

# Raumimpulsantwortmessung mit kompakter kugelförmiger Lautsprecheranordnung

Seminararbeit aus Algorithmen in Akustik und Computermusik 2, SE

Robert Baumgartner  
Katharina Egger

Betreuung: Dr. Franz Zotter, DI Matthias Frank  
Graz, 4. April 2011



institut für elektronische musik und akustik



## Zusammenfassung

Um bei Raumimpulsantwortmessungen auf die zugrunde liegenden Pfade der frühen Reflexionen rückschließen zu können, werden in dieser Arbeit erstmals Raumimpulsantworten für einen ganzen Satz von Abstrahlungsmustern gemessen. Diese entsprechen so genannten Kugelflächenfunktionen (eng.: spherical harmonics) und können von dem verwendeten Lautsprecherarray (Ikosaeder-Lautsprecher) wiedergegeben werden. Mit dem so ermittelten Datensatz kann über eine MVDR-Optimierung ein Abstrahlungsmuster generiert werden, sodass ein bestimmter Zeitpunkt der Raumimpulsantwort best möglich angeregt wird und dabei die Abstrahlungsrichtung des gesuchten Reflexionspfades anzeigt. Mit Hilfe von MATLAB wurde für diese Anwendung eine Benutzeroberfläche implementiert, welche eine intuitive Auswertung und Visualisierung der Ergebnisse ermöglicht.

## Inhaltsverzeichnis

<b>1</b>	<b>Motivation</b>	<b>4</b>
<b>2</b>	<b>Schallfeldbeschreibung mit Kugelflächenfunktionen</b>	<b>6</b>
2.1	Lösung der Wellengleichung in Kugelkoordinaten . . . . .	6
2.2	Sphärische Harmonische Transformation ( <i>SHT</i> ) für äußeres Randwertproblem . . . . .	9
2.3	Diskrete Sphärische Harmonische Transformation ( <i>DSHT</i> ) für äußeres Randwertproblem . . . . .	10
<b>3</b>	<b>Messaufbau</b>	<b>13</b>
3.1	Ikosaeder-Lautsprecher . . . . .	13
3.2	Geometrien . . . . .	13
3.3	Messsignal . . . . .	15
<b>4</b>	<b>Implementierung in MATLAB</b>	<b>16</b>
4.1	Berechnung der einzelnen Impulsantworten . . . . .	16
4.2	Bandbegrenzung . . . . .	17
4.3	Minimum-Variance-Distortionless-Response (MVDR) . . . . .	17
4.4	Graphical User Interface . . . . .	19
4.4.1	Mean Energy Time Curve . . . . .	19
4.4.2	Optimierte Impulsantwort . . . . .	20
4.4.3	Balloonplot . . . . .	20
<b>5</b>	<b>Ergebnisse</b>	<b>22</b>
<b>6</b>	<b>Ausblick</b>	<b>26</b>
<b>A</b>	<b>MATLAB Source Code</b>	<b>28</b>
<b>B</b>	<b>Messequipment</b>	<b>32</b>

# 1 Motivation

Um das akustische Übertragungsverhalten eines Raumes zu ermitteln, kann in der Regel angenommen werden, dass es sich bei dem Raum um ein lineares, zeitinvariantes System handelt. Messungen der Raumimpulsantwort zu unterschiedlichen Zeitpunkten führen in diesem Sinne zu qualitativ identischen Messergebnissen und identifizieren somit die Übertragungseigenschaften des Raumes. Üblicherweise werden diese Messungen mit omnidirektionalen Schallquellen, wie exemplarisch in Abbildung (1a) dargestellt, durchgeführt.



(a) Omnidirektionale Schallquelle für gewöhnliche Messmethode.



(b) Ikosaeder-Lautsprecher für erweiterte Messmethode.

Abbildung 1: Vergleich von Schallquellen zur Messung von Raumimpulsantworten.

Diskrete Reflexionspfade treten in der resultierenden Impulsantwort als Energiespitzen auf. Bei den herkömmlichen Verfahren kann die Richtung des am Messmikrofon eintreffenden Schalls jedoch nur aus geometrischen Überlegungen ermittelt werden.

Misst man hingegen nicht nur die Impulsantwort für omnidirektionale Schallausbreitung sondern auch für andere, genau definierte Abstrahlungsmuster, zum Beispiel für einen Satz von gebündelten Schallstrahlen in alle Richtungen, oder, wie in unserem Fall, für die so genannten Kugelflächenfunktionen, so können im Anschluss an die Messungen durch Superposition Impulsantworten für nahezu beliebige Abstrahlungsmuster generiert werden. Dies hat zur Folge, dass aufgrund der Zeitinvarianz Abstrahlungsmuster synthetisiert werden können, die eine bestimmte diskrete Reflexion am besten hervorrufen können und folglich die Abstrahlungsrichtung des gesuchten Reflexionspfades aufzeigen. Für diese Messmethode ist eine kompakte Lautsprecheranordnung notwendig, die in der Lage ist Abstrahlungen in Form von Kugelflächenfunktionen zu generieren. Für die durchgeführten Messungen wurde der Ikosaeder-Lautsprecher entsprechend Abbil-

dung (1b) verwendet.

In Abschnitt 2 werden zunächst die theoretischen Hintergründe erklärt, die eine holographische Beschreibung von Schallfeldern mit Hilfe von Kugelflächenfunktionen ermöglichen. Außerdem wird auf Randbedingungen und örtliche Diskretisierungsprobleme bei der Abstrahlungssynthese eingegangen. Der verwendete Messaufbau und Details der Messdurchführung werden in Abschnitt 3 präsentiert. Die erworbenen Messdaten werden mit MATLAB ausgewertet und über eine entwickelte Benutzeroberfläche visualisiert. Einzelheiten zu dieser Implementierung und ein Paar untersuchte Fallbeispiele sind in den darauf folgenden Abschnitten 4 und 5 zu finden.

## 2 Schallfeldbeschreibung mit Kugelflächenfunktionen

### 2.1 Lösung der Wellengleichung in Kugelkoordinaten

Ein quellenfreies, verlustloses Schallfeld kann mit einer partiellen, homogenen Wellengleichung der Form

$$\Delta p(\mathbf{r}, t) = \frac{1}{c^2} \frac{\partial^2 p(\mathbf{r}, t)}{\partial t^2} \quad (1)$$

beschrieben werden. Dabei repräsentiert  $p(\mathbf{r}, t)$  den Schalldruckverlauf am Punkt  $\mathbf{r} = (x \ y \ z)^T$ ,  $c$  die Schallgeschwindigkeit und  $\Delta$  den Laplace-Operator, der in kartesischen Koordinaten die zweite örtliche Ableitung wie folgt definiert:

$$\Delta = \nabla^T \nabla = \partial^2 / \partial x^2 + \partial^2 / \partial y^2 + \partial^2 / \partial z^2. \quad (2)$$

Da bei der theoretischen Beschreibung von akustischen Abstrahlungsprozessen meist Punktquellen superponiert werden empfiehlt sich aufgrund der Rotationssymmetrie eine Beschreibung in Kugelkoordinaten. Der Azimut  $0 \leq \varphi < 2\pi$  und der Zenitwinkel  $0 \leq \vartheta < \pi$  können hierbei im Richtungsvektor  $\boldsymbol{\theta} = \mathbf{r} / \|\mathbf{r}\|$  zusammengefasst werden.

$$\mathbf{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = r \boldsymbol{\theta} = r \begin{pmatrix} \cos(\varphi) \sin(\vartheta) \\ \sin(\varphi) \sin(\vartheta) \\ \cos(\vartheta) \end{pmatrix}, \quad \begin{pmatrix} r \\ \varphi \\ \vartheta \end{pmatrix} = \begin{pmatrix} \sqrt{x^2 + y^2 + z^2} \\ \arctan\left(\frac{y}{x}\right) \\ \arctan\left(\frac{\sqrt{x^2 + y^2}}{z}\right) \end{pmatrix} \quad (3)$$

In Kugelkoordinaten ergibt sich somit für den Laplace-Operator folgender Ausdruck (vgl. [Zot09]):

$$\Delta = \frac{1}{r^2} \frac{\partial^2 r^2}{\partial r^2} + \frac{1}{r^2 \sin^2(\vartheta)} \frac{\partial^2}{\partial \varphi^2} + \frac{1}{r^2 \sin(\vartheta)} \frac{\partial}{\partial \vartheta} \left( \sin(\vartheta) \frac{\partial}{\partial \vartheta} \right). \quad (4)$$

Unter der Annahme von separierbaren Variablen (vgl. [Wil99]) kann die Wellengleichung (1) mit einem Produktansatz der Form  $p(\mathbf{r}, t) = R(r)\Theta(\vartheta)\Phi(\varphi)T(t)$  gelöst werden. Die winkelabhängigen Terme  $\Theta(\vartheta)$  und  $\Phi(\varphi)$  werden zusammengefasst und mit orthonormalen Kugelflächenfunktionen  $Y_n^m(\boldsymbol{\theta})$  dargestellt. Unter Berücksichtigung zahlreicher Bedingungen wie der Sommerfeld'schen Abstrahlungsbedingung auslaufender Wellen, der Periodizität im Azimuth und der Forderung nach Kausalität ergeben sich somit (vgl. [Zot11]) folgende physikalisch sinnvolle Lösungsschar:

$$p(kr, \boldsymbol{\theta}) = \sum_{n=0}^{\infty} \sum_{m=-n}^n [b_{nm} j_n(kr) + c_{nm} h_n^{(2)}(kr)] Y_n^m(\boldsymbol{\theta}), \quad (5)$$

$$\text{bzw. } v(kr, \boldsymbol{\theta}) = \frac{i}{\rho_0 c} \sum_{n=0}^{\infty} \sum_{m=-n}^n [b_{nm} j_n'(kr) + c_{nm} h_n'^{(2)}(kr)] Y_n^m(\boldsymbol{\theta}). \quad (6)$$

Der Einfachheit wegen wird von stationären, harmonischen Schwingungen ausgegangen und deshalb der komplexe Schwingungsterm  $e^{i\omega t}$  in obiger Gleichung weggelassen. Es werden die Koeffizienten  $b_{nm}$  und  $c_{nm}$  eingeführt und in weiterer Folge als Wellenspektrum bezeichnet. Bei den radialen Komponenten der Basislösungen handelt es sich um sphärische Besselfunktionen abgeleitet auftreten. Das sphärische Wellenspektrum  $\psi_n^m(kr)$  entspricht dem mit radialen Funktionen gewichteten Wellenspektrum und ergibt sich demnach aus:

$$\psi_n^m(kr) = b_{nm}j_n(kr) + c_{nm}h_n^{(2)}(kr). \quad (7)$$

Die reellwertigen, sphärischen Besselfunktionen erster Art beschreiben hierbei ausschließlich einlaufende Wellen, während ausstrahlende Wellen von den komplexen, sphärischen Hankelfunktionen erfasst werden.

