

Fakultät für Physik der Ludwig-Maximilians-Universität München
Grundpraktikum in Experimentalphysik - Kurs P1
Blockpraktikum vom 23. Februar bis 23. März 2026

Versuch:					
Vorname 1:	Name 1:				
Vorname 2:	Name 2:				
Mit Abgabe der Auswertung wird bestätigt, dass diese eigenständig erstellt wurde!					
Die Abgabe ist vor dem Einreichen auf eine saubere äußere Form und Struktur zu kontrollieren. Bei ungenügender äußerer Form erfolgt zunächst keine Korrektur!				<input type="checkbox"/> OK?	
		1. Abgabe		2. Abgabe	
Alle Teilversuche vollständig ausgewertet?		Ja	Nein	Ja	Nein
Wurden immer korrekte Formeln angegeben und eigene Werte eingesetzt?		Ja	Nein	Ja	Nein
Wurde immer eine Fehlerrechnung durchgeführt?		Ja	Nein	Ja	Nein
Sind Endergebnisse immer angegeben und korrekt gerundet?		Ja	Nein	Ja	Nein
Wurde immer eine aussagekräftige Diskussion geführt?		Ja	Nein	Ja	Nein
Wurden alle Diagramme mit geeignetem Maßstab und Titel eingefügt?		Ja	Nein	Ja	Nein
Enthalten die Diagramme alle Messwerte, Beschriftungen u. Konstruktionen?		Ja	Nein	Ja	Nein
Sind ausgefülltes Deckblatt, Vorbereitung und Messprotokoll in der Abgabe enthalten?		Ja	Nein	Ja	Nein
Auswertung erhalten am:					
Auswertung zurückgegeben am:					
Nacharbeit notwendig bis:		nicht möglich			
Abzug 0,2 Punkte pro Nacharbeit/angefangene 3 Tage Verspätung:		-0,2	-0,2	-0,2	-0,2
Wird eine der obigen Fragen bei der ersten Abgabe mit Nein beantwortet ist eine Nacharbeit erforderlich!					
Punkte:		Datum, Abtestat:			

Bitte bewahren Sie Ihre Hefte nach dem Praktikum unbedingt auf.

Auswertung und Protokoll

Zum Versuch STO

Jonas Müther & Alejandro Schultheiss

P1 Praktikum

LMU München
Physik B.Sc.
Deutschland
2026

Inhaltsverzeichnis

1	Vorbereitung	2
1.1	Versuchsvorbereitung und Grundlagen des Versuchs	2
1.2	Versuchsprotokoll	9
2	Auswertung	20
2.1	Teilversuch I: Flugweiten verschiedener Kugeln	20
2.2	Teilversuch II: Elastischer Stoß von Kugeln gleicher Masse	21
2.2.1	Theoretischer Hintergrund	21
2.2.2	Grafische Analyse der Landepunkte	21
2.2.3	Abweichung von der Theorie	23
2.2.4	Veränderung der Landepunkte bei unterschiedlichen Massen	28
2.3	Teilversuch III: Bewegungsanalyse mit Hochgeschwindigkeitskamera	30
2.4	Teilversuch IV: Bestimmung der Erdbeschleunigung	31
2.4.1	Theoretischer Hintergrund	31
2.4.2	Erwartete Falldauer	31
3	Anhang	33
3.1	Python Scripts	33
3.1.1	TV I:	33
3.1.2	TV II:	33
3.1.3	TV III:	38
3.1.4	TV IV:	38
3.2	Verwendete Daten	38
3.2.1	TV I:	38
3.2.2	TV II:	38
3.2.3	TV III:	38
3.2.4	TV IV:	38

Kapitel 1

Vorbereitung

1.1 Versuchsvorbereitung und Grundlagen des Versuchs

Fakultät für Physik der Ludwig-Maximilians-Universität München
Grundpraktikum in Experimentalphysik - Kurs P1
Blockpraktikum vom 23. Februar bis 23. März 2026

Vorname:	<i>Sonja</i>	Name:	<i>Müller</i>	Gruppe:	<i>D1 - 1</i>
----------	--------------	-------	---------------	---------	---------------

Mit Abgabe der Vorbereitung wird bestätigt, dass diese eigenständig erstellt wurde!

Die Abgabe ist vor dem Einreichen auf eine saubere äußere Form und Struktur zu kontrollieren.
Bei ungenügender äußerer Form erfolgt zunächst keine Korrektur!

OK?

	Schriftliche Vorbereitung						Vortrag					
STV	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					
ROT	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					
FLU	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					
STW	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					
STO	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					
MOS	2,0	1,6	1,2	0,8	0,4	0,0	2,0	1,6	1,2	0,8	0,4	0,0
	Antestat:						Antestat:					

Bitte bewahren Sie Ihre Hefte nach dem Praktikum unbedingt auf.

Stöße (STO)

Vorbereitung

Jonas Mütter, 23.02.2025

Versuchsanordnung

Beteiligte Energieformen

- potentielle (Lage-) Energie $E_{\text{pot}} = m \cdot g \cdot h$
- kinetische Energie $E_{\text{kin}} = \frac{1}{2} m v^2$
- Rotationsenergie $E_{\text{rot}} = \frac{1}{2} I \omega^2$

Herleitung der Abfluggeschwindigkeit

Energieansatz:

$$E_{\text{pot}}^{(0)} = E_{\text{kin}}^{(1)} + E_{\text{rot}}^{(1)}$$

$$E_{\text{pot}} = \frac{1}{2} I \omega^2$$



$$r' = r \cdot \sin \frac{\alpha}{2}$$

$$m \cdot g \cdot h = \frac{1}{2} m v^2 + \frac{1}{2} \frac{2}{\sin^2 \frac{\alpha}{2}} m v^2$$

$$= \frac{1}{2} \cdot I \cdot \left(\frac{v}{r'}\right)^2$$

$$g \cdot h = \frac{1}{2} v^2 \left(1 + \frac{2}{\sin^2 \frac{\alpha}{2}}\right)$$

$$= \frac{1}{2} \cdot I \cdot \frac{v^2}{r' \sin^2 \frac{\alpha}{2}}$$

$$v = \sqrt{\frac{2g \cdot h}{1 + \frac{2}{\sin^2 \frac{\alpha}{2}}}}$$

$$= \frac{1}{2} \cdot \frac{2}{5} m \cdot r^2 \cdot \frac{v^2}{r^2 \sin^2 \frac{\alpha}{2}}$$

$$= \frac{1}{2} \cdot \frac{2}{5 \sin^2 \frac{\alpha}{2}} m v^2$$

Messinstrumente zur Längenmessung

1. Stab mit Längenskala (Abstand der Striche 1mm)

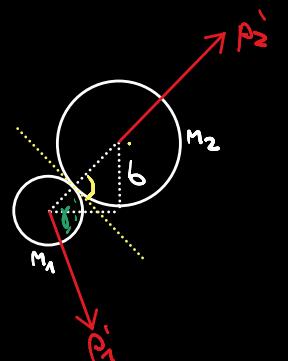
→ Auflösung von 0,5mm (durch präzises Abschätzen)

2. Schieblehre:

Messvorgang: 1. Ganze Millimeter: links vom Nullstrich der Noniusskala

2. Millimetersbruchteile: Strich der Noniusskala, der mit Strich der Millimeterskala übereinstimmt.

Stoßvorgang:



Stoßparameter δ

Variation des Stoßparameters: Drehen der Einstellschraube:
0,05mm Vorschub pro Skaleneinheit der Schraube

Energieerhaltung: $\frac{1}{2}mv_1^2 = \frac{1}{2}mv_1'^2 + \frac{1}{2}mv_2'^2$

Impulserhaltung: $\vec{p}_i = \vec{p}_1 + \vec{p}_2$

$$m_1 \vec{v}_1 = m_1 \vec{v}_1' + m_2 \vec{v}_2'$$

Resultate für verschiedene Relationen von Projektil- und Targetmasse

$m_p = m_t \Rightarrow \gamma = 90^\circ$, $m_p > m_t \Rightarrow \gamma < 90^\circ$, $m_p < m_t \Rightarrow \gamma > 90^\circ$

Charakteristische Ortskurven: $\frac{R_p}{R_t} = \frac{m_t}{m_p}$

Energieverlust durch Reibung

- Reibung beim Rollen auf der Schiene
- Rollreibung durch mikroskopische Verformung der Unterlage
- Gleitreibung durch Relativbewegung zwischen Kugeloberfläche und Schiene
- Verformung während des Stoßvorgangs \Rightarrow geringere Geschwindigkeit
- Nichtzentrale Stoß: Gleitreibung \Rightarrow Verkleinerung des Stoßuhlers

