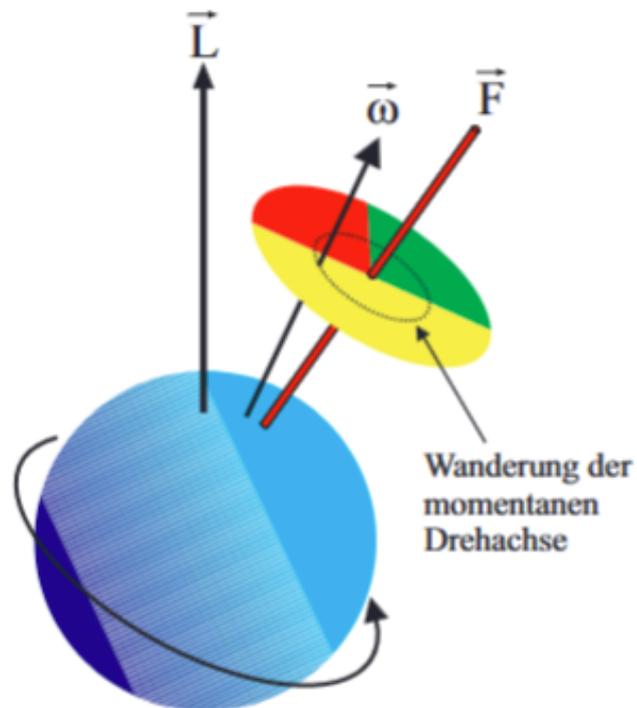


# Versuch 213 - Keisel

## PAP 2.1, [1]

13.01.2025



[2]

Teilnehmender Student: **Jonathan Rodemers**

Gruppe des Teilnehmenden: 4

Kurs: Montag

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Messverfahren . . . . .	1
1.3	Grundlagen aus der Physik . . . . .	1
1.3.1	Drehimpuls und Trägheitsmoment . . . . .	1
1.3.2	Präzession . . . . .	1
1.3.3	Nutation . . . . .	2
1.3.4	Dämpfung . . . . .	2
<b>2</b>	<b>Durchführung</b>	<b>3</b>
2.1	Messprotokol . . . . .	3
<b>3</b>	<b>Auswertung</b>	<b>6</b>
3.1	Dämpfung des Kreisels . . . . .	6
3.2	Präzessionsvergleich . . . . .	6
3.3	Bestimmung des Trägheitsmoments . . . . .	7
3.4	Vergleich der Trägheitsmomente . . . . .	8
3.5	Bestimmung von $I_x$ . . . . .	9
3.6	Bestimmung von $I_x$ die Zweite . . . . .	9
<b>4</b>	<b>Zusammenfassung und Diskussion</b>	<b>11</b>
<b>5</b>	<b>Anhang</b>	<b>12</b>
	<b>Quellen- und Literaturverzeichnis</b>	<b>19</b>

# 1. Einleitung

## 1.1 Motivation

Das Verhalten rotierender Körper unterscheidet sich signifikant von dem ruhender Objekte, insbesondere in Bezug auf Stabilität der bewegung. Dieses Experiment untersucht das Phänomen der Kreiselbewegung, bei dem ein rotierender Körper – anstelle einfach umzukippen – durch Gravitation und Eigenrotation komplexe Bewegungen wie Präzession und Nutation ausführt. Ziel ist es, die grundlegenden Prinzipien zu verstehen und physikalische Größen wie das Trägheitsmoment zu messen.

## 1.2 Messverfahren

Das Experiment wird mit einem luftgelagerten Kreisel durchgeführt, bestehend aus einer Stahlkugel und einem Aluminiumstab. Die Messungen erfolgen durch:

- **Präzession:** Nach Anbringen von Zusatzgewichten wird die Präzessionsdauer  $T_P$  in Abhängigkeit von Drehgeschwindigkeit und Masseverteilung beobachtet.
- **Nutation:** Mittels einer Farbscheibe und eines Stroboskops wird die Nutationsbewegung des Kreisels gemessen. Hierbei werden die Frequenzen der Eigenrotation  $\omega_F$  und Nutation  $\omega_N$  analysiert.
- **Dämpfung:** Die Abnahme der Rotationsfrequenz wird über einen definierten Zeitraum erfasst.

## 1.3 Grundlagen aus der Physik

Um die Bewegungen des Kreisels zu beschreiben, sind folgende physikalische Konzepte relevant:

### 1.3.1 Drehimpuls und Trägheitsmoment

Der Drehimpuls  $\vec{L}$  eines starren Körpers wird durch das Produkt aus Trägheitsmoment  $I$  und Winkelgeschwindigkeit  $\vec{\omega}$  beschrieben:

$$\vec{L} = I \cdot \vec{\omega} \quad (1.1)$$

Das Trägheitsmoment hängt von der Masseverteilung relativ zur Rotationsachse ab. Für die Figurenachse (Symmetrieachse) eines symmetrischen Kreisels ist der Drehimpuls zeitlich konstant, solange kein äußeres Drehmoment wirkt:

$$\frac{d\vec{L}}{dt} = \vec{M} \quad (1.2)$$

Ein symmetrischer Kreisel hat unterschiedliche Hauptträgheitsmomente  $I_x, I_y, I_z$ . Dabei gilt bei Symmetrie  $I_x = I_y \neq I_z$ .

