

**Synthesizer auf Grundlage der Impulse Pattern
Formulation (IPF) - Den parametrischen Raum musikalisch
spielbar gestalten**

Bachelor-Thesis

Zur Erlangung des akademischen Grades B. Sc.

Vorgelegt von: Moritz Ben Lucius Arndt



Studiengang: Medientechnik

Erstgutachter: Prof. Dr. Robert Mores

Zweitgutachter: Dr. Simon Linke

Hamburg, 09. August 2023

Zusammenfassung

Die Impulse Pattern Formulation (IPF) ist eine Top-Down-Methode zur Modellierung beliebiger gekoppelter Systeme mittels exponentiell abklingender Impulsfolgen. Sie eröffnet faszinierende Möglichkeiten zur detaillierten Beschreibung und synthetischen Nachbildung des komplexen Verhaltens musikalischer Klangeigenschaften. Diese Arbeit nutzt die IPF als Grundlage für die Entwicklung eines Synthesizers und präsentiert eine visuelle Darstellung des parametrischen Raums, die eine musikalisch intuitive Spielbarkeit ermöglicht. Durch diese innovative Herangehensweise wird ein neuer Weg zur Erzeugung vielfältiger Klanglandschaften und zur intuitiven Steuerung musikalischer Ausdrucksformen eröffnet.

Abstract

The Impulse Pattern Formulation (IPF) is a top-down method for modeling arbitrary coupled systems using exponentially decaying pulse sequences. It offers fascinating possibilities for detailed description and synthetic replication of the complex behavior of musical sound characteristics. This work utilizes the IPF as the foundation for synthesizer development and presents a visual representation of the parametric space, enabling musically intuitive playability. Through this innovative approach, a new way is opened for generating diverse soundscapes and intuitively controlling musical expressions.

Inhaltsverzeichnis

1. Einleitung	5
2. Grundlagen.....	6
2.1. Impulse Pattern Formulation (IPF).....	6
2.2. Synthese- und Modulationsverfahren	10
2.3. Latenz und Roundtrip.....	12
3. Konzipierung.....	13
3.1. Zielsetzung und Anforderungen.....	14
3.2. Problemdarstellung	14
3.3. User Persona	15
3.4. Lösungskonzepte.....	16
3.4.1. Visuelle Darstellung der IPF als Plot.....	16
3.4.2. Visualisierung des zu erwartenden Verhaltens durch Farben	17
4. Stabilitätsanalyse	18
4.1 Grundlegende Annahmen.....	18
4.2 Regeln zur Bestimmung der Zustände	18
4.3 Auflösung und Wertebereiche.....	19
4.4 Speicherung der Daten	20
4.5 Struktur des Analyseprogramms.....	21
5. Realisation.....	22
5.1 Schnittstelle des Synthesizers	22
5.2 Framework	23
5.2.1 Struktur des Frameworks.....	23
5.2.2 Bedienoberfläche	24
5.2.3 Audioverarbeitung.....	29
6. Validierung	31
6.1 Roundtrip	31

6.2 Prüfen der Lösungskonzepte.....	33
6.2.1 Visualisierung im Plot	33
6.2.1 Farbliche Darstellung.....	34
6.3 Erkenntnisse	37
7. Ausblick.....	38
8. Fazit.....	39
Abbildungsverzeichnis	40
Quellenverzeichnis	41
Anhang	42

1. Einleitung

Im Rahmen dieser wissenschaftlichen Arbeit wird mit einem neuen Ansatz der Klangsynthese auf einem bislang wenig erforschten Weg – mithilfe der Impulse Pattern Formulation (IPF) - erkundet. Als eine nicht lineare Top-Down-Methode zur Modellierung von gekoppelten Systemen mittels exponentiell abklingender Impulsfolgen, bietet die IPF eine neuartige Möglichkeit, die komplexen Klangeigenschaften musikalischer Klänge detailliert zu beschreiben und nachzubilden. Diese Arbeit verfolgt das übergeordnete Ziel, die IPF als Grundlage für die Schaffung eines Synthesizers zu nutzen, der den parametrischen Raum auf eine Weise gestaltet, die nicht nur technisch funktional, sondern auch intuitiv und musikalisch spielbar ist.

Die zentrale Herausforderung dieser Forschungsarbeit besteht darin, die Komplexität der IPF in eine greifbare und intuitive, musikalische Realität zu verwandeln. Mit einem gezielten Fokus auf die Gestaltung und Kontrolle der Parameter, strebt diese Arbeit danach, die IPF zu nutzen, um ein Werkzeug zu schaffen, das Musiker:innen ermöglicht, Klänge mühelos zu formen und zu steuern. Hierbei stellt sich die Frage, wie die nicht linearen Zusammenhänge der IPF in eine nutzerfreundliche und ästhetisch ansprechende Benutzeroberfläche übersetzt werden können.

Um diese Forschungsfrage zu beantworten, wird ein klarer Ablauf verfolgt. Es beginnt mit einer gründlichen Einführung in die IPF, erklärt ihre Grundlagen und erläutert ihre Anwendung im Bereich der Klangsynthese. Im Anschluss werden verschiedene Synthese- und Modulationsverfahren vorgestellt, die in Kombination mit der IPF eine große Klangvielfalt ermöglichen. Dabei steht die Intuition bei der Steuerung dieser Verfahren im Vordergrund, um eine nahtlose Verbindung zwischen Musik und Technologie zu schaffen.

Ein zentraler Aspekt dieser Arbeit liegt in der praktischen Umsetzung der Theorie in einen funktionsfähigen Synthesizer. Die technischen Schritte zur Realisierung werden ebenso beleuchtet wie die Entwicklung einer benutzerfreundlichen Oberfläche, die es Musiker:innen ermöglicht, ohne umfangreiche Vorkenntnisse über die IPF mit dem Instrument zu interagieren. Die gewonnenen Erkenntnisse und Entwicklungen werden abschließend durch eine umfassende Validierung gestützt. Durch einen Roundtrip-Test wird die Performance des Synthesizers im Hinblick auf die zeitliche Verzögerung überprüft. Damit wird sichergestellt, dass der Synthesizer nicht nur theoretisch überzeugt, sondern auch in der Praxis reibungslos funktioniert.

Diese Arbeit schließt nicht nur eine Lücke zwischen mathematischer Theorie und musikalischer Praxis, sondern öffnet auch neue Möglichkeiten der Klangkreation und -gestaltung.

2. Grundlagen

2.1. Impulse Pattern Formulation (IPF)

Die von Bader vorgestellte Impulse Pattern Formulation (IPF) ist eine systematische nicht lineare Top-Down-Methode, also ein Ansatz bei dem zunächst die Hauptkomponente betrachtet wird um dann in Untergruppen aufzuteilen, um beliebige gekoppelte Systeme und deren Zustände zu beschreiben (Bader 2013, S. 285). Ausgangspunkt ist ein initialer Impuls, der von System-Komponente ausgeht, an verschiedenen Punkten reflektiert und exponentiell gedämpft wird. Dieser Impuls erzeugt durch die Reflexionen eine Impulsreihe, welche durch die Verknüpfung der aktuellen und vorherigen Zustände auf sich selbst, und nachfolgende Impulse und das gesamte System Einfluss nehmen. Die resultierenden Zustände des Systems können stabiles, konvergierendes, divergierendes oder chaotisches Verhalten, wie Bifurkationen, annehmen. Diese Betrachtungsweise von Systemen lässt sich auch auf das Verhalten physikalischer Instrumente und ihrer Klangeigenschaften übertragen. Denn mit der Zusammensetzung individueller Bauteile bestehen diese auch aus gekoppelten Subsystemen, welche alle, durch ihre reflektierenden Eigenschaften, einen Einfluss auf den entstehenden Klang haben. Je nachdem wie präzise ein Instrument mithilfe der IPF abgebildet werden soll, lassen sich auch verschiedene Bauteile, als ein einzelnes Subsystem zusammenfassen. Verknüpft man nun die Realität mit der Theorie kann der gebende Impuls der IPF beispielsweise das Anschlagen einer Saite, die Bogenkraft beim Streichen einer Saite oder bei Blasinstrumenten der Blasdruck sein. In der Realität bewegt sich dieser Impuls durch das Instrument und wird an verschiedenen Punkten, den Subsystemen, zeitversetzt reflektiert und wirkt mit dieser Reflexion wiederum auf das gesamte System und den entstehenden Klang ein. Die IPF erlaubt es solche Vorgänge mit einer simplen Formel nachzubilden und bietet somit eine solide Grundlage für die Nachbildung realistischer musikalischer Klangeigenschaften.

Grundlage

Betrachtet man nur einen Reflexionspunkt, so lässt sich die IPF für einen iterativen Prozess in der simpelsten Form, mit

$$g_+ = g - \ln\left(\frac{g}{\alpha}\right), \quad (1)$$

formulieren (Bader 2013, S. 287).

α repräsentiert den gebenden Impuls, g den aktuellen und g_+ den folgenden Zustand des Systems. Für eine Berechnung muss ein Startwert g_0 gewählt werden. Welchen Einfluss die Wahl von g_0 auf den Zustand des Systems hat, hat Linke bereits ausführlich betrachtet und ist

dazu gekommen, dass große initiale Werte für g_0 zu langen Transienten führen, was bei der Beschreibung von Musikinstrumenten nicht besonders wünschenswert ist (Linke, Bader, und Mores 2019a, S.29). Generell ist aber kein bestimmter Wert für g_0 festgelegt, wenn dieser unbekannt ist, sondern lässt sich je nach Zusammenhängen des Systems frei wählen. Zwischen g und g_+ ist allgemein kein konkretes zeitlich gebundenes Intervall festgelegt. Es ist vielmehr die Zeit, die vergeht, bis eine neue Veränderung stattfindet. Nimmt man an, dass es sich um ein periodisches System handelt, kann man $\Delta t = 1/f$ festlegen. Somit beträgt die Zeit, zwischen g und g_+ , die Länge einer Periode, abhängig von der zuvor definierten Frequenz.

Möchte man sich mit der IPF dem Verhalten realistischer Klangeigenschaften annähern, erfordert die Komplexität, die Betrachtung von mehr als nur einem Reflexionspunkts. Für die Betrachtung mehrerer Reflexionspunkte lässt sich die allgemeinste Form

$$g_+ = g - \ln \left(\frac{1}{\alpha} \left(g - \sum_{k=1}^n \beta_k e^{g-g_{k-}} \right) \right) \quad (2)$$

der IPF verwenden (Bader 2013, S.291).

Wie in (2) dargestellt, kann die IPF durch die Integration beliebig vieler Reflexionspunkte bzw. Subsysteme erweitert werden. Im Kontext von physikalischen Eigenschaften werden Subsysteme mit höherer Ordnung k , als weiter vom ursprünglichen System entfernt betrachtet, was dazu führt, dass der ursprüngliche Impuls diese Punkte später und gedämpft erreicht. In diesem Zusammenhang stellt der Ausdruck $\beta_k e^{g-g_{k-}}$ ein Subsystem dar, welches durch die Verknüpfung mit früheren Systemzuständen, g_{k-} , und beeinflusst durch die individuellen Reflexionsstärken β_k , eine zeitlich verzögerte und gedämpfte Reaktion auf vorhergehende Zustände auslöst.

Systemverhalten und Stabilität

In Abbildung 1 ist dargestellt, wie sich der Zustand g abhängig von $\frac{1}{\alpha}$, bei nur einem Reflexionspunkt, verändert.

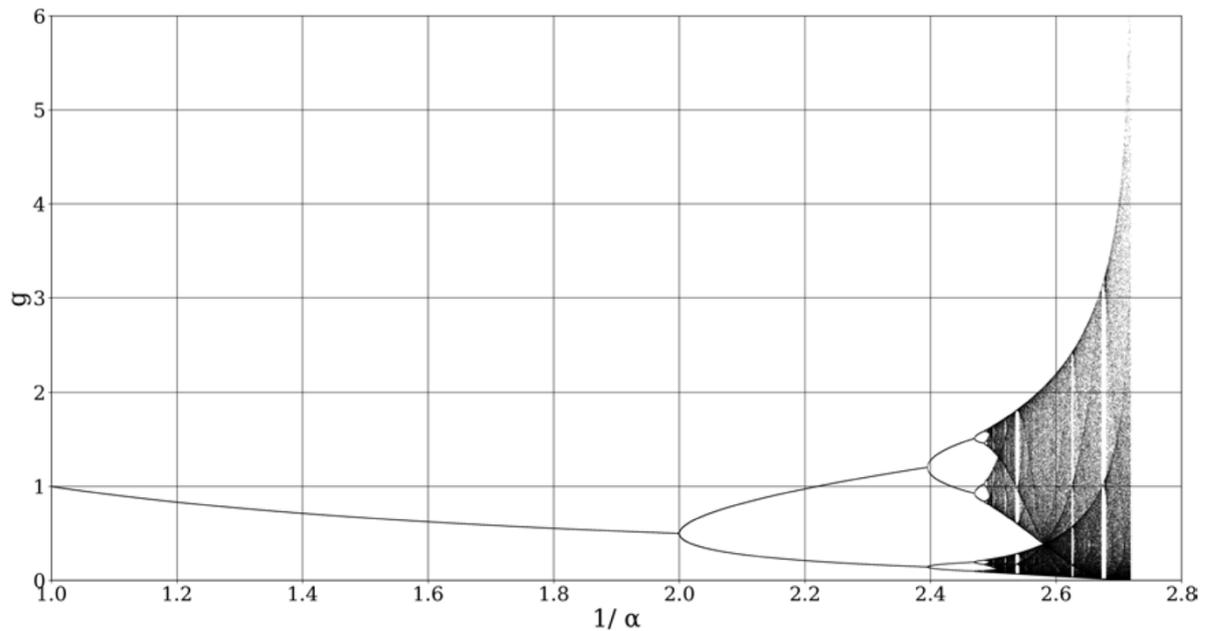


Abbildung 1: Bifurkationsszenario der IPF mit einem Reflexionspunkt

Quelle: (Linke, Bader, und Mores 2019b, 351)

Bis zu dem kritischen Punkt, $\alpha_c = 0.5$ bzw. $1/\alpha = 2.0$, welcher auch als erster Bifurkationspunkt bezeichnet wird, ist mit dem Resultat von jeweils nur einem Wert in Abhängigkeit von $1/\alpha$ und g , ein stabiles Verhalten zu beobachten. Wird dieser Wert jedoch überschritten, beginnt sich chaotisches Verhalten, was sich in ersten Bifurkationen äußert. Bifurkationen und in dem Sinne chaotisches Verhalten zeichnet sich dadurch aus, dass je nach Ordnung, zwei oder mehrere Zustände gleichzeitig existieren. Nach weiterer Erhöhung von $1/\alpha$, in Richtung $1/\alpha \geq 2.0$, steigt auch die Ordnung der Bifurkationen und endet in stark chaotischen Verhalten mit einer hohen Anzahl von gleichzeitig existierenden Zustandswerten. Diese Zustände lassen sich noch etwas besser nachvollziehen, wenn durch die IPF iteriert wird. Indem über 200 Iterationen die Folgezustände g_+ berechnet werden können die resultierenden Zustandswerte in einem Plot dargestellt werden.