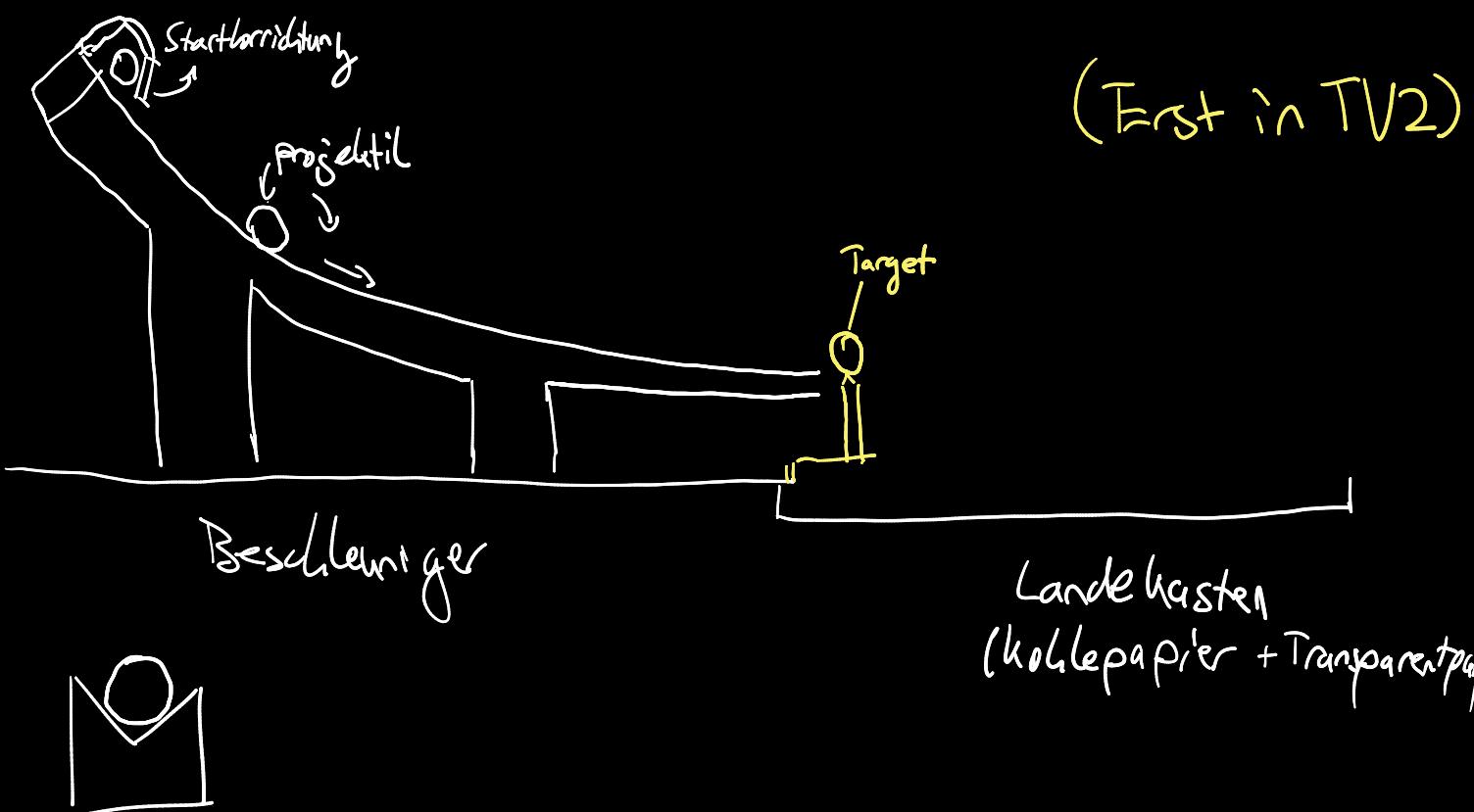
Reibungskoeffizient

$$K = \frac{h_1}{s} \left(1 - \frac{\delta_{\text{exp}}^2}{\delta_{\text{th}}^2} \right)$$

$s \hat{=}$ effektive Balllänge, die auf horizontale nötig wäre.

Versuchsaufbauplan:

TV1 & TV2



TV1:

Frage: Welche Kugel entspricht am Besten der Theorie für rutschfreies Rollen.

Durchführung: Kugeln verschiedener Materialien den Beschleunigungen entlang rollen lassen und Flugweite / Auftreffpunkte auf Landekästen bestimmen

TV2:

Untersuchung des theoretischen Stoßverhaltens für verschiedene Stoßparameter

Durchführung: Verschiedene Stoßparameter durch verschieben der Targetplattform einstellen und dann jeweils die Projektilkugel auf die Targetkugel beschleunigen und die Auftreffpunkte der Kugeln bestimmen

TV3:

Hochgeschwindigkeitsaufnahme einer Kugel auf dem Beschleuniger und in der Flugphase (500 fps)

Hardware:

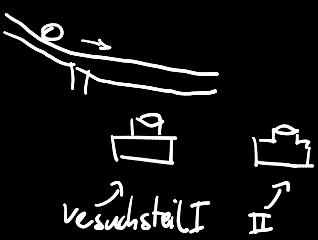
- Kameramodul mit Weitwinkel
- Hochgeschwind. Bildfassungskarte
- Leistungssstarkes Rechner

Software:

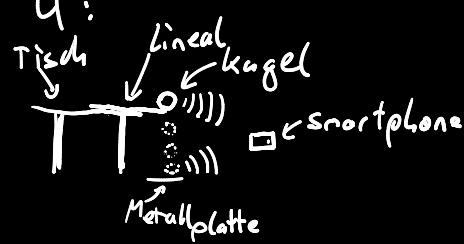
- Steuersoftware: Einstellung der Kameraparameter (Auflösung, Frame rate)

- Aufnahmesoftware: Schreibt Aufnahme in RAM, speichert auf Festplatte (Streampix)

- Videanalysesoftware: Bewegungsanalyse im Video (kann Objekte tracken)



TV 4:



Kugeln werden auf Metalllineal platziert, dann wird dies mit einem Stift weggehauen, so dass die Kugel beginnt zu fallen. Am Ende des Falls landet die Metallkugel auf einer Metallplatte.

Ein Smartphone mit Phyphox bestimmt die Dauer zwischen den Geräuschen des Weghauens des Lineals und dem Auftreffen auf der Metallplatte und somit indirekt die Fallzeit.
Der Versuch wird für verschiedene Höhen wiederholt.

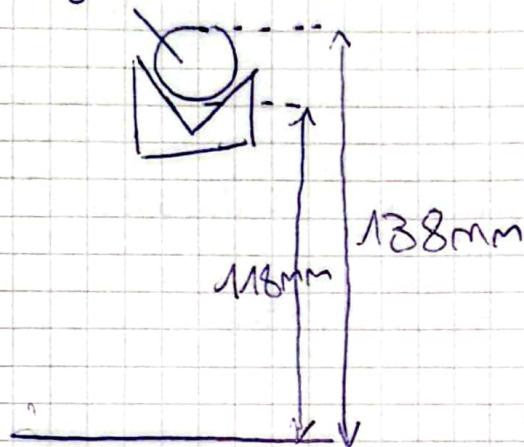
1.2 Versuchsprotokoll

(D)

Jonas Müthes, Alejandro Schultheiss

Teil I

$$d = 20\text{mm}$$



$$h_{\text{fall}} = 118 \text{ mm} \quad (\text{durchfallene Höhe})$$

Kugeldurchmesser

$$d_{1,1} = 20,00 \text{ mm} \quad (\text{Stahlkugel}) \quad \left. \right\} \text{Messdurchgang 1}$$

$$d_{2,1} = 16,30 \text{ mm} \quad (\text{Murmel})$$

$$d_{3,1} = 15,60 \text{ mm} \quad (\text{Plastikkugel})$$

$$d_{1,2} = 20,00 \text{ mm} \quad (\text{Stahlkugel}) \quad \left. \right\} \text{Messdurchgang 2}$$

$$d_{2,2} = 16,40 \text{ mm} \quad (\text{Murmel})$$

$$d_{3,2} = 15,50 \text{ mm} \quad (\text{Plastikkugel})$$

$$d_{1,3} = 20,00 \text{ mm} \quad (\text{Stahlkugel}) \quad \left. \right\} \text{Messdurchgang 3}$$

$$d_{2,3} = 16,35 \text{ mm} \quad (\text{Murmel})$$

$$d_{3,3} = 15,70 \text{ mm} \quad (\text{Plastikkugel})$$

Kugeldurchmesser final: (inkl. Fehler)

Stahlkugel:

$$\langle d_1 \rangle = \frac{d_{1,1} + d_{1,2} + d_{1,3}}{3} = \frac{20,00\text{mm} + 20,00\text{mm} + 20,00\text{mm}}{3} = 20,00\text{mm}$$

Fehler des Messschreibers: $\pm 0,03\text{mm}$ (Wird im folgenden auf individuellen Messfehler aufaddiert)

$$\Rightarrow d_1 = 20,00\text{mm} \pm 0,03\text{mm}$$

Murmel:

$$\langle d_2 \rangle = \frac{d_{2,1} + d_{2,2} + d_{2,3}}{3} = \frac{16,30\text{mm} + 16,40\text{mm} + 16,35\text{mm}}{3} = 16,35\text{mm}$$

$$\Rightarrow 16,35\text{mm} \pm 0,08\text{mm}$$

Plastikkugel:

$$\langle d_3 \rangle = \frac{d_{3,1} + d_{3,2} + d_{3,3}}{3} = \frac{15,60\text{mm} + 15,50\text{mm} + 15,70\text{mm}}{3} = 15,60\text{mm}$$

$$\Rightarrow 15,60\text{mm} \pm 0,13\text{mm}$$

Zuordnung von Kennzeichnung und Material:

O - Kreis \cong Plastikkugel

X - Kreuz \cong Stahlkugel

Ohne Kennzeichnung \cong Murmel

Fehlerschätzung beim Abschlagen des Schienen-elements:

Wir schätzen den Fehler auf 1,5mm (Stiftspitzenbreite)

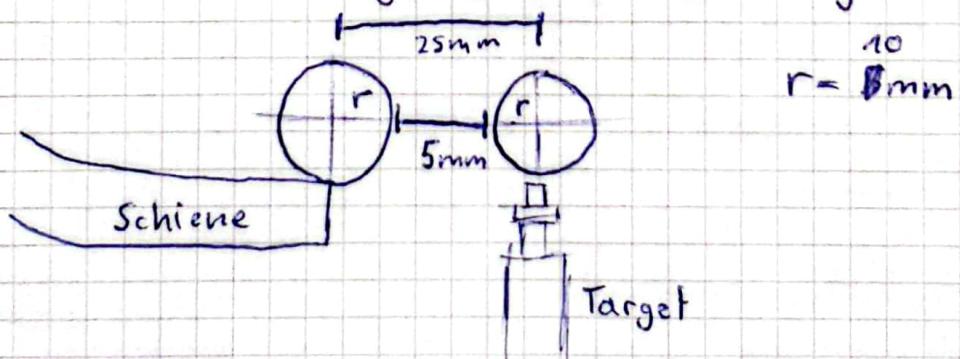
Wir entscheiden uns für die Stahlkugel, aufgrund der hohen ~~Konsistenz~~ bzw. höchsten Konsistenz der Auf treffpunkte, relativ zu den anderen Kugeln.

Wir nehmen an, dass dies aus der höchsten Trägheit (relativ zu den anderen Kugeln) und dem gleichmässigstem Radius (rel. zu den andern Kugeln) resultiert.

↳ siehe vorherige Seite

Teil II:

Um den Abstand auf 5mm (zwischen Schiene und Targetgestell) zu bringen, haben wir uns das anhand folgender Skizze überlegt:



Wir haben den Abstand zwischen Schiene und Target auf 25mm eingestellt.

