaten durch die Korrelation festzustellen. Wie in [Abschnitt 3.2](#) bereits dargestellt, macht sich der Zeitversatz in einer *außer-mittigen* Verschiebung des Peaks in der Kreuzkorrelation bemerkbar. Eine Korrektur um genau diesen zeitlichen Versatz („*dist*“) bringt die Daten auf eine zeitliche Linie.

Die Implementierung ist nur Teilautomatisiert, da händisch die früheste Funktion „NX“ ausgesucht werden muss. Im Anschluss kann y in „NY“ variiert werden. Die Entfernten Werte werden am

Ende durch Nuller aufgefüllt um die Signallänge für das Array zu erhalten. Die Abweichung betrug bei der Messung zwischen 1000 und 3000 Samples und somit < 100 ms.

```

1 %%align Measurement data
2 distance = []; % Ablage der Zeitabstände
3 mikro=mikrofonsignal(:,3); % früheste Messung im Array
4 NX=mikrofonsignal(:,3); % Referenzsignal
5 NY=mikrofonsignal(:,y); % variables Signal

7 korr=xcorr(NX,NY); % Korrelation bilden
8 [MaxW,MaxInd]=max(korr); % Index an dem korr maximal
9 dist=(length(korr)+1)/2 - MaxInd;
10 %dist beschreibt den zeitlichen Versatz der Aufnahmesignale
11 distance=[distance, dist]; % Ablage der Differenz
12 newNY=NY((abs(dist)+1):end,:); % Verschieben auf gleiche Zeit
13 newNY(numel(NX))=0; % Signal auf gleiche Länge bringen
14 mikro=[mikro newNY]; % Signale in Spalten ablegen
    
```

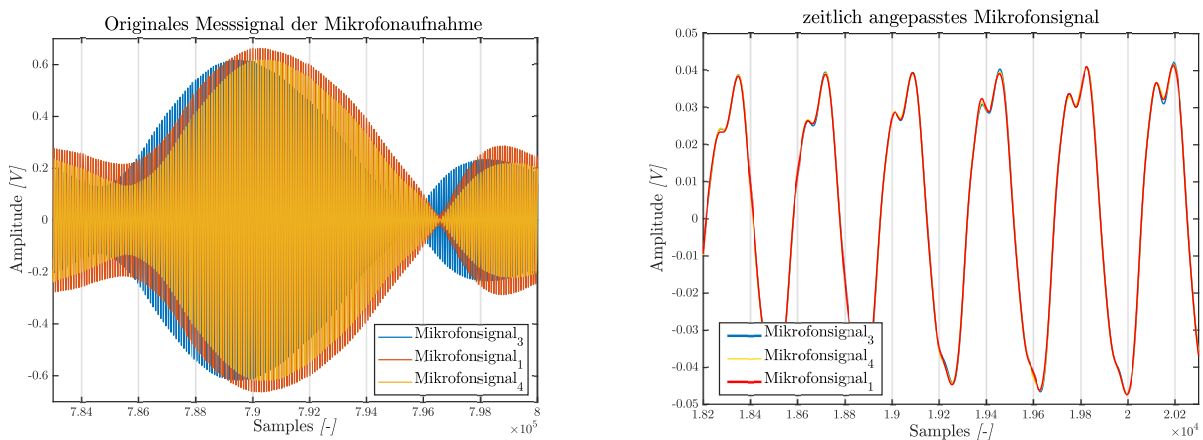


Abb. 3.3: zeitlicher Versatz der Mikrofonsignale infolge Soundkarte (vorher - nachher)

Das Entfernen der ersten Messdaten ist unkritisch, da das Messsignal beabsichtigt bei tiefen Frequenzen beginnt, welche die Boxen nicht erzeugen können. Deshalb müssen die Zeitsignale sinnvoll gekürzt werden. Hierbei stellt sich erst ab der Sekunde 8 ein brauchbarer Pegel bei 100 Hz ein. Am Messsignalende wurde das Signal ebenfalls auf etwa 5000 Hz bei Sekunde 29 reduziert. Das Lautsprechersignal wurde auf den gleichen Betrachtungszeitraum begrenzt.

In [Abb. 3.3](#) links sind 3 der originalen zeitversetzten Messsignale dargestellt. Im Bild rechts sind die zeitlich angepassten Kurvenverläufe.

```

1 mikro_neu = mikro(8*fs:29*fs-1,:); % Kürzen der Mikrofonsignale
2 mikrofonsignal = mikro_neu; % Abspeichern unter vorheriger Variable
3 noise = noise(8*fs:29*fs-1,:); % Kürzen des Sweep
    
```

Nun sind die Mikrofonsignale alle auf einen zufälligen zeitlichen Versatz des *Mikrofonsignal-3* angepasst. Jedoch bleibt der zufällige Versatz zwischen Ausgabe und Aufnahmedaten bestehen. Dieser kann nicht bestimmt werden und bleibt immer in den Daten und der späteren Impulsantwort.

Durch die erzielte Phasengleichheit der Mikrofonsignale entsteht konstruktive Überlagerung des Nutzsinal, aber zufällige Umgebungsgeräusche werden sich teilweise destruktiv auslöschen und damit reduzieren. In Codezeile 4 von [Listing 3.3.2](#) wird somit eine arithmetische Mittelung der

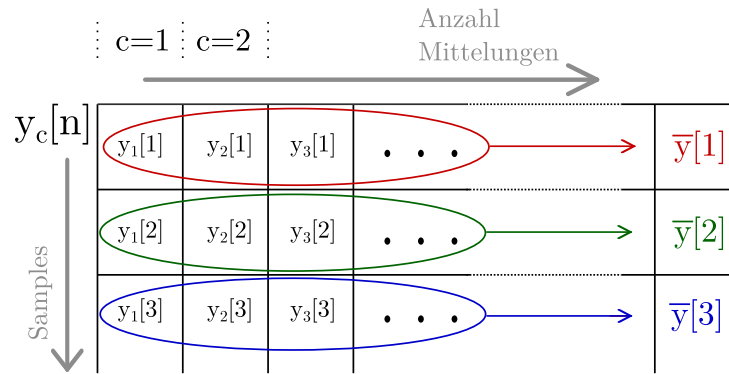


Abb. 3.4: schematisches Vorgehen zur zeitlichen Mittelung der Mikrofonaufnahmen

einzelnen Spalten getätigt um ein geräuschreduziertes Mikrofonsignal zu erhalten. Das Vorgehen ist nochmals in [Abb. 3.4](#) dargestellt. Die Speicherung in einem Array ist durchaus elegant, da eine Erweiterung frei wählbar ist und die Summierung ebenfalls sehr einfach umzusetzen ist. Zum Mikrofonsignal wird eine passende Zeitachse erstellt und diese in [Abb. 3.5](#) dargestellt.

```

1  %%% Mittelung der Aufnahmedaten zur SNR-Steigerung
2  %sum(...,2) creates the sum of all row-elements / size(,2) Array-width
3  %Effektivwert und danach Pegel des Mikrofonsignal berechnen
4  aufnahme = sum(mikrofonsignal,2) ./ size(mikrofonsignal,2);
5  aufnahme = aufnahme ./ kalibrierwert;

7  %%% Sweep für Messzeitraum schneiden und Zeitachse erstellen
8  zeit = 0:(1/fs):(length(aufnahme)-1)/fs;  zeit=zeit';

10 %%% Darstellen Schalldruck des gemittelten Mikrofonsignals
11 plot(zeit(:,1),aufnahme(:,1)), grid
12 xlabel('Zeit in {\it[s]}'), ylabel('Amplitude in {\it[Pa]}'),
13 title('Zeitsignal der gemittelten Aufnahmen'), xlim([0 (length(aufnahme)/fs)])

```

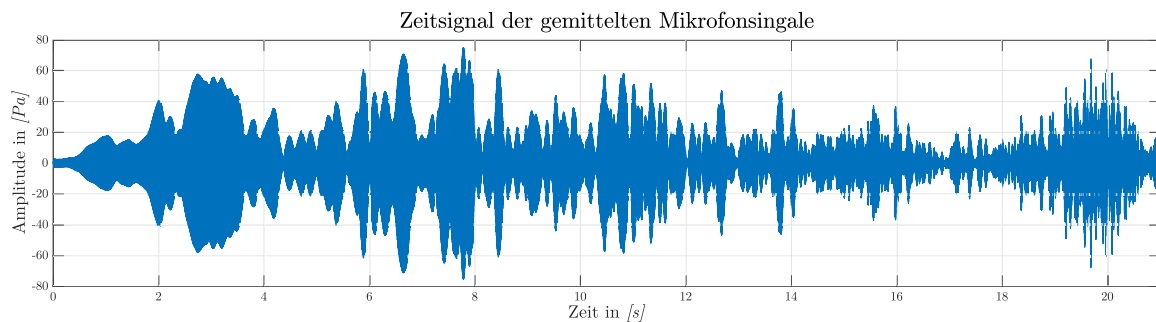


Abb. 3.5: gemitteltes Mikrofonsignal über die Messzeit

3.3.3 Kreuzkorrelation und Impulsantwort

Nach viel Datenaufbereitung wird nun nach [Grü14] mithilfe der Kreuzkorrelation die Impulsantwort des Raumes bestimmt. Hierzu wird die Funktion `xcorr()` verwendet. Der *biased*-Zusatz teilt die Amplitude durch die Messwerteanzahl. Dennoch wird eine Amplitude > 1 erzeugt, da der Kalibrierwert wie vorher beschrieben nicht korrekt ist. Auf welcher Seite der KKR die Impulsantwort vorhanden ist wird durch die Reihenfolge der Variablen in `xcorr()` bestimmt.