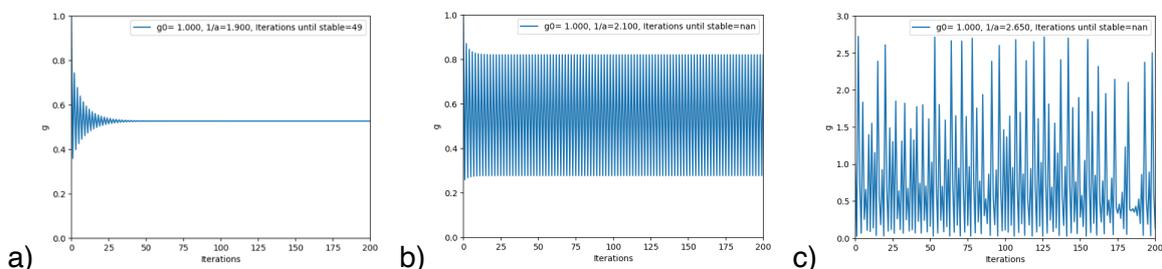


Abbildung 2: Verhalten der IPF bei Verwendung eines Reflexionspunkts, mit konstantem α und $g_0 = 1$, über 1000 Iterationsschritte a) Stabil: $1/\alpha = 1.9$ b) Bifurkation 1. Ordnung: $1/\alpha = 2.1$ und c) Chaotisches Verhalten: $1/\alpha = 2.65$,

Abbildung 2a) zeigt einen Fall für $1/\alpha \leq 2.0$ mit initialem Transienten und Einschwingverhalten bis zu einem stabilen Fixpunkt, ab welchem resultierenden Folgezustände keine Veränderung mehr aufweisen. Auch in Abbildung 2b) ist ein kurzer Transient sichtbar, jedoch oszillieren die Zustände anschließend konstant, da nun der kritische Punkt $\alpha_c = 0.5$ überschritten wurde und der 1. Bifurkationspunkt erreicht ist. Abschließend veranschaulicht Abbildung 2c) ein hochgradig chaotisches Verhalten, bei dem die Größe g keine harmonische Oszillation erfährt, sondern sich zwischen einer Vielzahl verschiedener Werte hin und her bewegt.

Laut Linke existiert für alle Kombinationen von α und β_k ein Fixpunkt g_s , der dadurch definiert ist, dass der Systemzustand zu allen Zeitpunkten konstant ist. Durch das Umstellen von (2

$$g_s = \alpha + \sum_{k=1}^n \beta_k \quad (3)$$

lässt sich g_s berechnen (Linke, Bader, und Mores 2019b, 352).

Bader zeigte, dass die Zustandswerte, bei einem konstanten α , in der Regel nach etwa $n < 300$ Iterationsschritten zu einem Fixpunkt g_s konvergieren, sofern das System stabil ist (Bader 2013, S.295).

Grenzwerte

Damit physikalische Eigenschaften, wie zum Beispiel die eines Musikinstruments, korrekt beschrieben werden können ist laut dem Energie-Erhaltungssatz vorausgesetzt, dass die Energien eines abgeschlossenen Systems immer konstant bleibt (von Helmholtz 1889). Dementsprechend können Subsysteme höherer Ordnung in der Summe auch nicht mehr Energie als α reflektieren, was sich durch

$$\alpha \geq \sum_{k=1}^n \beta_k \quad (4)$$

beschreiben lässt (Linke, Bader, und Mores 2019b, 352). Diese Einschränkung gilt außerdem fortlaufend für alle darauffolgenden Subsysteme. Wie bereits erwähnt erreicht der Impuls die Systeme mit höherer Ordnung k später und gedämpft, was zu den folgenden Einschränkungen

$$\alpha > \beta_1 > \beta_2 > \dots > \beta_k \quad (5)$$

und

$$\alpha + \sum_k^n \beta_k < 1 \quad (6)$$

führt (Bader 2013, S.291).

Linke zeigte, dass sich unter Verwendung von nur einem Reflexionspunkt, analytisch ein α_{min} berechnen lässt welches für alle g -Werte zutrifft und ist zu dem Ergebnis $1/\alpha_{min} = e \approx 2.7$ bzw. $\alpha_{min} \approx 0.37$ gekommen (Linke, Bader, und Mores 2019b, S.352). Diese Annahme bestätigt sich bei einem Blick auf Abbildung 1. Da in dieser Arbeit aber von mehreren Reflexionspunkten ausgegangen wird, ist eine analytische Lösung, durch den Bezug zu vorherigen Zuständen und der Abhängigkeit von g_0 , nicht möglich (Linke, Bader, und Mores 2019a, S.352).

2.2. Synthese- und Modulationsverfahren

Zur Entwicklung eines funktionalen Synthesizers werden grundlegende Synthese- und Modulationsverfahren eingesetzt. Statt akustischer Schallquellen kommen elektronische Schaltungen oder Software zum Einsatz. Wesentliche Bestandteile der Klangsynthese sind Modulationsverfahren wie Frequenzmodulation (FM), Amplitudenmodulation (AM), Phasenmodulation (PM) und Hüllkurvenmodulation. Diese Verfahren ermöglichen, in Kombination mit Syntheseverfahren wie der Wavetable-Synthese die Echtzeitmodulation von Parametern wie Tonhöhe, Lautstärke und Filterfrequenz.

Die Verknüpfung von Zustandswerten der IPF mit der Klangerzeugung erfordert eine Kombination dieser Synthese- und Modulationsverfahren, die anschließend durch die IPF-Werte kontrolliert werden können. Durch die Auswahl geeigneter Werte für α und β_k und die Verwendung einer geeigneten Wellenform kann der Klang eines physikalischen Instruments, synthetisch nachgebildet werden. Dieser Ansatz wurde bereits ausführlich von Linke, Bader und Mores untersucht und umgesetzt (Linke, Bader, und Mores 2019b). Für die Modellierung einer Dizi-Flöte wurden die Amplitudenwerte von fortlaufenden Gauß-Pulsen mit den Zustandswerten g_k verknüpft, während die Periodenlänge durch die Differenz der Zustandswerte mit

$$T_k = \frac{1}{f_0} (1 + (g_k - g_{k-})) \quad (7)$$

moduliert wurde (Linke, Bader, und Mores 2019b, 357).

Bei der Verwendung der IPF für die Klangsynthese bedarf es also einiger Grundlagen, welche im Folgenden näher betrachtet werden.

Wavetable-Synthese

Die Wavetable-Synthese ist eine Methode zur Erzeugung von Klängen, die auf der Speicherung und Manipulation von Wellenformen in einer Tabelle basiert. Die Wellenformen können aus natürlichen oder synthetischen Quellen stammen und werden in diskrete Samples zerlegt, die in einem Speicher abgelegt werden. Im Kontext von digitaler Audiotechnik sind Samples diskrete Werte, die den Pegel des Audiosignals zu bestimmten Zeitpunkten darstellen. Durch Ändern der Abspielgeschwindigkeit, der Startposition oder der Kombination verschiedener Wellenformen kann eine große Variation von Klängen erzeugt werden. In dem Anwendungsfall dieser Arbeit kann die Wavetable-Synthese verwendet werden um einfache Signalformen wie Sinus-, Dreieck-, Quadrat- oder Sägezahnsignale in gewünschter Frequenz abzuspielen und diese im Takt der Periode abhängig des aktuellen Zustand-Wertes zu manipulieren.

Amplitudenmodulation (AM)

Eine für die Untersuchung naheliegende Art der Modulation, im Zusammenhang mit der Beschreibung eines Einschwingverhaltens, ist die Amplitudenmodulation. Wie auch die anderen Modulationsverfahren findet dieses Konzept ihren Ursprung im Fachbereich der Nachrichtentechnik. Bei der Amplitudenmodulation wird die Amplitude eines Trägersignals, in Abhängigkeit eines Informationssignals moduliert. Im Kontext der IPF kann das Modulationssignal aus den Zustandswerten abgeleitet und das Trägersignal mittels Wavetable-Synthese erzeugt werden. Dabei erfolgt mit

$$A_k = g_k \tag{8}$$

im Takt der zugrundeliegenden Periodenlänge, eine Verrechnung der Zustandswerte g_k mit den Amplitudenwerten A_k .

Frequenzmodulation (FM)

Ein weiteres bedeutendes Modulationsverfahren, das im Kontext der IPF betrachtet wird, ist die Frequenzmodulation (FM). Analog zur Amplitudenmodulation basiert die Frequenzmodulation auf Konzepten aus der Nachrichtentechnik und ermöglicht die Manipulation der Frequenz eines Trägersignals mithilfe eines Modulationssignals. Eine mögliche Verknüpfung der IPF mit der Frequenzmodulation wird in Formel (7) dargestellt. Mit den berechneten Periodenlängen kann durch die Modulierung der Abspielgeschwindigkeit eines Wavetables, die Frequenzen beeinflusst werden.

Phasenmodulation (PM)

Bei der Phasenmodulation wird statt der Frequenz die Phasenbeziehung zwischen den Schwingungen eines Trägersignals mithilfe des Modulationssignals modifiziert. Dies führt zu einer Überlagerung des Trägersignals mit dem modulierenden Signal, wobei Obertonstrukturen und klangliche Variation entstehen. Um die Phasenverschiebung des zu modulierenden Signals, im Sinne der IPF, zu berechnen, kann mit

$$\varphi_k = ((g_k - g_{k-}) \cdot 360) \bmod 360 \quad (9)$$

auch hier die Differenz von zwei aufeinanderfolgenden Zuständen verwendet werden.

Es ist anzumerken, dass die exakte Verknüpfung der Zustandswerte mit den Modulationsverfahren nicht eindeutig festgelegt ist, da sie stark von den spezifischen Charakteristika des zu beschreibenden Systems abhängt. Die hier präsentierten Ansätze dienen lediglich als beispielhafte Darstellungen und sind teils aus anderen Forschungsarbeiten in diesem Bereich entnommen.

2.3. Latenz und Roundtrip

Auf der Steinberg Support-[Seite](#) heißt es „Bereits wenige Millisekunden Latenz werden von einigen Anwendern als störend empfunden, ab 20ms ist die Latenz auch von weniger geübteren Ohren deutlich wahrzunehmen.“ (Latenz der Audiokarte – Steinberg Support o. J.). Bei der Gestaltung eines digitalen Musikinstruments ist die Latenz ein besonders kritischer Faktor, welcher sich auf die Spielbarkeit und Verwendung auswirkt. Latenz wird allgemein als die Zeit, die zwischen einem Ereignis und seiner Wirkung vergeht, bezeichnet. Ein weiterer Begriff, welcher sich aus diesem Kontext ableitet, ist der Roundtrip. Der Roundtrip bezieht sich auf die Gesamtzeit, die für eine Aktion benötigt wird, einschließlich des Hinwegs und des Rückwegs. An welchen Punkten gemessen wird hängt aber ganz davon ab welche Qualitäten oder Schnittstellen geprüft werden sollen.

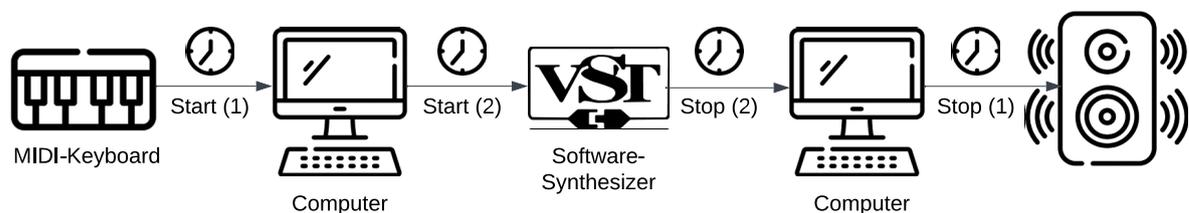


Abbildung 3: Möglichkeiten den Roundtrip eines Software-Synthesizers zu messen

In dem Fall von Anwendungen wie einem Software-Synthesizer, welcher über ein MIDI-Keyboards angesteuert wird, könnte, wie in Abbildung 3 dargestellt, die Zeit gemessen werden, die

benötigt wird, um ein MIDI-Signal vom Keyboard zum digitalen Instrument zu senden, es zu verarbeiten und das resultierende Audiosignal zurückzugeben, um es über Lautsprecher oder Kopfhörer auszugeben. Dabei sollte stets bedacht werden, dass Faktoren wie Abtastrate, die Blockgröße und Auflösung der zu verarbeitenden Samples, sowie die verwendeten Geräte und ihre Limitierungen einen Einfluss auf die Geschwindigkeit der Verarbeitung und Wiedergabe des Audiosignals haben.