### 1.3.2 Präzession

Wenn der Kreisel eine Zusatzmasse trägt, wird die Symmetrieachse aus der Vertikalen geneigt, und ein Drehmoment  $\vec{M}$  entsteht:

$$\vec{M} = \vec{l} \times m \cdot \vec{g} \quad (1.3)$$

Hier ist  $\vec{l}$  der Abstand zwischen dem Stützpunkt und dem Schwerpunkt. Dieses Drehmoment führt dazu, dass die Kreiselachse mit der Präzessionsfrequenz  $\omega_P$  um die Vertikale rotiert:

$$\omega_P = \frac{M}{L} = \frac{m \cdot g \cdot l}{I_z \cdot \omega_F} \quad (1.4)$$

Die Präzession ist unabhängig von der Ausrichtung des Kreisels, jedoch proportional zur Gewichtskraft  $m$ , dem Hebelarm  $l$  und invers proportional zur Eigenrotation  $\omega_F$

### 1.3.3 Nutation

Die Nutationsbewegung tritt auf, wenn der Kreisel durch einen Schlag oder eine Störung eine zusätzliche Kegelförmige Bewegung ausführt. Die Winkelgeschwindigkeit  $\vec{\omega}$  zerfällt dabei in zwei Komponenten:

$$\vec{\omega} = \vec{\omega}_N + \vec{\omega}_F \quad (1.5)$$

Dabei beschreibt  $\omega_N$  die Bewegung der Figurenachse auf einem sogenannten Nutationskegel um die Drehimpulsachse, und  $\omega_F$  die Eigenrotation des Kreisels. Die Nutationsfrequenz hängt von den Trägheitsmomenten und der Eigenrotation ab:

$$\omega_N \approx \frac{I_z}{I_x} \cdot \omega_F \quad (1.6)$$

Falls  $\omega_N$  zu beobachten ist, wird dies durch eine spezielle Farbscheibe sichtbar, die an der Figurenachse angebracht ist. Die momentane Drehachse zeigt durch Farbwechsel die Nutationsfrequenz an.

### 1.3.4 Dämpfung

Die Eigenrotation des Kreisels nimmt aufgrund von Luftreibung und Lagerverlusten allmählich ab. Dies wird durch eine exponentielle Abnahme der Winkelgeschwindigkeit beschrieben:

$$\omega(t) = \omega_0 \cdot e^{-\lambda t} \quad (1.7)$$

Hierbei ist  $\lambda$  die Dämpfungskonstante, und  $\omega_0$  die anfängliche Rotationsgeschwindigkeit. Die Halbwertszeit der Rotation kann ebenfalls aus den Messdaten bestimmt werden:

$$t_{1/2} = \frac{\ln(2)}{\lambda} \quad (1.8)$$

## 2. Durchführung

### 2.1 Messprotokoll

Jonathan Rademacher  
Mikaela Sorgenfrei

Messprotokoll  
213

13.01.25

Materialien

→ Stahlkugel ( $m = 4,164 \text{ kg}$ ,  $r = 5,08 \text{ cm}$ )

→ 52 Gewichte

→ Jantischeibe

→ Strabenschop

→ Stoppschuh

→ Motor mit Netzgerät

→

Aufgabe 2)

$t$

$\omega$

$\Delta t = 1 \text{ sec}$

$\Delta \omega = 5$

0

530

2

460

4

415

6

370

8

340

10

310

12

280

Antiquar 3 a) 1 Gewicht 20cm  
 $\varphi$  t  
 ca 30° 2,40 sec  
 ca 75° 2m 6,5 sec  
 ca 10° 2m 1 sec  
 $\Delta \varphi \approx 6^\circ$   $\Delta t = 0,5 \text{ sec}$

b) 1g bei 15cm 1g bei 20cm 2g bei 15cm 2g bei 20  
 $w_F$   $T_p$   $w_F$   $T_p$   $w_F$   $T_p$   $w_F$   $T_p$   
 $t_1$  420 2m 24s 610 2m 44sec 460 2min 7s 600 1min 38  
 $t_2$  300 1m 58s 495 2min 16s-7 390 1min 50s 540 1min 20  
 $t_3$  580 3m 21s 390 2min 05s 310 1min 30  
 $t_4$  455 2m 42s 330 2min 0s 560 2min 22s 350 32s  
 $\Delta \varphi = 10$   $\Delta t = 0,25$

4a) Reihenfolge Fandler.

b)  $\Delta \varphi \approx 10$   $\Omega$  für 10 Umläufe.  $\Delta t = 0,2 \text{ sec}$   
 $t_1 \sim 550 \sim 16,01$   
 $t_2 \sim 510 \sim 12,5$   
 $t_3 \sim 460 \sim 19,1$   
 $t_4 \sim 420 \sim 20,3$   
 $t_5 \sim 380 \sim 21,6$   
 $t_6 \sim 350 \sim 23,1$   
 $t_7 \sim 320 \sim 25,3$   
 $t_8 \sim 300 \sim 26,2$   
 $t_9 \sim 450 \sim 18,6$   
 $t_{10} \sim 520 \sim 15,8$