Einstellung Mikrometerschraube: 0,45

Teil III:

Wir benutzen die Stahlkugel
mit Durchmesser $d = \textcolor{blue}{\cancel{20}} \text{ mm}$.

Für die Kugel ergibt sich ein
Durchmesser inkl. Fehler von:

$$d = 20,00 \text{ mm} \pm 0,003 \text{ mm}$$

(S. vorherige Seite)

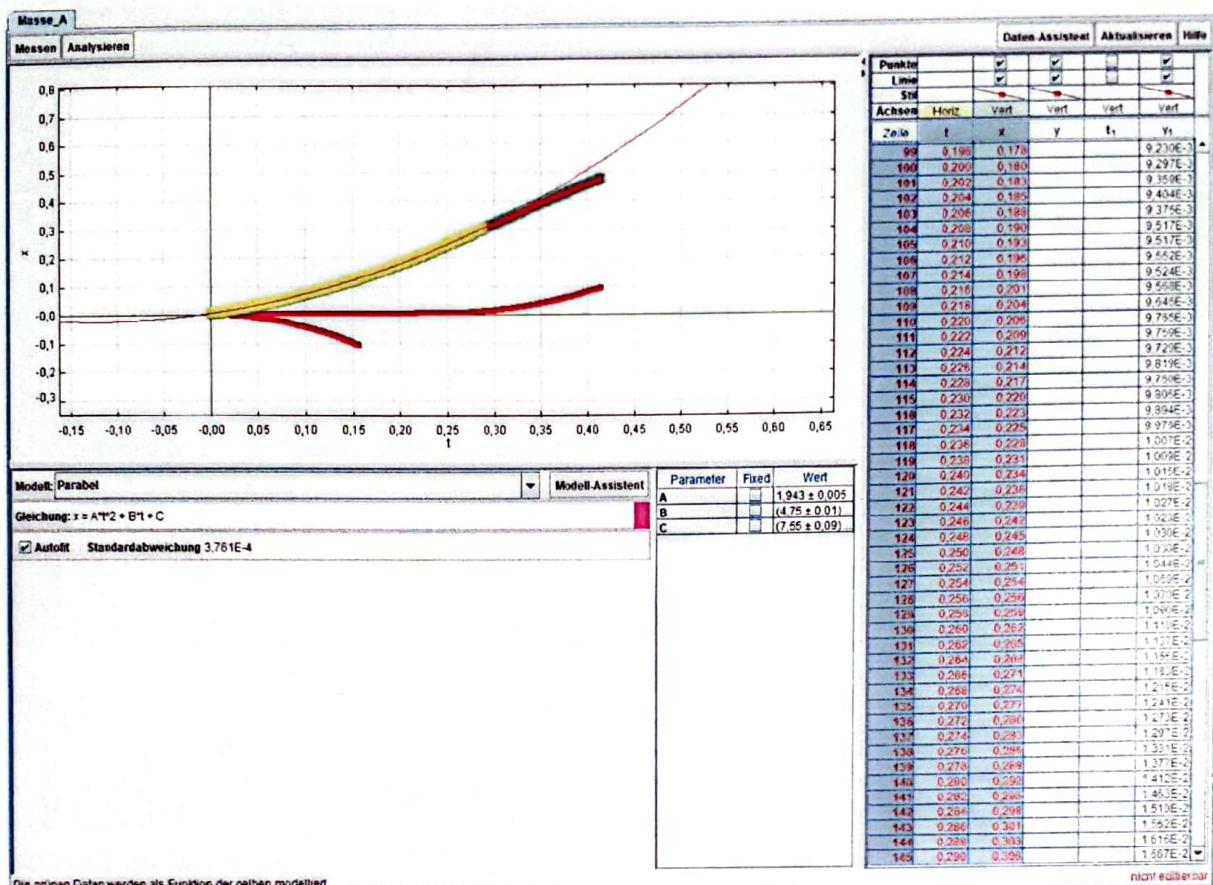
Video 1, welches das Herabrollen der Kugel
zeigt, haben wir wie folgt genannt:

"Jonas-Alejandro-1.avi"

Video 2, welches ~~die~~ den waagerechten
Wurf zeigt, haben wir wie folgt genannt:

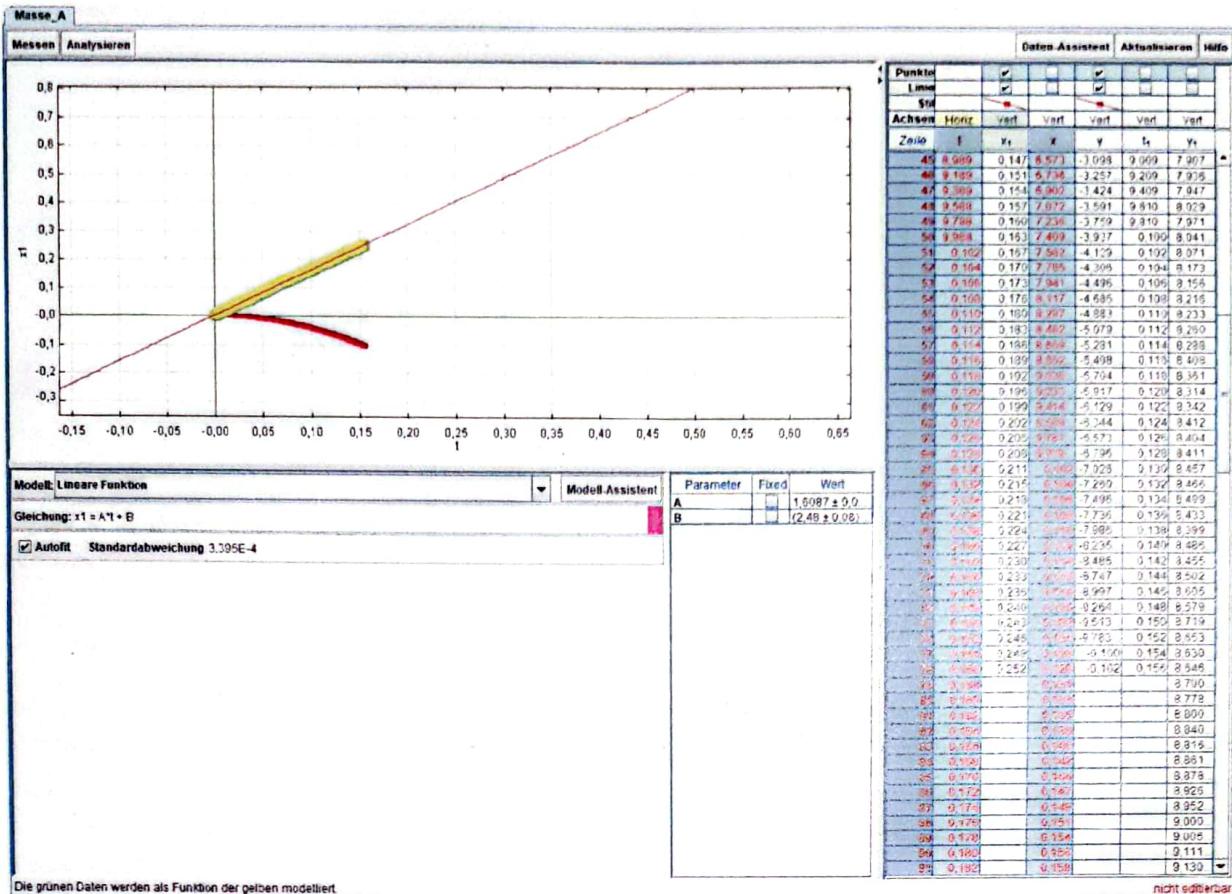
"Jonas-Alejandro-².avi"

Jonas-Alejandro-1.avi

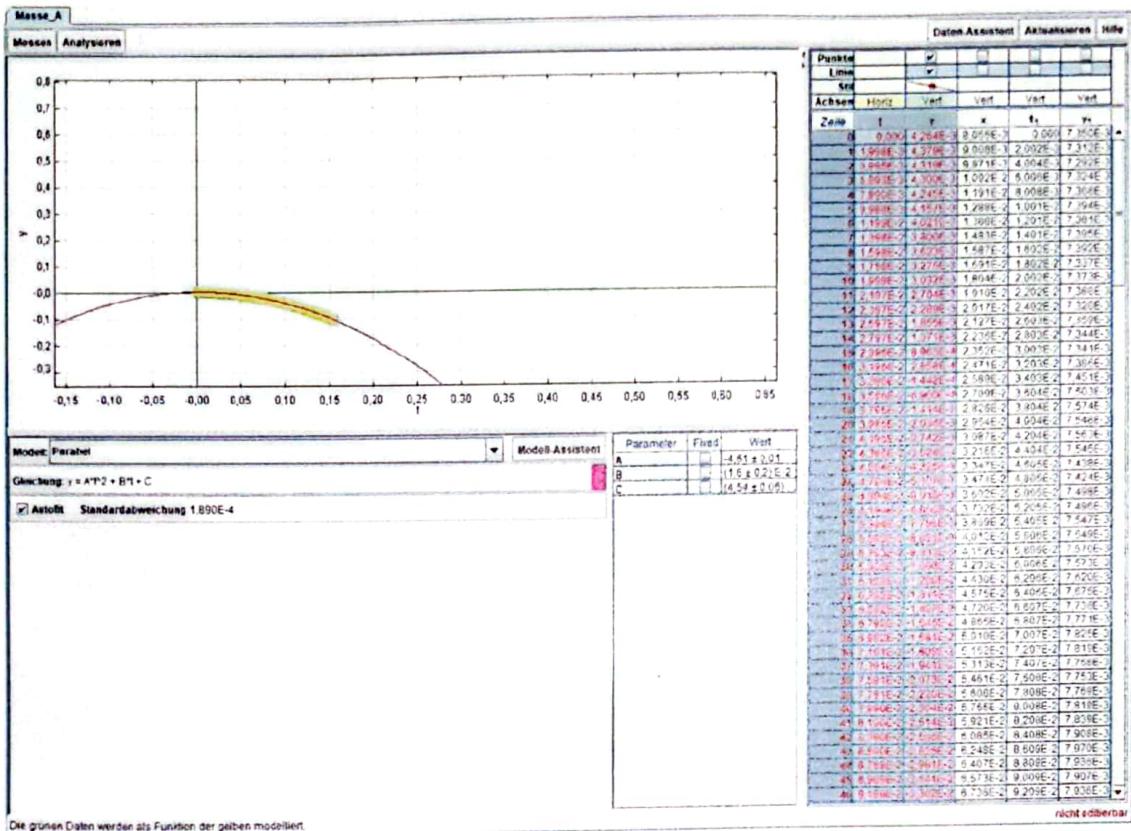


$$\frac{1}{2} \hat{a} = 1,943 \pm 0,005 \Rightarrow a = 3,886 \pm 0,01$$

$$u \hat{=} B = 4,75 \pm 0,01$$



Jonas_Alejandro_2.avi



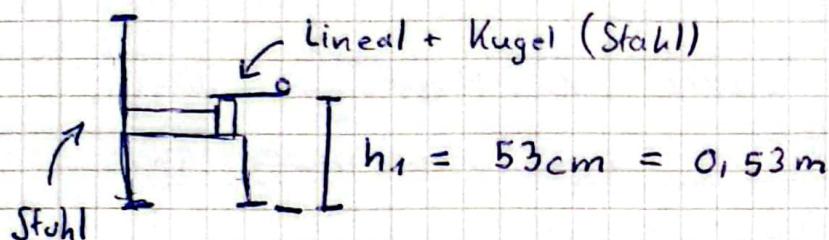
$$\frac{1}{2}g \hat{=} \text{Parameter } A = -4,61 \pm 0,01$$