3. Konzipierung

„Design thinking is a human centered and collaborative approach to problem solving, using a designed mindset to solve complex problems.“ (Brown zitiert nach Logan 2018). So wird Tim Brown, aktuell Vorsitzender von IDEO, einer internationalen Design- und Innovationsberatung, in einem Online-Artikel zitiert. (Logan 2018) Der Design Thinking Prozess ist in vielen Bereichen der Entwicklung ein fester Bestandteil. Angesichts dessen erscheint es sinnvoll, Grundlagen dieses Konzepts auch bei der Entwicklung des Synthesizers, auf Grundlage der IPF, anzuwenden.

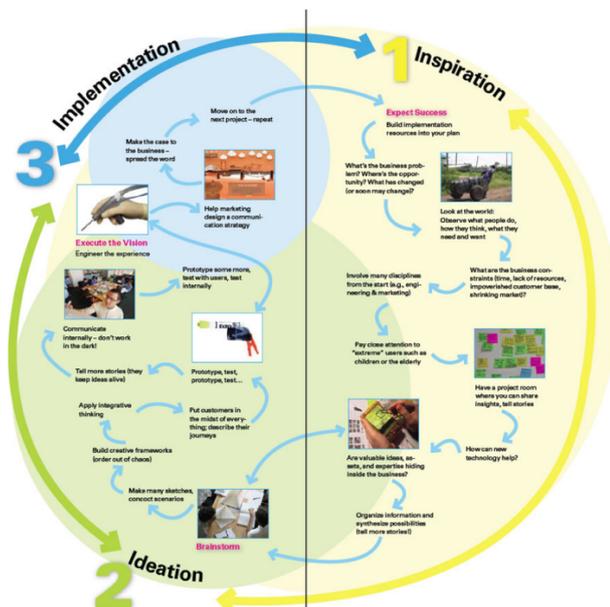


Abbildung 4: Tim Browns Design Thinking Process

Quelle: (Brown 2008)

Der Design Thinking Prozess gliedert sich laut Brown grundlegend in die Phasen der Inspiration, Ideenfindung und Umsetzung. Dabei kann aber zwischen den Schritten vor- und zurückgegangen werden, was zu einem dynamischen Prozess führt. Basierend auf diesen Schritten wird im Folgenden ein Konzept für die Realisierung des IPF-Synthesizers entwickelt. Damit

die Verständlichkeit und Nachvollziehbarkeit gewährleistet ist werden jedoch nicht alle Iterationsschritte aufgeführt und die Schritte werden in einem linearen Ablauf dargestellt.

3.1. Zielsetzung und Anforderungen

Das Ziel liegt darin, einen polyphonen Software-Synthesizer zu entwickeln welcher nicht nur musikalisch spielbar ist, sondern den Nutzer:innen auch einen benutzerfreundlichen und frustfreien Umgang ermöglicht. Der Fokus liegt auf einer klaren und verständlichen Darstellung der einzustellenden Parameter welche, ohne tiefgreifendes Fachwissen, intuitiv wählbar sein sollen. Gleichzeitig soll der Synthesizer ausreichend Raum für kreative Freiheiten bieten, sodass die Nutzer:innen ihre musikalischen Ideen verwirklichen können. Das Design des User-Interfaces soll den gängigen Regeln der User Experience folgen, um eine reibungslose Interaktion zu gewährleisten. Eine anschließende Validierung soll prüfen ob die vorgestellten Ansätze und Konzepte diesen Anspruch erfüllen können und die Grundlage für eine intuitive Wahl der Parameter liefert.

3.2. Problemdarstellung

Durch die nicht Linearität der IPF sind die Ergebnisse von verschiedenen Wertekombinationen der Kontrollparameter, ohne vorherige Berechnungen, nahezu unvorhersehbar. Es handelt sich teils um kleine nicht regelmäßig verteilte Wertebereiche, in welchen sich bewegt werden kann, um gewünschte Ergebnisse zu erhalten. Mit nur einem Reflexionspunkt lassen sich Grenzen und Bereiche noch analytisch ermitteln und Wertebereiche lassen sich wie in Abbildung 1 noch verständlich darstellen. Mit jedem weiteren Subsystem steigt die Möglichkeit der Werte-Kombinationen jedoch exponentiell und somit wird auch die Auswahl der Werte immer komplexer. In Abbildung 2 ist sichtbar, wie sich die Zustände mit nur einem weiteren Reflexionspunkt, für den Fall $\beta = 0.164$, verändern können. Eine Erhöhung von $1/\alpha$ führt nicht mehr linear von stabilen Zuständen, über erste Bifurkationen bis hin zu stark chaotischen Verhalten, sondern es kann auch Übergänge von chaotischen hin zu stabilen Verhalten und wieder zu Bifurkationen geben.

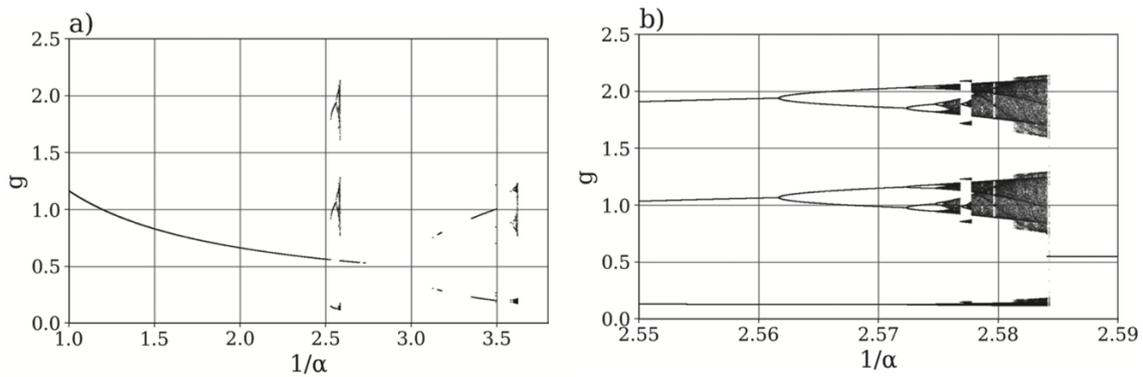


Abbildung 5: “Bifurcation scenario of the IPF with two reflection points in dependence of $1/\alpha$ with $\beta = 0.164$. The lower chart (b) is a zoom of the chaotic region.”

Quelle: (Linke, Bader, und Mores 2019a, S.29)

Um ein gewünschtes Verhalten des Systems trotzdem gezielt ansteuern zu können bedarf es Lösungskonzepten, welche intuitiv und nachvollziehbar ist.

3.3. User Persona

Im Rahmen der Entwicklung von Software-Instrumenten und Musiktechnologien ist es von wesentlicher Bedeutung, die Nutzergruppen zu identifizieren und ihre individuellen Anforderungen zu berücksichtigen. Diese User Persona beschreibt eine breite Gruppe von Musiker:innen die von professionellen Produzent:innen bis zu Hobby-Laien reichen, welche das besagte Software-Instrument verwenden könnten. Einige sind Expert:innen mit tiefem Verständnis für Musikproduktion und Instrumentierung, während andere sich erst am Anfang ihrer musikalischen Kompetenz befinden und nach intuitiven Werkzeugen suchen, um Klänge zu erzeugen. Die Anforderungen dieser vielfältigen Nutzergruppe können in verschiedene Kategorien unterteilt werden:

1. Kontrolle und Anpassung: Einige Musiker:innen wünschen sich umfassende Kontrolle über die Klangparameter, um ihre kreativen Ideen umzusetzen. Sie schätzen die Möglichkeit, feine Anpassungen vorzunehmen und in technische Details einzutauchen, um einzigartige Klänge zu erschaffen.

2. Benutzerfreundlichkeit: Andere Nutzer legen Wert auf Benutzerfreundlichkeit und möchten ohne lange Einarbeitungszeit direkt loslegen können. Sie bevorzugen eine intuitive Benutzeroberfläche, die es ermöglicht, auf einfache Weise ansprechende Klänge zu erzeugen, ohne sich in komplizierte Einstellungen vertiefen zu müssen.

3. Kreative Inspiration: Unabhängig vom Erfahrungsniveau suchen Musikerinnen und Musiker nach einer Lösung, die ihre kreative Inspiration fördert. Sie möchten spontan Klänge erforschen und experimentieren können, um neue musikalische Ideen zu entwickeln.

4. Flexibilität und Vielseitigkeit: Die Nutzerinnen und Nutzer möchten ein flexibles Software-Instrument, das verschiedene Genres und Musikstile unterstützt. Sie möchten sowohl traditionelle als auch innovative Klänge erzeugen können.

5. Lernkurve: Einige Nutzer sind bereit, Zeit und Mühe in das Instrument zu investieren, um seine Feinheiten zu verstehen. Andere suchen eher nach einer schnellen Erfolgserfahrung

Die Gestaltung des Software-Instruments sollte daher eine ausgewogene Balance zwischen Kontrollmöglichkeiten und Benutzerfreundlichkeit bieten, um die Bedürfnisse dieser vielfältigen Nutzergruppe zu erfüllen. Durch die Bereitstellung von anpassbaren Einstellungen, intuitiven Schnittstellen und kreativen Werkzeugen kann das Software-Instrument Musiker:innen auf unterschiedlichen Erfahrungsstufen gleichermaßen ansprechen und unterstützen.

3.4. Lösungskonzepte

Der folgende Abschnitt gibt Einblick in die zentralen Ansätze, um die gestellten Herausforderungen zu adressieren. Diese Konzepte dienen als grundlegende Bausteine für die erfolgreiche Umsetzung des angestrebten IPF-Synthesizers. Es werden die essenziellen Ideen und Konzepte vorgestellt, die dazu beigetragen haben, sowohl die technischen als auch die kreativen Anforderungen des Projekts zu erfüllen.

3.4.1. Visuelle Darstellung der IPF als Plot

Die Verwendung von Plots zur visuellen Darstellung von IPF-Werten und des daraus resultierenden Signals ist eine gute Möglichkeit, um die Verknüpfung der Einstellungen verständlicher zu gestalten. Dieser Ansatz ermöglicht es den Nutzer:innen, die Auswirkungen von Änderungen an den Parametern direkt nachvollziehen zu können und ein besseres Verständnis für die Zusammenhänge aufzubauen. Es ist ein gängiges Konzept in der Gestaltung von Benutzeroberflächen, visuelle Darstellungen zu verwenden, um komplexe Informationen leichter verständlich zu machen. Dies wird auch durch die "10 Usability Heuristics for User Interface Design" von Jakob Nielsen, einem renommierten Fachbuchautor, Redner und Berater im Bereich Software- und Webdesign, unterstützt (Nielsen 2020). Hier wird die Verwendung von „Sichtbarkeit des Systemstatus“ und „Erkennung statt Erinnerung“ als eine wichtige Regel für die Gestaltung von Benutzeroberflächen genannt. Aufgrund der Menge an Artikeln und Beiträgen im Internet, welche sich auf diese Regeln beziehen, ist davon auszugehen, dass sie, obwohl diese Regeln vor über zwanzig Jahren formuliert wurden, auch heute noch als relevantes

Werkzeug für die Gestaltung von Benutzerinteraktion und Benutzeroberflächen angesehen werden. Auch der nächste Ansatz macht sich dieses Konzept zunutze.

3.4.2. Visualisierung des zu erwartenden Verhaltens durch Farben

Eine zentrale Herangehensweise zur Erfüllung der zuvor dargelegten Anforderungen ist es das zu erwartende Verhalten schon vor der Auswahl der Werte zu visualisieren, um sichtbar zu machen in welchen Wertebereichen sich mit den Kontrollelementen bewegt werden kann und welches Verhalten zu erwarten ist. Dies geschieht durch die Anwendung einer farblichen Kodierung, die um die Kontrollelemente herum dargestellt wird und sich kontinuierlich an die aktuellen Einstellungen anpasst. Im Folgenden werden verschiedene Ansätze beschrieben, wie diese visuelle Farbgebung genutzt werden kann, um eine optimale Benutzererfahrung für verschiedene Anwendungsfälle abzudecken.

Farbzuordnungen für unveränderte Parameter: Für den Fall, dass die Kontrollelemente während des Spielens nicht verändert werden, ist es möglich, eine definitive Darstellung des resultierenden Verhaltens zu verwenden. Insbesondere für Anfänger stellt dies eine hilfreiche Methode dar, um den gewünschten Klang intuitiv zu erzeugen, ist dabei aber nicht auf die flexible Nutzung ausgelegt.

Wahrscheinlichkeitsvisualisierung für flexible Anpassungen: Für Musiker:innen die beabsichtigen die Kontrollparameter während des Spielens zu variieren, bietet sich die Nutzung einer dynamischen Wahrscheinlichkeits-Visualisierung in Bezug auf das zu erwartende Verhalten an. Wenn das Auftreten eines bestimmten Verhaltens für eine Reihe von gleichmäßig verteilten Anfangswerten von g_0 eine vorab festgelegte Schwelle überschreitet, erfolgt dessen Darstellung innerhalb der besagten Visualisierung. Sollte kein Verhalten diese Schwelle überschreiten, erfolgt die Darstellung des Verhaltens, welches am häufigsten auftritt. Diese Option ermöglicht es, sowohl Kontrolle als auch Flexibilität miteinander zu vereinen.

Eine Auswahl dieser Optionen in der Bedienoberfläche bietet für verschiedene Anwendungsfälle Wege, wie die Visualisierung durch Farben dazu beitragen kann, den Nutzer:innen eine effektive und ansprechende Interaktion, ohne weiteres Vorwissen zu der IPF, mit dem Synthesizer zu ermöglichen. Unabhängig von ihrem musikalischen Hintergrund oder

Erfahrungsniveau können Musiker:innen die Kontrolle über den Klang behalten und ihre kreativen Ideen umsetzen.

