

Physikalisches Anfängerpraktikum II

Sommersemester 2023

Versuch 213

Tutor: Felix Waldherr

Kreisel

1 Einleiung¹

1.1 Ziel des Versuchs

In diesem Experiment sollen das Verhalten und die Rotationseigenschaften eines Kreisels analysiert werden. Es zielt darauf ab, die Reibungsverluste, die Dämpfungskonstante sowie die Halbwertszeit zu bestimmen. Weiterhin soll aus der Präzessionsfrequenz eines schweren Kreisels das Trägheitsmoment um die Figurennachse ermittelt werden. Zudem soll durch die Größe und Richtung der Umlaufgeschwindigkeit der aktuellen Drehachse in Bezug auf die Figurennachse das Trägheitsmoment senkrecht zur Figurennachse bestimmt werden sowie das Trägheitsmoment aus der Nutationsfrequenz ermittelt.

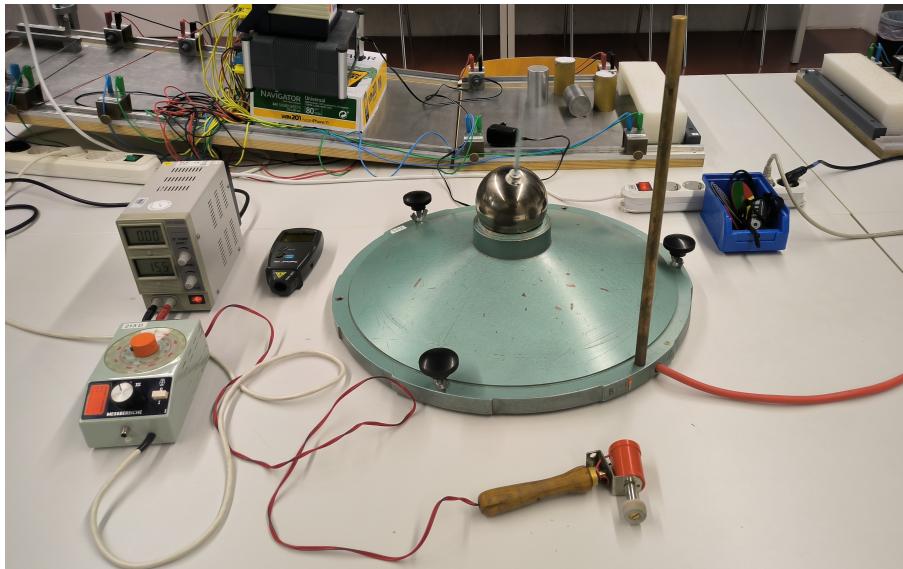


Abbildung 1: Versuchsaufbau

1.2 Der kräftefreie, symmetrische Kreisel

Wenn der Schwerpunkt eines Kreisels auf dem Unterstützungspunkt liegt, übt die Gewichtskraft kein äußeres Drehmoment \vec{M} aus und der Drehimpuls \vec{L} bleibt erhalten. In diesem Fall bezeichnet man ihn als "kräftefrei". Wenn zusätzlich zwei der

¹Dr. J.Wagner - Physikalisches Anfängerpraktikum - V. 2.0β Stand 04/2023 - Python Edition

Hauptträgheitsmomente des Kreisels gleich groß sind, ist der Kreisel symmetrisch. Zur Beschreibung der Kreiselbewegung definieren wir neben dem Drehimpuls noch die charakteristischen Symmetrieeachsen \vec{F} sowie die Richtung der Drehachse $\vec{\omega}$.

1.3 Nutation

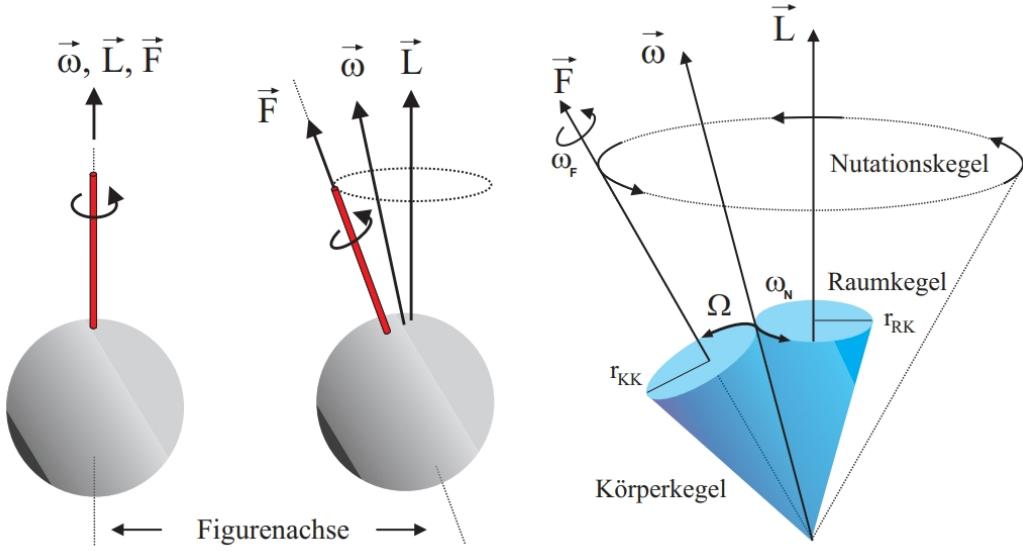


Abbildung 2: Bewegungen des kräftefreien symmetrischen Kreisels

Wenn die charakteristischen Kreiselachsen nicht parallel ausgerichtet sind, sondern unterschiedliche Orientierungen einnehmen, wird dann eine **Nutation** erzeugt. Bei dieser Kreiselbewegung bleibt \vec{L} konstant, während die Figurenachse \vec{L} auf einem Nutationskegel um die raumfeste Drehimpulsachse mit der Frequenz ω_N rotiert und gleichzeitig eine Eigenrotation $\vec{\omega}_F$ ausführt. Die resultierende Drehachse wird als **momentane Drehrichtung** $\vec{\omega}$ bezeichnet und ergibt sich wie folgt:

$$\vec{\omega} = \vec{\omega}_N + \vec{\omega}_F \quad (1)$$

Zerlegen wir die Komponente in x, z-Richtung, so lässt sich mit $\omega_x = \omega_N \sin \theta$ und $L_x = I_x \omega_x = L \sin \theta$ die Nutationsfrequenz berechnen, dabei ist θ der Winkel zwischen den beiden Achsen.

$$\omega_N = \frac{L}{I_x} \quad (2)$$

Für kleine Auslenkungen kann die durch Kleinwinkelnährung $L \approx L_z \omega \approx L_z \omega_F$ weiter geleitet werden:

$$\omega_N \approx \frac{I_z}{I_x} \omega_F \quad (3)$$

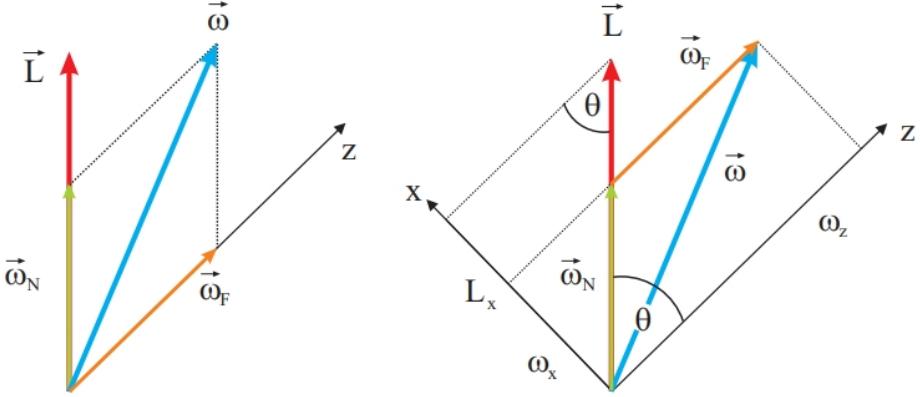


Abbildung 3: Das Koordinatensystem, körperfest und parallel zur Figurennachse orientiert, fällt mit den Hauptachsen des Kreisels zusammen.

