

Seminararbeit aus Algorithmen in Akustik und Computermusik 2, SE

## **Automatische Lautsprecherentzerrung mit Kautz-Filtern**

Sebastian Braun (0673052)  
Marco Schretter (0230932)

Betreuung: Franz Zotter  
Graz, 29. März.201



institut für elektronische musik und akustik



# 1. Einleitung

Lautsprecher ('LS') haben oft keinen idealen Frequenzgang, der den Klang grob beeinträchtigt. Für bestimmte Anwendungen wie z. B. Ambisonics und vor allem in Zusammenhang mit dem natürlichen Nachhall und den Moden eines Raumes ergeben sich besonders unvorteilhafte Klangverzerrungen.

Um einen Lautsprecher zu entzerren gibt es drei Möglichkeiten: Komplette nach Gehör, Messung des LS und anschließend händische Entzerrung, oder aber eine voll automatisierte Entzerrung.

Im letzten Fall wird die Impulsantwort des LS gemessen und sofort der Filter berechnet, mit welchem die LS-Impulsantwort einen linearen Frequenzgang gewährleisten soll.

Basierend auf dem Artikel *'Equalization of Loudspeaker and Room Responses Using Kautz Filters: Direct Least Squares Design'* von Matti Karjalainen und Tuomas Paatero [EJ, AES] soll ein solcher Patch in Pure Data mit Hilfe von Kautz Filtern realisiert werden.

## 2. Kautz Filter

### 2.1 Allgemeine Theorie

Die Motivation zur Verwendung von Kautz Filtern gegenüber anderen Methoden ergibt sich vor allem aus der Möglichkeit, Probleme bei der Invertierung (des Frequenzganges) einer gemessenen Impulsantwort und sog. 'ringing poles' (direkter EQ-Ansatz) zu umgehen und dabei auch die Filterordnung zum Erhalten befriedigender Entzerrungskurven gering zu halten [EJ S. 1].

Der Ansatz des Kautz Filters ist relativ einfach. Er geht von in Serie geschalteten Blöcken aus, die z. B. Verzögerungsglieder oder Allpässe sein können. Zwischen jedem Block wird ein Tap-Output abgegriffen, der jeweils mit einer Gewichtung  $w_i$  versehen wird. In Summe ergeben die Tap-Outputs das Ausgangssignal.

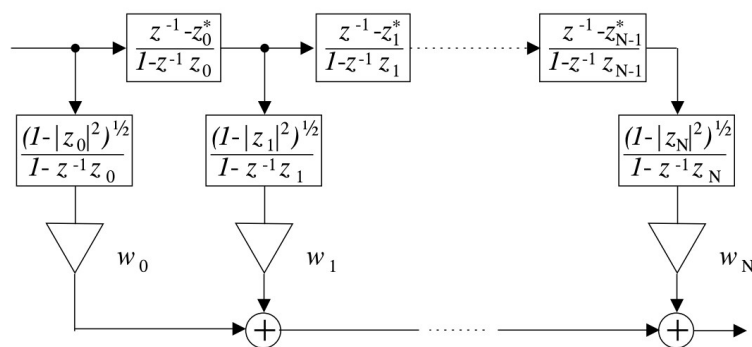


Abbildung 1: Allgemeine Struktur eines Kautzfilters [AES S. 2]

Die mathematische Darstellung im Zeitbereich ergibt sich zu

$$\hat{h}(n) = \sum_{i=0}^N w_i g_i(n),$$

wobei  $g_i$  die Impulsantworten der Filterblöcke sind, die auf Allpässen mit gewählten Polen basieren, auch als Basisfunktionen  $g_i$  zu bezeichnen. Das wichtige bei dieser Struktur ist, dass die Basisfunktionen orthogonal zueinander sind, da  $\langle g_i, g_k \rangle = 0$  für  $i \neq k$  [AES S. 3].

Der Ansatz für die direkte LS Entzerrung entsteht aus dem Ansatz, dass die Impulsantwort (IR) des Filters  $h_{EQ}(n)$  gefaltet mit der IR des gemessenen Systems (LS und Raum)  $h_R(n)$  die gewünschte IR  $h_T(n)$  ergeben soll. Für einen linearen Frequenzgang ist das ein Delta-Impuls,  $h_T(n) = \delta(n)$ .

$$h_E(n) = h_{EQ}(n) * h_R(n) \approx h_T(n) \quad [AES S. 3].$$

Zur Lösung der Gleichung wird ein Least-Squares Ansatz verwendet, um die gewünschte IR  $h_T(n)$  so gut wie möglich mit  $h_E(n)$  anzunähern.

$$\min (\sum (h_E(n) - h_T(n))^2),$$

wobei die entzerrte IR die Summe der mit  $w_i$  gewichteten Tap-Outputs ist:

$$h_E(n) = \sum w_i x_i(n), \quad x_i(n) = g_i(n) * h_R(n).$$

## 2.2 Gleichungen in Matrizenform

Um den Umgang mit den Gleichungen zu vereinfachen, schreibt man sich die Gleichungen in Matrix-Notation an.

Definition der Signale als Vektoren und Matrizen:

$$\mathbf{g}_i[n] = (g_i[0] \quad \dots \quad g_i[k-1])^T \quad \dots \text{ist die Impulsantwort der } i\text{-ten Kautz Filterstufe mit der Länge } k.$$

Fasst man alle Kautz Filterstufen mit der Anzahl  $I$  zusammen, so erhält man die Matrix  $\mathbf{G}$  mit den Dimensionen  $(I \times K)$ .

$$\mathbf{G} = \begin{pmatrix} g_1[0] & \dots & g_1[k-1] \\ g_2[0] & \dots & g_2[k-1] \\ \vdots & \ddots & \vdots \\ g_I[0] & \dots & g_I[k-1] \end{pmatrix}.$$

$\mathbf{x}[n] = (x[n] \quad x[n-1] \quad \dots \quad x[n-k+1])^T \quad \dots$  ist ein Vektor mit Mono-Eingangssamples und hat die selbe Länge wie die Impulsantworten der Kautz Filterstufen.

$$\mathbf{s}[n] = \mathbf{G} \cdot \mathbf{x}[n] = \begin{pmatrix} \mathbf{g}_1^T \cdot \mathbf{x}[n] \\ \vdots \\ \mathbf{g}_I^T \cdot \mathbf{x}[n] \end{pmatrix} \quad \text{Durch die Multiplikation von } \mathbf{s} = \mathbf{G} \mathbf{x} \text{ ergibt sich der (Kreuz-)}$$

Korrelationsvektor zwischen den Antworten der Filterausgänge und der Eingangssamples.  $\mathbf{s}$  ist ein Mehrkanalvektor der Länge  $I$  mit einem einzigen Ausgangssample, zB:

$$\mathbf{s}[0] = (s_1[0] \quad \dots \quad s_I[0])^T.$$

Ein Ausgangssample des Kautz Filters ergibt sich dann aus:

$$\mathbf{y} = \mathbf{s}^T \mathbf{w} = \mathbf{x}^T \mathbf{G}^T \mathbf{w} \quad \text{mit} \quad \mathbf{w} = (w_1 \quad \dots \quad w_I)^T \quad \text{als Gewichte der einzelnen Kautzstufen.}$$

Möchte man das gesamte Ausgangssignal als Vektor

$\mathbf{y} = (y[n] \quad \dots \quad y[B-1])^T$  mit der Länge B mit den Ausgangssamples  $y$  untereinander erhalten, so schreibt man zeitlich verschobene Versionen des Eingangssignalvektors  $\mathbf{x}$  zirkulär in die Matrix  $\mathbf{X}$  mit den Dimensionen (K x B) untereinander (Töplitzstruktur):

$$\mathbf{X}^T = \begin{pmatrix} x[0] & \dots & x[k-1] \\ x[1] & \dots & x[k] \\ \vdots & \ddots & \vdots \\ x[B-1] & \dots & x[k+B] \end{pmatrix}.$$

Es ergibt sich daraus die Gleichung  $\mathbf{y} = \mathbf{X}^T \mathbf{G}^T \mathbf{w} = \mathbf{S} \mathbf{w}$ , wobei  $\mathbf{y}$  zu einem Spaltenvektor

$$\mathbf{y} = (y[n] \quad \dots \quad y[B-1])^T \quad \text{und} \quad \mathbf{s} \quad \text{zu einer Matrix} \quad \mathbf{S} = \mathbf{X}^T \mathbf{G}^T \quad \text{wird.}$$

$\mathbf{S}$  ist dann die sog. channel convolution Matrix (B x I), in deren Zeilen jeweils ein  $\mathbf{s}^T$  Vektor steht:

$$\mathbf{S} = \begin{pmatrix} s_1[0] & \dots & s_I[0] \\ s_1[1] & \dots & s_I[1] \\ \vdots & \ddots & \vdots \\ s_1[B-1] & \dots & s_I[B-1] \end{pmatrix}.$$

Nun setzt man das Least Squares Problem so an, dass der Abstand zwischen Filteroutput  $\mathbf{y}$  und gewünschter IR  $\mathbf{h}_T$  minimal werden soll.

$$\begin{aligned} \|\mathbf{h}_T - \mathbf{y}\|^2 &\rightarrow \min \\ (\mathbf{h}_T - \mathbf{S}\mathbf{w})^T (\mathbf{h}_T - \mathbf{S}\mathbf{w}) &\rightarrow \min \end{aligned}$$

d.h. Die 1. Ableitung des Terms muss 0 werden:

$$\mathbf{S}^T \mathbf{S} \mathbf{w} - \mathbf{S}^T \mathbf{h}_T = 0.$$

Da die gewünschte Target IR  $h_T(n) = \delta(n - \Delta)$  mit einer bestimmten Verzögerung behaftet ist, schneidet der  $\mathbf{s}$ -Vektor die  $(\Delta+1)$ te Zeile aus der  $\mathbf{S}$  Matrix heraus.

Daraus folgt die Normalgleichung  $\mathbf{S}^T \mathbf{S} \mathbf{w} = \mathbf{s}$ .

Die Lösung, um die Gewichte zu bekommen, erreicht man über Inversion der Matrizen, was der Pseudo-Inversen entspricht.

$$\mathbf{w} = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{s} = (\mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T)^{-1} \mathbf{s}.$$

Die Inversion der Matrix wäre garantiert, wenn  $\mathbf{S}$  nur aus den zueinander orthogonalen Basisfunktionen  $g_i$  und dem Eingangssignal  $\mathbf{x}$  besteht:

$$\mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T = \mathbf{G} \mathbf{G}^T = \mathbf{I}.$$

Angenommen  $\mathbf{XX}^T$  wäre  $\mathbf{I}$  (die Einheitsmatrix), was einem LS mit idealem Frequenzgang entspricht, der nicht entzerrt werden muss. In diesem Fall sieht man, dass noch  $\mathbf{GG}^T$  invertiert werden müsste.

Das Versprechen und somit der große Vorteil der Kautz Filterstruktur ist, dass  $\mathbf{GG}^T$  bereits  $\mathbf{I}$  entspricht. Dies bedeutet, die Kautz Filterbank selbst verursacht also keine Schwierigkeiten in unserem Gleichungssystem.

Es können nur noch Probleme auftreten, wenn  $\mathbf{XX}^T$  nicht invertierbar ist.

### 3. Realisierung

#### 3.1 Grundgedanke

Für die Implementation in PD haben wir aufgrund der existierenden Filter von PD eine Realisierung des Kautz Filters mit Biquad-Blöcken und einfachen Nullstellen-Filtern gewählt [Abbildung 1]. Die Berechnung der Polradien und Filterkoeffizienten konnten wir offline in Matlab prozessieren und anschließend an PD übergeben. Somit ist die Filterordnung bzw. die Polverteilung unseres Kautz Filters durch die Vorverarbeitung in Matlab festgelegt, der PD-Patch kann aber problemlos durch automatisches Patchen in PD dynamisch an die übergebenen Filterordnungen aus Matlab angepasst werden.