	WF	WN	
9			
1	550	595	
2	480	450	
3	450	440	
4	340	320	
5	590	530	
6	500	450	
7	540	500	
8	480	450	
9	450	420	
10	400	390	
	$\sum f = 10$	$\sum f = 10$	P. Wein

### 3. Auswertung

#### 3.1 Dämpfung des Kreisels

Die Frequenz es Kreisels nimmt i.d.r Exponentiell ab, also liegt eine Einhüllende Dämpfungsfunktion vor, die Exponentiell mit der Dämpfungskonstante  $\lambda$  fällt. Wie wir in GLeichung 1.7 sehen können.

Plotten wir nun die abnahme der Frequenz als Funktion der Zeit, und legen eine Loarymtische Skala fest, so ergibt sich ein lineare verlauf, aus dessen Steigung wir Dämpfungskonstante ablesen können.

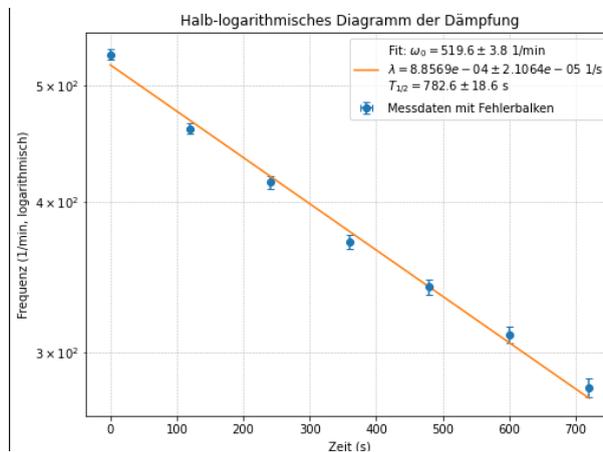


Abbildung 3.1: Dämpfungsdiagramm

Die Steigung, und damit auch die Dämpfungskonstante wurde dabei ermittelt mit:

$$\lambda = (8,86 \pm 0,21) \cdot 10^{-4} 1/min$$

Nach der Formel 1.8 können wir aus  $\lambda$  auch die Halbwärtszeit bestimmen.

$$T_{1/2} = (782 \pm 18) s$$

Dabei wurde die Fehlerformel:

$$\Delta T_{1/2} = \frac{\ln 2}{\lambda^2} \Delta \lambda \tag{3.1}$$

#### 3.2 Präzessionsvergleich

Wir haben die Präzessionszeiten für unterschiedliche Winkel gemessen und dabei die folgende Tabelle erhalten: Die Zeiten scheinen keinen Zusammenhang mit den Winkeln zu zeigen. Es

Winkel	Zeit
Wenig Winkel	$(2,40 \pm 0,5) s$
mittlerer Winkel	$(2,07 \pm 0,5) s$
großer Winkel	$(2,01 \pm 0,5) s$

Tabelle 3.1: Präzessionszeiten

ist keine identifizierbare Steigung etc. zu erkennen. Um dies genauer zu untersuchen wären

allerdings mehr und v.a. genauere Messungen notwendig. Die Signifikanz dieser Aussage ist also leider recht gering.

### 3.3 Bestimmung des Trägheitsmoments

Zunächst werden die angepassten Frequenzen  $\omega_{F_{ende}}$  berechnet. Diese erhalten wir über 1.7 und die jeweiligen Zeiten des Ende der Präzisions. Dabei benutzen wir die Fehlerformel:

$$\Delta\omega_{F_{ende}} = \sqrt{(e^{-t\lambda}\Delta\omega_o)^2 + \left(\frac{-t\omega_o}{e^{t\lambda}}\Delta\lambda\right)^2 + \left(\frac{-\omega_o\lambda}{e^{t\lambda}}\Delta t\right)^2} \quad (3.2)$$

Aus dem Anfangswert  $\omega_o$  und  $\omega_{F_{ende}}$  kann man dann einen Mittelwert bilden, dessen Fehler man ausrechnet mit:

$$\Delta\bar{\omega} = \sqrt{\left(\frac{1}{2}\Delta\omega_o\right)^2 + \left(\frac{1}{2}\Delta\omega_{F_{ende}}\right)^2} \quad (3.3)$$

Wir erhalten also:

Präzissionsdauer [sec]	Frequenz $\omega_0$ [1/min]	Frequenz $\omega_{F_{ende}}$ [1/min]	Frequenz $\bar{\omega}$ [1/min]
(144 ± 0.2)	(420 ± 5)	(370 ± 5)	(395 ± 3)
(118 ± 0.2)	(300 ± 5)	(270 ± 5)	(285 ± 3)
(201 ± 0.2)	(580 ± 5)	(485 ± 5)	(533 ± 3)
(162 ± 0.2)	(455 ± 5)	(394 ± 5)	(425 ± 3)

