



# PE SEMESTERPROJEKT

Bericht

Stutz Aline, Isliker Pascal, Scherrer Marcel  
ZHAW

## Inhaltsverzeichnis

1	Abstract.....	2
2	Teil 1.....	2
3	Teil 2: Würfel wird gestossen.....	3
3.1	Aufgabenstellung .....	3
3.2	Unser Experiment .....	4
3.3	Theorie .....	5
3.4	Berechnung der Energien .....	5
3.5	Vergleich mit Experiment .....	6
4	Teil 3a: Würfel stossen verzögert elastisch.....	7
5	Teil 3b: Würfel 1 wird gefangen.....	9
6	Teil 3c: Würfel 2 fällt.....	10
7	Rückblick .....	11
8	Abbildungsverzeichnis .....	11
9	Anhang .....	12

## 1 Abstract

Dieser Bericht beinhaltet die gewonnenen Erkenntnisse und Resultate der Experimente gemäss der Aufgabenstellung „PE\_Semesterprojekt\_FS22\_v08.pdf“. Die Experimente wurden dabei mithilfe von Unity simuliert.

## 2 Einleitung

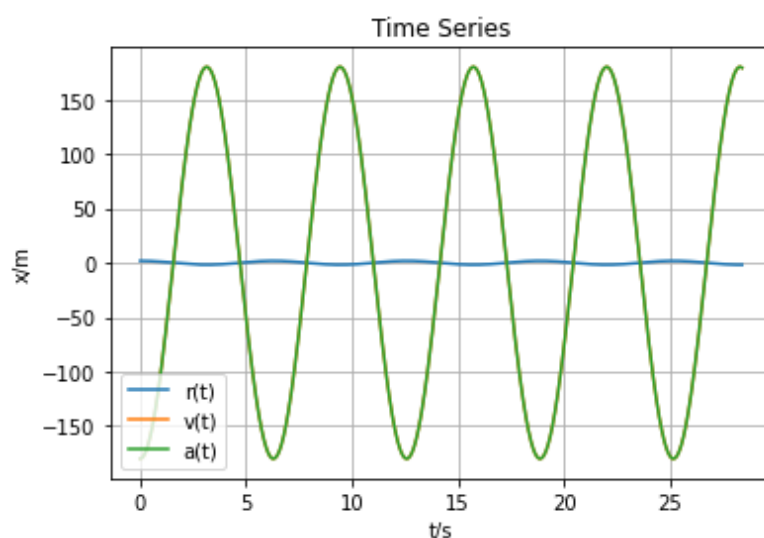
In diesem Bericht werden die gewonnenen Erkenntnisse aus den Experimenten gemäss der Anleitung Semesterprojekt PE FS 2022 dokumentiert. Dabei werden folgende Physikthemen behandelt:

- Harmonischer Oszillator
- Federkraft
- Kinetische Energie
- Federenergie
- Energieerhaltungsgesetz
- Impulsgesetz
- Elastischer Stoss
- Trockene Reibung
- Winkelgeschwindigkeit
- Arbeit
- Freier Fall

Für die Visualisierung der Experimente wird Unity verwendet. Zur Darstellung der Grafiken der erhaltenen Daten des Experimentes wird ein Python Skript verwendet.

## 3 Teil 1: Harmonischer Oszillators

Es soll ein Würfel erstellt werden, welcher eine Oszillation erfährt. Dieser soll dann eine Harmonische Schwingung widerspiegeln. In der unterstehenden Graphik, kann erkannt werden, dass die Reibung 0 beträgt und der Würfel eine gleiche Beschleunigung und Geschwindigkeit aufweist.



## 4 Teil 2: Würfel wird gestossen

### 4.1 Aufgabenstellung

Ein Würfel  $W_1$  ( $m = 1 \text{ kg}$ ) wird durch eine gespannte Feder von einer unendlich schweren Mauer  $M_1$  ( $m = \infty$ ) weggestossen. Dabei soll der Würfel, nachdem er abgestossen wurde, eine Geschwindigkeit ( $\vec{v} = 1 \text{ m/s}$ ) erreichen. Um diese Bedingungen zu erfüllen, müssen die Federkonstante  $k$  und die Kompression  $x$  entsprechend gewählt werden.