Um die Bewegung der Drehachse beobachten zu können, wird eine Farbscheibe auf die Figurennachse montiert. An einem Punkt verschwimmt die Farbe dieser bei Bewegung nicht und kennzeichnet somit die Orientierung der momentanen Drehrichtung. Die Farbenwechsel erfolgt mit Winkelgeschwindigkeit Ω , deren Betrag sich algebraisch über das Verhältnis zwischen den Radien des Körperkegels r_{KK} und des Raumkegels r_{RK} berechnen lässt:

$$\frac{\Omega}{\omega_N} = \frac{r_{RK}}{r_{KK}} \Rightarrow \Omega = \frac{I_x - I_z}{I_x} \omega_F \Rightarrow I_x - I_z = \frac{I_z}{\omega_F/\Omega - 1} \quad (4)$$

1.4 Präzession

Bei den schweren symmetrischen Kreiseln liegt der Schwerpunkt nicht mehr im Unterstützungspunkt, jedoch noch auf der Figurennachse. Im Versuch wird dieser durch Anbringung einer zusätzlichen Masse nach oben verschoben, somit wirkt ohne Berücksichtigung auf Nutation aufgrund der Schwerkraft folgender Drehmoment:

$$\vec{M} = \vec{l} \times m\vec{g} \quad (5)$$

Hierbei ist m die Zusatzmasse und \vec{l} der Vektor zwischen Unterstützungspunkt und Zusatzgewicht. Daraus folgt eine Änderung des Drehimpulses in Richtung des Drehmoments, dessen Betrag jedoch zeitlich konstant bleibt, was zu der seitlichen Ausweichung der Gewichtskraft führt. Man nennt dieses Phänomen **Präzession**:

$$\vec{M} \perp \vec{L} \Rightarrow d\vec{L} \parallel \vec{M} \Rightarrow d\vec{L} \perp \vec{L} \quad (6)$$

Mit $L = I_z \omega_F$ lässt sich die Präzessionsfrequenz darstellen als:

$$\omega_P = \frac{mgl}{I_z \omega_F} \quad (7)$$

Oder in allgemeiner Form:

$$\vec{M} = \vec{\omega}_P \times \vec{L} \quad (8)$$

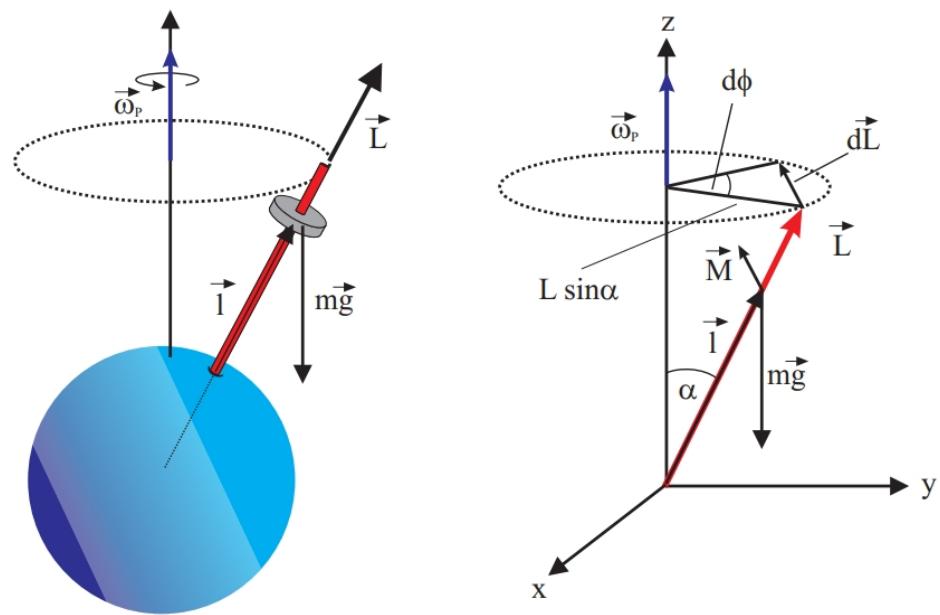


Abbildung 4: Geometrische Darstellung der Präzession

2 Versuchsdurchführung

Versuchsaufbau, Versuchsdurchführung und Messprotokoll siehe folgende Seiten.

Messgeräte

- Stahlkugel mit Al-Stab ($m=4.164 \text{ kg}$, $r=5.08 \text{ cm}$) als Kreisel gelagert in einer Luftkissenpfanne
- 2 Gewichte $r_a = 0.725 \text{ cm}$, $r_i = 0.325 \text{ cm}$, $h=1.1 \text{ cm}$, $m=9.85 \text{ g}$
- Farbscheibe, Scheibe mit konzentrischen Ringen
- Stroboskop (Umdrehungen · Motor mit Netzgerät)
- Stoppuhr (1/min) · Gyroskop zur Demonstration der Kreiseigenschaften.

I. Vorversuch

Wir machen uns zunächst vertraut mit dem Kreisel. Kreisel kräftefrei machen, indem wir die farbigen Scheiben verwenden.

II. Dämpfung des Kreisels

Der Kreisel kräftefrei einstellen, montieren 2 Gewichte am Stabende. Beschleunigen den Kreisel bei senkrechter Achse auf $600 - 700 \text{ min}^{-1}$. Messen celle 2 Minuten die Drehfrequenz des Kreisels über 12 min.

Tabelle 1: Der sinkende Verlauf der Frequenz über die Zeit

Zeit [min]	Drehfrequenz [rpm]	$\text{rpm} = \text{Umdrehung}/\text{min}$
0	650	700
2	500	630
4	370	580
6	280	520
8	180	470
10		420
12		390

III. Präzession

Farbscheibe montieren, kräftefrei. Im Abstand von 20 cm zur Kugelmitte wird ein Gewicht befestigt. Der Kreisel wird auf etwa 500 rpm beschleunigt. Der Stab wird bei gleichen Kreiselfrequenz möglichst ohne Vibration unter 3 Winkeln

des Stabs gegen die Vertikale losgelassen, und die Umlaufzeit (Präzessionszeit) bestimmen.

- Der Kreisel wird nun mit verschiedenen Massen konfiguriert belastet:
- ① Ein Gewichtsstück (GS) bei 15cm
 - ② Ein GS bei 20cm
 - ③ 2 GS'e bei 15cm
 - ④ 2 GS'e bei 20cm

Es wird für jede Gewichtseinstellung bei 4 verschiedenen Frequenzen von 300 rpm bis 600 rpm mit dem Stoppuhr die Präzessionsdauer bestimmt

Frequenz des Kreisels: $f = (500 \pm 10) \text{ min}^{-1}$

Tabelle 2: Umlaufdauern bei verschiedenen Winkeln der Achse

Winkel [°]	Präzessionsdauer T_p [s]
ca. 30°	124 131 66
ca. 60°	128 64
ca. 90°	124 62

Tabelle 3: T_p und ω_F (1 GS bei 15cm)

ω_F [min ⁻¹ rpm]	T_p [s]
300	52
400	46
500	42
600	39

Tabelle 4: T_p und ω_F (1 GS bei 20cm)

ω_F [min ⁻¹ rpm]	T_p [s]
300	38
400	32 32 36
500	35 35 37 37
600	25 25 146

Tabelle 5: T_p und w_F (2 GS'e bei 15cm)

w_F [min ⁻¹ rpm]	T_p [s]
300	7.36
400	4.4
500	4.4
600	4.4

Tabelle 6: T_p und w_F (2 GS'e bei 200 cm)

w_F [min ⁻¹ rpm]	T_p [s]
300	79.21
400	79.27
500	85.36
600	79.42

IV. Umlauf der momentanen Drehachse um die Figurenachse.

Es wird wieder auf kräftefrei justiert. Beschleunigung in senkrechter Richtung

Es wird durch Anlegen der Farbscheibe, unter Beachtung der Farbreihenfolge, die Umlaufrichtung bestimmt.

Es wird für 10 Frequenzen zwischen 300 rpm und 600 rpm die Umlaufzeit der momentanen Drehachse um die Figurenachse gemessen, dazu wird die Zeit für 10 Umläufe des Farbwheels der Sektorschreibe notiert. (gelb \rightarrow rot \rightarrow grün \rightarrow ...)

Tabelle 7: 10 Umlaufzeiten bei verschiedenen Frequenzen

w_F [rpm]	10 Farbumschw. perioden [s]
300	30.2
340	27.2
360	25.2
380	24.0
400	22.8
440	20.8
480	18.3
520	17.1
560	15.5
600	14.4

V. Nutation

Vertauschen den kräfte-freien Kreisel um die Achse in Nutation, dass die Öffnung des Nutationskegels um der Spitze des Stabes nur 1-2 cm betragen. 10 Wertepaare von ω_N, ω_F

Tabelle 8: Die Wertepaare der Frequenzen bei leichter Nutation

ω_F [rpm]	ω_N [rpm]
620	585
600	560
530	495
480	440
420	385
400	365
380	350
370	330
340	300
290	260

Fehler des Stroboskops: ± 5 rpm

Af.

3 Auswertung

Es wurde zuerst eine qualitative Untersuchung durchgeführt: Zu der Justierung des Kreisels dient unter anderem die Farbscheibe, die auch für stroboskopische Messungen benutzt wurde. Durch seitliches Drücken des Metaltrings erfolgte eine Ausrichtungsänderung, was die Nutation aber nicht die Präzession verursachte, und im Zentrum der Farbscheibe wurde eine Verschiebung gezeigt. Das Hinzufügen eines Gewichts am Stab führte zu Präzessionsbewegungen. Ohne zusätzliche Farbscheibe richtete sich der Stab auf, während mit einem Zusatzgewicht der Kreisel umfiel, abhängig von der Ausrichtung und Drehrichtung des Kreisels.