$$\Rightarrow |g| = 9,22 \pm 0,02$$

Teil IV:

Folgende Höhen ziehen wir in Betracht:

Höhe 1:



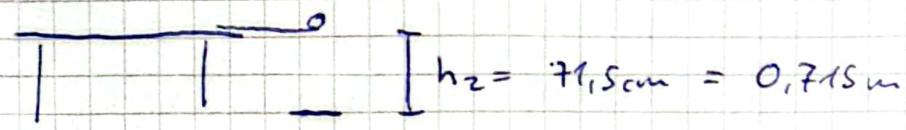
Inkl. Fehler: $0,53\text{m} \pm 0,003\text{m}$

Wir führen ~~8~~¹⁰ Messungen durch:

Anschlag mit Stift 1	Versuch 1	Fallzeit in s	Bemerkung (ggf.)	Schwelle: 0,1a.u.
	Versuch 2	0,104		Mind. Verzögerung: 0,18
	Versuch 3	0,346	Kuli wurde lang mitgeführt am Lineal	Hypothese
	Versuch 4	0,133		
	Versuch 5	0,334		
	Versuch 6	0,3309		
	Versuch 7	0,355		
	Versuch 8	0,340		
	Versuch 9	0,345		
	Versuch 10	0,342		

LMU München
Physikalische Praktika
Vor. Ich: <u>STO</u>
Datum: <u>23.02</u>
Betreuer: <u>Alexander Proschwitz</u>
Vorin

Höhe 2:



↑

Tisch

Inkl. Fehler: $0,715 \text{ m} \pm 0,003 \text{ m}$

Wir führen wieder 10 Messungen durch:

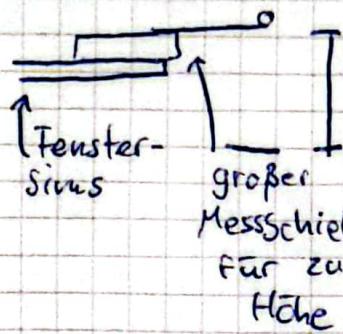
	Fallzeit in s	Bemerkung (ggf.)
Versuch 1	0,400	
Versuch 2	0,396	
Versuch 3	0,393	
Versuch 4	0,396	
Versuch 5	0,396	
Versuch 6	0,394	
Versuch 7	0,398	
Versuch 8	0,401	
Versuch 9	0,391	
Versuch 10	0,387	

Wir passen · Schwelle an da bei 0,1 a.u.

der Obere Stop nicht registriert wird.

Mit Schwelle 0,03 a.u. wird der obere Schlag registriert.

Höhe 3:



$$h_3 = 106,5 \text{ cm} = 1,06 \text{ m}$$

inkl. Fehler: $1,06 \pm 0,003 \text{ m}$

Wir führen 10 Messungen durch:

Versuch	Fallzeit in s	Bemerkung (ggf.)
Versuch 1	0,479	
Versuch 2	0,4775 0,476	
Versuch 3	0,421	
Versuch 4	0,477	
Versuch 5	0,489	
Versuch 6	0,477	
Versuch 7	0,469	
Versuch 8	0,478	
Versuch 9	0,476	
Versuch 10	0,483	

Schwelle bei 0,02 a. U.

Kapitel 2

Auswertung

2.1 Teilversuch I: Flugweiten verschiedener Kugeln

2.2 Teilversuch II: Elastischer Stoß von Kugeln gleicher Masse

2.2.1 Theoretischer Hintergrund

Im idealisierten Fall des Versuchsaufbaus liegen die Auftreffpunkte der Projektil- und Targetkugel unter Variation des Stoßparameters b jeweils auf Kreisbahnen. Der Mittelpunkt dieser Kreisbahnen liegt bei $\vec{M} = \vec{O} + \frac{1}{2}\vec{s}_{zentral}$, wobei $\vec{s}_{zentral}$ die Strecke zwischen Stoßpunkt \vec{O} und Auftreffpunkt der Targetkugel beim Zentralen Stoß beschreibt. Der Radius r_i dieser Kreisbahnen ist umgekehrt proportional zur Masse der jeweiligen Kugel m_i . Es gilt also:

$$\frac{r_{target}}{r_{projektil}} = \frac{m_{projektil}}{m_{target}} \quad (2.1)$$

Somit folgt für $m_{projektil} = m_{target}$, dass die Auftreffpunkte der Kugeln auf dem selben Kreis liegen. In unserem Fall mit Kugeln gleicher Masse gilt zudem für den Auftreffpunkt des Flugs ohne Stoß: (*Dieser muss noch um den Abstand zwischen Rampenende und Stoßursprungspunkt O von 2,5cm vergrößert werden*)

$$\vec{s}_{flug} = \vec{s}_{zentral} \quad (2.2)$$

und damit:

$$\vec{M} = \vec{O} + \frac{1}{2}\vec{s}_{flug} \quad (2.3)$$

2.2.2 Grafische Analyse der Landepunkte

Um die Landepunkte zu visualisieren, wurden die mittleren Landepunkte der Projektil- und Targetkugel bei Variation des Stoßparameters b ausgemessen und in Abbildung 2.1 dargestellt. Hierzu wurden die jeweiligen durchschnittlichen Landepunkte sowie deren Unsicherheiten mithilfe eines Python-Scripts (matplotlib) in ein Koordinatensystem eingezeichnet. Die Kreise um die Landepunkte der Kugeln stellen diese Unsicherheiten dar, welche sich hauptsächlich aus folgenden Aspekten zusammensetzt:

- Streuung der Landepunkte über 4 Durchgänge
- Ungenauigkeit des Abtragens des Rampenendes
- Messungenauigkeit durch das manuelle Ausmessen mithilfe eines Lineals
- Ungenauigkeit beim Schätzen des jeweiligen durchschnittlichen Landepunkts

Zusätzlich wurde die theoretisch vorhergesagte Kreisbahn eingezeichnet.

Anmerkung: Da die Projektilkugel für $b = 0$ auf der Vorrichtung für die Targetkugel landet und dann zufällig abprallt, wurde die Koordinate des Projektils für $b = 0$ auf den theoretischen Wert von (0, -2) korrigiert.

Landepunkte der Kugeln

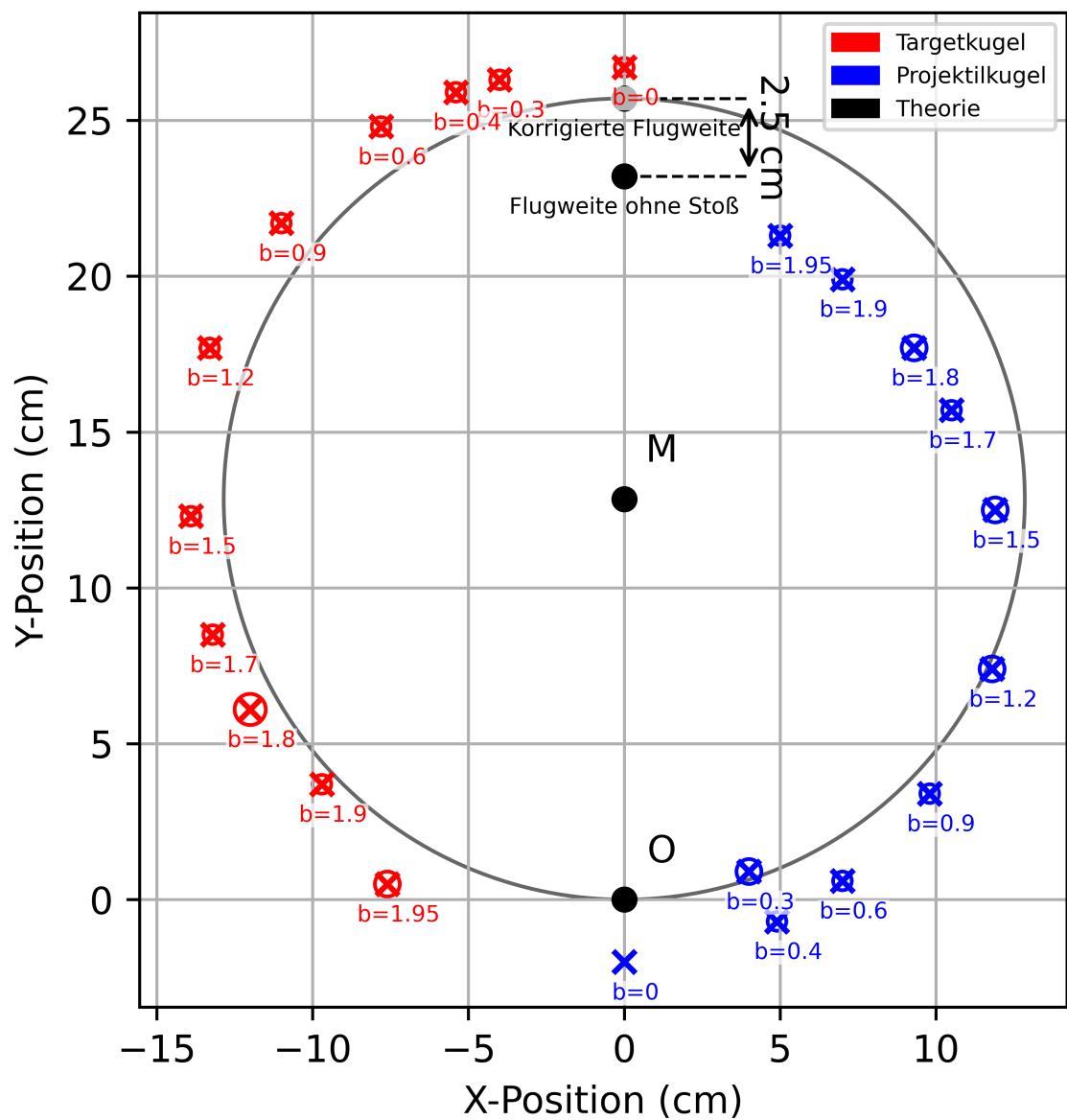


Abbildung 2.1: Landepunkte der Projektil- und Targetkugel bei Variation des Stoßparameters b