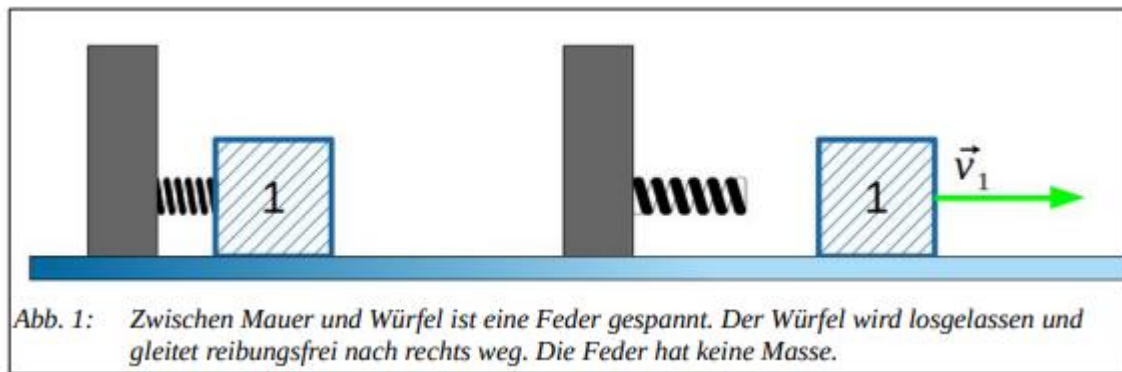


Abbildung 1 Ausschnitt der Aufgabe 2 aus der Aufgabenstellung

## 4.2 Unser Experiment

### Beschreibung

Nach dem Start des Programmes wird der Würfel auf eine Geschwindigkeit  $v = 1 \text{ m/s}$  in Richtung der Wand beschleunigt. An der Position  $x = 0$  trifft der Würfel auf die Feder. Dabei wird die Feder komprimiert und die Geschwindigkeit des Würfels verlangsamt sich, bis die maximale Kompression bei  $x = -1$ , erreicht wird. Ab diesem Zeitpunkt beschleunigt der Würfel in die entgegengesetzte Richtung (weg von der Wand). Die Beschleunigung erhöht sich bis der Würfel die Feder nicht mehr berührt. Die Geschwindigkeit beträgt ab diesem Zeitpunkt wieder konstant  $v = 1 \text{ m/s}$ .

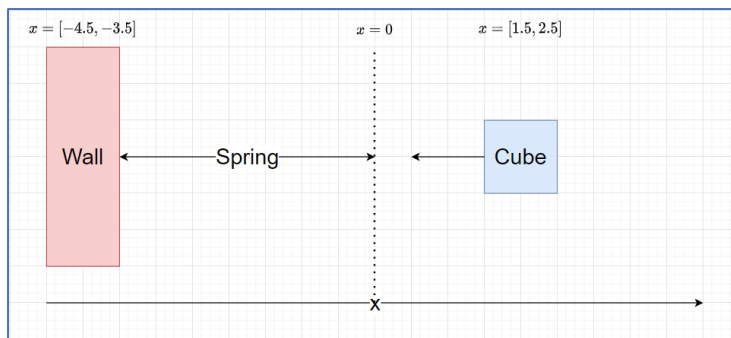


Abbildung 2: Top-Down Ansicht des Experimentes

### Umsetzung in Unity

Um dem Würfel eine initiale Geschwindigkeit von  $v = 1 \text{ m/s}$  zu geben, wendeten wir einmalig eine Kraft von 50 Newton auf den Würfel an. Dieser Kraftaufwand ist nötig, um den Würfel auf eine Geschwindigkeit  $1 \text{ m/s}$  zu beschleunigen, da in Unity ein einziger Tick ( $0.02 \text{ s}$ ) beträgt.

Nach der initialen Beschleunigung, bewegt sich der Würfel auf die Feder ( $x = 0$ ) mit konstanter Geschwindigkeit von  $1 \text{ m/s}$  zu. Sobald der Würfel auf die Feder trifft, können wir die Formel der Federkraft ( $\vec{F} = -kx$ ) anwenden. Die Feder wird maximal komprimiert und stösst den Würfel ab. Die Feder wächst anschliessend wieder auf ihre ursprüngliche Grösse an.