Die charakteristischen Kreiselfrequenzen werden mittels eines Stroboskops identifiziert. Die Farbscheibe auf dem Kreisel erscheint stillstehend, wenn ihre Rotationsfrequenz der Blitzfrequenz des Stroboskops entspricht oder ein **Vielfaches** davon ist. Bei Abweichungen werden mehrere gleichfarbige Sektoren sichtbar.

3.1 Dämpfung des Kreisels

Trotz der Luftschiene besteht im Wesentlichen eine starke Dämpfung, die wir bestimmen müssen. Aus den experimentell gemessenen Frequenz in Abhängigkeit von Zeit lässt sich die Dämpfungskonstante mittels:

$$\omega(t) = \omega_0 e^{-\delta t} \quad (9)$$

aus der Steigung im halblogarithmischen Diagramm ermitteln als:²

$$\underline{\underline{\delta = (8.27 \pm 0.14) \cdot 10^{-4} \text{ s}^{-1}}} \quad (10)$$

Nun kann die Halbwertszeit wie folgt berechnet werden:

$$T_{1/2} = \frac{\ln(2)}{\delta} \quad \Delta T_{1/2} = T_{1/2} \frac{\Delta \delta}{\delta} \quad (11)$$

$$\underline{\underline{T_{1/2} = (838 \pm 20) \text{ s}}} \quad (12)$$

3.2 Präzession

Der Kreisel wird nun senkrecht ausgerichtet und auf die gewünschte Geschwindigkeit gebracht. Dann wird die Achse schräg gestellt und kurz vor der Ablesemarke möglichst ohne Nutationsbewegungen losgelassen. Wir fügen zuerst im Abstand von $l = 20 \text{ cm}$ zur Kugelmitte ein Zusatzgewicht ($m = 9.85 \text{ g}$) auf den Stab hinzu, und bestimmen die Präzessionsdauer unter der Drehgeschwindigkeit von ungefähr 500 mim^{-1}

²Python Code 1

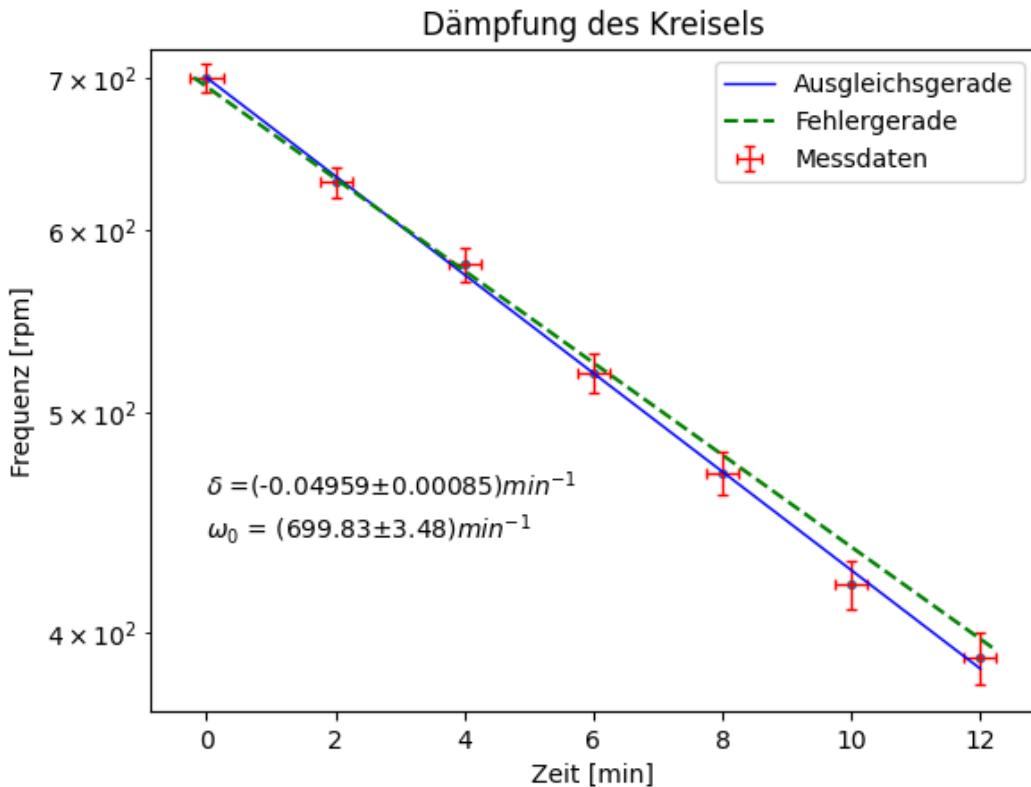


Abbildung 5: Dämpfungsmessung aufgetragen ins halblogarithmische Diagramm

Winkel [°]	T_p [s]
30	66
60	64
90	62

Tabelle 1: Präzessionsdauer und Winkel

Es ist offensichtlich, dass die drei Perioden sich nicht voneinander unterscheiden. Dieses Ergebnis ist jedoch nicht überraschend, da aus Gleichung (7) hervorgeht, dass die Präzessionsfrequenz unabhängig vom Winkel ist. Durch Berechnung des Mittelwert sowie der Standardabweichung, und Quadrieren mit systematischen Fehlern bekommen wir: ($g = 9.81 \text{ m/s}^2$)

$$T_P = (64 \pm 3) \text{ s} \quad (13)$$

Nun werden die Präzessionsdauern bei verschiedenen Frequenzen und Gewichtsverteilungen aufgetragen und daraus die Trägheitskomponente I_z aus Gl.(7) berechnet.³ Während der Präzessionsbewegung ändert sich aufgrund der Dämpfung des Kreisels die Drehfrequenz ω_F :

³Python Codes 2

$$\omega_F = \omega_0 e^{\delta T_P} \quad (14)$$

$$\Delta\omega_F = \omega_F \sqrt{\left(\frac{\Delta\omega_0}{\omega_0}\right)^2 + (\delta\Delta T_P)^2 + (T_P\Delta\delta)^2} \quad (15)$$

wobei ω_0 die Anfangsdrehfrequenzen sind. Aus den Werten für ω_F und ω_0 wird nun der Mittelwert $\bar{\omega}_F$ gebildet. T_p können wir auch analog wie die erste Aufgabe berechnen.

$$\bar{\omega}_F = \frac{\omega_F + \omega_0}{2} \quad (16)$$

$$\Delta\bar{\omega}_F = \frac{1}{2} \sqrt{(\Delta\omega_F)^2 + (\Delta\omega_0)^2} \quad (17)$$

Wir tragen jeweils die Präzessionsperiode T_P als Funktion von $\bar{\omega}_F$ auf und bestimmen für jedes äußere Drehmoment die Steigung s_i :

$$T_P = k\omega_F \quad (18)$$

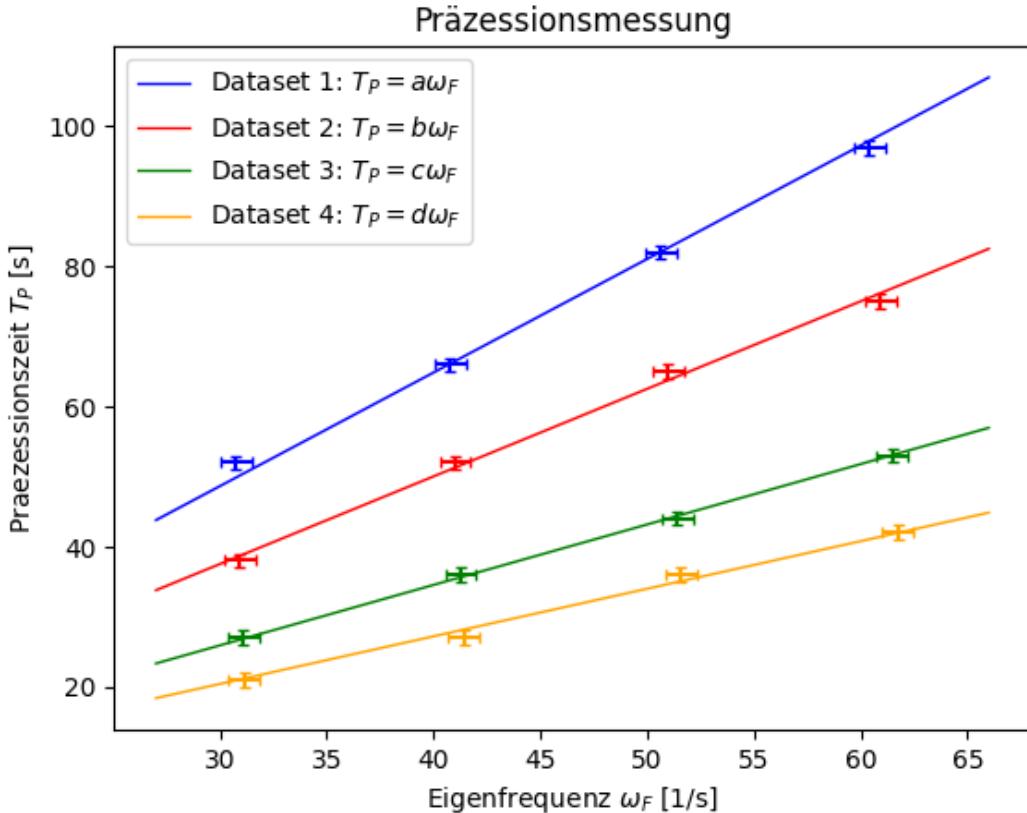


Abbildung 6: Präzessionsdauer gegen Eigenfrequenzen bei verschiedenen Gewichten sowie Drehmomenten, von oben bis unten in Reihe vom Messungen

