

Masterthesis

Entwicklung eines
3D-Audio-Postproduktionstools für
eine dreikanalige koinzidente
Stützmikrofonanordnung

Maurice Strobel (39256)

Hochschule der Medien, Stuttgart

Audiovisuelle Medien (Master of Engineering)

vorgelegt am 02.11.2021

Erstprüfer:

Prof. Dr. Frank Melchior

Zweitprüfer:

Prof. Oliver Curdt

Ehrenwörtliche Erklärung

Hiermit versichere ich, Maurice Strobel, ehrenwörtlich, dass ich die vorliegende Masterarbeit mit dem Titel „Entwicklung eines 3D-Audio-Postproduktionstools für eine dreikanalige koinzidente Stützmikrofonanordnung“ selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Ich habe die Bedeutung der ehrenwörtlichen Versicherung und die prüfungsrechtlichen Folgen (§ 23 Abs. 2 Master-SPO der HdM) einer unrichtigen oder unvollständigen ehrenwörtlichen Versicherung zur Kenntnis genommen.

Stuttgart, 02.11.2021 - Maurice Strobel

Danksagung

Ich möchte an dieser Stelle meinen Dank gegenüber einigen Personen aussprechen, die mir ermöglicht haben, diese Masterarbeit in der vorliegenden Form zu realisieren. An erster Stelle danke ich meinem Erstprüfer Prof. Dr. Frank Melchior, der in den zahlreichen stets konstruktiven und interessanten Gesprächen zur Entwicklung des Themas und zur Umsetzung der Arbeit beigetragen hat. Meinem Zweitprüfer Prof. Oliver Curdt möchte ich einerseits für seine Hilfsbereitschaft im Rahmen dieser Arbeit danken, andererseits insbesondere für die vielen prägenden Eindrücke und Möglichkeiten, die er im Rahmen des gesamten Masterstudiums geboten hat. Ebenfalls möchte ich Leon Hofmann für den freundlichen Kontakt und die Bereitschaft zur Beantwortung verschiedenster Fragen zu der von ihm entwickelten Mikrofonanordnung danken sowie für die Bereitstellung des Testmaterials. Auch meinem privaten Umfeld möchte ich danken, das mich stets verständnisvoll unterstützt und motiviert hat.

Zusammenfassung

Die vorliegende Arbeit beschäftigt sich mit der Entwicklung eines geeigneten Decodierungs- und Panningtools für die Bearbeitung von Aufnahmen eines dreikanalig-koinzidenten Stützmikrofons in mehrdimensionalen Mischungen. Decodierungsseitig wird eine stereofone MS-Decodierung in zwei Stereosignalpaare aufgezeigt und diskutiert. Es wird ein Amplitudenpanningkonzept entwickelt, das die vier decodierten Einzelsignale um eine imaginäre Schallquelle herum anordnet und mithilfe von zusätzlichen horizontalen und vertikalen Ausdehnungsparametern die Position und Ausdehnung einfach manipulierbar macht. Aus diesen Erkenntnissen wird zunächst ein allgemeines Signalverarbeitungs- und Bedienkonzept entwickelt, das dann prototypisch in ein Softwaretool implementiert wird. Dieser Prototyp wird dokumentiert und diskutiert mit einem Ausblick auf weitere Forschungsmöglichkeiten.

Abstract

This thesis presents and discusses a suitable decoding and panning method for the post-production of recordings of a coincident 3-channel-spot-microphone-arrangement in immersive audio productions. A stereophonic ms-based decoding method is proposed and discussed. For the decoded signals an amplitude panning method is developed that enables positioning of the expanded sound source consisting of the four decoded signals by distributing them around an imaginary sound source using expansion parameters, which enable simple manipulation of position and expansion. Based on these findings a general signal processing and interface concept is developed. This concept is then implemented into a software prototype which is documented and discussed.

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis	III
Tabellenverzeichnis	V
Abkürzungsverzeichnis	VI
Glossar	VII
Symbolverzeichnis	IX
1 Einleitung	1
2 Grundlagen	4
2.1 Lokalisation & räumliches Hören	4
2.2 Ohrsignalmerkmale	5
2.3 Summenlokalisation & Phantomschallquellen	5
2.4 Klangfärbung	6
2.5 Ausdehnung	7
3 Analysen	8
3.1 Die MHV-Mikrofonie	8
3.1.1 Verwandte Stereophonieverfahren	9
3.1.2 MS mit variabler Richtcharakteristik	12
3.1.3 MHV mit variabler Breite und Höhe	15
3.1.4 Das native B-Format-Mikrofon	15

3.2	Panningverfahren	16
3.2.1	VBAP	17
3.2.2	VBIP	20
3.2.3	MDAP	20
4	Konzeption	22
4.1	Signalverarbeitungskonzept	22
4.2	Parametrisierungskonzept	26
4.2.1	Decodierungsparameter	26
4.2.2	Panningparameter	26
4.3	Benutzeroberflächenkonzept	30
5	Implementierung	32
5.1	Tools	33
5.2	Signalverarbeitungskette	34
5.3	Parametrisierung	35
5.4	Benutzeroberfläche	35
5.5	Entwicklungskonzepte	40
5.5.1	Modularisierung	40
5.5.2	Dokumentation	44
5.5.3	Skriptsteuerung	45
5.5.4	Kommunikation über OSC	46
5.5.5	Namenskonventionen	48
5.5.6	Farbcodierung	48
5.6	Subjektive Bewertung	49
6	Fazit	51
6.1	Zusammenfassung	51
6.2	Ausblick	52
	Literatur	55

Abbildungsverzeichnis

2.1	Kopfbezogenes Koordinatensystem	5
3.1	Richtcharakteristika der MHV-Mikrofonie	9
3.2	Blumlein-Stereofonie	10
3.3	MS-Mikrofonie	11
3.4	Niere aus Kugel und Acht	14
3.5	MS mit variablem S-Anteil	14
3.6	Die äquivalenten Richtcharakteristika der LRBT-Signale	16
3.7	VBAP 3D	18
3.8	VBAP Tripel	19
3.9	MDAP 3D	21
4.1	Blockschaltbild Audio	22
4.2	Blockschaltbild: Decodierung	23
4.3	Blockschaltbild: Panning	24
4.4	ITU Konfiguration 4+7+0	25
4.5	ITU Layout 4+7+0	25
4.6	Panningkonzept: T-Orientiation (Raute)	28
4.7	Panningkonzept: X-Orientiation (Rechteck)	28
4.8	Panningkonzept: Discrete (Diskret)	30
5.1	Blockschaltbild: Gesamte Implementierung	36
5.2	4+7+0-Lautsprecheresetup	37
5.3	Benutzeroberfläche: Grafische Aufteilung	38
5.4	Benutzeroberfläche: Visualisierung der Richtcharakteristika	40
5.5	Benutzeroberfläche: Pan-Parameter-Modi	41

Abbildungsverzeichnis

5.6	Ausgangszuweisungsbereich der Benutzeroberfläche	42
5.7	Benutzeroberfläche: Speaker-Gain-Fenster	42
5.8	Max: Projektordner	43
5.9	Max: Selbstdokumentation	45
5.10	Max: OSC	46
5.11	Max: Patch-Identifizier	48
5.12	Max: Farbcodierung	49

Tabellenverzeichnis

4.1	Parameter der Benutzeroberfläche	31
5.1	Parameter der implementierten Benutzeroberfläche	39

Abkürzungsverzeichnis

- DAW** Digital Audio Workstation [34](#), [53](#)
HRFT Head-Related Transfer Function [5](#)
ICC [Inter-Channel Cross-Correlation Coefficient](#) [21](#)
ILD Interaural Level Difference [5](#)
ITD Interaural Time Difference [5](#)
M4L [Max For Live](#) [VII](#), [37](#), [39](#)
MDAP [Multiple-Direction Amplitude Panning](#) [3](#), [20](#), [21](#), [34](#), [49](#)
MHV [Mitte-Horizontal-Seite](#) [VII](#), [1–4](#), [16](#), [17](#), [26](#), [27](#), [32](#), [49](#), [53](#)
MS [Mitte-Seite](#) (auch [Summe-Differenz](#)) [2](#)
VBAP [Vector-Base Amplitude Panning](#) [VII](#), [3](#), [17](#), [19–22](#), [29](#), [30](#), [34](#)
VBIP [Vector-Base Intensity Panning](#) [3](#), [20](#)

Glossar

4+7+0 beschreibt eine mehrdimensionale Lautsprecheranordnung mit sieben Lautsprechern im unteren ([Lower Layer](#)) und vier Lautsprechern im oberen Layer ([Upper Layer](#)). [19](#), [23](#), [27](#), [28](#), [34](#), [53](#)

Amplitudenpanning ist ein rein pegelbasiertes Panningverfahren zur Positionierung von Schallquellen in einem zwei- oder dreidimensionalen Lautsprecher-setup. [VII](#), [VIII](#), [17](#), [18](#)

Downmix bezeichnet den Prozess einer Reduktion der zur Verfügung stehenden Kanäle auf eine geringere Anzahl, meist durch Summierung. [17](#)

Hörer* soll die in einem Versuchsaufbau hörende Person auf genderneutrale Weise bezeichnen. [VII](#), [4](#), [5](#), [17–19](#), [23](#)

Inter-Channel Cross-Correlation Coefficient bezeichnet die Kreuzkorrelation zwischen zwei Kanälen. [VI](#), [21](#)

Lower Layer (oder auch Bottom Layer) bezeichnet die untere horizontale Ebene in einer dreidimensionalen Lautsprecheranordnung. [VII](#), [23](#)

Max For Live ([M4L](#)) ist die für Ableton Live spezifische Implementierung von Max. [VI](#), [34](#)

Middle Layer bezeichnet die mittlere, auf Höhe des [Hörer*s](#) gelegene, horizontale Ebene in einer dreidimensionalen Lautsprecheranordnung. [20](#), [23](#), [26](#), [27](#)

Mitte-Horizontal-Seite ([MHV](#)) ist ein von Leon Hofmann entwickeltes koinzidentes Stützmikrofon-Array. [VI](#), [1](#)

Multiple-Direction Amplitude Panning ([VBAP](#)) ist ein von Ville Pulkki ([1999](#)) entwickeltes [Amplitudenpanningverfahren](#) als Modifikation von [VBAP](#). [VI](#), [20](#), [21](#)

Nutzer* soll den Nutzer (auch: User) einer Anwendung auf genderneutrale Weise bezeichnen. [26](#), [30](#), [34](#), [37](#), [39](#)

- Patch** bezeichnet die logische Abstraktion eines Bausteins in der Entwicklungsumgebung Max und stellt gleichzeitig deren Dateiformat dar. [33](#), [34](#)
- Upper Layer** (oder auch Top Layer) bezeichnet die obere horizontale Ebene in einer dreidimensionalen Lautsprecheranordnung. [VII](#), [20](#), [23](#), [37](#)
- Vector-Base Amplitude Panning** ist ein von Ville Pulkki ([1997](#)) entwickeltes [Amplitudenpanningverfahren](#). [VI](#), [VIII](#), [17](#), [18](#), [20](#)
- Vector-Base Intensity Panning** ist eine Abwandlung des [Vector-Base Amplitude Panning](#). [VI](#), [20](#)

Symbolverzeichnis

φ	Azimutwinkel im kopfbezogenen Koordinatensystem
δ	Elevationswinkel im kopfbezogenen Koordinatensystem
Φ	Mikrofonversatzwinkel in Relation zur Nullachse
Φ	horizontaler Ausdehnungswinkel im Panning der Schallquellen LRBT
Δ	vertikaler Ausdehnungswinkel im Panning der Schallquellen LRBT
o	vertikaler Versatzwinkel im Panning der Schallquellen LRBT
a	Skalierungsfaktor für den M-Anteil bei der MS-Decodierung
b	Skalierungsfaktor für den S-Anteil bei der MS-Decodierung
K	Skalierungsfaktor für den Druckanteil einer Niere
A	Skalierungsfaktor für den Druckgradientenanteil einer Niere
$\mathbf{p}$	dreidimensionaler Positionsvektor einer Schallquelle
$\mathbf{l}$	dreidimensionaler Positionsvektor eines Lautsprechers
$\mathbf{g}$	Gainfaktorenvektor in der Amplitudenpanningmatrizierung
$\mathbf{L}$	Matrix aus Positionsvektoren der Lautsprecher bei der Amplitudenpanningmatrizierung
M	Signal des M-Mikrofons der MS- oder MHV-Anordnung
S	Signal des S-Mikrofons der MS-Anordnung
H	Signal des H-Mikrofons der MHV-Anordnung
V	Signal des V-Mikrofons der MHV-Anordnung
$L, R,$	Signale der virtuellen Schallquellen LRBT
B, T	
$x[n]$	Eingangssignal in einer digitalen Signalverarbeitungskette
$y[n]$	Ausgangssignal in einer digitalen Signalverarbeitungskette

1 Einleitung

Die vorliegende Arbeit entwickelt ein Konzept für Decodierung und Panning der Signale einer dreikanalig-koinzidenten Stützmikrofonanordnung im Rahmen der Forschung zur Abbildung ausgedehnter akustischer Schallquellen in mehrdimensionalen Mischungen und implementiert dieses Konzept prototypisch in Form eines 3D-Audio-Postproduktionstools.

Diese Arbeit baut unter anderem auf der Forschung von Leon Hofmann (2020) auf, der im Rahmen einer Masterarbeit unter praktischen und audioästhetischen Aspekten ein koinzidentes Stützmikrofonieverfahren zur ausgedehnten Abbildung akustischer Instrumente namens **MHV** (Mitte-Horizontal-Seite) etabliert und prototypisch erprobt hat. Dieses Verfahren soll im Speziellen die Ausdehnung akustischer Schallquellen innerhalb einer Mehrkanalmischung variabel und formatkompatibel abbilden.

Diverse und ausgiebige Forschungen wurden bereits in Bezug auf koinzidente Hauptmikrofonen für mehrdimensionale Mischungen durchgeführt. Beispiele hierfür sind das sogenannte Doppel-MS (vgl. Wittek, Haut und Keinath 2006) oder auch das artverwandte native B-Format-Ambisonic-Mikrofon (vgl. Lee 2021). Die koinzidenten Verfahren setzen oft auf ein diskretes kanalbasiertes Routing der Signale auf die Lautsprecher eines 3D-Audio-Lautsprecher setups. Dadurch sind sie einerseits in der Postproduktion recht unflexibel, da eine Justierung der Schallquellenpositionen bzw. der Schallfeldorientierung nicht problemlos möglich ist, andererseits sind sie meist nur für bestimmte Lautsprecher setups geeignet und nicht universell kompatibel (vgl. Lee 2021). Ebenso versucht ein Hauptmikrofon möglichst natürlich das gesamte Schallfeld einzufangen – also im Falle einer musikalischen Ensembleaufnahme das gesamte instrumentale Ensemble inklusive des räumlichen Eindrucks (vgl. Weinzierl 2008, S. 600–603). Währenddessen

sollte ein Stützmikrofon möglichst isoliert nur das in einer Mischung zu stützende Klangelement – also beispielsweise ein einzelnes Instrument in einem größeren Ensemble – ohne räumliche Einflüsse aufzeichnen (vgl. Weinzierl 2008, S. 600–603).

Das Feld der 3D-Audio-Stützmikrofonie wird derzeit von größeren Herstellern mit Konzepten wie dem Schoeps 2plus2-Stützmikrofon erschlossen, das eine reduzierte Abwandlung des Schoeps ORTF-3D-Hauptmikrofons darstellt und auf ein diskretes Routing der decodierten Kanäle setzt (vgl. Hautz 2021). Hier ist ebenfalls ein Rückgriff auf vertraute Techniken aus Stereophonie und 3D-Hauptmikrofonie zu erkennen. Gleichmaßen zeigt das Interesse der Hersteller die aktuelle Relevanz des Themas und eine Notwendigkeit der Forschung in diesem Bereich auf. Das von Hofmann (2020) entwickelte MHV-Stützmikrofon bedient sich ebenfalls an Prinzipien der 3D-Hauptmikrofonie sowie stereofonen Konzepten wie MS und wendet diese auf ein Stützmikrofonkonzept an.

Ebenso wie für Hauptmikrofonien ist auch für Stützmikrofonanordnungen die Entwicklung geeigneter Postproduktionstools für Decodierung sowie auch Panning erforderlich. Für Hauptmikrofonien gibt es bereits einige – meist proprietäre – Tools, wie beispielsweise das Schoeps Double MS Plugin für die Schoeps Doppel-MS-Mikrofonie (vgl. Schoeps 2021), das die Decodierungsvorgänge visualisiert und sinnvoll parametrisiert. Ebenso versucht der Hersteller RØDE, das ehemalige Soundfield-Ambisonic-Mikrofon mit dem Soundfield by RØDE Plugin für andere Zwecke nutzbar zu machen (vgl. Hau 2019). Dies zeigt die Relevanz von geeigneten Softwaretools auf, die durch neue Signalverarbeitungskonzepte auch für etablierte Hardwarelösungen – wie zum Beispiel der Soundfield-Hauptmikrofonie – neue Anwendungen schaffen kann.

