

# Ortungssysteme auf einer flachen Ebene

Eine Präsentation von Janne K. und David B.

# Gliederung

## 1. Problemstellung

GPS

Schallbasis

## 2. Ansatz

Methodik

mathematische Grundlage

## 3. Umsetzung

Experiment

Simulation

## 4. Auswertung

Fazit

## 5. Ausblick

Anwendungsbereiche

# Problemstellung

# Global Positioning System

# Grundlage

- ursprünglich im Militärbereich vorgesehen für  
**Navigation und Positionsbestimmung**  
(USA, 1978)

# Grundlage

- ursprünglich im Militärbereich vorgesehen für  
**Navigation und Positionsbestimmung**  
(USA, 1978)
- Satelliten-Konstellationen, die durch Signale  
miteinander kommunizieren

# Grundlage

- ursprünglich im Militärbereich vorgesehen für  
**Navigation und Positionsbestimmung**  
(USA, 1978)
- Satelliten-Konstellationen, die durch Signale  
miteinander kommunizieren

=> berechnet Standort durch Trilaterationsverfahren

# Eckdaten

# Eckdaten

- **Abdeckung:** 95% der Zeit vollständige Abdeckung

# Eckdaten

- **Abdeckung:** 95% der Zeit vollständige Abdeckung
- **Medium:** elektromagnetischen Radiowellen

# Eckdaten

- **Abdeckung:** 95% der Zeit vollständige Abdeckung
- **Medium:** elektromagnetischen Radiowellen
- **Ausbreitung:** Lichtgeschwindigkeit

# Eckdaten

- **Abdeckung:** 95% der Zeit vollständige Abdeckung
- **Medium:** elektromagnetischen Radiowellen
- **Ausbreitung:** Lichtgeschwindigkeit
- **Genaugigkeit:** 5 - 10 Meter

# Eckdaten

- **Abdeckung:** 95% der Zeit vollständige Abdeckung
- **Medium:** elektromagnetischen Radiowellen
- **Ausbreitung:** Lichtgeschwindigkeit
- **Genaugkeit:** 5 - 10 Meter
- **Sendeintervall:** 3 Sekunden

# Verwendungszwecke

# Verwendungszwecke

- Navigation

# Verwendungszwecke

- Navigation
- Kartografie

# Verwendungszwecke

- Navigation
- Kartografie
- Zeitmessung

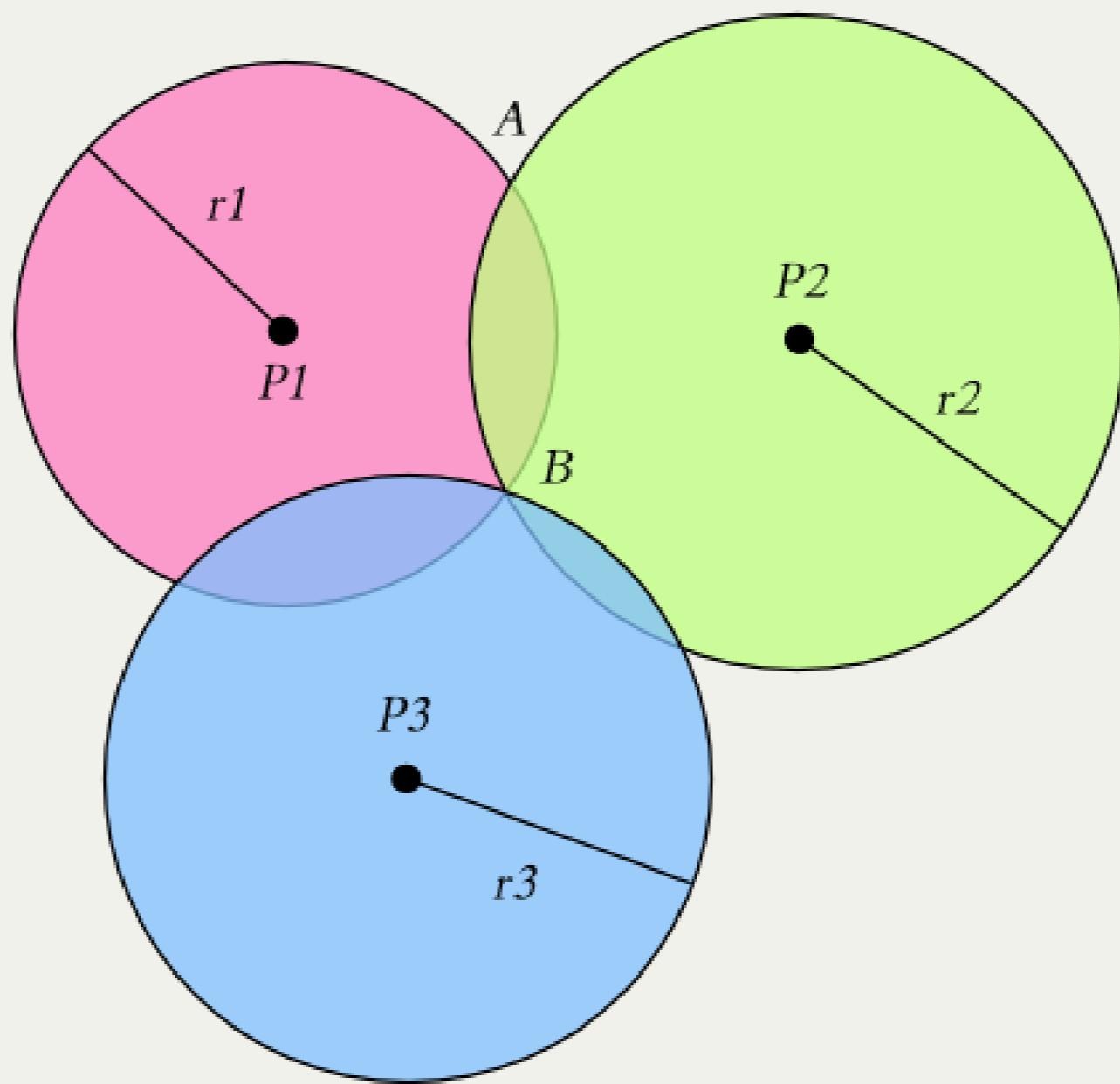
# Verwendungszwecke

- Navigation
- Kartografie
- Zeitmessung
- Positionsbestimmung

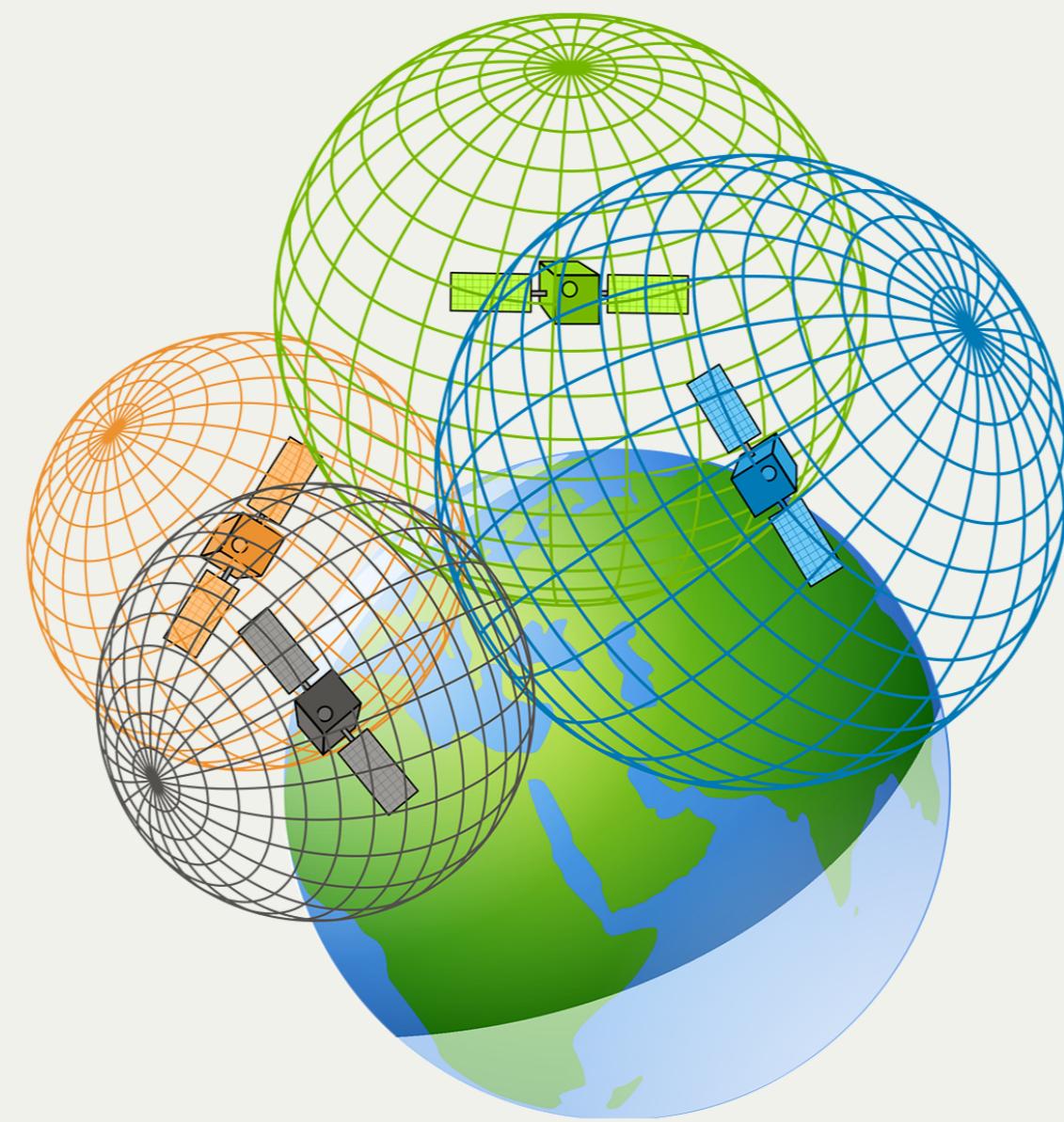
# Trilateration

# Trilateration

Verfahren, welches die Zeitunterschiede  
der empfangenen Signale benutzt,  
um einen Standort zu ermitteln



<https://de.wikipedia.org/wiki/Datei:Trilateration.png> [23.05.2023 - 13:23 Uhr]



<https://de.wikipedia.org/wiki/Datei:Trilateration.png> [23.05.2023 - 13:25 Uhr]

# Probleme

# Probleme

- extreme Distanzabweichungen bei Fehlmessungen,  
durch hohe Ausbreitungsgeschwindigkeit

# Probleme

- extreme Distanzabweichungen bei Fehlmessungen,  
durch hohe Ausbreitungsgeschwindigkeit
- $0,01s \cdot 299.792.458 \frac{m}{s} = 2.997.924,58m$

# Optimierungsansatz

# Optimierungsansatz

Durch anderes Medium geringere Ausbreitungsgeschwindigkeit

# Optimierungsansatz

Durch anderes Medium geringere Ausbreitungsgeschwindigkeit  
⇒ geringere Abweichungen bei gleicher Abtastrate

# Schallbasiertes Tracking

# Unterschied zum GPS Tracking

schall

---

elektromagnetisch

# Unterschied zum GPS Tracking

schall

elektromagnetisch

---

Ausbreitungsgeschwindigkeit      343 m/s      299.792.458 m/s

---

# Unterschied zum GPS Tracking

**schall**

**elektromagnetisch**

---

Ausbreitungsgeschwindigkeit	343 m/s	299.792.458 m/s
-----------------------------	---------	-----------------

---

Genauigkeit (191khz)	0,001796m	1.569,594m
----------------------	-----------	------------

# Nachteile

# Nachteile

- begrenzte Reichweite

# Nachteile

- begrenzte Reichweite
- starke Einschränkung durch Hindernisse

# Abgrenzung des Themas

# Abgrenzung des Themas

- Verwendung von GPS nur als Vergleich für unser Projekt

# Abgrenzung des Themas

- Verwendung von GPS nur als Vergleich für unser Projekt
- keine Vertiefung in das Thema GPS

Leitfrage:

*„Inwiefern wäre ein schallbasiertes Ortungssystem umsetzbar und nützlich?“*

# Mögliche Anwendungen

*„Autonomes Fahren“*

# „Autonomes Fahren“



<https://www.adac.de/rund-ums-fahrzeug/ausstattung-technik-zubehoer/autonomes-fahren/technik-vernetzung/aktuelle-technik/> [23.05.2023 - 15:59 Uhr]

„Staubsaugroboter“

# „Staubsaugroboter“



<https://www.digitaltrends.com/> [29.05.2023 - 12:52 Uhr]

# Ansatz

Vorgehen von Methodik & Experiment

# Methodik

## Grundlagen

Welle

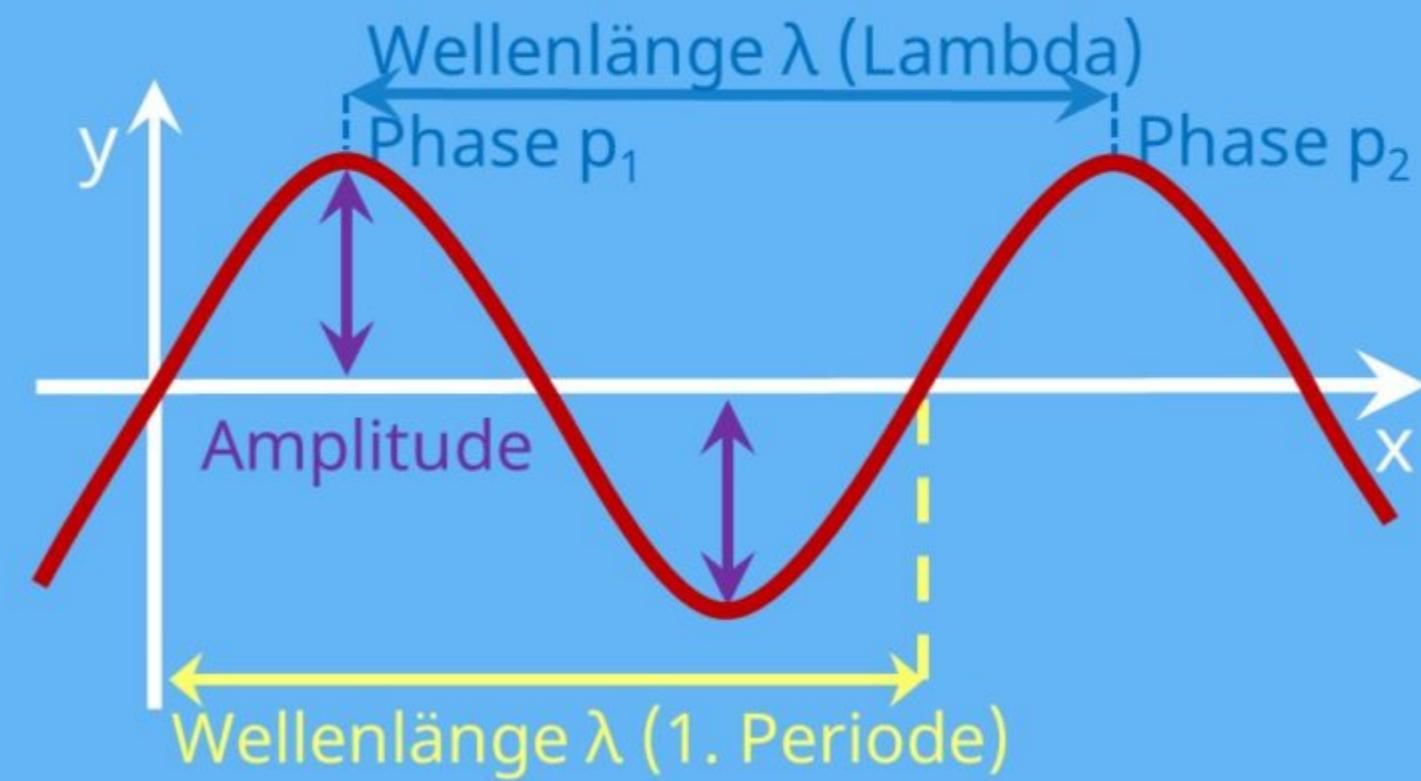
# Welle

- **longitudinal:** schwingt in Richtung ihrer Ausbreitung  
(z.B. Schallwelle)

# Welle

- **longitudinal:** schwingt in Richtung ihrer Ausbreitung  
(z.B. Schallwelle)
- **transversal:** schwingen senkrecht zu ihrer  
Ausbreitung (z.B. Wasserwelle)

## Wellenlänge und Amplitude



<https://studyflix.de/ingenieurwissenschaften/wellenlange-4447> [29.05.2023 - 17:33]

# Schallwellen

# Schallwellen

- Ausbreitung von Schwingungen in einem Medium, z.B. Luftmolekülen, die Schallwellen erzeugen.

