

Was sind die mathematischen Grundlagen moderner elektronischer Musik? - Die Fourier- Reihen und -Synthese

Verfasser: **Sören Richter** <soeren@dreieck-project.de>

Datum: **16.03.2017**

„Facharbeit im Seminarfach Entwicklung und Bedeutung der Mathematik im
Laufe der Jahrhunderte – berühmte Mathematiker, wichtige Erkenntnisse und
Anwendungen im Alltag“

Kaiserin-Auguste-Viktoria-Gymnasium Celle

Schuljahr 2016/2017

Inhaltsverzeichnis

1	Einleitung	4
2	Grundkonzepte und Hintergründe	4
2.1	Schall, Klänge und das Gehör.....	5
2.2	Schall digital speichern – das „Pulse-Code-Modulation“-Verfahren	5
2.3	Tonhöhen und deren Berechnung	7
2.4	Parameter der Sinus- und Kosinus-Funktion	8
3	Fourier-Reihen und die Additive Klangsynthese	9
3.1	Allgemeine Fourier-Reihe und der Fourier-Satz	9
3.2	Additive Klangsynthese	10
3.3	Wichtige Wellenformen und deren Fourier-Reihen	10
3.3.1	Sägezahn-Schwingung	10
3.3.2	Rechteck-Schwingung	11
3.3.3	Dreieck-Schwingung	12
3.4	„Vom Klang zur Reihe“: Die schnelle Fourier-Transformation (FFT) ..	12
3.5	Vor- und Nachteile der Fourier-Reihen	13
3.5.1	Gibbs-Phänomen	14
3.6	Klangformung – die ADSR-Hüllkurve.....	14
4	Vollständige Synthese eines Klangbeispiels	16
5	Fazit	17
6	Literaturverzeichnis	19
6.1	Printmedien	19
6.2	Internetquellen	19
7	Abbildungsverzeichnis	20
8	Anhang	21
8.1	Java-Umsetzung eines Synthesizers - Klassenübersicht.....	21
8.1.1	Die mainWindow-Klasse.....	21
8.1.2	Die Synth-Klasse	21

8.1.3	Die Oscillator-Klasse.....	22
8.1.4	Die Envelope-Klasse	22
8.1.5	Die Pitch-Klasse	23
8.1.6	Letzte Änderungen und bekannte Probleme.....	23
8.2	GeoGebra-Datei zur Darstellung der Hüllkurve mit Parametern	23

1 Einleitung

Der heutzutage gängige Weg zur Erstellung, Verarbeitung und Wiedergabe von Bild und Ton ist der Digitale. Information, Kommunikation und Emotion laufen in großem Maße über das präsenteste Datennetz der Welt, natürlich ausschließlich digital, mit Computern jeder Größe als Anbindung und Übersetzer der Signale, welche über Bildschirme und Lautsprecher in reale Sinneswahrnehmungen gewandelt werden. Einer der größten Vorteile ist, dass jeder der unzähligen Internetnutzer selbst zum Erzeuger neuer Inhalte werden kann – mit vergleichsweise geringem Aufwand kann jeder zur Berühmtheit in einem Videoportal oder auf einer Musik-Streaming Website werden.

Diese Facharbeit soll einen Einblick in die Funktionsweise von digitalen Synthesizern geben – jene Geräte und Programme, die Töne erzeugen, Klänge formen, sogar ganze Orchester nachahmen können und integraler Bestandteil der Produktion moderner Musik verschiedenster Genres sind. Gegenstand der Erörterung sind dabei die Grundkonzepte der digitalen Signalverarbeitung, hier die Speicherung von Schall mit dem „Pulse-Code-Modulation“-Verfahren (PCM), die mathematischen Grundlagen der additiven Klangsynthese, ausgegangen von den Fourier-Reihen des französischen Mathematikers Joseph Fourier (1768-1830)¹, und der scheinbare Widerspruch der Verbindung rationaler Mathematik, Physik und Elektronik mit der emotionalen Welt der Musik.

Das Ziel dieser Facharbeit ist die verständliche Darstellung der Zusammenhänge, darunter die Berechnung von digitalen akustischen Signalen und die klanglichen Auswirkungen der Anwendung bestimmter Fourier-Reihen und Hüllkurven zur Formung eines Klanges – abschließend zusammengefasst in der Darstellung einer vollständigen Synthese eines Klangbeispiels und der Übertragung der Konzepte in eine Java-Programmierung.

2 Grundkonzepte und Hintergründe

Um die Grundlage einer Facharbeit zu schaffen, die das Ziel der Erzeugung von Schall und Klängen hat, ist es essentiell zu wissen, was Schall im physikalischen Raum überhaupt ist, wie dieser digitalisiert werden kann und was die Eigenschaften und Grenzen unseres Gehörs sind.

¹ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 305.

2.1 Schall, Klänge und das Gehör

Wenn wir einen Ton über den Hörsinn wahrnehmen, dann legt dieser einen Weg zurück, der im Normalfall immer konzeptuell identisch ist. Eine gespannte Saite, Membran oder Oberfläche wird durch eine physikalische Kraft in Schwingung versetzt und gibt diese, oft durch einen Resonanzkörper verstärkt, an die Luft weiter. Auch diese „schwingt“ nun vereinfacht gesagt auf zumindest kurzem Wege gleichartig. Im näheren Sinne wurden also Druckunterschiede pro Zeiteinheit erzeugt, welche sich im Raum ausbreiten und schlussendlich auch das Trommelfell im menschlichen Ohr in Schwingung versetzen.² Die Abstandsdifferenz des schwingenden Mediums an einem Zeitpunkt zu dessen unbewegtem Zustand, also dessen Ausschlag, wird Amplitude genannt.³ Somit lässt sich eine Schallwelle auch mittels eines Zeit-Amplituden-Diagramms/Graphen graphisch darstellen.

Unterdessen bietet auch unser Gehör einige spannende mathematische Eigenschaften. So sind Amplituden- und Frequenzwahrnehmung nicht linear und relativ stark eingeschränkt. Der hörbare Frequenzbereich liegt zwischen etwa 16 Hertz und 20 Kilohertz.⁴ Der Tonhöhereindruck verläuft dabei vor allem in den etwas tieferen Lagen nahezu logarithmisch.⁵ Tiefere Frequenzen wie etwa von 100 bis 2000 Hertz können daher wesentlich einfacher voneinander unterschieden werden als jene im Bereich von 10 bis 20 Kilohertz, obwohl der Wertebereich deutlich größer ist.

2.2 Schall digital speichern – das „Pulse-Code-Modulation“-Verfahren

Um akustische Signale, welche physikalisch nur als hörbare Schallwellen existieren, auch digital mittels eines Computers zu speichern, ist ein Verfahren notwendig, welches sowohl die Grundvoraussetzungen eines Algorithmus erfüllt, als auch im Rahmen der Möglichkeiten der Technik liegt. Daraus folgt, dass das Speicherverfahren endlich sein muss, also nur eine begrenzte Menge von Schritten durchlaufen kann, genau definiert und somit theoretisch rekonstruierbar sein muss und nur mit Daten endlicher Genauigkeit rechnen kann, da Computerprozessoren auf bestimmte Wortgrößen bei der Signalverarbeitung beschränkt sind und der verwendete Speicherplatz begrenzt ist.⁶

² Vgl. SAUER, Der Aufbau des Ohres.

³ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 121.

⁴ Vgl. SAUER, Der Aufbau des Ohres.

⁵ Vgl. MÖSER, Wahrnehmung von Schall, S. 3-5.

⁶ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 298-300, 349-353.

Aus dem vorherigen Abschnitt ist bekannt, dass sich Schall als Zeit-Amplituden-Graph darstellen lässt. Dies gibt uns die Möglichkeit, gerundete Amplitudenwerte wiederholt zu Zeitpunkten eines bestimmten Intervalls abzulesen. Dieses Verfahren wird Quantisierung genannt und mittels der „Pulse-Code-Modulation“ (PCM) digital umgesetzt.⁷

Hier werden die einzelnen Amplitudenwerte der zu speichernden Schallwelle, die sogenannten Abtastwerte oder englisch Samples, mittels eines Analog-Digital-Wandlers, welcher die elektrischen Signale eines Mikrofons oder analogen Synthesizers umwandelt, gemessen oder mit digitalen Synthesizern berechnet und danach auf die Genauigkeit der Wortgröße, welche als Raster für die Amplitudenwerte angesehen werden kann, im Näheren bei PCM die Anzahl der verwendeten Bits pro Sample, gerundet.⁸

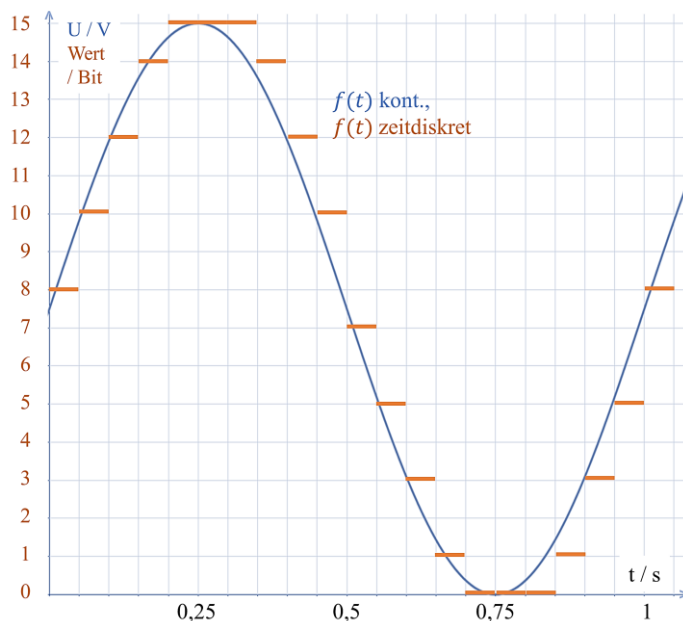


Abbildung 1: Quantisierung

Dieser Schritt des Verfahrens wird als Abtastung⁹ bezeichnet und muss regelmäßig in gleichen Zeitabständen geschehen, damit eine zusätzliche Speicherung der Zeitwerte eingespart werden kann, da diese somit nur noch gezählt werden müssen, und die Schallwelle wesentlich genauer gerastert wird und somit auch die

Rekonstruktion beziehungsweise Rückwandlung zu analogen Signalen, die das ursprüngliche Signal annähern, effizienter ist. Die Menge der Abtastungen pro Sekunde wird als Abtastrate beziehungsweise Samplerate bezeichnet und in der Einheit Hertz angegeben.¹⁰ Abbildung 1 zeigt beispielhaft die PCM-Quantisierung einer Sinus-Kurve der Frequenz 1 Hz mit einer Samplerate von 20 Hz und 4 Bit pro Sample.

⁷ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 352 – 354.

⁸ Vgl. ebd., S. 349 – 354.

⁹ Vgl. ebd., S. 350.

¹⁰ Vgl. ebd.

In der Signalverarbeitung wird das Konzept dieser Quantisierung auch als Umwandlung eines kontinuierlichen Signals, wie jenem einer Sinusfunktion mit einer unendlichen Menge an Funktionswerten oder dem einer in analoge elektrische Impulse gewandelten Schallwelle, zu einem zeitdiskreten Signal, hier der PCM-Datenstrom, bezeichnet.¹¹

Doch wie viele Samples pro Sekunde sind nötig, um den menschlich hörbaren Frequenzbereich bei der Speicherung akustischer Inhalte abzudecken? Diese Frage beantwortet das Shannon-Nyquist-Theorem, welches aussagt, dass die halbe Frequenz der Abtastrate grundsätzlich die Grenzfrequenz der PCM-Quantisierung ist¹², da darüber liegende Frequenzen zu kurze Wellenlängen für den Zeitabstand der Samples haben und somit keine eindeutige Rückwandlung mehr gegeben ist; es zeigt sich das Problem des sogenannten Aliasing.¹³ Doch leider tritt dieses Problem auch bereits bei Frequenzen unter der angenommenen Grenzfrequenz auf, da diese derart asynchron zur Abtastrate liegen, dass zusätzlich eine Überlagerungsfrequenz entsteht, wenn diese Schwingungen gerastert werden. In der Praxis werden diese unerwünschten Nebeneffekte unter Anwendung der gewünschten Grenzfrequenz $\cdot 2.2 =$ Abtastrate so gering wie möglich im hörbaren Frequenzbereich bis 20 kHz gehalten, woraus sich eine Abtastrate von etwa 44 kHz für die vollständig transparente Speicherung von zum Beispiel Musik, bei welcher eine hohe Qualität gefordert wird, ergibt. Eine herkömmliche CD als digitaler Tondatenträger speichert die akustischen Signale mittels PCM und verwendet ähnlich der oberen Annahme tatsächlich eine Abtastrate von 44,1 kHz mit 16 Bits pro Sample.¹⁴ Mit 2 Kanälen ergibt sich damit eine Datenmenge von 2 mal 16 Bits pro Sample mal 44100 Samples pro Sekunde = 1411200 Bits pro Sekunde, welche häufig in der alltäglichen Anwendung der Speicherung und Übertragung von Musik und Sprache viel zu groß ist und einen entscheidenden Nachteil der alleinigen PCM darstellt, welcher später durch die schnelle Fourier-Transformation in Abschnitt 3.4 gelöst werden kann.

2.3 Tonhöhen und deren Berechnung

Eine weitere wichtige Grundlage, um Klänge zur Verwendung in der Musik mittels eines Computers zu generieren, ist die Berechnung der Frequenzen der Töne der

¹¹ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 349-350.

¹² Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 817-818.

¹³ Vgl. ebd., S. 824-825.

