

Ikosaeder-Lautsprecher

Ansteuerungsfilter/Systemreduktion

Seminararbeit Algorithmen in Computermusik und Akustik 2, SE/UE

Thomas Kößler, Matr.Nr. 0530845
Johannes Zaar, Matr.Nr. 0473066

Graz, 25. März 2009
Betreuer: Franz Zotter



institut für elektronische musik und akustik



Zusammenfassung

In dieser Arbeit soll die Implementierung einer neuen Echtzeit-Ansteuerung für den von Franz Zotter am Institut für Elektronische Musik (IEM) entwickelten Ikosaeder-Lautsprecher besprochen werden. Bei dem Ikosaeder-Lautsprecher handelt es sich um ein mit 20 Speakern bestücktes, Ikosaederförmiges Lautsprecher-Array.

Die Hauptaufgabe in praktischer Hinsicht bestand darin, die bereits bestehende Implementierung der Ikosaeder-Ansteuerung mittels Matlab und Pure Data (PD) zu modifizieren, bzw. überhaupt neu zu programmieren. Dabei lag das Hauptaugenmerk auf der Rechenleistungsreduktion des Echtzeitsystems in PD, um dieses auch mit durchschnittlicher Computerleistung (z.B. mobil mit Laptop) einsetzbar zu machen.

Im ersten Teil dieser Dokumentation wird zunächst das bestehende Setup kurz beschrieben. Dann folgen die wesentlichen theoretischen und mathematischen Grundlagen für die Systembeschreibung bzw. -kontrolle. Der zweite Teil behandelt die praktische Umsetzung der Aufgabenstellung hinsichtlich der Methoden zur Reduzierung des Rechenaufwandes, d.h. die Implementierung sowie den resultierenden Signalfluss.

Inhaltsverzeichnis

1	Erfassung des Multiple-Input-Multiple-Output-Systems (MIMO-System)	4
1.1	Setup	4
1.2	Mikrofonseitige Abtastung	5
1.2.1	Erfassen der Frequenzgänge	5
1.2.2	Normieren der Frequenzgänge	6
1.2.3	Transformation in sphärische Harmonische	8
1.3	Zerlegung der Lautsprecher-signale	11
1.4	Gesamtsystem	13
1.5	MIMO-Systemkontrolle	13
2	Implementierung	15
2.1	Ansatz zur Reduktion des Rechenaufwandes	15
2.1.1	Ausgangssituation	15
2.1.2	Reduzierung der Auflösung - Unterabtastung	16
2.1.3	Reduktion der Filter-Anzahl - Maskierung	18
2.2	Modifikation der Filter	19
2.2.1	Akausalitäts-Problematik	19
2.2.2	Unterabtastung	20
2.2.3	Entzerrung der Frequenzgänge	22
2.2.4	Maskierung der Kontrollmatrix	24
2.2.5	Fenstern und Zyklisches Verdrehen der Impulsantworten	26
2.2.6	Grafischer Überblick - Filtermodifikation	28
2.3	Implementierung des Echtzeitsystems in Pure Data	29
2.3.1	Beschreibung der einzelnen Schritte	29
2.3.2	Grafischer Überblick - Echtzeit-System	31
3	Schlussfolgerungen und Ausblick	32

1 Erfassung des Multiple-Input-Multiple-Output-Systems (MIMO-System)

Als Basis für die nun folgende Beschreibung diente [FZ08] und [Pom08].

1.1 Setup

Um eine Vorstellung zu bekommen, worauf sich genau die in den nächsten Punkten folgenden mathematischen Ausdrücke beziehen, wird hier der Messaufbau und die verwendete Hardware beschrieben. Ausgangspunkt ist ein kugelförmiges Lautsprecherarray, das durch eine Ikosaederform (20-Flächner) angenähert wird. Der Ikosaeder gehört zu den Platonischen Körpern, von denen es insgesamt nur 5 gibt.

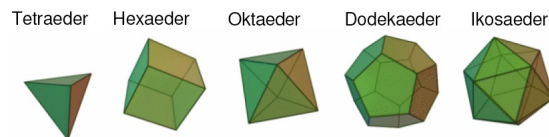


Abbildung 1: Die 5 Platonischen Körper

Er besteht aus 20 gleich großen Teilflächen, welche die Form gleichseitiger Dreiecke haben. Diese geometrische Form wurde aufgrund ihrer konstruktiven Regelmäßigkeit gewählt. Auf jeder Teilfläche sitzt ein Lautsprecher, der normal von dieser wegstrahlt. Zur Messung wurde ein halbkreisförmiges Mikrofonarray verwendet, das aus 18 Mikrofonen, die in der Elevation (ϑ) in 10° -Abständen angeordnet sind. Der Ikosaederlautsprecher wurde dann - drehbar gelagert - 36 Mal ebenfalls in 10° -Schritten in Azimuthrichtung (φ) weiterbewegt, damit die ganze Kugeloberfläche abgetastet werden konnte. Aus diesem Raster ergeben sich so 648 Mikrofonpositionen, die gleichwinklig auf einer Kugeloberfläche verteilt sind.

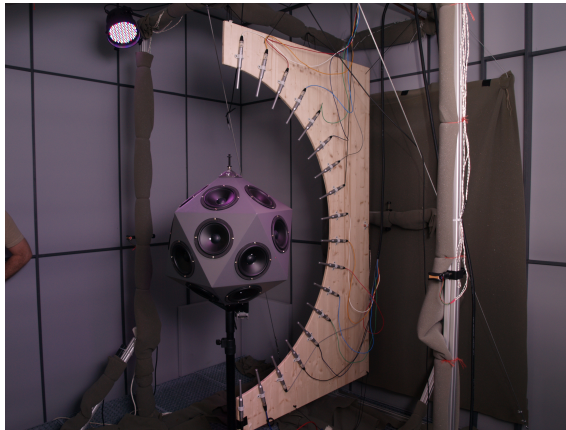


Abbildung 2: Messaufbau bestehend aus dem Icosaederlautsprecher und dem Mikrofonarray [FZ08, Pom08]

Aus dieser Messanordnung wurden in der Arbeit [Pom08] dann die Übertragungsfunktionen der 20 Lautsprecher zu den 648 Mikrofonen gewonnen, die als Basismaterial für die weitere Systembeschreibung herangezogen werden.

1.2 Mikrofonseitige Abtastung

1.2.1 Erfassen der Frequenzgänge

Die oben erwähnte Messanordnung stellt also ein $K \times L$ -MIMO-System dar, wobei in unserem Fall $K=648$ (Mikrofonpositionen) und $L=20$ (Lautsprecher) ist. Damit man dieses System beschreiben kann wurden die Impulsantworten $g_{ij}(t)$ aller möglichen Übertragungspfade (Lautsprecher zu Mikrofon) in [Pom08] gemessen.

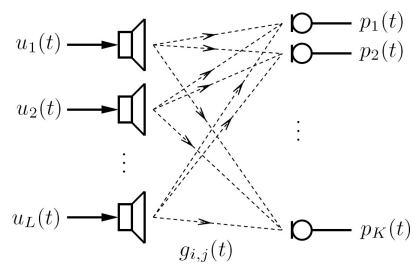


Abbildung 3: Resultierendes MIMO-System [Pom08]

Mittels Fouriertransformation erhält man somit eine Matrix $G(\omega)$, die alle Übertragungsfunktionen enthält.

$$\mathbf{G}(\omega) = \begin{pmatrix} g_{11}(\omega) & \cdots & g_{1L}(\omega) \\ \vdots & \ddots & \vdots \\ g_{K1}(\omega) & \cdots & g_{KL}(\omega) \end{pmatrix}$$

Somit kann das System als einfache Matrixmultiplikation im Frequenzbereich dargestellt werden:

$$\vec{p}(\omega) = \mathbf{G}(\omega)\vec{u}(\omega)$$

Jede Spalte der Matrix \mathbf{G} beschreibt hier die frequenzabhängige Richtcharakteristik von jedem der 20 Lautsprecher. Zur einfacheren Lesbarkeit wird die Variable ω , welche die Frequenzabhängigkeit kennzeichnet, in den folgenden Seiten weggelassen.

1.2.2 Normieren der Frequenzgänge

Da das Übertragungsverhalten der Einzellautsprecher variiert (Fertigungstoleranzen), muss ein Abgleich vorgenommen werden. Eine Möglichkeit wäre, dass man alle Lautsprecher auf unity-gain bringt, d.h. einen konstanten Frequenzgang erzeugt. Da die Lautsprecher aber bei tiefen (unter 100Hz) und hohen (über 800Hz) Frequenzen starke Dämpfungen aufweisen, müssten bis zu 40dB an Verstärkung aufgebracht werden, um diese Dämpfungen auszugleichen.

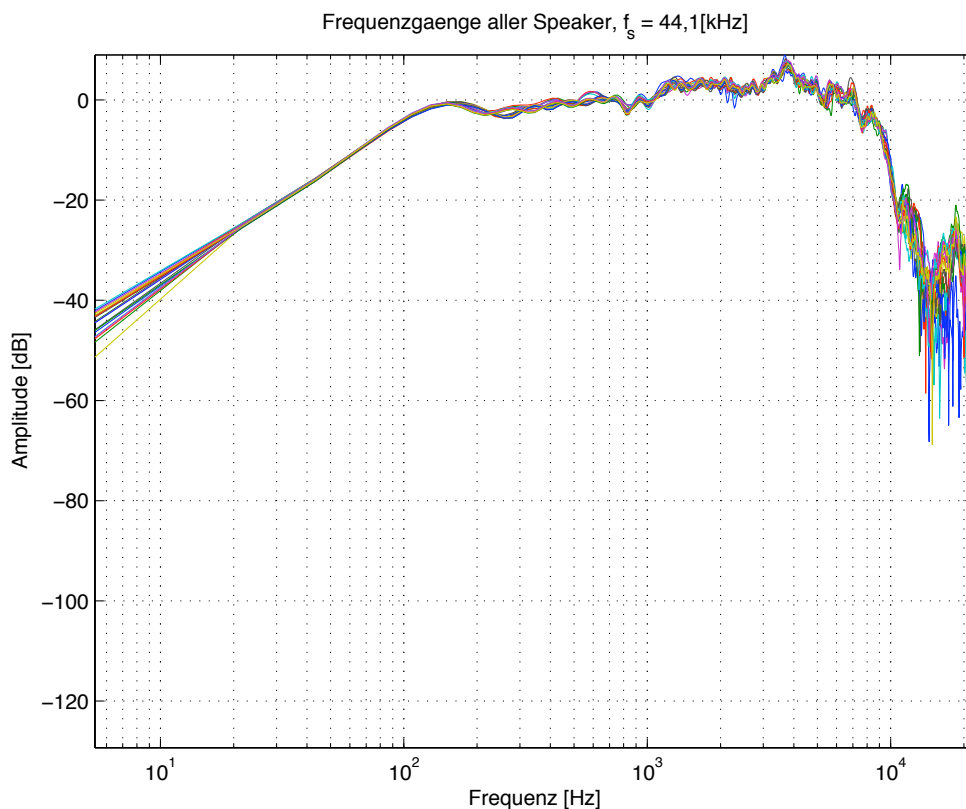


Abbildung 4: Aufgenommene Frequenzgänge der Lautsprecher

Darum werden alle Frequenzgänge der Lautsprecher auf einen Referenzfrequenzgang hin entzerrt, den Direktpfad eines willkürlich gewählten Lautsprechers.