# Schallwellen

- Ausbreitung von Schwingungen in einem Medium, z.B. Luftmolekülen, die Schallwellen erzeugen.
- **longitudinale Wellen**

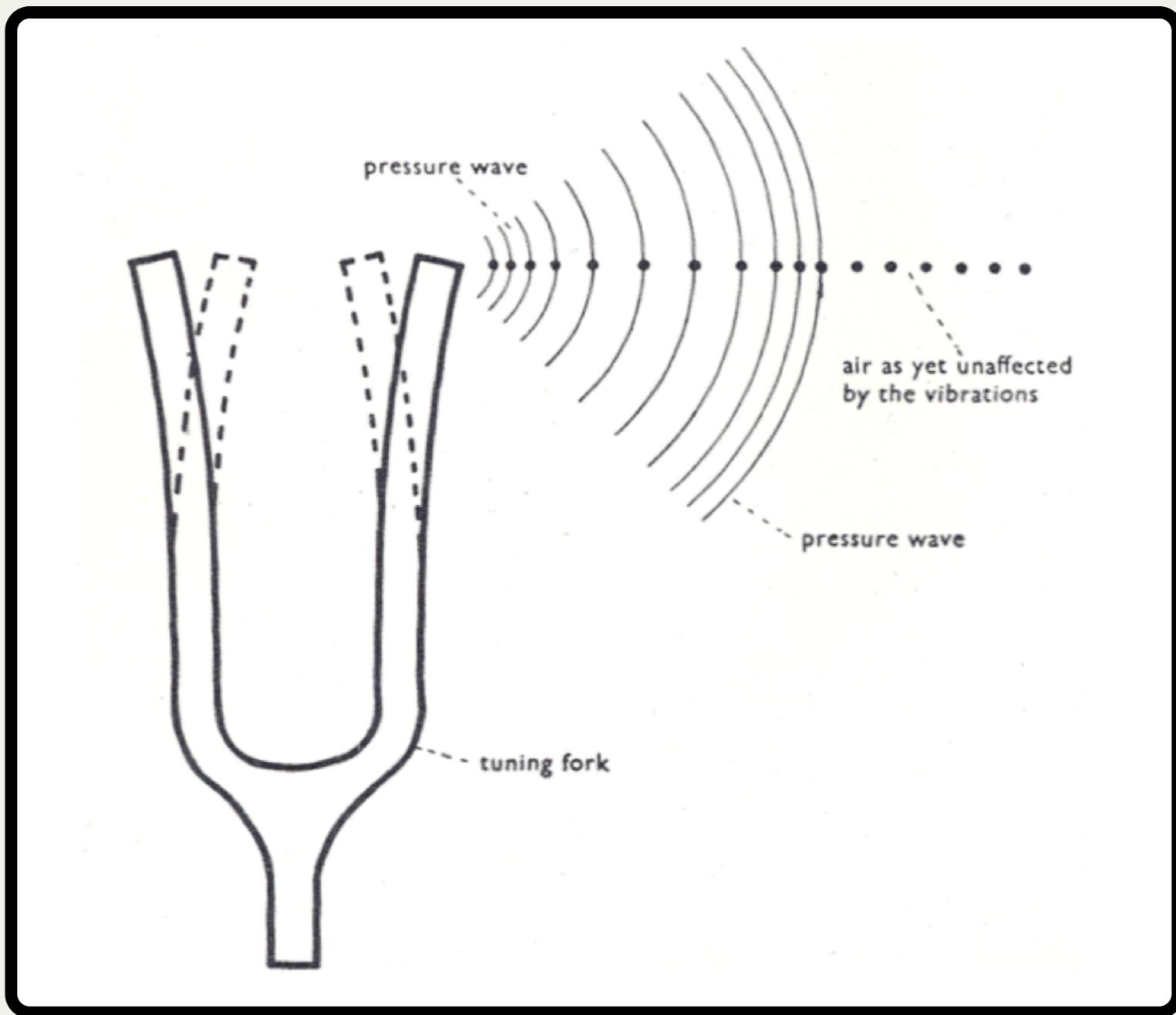


Abb. 1: Stimmgabel

(Ladefoged, 1996:4) [29.05.2023 - 00:29]

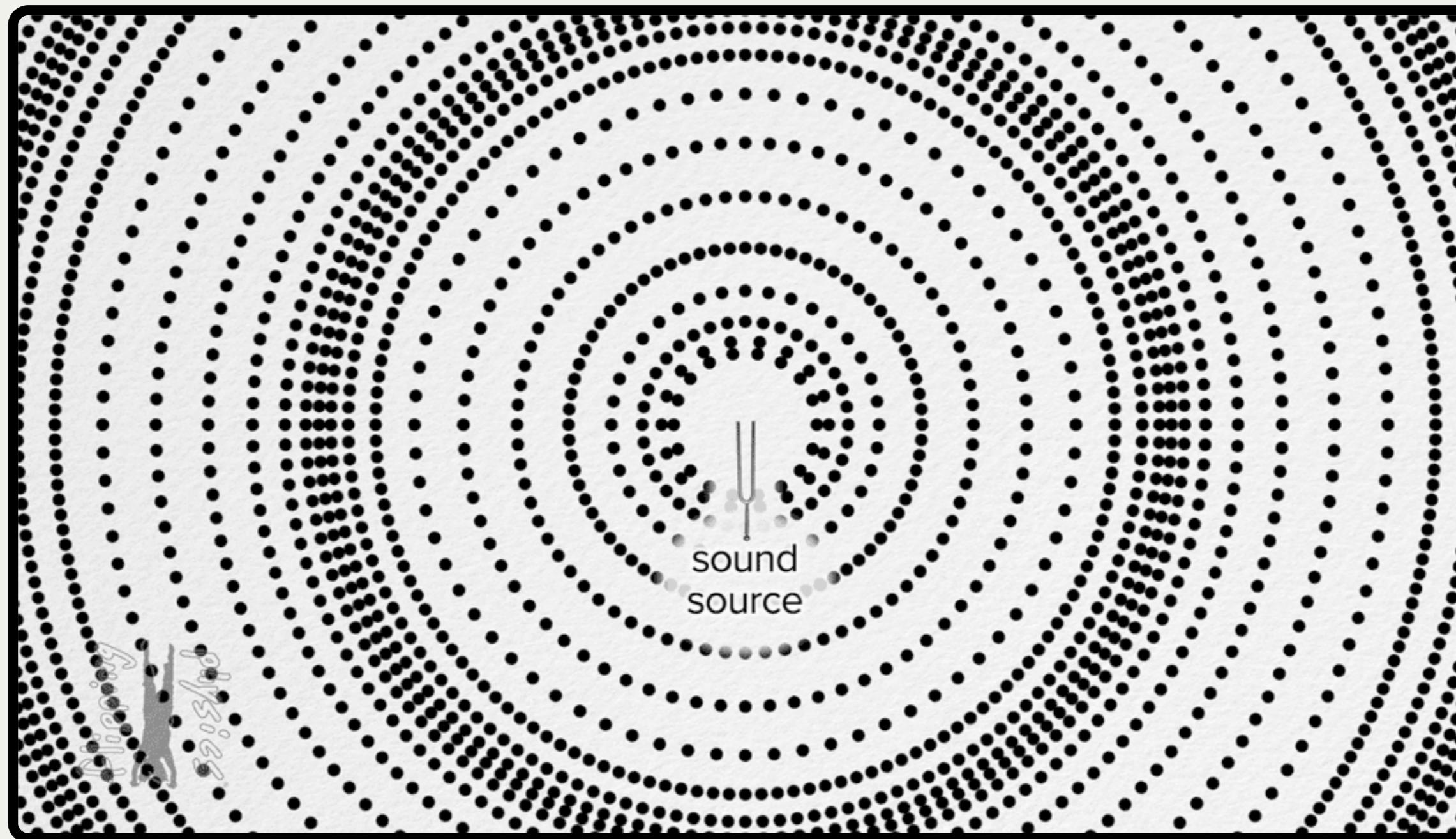


Abb. 2: Visuelle Darstellung von schwingenden Luftmolekülen

<https://www.flippingphysics.com/> [29.05.2023 - 12:55]

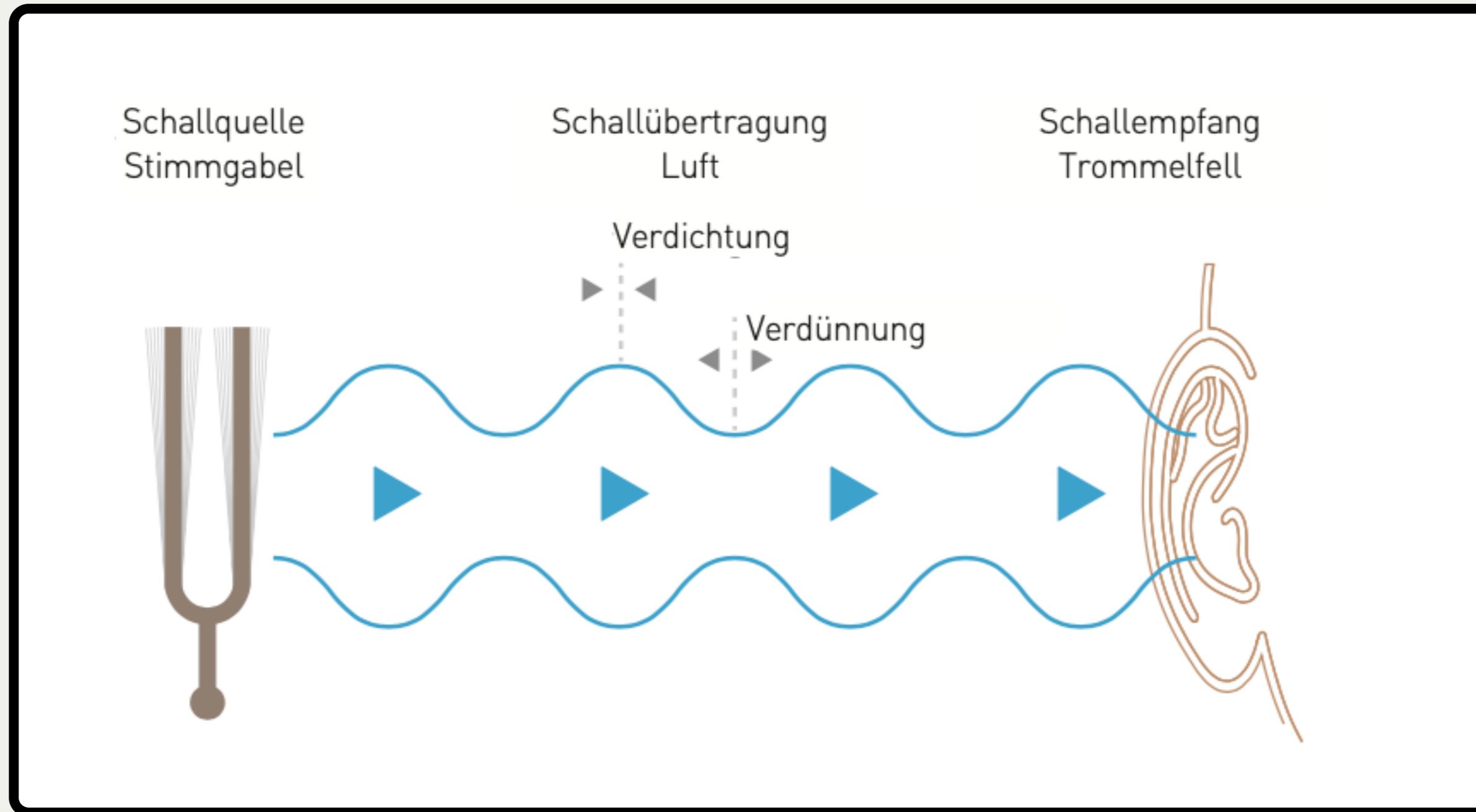


Abb. 3: Schallausbreitung

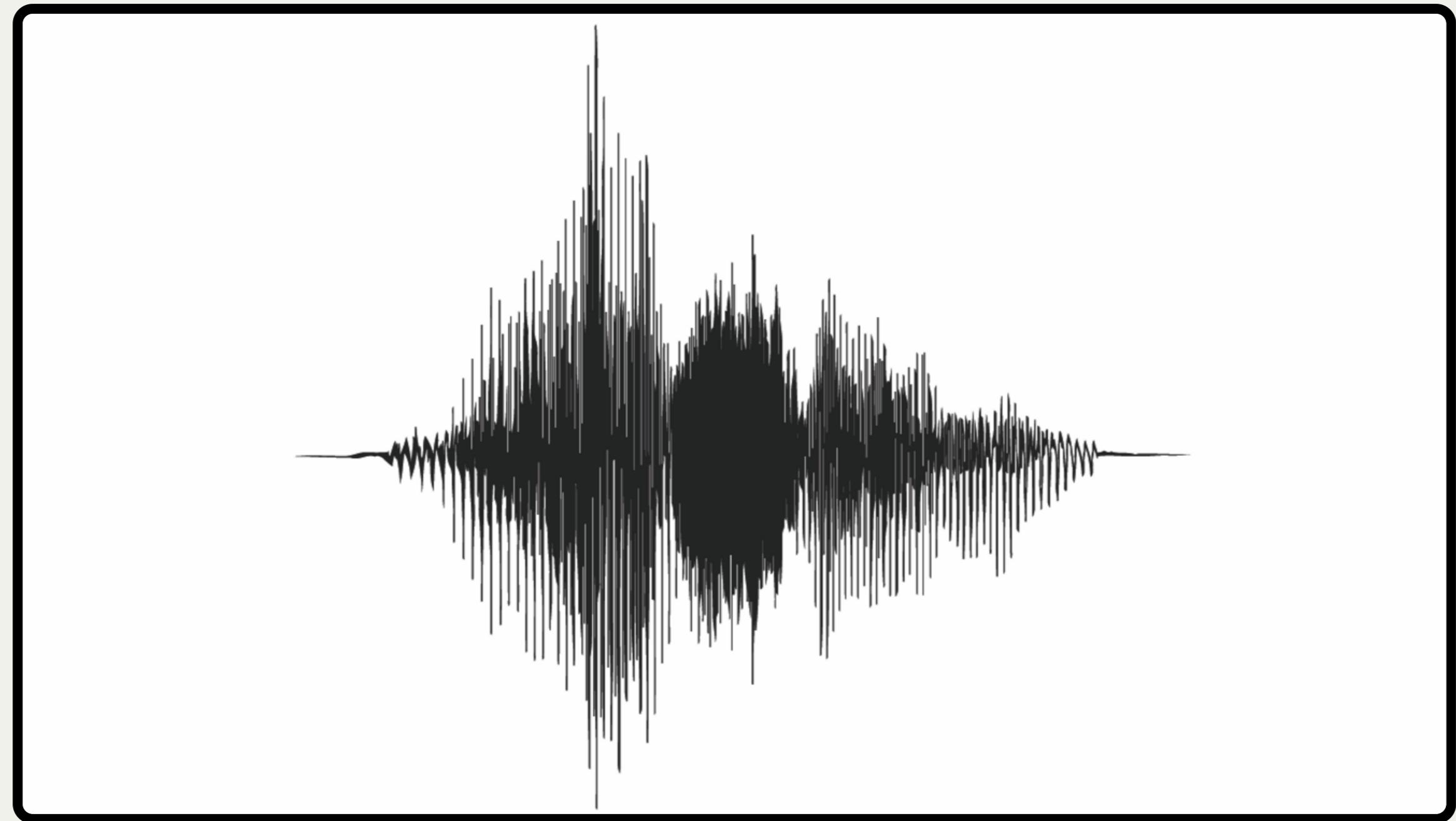
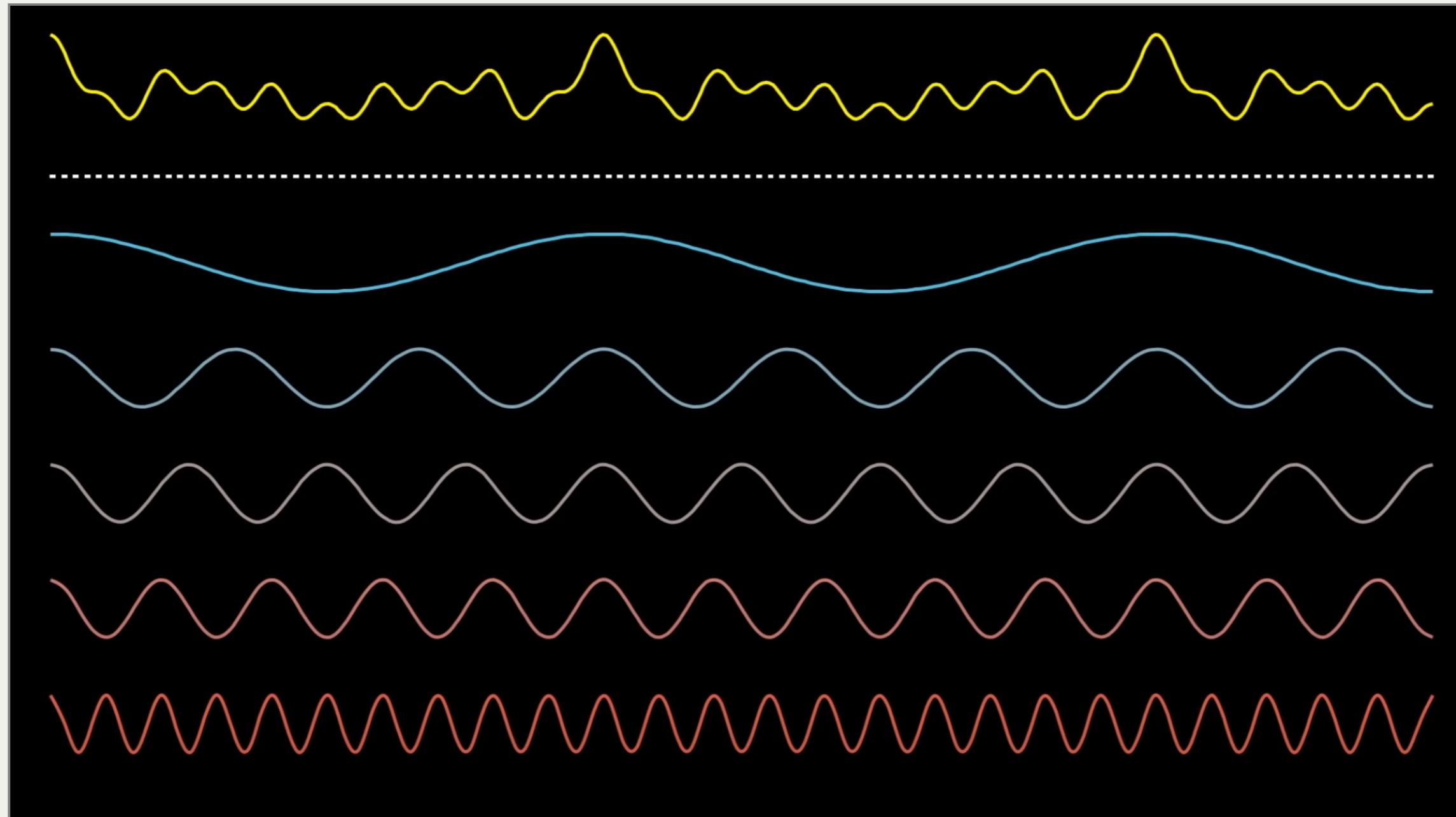