Die KKR wirkt nur so kurz, da die Messzeit so lange gewählt war.

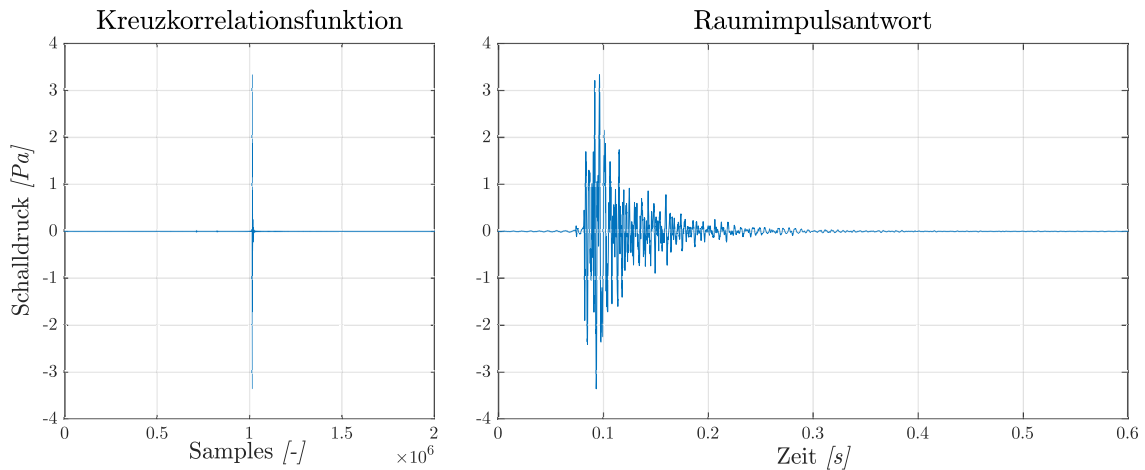


Abb. 3.6: Kreuzkorrelation und ausgeschnittene Raumimpulsantwort

```

1 %%% Kreuzkorrelation von Aus- und Eingang bilden und darstellen
2 kkor = xcorr(aufnahme, noise, 'biased');
3 plot(kkor)

```

Aus der Kreuzkorrelation wird nun ab der Mitte der Kreuzkorrelation die Raumimpulsantwort herausgetrennt. Dadurch bleibt der Zeitversatz des Direktschalls, sowie *leider* auch die Latenz der Soundkarte enthalten. Der Index des Maximums der Impulsantwort wird zur späteren Verwendung gespeichert.

Wie sich zeigt ist die Impulsantwort und damit die Nachhallzeit für den Raum überraschend kurz.

```

1 %%% Herausschneiden der Impulsantwort, aus der Kreuzkorrelation
2 kkor_mitte=round((length(kkor)+1)/2); % Bestimmen der Index-Mitte
3 imp = kkor(kkor_mitte:(kkor_mitte +1.5*fs),1); % Ausschneiden der Impulsantwort
4 [MaxWert, MaxImpInd] = max(imp); % Index der maximalen Amplitude der imp

```

3.3.4 ETC und Integrationsbeginn ermitteln

Durch die Quadrierung und Pegeldarstellung der Impulsantwort entsteht die ETC. Zur richtigen Pegeldarstellung wird die quadrierte Impulsantwort durch die quadrierte Hörschwelle geteilt und muss somit nur mit dem $10 \lg(\dots)$ multipliziert werden.

Der Zeitpunkt an dem die ETC in den Umgebungsgeräuschen untergeht stellt unserern späteren Integrationsbeginn der Rückwärtsintegration dar. Zu Beginn des Skriptes wurde diese Pegelabstand in der Variable *umgebungsabstand_dB* gespeichert. Die eigentliche ETC ist hierfür durch ihre Pegelschwankung ungeeignet. Stattdessen wird ein zeitliche Mittelung mit der Impuls-Zeitkonstanten ($\tau_I = 0,035 \text{ ms}$) verwendet. Mithilfe der Funktion *envelope* und der Einstellung (*rms*) kann diese zeitliche Glättung durchgeführt werden. Für diese Anwendung wird lediglich die obere Einhüllende verwendet und in die Grafik eingezeichnet. Ebenso wird der *noise-floor* grafisch hinzugefügt.

Mit der Funktion *find* wird der erste Wert bestimmt, der nach dem Maximum der geglätteten ETC unter unsere definierte Schwelle fällt. Formell wird die Gleichung $ETC_{rms} \leq (Umgebung + 10 \text{ dB})$ angewendet und zugehörige Index in *integrationsbeginn_idx* gespeichert.

```

1  %%% Darstellung der quadrierten Impulsantwort (ETC)
2  %die Kurve stellt den Schalldruckpegelverlauf dar
3  imp_quad= imp.^2;           %quadrierte Impulsantwort
4  L_impquad = 10*log10((imp_quad(1:end))./((2*10^-5)^2)); %ETC
5  zeit_imp = 0:(1/fs):(length(imp)-1)/fs; %Zeitachse
6  plot(zeit_imp,L_impquad); grid on; hold on;

8  %%% Bildung des RMS vom Messsignal, zur Bestimmung des Schnittpunktes mit Noise_floor
9  [imp_rms_upper,imp_rms_lower]=envelope(imp,round(0.035*fs),'rms');
10 %35ms Mittelung nach zeitlicher Impuls-Mittelung
11 imp_rms_db=20*log10(imp_rms_upper./(2*10^-5));
12 %%%Darstellen von Mittelungskurve und Noise_floor
13 plot(zeit_imp,imp_rms_db);
14 text = ['Umgebung = ',num2str(noise_floor),'dB'];
15 yline(noise_floor,'-',text,'color','green','LabelVerticalAlignment','top');

17 %%% Bestimmung des Index, an dem die ETC_rms unter die Schwelle fällt
18 integrationsbeginn_idx = find(imp_rms_db(MaxImpInd:end,1) < ...
    (noise_floor+umgebungsabstand_db),1) + MaxImpInd;
19 scatter(zeit_imp(1,integrationsbeginn_idx),imp_rms_db(integrationsbeginn_idx));

```

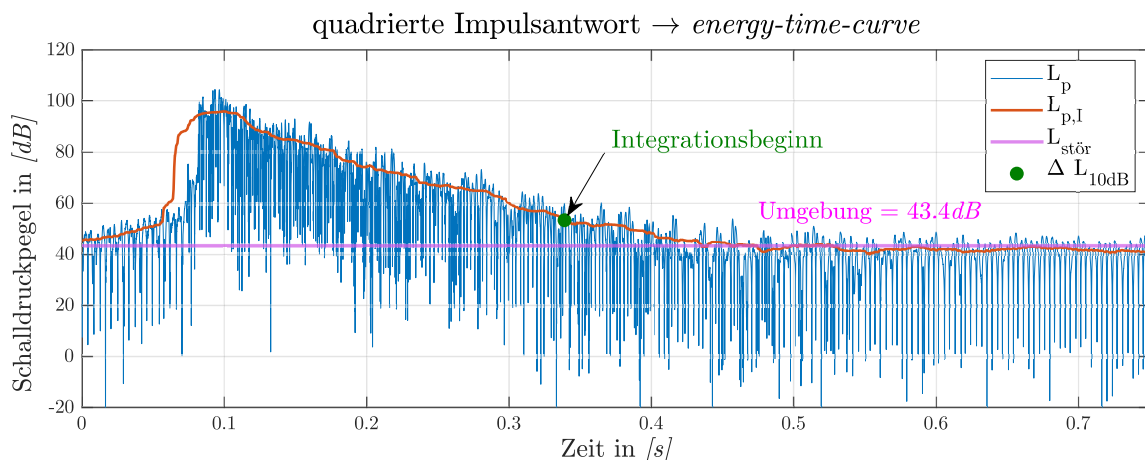


Abb. 3.7: ETC mit Umgebungsgeräuschpegel und Schwellwert

3.3.5 Rückwärtsintegration

Zu Beginn wird die transponierte¹, quadrierte Impulsantwort von Anfang bis zum Schwellwert mit der Funktion *fliplr* gedreht. Anschließend wird mit *cumtrapz* eine Integration (aufsummieren der diskreten Flächen) durchgeführt und wieder zurück gedreht. Dadurch wurde aus der ETC die EDC, welche in der Variable *integration1* gespeichert wird.

Eine Normierung der EDC auf 0 dB Ausgangspegel wird durch die Teilung auf den Startpegel erzielt.

```

1 %%Schröder-Rückwärtsintegration der ETC an Schwellwert
2 integration1 = fliplr(cumtrapz(fliplr(imp_quad(1:(integrationsbeginn_idx),1)')));
3 % Rückwärtsintegration
4 L = zeros(size(integration1));
5 L = 10 * log10(integration1(1:end)/integration1(1,1)); %normierte EDC
6 L_zeit = 0: (1/fs): (length(L)-1)/fs; %Zeitachse
7 plot(L_zeit,L)

```

3.3.6 lineare Regression und Nachhallzeitbestimmung

Nach [Deu09],[Ven13, Figure 2] liefert die EDC erst nach Abfallen der frühen Reflexionen ein repräsentative Aussage über die diffuse Energieabnahme im Raum. Aus diesem Grund werden die ersten 5 dB Pegelabfall nicht zur Nachhallzeitermittlung verwendet.