2.2.3 Abweichung von der Theorie

Es ist zu erkennen, dass die bestimmten Landepositionen der Projektil- und Targetkugel tatsächlich annähernd auf Kreisbahnen liegen. Jedoch stimmen diese Kreisbahnen nicht vollständig mit der theoretischen Vorhersage überein.

Zunächst fällt auf, dass der Landepunkt der Projektilkugel für $b = 0,3\text{cm}$ stark in y-Richtung von der Kreisbahn abweicht, die durch die anderen Messwerte bestimmt wird. Hier ist davon auszugehen, dass dieser Fehler, wie auch für $b = 0$ durch ein abprallen der Kugel von der Haltevorrichtung für die Targetkugel verursacht wurde. Physikalisch deutlich interessanter ist allerdings die generelle Abweichung der Kreisbahnen von Target- und Projektilkugel von der Theorie. Für diese Abweichung gibt es mehrere Ursachen, von welchen hier die relevantesten diskutiert werden sollen.

Abweichung durch endliche Radien der Kugeln

Aufgrund der endlichen Radien der Kugeln findet der Stoß der Kugeln nicht im Punkt O statt. Wie man mithilfe von Abbildung 2.2 erkennen kann ist der Startpunkt der Kugeln abhängig vom Stoßparameter b verschoben.

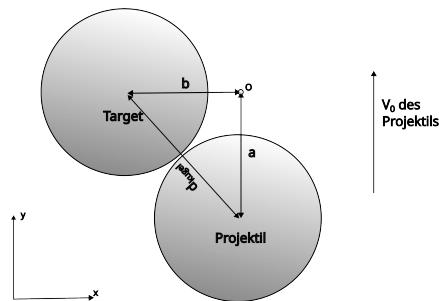


Abbildung 2.2: Abweichung der Startpunkte der Kugeln von Punkt O aufgrund der endlichen Radien der Kugeln

Entsprechend dieser Grafik müssen die gemessenen Landepunkte des Targets um b in x-Richtung und um $a = \sqrt{d_{\text{kugel}}^2 - b^2}$ in y-Richtung korrigiert werden. Diese Korrektur wurde in der folgenden Grafik 2.3 wieder mithilfe des Python-Skripts für jeden Landepunkt durchgeführt. *Anmerkung: In der Grafik wurden aus Gründen der Übersichtlichkeit nur die Mittelwerte angepasst, die Messungenauigkeit der Originalwerte gilt allerdings natürlich weiterhin.*

Landepunkte der Kugeln

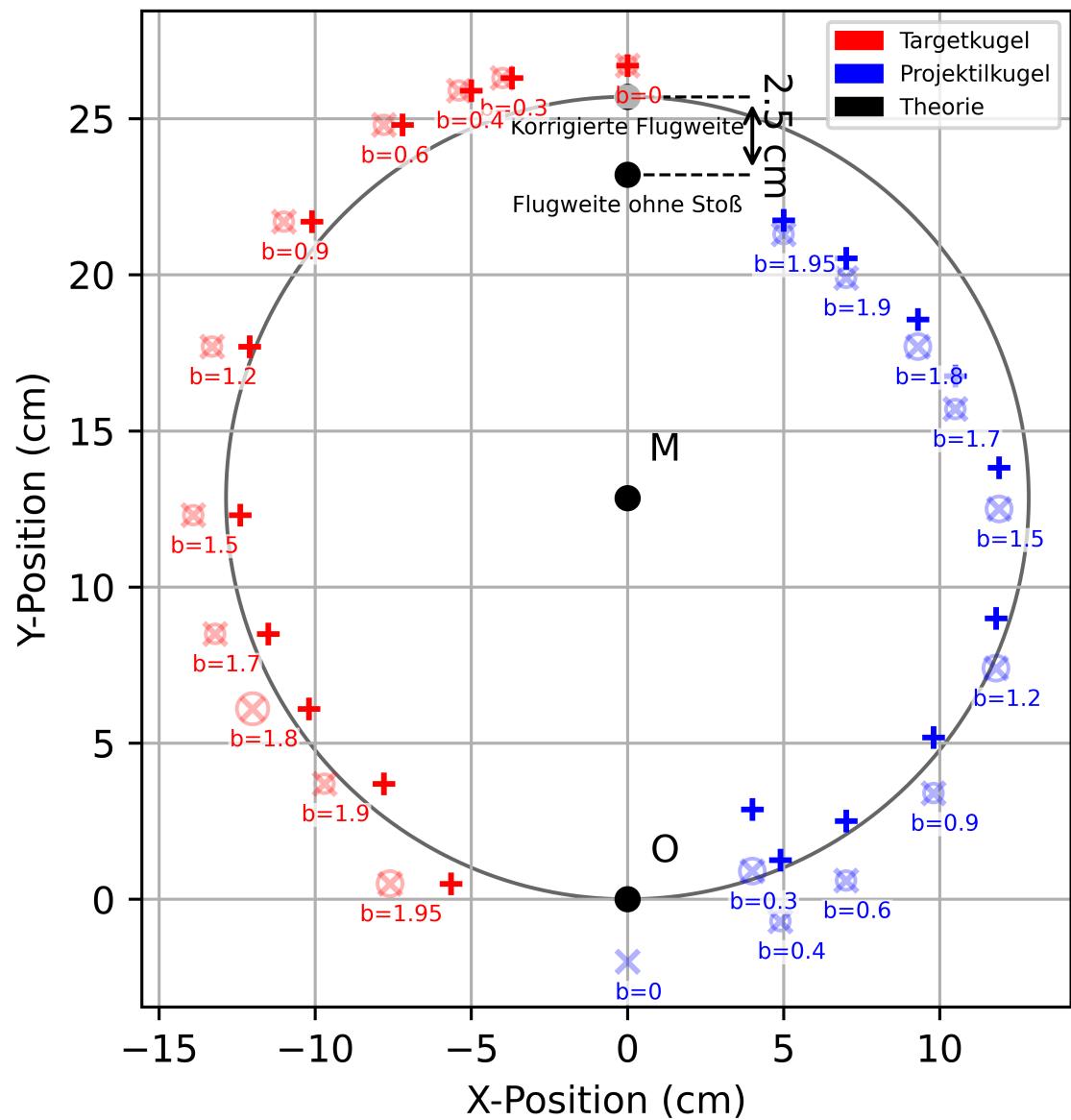


Abbildung 2.3: Abweichung der Startpunkte der Kugeln von Punkt O aufgrund der endlichen Radien der Kugeln

In der Grafik 2.3 ist zu erkennen, dass diese korrigierten Landepunkte bereits deutlich besser mit der theoretischen Vorhersage übereinstimmen. Insbesondere für große Stoßparameter b liegen die korrigierten Landepunkte der Projektilkugel allerdings immer noch relativ deutlich von der Theorie entfernt, was darauf hindeutet, dass es noch weititere Fehlerquellen gibt, die diese Abweichung verursachen. Bestimmt man die Abweichung in Abhängigkeit von b mithilfe eines weiteren Python Scripts, so ergibt sich ein Verlauf in Abhängigkeit des Stoßparameters b , wie er in Abbildung 2.4 dargestellt ist.

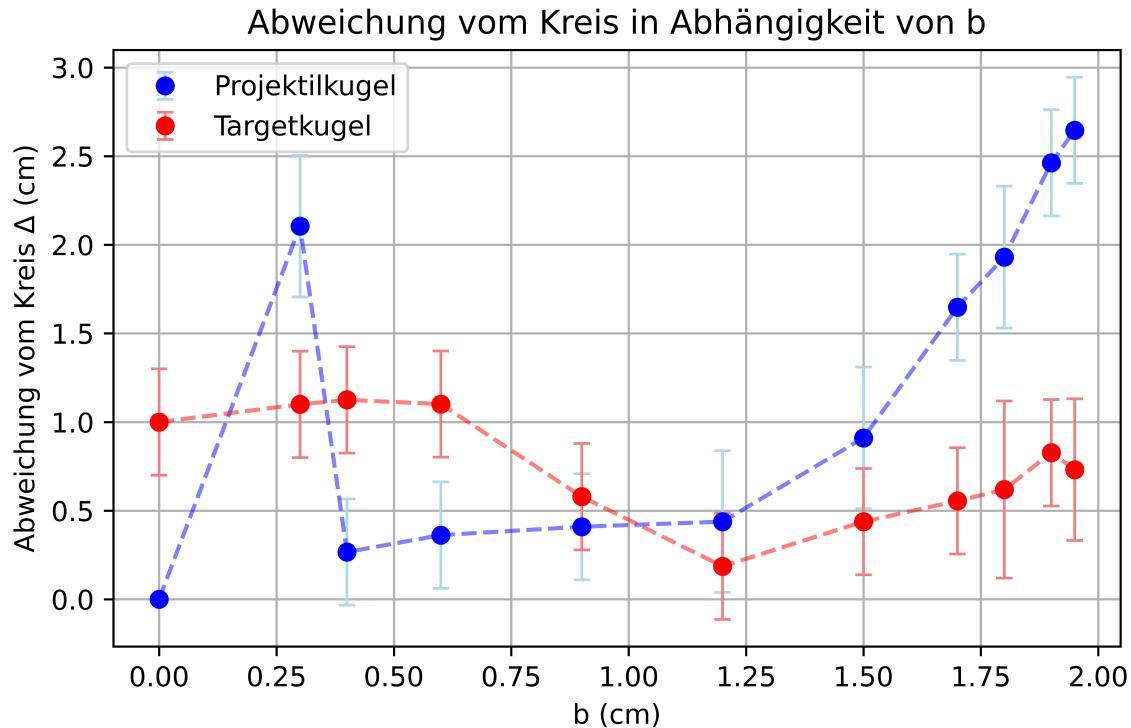


Abbildung 2.4: Abweichung der korrigierten Landepunkte der Kugeln von der theoretischen Vorhersage

Beobachtet man diesen Verlauf, so fällt auf, die Abweichung der Landepunkt der Targetkugel keine eindeutige Tendenz aufweist. Die Abweichung der Landepunkte der Projektilkugel hingegen nimmt (unter vernachlässigung des Ausreißers für $b = 0,3\text{cm}$) mit zunehmendem Stoßparameter b zu.