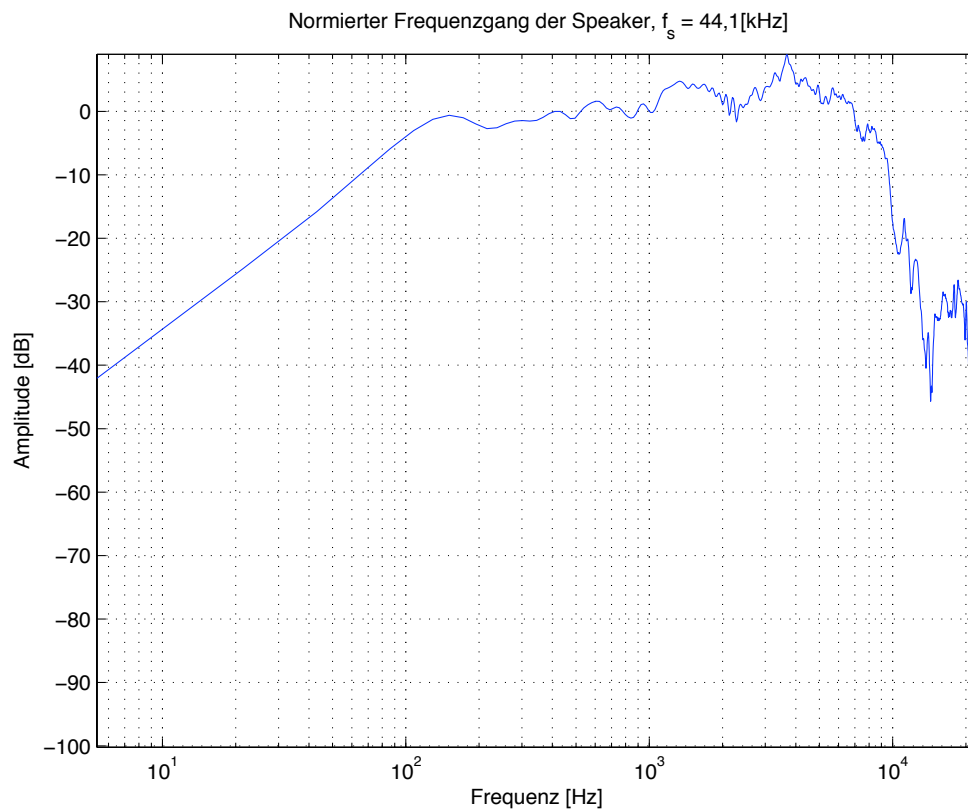
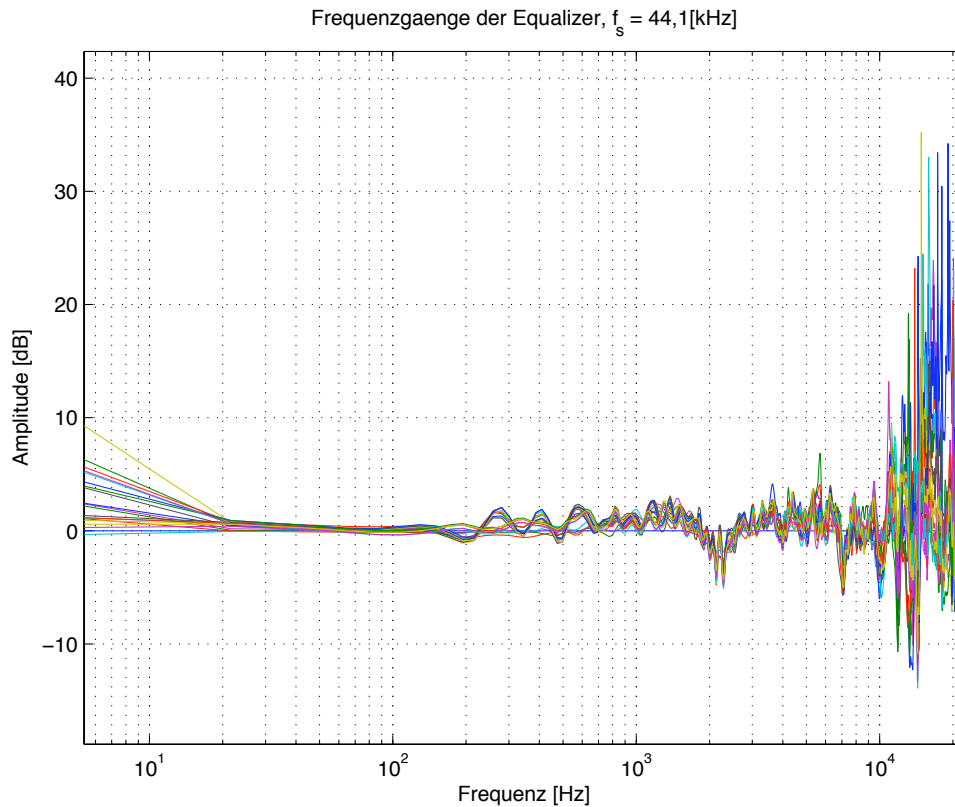


Abbildung 5: Referenzfrequenzgang für die Normierung

Im System wird also ein Satz von Equalizer-Filtern benötigt, der im Folgenden mit

$$\mathbf{G}_{EQ} = \text{diag}(\vec{g}_{EQ}(\omega))$$

bezeichnet wird.

Abbildung 6: Frequenzgänge der Equalizer \mathbf{G}_{EQ}

Für die Entzerrung der Mikrofone diente ein qualitativ hochwertiges Messmikrofon als Referenz. Die entsprechenden Entzerrungskurven sind bereits beim Messen in die Frequenzgänge eingeflossen und müssen daher nicht weiter berücksichtigt werden.

1.2.3 Transformation in sphärische Harmonische

Durch eine entsprechende Multiplikation der Matrix \mathbf{G} könnten bereits bestimmte Abstrahlcharakteristiken erreicht werden. Die gewünschte Richtcharakteristik kann sogar mit einem Steuerungsvektor \vec{p}_{ctl} vorgegeben werden, wenn die Matrixgleichung

$$\vec{p} = \mathbf{G}\mathbf{G}^\dagger \vec{p}_{ctl}$$

angesetzt wird. Darin ist \mathbf{G}^\dagger die Pseudoinverse der Systemmatrix \mathbf{G} , wenn für die Steuerbarkeit $\vec{p} \approx \vec{p}_{ctl}$ gelten soll.

Dies ist aber unpraktisch und der Steuerungsvektor wird bei größer werdender Anzahl der Mikrofone ebenfalls größer. Weiters entstehen durch die Näherungslösung mittels der Pseudoinversen Fehler, die schwer einschätzbar sind. Der Approximationsfehler, der bei $\mathbf{G}\mathbf{G}^\dagger$ entsteht hängt stark von der räumlichen Abtastung mit Mikrofonen ab und

ist i.A. räumlich nicht gleichmäßig verteilt. Sphärische Harmonische (SH) $Y_n^m(\varphi, \vartheta)$, die Basisfunktionen der Fouriertransformation auf der Kugel hingegen, erlauben die Definition einer gleichmäßigen räumlichen Auflösung. Damit, also mit der Begrenzung der SH-Ordnung, kann eine gleichmäßige Verteilung der Approximationsfehler auf der Kugel besser kontrolliert werden.

Analog zur Fouriertransformation, welche im 1-Dimensionalen die Zerlegung in orthogonale Basisfunktionen (\sin bzw. \cos) beschreibt, wird hier ein 3-dimensionales Feld mittels Linearkombination gewichteter Basisfunktionen beschrieben. Dies entspricht einer Fourierzerlegung auf der Kugeloberfläche. Jede Basisfunktion für sich ist räumlich bandbegrenzt, was eine saubere Synthese, d.h. ohne räumliches Aliasing, überhaupt möglich macht. Die sphärischen Harmonischen sind Teil der Lösung der Wellengleichung $(k^2 + \Delta)p = 0$, die mittels Produktansatz in Kugelkoordinaten gelöst wird.

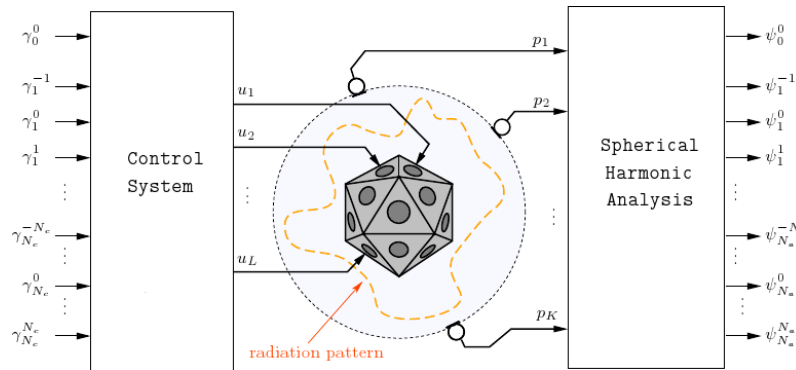


Abbildung 7: Blockschaltbild der gewünschten Systemzerlegung [Pom08]

Es gilt der Ansatz, dass jede über eine Fläche integrierbare Funktion

$$p(\theta) = \sum_{n=0}^{\infty} \sum_{m=-n}^n \chi_{mn} Y_n^m(\theta)$$

(in unserem Fall das Schallfeld bzw. die Schalldruckverteilung) auf der Einheitskugel durch eine Linearkombination von SHs beschrieben werden kann (siehe [Pom08]).

χ_{mn} ist der Koeffizientenvektor, der die Gewichte der Basisfunktionen $Y_n^m(\theta)$ enthält. $mn = n^2 + n + m + 1$, eine Laufvariable, die eine lineare Indizierung der Basisfunktionen ermöglicht. θ verweist auf den Abtastpunkt im Raum und ist abhängig von φ und ϑ .

n und m sind Grad und Ordnung der SHs. Je höher die Ordnung, desto feiner kann räumlich aufgelöst werden. Praktisch kann natürlich nur mit begrenzter Auflösung (räumlicher Bandbreite) gearbeitet werden.

Es gilt also, den Druckvektor \vec{p} in gewichtete SHs zu zerlegen. Dazu muss erst die Ordnung N festgelegt werden, die von der Anzahl der Abtastpunkte M abhängig und durch das Abtasttheorem

$$N \leq \sqrt{M} - 1$$

gegeben ist (siehe [FZ08]).