### Graphische Darstellung des Experiments

Das nachfolgende Diagramm wurde basierend auf den Daten des Unity-Experiments generiert. Das Resultat entspricht unserer Erwartung. Die Geschwindigkeit nimmt zuerst ab und anschliessend nimmt sie wieder zu, bis eine Geschwindigkeit von  $1 \text{ m/s}$  erreicht wird. Ab diesem Zeitpunkt bleibt die Geschwindigkeit konstant.

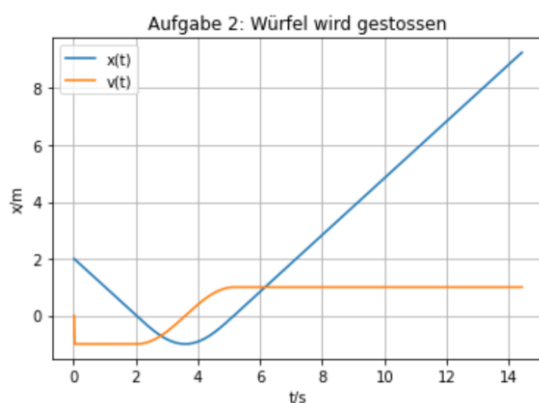


Abbildung 3: Orts- und Geschwindigkeit Diagramm

### 4.3 Theorie

Für die Umsetzung des Experimentes sind Kenntnisse einiger physikalische Formeln nötig, auf die wir in diesem Abschnitt genauer eingehen wollen.

#### Federkraft

Die Federkraft beträgt  $\vec{F} = -kx = ma$ . Sie zeigt entgegen der Auslenkung ( $-kx$ ). Die Beschleunigung  $a$  ist proportional zum Ort  $x$ .

#### Kinetische-Energie und Feder-Energie

Bei unserem Experiment entsteht sowohl Kinetische- als auch Feder-Energie. Diese Energien sind durch folgende Formeln definiert.

$$E_{spring} = \frac{1}{2} \cdot kx^2, \quad E_{kin} = \frac{1}{2} \cdot mv^2$$

#### Energie-Erhaltungssatz

Da es sich bei unserem Versuch um ein abgeschlossenes System handelt, kommt der Energie-Erhaltungssatz zum Zuge. Dieser besagt, dass sich die Energie in einem abgeschlossenen System mit der Zeit nicht verändert. In unserem Fall gibt es kinetische-  $E_{kin}$  und Feder-Energie  $E_{spring}$ , diese müssen zusammen konstant bleiben. Daraus lässt sich folgende Formel ableiten.

$$E_{spring} + E_{kin} = E_{tot}, \quad E_{tot} = \text{konstant}$$

### 4.4 Berechnung der Energien

In diesem Bereich werden wir die Energien, mithilfe der vorhergehenden Formel berechnen.

#### Vor Kompression ( $x = 0.5$ )

$$E_{kin} = \frac{1}{2} \cdot m \cdot v^2 = \frac{1}{2} \cdot 1 \cdot 1^2 = 0.5 \text{ J}, \quad E_{spring} = \frac{1}{2} \cdot k \cdot x^2 = \frac{1}{2} \cdot 1 \cdot 0^2 = 0 \text{ J}$$

$$E_{kin} + E_{spring} = E_{tot} = 0.5 + 0 = 0.5 \text{ J}$$

#### Während Kompression ( $x = -0.4$ )

Berechnung der Feder-Energie:

$$-E_{spring} = \frac{1}{2} \cdot k \cdot x^2 = \frac{1}{2} \cdot 1 \cdot (-0.4)^2 = 0.08 \text{ J}$$

Berechnung der Geschwindigkeit:

$$E_{tot} - E_{spring} = \frac{1}{2} \cdot m \cdot v^2 \rightarrow v = \sqrt{2 \cdot \frac{E_{tot} - E_{spring}}{m}} = \sqrt{2 \cdot \frac{0.42}{1}} = 0.917 \text{ m/s}$$