Folgende Abbildung 2 zeigt die Verläufe der sphärischen Besselfunktionen erster Art  $j_n(kr)$  und zweiter Art  $y_n(kr)$ , welche häufig auch als Neumannfunktionen bezeichnet werden.

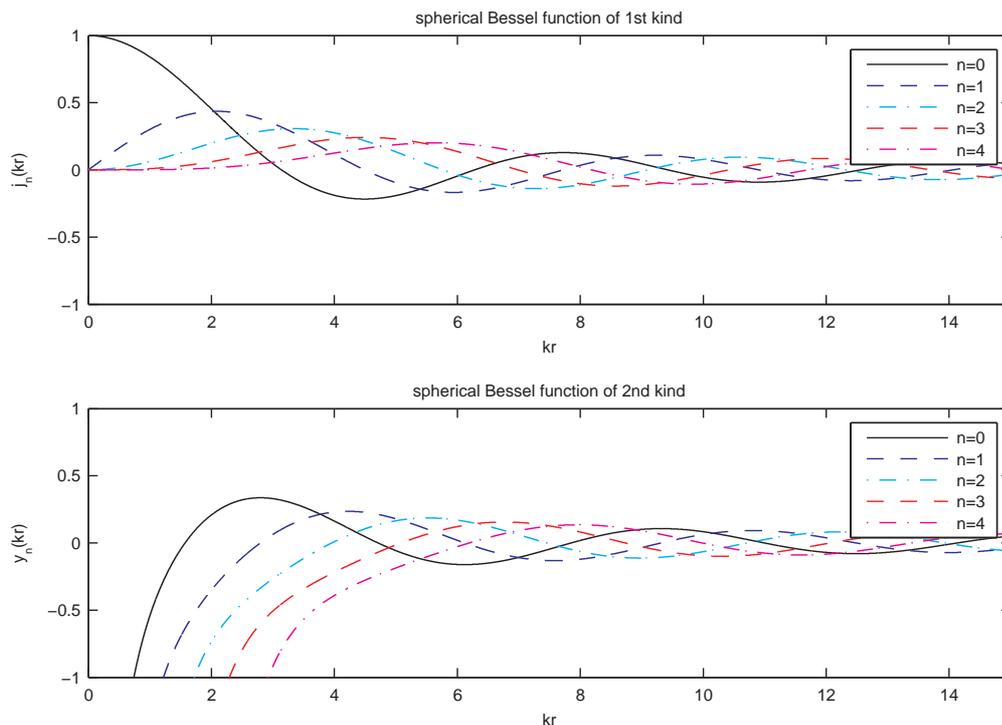


Abbildung 2: Sphärische Besselfunktionen erster und zweiter Art für  $n=0\dots4$ .

Der Realteil der sphärischen Hankelfunktion entspricht der sphärischen Besselfunktion erster Art und der Imaginärteil der sphärischen Neumannfunktion:

$$h_n^{(2)}(kr) = h_n^{(1)*} = j_n(kr) - iy_n(kr) \quad (8)$$

Wie der Verlauf der Neumannfunktion in Abbildung 2 zeigt, ergibt sich auch für die Hankelfunktion eine Singularität bei  $kr = 0$ . Außerdem lässt die Betragsdarstellung der

Hankelfunktion eine Unterscheidbarkeit zwischen Nah- und Fernfeld erkennen, wobei im Fernfeld der Betrag der Funktion die Ordnungsabhängigkeit verliert:

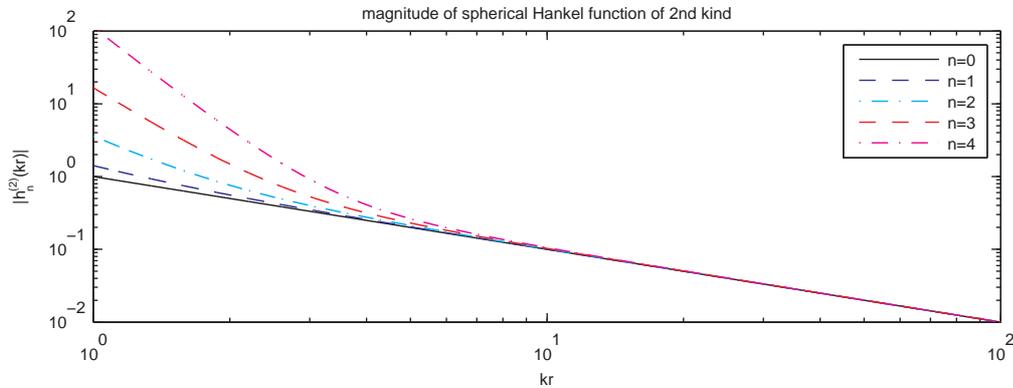


Abbildung 3: Betrag der sphärische Hankelfunktion zweiter Art für n=0...4.

Die orthonormalen Kugelflächenfunktionen  $Y_n^m(\theta)$  werden auch als Sphärische Harmonische (SH) bzw. Kugelharmonische bezeichnet und beruhen auf der Fourier-Zerlegung der Kugeloberfläche. Sie beschreiben den richtungsabhängigen Anteil der Basislösungen und werden bezüglich ihrer Ordnung  $n$  und ihrem Grad  $m$  unterteilt. Die Ordnung entspricht dabei der Anzahl an existierenden Knotenkreisen und der Grad der Anzahl an Knotenkreisen, die Nord- und Südpol schneiden. Bei einer Ordnung von  $n = N$  stehen insgesamt  $M = (N + 1)^2$  Funktionen zur Verfügung.

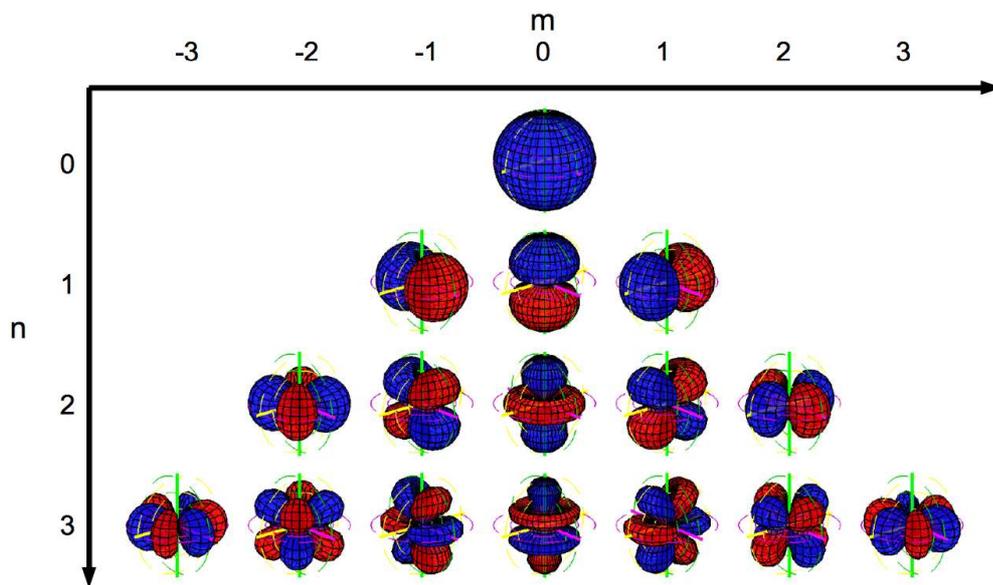


Abbildung 4: Kugelflächenfunktionen für n=0...3, Betrag distanzkodiert und Phasenlage farbkodiert; aus [Pom08].

## 2.2 Sphärische Harmonische Transformation ( $\mathcal{SHT}$ ) für äußeres Randwertproblem

Unter Vernachlässigung des Raumeinflusses kann bei Abstrahlungssynthesen von einem rein äußeren Randwertproblem, einer so genannten Dirichlet-Randwertbedingung (vgl. [Zot09]), ausgegangen werden. Dies bedeutet, dass sich entsprechend folgender Skizze alle Schallquellen im Inneren einer geschlossenen Kugeloberfläche mit dem Radius  $r_0$  befinden.

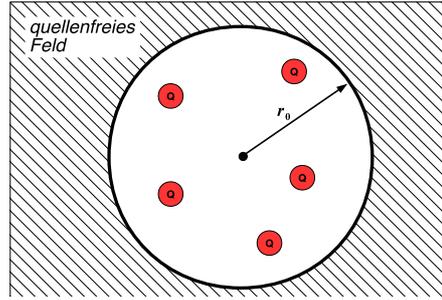


Abbildung 5: Skizze zu äußerem Randwertproblem.

Die Existenz von einlaufenden Wellen kann somit ausgeschlossen werden. Aus diesem Grund wird die entsprechende Komponente des Wellenspektrums  $b_{mn} \equiv 0$  gesetzt und die allgemeine Lösung (5) sowie die Zusammensetzung des sphärischen Wellenspektrums vereinfachen sich für diesen Spezialfall zu:

$$p(kr, \boldsymbol{\theta}) = \sum_{n=0}^{\infty} \sum_{m=-n}^n c_{mn} h_n^{(2)}(kr) Y_n^m(\boldsymbol{\theta}) \quad (9)$$

$$\psi_n^m(kr) = c_{nm} h_n^{(2)}(kr). \quad (10)$$

Möchte man den Schalldruckverlauf an einer beliebigen Position  $r' \geq r_0$  im Freifeld reproduzieren, muss die radiale Gewichtung durch die Hankelfunktion entsprechend angepasst werden:

$$\psi_n^m(kr') = \psi_n^m(kr_0) \frac{h_n^{(2)}(kr')}{h_n^{(2)}(kr_0)}. \quad (11)$$

Diese Filter werden üblicher Weise als sphärische Holografiefilter bezeichnet.