Da wir in unserem System 648 Abtastpunkte haben, erreichen wir eine mikrofonseitige Ordnung von $N_a = 24$, bis zu welcher die Koeffizienten noch ohne räumliches Aliasing abgeschätzt werden könnten, numerisch lässt sich das aber nur bis $N = 17$ rechnen, da die Pseudoinverse für $N > 17$ nahezu singular ist. Lautsprecherseitig ist das System nur bis zur Ordnung $N_c = 3$ steuerbar, d.h. die SH sind ohne räumliches Aliasing darstellbar. Daher wird die Zerlegung nun für $N = 3$ ausgelegt.

Es wird die Matrix

$$\mathbf{C} = \begin{pmatrix} Y_1(\theta_1) & \cdots & Y_{(N+1)^2}(\theta_1) \\ \vdots & \ddots & \vdots \\ Y_1(\theta_K) & \cdots & Y_{(N+1)^2}(\theta_K) \end{pmatrix}$$

definiert, die alle $(N + 1)^2$ Basisfunktionen beinhaltet bzw. der Vektor

$$\vec{\psi} = \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_{(N+1)^2} \end{pmatrix}$$

mit den Koeffizienten der SHs.

Der Vektor \vec{p} wird zerlegt in $\vec{p} = \mathbf{C}\vec{\psi}$. Es ist schnell ersichtlich, dass dieses Gleichungssystem nicht so einfach lösbar ist, da \mathbf{C} keine quadratische Matrix ist. Das Gleichungssystem ist überbestimmt - $K > (N + 1)^2$ - und hat keine exakte Lösung.

Es wird nun ein Vektor $\hat{\psi}$ eingeführt, der eine Schätzung des Koeffizientenvektors darstellt. Dabei soll der quadratische Fehler

$$\|\vec{p} - \mathbf{C}\hat{\psi}\|^2 \longrightarrow \min$$

minimiert werden.

Dazu muss die Ableitung (siehe [Pom08]) des quadratischen Fehlers Null werden. Der Ausdruck

$$\hat{\psi} = \underbrace{(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T}_{\mathbf{C}^\dagger} \vec{p}$$

ist nun die Lösung für den geschätzten Koeffizientenvektor.

$\mathbf{C}^\dagger = (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T$ wird Pseudoinverse genannt. Sie ist in diesem Fall eine „Linksinverse“, d.h. wenn sie von links auf die ursprüngliche Matrix \mathbf{C} multipliziert wird, resultiert daraus die Einheitsmatrix. Im praktischen Sinne hat sie die Eigenschaft, dass der quadratische Fehler minimiert wird. Schnell wird ersichtlich, dass sogar der Schätzer

$$\hat{\psi} = \underbrace{(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{C}}_I \vec{\psi}$$

eine exakte Lösung liefern kann, wenn die Schalldruckverteilung bandbegrenzt ($\vec{p} = \mathbf{C}\vec{\psi}$) ist.

Weil eine gleichwinklige Abtastung auf der Kugeloberfläche vorliegt, ist die Fehlerverteilung aber alles andere als gleichmäßig. So ist die Rasterung am Äquator am gröbsten und wird zu den Polen hin immer feiner.

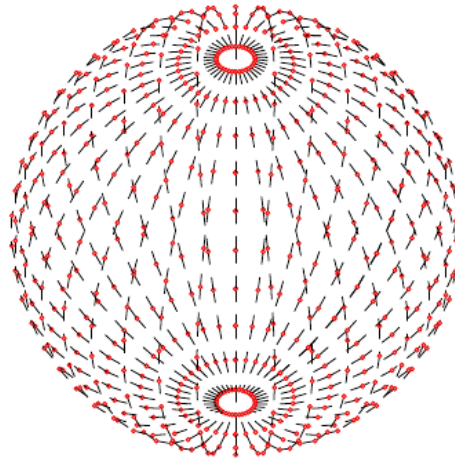


Abbildung 8: Rasterung der Kugeloberfläche [FZ08, Pom08]

Diese Unregelmäßigkeit wird korrigiert, indem eine Gewichtungsmatrix \mathbf{W} eingeführt wird. Sie hat die Form einer Diagonalmatrix und ändert die Struktur der Pseudoinversen nicht. Auf die Ermittlung der Gewichtungsmatrix wird in [Pom08] näher eingegangen. In den folgenden Schritten wird mit der gewichteten Pseudoinversen

$$\mathbf{C}^\dagger = (\mathbf{C}^T \mathbf{W} \mathbf{C})^{-1} \mathbf{C}^T \mathbf{W}$$

weitergerechnet.

Der Koeffizientenvektor kann nun mittels der Pseudoinversen berechnet werden:

$$\vec{\psi} = \mathbf{C}^\dagger \vec{p}$$

Unser Gesamtsystem $\vec{p} = \mathbf{G}\vec{u}$ zerlegt in SHs sieht nun folgendermaßen aus:

$$\vec{\psi} = \mathbf{C}^\dagger \mathbf{G}\vec{u}$$

1.3 Zerlegung der Lautsprechersignale

Nach der Zerlegung des Druckvektors in SHs liegt es nun nahe, auch lautsprecherseitig eine Zerlegung vorzunehmen. Dafür wird der selbe Ansatz verwendet wie oben, der besagt, dass jede über eine Fläche integrierbare Funktion mittels Linearkombination von gewichteten SHs auf der Einheitskugel dargestellt werden kann. Genau dies wird jetzt mit den 20 Lautsprechersignalen, die einer diskreten Spannungsverteilung auf der Oberfläche

des Ikosaeders entsprechen, gemacht.

Dafür werden die Lautsprechersignale als gewichtete Einheitsimpulse im Zentrum jedes Lautsprechers betrachtet:

$$u(\theta) = \sum_{l=1}^L \delta(\theta - \theta_l) u_l$$

Die Koeffizienten können nach Umrechnung (siehe [Pom08]) als

$$\Upsilon_{SH}^{(mn)} = \sum_{l=1}^L Y_{mn}(\theta_l) u_l$$

dargestellt werden.

Bringt man diese Linearkombination in Matrizen-Schreibweise, so ergibt sich die Form $\vec{\Upsilon}_{SH} = \mathbf{D}\vec{u}$, wobei

$$\mathbf{D} = \begin{pmatrix} Y_1(\theta_1) & \cdots & Y_1(\theta_L) \\ \vdots & \ddots & \vdots \\ Y_{(N+1)^2}(\theta_1) & \cdots & Y_{(N+1)^2}(\theta_L) \end{pmatrix}$$

die SHs an den Lautsprecherpositionen enthält und \vec{u}_{SH} die entsprechenden Koeffizienten.

Um nun die Lautsprechersignale durch die SHs ausdrücken zu können, muss das obige Gleichungssystem nach \vec{u} aufgelöst werden. Dabei versuchen wir zuerst die Pseudoinverse („Linksinverse,“)

$$\vec{u} = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \vec{\Upsilon}_{SH}$$

zu verwenden. Wie sich jedoch erkennen lässt, ist der Ausdruck $(\mathbf{D}^T \mathbf{D})^{-1}$ eine 20×20 -Matrix, die jedoch maximal Rang 16 haben kann und somit nicht vollen Rang besitzt. Das bedeutet, dass sie sich nicht invertieren lässt. Abhilfe schafft hier die Einführung der Matrix $\mathbf{D}^\dagger = \mathbf{D}^T (\mathbf{D} \mathbf{D}^T)^{-1}$, die in diesem Fall als „Rechtsinverse“ ihre Verwendung findet. Multipliziert man sie nämlich von rechts auf die ursprüngliche Matrix \mathbf{D} , so erhält man die Einheitsmatrix. Ihre Eigenschaft ist, dass sie minimale Varianz

$$\|\vec{u}\|^2 \rightarrow \min$$

besitzt und die Optimierungsaufgabe

$$\vec{\Upsilon}_{SH} - \mathbf{D}\vec{u} = \vec{0}$$

löst.

Die um $\mathbf{D}\mathbf{D}^\dagger = \mathbf{I}$ erweiterte Gleichung

$$\mathbf{D}\mathbf{D}^\dagger \vec{\Upsilon}_{SH} = \mathbf{D}\vec{u}$$

kann nun auf die gewünschten Lautsprechersignale

$$\vec{u} = \mathbf{D}^\dagger \vec{\Upsilon}_{SH}$$

umgeformt werden.

1.4 Gesamtsystem

D^\dagger wird Dekodermatrix genannt, da mit ihrer Hilfe die Lautsprecher-signale dekodiert werden können. Unser System kann somit wie folgt beschrieben werden:

$$\vec{\psi} = \mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger \vec{\Upsilon}_{SH}$$

1.5 MIMO-Systemkontrolle

Was jetzt noch fehlt, ist die Möglichkeit, das System zu steuern. Dazu wird der Steuerungsvektor $\vec{\gamma}$ eingeführt. Ihn gibt man quasi als Soll-Größe vor. Das System - falls es richtig arbeitet - wird dann bestmöglich reagieren um den Soll-Wert zu erreichen. Natürlich ist es sinnvoll, wenn Soll- und Ist-Wert die gleiche Struktur haben bzw. in der selben Domäne vorliegen. In unserem Fall sind dies Koeffizientenvektoren von SHs die linearkombiniert einer Druckverteilung entsprechen.

Zusätzlich wird deshalb noch eine Kontrollmatrix \mathbf{B} benötigt, die den Steuerungsvektor $\vec{\gamma}$ von der Druckdomäne auf die Spannungsdomäne transformiert.

$$\vec{u}_{SH} = \mathbf{B} \vec{\gamma}$$

Eingesetzt ins Gesamtsystem erhält man folgende Beziehung:

$$\vec{\psi} = \mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger \mathbf{B} \vec{\gamma}$$

Um nun die Steuerbarkeit des Systems zu gewährleisten, muss $\vec{\psi} \stackrel{!}{=} \vec{\gamma}$ gelten. Daraus folgt

$$\mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger \mathbf{B} \stackrel{!}{=} \mathbf{I}$$

was für die Kontrollmatrix \mathbf{B} bedeutet:

$$\mathbf{B} = (\mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger)^{-1}$$

Der Ausdruck $\mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger$ wird als Systemmatrix $\overset{o}{\mathbf{G}}$ bezeichnet. Da die Steuerung mittels Systeminversion funktioniert, spricht man bei der Kontrollmatrix von der der Systeminversion

$$\mathbf{B} = \overset{o}{\mathbf{G}}^{-1}$$

Somit ist das System kontrollierbar und lässt sich wie folgt beschreiben:

$$\vec{\psi} = \mathbf{C}^\dagger \mathbf{G} \mathbf{D}^\dagger \overset{o}{\mathbf{G}}^{-1} \vec{\gamma}$$

Nun sind nur noch die Lautsprecher zu entzerren, was mit dem errechneten Equalizer-Satz \mathbf{G}_{EQ} erreicht wird.