¹⁴ Vgl. OSTA, Audio Recording.

gleichstufigen Stimmung, nach welcher unter anderem auch Klaviere und Gitarren heutzutage gestimmt werden. Der Kammerton a^1 ist auf 440 Hz festgelegt.¹⁵ Mit jeder Oktave verdoppelt sich dieser Wert, da diese die harmonischen Oberschwingungen von 440 Hz beziehungsweise a^1 sind. Doch wie werden die dazwischenliegenden Tonhöhen berechnet? Die Tonleiter besteht innerhalb einer Oktave aus 12 Halbtonschritten.¹⁶ Sowohl die Verdopplung der Frequenz bei einem Oktavintervall und die etwa logarithmische Frequenzwahrnehmung des menschlichen Gehörs lassen für die Tonhöhen einzelner Tasten der Klaviatur einen exponentiellen Zusammenhang vermuten. Eine Funktion $f(n)$ / Hz muss also folgendes erfüllen: $f(0) = 1 \cdot 440$ Hz, $f(12) = 2 \cdot 440$ Hz, $f(24) = 4 \cdot 440$ Hz... Die Exponentialfunktion zur Berechnung der Frequenzen aller Tonstufen der Gleichstufigen Stimmung sieht demnach so aus:

$$f(n) = 2^{\frac{n}{12}} \cdot 440 \text{ Hz}$$

mit $a^1 \hat{=} n = 0$, n in Halbtonschritten¹⁷.

2.4 Parameter der Sinus- und Kosinus-Funktion

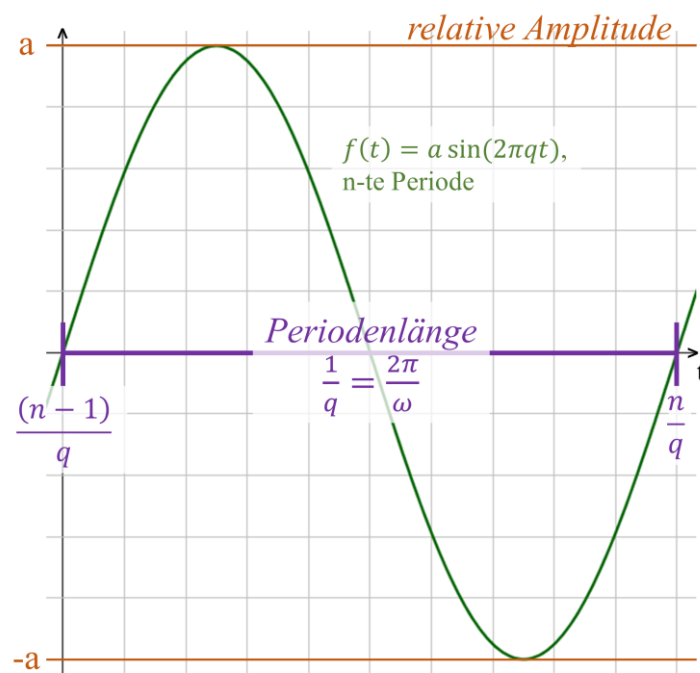


Abbildung 2: Sinus-Parameter

Um eine Sinus-Funktion zu bestimmen, die eine Schwingung gewünschter Frequenz in der Einheit Hertz repräsentiert und eine bestimmte Lautstärke im Verhältnis zur Verstärkung hat, welche durch die relative Amplitude bestimmt wird, sind jeweilige Parameter und der Ausgleich der

Periodenlänge

trigonometrischer Funktionen von 2π notwendig. Ähnlich der aus dem Mathematikunterricht bekannten Faktoren zur Stauchung beziehungsweise

¹⁵ Vgl. HERRMANN, Mathematik und Gott und die Welt, S. 52-53.

¹⁶ Vgl. ebd.

¹⁷ Vgl. ebd.

Streckung einer (Ko-)Sinus-Periode ergibt sich unter der Betrachtung, dass eine Periode einer vollständigen Schwingung entspricht, also bei 1 Hz (einer Schwingung pro Sekunde) mit t in s die Periodenlänge genau 1 sein soll, folgender Funktionsterm einer Sinus-Funktion:

$$f(t) = a \cdot \sin(2\pi qt)$$

Hier beschreibt der Vorfaktor a die relative Amplitude der Schwingung und q die Frequenz in Hertz aufgrund des Vorfaktors 2π , da $q = \frac{\omega}{2\pi}$ (Winkelgeschwindigkeit beziehungsweise Kreisfrequenz durch $2\pi =$ Schwingungsfrequenz in Hertz)¹⁸. Sinus- beziehungsweise Kosinus-Funktionen mit dem ganzzahligen Vielfachen $n \cdot q$ einer bestimmten Frequenz werden als harmonische Oberschwingungen n -ten Grads bezeichnet.¹⁹ Ein Vorzeichenwechsel entspricht einer Phasenumkehrung. Die Parameter sollten nicht 0 sein, da $f(t) = 0$ für alle t und eine Periode nicht existent wäre. Die Auswirkungen der Parameter der Sinusfunktion werden in Abbildung 2 graphisch verdeutlicht. Zu erkennen ist, dass die Periodenlänge der Kehrwert der Frequenz ist. Daher werden Schwingungen niedriger Frequenzen auch langwellig und jene höherer Frequenzen kurzwellig genannt.

3 Fourier-Reihen und die Additive Klangsynthese

3.1 Allgemeine Fourier-Reihe und der Fourier-Satz

Die Grundidee, der nach dem französischen Mathematiker Joseph Fourier benannten „Fourier-Reihen“, ist die Annahme, dass sich jedes beliebig aussehende periodische Signal als Summe einer Sinus- oder Kosinus-Funktion mit unendlich vielen bestimmten harmonischen Oberschwingungen schreiben lässt.²⁰ Diese mathematische Entdeckung wird auch als Fourier-Theorem bezeichnet. Die allgemeine Schreibweise der Fourier-Reihe²¹ beinhaltet die zwei von der Summen-Zählvariable k abhängigen Koeffizienten a_k und b_k , welche mittels partieller Integration²² einer Funktion bestimmt werden können.

$$f(t) = a_0 + \sum_{k=1}^{\infty} (a_k \cos(kt) + b_k \sin(kt))$$

¹⁸ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 121.

¹⁹ Vgl. BADER, Physik Gymnasium, S. 136-137.

²⁰ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 305.

²¹ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 703.

²² Vgl. ebd., S. 705.

Unterschieden wird zwischen geraden Funktionen $f(t)$, die eine Kosinus-Reihe ergeben ($b_k = 0$), da ihr Integral von $-\pi$ bis π ungleich 0 ist, und ungeraden Funktionen $f(t)$, die eine Sinus-Reihe ergeben ($a_k = 0$), mit dem Integral von $-\pi$ bis $\pi = 0$.²³

3.2 Additive Klangsynthese

Die sogenannte Additive Klangsynthese²⁴ nutzt das Prinzip der Überlagerung von Sinus- beziehungsweise Kosinus-Schwingungen mit bestimmten Anteilen ihrer harmonischen Oberschwingungen, um verschiedenartige neue periodische Schwingungen zu erhalten. Die Summenschreibweise von Fourier-Reihen aus Abschnitt 3.1 beschreibt genau diesen Vorgang, indem die Grundschwingung der Frequenz q mit ihren Oberschwingungen k von den Frequenzen $q \cdot k$ überlagert (addiert) werden, um eine bestimmte periodische Wellenform anzunähern.

3.3 Wichtige Wellenformen und deren Fourier-Reihen

Wie auch die Sinus-Funktion besitzen die drei folgenden Funktionsmodelle und somit auch ihre Annäherungen durch Fourier-Reihen besondere Eigenschaften und einen physikalischen Hintergrund, welche diese besonders geeignet für die additive Klangsynthese machen, ihnen aber auch einen historischen Kontext geben. Im Folgenden sind die Funktionsdefinitionen und Summenschreibweisen der Fourier-Reihen vereinfacht und an die Umsetzung in der Programmierung angepasst sowie die Parameter für Amplitude und Frequenz rotfarbig ergänzt. Außerdem wurde der Wertebereich grundsätzlich auf $[-1,1]$ begrenzt.