#### Maximale Kompression ( $x = -1$ )

$$E_{kin} = \frac{1}{2} \cdot m \cdot v^2 = \frac{1}{2} \cdot 1 \cdot 0^2 = 0 \text{ J}, \quad E_{spring} = \frac{1}{2} \cdot k \cdot x^2 = \frac{1}{2} \cdot 1 \cdot (-1)^2 = 0.5 \text{ J}$$

$$E_{kin} + E_{spring} = E_{tot} = 0 + 0.5 = 0.5 \text{ J}$$

#### 4.5 Vergleich mit Experiment

Vergleichen wir die berechnete Geschwindigkeit von  $v = 0.917 \text{ m/s}$ , welche während der Kompression besteht, mit der Geschwindigkeit in unseren Experiment, so stellen wir fest, dass diese an der Position  $x = -0.407$  ebenfalls eine Geschwindigkeit  $v = 0.917 \text{ m/s}$  aufweist.

## 5 Teil 3a: Würfel stossen verzögert elastisch

### 5.1 Beschreibung

Würfel 1 stösst mit Würfel 2 zusammen.

- Würfel 1 startet mit einer Geschwindigkeit  $v_1 = \dots$
- Würfel 2 startet mit einer Geschwindigkeit von  $v_0 = 0 \text{ m/s}$

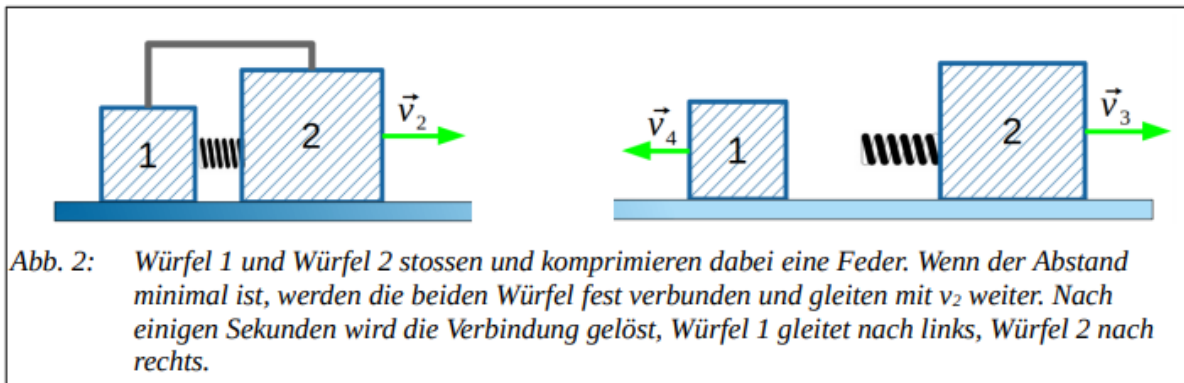
Zwischen den Würfeln wird eine Feder komprimiert.

Wenn die Feder maximal komprimiert ist, werden die beiden Würfel fest miteinander verbunden und gleiten gemeinsam weiter.

Nach einer Strecke von 5 Meter oder nach einigen Sekunden wird die feste Verbindung gelöst, die Würfel stossen sich voneinander ab. Würfel 1 gleitet nach links, Würfel 2 nach rechts.

Eigenschaften:

- Würfel 2 ( $m_2 = 2 \text{ kg}$ ) ist doppelt so schwer wie Würfel 1 ( $m_1 = 1 \text{ kg}$ ).
- Würfel 2 ( $V_2 = 1 \text{ m}^3$ ) ist gleich gross wie Würfel 1 ( $V_1 = 1 \text{ m}^3$ ).



### 5.2 Impuls Theorie

#### Massen und Geschwindigkeitsänderungen

$$\frac{m_1}{m_2} = -\frac{\Delta v_2}{\Delta v_1}$$

#### Impuls

- Impuls eines Körpers  $\vec{p} = m \cdot \vec{v}$
- Impuls vor Stoss  $\vec{p}_{vor} = m_1 \cdot \vec{v}_1 + m_2 \cdot \vec{v}_2$
- Impuls nach Stoss  $\vec{p}_{nach} = m_1 \cdot \vec{v}'_1 + m_2 \cdot \vec{v}'_2$