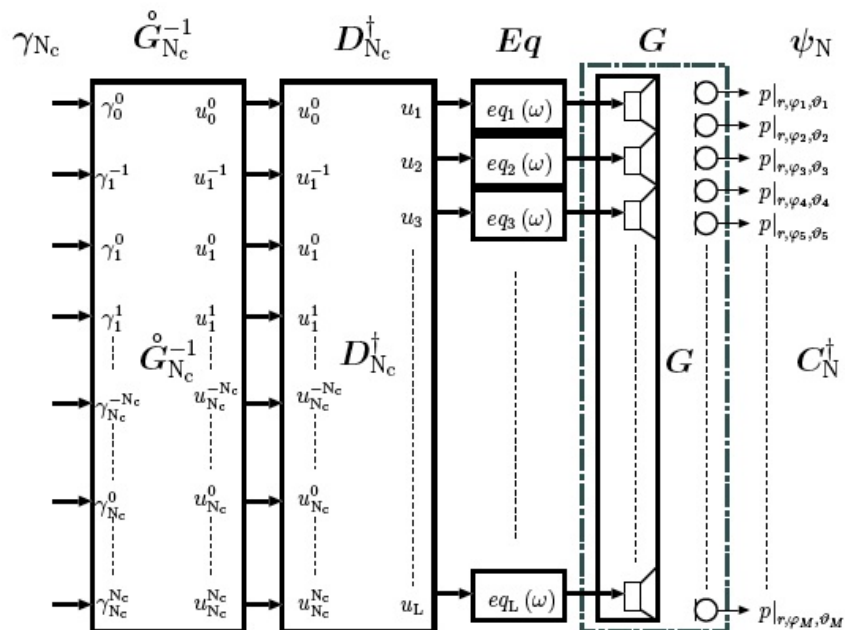


Abbildung 9: Gesamtsystemdarstellung des Icosaederlautsprechers [FZ08]

Um die Signale für die Ansteuerung der Lautsprecher zu erhalten, muss man also den vorgegebenen Steuerungsvektor mit der Systeminversen und der Dekoder-Matrix multiplizieren und dann die Lautsprecher mit der Equalizer-Matrix entzerren

$$\vec{u} = \mathbf{D}^\dagger \mathbf{G}^{-1} \vec{\gamma}.$$

Dies entspricht genau dem, was in Pure Data implementiert wurde (siehe nächster Abschnitt).

2 Implementierung

2.1 Ansatz zur Reduktion des Rechenaufwandes

2.1.1 Ausgangssituation

Um das System reduzieren zu können, war es zunächst einmal wichtig, die bereits vorhandenen Daten zu sichten und zu interpretieren. Das Material, auf das wir zurückgreifen konnten, bestand zum einen aus den Filtern der **Kontrollmatrix B**, welche die Inversion des Systemübertragungsverhaltens darstellen, und zum anderen aus der **Equalizer-Matrix**. Bei der Kontrollmatrix **B** handelt es sich um eine $16 \times 16 \times N_{fft}$ -Matrix, oder anders ausgedrückt um N_{fft} Matrizen der Dimension 16×16 .

$$Dim(\mathbf{B}) = 16 \times 16 \times N_{fft}$$

N_{fft} - Die Länge der Frequenzgänge im negativen und positiven Frequenzbereich - resultiert aus der doppelten Länge der gemessenen Impulsantworten - 1024 Samples bei einer Abtastrate von

$$f_s = 44,1[kHz]$$

und ergibt sich demnach zu $N_{fft} = 2 \cdot 1024 = 2048$.

$$N_{fft} = 2 \cdot 1024 = 2048$$

Somit ergeben sich $16 \cdot 16 = 256$ Filter über 2048 Frequenzbins.

Die Equalizer-Matrix hingegen besteht aus 20 Filtern über 2048 Frequenzbins ($20 \times N_{fft}$). Des Weiteren bot die bereits festgestellte geringe Relevanz eines Großteils der Einträge der Kontrollmatrix **B** einen konkreten Ansatzpunkt in Form einer für alle Frequenzbins gültigen **Maske** [FZ08], mittels derer die Kontrollmatrix deutlich reduziert werden kann. Eine zweite Möglichkeit ist die Reduzierung der Auflösung, also eine Unterabtastung. Somit gibt es zwei Parameter um die Anzahl der Rechenoperationen zu verändern:

- Die Auflösung/Anzahl der Frequenzbins (Equalizer- und Kontrollmatrix)
- Die Anzahl der Filter (Kontrollmatrix)

2.1.2 Reduzierung der Auflösung - Unterabtastung

Um unterabtasten zu können, muss zunächst einmal ermittelt werden, bis zu welcher oberen Grenzfrequenz (im Folgenden mit f_g abgekürzt) dies maximal möglich ist ohne relevante Information einzubüßen. Um das Nyquist-Kriterium ($f_g \leq f_s/2$) nicht zu verletzen, ist eine Antialias-Tiefpassfilterung unerlässlich, d.h. der darstellbare **Frequenzbereich wird kleiner**, je ungenauer die Auflösung bzw. je kleiner f_s wird.

Als **Kriterien** für die theoretische Ermittlung einer möglichst geringen - aber nicht verfälschenden - Abtastrate lassen sich **zwei Faktoren** identifizieren:

1. Räumliches Aliasing, welches durch die begrenzte Lautsprecher-Anzahl und deren gebündelte Schallabstrahlung bedingt ist. Im analytischen Sinne genau ist die Darstellung hier bis ca. $f_g = 800[Hz]$ (siehe Simulations-Grafiken unterhalb), allerdings kann man den Ikosaeder-Lautsprecher durchaus auch mit höherer Grenzfrequenz betreiben, wobei man natürlich Verfälschungen durch Aliasing in Kauf nehmen muss.

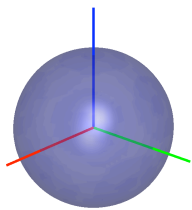


Abbildung 10: Räumliche Richtmuster für Y_0^0 bei $690Hz$

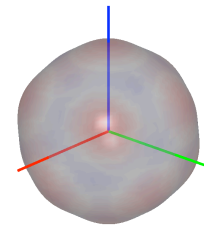


Abbildung 11: Räumliche Richtmuster für Y_0^0 bei $927Hz$

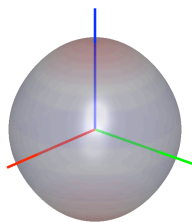


Abbildung 12: Räumliche Richtmuster für Y_2^0 bei $600Hz$

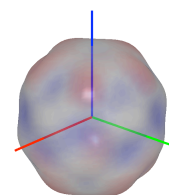


Abbildung 13: Räumliche Richtmuster für Y_2^0 bei $1430Hz$

2. Der Frequenzgang der Lautsprecher, der zwar normiert, aber natürlich von einem linearen Frequenzgang weit entfernt ist. Wie an den untenstehenden Grafiken deutlich zu erkennen ist, weisen die Lautsprecher ab ca. $8[kHz]$ eine starke Tiefpasswirkung auf.

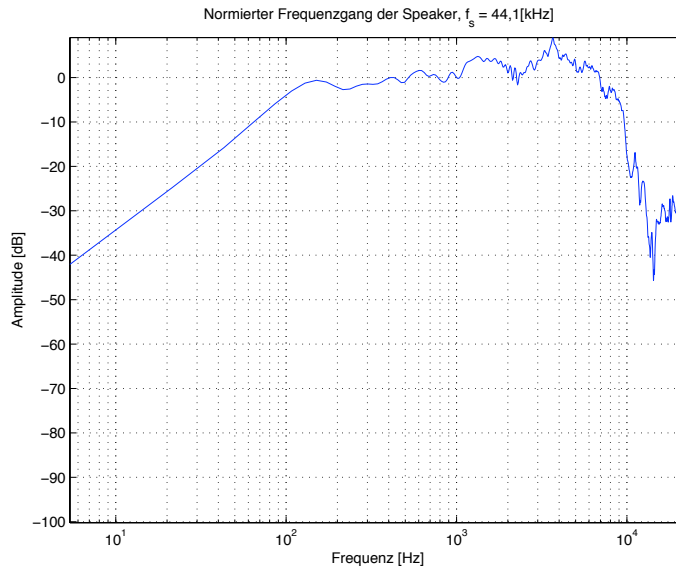


Abbildung 14: Frequenzgang der Lautsprecher - Überblick

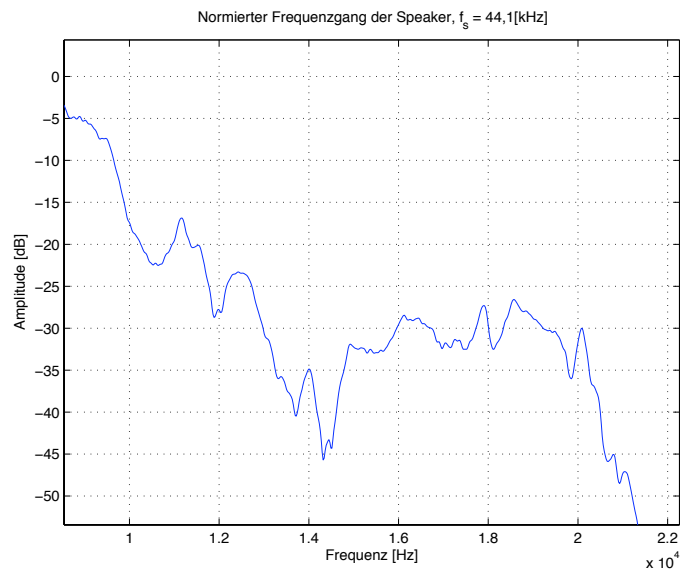


Abbildung 15: Frequenzgang der Lautsprecher - Höhere Frequenzen

Somit ist die kleinere und damit wichtigere obere Grenzfrequenz die, welche durch räumliches Aliasing bedingt ist.

Laut Analyse bedeutet das, dass eine **bis zu 16-fache Unterabtastung möglich** ist:

$$f_{s,min} = \frac{44,1[kHz]}{16} = 2,756[kHz] \Rightarrow f_{g,min} = \frac{44,1[kHz]}{2 \cdot 16} = 1.378[kHz] \quad (1)$$

2.1.3 Reduktion der Filter-Anzahl - Maskierung

Wie bei 2.1.1. erwähnt, stand bereits zu Beginn der Implementierung eine Maske für die Reduzierung der Kontrollmatrix \mathbf{B} zur Verfügung [FZ08]. Wie in *Abb. 16* zu sehen ist, wählt diese Maske die 46 wichtigsten (Betragsmäßig größten) der jeweils 256 Filter aus; dies muss für jeden Frequenzbin, also N_{fft} -mal geschehen.