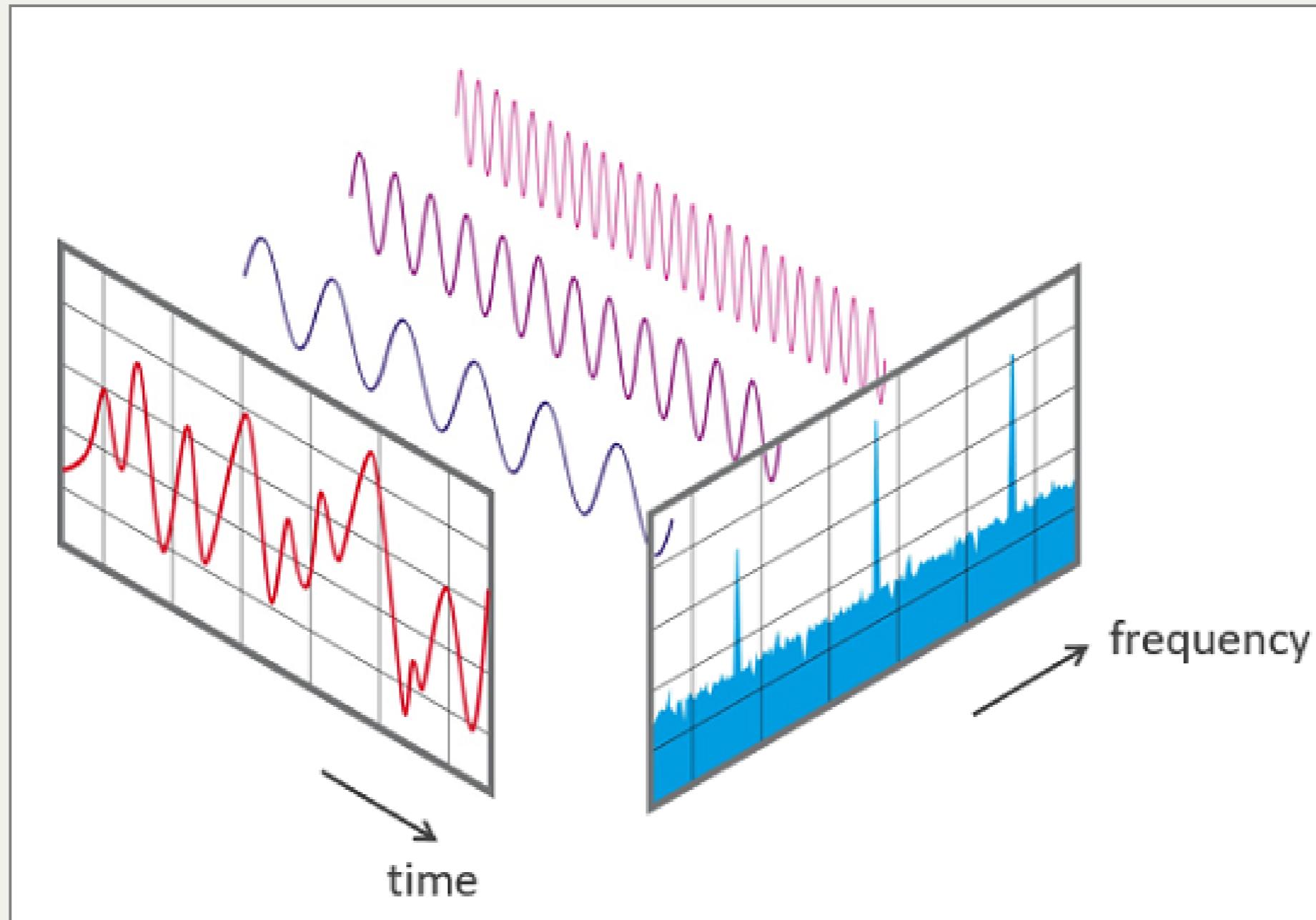
Abb. 4: Sound

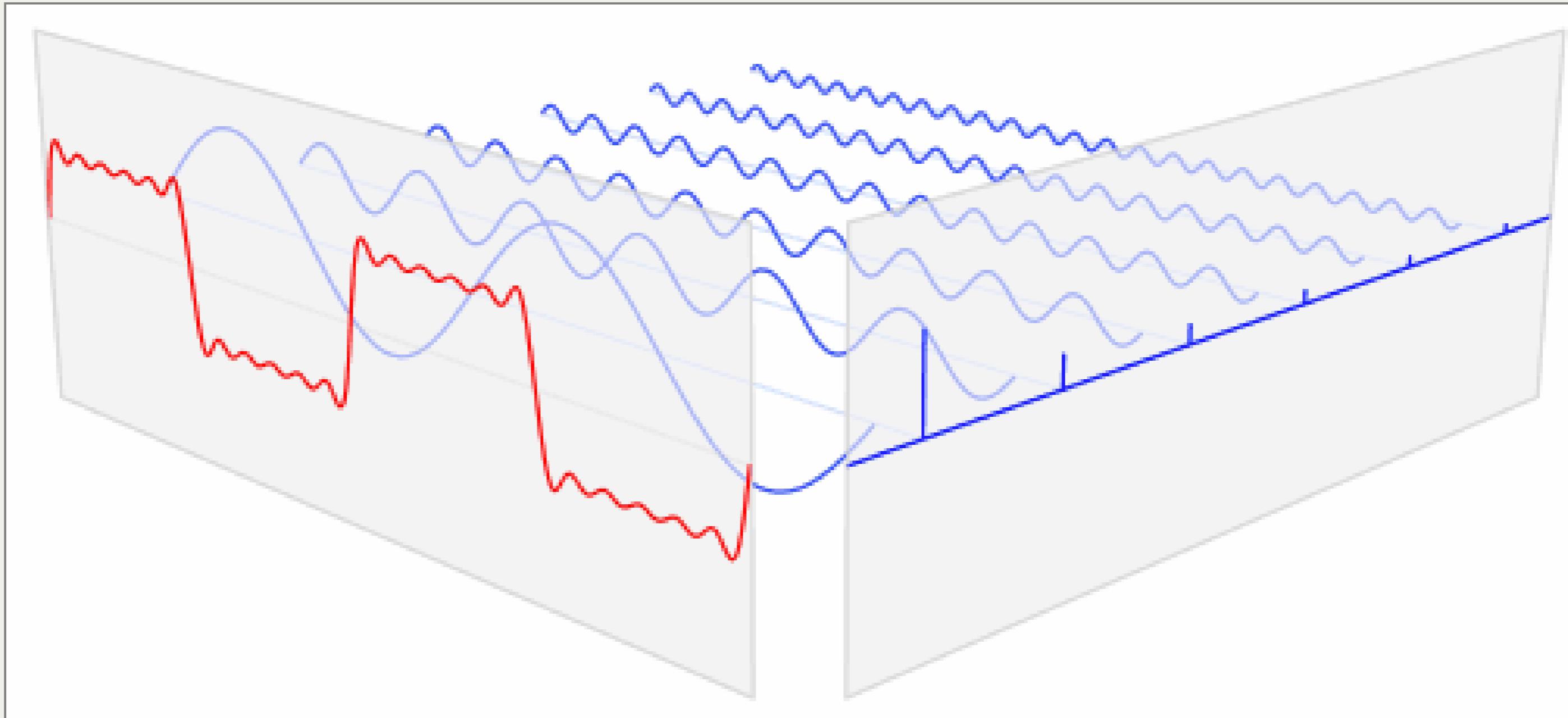
<https://www.weltderphysik.de/> [29.05.2023 - 01:52]

# Fourier Transformation



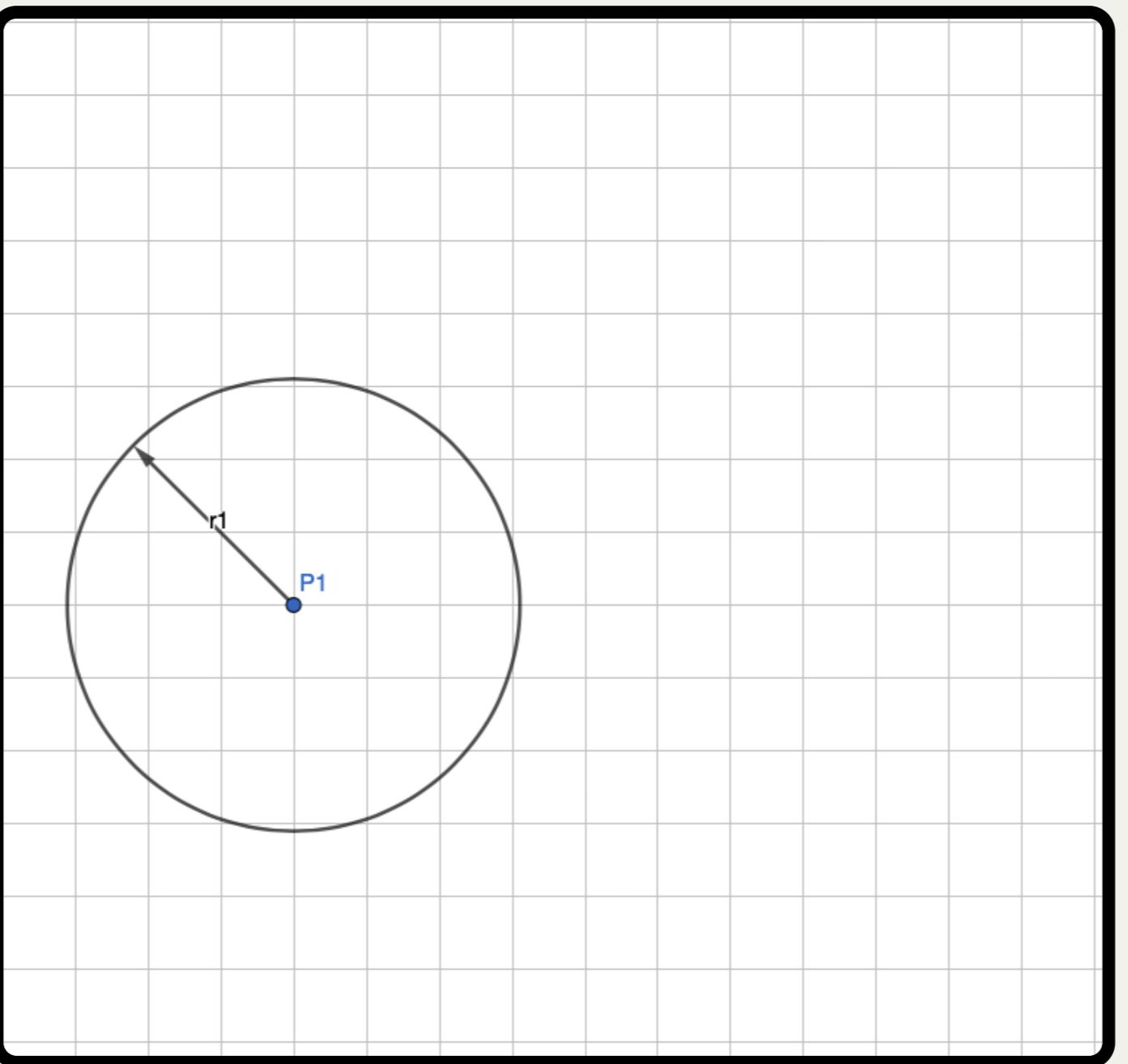
<https://www.youtube.com/watch?v=spUNpyF58BY> [23.05.2023 - 16:53 Uhr]