Somit gelten für zur Bestimmung der Nachhallzeit die obere Grenze L_{-5dB} und L_{10dB} als untere Grenze. Durch den fast vertikalen Abfall² gegen Ende der EDC, wurde nochmals ein Sicherheitsabstand von 35 ms zur Nachhallzeitermittlung eingefügt.

Zwischen den beiden Grenzen wird nun mit der Funktion *polyfit()* eine lineare Regressionsgerade erstellt. Die Koeffizienten der Gerade sind im Vektor *p* enthalten und mit Kurve *linreg* dargestellt. Zur Übersichtlichkeit wurden noch die Regressionsgrenzen hinzugefügt. Die EDC sowie die Regressionsgerade sind in Abb. 3.8 dargestellt.

```

1 %% Lineare Regression aus den zulässigen Schwellwerten
2 L_5db = find(L < max(L)-5, 1);
3 L_10db = (integrationsbeginn_idx) -(0.035*fs);
4 hold on;
5 %% ermitteln und einzeichnen der Regressionsgeraden in EDC
6 p = polyfit(L_zeit(:,L_5db:L_10db),L(:,L_5db:L_10db),1);
7 linreg = p(2) + L_zeit' .* p(1); %Regression einzeichnen
8 plot(L_zeit',linreg)

10 %% Regressionsgrenzen einzeichnen
11 scatter(L_zeit(:,L_5db),L(:,L_5db),'x');
12 scatter(L_zeit(:,L_10db),L(:,L_10db),'x');
13 hold off;

```

¹Die Impulsantwort muss nur wegen der Funktion *fliplr* transponiert werden, da diese nur für Zeilenvektoren funktioniert.

²Dies ist auf die Integration der sehr Amplitudenschwankenden Integration der ETC und die logarithmische Darstellung zurückzuführen.

Da die Steigung dieser Regessionsgeraden direkt die Energieabnahme enthält, muss dieser Wert noch auf den anerkannten Pegelabfall von 60 dB bezogen werden. Dadurch enthält $RT60$ die Nachhallzeit für den vermessenen Raum. Interessant für die Güte der Aussage ist aus welchem Pegelabfall die Nachhallzeit bestimmt wurde.

Die Nachhallzeit des Gewölbekeller scheint überraschend kurz, stimmt jedoch mit der Empfindung bei Händeklatschen überein. Was jedoch sehr unglaublich erscheint ist der nutzbare Pegelabfall von 35 dB. Trotz der maximalen Aussteuerung des Laustsprechers und hoher empfundener Lautstärke scheint dieser Wert zu groß zu sein. Eine Ursache für so hohe Pegeldifferenzen konnte nicht festgestellt werden.

Aus der ermittelten Nachhallzeit kann die äquivalente Absorptionsfläche von $A_W = 22,16 \text{ m}^2$ bestimmt werden. Durch die vielen auch hochkant im Raum stehenden Gegenstände scheint ein solcher Wert nicht abwegig zu sein.

$$A_W = 0,163 \cdot \frac{V}{T_S} = 0,163 \cdot \frac{50,7 \text{ m}^3}{0,37 \text{ s}} = 22,16 \text{ m}^2 \quad (3.1)$$

```

1 %%Nachhallzeit aus der Steigung der Regessionsgeraden und der
2 RT60 = 60/abs(p(1)); %p(1) ist die Steigung der Geraden
3 deltaRT= L(:,L_5db) - L(:,L_10db); % Pegeldifferenz der Nachhallzeitermittlung
4 fprintf('Nachhallzeit beträgt: %.2f Sekunden. \n Ermittelt aus einem nutzbaren ...
      Pegelabfall von: %.1f dB \n',RT60,deltaRT);

6 >> Nachhallzeit beträgt: 0.37 Sekunden.
7 >> Ermittelt aus einem nutzbaren Pegelabfall von: 34.5dB

```

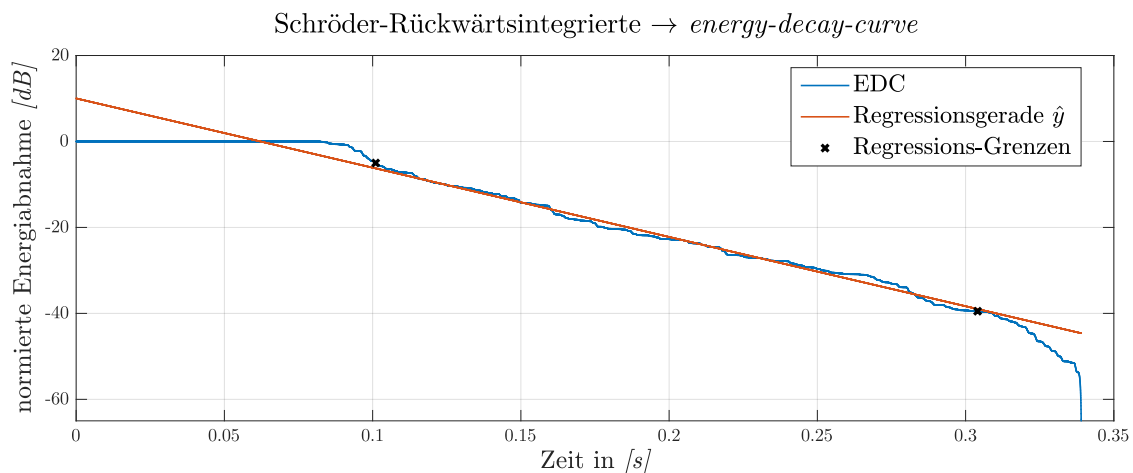


Abb. 3.8: EDC und Regressionsgerade zur Bestimmung der Nachhallzeit

3.3.7 Frequenzspektrum der Impulsantwort

Zur Bestimmung des zeitlichen Verlaufs des Frequenzspektrums der Impulsantwort wird die Matlab-Funktion `spectrogram` verwendet und mit `surf` 3-dimensional dargestellt.

Zu Beginn wird die Frequenzauflösung und damit die tiefste, messbare Frequenz sowie die Überlappung der FFT-Fenster eingegeben. Die Frequenzauflösung ist mit $\Delta f = 10$ Hz gewählt, sowie einer Überlappung von 75 % der einzelnen Hanning-Fenster. Die Spektrogramm-Funktion liefert als Ausgabe das komplexe Spektrum (`fft_data`) in Matrizenform, sowie die dazugehörigen äquidistanten Zeit- und Frequenzachsen als Vektoren.

Mit `abs(fft_data)/nfft` wird das komplexe Spektrum zum normierten Betragsspektrum. Die Darstellung der Surf-Funktion erfolgt als Schalldruckpegel durch $(20 \cdot \log_{10}(xxx)/(2 \cdot 10^{-5}))$ und wird über die beiden Achs-Vektoren aufgespannt. Zur zeitlichen Mittelung des Spektrogramms liefert `mean(...,2)` den Mittelwert der Frequenzlinien.

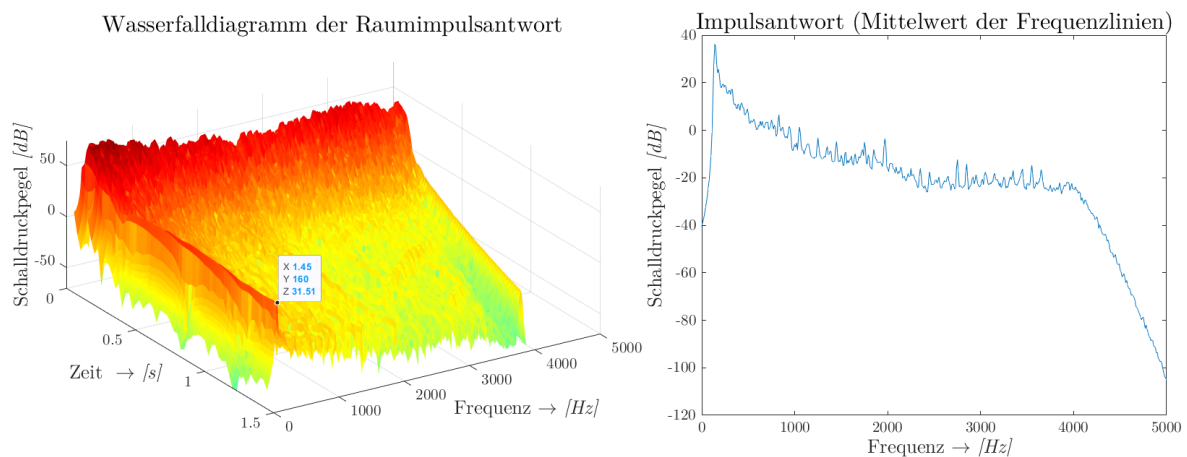


Abb. 3.9: Wasserfalldiagramm und Frequenzspektrum der Impulsantwort

```

1  %%% Frequenzanalyse Parameter
2  df= 10;                               %Frequenzauflösung /tiefeste Frequenz
3  overlap = 3/4;                          %Overlap fft-Fenster
4  nfft = fs/df;                            %Blocksize /Messwerte der FFT
5  window = hann(nfft);                    %Hanning Fenster erstellen
6  noverlap =ceil(nfft*(overlap));         %Bestimmung der Samples des Overlap

8  %%% Spektrogram: zeitlicher Verlauf des Frequenzspektrums eines Signals
9  [fft_data ,fachse ,tachse] = spectrogram(imp>window ,noverlap ,nfft , fs);
10     %Erstellung der Amplituden-,Frequenz- und Zeitachse aus Impulsantwort

12 %%% Darstellung des Wasserfall-Diagramms
13 subplot(1,2,1);
14 surf(tachse ,fachse ,(20*log10((abs(fft_data)/nfft)/(2*10^-5))), 'EdgeColor', 'none');
15     %Betragsspektrum der short time fourier transform
16 %view(0,90) %Campbell-Plot (Draufsicht)

18 %%% Darstellung der zeitlich gemittelten Frequenzanteile
19 subplot(1,2,2);
20 plot(fachse , mean((20*log10((abs(fft_data)/nfft)/(2*10^-5))),2));
21 %Amplitude über Frequenz plotten und 2
22 hold off;