3.3.1 Sägezahn-Schwingung

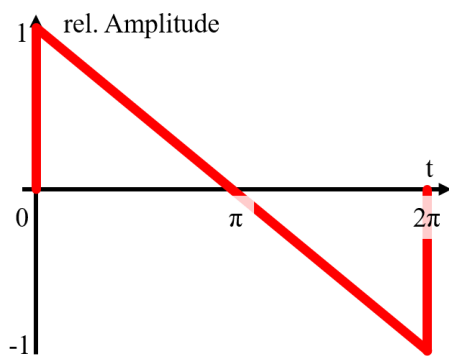


Abbildung 3: Sägezahn-Funktion

Ursprünglich durch die klassische Kippschaltung entstanden, wird die (fallende) Sägezahn-Funktion (Sawtooth-Waveform), siehe Abbildung 3, definiert als

$$saw(t) := \begin{cases} \frac{-t+\pi}{\pi}, & 0 < t < 2\pi, \\ 0, & t = 2\pi. \end{cases} \quad 2\pi\text{-}$$

periodisch²⁵ mit der Amplitude 1. Damit ergibt sich der theoretische Funktionsgraph

einer sich periodisch wiederholenden linearen Funktion mit regelmäßigen

²³ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 704.

²⁴ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 305.

²⁵ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 705.

Sprungstellen an Vielfachen von 2π . Perioden sind demnach an ihren Nullstellen punktsymmetrisch. Dieses Aussehen gab der Funktion ihren Namen, da die einzelnen Perioden wie scharfe Sägezähne wirken. Zur Annäherung dieses Erscheinungsbildes wird die Sinus-Grundschiwingung mit dem k -ten Anteil der k -ten Oberschiwingung überlagert. Daraus ergibt sich folgende Fourier-Reihe²⁶:

$$\text{saw}(t) = a \cdot \frac{2}{\pi} \cdot \sum_{k=1}^{\infty} \frac{\sin(2\pi \cdot q \cdot k \cdot t)}{k}$$

Auch klanglich kann der Sägezahnwellen-Bezeichnung durchaus eine Berechtigung zugesprochen werden, da der entstehende Ton äußerst stechend und metallartig klingt und üblicherweise als unangenehm empfunden wird.

3.3.2 Rechteck-Schwiingung

Aus regelmäßig wiederholten Spannungsstößen gleicher Länge entstand die Rechteck-Spannung, welche zu der Funktionsdefinition für die Rechteck-Schwiingung (Square-Waveform, Abbildung

4) führte:
$$sq(t) := \begin{cases} 1, & 0 < t < \pi, \\ 0, & t = 0, t = \pi, 2\pi- \\ -1, & \pi < t < 2\pi. \end{cases}$$

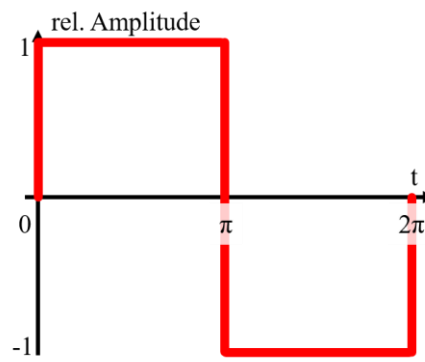


Abbildung 4: Rechteck-Funktion

periodisch²⁷ mit der Amplitude 1. Daraus

ergibt sich ein theoretischer Funktionsgraph bestehend aus π -breiten Rechtecken, gebildet aus dem Funktionswert von 1 mit wechselndem Vorzeichen sowie Sprungstellen an jedem Vielfachen von π , und erneut einer Punktsymmetrie an jeder Nullstelle der Funktion. Angenähert wird diese Gestalt mit jeder ungeraden Oberschiwingung einer Sinus-Grundschiwingung im Anteil ihrer Ordnung, woraus sich auf diese Fourier-Reihe schließen lässt²⁸:

$$sq(t) = a \cdot \frac{4}{\pi} \cdot \sum_{k=1}^{\infty} \frac{\sin(2\pi \cdot q \cdot (2k - 1) \cdot t)}{(2k - 1)}$$

Der Klang der Rechteck-Schwiingung lässt sich als deutlich angenehmer und weicher als jener der Sägezahn-Schwiingung charakterisieren und erinnert an frühere Einsatzgebiete dieser Wellenform, vor allem in historischen Videospieleen oder der ikonischen Vertonung unter anderem von Science-Fiction Robotern.

²⁶ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 705.

²⁷ Vgl. ebd., S. 734.

²⁸ Vgl. ebd., S. 735.

3.3.3 Dreieck-Schwingung

Die Dreieck-Schwingung (Triangle-Waveform, siehe Abbildung 5) repräsentiert eine periodische Funktion mit abwechselnd steigenden und fallenden linearen Abschnitten, welche achsen- und punktsymmetrische Dreiecksformen ergeben:

$$tr(t) := \begin{cases} \frac{-2t+\pi}{\pi}, & 0 \leq t < \pi, \\ \frac{2t-3\pi}{\pi}, & \pi \leq t < 2\pi. \end{cases} \quad 2\pi\text{-periodisch}^{29} \text{ mit der Amplitude 1. Die Fourier-}$$

Reihe lässt sich diesmal mit dem Kosinus deutlich einfacher darstellen³⁰:

$$tr(t) = a \cdot \frac{8}{\pi^2} \cdot \sum_{k=1}^{\infty} \frac{\cos(2\pi \cdot q \cdot (2k-1) \cdot t)}{(2k-1)^2}$$

Der Klang der Dreiecksfunktion kann als wesentlich angenehmer und deutlich dumpfer als jene der anderen Wellenformen wahrgenommen werden, was bereits im Aussehen der Schwingung erkennbar ist, da nun keine großen Sprungstellen vorhanden sind, die einen scharfen Klang erzeugen. Die Klangfarbe ist je nach subjektivem Eindruck möglicherweise vergleichbar mit der einer Glocke, die dauerhaft klingen würde.

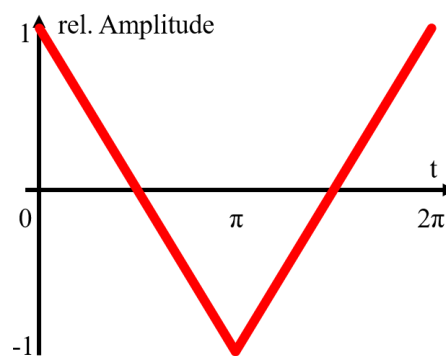


Abbildung 5: Dreieck-Funktion

3.4 „Vom Klang zur Reihe“: Die schnelle Fourier-Transformation (FFT)

Der mathematische Weg, um von jeglicher Schwingung entsprechende Fourier-Reihen zu berechnen, wird Fourier-Transformation genannt. Für zeitdiskrete Anwendungen existiert zudem die Diskrete Fourier-Transformation (DFT) und vor allem im Computer-Bereich die schnelle Fourier-Transformation (Fast Fourier Transform, FFT).³¹ Diese „zerlegt“ einen PCM-Datenstrom in die einzelnen Frequenzbestandteile, formt also aus den zeitdiskreten Werten eine Fourier-Reihe, welche die Wellenform vereinfacht im Frequenzspektrum darstellt.³²

Verwendet wird die schnelle Fourier-Transformation beispielsweise bei der Visualisierung von Schall in einem sogenannten Spektrogramm, welches den

²⁹ Dreieck-Definition eigenständig hergeleitet.