Die Steigungen ergeben sich als:

$$a = (1.621 \pm 0.014) \text{ } s^2 \quad b = (1.250 \pm 0.012) \text{ } s^2 \quad (19)$$

$$c = (0.863 \pm 0.003) \text{ } s^2 \quad d = (0.679 \pm 0.009) \text{ } s^2 \quad (20)$$

Mit Gleichung (7) lässt sich nun das Trägheitsmoment bei unterschiedlichen externen Bedingungen bestimmen:⁴

$$I_z = \frac{mgl}{\omega_P \bar{\omega}_F} = \frac{mgl}{2\pi} s \quad \Delta I_z = \frac{mgl}{2\pi} \Delta s \quad (21)$$

Mit unseren Messreihen können wir insgesamt 4 verschiedene Trägheitsmomente bekommen, deren Mittelwert unser gesuchtes Trägheitsmoment darstellt:

$$\underline{\underline{I_z = (3.94 \pm 0.17) \text{ } g \cdot m^2}} \quad (22)$$

3.3 Umlauf der momentanen Drehachse um die Figurennachse

Qualitativ sind die in der Einleitung beschriebenen visuellen Effekt an der Kreisscheibe beobachtbar. Durch die Verwendung der Farbscheibe können wir die Drehrichtung des Kreisels bestimmen. Die Abfolge der Farbänderungen geht von Rot über Grün nach Gelb, was darauf hinweist, dass der Kreisel sich im Uhrzeigersinn dreht.

In diesem Zusammenhang zeigt der Kreisel eine ähnliche Nutationsbewegung wie in Abbildung 2c), was dazu führt, dass $I_x > I_z$ gilt. Diese Schlussfolgerung lässt sich auch aus Gleichung (4) ableiten, da die Winkelgeschwindigkeit Ω , mit der die Farbwechsel stattfinden, deutlich kleiner ist als die Eigenfrequenz ω_F . Daher ergibt sich $I_x - I_z > 0$.

Mit einer Stoppuhr wurden Messungen für 10 Frequenzen im Bereich von 300 min^{-1} bis 600 min^{-1} vorgenommen und jeweils die Zeit t für 10 Umläufe der momentanen Drehachse um die Figurennachse notiert, um die Farbwechselrate zu ermitteln. Die Umlauffrequenz der momentanen Drehachse wird nun gegen ω_F aus der Tabelle aufgetragen. Dabei ergibt sich die erstere durch $\Omega = \frac{10 \cdot 2\pi}{T_K}$, wobei T_K die in der Tabelle angegebene Zeit für zehn Farbwechselperioden ist. Der Messfehler der Zeit wird auf 1s geschätzt:

$$\Omega = \frac{10 \cdot 2\pi}{T_K} \quad \Delta \Omega = \frac{10 \cdot 2\pi \Delta T_K}{T_K^2} \quad (23)$$

Das resultierende Diagramm zeigt unten, dabei ist die Steigung der Eichgerade $k = 14.87 \pm 0.13$, aus der wir das Trägheitsmoment des Kreisels aus Gl.(4) wieder bestimmen können.⁵

⁴Python Code 3

⁵Python Code4

$$I_x - I_z = \frac{I_z}{\omega_F/\Omega - 1} = \frac{I_z}{a - 1} \quad (24)$$

$$I_x = \frac{aI_z}{a - 1} \quad \Delta I_x = \sqrt{\left(\frac{a\Delta I_z}{a - 1}\right)^2 + \left(\frac{I_z\Delta a}{(a - 1)^2}\right)^2} \quad (25)$$

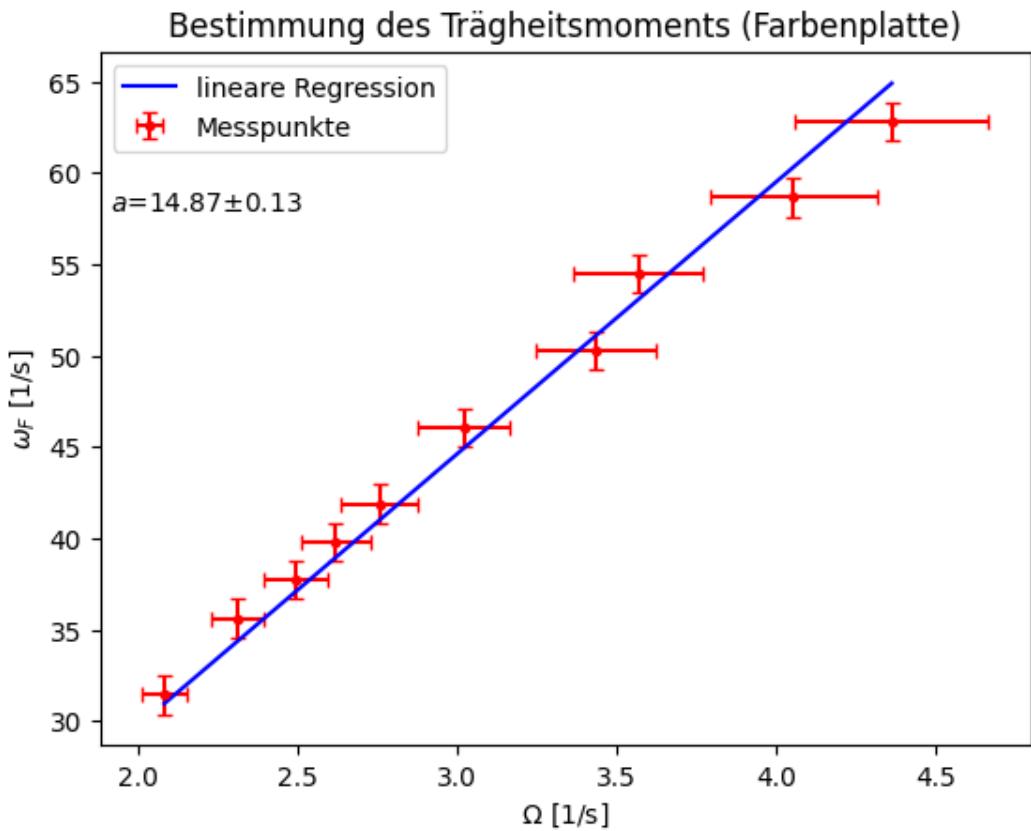


Abbildung 7: Die Farbwechselkreisfrequenz über die Drehkreisfrequenz

So ergibt sich das Trägheitsmoment in bzgl. x-Achse:

$$\underline{\underline{I_x = (4.23 \pm 0.18) g \cdot m^2}} \quad (26)$$

3.4 Nutation

Der kräftefreie Kreisel wurde nun durch vorsichtiges Anschlagen an die Achse in Nutation versetzt. Dabei wurden insgesamt 10 Paare von ω_F und ω_N bestimmt. Aufgrund der Luftreibung nehmen die Frequenzen ständig ab, weshalb die Messungen fehlerhaft bleiben, für die wir einen Fehler 10 rpm vom Stroboskop angenommen haben.