In Kapitel 2 der vorliegenden Arbeit werden relevante psychoakustische Grundlagen für die spätere Konzeption betrachtet. Dabei werden Lokalisation und räumliches Hören auf psychoakustischer Seite aufgegriffen und in einen tontechnischen Kontext gesetzt. Im Besonderen werden die Aspekte der Ausdehnung sowie die Herausforderung der Lokalisation und infolgedessen auch die ausgedehnte Abbildung beschrieben.

Kapitel 3 stellt relevante bestehende Techniken vor und analysiert diese. Die von Hofmann (2020) entwickelte MHV-Stützmikrofonanordnung wird vorgestellt und auf ihre Verwandtschaft zur MS- und XY-Mikrofonie hin untersucht. Zudem werden die Effekte der Decodierung des Verfahrens mit variablen Signalanteilen analysiert. Ebenfalls wird auf verschiedene Formen des Amplitudenpannings eingegangen. Im Speziellen werden VBAP, VBIP und MDAP nach Ville Pulkki vorgestellt.

Aufbauend auf den erarbeiteten Grundlagen und Analysen werden in Kapitel 4 geeignete Konzepte für die Decodierung sowie das Panning erarbeitet und vorgestellt. Dabei wird eine geeignete Signalverarbeitungskette, Parametrisierung sowie Benutzeroberfläche konzipiert.

Kapitel 5 stellt eine prototypische Implementierung der in Kapitel 4 vorgestellten Konzepte vor, dokumentiert diese und stellt sie in den Kontext der Erkenntnisse aus den vorangegangenen Kapiteln.

Abschließend wird der Entwicklungsprozess resümiert sowie die Erkenntnisse kritisch betrachtet. Es erfolgt ein Ausblick auf die Anwendbarkeit der Erkenntnisse und des Tools auf ähnliche mehrdimensionale Aufnahmen und Wiedergabeszenarien.

2 Grundlagen

Im Folgenden soll auf einige für diese Arbeit relevante psychoakustische Grundlagen eingegangen werden, um ein Grundverständnis für Lokalisation (2.1), Phantomschallquellen (2.3) und ausgedehnte Schallquellen (2.5) zu schaffen und die verwendeten Begrifflichkeiten zu erläutern. Für einen tiefergehenden psychoakustischen Einblick sei auf das Grundlagenwerk Blauerts (2008) sowie auf Hofmann (2020) verwiesen, der einige psychoakustischen Grundlagen bereits im Rahmen der Entwicklung der MHV-Anordnung aufgearbeitet hat.

2.1 Lokalisation & räumliches Hören

Zur Beschreibung von räumlichen Schallereignissen bzw. virtuellen Schallquellen und deren Positionen ist das Etablieren eines Koordinatensystems sinnvoll. Blauert (2008) etabliert ein kopfbezogenes Koordinatensystem, das den Hörer* in den Mittelpunkt einer Sphäre setzt und die Positionen von Schallereignissen mithilfe vom horizontalen Azimutwinkel φ , dem vertikalen Elevationswinkel δ und der Distanz r relativ zum Mittelpunkt bzw. zum Hörer* beschreibt (Abbildung 2.1). So ist es möglich ein beliebiges Schallereignis in seiner Position zu beschreiben.

Zusätzlich etabliert dieses kopfbezogene Koordinatensystem (Abbildung 2.1) dessen drei räumliche Ebenen und benennt sie mit Horizontal-, Median- und Frontalebene (vgl. Blauert und Braasch 2008, S. 88).

Diese Bezeichnungen der Winkel und Ebenen sollen in den folgenden Analysen und Beschreibungen verwendet werden.

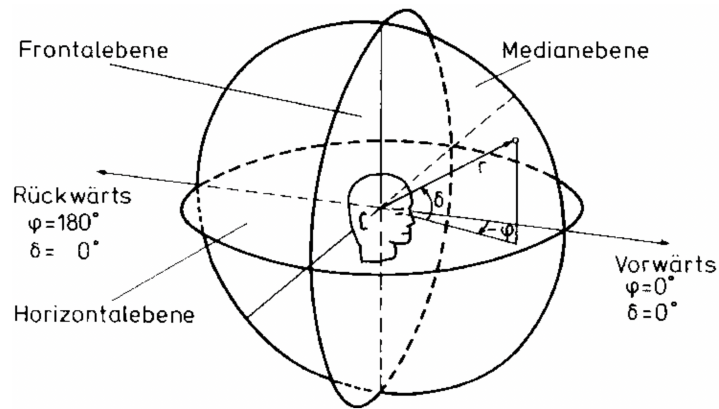


Abbildung 2.1: Das kopfbezogene Koordinatensystem (Blauert und Braasch 2008, S. 88)

2.2 Ohrsignalmerkmale

Die Ortung einer virtuellen Schallquelle hängt in großem Maße von der **ILD** (Interaural Level Difference) – also dem wahrgenommenen Lautstärkeunterschied des eintreffenden Schalls zwischen beiden Ohren – und der **ITD** (Interaural Time Difference) – also der zeitlichen Differenz des eintreffenden Ereignisses zwischen beiden Ohren – ab (vgl. Pulkki 1997). Zusätzlich zu diesen interauralen Ohrsignalmerkmalen kommen monaurale Ohrsignalmerkmale hinzu (vgl. Blauert und Braasch 2008, S. 88). Dies sind vornehmlich spektrale Merkmale, die durch Kopf- und Ohrform bedingt sind und deren Einfluss – zusammen mit **ILD** und **ITD** – mittels einer **HRFT** (Head-Related Transfer Function) beschrieben werden kann (vgl. Pulkki 1997).

2.3 Summenlokalisierung & Phantomschallquellen

Bei der Wiedergabe des gleichen Signals über zwei Lautsprecher, die sich an unterschiedlichen Punkten mit gleichem Abstand zum **Hörer*** befinden, ergibt sich über das psychoakustische Phänomen der Summenlokalisierung eine Ortung

einer sogenannten Phantomschallquelle zwischen den Lautsprechern (vgl. Blauert und Braasch 2008, S. 101). Die Lokalisationsschärfe ist dabei geringer als bei einer realen Schallquelle (vgl. Blauert und Braasch 2008, S. 101). Die Lokalisation und wahrgenommene Breite von Phantomschallquellen in der Horizontalen wurde ausgiebig von Matthias Frank (u.a. 2013) erforscht. Gribben & Lee (u.a. 2017) erforschen dies zusätzlich in der Vertikalen.

2.4 Klangfärbung

Beim paarweisen Panning zwischen Lautsprechern entstehen bei der Wiedergabe über zwei Lautsprecher Kammfiltereffekte an den beiden Ohrpositionen, die bei der Wiedergabe über einen einzelnen Lautsprecher nicht auftreten, wobei die Färbung bei einer größeren Dichte an Lautsprechern zunimmt (vgl. Zotter und Frank 2019, S. 36). Diese Färbung tritt jedoch nur bei einem bewegten Panning auf, während sie bei statischen Quellen laut einem Experiment von Theile nicht ins Gewicht fällt (vgl. Zotter und Frank 2019, S. 36). Die Färbung tritt ebenfalls auf bei einer sich verändernden Dichte an Lautsprechern (vgl. Zotter und Frank 2019, S. 36). Am auffälligsten sind die Kammfiltereffekte unter reflexionsarmen Bedingungen und werden in reflektiven Räumen kaschiert (vgl. Zotter und Frank 2019, S. 36–37). Das »Composite Loudness Level« (CLL) sagt die wahrgenommene Färbung voraus und errechnet sich aus der Summe der Lautheiten des Schalls an beiden Ohren in Terzbändern (vgl. Zotter und Frank 2019, S. 36). Damit zusammenhängend variiert die Lokalisationsschärfe je nach Anzahl gleichzeitig spielender Lautsprecher – so ist sie besonders groß, wenn nur ein Lautsprecher angesprochen wird, und wird kleiner je mehr Lautsprecher gleichzeitig spielen (vgl. Pulkki 2000).

Um Färbungen durch Kammfiltereffekte gering sowie die wahrgenommene Ausdehnung bei bewegten Schallquellen konstant zu halten, ist es empfehlenswert die Anzahl der simultan spielenden Lautsprecher möglichst konstant zu halten (Zotter und Frank 2019, S. 37).

2.5 Ausdehnung

In der akustischen Realität treten nur selten reine Punktschallquellen auf – die meisten Schallereignisse sind ausgedehnt und setzen sich aus mehreren räumlich verteilten einzelnen Punktschallquellen zusammen (vgl. Santala und Pulkki [2011](#)). Bei der Wahrnehmung der Ausdehnung eines solchen Schallereignisses muss das Ohr bzw. das Gehirn die gleichen Mechanismen nutzen, wie bei der Lokalisation von Punktschallquellen (vgl. Santala und Pulkki [2011](#)). Bei einem als ausgedehnt bzw. breit wahrgenommenen Schallereignis sind die aus unterschiedlichen Richtungen eintreffenden Schallkomponenten weniger korreliert als bei einem nicht ausgedehnt wahrgenommenen Ereignis (vgl. Santala und Pulkki [2011](#)).

3 Analysen

In diesem Abschnitt sollen die theoretischen Grundlagen für eine geeignete Signalverarbeitungskette für die Decodierung sowie das Panning der nativen Signale der MHV-Stützmikrofonanordnung erarbeitet werden. In diesem Zuge wird zunächst die MHV-Mikrofonie im mathematisch-physischen Detail beschrieben und dann ein geeigneter Panningalgorithmus evaluiert bevor im Folgekapitel eine gesamte Signalverarbeitungskette erarbeitet wird.

3.1 Die MHV–Mikrofonie

Die folgenden Beschreibungen gehen zunächst von einer idealen und für alle Frequenzen gleichen Richtwirkung der Mikrofone aus (vgl. Benjamin und Chen 2005). Dabei weichen in der Praxis die Polardiagramme besonders für hohe Frequenzen vom Ideal ab und ergeben entsprechend auch in der Matrizierung und Wechselwirkung von der theoretischen Abbildung abweichende und eventuell unerwünschte Effekte. Auch gehen die Beschreibungen von einer idealen koinzidenten Anordnung der Mikrofonkapseln aus, bei der sich die Mikrofone auf exakt der gleichen Stelle befinden und es somit keine Laufzeitunterschiede gibt (vgl. Benjamin und Chen 2005). Ebenso vernachlässigen die folgenden Polardiagrammdarstellungen die Polaritätsdrehung, die bei rückwärtig eintreffendem Schall bei Druckgradientenempfängern auftritt (vgl. Dickreiter u. a. 2014, S. 254).

Betrachtet man die von Hofmann (2020) beschriebene MHV-Mikrofonie und zerlegt sie in zweidimensionale Stereophonieverfahren, erhält man auf der Horizontalebene in Bezug zur 0° -Richtung mit Winkel $\varphi = 0^\circ$ der Schallquelle durch die Signale M und H eine Stereo-MS-Mikrofonie. Aus den Signalen M und V ergibt sich auf der Medianebene ebenfalls eine Stereo-MS-Mikrofonie, jedoch um 90° um

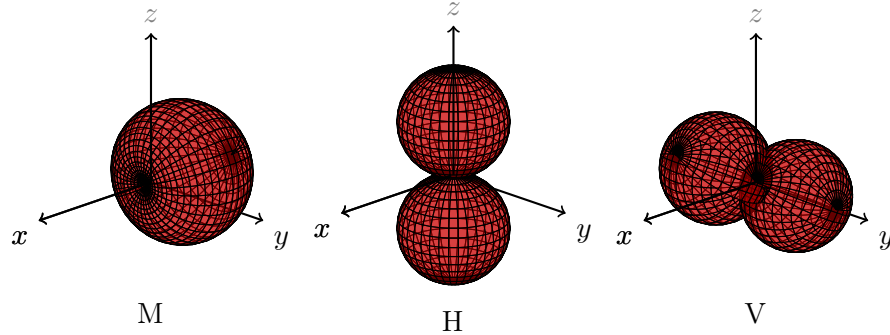


Abbildung 3.1: Die Richtcharakteristika der MHV-Mikrofonie

die x-Achse des kartesischen Koordinatensystems rotiert. Zuletzt ergibt sich aus den Signalen H und V ein 90° um die x-Achse sowie 45° um die y-Achse rotiertes Blumlein-Stereofonie-Verfahren.

Auf diese Ähnlichkeiten soll im Folgenden mathematisch eingegangen werden, um schließlich das MHV-Verfahren auf verschiedene Weisen beschreiben und in Signalketten einbinden zu können.

Das MHV-System entspricht außerdem dem als natives B-Format (vgl. Benjamin und Chen 2005) bezeichneten Verfahren, jedoch mit dem Unterschied, dass die Richtungskomponente X nicht nativ vorhanden ist und das M-Signal aus einer Niere besteht, also keiner reinen Druckkomponente W entspricht.

3.1.1 Verwandte Stereofonieverfahren

Die allgemeinen Formeln für die MS-Matrizierung der Mikrofonsignale M und S lauten $L = \frac{1}{2}(M + S)$ für den linken Stereokanal sowie $R = \frac{1}{2}(M - S)$ für den rechten Stereokanal (vgl. Görne 2015, S. 305). Zur Kompensation der Addition der matrizen Signale – besonders im Falle einer erneuten Summierung – werden die addierten Mikrofonsignale pro Stereokanal mit einem Faktor multipliziert. Görne (2015) verwendet hier $\frac{1}{\sqrt{2}}$, was einer Absenkung von 3dB entspricht. Der jeweilige Wert ist jedoch primär in der tatsächlichen Decodierung zu berücksichtigen und

abhängig vom Korrelationsgrad der Stereosignale (vgl. Görne 2015, S. 305). Im Folgenden soll der Einfachheit halber mit dem linearen Faktor $\frac{1}{2}$ gerechnet werden.

Die Beschreibung von XY- und MS-Mikrofonie geht auf die Patente Alan Dower Blumleins zurück (Zotter und Frank 2019, S. 1). Die Blumlein-Mikrofonie besteht aus zwei Acht-Mikrofonen im 90°-Winkel (Abbildung 3.2). Die Richtcharakteristik der beiden Druckgradientenempfänger X und Y kann im zweidimensionalen Schnitt in der horizontalen Ebene mithilfe der Cosinusfunktion in Bezug auf den Schalleinfallswinkel φ mit Mikrofonversatzwinkel ϕ beschrieben werden (Zotter und Frank 2019, S. 2).

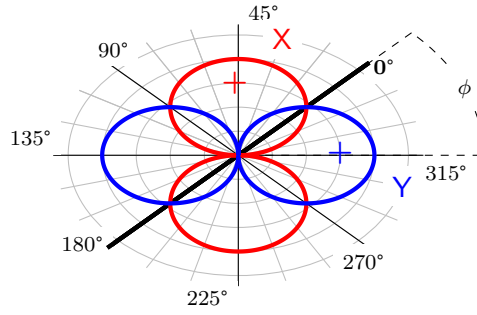


Abbildung 3.2: Die Blumlein-Stereofonie mit zwei Achten mit Mikrofonversatzwinkel $\phi = 45^\circ$ (nach Zotter und Frank 2019)

Der vom Schallrichtungswinkel φ abhängige Pegelverlauf einer aus zwei Achten im 90°-Winkel bestehenden Blumlein-Mikrofonie mit Mikrofonversatzwinkel $\phi = 45^\circ$ lässt sich wie folgt berechnen (vgl. Zotter und Frank 2019, S. 2):

$$\mathbf{g}_{XY}(\varphi) = \begin{bmatrix} L \\ R \end{bmatrix} = \begin{bmatrix} \cos(\varphi + \phi) \\ \cos(\varphi - \phi) \end{bmatrix} = \begin{bmatrix} \cos(\varphi + 45^\circ) \\ \cos(\varphi - 45^\circ) \end{bmatrix} \quad (3.1)$$

Der Winkel φ entspricht dabei dem Winkel der Schallquelle in Bezug zur Haupteinfallrichtung mit $\varphi = 0^\circ$ (Zotter und Frank 2019, S. 2).