Das sphärische Wellenspektrum kann außerdem über eine Transformation, deren Basisfunktionen die Kugelflächenfunktionen darstellen, direkt aus dem Schalldruck ermittelt werden. Diese örtliche Transformation wird als Sphärische Harmonische Transformation ( $\mathcal{SHT}$ ) bezeichnet:

$$\psi_n^m(kr) = \mathcal{SHT}\{p(kr, \boldsymbol{\theta})\} := \int_{\mathbb{S}^2} p(kr, \boldsymbol{\theta}) Y_n^m(\boldsymbol{\theta}) d\boldsymbol{\theta} \quad (12)$$

$$p(kr, \boldsymbol{\theta}) = \mathcal{ISHT}\{\psi_n^m(kr)\} := \sum_{n=0}^{\infty} \sum_{m=-n}^n \psi_n^m(kr) Y_n^m(\boldsymbol{\theta}). \quad (13)$$

## 2.3 Diskrete Sphärische Harmonische Transformation (DSHT) für äußeres Randwertproblem

### Matrixnotation für begrenzte örtliche Auflösung

Da bei der praktischen Realisierung von kompakten Lautsprecheranordnungen zur holo-phonen Abstrahlungssynthese nicht beliebig viele Chassis auf der Kugeloberfläche verteilt werden können, kann auch nur bis zu einer bestimmten maximalen SH-Ordnung  $n = N$  synthetisiert bzw. kontrolliert wiedergegeben werden. Somit ergibt sich eine Gesamtanzahl an verfügbaren Kugelflächenfunktionen von  $M = (N + 1)^2$ , auf welche sich das örtliche Auflösungsvermögen beschränkt. In Hinblick auf die Implementierung der Filter empfiehlt sich für die Definition von Matrizen eine eindeutige Reihung der Kugelflächenfunktionen. So wird ein neuer Index  $\nu$  eingeführt, der die Kombinationsmöglichkeiten der Indizes  $n$  und  $m$  in linearer Abfolge repräsentiert:

$$\nu = \sum_{m'=-n}^m 1 + \sum_{n'=0}^{n-1} \sum_{m'=-n'}^{n'} 1 = n^2 + n + m + 1. \quad (14)$$

Die begrenzte Form des Wellenspektrums  $\mathbf{c}_N$  lässt sich dadurch wie folgt beschreiben:

$$\mathbf{c}_N := \begin{pmatrix} c_{0,0} \\ c_{1,-1} \\ c_{1,0} \\ c_{1,1} \\ \vdots \\ c_{N,N} \end{pmatrix}. \quad (15)$$

Dementsprechend kann auch eine Diagonalmatrix  $\mathbf{H}_N$  für die Hankelfunktion  $h_n^{(2)}(kr)$  eingeführt werden:

$$\mathbf{H}_N(kr) := \begin{pmatrix} h_0^{(2)}(kr) & 0 & 0 & \dots & 0 \\ 0 & h_1^{(2)}(kr) & 0 & \dots & 0 \\ 0 & 0 & h_1^{(2)}(kr) & \ddots & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_N^{(2)}(kr) \end{pmatrix}, \quad \dim\{\mathbf{H}_N(kr)\} = M \times M. \quad (16)$$

Es gilt zu beachten, dass die Hankelfunktionswerte nur von der Ordnung  $n$  nicht aber vom Grad  $m$  abhängen und somit pro Ordnung immer  $2n + 1$  identische Werte zu finden

sind.

Die vektorisierte Form des sphärischen Wellenspektrums berechnet sich folglich aus:

$$\boldsymbol{\psi}_N(kr) = \mathbf{H}_N \mathbf{c}_N. \quad (17)$$

Die SH-Matrix  $\mathbf{Y}_N$  repräsentiert die Kugelflächenfunktionen  $Y_n^m(\boldsymbol{\theta})$ , die an den  $L$  Lautsprecherpositionen  $\boldsymbol{\theta}^{(l)}$  abgetastet werden:

$$\mathbf{Y}_N := \text{mtx}_N\{Y_n^m(\boldsymbol{\theta}^{(l)})\} = \begin{pmatrix} Y_0^0(\boldsymbol{\theta}^{(1)}) & Y_1^{-1}(\boldsymbol{\theta}^{(1)}) & Y_1^0(\boldsymbol{\theta}^{(1)}) & \dots & Y_N^N(\boldsymbol{\theta}^{(1)}) \\ Y_0^0(\boldsymbol{\theta}^{(2)}) & Y_1^{-1}(\boldsymbol{\theta}^{(2)}) & Y_1^0(\boldsymbol{\theta}^{(2)}) & \dots & Y_N^N(\boldsymbol{\theta}^{(2)}) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Y_0^0(\boldsymbol{\theta}^{(L)}) & Y_1^{-1}(\boldsymbol{\theta}^{(L)}) & Y_1^0(\boldsymbol{\theta}^{(L)}) & \dots & Y_N^N(\boldsymbol{\theta}^{(L)}) \end{pmatrix},$$

$$\dim\{\mathbf{Y}_N\} = L \times M. \quad (18)$$

### Inversionsproblematik der *DSHT*

Mit Hilfe dieser Matrixschreibweise lässt sich nun in Anlehnung an (12) auch für örtlich diskrete Randwertprobleme eine Sphärische Harmonische Transformation definieren:

$$\mathbf{p}(kr) = \mathcal{IDSHT}\{\boldsymbol{\psi}_N(kr)\} := \mathbf{Y}_N \boldsymbol{\psi}_N(kr), \quad (19)$$

$$\boldsymbol{\psi}_N(kr) = \mathcal{DSHT}\{\mathbf{p}(kr)\} := \mathbf{Y}_N^\dagger \mathbf{p}_L(kr). \quad (20)$$

Da im Allgemeinen die Anzahl an darstellbaren Kugelflächenfunktionen ( $M$ ) nicht der Anzahl an Abtastpunkten ( $L$ ) entspricht, wird in der Analysegleichung (20) die Pseudoinverse, mit  $\dagger$  gekennzeichnet, verwendet. Es handelt sich bei der SH-Matrix  $\mathbf{Y}_N$  demnach in der Regel nicht um eine quadratische Matrix und somit kann auch keine direkte Inversion durchgeführt werden. Ein mögliches Verfahren zur Inversion von beliebig rechteckigen Matrizen stellt die Singulärwertzerlegung dar, welche auch in der späteren Implementierung zur Auswertung der Raumimpulsantwortmessung Verwendung findet.

## Singulärwertzerlegung

Mit Hilfe dieser Singulärwertzerlegung (SVD, eng.: singular value decomposition) kann die Matrix  $\mathbf{Y}_N$  als ein Produkt von drei speziellen, regulären Matrizen  $\mathbf{U}$ ,  $\mathbf{S}$  und  $\mathbf{V}$  dargestellt werden:

$$\mathbf{Y}_N = \mathbf{U} \mathbf{S} \mathbf{V}^* \quad (21)$$

$$\mathbf{S} = \text{diag}\{[s_i]_i\} = \begin{pmatrix} s_1 & 0 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & \cdots & 0 \\ 0 & 0 & s_3 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & s_M \end{pmatrix}. \quad (22)$$

Die Diagonalmatrix  $\mathbf{S}$  beinhaltet hierbei die Singulärwerte  $s_i$ , die vergleichbar mit Eigenwerten von quadratischen Matrizen, die Eigenschaften einer Matrix spezifizieren. Eine Inversion der Matrix  $\mathbf{Y}_N$  kann somit sehr einfach durchgeführt werden:

$$\mathbf{Y}_N^\dagger = \mathbf{V} \text{diag}\left\{\left[\frac{1}{s_i}\right]_i\right\} \mathbf{U}^*. \quad (23)$$

Da bei der Inversion der einzelnen Singulärwerte die Gefahr besteht, dass sehr geringe Singulärwerte bei der Inversion große numerische Schwierigkeiten in Bezug auf den Wertebereich erzeugen, wird in diesem Zusammenhang eine Konditionszahl  $\kappa$  definiert, welche das Verhältnis von maximalem zu minimalem Singulärwert angibt:

$$\kappa = \frac{s_{max}}{s_{min}}. \quad (24)$$

Ist die Dynamik der Singulärwerte und somit die Konditionszahl sehr groß, können einige Vektoren der  $\mathbf{Y}_N$ -Matrix nur schwer dargestellt werden. Um die Recheneffizienz zu erhöhen und gleichzeitig die Gefahr von rechnerischen Fehlinterpretationen zu verhindern, sollte deshalb die Konditionszahl möglichst gering gehalten werden, indem kleine Singulärwerte eliminiert werden.

Mit anderen Worten kann man sagen, dass die Inversion eines allgemeinen Gleichungssystems  $\mathbf{Y}\mathbf{x} = \mathbf{b}$ , wie auch bei der gewöhnlichen Division, einem Vergleich der in der Matrix enthaltenen Vektoren mit dem Vektor  $\mathbf{b}$  entspricht. Sei  $\mathbf{b}$  nun irgendein Vektor, der aus der Matrix  $\mathbf{Y}$  nicht hergestellt werden kann, so ist es geschickt, sich bei der Inversion nur auf in ausreichendem Maße vorhandene Komponenten zu stützen.

## 3 Messaufbau

### 3.1 Ikosaeder-Lautsprecher

Für die Messungen wurde die ikosaederförmige Lautsprecheranordnung mit  $L = 20$  Lautsprechern, welche am IEM entwickelt wurde, verwendet. Mittels sphärischem Beamforming ist damit eine beliebig-direktionale Schallabstrahlung, zum Beispiel in Form von Kugelflächenfunktionen, möglich. Ein stark gerichtetes Schallstrahlenbündel kann über die Superposition von Kugelflächenfunktionen in eine beliebige Richtung gelenkt werden und erlaubt damit das Anregen einer bestimmten Reflexion.

Das Lautsprecherarray ist laut [ZS07] in der Lage, in einem Frequenzbereich von etwa  $40 - 1500\text{Hz}$  Abstrahlungssynthese zu betreiben. Darüber hinaus wird der Einfluss der Richtwirkung der einzelnen Lautsprecher immer größer und somit nicht mehr vernachlässigbar.

Das örtliche Auflösungsvermögen der Lautsprecheranordnung wird durch die Anzahl  $L = 20$  der gleich verteilt angebrachten Lautsprechermembranen limitiert:

$$M = (N_{ico} + 1)^2 \stackrel{!}{<} 20 = L \Rightarrow N_{ico} = \lfloor \sqrt{L} - 1 \rfloor \quad (25)$$

Der Ikosaeder-Lautsprecher kann somit Kugelflächenfunktionen bis zur Ordnung  $N_{ico} = 3$  wiedergeben.