³⁰ Die Dreiecksfunktion entspricht die Stammfunktion der Rechteckfunktion.

³¹ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 785-788, 800-801.

³² Vgl. BADER, Physik Gymnasium, S. 142-145.

Frequenz-Amplituden-Zeit-Zusammenhang eines Signals graphisch darstellt. Effekte zur Veränderung von Tonhöhe, Geschwindigkeit oder Amplitude einzelner Frequenzbereiche von Audiodaten verwenden auch die FFT mit entsprechend im Zeit- oder Frequenzbereich veränderten Fourier-Reihen zur Rücktransformation.³³ In der Musikproduktion wird diese Möglichkeit oft für das „Sampling“ und die Synthese reeller Instrumentalklänge, etwa von ganzen Orchestern, verwendet oder als Möglichkeit genutzt, schiefen Gesang in die richtige Tonlage zu bringen. Außerdem kann der große Nachteil riesiger Datenmengen der Pulse-Code-Modulation mit der FFT gelöst werden. Sogenannte verlustbehaftete Kompressionsverfahren streichen mithilfe des psychoakustischen Modells viele nicht hörbare Frequenzbestandteile aus dem Spektralbereich einer Schallwelle und speichern die somit deutlich kleineren, jedoch gelegentlich qualitativ beeinflussten, Fourier-Reihen. Allgemein bekannt ist die Anwendung dieses Verfahrens im MP3-Format aber auch im JPG- und MP4-Format da sich die FFT in sehr ähnlicher Weise auch auf Bilddaten anwenden lässt.³⁴

3.5 Vor- und Nachteile der Fourier-Reihen

Eine andere Perspektive auf dieses Thema bietet sich bezüglich der Vor- und Nachteile der Fourier-Reihen im Vergleich zu den Funktionsdefinitionen, insbesondere mit Verweis auf die spätere algorithmische Umsetzung der herkömmlichen Synthesizer-Schwingungen, an.

Grundsätzlich sollte die Verwendung von Fourier-Reihen bevorzugt werden, da jedes Instrument und jede erdenkliche Wellenform angenähert werden können. Die Nutzung der Fourier-Transformation gibt einem digitalen Synthesizer eine äußerst hohe Effizienz bei der Berechnung, Formung und Veränderung von Klängen, beispielsweise in Tonhöhe und Geschwindigkeit. Da nur die Parameter der errechneten Fourier-Reihe verändert und die Funktionswerte einiger überlagerter Sinus-Funktionen neu berechnet werden müssen, ist der Rechenaufwand vergleichsweise gering.³⁵ Auch der Programmieraufwand ist entsprechend gering, da sich die Problemlösungen sehr allgemein formulieren lassen.

Da eine existente Fourier-Reihe zudem nur aus summierten trigonometrischen Funktionen besteht, ist das kontinuierliche Signal der Reihe immer stetig differenzierbar und vermeidet bei der Umwandlung zu einem zeitdiskreten Signal

³³ Vgl. **MATHWORKS®**, Pitch Shifting and Time Dilation.

³⁴ Vgl. **GOEBBELS; RITTER**, Mathematik verstehen und anwenden, S. 706 – 707.

³⁵ Vgl. ebd. S. 804.

mögliche Alias-Effekte an Sprungstellen, jedoch können Überschwingungen, wie das Gibbs-Phänomen entstehen.

3.5.1 Gibbs-Phänomen

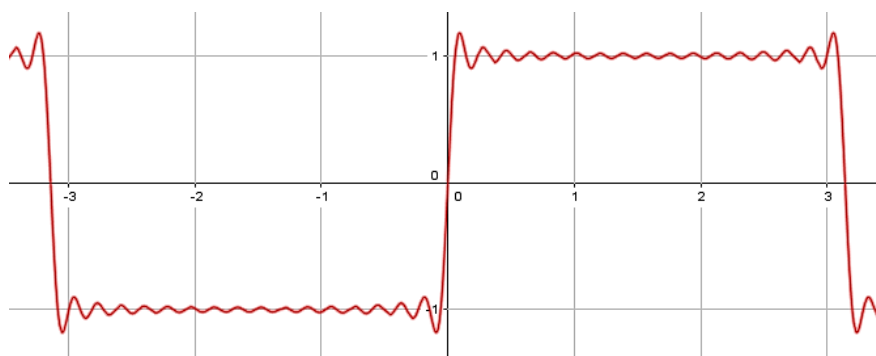


Abbildung 6: Gibbs-Phänomen

Besonders bei der Fourier-Reihe für die Rechteck-Schwingung entsteht ein Problem. Unmittelbar neben den Sprungstellen der Funktion entstehen recht starke Überschwingungen, welche die Funktion an diesen Stellen schlechter gegen die Funktionsdefinition konvergieren lassen. Diese Fehler werden als Gibbs-Phänomen bezeichnet und betragen insgesamt etwa eine Überschwingung von ca. 8,9 % zu der relativen Amplitude der Funktion.³⁶ Abbildung 6 zeigt das Gibbs-Phänomen bei der Annäherung der Rechteck-Funktion mit 15 Obertönen.

3.6 Klangformung – die ADSR-Hüllkurve

Um nach der Generation eines periodischen Signals mittels dessen Fourier-Reihe einen Ton zu formen, der keine gleichbleibende Schwingung ist, sondern bestimmte Klangcharakteristika eines Instruments, wie zum Beispiel den Anschlag oder Nachklang, nachahmt, ist eine zusätzliche Funktion notwendig, deren Funktionswerte als Vorfaktoren für die Amplitude des Signals verwendet werden sollen. Diese wird *Hüllkurve* (Englisch: *Envelope-Shape*) genannt.³⁷

Im herkömmlichen Falle der ADSR-Hüllkurve besteht diese aus vier Parametern, welche sich an den Artikulationsmöglichkeiten und Spielweisen herkömmlicher Instrumente anlehnen und von Zeit t sowie Notenlänge n abhängen: Anschlagszeit (**A**ttack), Abklingzeit (**D**ecay), Klanghaltungsfaktor (**S**ustain) und Nachklangzeit (**R**elease).³⁸

Zur Vereinfachung wird im Folgenden von einem linearen Verlauf der Teilfunktionen von $env(t)$ ausgegangen. Eine Verwendung ganzrationaler

³⁶ Vgl. GOEBBELS; RITTER, Mathematik verstehen und anwenden, S. 734-738.

³⁷ Vgl. RUSCHKOWSKI, Elektronische Klänge, S. 162-165.

³⁸ Vgl. ebd.