Analog lautet die Formel für ein natives MS aus Kugel und um $\phi = 90^\circ$ gedrehter Acht (vgl. Zotter und Frank 2019, S. 4):

$$\mathbf{g}_{MS}(\varphi) = \begin{bmatrix} M \\ S \end{bmatrix} = \begin{bmatrix} 1 \\ \cos(\varphi - 90^\circ) \end{bmatrix} = \begin{bmatrix} 1 \\ \sin(\varphi) \end{bmatrix} \quad (3.2)$$

Das MHV-Mikrofonverfahren wendet für das M-Signal sowohl auf der Horizontalen als auch in der Vertikalen eine Niere an, wodurch M kein reiner Druckempfänger mehr ist:

$$\mathbf{g}_{MS}(\varphi) = \begin{bmatrix} M \\ S \end{bmatrix} = \begin{bmatrix} \frac{1}{2} + \frac{1}{2}\cos(\varphi) \\ \cos(\varphi - 90^\circ) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} + \frac{1}{2}\cos(\varphi) \\ \sin(\varphi) \end{bmatrix} \quad (3.3)$$

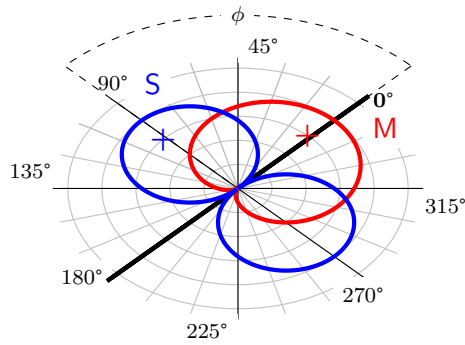


Abbildung 3.3: Die MS-Mikrofonie mit Niere M , Acht S und Mikrofonversatzwinkel $\phi = 90^\circ$

Die Decodierung der Signale M und S auf die Stereokanäle L und R kann durch eine Matrizierung ausgedrückt werden (vgl. Zotter und Frank 2019, S. 5):

$$\begin{bmatrix} L \\ R \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} M \\ S \end{bmatrix} \quad (3.4)$$

Angewandt auf die Richtcharakteristika der beiden Mikrofone M und S ergibt sich daraus:

$$\begin{bmatrix} L \\ R \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} M \\ S \end{bmatrix} \quad (3.5)$$

$$= \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} + \frac{1}{2} \cos(\varphi) \\ \cos(\varphi - 90^\circ) \end{bmatrix} \quad (3.6)$$

$$= \frac{1}{2} \begin{bmatrix} \frac{1}{2} + \frac{1}{2} \cos(\varphi) + \cos(\varphi - 90^\circ) \\ \frac{1}{2} + \frac{1}{2} \cos(\varphi) - \cos(\varphi - 90^\circ) \end{bmatrix} \quad (3.7)$$

$$= \frac{1}{2} \begin{bmatrix} \frac{1}{2} + \frac{1}{2} \cos(\varphi) + \sin(\varphi) \\ \frac{1}{2} + \frac{1}{2} \cos(\varphi) - \sin(\varphi) \end{bmatrix} \quad (3.8)$$

3.1.2 MS mit variabler Richtcharakteristik

Jede MS-Anordnung lässt sich in eine äquivalente XY-Anordnung umrechnen (Hibbing 1989), wodurch die beiden resultierenden Lautsprechersignale L und R wiederum Richtcharakteristika entsprechen (vgl. Wittek, Haut und Keinath 2006). So entsprechen die addierten bzw. subtrahierten Richtcharakteristika von Niere mit $r_M(\varphi) = \frac{1}{2} + \frac{1}{2} \cos(\varphi)$ und um 90° gedrehter Acht mit $r_S(\varphi) = \cos(\varphi - 90^\circ) = \sin(\varphi)$ (vgl. Abbildung 3.3) jeweils $r_X(\varphi) = \frac{1}{2} + \frac{1}{2} \cos(\varphi) + \sin(\varphi)$ bzw. $r_Y(\varphi) = \frac{1}{2} + \frac{1}{2} \cos(\varphi) - \sin(\varphi)$ (vgl. Abbildung 3.5). Da die Anteile der beiden Mikrofonsignale in der Praxis variabel sein sollten, verändern sich auch die äquivalenten Richtcharakteristika. Zur Skalierung sollen die beiden Faktoren a für den M -Anteil und b für den S -Anteil eingeführt werden.

Die beiden resultierenden addierten Formeln entsprechen einem mathematischen Spezialfall, da hier zwei zueinander orthogonale trigonometrische Funktionen addiert werden. Die allgemeine Ausgangsformel dafür lautet $f(\varphi) = a \cos(\varphi) + b \sin(\varphi)$. Aus den trigonometrischen Additionstheoremen ergibt sich $f(\varphi) = a \cos(\varphi) + b \sin(\varphi) = A \cos(\varphi - \phi)$, wobei $A = \sqrt{a^2 + b^2}$ und $\phi = \tan^{-1}(\frac{b}{a})$ mit $a \geq 0$ und $b \geq 0$ gilt.

Angewandt auf $r(\varphi) = \frac{1}{2} + \frac{1}{2} \cos(\varphi) \pm 1 \sin(\varphi)$ ergibt sich die auf eine Cosinus-Komponente reduzierte Form:

$$\begin{bmatrix} L \\ R \end{bmatrix} = \begin{bmatrix} X \\ Y \end{bmatrix} = \frac{1}{2} \begin{bmatrix} a & b \\ a & -b \end{bmatrix} \begin{bmatrix} M \\ S \end{bmatrix} \quad (3.9)$$

$$= \frac{1}{2} \begin{bmatrix} \frac{a}{2} + \frac{a}{2} \cos(\varphi) + b \sin(\varphi) \\ \frac{a}{2} + \frac{a}{2} \cos(\varphi) - b \sin(\varphi) \end{bmatrix} \quad (3.10)$$

$$= \frac{1}{2} \begin{bmatrix} \frac{a}{2} + \sqrt{\frac{a^2}{2} + b^2} \cos(\varphi - \tan^{-1}(\frac{2b}{a})) \\ \frac{a}{2} + \sqrt{\frac{a^2}{2} + b^2} \cos(\varphi + \tan^{-1}(\frac{2b}{a})) \end{bmatrix} \quad (3.11)$$

Diese Schreibweise ist für die Beschreibung der resultierenden Mikrofoncharakteristik von Vorteil, da hier Druckgradienten- und Druckanteil einer Nierencharakteristik direkt ersichtlich sind (vgl. Dickreiter u. a. 2014, S. 158). Der Faktor $\sqrt{\frac{a^2}{2} + b^2}$ bestimmt bzw. skaliert den Druckgradientenanteil $\cos(\dots)$ und der Summand $\frac{a}{2}$ den Druckanteil der resultierenden Nierencharakteristik. Der Summand $\tan^{-1}(\frac{2b}{a})$ gibt die Phasenverschiebung bzw. Drehung des Druckgradienten an.

In der Praxis ist eine lineare Überblendung der beiden Mikrofonsignale geeigneter als die getrennte Skalierung durch die Faktoren a und b . Dafür ist es sinnvoll b von a mit der Bedingung $b = 1 - a$ ($0 \leq a \leq 1$) abhängig zu machen, sodass jede Richtcharakteristik nur über a eindeutig beschrieben werden kann. Dies hat den zusätzlichen Vorteil, dass die resultierende Richtcharakteristik in ihrer jeweiligen Nullrichtung ihren Pegel nicht verändert. Abbildung 3.5 zeigt drei Fälle der Parametrisierung von a und b mit $b = 1 - a$ ($0 \leq a \leq 1$) mit dem resultierenden Mikrofonversatzwinkel ϕ und dem resultierenden Druckgradientenanteil A und Kugelanteil K (vgl. auch Wittek, Haut und Keinath 2006).

Faktor a ($a \geq 0$) skaliert also den Signalanteil des Nierenmikrofons M bzw. den mathematischen Cosinusanteil, während b ($b \geq 0$) den Signalanteil des orthogonalen Mikrofons S mit Achtcharakteristik bzw. den mathematischen Sinusanteil skaliert. Ein größerer Faktor b vergrößert den Öffnungswinkel des resultierenden XY-Verfahrens sowie die Richtwirkung der äquivalenten Mikrofonsignale X und Y .

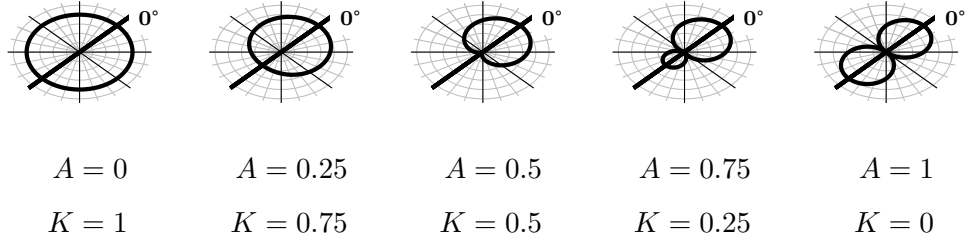


Abbildung 3.4: Die Niere aus einem Mischverhältnis von Kugelanteil K und Achtanteil A mit der Richtcharakteristik $r(\varphi) = K + A \cos(\varphi)$ (nach Dickreiter u. a. 2014, S. 178)

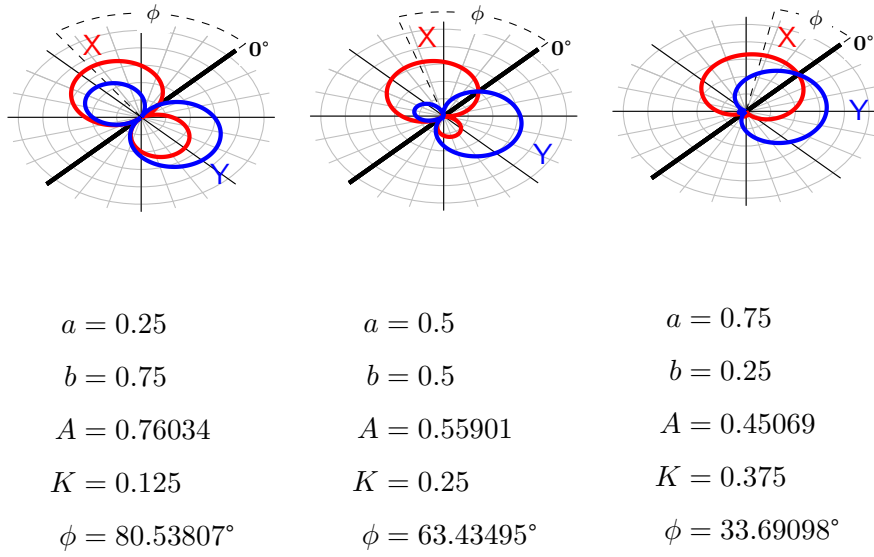


Abbildung 3.5: MS mit variablen Anteilen von M und S mit resultierendem Mikrofonversatzwinkel ϕ , Druckgradientenanteil A und Kugelanteil K

3.1.3 MHV mit variabler Breite und Höhe

Die Äquivalenz zur XY-Stereofonie lässt sich sowohl auf die MS-Matrizierung von M und H mit dem horizontalen Schalleinfallswinkel φ anwenden als auch auf M und V mit dem vertikalen Schalleinfallswinkel δ .

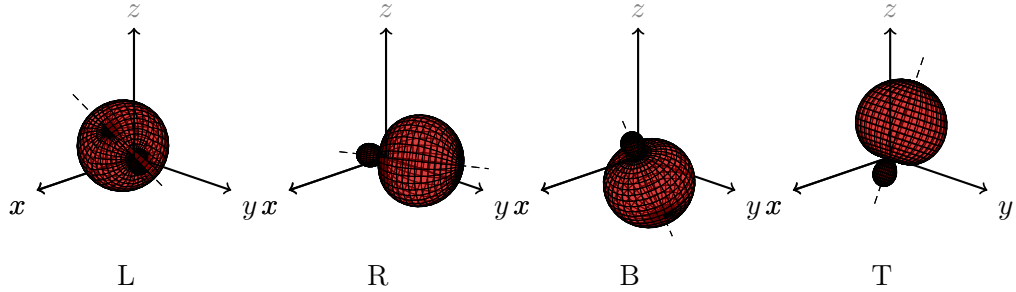
Hofmann (2020) schlägt für die Decodierung des MHV-Verfahrens eine Stereo-Decodierung der beiden beschriebenen MS-Systeme aus M und H bzw. M und V auf zwei Stereosignalpaare vor, wobei das horizontale Stereosignalpaar als LR (Left/Right) mit $L = M + H$ bzw. $R = M - H$ und das vertikale Stereosignalpaar als BT (Bottom/Top) mit $B = M + V$ bzw. $T = M - V$ bezeichnet wird (vgl. Hofmann 2020, S. 49). Die Decodierung der drei Mikrofonssignale auf die zwei Stereosignalpaare lässt sich unter Berücksichtigung der in Formel 3.11 eingeführten Gewichtungsfaktoren a und b in einer Matrix ausdrücken. Um die beiden Signalpaare LR bzw. MH und BT bzw. MV unabhängig gewichten zu können, müssen unabhängige Faktoren a und b verwendet werden. Die Gewichtung von M zu H wird durch a_{MH} und b_{MH} bestimmt, die Gewichtung von M zu V durch a_{MV} und b_{MV} . So entsteht die Decodierungsgleichung:

$$\begin{bmatrix} L \\ R \\ B \\ T \end{bmatrix} = \frac{1}{2} \begin{bmatrix} a_{MH} & b_{MH} & 0 \\ a_{MH} & -b_{MH} & 0 \\ a_{MV} & 0 & b_{MV} \\ a_{MV} & 0 & -b_{MV} \end{bmatrix} \begin{bmatrix} M \\ H \\ V \end{bmatrix} \quad (3.12)$$

Die Signale L , R , B und T entsprechen wie auch schon im Falle der beschriebenen MS-Stereofonie äquivalenten XY-Richtcharakteristika. Diese sind in Abbildung 3.6 in dreidimensionaler Form zu sehen. Analog zur Abbildung 3.5 ist auch hier die Richtwirkung sowie der Mikrofonversatzwinkel abhängig von den Faktoren a und b bzw. a_{MH} , b_{MH} , a_{MV} und b_{MV} .

3.1.4 Das native B-Format-Mikrofon

Das MHV-Mikrofonieverfahren weist eine große Ähnlichkeit zum von Benjamin und Chen (2005) als natives B-Format-Mikrofon bezeichneten Mikrofoniever-



Abbildungung 3.6: Die äquivalenten Richtcharakteristika der LRBT-Signale mit gleicher Gewichtung aller Mikrofon-signale

fahren auf, das aus einem Druckempfänger mit der Bezeichnung W sowie drei Druckgradientenempfängern X , Y und Z besteht. Diese drei Mikrofone sind wiederum auf den drei Achsen des dreidimensionalen kartesischen Koordinatensystems angeordnet (vgl. Zotter und Frank 2019, S. 9). So entspricht auch beim MHV-Mikrofonieverfahren der Mikrofonkanal H der Ambisonics-Komponente Y und der Mikrofonkanal V der Ambisonics-Komponente Z . Nur M stellt mit seiner Nierencharakteristik eine Ausnahme dar und entspricht nur im Kugelanteil von M nativ der Ambisonics-Komponente W sowie im Druckgradientenanteil der Richtungskomponente X .

Diese Betrachtungsweise kann der Startpunkt einer weitergehenden Analyse und einer alternativen Decodierung sein, wird jedoch im Rahmen dieser Arbeit nicht weiter verfolgt.

3.2 Panningverfahren

Es gibt verschiedene Möglichkeiten mit den im Kapitel 3.1 beschriebenen Matrixierungen aus den Signalen der MHV-Anordnung fortzufahren. Wie Hofmann bereits beschrieben hat, ist das Ziel der Anordnung nicht die akkurate Abbildung bzw. Lokalisation jedes Schallereignisses, sondern vornehmlich die realistische und klangästhetisch ansprechende Abbildung der akustischen Ausdehnung der

Schallquelle (vgl. Hofmann 2020). Auch sei erwähnt, dass das Mikrofon nicht als Hauptmikrofon mit dem Ziel einer Reproduktion des Höreindrucks an einer virtuellen Hörerposition entwickelt wurde, sondern als ergänzendes oder auch eigenständiges Stützmikrofon (vgl. Hofmann 2020). Ein wichtiger Aspekt in der Entwicklung des MHV-Stützmikrofonverfahrens war die gute Abwärtskompatibilität koinzidenter Mikrofonien (vgl. Hofmann 2020, S. 48). Dieser Gedanke soll auch auf der Seite des Pannings und der Wiedergabe fortgesetzt werden. Daher beschränkt sich die vorliegende Arbeit auf die Untersuchung der Positionierung der Signale des MHV durch [Amplitudenpanning](#). Die gute Abwärtskompatibilität koinzidenter Mikrofonsysteme rührt daher, dass diese ausschließlich Lautstärkeunterschiede aus verschiedenen Richtungen an einem einzigen Punkt im Raum aufzeichnen und die Komponente Laufzeit aufnahmeseitig außen vor lassen (vgl. Weinzierl 2008, S. 593). Daher können diese Signale im Falle eines [Downmix](#), also dem Prozess einer Reduktion der zur Verfügung stehenden Kanäle durch Summierung selbiger, summiert werden, ohne dass unerwünschte Kammfiltereffekte auftreten (vgl. Weinzierl 2008, S. 566). Analog dazu erreicht [Amplitudenpanning](#) Lokalisation ausschließlich durch in der Lautstärke unterschiedliche Wiedergabe des gleichen Signals auf Lautsprechern, die sich üblicherweise im gleichen Abstand zum [Hörer*](#) befinden (vgl. Pulkki 1997). Um die Ausdehnungsabbildung innerhalb eines mehrdimensionalen Amplitudenpanningsystems theoretisch zu analysieren, ist ein grundlegendes Verständnis von selbigem nötig. Dazu soll zunächst auf das von Ville Pulkki (1997) entwickelte [Vector-Base Amplitude Panning](#) eingegangen werden, das eine Variante des [Amplitudenpannings](#) darstellt.