Die Hauptdiagonale ist vollbesetzt, d.h. jede Eingangs-SH wird auf die entsprechende Ausgangs-SH gleicher Ordnung und gleichen Grades matriziert, was zu erwarten war. Idealerweise sollte nur die Hauptdiagonale besetzt sein, was aufgrund der begrenzten Auflösung der Abtastung allerdings nicht möglich ist. Die anderen gefärbten Matrix-Elemente stehen für Eingangs-SH's, die in Ausgangs-Basisfunktionen anderer Ordnung und/oder anderen Grades mit einfließen, allerdings mit geringeren Beträgen als die jeweils entsprechenden SH aus der Hauptdiagonale.

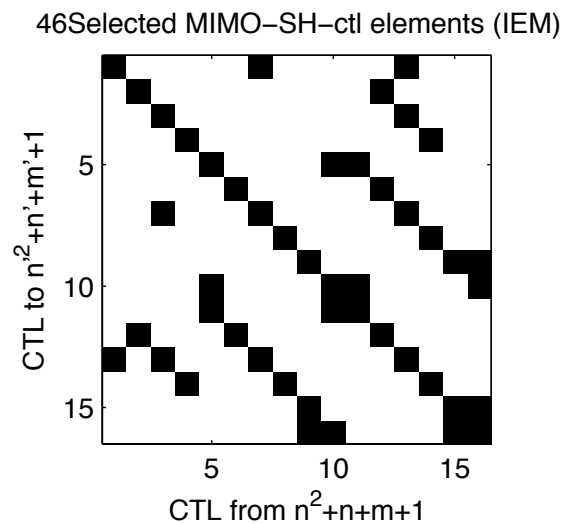


Abbildung 16: Maske für alle 16×16 -Kontrollmatrizen [FZ08]

2.2 Modifikation der Filter

2.2.1 Akausalitäts-Problematik

Um - wie in den folgenden Punkten beschrieben werden wird - die Filter unterabzutasten, sowie um deren Frequenzantworten zu entzerren, ist es nötig, verschiedene Filterungen durchzuführen.

Allerdings wird bei der Durchsicht der Impulsantworten der Kontrollmatrix **B** schnell klar, dass diese durch die akustischen Laufzeiten im System zyklisch verdreht sein können und durch die starken Anhebungen der Frequenzen unterhalb von $100[\text{Hz}]$ in den Frequenzgängen der Filter keinen erkennbaren Impuls aufweisen.

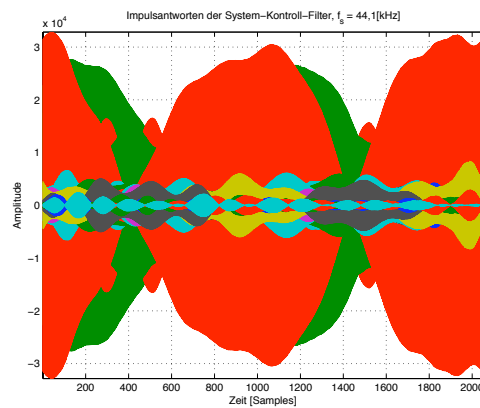


Abbildung 17: Impulsantworten der Kontrollmatrix **B**

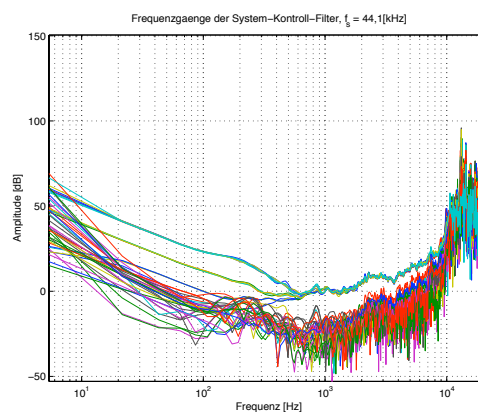


Abbildung 18: Frequenzgänge der Kontrollmatrix **B**

Das Problem stellt sich nun folgendermaßen dar:

Einerseits sind die Impulsantworten verschoben, was aus der durch den Abstand zwischen Lautsprechern und Messmikrofonen entstehenden Laufzeit resultiert. Das bedeutet, dass vor jeder Faltung im Zeitbereich ein zyklisches Zurechtdrehen vorangehen muss um korrekte Ergebnisse erzielen zu können.

Andererseits sind die Impulsantworten derart verfälscht, dass sich ohne Veränderung der Frequenzgänge schwerlich herausfinden lässt, inwieweit eine zyklische Verdrehung vorliegt.

Als Lösung werden sämtliche Filterungen im Frequenzbereich durchgeführt, also eine **zirkuläre Faltung**. Somit ist es möglich, bereits vor einem zyklischen Zurechtdrehen die Frequenzgänge zu verändern und dabei keine nützliche Information zu verlieren.

2.2.2 Unterabtastung

Nachdem zwar die räumlich maximal zulässige Grenzfrequenz bei ca. $800[Hz]$ liegt, das Gehör allerdings diese räumlichen Fehler vermutlich verzeiht, entscheiden wir uns dafür, nicht bis an die Schmerzgrenze der Unterabtastung zu gehen. Wir führen eine Unterabtastung mit dem Faktor 4 durch. Die dadurch modifizierten Parameter **Abtastrate** ($f_{s,new}$) und **obere Grenzfrequenz** ($f_{g,new}$) ergeben sich zu:

$$f_{s,new} = \frac{f_s}{4} = \frac{44100[Hz]}{4} = 11025[Hz]$$

$$f_{g,new} = \frac{f_g}{4} = \frac{22050[Hz]}{4} = 5512,5[Hz]$$

Der benötigte **Antialias-Tiefpassfilter** wird als Butterworth-Filter vierter Ordnung mit der Grenzfrequenz

$$f_{g,TP} = 4,5[kHz]$$

implementiert.

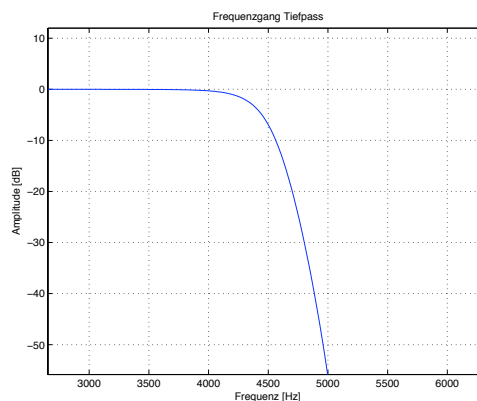


Abbildung 19: Frequenzgang des Antialias-Tiefpass-Filters

Die Unterabtastung und die zugehörige Filterung werden auf die Equalizer-Matrix und auf die Kontrollmatrix \mathbf{B} angewendet. Die Faltung wird als zirkuläre Faltung im Frequenzbereich durchgeführt; anschließend wird im Zeitbereich jedes vierte Sample ausgewählt, der Rest verworfen. In *Abb. 20, 21, 22 und 23* sind die durch die Tiefpass-Filterung erzeugten Veränderungen der Frequenzgänge im alten ($f_{g,old} = 22,05[kHz]$) und im durch die Unterabtastung verkleinerten neuen Frequenzbereich ($f_{g,new} = 5,5125[kHz]$) zu sehen.

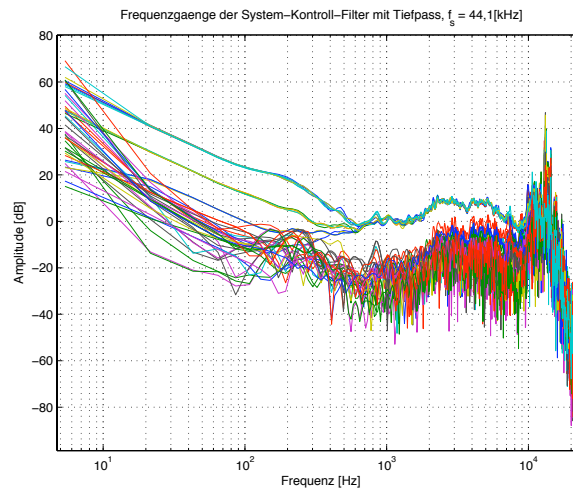


Abbildung 20: Frequenzgänge der Kontrollmatrix \mathbf{B} mit $f_{s,old} = 44,1[kHz]$ und Antialias-Tiefpass-Filter

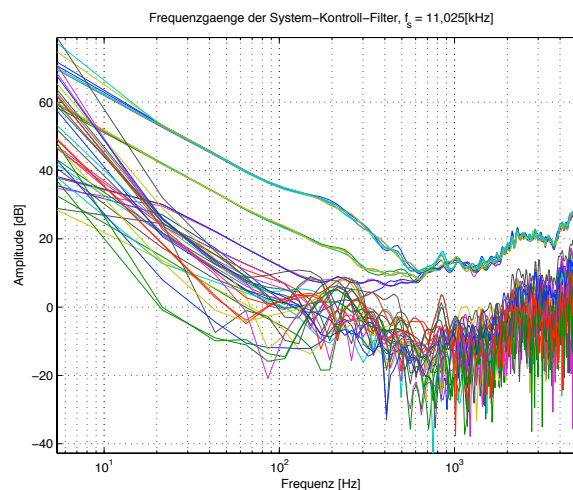


Abbildung 21: Frequenzgänge der Kontrollmatrix \mathbf{B} mit $f_{s,new} = 11,025[kHz]$ und Antialias-Tiefpass-Filter

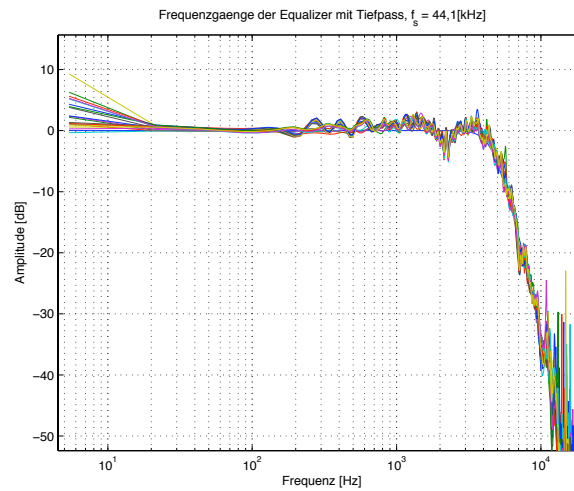


Abbildung 22: Frequenzgänge der Equalizer-Matrix mit $f_{s,old} = 44,1$ [kHz] und Antialias-Tiefpass-Filter

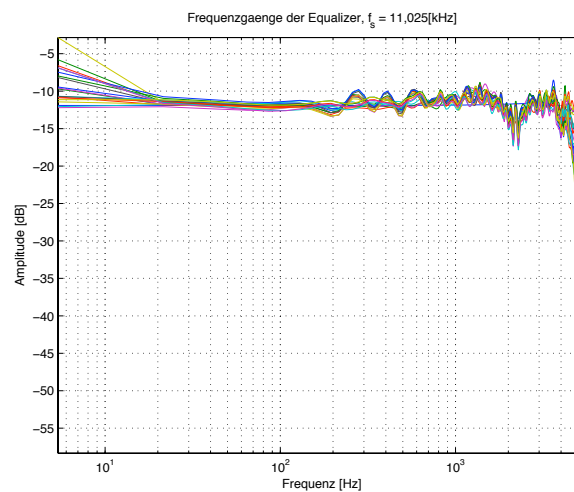


Abbildung 23: Frequenzgänge der Equalizer-Matrix mit $f_{s,new} = 11,025$ [kHz] und Antialias-Tiefpass-Filter

2.2.3 Entzerrung der Frequenzgänge

Einerseits sind die Impulsantworten der Kontrollmatrix \mathbf{B} durch die extremen Anhebungen in den tiefen Frequenzen nach wie vor sehr stark verfälscht. Andererseits übertragen die Lautsprecher ohnehin in diesem Bereich kaum mehr etwas - deshalb ist eine weitere Filterung notwendig.

