UNIVERSITÄT DES SAARLANDES

Theoretische Physik Ib

Analytische Mechanik – Tutorium 04

Bisherige Formalismen

Python-Einführung

Bisherige Formalismen

- Newton
- D'Alembert
- Lagrange I
- Lagrange II

Was ist was?

Ein paar Minuten
zum Nachdenken, Notizenwälzen,
Austausch nach rechts/links/vorne/hinten
Vgl. Notizen S. 11c (3. VL), 12a & 17 (4. VL)

Newton

- Kraftfreie Körper bewegen sich geradlinig und gleichförmig

  (lex prima / Trägheitsgesetz)

- Bewegungsgesetz:

$$\vec{a} := \frac{d\vec{v}}{dt} \Rightarrow \vec{F} = \dot{\vec{p}} = m\vec{a}$$

  (lex secunda / Bewegungsgesetz)

- actio = reactio

  (lex tertia / Reaktionsgesetz)

- Lineare Superposition von Kräften

  (lex quarta / Superpositionsgesetz)

D'Alembertsches Prinzip

$$\sum_{i=1}^{N}(m_i\ddot{\vec{r}}_i - \vec{X}_i)\cdot\delta\vec{r}_i = 0$$

- Kombination von Newtons Gesetz mit dem Prinzip der virtuellen Arbeit
- δri : virtuelle Verschiebung, erlaubt durch Zwangsbedingungen
- Zwangskräfte leisten keine virtuelle Arbeit

Lagrange Gleichung 1. Art

$$m_j \ddot{x}_j - K_j - \sum_{\mu=1}^{k} \lambda_\mu \, \Phi_j^\mu = 0 \qquad \text{mit} \qquad \Phi_j^\mu := \frac{\partial \Phi^\mu}{\partial x_j}$$

- Mit holonomen Zwangsbedingungen:

$$\Phi^\mu(x_1, ..., x_N, t) = 0 \Rightarrow \Phi_j^\mu = \frac{\partial f^\mu}{\partial x_j}$$

also

$$m_j \ddot{x}_j - K_j - \sum_{\mu=1}^{k} \lambda_\mu \frac{\partial f^\mu}{\partial x_j} = 0$$

Lagrange-Gleichung 2.Art

- Für j=1,...,fj:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} = Q_j$$

- Für konservative Kräfte: $Q_j = -\frac{\partial V}{\partial q_j}$

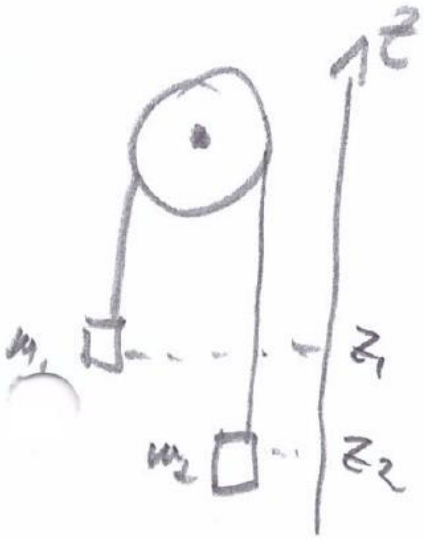
$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} = -\frac{\partial V}{\partial q_j}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial (T-V)}{\partial \dot{q}_j}\right) - \frac{\partial (T-V)}{\partial q_j} = 0$$

$$L = T - V \longrightarrow \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} = 0$$

Atwoodsche Fallmaschine

Case study Atwood's machine



(i) Newton $\underline{F} = m\underline{a}$

(ii) D'Alembert $\sum_{i=1}^{N} \underline{z}_i \cdot \delta \underline{r}_i = 0 = \sum_{i=1}^{N} (m_i \ddot{\underline{r}}_i - \underline{X}_i) \delta \underline{r}_i$

(iii) Lagrange I $m_j \ddot{x}_j - K_j - \sum_{\mu=1}^{\Lambda} \lambda_\mu \frac{\partial L_\mu}{\partial x_j} = 0$

$\Rightarrow \ddot{z}_1 = -\frac{m_1 - m_2}{m_1 + m_2} g$

(iv) Lagrange II: $\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = 0 \quad , \quad j = 1, \ldots, f$

Let's code...

Was sind eure Erfahrungen mit Coding?

One-Stop-Shop-Lösung: Anaconda

https://www.anaconda.com/download

Some Python code to get started

UNIVERSITÄT DES SAARLANDES

Philipp Hoevel, Philipp Hoevel, Saarland University

Date: **May 08, 2025**

Inspired by Hans Petter Langtangen, RIP

Why programming?

> *Everybody in this country should learn how to program a computer... because it teaches you how to think*. Steve Jobs, 1955-2011.

The learning strategy is about doing exercises

- Study and try to understand examples
- Program a lot!
- Try to solve all excercise.

This introduction 1 is about getting started

- printing text and numbers
- playing with matrices
- testing data types
- playing with turtles

TPIb_Tutorium04_coding_SS25.ipynb

Bis bald…