Prof. Dr. Katja Mombaur,
Benjamin Reh,
Silvan Lindner

Robotics 1 (WS 2018/2019)

Exercise Sheet 1

Presentation during exercises in calendar week 45

Exercise 1.1 – Rolling sphere

A solid sphere (ball) rolls down a slope that is inclined towards the vertical by $\alpha = 60\,^\circ = \frac{\pi}{3}$. The experiment starts with the ball at rest, e.g. there is no initial velocity.

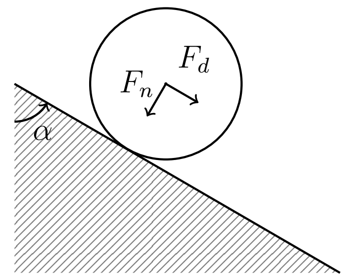
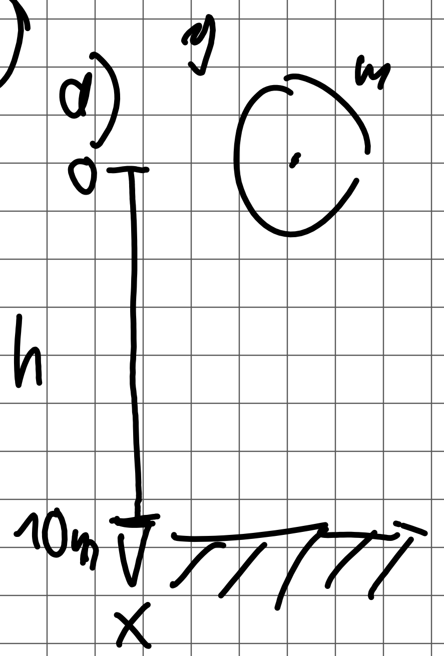


Figure 1: Sphere on slope

1. *Free fall*: First we assume the ball to be a point mass that falls freely under the influence of the gravitational acceleration $g = 9.81\,\frac{\mathrm{m}}{\mathrm{s}^2} = 9.81\,\frac{\mathrm{N}}{\mathrm{kg}}$ without the slope or any friction.
   (a) Calculate the end velocity after falling 10 m using the Conservation of Energy.
   (b) How long does the mass fall?
2. *Frictionless slinding*: The ball is still assumed to be a point mass but slides without friction down the slope.
   (a) Determine the force F_d that drives the mass down the slope using trigonometry. Resolve the gravitational force F_g into a normal force F_n that is orthogonal to the slope and a downhill force F_d that acts parallel to the slope.
   (b) Calculate the end velocity after sliding down on the slope for 10 m.
   (c) How long does the mass slide?
3. *Rolling sphere:* The ball is solid with a constant density rolling down the slope. There is no friction and the ball does not slide.
   (a) Deduce the rolling condition: Find the relation between the rotation angle φ and the translation along the slope s. Note that the sphere performs a full rotation while it rolls the length of a full circumference. Determine the time derivative of the rolling condition to get the relation between ω and v.
   (b) Calculate the end velocity for the sphere after rolling down on the slope for 10 m. Use the Conservation of Energy.

1)

a)



$$E_{pot} = E_{kin}$$

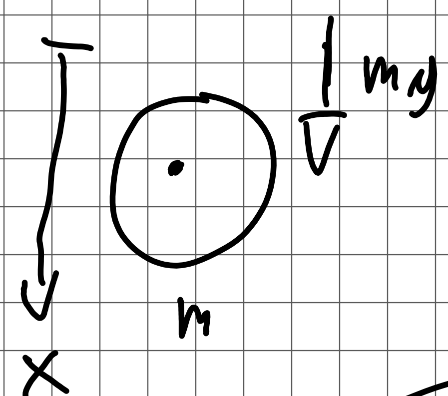
$$\cancel{m} g h = \frac{1}{2} \cancel{m} v^2$$

$$v = \sqrt{2 g h_{10}}$$

oder: $v(t) = g t$

$$v(t_{10}) = g \sqrt{\frac{2 h_{10}}{g}} = \sqrt{2}\sqrt{g}\sqrt{h_{10}}$$

b)