```

3.3.8 Vergleich der Messergebnisse (*Room EQ Wizard*)

Da der Messaufbau bereits funktionsfähig war, entschied ich mich eine Messung mit der kostenlosen Messsoftware *Room EQ Wizard* - (*REW*) nochmals als Kontrolle durchzuführen. Es soll als objektivere Meinung des Messaufbaus und des Raumes und Kontrolle des Matlab-Skripts dienen. Diese Software verwendet ebenfalls einen logarithmischen Sweep im Frequenzbereich von 30 bis 6000 Hz mit einer Messzeit von 21,8 s. Es wurde jedoch (zur Schonung der Nachbarschaft) nur eine Messung durchgeführt.

Besonders interessant ist die objektiv vergleichbare Nachhallzeit. Diese wird hier frequenzabhängig in Terzschritten angegeben. Die Kurve T_{20} ist die Nachhallzeit aus 20 dB Pegelabfall berechnet, T_{30} respektiv aus 30 dB. Die Werte unter 100 Hz sind wegen geringem Energieeintrag nach [Abb. A.1](#) nicht zu verwenden. Auffällig ist, dass die T_{20} deutlich stärker schwankt als T_{30} , jedoch fallen ab 1 kHz alle Nachhallzeiten auf < 300 ms ab.

Leider stellt REW keinen Einzahlwert der Nachhallzeit zur Verfügung, jedoch deckt sich das Ergebnis in groben Maß mit der Nachhallzeit aus den Matlab-Messungen. Im [Abschnitt A.2](#) ist das ermittelte Wasserfall-Diagramm, sowie der erzeugte Schalldruckpegelverlauf (Absolutpegel nicht korrekt) zu finden.

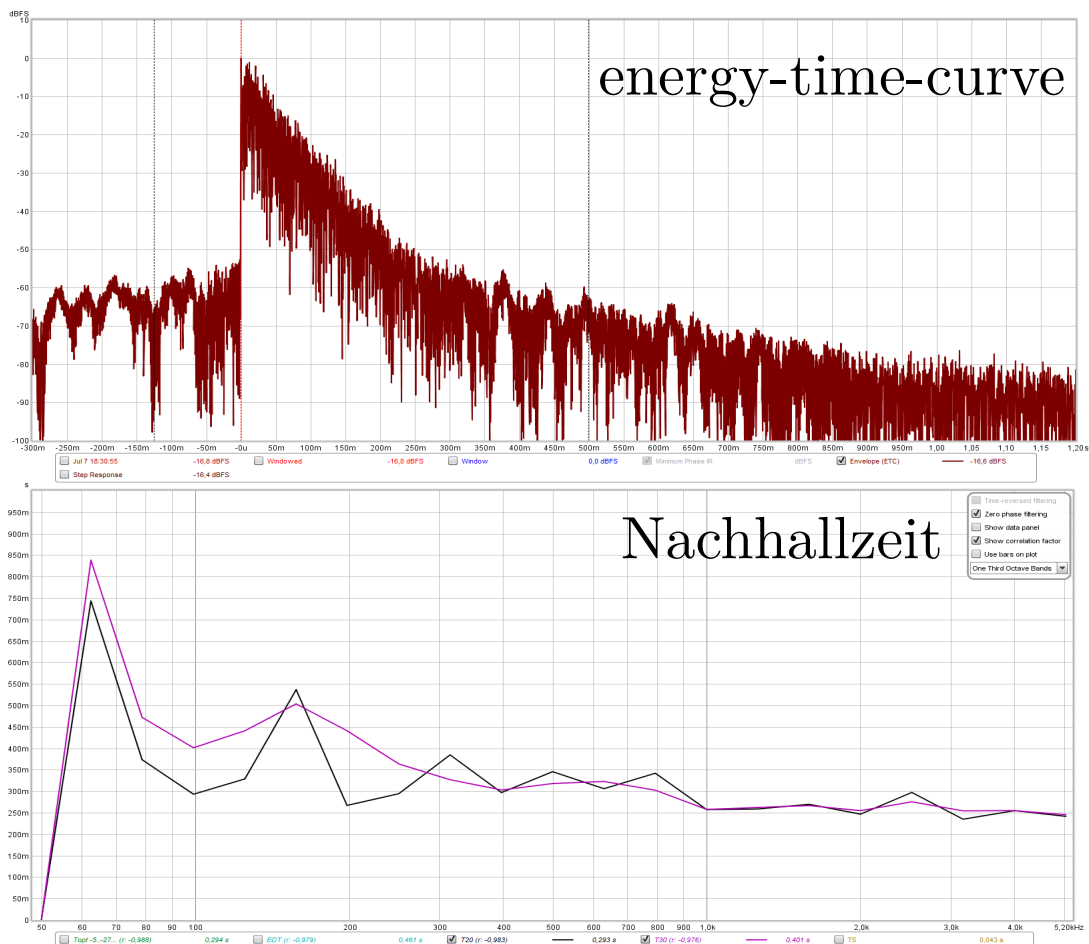


Abb. 3.10: ETC und frequenzabhängige Nachhallzeit aus REW-Messung

3.4 Diskussion zur Zuverlässigkeit der Ergebnisse

Bei dieser Arbeit ist die Umsetzung von akustischen Messungen mit einfachen Mitteln und der Programmierung in Matlab untersucht worden. Dadurch hat die verwendete Hardware nicht nur Einfluss auf die eigentliche Messung, sondern auch auf die Skripterstellung. Deshalb sei hier gesagt, dass dieses Skript nur unter diesen Messkette getestet wurde und andere Systeme, andere Probleme mit sich bringen können.

Die wesentlichen Ergebnisse wurden zur Verdeutlichung des Skriptes bereits in [Abschnitt 3.3](#) visualisiert. Ein essentieller Beeinflussungsfaktor ist das Aufnahme und Wiedergabesignal. Denn mit unbrauchbaren Eingangsdaten kann keine Software sinnvolle Ergebnisse ziehen.

Das Mikrofon liefert durch linearen Frequenzgang und omnidirektionales Richtcharakteristik eine geeignete Aufzeichnung. Leider wird durch die Lautstärkeregel am Mischpult keine korrekter Absolutpegel gemessen. Die Pegeldifferenzen bleiben jedoch fehlerfrei und können zur Messung von Pegelabfällen verwendet werden.

Der Lautsprecher weist durch sein gerichtetes Abstrahlverhalten von mittleren und hohen Frequenzen und dem nichtlinearen Pegelverlauf eine geringe Eignung für die Messung auf. Zunächst wird durch die Richtwirkung kein gleichmäßig verteiltes Schallenergiefeld im Raum entstehen und somit eine eher zu kurze Nachhallzeit gemessen. Durch den ungleichmäßigen Pegelverlauf sind leider auch Frequenzen über und untergewichtet und somit unterschiedlich stark im Messsignal enthalten. Diese nicht-Linearität wird direkt in die Impulsantwort übertragen.

Die verwendete Soundkarte stellt das schwierigste Element der Messkette dar. Durch den nicht reproduzierbaren zeitlichen Versatz der Signale, sowie weitere [zufällige Artefakte](#) musste erheblich mehr Aufwand in die Skripterstellung gesteckt werden. Dies war der Erzeugung von brauchbaren Daten geschuldet und nahm etwa die Hälfte der gesamten Bearbeitungszeit in Anspruch.

Bei der Skripterstellung wurden erste Versuche mit unkorreliertem Rauschen gestartet. Durch den erheblich höheren Energieeintrag, sowie die bessere Nachbereitung (infolge Soundkarten-Verunreinigung) wurde das Rauschen durch den logarithmischen Sweep als Erregungssignal ersetzt. Zur Steigerung der SNR wurde die [Zeitsynchrone Mittelung](#) verwendet. Das händische Eingreifen zur Mittelung der zeitversetzten Mikrofonsignale in [Abschnitt 3.3.2](#) stellt stets eine potenzielle Fehlerquelle dar. Jedoch konnte bereits bei 5 Mittelungen ein sehr gutes Rauschniveau wie in [Raumimpulsantwort](#) ersichtlich, erzielt werden. Dieser Mehraufwand kann vermieden werden, wollte ich aufgrund der deutlichen Verbesserung nicht missen. Der Zeitversatz der Soundkarte bleibt jedoch auch bei den gemittelten Signalen stets enthalten.

Weiterhin auffällig ist, dass der Peak der [Raumimpulsantwort](#) nicht zu Beginn auftritt. Die einzige Erklärung hierfür liegt in der Ausrichtung der Box, wodurch der gerichtete Schall des Hochtöners nicht per Direktschall auf das Mikrofon trifft.