Wir beschneiden die Frequenzgänge der System-Kontroll-Filter durch eine - ebenfalls zirkuläre - Faltung mit einem **Hochpass-Filter** vierter Ordnung mit der Grenzfrequenz

$$f_{g,HP} = 100[Hz]$$

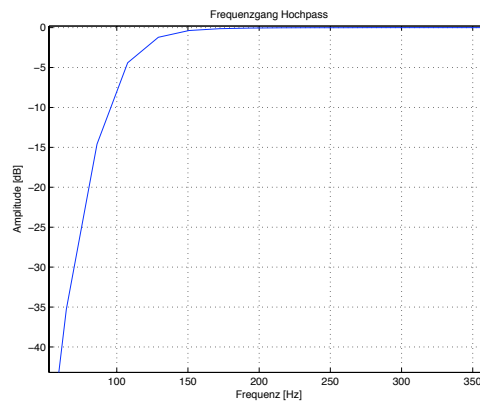


Abbildung 24: Frequenzgang des Hochpass-Filters mit $f_{g,HP} = 100[Hz]$

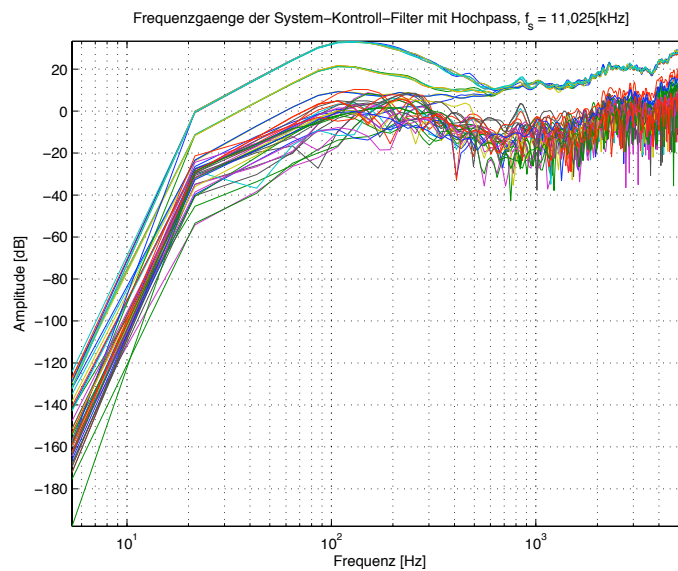


Abbildung 25: Frequenzgänge der Kontrollmatrix mit Hochpass-Filter, $f_{g,HP} = 100[Hz]$

Tatsächlich lassen sich durch diese einfache Maßnahme die Impulsantworten deutlich verbessern. Es ist nun deutlich erkennbar, wie weit diese verschoben sind (siehe Abb. 26).

Anhand der Grafik ist es möglich, das unter 2.2.5 beschriebene zyklische Zurechtdrehen vorzunehmen.

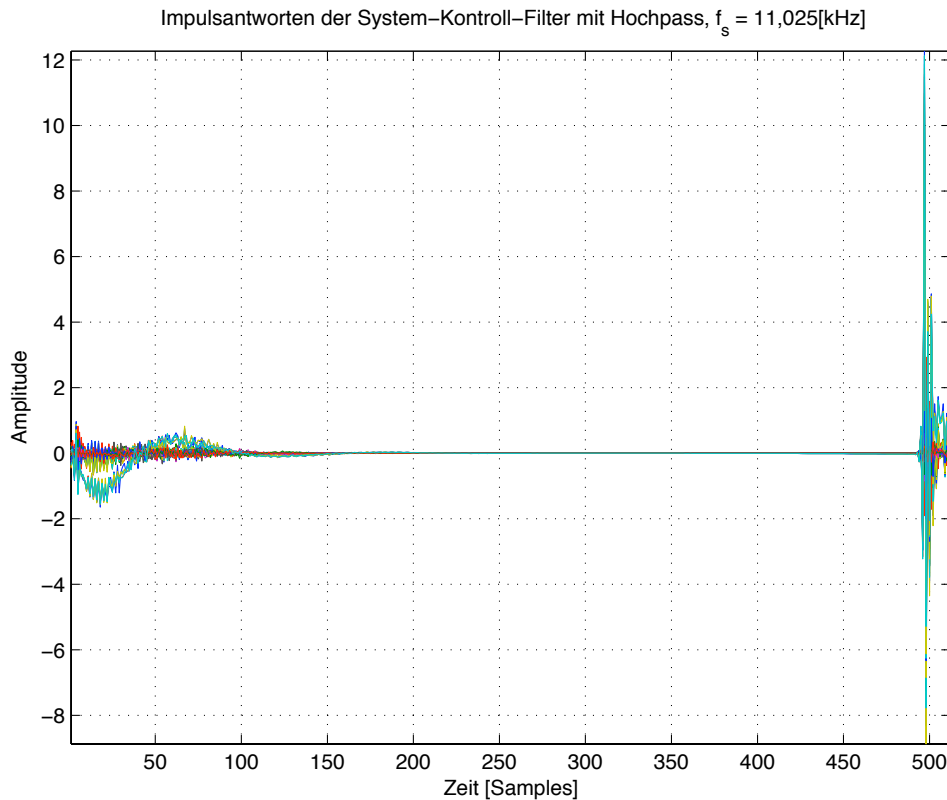


Abbildung 26: Impulsantworten der Kontrollmatrix mit Hochpass-Filter, $f_{g,HP} = 100[\text{Hz}]$

2.2.4 Maskierung der Kontrollmatrix

Umformung der Kontroll-Matrix Um die Maske (siehe dazu 2.1.3) auf die Kontrollmatrix \mathbf{B} anzuwenden, werden die Elemente Zeile für Zeile durchnummeriert, d.h. in der ersten Zeile stehen die Elemente 1 bis 16, in der zweiten Zeile die Elemente 17 bis 32, etc.

Die Einträge der Maske können die Werte Eins oder Null annehmen. Die Einträge mit Eins geben die relevanten Elemente der Kontrollmatrix \mathbf{B} an, die Einträge mit Null die zu verwerfenden.

Wir bilden aus der 16×16 -Matrix einen Vektor der Länge $16 \cdot 16 = 256$ und suchen anschließend die Indices der Einträge gleich Eins. Gleichmaßen wird auch die Kontrollmatrix \mathbf{B} um eine Dimension reduziert, nämlich von $16 \times 16 \times 512$ auf 256×512 - wobei $N_{fft} = 512$ aus dem vierfachen Unterabtasten resultiert.

Als letzter Schritt dieses Vorgangs sind nur noch die 46 Indices der Maske auf die Kontrollmatrix \mathbf{B} anzuwenden, d.h. es werden alle durch die Masken-Indices ausgewählten

Zeilen (also Übertragungsfunktionen) in Abhängigkeit ihrer Index-Größe der Reihe nach in die neue Kontrollmatrix B_{red} des reduzierten Systems geschrieben.

Die Dimensionen der neuen Kontrollmatrix B_{red} ergeben sich daher zu:

$$\text{Dim}(B_{red}) = 46 \times N_{fft} = 46 \times 512$$

Indizierungs-Matrizen Da wir im Echtzeitsystem die Information benötigen, welche Eingangssignals-SH mit welcher Übertragungsfunktion gefaltet werden muss und welche der Faltungsergebnisse für welchen SH-Ausgangs aufsummiert werden müssen, ist es nötig, die entsprechenden Informationen zu übergeben.

Wir realisieren dies durch die Übergabe von zwei weiteren Matrizen, bzw. einem Vektor und einer Matrix:

howmanyins: Dabei handelt es sich um einen 16×1 -Vektor, dessen Zeilen jeweils einen ganzzahligen Eintrag zwischen 2 und 4 beinhalten. Jede Zeile steht für einen der 16 SH-Ausgänge; die Höhe des jeweiligen Eintrages gibt an, wieviele Faltungen für den entsprechenden Ausgang durchzuführen sind. Wie an der Maske (siehe *Abb. 16*) zu erkennen ist, sind in jeder Zeile nicht mehr als vier und nicht weniger als zwei relevante Einträge.

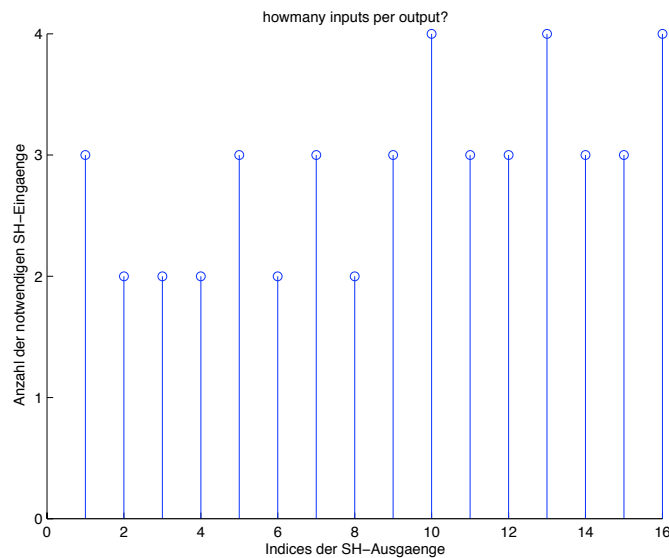


Abbildung 27: Howmanyins - Anzahl der SH-Eingänge für jeden SH-Ausgang

whichins: Um nicht nur die richtige Anzahl an Faltungen pro SH-Ausgang, sondern auch die Indices der jeweiligen SH-Eingänge zu kennen, erstellen wir eine 16×4 -Matrix, welche in jeder Zeile maximal vier Indices zwischen 1 und 16 beinhaltet, der Rest ist Null.

Jede Zeile gibt nun an, welche SH-Eingänge für die entsprechenden SH-Ausgänge aufsummiert werden müssen.