#### Elastischer Stoss

$$\vec{v}'_1 = \frac{m_1 - m_2}{m_1 + m_2} \cdot \vec{v}_1 + \frac{2m_2}{m_1 + m_2} \vec{v}_2$$

$$\vec{v}'_2 = \frac{m_2 - m_1}{m_1 + m_2} \cdot \vec{v}_2 + \frac{2m_1}{m_1 + m_2} \vec{v}_1$$

### 5.3 Umsetzung in Unity



## 5.4 Berechnungen

## 6 Teil 3b: Würfel 1 wird gefangen

### 6.1 Beschreibung

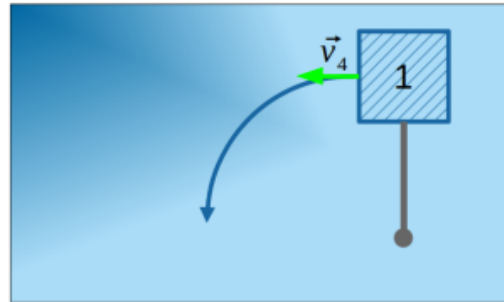
Würfel 1 gleitet (mit einer Geschwindigkeit von  $v_2'$ ) eine kurze Strecke nach links.

Dann wird ein virtuelles masseloses Seil von 5 m Länge am Schwerpunkt des Würfels befestigt.

Würfel 1 dreht in der Ebene um den Befestigungspunkt des Seils. Der Befestigungspunkt befindet sich seitlich, 5 m senkrecht zur Geschwindigkeit.

Am Boden wirkt trockene Reibung, sodass der Würfel nach einem Viertelkreis zum Stehen kommt.

*Abb. 3: Ansicht von oben. Würfel 1 kreist um den Befestigungspunkt des Seils und kommt nach einem Viertelkreis zum Stehen.*



## 6.2 Theorie

### Trockene Reibung

Trockene Reibung ist proportional zur Normalkraft

- $\mu$  = Proportionalitätsfaktor / Gleitreibungskoeffizient

$$\vec{F}_R = \mu \vec{F}_N$$

### Winkelgeschwindigkeit

$$\omega = \frac{d\phi}{dt}$$

### Arbeit

$$\text{Arbeit} = \text{Kraft} \cdot \text{Weg}, \quad W = F \cdot s$$

### Lösungsweg

$F_Z$ : Zentripetalkraft

$$\vec{F}_Z = m \cdot \frac{v^2}{R}, \quad z = (x_0 - x, y_0 - y)$$

$$\vec{F}_z = |\vec{F}_z| \cdot \vec{e}_z$$

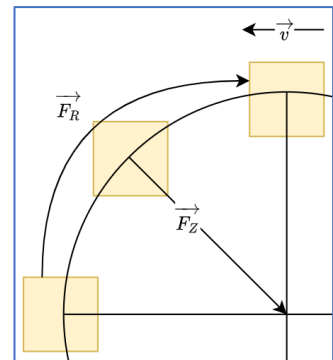
$F_R$ : Reibungskraft

$$W = F_R \cdot s = mg \cdot \mu \cdot \frac{2}{4}R = \frac{1}{2}mv^2, \quad \mu = \frac{v^2}{g\pi R}$$