Die Grenze der Rückwärtsintegration wurde durch den I-bewerteten RMS-Pegel aus der [ETC](#) und dem geforderten 10 dB Pegelabstand zu den Umgebungsgeräuschen ermittelt. Diese einfache aber robuste Methode liefert sehr brauchbare Ergebnisse. Das Ende des linearen Pegelabfalls, beziehungsweise die geforderten 10 dB Pegelabstand werden somit wiederholbar bestimmt. Diese objektive Erkennung ermöglicht damit die vollständige Automatisierung bei vorhandener Impulsantwort.

Die Regressionsgerade der **EDC** nähert den linearen Pegelabfall sehr gut an. Ebenfalls ersichtlich ist der verstärkte Abfall der frühen Reflexionen in den Ersten 5 dB Pegelabfall. Dennoch bleibt fraglich, ob die Box wirklich einen so hohen Energiemenge einbringen konnte. Jedoch zeigt die Messung **REW-Messung** mit der externen Akustiksoftware in der **EDC** ähnlich hohe Pegelabfälle und kann zumindest nicht auf einen Fehler im Matlab-Skript zurückgeführt werden.

Im **Wasserfall-Diagramm** ist ein spektral gleichmäßiger Pegelabfall vor 0,5 s zu erkennen. Die Pegelspitze bei 160 Hz kann nicht erklärt werden. Da der Pegel kein Abklingverhalten zeigt, sollte es sich um ein Störgeräusch der Umgebung oder der Messkette handeln.

Die kurze Nachhallzeit dieses Kellerraums konnte somit objektiviert werden. Die breitbandige Absorption durch die vielen großen Gegenstände im Raum scheint realistisch. Anhand dieser Messwerte konnte ich meinen Eltern einen objektiven Nachweis liefern, dass der Raum mit zu viel Krempel vollgestellt ist.

Zusammenfassend lässt sich also sagen dass die Auswertung der Daten im weitesten Sinne funktioniert. Jedoch stellt die brauchbare Datenerzeugung eine erhebliche Hürde der Messung mit einfachen Mitteln dar. Dies erschwert somit die eigentliche Datenverarbeitung des Matlab-Skriptes um diesen Fehlern zu umgehen. Somit sind solche *Hobby-Messungen* als eine grobe Abschätzung brauchbar, können jedoch niemals eine professionelle Untersuchung mit geeigneter Messtechnik ersetzt werden.

4 Zusammenfassung und Ausblick

Das Ziel der Belegarbeit war die Erstellung eines Matlab-Skriptes zur Bestimmung der Impulsantwort und Nachhallzeit mithilfe der Kreuzkorrelation sowie der Rückwärtsintegration nach Schröder. Bestandteil der Arbeit sollte auch die Verwendung und Anpassung des Skriptes mit einfachen Messmitteln („Consumer Bereich“) in einem beliebigen Raum sein.

Im Verlauf dieser schriftlichen Ausarbeitung konnte die Implementierung und der Umgang mit dem Skriptes erklärt werden. Das Skript umfasst die Erstellung eines logarithmischen Sweep und die Ermittlung der Impulsantwort, sowie die Nachhallzeitbestimmung aus der *energy-time-curve*. Zusätzlich wurde das geforderte Wasserfall-Diagramm der Impulsantwort dargestellt. Die Integrationsgrenzen der Schröder-Rückwärtsintegration konnten aus einer zusätzlichen Messung des Umgebungspegels ermittelt. Eine vollständige Automatisierung des Skripts war durch den zufälligen Zeitversatz der Soundkarte wie in [Abschnitt 3.3.2](#) beschrieben nicht möglich und bedarf händisches Eingreifen. Als objektive Prüfung des Matlab-Skripts wurden mit dem bestehenden Messaufbau eine Untersuchung mit der kostenlosen Messsoftware *Room EQ Wizard* durchgeführt. Hierbei ergaben sich vergleichbare Werte für die Nachhallzeit und den qualitativen Verlauf der ETC, relativ zum Matlab Skript.

Es wurde gezeigt, dass mit einfachen Mitteln raumakustische Untersuchungen stattfinden können. Dennoch weisen Sie erhebliche Unsicherheiten und Nachteile auf, die in [Abschnitt 3.4](#) näher diskutiert werden. Besonders störend ist hierbei die Soundkarte, deren zufällige Fehlerquellen eine stetige Datenkontrolle benötigen und immer einen deutlichen Mehraufwand zur Erstellung der Behelfslösung notwendig machten. Somit kann festgehalten werden, dass Messungen mit einfachen Mitteln zwar möglich sind, aber im Besten Fall nur für grobe Tendenzen der Raumeigenschaften geeignet sind.

Ausblick

Für eine vollautomatisierte Auswertung sind nur noch wenige, aber zeitintensive Schritte bei der Mittelung der Mikrofonsignale notwendig. Diese Implementierung würde den Zeitaufwand der Nachauswertung nochmals drastisch senken. Besonders sinnvoll wäre bei der Messung mit günstiger Soundkarte die Einführung von Kontrollparametern, die die Plausibilität der Messdaten prüft und gegebenenfalls eine Warnung ausgibt. Dadurch kann die händische Kontrolle reduziert werden.

Die Nachhallzeit wird üblicherweise in Terzen angegeben. Bisher wurde die Nachhallzeit im Skript jedoch nur als Einzahlwert ermittelt und somit von den lang nachklingenden Frequenzen dominiert. Eine Umsetzung der frequenzabhängigen Nachhallzeit erzielt eine erhebliche Steigerung des Informationsgehalts aus der Messung.

Selbstverständlich wäre ein Vergleich einer Raummessung zwischen professioneller Messtechnik und den hier verwendeten Mittel sehr interessant. Nur ein solch direkter Vergleich ermöglicht eine wirkliche Aussage über die Qualität der hier verwendet Messdurchführung.

A Anhang

A.1 vollständiges Skript

```
1 %%Ermittlung der Raumimplusantwort eines Raumes mit Hilfe der Kreuzkorrelation
2 %%%—Einleitung und Aufteilung—
3 % Belegaufgabe – Akustische Messtechnik —> Abgabe 30.09.2020
4 %Ingenieurakustik Durchgang 5
5 %Daniel Steinshorn
6
7 %—Benötigte Toolboxes – Version: MatlabR2020a (9.8.0.1323502)
8 % Communications Toolbox
9 % Signal Processing Toolbox
10
11 %Aufteilung des Codes:
12 % – Parametereingabe
13 % – Aufnahme /Laden der Messdaten
14 % – Bestimmen der Impulsantwort
15 % – Ermittlung der Nachhallzeit (Schröder-Rückwärtsintegrator)
16 % – Frequenzspektrum
17 %Mic 1 sec an um 1. Soundkarten "start" zu umgehen und 2. kann daraus der
18 %Umgebungspegel ermittelt werden—> +10 dB abschneiden der Nachhallzeit
19 %
20
21 %%%Eingabe der Parameter
22 hold off
23
24 %—Eingabe der Parameter
25 %%% Auswahl der richtigen Ein/Ausgänge
26 audio = audiodevinfo; %Informationen über Audiodevice auslesen
27 audioIN = 1; %Mikrofon
28 audioOUT= 4; %Lautsprecher
29
30 %%% Parameter der Messdaten
31 fs = 48000;%Hz % Abtastrate [Hz]
32 nBits = 24; % Bitrate
33 nChannels = 1; % Mono Eingang
34 kalibrierwert = 0.0103;%V/Pa % 10,3mV/Pa @1kHz (Behringer ECM8000)
35
36 %%% Messdauer und Mittelungsanzahl
37 %messzeit > erwartete Nachhallzeit
38 %und die tiefste Frequenz muss mindestens einmal hineinpassen
39 messzeit = 30;%s
40 % Anzahl von zeitlichen Mittelungsvorgängen
41 Naverage = 5;
42 %Pegelabstand zur Umgebung (ISO3382 fordert 10dB)
43 umgebungsabstand_db = 10;%dB
44
45 %%% Erstellen von Messgerät/Player/Rauschen
46 messgeraet=audiorecorder(fs , nBits , nChannels , audioIN);
47 noise=expsweep(30,6000, messzeit , fs); %Sweep erzeugt
48 player = audioplayer(noise , fs , nBits , audioOUT);
49
50
51 %%% Bestimmung des Umgebungspegel (Noise Floor) mittels Effektivwert
52 pause(0.2)%s %Vermeidung von Tastaturanschlägen
53 recordblocking(messgeraet ,20); %Umgebungsgeräusche aufnehmen
54 umgebung = getaudiodata(messgeraet);
55 umgebung_kal = (umgebung(1*fs:end))./ kalibrierwert;
```



```

56  umgebung_rms = rms(umgebung_kal); %Pa  %Effektivwert Umgebungs-schall
57  noise_floor = 20*log10(umgebung_rms/(2*10^-5)); %dB
58      %Schalldruckpegel der Umgebung als noise_floor

60  %%% Aufzeichnung, Wiederholung, Speicherung des Mikrofondaten
61  %c-te Aufnahmedaten in c-te Spalte schreiben
62  c=1; mikrofonsignal=[];
63  for c=c:Naverage          % Wiederholung
64  play(player);            % Sweep abspielen
65  recordblocking(messgeraet, messzeit); %aufnehmen
66  pause(2);                % Ausklingen des Raums
67  messdaten_messzeit = getaudiodata(messgeraet);
68  mikrofonsignal=[mikrofonsignal messdaten_messzeit];
69  end

71  %%% Mittelung der Aufnahmedaten zur SNR-Steigerung
72  %sum(...,2) creates the sum of all row-elements (Mittelwert)
73  %Effektivwert und danach Pegel des Mikrofon-signal berechnen
74  aufnahme = sum(mikrofonsignal,2)./size(mikrofonsignal,2);
75  aufnahme = aufnahme./kalibrierwert;

77  %%% Sweep für Messzeitraum schneiden und Zeitachse erstellen
78  zeit = 0:(1/fs):(length(aufnahme)-1)/fs;  zeit=zeit';

80  %%% Darstellen Schalldruck des gemittelten Mikrofon-signals
81  plot(zeit(:,1),aufnahme(:,1)), grid
82  xlabel('Zeit in \{it [s]\}'), ylabel('Amplitude in \{it [Pa]\}'),
83  title('Zeitsignal der gemittelten Aufnahmen'), xlim([0 (length(aufnahme)/fs)])

86  %%% Kreuzkorrelation von Aus- und Eingang bilden und Darstellen
87  %'biased' teilt durch Linienanzahl (normierung)
88  %Reihenfolge in xcorr bestimmt Seite der Impulsantwort
89  % kcorr kann >1 sein, da das Mikrofon größer als 1 ist
90  kcorr = xcorr(aufnahme, noise, 'biased');
91  plot(kcorr), grid, title('Kreuzkorrelationsfunktion'),
92  xlabel('Samples'), ylabel('Schalldruck \{it [Pa]\}')
93  %%%Herausschneiden der Impulsantwort, aus der Kreuzkorrelation
94  kcorr_mitte=round((length(kcorr)+1)/2);
95  imp = kcorr(kcorr_mitte:(kcorr_mitte +1.5*fs),1);%
96  [MaxWert,MaxImpInd] = max(imp);

99  %%% Darstellung der quadrierten Impulsantwort
100 %die Kurve stellt den Schalldruckpegelverlauf dar
101 imp_quad= imp.^2;
102 L_impquad = 10*log10((imp_quad(1:end))./((2*10^-5)^2));

104 %%% Bildung des RMS vom Messsignal, zur Bestimmung des Schnittpunktes
105 %%%von Messsignal = Umgebungspegel + 10dB
106 %->Dies ist der Punkt der als Integrationsanfang genutzt wird
107 [imp_rms_upper,imp_rms_lower]=envelope(imp,round(0.035*fs),'rms');
108     %35ms Mittelung nach Impuls-Mittelung
109 clearvars imp_rms_lower;
110 zeit_imp = 0:(1/fs):(length(imp)-1)/fs;  %zeit_imp=zeit_imp';
111 imp_rms_db=20*log10(imp_rms_upper./(2*10^-5));
112 plot(zeit_imp,L_impquad); grid on;
113 title('quadrierte Impulsantwort \rightarrow \{it {energy-time-curve}\}'),
114 xlabel('Zeit in \{it [s]\}'), ylabel('Schalldruckpegel in \{it [dB]\}'),
115 hold on;
116 plot(zeit_imp,imp_rms_db);
117 text = ['Umgebung = ', num2str(noise_floor), 'dB'];
118 yline(noise_floor, '-','text','color','green','LabelVerticalAlignment','top');