3.2.1 VBAP

Das von Ville Pulkki (1997) beschriebene [Vector-Base Amplitude Panning](#) (VBAP) ist ein Algorithmus, der virtuelle Schallquellen auf einer sphärischen Anordnung beinahe beliebig vieler Lautsprecher abbilden kann, indem es mithilfe der am nächsten an der Schallquellenposition gelegenen Lautsprecher Phantomschallquellen erzeugt (vgl. Zotter und Frank 2019, S. 41). [VBAP](#) stellt so eine Erweiterung

des stereofonen [Amplitudenpannings](#) für die Verwendung mit mehr als zwei Lautsprechern dar (vgl. Pulkki 1999). Im zweidimensionalen Fall – also mit Lautsprecher- und Schallquellenpositionen auf der Horizontalebene – geben ein bis zwei Lautsprecher gleiche Signal mit unterschiedlichen Lautstärken bzw. Gains wieder (vgl. Zotter und Frank 2019, S. 41). Im dreidimensionalen Fall – also mit Lautsprecher- und Schallquellenpositionen auf einer Sphäre – sind es ein bis drei Lautsprecher (vgl. Zotter und Frank 2019, S. 41). Um die Position der virtuellen Schallquelle in Relation zum [Hörer*](#) dreidimensional zu beschreiben werden Einheitsvektoren genutzt (vgl. Reiss und McPherson 2014, S. 219).

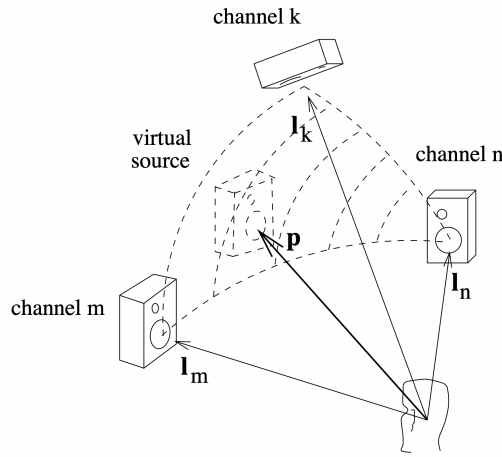


Abbildung 3.7: Dreidimensionales [Vector-Base Amplitude Panning](#) (Pulkki 1999)

Dabei zeigt auf jeden Lautsprecher ein Einheitsvektor $\mathbf{l}$ (Abbildung 3.7), der dessen Position beschreibt (vgl. Reiss und McPherson 2014, S. 219).

$$\mathbf{l}_n = \begin{bmatrix} l_{n1} \\ l_{n2} \\ l_{n3} \end{bmatrix} \quad (3.13)$$

Der Positionsvektor $\mathbf{p}$ (Abbildung 3.7) der virtuellen Schallquelle lässt sich mithilfe der Gainfaktoren g_n , g_m und g_k sowie der Lautsprecherpositionen $\mathbf{l}_n$, $\mathbf{l}_m$

und $\mathbf{l}_k$ der Lautsprecher n , m und k berechnen (vgl. Reiss und McPherson 2014, S. 219).

$$\mathbf{p} = \begin{bmatrix} p_n \\ p_m \\ p_k \end{bmatrix} = \begin{bmatrix} \mathbf{l}_n & \mathbf{l}_m & \mathbf{l}_k \end{bmatrix} \begin{bmatrix} g_n \\ g_m \\ g_k \end{bmatrix} \quad (3.14)$$

VBAP kann mit einem beinahe beliebigen mehrdimensionalen Lautsprecher-setup mit mindestens drei Lautsprechern verwendet werden (vgl. Pulkki 1997). Dazu wird die Sphäre, auf der sich die Lautsprecher befinden, in nicht-überlappende Tripel eingeteilt (Abbildung 3.8) (vgl. Pulkki 1997) – dieser Prozess wird auch als Triangulation bezeichnet (vgl. Zotter und Frank 2019, S. 48). Liegt die virtuelle Schallquelle mit der Position $\mathbf{p}$ innerhalb eines Lautsprecher-Tripel $\mathbf{L}_{kmn}$, werden die Gainfaktoren $\mathbf{g}$ für dieses berechnet (vgl. Pulkki 1997).

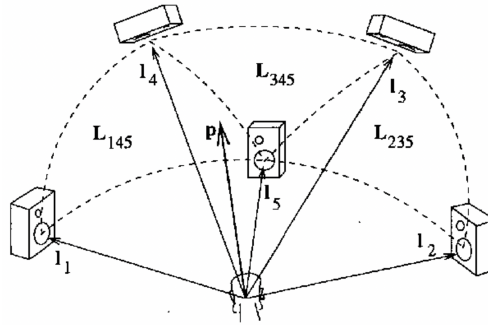


Abbildung 3.8: Beispielhafte Tripelbildung der Lautsprecher beim VBAP (Pulkki 1997)

Die rein mathematische Triangulation bringt jedoch besonders auf herkömmlichen dreidimensionalen Lautsprecheranordnungen wie 4+7+0 teilweise unerwünschte Resultate hervor, da in der Regel für den Hörer* symmetrische Tripel wünschenswert sind (vgl. Zotter und Frank 2019, S. 48). Dazu können ein oder mehrere Phantomlautsprecher bzw. »imaginary speakers« platziert werden – in der Regel im Zenit und Nadir der Lautsprecheranordnung –, sodass eine gleichmäßige Aufteilung erreicht und Sprünge von Schallquellen zwischen Tripeln reduziert

werden können (vgl. Zotter und Frank 2019, S. 48). Auch Schallquellen, die unterhalb oder oberhalb des **Middle Layer** liegen, können so zwar nicht an ihrer tatsächlichen Position abgebildet werden, jedoch trotzdem sinnvoll auf dem **Middle Layer** oder **Upper Layer** repräsentiert werden (vgl. Zotter und Frank 2019, S. 49).

Beim Panning per **VBAP** ist zu beachten, dass nur virtuelle Schallquellenpositionen auf der durch die Lautsprecher definierten Sphäre abgebildet werden können (vgl. Pulkki 1997). Ebenso können virtuelle Schallquellen nur innerhalb des jeweils aktiven Lautsprecherdreiecks abgebildet werden (Pulkki 1997).

Problematisch beim **VBAP** ist die entstehende Klangfärbung sowie die wahrgenommene Breite der Schallquelle bei der Bewegung der virtuellen Schallquelle auf der Sphäre, da je nach Position – teilweise sprunghaft – die Anzahl der gleichzeitig spielenden Lautsprecher variiert (vgl. Zotter und Frank 2019, S. 44).

3.2.2 VBIP

Vector-Base Intensity Panning (VBIP) stellt eine Abwandlung des **Vector-Base Amplitude Panning** dar, indem es die Werte des Gain-Vektors aus Gleichung 3.14 jeweils quadriert (vgl. Zotter und Frank 2019, S. 44). Dies trägt zu einer größeren Übereinstimmung von wahrgenommener virtueller Schallquelle und dem rechnerischen Richtungsvektor p bei (vgl. Zotter und Frank 2019, S. 44).

3.2.3 MDAP

Das **Multiple-Direction Amplitude Panning** (MDAP) ist eine von Ville Pulkki (1999) vorgeschlagene Abwandlung des **VBAP**, die sich der Problematik der inkonsistenten Klangfärbung und wahrgenommenen Breite der virtuellen Schallquelle an verschiedenen Positionen annimmt (Pulkki 2001). **MDAP** hält die Anzahl der gleichzeitig spielenden Lautsprecher mit variierender Position der virtuellen Schallquelle möglichst konstant, um Klangverfärbungen zu vermeiden (vgl. Zotter und Frank 2019, S. 41). Dies wird erreicht, indem zusätzlich zur eigentlichen virtuellen Schallquellenposition virtuelle Schallquellen umliegend mit einem de-

finierten Positionsversatz platziert werden, was in Abbildung 3.9 sichtbar wird (vgl. Zotter und Frank 2019, S. 45).

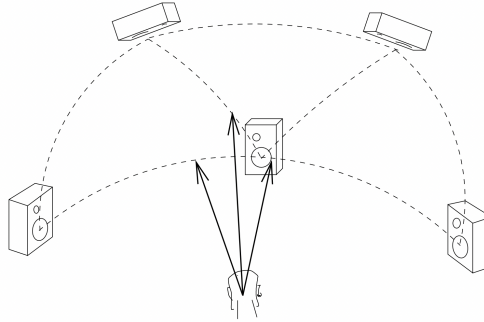


Abbildung 3.9: Dreidimensionales [Multiple-Direction Amplitude Panning](#) mit Spreizung der virtuellen Schallquelle auf drei Positionen (Pulkki 1999)

Implementierungsseitig kann [MDAP](#) in Form eines Spread-Parameters realisiert werden, der kontrolliert, in welchem Anteil und mit welcher Verteilung die zusätzlichen Schallquellenpositionen einbezogen werden sollen oder ob nur die ursprüngliche Schallquellenposition per herkömmlichem [VBAP](#) gepannt wird (vgl. Pulkki 2000).

Im Kontext der Wiedergabe von ausgedehnten Schallquellen bzw. aufnahmeseitiger Diffusität ist [MDAP](#) jedoch ungeeigneter, da beim Panning mehrere korrelierte Signale dabei der [ICC](#) zwischen den jeweils angesprochenen Lautsprecherkanälen deutlich größer wird, was zu einer schlechteren Trennung der Signale führt. Dies ist darin begründet, dass die wahrgenommene Diffusität stark vom [Inter-Channel Cross-Correlation Coefficient \(ICC\)](#) abhängig ist und bei niedrigem [ICC](#) und niedriger Anzahl gleichzeitig spielender Lautsprecher höher ist (vgl. Cousins, Bleack und Fazi 2017).

4 Konzeption

Im Kern ist die Aufgabe des zu entwickelnden Postproduktionstools die Decodierung der Mikrofonsignale (Abschnitt 3.1) in vier Einzelsignale sowie das Panning, also die Positionierung dieser Schallquellen im mehrdimensionalen Lautsprechersetup (Abschnitt 3.2). Dazu soll im Folgenden ein Konzept für eine Signalverarbeitungskette, für die Parametrisierung selbiger sowie für eine geeignete Benutzeroberfläche vorgestellt werden.

Dieses Kapitel ist als verallgemeinerbares Konzept zu verstehen, welches unabhängig von der jeweiligen Implementierung Bestand hat. Ein konkreter Implementierungsvorschlag dieses Konzeptes ist in Kapitel 5 zu finden.

4.1 Signalverarbeitungskonzept

Signalverarbeitungstechnisch gibt es eine Aufteilung in zwei Blocks – Decodierung und Panning (Abbildung 4.1). Der Decodingblock orientiert sich hier an der mathematischen Beschreibung aus Formel 3.12 und ist in Abbildung 4.2 zu sehen. Der Panningblock sieht ein VBAP vor. Abbildung 4.3 zeigt beispielhaft die Multiplikation eines Mono-Eingangssignals mit den in Abschnitt 3.2 beschriebenen Lautsprechergainfaktoren g – dieser Block muss viermal implementiert werden, um die vier Schallquellen L , R , B und T positionieren zu können.

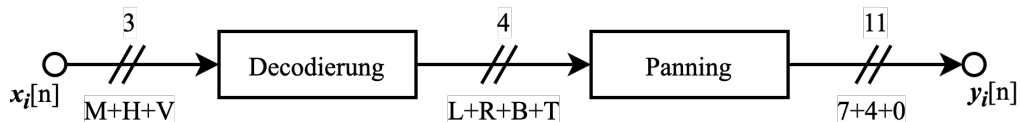


Abbildung 4.1: Das Blockschaltbild der Audiosignalverarbeitungskette

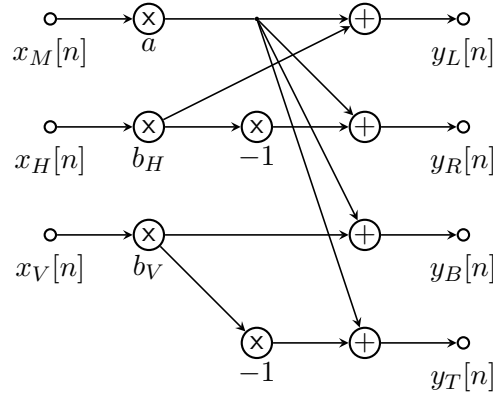


Abbildung 4.2: Blockschaltbild: Decodierung der Mikrofon-signale in zwei Stereo-signalpaare

Lautsprecher-setup

Zur Abbildung von Höhe und vertikaler Ausdehnung ist eine Erweiterung des herkömmlichen zweidimensionalen Stereo- oder Surround-Setups nötig (vgl. Nettingsmeier 2012). Dazu wird das sogenannte **Middle Layer**, das sich auf Höhe des Hörer*s befindet, um ein **Upper Layer** erweitert, das sich wiederum über dem Hörer* befindet (vgl. ITU 2018). Optional kann zusätzlich ein **Lower Layer** unterhalb des **Middle Layer** genutzt werden. Alle Lautsprecher befinden sich idealerweise auf einer Sphäre (vgl. ITU 2018).

Es sei an dieser Stelle zu betonen, dass das hier entwickelte Konzept für beliebige dreidimensionale sphärische Lautsprecheranordnung Anwendung finden kann. Hier soll auf das später in der Implementierung verwendete Setup **4+7+0** eingegangen werden. Für dieses Lautsprecher-setup kommen in der Fachliteratur verschiedene Bezeichnungen zum Einsatz wie »7.4.0« oder auch »11.0«. Lee (2021) erwähnt auch die Verwechslungsgefahr, die durch die gleiche Benennung von beispielsweise Lautsprecher L und Mikrofon-signal L auftreten kann. Rec. ITU-R BS.2051-2 (ITU 2018) spezifiziert das Layout als **4+7+0**, wobei das Benennungsschema **Upper Layer** + **Middle Layer** + **Lower Layer** zum Einsatz

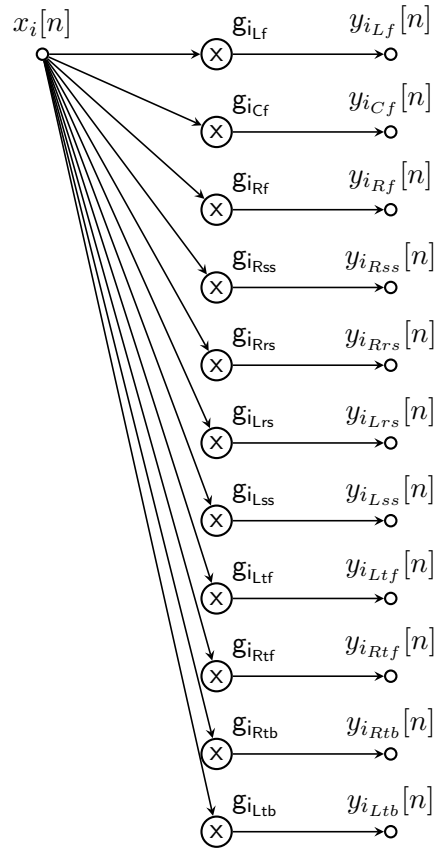


Abbildung 4.3: Blockschaltbild: Beispielhaftes Amplitudenpanning eines Eingangssignals i

Loudspeaker configuration for Sound System J (4+7+0)

SP Label	Channel		Azimuth	Elevation
	Label	Name	Range	Range
M+030	L	Left	+30 .. +45	0
M-030	R	Right	-30 .. -45	0
M+000	C	Centre	0	0
LFE1	LFE	Low frequency effects	–	–
M+090	Lss	Left side surround	+85 .. +110	0
M-090	Rss	Right side surround	-85 .. -110	0
M+135	Lrs	Left rear surround	+120 .. +150	0
M-135	Rrs	Right rear surround	-120 .. -150	0
U+045	Ltf	Left top front	+30 .. +45	+30 .. +55
U-045	Rtf	Right top front	-30 .. -45	+30 .. +55
U+135	Ltb	Left top back	+100 .. +150	+30 .. +55
U-135	Rtb	Right top back	-100 .. -150	+30 .. +55

Abbildung 4.4: Empfehlung Lautsprecherkonfiguration 4+7+0 nach Rec. ITU-R BS.2051-2 (ITU 2018)

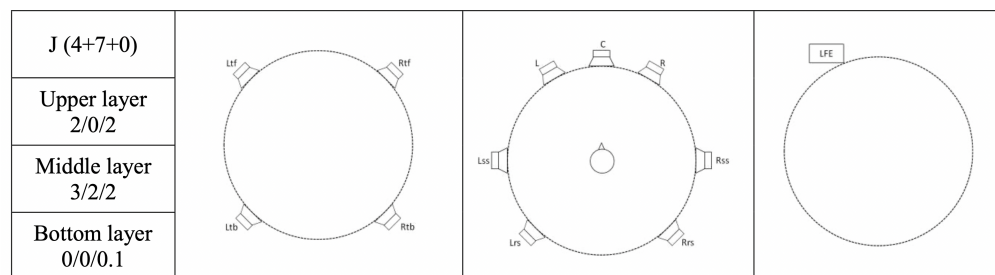


Abbildung 4.5: Empfehlung Lautsprecherlayout 4+7+0 nach Rec. ITU-R BS.2051-2 (ITU 2018)

kommt. Diese Bezeichnung soll im Folgenden verwendet werden. Um jedoch die Verwechslung mit den Bezeichnungen der äquivalenten Mikrofon-signale des **MHV** zu vermeiden, soll nachfolgend den frontalen Lautsprechern des **Middle Layers** ein »f« angehängt werden.