4. Stabilitätsanalyse

Das zentrale Lösungskonzept aus Abschnitt 3.4.2, welches sich die farbliche Darstellung zunutze macht, setzt eine Stabilitätsanalyse der Wertekombinationen, welche sich über die Kontrollparameter einstellen lassen, voraus. Aufgrund der großen Menge an verschiedenen Kombinationen ist es sinnvoll diese Analyse mithilfe eines Programms durchzuführen welches automatisch durch alle Kombinationen iteriert und diese nach resultierendem Verhalten einordnet. Zudem müssen die Ergebnisse in einem geeigneten Format gespeichert werden, um von der Synthesizer-Software gelesen werden zu können. Für die Umsetzung des Analyse-Programms wurde die Programmiersprache Python verwendet.

4.1 Grundlegende Annahmen

Bei allen Berechnungen wird von 3 Reflexionspunkten α , β und γ und den bestehenden Grenzen und Verhaltensregeln aus Absatz 2.1, welche auch auf mehrere Reflexionspunkte zutreffen, ausgegangen. Auch wenn Bifurkationen in der Theorie Teil des chaotischen Verhaltens darstellen, werden Bifurkationen bis zur 3. Ordnung in diesen Untersuchungen als eine eigene Einordnung betrachtet. Des Weiteren wird in stabiles und chaotisches Verhalten unterteilt. Die Grenzen für die Einordnung des stabilen Verhaltens werden so gewählt, dass sie im musikalischen Kontext sinnvoll sind. Divergierendes Verhalten wird mit allen ungültigen Berechnungen zusammengefasst.

4.2 Regeln zur Bestimmung der Zustände

Stabilität: Aus vorherigen Erkenntnissen ist bekannt, dass sich ein stabiles System in der Regel, bei nur einem Reflexionspunkt und konstantem Kontrollparameter α , nach $n \leq 300$ Iterationsschritten dem Konvergenzwert angenähert haben sollte. In Bezug auf die Darstellung musikalischer Klangeigenschaften wird diese Grenze auf $n \leq 250$ reduziert, um zu lange Transienten auszuschließen. Somit wird eine Wertekombination als stabil bewertet, wenn g nach mindestens $n \leq 250$ zu g_s konvergiert ist. Wird dieser Punkt innerhalb der vorgegebenen Anzahl an Iterationen nicht erreicht so ist es wahrscheinlich, dass es sich um chaotisches Verhalten handelt und wird auch so eingeordnet.

Bifurkationen: Für den Fall einer Bifurkation ist es nicht mehr möglich sich nach g_s zu richten, da dieser Punkt langfristig niemals erreicht wird. Bei einer Bifurkation spalten sich die Zustandswerte, wie auch in Abbildung 2 am ersten Bifurkationspunkt sichtbar, in zwei weitere Werte auf. Um diesen Fall zu bestimmen kann geprüft werden wie viele einzigartige Werte nach dem Einschwingverhalten vorhanden sind. In diesem Fall gilt es also zu bewerten, ob eine Bifurkation auftritt und in wie viele Zustände diese sich aufteilt.

Ermittelt man die Menge an einzigartigen Zahlen aus einer Reihe an Werten, welche einem Abschnitt nach $n \geq 250$ Iterationen entstammt, so erhält man nicht nur die Information darüber, ob es zu einer Bifurkation gekommen ist, sondern auch in wie viele Zustände diese sich aufgeteilt hat. Alle Fälle bei denen es mehr als 1 aber weniger als acht einzigartige Zustände gibt werden als Bifurkation eingestuft. Also lassen sich für diesen Wert folgende Einstufungen festlegen:

Tabelle 1: Werte zur Einordnung von Bifurkationen und deren Ordnung

Bifurkation	Menge an einzigartigen Werten nach $n \geq 250$ Iterationen
1. Ordnung	2
2. Ordnung	4
3. Ordnung	8
Allg. Bifurkation	$> 1 \text{ und } \leq 8$
chaotisch	> 8

Divergenz und ungültige Werte: Eine Divergenz lässt sich feststellen indem geprüft wird, ob der berechnete Zustand komplex ist oder der Realteil negativ ist. Des Weiteren ist eine Berechnung ungültig, wenn das Argument des Logarithmus 0 ist. Dies kann geprüft werden, bevor die nächste Berechnung durchgeführt wird. Wenn einer dieser Fälle zutrifft, wird die Berechnung abgebrochen und die Wertekombination wird zusammen mit den Fällen für Divergenz als ungültig eingestuft.

4.3 Auflösung und Wertebereiche

Die Auflösung der zu iterierenden Werte hat einen maßgeblichen Einfluss auf die Größe der endgültigen Datenreihe, die die zu speichernden Informationen enthält. Es ist von besonderer Bedeutung, sicherzustellen, dass diese Datenmenge nicht unnötig groß wird. Eine erhöhte Datenmenge erfordert längere Lesezeiten und verlangsamt den Prozess der Identifizierung

spezifischer Datensätze, was sich negativ auf die Gesamtperformance des Programms auswirken kann. Angesichts der Ausrichtung der Anwendung auf die Verwendung eines MIDI-Controllers genügt es, die Auflösung der Werte auf maximal 128 Schritte festzulegen, da eine MIDI-Nachricht einen Informationsgehalt von 7 Bit übertragen kann. In diesem Fall wurde sich der Einfachheit für eine Auflösung von 101 Schritten mit einer Schrittgröße von 0.1 entschieden. Für die g_0 -Werte wird ein Bereich von 0 bis 4 mit einer Schrittgröße von 0.1 gewählt, um den Bereich für zufällig auftretende Werte grob abzudecken.

4.4 Speicherung der Daten

Die Verwendung einer CSV-Datei (Comma-separated values) als Speicherformat für Analyse-Daten bietet eine effiziente und standardisierte Methode, um wichtige wissenschaftliche Informationen zu erfassen und zu organisieren. Die simple Struktur ermöglicht eine gute Lesbarkeit für das menschliche Auge und auch für Computeranwendungen ist es einfach die Daten auszulesen. Da für den Lösungsansatz der farblichen Darstellung mehrere unterschiedliche Ansichten vorgesehen sind müssen die Daten in unterschiedlichen Tabellen gespeichert werden, welche je nach Anforderung unterschiedlich formatiert sind. Diese Formatierungen sind in den folgenden beiden Tabellen aufgeführt. Die eingetragenen Daten sind lediglich Beispieldaten und stellen nicht die tatsächlichen Ergebnisse dar.

Tabelle 2: Formatierung für eindeutig berechnetes Verhalten aller Kombinationen inklusive des Startwerts g_0

g_0	α	β	γ	Verhalten
0.3	0.6	0.2	0.1	chaotisch
0.4	0.6	0.2	0.1	Bifurkation
0.5	0.6	0.2	0.1	stabil
...

Tabelle 3: Formatierung der Liste für die Darstellung welches Verhalten, über alle g_0 -Werte, über einen definierten Prozentsatz aufgetreten ist.

α	β	γ	Zustand welcher über x% über alle g-Werte auftritt
0.6	0.2	0.1	stabil
0.7	0.2	0.1	chaotisch
0.8	0.2	0.1	Bifurkation
...

4.5 Struktur des Analyseprogramms

Der erste Bestandteil des Analyseprozess sind die verschachtelten Iterationsschleifen über die Wertebereiche der Kontrollparameter g_0 , α , β und γ . Die Wertebereiche dieser Parameter werden in je einem Array dargestellt, welches sich über die Iterationen hinweg an die zuvor definierten Beschränkungen dynamisch anpasst mit einer Auflösung von 101 Schritten durchlaufen wird. Für jeden Schritt der verschachtelten Iterationen wird ein Analyse-Prozess ausgelöst, welcher die momentanen Werte für die Kontrollparameter übernimmt und eine Berechnung der IPF über 500 Iterationen ausführt. Während dieser Prozess durchläuft, wird nach den definierten Regeln zur Bestimmung des Verhaltens aus Abschnitt 4.2 bestimmt um welches Verhalten es sich bei dieser Wertekombination handelt und in einem Array für die Resultate, abgespeichert. In der verschachtelten Iterationsschleife steht die Iteration über die Startwerte g_0 an letzter Stelle. Dadurch besteht die Möglichkeit, für jede Kombination der Parameter α , β und γ das Verhalten für alle g_0 -Werte zu bestimmen und zu überprüfen, wie häufig jede Verhaltensmöglichkeit auftritt. Wenn dieser Prozess für alle Wertekombinationen abgeschlossen ist, werden die Ergebnisse der Arrays jeweils in eigenen CSV-Dateien gespeichert.

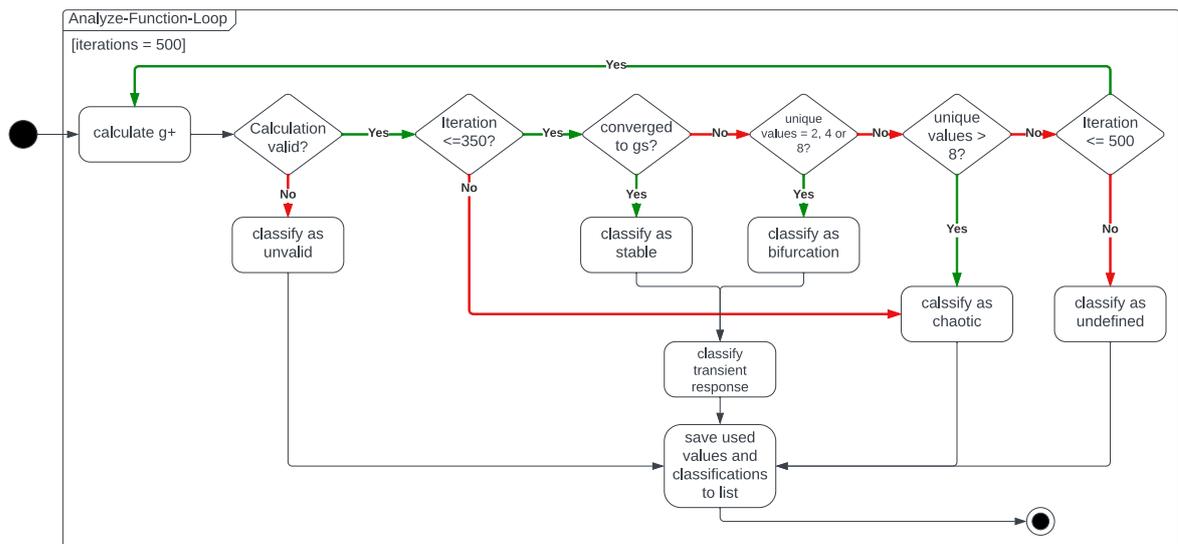


Abbildung 6: Vereinfachte Darstellung des Analyse-Prozesses

In Abbildung 6 ist der Analyse-Prozess in vereinfachter Form mit den entscheidenden Abfragen dargestellt.

5. Realisation

5.1 Schnittstelle des Synthesizers

Die Wahl der Schnittstelle für die Entwicklung des Synthesizers hat einen entscheidenden Einfluss darauf, wie vielseitig er genutzt werden kann. Eine Umfrage zur Nutzung digitaler musikalischer Instrumente zeigt, dass rund 70 Prozent der befragten Musikschaaffenden, die elektronische Musikinstrumente (EMIs) verwenden, Computersoftware und MIDI-Controller einsetzen. Daher ist es sinnvoll einen Weg zu finden in welchem sich beide Optionen integrieren lassen.

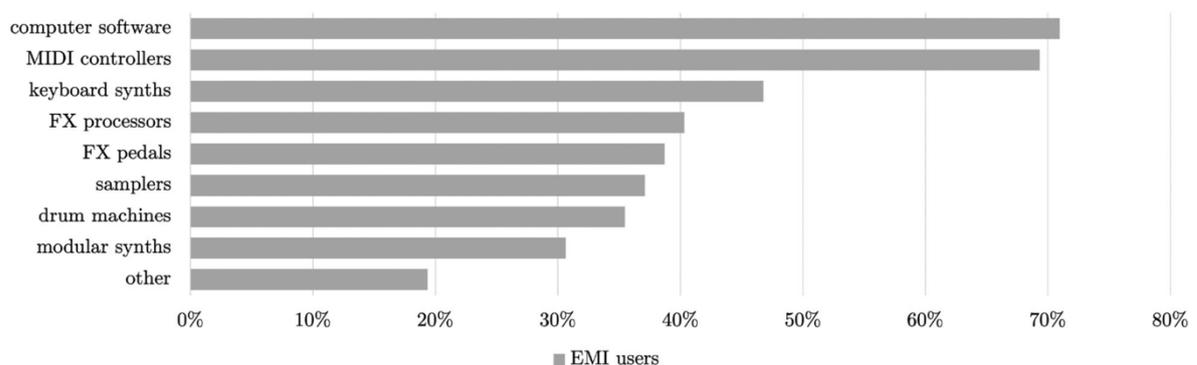


Abbildung : Prozent an EMI Nutzern (N=62) welche jenes EMI jeweils verwenden (Sullivan, Guastavino, und Wanderley 2021, S. 477)

Virtual Studio Technology (VST) ist eine etablierte Schnittstelle für Audio-Plugins, das ursprünglich im Jahr 1996 für die Digital Audio Workstation (DAW) Cubase von Steinberg Media Technologies entwickelt wurde und sich später als Industriestandard durchgesetzt hat. Die Verwendung der VST-Struktur als Grundlage für einen virtuellen Synthesizer bringt verschiedene Vorteile mit sich. Aufgrund ihres Status wird das VST-Format von allen gängigen DAWs sowie auch von einigen Schnittprogrammen wie Davinci Resolve oder Adobe Premiere unterstützt. Diese breite Unterstützung ermöglicht den Einsatz eines Synthesizers im VST-Format in einer Vielzahl von Musikproduktionsumgebungen und gewährleistet dessen Wettbewerbsfähigkeit in der Branche. Zusätzlich wird den Benutzer:innen eine nahtlose Integration und Verwaltung des Synthesizers in bestehende Produktions- und Aufnahmeworkflows ermöglicht. Durch die Unterstützung des MIDI-Protokolls ist es möglich, den Synthesizer mittels eines MIDI-Controllers anzusteuern, wodurch ein natürliches Kontroll- und Spielerlebnis ermöglicht wird. Infolgedessen wird die VST-Schnittstelle für die Entwicklung genutzt, um auf die genannten Vorteile zuzugreifen.