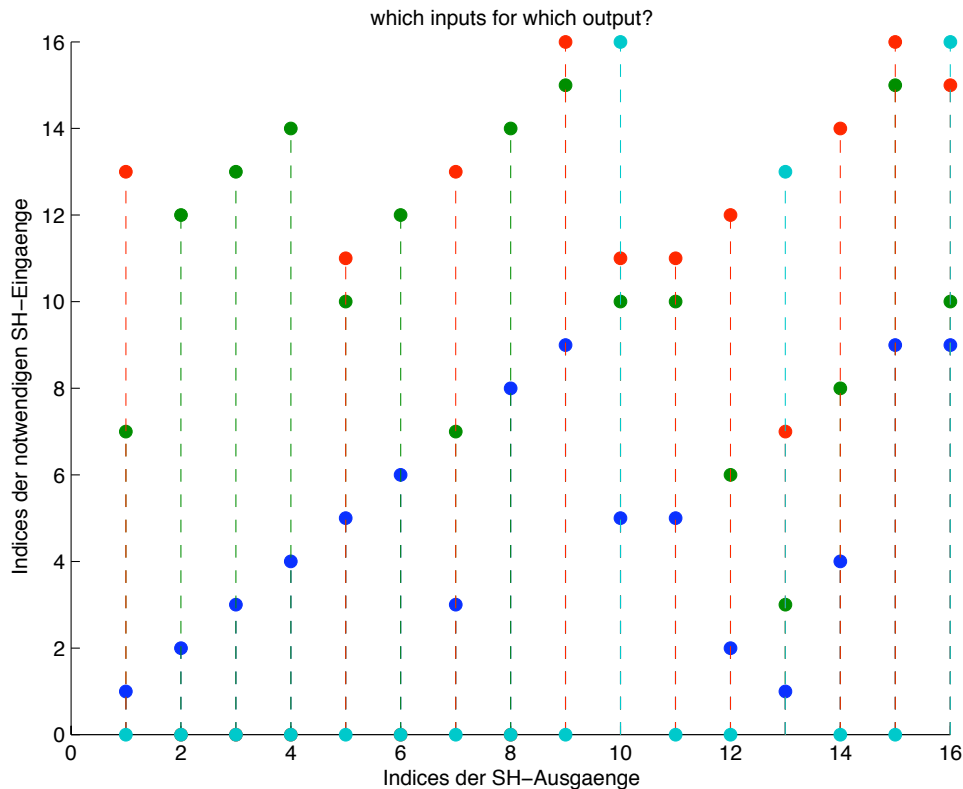


Abbildung 28: Whichins - Indices der SH-Eingänge für jeden SH-Ausgangs

2.2.5 Fenstern und Zyklisches Verdrehen der Impulsantworten

Schließlich ist noch das zyklische Zurechtdrehen offen. Aus den Hochpass-gefilterten Impulsantworten von B_{red} können wir die Größe der Verschiebung grafisch ermitteln (Abb. 26). Die Impulsantworten verschieben wir nach hinten, während der dadurch am Ende herausfallende Teil wieder vorne eingefüllt wird. Mit einer Verschiebung um 30 Samples erzielen wir das untenstehende Ergebnis. Zudem nehmen wir eine Fensterung der Impulsantworten mittels eines Hanning-Fensters vor (10 Samples am Anfang, 300 Samples am Ende), sodass diese sanft ein- und ausgeblendet werden.

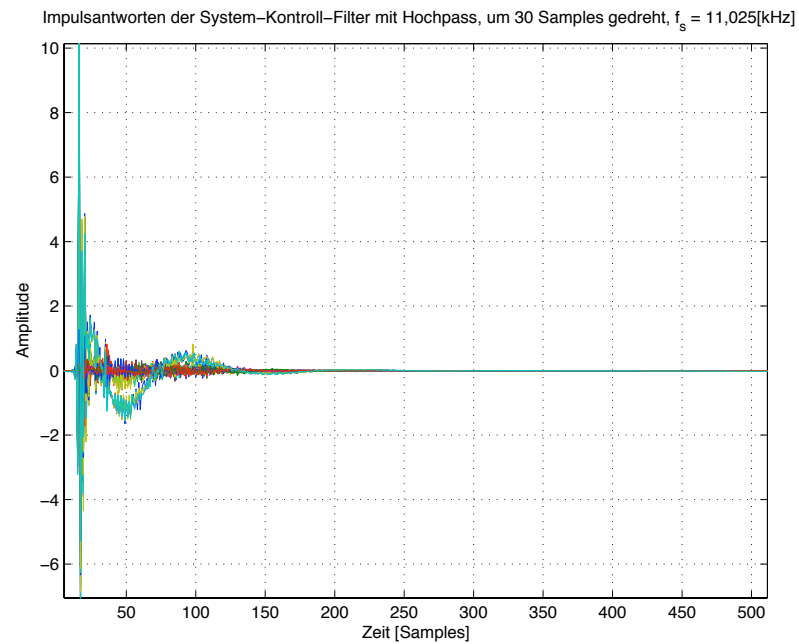


Abbildung 29: Impulsantworten von B_{red} , um 30 Samples zyklisch verdreht

Eine leichte Verdrehung nehmen wir auch bei den Equalizern vor, allerdings ist hier nur mit einer Verschiebung um 3 Samples:

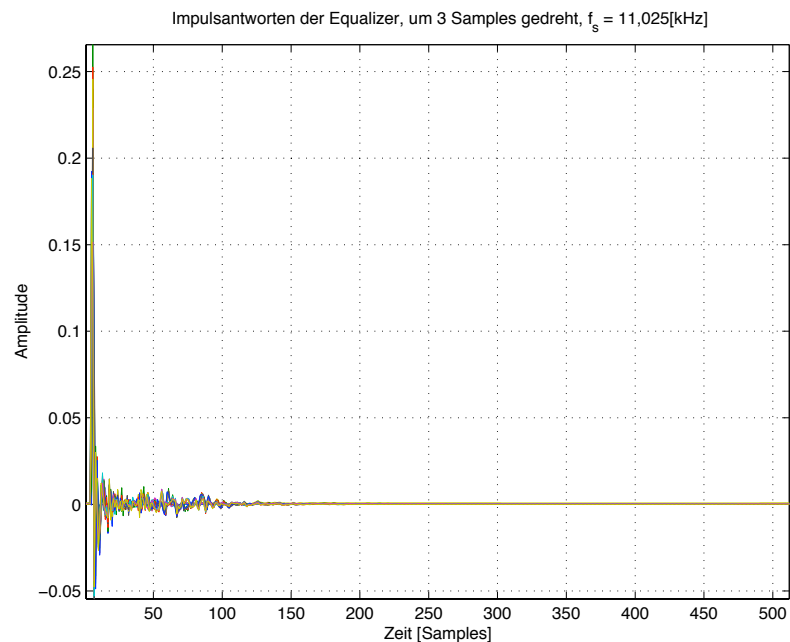


Abbildung 30: Impulsantworten der Equalizer, um 3 Samples zyklisch verdreht

2.2.6 Grafischer Überblick - Filtermodifikation

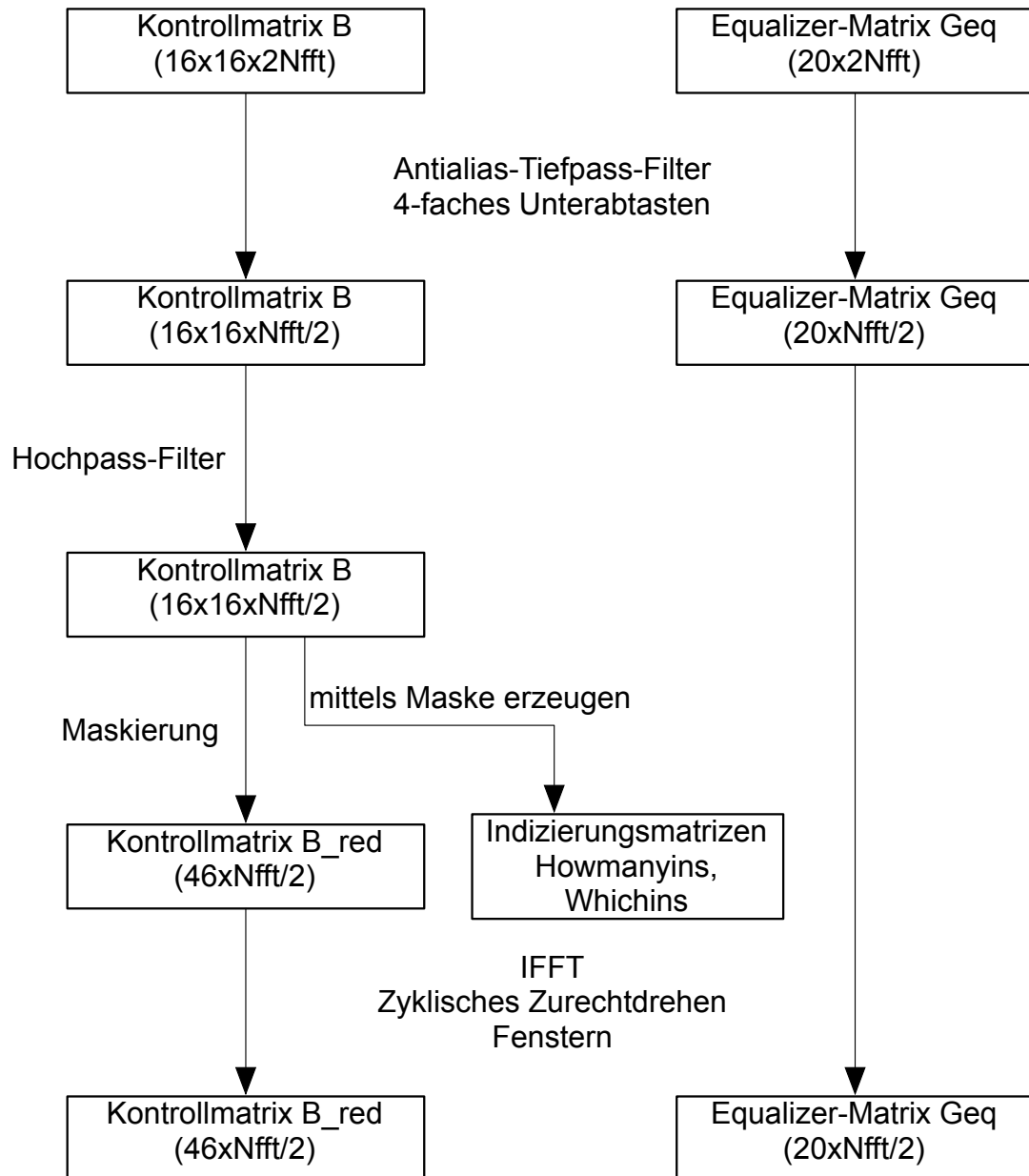


Abbildung 31: Modifikation der Filter-Matrizen - Schema und Reihenfolge

2.3 Implementierung des Echtzeitsystems in Pure Data

2.3.1 Beschreibung der einzelnen Schritte

Mit den - wie in 2.2 beschrieben - modifizierten Matrizen haben wir nun alle nötigen Informationen um die Echtzeit-Implementierung in Pure Data realisieren zu können.