```

```

119     integrationsbeginn_idx = find(imp_rms_db(MaxImpInd:end,1) < ...
120         (noise_floor+umgebungsabstand_db),1) + MaxImpInd;
121     scatter(zeit_imp(1,integrationsbeginn_idx),imp_rms_db(integrationsbeginn_idx));
122     hold off;

123     %%%Schröder-Rückwärtsintegration der quadrierten Impulsantwort
124     % Rückwärtsintegrierte mithilfe von cumtrapz auf 0dB normieren
125     % In Pegel umrechnen, Zeitachse erstellen und plotten
126     integration1 = fliplr(cumtrapz(fliplr(imp_quad(1:(integrationsbeginn_idx),1)')));
127     L = zeros(size(integration1));
128     L = 10 *log10(integration1(1:end)/integration0(1,1));
129     L_zeit = 0: (1/fs): (length(L)-1)/fs;
130     plot(L_zeit,L),grid
131     title('Schröder-Rückwärtsintegrierte \rightarrow \it{energy-decay-curve}')
132     xlabel('Zeit in \it{s}'), ylabel('normierte Energieabnahme [dB] ');

134     %%% Lineare Regression
135     % Bestimmung der Indizes, an denen die Bedingung erfüllt sind
136     % (1): 5dB Pegelabfall vom Maxima
137     % (2): 10dB höherer Pegel als der Noise-Floor
138     L_5db= find( L < max(L)-5, 1);
139     %%%-> manuelle Eingabe des unteren Pegels
140     %L_10db = find(L<(-20),1);
141     L_10db = (integrationsbeginn_idx) -(0.035*fs);
142     %Sicherheitsabstand zu starkem Anstieg durch Integration

144     %%% ermitteln und einzeichnen der Regressionsgeraden in EDC
145     p= polyfit(L_zeit(:,L_5db:L_10db),L(:,L_5db:L_10db),1);
146     hold on;
147     %durgezogene Linie einzeichnen
148     linreg = p(2) + L_zeit'.*p(1);
149     plot(L_zeit',linreg)
150     %show the range of linear regression (start/end)
151     scatter(L_zeit(:,L_5db),L(:,L_5db),'x');scatter(L_zeit(:,L_10db),L(:,L_10db),'x');
152     hold off;

154     %%%Nachhallzeit aus der Steigung der Regressionsgeraden und der
155     %%%genutzten Pegeldifferenz
156     RT60 = 60/abs(p(1)); %p(1)ist die steigung der RT60
157     deltaRT= L(:,L_5db) - L(:,L_10db);
158     fprintf('Nachhallzeit beträgt: %.2f Sekunden. \n Ermittelt aus einem nutzbaren
159         Pegelabfall von: %.1f dB \n',RT60,deltaRT);
160     Frequenzanalyse

161     %%%Frequenzanalyse der Impulsantwort
162     df= 10;
163     nfft = fs/df; %Blocksize /Messwerte der FFT
164     window = hann(nfft); %Hanning Fenster erstellen
165     overlap = 3/4; %Overlap fft-Fenster
166     noverlap = ceil(nfft*(overlap)); %Bestimmung des "Zeitwerts" des Overlap

168     %—Bestimmen der FFT und insgesamte Frequenzanalyse
169     [fft_data, fchse, tachse] = spectrogram(imp,window,noverlap,nfft,fs);
170     %— Darstellung der FFT über Zeit und Frequenz (3D)—
171     subplot(1,2,1);
172     surf(tachse,fchse,(20*log10((abs(fft_data)/nfft)/(2*10^-5))),'EdgeColor','none');
173     xlabel('Zeit \rightarrow \it{s}'), ylabel('Frequenz \rightarrow \it{Hz}');
174     zlabel('Schalldruckpegel \it{dB}'), title('Wasserfalldiagramm der Raumimpulsantwort')
175     ylim([0 5000]), view([-0.0 90.0]) % Begrenzt die Frequenz im Plot

177     % normiert durch die nfft: Außerdem ist die FFt komplex, doch wir benötigen
178     % nur die Absolutwerte hier
179     %view(2); %view(90,0); %view(0,0) %Amplitude über Frequenz bzw. %Amplitude über Zeit
180     subplot(1,2,2);

```

```

181 plot(fachse , mean((20*log10((abs(fft_data)/nfft)/(2*10^-5)),2));
182 %Amplitude über Frequenz plotten und 2 Dimensional darstellen
183 title('Impulsantwort (Mittelwert der Frequenzlinien)')
184 xlabel('Frequenz \rightarrow {\it [Hz]}'), xlim([0 5000])
185 hold off;