5.2 Framework

Für die Umsetzung des IPF-Synthesizers wurde das JUCE-Framework als Entwicklungsumgebung gewählt, welches auf der Programmiersprache C++ basiert. JUCE ist ein plattformübergreifendes Framework für die Audioprogrammierung mit einer umfangreichen Sammlung von Audio- und DSP (Digital-Signal-Processor)-Klassen sowie vorgefertigten Funktionen zur Integration von MIDI-Steuerung. Das Framework wird von einer aktiven Community unterstützt und bietet eine solide Grundlage für die Entwicklung eines Audio-Plugins. Ein weiterer Vorteil ist außerdem, dass sich das entwickelte Plugin auch als Stand-Alone-Programm kompilieren lässt, in das Einstellungsoptionen wie die Auswahl des zu verwendenden MIDI-Geräts und der Abtastrate automatisch integriert sind.

5.2.1 Struktur des Frameworks

In dem Framework werden grundlegend zwei Klassen unterschieden: der Plugin-Processor und der Plugin-Editor. Die Plugin-Processor-Klasse ist für die eigentliche Audioverarbeitung des Plugins verantwortlich, wie zum Beispiel das Generieren von Tönen oder das Anwenden von Modulationen auf das Audiosignal. Hierbei handelt es sich um die Kernfunktionen des Plugins im Audiopfad.

Die Plugin-Editor-Klasse dient hingegen der Erstellung und Verwaltung der grafischen Benutzeroberfläche (GUI) des Plugins. Hier werden Elemente wie Regler, Schalter oder Anzeigefelder erzeugt, die es dem Benutzer ermöglichen, die Parameter des Plugins zu steuern. Die Plugin-Editor-Klasse arbeitet eng mit dem Plugin-Processor zusammen, indem sie die Parameter des Audioprozessors darstellt und dem Benutzer eine intuitive Möglichkeit bietet, sie zu steuern.

Zusammen bilden die Plugin-Processor- und Plugin-Editor-Klassen eine zentrale Komponente für die Entwicklung von Audio-Plugins im JUCE-Framework. Sie ermöglichen die Trennung von Audioverarbeitung und Benutzeroberfläche. Allerdings ist man bei der Entwicklung von Audio-Anwendungen nicht auf die vorgefertigten Klassen beschränkt. Da es sich weiterhin um eine C++-Umgebung handelt, stehen alle herkömmlichen Regeln und Möglichkeiten der C++-Programmierung zur Verfügung. Das bedeutet, dass Entwickler ihre eigenen Klassen erstellen können, um spezifische Funktionalitäten zu implementieren, die über die grundlegenden Funktionen der Plugin-Processor- und Plugin-Editor-Klassen hinausgehen. Das Framework dient hierbei als unterstützendes Werkzeug, das Entwickler:innen dabei hilft, bestimmte Aufgaben

einfacher zu gestalten und ihnen Arbeit abzunehmen, indem es eine solide Grundstruktur und Funktionalitäten für die Audioprozessierung und die GUI-Entwicklung bereitstellt.

5.2.2 Bedienoberfläche

Für die Kontrolle über die Prozesse, die bei der Audioverarbeitung durchgeführt werden, benötigt es zunächst einige Kontrollelemente. Dies umfasst eine Reihe an Slidern, Buttons und Auswahl-Menüs, welche alle über das MIDI-Protokoll ansteuerbar sind, um momentane Einstellungen zu speichern, Automationen zu erstellen oder über einen MIDI-Controller anzusteuern. Für die verwendete Sprache, innerhalb der Benutzeroberfläche, wurde Englisch gewählt, um das Programm für eine breitere Zielgruppe verständlich zu gestalten.

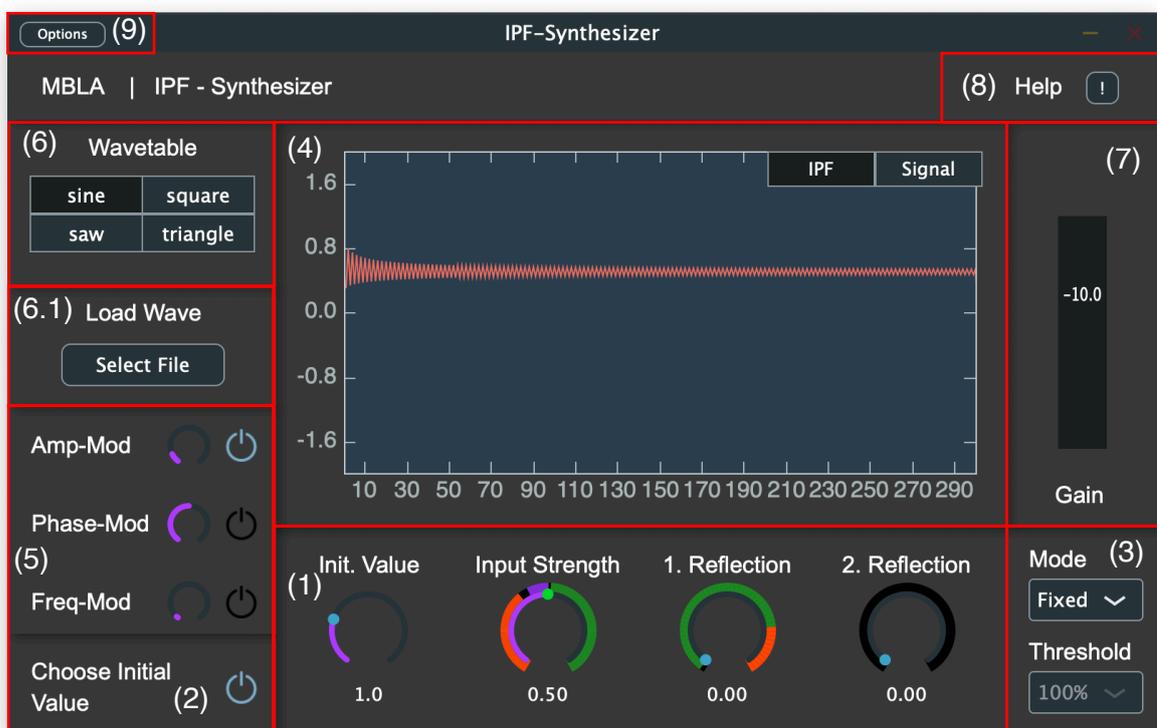


Abbildung 7: Screenshot der IPF-Synthesizer Benutzeroberfläche mit den Basis-Einstellungen

Die Benutzeroberfläche teilt sich in verschiedene Bereiche auf welche jeweils unterschiedliche Funktionen des Synthesizers kontrollieren oder darstellen und werden im Folgenden näher beschrieben.

(1) IPF-Kontrollparameter

In diesem Bereich befinden sich die zentralen Kontrollelemente für die Parameter der IPF. Um die Verständlichkeit zu erhöhen, sind die Parameter α , β und γ als "Input Strength", "1. Reflection" und "2. Reflection" bezeichnet. Zur weiteren Feinabstimmung besteht die Möglichkeit, den Ausgangswert g_0 über den „Init. Value“-Regler zu wählen, abhängig davon, ob die Schaltfläche "Choose Initial Value" aktiviert ist. Wenn diese nicht aktiviert ist, wird g_0 gemäß der Regel $g_0 = 1/\alpha$ festgelegt. Der „Init. Value“-Regler ermöglicht Werte im Bereich von 0 bis 4 in Schritten von 0.1.

Die Wertebereiche der anderen Kontrollparameter aus diesem Bereich sind gemäß den Formeln (4, (5 und (6 festgelegt und passen sich dynamisch den aktuellen Einstellungen an. Der in Abschnitt 3.4.2 vorgestellte Lösungsansatz wird hier durch die farbliche Darstellung des zu erwartenden Verhaltens rund um die Regler veranschaulicht. Grüne Farbe steht für stabiles Verhalten, Lila für bifurkatives Verhalten bis zur 3. Ordnung, Schwarz für chaotisches Verhalten und Rot für ungültige Berechnungen oder divergierendes Verhalten, bei dem kein Ton erzeugt wird.

Die Verwendung dieser Kontrollelemente ermöglicht es den Nutzern, die Einstellungen der IPF präzise anzupassen. Die Benennung der Parameter in verständlicher Sprache, erleichtert die Interpretation der Funktionen. Die Option, den Ausgangswert g_0 anzupassen oder automatisch zu berechnen, bietet Flexibilität in der Handhabung. Die dynamische Anpassung der Wertebereiche und die visuelle Farbcodierung tragen dazu bei, die Auswirkungen der Einstellungen auf das Verhalten des Systems intuitiv zu erfassen, was das Risiko von fehlerhaften Einstellungen minimiert.

(2) Auswahl des Startwerts g_0

In diesem Bereich befindet sich die Option zur Festlegung eines festen Werts für g_0 oder ob dieser Wert von anderen Einflüssen bestimmt werden soll. Bei aktiviertem "Choose Start Value" kann der Wert festgelegt werden. Wenn diese Schaltfläche deaktiviert ist, wird das g_0 -Kontrollelement ausgegraut, kann nicht mehr manuell verändert werden und wird wie bereits erwähnt durch $g_0 = 1/\alpha$ festgelegt.

(3) Dropdown-Menüs zur Auswahl des Spielmodus bzw. der Anzeige der farblichen Darstellung

Der vorgestellte Lösungsansatz aus Abschnitt 3.4.2 beinhaltet eine Auswahlmöglichkeit für die Darstellung des zu erwartenden Verhaltens. Mithilfe des „Mode“-Menüs können diese Darstellungen mit den, in Abbildung 8 dargestellten, Optionen ausgewählt werden. Der "Fixed"-Modus bezieht sich auf den Ansatz für die Anzeige des definitiven Verhaltens, bei welchem die Kontrollparameter während des Spielens unverändert bleiben.

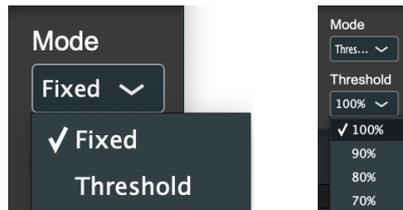


Abbildung 8: Screenshot der Dropdown-Menüs für die Auswahl des Modus und des Schwellenwerts für den Threshold-Modus innerhalb des IPF-Synthesizers

Der "Threshold"-Modus ermöglicht die Realisierung der "Wahrscheinlichkeitsvisualisierung für flexible Anpassungen". Bei Aktivierung dieses Modus kann mithilfe des "Threshold"-Menüs bestimmt werden, welche Häufigkeit des Auftretens ein Verhalten überschreiten muss, um angezeigt zu werden. Die farbliche Darstellung für einen Parameter bezieht sich für beide Modi jeweils auf die aktuellen Einstellungen der anderen beiden Parameter. Bei einer Veränderung von α und β wird also abhängig dieser Werte angezeigt in welchem Bereich γ sich bewegen kann.

(4) Darstellung der Zustandswerte und des modulierten Wavetable-Signals

Hier erfolgt die Visualisierung der IPF gemäß dem Ansatz aus Abschnitt 3.4.1. Die IPF wird über die ersten 300 Iterationen dargestellt und bei jeder Veränderung der Kontrollparameter automatisch aktualisiert. Zusätzlich besteht die Option, im selben Bereich die ersten 64000 Samples des resultierenden Signals mit den Auswirkungen der angewandten Modulationsverfahren anzeigen zu lassen. Die Darstellung erfolgt mittels der "CustomMatplot"-Bibliothek, die eine modifizierte Version der Matplotlib-Bibliothek für die Python-Programmierungsumgebung ist. Diese Bibliothek ermöglicht es, in den Plot hinein- oder herauszuzoomen, um die Darstellungen genauer zu untersuchen. Die visuelle Darstellung der IPF-Zustandswerte sowie des modulierten Wavetable-Signals bietet den Nutzer:innen eine sofortige visuelle Rückmeldung über die Effekte ihrer Einstellungen. Die fortlaufende Aktualisierung bei Änderungen der

Kontrollparameter ermöglicht es den Nutzern, unmittelbar die Auswirkungen ihrer Anpassungen zu beobachten.

(5) Kontrolle über Modulationsverfahren

Hier können die Modulationsverfahren ausgewählt werden. Für die Phasen- oder die Frequenzmodulation besteht nur die Möglichkeit eins von beidem zurzeit zu wählen. Beides gleichzeitig ist nicht möglich, aber beide können mit der Amplitudenmodulation kombiniert werden. Diese Entscheidung wurde getroffen, um die Komplexität zu verringern. Für jedes Modulationsverfahren steht ein Regler zur Verfügung, um den Einflussgrad zu steuern. Die Standardverstärkung beträgt 1, was die reine Form des Modulationsverfahrens darstellt. Falls kein Modulationsverfahren aktiviert ist, wird das unveränderte Wavetable-Signal abgespielt.

(6) Auswahl des Wavetables

Hier kann zwischen verschiedenen Wavetables gewählt werden welche für die Modulation verwendet und als abschließendes Signal ausgegeben werden soll. Es stehen vier Basis-Wavetables zur Auswahl, von denen jedes eine Periode der entsprechenden Signalform, mit 64 diskreten Samples, enthält. Diese umfassen Sinus-, Rechteck-, Sägezahn- und Dreiecksignalformen. Zu einem gegebenen Zeitpunkt kann immer nur ein Wavetable ausgewählt werden, und das aktive Wavetable wird durch Abdunkeln des entsprechenden Feldes angezeigt. Die Möglichkeit, aus verschiedenen Wavetables zu wählen, eröffnet eine breitere Palette an Klangmöglichkeiten, die erzeugt werden können.