Newton III:

$$\sum F_x = m \ddot{x}$$

$$\cancel{m}\ddot{x} = \cancel{m} g$$

$$\frac{dx}{dt^2} = g \quad \Big| \int\int dt$$

$$x = \frac{1}{2} g t^2 = h_{10}$$

$$\frac{2 h_{10}}{g} = t^2 \quad \Big| \sqrt{\ }$$

$$t_{10} = \sqrt{\frac{2 h_{10}}{g}}, \quad h = 10\,m$$

alternativ:

$$\frac{dV}{dt} = g \quad \Big| \int dt$$

$V = g t$ siehe a)

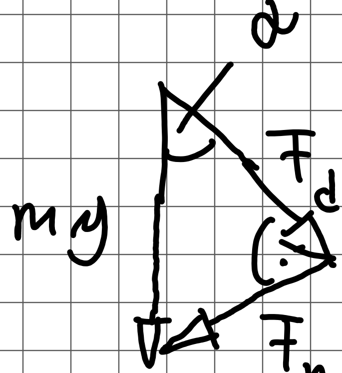
$$t_{10} = \frac{V_{nach\,10m}}{g}$$

$$t_{10} = \frac{\sqrt{2 g h_{10}}}{g} = \sqrt{2}\,\frac{g^{1/2}}{g}\sqrt{h_{10}}$$

$$t_{10} = \sqrt{2}\,\frac{1}{\sqrt{g}}\sqrt{h_{10}}$$

[2]

2 a

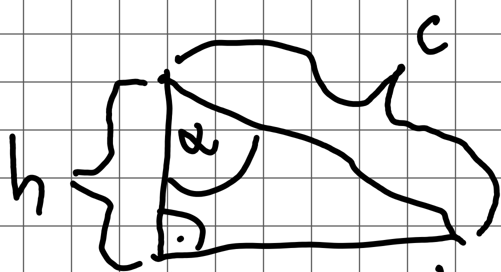


$F_d = mg\cos\alpha$

$F_n = mg\sin\alpha$

2 b & c

$V_{10} = ?$



$h = \cos\alpha \, c$

$c = \dfrac{h}{\cos\alpha}$

→ nur ein Freiheitsgrad, in F_d-Richtung
↳ Ball wird mit F_d entlang c geschoben

$$\cancel{m}\dot{v} = F_d = \cancel{m}g\cos\alpha$$

$$\dot{v} = g\cos\alpha \quad \Big| \int dt$$

$$v(t) = g\cos\alpha \, t \rightarrow \text{suche } t_c \text{ mit } x(t)$$

$$x(t) = \frac{1}{2} g\cos\alpha \, t^2 \overset{!}{=} c$$

c) $t_c = \sqrt{\dfrac{2c}{g\cos\alpha}} = 2{,}02\,s$

b) $V(t = t_c) = \dfrac{g\cos\alpha}{(g\cos\alpha)^{1/2}} \sqrt{2c} = \underline{\underline{\sqrt{2c\,g\cos\alpha}}} \approx 9{,}90\,\frac{m}{s}$