4.2 Parametrisierungskonzept

Tabelle 4.1 zeigt alle nötigen Parameter, die dem **Nutzer*** zugänglich gemacht werden sollen mit ihren Bezeichnungen, Wertebereichen, Einheiten und Funktionen.

4.2.1 Decodierungsparameter

Für die Decodierung der drei Signale M , H und V des **MHV**-Arrays in vier Monosignale L , R , B und T (vgl. Decodierungsgleichung 3.12) ist die Kontrolle über die Gewichtung von H und V im Verhältnis zu M nötig. Dies ermöglichen die in Abschnitt 3.1.2 etablierten MS-Gewichtungsfaktoren a und b . Angewandt auf die beiden MS-Signalpaare ergeben sich die Faktoren a_{MH} , b_{MH} , a_{MV} , b_{MV} aus der Gleichung 3.12 in Abschnitt 3.1.3.

4.2.2 Panningparameter

Eine Möglichkeit des Pannings der Signale des **MHV**-Arrays ist eine freie und unabhängige Platzierung der decodierten Einzelsignale L , R , B und T (vgl. Abschnitt 3.1.3) auf der Lautsprechersphäre. Diese Variante bietet dem **Nutzer*** die maximale Flexibilität und ist zudem mit den meisten herkömmlichen VBAP-Pannern problemlos möglich. Die Problematik besteht jedoch dabei primär in der Nutzerfreundlichkeit, da hier dem **Nutzer*** eine sehr große Zahl an Parametern offen gelegt werden muss. So braucht eine entsprechende Benutzeroberfläche bei einer nicht-grafischen Lösung Azimut- und Elevationsregler für alle vier Signale, was zu einer Gesamtzahl von acht Panningpositionsparametern führt, die in die Benutzeroberfläche implementiert werden müssen.

Gegen diese Offenlegung aller Einzelparameter sprechen ebenfalls einige Faktoren. So entfernt sich diese Variante insbesondere von dem konzeptionellen Verständnis der Signale als Komponenten eines einzelnen Schallereignisses – im Falle des intendierten Einsatzes des [MHV](#)-Arrays als Stützmikrofon ist dieses wahrgenommene Schallereignis idealerweise das mikrofonierte Instrument.

Es bietet sich daher an die Position des Schallereignisses und unabhängig davon dessen Ausdehnung in Breite und Höhe zu beschreiben. So sollen für die Position ein globaler Azimut-Parameter φ_I und ein globaler Elevationsparameter δ_I eingeführt werden, die die Position einer imaginären Schallquelle I beschreiben, um die sich wiederum die Positionen der realen Signale anordnen. Zusätzlich müssen dafür Parameter eingeführt werden, die die Ausdehnung beschreiben – also den Abstand der Einzelsignale von der Schallquelle I bzw. den Basiswinkel der virtuellen Stereosignalpaare. Dafür soll ein horizontaler Ausdehnungswinkelparameter Φ sowie ein vertikaler Ausdehnungswinkelparameter Δ eingeführt werden. Aufgrund der Beschaffenheit des verwendeten Lautsprechersystems [4+7+0](#) ist zusätzlich ein vertikaler Offsetparameter o angebracht, der die BT-Schallquellenpositionen vertikal in Relation zur Nullposition verschieben kann, sodass die Position von B bei einem Winkel $\delta_I = 0^\circ$ nicht unterhalb der Horizontalebene bzw. des [Middle Layer](#) liegen muss. Aus den Positionsdaten der Schallquelle mit horizontaler und vertikaler Ausdehnung ergeben sich die Positionen $(\varphi_{L,R,B,T}, \delta_{L,R,B,T})$ der Einzelsignale nach folgender Abhängigkeit:

$$\begin{aligned}
 \varphi_L &= \varphi_I - \Phi/2 \\
 \delta_L &= \delta_I \\
 \varphi_R &= \varphi_I + \Phi/2 \\
 \delta_R &= \delta_I \\
 \varphi_B &= \varphi_I \\
 \delta_B &= \delta_I - \Delta/2 + o \\
 \varphi_T &= \varphi_I \\
 \delta_T &= \delta_I + \Delta/2 + o
 \end{aligned} \tag{4.1}$$

Das Panningkonzept »T-Orientatation« (Raute) ist schematisch in Abbildung 4.6 am Beispiel der fünf vorderen Lautsprecher eines 4+7+0-Setups zu sehen. Der horizontale Abstand zwischen der imaginären Schallquelle I und Schallquelle L und R beträgt hier die Hälfte des horizontalen Ausdehnungswinkels Φ , während der vertikale Abstand zwischen I und Schallquelle B und T die Hälfte des vertikalen Ausdehnungswinkels Δ beträgt. B und T befinden sich auf der gleichen Azimutposition wie die imaginäre Schallquelle I , während L und R sich auf dem gleichen Elevationsniveau wie I befinden. Eine ähnliche Positionierung wurde bereits von Hofmann (vgl. Hofmann 2020) vorgeschlagen und erprobt.

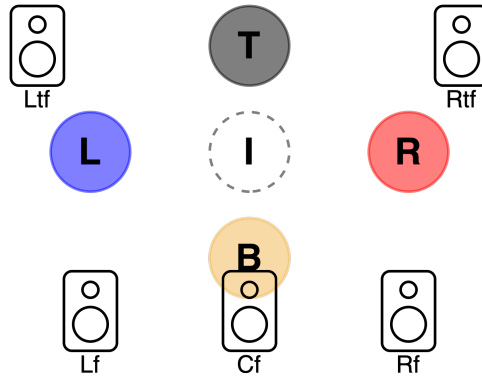


Abbildung 4.6: Panningkonzept »T-Orientatation« (Raute)

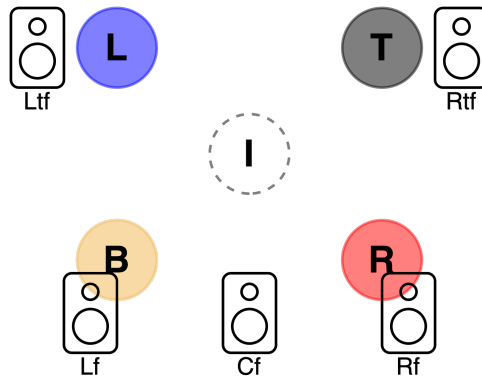


Abbildung 4.7: Panningkonzept »X-Orientatation« (Rechteck)

Eine alternative Positionierung um die imaginäre mittlere Schallquelle I herum ist die Anordnung in einer x-Form, also effektiv einer um 45° gedrehten Variation der »T-Orienta-tion«. Dieses Panningkonzept »X-Orienta-tion« (Rechteck) mit horizontalem Ausdehnungswinkel Φ und vertikalem Ausdehnungswinkel Δ ist in Abbildung 4.7 zu sehen. Die horizontale Winkeldifferenz der imaginären Schallquelle I zur Schallquelle L und B sowie zu R und T beträgt hier die Hälfte des horizontalen Ausdehnungswinkels Φ , während der vertikale Winkelversatz von I zu Schallquelle L sowie zu T bzw. B und R die Hälfte des vertikalen Ausdehnungswinkels Δ beträgt. L und T haben hier die gleiche Elevation, B und R ebenfalls. B und L befinden sich auf der gleichen Azimutposition, ebenso R und T . Die Schallquellenpositionen lassen sich somit durch folgende Anpassung der Gleichungen aus 4.1 berechnen:

$$\begin{aligned}
 \varphi_L &= \varphi_I - \Phi/2 \\
 \delta_L &= \delta_I + \Delta/2 \\
 \varphi_R &= \varphi_I + \Phi/2 \\
 \delta_R &= \delta_I - \Delta/2 \\
 \varphi_B &= \varphi_I - \Phi/2 \\
 \delta_B &= \delta_I - \Delta/2 \\
 \varphi_T &= \varphi_I + \Phi/2 \\
 \delta_T &= \delta_I + \Delta/2
 \end{aligned} \tag{4.2}$$

Ein wiederum alternativer Ansatz ist das diskrete Routing der Signale auf Lautsprecher – also effektiv ein rein kanalbasierter Ansatz. Dazu können verschiedene Kombinationen aus Lautsprecherkanälen als Quadrupel vordefiniert werden. Abbildung 4.8 zeigt beispielhaft ein Quadrupel aus Lf , Rf , Ltf und Rtf . Je nach Implementierung kann signalverarbeitungstechnisch ein direktes Routing oder eine Positionierung der Signale per VBAP auf exakten Lautsprecherpositionen erfolgen.

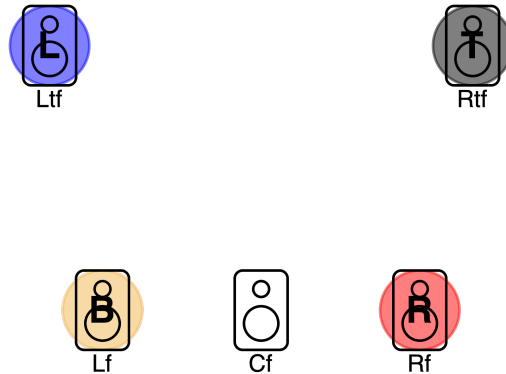


Abbildung 4.8: Panningkonzept »Discrete« (Diskret)

4.3 Benutzeroberflächenkonzept

Die Benutzeroberfläche des Tools soll die Aufgabe haben dem **Nutzer*** alle Parameter aus Tabelle 4.1 in geeigneter und übersichtlicher Weise offenzulegen und gleichzeitig die klanglichen Vorgänge in geeigneter Weise zu visualisieren. So sollte die in Abschnitt 3.1.2 etablierte Visualisierung aus Abbildung 3.5 auch in der Benutzeroberfläche implementiert werden. Ebenfalls ist die Visualisierung der Panningpositionen der virtuellen Schallquellen sowie des Lautsprechersystems eine wichtige Veranschaulichung für den **Nutzer*** und sollte sich an Abbildung 5.2 orientieren. Als Hilfe, um das Übersprechen der Signale auf gleiche Lautsprecher zu überwachen, ist die Visualisierung der Gainfaktoren des **VBAP** sinnvoll.

Name	Abkürzung	Wertebereich	Einheit	Beschreibung
M-Level	m	-70 ... 6	dB	Lautstärkeskalierungsfaktor M-Mikrofonsignal
H-Level	h	-70 ... 6	dB	Lautstärkeskalierungsfaktor H-Mikrofonsignal
V-Level	v	-70 ... 6	dB	Lautstärkeskalierungsfaktor V-Mikrofonsignal
azimuth	α	-180 ... 180	°	horizontaler Azimutwinkel von I
elevation	φ	-180 ... 180	°	vertikaler Elevationswinkel von I
hor. spread	hsread	0 ... 120	°	horizontaler spread-Winkel
vert. spread	vsread	0 ... 120	°	vertikaler spread-Winkel
vert. offset	voffset	0 ... 60	°	vertikaler spread-Winkel
orientation	orient	0, 45	°	Schalter für Ausrichtung der Raute
fixed preset	index	0, 1 .. 12		Wahlschalter für fixed-Preset
pan algorithm	panalgo	vbap, vbip, lbap		Wahlschalter für Panningalgorithmus
phantom speakers	phantomon	0, 1		Wahlschalter für Phantomlautsprecher in der Triangulation
Plot-View	plotview	0, 1		Wahlschalter für native oder äquivalente Polardiagramm-Visualisierung
Input-Selector	inselect	0 ... 32		Wahlmenü für das Audio-Input-Tripel

Tabelle 4.1: Parameter der grafischen Benutzeroberfläche

5 Implementierung

Dieses Kapitel erfüllt neben dem beispielhaften Vorschlag einer Implementierung eines Postproduktionstools für die [MHV](#)-Stützmikrofonanordnung auch die Dokumentation des Entwicklungsprozesses des entstandenen »MHV-Panners«.

Um die in Kapitel 4 erarbeitete Signalverarbeitungskette, ihre Parametrisierung sowie die Benutzeroberfläche prototypisch zu implementieren, wird die Programmierumgebung Cycling74 Max 8 verwendet. Zusätzlich zu einer Max-Standalone-App, die fünf Instanzen des »MHV-Panners« enthält, wird ein Max for Live Device erstellt, welches den ursprünglich intendierten Einsatz der Software als DAW-Plugin emulieren soll. Beide Versionen des Tools sind größtenteils deckungsgleich mit einigen wenigen unterscheidenden Spezifika.

Beide Varianten der prototypischen Implementierung weisen neben den genannten Vorteilen ebenfalls einige durch die verwendeten Mittel bedingte Nachteile auf, worauf am Ende des Kapitels eingegangen werden soll. Ambition dieser Arbeit ist nicht die Erstellung eines marktfähigen Plugins, sondern eines Prototyps, der anschaulich und praktisch die in Abschnitt 4.1 etablierte Signalverarbeitungskette implementiert und dabei Vorschläge hinsichtlich Implementierung und Design gibt. Daher werden die jeweiligen Limitationen der verwendeten Entwicklungsumgebungen sowie funktional unkritische Fehler der Implementierung in Kauf genommen.

Entsprechend der Definition von »Prototyping« nach Lackes (2018) stellt der »MHV-Panner« im derzeitigen Stadium eine erste Version dar, die die Testung und Weiterentwicklung sowie weitergehende Forschungen – in diesem speziellen Fall im Feld der ausgedehnten Schallquellen – ermöglichen.

5.1 Tools

Im Folgenden soll kurz auf die verwendeten Tools und die Entwicklungsumgebung eingegangen werden. Hauptaugenmerk bei der Wahl der Tools war die einfache Bedienbarkeit im Entwicklungsprozess mit überschaubarer Lernkurve, gute Nachvollziehbarkeit der Ergebnisse, eine gute Dokumentation sowie eine hohe Klangqualität der das Audiosignal verarbeitenden Elemente.

Max

Max ist eine von Cycling74 entwickelte Programmierungsumgebung in der aktuellen Version 8.2¹ (vgl. Cycling74 2021). Sie bietet in modularer Weise Logik- und Signalverarbeitungsbausteine für Audio und Video in Form von Max-Objekten an, die node-artig verbunden werden können und teilweise bereits eine Benutzeroberfläche aufweisen. Organisiert werden die Bausteine innerhalb von [Patches](#), die wiederum in Projekten organisiert werden können. Neben dem proprietären [Patch](#)-Format können Medien, Skripte und andere Dateien eingebunden werden. Zusätzlich gibt es Externals, die als Max-Objekte eingebunden werden und die Funktionalität der Anwendung erweitern können.

Die Vorteile von Max für das Prototyping von Audiosoftware sind die gute Optimierung auf Echtzeitaudiosignalverarbeitung, die relativ niedrige Lernkurve im Vergleich zu anderen rein Code-basierten Programmierungsumgebungen, die signalverarbeitungstechnische Anschaulichkeit der Ergebnisse sowie die einfache Möglichkeit rudimentäre Benutzeroberflächen zu erstellen.

Spat

Spat ist ein vom IRCAM-Institut entwickeltes Software-Paket mit Max-Externals in der aktuellen Version 5.2.4.², welches der Echtzeit-Spatialisierung von Audio-

¹Während der Entwicklung des MHV-Panners wurde Version 8.1.11 verwendet.

²Während der Entwicklung des MHV-Panners wurde Version 5.2.1 verwendet.

signalen dient (vgl. IRCAM 2021). Es bietet eine umfangreiche Sammlung an ausführlich dokumentierten Audio- und OSC-Externals.

Live

Live ist eine von Ableton entwickelte DAW (Digital Audio Workstation) in der aktuellen Version 11.0.11 (vgl. Ableton 2021). Mit Versionsnummer 8 wurde 2009 die Max-Anbindung [Max For Live](#) eingeführt (vgl. AudioNewsRoom 2009), mithilfe derer in Max entwickelte [Patches](#) in Form eines Max for Live Devices exportiert und in Live als Plugin genutzt werden können.

5.2 Signalverarbeitungskette

Die implementierte Signalverarbeitungskette orientiert sich in großem Maße an dem in 4.1 etablierten Konzept. Zusätzlich wird hier der Binauralisierungsblock ergänzt, um einerseits dem [Nutzer*](#) die Kopfhörerwiedergabe zu ermöglichen und andererseits den Entwicklungsprozess einfacher und flexibler zu gestalten. Für die Binauralisierung der Lautsprecher-signale wird das `spat5.virtualspeakers~`-Objekt in einer Konfiguration ohne Bodenreflexionssimulation und mit der KEMAR-Kunstkopf-HRFT verwendet. Der Binauralisierungsblock kann DSP-seitig deaktiviert und die Lautsprecher-signale direkt abgegriffen werden. Das Amplitudenpanning wird mit dem `spat5.pan~`-Objekt realisiert. Dieses wird mit einem Source-Spread von 0 parametrisiert, was einem reinen [VBAP](#) im Gegensatz zum [MDAP](#) entspricht. Der Panningalgorithmus ist umschaltbar zwischen `vbap3d`, `vbip3d` und `lbap`. `Lbap` realisiert dabei jeweils ein zweidimensionales Amplitudenpanning auf den beiden horizontalen Layern des Lautsprechersystems und überblendet bei Schallquellenpositionen zwischen den Layern. Phantomlautsprecher werden optional schaltbar über den Modus `auto` realisiert. Im Falle des verwendeten Lautsprechersystems [4+7+0](#) nach Empfehlung der ITU (2018) wird dabei automatisch ein Lautsprecher im Zenit sowie ein weiterer im Nadir positioniert. Die Decodierung der Mikrofon-signale nach Blockschaltbild 4.2

wird mithilfe von einfacher Subtraktion und Summierung der Signale in eigener Implementierung realisiert.

