



**Hochschule Konstanz**  
Technik, Wirtschaft und Gestaltung

**Signale, Systeme und Sensoren**

# **Fourieranalyse und Akustik**

**Gina Kokoska, Simon Winter**

**Konstanz, 31. Mai 2021**

## **Zusammenfassung (Abstract)**

Thema:	Fourieranalyse und Akustik	
Autoren:	Gina Kokoska	gi161kok@htwg-konstanz.de
	Simon Winter	si411win@htwg-konstanz.de
Betreuer:	Prof. Dr. Matthias O. Franz	mfranz@htwg-konstanz.de
	Jürgen Keppler	juergen.keppler@htwg-konstanz.de
	Daniel Castelo	dacastel@htwg-konstanz.de

In diesem Versuch wird ein akustisches Signal mit Hilfe der Fourieranalyse untersucht. Ziel dieses Versuches ist es, ein analoges Signal mit Hilfe der Fouriertransformation in Frequenzanteile zu zerlegen um daraus die Tonhöhe in Hertz zu rekonstruieren.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>III</b>
<b>Tabellenverzeichnis</b>	<b>IV</b>
<b>Listingverzeichnis</b>	<b>V</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Versuch 1</b>	<b>2</b>
2.1 Fragestellung, Messprinzip, Aufbau, Messmittel . . . . .	2
2.2 Messwerte . . . . .	3
2.3 Auswertung . . . . .	4
2.4 Interpretation . . . . .	4
<b>Anhang</b>	<b>5</b>
A.1 Quellcode . . . . .	5
A.1.1 Quellcode Versuch 1 . . . . .	5
<b>Literaturverzeichnis</b>	<b>7</b>

# **Abbildungsverzeichnis**

# **Tabellenverzeichnis**

# Listingverzeichnis

LaTeX_Template/src/codea3.txt . . . . .	5
---	---

# Kapitel 1

## Einleitung

Dieser Versuch nutzt ein Kondensatormikrofon, welches einen analog erzeugten konstanten Ton digitalisiert. Das erzeugte akustische Signal wird anschliessend mit Hilfe der Fourieranalyse in ihre Fourier-transformierten zerlegt, um diese anschliessend auf . Alle Zwischenergebnisse werden graphisch dargestellt.

[1] [2]

# Kapitel 2

## Versuch 1

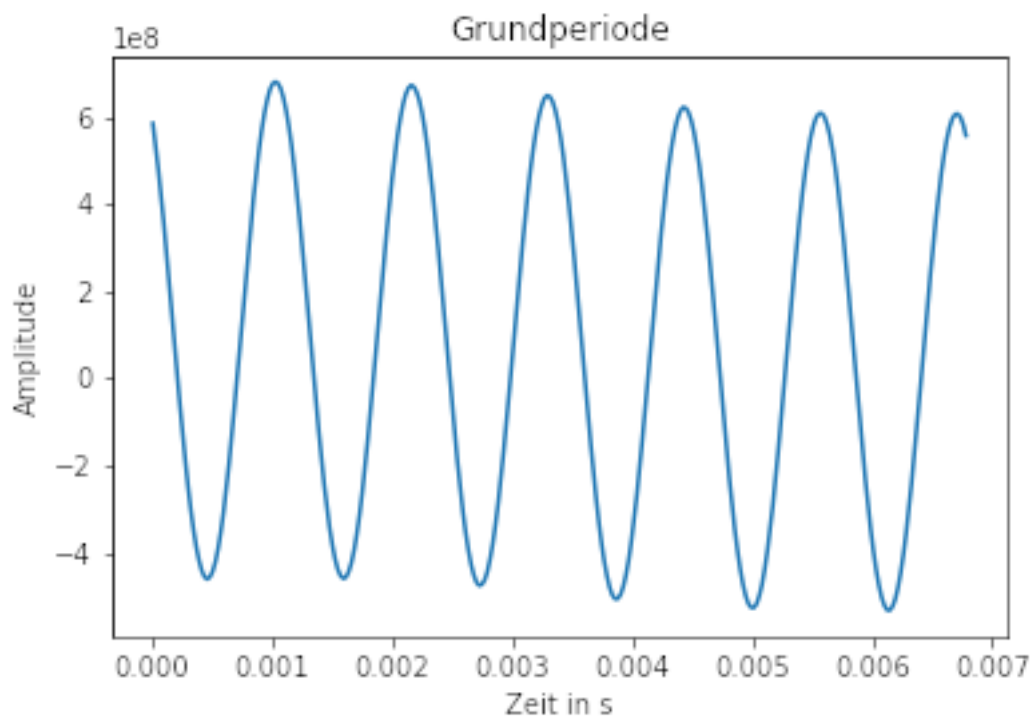
### 2.1 Fragestellung, Messprinzip, Aufbau, Messmittel