Die gemessenen Werte von ω_F , ω_N werden gegenseitig ins folgende Diagramm eingebracht sowie deren Steigung ermittelt:⁶

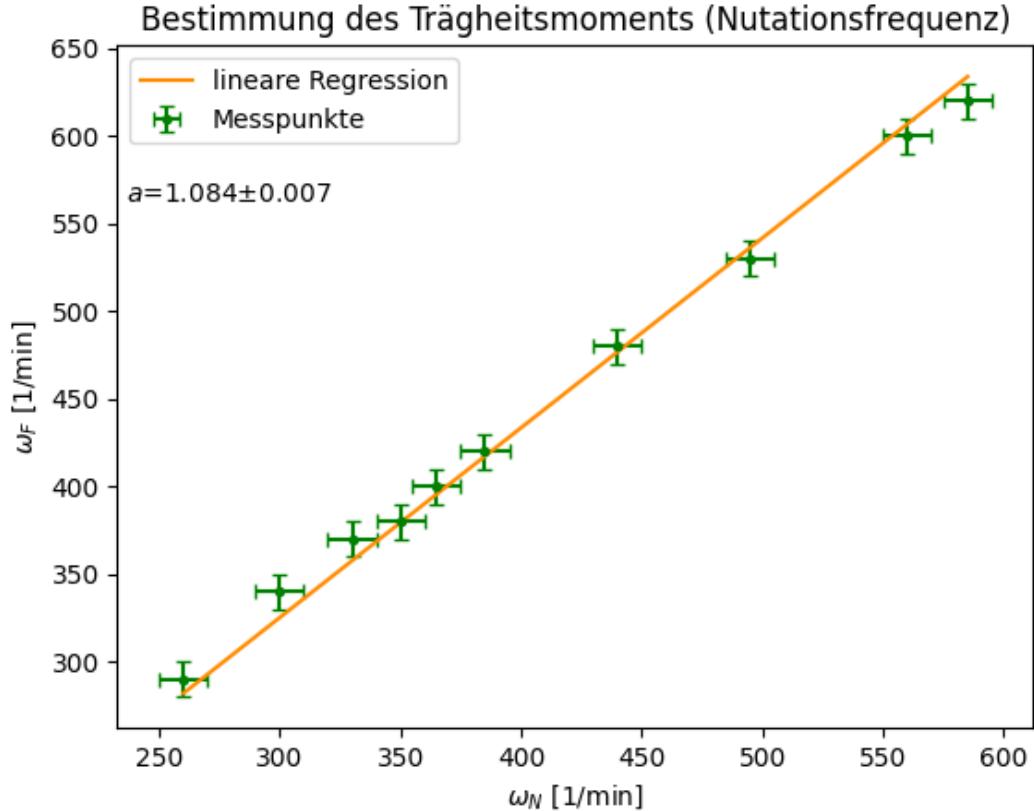


Abbildung 8: Nutationsmessung

Mit der Gl.(3) lässt sich das Trägheitsmoment I_x durch Nutationsfrequenz näherungsweise bestimmen als:

$$I'_x = \frac{I_z \omega_F}{\omega_N} = I_z a' \quad \Delta I'_x = \sqrt{(I_z \Delta a')^2 + (a' \Delta I_z)^2} \quad (27)$$

$$\underline{I'_x = (4.27 \pm 0.18) \text{ g} \cdot \text{m}^2} \quad (28)$$

Im Vergleich zu I_x beträgt die Fehlerabweichung:

$$\frac{|I_x - I'_x|}{\sqrt{(\Delta I_x)^2 + (\Delta I'_x)^2}} \approx 0.16 \quad (29)$$

Die Fehlerabweichung befindet sich innerhalb von 3σ und ist damit nicht signifikant.

4 Diskussion

In diesem Experiment haben wir sowohl qualitativ als auch quantitativ die Kreiselrotation beobachtet und die Trägheitsmomente mit verschiedenen Methoden bestimmt.

⁶Python Code 5

Dabei wird die allgemeine Bewegung eines kräftefreien Kreisels durch eine Nutationsbewegung charakterisiert. Die Figurenachse dreht sich mit der Eigenfrequenz ω_F und rotiert gleichzeitig mit der Nutationsfrequenz ω_N um die raumfeste Drehimpulsachse. Die Winkelgeschwindigkeit variiert mit Ω um die Figurenachse und kann mithilfe einer farbigen Sektorscheibe beobachtet werden. Bei einem schweren Kreisel, dessen Auflagepunkt nicht im Schwerpunkt liegt, führt die Gewichtskraft zu einem Drehmoment und damit zu einer Präzession. Der Drehimpuls bewegt sich dabei mit der Frequenz $\omega\omega_P$ auf einem Kegelmantel um die Richtung der Gewichtskraft.

Da Reibung irgendwie eine wichtige Rolle spielt wurde zuerst die Dämpfungskoeffiziente ermittelt, indem wir die Steigung von Eigenfrequenz-Zeit Regression betrachten. Bei der Messung gestaltet es sich aufgrund menschlicher Ungenauigkeiten bei der Bedienung der Stoppuhr sowie der visuellen Wahrnehmung als herausfordernd, die Drehfrequenz exakt zum bestimmten Zeitpunkt zu erfassen. Aus diesem Grund sind wir bei der Fehlerabschätzung bisschen übertrieben vorgegangen. Die Plausibilität des Werts für die Dämpfungskonstante lässt sich leicht anhand des erhaltenen Werts für die Halbwertszeit überprüfen. Bei Betrachtung der Tabelle im Messprotokoll wird deutlich, dass die Frequenz nach 720 s (dem Maximum der Messreihe) noch nicht vollständig auf die Hälfte abgefallen ist. Wenn man sich den Verlauf der exponentiellen Zerfallskurve aber weiter führt, erscheint der errechnete Wert für die Halbwertszeit grob abgeschätzt, aber dennoch sinnvoll und praktisch für die Vorhersage.

Darüber hinaus führen wir Messungen zur Präzession durch, indem wir dem Kreisel bestimmte Gewichte in verschiedenen Positionen hinzufügen. Die Ergebnisse für die Präzessionsdauern stimmen gut überein, unabhängig vom eingestellten Winkel der Achse, was auch der Erwartung von Gl.(7) entspricht. Durch Anwenden von Mittelwertmethoden können wir die Präzessionsperioden gegen die Eigenfrequenzen auftragen und das Trägheitsmoment bezüglich der z-Achse durch die Steigung bestimmen. In Abbildung 6 zeigen sich unterschiedliche Tendenzen bei den Eichgeräten, wobei der Kreisel mit mehr Gewichten und größerem Drehmoment (größerer Abstand) stärker und schneller präzessiert. Diese Beobachtungen lassen sich theoretisch ebenso durch Gleichung (7) erklären.

Verriegelt sich die Aluminiumstange auf der Kugel auch nur geringfügig, ist der Versuch schwer durchführbar, was die starke Sensitivität des Kreisels gezeigt hat. Eine mögliche systematische Fehlerquelle ist deswegen wahrscheinlich die nicht perfekt gerade Stange wegen der falschen Operationen von vorherigen Gruppen. Allerdings können wir das ermittelte Trägheitsmoment durch die theoretische Trägheitsmomentformel für eine Kugel überprüfen. Dabei nutzen wir die Gleichung:

$$I_{\text{Kugel}} = \frac{2}{5} m R^2 \quad (30)$$

unter der Annahme, dass die Masse sowie die geometrische Umformung der Alu-Stange vernachlässigbar sind. Dies ergibt sich ein Trägheitsmoment $I = 4.30 \text{ g} \cdot \text{m}^2$, wobei eine Fehlerabweichung mit dem Experiment 2.12σ beträgt, die nicht signifikant

ist und sogar wegen der Zusatzmassen und der inhomogenen Massenverteilung der Kugel sehr überzeugend erscheint.

Allerdings erscheint die Mittelwertsmethode in diesem Fall als kritisch, da die Winkelgeschwindigkeit exponentiell mit der Zeit abnimmt. Daher lässt sich die mittlere Geschwindigkeit nicht einfach durch die halbe Addition des Anfangs- und Endwerts darstellen. Eine bessere Vorgehensweise wäre, zunächst die Integration zu berechnen, die aus einer exponentiellen Funktion stammt und daher analytisch lösbar ist. Anschließend können wir diese durch das gesamte Zeitintervall aufteilen.

Im dritten Teil des Versuchs ist vorgesehen, das Trägheitsmoment bezüglich der x-Achse sowohl mithilfe der Farbenscheibe als auch der Nutationsfrequenz zu bestimmen, wobei Gleichungen (3) und (4) verwendet werden. Es ist jedoch zu beachten, dass diese Gleichungen unter Verwendung der Kleinwinkelnäherung hergeleitet werden. Daher sollte die Öffnung des Nutationskegels an der Spitze des Stabes nur zwischen 1 und 2 cm betragen, um Fehler zu vermeiden, insbesondere wenn die Winkel groß sind. Die visuelle Ermüdung durch die längere Betrachtung der Farbenplatte sowie des Blitzlichts kann die Genauigkeit ebenfalls beeinträchtigen. Aus diesem Grund wird empfohlen, dass das Zählen der Farbenwechsel und die Eichung der Nutationsfrequenz durch ein Stroboskop automatisch vom Computer durchgeführt werden, um die Genauigkeit der Messungen zu erhöhen.

Die Fehlerabweichung zwischen beiden Ergebnissen liegt innerhalb von 3σ , was die Konsistenz des Experiments mit der theoretischen Vorhersage zeigt. Im Vergleich zum Trägheitsmoment I_z wird unsere vorherige Behauptung bestätigt, dass $I_x > I_z$ ist. Dies liegt daran, dass, wenn sich der Körper um die z-Achse dreht, die Aluminiumstange kaum eine Rolle spielt. Jedoch, wenn wir die Rotationen um die x- und y-Achsen betrachten, wird deutlich, dass es relativ schwieriger ist, solche Rotationen anzutreiben, da die effektive Länge der Drehung des Objekts deutlich zunimmt.