③ a) $s = f(\varphi) = \, ?$

$s(2\pi) = 2\pi r$

$\Rightarrow s = \varphi r \quad \Big| \frac{d}{dt}$

$v = \omega r$

b) $E_{kin} = E_{rot} + E_{trans} = E_{pot}$

$= \frac{1}{2} m v^2 + \frac{1}{2} \Theta_M \omega^2$

$\rightarrow v = \omega r$

$mgh = \frac{1}{2}(\Theta_M + mr^2)\,\omega^2$

$mgh = \frac{1}{2}\left(\Theta_M + mr^2\right) v^2 r^2$

$$v^2 = \frac{2mgh}{(\Theta_M + mr^2)\,r^2}$$

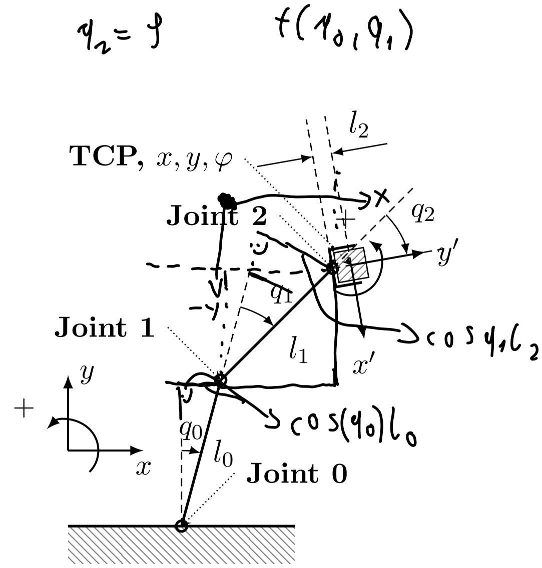


Figure 2: Scheme of extended model

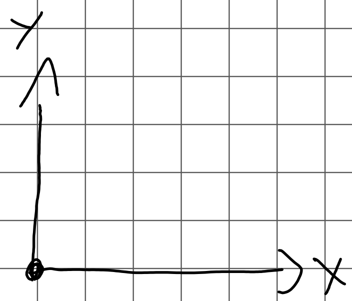
Exercise 1.2 – Create your own lua-model

In the following session, we will create a three limb model as shown in Figure 2. The model should consist of 3 segments with 3 planar revolute degrees of freedom and a free-moving cube (linear motion about x, y and rotation about φ). The basis is fixed to the ground.

- Write a lua-model for the three limb manipulator with segment lengths $l_0 = 1.0$, $l_1 = 1.0$ and $l_2 = 0.15$.
- Write the `COLUMNS` section for an animation file, such that they are ordered in the following way: `| time | x | y |` φ `| q0 | q1 | q2 |`
- Test your model using the provided `"animation.csv"`.

Hint 1: Revisit the lua-model and animation file from exercise 0.2 on the previous sheet.

Hint 2: You can use `DATA_FROM:animation.csv` instead of `DATA:` inside the animation file to import data from an additional file.