$$|\vec{F}_R| = mg\mu, \quad \vec{F}_R = mg\mu \cdot -\vec{e}_z$$

$F_{res}$ : Resultierende Kraft

$$\vec{F}_{res} = \vec{F}_z + \vec{F}_R$$

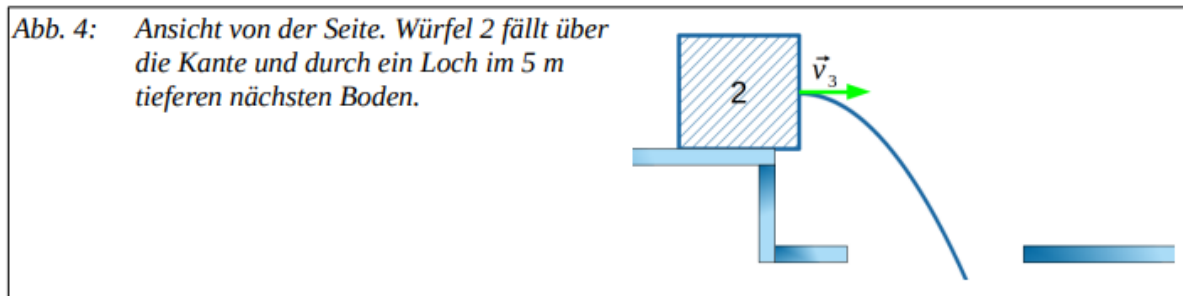


## 7 Teil 3c: Würfel 2 fällt

### 7.1 Beschreibung

Würfel 2 gleitet (mit einer Geschwindigkeit von  $v_2'$ ) auf eine Kante zu und fällt ins Leere ( $g = 9.81 \text{ m/s}^2$ ).

Weit unten, 5 m tiefer, gibt es ein Loch im Boden, durch das der Würfel fallen soll. Wählen Sie die Position des Lochs so, dass der Würfel mittig trifft.



### 7.2 Theorie

Falldistanz  $s_1$  und Fall-Geschwindigkeit  $v_1$  nach einer Zeit  $t$

$$s = s_0 + \frac{g}{2} \cdot t^2, \quad v = at$$

Horizontale Distanz  $s_2$  und Geschwindigkeit  $v_2$  nach einer Zeit  $t$

$$s_2 = s_0 + v_3 \cdot t, \quad v_2 = v_3$$

Drehmoment?

## 8 Rückblick

## 9 Abbildungsverzeichnis

Abbildung 1 Ausschnitt der Aufgabe 2 aus der Aufgabenstellung .....	3
Abbildung 2: Top-Down Ansicht des Experimentes.....	4
Abbildung 3: Orts- und Geschwindigkeit Diagramm .....	4

## 10 Anhang

### 10.1 Aufgabe 2

```
public class CubeController : MonoBehaviour
{
    private Rigidbody rigidBody;

    public int springConstant; // N/m

    private float currentTimeStep; // s

    private List<List<float>> timeSeries;

    // Start is called before the first frame update
    void Start()
    {
        rigidBody = GetComponent<Rigidbody>();
        timeSeries = new List<List<float>>();
    }

    // Update is called once per frame
    void Update()
    {
    }

    private float featherPosXFrom = 0; // m
    private double featherPosXTo = -3.5; // m
    private bool featherEnabled = false;
    private bool setInitialForce = false;

    // FixedUpdate can be called multiple times per frame
    void FixedUpdate() {
        float forceX = 0; // N

        rigidBody.mass = 1; // kg
        springConstant = 1; // N/m

        // Assuming the Feather starts at position x = 0
        float compression = rigidBody.position.x;

        //Set Initial Force before Feather is touched
        if (!setInitialForce)
        {
            //Apply Force
            forceX = -50; // N
            rigidBody.AddForce(new Vector3(forceX, 0f, 0f));
            setInitialForce = true;
        }

        //Pre-Feather
        if (rigidBody.position.x < featherPosXFrom && rigidBody.position.x >
featherPosXTo)
        {
            featherEnabled = true;
        } else
        {
            featherEnabled = false;
        }

        Debug.Log("Position: " + rigidBody.position.x
            + ", Velocity: " + rigidBody.velocity.x

```

```

        + ", Acceleration: " + forceX / rigidBody.mass);

//Feather
if (featherEnabled)
{
    //F = -x*k
    forceX = -compression * springConstant;

    //Apply Force
    rigidBody.AddForce(new Vector3(forceX, 0f, 0f));
}

//Post-Feather

currentTimeStep += Time.deltaTime;
timeSeries.Add(new List<float>() {currentTimeStep, rigidBody.position.x,
rigidBody.velocity.x, forceX });
}

void OnApplicationQuit() {
    WriteTimeSeriesToCSV();
}

void WriteTimeSeriesToCSV() {
    using (var streamWriter = new StreamWriter("time_series.csv")) {
        streamWriter.WriteLine("t,z(t),v(t),a(t) (added)");

        foreach (List<float> timeStep in timeSeries) {
            streamWriter.WriteLine(string.Join(",", timeStep));
            streamWriter.Flush();
        }
    }
}
}

```

## 10.2 Aufgabe 3