Die komplette implementierte Signalverarbeitungskette ist in Abbildung 5.1 zu sehen. Das praktisch verwendete und im Tool implementierte Lautsprechersetup ist in Abbildung 5.2 zu sehen. Dieses orientiert sich an den Empfehlungen der ITU (vgl. Abbildungen 4.4 und 4.5).

5.3 Parametrisierung

Tabelle 5.1 referenziert die in Tabelle 4.1 etablierten relevanten Parameter mit ihrem Max- und Live-spezifischen Parameterautomationsnamen, dem verwendeten Max-Objektyp und dem sichtbaren Label in der Benutzeroberfläche.

5.4 Benutzeroberfläche

Für die Benutzeroberfläche kommt primär eine Kombination aus nativen Max-Objekten mit ihrer jeweiligen eigenen grafischen Benutzeroberfläche zum Einsatz. Um zusätzlich auch in Verbindung mit Ableton Live ein kohärentes Aussehen und Verhalten zu erreichen, werden primär `live.`-Objekte eingesetzt:

- `live.numbox` für Parameter mit kontinuierlichen Zahlenwerten.
- `live.text` für Parameter mit alternierenden Werten.
- `live.menu` für Parameter mit mehreren möglichen Textwerten.
- `live.gain` für Lautstärkeparameter mit zusätzlichem Metering von Levels.

Die Visualisierung der nativen und äquivalenten Richtcharakteristika der Mikrofone erfolgt mithilfe eines speziell für diesen Zweck erstellten Javascripts innerhalb eines `jsui`-Objekts (Abbildung 5.4). Dieses Skript implementiert die Grafiken 3.3 interaktiv anhand der Formel 3.11 und ist interaktiv abhängig von den Gewichtungsfaktoren a , b_{MH} und b_{MH} als Eingabe aus der Benutzeroberfläche.

Für die Visualisierung der Panning- und Lautsprecherpositionen kommt das

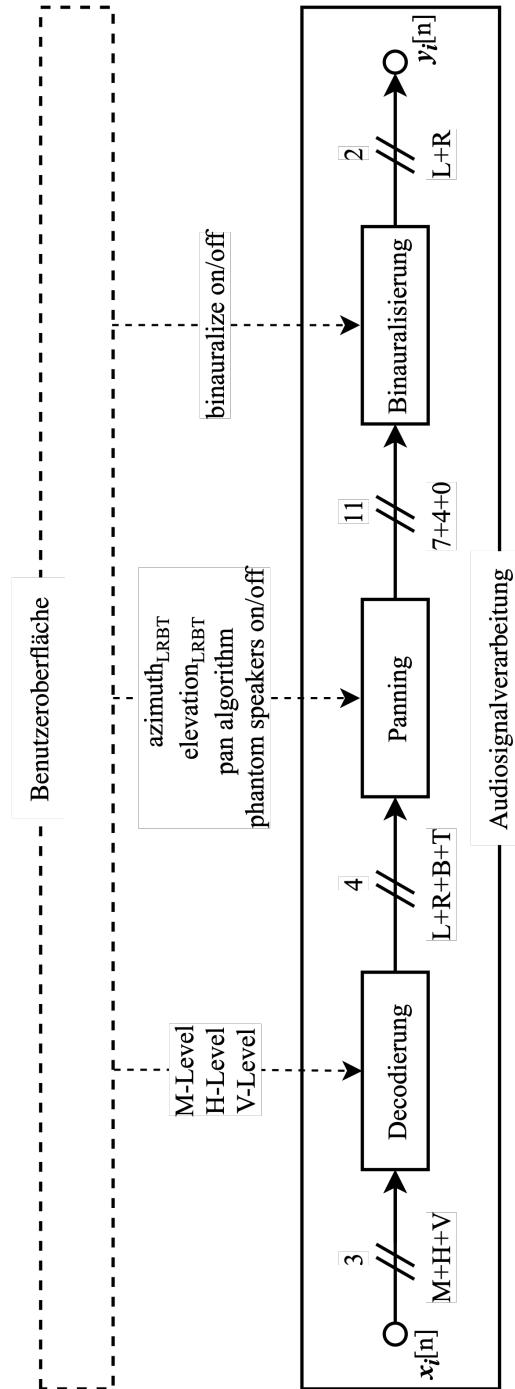


Abbildung 5.1: Blockschaltbild der gesamten Implementierung der Signalverarbeitungskette

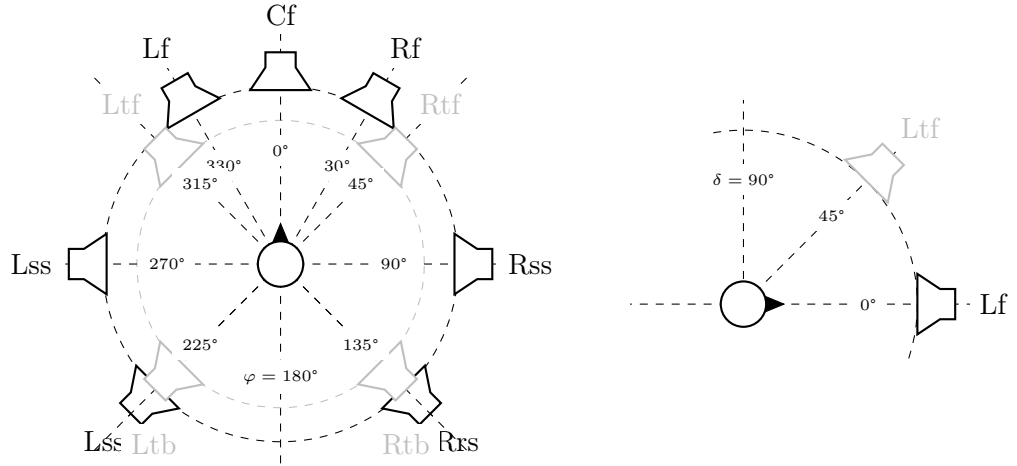


Abbildung 5.2: Das 4+7+0-Lautsprechersystem in Draufsicht (links) mit horizontalem Azimutwinkel φ und seitlichem Querschnitt (rechts) mit vertikalem Elevationswinkel δ , wobei die vier Lautsprecher im **Upper Layer** (grau) eine Elevation von $\delta = 45^\circ$ aufweisen

`spat5.viewer.embedded`-Objekt zum Einsatz, das in Aussehen, Zoom und Farbcodierung der sonstigen Benutzeroberfläche angepasst ist. Die Performance und grafische Auflösung ist aufgrund der Nutzung innerhalb der rein CPU-basierten Max-Oberfläche zwar recht beschränkt, jedoch erfüllt das Objekt seinen Zweck der Visualisierung innerhalb dieses Prototyps ausreichend.

Für das Gesamtlayout der Benutzeroberfläche (Abbildung 5.3) bietet Ableton Live Vorgaben, um zu gewährleisten, dass die Benutzeroberfläche des jeweiligen **M4L**-Devices innerhalb der Live-Oberfläche sichtbar und nutzbar ist. So ist die Höhe des Devices auf 169 Pixel festgelegt. Die Breite ist frei wählbar und bei entsprechender Implementierung auch für den **Nutzer*** anpassbar. Der MHV-Panner macht von dieser Möglichkeit Gebrauch, indem er den zusätzlichen **M4L**-spezifischen Routingbereich des Devices ein- und ausblendbar macht (Abbildung 5.6). Dieser Bereich ist aufgrund der speziellen Bedürfnisse der Live-

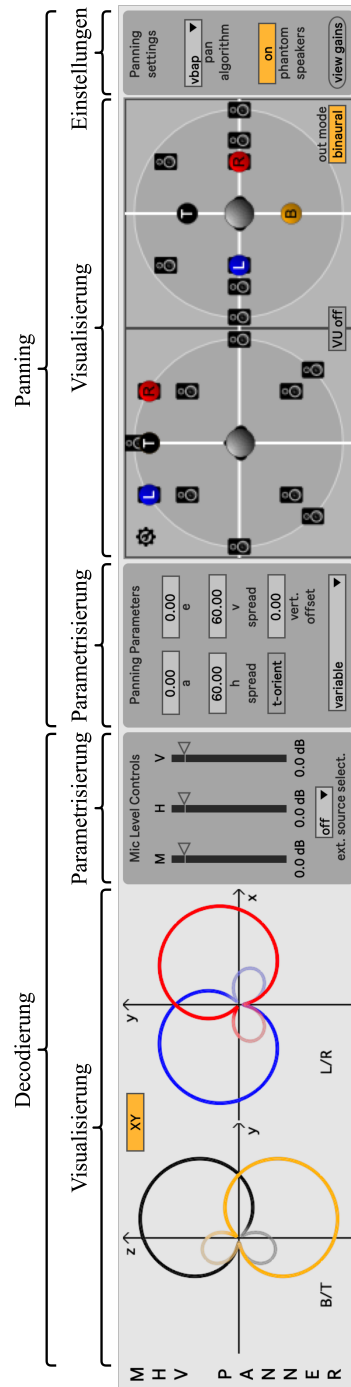


Abbildung 5.3: Benutzeroberfläche: Grafische Aufteilung in Decodierung und Panning

Name	Automationsname	Label	Objekttyp
M-Level	gain_mlevel	M	live.gain
H-Level	gain_hlevel	H	live.gain
V-Level	gain_vlevel	V	live.gain
azimuth	numbox_azimuth	a	live.numbox
elevation	numbox_elevation	e	live.numbox
hor. spread	numbox_hspread	h spread	live.numbox
vert. spread	numbox_vspread	v spread	live.numbox
vert. offset	numbox_voffset	vert. offset	live.numbox
orientation	button_orientation	x-orient/t-orient	live.text
fixed preset	numbox_indexpos	position	live.numbox
pan algorithm	menu_panmode	vbap, vbip, lbap	live.menu
phantom speakers	button_phantomspeaker	none/yes	live.text
Plot-View	button_plotmode	XY/MS	live.text
Input-Selector	menu_inputselect	ext. source select	live.menu

Tabelle 5.1: Parameter der grafischen Benutzeroberfläche der Implementierung in
Bezug auf Tabelle 4.1

Implementierung in Oberfläche und Logik dem nativen [M4L](#)-Surroundpanner entnommen.

Ebenfalls ist ein zusätzliches Fenster implementiert, das die Lautsprechergains in Interaktion mit der sonstigen Parametrisierung visualisiert (Abbildung 5.7). Dieses kann der [Nutzer*](#) über einen dedizierten Button in der Benutzeroberfläche öffnen.

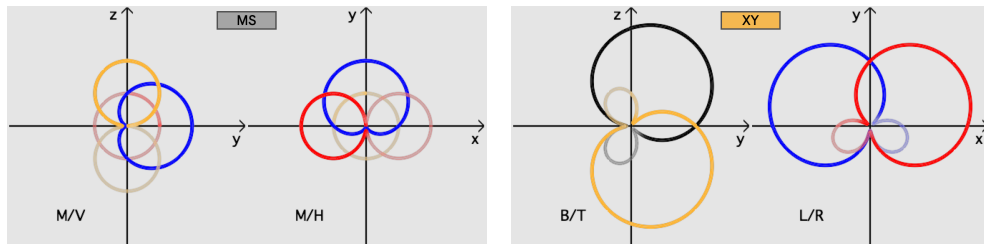


Abbildung 5.4: Benutzeroberfläche: Umschaltbare Visualisierung der nativen (links) und äquivalenten (rechts) Richtcharakteristika

5.5 Entwicklungskonzepte

Bei der Implementierung des »MHV-Panners« kamen einige Entwicklungskonzepte zum Einsatz, die im Folgenden erläutert werden sollen.

5.5.1 Modularisierung

Bei der Implementierung wurde ein möglichst modularer Aufbau angestrebt. So sind alle wichtigen und potenziell mehrfach verwendeten Blocks in eigene Patches oder Klassen aufgeteilt, um Wiederholungen in Code und Patch zu vermeiden, Anpassungen in der Implementierung leichter vornehmen zu können und die generelle Übersichtlichkeit und Nachvollziehbarkeit zu erhöhen. Insbesondere die Trennung von Benutzeroberfläche und Backend ist im Programm Max herausfordernd, da dies konzeptionell die Grenze zwischen Front- und Backend stark verschwimmen lässt. Die Patches weisen selbst – insbesondere durch den Präsentationsmodus mit reduziertem Layout – bereits eine Form von Frontend auf. Auch um ein allgemeines Konzept direkt umzusetzen, wie das in Kapitel 4 vorgestellte, ist eine klare Trennung sinnvoll. Diese Trennung macht außerdem perspektivisch den Wechsel zu einem auf einem anderen Framework basierenden alternativen Frontend einfacher, da dies nur die bestehenden Parameter (vgl. Abb. 4.1) an das Backend kommunizieren können muss. Daher ist hier die Aufteilung in unterschiedliche Patches sinnvoll.

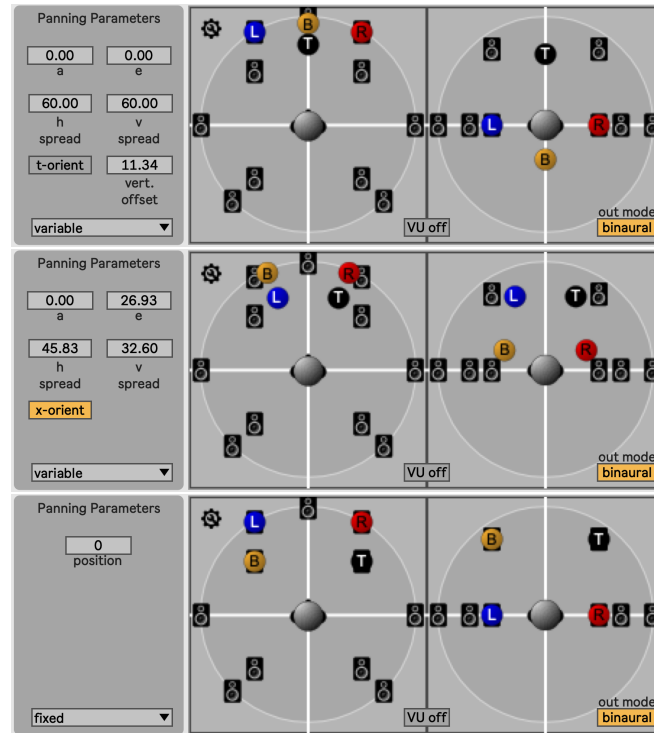


Abbildung 5.5: Benutzeroberfläche: Die drei Modi der Panningpositionsparameter

Abbildung 5.8 zeigt alle Max-Patches sowie die beiden Javascript-Skripte für die M4L-Version und die Standalone-Version.

MHV_RACK ist das Standalone-spezifische Containerpatch, das wiederum mehrere Instanzen des Frontend-Patches MHV_FE enthält und gleichzeitig als grafisches Frontend dient.

MHV_Panner_M4L ist das M4L-spezifische Containerpatch, das in Ableton als Plugin aufgerufen werden kann und wiederum das Frontendpatch MHV_FE enthält.

MHV_FE ist das Frontendpatch, das neben der Benutzeroberfläche und Parametersteuerung wiederum das Patch MHV_BE enthält.