Der Lautsprecher wird mittels 42-Multipin-Connector mit den Verstärkern verbunden. Über eine MAD1 - Schnittstelle und DA - Konverter wird die Verbindung zum PC hergestellt. Die Signale werden mit Hilfe eines bestehenden *pd*-Patches (puredata) eingespielt.

### 3.2 Geometrien

Die Messungen wurden im IEM CUBE durchgeführt, ein zur Außenwelt schallgedämmter Musikwiedergabe-, Konzert- und Vortragsraum zugleich.

Lautsprecher und Mikrofon wurden in einer Höhe von  $1.36\text{m}$  montiert. Der Abstand zwischen Lautsprecher und Mikrofon und somit der Direktpfad betrug  $2.5\text{m}$ . Der genaue Messaufbau ist in Abbildung 6 ersichtlich.

Der zurückgelegte Weg der Bodenreflexion beträgt somit  $2 \cdot \sqrt{\left(\frac{2.5}{2}\right)^2 + 1.36^2} \cong 3.7\text{m}$ . Zeitlich gesehen kommt die Bodenreflexion  $t_d = \frac{3.7-2.5}{343} \cong 3.5\text{ms}$  später als der Direkt-schall am Mikrofon an.

Um eine zweite repräsentative Reflexion in der Impulsantwort erkennen zu können, wurde die Tafel  $1.8\text{m}$  parallel zur Achse Lautsprecher-Mikrofon positioniert.

Der Weg der dadurch entstandenen seitlichen Reflexion beträgt  $2 \cdot \sqrt{\left(\frac{2.5}{2}\right)^2 + 1.8^2} \cong$

4.4m. Der Zeitunterschied zwischen Direktpfad und seitlicher Reflexion ergibt sich somit zu  $t_d = \frac{4.4-2.5}{343} \cong 5.5ms$ .

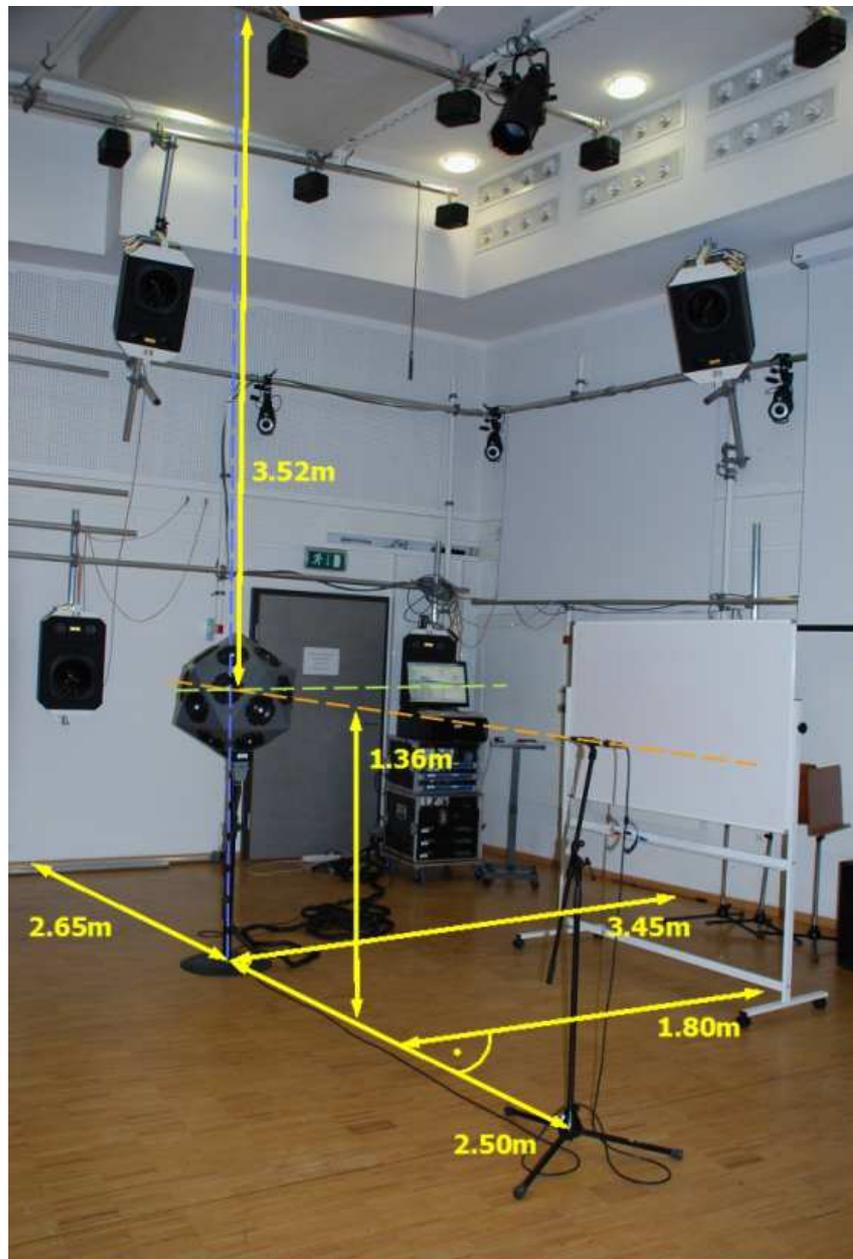


Abbildung 6: Messaufbau mit eingezeichneten Abständen

### 3.3 Messsignal

Die Raumimpulsantwort wurde mit Hilfe von exponentiellen Sweeps ermittelt. Wie in [Maj08] erläutert, bietet die Sweepmessung gegenüber der MLS-Messung den Vorteil der geringeren Empfindlichkeit in Hinblick auf harmonischen Verzerrungen, da hier der lineare Teil von etwaigen harmonischen Obertönen getrennt erfasst wird. Der Sweep ergibt sich zu (vgl. [Maj08]):

$$x(t) = \sin[A(e^{\frac{t}{\tau}} - 1)] \quad (26)$$

$$\text{mit } A = \frac{T \cdot \omega_1}{\ln \frac{\omega_2}{\omega_1}} \text{ und } \tau = \frac{T}{\ln \frac{\omega_2}{\omega_1}}$$

Die Start-  $f_1$  und Stopfrequenz  $f_2$  wurden wie folgt gewählt:

$$f_1 = 50 \text{ Hz}, \quad f_2 = \frac{f_s}{2}, \\ f_s = 44100 \text{ Hz.}$$

Das Sweepsignal wird nun mit dem Abstrahlmuster der jeweiligen Kugelflächenfunktionen beaufschlagt und für jede Kugelflächenfunktion separat wiedergegeben. Es werden daher  $M = 16$  Messungen der Raumantwort durchgeführt.

Diese aufgenommenen Signale bilden wiederum die Grundlage für die entwickelte MATLAB Simulation.

## 4 Implementierung in MATLAB

Ziel der Implementierung ist es, die Raumimpulsantwort hinsichtlich der bestehenden Reflexionspfade im Raum analysieren zu können. Die einzelnen Reflexionspfade sollen in Hinblick auf deren Abstrahlungsrichtung nachvollzogen und anschaulich dargestellt werden können. Die Auswahl der gesuchten Reflexion soll hierbei durch Festlegung eines bestimmten Zeitpunkts von dem Benutzer vorgenommen werden.

### 4.1 Berechnung der einzelnen Impulsantworten

Zu Beginn muss für jede Kugelflächenfunktion die entsprechende Impulsantwort berechnet werden.

Von den aufgenommenen Raumantworten ausgehend können die einzelnen Impulsantworten  $r_\nu(t)$  mit Hilfe der diskreten Fouriertransformation berechnet werden. Mit  $x(t)$  (Spektrum:  $X(\omega)$ ) nach (26) und  $y_\nu(t)$  ( $Y_\nu(\omega)$ ) als zugehöriges gemessenes Antwortsignal ergibt sich:

$$y_\nu(t) = x(t) * r_\nu(t) \quad (27)$$

$$Y_\nu(\omega) = X(\omega) \cdot R_\nu(\omega) \quad (28)$$

$$R_\nu(\omega) = Y_\nu(\omega) \cdot X^{-1}(\omega) \quad (29)$$

$$r_\nu(t) = \mathcal{IFFT}(R_\nu(\omega)) \quad (30)$$

$$\nu = 1 \dots M = 16$$

Durch Superposition der erhaltenen zeitdiskreten Impulsantworten  $r_\nu[n] = r_\nu(\frac{n}{f_s})$  können nun Impulsantworten für nahezu beliebige Abstrahlmuster  $r_\xi[n]$  generiert werden, indem man sie entsprechend der Anteile der einzelnen Kugelflächenfunktionen mit den Koeffizienten  $c_\nu$  gewichtet.

$$\tilde{\mathbf{r}}_\nu = [r_\nu[n]]_{\nu=1\dots M} \quad (31)$$

$$\mathbf{R} = [ \tilde{\mathbf{r}}_1 \quad \tilde{\mathbf{r}}_2 \quad \dots \quad \tilde{\mathbf{r}}_{16} ] \quad (32)$$

$$\mathbf{c} = \text{vec}\{c_\nu\} = [ c_1 \quad c_2 \quad \dots \quad c_{16} ]^T \quad (33)$$

$$\tilde{\mathbf{r}}_\xi = \mathbf{R}\mathbf{c} \quad (34)$$

Soll die Impulsantwort mit dem Ergebnis einer herkömmlichen Messung mit omnidirektionaler Schallquelle verglichen werden, betrachtet man nur die Impulsantwort für die Kugelflächenfunktion nullter Ordnung  $r_1[n]$ . Dies entspricht der Multiplikation mit den Koeffizienten

$$c_\nu = \begin{cases} 1 & \nu = 1 \\ 0 & \nu \neq 1 \end{cases} \quad (35)$$

## 4.2 Bandbegrenzung

Im Anschluss werden die Impulsantworten  $r_\nu[n]$  bandbegrenzt und auf  $f'_s = 2kHz$  unterabgetastet.

Dies geschieht, um die entstehenden Artefakte durch Auftreten räumlichen Aliasing bei hohen Frequenzen zu vermeiden. Grund hierfür ist die diskrete Abtastung der Kugeloberfläche. Die endliche Anzahl der Lautsprecher mit endlichem Abstand zueinander stellt eine nicht ausreichend feine, räumliche Diskretisierung der Kugeloberfläche dar. Bei hohen Frequenzen kommt es daher zu störenden Interferenzmustern und es kann die gewünschte Abstrahlcharakteristik nicht mehr vollständig nachgebildet werden.