Funktionen höheren Grades oder Exponentialfunktionen ist darüber hinaus denkbar, um das klangliche Ergebnis weitergehend zu beeinflussen. Abbildung 7 zeigt eine mögliche Gestalt der Hüllkurve mit linearen Verläufen.

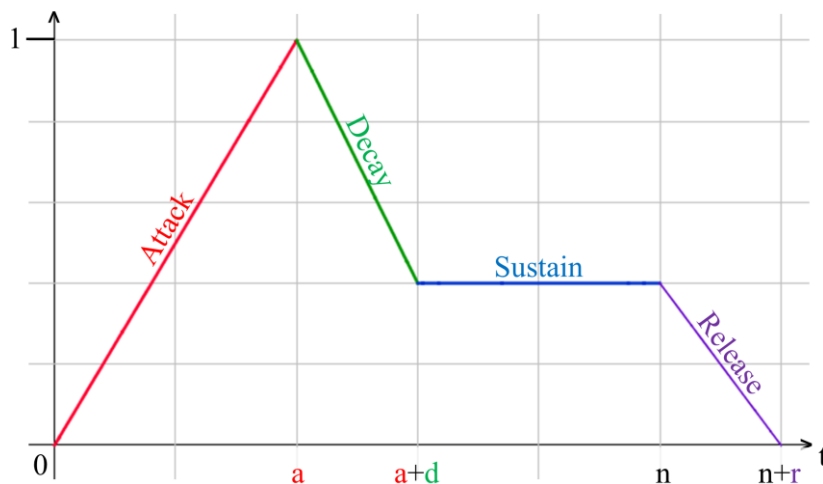


Abbildung 7: ADSR-Hüllkurve

Die Anschlagszeit ist als Teilfunktion mit folgenden Eigenschaften zu definieren: $A(0) := 0$, $A(a) := 1$, mit $a > 0$. Für $a = 0$ ist $A(t)$ nicht existent³⁹, für $a < 0$ im Definitionsbereich von $env(t)$ nicht existent. Die Abklingzeit ist abhängig von der Anschlagszeit und definiert sich so: $D(a) := 1$, $D(a + d) := s$, mit gleichen Bedingungen für d wie für a und s als Klanghaltungsfaktor mit $0 \leq s \leq 1$, dessen darauffolgende Funktion wiederum nur als $S(t) := s$ innerhalb der verbleibenden Notenlänge $a + d < t \leq n$ definiert wird. Die Nachklangzeit wird mit $R(n) := s$, $R(n + r) := 0$ definiert, hier endet die Hüllkurve eines Tons.

Aus allen Bedingungen für die Teilfunktionen folgt diese Funktionsdefinition für $env(t)$:⁴⁰

$$env(t) := \begin{cases} \text{Attack } A(t) := \frac{t}{a}, & 0 \leq t \leq a, \\ \text{Decay } D(t) := \frac{-1 + s}{d}t + \frac{1 - s}{d}(a + d) + s, & a < t \leq a + d, \\ \text{Sustain } S(t) := s, & a + d < t \leq n, \\ \text{Release } R(t) := \frac{-s}{r}t + \frac{s}{r}n + s, & n < t \leq n + r. \end{cases}$$

Im Definitionsbereich $0 \leq t \leq n + r$.

³⁹ Trotz der theoretischen Division durch 0 ist $a = 0$ in der Praxis ein gültiger Parameter.

⁴⁰ Funktionsdefinition auf Grundlage von RUSCHKOWSKI, Elektronische Klänge, S.162-163 eigenständig erarbeitet.

4 Vollständige Synthese eines Klangbeispiels

Zusammenfassend sollen alle Schritte der Klangsynthese anhand eines Beispiels durchgeführt sowie abschließend in den Ablauf für eine Programmierung überführt werden, für die hier eine allgemein verständliche Darstellung als Pseudocode gewählt wird.

Zu Beginn sind Parameter für die Tonhöhe, die Amplitude, die Hüllkurve und die Wellenform festzulegen. Der Ton soll ein recht kurz gespieltes, einige Zeit ausklingendes c^2 mit der relativen Amplitude von 80% sein. Als Wellenform wird die Rechteckschwingung gewählt, außerdem werden die Attack-Zeit auf 10 ms, Decay-Zeit auf 200 ms, Sustain-Amplitude auf 10% und die Halte- sowie Release-Zeit auf 400 ms festgelegt. Das zweigestrichene c liegt drei Halbtöne über dem Kammerton a^1 , daher berechnet sich dessen Frequenz in Hertz als $f(3) = 2^{\frac{3}{12}} * 440 \approx 523.25$. Demnach lässt sich das Klangbeispiel mathematisch wie folgt aufschreiben:

$$bsp(t) = 0.8 \text{rect}_{bsp}(t) \text{env}_{bsp}(t), \text{ mit } \text{rect}_{bsp}(t) = \frac{4}{\pi} \sum_{k=1}^{\infty} \frac{\sin(2\pi \times 523.25 \times (2k-1)t)}{(2k-1)}$$
$$\text{und } \text{env}_{bsp}(t) = \begin{cases} \frac{t}{0.01}, & 0 \leq t \leq 0.01, \\ \frac{-1+0.1}{0.2}t + \frac{1-0.1}{0.2}(0.01 + 0.2) + 0.1, & 0.01 < t \leq 0.01 + 0.2, \\ 0.1, & 0.01 + 0.2 < t \leq 0.4, \\ \frac{-0.1}{0.4}t + \frac{0.1}{0.4}0.4 + 0.1, & 0.4 < t \leq 0.4 + 0.4. \end{cases} \text{ im}$$

Intervall $[0, 0.8]$ mit t in s. Zuerst betrachten wir den Pseudocode der Hüllkurven-Funktion $\text{env}(t)$ und den zuvor festgelegten Variablen:

```
length = 0.4; freq = 523.25; amp = 0.8;
attack = 0.01; decay = 0.2; sustain = 0.1; release = 0.4;
```

Funktion **env(t)**:

```
Wenn t <= attack:
    Rückgabe von t/attack;
Wenn t > attack und t <= attack + decay:
    Rückgabe von (-1 + sustain) / decay * t + (1 -
    sustain) / decay * (attack + decay) + sustain;
Wenn t > attack + decay und t <= length:
    Rückgabe von sustain;
Wenn t > length:
    Rückgabe von (-sustain) / release * t + sustain /
    release * length + sustain;
```


Nun wird die algorithmische Umsetzung der Fourier-Reihe für die Rechteck-Schwingung betrachtet. Da eine unendliche Schleife nicht möglich ist, ist ein Abschluss der Berechnung notwendig und die Fourier-Reihe kann nur bis zu einem bestimmten Oberton angenähert werden. Aus Abschnitt 2.2 ist bereits das Shannon-Nyquist-Theorem bekannt, welches Aufschluss über diese Grenzfrequenz gibt. Die Schleife soll demnach nur solange laufen, bis die Frequenz des zu berechnenden Obertons unter der halben Abtastrate⁴¹ liegt:

```
Funktion sq(t) :  
    funktionswert = 0;  
    von order = 1 bis 22050/freq/2, Schritt 1:  
        funktionswert = funktionswert + (sin(2 * PI * freq *  
            (order * 2 - 1) * t) / (order * 2 - 1));  
    Rückgabe von 4 / PI * funktionswert;
```

Zur Speicherung der Funktionswerte in einer anhörbaren WAV-Datei wird die Pulse-Code-Modulation mit einer Samplerate von 44100 Hz und 16 Bits pro Sample verwendet, es sollen also die (gerundeten) Funktionswerte von jedem Vielfachen von $\frac{1}{44100}$ bis $\frac{n}{44100} = 0.8$ berechnet und geschrieben werden:

```
Methode synthesize() :  
    Beginne PCM-Datenstrom der WAV-Datei;  
    von sample = 0 bis 44100 * (length + release), Schritt 1:  
        schreibe amp * sq(sample/44100) * env(sample/44100);  
    Schließe PCM-Datenstrom;
```

Aus diesem Pseudocode lässt sich relativ leicht ein grundlegender Synthesizer-Programmcode entwickeln.