**Tabelle 3.2:** 1 Gewicht bei 15cm

Präzissionsdauer [sec]	Frequenz $\omega_0$ [1/min]	Frequenz $\omega_{F_{ende}}$ [1/min]	Frequenz $\bar{\omega}$ [1/min]
(164 ± 0.2)	(610 ± 5)	(528 ± 5)	(569 ± 3)
(136 ± 0.2)	(495 ± 5)	(439 ± 5)	(467 ± 3)
(125 ± 0.2)	(390 ± 5)	(349 ± 5)	(370 ± 3)
(120 ± 0.2)	(330 ± 5)	(297 ± 5)	(313 ± 3)

**Tabelle 3.3:** 1 Gewicht bei 20cm

Präzissionsdauer [sec]	Frequenz $\omega_0$ [1/min]	Frequenz $\omega_{F_{ende}}$ [1/min]	Frequenz $\bar{\omega}$ [1/min]
(127 ± 0.2)	(460 ± 5)	(411 ± 5)	(436 ± 3)
(110 ± 0.2)	(390 ± 5)	(354 ± 5)	(372 ± 3)
(90 ± 0.2)	(310 ± 5)	(286 ± 5)	(298 ± 3)
(142 ± 0.2)	(560 ± 5)	(521 ± 5)	(540 ± 3)

**Tabelle 3.4:** 2 Gewicht bei 15cm

Präzissionsdauer $T_P$ [sec]	Frequenz $\omega_0$ [1/min]	Frequenz $\omega_{F_{ende}}$ [1/min]	Frequenz $\bar{\omega}$ [1/min]
(98 ± 0.2)	(600 ± 5)	(550 ± 5)	(575 ± 3)
(80 ± 0.2)	(540 ± 5)	(503 ± 5)	(522 ± 3)
(62 ± 0.2)	(430 ± 5)	(407 ± 5)	(419 ± 3)
(52 ± 0.2)	(350 ± 5)	(334 ± 5)	(342 ± 3)

**Tabelle 3.5:** 2 Gewichte bei 20cm

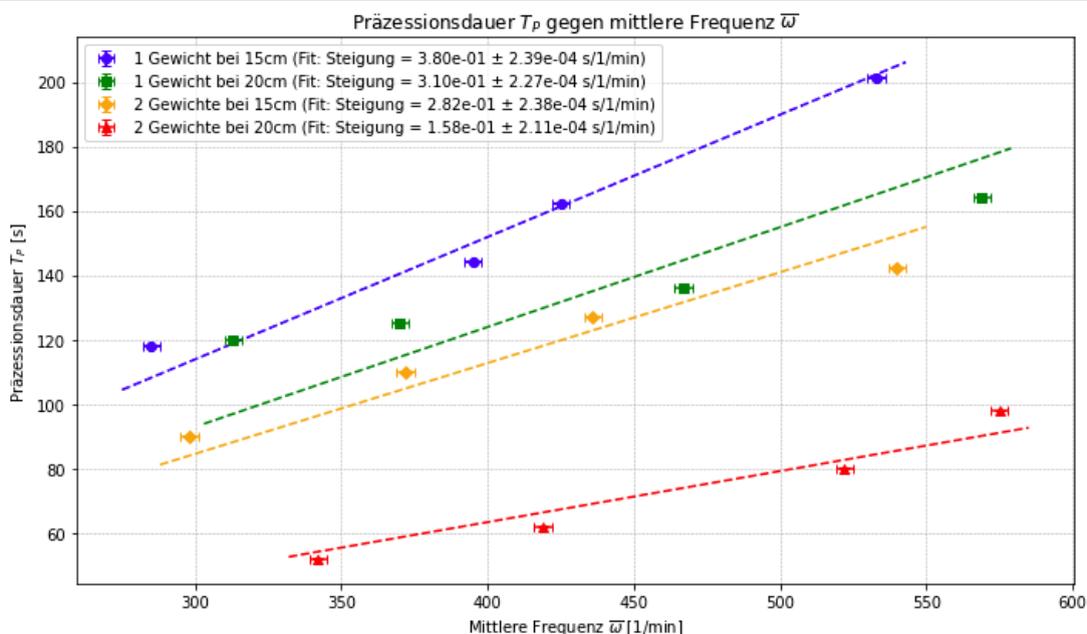


Abbildung 3.2: Präzissionsplots

Nun kann man  $T_P$  als Funktion von  $\bar{\omega}$ , so erhält man geraden, deren Steigung dem Kehrwert der Frequenzproduktes.

Stellen wir Gleichung 1.4 nach Dem Trägheitsmoment um, so kann man damit auch die Trägheitsmoment  $I_z$  bestimmen.