## 4.3 Minimum-Variance-Distortionless-Response (MVDR)

Um ein Abstrahlungsmuster zu generieren, welches einen bestimmten Zeitpunkt der Raumimpulsantwort bestmöglich anregt, werden die Koeffizienten mit Hilfe der Minimum-Variance-Distortionless-Response - Methode optimiert.

Ziel der Optimierung ist es, durch geeignete Wahl der Koeffizienten  $\mathbf{c}$  die Energie der Impulsantwort in einem Abtastwert zu maximieren und gleichzeitig die Energie in allen anderen Abtastwerte zu minimieren. Die resultierenden Koeffizienten  $\mathbf{c}$  beschreiben dann die Abstrahlcharakteristik, welche die Schallquelle haben müsste, um den gewählten Zeitpunkt und somit den zugehörigen Reflexionspfad in der Impulsantwort bestmöglich anzuregen.

Die Energie der resultierenden Impulsantwort zum ausgewählten Zeitpunkt  $T$  wird durch eine Nebenbedingung in (36) auf  $r_\xi[n = T] = 1$  gesetzt (mit MATLAB-Indizierung):

$$\mathbf{r}^T \mathbf{c} = 1 \text{ mit } \mathbf{r}^T = \mathbf{R}(T, :) \quad (36)$$

Ziel ist es, unter dieser Nebenbedingung die Gesamtenergie im Sinne des LS-Verfahrens (*Least Squares*) zu minimieren:

$$\|\mathbf{R}\mathbf{c}\|^2 = \mathbf{c}^T \mathbf{R}^T \mathbf{R} \mathbf{c} \quad (37)$$

Um die Nebenbedingung zu berücksichtigen, wird der Lagrange-Multiplikator  $\lambda$  eingeführt. Die endgültige Kostenfunktion ergibt sich dann zu:

$$J(\mathbf{c}) = \mathbf{c}^T \mathbf{R}^T \mathbf{R} \mathbf{c} + \lambda(\mathbf{c}^T \mathbf{r} - 1) \quad (38)$$

Mit Hilfe der ersten Ableitung der Kostenfunktion  $J(\mathbf{c})$  nach  $\mathbf{c}$  und anschließendem Nullsetzen wird das Minimum von (38) und somit der optimierte Koeffizientensatz  $\mathbf{c}_{opt}$  ermittelt.

$$\begin{aligned} \frac{\partial}{\partial \mathbf{c}^T} J(\mathbf{c}) &\stackrel{!}{=} 0 \\ \mathbf{R}^T \mathbf{R} \mathbf{c} + \lambda \mathbf{r} &= 0 \end{aligned}$$

$$\mathbf{c}_{opt} = -\lambda(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{r} \quad (39)$$

Dieses Ergebnis enthält jedoch noch den unbekannt Parameter  $\lambda$ . Um eine zweite Gleichung für diese Unbekannte zu bekommen wird die Kostenfunktion nun analog zu (39) bezüglich  $\lambda$  differenziert:

$$\begin{aligned} \frac{\partial}{\partial \lambda} J(\mathbf{c}) &\stackrel{!}{=} 0 \\ \mathbf{c}^T \mathbf{r} - 1 &= 0 \\ \mathbf{r}^T \mathbf{c} &= 1. \end{aligned}$$

$\mathbf{c}_{opt}$  aus (39) kann nun eingesetzt werden und liefert einen von  $\mathbf{c}$  unabhängigen Ausdruck für  $\lambda$ :

$$\begin{aligned} -\lambda \mathbf{r}^T (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{r} &= 1 \\ \lambda &= \frac{-1}{\mathbf{r}^T (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{r}}. \end{aligned}$$

Dies wiederum in (39) eingesetzt ergibt die optimalen Koeffizienten:

$$\mathbf{c}_{opt} = \frac{(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{r}}{\mathbf{r}^T (\mathbf{R}^T \mathbf{R})^{-1} \mathbf{r}}. \quad (40)$$

Um die Matrix  $\mathbf{R}$  zu invertieren wird zunächst die Singulärwertzerlegung entsprechend Abschnitt 2.3 durchgeführt. Es hat sich gezeigt, dass eine Konditionszahl von  $\kappa = 15$  aussagekräftige Ergebnisse liefert. Aufgrund der relativ großen Pegelunterschiede der Abtastwerte kann der Wert zum gewünschten Zeitpunkt bereits relativ klein werden. Deshalb ist eine solch hohe Konditionszahl in Anbetracht der Signaldynamik durchaus sinnvoll. Ein exemplarischer Vergleich unterschiedlicher Konditionszahlen folgt in Abschnitt 4.4.3.

Die optimierte Impulsantwort  $\tilde{\mathbf{r}}_\xi$  berechnet sich dann wie in (34) zu:

$$\tilde{\mathbf{r}}_\xi = \mathbf{R} \mathbf{c}_{opt}.$$

## 4.4 Graphical User Interface

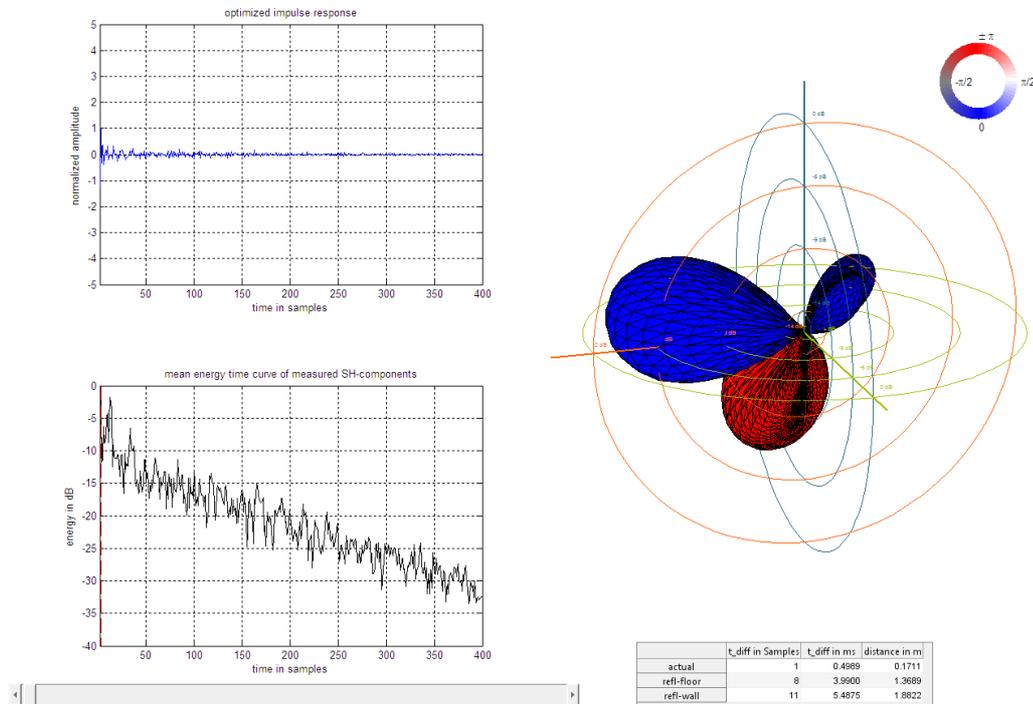


Abbildung 7: Benutzeroberfläche

Abbildung 7 zeigt die Benutzeroberfläche, welche sich in folgende drei Abschnitte gliedert.

### 4.4.1 Mean Energy Time Curve

Im linken unteren Fenster ist die mittlere Energie  $E$  der gemessenen SH-Komponenten logarithmisch über der Zeit  $t$  aufgetragen.

$$E(t) = 10 \cdot \log \left( \frac{\frac{\sum_{\nu=1}^M r_{\nu}^2(t)}{M}}{\max(\frac{\sum_{\nu=1}^M r_{\nu}^2(t)}{M})} \right) \quad (41)$$

Dieses Fenster dient gleichzeitig zur Eingabe des Optimierungszeitpunktes  $T$  der MVDR (vgl. dazu Abschnitt 4.3). Dieser Zeitpunkt kann entweder über Setzen des Cursors im Diagramm, oder mit Hilfe des Sliders darunter eingestellt werden. Außerdem ist im rechten unteren Eck eine Tabelle dargestellt, die zum einen die aktuell gewählte Laufzeitverzögerung in Samples und Millisekunden angibt und zum zweiten als Anhaltspunkte die entsprechenden Werte der beiden repräsentativsten Reflexionen anführt.

Es hat wenig Sinn, den Optimierungszeitpunkt auf einen Zeitpunkt der Impulsantwort zu setzen, an dem ohnehin keine oder nur wenig Energie vorhanden ist. Es wäre unmöglich, die Energie genau an diesem Zeitpunkt der Raumimpulsantwort zu normieren. Um geeignete Stellen leichter auswählen zu können, wird in dem Diagramm die Energie über der Zeit statt der Impulsantwort dargestellt.

#### 4.4.2 Optimierte Impulsantwort

Im Diagramm darüber wird die hinsichtlich des links unten gewählten Zeitpunktes optimierte Raumimpulsantwort  $r_\xi$  dargestellt. Bei erfolgreicher Optimierung tritt der ausgewählte Abtastwert als globales Maximum der Impulsantwort mit dem normalisierten Wert 1 hervor. Ist dies nicht der Fall, muss der Optimierungszeitpunkt variiert werden, um ein aussagekräftiges Abstrahlmuster zu erhalten.

#### 4.4.3 Balloonplot

Auf der rechten Seite wird der so genannte Balloonplot abgebildet. Dieses Diagramm stellt das Abstrahlmuster dar, welches auf Grund der berechneten Gewichtungskoeffizienten  $c_{opt}$  der SH-Komponenten erzeugt werden müsste, um die optimierte Impulsantwort  $r_\xi[n]$  zu erhalten. Die Zuordnung der Achsen ist in Abbildung 8 farblich dargestellt.