5 Fazit

In dieser Facharbeit konnten physikalische, musikalische und technische Grundlagen moderner elektronischer Musik, welche viele Millionen von jungen Menschen weltweit fasziniert, mithilfe der Mathematik und Algorithmik aufgezeigt und gelöst werden. Von dem Konzept der Schallwelle über deren digitale Speicherung, die Berechnung der Töne der gleichstufigen Stimmung, welche

⁴¹ Für die Programmierung wird vereinfacht die halbe Abtastrate als Grenzfrequenz gewählt und eine genauere Annäherung bevorzugt.

unsere Musik seit Jahrhunderten bilden, bis hin zu Fourier-Reihen für besondere Wellenformen und mittels Fourier-Transformation für praktisch jeden Klang der Welt, Hüllkurven zur Nachahmung bestimmter Artikulationen und Spielweisen verschiedener Instrumente und der Programmierung eines einfachen Fourier-basierten Synthesizers, wurde eine Zusammenfassung geschaffen, welche die Hintergründe digitaler Klangerzeugung gesammelt erörtert und als Vorlage zur Programmierung eines eigenen Synthesizers für jeden Interessierten dienen kann. Doch all das ist lediglich der Anfang der ständig wachsenden, immer komplexer und rechenintensiver werdenden Welt elektronischer Musik und Klangkunst.

Weit über die Verwendung einer einzelnen Fourier-Reihe als Oszillator und einer einzigen Hüllkurve als Nachbearbeitung hinaus ist denkbar, viele Oszillatoren und verschiedenste Wellenformen zu benutzen, um eine große Bandbreite an Klangvielfalt zu generieren. Außerdem lassen sich Fourier-Reihen auch besonders gut für Effekte verwenden, um deren Parameter von unterschiedlich periodisch wechselnden Werten abhängig zu machen. Damit lässt sich beispielsweise durch Anwendung einer Fourier-Reihe auf die Tonhöhe oder Lautstärke eines synthetisierten Signals eine Art Tremolo-Effekt erzeugen. Meist wird eine solche zusätzliche Implementation eines langsam schwingenden Oszillators als LFO (Low Frequency Oscillator, Niedrigfrequenzoszillator) bezeichnet. Der Fantasie der Synthesizer-Entwickler sind keine Grenzen gesetzt.

Weiterhin wichtig anzumerken ist, dass diese Art der Klangsynthese dank hoher Effizienz und einfacher Bedienbarkeit vielen Millionen von Künstlern in den letzten Jahrzehnten aufgrund wachsender Verbreitung von Computern und günstiger Musik-Produktionssoftware die Möglichkeit eröffnen konnte, an der Erschaffung elektronischer Musik mitzuwirken und an einer schier endlosen Auswahl bekannter sowie unbekannter Instrumente und Klänge ohne Mehrkosten teilzuhaben.

Ohne die Fourier-Transformation zur Kompression von Audio- und Bilddaten wären außerdem keine hochauflösenden Medien, wie Musik- und Video-Streaming-Dienste über das Internet oder Digitalfernsehen über Satellit, möglich, da unkomprimierte Datenmengen zeitdiskreter Speicherung maßlos wären.

Abschließend lässt sich anmerken, dass diese Facharbeit ihre Intention der Darstellung der Hintergründe digital gespeicherter und erzeugter Musik erreichte und gezeigt werden konnte, dass sich die emotionale Welt von Ton und Musik mathematisch rational beschreiben lässt.

6 Literaturverzeichnis

6.1 Printmedien

BADER, Franz, Dorn.Bader. Physik Gymnasium (G8) 11/12, Braunschweig: Schroedel 2010.

GOEBBELS, Steffen; **RITTER**, Stefan, Mathematik verstehen und anwenden. Von den Grundlagen bis zu Fourier-Reihen und Laplace-Transformation, Berlin, Heidelberg: Springer-Verlag 2013.

HERRMANN, Norbert, Mathematik und Gott und die Welt. Was haben Kunst, Musik oder Religion mit Mathematik am Hut?, Berlin, Heidelberg: Springer-Verlag 2014.

RUSCHKOWSKI, André, Elektronische Klänge und musikalische Entdeckungen, Stuttgart: Reclam 2010.

6.2 Internetquellen

MATHWORKS® Documentation, Pitch Shifting and Time Dilation Using a Phase Vocoder in MATLAB, <https://de.mathworks.com/help/audio/examples/pitch-shifting-and-time-dilation-using-a-phase-vocoder-in-matlab.html>, abgerufen am 14.03.2017.

MÖSER, Michael, Wahrnehmung von Schall, aus: **MÖSER**, Michael, Technische Akustik, Berlin, Heidelberg: Springer-Verlag 2003, http://www.springer.com/cda/content/document/cda_downloaddocument/9783540713869-c1.pdf, abgerufen am 14.03.2017.

OPTICAL STORAGE TECHNOLOGY ASSOCIATION (OSTA), Audio Recording, <http://www.osta.org/technology/cdqa8.htm>, abgerufen am 14.03.2017.

SAUER, Michael, Der Aufbau des Ohres, <http://www.hoerakustik-sauer.de/rundums-hoeren/das-ohr.html>, abgerufen am 14.03.2017.

7 **Abbildungsverzeichnis**

Abbildung 1, S. 6, zeitdiskrete Quantisierung einer Sinusfunktion, eigene Darstellung basierend auf **RUSCHKOWSKI**, S. 350.

Abbildung 2, S. 8, Sinus-Funktion mit Parametern, eigene Darstellung.

Abbildung 3, S. 10, Sägezahnfunktion, eigene Darstellung.

Abbildung 4, S. 11, Rechteckfunktion, eigene Darstellung.

Abbildung 5, S. 12, Dreiecksfunktion, eigene Darstellung.

Abbildung 6, S. 14, Gibbs-Phänomen, eigene Darstellung basierend auf **GOEBBELS; RITTER**, S. 738.

Abbildung 7, S. 15, ADSR-Hüllkurve, eigene Darstellung basierend auf **RUSCHKOWSKI**, S. 163.