Mit der Bedingung, dass die Geraden druch den Ursprung gehen sollen, sehen die Fits so aus:

Daraus ergibt sich folgende Tabelle:

Was	Steigung $s$	Trägheitsmoment $I_z$ [ $kgm^2$ ]
1G15cm	$(0,38000 \pm 0,00024)$	$(16,140 \pm 0,015)$
1G20cm	$(0,31000 \pm 0,00023)$	$(26,370 \pm 0,019)$
2G15cm	$(0,28200 \pm 0,00024)$	$(21,740 \pm 0,018)$
2G20cm	$(0,15800 \pm 0,00021)$	$(51,58 \pm 0,07)$

Tabelle 3.6: Trägheitsmomente

Dabei wurde der Fehler des Trägheitsmoment bestimmt mit:

$$\Delta I_z = \frac{mgl}{s} \cdot \Delta s \tag{3.4}$$

Für die Mittelwertsbestimmung benutzen wir im Folgenden nur die ersten 3 Werte, da der letzte mit Ziemlicher sicherheit auf einen Messfehler zurückzuführen ist und nicht in die Ergebnisreihe passt.

Wir erhalten:

$$\bar{I}_z = (19,009 \pm 0,008)kgm^2$$

### 3.4 Vergleich der Trägheitsmomente

Es gilt, dass  $I_x > I_z$ , da sich die Stange i.d.R. in der z-Achse befunden hat.

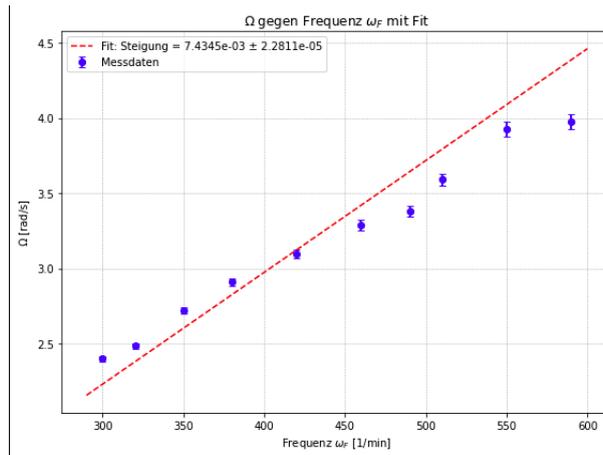


Abbildung 3.3:  $\Omega$  vs  $\omega_F$

### 3.5 Bestimmung von $I_x$

Dadurch betrachten wir zunächst die Dauer  $T$  für 10 Farbwechseldruchläufe. Wir können die Frequenz dieses Farbwechsels angeben mit:

$$\Omega = 2\pi \frac{10}{T} \quad (3.5)$$

Mit der Fehlerformel:

$$\Delta\Omega = \frac{-20\pi}{T^2} \Delta T \quad (3.6)$$

Wir tragen nun die gemessenen  $\Omega$  als eine Funktion gegen  $\omega_F$  auf und erhalten dabei folgedes Bild:

Wir erhalten also eine Steigung von:

$$s_2 = (0,007435 \pm 0,000023) \text{rad min/s}$$

Mithilfe der Formel:

$$I_x = \frac{I_z}{1 - s_2} \quad (3.7)$$

und dem Fehler:

$$\Delta I_x = \sqrt{\left(\frac{1}{1 - s_2} \Delta I_z\right)^2 + \left(\frac{I_z}{(1 - s_2)^2} \Delta s_2\right)^2} \quad (3.8)$$

Erhalten wir die Tragheitsmoment in x Richtung mit:

$$I_x = (19,151 \pm 0,008) \text{kgm}^2$$

Also nur geringfügig größer.

### 3.6 Bestimmung von $I_x$ die Zweite

Wir tragen nun  $\omega_N$  als Funktion von  $\omega_F$  ab und betrachte wieder die Steigung  $s_3$

Wir erhalten also eine Steigung von:

$$s_3 = (1,077 \pm 0,007) \text{rad min/s}$$

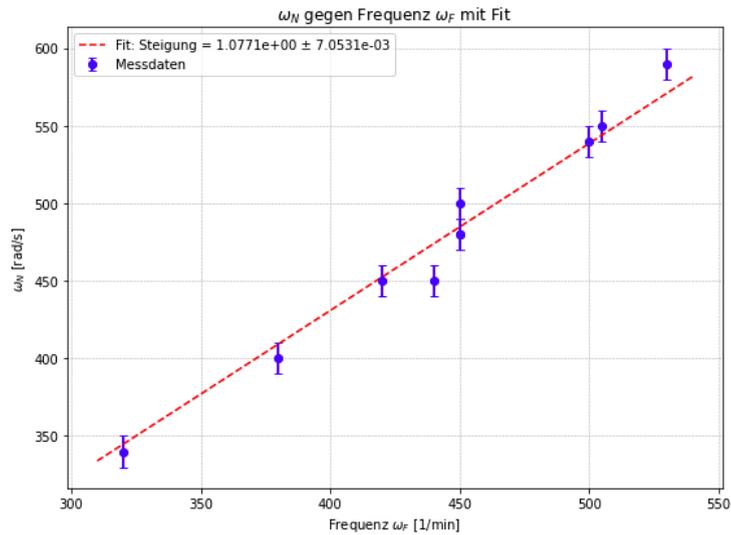


Abbildung 3.4:  $\omega_N$  vs  $\omega_F$

Mit der Formel:

$$I_x = \frac{I_z}{s_3} \quad (3.9)$$

Und der Fehlerformel:

$$\Delta I_x = \sqrt{\left(\frac{1}{s_3} \Delta I_z\right)^2 + \left(\frac{-I_z}{s_3^2} \Delta s_3\right)^2} \quad (3.10)$$

erhalten wir:

$$I_x = (17,65 \pm 0,12) \text{kgm}^2$$

Dieser Wert weicht um  $12,4 \sigma$  von dem andern Wert ab. Dies spricht für eine Signifikante Abweichung, welche vorraussichtlich auf die ungenaue Messung zurückzuführen ist.