Die Filter-Matrizen, d.h. G_{EQ} und B_{red} werden zur Vereinfachung im **Zeitbereich** übergeben. Außerdem importieren wir die beiden Indizierungsmatrizen „Howmanyins“ und „Whichins“.

Eine kurze Zusammenfassung der einzelnen Rechenschritte innerhalb des Echtzeitsystems folgt:

Transformation des Eingangssignals in SH's Die aktuelle Implementierung arbeitet mit nur einem Mono-Eingangssignal, welches mittels Steuerung der Azimuth- und Elevationswinkel in die gewünschte Richtung abgestrahlt wird. Um das Eingangssignal in sphärische Harmonische zu zerlegen verwenden wir ein bereits vorhandenes External (mtx_spherical_harmonics), das für die beiden eingegebenen Winkel die Anteile des Signals für jede SH berechnet.

Variabilität des Radius' Um nicht auf den Radius der Messung (80cm) beschränkt zu sein, verwenden wir auch einen am IEM programmierten Beamformer (sh_radial_beamformer), welcher die Abstrahlungsrichtung mit der Abstands-Information versieht.

Unterabtasten und Blocken des Eingangssignals Natürlich muss auch das Eingangssignal auf ein Viertel der ursprünglichen Abtastrate gebracht werden. Dazu wird jedes SH-Signal mit einem Butterworth-Tiefpass-Filter (6. Ordnung/ $f_g = 3,7[kHz]$) bearbeitet um Aliasing zu vermeiden.

Anschließend werden die Signale vierfach unterabgetastet und auf die erforderliche Blockgröße $N_{fft}/2$ gebracht.

Overlap Um die Filterung durchzuführen müssen die Eingangssignale auf die doppelte Blocklänge überlappt werden, d.h. ein aktueller Signalblock er Länge $N_{fft}/2$ wird an einen alten Signalblock der Länge $N_{fft}/2$ gereiht. Die Faltung wird also im Frequenzbereich mit der neuen Länge

$$N_{fft} = 1024$$

gerechnet. Die 16 SH-Signale werden zudem in Matrizenform verpackt und als solche übergeben.

FFT Um die Faltung in PD umzusetzen müssen zunächst die Kontrollmatrix B_{red} und das in sphärische Harmonische zerlegte Eingangssignal in den Frequenzbereich transformiert werden. Die N_{fft} Punkte lange Transformation in PD liefert den imaginären und

den reellen Anteil in zwei getrennten Matrizen, d.h. die folgenden Schritte werden jeweils doppelt ausgeführt.

Umformen der SH-Eingangssignale Mittels der Indizierungsmatrizen „Howmany-ins“ und „Whichins“ läuft eine Schleife über die Indices der SH-Eingänge, wodurch diese ausgewählt und auf die gleiche (Matrix-) Form gebracht werden wie B_{red} , also 46×1024 . Jedes SH-Eingangssignal wird in jene Zeile geschrieben, in welcher B_{red} die zugehörige Übertragungsfunktion enthält.

Zyklische Faltung Durch Zeilenweise Multiplikation (im Frequenzbereich) der Eingangsmatrix und B_{red} wird die Faltung berechnet. Anschließend wird die inverse Fourier-Transformation berechnet und die Signale so wieder in den Zeitbereich gebracht.

Verwerfen des zyklischen Faltungsergebnisses Die immer noch mit doppelter Blockgröße vorhandenen Signale werden wieder auf die ursprüngliche Blockgröße gebracht, indem der Block mit dem unerwünschten zyklischen Faltungsergebnis verworfen wird.

Summierung der Signale für jeden SH-Ausgang Hier wird mittels der Matrix Howmanyins eine Schleife durchlaufen, welche die gefilterten Eingangs-Basisfunktionen der richtigen Ausgangs-Basisfunktion zuordnet. Die jeweiligen Eingangs-Basisfunktionen werden dann für jede Ausgangs-Basisfunktion aufsummiert.

Dekodierung Die nun entstandenen 16 SH-Ausgänge werden mit einem Ambisonics-Dekoder auf 20 Lautsprechersignale matriziert.

Equalizer Um schließlich die Lautsprecher noch zu entzerren, werden die 20 Ausgangssignale mit den 20 Equalizer-Übertragungsfunktionen gefaltet. Dies geschieht im Prinzip genauso wie oben beschrieben, also Overlap and Add, FFT, Multiplikation der imaginären und reellen Matrizen, Rücktransformation, Verwerfen des zyklischen Faltungsergebnisses. Im Unterschied zu dem obigen Prozedere wird hier allerdings aus jedem Eingang genau ein Ausgang berechnet (single Input, single Output).

Überabtasten Die Lautsprecher-Signale werden wieder auf die ursprüngliche Abtastrate ($f_{s,old} = 44,1[kHz]$) gebracht. Zum Glätten der Signale wird zudem noch eine weitere Antialias-Tiefpass-Filterung (6. Ordnung/ $f_g = 3,7[kHz]$) durchgeführt.

2.3.2 Grafischer Überblick - Echtzeit-System

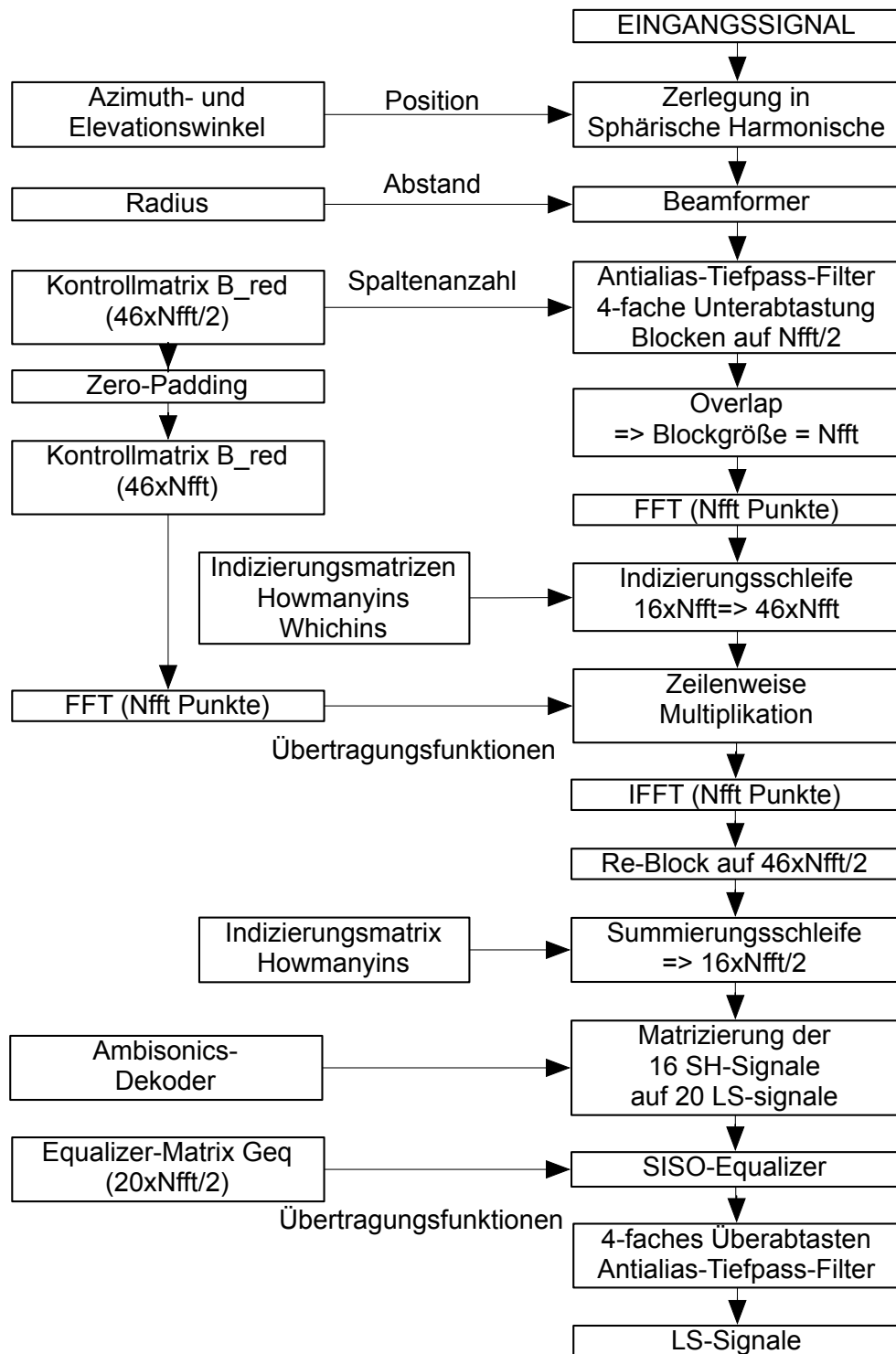


Abbildung 32: Echtzeitsystem - Signalfluss

3 Schlussfolgerungen und Ausblick

Nach einigen Anfangsproblemen funktioniert das neue System sehr gut, wenn auch bislang noch keine Evaluation mittels Hörversuchen oder Ähnlichem stattfand. Der Rechenaufwand darf - obwohl es auch diesbezüglich noch keine genauere Untersuchung gibt - als deutlich verringert angenommen werden. Diese Annahme ergibt sich schon alleine aus der Tatsache, dass die Anzahl der in Echtzeit zu berechnenden Filter von 256 auf 46 reduziert wurde. Das entspricht einem knappen Fünftel des nichtreduzierten Systems. Trotzdem ist eine weitere Rechenleistungs-Optimierung dieses Systems sicherlich möglich. Einerseits besteht Möglichkeit einer weiteren 4-fachen Unterabtastung, d.h. insgesamt 16-fache Unterabtastung, was allerdings eine Grenzfrequenz von $f_g = 1,378[kHz]$ zur Folge hat. Andererseits könnte man auch durch eine härtere Fensterung der Impulsantworten möglicherweise noch Rechenaufwand einsparen.

Da in absehbarer Zeit die Lautsprecher des Iksaeder-Lautsprecher-Arrays ausgewechselt werden sollen, wird dieser auch neu vermessen werden müssen. Nicht zuletzt deshalb wurde bei der Implementierung darauf geachtet, das System möglichst flexibel anzulegen, sodass die neuen Daten hoffentlich reibungslos übernommen werden können.

Literatur

- [FZ08] A. S. F. Zotter, H. Pomberger, "Efficient directivity pattern control for spherical loudspeaker arrays," Institute of Electronic Music and Acoustics, Graz-Austria, Tech. Rep., 2008.
- [Pom08] H. Pomberger, "Angular and radial directivity control for spherical loudspeaker arrays," Master's thesis, Institute of Electronic Music and Acoustics, Graz-Austria, 2008.