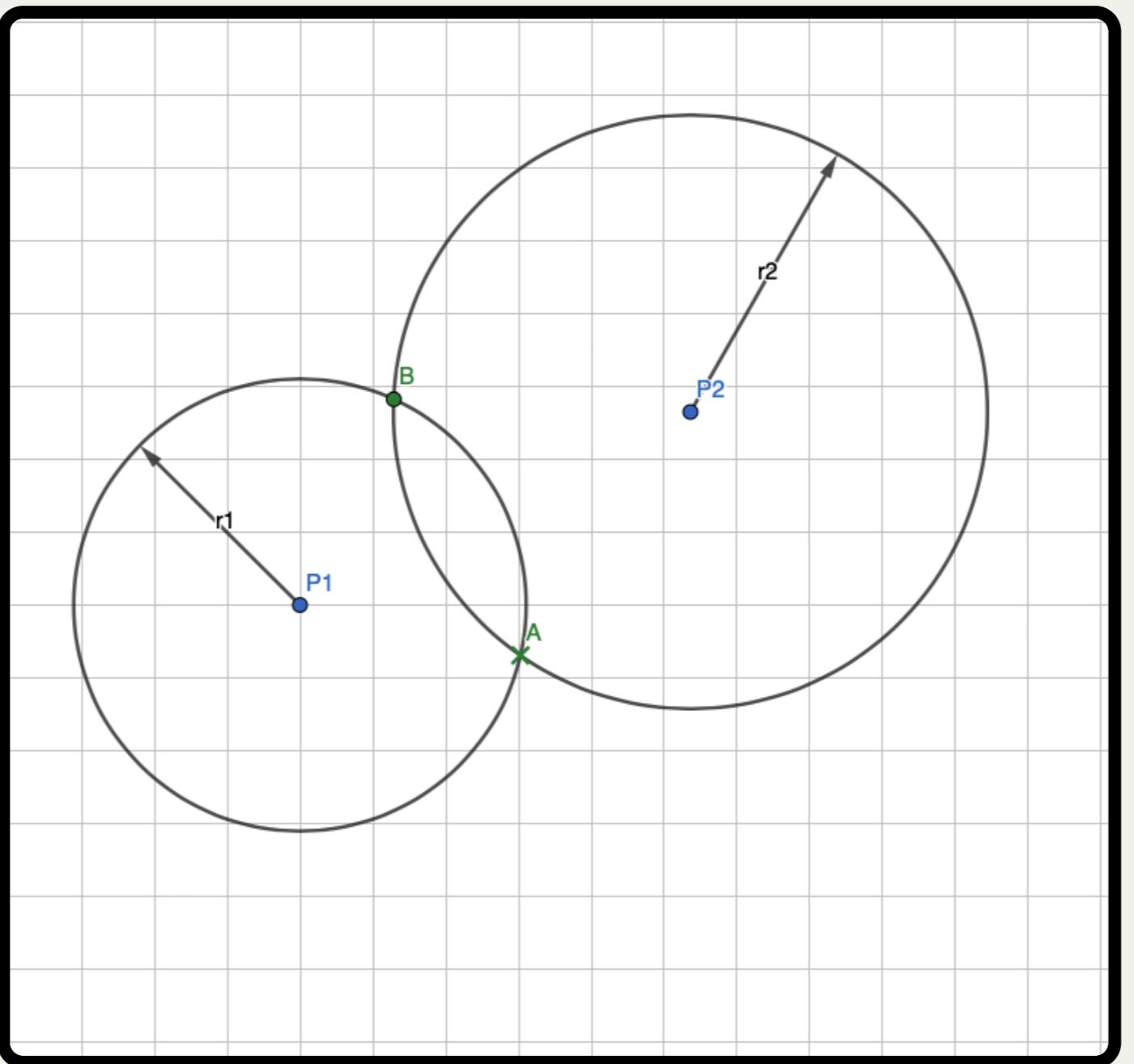


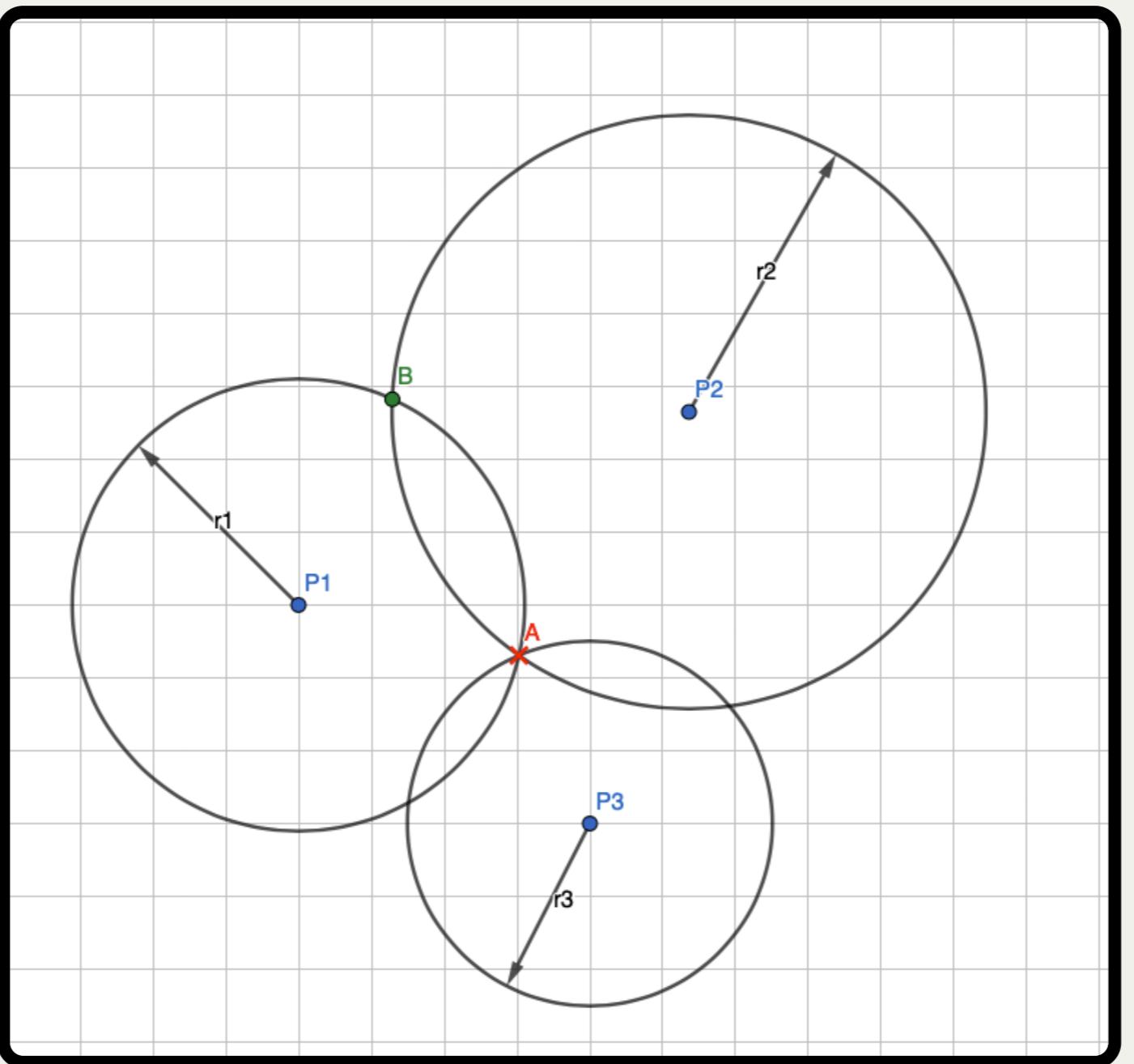


<http://www.land-of-kain.de/> [29.05.2023 - 12:33 Uhr]

# Trilateration







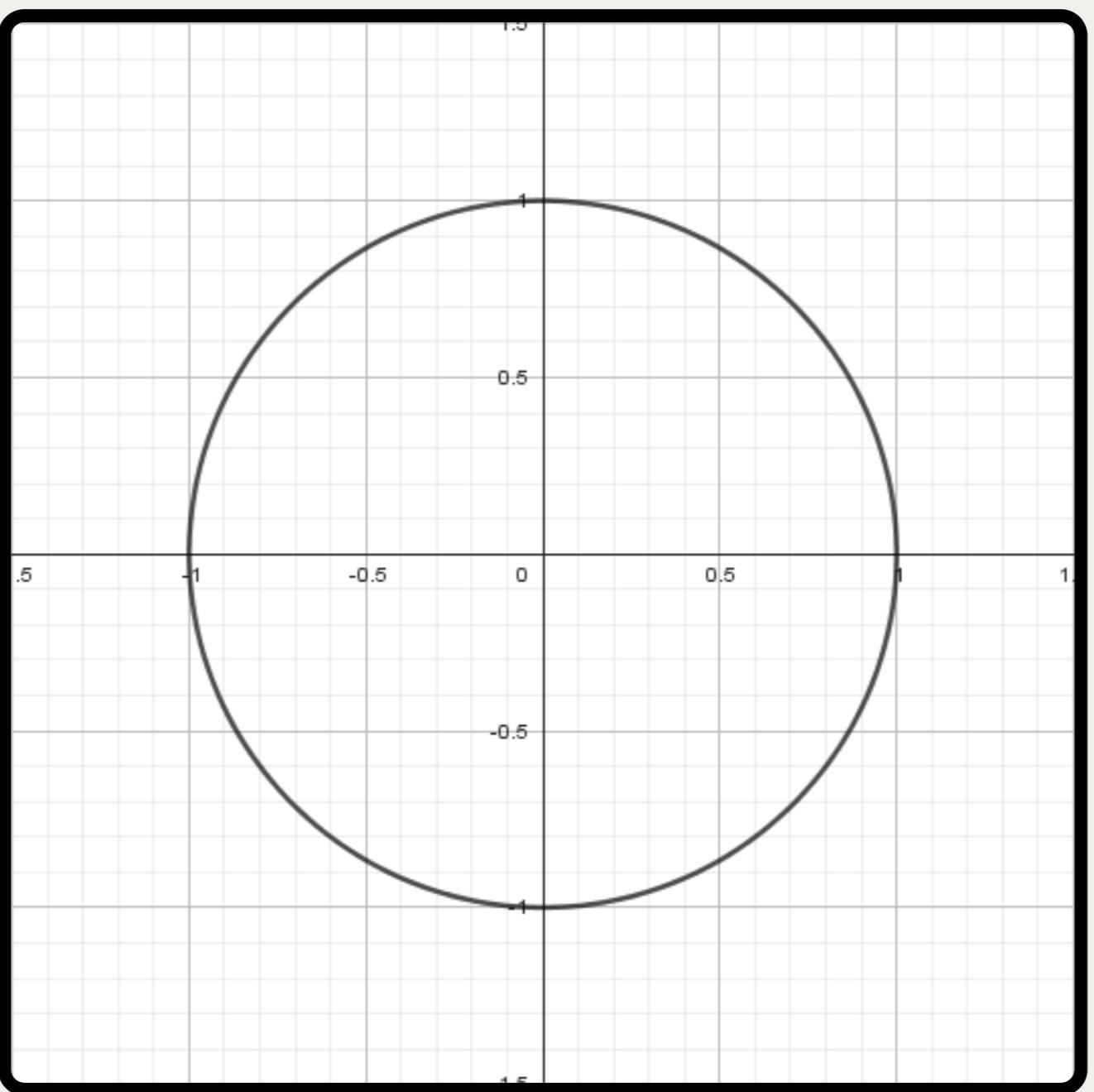
# Mathematische Grundlage

## Trilateration

# Distanz zu Punkt

# Distanz zu Punkt

$$(x - x_r)^2 + (y - y_r)^2 = r^2$$



$$x^2 + y^2 = 1^2$$

Beispiel:

# Beispiel:

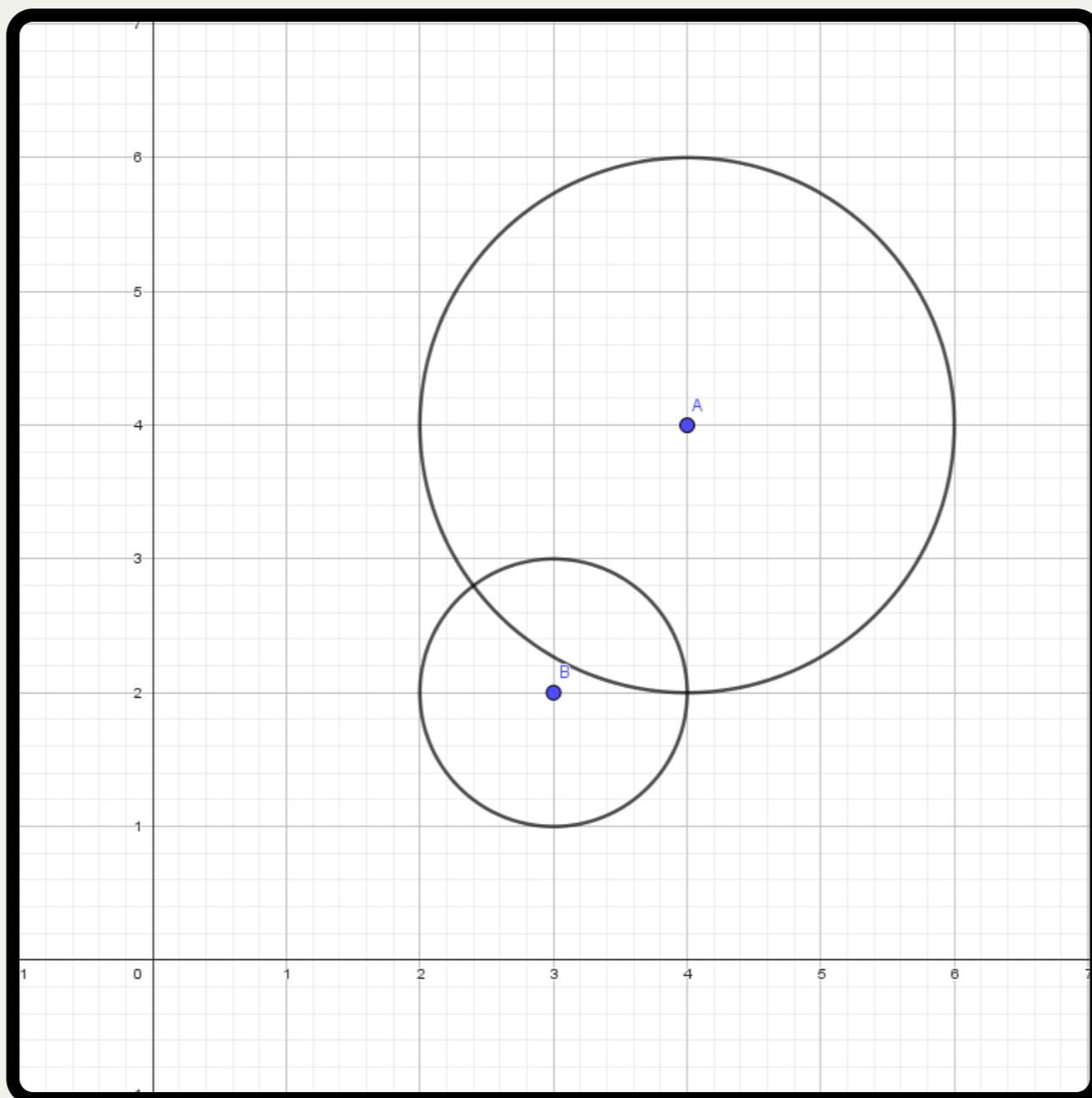
$$x_1 = 3; y_1 = 2; r_1 = 1$$

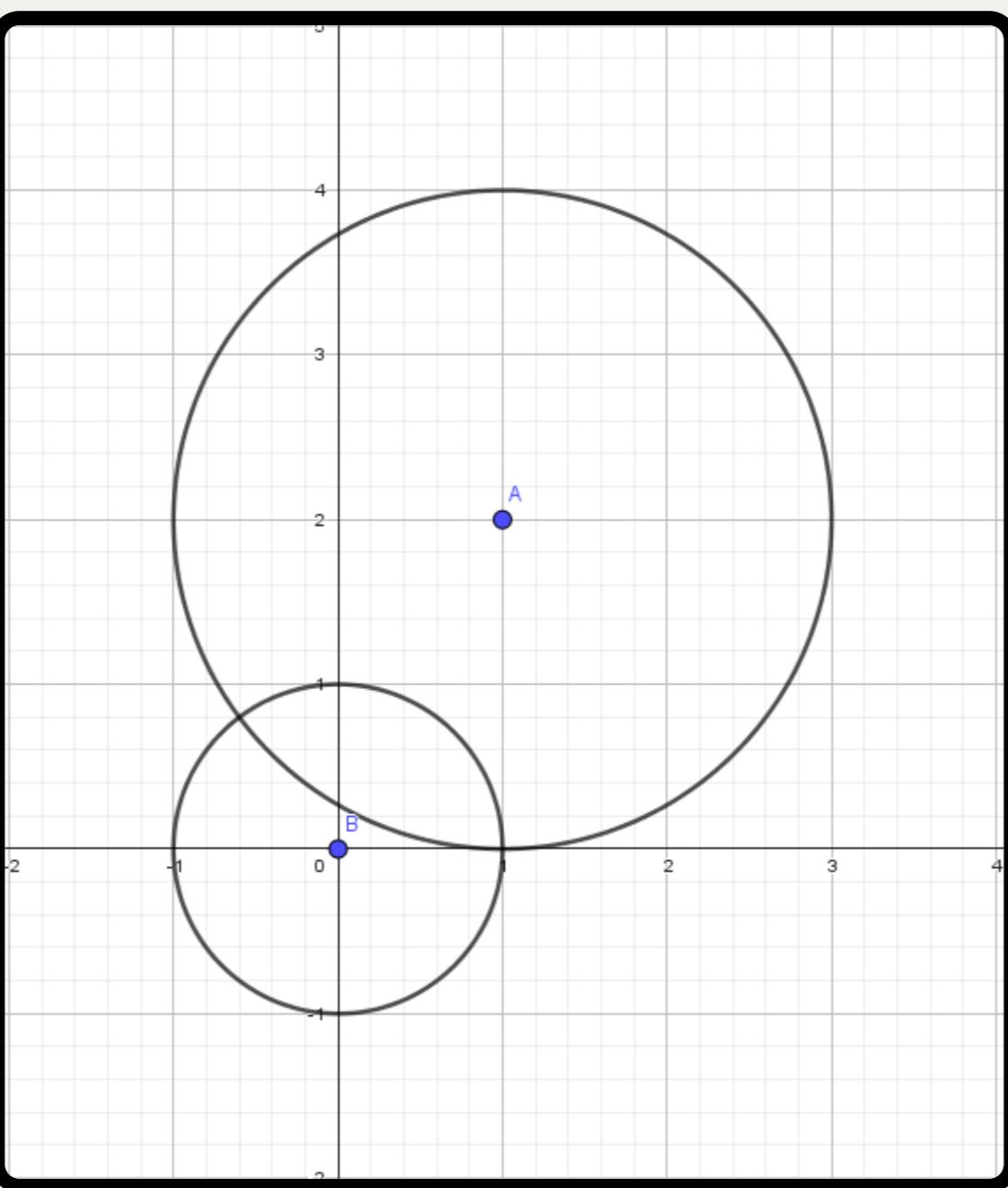
$$x_2 = 4; y_2 = 4; r_2 = 2$$

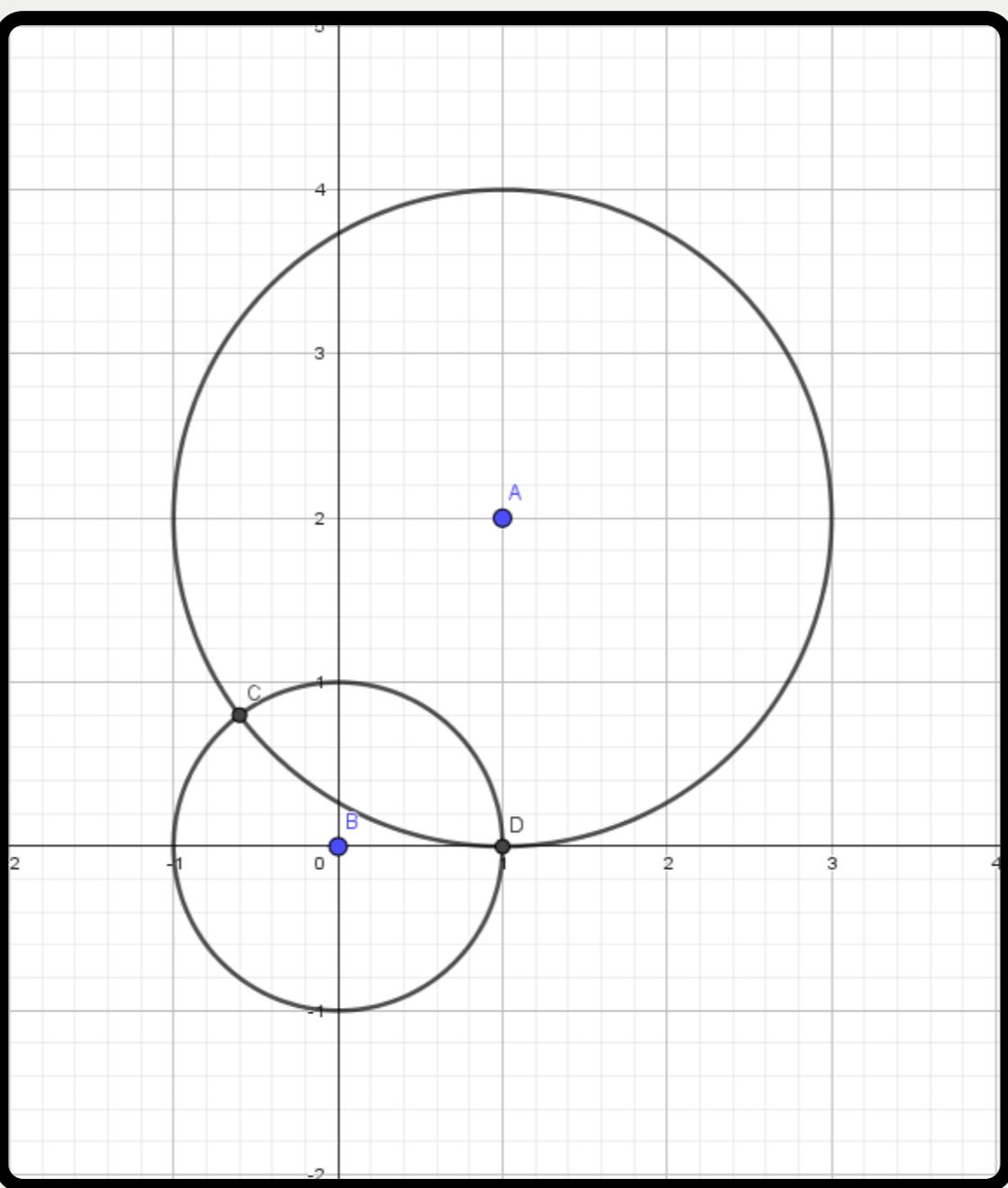
# Beispiel:

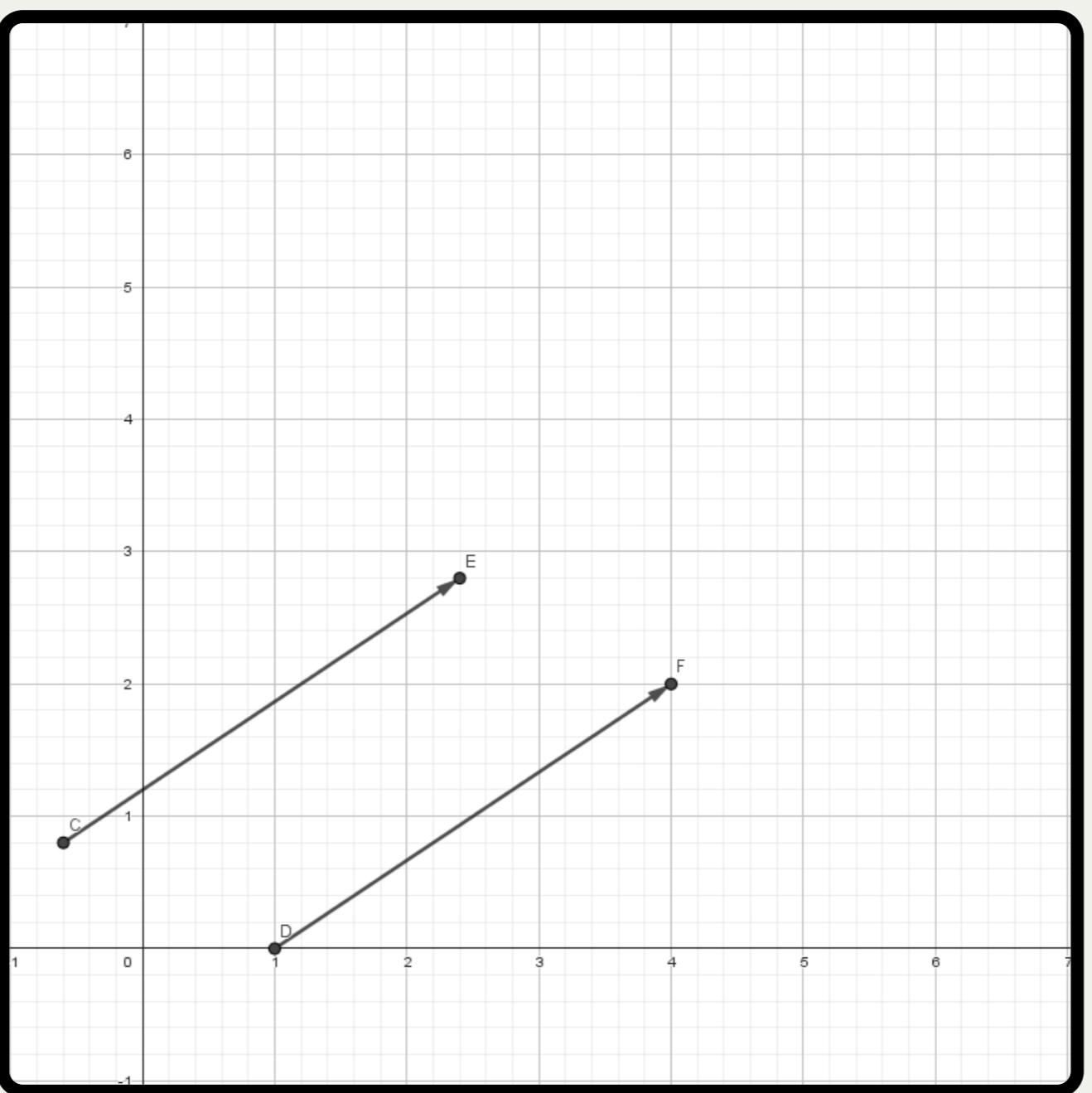
$$x_1 = 3; y_1 = 2; r_1 = 1$$

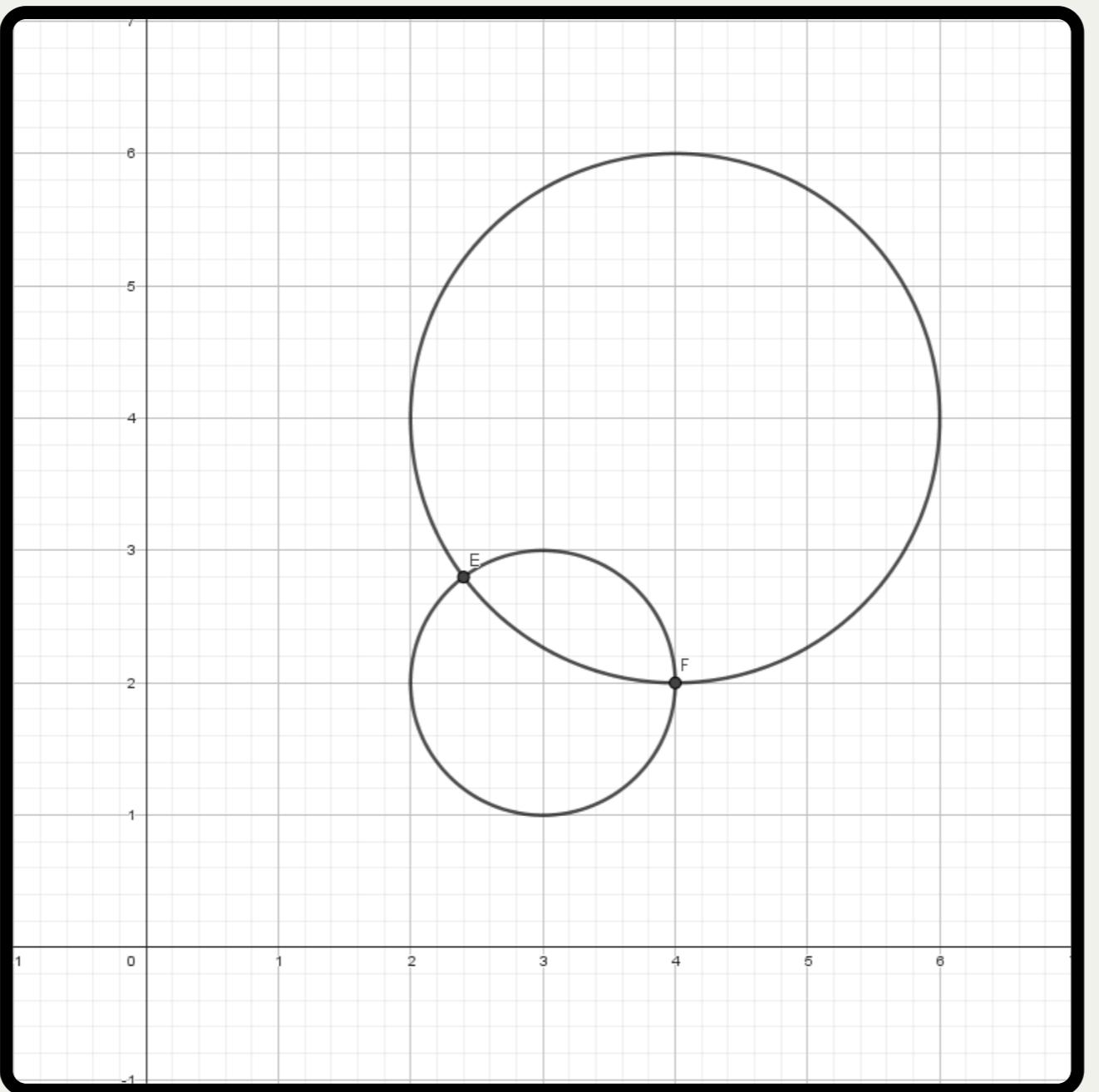
$$x_2 = 4; y_2 = 4; r_2 = 2$$











Schnittpunkte von 2 Kreisen

$$p_1=(x_1;y_1);\;\text{distanz}=r_1$$

$$p_2=(x_2;y_2);\;\text{distanz}=r_2$$

$$p_1 = (x_1; y_1); \text{ distanz} = r_1$$

$$p_2 = (x_2; y_2); \text{ distanz} = r_2$$

$$p_r = (x_2 - x_1; y_2 - y_1)$$

$$x^2+y^2=r_1^2$$

$$(x+x_r)^2+(y+y_r)^2=r_2^2$$

$$(x+x_r)^2+(y+y_r)^2=r_2^2$$

$$(x^2+2xx_r+x_r^2)+(y^2+2yy_r+y_r^2)=r_2^2$$

$$(x^2+2xx_r+x_r^2)+(y^2+2yy_r+y_r^2)-(x^2+y^2)=r_2^2-r_1^2$$

$$2xx_r+x_r^2+2yy_r+y_r^2=r_2^2-r_1^2\quad|- (r_2^2-r_1^2)$$

$$2xx_r+x_r^2+2yy_r+y_r^2-r_2^2+r_1^2=0\quad|-2xx_r$$

$$x_r^2+2yy_r+y_r^2-r_2^2+r_1^2=-2xx_r\quad|\div -2x_r$$

$$\frac{-(-r_2^2+r_1^2+2yy_r+y_r^2+x_r^2)}{2x_r}=x$$

$$x = \frac{r_2^2 - r_1^2 \boxed{-2yy_r} - y_r^2 - x_r^2}{\boxed{2x_r}} = \frac{\boxed{-2yy_r}}{\boxed{2x_r}} + \underbrace{\frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r}}_{\sigma}$$

$$\sigma = \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r}$$

$$x = \frac{-2yy_r}{2x_r} + \sigma$$

$$\boxed{x = \frac{-yy_r}{x_r} + \sigma}$$

$$\boxed{x = \frac{-yy_r}{x_r} + \sigma} \quad \text{einsetzen in} \quad \boxed{x^2 + y^2 = r_1^2}$$

$$\left( \frac{-yy_r}{x_r} + \sigma \right)^2 + y^2 = r_1^2$$

$$\frac{y_r^2}{x_r^2}y^2 - 2\frac{y_r\sigma}{x_r}y + \sigma^2 + y^2 = r_1^2$$

$$\boxed{\left(1 + \frac{y_r^2}{x_r^2}\right)y^2 + \left(-2\frac{y_r\sigma}{x_r}\right)y + (\sigma^2 - r_1^2) = 0}$$

$$\boxed{(1+\frac{y_r^2}{x_r^2})y^2+(-2\frac{y_r\sigma}{x_r})y+(\sigma^2-r_1^2)=0}$$

$$a=1+\frac{y_r^2}{x_r^2}$$

$$b=-2\frac{y_r\sigma}{x_r}$$

$$c=\sigma^2-r_1^2$$

$$ay^2+by+c=0$$

$$\boxed{\frac{-b\pm\sqrt{b^2-4ac}}{2a}=y_{1,2}}$$

$$\begin{aligned} a &= 1 + \frac{y_r^2}{x_r^2} \\ b &= -2\frac{y_r\sigma}{x_r} \\ c &= \sigma^2 - r_1^2 \end{aligned}$$

$$\sigma = \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r}$$

$$\boxed{\frac{2\frac{y_r\sigma}{x_r^2}\pm\sqrt{4\frac{y_r^2\sigma^2}{x_r^2}-4\cdot(1+\frac{y_r^2}{x_r^2})\cdot(\sigma^2-r_1^2)}}{2+2\frac{y_r^2}{x_r^2}}=y_{1,2}}$$



$$\frac{2 \frac{y_r \left( \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r} \right)}{x_r^2} \pm \sqrt{4 \frac{y_r^2 \left( \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r} \right)^2}{x_r^2} - 4 \cdot (1 + \frac{y_r^2}{x_r^2}) \cdot (\left( \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r} \right)^2 - r_1^2)}}{2 + 2 \frac{y_r^2}{x_r^2}} = y_{1,2}$$

$$\frac{2 \frac{x_r \left( \frac{r_2^2 - r_1^2 - x_r^2 - y_r^2}{2y_r} \right)}{y_r^2} \pm \sqrt{4 \frac{x_r^2 \left( \frac{r_2^2 - r_1^2 - x_r^2 - y_r^2}{2y_r} \right)^2}{y_r^2} - 4 \cdot (1 + \frac{x_r^2}{y_r^2}) \cdot (\left( \frac{r_2^2 - r_1^2 - x_r^2 - y_r^2}{2y_r} \right)^2 - r_1^2)}}{2 + 2 \frac{x_r^2}{y_r^2}} = x_{1,2}$$