Abweichung durch Reibung und Abstand zwischen Rampe und Projektilhalterung

Grafik 2.4 zeigt insbesondere, dass die Abweichung der Landepunkte der Projektilkugel vom theoretischen Verlauf mit zunehmendem Stoßparameter b deutlich zunimmt. Dies liegt vermutlich unter anderem daran, dass die Gleitreibung während des Stoßes eine große Rolle spielt. Diese führt dazu, dass die Kugeln, welche mit zunehmendem Stoßparameter b länger aneinander entlang gleiten, stärker abgebremst werden und so eine geringere Flugweite erreichen.

Hinzu kommt, dass infolge der Geometrie des Versuchsaufbaus das Projektil für größere Stoßparameter eine größere Strecke zwischen Rampe und Targetkugel unter Einfluss der Erdbeschleunigung zurücklegt, bevor es die Targetkugel trifft. Dies führt dazu, dass die Targetkugel tiefer getroffen wird (Siehe Abbildungen 2.5 und 2.6), was dazu führt, dass die Projektilkugel eine größere Komponente in negative z-Richtung erhält. Hierdurch verkürzt sich die Flugweite, weshalb der Fehler mit zunehmendem Stoßparameter b zunimmt.

Diese Effekte erklären auch, weshalb die Abweichung für die Targetkugel mit zunehmendem Stoßparameter b keine eindeutige Tendenz aufweist. Zwar ist auch die Targetkugel von der Gleitreibung betroffen, was eine Verkürzung der Flugweite verursacht, allerdings wirkt für die Targetkugel der Effekt, tiefer getroffen zu werden in entgegengesetzter Richtung. Da die targetkugel durch einen tieferen Treffer eine größere Komponente in positive z-Richtung erhält, verlängert sich die Flugweite. Dadurch könnte es dazu kommen, dass die beiden Effekte sich teilweise gegenseitig aufheben, wodurch keine eindeutige Tendenz der Abweichung erkennbar wird.

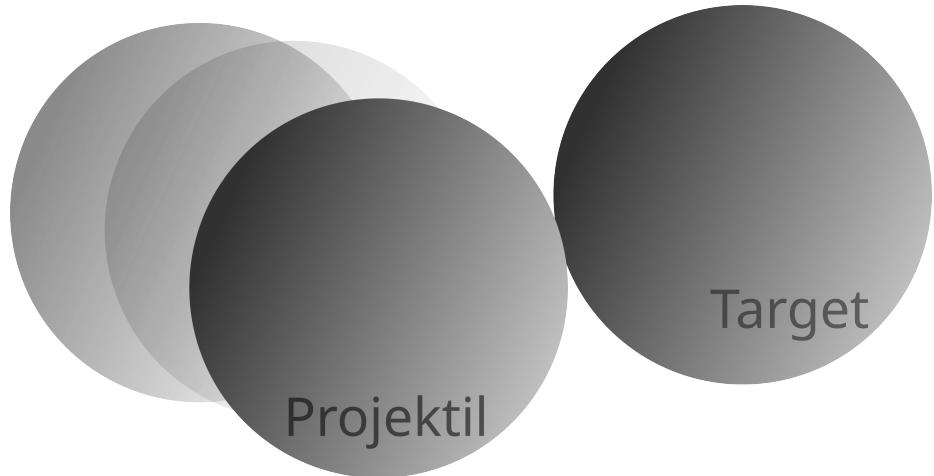


Abbildung 2.5: Stoßhöhe für $b = 0$

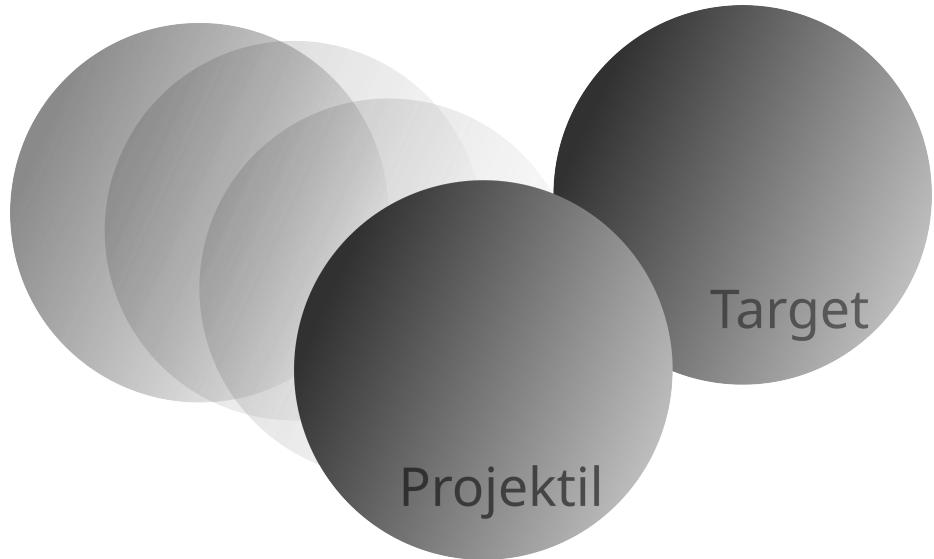


Abbildung 2.6: Stoßhöhe für $b > 0$

2.2.4 Veränderung der Landepunkte bei unterschiedlichen Massen

Wie bereits erwähnt gilt folgender Zusammenhang zwischen Massen und den Radien der Kreisbahnen der Landpunkte:

$$\frac{r_{target}}{r_{projektil}} = \frac{m_{projektil}}{m_{target}} \quad (2.4)$$

Hierraus folgt, dass die Radien der Landekreise der Kugeln bei unterschiedlichen Massen unterschiedlich groß sein müssen. Für $m_{projektil} < m_{target}$ muss der Radius des Landekreises der Projektilkugel größer sein als der Radius für die Targetkugel, da die leichtere Projektilkugel aufgrund ihrer geringeren Masse einen stärker abgelenkt wird und so eine größere Flugweite erreicht. Außerdem führt dies dazu, dass der Winkel zwischen den Geschwindigkeitsvektoren größer als 90° ist. Für $m_{projektil} > m_{target}$ muss der Radius des Landekreises der Projektilkugel kleiner sein als der Radius für die Targetkugel, da die schwerere Projektilkugel aufgrund ihrer größeren Masse weniger abgelenkt wird und so eine geringere Flugweite erreicht. Außerdem führt dies dazu, dass der Winkel zwischen den Geschwindigkeitsvektoren kleiner als 90° ist. Daraus folgen folgen die Ortskurven, die in den Abbildungen 2.7, 2.8 und 2.9 dargestellt sind.