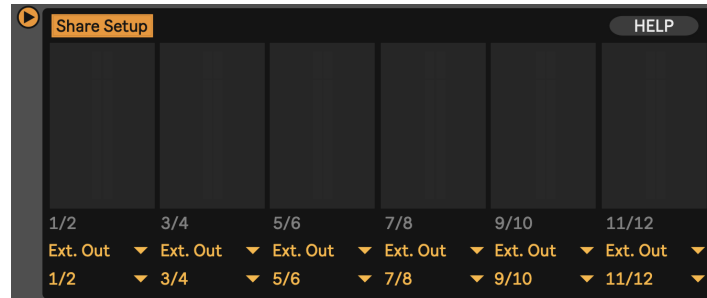


Abbildung 5.6: Der zusätzliche einklappbare Teil der Benutzeroberfläche in der M4L-Version für die Ausgangszuweisung

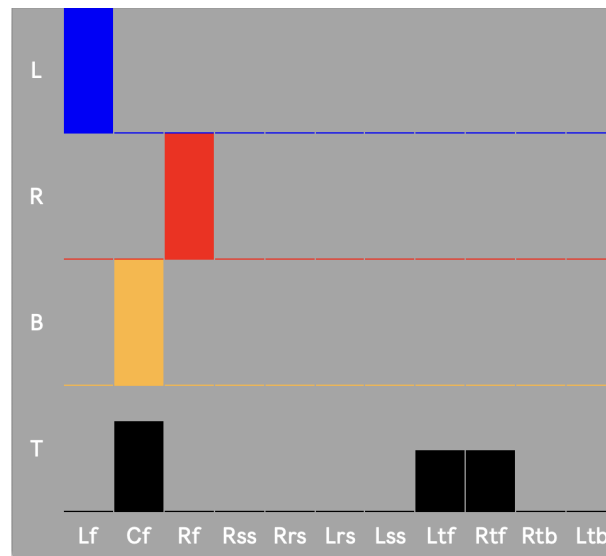


Abbildung 5.7: Benutzeroberfläche: Zusätzliches Fenster zur Visualisierung der Speaker-Gains

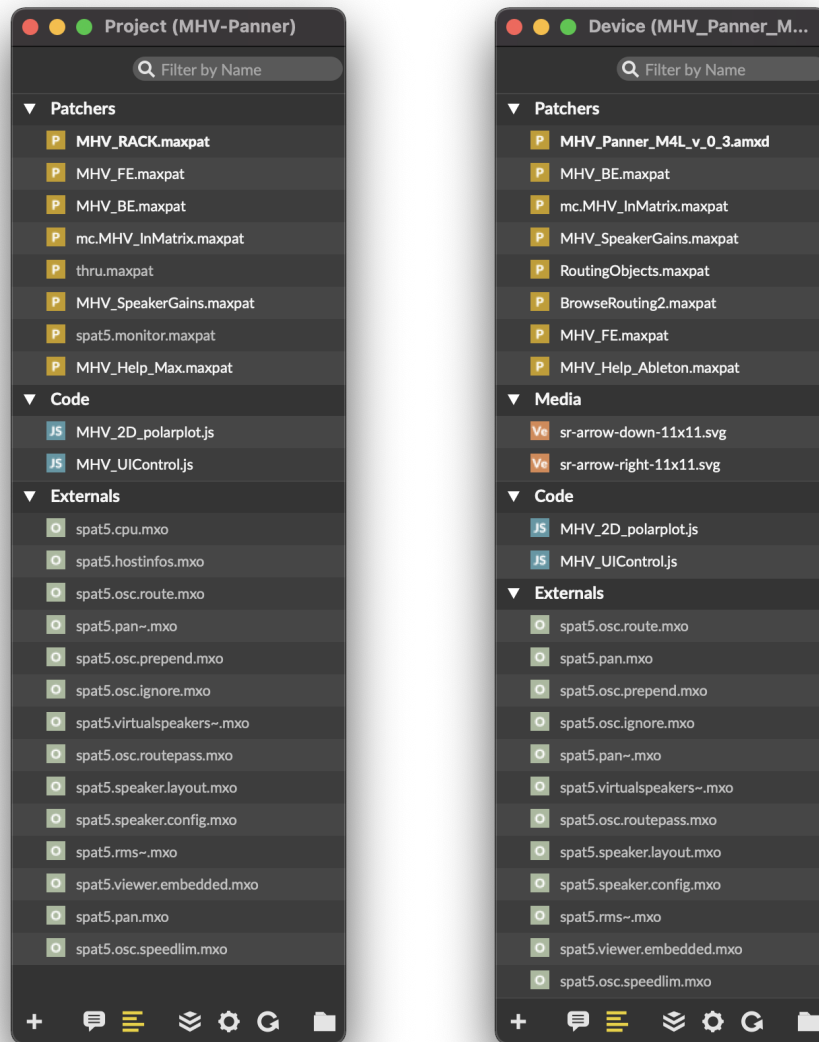


Abbildung 5.8: Max: Projektordner mit Patches, Javascripts und Externals von Standalone-Version (links) und M4L-Version (rechts)

MHV_BE ist das Backendpatch, das die komplette Audiosignalverarbeitung verwaltet.

mc.MHV_InMatrix ist das Patch, welches die Decodierung der Mikrofonssignale in zwei Stereosignalspaare vornimmt.

MHV_SpeakerGains ist ein zusätzliches Frontend-Patch, das die Speaker-Gains in einem Schwebefenster visualisiert.

MHV_Help_Max ist das Standalone-spezifische Hilfe-Patch, das aus der Benutzeroberfläche heraus geöffnet werden kann.

thru ist ein Patch für die Übersichtlichkeit innerhalb des Audio-Backends.

MHV_2D_polarplot ist das Javascript-Skript, das die Grafikbefehle für das JSUI-Objekt liefert.

MHV_UIControl ist das Javascript-Skript, das das Verhalten der Max-Objekte im Frontendpatch MHV_FE steuert.

sr-arrow sind zwei Bilddateien aus dem M4L-Surroundpanner für einen Button im Live-Look.

RoutingObjects ist ein Patch aus dem M4L-Surroundpanner für die Outputzuweisung.

BrowseRouting2 ist ein Patch aus dem M4L-Surroundpanner für die Outputzuweisung.

5.5.2 Dokumentation

Zur Übersicht und Nachvollziehbarkeit sind alle Bereiche der Entwicklung bestmöglich in sich dokumentiert. So gibt es Kommentare innerhalb des Max-Backends, die die Funktionen einzelner Abschnitte und Blöcke logisch und signalverarbeitungstechnisch dokumentieren (vgl. Abbildung 5.9). Die Benutzeroberfläche beinhaltet ebenfalls möglichst selbsterklärende Beschriftungen der Bedienelemente (vgl. Abbildung 5.3).

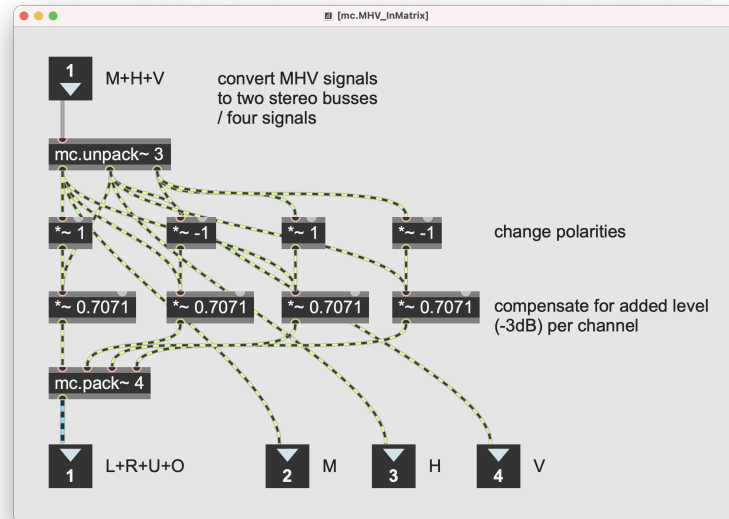


Abbildung 5.9: Max: Beispielhafter Ausschnitt der Decodermatrix mit Kommentierung

5.5.3 Skriptsteuerung

Für die Steuerung der Benutzeroberfläche wird ein Javascript mithilfe des `js-` Objekts genutzt. Dieses steuert vornehmlich die Sichtbarkeit verschiedener UI-Elemente. Zusätzlich übernimmt es die Berechnung der Positionen der vier virtuellen Schallquellen aus den in der Benutzeroberfläche angezeigten Werten und sendet diese weiter an das Audio-Backend.

Besonders für die Berechnung und Zuweisung der Quellpositionen aus den Lautsprecherpositionen – entsprechend dem Konzept »Discrete« nach Abschnitt 5.5 – ist die Code-basierte Berechnung deutlich übersichtlicher. So sind im `MHV_UIControl`-Skript die Lautsprecherpositionen im Format [Azimut, Elevation, Distance] in einem Array hinterlegt:

```
var speakerpos = [[-30., 0., 1.], [0., 0., 1.], ... ];
```

Ebenso sind die möglichen Kombinationen des diskreten Routings in einem Array hinterlegt und können als Presets abgerufen werden:

```
var speakercomb = [[0, 2, 7, 8], [0, 7, 2, 8], ... ];
```

Auch die in Abschnitt 5.5 entwickelten Formeln für die Panningkonzepte »X-Orientation« und »T-Orientation« werden innerhalb des `MHV_UIControl`-Skripts berechnet. Die Azimut- und Elevationswinkel der vier Schallquellen werden dazu ebenfalls in einem Array hinterlegt. Zur Berechnung der variablen Positionen im Verhältnis zur imaginären Schallquelle I werden die aktuellen Parameterwerte berechnet und ins Array geschrieben. Hier beispielhaft die Berechnung für die Positionswerte von L entsprechend der Gleichung 4.2:

```
sourceposvar[0][0] = a - hspread/2;
sourceposvar[0][1] = e;
...
```

Diese vom jeweiligen Panningmodus abhängigen unterschiedlichen Rechengänge und Zwischenspeicherungen von Werten wären zwar in Max ebenfalls möglich, jedoch deutlich unübersichtlicher und komplexer, weshalb diese in das Javascript ausgelagert sind.

5.5.4 Kommunikation über OSC

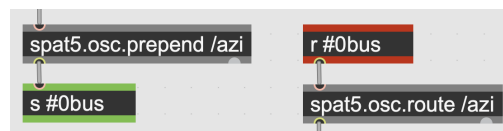


Abbildung 5.10: Max: OSC-Logik mit `spat5.osc.prepend` und `spat5.osc.route`

Für die Kommunikation zwischen den Max-Objekten und verschiedenen Patch-Instanzen – insbesondere zwischen Frontend und Backend – werden die Konventionen von OSC (OpenSoundControl 2021) verwendet, die in Max über die

entsprechenden Spat-OSC-Externals nutzbar werden. OSC ist ein offener Kommunikationsstandard, der insbesondere in Netzwerkkommunikationsstrukturen genutzt wird und ein hohes Maß an Flexibilität und Simplizität aufweist (OpenSoundControl 2021). Statt eines Netzwerkprotokolls wird hier die proprietäre Send-Receive-Struktur von Max genutzt – eine Kommunikation über serielle Ports oder Netzwerkschnittstellen wäre jedoch einfach möglich. So wird lediglich ein universeller Nachrichten-Send-Receive-Bus genutzt, auf den alle Nachrichten mit einem entsprechenden Präfix über das Objekt `spat5.osc.prepend` geschickt werden. Analog kann dieser Bus an beliebiger Stelle in den Patches mithilfe des Max-Objekts `spat5.osc.route` abgegriffen werden. Abbildung 5.10 zeigt beispielhaft ein solches Konstrukt mit dem Azimut-Parameter, der von einem UI-Element ausgegeben wird und mit dem Präfix `/azi` versehen wird und schließlich durch Filterung nach dem Präfix wieder abgegriffen wird. Dieser Aufbau ist im Besonderen im Debugging von Vorteil, da alle Nachrichten über den Bus abgegriffen und direkt zugeordnet werden können. Auch im Sinne der universellen Erweiterbarkeit ist dieser Aufbau vorteilhaft. So könnte auch externe Software oder Hardware – beispielsweise ein Headtracker – mit wenig Implementierungsaufwand nativ mit dem MHV-Panner-Backend kommunizieren. Nicht zuletzt fiel wegen der nativen Kommunikation der `spat5`-Audio-Objekte per OSC die Wahl auf diesen Kommunikationsstandard.

Für den Fall, dass mehrere Instanzen des MHV-Panners gleichzeitig geöffnet sind, wie beispielsweise im Plugin- oder Rack-Betrieb, müssen die Kommunikationswege der globalen Busse getrennt werden. Dazu wird eine Max-spezifische Funktion genutzt, die jeder Patch-Instanz einen Identifier übergibt. So ist der wird den `send`- und `receive`-Objekten ein zusätzliches `#0` im Namen hinzugefügt, das bei der Instanziierung in einen einzigartigen numerischen Identifier gewandelt wird (vgl. Abbildung 5.11). So ist eine unabhängige Kommunikation innerhalb der Instanzen möglich.



Abbildung 5.11: Max: Paar aus **send**- und **receive**-Objekt und Javascript zur UI-Steuerung mit Patch-Instanz-Identifizier

5.5.5 Namenskonventionen

Als Namenskonvention wird der sogenannte Camel Case genutzt. Zwar gibt es seitens Max keine funktionale Notwendigkeit dafür, jedoch der Einheitlichkeit und Übersichtlichkeit halber folgt die Benennung von Objekten, Sends und OSC-Adressen diesem Prinzip. Eine Ausnahme stellt die Benennung der UI-Elemente dar, die dem Schema `Typ_Parameter` folgt, um im Falle des M4L-Devices eine leichtere Lesbarkeit der Automationsparameter innerhalb der DAW zu erreichen.

5.5.6 Farbcodierung

Sowohl innerhalb des Max-Backends als auch in der Benutzeroberfläche wird eine einheitliche Farbcodierung etabliert. Seitens der Benutzeroberfläche werden die Visualisierungen der äquivalenten Richtcharakteristika, die visualisierten Positionen der entsprechenden virtuellen Schallquellen sowie die Visualisierung der Lautsprechergegainfaktoren einheitlich farbcodiert. Hier entspricht *L* stets *blau*, *R* stets *rot*, *B* stets *gelb* und *T* stets *schwarz*. Außerdem ist eine M4L-spezifische Farbfunktion integriert, um eine gute Lesbarkeit sowie ein einheitliches Aussehen in den verschiedenen Live-Looks mit variierenden Farbpaletten zu gewährleisten. So werden einige Elemente abhängig von der jeweils gewählten Live-Farbpalette mithilfe des `live.colors`-Objekts eingefärbt. Zusätzlich werden in der Benutzeroberfläche einige `live.`-Objekte genutzt, deren Aussehen ebenfalls auf die Live-Farbpalette reagiert. Generell werden vornehmlich Farben aus der Live-Farbpalette genutzt, um einen kohärenten Look innerhalb von Live zu erzeugen. Innerhalb des Max-Backends wird zur besseren Übersicht in der Entwicklung sowie

für die bessere Übersichtlichkeit für Außenstehende ebenfalls eine Farbcodierung für verschiedene Objekttypen eingesetzt (Abbildung 5.12).

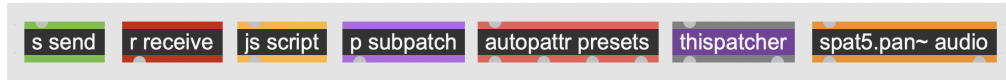


Abbildung 5.12: Die verwendete Farbcodierung nach Objekttyp innerhalb des Max-Backends anhand von beispielhaften Objekten

5.6 Subjektive Bewertung

Nach einem ersten subjektiven Test anhand der von Hofmann (2020) produzierten Aufnahmen aus einem reflexionsarmen Raum, lassen sich einige frühe Resümees ziehen. Eine ausführliche Evaluation in Form einer Usability-Studie oder eines Hörvergleichs ist im Rahmen dieser Arbeit nicht vorgesehen, kann jedoch der nächste konsequente Schritt in nachfolgenden Forschungen sein.

So ist die Bearbeitung der Aufnahmen der [MHV](#)-Stützmikrofonanordnung deutlich intuitiver geworden, da die decodierten Signale zusammengefasst zu einer imaginären Schallquelle bearbeitet werden können und keine einzelne Parametrisierung und Positionierung der Signale mehr notwendig ist. Die Effekte und der Mehrgewinn der ausgedehnten Abbildung der Schallquellen sind sowohl in der binauralen Kopfhörerwiedergabe als auch in der Lautsprecherwiedergabe klar zu erkennen, wobei sie, wie zu erwarten, in der Vertikalen deutlich subtiler ausfallen. Interessant sind die unterschiedlichen Effekte der beiden intendierten Möglichkeiten die Ausdehnung zu beeinflussen: die Gewichtung des Mikrofonsignals und die räumliche Verteilung der decodierten Signale. Die Visualisierung der Speakergains lässt Übersprechen zwischen Kanälen auf den gleichen Lautsprecher leicht erkennen und zum Teil die wahrgenommene Ausdehnung vorhersagen. Ebenfalls wie zu erwarten wirkt sich ein [MDAP](#)-Spread der Schallquellen negativ auf die wahrgenommene Ausdehnung aus, weshalb er in der Benutzeroberfläche nicht implementiert wurde. Ebenfalls positiv tragen der LBAP-Algorithmus sowie

die zuschaltbaren Phantomlautsprecher zur Wahrnehmung von Ausdehnung bei. Interessant ist der Effekt der verschiedenen Drehungen und Anordnungen der Signale um die imaginäre Schallquelle, da hier auch unrealistische Anordnungen klangästhetisch ansprechende Resultate hervorbringen.

Die Implementierung des Tools leidet teilweise unter den Beschränkungen der verwendeten Mittel. So ist die Integration in einen Postproduktions-Workflow noch nicht gänzlich möglich. Die Standalone-Variante erschwert das Processing der gepannten Signale, da die Summierung der gepannten Signale außerhalb der jeweiligen DAW stattfindet. Gleichmaßen nähert sich die M4L-Variante einem integrierten Workflow zwar an, kann jedoch durch die mangelnde Eignung von Ableton Live für Mehrkanalaudio nicht ihr ganzes Potenzial entfalten. So ist eine Insertierung von anderen Plugins vor oder nach dem MHV-Panner nicht sinnvoll möglich, da der MHV-Panner nicht (nur) den Stereobus des jeweiligen Kanalszugs nutzt, sondern zusätzliche Audiowege unabhängig von der DAW-Infrastruktur nutzen muss. Hier drängt sich die Notwendigkeit der Integration des MHV-Panners in andere mehrkanalfähigen DAWs wie Cockos Reaper auf, was jedoch nur mit der Implementierung in ein gängiges Pluginformat wie VST möglich ist. Auch hier besteht ein Potenzial für weitere Forschungen und Entwicklungen.