5 Anhang

Python Code 213

January 15, 2024

0.0.1 Python Code Versuch 213 Yuting Shi

Python Code 1

```
[12]: import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import scipy.integrate as integrate
from scipy.stats import chi2
from IPython.display import HTML, display
from scipy.special import gamma
%matplotlib inline
```

```
[13]: t=np.array([0,2,4,6,8,10,12])
err_t=np.ones(len(t))*0.25
f=np.array([700, 630, 580, 520, 470, 420, 390])
err_f=np.ones(len(f))*10 # Fehler aus Ableseungenauigkeit: 10
# Der Ausdruck np.ones(len(f)) erstellt ein Array mit Einsen

def gerade(x,m,b):
    return b*(np.exp(m*x))

#Messwerte mit Fehler
plt.yscale('log')
plt.plot(t,f,'.')
plt.
    errorbar(t,f,yerr=err_f,xerr=err_t,linewidth=1,fmt='none',capsize=2,color='red',label='Messwerte')

#Ausgleichsgerade
popt, pcov=curve_fit(gerade,t,f)
plt.plot(t,gerade(t,*popt), 'b-', linewidth=1,label='Ausgleichsgerade')

#Fehlergerade
plt.plot(np.array([t[0]-0.2,t[6]+0.2]),np.
    array([f[0],f[6]+4]), 'g--',label='Fehlergerade')

plt.ylim(370, 720)
plt.text(0, 460, '$\delta = (-0.04959 \pm 0.00085) \min^{-1}$')
```

```

plt.text(0, 440, '$\omega_0$ = (699.83$\pm$3.48)$min^{-1}$')
plt.title('Dämpfung des Kreisels')
plt.ylabel('Frequenz [rpm]')
plt.xlabel('Zeit [min]')
plt.legend(loc='upper right')
plt.savefig('kreisel1.pdf', format='PDF')

```

```

print('m =', round(popt[0], 5), '+/-', round(pcov[0, 0]**0.5, 5))
print('b =', round(popt[1], 2), '+/-', round(pcov[1, 1]**0.5, 2))

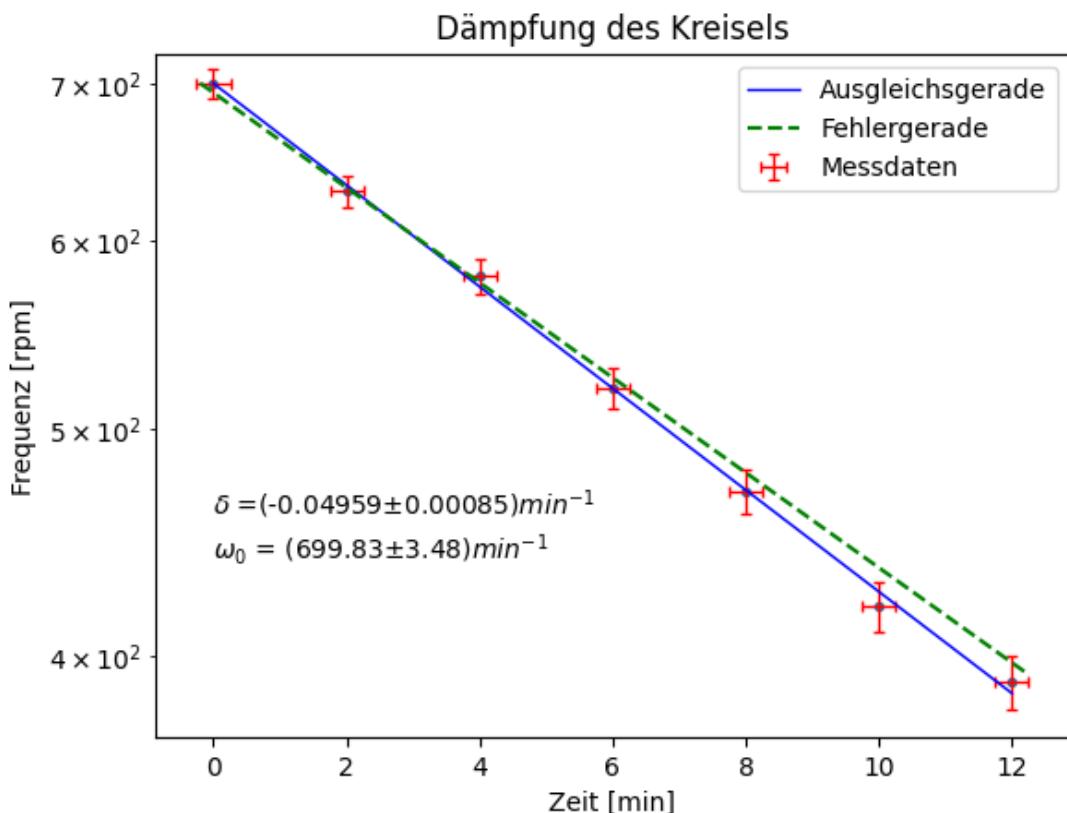
```

```

<>:23: SyntaxWarning: invalid escape sequence '\d'
<>:24: SyntaxWarning: invalid escape sequence '\o'
<>:23: SyntaxWarning: invalid escape sequence '\d'
<>:24: SyntaxWarning: invalid escape sequence '\o'
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\4280320668.py:23:
SyntaxWarning: invalid escape sequence '\d'
    plt.text(0, 460, '$\delta$ =(-0.04959$\pm$0.00085)$min^{-1}$')
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\4280320668.py:24:
SyntaxWarning: invalid escape sequence '\o'
    plt.text(0, 440, '$\omega_0$ = (699.83$\pm$3.48)$min^{-1}$')

m = -0.04959 +/- 0.00085
b = 699.83 +/- 3.48

```



Python Code 2

```
[14]: m=np.array([])
d_m=np.array([])
r=popt[0]/60
d_=np.sqrt(pcov[0,0])/60

#frequenz in 1/s
f_1=np.array([300, 400, 500, 600])/60 #1 bei 15cm
f_2=np.array([300, 400, 500, 600])/60 #1 bei 20cm
f_3=np.array([300, 400, 500, 600])/60 #2 bei 15cm
f_4=np.array([300, 400, 500, 600])/60 #2 bei 20cm
d_f=10/60

#drehfrequenz anfangswerte
w_0_1=2*np.pi*f_1
w_0_2=2*np.pi*f_2
w_0_3=2*np.pi*f_3
w_0_4=2*np.pi*f_4
d_w_0=2*np.pi*d_f

#zeit in s
t_1=np.array([52, 66, 82, 97])
t_2=np.array([38, 52, 65, 75])
t_3=np.array([27, 36, 44, 53])
t_4=np.array([21, 27, 36, 42])
d_t=1

#drehfrequenz endwerte
w_e_1=np.exp(r*t_1)*w_0_1
w_e_2=np.exp(r*t_2)*w_0_2
w_e_3=np.exp(r*t_3)*w_0_3
w_e_4=np.exp(r*t_4)*w_0_4

d_w_e_1=w_e_1*np.sqrt((d_w_0/w_0_1)**2+(t_1*d_)**2+(r*d_t)**2)
d_w_e_2=w_e_2*np.sqrt((d_w_0/w_0_2)**2+(t_2*d_)**2+(r*d_t)**2)
d_w_e_3=w_e_3*np.sqrt((d_w_0/w_0_3)**2+(t_3*d_)**2+(r*d_t)**2)
d_w_e_4=w_e_4*np.sqrt((d_w_0/w_0_4)**2+(t_4*d_)**2+(r*d_t)**2)

#drehfrequenz mittelwerte
w_1=0.5*(w_0_1+w_e_1)
w_2=0.5*(w_0_2+w_e_2)
w_3=0.5*(w_0_3+w_e_3)
w_4=0.5*(w_0_4+w_e_4)
```

```

d_w_1=0.5*np.sqrt(d_w_0**2+d_w_e_1**2)
d_w_2=0.5*np.sqrt(d_w_0**2+d_w_e_2**2)
d_w_3=0.5*np.sqrt(d_w_0**2+d_w_e_3**2)
d_w_4=0.5*np.sqrt(d_w_0**2+d_w_e_4**2)