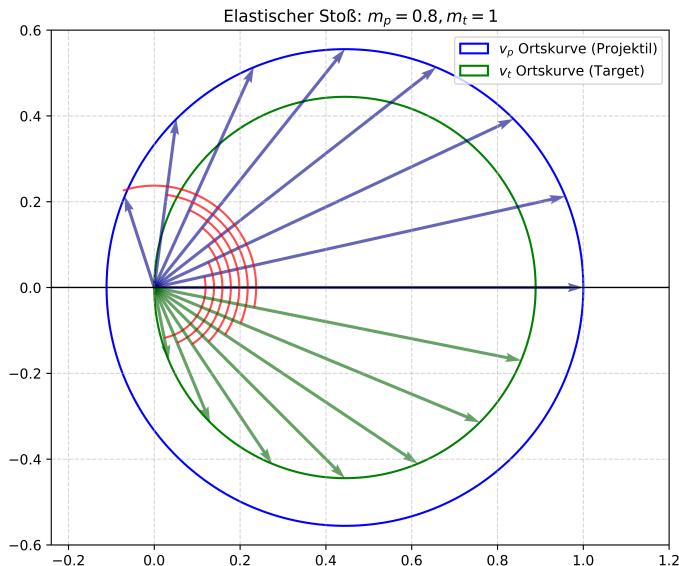


Abbildung 2.7: Ortskurve für $m_{projektil} < m_{target}$

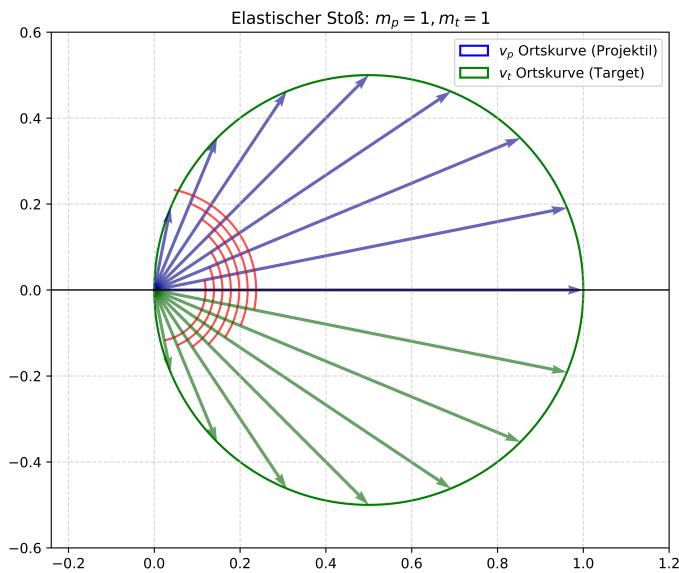


Abbildung 2.8: Ortskurve für $m_{projektil} = m_{target}$

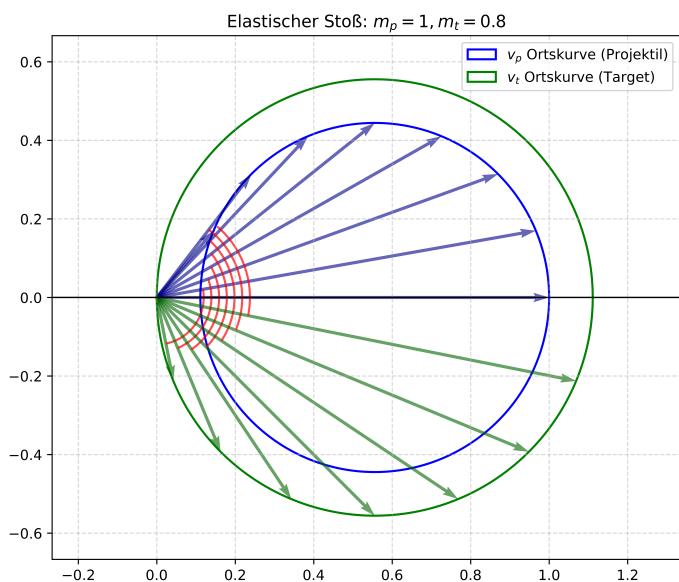


Abbildung 2.9: Ortskurve für $m_{projektil} > m_{target}$

2.3 Teilversuch III: Bewegungsanalyse mit Hochgeschwindigkeitskamera

2.4 Teilversuch IV: Bestimmung der Erdbeschleunigung

Der Versuch wurde für drei Höhen (0.530m, 0.715m, 1,065m) durchgeführt. Die Auswertung erfolgt im folgenden parallel für alle drei Höhen.

2.4.1 Theoretischer Hintergrund

Für die Bewegung im Freien Fall gilt folgende Formel:

$$y(t) = \frac{1}{2}gt^2 \quad (2.5)$$

2.4.2 Erwartete Falldauer

Um die Erwartete Falldauer zu berechnen formt man die Formel 2.5 zu folgender Form um:

$$t_{fall} = \sqrt{\frac{2h}{g}} \quad (2.6)$$

Hiermit errechnet man für die verschiedenen Höhen die mittleren Fallzeiten:

$$\begin{aligned} t^{(1)}_{fall} &= \sqrt{\frac{2h^{(1)}}{g}} = \sqrt{\frac{2 \cdot 0.530m}{g}} = 0.3288s \\ t^{(2)}_{fall} &= \sqrt{\frac{2h^{(2)}}{g}} = \sqrt{\frac{2 \cdot 0.715m}{g}} = 0.3818s \\ t^{(3)}_{fall} &= \sqrt{\frac{2h^{(3)}}{g}} = \sqrt{\frac{2 \cdot 1.065m}{g}} = 0.4660s \end{aligned}$$

Für die Unsicherheiten der Messwerte erhält man mithilfe von Linearisierung:

$$\Delta t = \frac{\partial t_{fall}}{\partial h}|_{h=h^*} \cdot \Delta h = \frac{1}{\sqrt{2gh^*}} \cdot \Delta h \quad (2.7)$$

Damit ergibt sich für die verschiedenen Höhen, die jeweils mit einer Unsicherheit von $\Delta h = \pm 3 \cdot 10^{-3}m$ gemessen wurden:

$$\begin{aligned} \Delta t^{(1)} &= \frac{3 \cdot 10^{-3}m}{\sqrt{2g \cdot 0.530m}} = 9.4 \cdot 10^{-4}s \\ \Delta t^{(2)} &= \frac{3 \cdot 10^{-3}m}{\sqrt{2g \cdot 0.715m}} = 8.1 \cdot 10^{-4}s \\ \Delta t^{(3)} &= \frac{3 \cdot 10^{-3}m}{\sqrt{2g \cdot 1.065m}} = 6.6 \cdot 10^{-4}s \end{aligned}$$

Somit sind die errechneten Fallzeiten:

$$\begin{aligned} t^{(1)}_{fall_{the}} &= (0.3288 \pm 0.0010)s \\ t^{(2)}_{fall_{the}} &= (0.3818 \pm 0.0008)s \end{aligned}$$

$$t_{fall_{the}}^{(3)} = (0.4660 \pm 0.0007)s$$

Die während des Experiments gemessenen Werte sind in der Tabelle 2.1 dokumentiert. Hierbei sind starke Ausreißerwerte, die höchstwahrscheinlich auf eine Fehlmessung zurückzuführen sind geklammert. Bei den Messwerten, die nur knapp über der eingestellten Mindestverzögerung von 0.1s liegen ist mit hoher Sicherheit davon auszugehen, dass das Startgeräusch nicht registriert wurde und dann zwei mal kurz aufeinander das Aufprallgeräusch registriert wurde.

Tabelle 2.1: Fallzeiten in Abhängigkeit von der Fallhöhe

h	$t_1(s)$	$t_2(s)$	$t_3(s)$	$t_4(s)$	$t_5(s)$	$t_6(s)$	$t_7(s)$	$t_8(s)$	$t_9(s)$	$t_{10}(s)$
0.530m	(0.160)	(0.104)	0.346	(0.133)	0.334	0.339	0.355	0.340	0.345	0.342
0.715m	0.400	0.396	0.393	0.396	0.396	0.394	0.398	0.401	0.391	0.387
1.065m	0.479	0.476	(0.421)	0.477	0.489	0.477	0.469	0.478	0.476	0.483

Berechnet man jeweils das Arithmetische Mittel der Messwerte erhält man folgende Mittelwerte:

$$\bar{t}_{fall_{mess}}^{(1)} = 0.343s$$

$$\bar{t}_{fall_{mess}}^{(2)} = 0.395s$$

$$\bar{t}_{fall_{mess}}^{(3)} = 0.478s$$

Die Messungenauigkeit kann man jeweils so abschätzen, dass ca. 2/3 der Messwerte innerhalb dieser Schranken liegen. (*Die Messungenauigkeit liegt mit dieser Abschätzung für jede der drei Messreihen bei 0.04s*)

$$t_{fall_{mess}}^{(1)} = (0.343 \pm 0.04)s$$

$$t_{fall_{mess}}^{(2)} = (0.395 \pm 0.04)s$$

$$t_{fall_{mess}}^{(3)} = (0.478 \pm 0.04)s$$

Die gemessenen Werte unterscheiden sich somit so weit von den theoretischen werten, dass

Kapitel 3

Anhang

3.1 Python Scripts

3.1.1 TV I:

3.1.2 TV II:

Landepunkte der Kugeln

```
1      import matplotlib.pyplot as plt
2      from matplotlib.patches import Circle, Patch
3      from math import sqrt
4
5      mode = "korrigiert" # "korrigiert" oder "unkorrigiert"
6
7
8
9      x_p = [0, 4, 4.9, 7, 9.8, 11.8, 11.9, 10.5, 9.3, 7, 5]
10     y_p = [-2, 0.9, -0.7, 0.6, 3.4, 7.4, 12.5, 15.7, 17.7, 19.9, 21.3]
11     unsicherheit1 = [0, 0.4, 0.3, 0.3, 0.3, 0.4, 0.4, 0.3, 0.4, 0.3, 0.3]
12
13    x_t = [0, -4, -5.4, -7.8, -11.0, -13.3, -13.9, -13.2, -12, -9.7, -7.6]
14    y_t = [26.7, 26.3, 25.9, 24.8, 21.7, 17.7, 12.3, 8.5, 6.1, 3.7, 0.5]
15    unsicherheit2 = [0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.5, 0.3, 0.4]
16
17    b = [0, 0.3, 0.4, 0.6, 0.9, 1.2, 1.5, 1.7, 1.8, 1.9, 1.95]
18    a = []
19    for i in b:
20        a.append(sqrt(2*2 - i*i))
21
22    y_p_corr = [y_p[i] + a[i] for i in range(len(y_p))]
23    x_t_corr = [x_t[i] + b[i] for i in range(len(x_t))]
24
25    fig, ax = plt.subplots()
26
27    if mode == "korrigiert":
28        ax.scatter(x_p, y_p_corr, marker='+', color='blue', zorder=3)
29        ax.scatter(x_t_corr, y_t, marker='+', color='red', zorder=3)
30        filename = "LandepunkteStossUndKorrektur.png"
31        alphaOrig = 0.3
32    else:
```

```

33     alphaOrig = 1
34     filename = "LandepunkteStoss.png"
35
36     ax.scatter(0, 0, marker='o', color='black', zorder=4, label='Ursprung')
37     ax.annotate('0', (0, 0), textcoords="offset points", xytext=(10,10), ha='center')
38     ax.scatter(0, 25.7/2, marker='o', color='black', zorder=4, label='M')
39     ax.annotate('M', (0, 25.7/2), textcoords="offset points", xytext=(10,10), ha='center')
40     ax.scatter(0, 23.2, marker='o', color='black', zorder=4, label='Flugweite ohne Stoss')
41     ax.annotate('Flugweite ohne Stoss', (0, 23.2), textcoords="offset points", xytext=(0,-10), ha='center', fontsize=6)
42     ax.scatter(0, 25.7, marker='o', color='black', zorder=4, label='Korrigierte Flugweite')
43     ax.annotate('Korrigierte Flugweite', (0, 25.7), textcoords="offset points", xytext=(0,-10), ha='center', fontsize=6)
44     kreis_m = Circle((0, 25.7/2), 25.7/2,
45                         edgecolor='black',
46                         facecolor='none',
47                         alpha=0.6,
48                         linewidth=1)
49     ax.add_patch(kreis_m)
50
51     ax.scatter(x_p, y_p, marker='x', color='blue', zorder=3, alpha=alphaOrig)
52     ax.scatter(x_t, y_t, marker='x', color='red', zorder=3, alpha=alphaOrig)
53
54
55
56     for xi, yi, r, bi in zip(x_p, y_p, unsicherheit1, b):
57         if r > 0: # keine Kreise bei Radius 0
58             kreis = Circle((xi, yi), r,
59                             edgecolor='blue',
60                             facecolor='none',
61                             alpha=alphaOrig,
62                             linewidth=1)
63             ax.add_patch(kreis)
64             ax.annotate(f"b={bi}",
65                         (xi, yi),
66                         backgroundcolor='white',
67                         bbox=dict(
68                             facecolor='white',
69                             alpha=0.7,
70                             edgecolor='none',
71                             boxstyle='square', pad=0.05,
72                         ),
73                         zorder=5,
74                         textcoords="offset points",
75                         xytext=(3, -10),
76                         fontsize=6,
77                         color='blue',
78                         ha = 'center')
79     for xi, yi, r, bi in zip(x_t, y_t, unsicherheit2, b):
80         kreis = Circle((xi, yi), r,
81                         edgecolor='red',
82                         facecolor='none',
83                         alpha=alphaOrig,
84                         linewidth=1)
85         ax.add_patch(kreis)
86         ax.annotate(f"b={bi}",
87                     (xi, yi),