6 Fazit

6.1 Zusammenfassung

Im Rahmen der vorliegenden Arbeit wurde ein Postproduktionstool für eine dreikanalig-koinzidente Stützmikrofonanordnung – im Speziellen die MHV-Anordnung nach Hofmann (2020) – konzipiert. Das Konzept wurde in Form des »MHV-Panners« prototypisch implementiert und dokumentiert.

Für die Decodierung wurde eine Stereo-MS-Matrix für zwei Signalaare aus M und H sowie M und V erarbeitet, die die vier XY-äquivalenten Signale L und R sowie B und T ausgibt.

Für die Platzierung bzw. das Panning der vier aus der Decodierung resultierenden Signale als virtuelle Schallquellen wurden drei Parametrisierungskonzepte für das Amplitudenpanning entwickelt. Das Konzept »T-Orientatation« positioniert die vier Schallquellen um eine imaginäre Schallquellenposition herum, wobei hier das vertikale Stereopaar aus B und T die vertikale Ausdehnung beschreiben kann und das horizontale Stereopaar aus L und R die horizontale Ausdehnung. Das Konzept »X-Orientatation« dreht die Anordnung »T-Orientatation« um 45° und entfernt sich von der ursprünglichen Aufstellung der Mikrofonanordnung. Das Konzept »Discrete« beschränkt sich auf ein diskretes Panning der Einzelsignale auf die Lautsprecher.

Es wurden zudem Konzepte für eine Benutzeroberfläche erarbeitet, um die Decodierung sowie das Panning angemessen zu bedienen und zu visualisieren. Für die Visualisierung der Decodierung wurde auf die Darstellung der zweidimensionalen Polardiagramme der nativen sowie XY-äquivalenten Richtcharakteristika der decodierten Mikrofonsignale zurückgegriffen. Für die Visualisierung des Pannings

wurde eine etablierte dreidimensionale Darstellung des Lautsprecher setups mit den Einzelsignalen als virtuelle Schallquellen herangezogen.

Die Implementierung des »MHV-Panners« wurde mithilfe der Programmierumgebung Max einerseits als Standonetool sowie andererseits als Max for Live Device für Ableton Live vorgenommen. Dazu wurden die zuvor entwickelten Signalverarbeitungskonzepte prototypisch implementiert. Ergänzt wurden diese mit einer Binauralisierung, um eine flexiblere Testumgebung zu schaffen. Die Implementierung wurde im Detail dokumentiert und begründet.

6.2 Ausblick

Sowohl das MS-basierte Decodierungskonzept als auch die Platzierung der resultierenden Signale durch Amplitudenpanning stellen Vorschläge dar. Möglich wäre auch eine zusätzliche oder alternative frequenzabhängige Decodierung, um die frequenzabhängige Wahrnehmung von Ausdehnung auszunutzen. Ebenfalls wäre das Panning mit zeitbasierten Verfahren oder auch Higher Order Ambisonics möglich und kann ein neues interessantes Forschungsfeld darstellen. Einen Vergleich der verschiedener Panning-Ansätze könnte im Rahmen von Hörtests evaluiert werden.

Die Implementierung des »MHV-Panners« ist ein erster Schritt, die Signale der MHV-Stützmikrofonanordnung in einem DAW-Workflow integriert und intuitiv bearbeitbar zu machen. Somit steht als nächster potenzieller Schritt in dieser Forschung der Beta-Test und eine Usability-Studie des Tools bevor, um besonders die Benutzeroberfläche aufgrund von Feedback weiter zu optimieren. Ebenso kann der »MHV-Panner« Teil einer Hörevaluation sein, um Wert und Effekt von Ausdehnungsabbildung zu erforschen und zu quantifizieren – sowohl für Nutzer*innen als auch Hörer*innen. Dabei gilt es außerdem den Einfluss verschiedener Lautsprecher setups sowie auch der binauralen Wiedergabe und in diesem Rahmen auch allgemein die Ausdehnungsabbildung in der binauralen Kopfhörerwiedergabe mit und ohne Headtracking zu erforschen.

Die präsentierte Implementierung des erarbeiteten Konzepts in Form des »MHV-

Panners« weist sowohl grafisch als auch infrastrukturell einige Beschränkungen auf. Besonders unter dem Aspekt einer eventuellen professionellen oder kommerziellen Verwendung des Tools, besteht hier jedoch ein großes Potenzial das erarbeitete Konzept plattformübergreifend in den gängigen Pluginformaten wie VST, AU und AAX zu implementieren. Insbesondere die mögliche Verwendung innerhalb von [DAWs](#), die für Mehrkanalaudio optimiert sind, wäre ein großer Gewinn. Zu diesem Zweck hat diese Arbeit die Konzepte zunächst unabhängig von der Implementierung präsentiert.

Das Forschungsfeld dieser Arbeit ist in vielen Aspekten bewusst eng gesteckt. Den Ausgangspunkt bietet die von Hofmann (2020) entwickelte [MHV-Stützmikrofonanordnung](#). Das Zielformat der Implementierung ist ein kanalbasiertes [4+7+0](#)-Lautsprecheresetup bzw. die Binauralwiedergabe. Die verwendete Plattform ist Cycling74 Max bzw. Ableton Live. Das Aufnahmeprinzip der [MHV-Stützmikrofonanordnung](#) ist dabei rein koinzident – also pegelbasiert, ebenso ist die Decodierung der Signale eine reine Summierung bzw. Subtraktion. Die Platzierung der Signale findet durch reines Amplitudenpanning statt, das auf dem Prinzip der Summenlokalisation basiert. Trotzdem bieten die Ergebnisse Anknüpfungspunkte für verwandte Szenarien und Forschungsfragen und müssen nicht in einem Vakuum existieren.

Die Notwendigkeit für die Entwicklung neuer Tools für die Erstellung immersiver mehrdimensionaler Audioproduktionen besteht insbesondere in der Hinsicht, dass Streamingservices wie Apple Music den Konsument*innen nun immersive Audioproduktionen in binauraler Wiedergabe mit optionalem Headtracking anbieten. Dies bestärkt die Relevanz bestehende und etablierte Stereo-Aufnahmetechniken in immersive Workflows zu integrieren und zu erweitern. Die [MHV-Stützmikrofonanordnung](#) stellt hier einen aufnahmeseitigen Ansatz dar, eine Stereomikrofonie zu erweitern. Es ist auch vorstellbar den »MHV-Panner« in aktueller oder angepasster Form für den Upmix von bestehenden MS-, XY- oder auch laufzeitbasierten Mikrofonaufnahmen oder anderer stereofoner Klangquellen in immersive Formate zu erproben.

Abschließend sei festzuhalten, dass im Rahmen dieser Arbeit ein Konzept

sowie eine prototypische Implementierung eines Postproduktionstools für ein dreikanalig-koinzidentes Stützmikrofonverfahren entwickelt wurde. Es wurden Erfolge, Potenziale und Optimierungsmöglichkeiten aufgezeigt. Wie auch schon die vorangegangene Arbeit Hofmanns ([2020](#)) kann die vorliegende Arbeit einen Ausgangspunkt für weitere Forschungen darstellen.

Literatur

- Ableton (2021). *Live 11 Release Notes / Ableton*. Ableton. URL: <https://www.ableton.com/de/release-notes/live-11/> (besucht am 23.10.2021).
- AudioNewsRoom (2009). *Winter Namm 2009: A Few Picks / AudioNewsRoom - ANR*. AudioNewsRoom -ANR. URL: <https://audionewsroom.net/2009/01/winter-namm-2009-a-few-picks.html> (besucht am 23.10.2021).
- Benjamin, Eric und Thomas Chen, Hrsg. (2005). *The Native B-Format Microphone*. J. Audio Eng. Soc. URL: <http://www.aes.org/e-lib/browse.cfm?elib=13348> (besucht am 18.06.2021).
- Blauert, Jens und Jonas Braasch (2008). „Räumliches Hören“. In: *Handbuch Der Audiotechnik*. Hrsg. von Stefan Weinzierl. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 87–121. ISBN: 978-3-540-34301-1. DOI: [10.1007/978-3-540-34301-1_3](https://doi.org/10.1007/978-3-540-34301-1_3). URL: https://doi.org/10.1007/978-3-540-34301-1_3 (besucht am 13.06.2021).
- Cousins, Michael, Stefan Bleeck und Filippo Fazi (2017). *The Effect of Inter-Channel Cross-Correlation Coefficient on Perceived Diffuseness*. URL: https://www.researchgate.net/publication/320619455_The_effect_of_inter-channel_cross-correlation_coefficient_on_perceived_diffuseness (besucht am 13.10.2021).
- Cycling74 (2021). *Cycling '74 Website*. Cycling '74 Max. URL: <https://cycling74.com/> (besucht am 23.10.2021).
- Dickreiter, Michael u. a. (2014). *Handbuch der tonstudiotechnik*. 8., überarbeitete und erweiterte Auflage. De Gruyter Saur reference. Berlin: De Gruyter Saur. 2 S. ISBN: 978-3-11-028978-7. URL: <https://doi.org/10.1515/9783110316506> (besucht am 18.06.2021).

- Frank, Matthias (2013). „Phantom Sources Using Multiple Loudspeakers in the Horizontal Plane“. University of Music and Performing Arts Graz (Austria).
- Görne, Thomas (2015). *Tontechnik: Hören, Schallwandler, Impulsantwort Und Faltung, Digitale Signale, Mehrkanaltechnik, Tontechnische Praxis*. 4., aktualisierte Auflage. Medien. München: Hanser. 384 S. ISBN: 978-3-446-44149-1.
- Gribben, Christopher und Hyunkook Lee (2017). „A Comparison between Horizontal and Vertical Interchannel Decorrelation“. In: *Applied Sciences* 7, S. 1202. DOI: [10.3390/app7111202](https://doi.org/10.3390/app7111202).
- Hau, Dr. Andreas (2019). *Rode NT-SF1 - Ambisonics-Mikrofon im Test*. SOUND & RECORDING. URL: <https://www.soundandrecording.de/equipment/rode-nt-sf1-%e2%88%92-ambisonics-mikrofon-im-test/> (besucht am 31.10.2021).
- Hautz, Christian (2021). *Schoeps stellt das ORTF-2plus2 Stützmikrofon für 3D-Audio vor*. bonedo.de - Das Musikerportal. URL: <https://www.bonedo.de/artikel/einzelansicht/schoeps-stellt-das-ortf-2plus2-stuetzmikrofon-fuer-3d-audio-vor.html> (besucht am 31.10.2021).
- Hibbing, Manfred, Hrsg. (1989). *XY and MS Microphone Techniques in Comparison*. URL: <http://www.aes.org/e-lib/browse.cfm?elib=5883> (besucht am 18.06.2021).
- Hofmann, Leon (2020). „Konzeption Und Erprobung Eines Stützmikrofonverfahrens Zur Ausgedehnten Abbildung Akustischer Instrumente in Mehrdimensionalen Audiomischungen“. 136 S. URL: https://curdt.home.hdm-stuttgart.de/PDF/Hofmann_Leon.pdf (besucht am 01.05.2021).
- IRCAM (2021). *Spat - Ircam Forum*. Ircam Forum. URL: <https://forum.ircam.fr/projects/detail/spat/> (besucht am 23.10.2021).
- ITU (2018). *Recommendation ITU-R BS.2051-2: Advanced Sound System for Programme Production*. International Telecommunication Union, S. 22. URL: <https://www.itu.int/rec/R-REC-BS.2051-2-201807-I/en> (besucht am 13.06.2021).
- Lackes, Prof Dr Richard und Dr. Markus Siepermann (2018). *Definition: Prototyping*. <https://wirtschaftslexikon.gabler.de/definition/prototyping-44360>. URL:

- <https://wirtschaftslexikon.gabler.de/definition/prototyping-44360/version-267672> (besucht am 24.10.2021).
- Lee, Hyunkook (2021). „Multichannel 3D Microphone Arrays: A Review“. In: *Journal of the Audio Engineering Society. Audio Engineering Society* 69, S. 5–26. DOI: [10.17743/jaes.2020.0069](https://doi.org/10.17743/jaes.2020.0069).
- Nettingsmeier, Jörn (2012). *The Why and How of With-Height Surround Sound*. URL: https://www.researchgate.net/publication/308785677_The_Why_and_How_of_With-Height_Surround_Sound (besucht am 13.06.2021).
- OpenSoundControl (2021). *OSC Index*. OpenSoundControl. URL: <http://opensoundcontrol.org/> (besucht am 23.10.2021).
- Pulkki, Ville (1997). „Virtual Sound Source Positioning Using Vector Base Amplitude Panning“. In: *J. Audio Eng. Soc* 45.6, S. 456–466. URL: <http://www.aes.org/e-lib/browse.cfm?elib=7853>.
- (1999). „Uniform Spreading of Amplitude Panned Virtual Sources“. In: *Proceedings of the 1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. WASPAA'99 (Cat. No.99TH8452)*, S. 187–190. DOI: [10.1109/ASPAA.1999.810881](https://doi.org/10.1109/ASPAA.1999.810881). URL: <https://ieeexplore.ieee.org/document/810881> (besucht am 18.06.2021).
- (2000). „Generic Panning Tools for MAX/MSP“. In: .
- (2001). „Localization of Amplitude-Panned Virtual Sources II: Two- and Three-Dimensional Panning“. In: *J. Audio Eng. Soc* 49, S. 753–767.
- Reiss, Joshua D. und Andrew McPherson (2014). *Audio Effects : Theory, Implementation and Application*. Bosa Roca, UNITED STATES: Taylor & Francis Group. ISBN: 978-1-4665-6029-1. URL: <http://ebookcentral.proquest.com/lib/hdm-stuttgart/detail.action?docID=1661903>.
- Santala, Olli und Ville Pulkki (2011). „Directional Perception of Distributed Sound Sources“. In: *The Journal of the Acoustical Society of America* 129.3, S. 1522–1530. ISSN: 0001-4966. DOI: [10.1121/1.3533727](https://doi.org/10.1121/1.3533727). URL: <https://doi.org/10.1121/1.3533727> (besucht am 11.10.2021).
- Schoeps (2021). *Schoeps Double MS Plugin Manual*. URL: <https://schoeps.de/produkte/zubehoer/zubehoer-surround-3d/zubehoer-surround-3d>

doppel - ms / plugin - double - ms - tool / double - ms - plugin - 2018 .
html ? tx _ igproducts _ product % 5Burl % 5D = fileadmin % 2Fuser _
upload % 2Fuser _ upload % 2Fdownloads % 2FBedienungsanleitungen %
2FDoubleMSPluginUserGuide1 . 0 . 0 . pdf & tx _ igproducts _ product %
5Baction % 5D = downloadFile & tx _ igproducts _ product % 5Bcontroller %
5D = Product & cHash = 364bbe428e513c8f4a3a82c24fda4e8 (besucht am
31.10.2021).

Weinzierl, Stefan, Hrsg. (2008). *Handbuch Der Audiotechnik*. [Ausgabe in 2
Bänden]. Berlin; Heidelberg: Springer. ISBN: 978-3-540-34301-1.

Wittek, Helmut, Christopher Haut und Daniel Keinath (2006). „Doppel-MS –
Eine Surround- Aufnahmetechnik Unter Der Lupe“. In: *Tonmeistertagung*. URL:
https://hauptmikrofon.de/HW/TMT2006_Wittek_DoubleMS_neutral.pdf
(besucht am 09.05.2021).

Zotter, Franz und Matthias Frank (2019). *Ambisonics: A Practical 3D Audio
Theory for Recording, Studio Production, Sound Reinforcement, and Virtual
Reality*. Springer Topics in Signal Processing. Cham: Springer International
Publishing; Imprint: Springer. 210 S. ISBN: 978-3-030-17207-7. DOI: [10.1007/
978-3-030-17207-7](https://doi.org/10.1007/978-3-030-17207-7).

Digitaler Anhang

Der Arbeit liegt in einfacher Ausführung eine Daten-CD mit der digitalen Fassung dieser Arbeit als PDF bei.

In zweifacher Ausführung liegt ein USB-Stick mit folgendem Inhalt bei:

- Digitale Fassung dieser Arbeit als PDF
- Manuals, Papers und Snapshots von Internetseiten als PDF-Dateien
- Max-Development-Projekt-Ordner mit allen Patches und Externals und lauffähiger RACK-Standalone-Version des »MHV Panners«
- Ableton Live Demo Session mit Max for Live Device »MHV Panner« (beinhaltet Audiodateien von Leon Hofmann ([2020](#)))
- Reaper Demo Session für Verwendung mit Standaloneversion (beinhaltet Audiodateien von Leon Hofmann ([2020](#)))