y
x

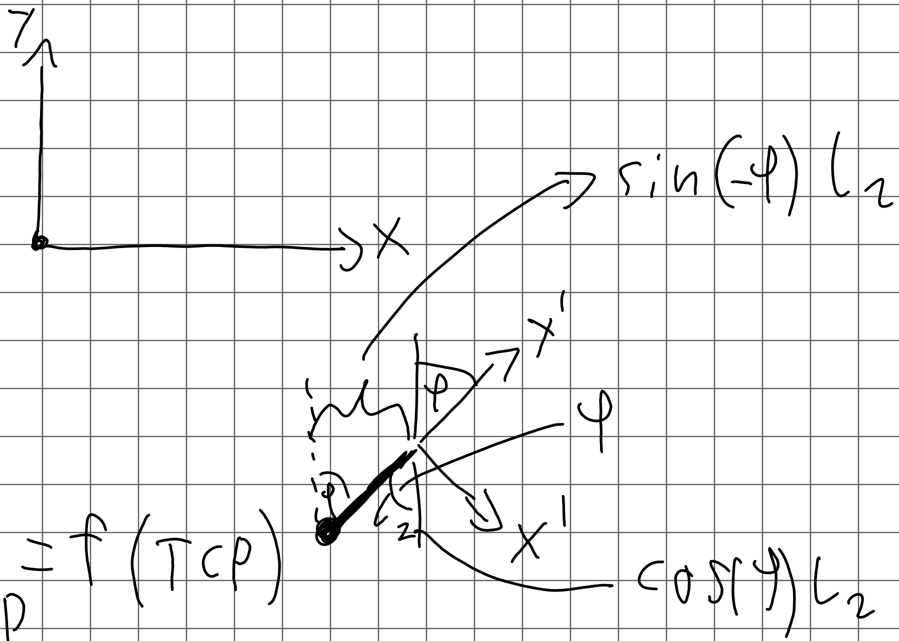
Exercise 1.3 – Inverse Kinematics

Imagine the following case: the simple planar scheme (Figure 2) models the kinematics of a real planar industrial manipulator and the cube is a tool block for a specific production task. Now, the block should move along a given trajectory $TCP(t) = [x(t), y(t), \varphi(t)]^T$ to avoid collisions during a manufacturing process. This is a path planning and optimization challenge devoted to your cooperation partner. The question is now, what are the trajectories of $q(t) = [q_0(t), q_1(t), q_2(t)]^T$ dependent on the final motion of the block?

- Express the trajectories $q(t)$ analytically in terms of the cubes trajectory $TCP(t)$.
  Hint: Calculate the position of joint 2 depending on $TCP(t)$ first. Then calculate the joint angles using trigonometry.
- **(optional)** Are there constraints for $x(t)$, $y(t)$ and $\varphi(t)$?
- Use the provided `C++` skeleton `"inverse_kinematic/main.cpp"` and implement your analytic formulation such that `q = invserseKinematic(TCP)`. Run the code and check the console output to see if your solution is correct.
- **(optional)** The `C++` code exports a new `animation.csv`. Visualize the trajectory using the lua-model you created in the previous exercise.

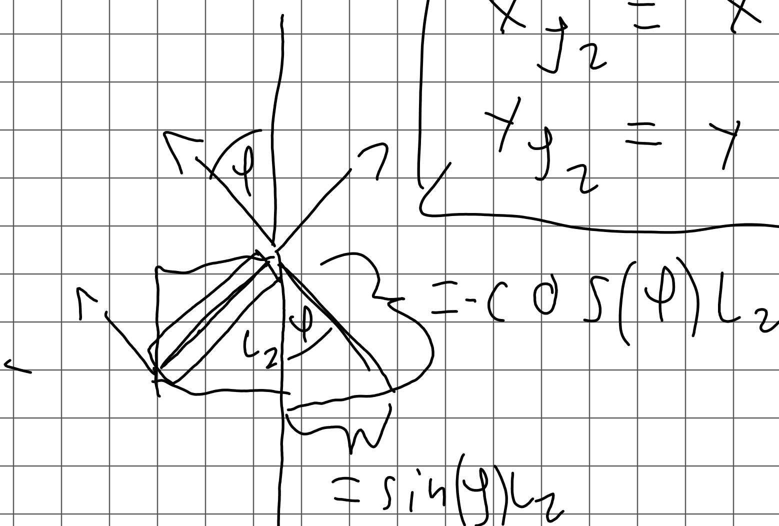
Mechanics Cheat Sheet

Position	s	$1\,\mathrm{m}$	Angle	φ	$1\,\mathrm{rad}$
Mass	m	$1\,\mathrm{kg}$	Inertia	I	$1\,\mathrm{kg}\cdot\mathrm{m}^2$
Velocity	$v = \dot{s} = \frac{d}{dt}s$		Angular Velocity	$\omega = \dot{\varphi} = \frac{d}{dt}\varphi$	
Acceleration	$a = \ddot{s} = \frac{d}{dt^2}s$		Angular Acceleration	$\dot{\omega} = \ddot{\varphi} = \frac{d}{dt^2}\varphi$	
Eq. of Motion	$s = \frac{1}{2}at^2 + v_0 t + s_0$		Eq. of Motion	$\varphi = \frac{1}{2}\dot{\omega}t^2 + \omega_0 t + \varphi_0$	
Impulse	$p = mv$		Angular Momentum	$H = I\omega$	
Kinetic Energy	$E_{\mathrm{kin,trl}} = \frac{1}{2}mv^2$		Kinetic Energy	$E_{\mathrm{kin,rot}} = \frac{1}{2}I\omega^2$	
Potential Energy	$E_{\mathrm{pot}} = mgh$				