8 Anhang

8.1 Java-Umsetzung eines Synthesizers - Klassenübersicht

Im Anhang befindet sich im Ordner „fourier-synthesizer-facharbeit“ eine Java-Programmierung, welche die Klangsynthese mit Fourier-Reihen und Hüllkurven, die Speicherung im PCM-Format und die Frequenzberechnung in der gleichstufigen Stimmung umsetzt. Das Programm ermöglicht die Erzeugung von WAV-Dateien einstellbarer Länge mit den vier behandelten Wellenformen (Sinus, Sägezahn, Rechteck, Dreieck), inklusive aller Parameter für Frequenz und Amplitude sowie derer, die eine Hüllkurve auf die generierte Schwingung anwenden, um eine gewünschte Klangwirkung zu erhalten. Mit den Reglern und Zählschaltern kann an den Parametern weitgehend beliebig experimentiert werden, um die Klangwirkung bestimmter Einstellungen spielerisch herauszufinden.

Die Programmierung ist objektorientiert und besteht aus verschiedenen Klassen mit ihren entsprechend abgegrenzten Aufgaben, welche im Folgenden näher erläutert werden und im Quellcode dokumentiert sind. Zur Ausführung des Programms ist die aktuelle Version des Java-Runtime-Environments (JRE) beziehungsweise zur Kompilierung des Quellcodes eine aktuelle Java-IDE und die neuste Version des Java-Development-Kits (JDK) notwendig.

8.1.1 Die mainWindow-Klasse

In der Hauptklasse des Programms befinden sich die Startmethode der Anwendung, der Aufbau der Benutzeroberfläche und die Aktionsmethoden der Steuerelemente. Darunter finden sich die Anzeige der aktuellen Parametereinstellungen, vor allem bei der Hüllkurve, in Beschriftungen, die Umrechnung des Halbtons in die Frequenz in Hertz beziehungsweise in die andere Richtung einer Frequenz in den entsprechenden Halbton inklusive Abweichung in Hundertstel-Halbtönen (cents) und der Aufruf der Klangsynthese mit Parameterübergabe und darauffolgender Speicherung als WAVE-Datei. Zusätzlich gibt es noch eine Schaltfläche, welche die Parameter auf zufällige Werte setzt, und eine weitere, die Programminformationen anzeigt.

8.1.2 Die Synth-Klasse

Diese Klasse beschreibt Objekte, welche die Parameter des Synthesizers als Attribute aufnehmen können und via Methodenaufruf von `synthesize_standard()` die PCM-Daten der Klangsynthese erzeugen sowie die anschließende Speicherung mithilfe eines Pfadauswahldialogfensters durchführen.

Beim Aufruf von `synthesize_standard()` wird zuerst die Pfadauswahl getätigt, die bei erfolgreicher Speicherortfestlegung den Dateipfad zurückgibt und somit die Ausführung des Synthesizer-Algorithmus ermöglicht. Andernfalls wird ein leerer Wert zurückgegeben, keine Synthese durchgeführt und nach entsprechendem Hinweisdialog die Methodenausführung beendet.

Wird jedoch die Klangsynthese gestartet, werden Objekte der Klassen `Oscillator`, mit der Wellenform und Grenzfrequenz (`cutoff`) als Attribute, und `Envelope`, mit den ADSR-Hüllkurven-Parametern und der Haltelänge des Tons, initialisiert. Im nächsten Schritt wird eine Reihe (ein sogenanntes Array) von `Samplerate*Länge` vielen Fließkommawerten als Buffer (Zwischenspeicher) deklariert, in welche im Anschluss nach dem im Pseudocode beschriebenen Konzept die in der Synthese berechneten Werte nacheinander geschrieben werden. Der Buffer wird abschließend in eine Methode `save_as_wave(Fließkommaarray, Datei)` weitergegeben, welche die Fließkommawerte in einen 16 Bit-Datenstrom umwandelt und mithilfe der Java-Sound API die Speicherung der WAVE-Datei übernimmt.

8.1.3 Die Oscillator-Klasse

In dieser Klasse sind die Summenschreibweisen der Fourier-Reihen für die Wellenformen Sägezahn, Rechteck und Dreieck entsprechend dem Pseudocode-Konzept algorithmisch umgesetzt und um eine zusätzliche Funktion für eine einfache Sinus-Schwingung ergänzt.

Ein `Oscillator`-Objekt besitzt Attribute für die Wellenform und die Grenzfrequenz (`cutoff`) der Fourier-Reihen-Berechnung.

Über die Funktion `getSample(Kommazahl Frequenz, Kommazahl Zeit)` gibt ein Objekt der Klasse den Wert der festgelegten Wellenform einer Periodenlänge von $1/\text{Frequenz}$ zum Zeitpunkt `Zeit` zurück.

8.1.4 Die Envelope-Klasse

Auch der Aufbau dieser Klasse orientiert sich am Konzept aus Abschnitt 5. Ein Hüllkurven-Objekt der Klasse `Envelope` besitzt die konstanten Attribute für `Attack`, `Decay`, `Sustain` und `Release` sowie ein veränderbares Attribut für die Länge eines Tons (`Hold`, oder hier: `length`). Über `getAmplitude(Kommazahl Zeit)` wird die relative Amplitude zum Zeitpunkt `Zeit` berechnet und zurückgegeben. `setLength(Kommazahl Länge)` legt die Haltezeit der Hüllkurve für einen Ton neu

`fest.getFullLength()` gibt die vollständige Länge eines Tons (Haltezeit+Release) zurück.

8.1.5 Die Pitch-Klasse

Die statische Klasse Pitch stellt keine Vorlage für Objekte dar, sondern stellt lediglich Funktionen zur Umrechnung von Halbtonschritten vom Kammerton a aus zu Frequenzwerten in Hertz (`freq(Kommazahl key)`) und umgekehrt (`key(Kommazahl freq)`) zur Verfügung.

8.1.6 Letzte Änderungen und bekannte Probleme

In der derzeitigen Version 0.1.5 sind bisher keine bekannten Probleme vorhanden. Ich bitte um Rückmeldung zur zeitnahen Korrektur gegebenenfalls auftretender Fehler.

Die neueste Version beinhaltet Code-Optimierungen und Verhaltenskorrekturen für die Weitergabe von Parametern sowie verbesserte Thread-Sicherheit. Änderungen in 0.1.4: eine Funktion für zufällige Parameter wurde hinzugefügt. Änderungen in 0.1.3 und 0.1.2: Code-Optimierungen, Dialogfeld für Programminformationen, Korrektur des Verhaltens der RadioButtons.

8.2 GeoGebra-Datei zur Darstellung der Hüllkurve mit Parametern

Ein weiterer Bestandteil des Anhangs ist die GeoGebra-Datei „adsr-envelope.ggb“, welche zur interaktiven graphischen Veranschaulichung der Funktionsweise der ADSR-Hüllkurve, deren Parameter und Funktionsterme dient. Die Parameter für Attack, Decay, Sustain und Release lassen sich mittels Schieberegler verändern, die die Auswirkungen auf die Gestalt der Hüllkurve unmittelbar zeigen.

Hiermit versichere ich, dass ich die vorliegende Facharbeit selbstständig angefertigt, keine anderen als die angegebenen Hilfsmittel benutzt und die Stellen der Facharbeit, die im Wortlaut oder im wesentlichen Inhalt aus anderen Werken entnommen wurden, mit genauer Quellenangabe kenntlich gemacht habe.

Hiermit erkläre ich, dass ich damit einverstanden bin, wenn die von mir verfasste Facharbeit der schulinternen Öffentlichkeit zugänglich gemacht wird.