```

```

88         backgroundcolor='white',
89         bbox=dict(
90             facecolor='white',
91             alpha=0.7,
92             edgecolor='none',
93             boxstyle='square,pad=0.05',
94         ),
95         zorder=5,
96         textcoords="offset points",
97         xytext=(3, -10),
98         fontsize=6,
99         color='red',
100        ha = 'center')
101
102
103     y1 = 23.2
104     y2 = 25.7
105     x_dim = 4
106
107     ax.annotate(
108         "n",
109         xy=(x_dim, y1),
110         xytext=(x_dim, y2),
111         arrowprops=dict(arrowstyle="<->")
112     )
113
114     ax.plot([0, x_dim], [y1, y1], color='black', linewidth=0.8, linestyle='dashed')
115     ax.plot([0, x_dim], [y2, y2], color='black', linewidth=0.8, linestyle='dashed')
116
117     ax.text(
118         x_dim + 1.3,
119         (y1 + y2) / 2,
120         f"{{y2 - y1:.1f}} cm",
121         rotation=-90,
122         va="center",
123         ha="right"
124     )
125
126     ax.set_xlabel("X-Position (cm)")
127     ax.set_ylabel("Y-Position (cm)")
128     ax.set_title("Landepunkte der Kugeln")
129
130     ax.set_aspect('equal', 'box')
131
132     ax.grid(True)
133
134     red_patch = Patch(color='red', label='Targetkugel')
135     blue_patch = Patch(color='blue', label='Projektilkugel')
136     plt.legend(handles=[red_patch, blue_patch, Patch(color='black', label='Theorie')],
137                 loc='upper right', fontsize=6)
138
139     plt.savefig(f"./01_ST0/Images/{filename}", dpi=1000, bbox_inches='tight')

```

Abweichung der Landepositionen

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from math import sqrt
4
5 x_p = [0, 4, 4.9, 7, 9.8, 11.8, 11.9, 10.5, 9.3, 7, 5]
6 y_p = [-2, 0.9, -0.7, 0.6, 3.4, 7.4, 12.5, 15.7, 17.7, 19.9, 21.3]
7 unsicherheit1 = [0, 0.4, 0.3, 0.3, 0.3, 0.4, 0.4, 0.3, 0.4, 0.3, 0.3]
8
9 x_t = [0, -4, -5.4, -7.8, -11.0, -13.3, -13.9, -13.2, -12, -9.7, -7.6]
10 y_t = [26.7, 26.3, 25.9, 24.8, 21.7, 17.7, 12.3, 8.5, 6.1, 3.7, 0.5]
11 unsicherheit2 = [0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.5, 0.3, 0.4]
12
13 b = [0, 0.3, 0.4, 0.6, 0.9, 1.2, 1.5, 1.7, 1.8, 1.9, 1.95]
14 a = [sqrt(4 - i**2) for i in b]
15
16 y_p_corr = [y_p[i] + a[i] for i in range(len(y_p))]
17 x_t_corr = [x_t[i] + b[i] for i in range(len(x_t))]
18
19 xc, yc = 0, 25.7/2
20 R = 25.7/2
21
22 x_proj = np.array(x_p)
23 y_proj = np.array(y_p_corr)
24 b_proj = np.array(b)
25 uns_proj = np.array(unsicherheit1)
26
27 x targ = np.array(x_t_corr)
28 y targ = np.array(y_t)
29 b targ = np.array(b)
30 uns targ = np.array(unsicherheit2)
31
32 dev_proj = np.abs(np.sqrt((x_proj - xc)**2 + (y_proj - yc)**2) - R)
33 dev targ = np.abs(np.sqrt((x targ - xc)**2 + (y targ - yc)**2) - R)
34
35 fig2, ax2 = plt.subplots(figsize=(6,4))
36
37 ax2.errorbar(b_proj, dev_proj, yerr=uns_proj, fmt='o', color='blue',
38               ecolor='lightblue', elinewidth=1, capsized=3, label='Projektilkugel')
39
40 ax2.errorbar(b targ, dev targ, yerr=uns targ, fmt='o', color='red',
41               ecolor='lightcoral', elinewidth=1, capsized=3, label='Targetkugel')
42
43 ax2.plot(b_proj, dev_proj, color='blue', linestyle='--', alpha=0.5)
44 ax2.plot(b targ, dev targ, color='red', linestyle='--', alpha=0.5)
45
46 ax2.set_xlabel("b (cm)")
47 ax2.set_ylabel("Abweichung vom Kreis (cm)")
48 ax2.set_title("Abweichung vom Kreis in Abhaengigkeit von b")
49 ax2.grid(True)
50 ax2.legend()
51
52 plt.tight_layout()
53 plt.savefig("./01_ST0/Images/AbweichungenKorrigierteLandepunkte.png", dpi=1000,
54 bbox_inches='tight')
55 plt.show()
```

Ortskurve verschiedene Massen

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from matplotlib.patches import Arc
4
5
6 def plot_collision_diagram(mp, mt, v0=1.0):
7     vs = v0 * (mp / (mp + mt))
8
9     up = v0 * (mt / (mp + mt))
10    ut = vs
11
12    Rp = up
13    Rt = ut
14
15    fig, ax = plt.subplots(figsize=(8, 8))
16
17    circle_p = plt.Circle((vs, 0), Rp, color='blue', fill=False, label='$v_p$ Ortskurve (Projektil)', linewidth=1.5)
18    circle_t = plt.Circle((vs, 0), Rt, color='green', fill=False, label='$v_t$ Ortskurve (Target)', linewidth=1.5)
19    ax.add_artist(circle_p)
20    ax.add_artist(circle_t)
21
22    angles = np.linspace(0, np.pi, 8, endpoint=False)
23
24    for theta in angles:
25        vpx = vs + Rp * np.cos(theta)
26        vpy = Rp * np.sin(theta)
27        ax.quiver(0, 0, vpx, vpy, angles='xy', scale_units='xy', scale=1,
28                   alpha=0.6, width=0.005, color='darkblue')
29
30        vtx = vs + Rt * np.cos(theta + np.pi)
31        vty = Rt * np.sin(theta + np.pi)
32        ax.quiver(0, 0, vtx, vty, angles='xy', scale_units='xy', scale=1,
33                   alpha=0.6, width=0.005, color='darkgreen')
34
35        arc_radius = 0.2 + (theta * 0.1)
36
37        angle_p = np.degrees(np.arctan2(vpy, vpx))
38        angle_t = np.degrees(np.arctan2(vty, vtx))
39
40        arc = Arc((0, 0), arc_radius, arc_radius, angle=0,
41                   theta1=angle_t, theta2=angle_p, color='red', lw=1.5, alpha=0.7)
42        if theta > 0:
43            ax.add_patch(arc)
44
45        limit = max(vs + Rp, vs + Rt) * 1.2
46        ax.set_xlim(-limit * 0.2, limit)
47        ax.set_ylim(-limit/2, limit/2)
48        ax.axhline(0, color='black', lw=1)
49        ax.set_aspect('equal')
50        ax.set_title(f'Elastischer Stoß: $m_p={mp}, m_t={mt}$')
51        ax.legend(loc='upper right')
52
53        plt.grid(True, linestyle='--', alpha=0.5)
54
55        if mp < mt:
```

```

57     relation = '<'
58 elif mp > mt:
59     relation = '>'
60 else:
61     relation = '='
62
63 plt.savefig(f'Ortskurve_m_p_{relation}_m_t.png', dpi=1000, bbox_inches='tight')
64
65
66 plot_collision_diagram(mp=0.8, mt=1)
67 plot_collision_diagram(mp=1, mt=1)
68 plot_collision_diagram(mp=1, mt=0.8)

```

3.1.3 TV III:

3.1.4 TV IV:

3.2 Verwendete Daten

3.2.1 TV I:

3.2.2 TV II:

Die folgenden Daten sind alle in cm angegeben.

Stossparameter b	Datensatz Projektil				Datensatz Target			
	x_p	y_p	y_p^{corr}	Δr_p	x_t	x_t^{corr}	y_t	Δr_t
0.00	0.0	-2.0	0.0	0.0	0.0	0.0	26.7	0.3
0.30	4.0	0.9	2.9	0.4	-4.0	-3.7	26.3	0.3
0.40	4.9	-0.7	1.3	0.3	-5.4	-5.0	25.9	0.3
0.60	7.0	0.6	2.5	0.3	-7.8	-7.2	24.8	0.3
0.90	9.8	3.4	5.2	0.3	-11.0	-10.1	21.7	0.3
1.20	11.8	7.4	9.0	0.4	-13.3	-12.1	17.7	0.3
1.50	11.9	12.5	13.8	0.4	-13.9	-12.4	12.3	0.3
1.70	10.5	15.7	16.8	0.3	-13.2	-11.5	8.5	0.3
1.80	9.3	17.7	18.6	0.4	-12.0	-10.2	6.1	0.5
1.90	7.0	19.9	20.5	0.3	-9.7	-7.8	3.7	0.3
1.95	5.0	21.3	21.7	0.3	-7.6	-5.7	0.5	0.4

3.2.3 TV III:

3.2.4 TV IV: