Adaptive MISO-Filter zur möglichst konstruktiven Mehrkanalsignalüberlagerung

TI Projekt

Rainer Mittmannsgruber

Betreuung: Dr. Franz Zotter

Graz, 18. Mai 2016

institut für elektronische musik und akustik

Zusammenfassung

Das IEM verfügt über die Möglichkeit, mit einer umgebenden Kugelmikrofonanordnung Musikinstrumente aufzunehmen. Es ist ein sinnvoller Ansatz, das rundum abgestrahlte Schallsignal eines Instrumentes als ein Quelle-Abstrahlungsfitermodell zu betrachten. Danach ergibt sich der Schall, der in eine Richtung abgegeben wird aus dem Quellsignal (Ursignal) gefiltert mit dem jeweiligen Abstrahlungsfilter. Nur sind beide, das Quellsignal und die Abstrahlungsfilter, nicht bekannt und müssen erst aus der umgebenden Mikrofonkugelaufnahme gewonnen werden. Diese Arbeit be schäftigt sich mit der Umsetzung eines Quelle-Filter-Modells für diese Anwendung und analysiert die Qualität eines solchen Ansatzes. Eine solche Zerlegung der Schallabstrahlung ermöglicht eine aufschlussreiche Darstellung und bietet auch trickreiche Vereinfachungsmöglichkeiten, wie zum Beispiel die Datenreduktion für die Übertragung einer solchen mehrkanaligen Aufnahme.

Abstract

The facilities at IEM provide the opportunity to record a musical instrument within a spherical microphone array. It is a reasonable approach to consider a source filter model for the sound radiation of a musical instrument. According to this approach, the emitted sound in a certain direction consists of a primary signal filtered by the associated radiation filter. However, both primary signal and radiation filters are unknown and have to be obtained from the recorded microphone signals. This project concentrates on the implementation of a source filter model for this task and analyzes the quality of this approach. The decomposed sound radiation offers insight and illustration methods as well as tricky simplifications, such as a data reduction for transmission of such multichannel recordings.

Inhaltsverzeichnis

1 Einleitung

Die Möglichkeit zur Aufnahme eines Musikinstruments mit einer umgebenden Kugelmikrofonanordnung wurde im Rahmen der Diplomarbeit von F. Hohl [Hoh09] konstruiert und ist in Abb. 1 dargestellt. Eine derartige Aufnahme bietet vielzählige Möglichkeiten zur Analyse und Darstellung der Klangquelle.

Abbildung 1: Umgebende Kugelmikrofonanordnung [Hoh09]

Bei einer mehrkanaligen Aufnahme dieser Art drängt sich die Frage auf, ob aus den aufgenommenen Signalen ein Signal, im weiteren Ursignal genannt, generiert werden kann, welches alle spektralen Anteile besitzt und somit das Ursprungssignal darstellt. Für diese Problemstellung finden sich mehrere Lösungsansätze. Der Ansatz, die Betragsspektren der einzelnen Signale aufzusummieren und aus der resultierenden Summe das zugehörige Phasenspektrum abzuschätzen, wurde in der Arbeit von S. Süss untersucht [Süs11]. Dabei basiert der Lösungsweg auf dem Algorithmus von Griffin und Lim [GL84], wobei hier die vorhandene Phaseninformation vernachlässigt wird. Ist das Ursignal anschließend bekannt, besteht die Möglichkeit die Datenmenge bei einer Übertragung zu reduzieren, wie in Abb. 2 dargestellt ist. Andere Ansätze welche die bekannte Phaseninformation nicht verwerfen, wurden hierbei ebenfalls umgesetzt, führten aber zu keinem zufriedenstellenden Ergebnis. In der Arbeit von C. Hollomey [Hol14] wurde versucht einen ähnlichen Ansatz, der das Phasenspektrum vom Betragsspektrum abschätzt, in Echtzeit zu implementieren.

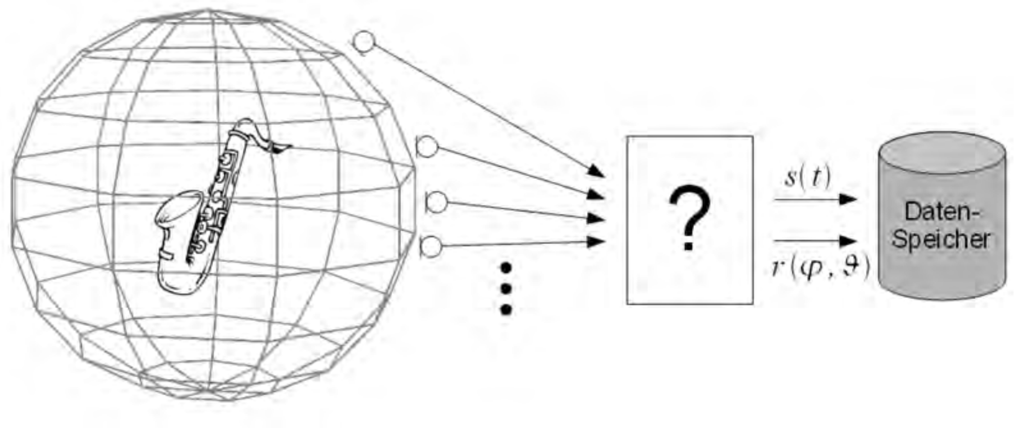

Abbildung 2: Schema zur Kodierung von Schallabstrahlung in Ursignal und Abstrahlungsrichtung [Süs11]

In dieser Arbeit soll nun die Berechnung eines Ursignals auf einem Quelle - Filter - Modell basieren. Abb. 3 zeigt eine derartige Zerlegung in Quelle und dem jeweiligen Mikrofonsignal zugeordneten Abstrahlungsfilter.

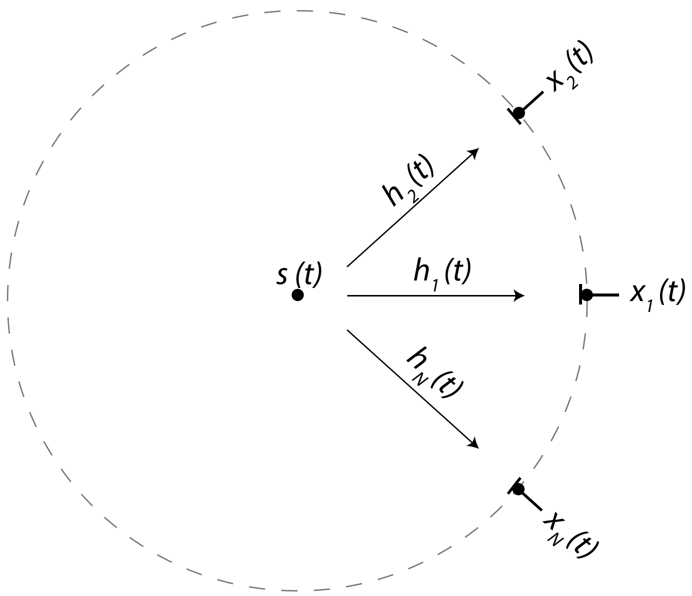

Abbildung 3: Schallabstrahlung als Quelle - Filter Modell

Basierend auf diesem Modell können mit Hilfe von SIMO - Signalverarbeitung (*Single Input Multiple Output*) die jeweiligen Abstrahlungsfilter, ohne Kenntnis des Ursignals, abgeschätzt werden. Sind die Filter bekannt, kann mit Hilfe des MINT - Theorems (*Multi-Channel Inverse Theorem*) ([HBC06] Kap. 7) das Ursignal schlussendlich berechnet werden. Der analytische Ansatz für die blinde Kanalidentifikation basiert auf den Ausführungen von J. Huang, J. Benesty und J. Chen [HBC06].

Zudem wird in dieser Arbeit noch ein zweiter, neuer Ansatz verfolgt: Neben der Abschätzung der Abstrahlungsfilter und daraus folgender Berechnung mittels dem MINT - Theorem, lassen sich nämlich auch solche Filter berechnen, die das Ursignal direkt aus den Mikrofonsignalen abschätzen. Dabei sorgt ein Algorithmus dafür, dass die Signalenergie des resultierenden Signals maximiert wird. Im folgenden Abschnitt werden beide Ansätze analytisch diskutiert.

2 Analytische Betrachtung und Algorithmen

In diesem Abschnitt werden zwei Möglichkeiten aufgezeigt, um ein Ursignal aus einer Aufnahme mit einer umgebenden Mikrofonanordnung zu gewinnen. In beiden Algorithmen ist die zirkuläre Faltung ein wesentlicher Bestandteil der mathematischen Formulierung, in Vorbereitung einer schnellen blockbasierten Umsetzung mittels FFT. Im folgenden Abschnitt wird dieser Sachverhalt näher betrachtet, um im Anschluss bei der Herleitung der Algorithmen Anwendung zu finden.

2.1 Zirkuläre Faltung im Frequenzbereich

Handelt es sich bei einem Signal x um eine sehr lange, im Prinzip unendliche Eingangsfolge, kann die direkte Implementation der Faltung mit einem Signal h,

$$y(t) = x(t) * h(t)\,, \tag{2.1}$$

zu einer sehr rechen- und zeitaufwändigen mathematischen Operation werden. Es ist daher ein sinnvoller Ansatz die Eingangsfolge x in kurze Teilfolgen zu zerlegen, deren Faltung mit dem Signal h zu berechnen und aus den so gewonnenen Teilsignalen das Ergebnis y zusammenzusetzen. In Vektorschreibweise kann (2.1) als

$$y(k) = \boldsymbol{x}^T(k)\boldsymbol{h} \tag{2.2}$$

angeschrieben werden. Betrachtet man nun eine Blockverarbeitung mit einer Blocklänge von $2L$ und dem Index b, so entspricht

$$\tilde{\boldsymbol{y}}(b) = \boldsymbol{X}(b)\tilde{\boldsymbol{h}}(b) \tag{2.3}$$

dem Ergebnis der zyklischen Faltung. Angestrebt wird zwar das Ergebnis der linearen Faltung, die zyklische Faltung erlaubt allerdings die effiziente Umsetzung als schnelle Faltung. Dabei ist

$$\boldsymbol{X}(b) = \begin{bmatrix} x(bL) & x(bL+2L+1) & \dots & x(bL+1) \\ x(bL+1) & x(bL) & \dots & x(bL+2) \\ \vdots & \vdots & \ddots & \vdots \\ x(bL+2L+1) & \dots & \dots & x(bL) \end{bmatrix}$$

die $2L$x$2L$ Zirkulantmatrix eines Datenblocks. Dieser Datenblock wird alle L Abstastzyklen gebildet, sodass sich die resultierenden Datenblöcke um 50% überlappen. Für eine lineare Faltung sind in (2.3) sowohl das Ergebnis, als auch $\tilde{\boldsymbol{h}}(b)$ zu schneiden. Dazu werden die ersten L-Punkte von $\tilde{\boldsymbol{h}}(b)$ und die letzten L-Punkte im Ergebnis mit den Fenstermatrizen $\boldsymbol{W}^{10}$ und $\boldsymbol{W}^{01}$ freigestellt. (2.3) wird damit zu, vgl. [HBC06],

$$\boldsymbol{y}(b) = \boldsymbol{W}^{01}\boldsymbol{X}(b)\boldsymbol{W}^{10}\boldsymbol{h}(b)\,, \tag{2.4}$$

wobei die entsprechenden Fenstermatrizen sich aus Nullmatrizen und Einheitsmatrizen zusammensetzen:

$$\boldsymbol{W}^{01} = \begin{bmatrix} \mathbf{0}_{LxL} & \mathbf{0}_{LxL} \\ \mathbf{0}_{LxL} & \mathbf{1}_{LxL} \end{bmatrix} \qquad \boldsymbol{W}^{10} = \begin{bmatrix} \mathbf{1}_{LxL} & \mathbf{0}_{LxL} \\ \mathbf{0}_{LxL} & \mathbf{0}_{LxL} \end{bmatrix} . \tag{2.5}$$

Die Fourier-Transformationsmatrix

$$\boldsymbol{F} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & e^{-j\frac{\pi}{L}} & e^{-j\frac{2\pi}{L}} & \dots & e^{-j\frac{\pi(2L-1)}{L}} \\ 1 & e^{-j\frac{2\pi}{L}} & e^{-j\frac{4\pi}{L}} & \dots & e^{-j\frac{2\pi(2L-1)}{L}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{\pi(2L-1)}{L}} & e^{-j\frac{2\pi(2L-1)}{L}} & \dots & e^{-j\frac{\pi(2L-1)^2}{L}} \end{bmatrix}$$

lässt uns das Verfahren im Frequenzbereich anschreiben. Weil sich ein zyklischer zeitlicher Versatz im Frequenzbereich durch komponentenweise komplexe skalare Faktoren (Einheitszeiger mit linearer Phase über der Frequenz) ausdrückt, diagonalisiert eine beidseitig angewandte Fouriertransformationsmatrix die Zirkulantmatrix X

$$\boldsymbol{X}(b) = \boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F}}(\omega, b)\boldsymbol{F} . \tag{2.6}$$

Die Einträge in der Diagonalmatrix entsprechen dadurch der Fouriertransformation eines Zeitblocks

$$\boldsymbol{X}_{\mathrm{F}}(\omega, b) = diag\left(FFT\{\boldsymbol{x}(b)\}\right) . \tag{2.7}$$

Wird der (ungeschnittene) Vektor $\boldsymbol{h}$ auch aus dem Frequenzbereich erzeugt, so wird er ausgedrückt durch

$$\tilde{\boldsymbol{h}}(b) = \boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F}}(\omega, b) . \tag{2.8}$$

Setzt man nun (2.6) und (2.8) in die Definition eines Datenblocks im Zeitbereich ein, vgl. (2.4), so erhält man mit

$$\boldsymbol{y}(b) = \boldsymbol{W}^{01}\boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F}}(\omega, b)\boldsymbol{F}\,\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F}}(\omega, b) \tag{2.9}$$

das lineare Faltungsergebnis eines Datenblocks mit dem Vektor $\boldsymbol{h}$. Durch die Diagonalisierung der Matrix $\boldsymbol{X}(b)$, entsteht das Faltungsergebnis durch punktweise Multiplikation im Fourierbereich. Da die Transformationen $\boldsymbol{F}^{-1}$ und $\boldsymbol{F}$ nicht durch Matrixmultiplikationen gelöst werden müssen, sondern durch die schnelle Fouriertransformation implementiert werden können, existiert eine hoch effiziente Umsetzung der Kompaktschreibweise in (2.9). Die FFT und die Kompaktschreibweise ermöglichen uns die Herleitung effizienter adaptiver Filter.

2.2 Blinde Kanalidentifikation und Ursignalbildung

Der grundsätzliche Ansatz von blinder SIMO - Kanalidentifikation (BCI) [HBC06] besteht darin, dass sich aus der Definition eines Mikrofonsignals als die Faltung eines unbekannten Quellsignals mit einem unbekannten Filter

$$x_1 = s * h_1 \tag{2.10}$$

der Zusammenhang

$$x_1 * h_2 = s * h_1 * h_2 = x_2 * h_1 \tag{2.11}$$

herstellen lässt. Sind die Signale x_1 und x_2 bekannt, so lassen sich die Filter h_1 und h_2 ohne Wissen über das Quellsignal s berechnen, sofern die Filterübertragungsfunktionen keine gemeinsame Nullstellen haben. Allgemein lässt sich dieser Zusammenhang in Vektorschreibweise als

$$\boldsymbol{x}_i^T(k)\boldsymbol{h}_j = \boldsymbol{x}_j^T(k)\boldsymbol{h}_i \qquad \forall k \quad i,j = 1,2,\dots,N \qquad i \neq j \tag{2.12}$$

zusammenfassen [HBC06]. Wird nun die in (2.4) gezeigte Schreibweise für das lineare Faltungsergebnis eines Datenblocks angewandt, so ergibt sich (2.12) zu

$$\boldsymbol{W}^{01}\boldsymbol{X}_i(b)\boldsymbol{W}^{10}\boldsymbol{h}_j(b) = \boldsymbol{W}^{01}\boldsymbol{X}_j(b)\boldsymbol{W}^{10}\boldsymbol{h}_i(b)\,, \tag{2.13}$$

wobei $\boldsymbol{X}_i(b)$ die $2L$x$2L$ Zirkulantmatrix eines Datenblocks des jeweiligen Kanals, und $\tilde{\boldsymbol{h}}_i(b)$ $\tilde{\boldsymbol{h}}_j(b)$ die dem Kanal zugeordnete Filterimpulsantwort ist.

Werden nun die Signale gemäß (2.6) bis (2.8) aus dem Frequenzbereich erzeugt, ergibt sich

$$\boldsymbol{W}^{01}\boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F},i}(\omega,b)\boldsymbol{F}\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega,b) - \boldsymbol{W}^{01}\boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F},j}(\omega,b)\boldsymbol{F}\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F},i}(\omega,b) = 0\,. \tag{2.14}$$

Wenn die Filter $\tilde{\boldsymbol{h}}_{\mathrm{F},i}$ und $\tilde{\boldsymbol{h}}_{\mathrm{F},j}$ nicht genau bekannt sind und die Signalkette auch Störungen beinhaltet, wird (2.14) nicht mehr zu 0 sondern es entsteht der Fehler $\boldsymbol{e}_{ij}$. Nun gilt es jene Filter zu finden, welche zum minimal erreichbaren Fehler führen. Hierzu lässt sich ein LMS-Verfahren zur Fehlerminimierung, mit der Nebenbedingung von normierten Filterkoeffizienten, ansetzen:

$$\sum_{i,j=1}^{N} \|\boldsymbol{e}_{ij}\|^2 \longrightarrow \min\,,$$

$$\text{Nebenbedingung} \quad \|\boldsymbol{h}(b)\| = 1\,, \quad \text{wobei} \quad \boldsymbol{h}(b) = \left[\boldsymbol{h}_1^T\,\boldsymbol{h}_2^T\,\dots\,\boldsymbol{h}_N^T\right]^T\,. \tag{2.15}$$

Diese Nebenbedingung dient zur Sicherstellung, dass der Algorithmus nicht zu einer trivialen Lösung durch einen Nullvektor konvergiert. Die Kostenfunktion lässt sich mit dem Lagrangemultiplikator λ als

$$J(b) = \sum_{i,j=1}^{N} \|\boldsymbol{e}_{ij}(b)\|^2 + \lambda\left(\|\boldsymbol{h}(b)\| - 1\right) \tag{2.16}$$

anschreiben, die bei korrekten Filterkoeffizienten $\tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega,b)$ minimal wird. Es ist jedoch leichter, mit einer Kostenfunktion zu rechnen die von normierten Koeffizienten ausgeht

$$J(b) = \sum_{i,j=1}^{N} \|\boldsymbol{e}_{ij}(b)\|^2\,. \tag{2.17}$$

Für das LMS-Verfahren bildet sich die Kostenfunktion im Sinne des MMSE über den Erwartungswert

$$J = \mathbb{E}\left\{J(b)\right\} = \mathbb{E}\left\{\sum_{i,j=1}^{N} \left\|\boldsymbol{e}_{ij}\right\|^2\right\}, \tag{2.18}$$

der durch den instantanen Wert $J(b)$ abgeschätzt wird. Die Aktualisierungsgleichung der in Richtung einer fallenden Kostenfunktion (Gradient)

$$\tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega, b) = \tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega, b-1) - \mu \nabla J(b) \tag{2.19}$$

verbessert die Filterkoeffizienten schrittweise. Der Gradient von $J(b)$ errechnet sich mit:

$$\begin{aligned} \frac{\partial J}{\partial \tilde{\boldsymbol{h}}^H_{\mathrm{F},j}(\omega, b)} &= \sum_{i,j=1}^{N} \boldsymbol{e}_{ij}^H \boldsymbol{e}_{ij} \\ &= 2 \sum_{i=1}^{N} \left(\boldsymbol{W}^{01}\boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F},i}(\omega, b)\boldsymbol{F}\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\right)^H \boldsymbol{e}_{ij} \end{aligned} \tag{2.20}$$

Eingesetzt in (2.19) ergibt sich für die Aktualisierungsgleichung

$$\tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega, b) = \tilde{\boldsymbol{h}}_{\mathrm{F},j}(\omega, b-1) - 2\tilde{\mu} \sum_{i=1}^{N} \left(\boldsymbol{W}^{01}\boldsymbol{F}^{-1}\boldsymbol{X}_{\mathrm{F},i}(\omega, b)\boldsymbol{F}\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\right)^H \boldsymbol{e}_{ij} \tag{2.21}$$

Damit die Aktualisierungsgeschwindigkeit weniger von der Energie der Korrelation abhängig ist, wird die Schrittweite zusätzlich auf die 2-Norm der Mikrofonsignale bezogen

$$\tilde{\mu} = \frac{\mu}{\left\|\boldsymbol{x}_{\mathrm{F}}(b)\right\|^2 + reg}\,. \tag{2.22}$$

Um der angenommenen Vereinfachung gerecht zu werden, und um zu verhindern, dass der Algorithmus zu einer trivialen Lösung durch einen Nullvektor konvergiert, erfolgt nach jeder Aktualisierung eine Normierung der Koeffizienten mit

$$\boldsymbol{h}(b) = \frac{\tilde{\boldsymbol{h}}(b)}{\left\|\tilde{\boldsymbol{h}}(b)\right\| + reg}\,. \tag{2.23}$$

2.2.1 Berechnung des Ursignals

Aus den gewonnenen Kanalantworten lässt sich ein interferenzfreies, aber noch klanggefärbtes Ursignal durch die Summe aller Mikrofonsignale bilden, nachdem jedes der Mikrofonsignale mit seiner zeitlich umgekehrten Kanalantwort gefaltet wird

$$\tilde{s}(t) = \sum_{i}^{N} x_i(t) * h_i(-t)\,. \tag{2.24}$$

Mit Hilfe eines Entzerrungsfilters wird das resultierende Signal $\tilde{s}(t)$ von den Klangfärbungen durch die zeitumgekehrten Kanalantworten befreit

$$s(t) = \tilde{s}(t) *^{-1} h_{EQ}\,. \tag{2.25}$$

Erfolgt die Berechnung des Ursignals ebenfalls mit Blockverarbeitung der Länge $2L$, so kann (2.24) wieder als Ergebnis einer zyklischen Faltung angeschrieben werden

$$\tilde{\boldsymbol{s}}(b) = \sum_{i=1}^{N} \boldsymbol{H}_{\mathrm{rev},i}(b)\boldsymbol{x}_i(b)\,, \tag{2.26}$$

wobei

$$\boldsymbol{H}_{\mathrm{rev},i}(b) = \begin{bmatrix} h_i(bL+2L+1) & h_i(bL) & \dots & h_i(bL+2L) \\ h_i(bL+2L) & h_i(bL+2L+1) & \dots & h_i(bL+2L-1) \\ \vdots & \vdots & \ddots & \vdots \\ h_i(bL) & \dots & \dots & h_i(bL+2L+1) \end{bmatrix}$$

die Zirkulantmatrix einer zeitlich umgekehrten Impulsantwort ist. Wendet man die Definitionen in (2.6) und (2.8) analog auf (2.26) an, so ergibt sich

$$\tilde{\boldsymbol{s}}(b) = \boldsymbol{W}^{01}\boldsymbol{F}^{-1} \sum_{i=1}^{N} \boldsymbol{H}_{\mathrm{F,rev},i}(\omega, b)\boldsymbol{x}_{\mathrm{F},i}(\omega, b)\,, \tag{2.27}$$

wobei $\boldsymbol{H}_{\mathrm{F,rev},i}(\omega, b)$ die Diagonalmatrix einer zeitlich umgekehrten, gefensterten Impulsantwort im Fourierbereich darstellt. Die anschließende Filterung kann ebenfalls als Ergebnis einer zyklischen Faltung angeschrieben werden

$$\boldsymbol{s}(b) = \boldsymbol{H}_{sum}^{-1}\tilde{\boldsymbol{s}}(b)\,, \tag{2.28}$$

wobei $\boldsymbol{H}_{sum}$ den Entzerrungsfilter in Form einer Impulsantwort als Zirkulantmatrix darstellt. Im Fourierbereich wird der Ausdruck zu

$$\boldsymbol{s}(b) = \boldsymbol{F}^{-1}\tilde{\boldsymbol{H}}_{\mathrm{F},sum}^{-1}\tilde{\boldsymbol{s}}_{\mathrm{F}}(\omega, b)\,. \tag{2.29}$$

Für die Bildung des Entzerrungsfilters stehen mehrere Möglichkeiten zur Verfügung, je nachdem worauf normiert werden soll:

$\boldsymbol{h}_{\mathrm{F},sum}(\omega,b)=\sum_{i=1}^{N}\lvert\boldsymbol{h}_{\mathrm{F},i}(\omega,b)\rvert^2$	Normierung auf ein spektral weißes Ursignal
$\boldsymbol{h}_{\mathrm{F},sum}(\omega,b)=\sqrt{\sum_{i=1}^{N}\lvert\boldsymbol{h}_{\mathrm{F},i}(\omega,b)\rvert^2}$	Normierung auf interverenzfreie, klangneutrale Ursignalbildung (Pseudoallpass)
$\boldsymbol{h}_{\mathrm{F},sum}(\omega,b)=\frac{\sum_{i=1}^{N}\lvert\boldsymbol{h}_{\mathrm{F},i}(\omega,b)\rvert^2}{max_i\lvert\lvert\boldsymbol{h}_{\mathrm{F},i}(\omega,b)\rvert}$	Normierung auf Klangfärbung des lautesten Kanals

Es ist jedoch zu beachten, dass die Inverse des Filters gebildet werden muss. Weil die zeitliche Länge von Inversfiltern im Allgemeinen unendlich ist, können zyklische Überlappungen etwa dadurch minimiert werden, dass das Filter minimalphasig gemacht wird. Von Nutzen ist dabei die Eigenschaft des Cepstrums, dass alle minimalphasigen Nullstellen kausale Signalanteile, und maximalphasige Nullstellen akausale Signalanteile erzeugen. Wird nun mit Hilfe der Hilbert-Transformation ein analytisches Cepstralsignal erzeugt, der negative Quefrenzanteil also unterdrückt, erzeugt das ein minimalphasiges Filter.

Zusammengefasst bildet dieser Ansatz ein Ursignal, indem zuerst jene Impulsantworten abgeschätzt, die am ehesten in der Lage sind, aus einem einzelnen Signal die Mikrofonsignale zu erzeugen, um dann mit der Umkehr dieser Impulsantworten auf das Ursignal zurückzurechnen. Angesichts dieser zweistufigen Implementation drängt sich die Frage auf, ob es nicht möglich ist, das Ursignal direkt aus den adaptiv gefilterten Mikrofonsignalen zu gewinnen. Diese Überlegung führt uns zu einem neuen Ansatz, eines Allpassverfahrens, das im kommenden Abschnitt vorgestellt wird.

2.3 Allpassverfahren

Werden alle Mikrofonsignale ungefiltert überlagert, können bei dem resultierenden Summensignal durch Interferenz Signalanteile verstärkt oder abgeschwächt werden. Möchte man ein Signal bilden, das alle Signalanteile enthält, lässt sich diese Forderung dadurch ausdrücken, dass sich alle Mikrofonsignale möglichst konstruktiv überlagern sollen. Demzufolge müsste ein durch Filterung und Summation gewonnenes Ursignal die Energie maximieren. Auf Basis dieser Überlegung lässt sich nun eine Kostenfunktion für ein neues Verfahren ansetzen, das adaptiv geeignete Filter bestimmt.

2.3.1 Berechnung des Ursignals mit Hilfe der Fouriertransformation

Gesucht sind die Filter $h_i(t)$ mit denen das Ursignal $\hat{s}(t)$ direkt aus den Mikrofonsignalen abgeschätzt werden kann. Dabei ergeben sich N Teilsignale

$$s_i(t) = x_i(t) * h_i(t)\,, \tag{2.30}$$

deren Summe die gesuchte Schätzung des Ursignals ergibt:

$$s(t) = \sum_{i=1}^{N} x_i(t) * h_i(t)\,. \tag{2.31}$$

Verwendet man für ein Teilsignal $s_i(t)$ die Schreibweise aus (2.4), um eine Blockverarbeitung einzuführen und das lineare Faltungsergebnis zu erhalten, so ergibt sich für das Ursignal

$$\boldsymbol{s}(b) = \sum_{i=1}^{N} \boldsymbol{W}^{01}\boldsymbol{X}_i(b)\boldsymbol{W}^{10}\boldsymbol{h}_i(b)\,. \tag{2.32}$$

Werden die Eingangssignale und die Filterimpulantworten aus dem Frequenzbereich erzeugt, kann, unter Verwendung von (2.6) bis (2.8), das Ursignal kompakt als

$$\boldsymbol{s}(b) = \boldsymbol{W}^{01}\boldsymbol{F}^{-1}\sum_{i=1}^{N} \boldsymbol{X}_{\mathrm{F},i}(\omega, b)\boldsymbol{F}\,\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F},i}(\omega, b)\,. \tag{2.33}$$

angeschrieben werden. Die Summe aller Multiplikationen der Eingangssignale mit den jeweiligen Filterkoeffizienten im Frequenzbereich führt nach Rücktransformation und Fensterung zu einem Signalblock des Ursignals.

2.3.2 Gradientenverfahren zur Maximierung der Signalenergie

Ziel dieses Gradientenverfahrens ist es nun, die Energie des Ursignals in einem Block zu maximieren. Wobei gilt, dass die Gewichtsfaktoren in diesem Zeitabschnitt konstant und normiert sind.

$$\sum_{b=c}^{d} \|\boldsymbol{s}(b)\|^2 \quad \rightarrow \quad \max\,, \tag{2.34}$$

$$\text{Nebenbedingung} \quad |\boldsymbol{h}_n(\omega, b)|_p = 1\,. \tag{2.35}$$

Um die Aufgabe wie gewohnt als Minimierungsaufgabe zu stellen, wird die Kostenfunktion mit negativen Quadraten formuliert und für die Nebenbedingung der Lagrangemultiplikator λ angesetzt

$$J = -\left\|\boldsymbol{s}(b)\right\|^2 + \lambda(|\boldsymbol{h}(\omega, b)|_p - 1)\,. \tag{2.36}$$

Zur Aktualisierung der Koeffizienten wird wiederum ein Gradientenabstiegsverfahren angesetzt:

$$\tilde{\boldsymbol{h}}_n(\omega, b) = \tilde{\boldsymbol{h}}_n(\omega, b-1) - \mu \frac{\partial J}{\partial \tilde{\boldsymbol{h}}_n^H(\omega, b)} \qquad \text{mit} \quad n = 1, 2, \ldots, N \tag{2.37}$$

Es ist jedoch, wie oben, leichter mit einer Kostenfunktion zu rechnen, die von normierten Koeffizienten ausgeht:

$$J = -\left\|\boldsymbol{s}(b)\right\|^2 = -\boldsymbol{s}^H(b)\boldsymbol{s}(b) \tag{2.38}$$

Der Gradient dieser Kostenfunktion errechnet sich nun mit:

$$\begin{aligned}
\frac{\partial J}{\partial \tilde{\boldsymbol{w}}_n^H(\omega, b)} &= \frac{\partial}{\partial \tilde{\boldsymbol{h}}_n^H(\omega, b)} \Big[-\boldsymbol{s}^H(b)\boldsymbol{s}(b) \Big] \\
&= -\frac{\partial}{\partial \tilde{\boldsymbol{h}}_n^H(\omega, b)} \left[\left(\boldsymbol{W}^{01}\boldsymbol{F}^{-1} \sum_{i=1}^{N} \boldsymbol{X}_{\mathrm{F},i}(\omega, b)\boldsymbol{F}\,\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F},i}(\omega, b) \right)^H \boldsymbol{s}(b) \right] \\
&= -2\left(\boldsymbol{F}^{-H}\boldsymbol{W}^{10}\boldsymbol{F}^H\boldsymbol{X}_{\mathrm{F},n}^*(\omega, b)\boldsymbol{F}^{-H}\boldsymbol{W}^{01} \right) \boldsymbol{s}(b)
\end{aligned} \tag{2.39}$$

Eingesetzt in (2.37), ergibt sich damit die Aktualisierungsgleichung

$$\tilde{\boldsymbol{h}}_n(\omega, b) = \tilde{\boldsymbol{h}}_n(\omega, b-1) + 2\tilde{\mu}\,\boldsymbol{F}^{-H}\boldsymbol{W}^{10}\boldsymbol{F}^H\boldsymbol{X}_{\mathrm{F},n}^*(\omega, b)\boldsymbol{F}^{-H}\boldsymbol{W}^{01}\boldsymbol{s}(b)\,. \tag{2.40}$$

Die zyklische Kreuzkorrelation des n-ten Kanalsignals mit der zweiten Hälfte des geschätzten Ursignals $\tilde{\boldsymbol{s}}(b)$, also des linearen Faltungsergebnisses, ergibt, nach Schneiden auf die ersten L Abtastwerte und multipliziert mit einer Schrittweite, den Aktualisierungsterm.

Damit die Aktualisierungsgeschwindigkeit weniger von der Energie der Korrelation abhängig ist, wird die Schrittweite zusätzlich auf die 2-Norm der Mikrofonsignale bezogen

$$\tilde{\mu} = \frac{\mu}{\left\|\boldsymbol{x}_{\mathrm{F}}(b)\right\|^2 + reg}\,. \tag{2.41}$$

Um der bei der Kostenfunktion getroffenen Vereinfachung gerecht zu werden, muss nach der Aktualisierung den Koeffizienten die Normierung aufgeprägt werden, mit welcher das Filterverhalten definiert wird. Für eine Allpasscharakteristik erfolgt die Normierung mit

$$h_n(\omega, b) = \frac{\tilde{h}_n(\omega, b)}{|\tilde{h}_n(\omega, b)| + reg}\,. \tag{2.42}$$

Das gewonnene Ursignal besteht damit stets aus allen mit der Stärke 1 überlagerten Kanalsignalen, wobei der Algorithmus für konstruktive Phasenlagen sorgt.

Alternativ ist es möglich, mit der L2-Norm für jede Frequenz individuell zu normieren:

$$\boldsymbol{h}_n(\omega, b) = \frac{\tilde{\boldsymbol{h}}_n(\omega, b)}{\left\|\tilde{\boldsymbol{h}}_n(\omega, b)\right\|_2 + reg} \,. \tag{2.43}$$

Dadurch müssen nicht alle Signale an jeder Frequenz voll beitragen, wodurch tendenziell eher die lautesten Signale herangezogen werden, um das Ursignal zu bilden.

2.4 Implementierung

Die Transformation in den Frequenzbereich und zurück, analytisch mit der Matrix $\boldsymbol{F}$, entspricht bei der Programmierung einer *FFT*, bzw *IFFT*. Durch die damit einhergehende Diagonalisierung, vgl. (2.6), kann die Matrizen-Vektor-Multiplikation im Zeitbereich durch eine punktweise Multiplikation im Frequenzbereich ersetzt werden. In der folgenden Darstellung der Arbeitsschritte in Listenform wird diese punktweise Multiplikation mit dem Hademard-Produkt dargestllt. Beispielsweise wird damit die Matrizen-Vektor-Multiplikation von $\boldsymbol{X}_{\mathrm{F},i}(\omega, b)$ und $\tilde{\boldsymbol{h}}_{\mathrm{F},i}(\omega, b)$ in (2.33),

$$\boldsymbol{s}(b) = \boldsymbol{W}^{01}\boldsymbol{F}^{-1}\sum_{i=1}^{N}\boldsymbol{X}_{\mathrm{F},i}(\omega, b)\boldsymbol{F}\,\boldsymbol{W}^{10}\boldsymbol{F}^{-1}\tilde{\boldsymbol{h}}_{\mathrm{F},i}(\omega, b)\,,$$

zu der punktweisen Multiplikation von $\boldsymbol{X}_i$ und $\boldsymbol{H}_i$

$$\boldsymbol{S} = \sum_{i=1}^{N}\boldsymbol{X}_i \odot \boldsymbol{H}_i\,.$$

Wobei $\boldsymbol{X}_i$ einen Datenblock, und $\boldsymbol{H}_i$ die zugehörige Kanalimpulsantwort in Form eines Vektors im Frequenzbereich darstellen. Auf der folgenden Seite sind die Arbeitsschritte beider Algorithmen für einen Schleifendurchlauf gegenübergestellt.

BCI + Ursignalbildung		***Allpassverfahren***	
Initialisierung			
stepsize $\mu > 0$, Regularisierungsfaktor $\rho > 0$		stepsize $\mu > 0$, Regularisierungsfaktor $\rho > 0$	
$\boldsymbol{h}_i = [1\ 0 \ldots 0]^T \quad i = 1, 2, \ldots, N$		$\boldsymbol{H}_i = [1\ e^{-\frac{-2\pi j}{N}}\ e^{-\frac{-4\pi j}{N}}\ \ldots\ e^{-\frac{-L\pi j}{N}}\ ]^T \quad i = 1, 2, \ldots, N$	
$\boldsymbol{h} = [\boldsymbol{h}_1^T\ \boldsymbol{h}_2^T\ \ldots\ \boldsymbol{h}_N^T]$		$\boldsymbol{H} = [\boldsymbol{H}_1^T\ \boldsymbol{H}_2^T\ \ldots\ \boldsymbol{H}_N^T]$	
$\boldsymbol{h} = \boldsymbol{h}/\sqrt{N}$, Normalisierung			
$\boldsymbol{H} = FFT\{\boldsymbol{h}\}$			
$\boldsymbol{w} = [2\ 2\ \ldots\ 2]^T \quad length(L)$			
$w_1 = 1; \quad w_{L/2+1} = 1; \quad w_{L/2+2:end} = 0$			
Schleifenberechnung: for b = 1,2,...			
(1)	$\boldsymbol{X}_i = FFT\{\boldsymbol{x}_i\,((b-1)*L/2 + (1:L))\}$	*(1)*	$\boldsymbol{X}_i = FFT\{\boldsymbol{x}_i\,((b-1)*L/2 + (1:L))\}$
(2)	$\boldsymbol{E}_{ij} = \boldsymbol{X}_i \odot \boldsymbol{H}_j - \boldsymbol{X}_j \odot \boldsymbol{H}_i \qquad \forall i, j$	*(2)*	$\boldsymbol{S} = \sum_{i=1}^{N} \boldsymbol{X}_i \odot \boldsymbol{H}_i$
(3)	$\boldsymbol{e}_{ij} = IFFT\{\boldsymbol{E}_{ij}\}$	*(3)*	$\boldsymbol{s} = IFFT\{\boldsymbol{S}\}$
(4)	Fensterung von $\boldsymbol{e}_{ij}$ mit $\boldsymbol{W}^{01}$	*(4)*	Fensterung von $\boldsymbol{s}$ mit $\boldsymbol{W}^{01}$
(5)	$\boldsymbol{E}_{ij} = FFT\{\boldsymbol{e}_{ij}\}$	*(5)*	$\boldsymbol{S} = FFT\{\boldsymbol{s}\}$
(6)	$\Delta\boldsymbol{H}_j = 2\mu\frac{1+\rho}{\lVert\boldsymbol{X}\rVert^2+\rho} \sum_{i=1}^{N} \boldsymbol{X}_i^* \odot \boldsymbol{E}_{ij}$	*(6)*	$\Delta\boldsymbol{H}_j = 2\mu\frac{1+\rho}{\lVert\boldsymbol{X}\rVert^2+\rho} \sum_{i=1}^{N} \boldsymbol{X}_i^* \odot \boldsymbol{S}$
(7)	$\Delta\boldsymbol{h}_j = IFFT\{\Delta\boldsymbol{H}_j\}$	*(7)*	$\Delta\boldsymbol{h}_j = IFFT\{\Delta\boldsymbol{H}_j\}$
(8)	Fensterung von $\Delta\boldsymbol{h}_j$ mit $\boldsymbol{W}^{10}$	*(8)*	Fensterung von $\Delta\boldsymbol{h}_j$ mit $\boldsymbol{W}^{10}$
(9)	Update und Normierung: $\boldsymbol{h} = \frac{\boldsymbol{h}-\Delta\boldsymbol{h}}{\lVert\boldsymbol{h}-\Delta\boldsymbol{h}\rVert}$	*(9)*	$\Delta\boldsymbol{H}_j = FFT\{\Delta\boldsymbol{h}_j\}$
(10)	$\boldsymbol{H} = FFT\{\boldsymbol{h}(-t)\}$	*(10)*	Update: $\boldsymbol{H} = \boldsymbol{H} + \Delta\boldsymbol{H}$
(11)	$\tilde{\boldsymbol{S}} = \sum_{i=1}^{N} \boldsymbol{X}_i \odot \boldsymbol{H}_i$	*(11)*	Allpass Normierung: $\boldsymbol{H} = \frac{\boldsymbol{H}}{\lvert\boldsymbol{H}\rvert+\rho}$
(12)	$\boldsymbol{H}_{sum} = \sum_{i=1}^{N} \lvert\boldsymbol{H}_i\rvert^2$		
(13)	$\boldsymbol{H}_{sum} = exp(FFT\{IFFT\{log(\boldsymbol{H}_{sum})\} \odot \boldsymbol{w}\})$		
(14)	$\boldsymbol{S} = \tilde{\boldsymbol{S}} \odot \boldsymbol{H}_{sum}^{-1}$		
(15)	$\boldsymbol{s} = IFFT\{\boldsymbol{S}\}$		
(16)	Fensterung von $\boldsymbol{S}$ mit $\boldsymbol{W}^{01}$		

Tabelle 1: Vergleich der Programmierung beider Algorithmen

2.4.1 Rechenaufwand

Der Rechenaufwand einer Operation in MATLAB lässt sich durch die Anzahl an *floating-point operations (FLOP)* beziffern. Ein Rückschluss auf die Berechnungszeit ist jedoch nicht möglich, da hier die maschinenabhängige Rechenleistung die wesentliche Rolle spielt. Die genaue Anzahl an FLOPs lässt sich mit aktuellen Versionen von MATLAB nicht mehr direkt berechnen. Deshalb erfolgt eine Abschätzung auf Basis des Rechenaufwandes folgender Operationen:

— FFT / IFFT eines Kanals mit FFT-Länge L: $L\log(L)$ FLOPs
— Addition / Subtraktion zweier Vektoren der Länge L: L FLOPs
— Skalare Multiplikation zweier Vektoren der Länge L: L FLOPs

Mit diesen Definitionen ergibt sich für die BCI mit Ursignalbildung pro Schleifendurchgang

$$flop = 5NL\log(L) + 4N^2L + L\left(15N + 3\log(L) + 15\right) \ , \tag{2.44}$$

und für das Allpassverfahren

$$flop = 3NL\log(L) + L\left(11N + 2\log(L) + 1\right) \ . \tag{2.45}$$

Für eine FFT-Länge von $L = 4096$ ergibt sich der in Abb. 4 gezeigte Rechenaufwand für unterschiedliche Kanalanzahl.

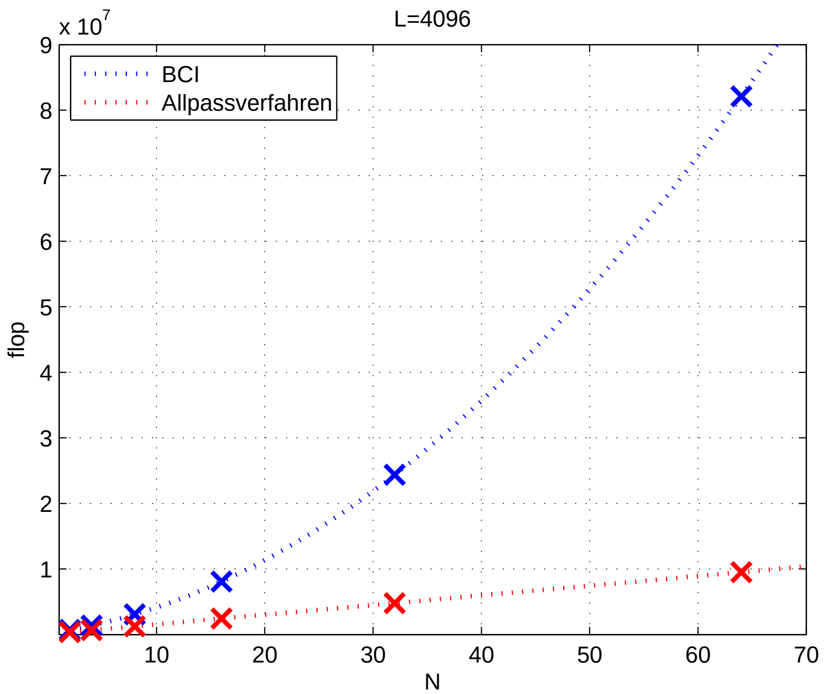

Abbildung 4: Berechnungsaufwand pro Schleifendurchlauf

Diese Betrachtung liefert keinesfalls eine genaue Anzahl an Berechnungsschritten, jedoch eine Abschätzung, mit der sich wesentliche Beobachtungen machen lassen.

Der zweite Term in (2.44) beschreibt den Aufwand für die Berechnung der Fehler $\boldsymbol{e}_{ij}$, und damit den wesentlichen Unterschied zu dem Allpassverfahren. Mit dieser Fehlerberechnung hat die BCI mit Ursignalbildung eine Komplexität von $O(N^2)$, wobei das Allpassverfahren im Vergleich nur einen linearen Zusammenhang $O(N)$ besitzt. Zudem erfolgen bei ersterem 5 Fourier-Transformationen aller Kanäle und beim Allpassverfahren 3 Fourier-Transformationen aller Kanäle in einem Schleifendurchlauf.

Bei Berechnung eines Ursignals aus allen 64 Kanälen der Mikrofonanordnung, ergibt sich für BCI mit Ursignalbildung im Vergleich zu dem Allpassverfahren ein mehr als 8-facher Rechenaufwand pro Datenblock. Für eine steigende Anzahl an Kanälen ist das Allpassverfahren bezüglich Rechenaufwand theoretisch besser zur Ursignalbildung aufgestellt.

3 Simulation

Zur Anwendung der Algorithmen werden diese in MATLAB ausprogrammiert und auf Aufnahmen der kugelförmigen Mikrofonanordnung angewandt. Zuvor soll jedoch die Funktionsweise mit Hilfe von Testfällen überprüft werden.

3.1 Testfall: Weißes Rauschen

In diesem Testfall wird ein bekanntes Ursignal als weißes Rauschen definiert. Daraus werden mit vordefinierten Impulsantworten N Mikrofonsignale generiert, welche bei der anschließenden Simulation als Quellsignale dienen. Die dem Kanal i zugehörige Impulsantwort entspricht einem um i Abtastwerte verzögerten Impuls. Jedes Mikrofonsignal x_i entspricht also einer um i Abtastwerte verzögerte Version des Ursignals.

3.1.1 Kanalidentifikation mit Ursignalbildung

Die Filter werden mit $\boldsymbol{h}_i = [\frac{1}{\sqrt{N}}\ 0\ \ldots\ 0]^T$ initialisiert. Nach einer Adaptionszeit von 5000 Datenblöcken der Länge $L = 64$ ergeben sich die in Abb. 5 und Abb. 6 gezeigten Impulsantworten. Bei weißem Rauschen kommt der Algorithmus mit relativ kurzer FFT-Länge rasch zu einer originalgetreuen Lösung.

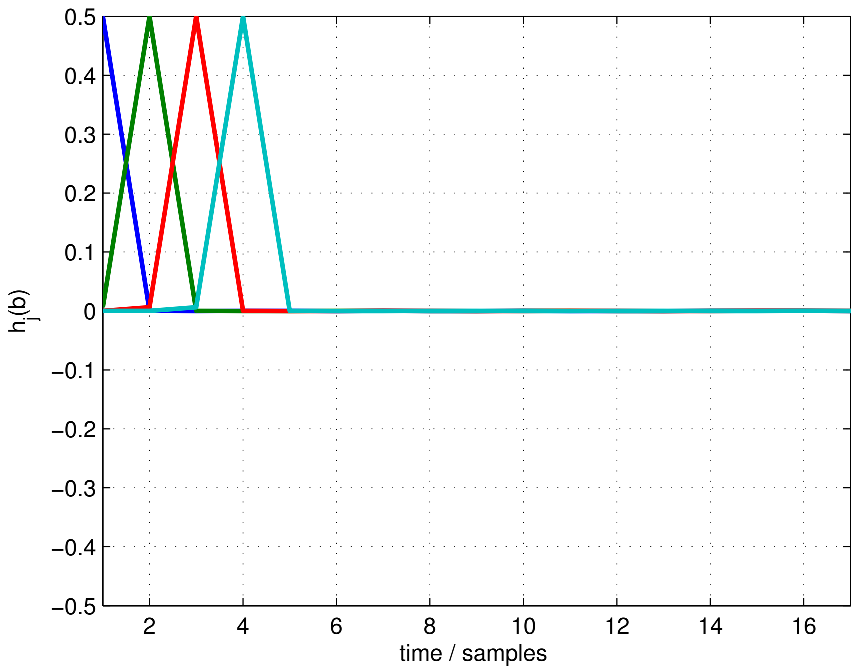

Abbildung 5: Mit BCI errechnete Impulsantworten im Zeitbereich

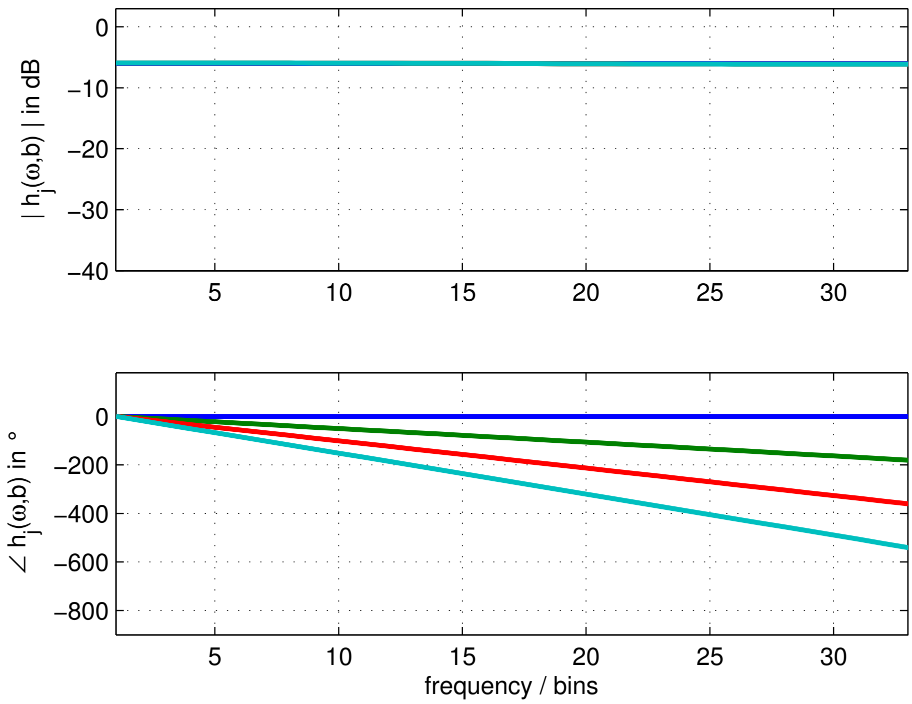

Abbildung 6: Mit BCI errechnete Impulsantworten im Frequenzbereich

3.1.2 Allpassverfahren

Auf den gleichen Testfall wird nun das Allpassverfahren angewandt. Die Filter $\boldsymbol{h}_i$ werden mit linearer Phase und Allpass-Charakteristik, und einem Delay von $L/4+1$ initialisiert. Mit der gleichen Adaptionszeit und FFT-Länge wie vorhin ergeben sich die in Abb. 7 und Abb. 8 gezeigten Impulsantworten. Da bei diesem Verfahren den Koeffizienten ein Allpass-Kriterium aufgeprägt wird, entsprechen die Imupulsantworten nicht dem Original, welches nur eine zeitliche Verzögerung darstellt. Diese Verzögerung wird jedoch, wie in Abb. 7 ersichtlich, erkannt.

Bei diesem Verfahren wird aus den Mikrofonsignalen ein Summensignal gebildet, wobei es die Aufgabe des Algorithmus ist, für konstruktive Phasenlagen zu sorgen. Summieren sich die Signale maximal konstruktiv, so kann bei Verdoppelung der Anzahl an Kanälen, bei Leistungsbetrachtung, ein Gewinn von $+3dB$ erzielt werden. Abb. 9 zeigt den Energiezuwachs des gebildeten Ursignals für eine FFT-Länge von $L = 128$. Dabei zeigt sich, dass mit steigender Anzahl an Kanälen das Maximum nicht mehr erreicht wird. Bei steigender Kanal-Anzahl ist die in Abb. 9 zeigte FFT Länge zu kurz um die Impulsantworten vollständig abzubilden. Abhilfe schafft eine Steigerung der FFT-Länge, wie in Abb. 10 gezeigt, wodurch wieder das gewünschten Ergebnis erreicht wird.

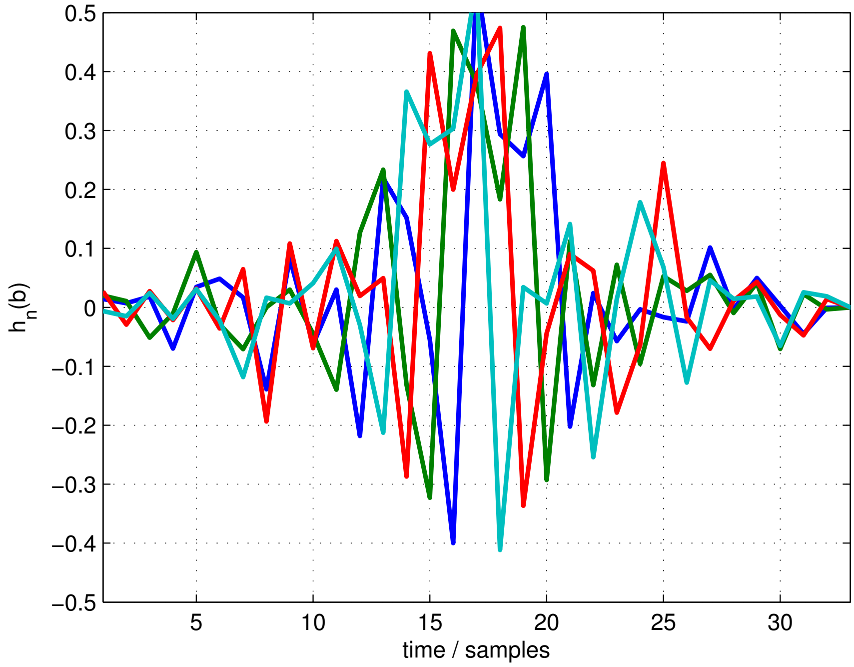

Abbildung 7: Mit dem Allpassverfahren errechnete Impulsantworten im Zeitbereich

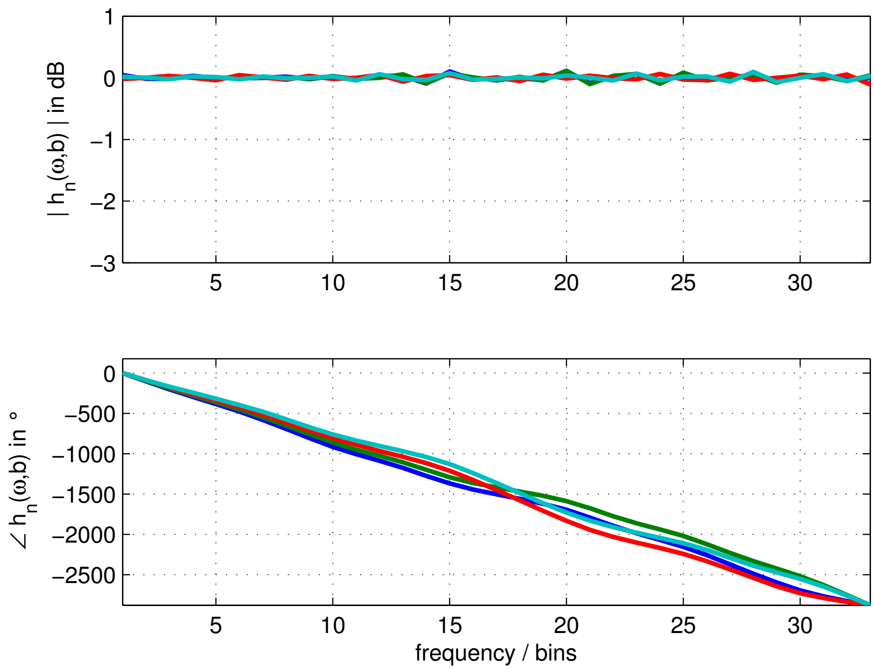

Abbildung 8: Mit dem Allpassverfahren errechnete Impulsantworten im Frequenzbereich

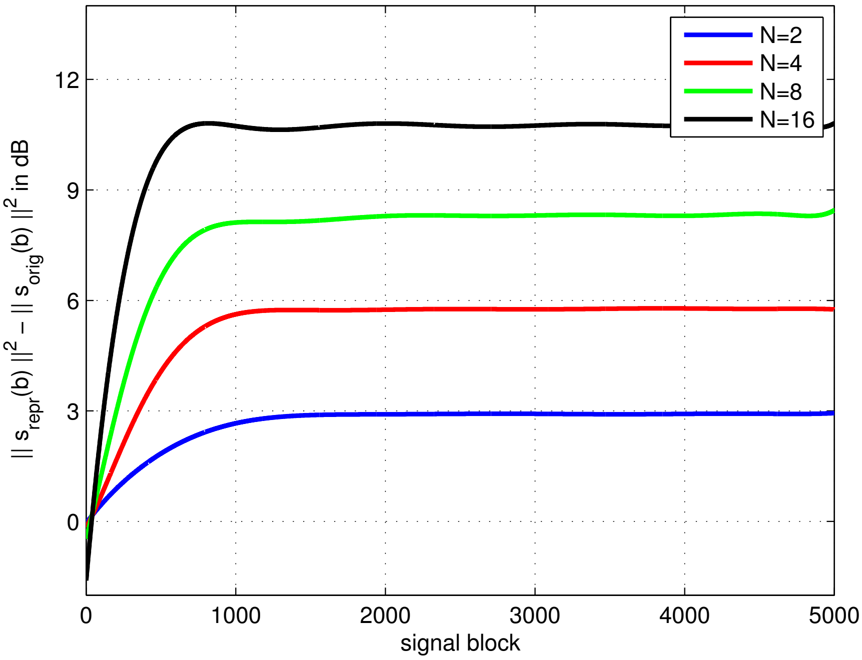

Abbildung 9: Energiegewinn gegenüber einem Kanal

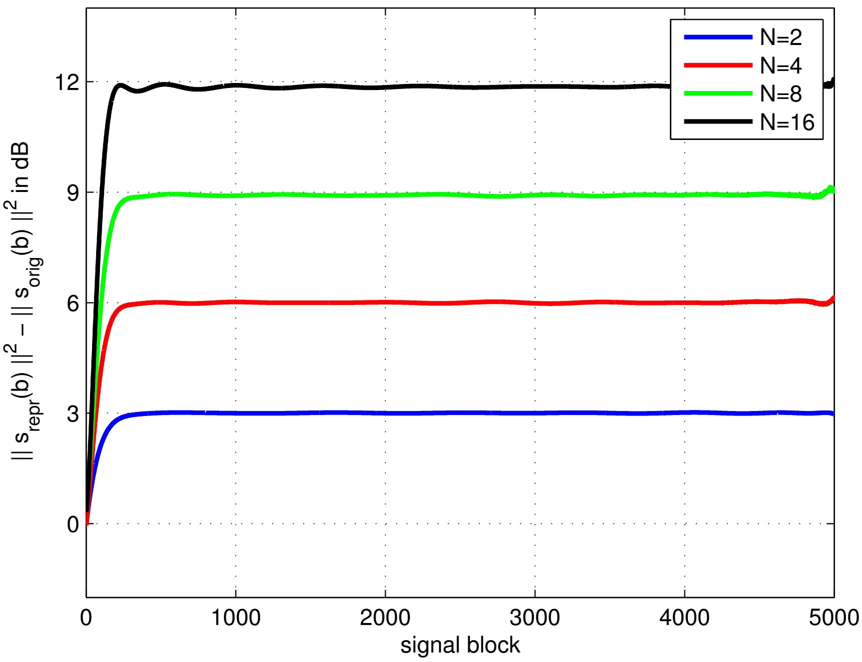

Abbildung 10: Energiegewinn gegenüber einem Kanal

3.2 Testfall: Musikalisches Signal

Dieser Testfall verwendet die selben Impulsantworten wie vorhin, jedoch bildet nicht mehr weißes Rauschen das Ursignal, sondern ein musikalisches Signal. Für diesen Zweck wird ein Kanal einer Aufnahme des Trompeten-Parts von „Gaelforce“ von Peter Graham verwendet.

3.2.1 Kanalidentifikation mit Ursignalbildung

Bei diesem musikalischen Ursignal ist es sinnvoll, die Konvergenz und den Einfluss der einzelnen Parameter zu beobachten. Als Maß zur Ermittlung der Konvergenzzeit wird das *Normalized Projection Misalignment (NPM)* verwendet, welches sich mit

$$NPM(b) = \frac{\|\boldsymbol{\beta}(b)\|}{\|\boldsymbol{h}\|}, \qquad \boldsymbol{\beta}(b) = \boldsymbol{h} - \frac{\boldsymbol{h}^T\hat{\boldsymbol{h}}(b)}{\hat{\boldsymbol{h}}(b)^T\hat{\boldsymbol{h}}(b)}\hat{\boldsymbol{h}}(b) \tag{3.1}$$

berechnet, wobei $\boldsymbol{h}$ die bekannten Impulsantworten und $\hat{\boldsymbol{h}}(b)$ die abgeschätzten Impulsantworten im Zeitblock b darstellen. Mit Hilfe des NPM werden die Einflüsse von FFT-Länge, Schrittweite und Regularisierungsfaktor auf die Konvergenzzeit in Abb. 11-13 dargestellt. Dabei erfolgt die Darstellung in einer Weise, als würde der Algorithmus in Echtzeit angewandt. Die Aufnahme mit der die Testsignale generiert werden, hat eine Dauer von 10 Sekunden. Um dem Algorithmus Daten zur Adaption zur Verfügung zu stellen, werden diese Testsignale in einer Schleife wiederholt durchlaufen.

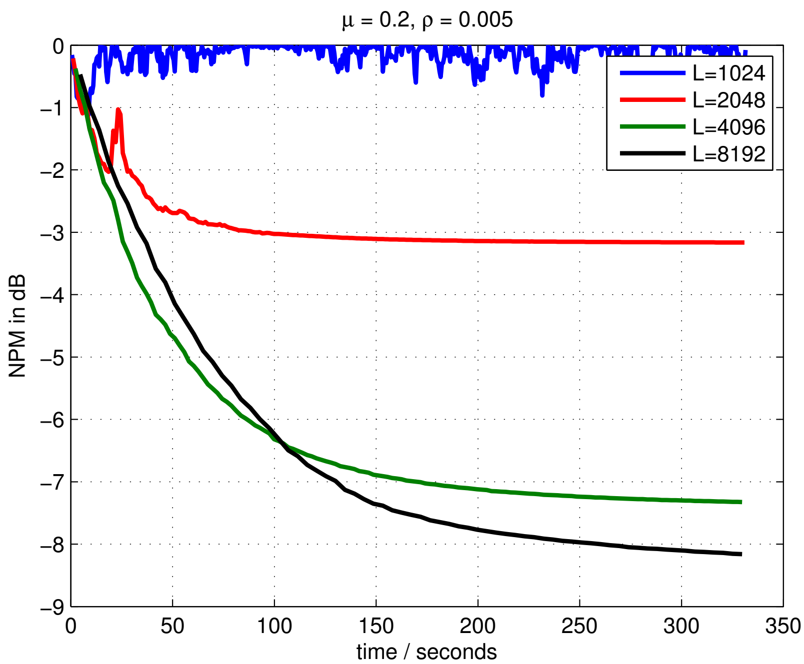

Abbildung 11: *NPM* bei Variation der FFT-Länge

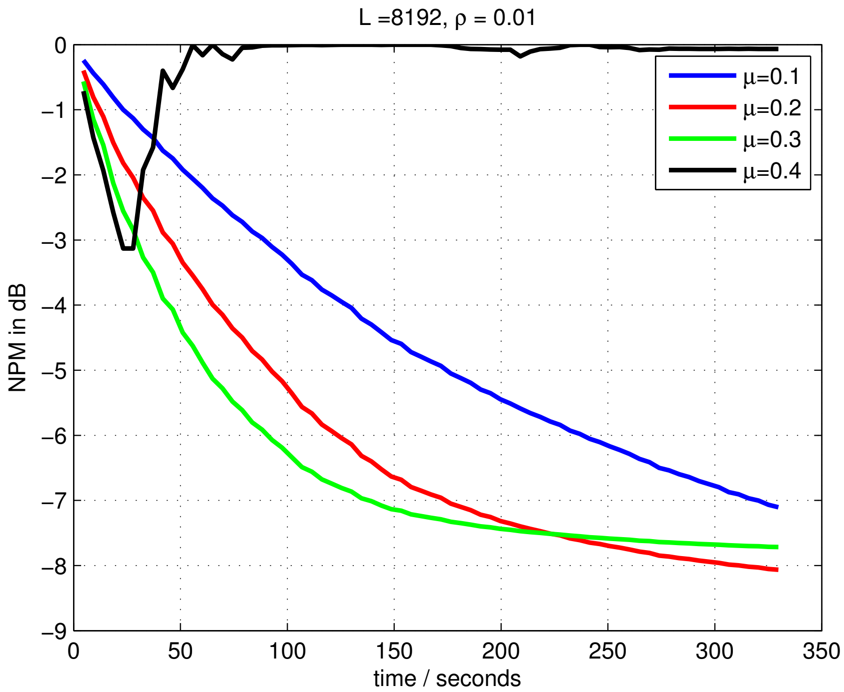

Abbildung 12: *NPM* bei Variation der Schrittweite

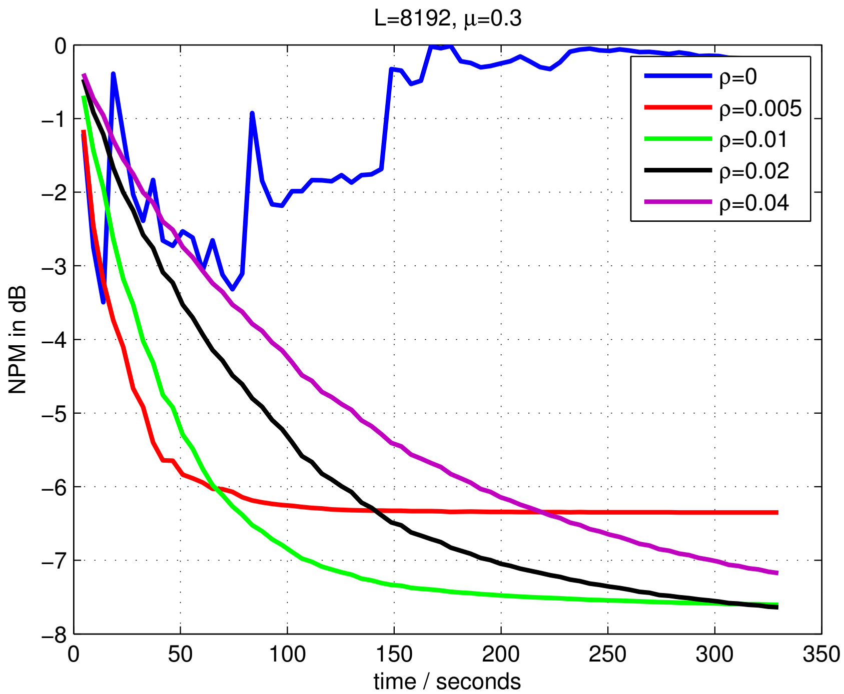

Abbildung 13: *NPM* bei Variation des Regularisierungsfaktors

Bei Variation der Parameter lassen sich folgende Eigenschaften beobachten:

Zu Abb. 11:
- Eine Erhöhung der FFT-Länge hat keinen wesentlichen Einfluss auf die Konvergenzzeit, erhöht aber die Adaptionsgenauigkeit
- Bei geringen FFT-Längen führen höhere Schrittweiten nicht mehr zur Konvergenz.

Zu Abb.12:
- Eine Erhöhung der Schrittweite führt, wie zu erwarten, zu einer kürzeren Konvergenzzeit. Es existiert jedoch ein Maximum in der Konvergenzgeschwindigkeit, das von der FFT-Länge abhängig ist.
- Die erreichbare Adaptionsgenauigkeit verschlechtert sich bei Erhöhung der Schrittweite.

Zu Abb.13:
- Eine Regularisierung ist notwendig damit die Adaption konvergiert.
- Ein kleiner Regularisierungsfaktor führt zwar einerseits zu einer kürzeren Konvergenzzeit, verringert aber die Adaptionsgenauigkeit.
- Mit steigendem Regularisierungsfaktor steigt auch die Konvergenzzeit drastisch an.
- Bei Verwendung der für diesen Testfall ermittelten optimalen Parameter, lässt sich eine Konvergenzzeit von $\sim$60 Sekunden erreichen.

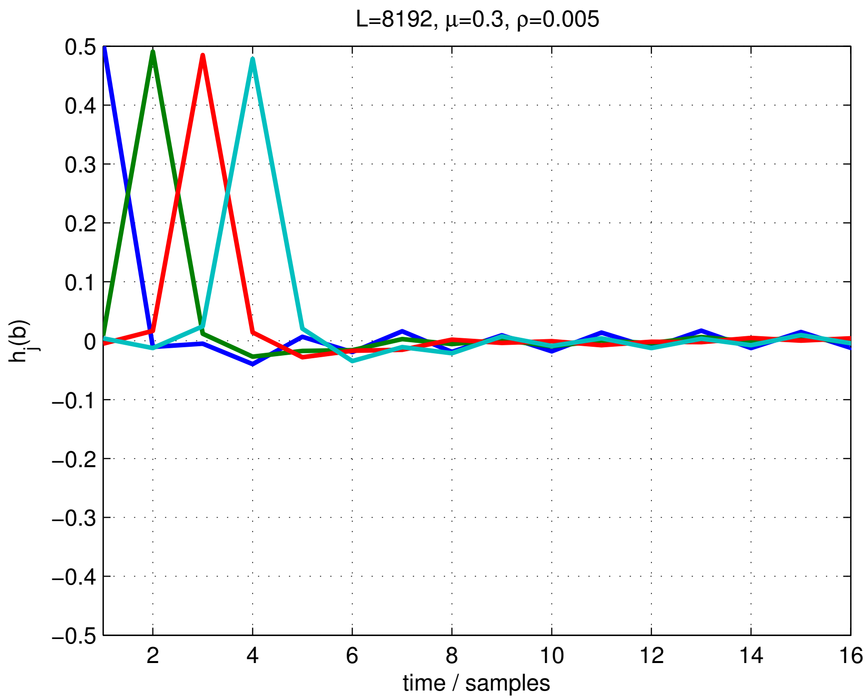

Abbildung 14: Mit BCI errechnete Impulsantworten im Zeitbereich

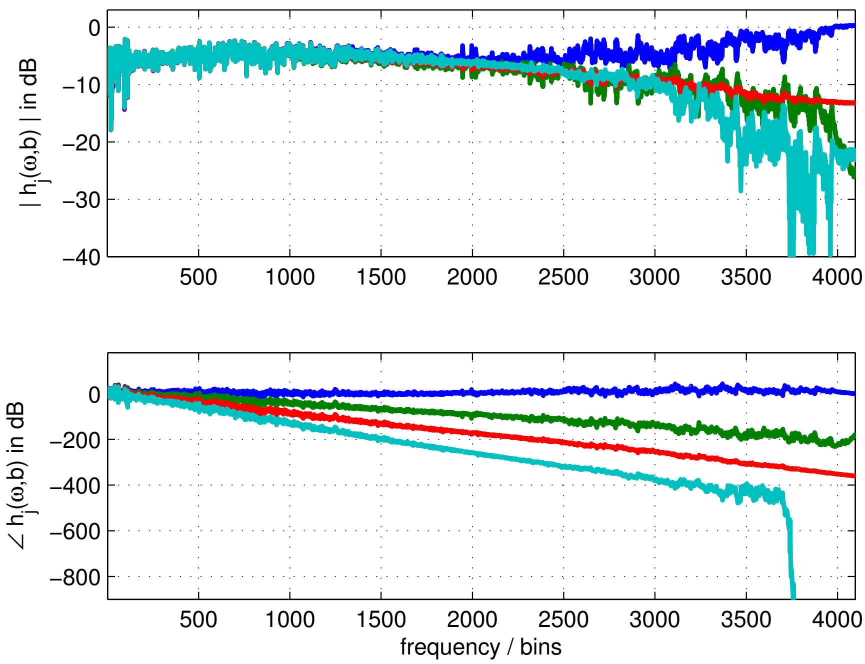

Abbildung 15: Mit BCI errechnete Impulsantworten im Frequenzbereich

Die ermittelten Impulsantworten nach in etwa 20 Durchläufen der Testdaten sind in Abb. 14 und Abb. 15 dargestellt. Im Vergleich mit weißem Rauschen als Ursignal, vgl. Abb. 5 und Abb. 6, und auch bei Beobachtung des Adaptionsvorgangs, scheint die Anregung des musikalischen Signals bei hohen Frequenzen nicht ausreichend zu sein, um eine ideale Abschätzung zu liefern.

3.2.2 Allpassverfahren

Die Variation der Parameter hat auch bei diesem Algorithmus den selben Einfluss wie vorhin beschrieben. Jedoch ist der Algorithmus sensibler auf die Regularisierung. Erst ein Wert von $\rho \simeq 0.05$ führt zu einem stabilen Ergebnis. Ein Vergleich mittels NPM ist hier jedoch nicht möglich, da die originalen Impulsantworten nicht bekannt sind. Auch das Allpassverfahren profitiert von einer höheren Frequenzauflösung, wie schon der Vergleich verschiedener L bei dem Testfall mit weißem Rauschen gezeigt hat. Das Adaptionsergebnis in Abb. 16 wurde mit einer FFT-Länge von $L = 8192$ erzielt und zeigt den selben Ausschnitt von 64 Samples wie Abb. 7 um die beidem vergleichen zu können. Die vollständig berechneten Impulsantworten haben eine Länge von $L/2 = 4096$ Samples. Auch hier kann die Verzögerung detektiert werden, jedoch nicht für jeden Kanal gleichwertig, wie im Vergleich zu Abb. 8 zu sehen ist. Das Betrags- und Phasenspektrum in Abb. 17 zeigt zudem Fehler bei der Allpassnormierung für tiefen Frequenzen.

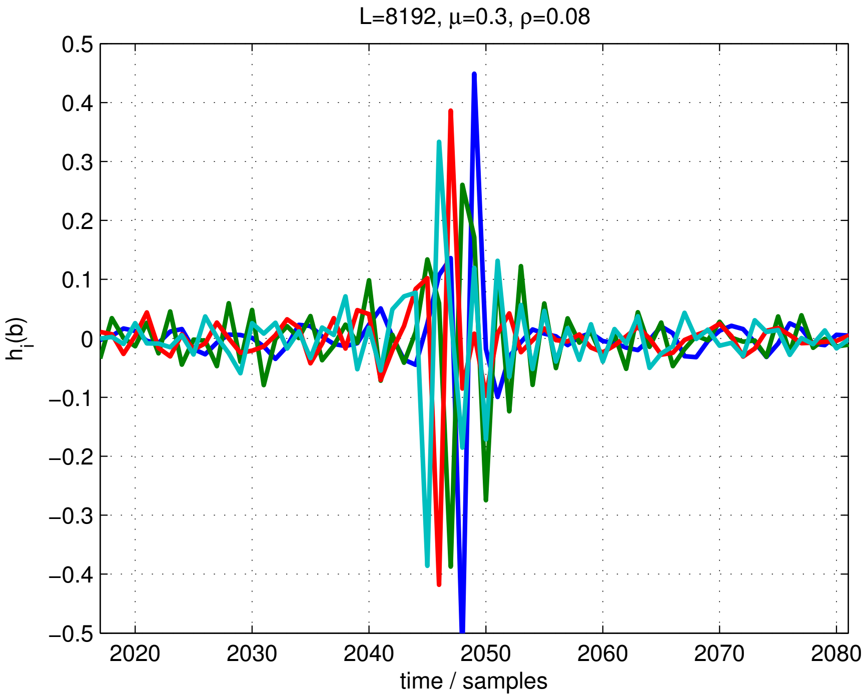

Abbildung 16: Mit dem Allpassverfahren errechnete Impulsantworten im Zeitbereich

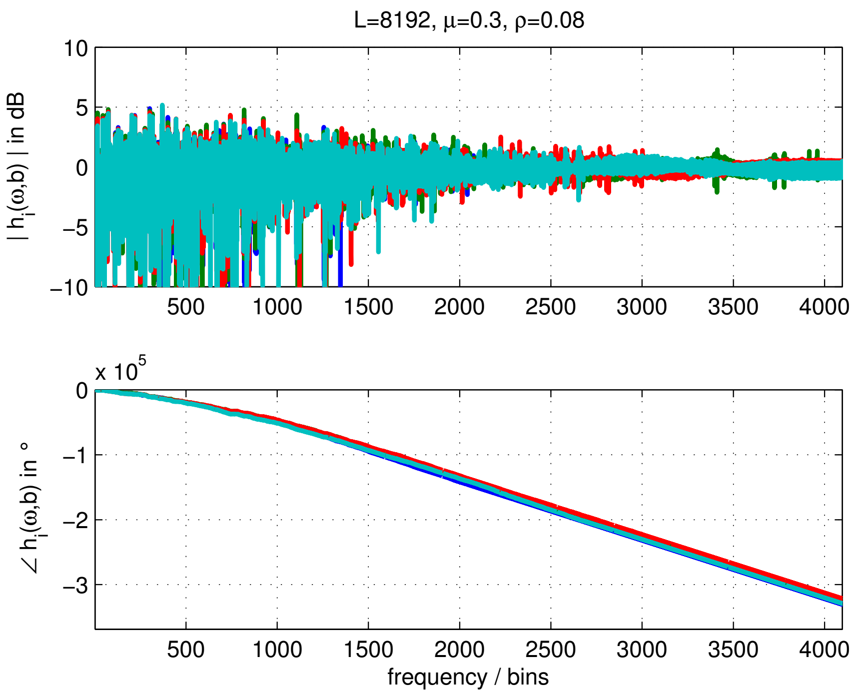

Abbildung 17: Mit dem Allpassverfahren errechnete Impulsantworten im Frequenzbereich

3.3 Anwendung auf Mehrkanalaufnahmen mit unbekannten Kanalimpulsantworten

Es werden nun beide Algorithmen auf die vollständige Aufnahme von „Gaelforce“ angewandt. Alle Impulsantworten und das zugehörige Ursignal sind somit unbekannt. Eine Frequenzauflösung von $L = 8192$ hat sich bei dem vorhergehenden Testfall bewährt und wird deshalb beibehalten. Abb. 18 zeigt die ersten 128 Abtastwerte der resultierenden Impulsantworten für eine Berechnung mit 8 Kanälen, wobei deutlich verschiedene Verzögerungen erkannt werden können.