$$y_{zp} = f(TCP)$$

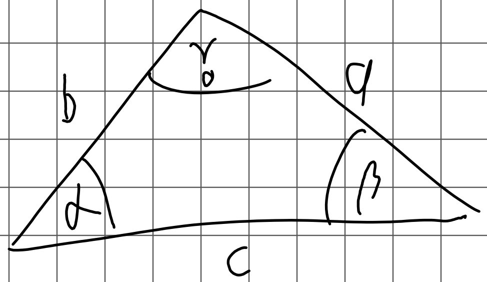
$$\boxed{\begin{aligned} X_{J_2} &= X + \sin(+\varphi)\, l_2 \\ Y_{J_2} &= Y - \cos(\varphi)\, l_2 \end{aligned}}$$



$$\cos(x+90) = -\sin(x)$$

musterlösung

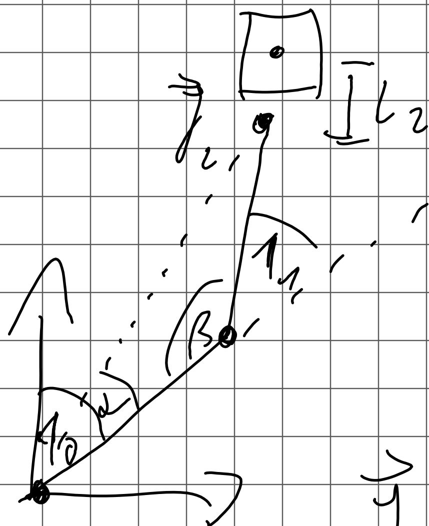
law of cosine



$$c^2 = a^2 + b^2 - 2ab\cos(\gamma)$$

law of sine

$$\frac{a}{\sin \alpha} = \frac{b}{\sin \beta} = \frac{c}{\sin c}$$



Hint

A

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} y_{2x} \\ y_{2y} \end{pmatrix} + \begin{pmatrix} l_2 \cos\varphi \\ l_2 \sin\varphi \end{pmatrix}$$

$$\vec{y}_2 = \begin{pmatrix} y_{2x} \\ y_{2y} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} - A$$

$$\psi_0 = \frac{\pi}{2} - (\theta - \alpha)$$

$$\psi_1 = \pi - \beta$$

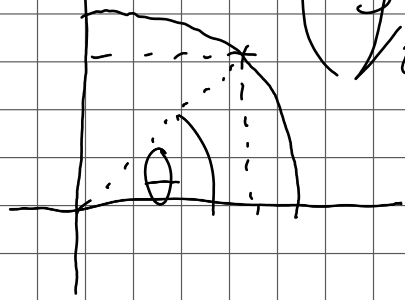
$$\varphi = \psi_0 + \psi_1 + \psi_2 + \frac{\pi}{2}$$

$$|\vec{y}_2|^2 = l_0^2 + l_1^2 + 2 l_0 l_1 \cos\beta$$

$$\rightarrow \beta = \arccos\left(\frac{l_0^2 l_1^2 - |\vec{y}_2|^2}{2 l_0 l_1}\right)$$

$$\frac{\sin\alpha}{l_1} = \frac{\sin\beta}{|\vec{y}_2|^2} \qquad \alpha = \operatorname{acsin}\left(l_1 \frac{\sin\beta}{|\vec{y}_2|}\right)$$

$$\theta = \operatorname{atan}\left(\frac{J_{2x}}{J_{2y}}\right) + \frac{\pi}{2}$$



$$\rightarrow \theta = \operatorname{atan2}(J_{2y}, J_{2x})$$

$\hookrightarrow$ implemented in eigen