(6.1) Möglichkeit eigene Wavetable zu laden und zu verwenden

Des Weiteren bietet sich die Option, eigene Wavetables zu integrieren. Diese Möglichkeit steht jedoch nur zur Verfügung, wenn das Wavetable denselben Anforderungen genügt wie die Basis-Wavetables. Das bedeutet eine Periode eines Signals mit einer Länge von 64 diskreten Samples.

(7) Kontrolle über die Ausgangslautstärke

Zur Feinabstimmung der Gesamtausgangslautstärke des Signals wird ein Gain-Regler mit logarithmischer Skala implementiert.

(8) Hilfe-Button

Um Nutzer:innen über die Farbkodierung und die verschiedenen Modi aufzuklären, wird ein Button implementiert mit welchem sich Informationen für die Bedienungshilfe aufrufen lassen.

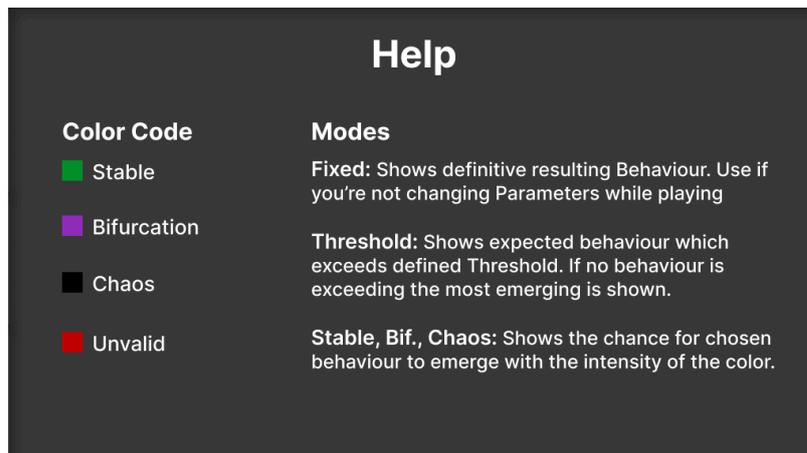


Abbildung 9: Screenshot der angezeigten Bedienungshilfe innerhalb des IPF-Synthesizers

Bei Betätigung erscheint, im selben Bereich wie die Darstellung der Zustandswerte, eine Erklärung zu den Farben und Modi. Während der Hilfebildschirm aktiv ist, wird der Plot ausgeblendet. Diese Funktion unterstützt die Nutzer:innen dabei, die visuellen Darstellungen und die verschiedenen Modi besser zu verstehen und optimal zu nutzen.

(9) Übergeordnete Einstellungen für die Standalone-Verwendung

Beim Kompilieren des Programms als eigenständige Version wird eine zusätzliche Leiste hinzugefügt. Diese Leiste enthält einen Button für erweiterte Optionen, in denen essenzielle Audio- und MIDI-Einstellungen verfügbar sind. Außerdem bieten diese Optionen die Möglichkeit, die aktuelle Konfiguration zu speichern, zu laden oder zurückzusetzen.



Abbildung 10: Screenshot der Standardoptionen des Plugins

Diese Funktion eröffnet auch im Standalone-Betrieb die Möglichkeit, eine Erfahrung zu schaffen, die sich nahtlos an die individuellen Systemeigenschaften und die spezifische Verwendung von verschiedenen Geräten anpasst.

Bei jeder Modifikation der Parameter erfolgt die Übermittlung der aktualisierten Informationen an den Audio-Prozessor. Diese Schritte sind von essenzieller Bedeutung, um die Klangerzeugung zu ermöglichen. Im anschließenden Abschnitt wird dieser Prozess näher erläutert.

5.2.3 Audioverarbeitung

Die Realisierung eines polyphonen Wavetable-Synthesizers erfordert eine Reihe grundlegender Komponenten bei der Audioverarbeitung. Die Kernelemente umfassen eine zentrale Klasse zur Generierung der Wavetables, welche die Basiswellenformen für den Synthesizer bereitstellt. Außerdem eine dedizierte Klasse zur Verwaltung der insgesamt 128 Oszillatoren des Synthesizers. Diese Klasse regelt die Zuweisung und Steuerung der Oszillatoren, um die korrekte Wiedergabe mehrerer gleichzeitiger Töne zu gewährleisten. Und eine Klasse für den individuellen Oszillator, der die Parameter und Verarbeitung für jede Note des Synthesizers steuert. Jeder Oszillator erzeugt Samples basierend auf den zugewiesenen Parametern und dem im Plugin-Editor ausgewählten Wavetable. Die Zusammenarbeit dieser Klassen ermöglicht den Prozess der Sample-Generierung, wobei die erzeugten Samples im Audio-Buffer des Plugin-Prozessors abgelegt und abgespielt werden. MIDI-Noten-Signale werden an entsprechende Klassen weitergeleitet und liefern Informationen über Tonhöhe und Signallänge, um das Instrument spielbar zu machen. Alle einzustellenden Parameter werden von der Plugin-Editor-Klasse übernommen.

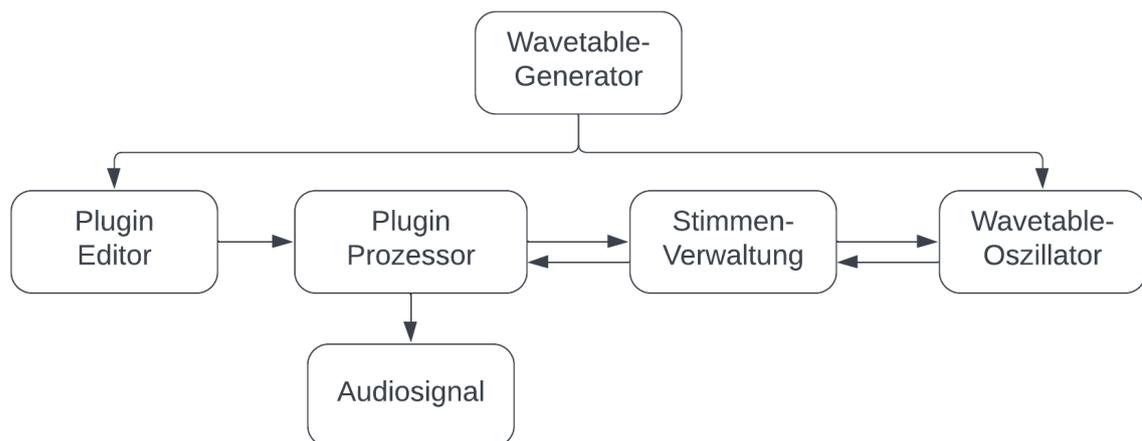


Abbildung 11: Darstellung der grundlegenden Struktur des Programms

Nach der erfolgreichen Umsetzung dieser Konzepte kann die Klasse für die Stimmen-Verwaltung mit weiterführenden Anpassungen unter Verwendung der IPF und verschiedener Modulationsverfahren erweitert werden.

Berechnung der IPF

Die Berechnung der IPF erfolgt individuell für jede Stimme des Synthesizers, um die Polyphonie sowohl im Modulationsbereich als auch bei der Klanggestaltung zu gewährleisten. Hierbei wird die Berechnung nach Erhalt eines MIDI-Notensignals, welches das Spielen eines Tons

anfordert, ausgelöst und erst mit dem Abschluss dieses Befehls beendet. Für den Beginn einer IPF-Berechnung mit mehr als einem Reflexionspunkt ist Vorwissen über vorangegangene Zustände erforderlich. Um diesen Anspruch zu umgehen, werden die zusätzlichen Reflexionspunkte erst in die Berechnung einbezogen, sobald aus der vorherigen Berechnung Werte für die vorherigen Zustände abgeleitet werden können.

Der zeitliche Abstand zwischen den Berechnungsschritten wird durch die Periodenlänge der zu erzeugenden Frequenz bestimmt. Hierbei spielt die Information über die Abtastrate des Systems eine wichtige Rolle, da sie benötigt wird, um zu ermitteln, wie viele diskrete Samples eine einzige Periode des resultierenden Signals umfasst. Nach Festlegung dieser Parameter ermöglicht ein einfacher Zählermechanismus, der bei jeder Berechnung eines Samples inkrementiert wird, die Identifizierung des Zeitpunkts, an dem eine vollständige Periode abgeschlossen ist. Sobald dieser Zustand erreicht ist, wird die darauffolgende Berechnung der IPF durchgeführt, und gleichzeitig wird der Zähler auf den Ausgangswert zurückgesetzt, um den Zyklus erneut zu beginnen.

Amplitudenmodulation

In der Amplitudenmodulation werden gemäß Formel (8) alle Samples innerhalb einer Periode der aktuellen Signalform mit dem gegenwärtigen IPF-Wert verrechnet. Durch die Nutzung des Modulationsreglers in der Benutzeroberfläche erfolgt vor der Verrechnung eine Multiplikation der Zustandswerte mit dem Wert des Reglers. Bei der digitalen Audioverarbeitung ist ein Höchstwert der Amplitude von 1 erlaubt. Jeder Wert, der größer als 1 ist, führt zu einer Übersteuerung, bei der das Signal verzerrt wird. Da die IPF-Werte ebenfalls Werte größer als 1 annehmen können, wird das Signal in diesen Fällen, mit einer Amplitude von 1 für alle überschreitenden Werte, abgeschnitten und verzerrt.

Phasenmodulation

Die Implementierung der Phasenmodulation erfolgt gemäß Formel (9). Sobald ein neuer Zustandswert der IPF erfolgreich berechnet wurde, wird, falls verfügbar, die Differenz von 2 aufeinanderfolgenden Werten ermittelt. Anschließend wird diese Differenz mittels einer Modulo-Operation in den Bereich von 0 bis 360 Grad umgewandelt. Die Verstärkung des Phasenmodulationsreglers sorgt in diesem Fall für eine Multiplikation des Regler-Werts mit der Differenz der Zustandswerte. Ist die Phasenmodulation aktiviert wird für jede Periode ein zusätzliches, phasenverschobenes Signal mit gleicher Periodenlänge hinzugefügt. Zur Vermeidung

möglicher Signalübersteuerungen werden sowohl die Samples des ursprünglichen Signals als auch die des phasenverschobenen Signals durch 2 dividiert.

Frequenzmodulation

Die Frequenzmodulation basiert auf Formel (7 für die Berechnung der Modulationswerte. Hierbei gelten die gleichen Regeln wie bei der Phasenmodulation bei der Ermittlung der Differenz und Verstärkung. Die Resultate aus den Berechnungen werden verwendet, um die aufeinanderfolgenden Schwingungen in der Phasenlänge zu modulieren.

6. Validierung

6.1 Roundtrip

Der Einleitung von Abschnitt 2.3 lässt sich entnehmen, dass es erstrebenswert ist, einen Roundtrip mit einer maximalen Länge von $20ms$ zu erreichen. Ab dieser Grenze wird wohl eine Verzögerung auch für weniger geübte Gehöre wahrnehmbar. Obwohl es aufgrund der in Abschnitt 2.3 genannten Einflüsse nicht möglich ist, eine definitive Zeit für den Roundtrip festzulegen, bieten folgende Messungen dennoch einen ersten Eindruck von der Performance. Folgende Messungen spiegeln daher lediglich ein Abbild für die verwendeten Geräte und Einstellungen dar.

Messaufbau

Für die durchgeführten Messungen wurde ein Kondensator-Mikrofon verwendet, das über ein Steinberg UR22 Audio-Interface mit einem Macbook M1 verbunden war. Das Mikrofon wurde in einer Entfernung von etwa 20 Zentimetern von einem Lautsprecher und einem M-Audio Keystation Mini 32 MIDI-Keyboard aufgestellt. Das Audiosignal für den Lautsprecher wurde über das Interface an einen Denon PMA-520AE Verstärker gesendet. Im Anschluss wurde das Geräusch erfasst, das durch Drücken der äußersten Taste des Keyboards mit einem schnellen Stoß eines Kugelschreibers erzeugt wurde, sowie das resultierende Signal, das aus dem Lautsprecher kam.

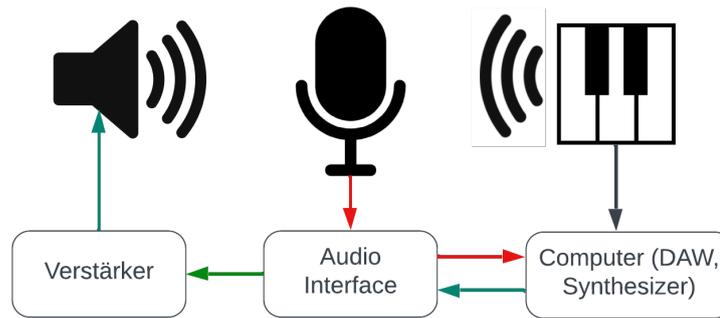


Abbildung 12: Aufbau für die Roundtrip-Messung, Grüne Pfeile sind ausgehende Audiosignale, rote Pfeile sind eingehende Audiosignale und schwarze Pfeile sind MIDI-Signale

Insgesamt wurden für drei verschiedene Szenarien jeweils zehn Aufnahmen mit der Audio-Software Studio One erstellt. Im ersten Szenario wurde keines der Modulationsverfahren aktiviert, was dazu führt, dass nur das reine Signal des Wavetables abgespielt wird. Im zweiten Szenario wurde ausschließlich die Amplitudenmodulation aktiviert. Im dritten Szenario wurde zusätzlich zur Amplitudenmodulation auch die Frequenzmodulation hinzugeschaltet. Das Programm wurde auf eine Audio-Buffer-Größe von 256 Samples und eine Abtastrate von 44,1 kHz eingestellt. Sowohl das Audio-Interface als auch das Aufnahmeprojekt wurden entsprechend diesen Einstellungen angepasst.

Auswertung

Die Zeit, die zwischen dem Drücken der Taste und dem Ertönen des Signals vergeht, lässt sich mit Studio One, Samplegenau bestimmen. Dafür muss der Cursor, wie in Abbildung 13 dargestellt, an die gewünschte Position bewegt werden und für diesen Zeitpunkt wird Millisekunden genau die Zeit angezeigt.