```

```

[29]: def linear(x,a):
    return a*x
#fehler einzeichnen

#Punkt graphisch darstellen Anzahl 1 bei 15cm
popt, pcov=curve_fit(linear, w_1, t_1, maxfev=1000)
plt.errorbar(w_1, t_1, yerr=d_t, xerr=d_w_1, fmt='none', capsize=2, color='blue')
#graph plotten
x=np.linspace(27,66,1000)
plt.plot(x, linear(x, *popt), label='Dataset 1: $T_P = a\omega_F$', linewidth=1, color='blue')

# Extract the slope and its error
slope_1 = popt[0]
slope_error_1 = np.sqrt(pcov[0, 0])
# Print the slope and its error for the first dataset
print(f"Slope for Dataset 1: {slope_1:.3f} ± {slope_error_1:.3f}")

#Punkt graphisch darstellen Anzahl 1 bei 20cm
popt, pcov=curve_fit(linear, w_2, t_2, maxfev=1000)
plt.errorbar(w_2, t_2, yerr=d_t, xerr=d_w_2, fmt='none', capsize=2, color='red')
#graph plotten
x=np.linspace(27,66,1000)
plt.plot(x, linear(x, *popt), label='Dataset 2: $T_P = b\omega_F$', linewidth=1, color='red')

# Extract the slope and its error
slope_2 = popt[0]
slope_error_2 = np.sqrt(pcov[0, 0])
# Print the slope and its error for the first dataset
print(f"Slope for Dataset 2: {slope_2:.3f} ± {slope_error_2:.3f}")

#Punkt graphisch darstellen Anzahl 2 bei 15cm
popt, pcov=curve_fit(linear, w_3, t_3, maxfev=1000)
plt.errorbar(w_3, t_3, yerr=d_t, xerr=d_w_3, fmt='none', capsize=2, color='green')
#graph plotten
x=np.linspace(27,66,1000)
plt.plot(x, linear(x, *popt), label='Dataset 3: $T_P = c\omega_F$', linewidth=1, color='green')

```

```

# Extract the slope and its error
slope_3 = popt[0]
slope_error_3 = np.sqrt(pcov[0, 0])
# Print the slope and its error for the first dataset
print(f"Slope for Dataset 3: {slope_3:.3f} ± {slope_error_3:.3f}")

#Punkt graphisch darstellen Anzahl 2 bei 20cm
popt, pcov=curve_fit(linear, w_4, t_4, maxfev=1000)
plt.errorbar(w_4, t_4, yerr=d_t, xerr=d_w_4, fmt='none', capsized=2, color='orange')
#graph plotten
x=np.linspace(27,66,1000)
plt.plot(x, linear(x, *popt), label='Dataset 4: $T_P = d\omega_F$', linewidth=1, color='orange')

# Extract the slope and its error
slope_4 = popt[0]
slope_error_4 = np.sqrt(pcov[0, 0])
# Print the slope and its error for the first dataset
print(f"Slope for Dataset 4: {slope_4:.3f} ± {slope_error_4:.3f}")

# Plot settings
plt.ylabel('Praezessionszeit $T_P$ [s]')
plt.xlabel('Eigenfrequenz $\omega_F$ [1/s]')
plt.title('Präzessionsmessung')
plt.legend()
plt.savefig('kreisel2.pdf', format='PDF')
plt.show()

```

```

<>:10: SyntaxWarning: invalid escape sequence '\o'
<>:23: SyntaxWarning: invalid escape sequence '\o'
<>:36: SyntaxWarning: invalid escape sequence '\o'
<>:49: SyntaxWarning: invalid escape sequence '\o'
<>:59: SyntaxWarning: invalid escape sequence '\o'
<>:10: SyntaxWarning: invalid escape sequence '\o'
<>:23: SyntaxWarning: invalid escape sequence '\o'
<>:36: SyntaxWarning: invalid escape sequence '\o'
<>:49: SyntaxWarning: invalid escape sequence '\o'
<>:59: SyntaxWarning: invalid escape sequence '\o'
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\2271576670.py:10:
SyntaxWarning: invalid escape sequence '\o'
    plt.plot(x, linear(x, *popt), label='Dataset 1: $T_P = a\omega_F$', linewidth=1, color='blue')
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\2271576670.py:23:
SyntaxWarning: invalid escape sequence '\o'
    plt.plot(x, linear(x, *popt), label='Dataset 2: $T_P = b\omega_F$', linewidth=1, color='red')

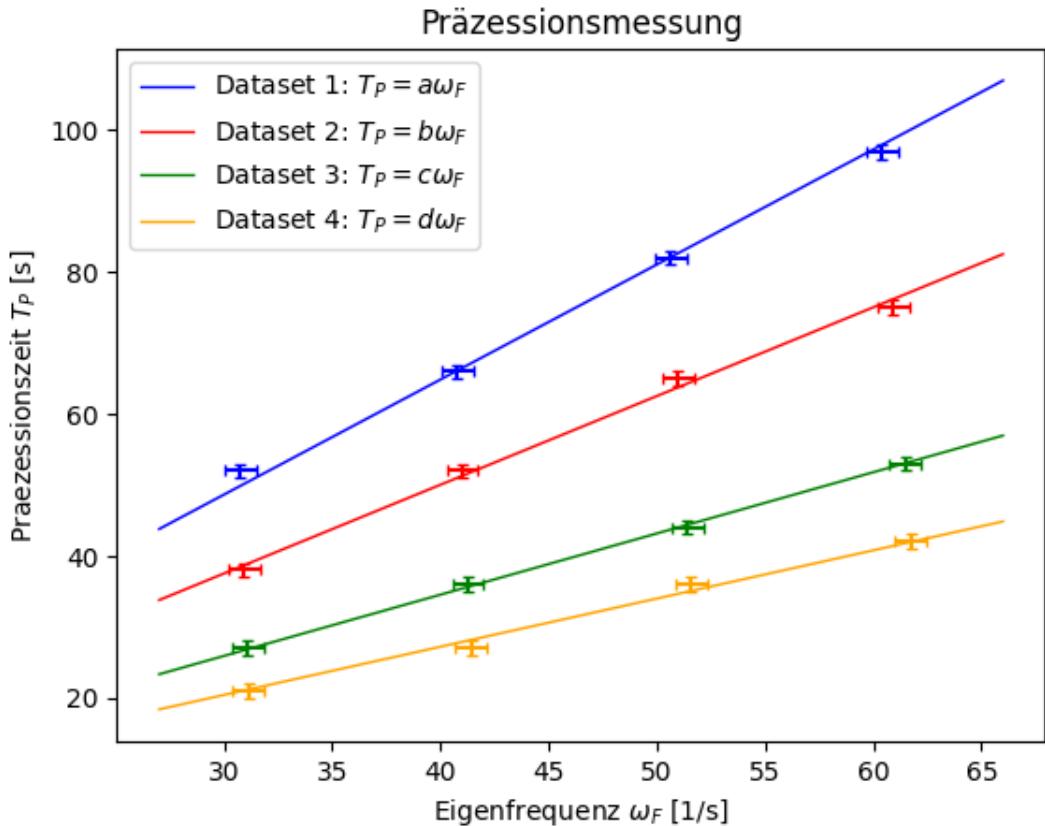
```

```

C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\2271576670.py:36:
SyntaxWarning: invalid escape sequence '\o'
    plt.plot(x, linear(x, *popt), label='Dataset 3: $T_P = c\omega_F$', linewidth=1, color='green')
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\2271576670.py:49:
SyntaxWarning: invalid escape sequence '\o'
    plt.plot(x, linear(x, *popt), label='Dataset 4: $T_P = d\omega_F$', linewidth=1, color='orange')
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\2271576670.py:59:
SyntaxWarning: invalid escape sequence '\o'
    plt.xlabel('Eigenfrequenz $\omega_F$ [1/s]')

Slope for Dataset 1: 1.621 ± 0.014
Slope for Dataset 2: 1.250 ± 0.012
Slope for Dataset 3: 0.863 ± 0.003
Slope for Dataset 4: 0.679 ± 0.009

```



Python Code 3

```
[30]: #traegheitsmoment I_z bestimmen
```

```
l=np.array([0.15,0.20,0.15,0.20]) #position der gewichte in meter
m=np.array([9.85,9.85,2*9.85,2*9.85]) #masse der gewichte in gramm
g=9.81 #in m/s^2

#steigung der praezessionsgeraden
a=np.array([1.621, 1.250, 0.863, 0.679])
da=np.array([0.014, 0.012, 0.003, 0.009])

#traegheitsmoment berechnen
Iz=a*m*g*l/(2*np.pi)
dIz=a*m*g*l/(2*np.pi)*da/a

#mittelwert uns standardabweichung berechnen
mean_Iz=np.mean(Iz)
std_Iz=np.std(Iz)

#gesamtfehler berechnen
dIz_ges=np.mean(np.sqrt(std_Iz**2+dIz**2))

print('I_z=', mean_Iz, '+/-', dIz_ges)
```

I_z= 3.93565362110101 +/- 0.1678767693567657

Python Code 4

```
[43]: #Umlauf der momentanen Drehachse um die Figurennachse
```

```
Zeiten = 1/10 * np.array([30.2, 27.2, 25.2, 24.0, 22.8, 20.8, 18.3, 17.6, 15.5, ↳14.4])
Fehler_Zeiten = 0.1 * 1

Omega = (2*np.pi)/Zeiten
Fehler_Omega = 2*np.pi*Fehler_Zeiten/Zeiten**2

Omega_F = 2*np.pi * 1/60 * np.array([300, 340, 360, 380, 400, 440, 480, 520, ↳560, 600])
Fehler_Omega_F = 2*np.pi * 1/60 * 10

plt.title('Bestimmung des Trägheitsmoments (Farbenplatte)')
plt.ylabel(r'$\omega_F$ [1/s]')
plt.xlabel(r'$\Omega$ [1/s]')
plt.errorbar(Omega, Omega_F, xerr=Fehler_Omega, yerr=Fehler_Omega_F, ↳color='red', fmt='.', capsiz=3, label='Messpunkte')
popt, pcov = curve_fit(linear, Omega, Omega_F)
a_1 = popt[0]
err_a_1 = np.sqrt(pcov[0][0])
```