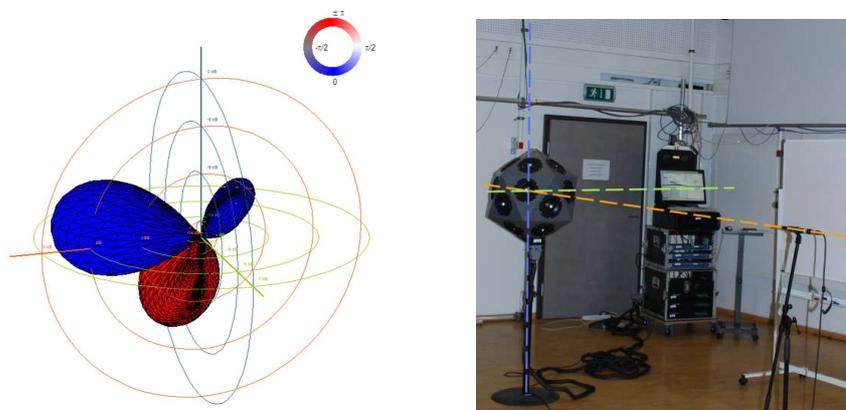


Abbildung 8: Balloonplot

Die Betragsinformation steckt im Abstand zum Koordinatenursprung des Diagramms. Dieser Schalldruckbetrag in dB wird wiederum nach (41) energiegewichtet, um nur die sinnvollen Abstrahlmuster hervorzuheben. Die Phase wird farbkodiert dargestellt, genauer gesagt lassen die beiden Farben Rot und Blau Gegenphasigkeiten erkennen. Herrscht somit im blauen Bereich ein Druckmaximum, liegt gleichzeitig im roten Teil ein Druckminimum vor und umgekehrt.

Weiters wird die Dynamik der Darstellung auf 10dB begrenzt. Dies erlaubt eine deutlichere Erkennbarkeit der Strahlrichtungen, da kleinere Nebenkeulen nicht mehr dargestellt

werden. Um ein Ablesen des Betrags zu erleichtern, sind Ringe im 3dB Abstand eingezeichnet.

Sowohl die Dynamikbegrenzung als auch die Konditionszahl  $\kappa$  sind Designparameter, deren Variation (vor allem) im Balloonplot optische Veränderungen mit sich bringen. Diese können vom Benutzer im MATLAB Code einfach angepasst werden.

Eine Änderung der Konditionszahl  $\kappa$  hat qualitative Auswirkung auf die Berechnung der Koeffizienten. Abbildung 9 zeigt den Balloonplot des Direktpfades ( $T = 1$ ) für drei verschiedene Werte von  $\kappa$ .

Das linke Bild zeigt das Abstrahlmuster für  $\kappa = 400$ , also praktisch ohne Regularisierung, das heißt es werden fast keine Singulärwerte eliminiert. Es können sehr viele Nebenkeulen erkannt werden. Das rechte Bild zeigt das Abstrahlmuster für  $\kappa = 5$ . Hier ist eindeutig zu erkennen, dass zu stark konditioniert wurde. Der Beam in Direkttrichtung wird breiter und verändert seine Richtung. Das mittlere Bild für  $\kappa = 15$  zeigt den Balloonplot im gut konditionierten Fall. Dieser Wert wurde empirisch ermittelt.

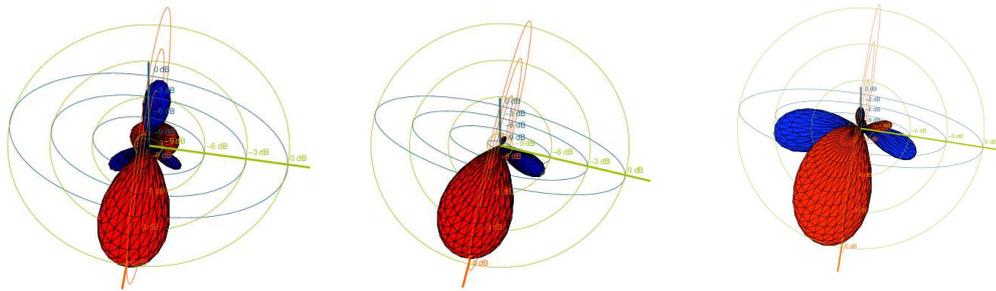


Abbildung 9: Variation der Konditionszahl:  $\kappa = 400$ ,  $\kappa = 15$  und  $\kappa = 5$  (von links nach rechts)

## 5 Ergebnisse

### a) Direktpfad

Will man den Direktanteil in der Impulsantwort maximieren, das heißt die Energie im ersten Sample, muss die Abstrahlrichtung der Schallquelle direkt auf das Mikrofon fokussiert werden. Wie in Abbildung 10 sehr gut erkennbar, werden die Komponenten der Kugelharmosischen wie zu erwarten so gewichtet, dass ein stark ausgeprägter Beam in die Richtung des Direktpfades zeigt.

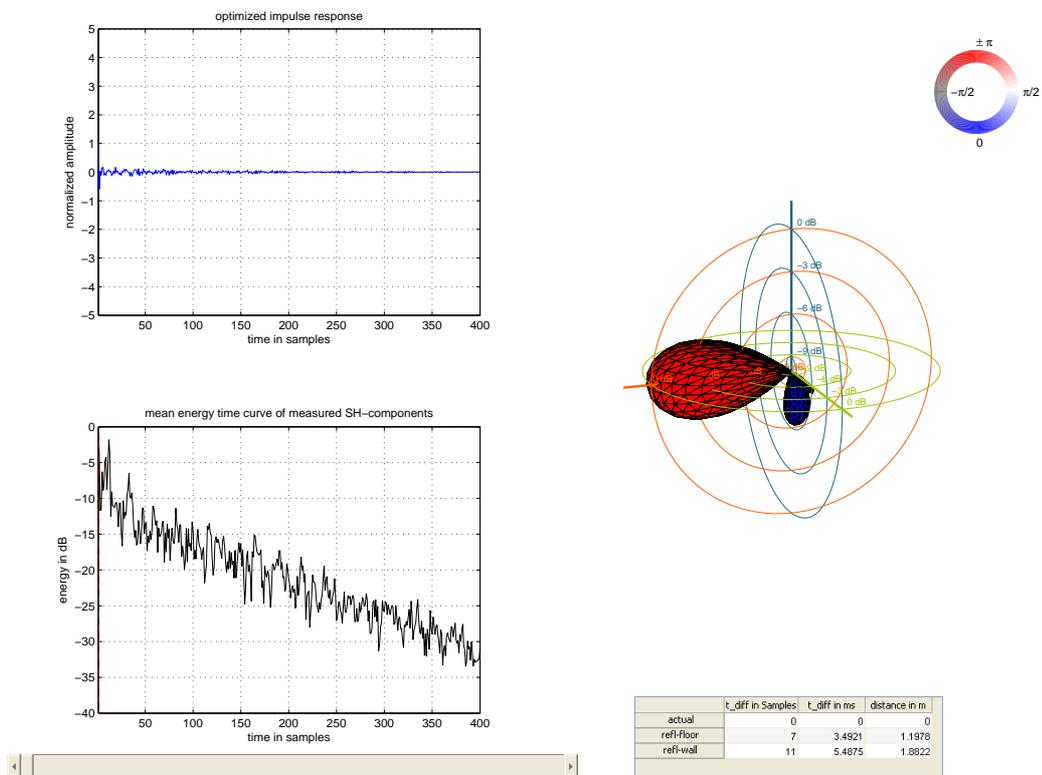


Abbildung 10: Direktpfad

b) Bodenreflexion

Die erste eintreffende Reflexion entspricht der Bodenreflexion. Leitet man den Schallstrahl in diese Richtung, wird die Reflexion besonders gut angeregt und somit ergibt sich in der resultierenden Impulsantwort links oben ein globales Energiemaximum bei der entsprechenden Laufzeitverzögerung (siehe Abb. 11).

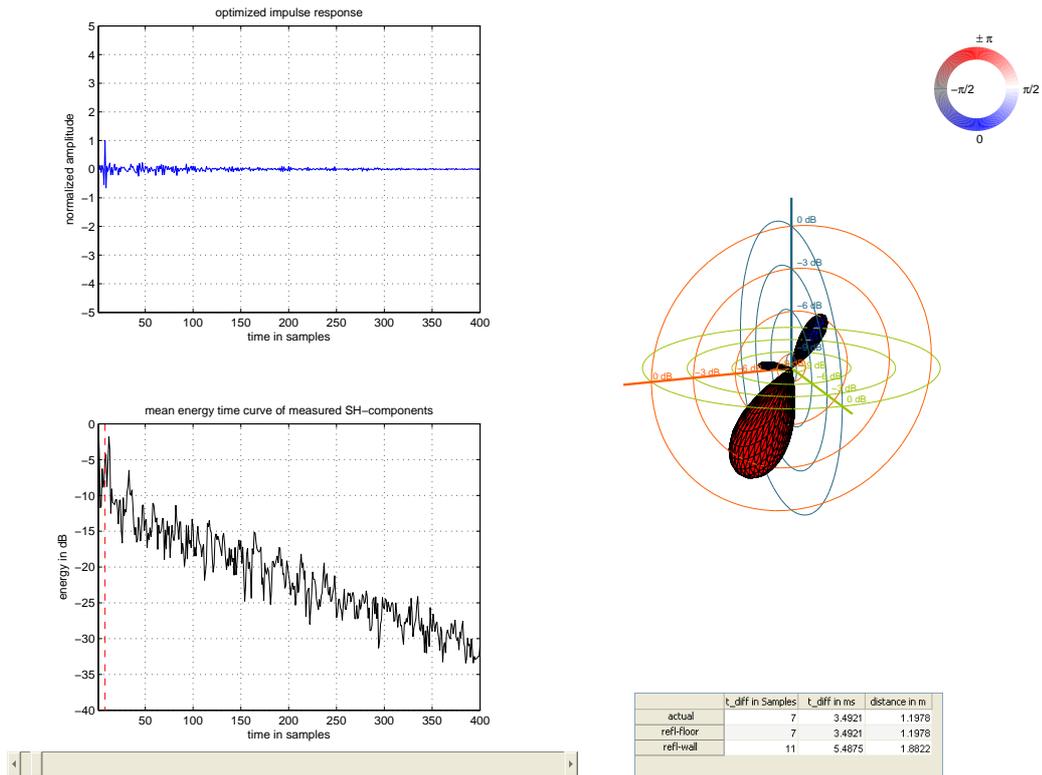


Abbildung 11: Bodenreflexion

c) Seitliche Reflexion an der Tafel

Die zweite Reflexion, die eindeutig im mittleren Energieverlauf hervorsteicht, repräsentiert die seitliche Reflexion am Whiteboard. Um die Energie, die über diesen Reflexionspfad am Mikrofon eintrifft zu maximieren, muss die Schallabstrahlung, wie in Abbildung 12 ersichtlich, in die Richtung der Tafel fokussiert werden.