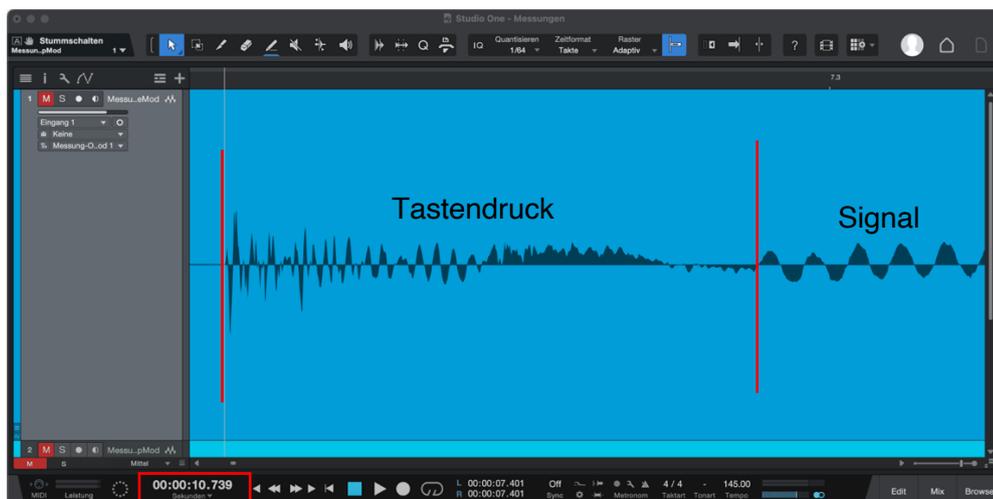


Abbildung 13: Screenshot einer Messaufnahme aus Studio One, Rote vertikale Striche sind Messpunkte, im roten Kasten ist der Zeitpunkt an der Stelle des Cursors angezeigt

Auch wenn ein Punkt Samplegenau bestimmt werden kann, kann es trotzdem zu Abweichungen kommen, wenn nicht ganz klar ist welche Samples noch zu dem Drücken der Taste gehören und wo genau der resultierende Klang aus den Lautsprechern anfängt. Mittels eines Vergleichs von einzelnen Aufnahmen des Tastendrucks ohne ein resultierendes Klangereignis aus den Lautsprechern und dem reinen Klang aus den Lautsprechern können die Grenzen für die Bestimmung also mit hoher Genauigkeit bestimmt werden. Diese Abweichung ist daher also zu vernachlässigen, da Sie keinen maßgeblichen Einfluss auf die Dauer hat. Genauso wie der Abstand des Mikrofons zu dem Lautsprecher und Keyboard. Mit einer Schallgeschwindigkeit von 343m/s benötigt der Schall bei 20 Zentimetern etwa 0,6 Millisekunden bis zum Mikron und beeinflusst die Messungen somit nicht maßgeblich, ist aber trotzdem wichtig zu erwähnen. Im Folgenden sind die durchschnittlichen Zeitdifferenzen zwischen Tastendruck und Ertönen des Signals aufgeführt:

Tabelle 4: Durchschnitt der gemessenen Zeitdifferenzen für 3 verschiedene Fälle mit jeweils 10 Messungen

Einstellungen	Durchschnitt der gemessenen Zeitdifferenzen
Ohne Modulation	17,5 ms
Amplitudenmodulation	18,6 ms
Amplituden- und Frequenzmodulation	17,9 ms

Die Messergebnisse zeigen, dass die Dauer des Roundtrips im Durchschnitt im Bereich von 17,5 bis maximal 18,6 Millisekunden liegt. Diese Werte liegen damit also unter der Grenze von 20 Millisekunden und liegen somit in einem akzeptablen Bereich, um die Anwendung im Live-Betrieb zu verwenden und ein natürliches Spielerlebnis zu gewährleisten.

6.2 Prüfen der Lösungskonzepte

Für die umfassende Validierung der umgesetzten Lösungskonzepte ist ein Vergleich von entscheidender Bedeutung, der die Zuverlässigkeit der Visualisierungen in Bezug auf die tatsächlichen Klangresultate untersucht und das erwartete Verhalten zutreffend abbildet.

6.2.1 Visualisierung im Plot

Im ersten Schritt wurde die Visualisierung des durch die Amplitudenmodulation modulierten Signals innerhalb des Plugins mit 3 unterschiedlichen Einstellungen mit den entsprechenden Aufnahmen innerhalb einer Audio-Software verglichen.

Diese Gegenüberstellung ermöglicht eine sorgfältige Analyse der Visualisierungsergebnisse und deren Genauigkeit im Vergleich zu den realen Klangausgaben. Dabei wurde insbesondere darauf geachtet, ob das modulierte Signal den erwarteten Verlauf zeigte und ob die durch die Parameter gesteuerten Effekte im Plugin mit denen in der Audio-Software übereinstimmten. Für diese Untersuchungen wurde für $g_0 = 1$ verwendet.

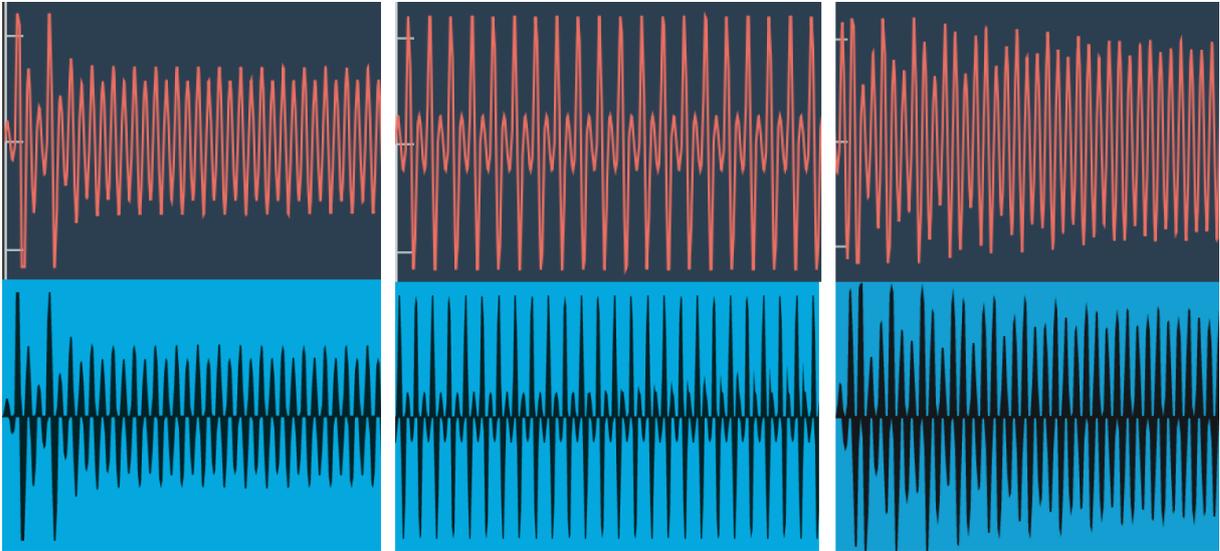


Abbildung 14: Visualisierung der durch die IPF amplitudenmodulierten Signale mit 3 unterschiedlichen Einstellungen, Innerhalb des Plugins (oben), Tatsächliche Resultate einer Aufnahme (unten)

Die in Abbildung 14 ersichtlichen Ergebnisse dieses Vergleichs erwiesen sich als äußerst vielversprechend, da die im Plugin dargestellten Visualisierungen eine hohe Übereinstimmung mit den tatsächlichen Klangresultaten aufwiesen. Die präzise Abbildung der Amplitudenmodulation entsprach den Erwartungen. Dieser Vergleich erfolgte visuell, da dieser Ansatz lediglich einen Eindruck von den Veränderungen vermitteln sollte und nicht auf Sample-Ebene exakt sein muss.

6.2.1 Farbliche Darstellung

Auswertung Fixed-Modus

Im folgenden Schritt erfolgt die Überprüfung der Visualisierung des zu erwartenden Verhaltens. Die erste Auswertung bezieht sich auf den Fixed-Modus, bei dem vorausgesetzt wird, dass die Parameter während des Spielens unverändert bleiben, um ein eindeutiges Verhalten mithilfe der Farbgestaltung um die Kontrollparameter herum zu gewährleisten. Hierzu wurden stichprobenartig Aufnahmen mit einer repräsentativen Auswahl an Kontrollparametern gemacht, von denen aufgrund der visuellen Darstellung erwartet wurde, dass sie zu unterschiedlichen Verhaltensweisen führen. Diese Aufnahmen wurden anschließend mit dem visuell

erwarteten Verhalten verglichen. Bei den Aufnahmen wurde ausschließlich die Amplitudenmodulation verwendet, wobei der Startwert immer auf $g_0 = 1$ festgelegt wurde. Als korrekt dargestellt wird eine Konfiguration eingeordnet, wenn die farblichen Darstellungen, um die Kontrollelemente, bei ihren aktuellen Werten, alle das gleiche zu erwartende Verhalten anzeigen und dieses Verhalten auch in der Aufnahme hörbar und sichtbar wird.

Die Darstellung des zu erwartenden Verhaltens, mit der Bedingung $\beta = 0$, $\gamma = 0$, und variablen Werten für α wurde in einer Stichprobe von 10 unterschiedlichen Einstellungen zu 90 Prozent korrekt wiedergegeben. Es gibt einen Grenzfall bei $\alpha = 0.5$, bei dem eine chaotische Verhaltensweise farblich vorhergesagt wird. Es ist jedoch wichtig anzumerken, dass es sich hierbei um den ersten Bifurkationspunkt handelt. Obwohl in der Theorie eine Bifurkation auch chaotisches Verhalten darstellen kann, wurde in diesem Kontext klargestellt, dass Bifurkationen und chaotisches Verhalten getrennt betrachtet werden sollten. In diesem Fall könnte eine fehlerhafte Zuordnung bei der Stabilitätsanalyse die Ursache für die Abweichung sein.

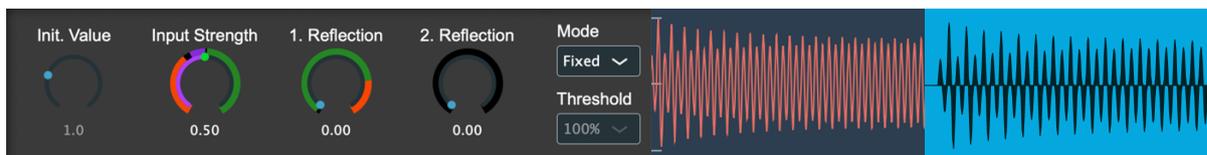


Abbildung 15: Grenzfall der Darstellung des Verhaltens, bei Verwendung von einem Reflexionspunkt. Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme

Bei Einbeziehung von β treten leicht erhöhte Abweichungen auf. In vielen Fällen stimmt die farbliche Darstellung zwar mit einem stabilen Verhalten überein, jedoch gab es auch einige Situationen, in denen ein divergierendes Verhalten vorhergesagt wurde, obwohl ein stabiles Klangverhalten auftrat. Besonders in Situationen, wie bei den in Abbildung 16 dargestellten Einstellungen, wurde festgestellt, dass keines der prognostizierten Verhaltensweisen tatsächlich auftrat. Die farbliche Darstellung gibt entweder ein stabiles oder chaotisches Verhalten an, während in der tatsächlichen Audioaufnahme ein bifurkatives Verhalten hörbar und sichtbar ist. Hierbei könnte eine mögliche Fehlerquelle in der verwendeten Datenreihe der Stabilitätsanalyse liegen oder in einer fehlerhaften Kategorisierung der Zustände.

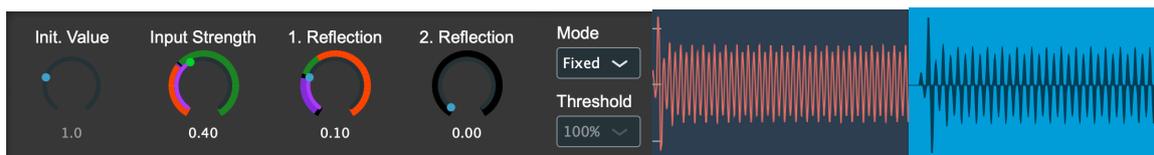


Abbildung 16: Fehlerhafte Darstellung des zu erwartenden Verhaltens bei Verwendung von zwei Reflexionspunkten. Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme

Ähnlich ungenaue Darstellungen traten auch auf, wenn zusätzlich noch γ einbezogen wurde. Ab einem Wert von $\alpha \geq 0.5$ wurden jedoch weniger Probleme bei der Bestimmung der stabilen Zustände festgestellt. In diesem Bereich wurden aber größere Abschnitte, die immer noch zu stabilem Verhalten führten, für den zweiten und dritten Reflexionspunkt irrtümlich rot markiert, was sie als divergierend oder unbrauchbar kennzeichnete. Dieses Phänomen ist in Abbildung 17 sichtbar.

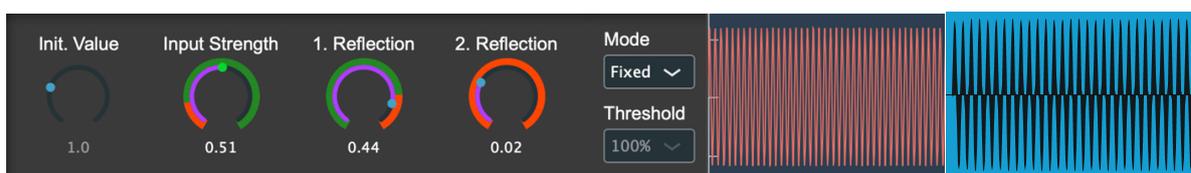


Abbildung 17: Fehlerhafte Darstellung des zu erwartenden Verhaltens in Bezugnahme von β und γ , Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme

Auswertung Threshold-Modus

Für den Threshold-Modus wird eine Analyse der Zuverlässigkeit für der Schwellenwerte durchgeführt. Hierbei werden verschiedene Werte für fünf ausgewählte Parameter-Konfigurationen sowie eine breite Palette von 41 unterschiedlichen Werten für g_0 von 0 bis 4 in Schritten von 0.1 verwendet. Durch die Aufnahme von Klangbeispielen wird überprüft, ob die visuellen Informationen die tatsächlichen Ergebnisse widerspiegeln. Ähnlich wie in der vorherigen Untersuchung werden auch hier die drei Szenarien mit der Verwendung von einem, zwei und drei Reflexionspunkten geprüft. Zusätzlich werden zwei verschiedene Schwellenwert-Grenzen getestet, um die Zuverlässigkeit des Threshold-Modus in verschiedenen Kontexten zu bewerten.