## 4. Zusammenfassung und Diskussion

alle messungen butcherd durch dummes 2x geschwindigkeit

In diesem Experiment haben wir das Verhalten des rotierenden Kreisls untersucht, insbesondere die Phänomene der Präzession, Nutation und Dämpfung, um grundlegende physikalische Prinzipien wie das Trägheitsmoment des Kreisels zu bestimmen. Dabei haben wir verschiedene Messmethoden genutzt.

Bei der Bestimmung des Trägheitsmoments haben wir die Frequenzen der Eigenrotation und die Präzessionszeiten gemessen, um aus diesen Werten das Trägheitsmoment des Kreisels zu berechnen. Unsere Ergebnisse zeigten eine exponentielle Abnahme der Frequenz der Eigenrotation, was die Dämpfung des Kreisels belegt. Dabei ergab sich ein Dämpfungsfaktor von:

$$\lambda = (8,86 \pm 0,21) \cdot 10^{-4} 1/min$$

Die gemessenen Präzessionszeiten variierten mit der Masseverteilung, jedoch konnten keine eindeutigen Zusammenhänge zwischen den Messwerten und den Winkeln erkannt werden.

Die Werte für das Trägheitsmoment, das wir aus den Präzessionsdaten abgeleitet haben, lagen in einem Bereich von

$$16 - 51 kgm^2$$

, wobei der Mittelwert für das Trägheitsmoment auf

$$I_z = (19,009 \pm 0,008) kgm^2$$

berechnet wurde. Trotz dieses Resultats gibt es einige Unsicherheiten und potenzielle Fehlerquellen, die die Genauigkeit unserer Messungen beeinträchtigt haben. Besonders bei der Frequenzmessung mit dem Stroboskop traten Probleme auf, da die Messung der Frequenz zu Unsicherheiten führte. Es war nicht eindeutig, ob beispielsweise 300 oder 600 1/min gemessen wurden, was zu fehlerhaften Werten führte.

Diese Messungen wurden durch die Geschwindigkeit der Rotation und die Handhabung des Stroboskops erschwert, was die Präzision der gemessenen Frequenzen ebenfalls beeinträchtigte.

Auch die Bestimmung der Dämpfungskonstanten könnte durch Ungenauigkeiten bei der Zeitmessung und der Ablesung der Frequenzwerte beeinflusst worden sein. Dieser Fehler würde sich dann in allen Werten widerspiegeln.

Trotz dieser Unsicherheiten können wir jedoch sagen, dass das Experiment (in ein paar Aspekten) erfolgreich war. Besonders die Bestimmung des Trägheitsmoments und die Untersuchung der Dämpfung wurden mit zufriedenstellender Genauigkeit durchgeführt. Dennoch sollten für zukünftige Experimente eine verbesserte Technik und genauere Instrumente verwendet werden, um die Messungen noch präziser und verlässlicher zu gestalten.

## 5. Anhang

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

# Beispiel-Daten (Zeit in Sekunden, Frequenz in 1/min)
zeit = np.array([0, 120, 240, 360, 480, 600, 720]) # Zeit in Sekunden
frequenz = np.array([530, 460, 415, 370, 340, 310, 280]) # Frequenz in Hz

# Fehler in x (Zeit) und y (Frequenz)
fehler_x = np.array([1, 1, 1, 1, 1, 1, 1]) # Fehler in Sekunden
fehler_y = np.array([5, 5, 5, 5, 5, 5, 5]) # Fehler in 1/min

# Logarithmische Anpassungsfunktion
def log_decay(t, omega0, lambda_):
    return omega0 * np.exp(-lambda_ * t)

# Curve Fitting (Anpassung der logarithmischen Funktion an die Daten)
popt, pcov = curve_fit(log_decay, zeit, frequenz, sigma=fehler_y, absolute_sigma=True)
omega0, lambda_ = pop

fit_errors = np.sqrt(np.diag(pcov)) # Unsicherheiten der Fit-Parameter
omega0_error, lambda_error = fit_errors

# Berechnung der Halbwertszeit
halbwertszeit = np.log(2) / lambda_
halbwertszeit_error = lambda_error * halbwertszeit / lambda_ # Fehlerfortpflanzung