Schnittpunkte =  $(x_1 + x_b, y_1 + y_b), (x_2 + x_b, y_2 + y_b)$

# Schnittpunkt - Abgleich

# Schnittpunkt - Abgleich

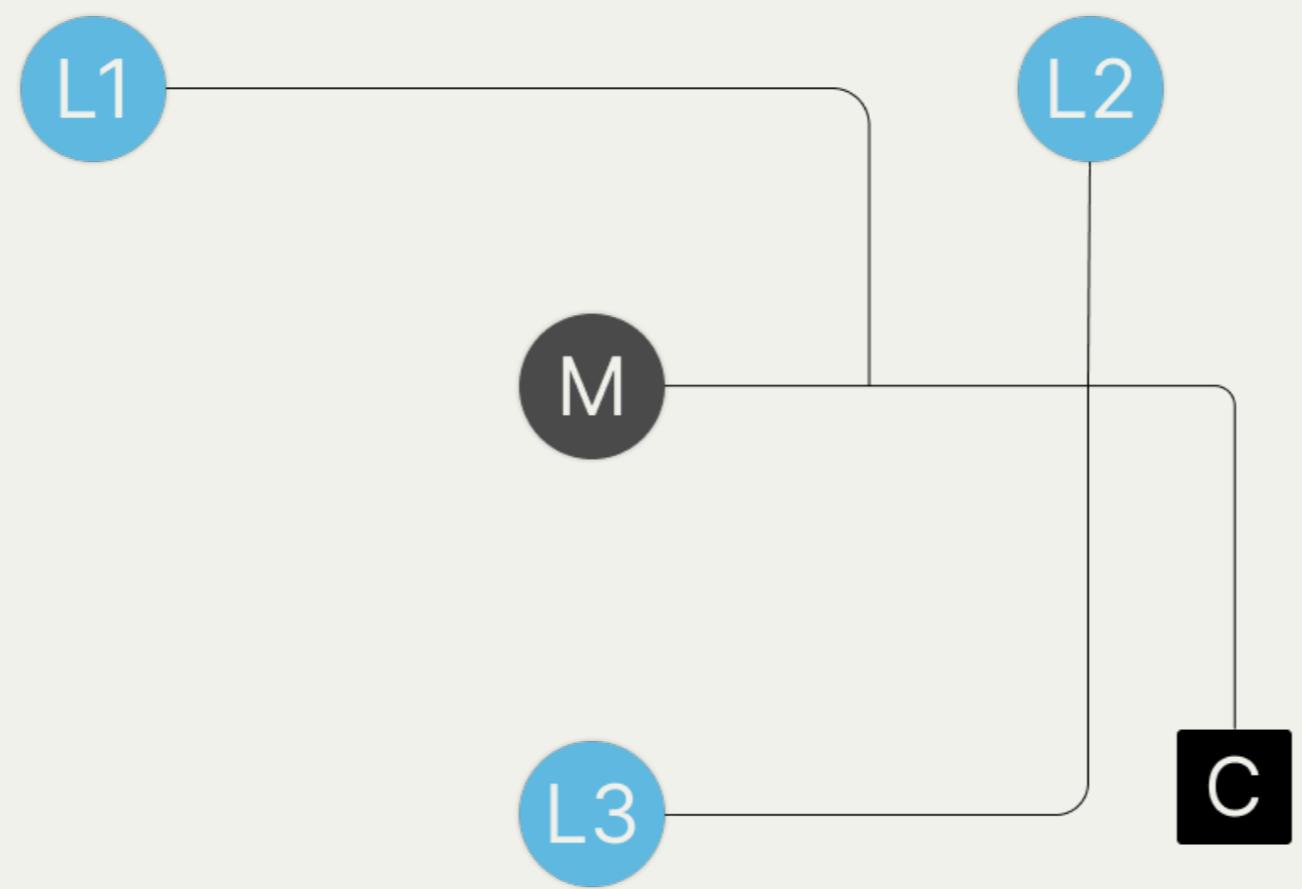
- gemessene Distanz zwischen Satellit und Empfänger  
    ⇒ Distanz zwischen Satellit und Schnittpunkten

# Schnittpunkt - Abgleich

- gemessene Distanz zwischen Satellit und Empfänger  
    ⇒ Distanz zwischen Satellit und Schnittpunkten
- Schnittpunkt mit geringerer Abweichung

# Experiment

# Aufbau



# Durchführung

# Durchführung

1. Aufnahme durch Mikrofon startet

# Durchführung

1. Aufnahme durch Mikrofon startet
2. 2 Sekunden warten

# Durchführung

1. Aufnahme durch Mikrofon startet
2. 2 Sekunden warten
3. Lautsprecher spielen Ton mit fester (jeweils unterschiedlicher) Frequenz

# Durchführung

1. Aufnahme durch Mikrofon startet
2. 2 Sekunden warten
3. Lautsprecher spielen Ton mit fester (jeweils unterschiedlicher) Frequenz
4. 2 Sekunden warten

# Durchführung

1. Aufnahme durch Mikrofon startet
2. 2 Sekunden warten
3. Lautsprecher spielen Ton mit fester (jeweils unterschiedlicher) Frequenz
4. 2 Sekunden warten
5. Aufnahme beenden

# Berechnung I

# Berechnung I

1. iterieren über die Länge der Aufnahme in "Frames"  
(Dauer der Aufnahme / Sample Rate)

# Berechnung I

1. iterieren über die Länge der Aufnahme in "Frames"  
(Dauer der Aufnahme / Sample Rate)
2. interval = [index; index + chunk Size]

# Berechnung I

1. iterieren über die Länge der Aufnahme in "Frames"  
(Dauer der Aufnahme / Sample Rate)
2. interval = [index; index + chunk Size]
3. Fourier Transformation über Intervall  $\Rightarrow$  Lautstärke der gespielten Frequenzen

# Berechnung I

1. iterieren über die Länge der Aufnahme in "Frames"  
(Dauer der Aufnahme / Sample Rate)
2. interval = [index; index + chunk Size]
3. Fourier Transformation über Intervall  $\Rightarrow$  Lautstärke der gespielten Frequenzen
4. ggf. speichere Frame Index

# Berechnung I

1. iterieren über die Länge der Aufnahme in "Frames"  
(Dauer der Aufnahme / Sample Rate)
2. interval = [index; index + chunk Size]
3. Fourier Transformation über Intervall  $\Rightarrow$  Lautstärke der gespielten Frequenzen
4. ggf. speichere Frame Index
5. 
$$t_p = \frac{frameIndex}{sampleRate}$$

# Berechnung II

# Berechnung II

1.

$$t_r = t_p - \text{delay}$$

(delay = 2s)

# Berechnung II

1.

$$t_r = t_p - \text{delay}$$

(delay = 2s)

2.

$$s_r = t_r \cdot 343 \frac{m}{s}$$



```
● ● ●

def getDistanceToSpeaker(self, FREQUENCYPLAY, TIMEPLAYING, DEBUG, VELOCITY, THRESHHOLD):
    # Function to calculate Distance from Speaker and Microphone
    chunk = (TIMEPLAYING + self.delay) * self.RATE

    frequencyPlayed = int((FREQUENCYPLAY - (FREQUENCYPLAY % (self.RATE / self.REALCHUNK))))
    indexOfFrequency = int((FREQUENCYPLAY - (FREQUENCYPLAY % (self.RATE / self.REALCHUNK))) / (self.RATE/self.REALCHUNK))
    # Calculating index of frequency

    audio = pyaudio.PyAudio()
    STREAM = audio.open(format=self.FORMAT, channels=self.CHANNELS,
                         rate=self.RATE, input=True,
                         frames_per_buffer=chunk, input_device_index=1)

    threadPlayer = Thread(target=self.threadPlaySound, args=((frequencyPlayed, TIMEPLAYING)))
    threadPlayer.start()

    timeRecordingStart = time.time_ns() # Messuring recording time
    hugeChunk = STREAM.read(chunk)
    timeRecordingEnd = time.time_ns()

    self.writeToFile("recording.wav", hugeChunk)

    timeTillRecordingRealStarted = ((timeRecordingEnd - timeRecordingStart) * 1e-9) - ((TIMEPLAYING + self.delay))

    frameFoundAt = 0

    bar = progressbar.ProgressBar(max_value=chunk)

    readData = np.frombuffer(hugeChunk, dtype='h') # Converting Chunk to readable Chunk
    readData = np.array(readData, dtype='h')/140 + 255

    indexOfFrequency = int((frequencyPlayed - (frequencyPlayed % (self.RATE / self.REALCHUNK))) / (self.RATE/self.REALCHUNK))
    # Calculating index of frequency we want to listen to

    frameFoundAt = -1
    arr = []

    for i in range(0, (chunk) - self.REALCHUNK * 2): # Checking through all sub-chunks, detects first
        bar.update(i)
        chunk = readData[i:i+self.REALCHUNK*2]
        realValues = np.abs(fft(chunk)[0:self.REALCHUNK]) * 2 / (128 * self.REALCHUNK)
        valueOfFrequency = realValues[indexOfFrequency]
        arr.append(realValues[indexOfFrequency])

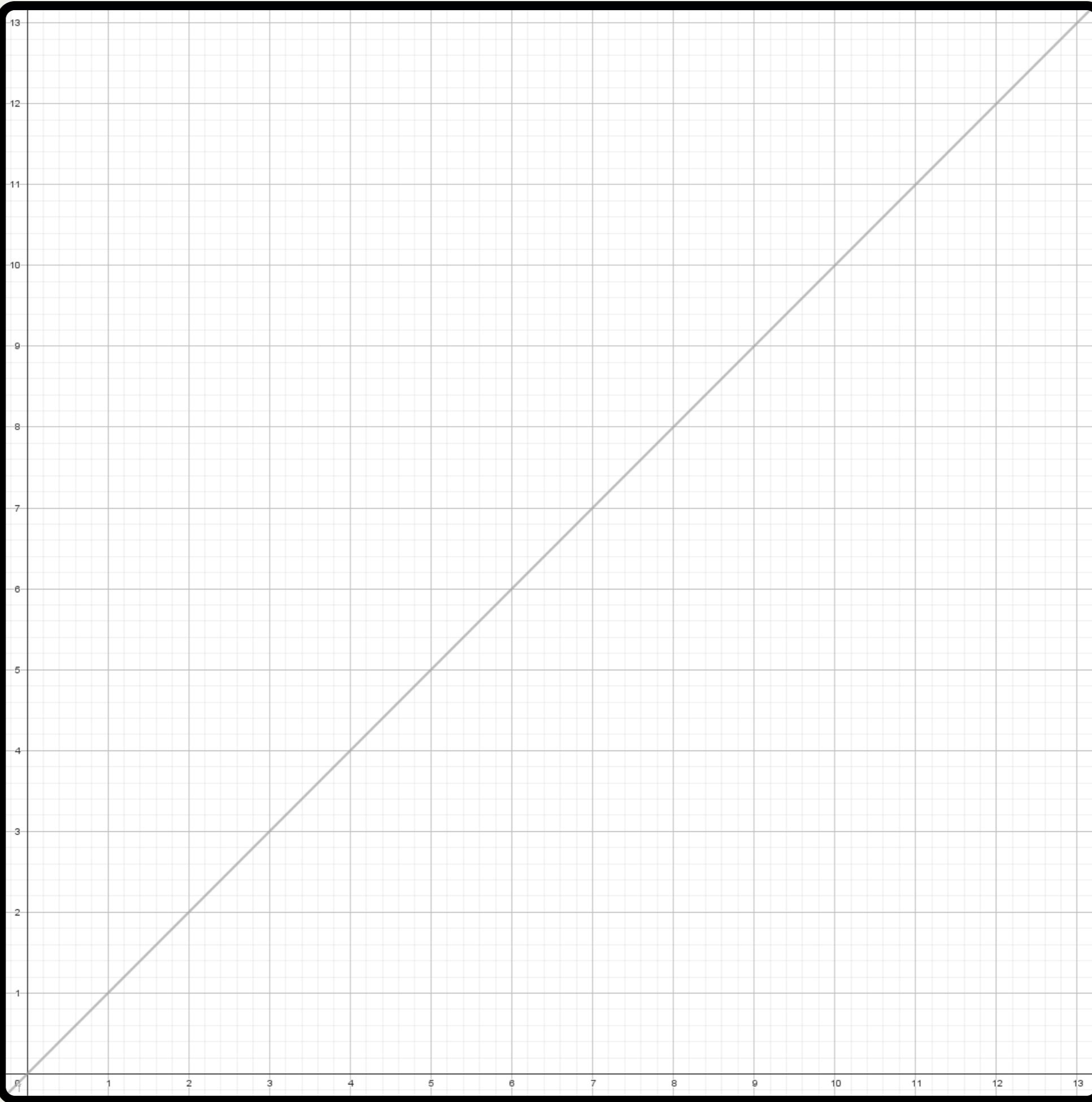
        if (valueOfFrequency > THRESHHOLD and frameFoundAt == -1):
            frameFoundAt = i

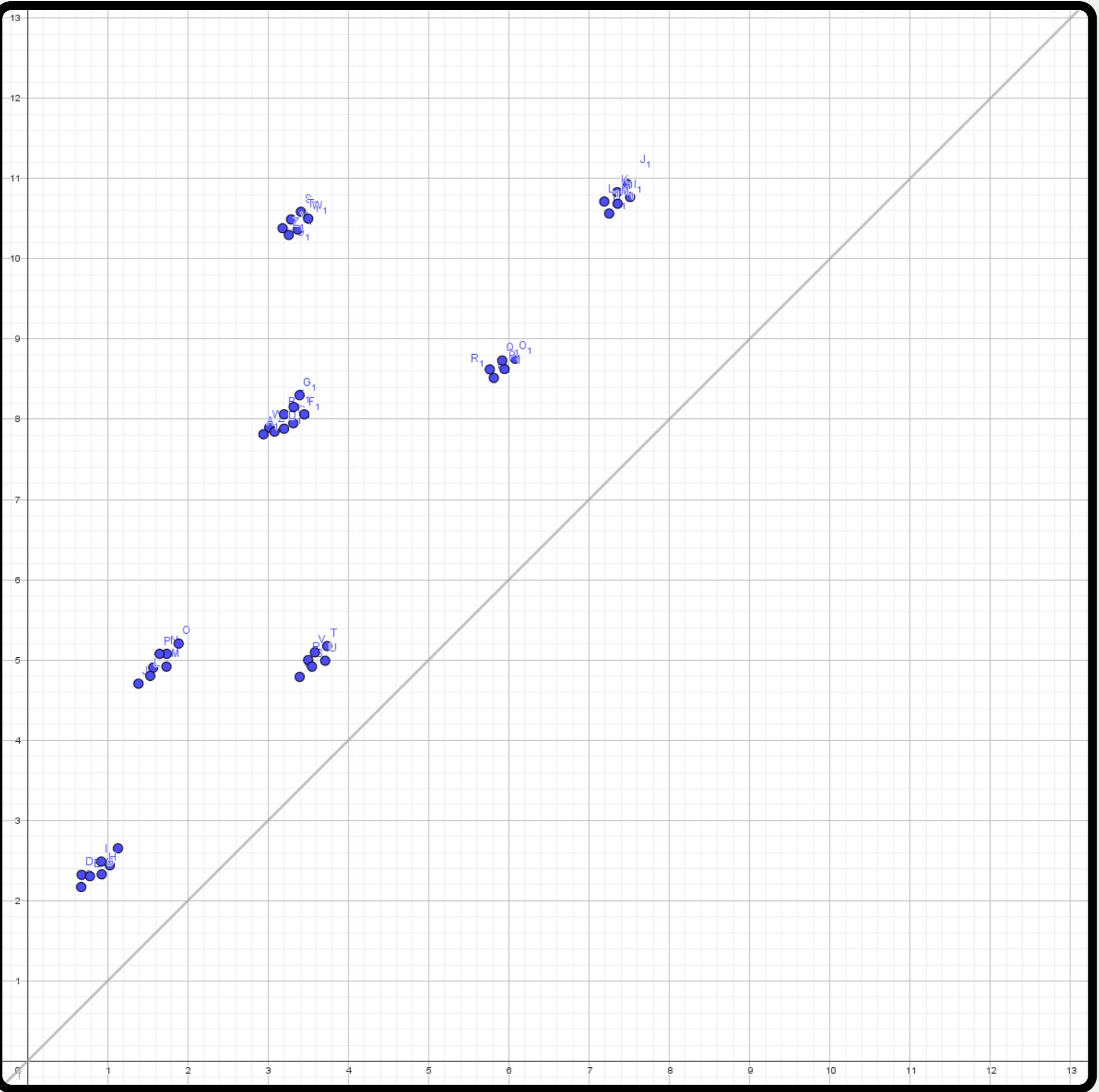
    timeDistance = ((frameFoundAt / self.RATE) + timeTillRecordingRealStarted) * 1e+9 # Time (relative) we heard the sound at
    # (correction for delay in the STREAM.READ() methode) (ns)
    timeUntilSoundPlayed = timeSoundPlayedAt - timeRecordingStart # Time (relative) we heard the sound at (ns)
    timeRec = timeDistance - timeUntilSoundPlayed # Time (relative) it took the sound to travel to the microphone (ns)
    relativeTime = timeRec * 1e-9 # Time (relative) it took the sound to travel to the microphone (s)
    distance = relativeTime * VELOCITY # Distance calculated by the time the sound traveled

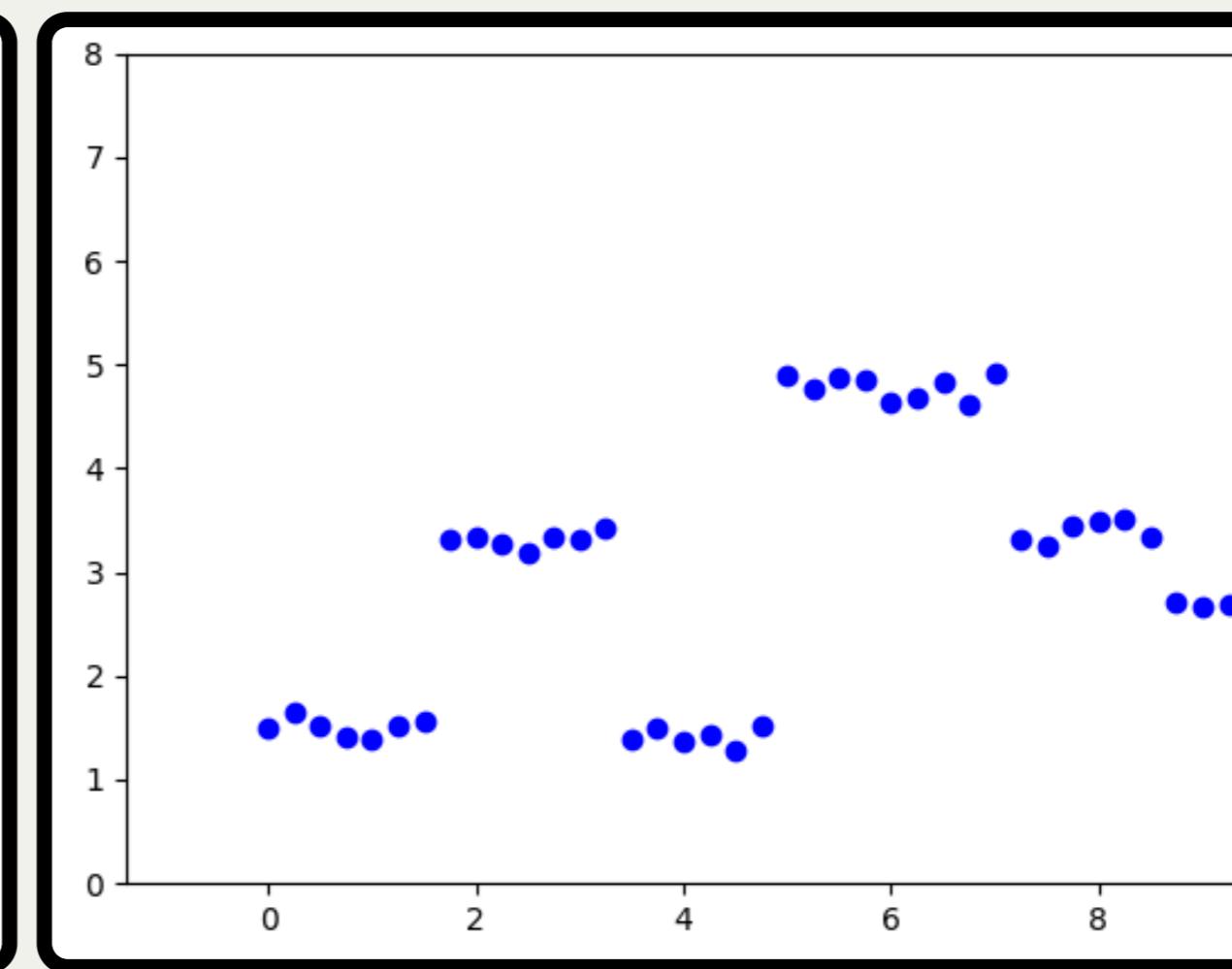
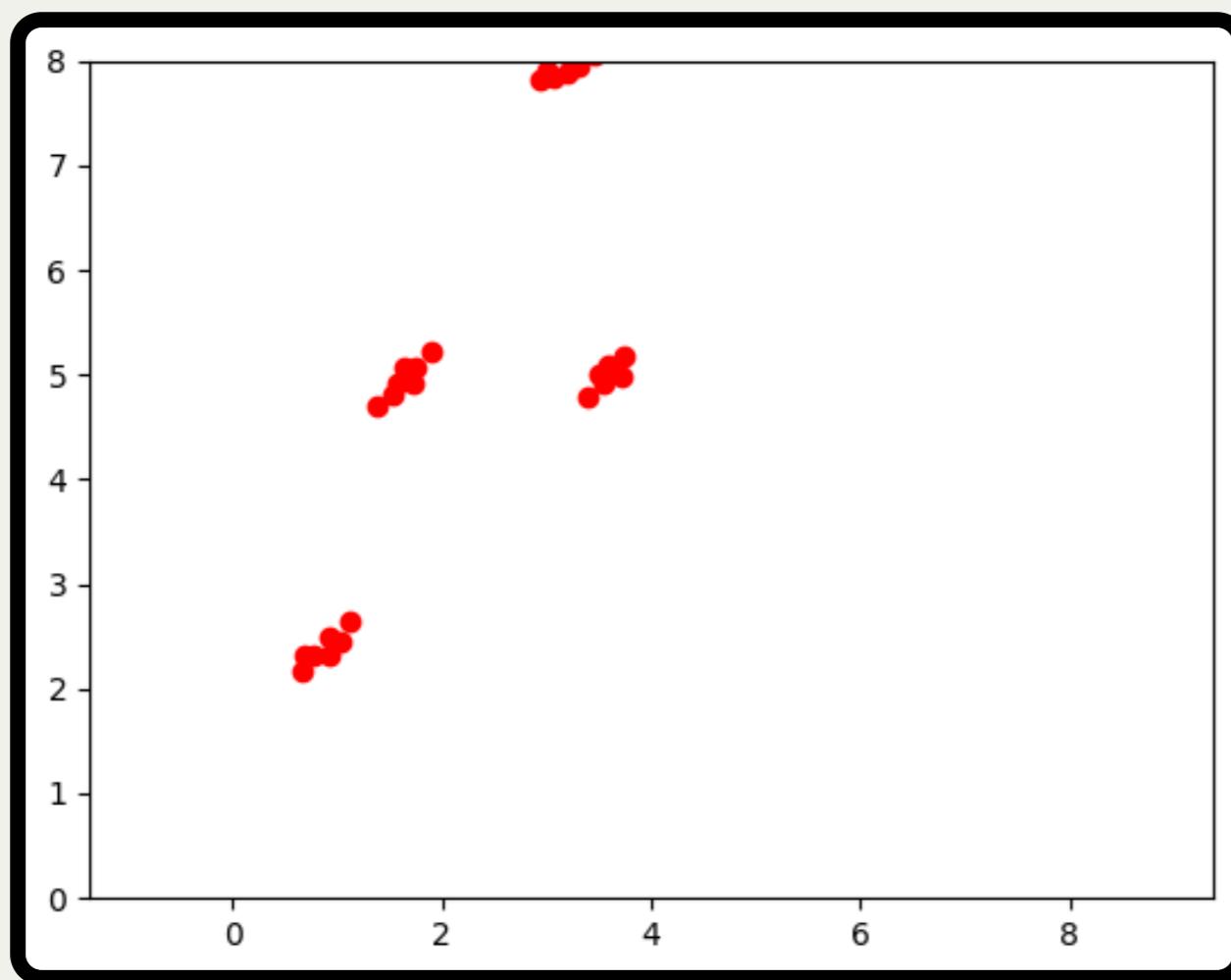
    frameItShouldBeOn = (timeSoundPlayedAt - timeRecordingStart) / self.RATE
    frameItIsOn = frameFoundAt

    return distance
```