Dier erste Untersuchung mit der Nutzung eines Reflexionspunktes zeigt ähnlich wie im Fixed-Modus eine hohe Zuverlässigkeit. Mit einem Schwellenwert von 80 Prozent ist die farbliche Darstellung für alle 5 getesteten Parameterkonfigurationen korrekt. Auch für den Schwellenwert von 90 Prozent werden die Erwartungen in allen Fällen erfüllt und bieten eine korrekte Darstellung. Die einzige Ausnahme bildet auch hier der Fall $\alpha = 0.5$.

Bei der Anwendung von 2 oder 3 Reflexionspunkten zeigt sich, dass die Verwendung eines niedrigeren Schwellenwerts in Erwägung gezogen werden sollte, da mit zunehmender Anzahl der Reflexionspunkte weniger Zustände die höheren Schwellenwerte überschreiten. Bei Verwendung des Schwellenwerts von 60 Prozent wurde eine recht hohe Erfolgsquote erzielt. In insgesamt 7 verschiedenen Parametereinstellungen wurde das erwartete Verhalten korrekt

abgebildet. Jedoch traten in 3 weiteren Fällen Fehler in der Darstellung auf. Dies deutet darauf hin, dass die Zuverlässigkeit der visuellen Darstellung nicht in vollem Umfang gewährleistet ist. Mögliche Gründe für diese Abweichungen könnten fehlerhafte Kategorisierungen in der Stabilitätsanalyse sein oder Ungenauigkeiten in der Umsetzung des ausführenden Programms selbst. Diese Ergebnisse legen nahe, dass weitere Arbeit erforderlich ist, um auf die farbliche Anzeige als alleinige Informationsquelle vertrauen zu können.

6.3 Erkenntnisse

Die Validierung der entwickelten Lösungskonzepte hat wertvolle Einblicke in die Zuverlässigkeit und Funktionalität des Synthesizers auf Basis der Impulse Pattern Formulation (IPF) geliefert. Die Roundtrip-Messungen bestätigen die Anwendbarkeit des Synthesizers im Live-Betrieb, da die durchschnittlichen Zeitdifferenzen zwischen Tastendruck und Ertönen des Signals im Bereich von 17,5 bis maximal 18,6 Millisekunden liegen und somit unterhalb der gewünschten Grenze von 20 Millisekunden.

Die Auswertung zeigt eine hohe Übereinstimmung zwischen der Darstellung der IPF und des resultierenden Signals im Plot mit den tatsächlichen Klangergebnissen. Für einen Reflexionspunkt funktioniert das Lösungskonzept mit der farblichen Darstellung zwar präzise, jedoch benötigt es gerade bei der Verwendung von mehreren Reflexionspunkten klarere Regeln und eine genaue Überprüfung der korrekten Umsetzung und der verwendeten Datenreihen. Die dynamische Farbanpassung bietet nicht immer klare Hinweise auf das erwartete Verhalten, was die Verlässlichkeit beeinträchtigt.

Die Erkenntnisse nach der Auswertung verdeutlichen die Notwendigkeit einer detaillierten Überarbeitung der Farbgebung und der Anzeigeregeln, um eine zuverlässigere Interpretation des Verhaltens zu ermöglichen. Die Stabilitätsanalyse muss ebenfalls erneut geprüft werden, um sicherzustellen, dass das Verhalten korrekt eingeordnet wird. Die visuellen Informationen müssen als verlässliche Grundlage für die Parameterwahl etabliert und mögliche Fehlerquellen eliminiert werden. Insgesamt sind weitere Schritte erforderlich, um die auf Grundlage der IPF präziser und verlässlicher kontrollieren zu können.

7. Ausblick

Im Hinblick auf eine weiterführende Entwicklung des Synthesizers könnten, neben der Optimierung und Beseitigung der aus den Erkenntnissen entdeckten Probleme, verschiedene Erweiterungen dazu beitragen, sowohl die Vielfalt der Klänge zu erhöhen als auch eine realistische Nachbildung von Instrumenten zu ermöglichen.

Um die realistische Reproduktion physikalischer Klangeigenschaften zu verbessern, wäre es sinnvoll, die Option zur Hinzufügung zusätzlicher Reflexionspunkte zu integrieren. Dadurch ließen sich die Einflüsse der individuellen Subsysteme noch präziser beschreiben. Ebenso könnten durch die Integration weiterer Methoden zur Beeinflussung der Zustandswerte auf die Modulationsverfahren Klänge individueller gestaltet werden. Die Erweiterung um zusätzliche Oszillatoren, die simultan erklingen und individuell einstellbar sind, würde zudem eine deutlich dynamischere Klangpalette ermöglichen. Die Möglichkeit zur nahtlosen Integration eigener Wavetables oder zur präzisen Auswahl bestimmter Bereiche innerhalb eines Wavetables könnte zudem die passende Grundlage für die gezielte Klanggestaltung bieten.

Spannend wäre außerdem die Integration der Anschlagsstärke des MIDI-Keyboards, um den Parameter Alpha zu beeinflussen. Eine gründliche Untersuchung dieser Einflussnahme könnte zeigen, in welchem Maße sie sinnvoll ist. Diese Anpassung könnte die expressiven Möglichkeiten des Synthesizers erweitern, da die Dynamik des Spiels direkten Einfluss auf die Klangentstehung nehmen würde.

Diese skizzierten Möglichkeiten sollen verdeutlichen welches Potenzial zukünftige Erweiterungen haben könnten, um die Leistungsfähigkeit des Synthesizers in Bezug auf Klangkreation und -gestaltung weiter zu steigern.

8. Fazit

Die vorliegende Arbeit konzentrierte sich darauf, einen Synthesizer auf Basis der IPF zu entwickeln und den parametrischen Raum dieses nicht linearen Systems musikalisch spielbar zu gestalten. Diese Zielsetzung wurde konzeptionell und zu Teilen auch in der Praxis erfolgreich umgesetzt.

Die Benutzeroberfläche des Synthesizers ermöglicht eine intuitive Steuerung der IPF-Parameter und Modulationsverfahren. Die Implementierung visueller Darstellungen und Farbcodierungen erleichtert die Interpretation und Anpassung der Einstellungen. Die Ergebnisse der Round-trip-Messungen bestätigen die Anwendbarkeit des Synthesizers im Live-Betrieb, da die gemessenen Verzögerungen innerhalb eines akzeptablen Rahmens liegen.

Trotz dieser Erfolge ist anzumerken, dass weitere Arbeit notwendig ist, um die gesteckten Ziele vollständig zu erreichen. Die praktische Umsetzung der Visualisierung von Parameterbereichen und zu erwartendem Verhalten eines nicht linearen Systems war nicht in allen Aspekten erfolgreich. Die Darstellung für einen Reflexionspunkt erwies sich als funktionstüchtig, jedoch bleibt das Ziel, die Problematik bei mehreren Punkten zu lösen, noch ausstehend. Es ist erkennbar, dass die Umsetzung noch eine verfeinerte Herangehensweise erfordert, um die Zuverlässigkeit und Eindeutigkeit der visuellen Darstellungen zu gewährleisten.

Insgesamt legt diese Arbeit eine solide Grundlage für die weitere Entwicklung des Synthesizers auf Basis der IPF. Während einige Herausforderungen noch überwunden werden müssen, wurde eine vielversprechende Grundlage für die weitere Entwicklung und Darstellung von nicht linearen Systemen geschaffen. Die Arbeit verdeutlicht die enge Verbindung von Wissenschaft und Kunst in der Musikproduktion und eröffnet neue Wege für die musikalische Kreativität.

Abbildungsverzeichnis

Abbildung 1: Bifurkationsszenario der IPF mit einem Reflexionspunkt	8
Abbildung 2: Verhalten der IPF bei Verwendung eines Reflexionspunkts, mit konstantem α und $g_0 = 1$, über 1000 Iterationsschritte a) Stabil: $1/\alpha = 1.9$ b) Bifurkation 1. Ordnung: $1/\alpha = 2.1$ und c) Chaotisches Verhalten: $1/\alpha = 2.65$,	8
Abbildung 3: Möglichkeiten den Roundtrip eines Software-Synthesizers zu messen	12
Abbildung 4: Tim Browns Design Thinking Process	13
Abbildung 5: "Bifurcation scenario of the IPF with two reflection points in dependence of $1/\alpha$ with $\beta = 0.164$. The lower chart (b) is a zoom of the chaotic region."	15
Abbildung 6: Vereinfachte Darstellung des Analyse-Prozesses	21
Abbildung 7: Screenshot der IPF-Synthesizer Benutzeroberfläche mit den Basis-Einstellungen	24
Abbildung 8: Screenshot der Dropdown-Menüs für die Auswahl des Modus und des Schwellenwerts für den Threshold-Modus innerhalb des IPF-Synthesizers	26
Abbildung 9: Screenshot der angezeigten Bedienungshilfe innerhalb des IPF-Synthesizers	28
Abbildung 10: Screenshot der Standardoptionen des Plugins	28
Abbildung 11: Darstellung der grundlegenden Struktur des Programms	29
Abbildung 12: Aufbau für die Roundtrip-Messung, Grüne Pfeile sind ausgehende Audiosignale, rote Pfeile sind eingehende Audiosignale und schwarze Pfeile sind MIDI-Signale	32
Abbildung 13: Screenshot einer Messaufnahme aus Studio One, Rote vertikale Striche sind Messpunkte, im roten Kasten ist der Zeitpunkt an der Stelle des Cursors angezeigt	32
Abbildung 14: Visualisierung der durch die IPF amplitudenmodulierten Signale mit 3 unterschiedlichen Einstellungen, Innerhalb des Plugins (oben), Tatsächliche Resultate einer Aufnahme (unten)	34
Abbildung 15: Grenzfall der Darstellung des Verhaltens, bei Verwendung von einem Reflexionspunkt. Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme	35
Abbildung 16: Fehlerhafte Darstellung des zu erwartenden Verhaltens bei Verwendung von zwei Reflexionspunkten. Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme	35
Abbildung 17: Fehlerhafte Darstellung des zu erwartenden Verhaltens in Bezugnahme von β und γ , Links: Einstellungen des Plugins, Mitte: Vorschau des Signals im Plugin, Rechts: Tatsächliche Aufnahme	36

Quellenverzeichnis

Bader, Rolf. 2013. „Impulse Pattern Formulation“. In *Nonlinearities and Synchronization in Musical Acoustics and Music Psychology, Current Research in Systematic Musicology*, hrsg. Rolf Bader. Berlin, Heidelberg: Springer: 285–96. https://doi.org/10.1007/978-3-642-36098-5_8 (12. Juli 2023).

Brown, Tim. 2008. „Design Thinking“. *Harvard Business Review* 86(6): 84–92.

von Helmholtz, H. 1889. *Über die Erhaltung der Kraft*. W. Engelmann. <https://books.google.de/books?id=qu44AAAAMAAJ>.

„Latenz der Audiokarte – Steinberg Support“. <https://helpcenter.steinberg.de/hc/de/articles/206111444-Latenz-der-Audiokarte> (9. August 2023).

Linke, Simon, Rolf Bader, und Robert Mores. 2019a. „The Impulse Pattern Formulation (IPF) as a Model of Musical Instruments—Investigation of Stability and Limits“. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 29(10): 103–9.

Linke, Simon, Rolf Bader, und Robert Mores. 2019b. „The Impulse Pattern Formulation (IPF) as a Nonlinear Model of Musical Instruments“.

Logan, Mark. 2018. „Design Thinking for Entrepreneurs“. *idealect*. <https://medium.com/idealect/design-thinking-for-entrepreneurs-392c8cbdcc24> (12. Juli 2023).

Nielsen, Jakob. 2020. „10 Usability Heuristics for User Interface Design“. Nielsen Norman Group. <https://www.nngroup.com/articles/ten-usability-heuristics/> (12. Juli 2023).

Sullivan, John, Catherine Guastavino, und Marcelo M. Wanderley. 2021. „Surveying Digital Musical Instrument Use in Active Practice“. *Journal of New Music Research* 50(5): 469–86.

Anhang

Auf dem USB-Stick im Anhang befinden sich folgende Dateien:

1. VST-Plugin des IPF-Synthesizers
Dateiname: IPF-Synthesizer-Mac.vst3
2. AU-Plugin des IPF-Synthesizers
Dateiname: IPF-Synthesizer-Mac.component
3. Standalone-Plugin des IPF-Synthesizers für Mac
Dateiname: IPF-Synthesizer-Mac.app
4. Standalone-Plugin des IPF-Synthesizers für Windows
Dateiname: IPF-Synthesizer-Mac.exe
5. Projektordner des Programms inkl. Sourcecode und verwendeten Datenreihen
Dateiname: IPF-Synthesizer-Projektordner.zip

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit mit dem Titel:

Synthesizer auf Grundlage der Impulse Pattern Formulation (IPF) - Den parametrischen Raum musikalisch spielbar gestalten

selbständig und nur mit den angegebenen Hilfsmitteln verfasst habe. Alle Passagen, die ich wörtlich aus der Literatur oder aus anderen Quellen wie z. B. Internetseiten übernommen habe, habe ich deutlich als Zitat mit Angabe der Quelle kenntlich gemacht.

09. August 2023

Datum

Unterschrift