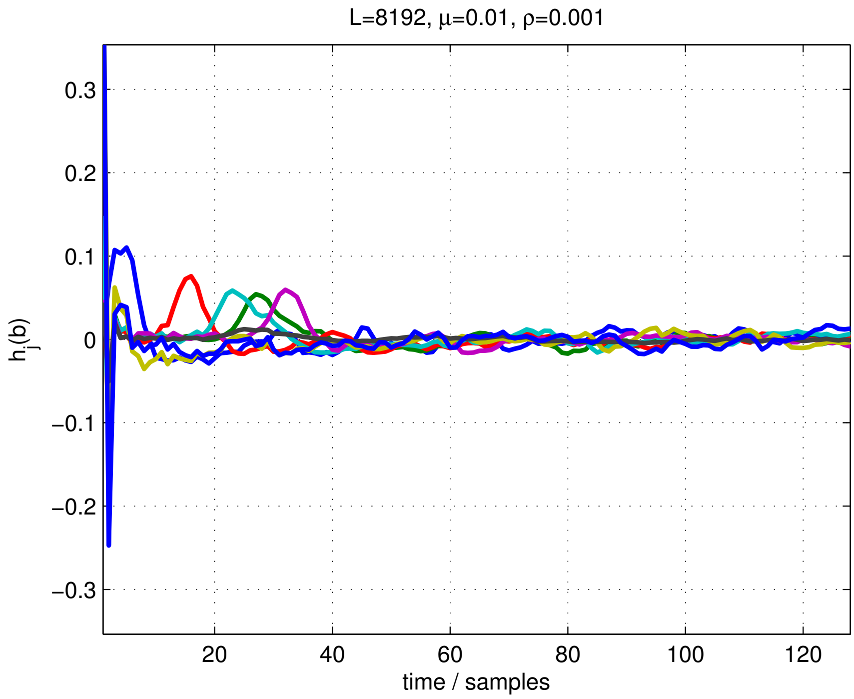

Abbildung 18: Mit BCI errechnete Impulsantworten einer Mehrkanalaufnahme

Die in Abb. 18 sichtbaren Verzögerungen entsprechen bei einer Abtastrate von $Fs = 44100Hz$ einer Distanz von 5 bis $25cm$, was bei einer Trompeten-Aufnahme durchaus realistisch ist.

Die im vorherigen Testfall ermittelten Parameter zur Optimierung der Konvergenzzeit erweisen sich bei der realen Anwendung als nicht praktikabel. Bei zu starker Erhöhung der Schrittweite, werden den Impulsantworten zunehmend Störungen überlagert, denen auch eine Erhöhung des Regularisierungsfaktors nicht entgegengewirkt. Um zu dem erreichbarem Optimum zu gelangen wird eine Eigenschaft ausgenutzt, die sich in Abschnitt 3.2.1 Abb. 12 abzeichnet. Eine deutlich niedrigere Schrittweite führt zwar zu einer erhöhten Konvergenzzeit, die Adaptionsgenauigkeit wird jedoch erhöht.

Betrachtet man die erhaltenen Impulsantworten im Frequenzbereich, Abb. 19, erkennt man im Betragsspektrum eine zunehmende Erhöhung bei hohen Frequenzen.

Das ist vermutlich dadurch zu erklären, dass die Anregungsenergie der Aufnahme bei hohen Frequenzen zu gering ist. Dieser Zustand zeigt sich auch dadurch, dass bei manchen Kanälen ein Großteil der Energie noch im Initialisierungs-Impuls steckt. Eine separate Darstellung der Impulsantworten findet sich im Anhang.

Die durch das Allpassverfahren erhaltenen Impulsantworten weisen in Abb. 20 ebenfalls jene Verzögerungen auf, die auch vom vorhergehenden Algorithmus erkannt wurden. Es ist jedoch ein vergrößerter Ausschnitt gezeigt, da zwar Verzögerungen erkannt werden, die Hauptenergie aber noch bei allen Kanälen im Initialisierungspuls steckt. Eine Optimierung der Parameter führen zu keinem besseren Ergebnis als in Abb. 20 gezeigt. Eine vollständige Darstellung der einzelnen Impulsantworten findet sich im Anhang. Die Allpassnormierung wird kaum verletzt, wie Abb. 21 zeigt. Eine grafische Darstellung des Ursignals ist wenig aussagekräftig, und eine Einschätzung der erreichten klanglichen Qualität wäre zur Beurteilung entscheidend.

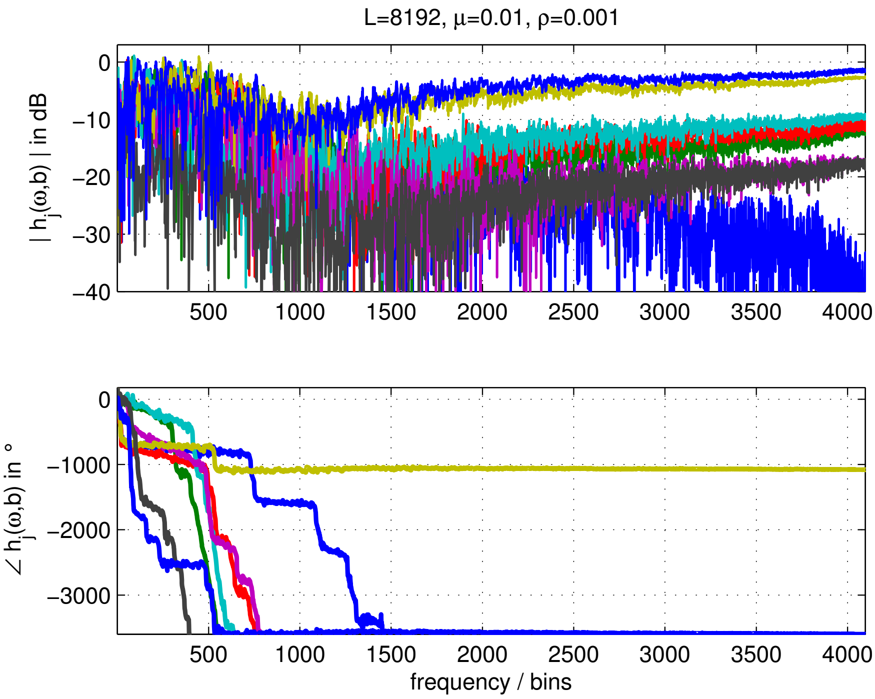

Abbildung 19: Mit BCI errechnete Impulsantworten einer Mehrkanalaufnahme

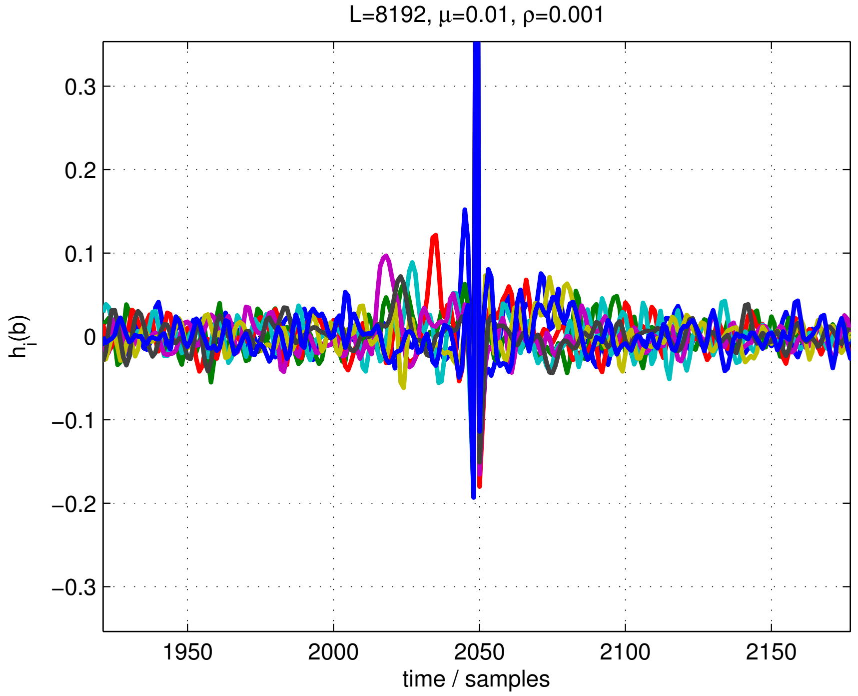

Abbildung 20: Mit dem Allpassverfahren errechnete Impulsantworten einer Mehrkanalaufnahme

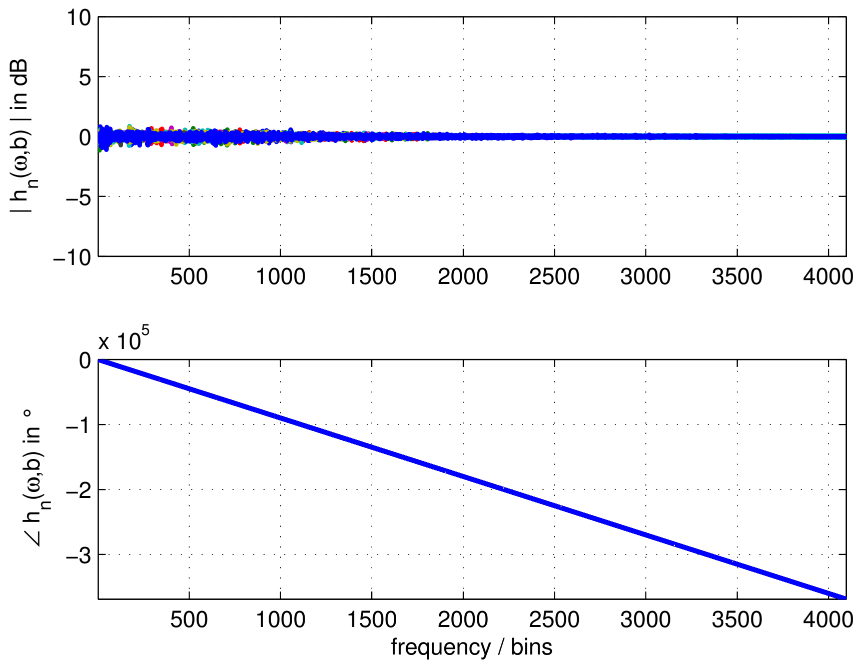

Abbildung 21: Mit dem Allpassverfahren errechnete Impulsantworten einer Mehrkanalaufnahme im Frequenzbereich

4 Diskussion und Schlussfolgerung

Klangeindruck des Ursignals Da eine grafische Darstellung des generierten Ursignals wenig aussagekräftig ist, wird die klangliche Qualität zum wesentlichen Beurteilungskriterium. Auffällige Eindrücke werden daher an dieser Stelle diskutiert.

Blinde Kanalidentifikation mit Ursignalbildung:

Eine Aufsummierung aller Mikrofonsignale ohne Adaption führt zu einem Ursignal, welches, verglichen zu einem Kanal, einen deutlichen Tiefpasscharakter hat. Bei einer Adaption und blockweiser Berechnung des resultierenden Ursignals zeigt sich, dass der Tiefpasscharakter mit zunehmender Adaptionszeit deutlich abnimmt, und das Signal an Präsenz gewinnt. Das Signal beinhaltet außerdem keine hörbaren Artefakte, die Fensterung der Signale und die minimalphasige Entzerrung scheinen also gut zu funktionieren.

Allpassverfahren:

Das mit dem Allpassverfahren resultierende Ursignal zeigt mit zunehmender Adaptionszeit die Eigenschaft von räumlich übertragenen und überlagerten Signalen. Werden alle Mikrofonsignale mit Allpässen gefiltert, die die Phaseninformation nicht beeinflussen, so würde ein Ursignal resultieren, welches keinen räumlichen Eindruck erzeugt, sondern durch Interferenz gefärbte Klangeigenschaften besitzt. Bei der Veränderung der Phaseninformation scheint der Algorithmus für die Signalmaximierung zu viele Freiheiten zu besitzen.