# Ergebnisse





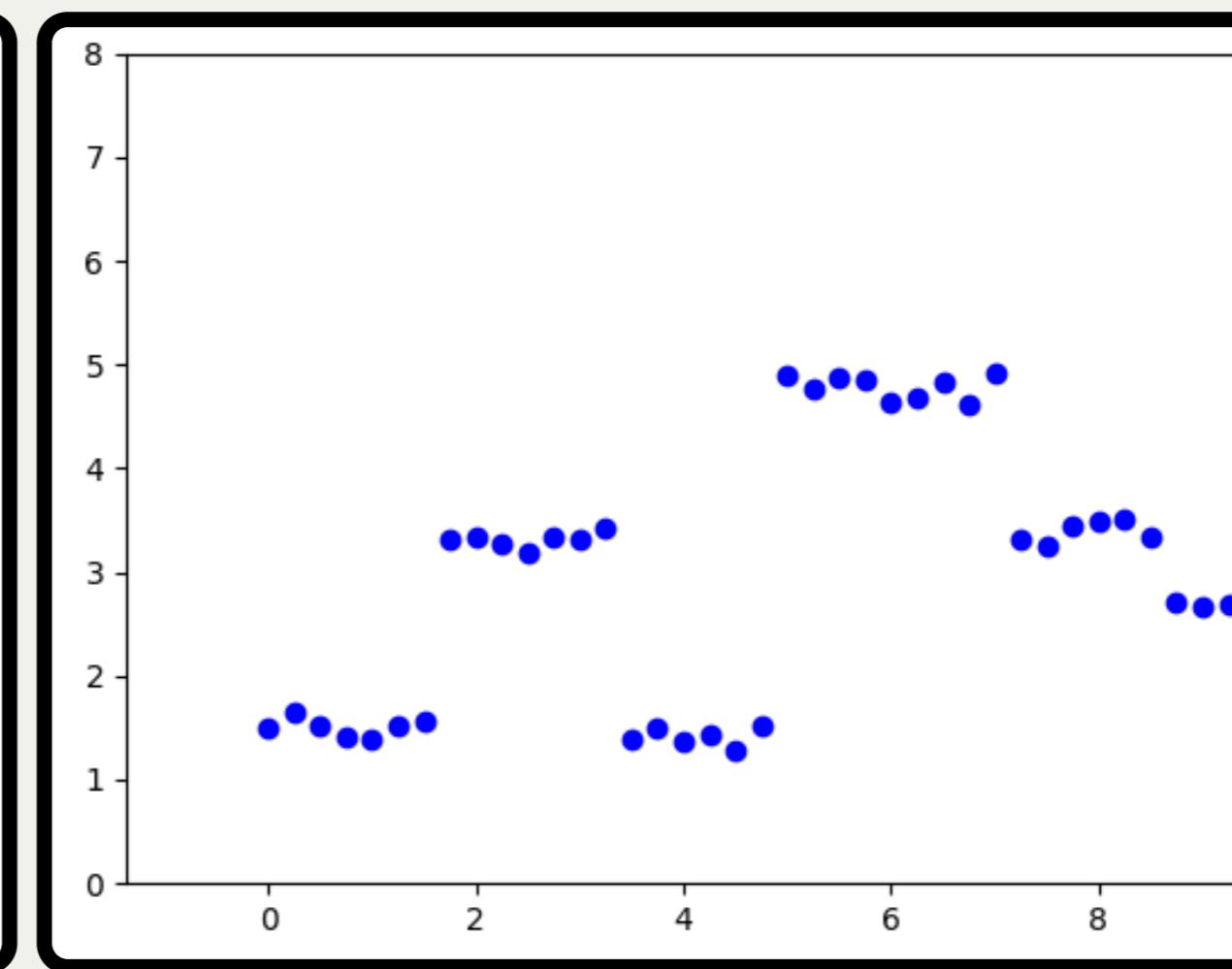
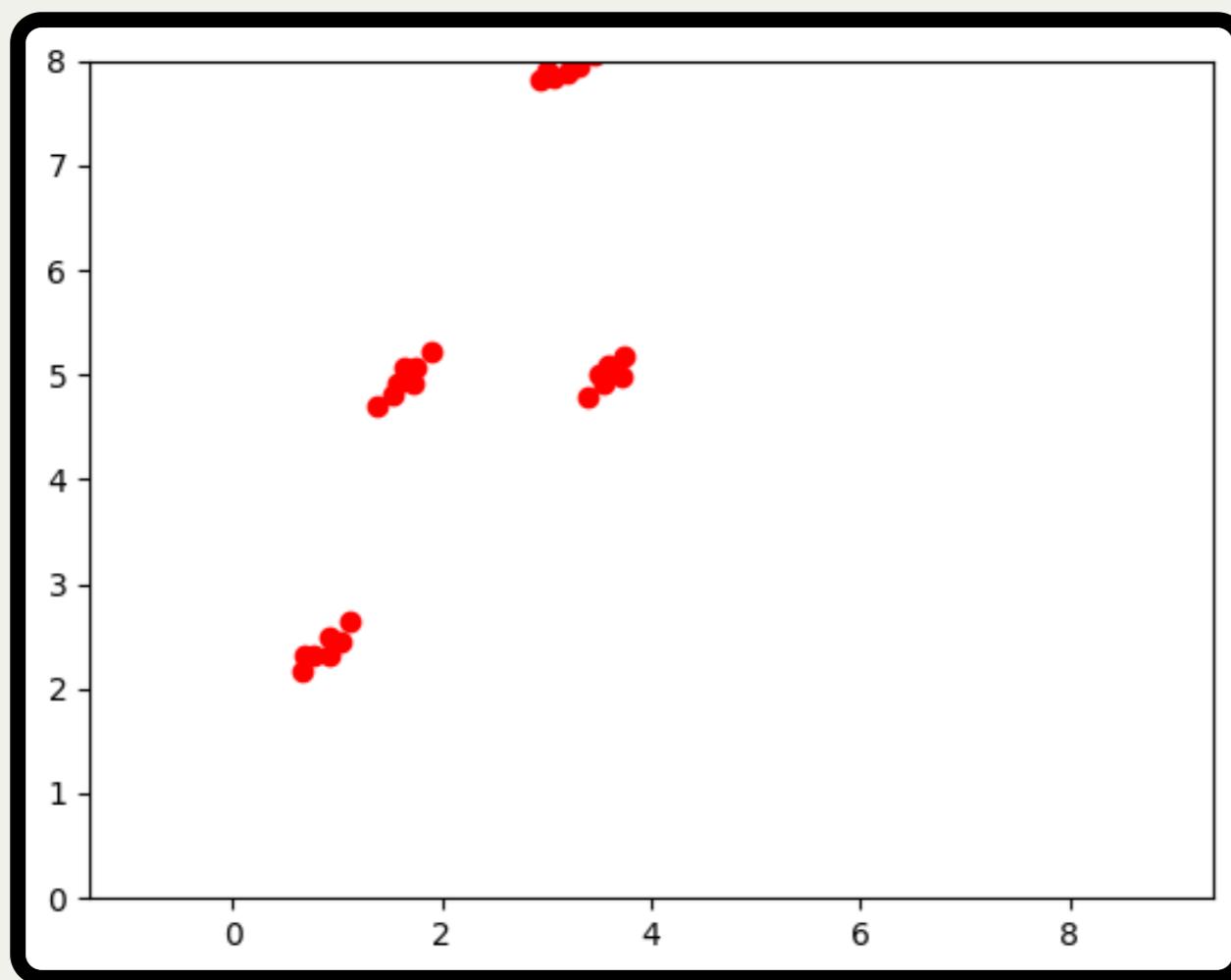


```
1  {
2      'index': 1,
3      'structVersion': 2,
4      'name': 'Mikrofonarray (5- USB PnP Audio)'
5      'hostApi': 0,
6      'maxInputChannels': 2,
7      'maxOutputChannels': 0,
8      'defaultLowInputLatency': 0.09,
9      'defaultLowOutputLatency': 0.09,
10     'defaultHighInputLatency': 0.18,
11     'defaultHighOutputLatency': 0.18,
12     'defaultSampleRate': 44100.0
13 }
```

```
1 {
2     'index': 1,
3     'structVersion': 2,
4     'name': 'Mikrofonarray (5- USB PnP Audio)'
5     'hostApi': 0,
6     'maxInputChannels': 2,
7     'maxOutputChannels': 0,
8     'defaultLowInputLatency': 0.09,
9     'defaultLowOutputLatency': 0.09,
10    'defaultHighInputLatency': 0.18,
11    'defaultHighOutputLatency': 0.18,
12    'defaultSampleRate': 44100.0
13 }
```