```

A.2 Room EQ Wizard

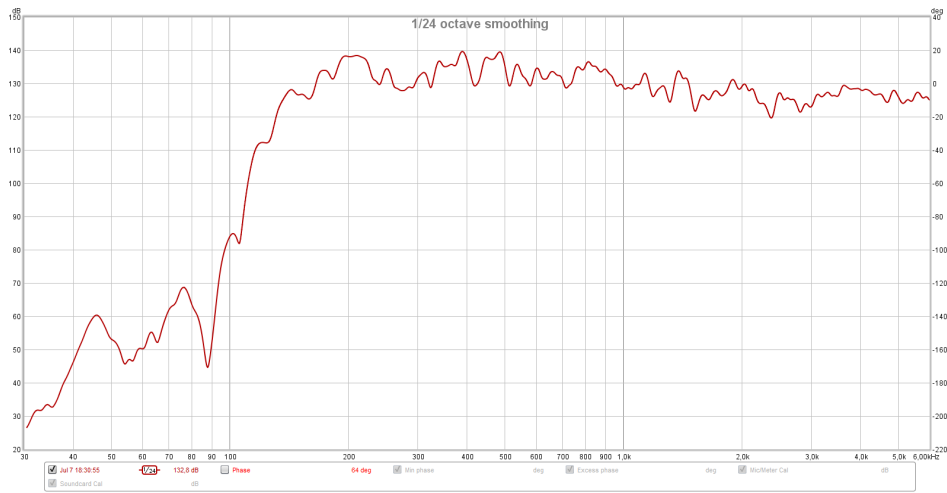


Abb. A.1: frequenzabhängiges SPL der REW-Messung im Raum

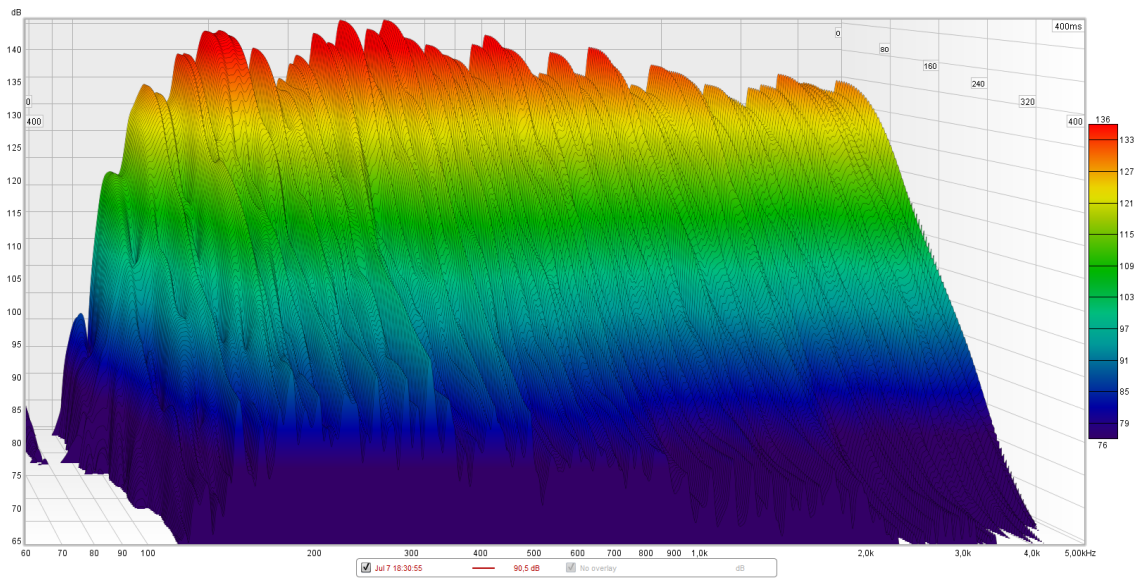


Abb. A.2: Wasserfalldiagramm mit 400 ms Zeitraum

A.3 Latenz der Soundkarte bestimmen

```

1 %% Ermittlung der Latenz einer Soundkarte

3 %{Es wird durch eine Loopback-Schleife der Lautsprecher Ausgang über ein
4 %AUX-Kabel direkt wieder auf den Mikrofoneingang geleitet. Dadurch wird die
5 %Latenz/ der zeitliche Versatz zwischen Eingang und Ausgang ermittelt.
6 %Für eine externe USB-Soundkarte wurde mit weißem Rauschen eine
7 %signifikanter Zeitversatz ermittelt, der sich auch für jede Messung
8 %verändert.%}

10 %—— Eingabe bevor vor Nutzung——
11 fs= 48000; %Abtastrate [Hz]
12 audio = audiodevinfo; %Informationen über Audiodevice
13 audioIN = 1; %Mikrofon-Eingangs ID
14 audioOUT= 4; %Lautsprecher-Ausgangs ID auswählen

16 %——Messgerät definieren——
17 messzeit = 2; %Aufnahmedauer, je Messung
18 nBits=24; nChannels = 1; %Bitrate %Mono-Eingang
19 messgeraet = audiorecorder(fs, nBits, nChannels, audioIN); %Messgerät anlegen

21 %—— Anzahl der Durchläufe der Latenzmessung
22 %—— Es wird für jede Schleife neu das White Noise erzeugt
23 for counter= 1:5
24     noise = wgn(messzeit*fs, 1, 0); %Erstellt das Weiße Rauschen als Vektor (n
        x 1)
25     % Weißes Rauschen mit messzeit in Sekunden, mit der Leistung "0dBW" = 1 Watt
26     varianz=var(noise); %Bestimmt die Varianz des Rauschsignals

28     player = audioplayer(noise, fs, nBits, audioOUT); %Erstellung des Abspielgerätes
        (Bewusst in der Schleife
29     %zur neuen Initialisierung der 'noise' Variable)
30     pause(player); %warten bis player fertig initialisiert ist
31     record(messgeraet, messzeit); %Aufnahme mit Messgerät nach Messdauer
        starten
32     play(player); %Wiedergabe des weißen Rauschen über
        gewählte Länge

34     pause(messzeit+0.5); %Skript solange pausieren bis Messung
        abgeschlossen

36     messdaten = getaudiodata(messgeraet); %Messdaten abspeichern
37     rauschenAKR = xcorr(noise, 'biased'); %Autokorrelation des Rauschens bestimmen
38     kkorr = xcorr(noise, messdaten, 'biased'); %Kreuzkorrelation der Signale
39     %——'biased' Teilt den Wert der Autokorrelation durch die Anzahl der Messwerte

41 %—— Bestimmen des Versatzes von von Eingang und Ausgang——
42 Zero_lag = round(length(kkorr)/2); %Wenn kein Versatz, liegt die Impuls in der
        mitte der KKF
43 [MaxWert, MaxWertIndex] = max(kkorr); %Maximalwert Bestimmung in der
        Kreuzkorrelation
44 lag = abs((MaxWertIndex-Zero_lag)/fs); %Latenz in der Soundkarte

46     fprintf(['Die Latenz der Soundkarte liegt bei %s ' ... %Ausgabe
47             'Sekunden mit einer Rauschvarianz von %s.\n']...
48             , num2str(lag, 5), num2str(varianz, 4));
49 end

51 %——Visualisierung anhand der zuletzt generierte Messdaten
52 close all;
53 subplot(3,1,1); %Erstellt von Plots mit
54 histogram(noise); %stat. Verteilung der der Werte im Rauschsignal

```

```

56 subplot(3,1,2);
57 plot(rauschenAKR); title('Autokorrelation White-Noise')
58 xlabel('Samples') %Autokorrelationsrauschen

60 subplot(3,1,3);
61 plot(kkorrr); title('Kreuzkorrelation ')%Kreuzkorrelationsrauschen
62 xlabel('Samples')
63 %—Es kann an den Plots visuell der zeitliche Versatz im Signal erkannt werden
    
```

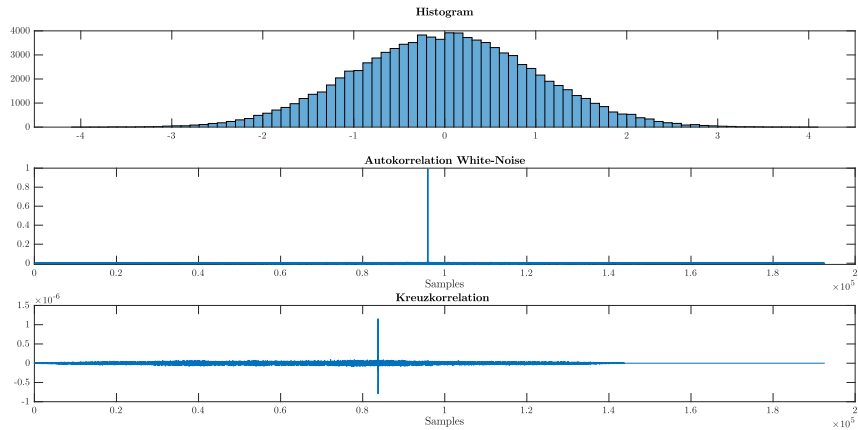


Abb. A.3: Darstellung des Zeitversatzes der Soundkarte

```

1 %%Command Window
2 >> LatenzSoundkarte
3 Die Latenz der Soundkarte liegt bei 0.26871 Sekunden mit einer Rauschvarianz von 0.9946.
4 Die Latenz der Soundkarte liegt bei 0.25033 Sekunden mit einer Rauschvarianz von 1.0053.
5 Die Latenz der Soundkarte liegt bei 0.24879 Sekunden mit einer Rauschvarianz von 1.0011.
6 Die Latenz der Soundkarte liegt bei 0.25540 Sekunden mit einer Rauschvarianz von 1.0057.
7 Die Latenz der Soundkarte liegt bei 0.24975 Sekunden mit einer Rauschvarianz von 0.9974.
    
```

A.4 exponentieller Sweep erstellen

Erstellung eines Sweep mit exponentiell ansteigender Frequenz bei gleicher Amplitude.