Es wird die Tonhöhe eines akustische Signals gesucht. Hierfür wird ein konstanter Ton mit einem Mikrofon aufgenommen und in ein Python-Programm implementiert. Die Grundperiode (in ms) und die Grundfrequenz (in Hz) des aufgenommenen Signals wird in einem Graphen geplottet. Innerhalb des Programms wird die Signaldauer (in s), die Abtastfrequenz (in Hz), die Signallänge  $M$  und das Abtastintervall  $\Delta t$  (in s) berechnet. Letzlich wird die Fouriertransformierte des Signals berechnet um daraus die Frequenz, also die Tonhöhe zu erhalten. Das verwendete Mikrofon ist ein Kondensatormikrofon welches einen elektrostatischen Wandler nutzt, um Luftschall in eine elektrische Spannung um zu wandeln. Es nutzt das Prinzip der Kapazität, in welchem der Abstand einer leitfähigen Membran zur Gegenelektrode gemessen wird. Hierbei wird eine polarisierte Kapsel bestehend aus einer elastischen Membran und einer Gegenelektrode verwendet. Diese bilden die Platten eines Kondensators der Gleichspannung als elektrische Kapazität speichern kann. Auftreffender Schall bringt die Membran zum Schwingen, wodurch sich der Abstand zwischen Membran und Gegenelektrode und damit auch die Kapazität des Kondensators verändert. Je geringer der Abstand desto grösser der eintreffende Schall (antiproportionales Signal). Das Mikrofon ist mit einem USB-Kabel an einen Computer angeschlossen. Zur Erzeugung des Tons wird die App Music Studio Lite verwendet. Das gewählte Instrument ist ein Synth Bass und spielt den Ton C4 ab. Als Messmittel wird ein AUNA MIC-900B Mikrofon verwendet.



## 2.2 Messwerte

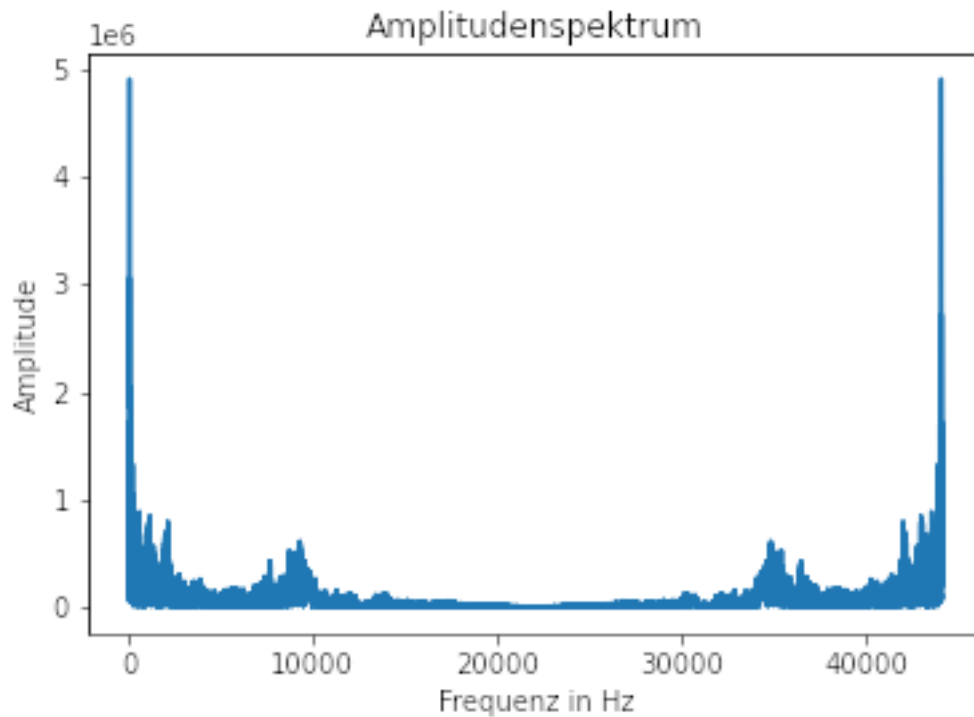
Das Programm lieferte folgenden Periodenausschnitt:



Aus der Grafik ist eine ungefähre Periodendauern von 0.00120s zu entnehmen.

## 2.3 Auswertung

Das Berechnen der Fouriertransformierten lieferte folgende Grafik:



## 2.4 Interpretation

Aus dem Ergebnis lässt sich schliessen, dass die errechnete Tonhöhe 833Hz mit einem Messfehler von 393Hz von der eigentlichen Tonhöhe 440Hz abweicht. Zusammenfassend kann gesagt werden, dass die Fourieranalyse eine passende Methode ist, um die Tonhöhe eines Signals zu erhalten. Natürlich muss beachtet werden, dass in unserem Falle keine Filter angewandt wurden, um das tatsächliche Signal zu erhalten, somit sind die Messfehler in dem Ergebnis nicht unerwartet.

# Anhang

## A.1 Quellcode

### A.1.1 Quellcode Versuch 1

```
1 # Aufgabe 1
2 import matplotlib.pyplot as plt
3 import pyaudio
4 import numpy as np
5 from scipy.io.wavfile import read, write
6
7 FORMAT = pyaudio.paInt16
8 SAMPLEFREQ = 44100 # 44 kbps
9 FRAMESIZE = 1024 # each buffer contains 1024 samples
10 NOFFFRAMES = 220
11 WAVE_OUTPUT_FILENAME = "output.wav"
12 p = pyaudio.PyAudio()
13 print('running')
14
15 stream = p.open(format=FORMAT, channels=1, rate=SAMPLEFREQ, input=True,
16                 frames_per_buffer=FRAMESIZE)
17 data = stream.read(NOFFFRAMES*FRAMESIZE)
18 decoded = np.fromstring(data, np.int)
19
20 stream.stop_stream()
21 stream.close()
22 p.terminate()
23 deltaT = 1 / SAMPLEFREQ
24 xvalue = np.arange(0, 300) * deltaT
25
26 plt.plot(xvalue, decoded[0:300])
27 plt.title("Grundperiode")
```

```

28 plt.xlabel("Zeit in s")
29 plt.ylabel("Amplitude")
30 plt.savefig('frequenzspektrum.png')
31
32 plt.show()
33 dataRead = read('output.wav')
34 dataRead = np.array(dataRead[1], dtype=int)
35 np.array(dataRead[1], dtype=float)
36 np.savetxt("outputData.csv", dataRead, delimiter=',')
37
38
39 fft = np.fft.fft(dataRead)
40 spec = np.abs(fft)
41 freq = np.linspace(0, FRAMESIZE, len(spec))
42 xvalue_sp = np.arange(0, 220160) / (220160 * deltaT)
43 plt.plot(xvalue_sp, spec)
44 plt.title('Amplitudenspektrum')
45 plt.xlabel('Frequenz in Hz')
46 plt.ylabel('Amplitude')
47 plt.savefig('amplitudenspektrum.png')
48 plt.show()
49
50 T = (0.0022 - 0.001)
51 Tonhoehe = 1 / T

```

# Literaturverzeichnis

- [1] Franz, Prof. Dr. Matthias O.: *Vorlesung 10 - Sprache und Spracherkennung: Kurzzeit-Fouriertransformation, Erzeugung und Wahrnehmung von Sprache, Mustererkennung durch Korrelation*. In: *Vorlesung Technische Grundlagen der angewandten Informatik*, 2016.
- [2] Franz, Prof. Dr. Matthias O.: *Vorlesung 1 - Einführung*. In: *Vorlesung Technische Grundlagen der angewandten Informatik*, 2016.