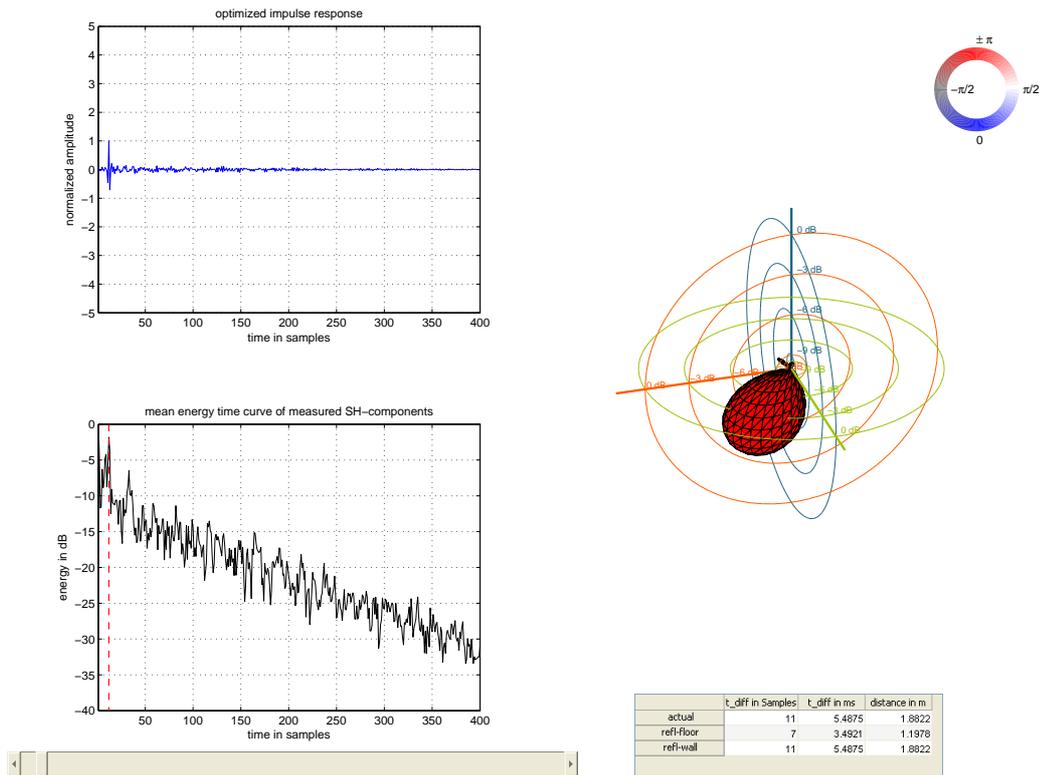


Abbildung 12: Seitliche Reflexion

d) Späte Reflexionen - Diffusfeld

Wählt man als Optimierungszeitpunkt  $T$  ein Sample, bei welchem ohnehin wenig Signalenergie vorhanden ist, kann aus rein physikalischen Gründen kein zugehöriges Abstrahlmuster gefunden werden, das in erster Linie diesen Verzögerungszeitpunkt anregt. Wie in Abbildung 13 erkennbar liefert die MVDR-Optimierung zwar eine Impulsantwort, bei welcher der ausgewählte Abtastwert auf 1 gesetzt wurde, jedoch konnte die restliche Signalenergie nur sehr schlecht minimiert werden. Daher treten Amplituden auf, die bis zu fünfmal größer als beim Optimierungszeitpunkt sind. Dementsprechend kann auch keine Vorzugsrichtung in der Abstrahlcharakteristik erkannt werden. Da die Wahl des Zeitpunktes nicht sinnvoll erscheint, wird der Betrag des Abstrahlmusters negativ gewichtet und verschwindet nahezu ganz.

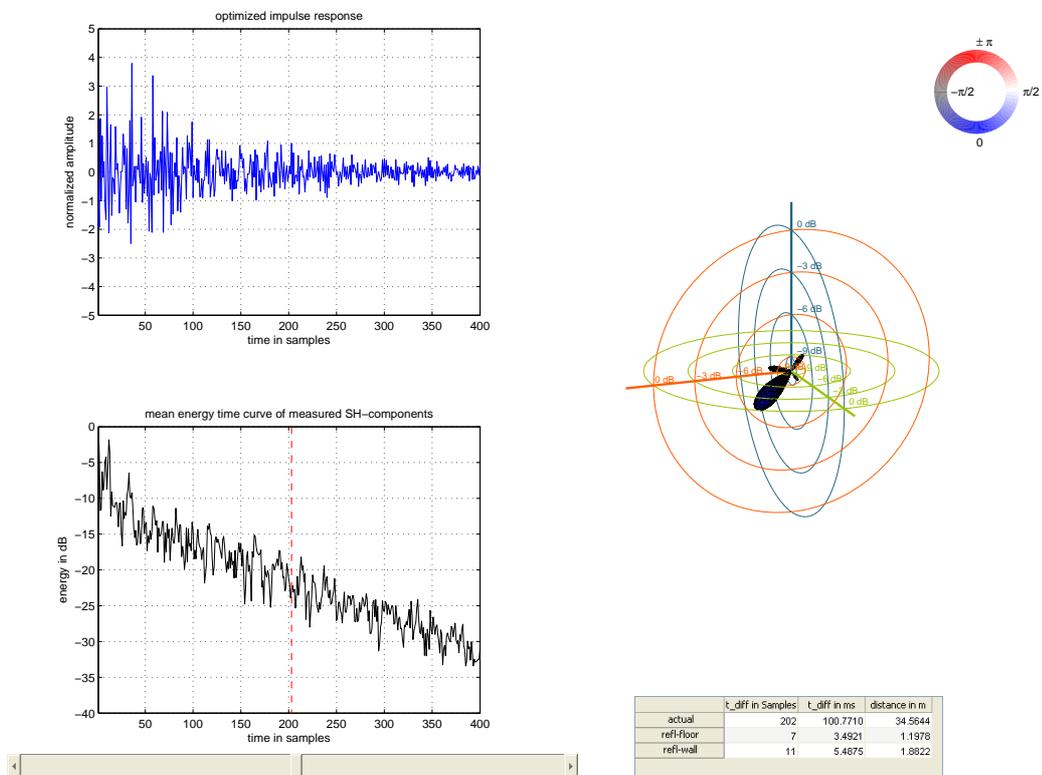


Abbildung 13: Späte Reflexion - Diffusfeld

## 6 Ausblick

In dieser Seminararbeit wurden zusätzlich zu dem herkömmlichen Verfahren mit omnidirektionaler Schallquelle auch Raumimpulsantworten für Abstrahlcharakteristiken gemessen, die Sphärischen Harmonischen höherer Ordnungen entsprechen. Dies bietet den Vorteil, mittels Superposition Impulsantworten für nahezu beliebige Abstrahlcharakteristiken generieren zu können. Ein besonderes Augenmerk wurde dabei auf den Einsatz eines Algorithmus gelegt, der eine automatisierte Auswertung der Richtungsinformation ermöglichen soll.

Eine weitere Verbesserungsmöglichkeit wäre nun, den einzelnen Druckempfänger mit einem kompakten kugelförmigen Mikrofonarray zu ersetzen.

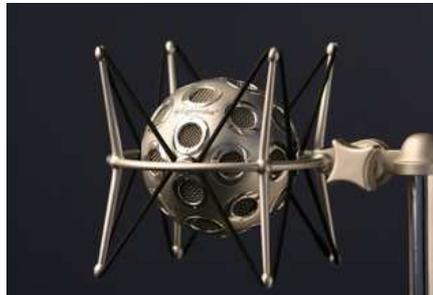


Abbildung 14: *Eigenmike* der Firma *mh acoustics LLC* als Beispiel für kugelförmiges Mikrofonarray

Dadurch bekäme man zusätzliche Richtungsinformation über das Eintreffen des Schalls am Messort. Man könnte so die Schallenergie bestimmen, die über einen bestimmten Reflexionspfad wann, wie stark und von welcher Seite am Sensor eintrifft.

Dies könnte beispielsweise Aufschluss über Reflexionen bieten, deren Einfallrichtung und Intensität sich störend auf die räumliche Klangwahrnehmung eines bestimmten Sitzplatzes in einem Konzertsaal auswirken.

## Literatur

- [Maj08] P. Majdak, *Vorlesungsunterlagen zu Algorithmen in Akustik und Computermusik, VO+UE, WS2008*, Ausgabe 2008.
- [Pom08] H. Pomberger, *Angular and Radial Directivity Control for Spherical Loudspeaker Arrays*, Graz, A, 2008.
- [Wil99] E. G. Williams, *Fourier Acoustics*. Academic Press, 1999.
- [Zot09] F. Zotter, *Analysis and Synthesis of Sound-Radiation with Spherical Arrays*, 2009.
- [Zot11] Zotter, *Vorlesungsskriptum: Akustische Holografie und Holografie*, Feb 2011.
- [ZS07] F. Zotter and A. Sontacchi, *IEM Report 39/27: Icosahedral loudspeaker array, Graz, Tech. Rep.*, 2007. [Online]. Available: [http://iem.at/projekte/publications/iem\\_report/report39\\_07/report39\\_07/file\\_view](http://iem.at/projekte/publications/iem_report/report39_07/report39_07/file_view)

## A MATLAB Source Code

### *gui.m*

```

1 % Algorithmen in Akustik und Computermusik 2 (WS 2010/11)
2 % Raumimpulsantwortmessung mit kompakter kugelfoermiger Lautsprecheranordnung
3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4 % Implemented by %
5 % Baumgartner Robert, Egger Katharina %
6 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
7
8 clear all
9 close all
10
11 global data;
12 global k;
13
14 N=3; % SH order
15 Ldown=22; % down sampling factor
16 k=15; % condition number
17
18 % IR calculation
19 if exist('ir.mat','file')==2
20     load('ir.mat');
21 else
22     for ii=1:(N+1)^2
23         % [sw,fs_sw]=wavread(['Messungen/CubeSH25102010/sweepresp_' num2str(ii) '.wav']);
24         [sw,fs_sw]=wavread(['Messungen/02122010_sweep2sek/sweepresp_' num2str(ii) '.wav'
25         ]);
26         irfull=irswep(sw,fs_sw);
27         ir(:,ii)=resample(irfull,1,Ldown);
28     end
29 end
30 fs=fs_sw/Ldown;
31
32 [aa,indx] = max(mean((ir).^2,2));
33 cutoff = min(indx)-1;
34 ir = ir(cutoff:end,:);
35
36 fig = figure('ToolBar','figure');
37
38 pos=get(0,'ScreenSize');
39 set(fig,'Position',[10,50,pos(3)-20,pos(4)-150])
40
41 tinit=2; % delay of direct wave
42 slidMax=400; % 400 samples = about 200ms
43 slidMin=tinit -1;
44 tframe=uicontrol('style','slider','Units','normalized','Position',...
45     [0.05 0.03 0.5 0.03],'Callback','gui_callback','SliderStep',[1/(slidMax-slidMin)
46     10/(slidMax-slidMin)],...
47     'Value',tinit-1,'Max',slidMax,'Min',slidMin);
48
49 cnames = {'t_diff in Samples','t_diff in ms','distance in m'};
50 rnames = {'actual','refl-floor','refl-wall'};
51 [diff_t diff_m]=distance(0, fs);
52 dat = [0, diff_t, diff_m; ...
53     7, 3.4921, 1.1978, ; ...
54     11, 5.4875, 1.8822];
55 tab=uitable('Data',dat,'ColumnName',cnames,'RowName',rnames,'Units','normalized','
56     Position',[0.6 0.03 0.36 0.15],...
57     'CellEditCallback','gui_callback');
58
59 [c_opt ir_beam]=mvdr(ir,tinit,k);
60 E=10*log10(mean((ir).^2,2)/max(mean((ir).^2,2)));