Einen interessanten Versuch stellt die Adaption nicht in Richtung des positiven, sondern des negativen Gradienten dar. Es soll also ein Energieminimum gefunden werden, indem die Aufsummierung der Mikrofonsignale möglichst destruktiv werden soll. Klanglich äußert sich dieser Versuch so, als würde sich das Ursignal vom Hörer räumlich entfernen, also immer diffuser werden. Jedoch entfernen sich nur die tonalen Anteile. Die Anblasgeräusche bleiben weiterhin räumlich beim Hörer, da diese scheinbar zu kurz und in der Amplitude zu gering sind, um eine ausreichend rasche Adaption auszulösen.

Kanal-Reproduktion Bei BCI mit anschließender Ursignalbildung sollte eine Filterung des Ursignals mit der jeweiligen Impulsantwort das zugehörige Mikrofonsignal reproduzieren. Ein Vergleich mit den originalen Aufnahmen zeigt, dass die resultierende Klangfarbe, der durchaus unterschiedlichen Mikrofonsignale, dem Original sehr nahe kommen. Es ist jedoch eine Amplitudenanhebung hochfrequenter Signalanteile hörbar, die vermutlich daraus resultiert, dass die Mikrofonsignale bei hohen Frequenzen zu wenig Anregungsenergie für eine Adaption liefern und somit in den Kanalimpulsantworten der Initialisierungspuls enthalten bleibt. Wird die Reproduktion mit einer konstanten Impulsantwort durchgeführt, ist das Ergebnis zudem statischer Natur. Bewegungen des Musikers bei der Aufnahme und damit verbundene Klangfärbungen gehen dadurch also verloren.

Schlussfolgerung Beide hier vorgestellten Algorithmen erzeugen aus einer beliebigen Anzahl an Einzelsignalen ein Summensignal, wobei den Algorithmen ein Quelle-Filter-Modell zugrunde liegt. Aus der Programmierung und Simulation lassen sich einige Schlussfolgerungen ziehen, welche hier kurz zusammengefasst werden sollen:

— Die Bildung eines Ursignals mittels blinder Kanalidentifikation und anschließender Anwendung des *Multi-Channel Inverse Theorems* ist durchführbar. Die Berechnung auf diesem zweistufigen Weg ist aber sehr rechenintensiv, besonders für hohe Kanalanzahl.

— Die Suche nach alternativen Wegen führt zu einem neuen Algorithmus zur konstruktiven Kanalsummierung, dem Allpassverfahren. Der Rechenaufwand zur Bestimmung eines Ursignals ist, verglichen mit BCI/MINT, deutlich geringer. Das Ergebnis zeigt allerdings klangliche Verfälschungen. Der Ursprung dieses Verhaltens bedarf jedoch noch weiteren Untersuchungen und Verbesserungen.

— Auch wenn das Allpassverfahren einen geringeren Rechenaufwand vorweist, so sind beide Algorithmen nicht für die Anwendung in Echtzeit geeignet, da die Konvergenzeit schon im Testfall sehr hoch ist. Die Anwendung auf eine reale Aufnahme zeigt außerdem, dass höhere Schrittweiten nicht umgesetzt werden können ohne zu starken Fehlern zu führen. Die somit nötige Verwendung von niedrigen Schrittweiten und dadurch weitere Zunahme der Konvergenzzeit, machen eine Anwendung in Echtzeit unmöglich.

— Die Qualität des berechneten Ursignals und der Impulsantworten ist stark vom verwendeten Klangmaterial abhängig. Für eine erfolgreiche Anwendung der Algorithmen liefert die Klangquelle idealerweise ausreichend Anregung in allen Frequenzbereichen. Eine Forderung, die bei musikalischen Signal meist nicht erfüllt werden kann. Die Anwendung auf eine Aufnahme mit breiterem Klangspektrum, wie der hier verwendete Trompeten-Part, führt zu einem deutlich besserem Ergebnis, als die Anwendung auf die Aufnahme eines Saxophon-Parts.

Ausblick Das Allpassverfahren stellt einen interessanten Ansatz zur konstruktiven Signalsummierung dar. In der derzeitigen Implementation weist das Ergebnis jedoch Fehler auf, deren Ursprung noch untersucht werden muss. Mit dem Ziel die Signalenergie zu maximieren, kann die einzige Forderung, dass es sich bei den Filtern um Allpässe handeln muss, den Algorithmus zu wenig in seinen Freiheiten einschränken um zu dem gewünschten Ergebnis zu gelangen. Hier könnte also bei den Nebenbedingungen angesetzt werden um den Algorithmus zu optimieren.

Eine Kombination mit der Spektrogramminversion, die nicht an einen zeitlich teilstationären Verlauf von Kanalantworten gebunden ist, könnte eine interessante Herausforderung sein.

A Anhang

Vollständige Darstellung der Impulsantworten aus Abb. 16:
Allpassverfahren, Mehrkanalaufnahme „Gaelforce“
$L = 8192, \mu = 0.01, \rho = 0.001$

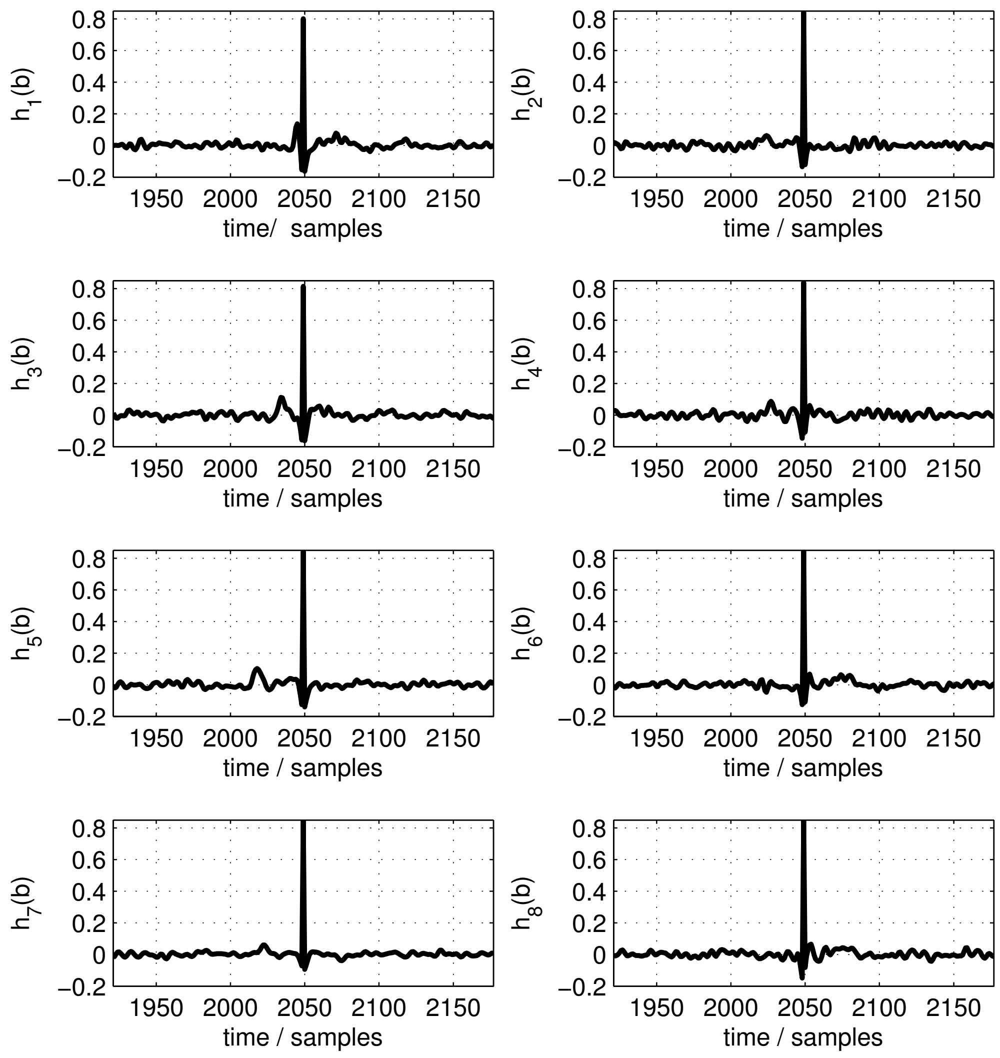

Abbildung 22: Errechnete Impulsantworten mit dem Allpassverfahren

Vollständige Darstellung der Impulsantworten aus Abb. 18:
BCI mit Ursignalbildung, Mehrkanalaufnahme „Gaelforce“
$L = 8192, \mu = 0.01, \rho = 0.001$

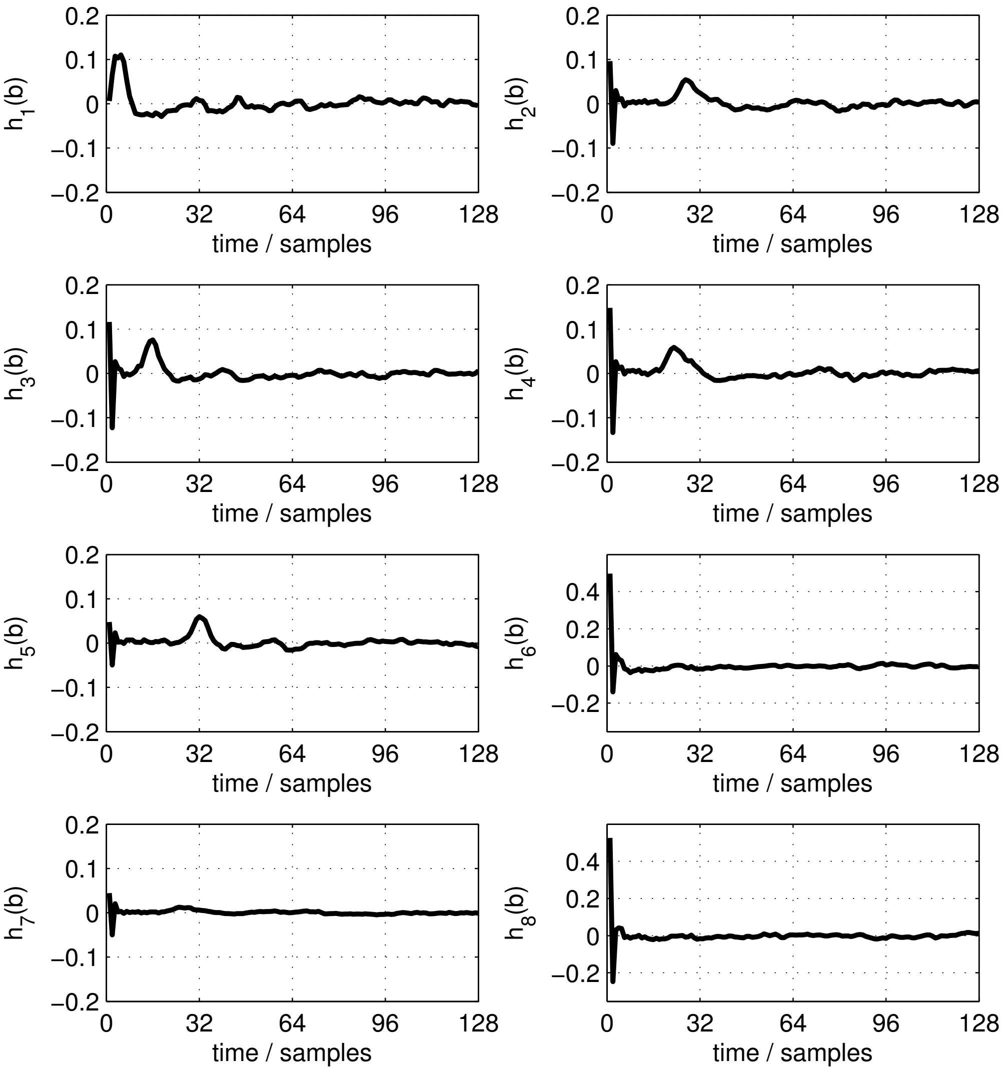

Abbildung 23: Errechnete Impulsantworten mit BCI

Literatur

[GL84] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speechm and Signal Processing*, vol. ASSP 32, 1984.

[HBC06] Y. Huang, J. Benesty, and J. Chen, *Acoustic MIMO Signal Processing.* Springer, 2006, ch. 6-7.

[Hoh09] F. Hohl, "Kugelmikrofonarray zur Abstrahlungsvermessung von Musikinstrumenten," Thesis, Universität für Musik und darstellende Kunst, Technische Universität Graz, 2009.

[Hol14] C. Hollomey, "Real Time Spectrogram Inversion," Thesis, Universität für Musik und darstellende Kunst, Technische Universität Graz, 2014.

[Süs11] S. Süss, "Auffinden von Ursignalen aus Aufnahmen umgebender kugelförmiger Mikrofonanordnung," TI-Projekt, Universität für Musik und darstellende Kunst, Technische Universität Graz, 2011.