# Plot: Halb-logarithmisches Diagramm mit Fehlerbalken
plt.figure(figsize=(8, 6))
plt.errorbar(zeit, frequenz, xerr=fehler_x, yerr=fehler_y, fmt='o', label='Messdaten')
# Fehlerbalken
plt.semilogy(zeit, log_decay(zeit, *popt), '-', label=(
    f'Fit:  $\omega_0 = \{omega0:.1f\} \pm \{omega0\_error:.1f\} \text{ 1/min}$ \n'
    f' $\lambda = \{lambda_:.4e\} \pm \{lambda\_error:.4e\} \text{ 1/s}$ \n'
    f' $T_{\{1/2\}} = \{halbwertszeit:.1f\} \pm \{halbwertszeit\_error:.1f\} \text{ s}$ '
)) # Fit mit Fehlerangaben

# Achsen und Titel
plt.xlabel('Zeit (s)')
plt.ylabel('Frequenz (1/min, logarithmisch)')
plt.title('Halb-logarithmisches Diagramm der Dämpfung')

```

```

plt.legend()
plt.grid(True, which="both", linestyle="--", linewidth=0.5)
plt.show()

# Ergebnisse ausgeben
print(f"D mpfungskonstante      : {lambda_:.4e}      {lambda_error:.4e} 1/s")
print(f"Anfangsfrequenz        : {omega0:.1f}      {omega0_error:.1f} 1/min")
print(f"Halbwertszeit T /      : {halbwertszeit:.1f}      {halbwertszeit_error:.1f}")

-----

w_o = [420,300,580,455]
Deltaw_o = [5,5,5,5]
time= [144,118,201,162]
Deltatime = [0.2,0.2,0.2,0.2]
d = 8.8569e-04
Deltad= 2.1064e-05
w_e1 = []
errw_e1 = []
meanw = []
errmean =[]

def w_e(t,w_o,d):
    return w_o*np.exp(-d*t)

def errw_e(t,w_o,d,Deltaw_o,Deltatime,Deltad):
    return np.sqrt( (np.exp(-d*t)*Deltaw_o      )**2      +(-t*w_o*np.exp(-t*d) * Del
)**2      +( -w_o*d*np.exp(-t*d)*Deltatime  )**2  )

def mean(w_e,w_o):
    return 0.5*(w_o+w_e)
def meanerr(Deltaw_o,errw_e):
    return 0.5*np.sqrt(Deltaw_o**2+errw_e**2)

for i in range(len(w_o)):
    r = w_e(time[i],w_o[i],d)
    r_2 = errw_e(time[i],w_o[i],d,Deltaw_o[i],Deltatime[i],Deltad)
    w_e1.append(r)
    errw_e1.append(r_2)

for i in range(len(w_o)):

```

```
r_3 = mean(w_e1[i],w_o[i])
r_4 = meanerr(Deltaw_o[i],errw_e1[i])
meanw.append(r_3)
errmean.append(r_4)
```

```
print(w_e1)
print(errw_e1)
print(meanw)
print(errmean)
```

---

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
```

```
# Daten aus den Tabellen (Pr zessionsdauer T_P in Sekunden, mittlere Frequenz
data = {
```

```
    "1 Gewicht bei 15cm": {
        "T_P": np.array([144, 118, 201, 162]),
        "omega_avg": np.array([395, 285, 533, 425]),
        "T_P_err": np.array([0.2, 0.2, 0.2, 0.2]),
        "omega_avg_err": np.array([3, 3, 3, 3]),
    },
```

```
    "1 Gewicht bei 20cm": {
        "T_P": np.array([164, 136, 125, 120]),
        "omega_avg": np.array([569, 467, 370, 313]),
        "T_P_err": np.array([0.2, 0.2, 0.2, 0.2]),
        "omega_avg_err": np.array([3, 3, 3, 3]),
    },
```

```
    "2 Gewichte bei 15cm": {
        "T_P": np.array([127, 110, 90, 142]),
        "omega_avg": np.array([436, 372, 298, 540]),
        "T_P_err": np.array([0.2, 0.2, 0.2, 0.2]),
        "omega_avg_err": np.array([3, 3, 3, 3]),
    },
```

```
    "2 Gewichte bei 20cm": {
        "T_P": np.array([98, 80, 62, 52]),
        "omega_avg": np.array([575, 522, 419, 342]),
        "T_P_err": np.array([0.2, 0.2, 0.2, 0.2]),
        "omega_avg_err": np.array([3, 3, 3, 3]),
    },
},
```

```

}

# Lineare Funktion f r den Fit
def linear(x, slope):
    return slope * x

# Farben und Marker f r die unterschiedlichen Datens tze
colors = ['blue', 'green', 'orange', 'red']
markers = ['o', 's', 'D', '^']

# Plot erstellen
plt.figure(figsize=(10, 6))
fit_results = {} # F r Ergebnisse des Fits

for i, (label, dataset) in enumerate(data.items()):
    T_P = dataset["T_P"]
    omega_avg = dataset["omega_avg"]
    T_P_err = dataset["T_P_err"]
    omega_avg_err = dataset["omega_avg_err"]

    # Fit durchf hren (Fixpunkt Ursprung: offset=0)
    popt, pcov = curve_fit(linear, omega_avg, T_P, sigma=T_P_err, absolute_sigma=True)
    slope = popt[0]
    slope_err = np.sqrt(np.diag(pcov))[0]
    fit_results[label] = {"slope": slope, "slope_err": slope_err}

# Plot der Daten mit Fehlerbalken
plt.errorbar(
    omega_avg,
    T_P,
    xerr=omega_avg_err,
    yerr=T_P_err,
    fmt=markers[i],
    color=colors[i],
    label=f"{label} (Fit: Steigung = {slope:.2e}      {slope_err:.2e} s/1/min)",
    capsizes=3,
)

# Fit-Linie zeichnen
omega_fit = np.linspace(min(omega_avg) - 10, max(omega_avg) + 10, 100)
plt.plot(omega_fit, linear(omega_fit, slope), color=colors[i], linestyle='--')