```

```

61 cursorx = tinit-1;
62 cursory = -40;
63
64 subplot(2,2,1)
65 cla
66 p=plot(0:length(ir_beam)-1,ir_beam);
67 axis ([slidMin slidMax -5 5]);
68 xlabel('time in samples')
69 ylabel('normalized amplitude')
70 title('optimized impulse response')
71 grid on
72 subplot(2,2,3)
73 m=plot(0:length(E)-1,E,'-k');
74 xlabel('time in samples')
75 ylabel('energy in dB')
76 title('mean energy time curve of measured SH-components')
77 hold on
78 grid on
79 c=stem(cursorx,cursory,'--r','Marker','none');
80 axis ([slidMin slidMax -40 0]);
81
82
83
84 set(gca,'ButtonDownFcn','gui_callback');
85 set(get(gca,'Children'),'ButtonDownFcn','gui_callback');
86
87 subplot(2,2,[2 4])
88 balloonplot(c_opt,E(tinit))
89
90 data=struct('dat',dat,'tframe',tframe,'tab',tab,'N',N,'ir',ir, ...
91           'cursorx',cursorx,'cursory',cursory,'p',p,'c',c,'E',E,'fs',fs);

```

### **gui\_callback.m**

```

1 function gui_callback
2
3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4 % Implemented by %
5 % Baumgartner Robert, Egger Katharina %
6 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
7
8 global data;
9 global k;
10
11 if gcbo==data.tframe
12     tset = floor(get(data.tframe,'Value'));
13 else
14     tset=get(gca,'CurrentPoint');
15     tset=floor(tset(1));
16     set(data.tframe,'Value',tset);
17 end
18
19 [diff_t diff_m]=distance(tset-1, data.fs);
20 data.dat(1,1:3) = [tset-1 diff_t diff_m];
21
22 set(data.tab,'Data',data.dat);
23 [c_opt ir_beam]=mvdr(data.ir,tset+1,k);
24
25 data.cursorx = tset;
26
27 set(data.p,'YData',ir_beam)
28 set(data.c,'XData',data.cursorx)
29
30 subplot(2,2,[2 4])
31 cla
32 balloonplot(c_opt,data.E(tset))
33
34

```

35 drawnow

**mvdr.m**

```

1 function [c_opt ir_beam]=mvdr(ir,tset,k)
2 % MVDR (Minimum Variance Distortionless Response)
3 % Usage: [c_opt ir_beam]=mvdr(ir,tset,[k])
4 % Input:  ir      ... IRs for each SH
5 %         tset    ... time sample to be set to 1
6 %         k      ... condition number (default: 10)
7 % Output: c_opt  ... optimal SH coeffs
8 %         ir_beam ... IR of resulting beam
9 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10 % Implemented by
11 % Baumgartner Robert, Egger Katharina
12 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13
14 if ~exist('k','var')
15     k=10;
16 end
17
18 [U,S,V]=svd(ir'*ir); % U S V' = M
19 s=diag(S);
20 smax=max(s);
21 idx=find(s>smx*1/k);
22 s=s(idx);
23 Sinv=diag(1./s);
24 Cinv=V(:,idx)*Sinv*U(:,idx)'; %inv(M)=V S^-1 U'
25
26 c_opt=Cinv*ir(tset,:)'./(ir(tset,:)*Cinv*ir(tset,:));
27 ir_beam=mean(ir*c_opt,2);
28
29 end

```

**balloonplot.m**

```

1 function balloonplot(c_beam,E,dyn,viewangle,wghtFac)
2 % BALLOONPLOT plots 3-dimensional radiation patterns for given sh-coefficients
3 % Usage: balloonplot(c_beam,[viewangle])
4 % input arguments: c_beam ... sh-coefficients
5 %                  E      ... energy of pattern
6 %                  dyn    ... maximal dynamic of radiation pattern in dB
7 %                  viewangle... optional viewangle definition according
8 %                          to [az el] in degrees
9 %                  wghtFac ... weighting factor for energy weighting
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 % Implemented by
12 % Baumgartner Robert, Egger Katharina
13 % based on implementation of Fabian Hohl
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15
16 if ~exist('dyn','var')
17     dyn=10; % [dB]
18 end
19 if ~exist('viewangle','var')
20     viewangle=[170 0];
21 end
22 if ~exist('wghtFac','var')
23     wghtFac = 0.025;
24 end
25
26
27 %f = figure('NumberTitle','off');
28 %clf
29
30 N=sqrt(length(c_beam))-1;
31

```

```

32 [tri,phi,theta]=spheregridXY(80,40);
33
34 X=cos(phi).*sin(theta);
35 Y=sin(phi).*sin(theta);
36 Z=cos(theta);
37
38 tri=trimesh(tri,X,Y,Z,zeros(size(X)));
39 axis equal
40 set(gca,'Visible','off');
41
42 Ysh=i*sh_matrix_real(N,phi,theta);
43
44
45 %%% colors
46 orange = [1 0.3686 0];
47 %hblau = [0.3254 0.5490 0.7176];
48 dblau = [0.1333 0.4039 0.5019];
49 hgruen = [0.6156 0.7607 0.019];
50
51 % axes
52 hold on
53 maxr = 1.2; % Skalierungsfaktor
54 alpha=0*pi/180; % fr Azimuthversatz der Instrumentenachse (um 55 Grad im Bogenmass um z-
    Achse drehen (auf Stern 33) für 'cnt)
55 Beta=0; %Grad
56 beta=Beta*pi/180; % fr Elevationsversatz der Instrumentenachse
57 plot3([0 maxr*cos(alpha)*cos(beta)],[0 maxr*-sin(alpha)*cos(beta)],[0 maxr*-sin(beta)],'
    Color',orange,'LineWidth',2);
58 plot3([0 maxr*sin(alpha)*cos(beta)],[0 maxr*cos(alpha)*cos(beta)],[0 0],'Color',hgruen,'
    LineWidth',2);
59 plot3([0 maxr*cos(alpha)*cos(pi/2-beta)],[0 maxr*sin(alpha)*cos(pi/2-beta)],[0 maxr*sin(
    pi/2-beta)],'Color',dblau,'LineWidth',2)
60
61 theta_circ=linspace(0,2*pi,80);
62 x_circ1=sin(theta_circ);
63 y_circ1=zeros(size(theta_circ));
64 z_circ1=cos(theta_circ);
65 for r=1:-3/dyn:0
66     plot3(r*x_circ1,r*y_circ1,r*z_circ1,'Color',orange);
67     text(r,0.03,0.05,[num2str(round((r-1)*dyn)),' dB'],'FontSize',5,'Color',orange);
68     plot3(r*cos(pi/2-beta)*z_circ1,r*x_circ1,r*sin(pi/2-beta)*z_circ1,'Color',dblau);
69     text(-0.03,r+0.01,0.05,[num2str(round((r-1)*dyn)),' dB'],'FontSize',5,'Color',hgruen)
70
71     plot3(r*-cos(beta)*z_circ1,r*x_circ1,r*sin(beta)*z_circ1,'Color',hgruen);
72     text(-0.03,0.03,r+0.05,[num2str(round((r-1)*dyn)),' dB'],'FontSize',5,'Color',dblau);
73 end
74 Pplot=Ysh*c_beam;
75
76 R=20*log10(abs(Pplot));
77
78 %Dynamikbegrenzung auf dyn
79 R=((R-max(R))+dyn)/dyn;
80
81 R=R+wghtFac*E;
82 R(find(R<0))=0;
83
84 X=X.*R;
85 Y=Y.*R;
86 Z=Z.*R;
87
88 axis([-1 1 -1 1 -1 1]);
89 set(tri,'vertices',[X Y Z],'FaceVertexCData',angle(Pplot));
90 caxis([-pi pi]);
91 load C.mat;
92 colormap(C);
93 set(tri,'facecolor','interp');
94 set(tri,'edgecolor','k');

```

```

95 view(viewangle);
96 load phase_cmap;
97 farbrad_phase;
98
99
100 end

```

### *distance.m*

```

1 function [diff_t diff_m]=distance(noSmpl, fs)
2 % calculates the responding time difference and distance of number of
3 % Samples
4 % Usage: [diff_t diff_m]=distance(noSmpl, fs)
5 % Input: noSmpl ... number of Samples
6 %        fs ... sampling rate
7 % Output: diff_t ... difference in ms
8 %        diff_m ... difference in m
9 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10 % Implemented by %
11 % Baumgartner Robert, Egger Katharina %
12 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13
14 diff_t = noSmpl * 1000 / fs;
15 diff_m = diff_t * 343 / 1000;

```

### *irsweep.m*

```

1 function ir=irsweep(sweepresp,fs)
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 % Implemented by %
4 % Baumgartner Robert, Egger Katharina %
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6
7 sweep=expsweep(50,fs/2,1,fs);
8 isw=invexpsweep(sweep,50,fs/2,fs);
9
10 nfft=length(sweepresp)+length(isw)-1;
11 ir=real(ifft( fft(sweepresp,nfft).*fft(isw,nfft) )) / length(sweepresp);
12
13
14 end

```

## B Messequipment

- Mikrofon: Neutrik SPL Druckempfänger
- Mikrofonvorverstärker: Behringer Eurorack MX 602A
- D/A-Wandler: RME ADI-8 DS
- Verstärker: Bittner Audio 8X 100