$$\text{delay}_{\text{aufnahme}} = [0.09s; 0.18s]$$

$$\text{abweichung}_{\text{aufnahme}} = [0.09s \cdot 343 \frac{m}{s}; 0.18s \cdot 343 \frac{m}{s}] = [30.87m; 61.74m]$$



```
1 {
2     'index': 19,
3     'structVersion': 2,
4     'name': 'Realtek HD Audio 2nd output',
5     'hostApi': 0,
6     'maxInputChannels': 0,
7     'maxOutputChannels': 2,
8     'defaultLowInputLatency': 0.09,
9     'defaultLowOutputLatency': 0.09,
10    'defaultHighInputLatency': 0.18,
11    'defaultHighOutputLatency': 0.18,
12    'defaultSampleRate': 44100.0
13 }
```

```
1 {
2     'index': 19,
3     'structVersion': 2,
4     'name': 'Realtek HD Audio 2nd output',
5     'hostApi': 0,
6     'maxInputChannels': 0,
7     'maxOutputChannels': 2,
8     'defaultLowInputLatency': 0.09,
9     'defaultLowOutputLatency': 0.09,
10    'defaultHighInputLatency': 0.18,
11    'defaultHighOutputLatency': 0.18,
12    'defaultSampleRate': 44100.0
13 }
```

$$\text{delay}_{\text{abspielen}} = [0.09s; 0.18s]$$

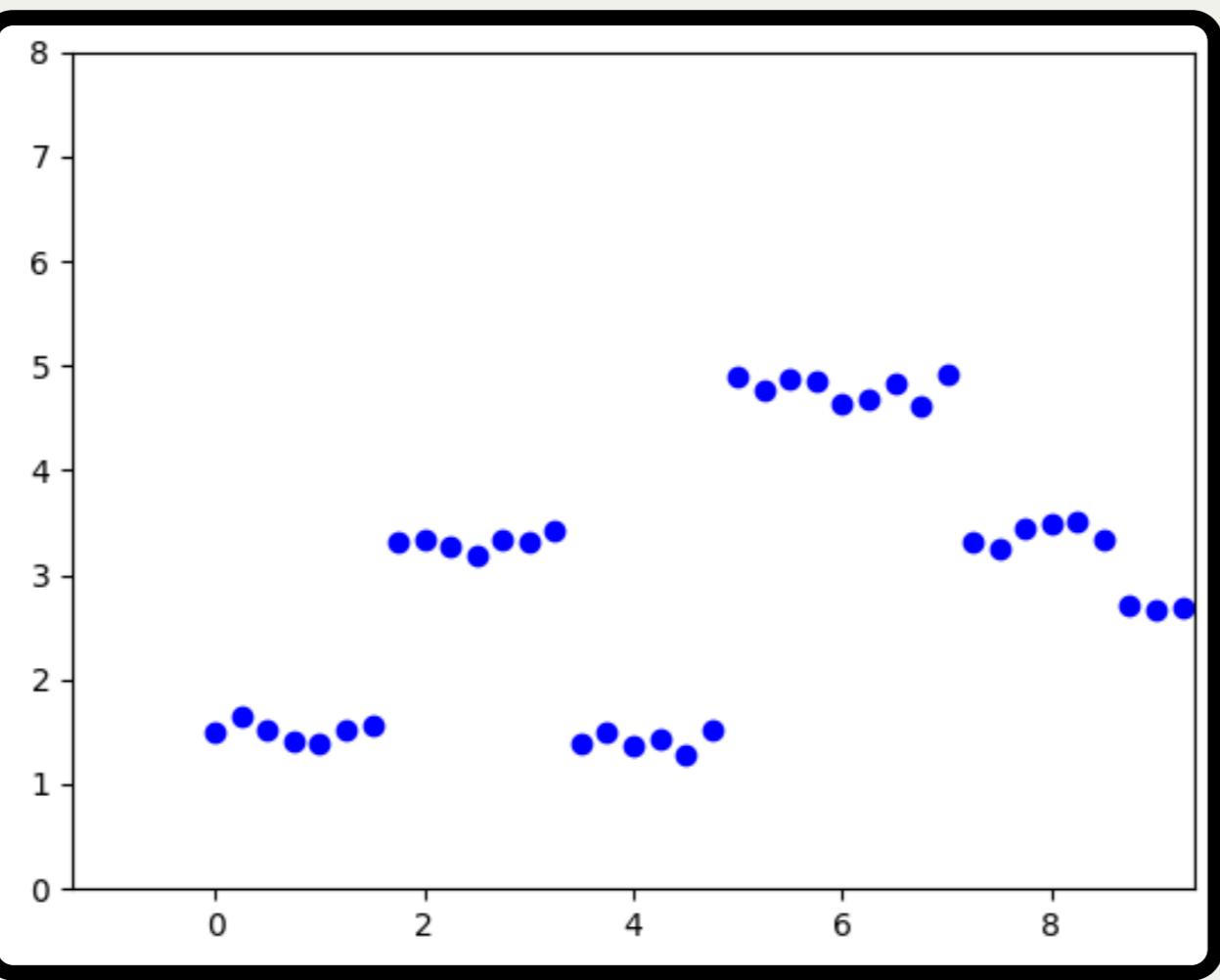
$$\text{abweichung}_{\text{abspielen}} = [0.09s \cdot 343 \frac{m}{s}; 0.18s \cdot 343 \frac{m}{s}] = [30.87m; 61.74m]$$

$$\text{abweichung}_{gesamt-minimal} = \min(\text{abweichung}_{abspielen}) - \max(\text{abweichung}_{aufnahme})$$

$$\text{abweichung}_{gesamt-minimal} = 30.87m - 61.74m = -30.87m$$

$$\text{abweichung}_{gesamt-maximal} = \max(\text{abweichung}_{abspielen}) - \min(\text{abweichung}_{aufnahme})$$

$$\text{abweichung}_{gesamt-maximal} = 61.74m - 30.87m = 30.87m$$



# Probleme

# Hardware Limitationen

## Hardware Limitationen

- Audio-Treiber Latenz nicht nachvollziehbar  
⇒ stark abweichende Resultate

# Lösungen

Lightweight Umgebung wählen:

# Lösungen

Lightweight Umgebung wählen:

- Linux

# Lösungen

Lightweight Umgebung wählen:

- Linux
- Arduino

# Lösungen

Lightweight Umgebung wählen:

- Linux
- Arduino
- Raspberry Pi

# Simulation



(960, 120)



(960, 400)



(1344, 480)



(634, 584)

0.82s

1.09s

1.14s

No Int.

Delay

343 m/s

Ergebnisse

# Berechnung



# Distanzen:

Satelite 0:  $0.82\text{s} \cdot 343\text{ m/s} \Rightarrow 280.00\text{m}$

Satelite 1:  $1.09\text{s} \cdot 343\text{ m/s} \Rightarrow 374.69\text{m}$

Satelite 2:  $1.14\text{s} \cdot 343\text{ m/s} \Rightarrow 392.24\text{m}$

# Distanzen:

Satelite 0:  $0.82\text{s} \cdot 343 \text{ m/s} \Rightarrow 280.00\text{m}$

Satelite 1:  $1.09\text{s} \cdot 343 \text{ m/s} \Rightarrow 374.69\text{m}$

Satelite 2:  $1.14\text{s} \cdot 343 \text{ m/s} \Rightarrow 392.24\text{m}$

# Positionen:

Satelite 0 : (960, 120)

Satelite 1 : (634, 584)

Satelite 2 : (1344, 480)

$$a = 1 + \frac{y_r^2}{x_r^2}$$

$$b = -2\frac{y_r\sigma}{x_r}$$

$$c = \sigma^2 - r_1^2$$

$$\sigma = \frac{r_2^2 - r_1^2 - y_r^2 - x_r^2}{2x_r}$$

$$\boxed{\frac{2\frac{y_r\sigma}{x_r^2} \pm \sqrt{4\frac{y_r^2\sigma^2}{x_r^2} - 4\cdot\left(1+\frac{y_r^2}{x_r^2}\right)\cdot\left(\sigma^2 - r_1^2\right)}}{2+2\frac{y_r^2}{x_r^2}} = y_{1,2}}$$

$$x_{1,2} = \frac{-(-581.90) \pm \sqrt{(-581.90)^2 - 4 \cdot (1.49) \cdot (30676.56)}}{2 \cdot 1.49} = [326.40, 62.87]$$

$$x_{real} = [x_1 + x_{sat}, x_2 + x_{sat}] = [326.40 + 633.60, 62.87 + 633.60] = [960.00, 696.47]$$

$$y_{1,2} = \frac{-(1671.68) \pm \sqrt{(1671.68)^2 - 4 \cdot (3.02) \cdot (205314.15)}}{2 \cdot 3.02} = [-184.00, -369.38]$$

$$y_{real} = [y_1 + y_{sat}, y_2 + y_{sat}] = [-184.00 + 584.00, -369.38 + 584.00] = [400.00, 214.62]$$

$$p_{1,2} = [(960, 400), (696, 215)]$$

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3:

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m  
 $(960, 400) \Rightarrow$  Abstand: 392.24m

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m

(960, 400)  $\Rightarrow$  Abstand: 392.24m

(696, 215)  $\Rightarrow$  Abstand: 699.80m

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m

(960, 400)  $\Rightarrow$  Abstand: 392.24m  $\Rightarrow$  0m

(696, 215)  $\Rightarrow$  Abstand: 699.80m

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m

(960, 400)  $\Rightarrow$  Abstand: 392.24m  $\Rightarrow$  0m

(696, 215)  $\Rightarrow$  Abstand: 699.80m  $\Rightarrow$  307.5530207841283m

Simulation

# Ergebnisse

## Schnittpunkte von Satellit eins und Satellit zwei

Radius R3: 392.24m

(960, 400)  $\Rightarrow$  Abstand: 392.24m  $\Rightarrow$  0m

(696, 215)  $\Rightarrow$  Abstand: 699.80m  $\Rightarrow$  307.5530207841283m

Schnittpunkt mit geringerer Abweichung: (960, 400)

Simulation

# Fazit

*„Inwiefern wäre ein schallbasiertes Ortungssystem umsetzbar und nützlich?“*

# Umsetzbarkeit

# Funktionierendes Grundsystem

# Funktionierendes Grundsystem

- starke Einschränkungen in der Praxis

# Funktionierendes Grundsystem

- starke Einschränkungen in der Praxis
- Störgeräusche und Reflexionen

# Funktionierendes Grundsystem

- starke Einschränkungen in der Praxis
- Störgeräusche und Reflexionen
- Distanz-Einschränkung durch Luftreibung und Hindernisse

# Nützlichkeit

# Nützlichkeit

Hohe Genauigkeit

# Ursprüngliche Anwendungsbeispiele

# Autonomes Fahren

# Autonomes Fahren

- Doppler-Effekt durch Bewegung der Autos mit unbekannter Geschwindigkeit

# Autonomes Fahren

- Doppler-Effekt durch Bewegung der Autos mit unbekannter Geschwindigkeit
- starke störgeräusche durch andere Autos, Straßenlärm und Gegenwind

# Autonomes Fahren

- Doppler-Effekt durch Bewegung der Autos mit unbekannter Geschwindigkeit
  - starke störgeräusche durch andere Autos, Straßenlärm und Gegenwind
- ⇒ nicht anwendbar

Staubsaugroboter

# Staubsaugroboter

- durch geringe Geschwindigkeit Doppler-Effekt vernachlässigbar

# Staubsaugroboter

- durch geringe Geschwindigkeit Doppler-Effekt vernachlässigbar
- Positionsbestimmung basierend auf Nachverfolgung von Bewegungen

# Staubsaugroboter

- durch geringe Geschwindigkeit Doppler-Effekt vernachlässigbar
- Positionsbestimmung basierend auf Nachverfolgung von Bewegungen
- Präzision nicht umbedingt benötigt

# Ausblick

# Experiment

# Experiment

- weiterführung auf anderer Platform

# Experiment

- weiterführung auf anderer Platform
- optimierung der Genauigkeit durch bessere Hardware und Trilaterationsverfahren mit mehrfachen Satelliten

# Experiment

- weiterführung auf anderer Platform
- optimierung der Genauigkeit durch bessere Hardware und Trilaterationsverfahren mit mehrfachen Satelliten
- optimierung der Laufzeit durch implementierung in C und implementierung von "Binärer Suche"

# Quellen

# Internetquellen

- <https://www.chemie.de/lexikon/Funkwelle.html> [16-10-2022 16:10 Uhr]
- <https://www.hans-riegel-fachpreise.com/fileadmin/hans-riegel-fachpreise/Module/ausgezeichnete-arbeiten/hans-riegel-fachpreise-seminararbeit-vwa-2014-koch.pdf> [10-12-2022 17:40 Uhr]
- <https://www.ias.ac.in/article/fulltext/sadh/001/03/0235-0249> [10-12-2022 18:00 Uhr]
- [https://de.wikipedia.org/wiki/Differential\\_Global\\_Positioning\\_System](https://de.wikipedia.org/wiki/Differential_Global_Positioning_System) [11-12-2022 17:45 Uhr]
- [https://de.wikipedia.org/wiki/Global\\_Positioning\\_System](https://de.wikipedia.org/wiki/Global_Positioning_System) [12-12-2022 18:00 Uhr]
- [https://www.researchgate.net/figure/Trilateration-in-a-2-D-space-with-two-anchor-points\\_fig1\\_224115960](https://www.researchgate.net/figure/Trilateration-in-a-2-D-space-with-two-anchor-points_fig1_224115960) [16-12-2022 18:40 Uhr]
- <https://www.oxts.com/de/trilateration/> [17-12-2022 15:25 Uhr]
- <https://www.youtube.com/watch?v=nmgFG7PUHfo> [20-12-2022 18:30 Uhr]
- [https://en.wikipedia.org/wiki/Fast\\_Fourier\\_transform](https://en.wikipedia.org/wiki/Fast_Fourier_transform) [21-12-2022 19:40 Uhr]
- <https://www.youtube.com/watch?v=spUNpyF58BY> [21-12-2022 23:20 Uhr]
- [https://www.youtube.com/watch?v=mkGsMWi\\_j4Q](https://www.youtube.com/watch?v=mkGsMWi_j4Q) [22-12-2022 18 Uhr]
- [https://en.wikipedia.org/wiki/Goertzel\\_algorithm](https://en.wikipedia.org/wiki/Goertzel_algorithm) [23-12-2022 18:00 Uhr]
- <https://courses.cs.washington.edu/courses/cse466/12au/calendar/Goertzel-EETimes.pdf> [23-12-2022 18:30 Uhr]
- <https://www.youtube.com/watch?v=aQKX3mrDFoY> [24-12-2022 10:20 Uhr]
- <https://pypi.org/project/sounddevice/> [02-01-2023 11:30 Uhr]
- <https://pypi.org/project/matplotlib/> [02-01-2023 12:10 Uhr]
- <https://docs.scipy.org/doc/scipy/reference/fft.html#module-scipy.fft> [16-02-2023 19:00 Uhr]
- <https://numpy.org/doc/stable/user/> [16-02-2023 19:05 Uhr]
- <https://revealjs.com/> [04-03-2023 17:30 Uhr]
- <https://brockhaus.de/ecs/enzy/article/fourier-transformation> [08-04-2023 18:00 Uhr]
- [https://www.hs-albsig.de/fileadmin/user\\_upload/hsas/02\\_organisationseinheiten/studentische\\_abteilung/downloads/formulare/eigenstaendigkeitserklaerung\\_20190117.pdf](https://www.hs-albsig.de/fileadmin/user_upload/hsas/02_organisationseinheiten/studentische_abteilung/downloads/formulare/eigenstaendigkeitserklaerung_20190117.pdf) [10-05-2023 18:30 Uhr]
- <https://home.uni-leipzig.de/jtrommer/phonetik07/k2b.pdf> [04-03-2023 16:00 Uhr]
- <https://getcssscan.com/css-checkboxes-examples> [24-05-2023 15:00 Uhr]

# Bildquellen

- <http://www.land-of-kain.de/> [29.05.2023 - 12:33 Uhr]
- <https://www.nti-audio.com/> [25.05.2023 - 11:24 Uhr]
- <https://www.youtube.com/watch?v=spUNpyF58BY> [23.05.2023 - 16:53 Uhr]
- <https://www.flippingphysics.com/> [29.05.2023 - 12:55]
- <https://www.glastroesch.com/> [29.05.2023 - 01:35]
- <https://www.weltderphysik.de/> [29.05.2023 - 01:52]
- (Ladefoged, 1996:4) [29.05.2023 - 00:29]
- <https://studyflix.de/ingenieurwissenschaften/wellenlange-4447> [29.05.2023 - 17:33]
- <https://www.adac.de/rund-ums-fahrzeug/ausstattung-technik-zubehoer/autonomes-fahren/technik-vernetzung/aktuelle-technik/> [23.05.2023 - 15:59 Uhr]
- <https://www.digitaltrends.com/> [29.05.2023 - 12:52 Uhr]
- <https://de.wikipedia.org/wiki/Datei:Trilateration.png> [23.05.2023 - 13:23 Uhr]
- <https://de.wikipedia.org/wiki/Datei:Trilateration.png> [23.05.2023 - 13:25 Uhr]