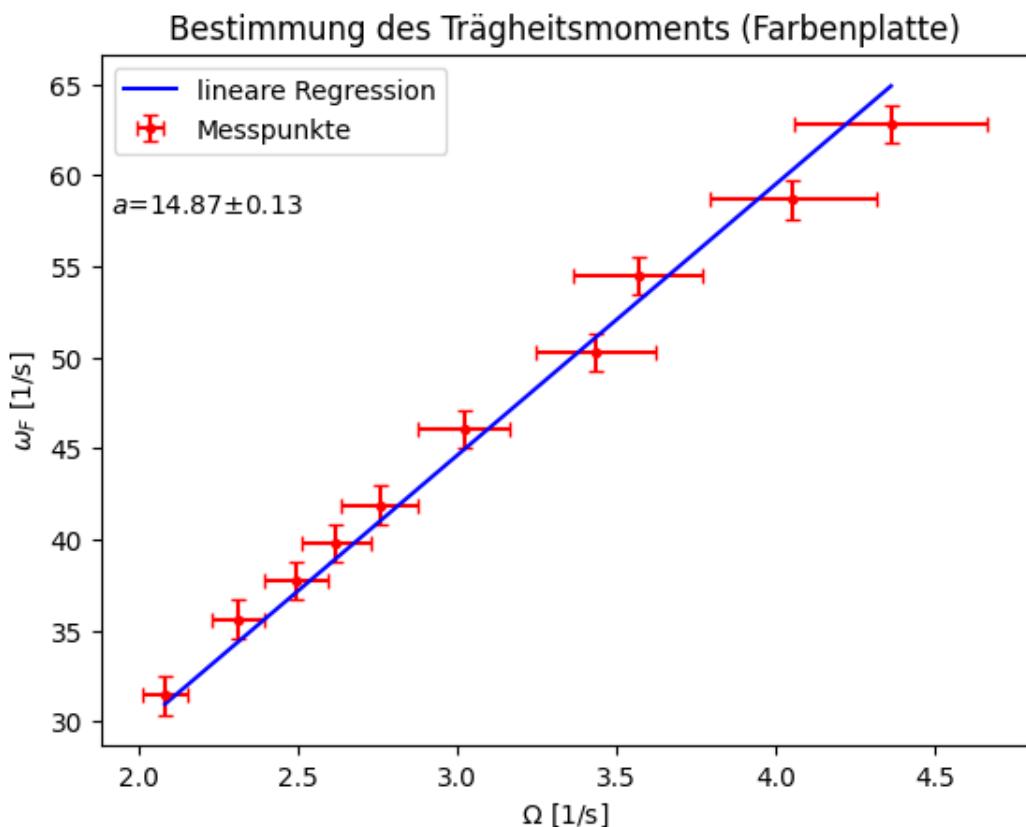
```

plt.plot(Omega, linear(Omega,*popt), label='lineare Regression', color='blue')
plt.annotate(r'$a$={val1:.2f}$\pm${val2:.2f}'.format(val1=popt[0], val2=np.
    sqrt(pcov[0,0])), xy=(0.01, 0.77), xycoords='axes fraction')
plt.legend(loc='best')
plt.savefig('kreisel6.pdf', format='PDF')
print(a_1)
print(err_a_1)

print(Omega)
print(Omega_F)

```

14.873149247230804
0.12719222906508088
[2.08052494 2.3099946 2.4933275 2.61799388 2.75578303 3.02076217
3.43343459 3.56999165 4.05366794 4.36332313]
[31.41592654 35.60471674 37.69911184 39.79350695 41.88790205 46.07669225
50.26548246 54.45427266 58.64306287 62.83185307]



Python Code 5

```
[42]: #bestimmung der nutation
```

```
RPM_F=np.array([620,600,530,480,420,400,380,370,340,290])
RPM_N=np.array([585,560,495,440,385,365,350,330,300,260])
err_RPM=10

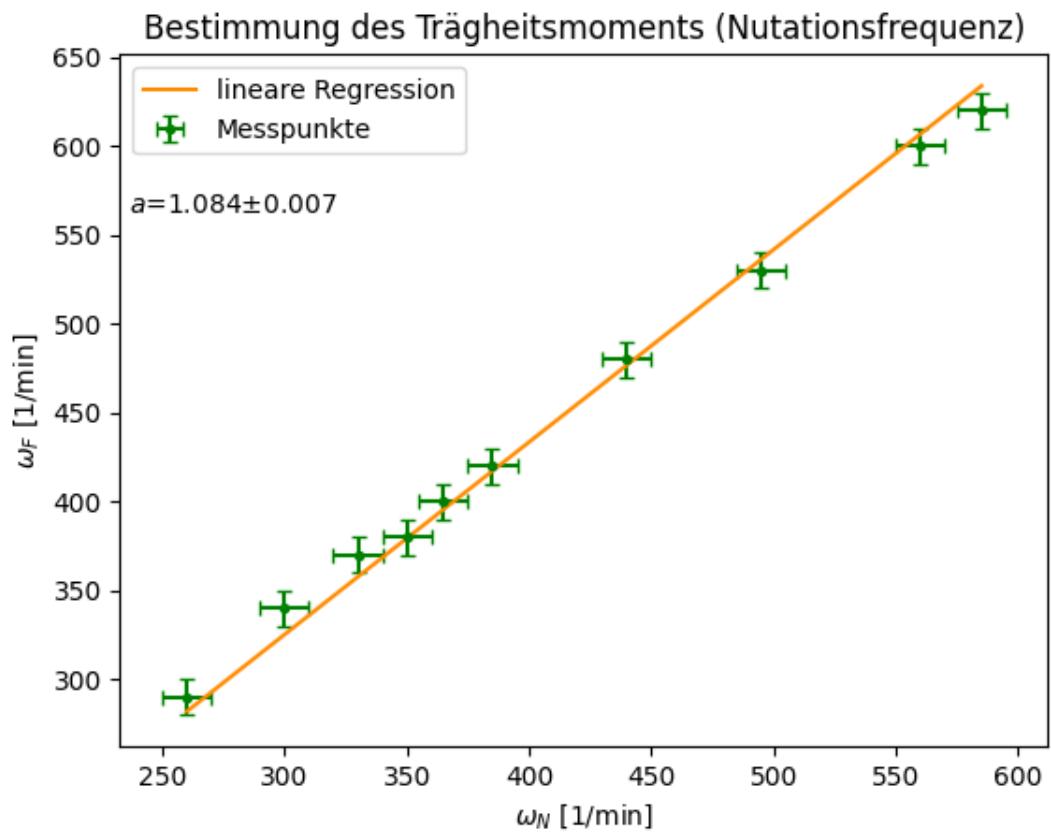
plt.title('Bestimmung des Trägheitsmoments (Nutationsfrequenz)')
plt.ylabel(r'$\omega_F$ [1/min]')
plt.xlabel(r'$\omega_N$ [1/min]')
plt.errorbar(RPM_N, RPM_F, xerr=err_RPM, yerr=err_RPM, color='green', fmt='.', capsize=3, label='Messpunkte')

#kurve fitten
popt, pcov=curve_fit(linear, RPM_N, RPM_F)
a_2=popt[0]
err_a_2=np.sqrt(pcov[0][0])
plt.plot(RPM_N, linear(RPM_N,*popt), label='lineare Regression', color='darkorange')
plt.annotate('$a={val1:.3f}\pm{val2:.3f}'.format(val1=popt[0], val2=np.sqrt(pcov[0,0])), xy=(0.01,0.77), xycoords='axes fraction')
plt.legend(loc='best')
plt.savefig('kreisel7.pdf', format='PDF')

print(a_2)
print(err_a_2)
```

```
<>:17: SyntaxWarning: invalid escape sequence '\p'
<>:17: SyntaxWarning: invalid escape sequence '\p'
C:\Users\shiy0\AppData\Local\Temp\ipykernel_8176\4166434854.py:17:
SyntaxWarning: invalid escape sequence '\p'
    plt.annotate('$a={val1:.3f}\pm{val2:.3f}'.format(val1=popt[0],
val2=np.sqrt(pcov[0,0])), xy=(0.01,0.77), xycoords='axes fraction')

1.0836024703719827
0.006937368987075183
```



[]:

6 Quelle

- Wagner, J. (April 2022). Physikalisches Praktikum PAP 2.1 für Studierende der Physik [Praktikumsanleitung]. Ruprecht-Karls-Universität Heidelberg. Abgerufen am 29. Oktober 2023, von https://www.physi.uni-heidelberg.de/Einrichtungen/AP/info/Corona /PAP2_1_2023.pdf