#### 3.2 Verwendete Struktur des Kautz Filters

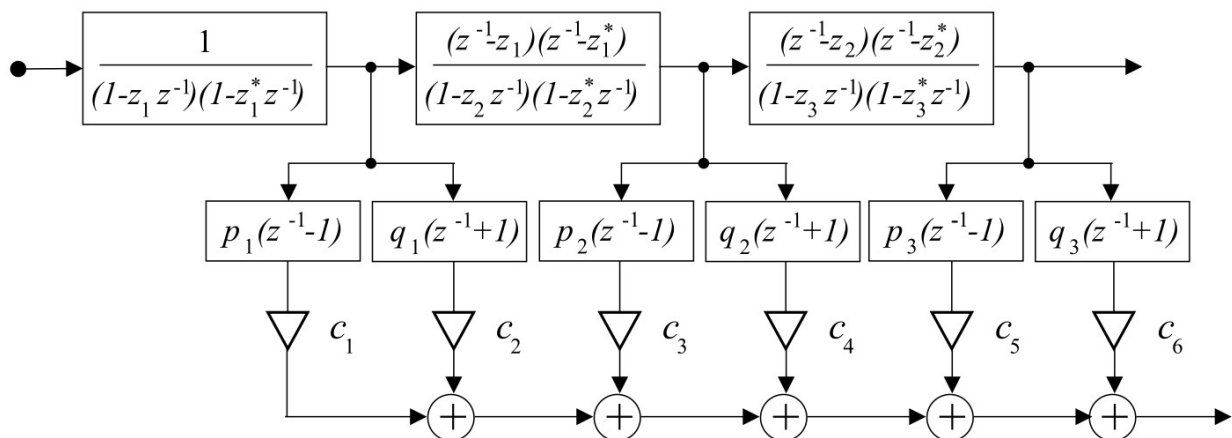


Abbildung 2: Verwendete Kautzfilter Realisierung mit Biquad-Blöcken [EJ S. 4]

In unserer Struktur wurden Allpassfilter 2. Ordnung verwendet, in Abbildung 2 in der oberen Reihe zu sehen. Für jeden Allpassblock gibt es zwei Anzapfungen, jeweils einem komplex konjugierten Polpaar [EJ S. 3] entsprechend. Diese Anzapfungen gehen durch Normalisierungsterme, die nur aus Nullstellenfiltern bestehen und werden anschließend mit  $c_i$  gewichtet (Anm.: hier entspricht  $c_i = w_i$ ).

### 3.3 Wahl der Polstellen

Bei einer fixen Wahl der Polstellen, wie sie in unserem Fall verwendet wurde, müssen nur noch die Gewichte für den Filter berechnet werden. Dabei wird die gemessene IR  $h_R(n)$  durch den Kautz Filter geschickt und die Anzapfungen vor der Gewichtung abgegriffen. Diese Anzapfsignale  $s[n]$  werden dann spaltenweise in die Matrix  $S$  geschrieben.

Um eine möglichst sinnvolle Verteilung der Pole zu erhalten, welche die Auflösung des Filters um die entsprechende Frequenz bestimmen, wurde eine logarithmische Frequenzauflösung mit einer auf einer log-Achse konstanten Güte gewählt. Dieses entspricht am besten der Auflösung des menschlichen Gehörs in Anlehnung an die ERB Skala [AES S. 5].

Da eine fixe Polverteilung gewählt wurde, konnte die Berechnung der Polstellen auch extern in Matlab erfolgen. Es wurden 23 Pole gewählt, die logarithmisch von 100 Hz an startend aufwärts verteilt sind, zu sehen in Abbildung 3. Die Q-Auflösung der Bandpässe ergibt sich durch ein logarithmisches Abklingen der Pol-Radien, wie in Abbildung 4 dargestellt.

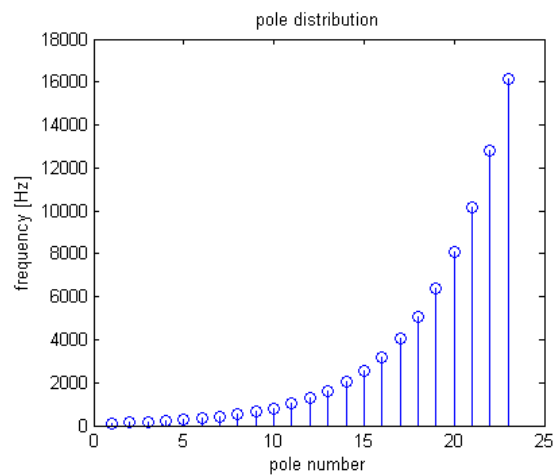


Abbildung 3: logarithmische Verteilung der Polfrequenzen

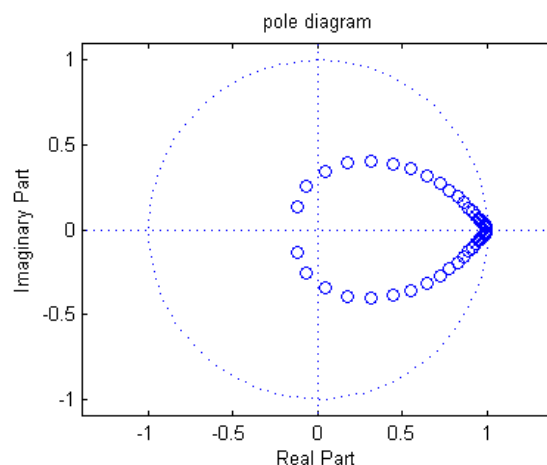


Abbildung 4: Poldiagramm mit komplex konjugierten Polpaaren

Die Pole wurden in Matlab direkt in die benötigten Biquad- und Nullstellen-Filterkoeffizienten umgerechnet (lt. PD-Hilfe Quotientengleichungen) und automatisch als Matrix-Datei exportiert, die von PD eingelesen werden kann.

Somit muss man nur die Startfrequenz des tiefsten Pols und die Anzahl der Pole im Skript wählen und es ausführen, um ein anderes Polset zu bekommen. Anschließend kann man direkt den PD Patch ausführen, wenn er im selben Verzeichnis wie das Matlab Skript liegt.

### ***3.4 IR-Messung und Kautz Filter in PureData (PD)***

Die IR-Messung des LS (incl. des Raumes), der Aufbau des Kautz Filters, die Berechnung der Gewichte und die Ausgabe des frequenzgangkorrigierten Signals erfolgt in PD.

Wir benötigen folgende Tools zur Realisierung: iemmatrix, zexy und iemlib.

Beim Start des Patches wird bereits die von Matlab generierte Matrix mit den Filterkoeffizienten eingelesen. Der Kautz Filter wird dynamisch nach Eingabe der Ordnung (Anzahl der Polpaare) generiert (default 23 Polpaare).

Der erste Schritt ist die Messung des LS (plus Raumimpulsantwort). Es werden 3 exponentielle Sinussweeps ausgespielt und aufgenommen. Diese werden gemittelt und die minimalphasige Impulsantwort berechnet.

Danach wird die errechnete IR in den dynamisch aufgebauten Kautz Filter geschickt, die Tap-Outputs abgegriffen und in die **S**-Matrix geschrieben.

Durch die Minimalphasigkeit der IR steht der gewünschte **s**-Vektor als Ergebnis der Kreuzkorrelation der Tap-Outputs und unserer Target IR (Delta-Impuls) für die anschließende Multiplikation mit der Pseudo-Inversen von **S** ohne das o.a. Delay in der ersten Zeile von **S**.

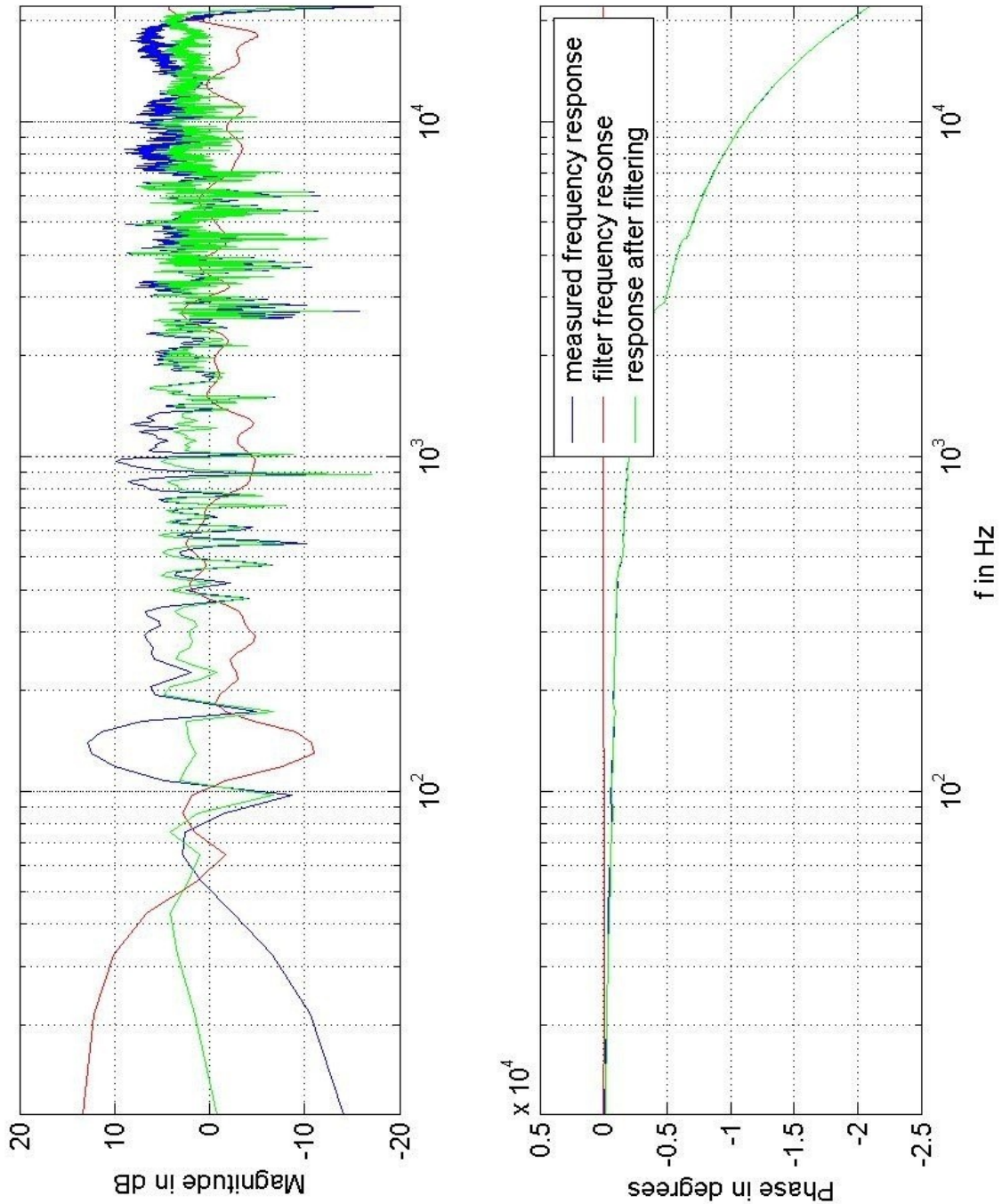
Als letzter Schritt wird die Gleichung  $\mathbf{w} = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{s} = (\mathbf{X} \mathbf{G}^T \mathbf{G} \mathbf{X}^T)^{-1} \mathbf{s}$  ausgeführt und die Gewichte in den Kautz Filter geschrieben, womit der LS Frequenzgang entzerrt ist.

Zum Vergleich der IR-Messung vor der Korrektur mit dem korrigierten Frequenzgang in Matlab werden die Ergebnisse in wav-Files geschrieben. Gleichzeitig wird auch die Impulsantwort des Entzerrungsfilters gemessen und als wave Datei gespeichert. Die Beurteilung und das Plotten erfolgt im Matlab Skript „filteranalyse“.

### 3.5 Messergebnisse

Messsetup Abbildung 5:

Raum: Zimmer Schretter (23 m<sup>2</sup>), Lautsprecher: Adam A7 (Gain 0 dB), Messmikro: Behringer ECM-8000, Soundkarte & Preamp: RME FF-400 (Gain 40 dB), Messdistanz: 1,5 m





## 4. Schlussfolgerungen und Ausblick

Mit der von uns verwendeten Kautz-Struktur mit logarithmisch angeordneten komplexen Polpaaren hat sich bei einer Filterordnung von 23 eine recht zufriedenstellende Idealisierung des Frequenzganges realisieren lassen.

Durch das Versprechen der Orthogonalität der Filterimpulsantworten der Kautz-Stufen vereinfachte sich das Lösen der Gleichungen und des Least Squares Problems.

Entscheidender Vorteil ist die Möglichkeit recht einfach dem System eine gewünschte Zielimpulsantwort vorgeben zu können. In unserem Fall haben wir ein lineares Ziel-Frequenzspektrum angestrebt, es kann jedoch auch zB ein HP-gefilterter Dirac-Impuls als Zielfunktion dienen, um zu starke Korrekturen (siehe Abb. 5) des Systems im tieffrequenten Bereich zu vermeiden .

Natürlich können Notches im LS Frequenzgang nicht kompensiert werden, jedoch Überhöhungen und breitbandigere Unebenheiten. Im Gesamtbild sind steile Notches jedoch weniger auffällig als Überhöhungen. Das Verhalten des Kautz Filters ist aber stark von der Filterordnung und der Wahl der Polstellen der Filterstufen abhängig.

## 5. Literaturverzeichnis

[EJ] Karjalainen, Matti und Paatero, Tuomas, *Equalization of Loudspeaker and Room Responses Using Kautz Filters: Direct Least Squares Design* (EURASIP Journal on Advances in Signal Processing, 2007, Article 60949).

[AES] Paatero, Tuomas und Karjalainen, Matti, *Equalization of audio systems using Kautz Filters with log-like frequency resolution* (Paris: Audio Engineering Society, 2006, Convention Paper 6767).

Paatero, Tuomas - Kautz Filter Design (incl. MatLab Examples):

<http://www.acoustics.hut.fi/software/kautz/kautz.htm>