- f_{start} ist die untere Startfrequenz des Sweep
- f_{end} ist die obere Endfrequenz des Sweep
- T ist die Dauer des Sweep in Sekunden
- f_s ist die Samplerate des Sweep

```

1 function vec=expsweep(fstart,fend,T,fs)
2 % vec=expsweep(fstart,fend,T,fs)
3 %T=T/fs;
4 w1=2*pi*fstart;
5 w2=2*pi*fend;
6 A=T*w1/log(w2/w1); %log = log. naturalis
7 tau=T/log(w2/w1);
8 t=0:1/fs:T-1/fs;
9 vec=sin(A*(exp(t/tau)-1));
    
```

A.5 Messanordnung

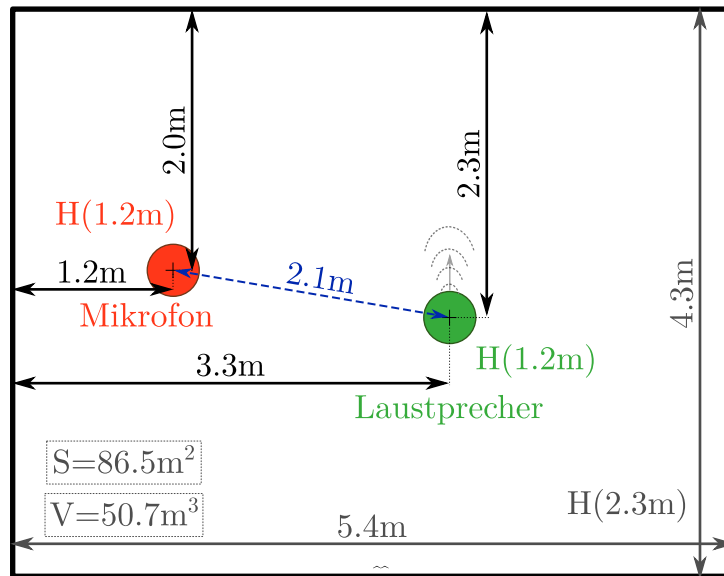


Abb. A.4: Anordnung von Lautsprecher und Mikrofon im Raum

A.6 Probleme und Artefakte der Messkette

Bei diesem Fehler tritt bei gleichzeitigem Start des Rauschsignals und aufzeichnen des Mikrofons auf. Wie in [Abb. A.5](#) zu erkennen ist tritt zunächst eine abklingende Funktion auf, bevor bei 0,5s die zu erwartende Rauschaufzeichnung startet. Der Zeitversatz von Soundkarten zwischen

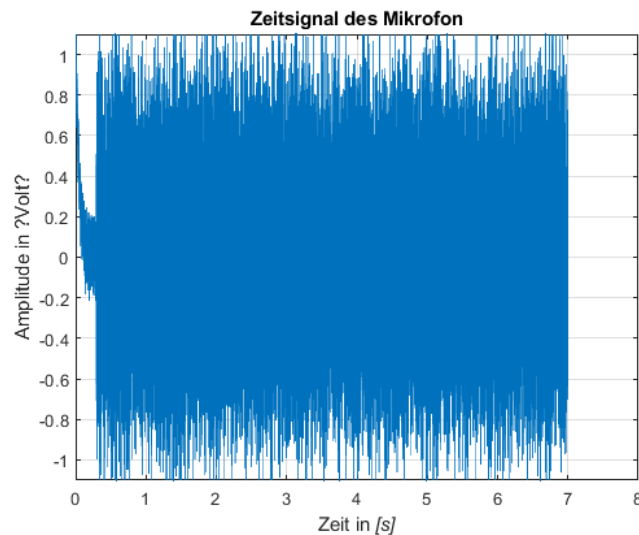


Abb. A.5: Mikrofon-Aufnahmefehler

den Ein- und Ausgabesignalen ist bekannt. Jedoch war ich verblüfft um Abstände es sich handelt, wie in [Abb. A.6](#) gezeigt.

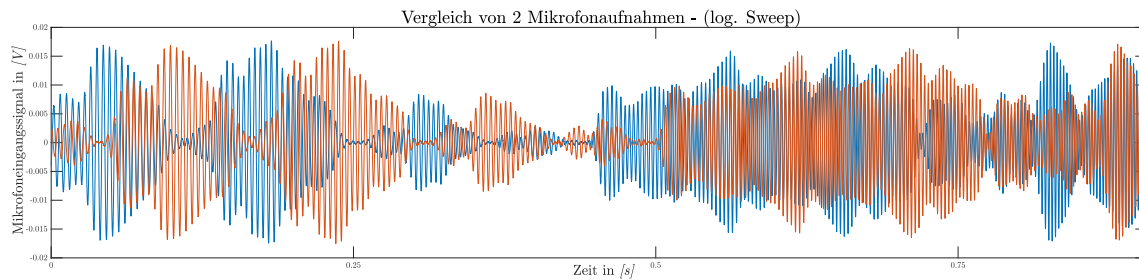


Abb. A.6: Zeitversatz bei Aufzeichnungen

Ein interessanter Fehler trat durch den Anschluss eines externen Monitors auf. Häufig kann durch die Netzspannung ein Vielfaches der Grundfrequenz 50 Hz in einem Audiosignal (auch Netzbrummen genannt) auftreten. Interessant an [Abb. A.7](#) ist jedoch klar die Frequenz 1000 Hz und Vielfache aufgetreten. Besonders überraschend war, dass diese Spannung nicht über das Stromnetz, sondern durch das HDMI-Kabel übertragen wurde.

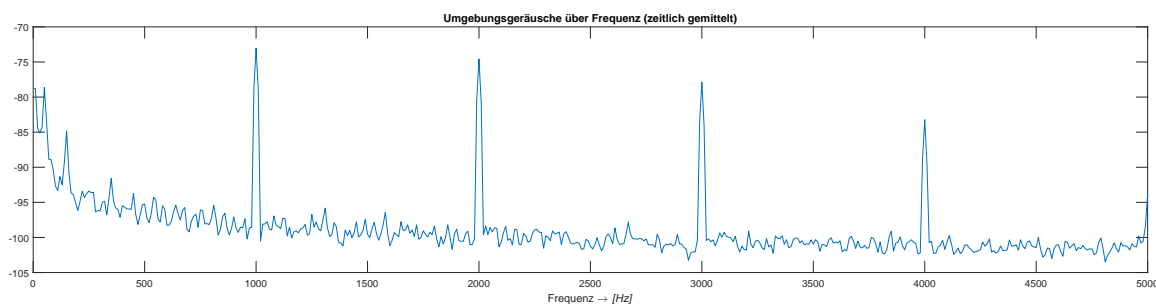


Abb. A.7: Störgeräusche im Audiosignal

Nicht alle Fehler konnten hier aufgezeigt werden, da Sie jeweils nur einmalig auftraten und nicht dokumentiert sind.

Abbildungsverzeichnis

2.1	Impulsantwort im diskreten Zeit- und Bildbereich [Grü14, Bild 4.12]	2
2.2	Amplitudengang bei Abweichung vom Einheitsimpuls [Grü14, Bild 4.13]	2
3.1	Vermessene Raum mit Inhalt und Messtechnik	9
3.2	Messequipment - Eingänge und Ausgänge	10
3.3	zeitlicher Versatz der Mikrofonsignale infolge Soundkarte (vorher - nachher) . . .	13
3.4	schematisches Vorgehen zur zeitlichen Mittelung der Mikrofonaufnahmen	14
3.5	gemitteltes Mikrofonsignal über die Messzeit	14
3.6	Raumimpulsantwort	15
3.7	ETC	16
3.8	EDC	18
3.9	Wasserfall-Diagramm	19
3.10	REW-Messung	20
A.1	frequenzabhängiges SPL der REW-Messung im Raum	27
A.2	Wasserfalldiagramm mit 400 ms Zeitraum	27
A.3	Darstellung des Zeitversatzes der Soundkarte	29
A.4	Anordnung von Lautsprecher und Mikrofon im Raum	30
A.5	Mikrofon-Aufnahmefehler	30
A.6	Zeitversatz bei Aufzeichnungen	31
A.7	Störgeräusche im Audiosignal	31

Literaturverzeichnis

- [Deu09] DEUTSCHES INSTITUT FÜR NORMUNG E.V.: *Messung von Parametern der Raumakustik – Teil 1: Aufführungsräume (ISO 3382-1:2009)*; Oktober 2009
- [Fro18] FROST, Ira: *Einfache lineare Regression: Die Grundlage für komplexe Regressionsmodelle verstehen*. Wiesbaden : Springer VS, 2018 (essentials). <http://dx.doi.org/10.1007/978-3-658-19732-2>. <http://dx.doi.org/10.1007/978-3-658-19732-2>. – ISBN 9783658197315
- [Grü14] GRÜNIGEN, Daniel C.: *Digitale Signalverarbeitung: Mit einer Einführung in die kontinuierlichen Signale und Systeme*. 5., neu bearb. Aufl. München : Fachbuchverl. Leipzig im Carl Hanser Verl., 2014 (Hanser eLibrary). <http://dx.doi.org/10.3139/9783446439917>. <http://dx.doi.org/10.3139/9783446439917>. – ISBN 9783446440791
- [RCI11] RAHMAN, Abdul Ghaffar A. ; CHAO, Ong Z. ; ISMAIL, Zubaidah: Effectiveness of Impact-Synchronous Time Averaging in determination of dynamic characteristics of a rotor dynamic system. In: *Measurement* 44 (2011), Nr. 1, 34–45. <http://dx.doi.org/10.1016/j.measurement.2010.09.005>. – DOI 10.1016/j.measurement.2010.09.005. – ISSN 0263–2241
- [Sch65] SCHROEDER, M. R.: New Method of Measuring Reverberation Time. In: *The Journal of the Acoustical Society of America* 37 (1965), Nr. 3, S. 409–412. <http://dx.doi.org/10.1121/1.1909343>. – DOI 10.1121/1.1909343. – ISSN 0001–4966
- [Ven13] VENNERØD, Jakob: *The Hard Case - Improving Room Acoustics in Cuboid Rooms by using Diffusors: Scale Model Measurements*. Unpublished, 2013. <http://dx.doi.org/10.13140/2.1.1525.4721>. <http://dx.doi.org/10.13140/2.1.1525.4721>