```

```

# Achsenbeschriftungen , Titel und Legende
plt.xlabel(r'Mittlere Frequenz  $\overline{\omega}$  [1/min]')
plt.ylabel(r'Pr zessionsdauer  $T_P$  [s]')
plt.title('Pr zessionsdauer  $T_P$  gegen mittlere Frequenz  $\overline{\omega}$ ')
plt.legend()
plt.grid(True, linestyle="--", linewidth=0.5)

# Plot anzeigen
plt.tight_layout()
plt.show()

# Fit-Ergebnisse ausgeben
print("Fit-Ergebnisse (Steigung Fehler):")
for label, result in fit_results.items():
    print(f"{label}: Steigung = {result['slope']:.2e} {result['slope_err']:.2e}")

```

---

```

# Konstanten
g = 9.81 # Erdbeschleunigung in m/s2
m_kugel = 4.164 # Masse der Kugel inkl. Stab in kg
abstaende = {"15cm": 0.15, "20cm": 0.20} # Abstände der Zusatzmassen in Metern

# Berechnung des Tr gheitsmoments aus den Steigungen
for label, result in fit_results.items():
    # Abstand abhängig von der Messkonfiguration
    if "15cm" in label:
        l = abstaende["15cm"]
    elif "20cm" in label:
        l = abstaende["20cm"]

    # Steigung und Fehler
    slope = result["slope"]
    slope_err = result["slope_err"]

    # Berechnung von I_z
    I_z = m_kugel * g * l / slope
    I_z_err = m_kugel * g * l / slope**2 * slope_err # Fehlerfortpflanzung

    # Ergebnisse ausgeben
    print(f"{label}:")
    print(f"Tr gheitsmoment I_z = {I_z:.4e} {I_z_err:.4e} kg m ")

```

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

# Gegebene Daten
fre = np.array([590, 550, 510, 490, 460, 420, 380, 350, 320, 300]) # Frequenz in
t_2 = np.array([15.8, 16.01, 17.5, 18.6, 19.1, 20.3, 21.6, 23.1, 25.3, 26.2])
# Zeit in Sekunden
dt = 0.2 # Fehler in t_2

# Funktionen zur Berechnung von omega und Fehler
def omega(t):
    return 2 * np.pi * (10 / t)

def omegaerr(t, dt):
    return (-20 * np.pi) / (t**2) * dt

# Berechnung von omega und Fehler
o = omega(t_2)
oerr = omegaerr(t_2, dt)

# Lineare Funktion (durch den Ursprung)
def linear(x, slope):
    return slope * x

# Curve Fit
popt, pcov = curve_fit(linear, fre, o, sigma=oerr, absolute_sigma=True)
slope = pop[0]
slope_err = np.sqrt(np.diag(pcov))[0]

# Plot
plt.figure(figsize=(8, 6))
plt.errorbar(fre, o, yerr=oerr, fmt='o', label='Messdaten', capsize=3, color='blue')
fre_fit = np.linspace(min(fre) - 10, max(fre) + 10, 100)
plt.plot(fre_fit, linear(fre_fit, slope), 'r--', label=f'Fit: Steigung = {slope}')

# Achsenbeschriftungen und Titel
plt.xlabel(r'Frequenz  $\omega_F$  [1/min]')
plt.ylabel(r' $\Omega$  [rad/s]')

```

```
plt.title(r'\Omega$ gegen Frequenz $\omega_F$ mit Fit')
plt.grid(True, linestyle="--", linewidth=0.5)
plt.legend()
plt.tight_layout()

# Plot anzeigen
plt.show()

# Ergebnisse ausgeben
print(f"Fit-Ergebnisse:")
print(f"Steigung = {slope:.4e}    {slope_err:.4e} r a d    s    / (1/min)")
```

## Quellen- und Literaturverzeichnis

- [1] CAPTAIN JONI: *pap1-tex-vorlage*. <https://github.com/captain-joni/pap1-tex-vorlage>. – [Online; Stand 28.08.2024]
- [2] DR. J. WAGNER: *Physikalisches Praktikum 1 für Studierende der Physik B.Sc.* <https://www.physi.uni-heidelberg.de/Einrichtungen/AP/info/Corona/PAP1.pdf>. – [Online; Stand 01/2014]