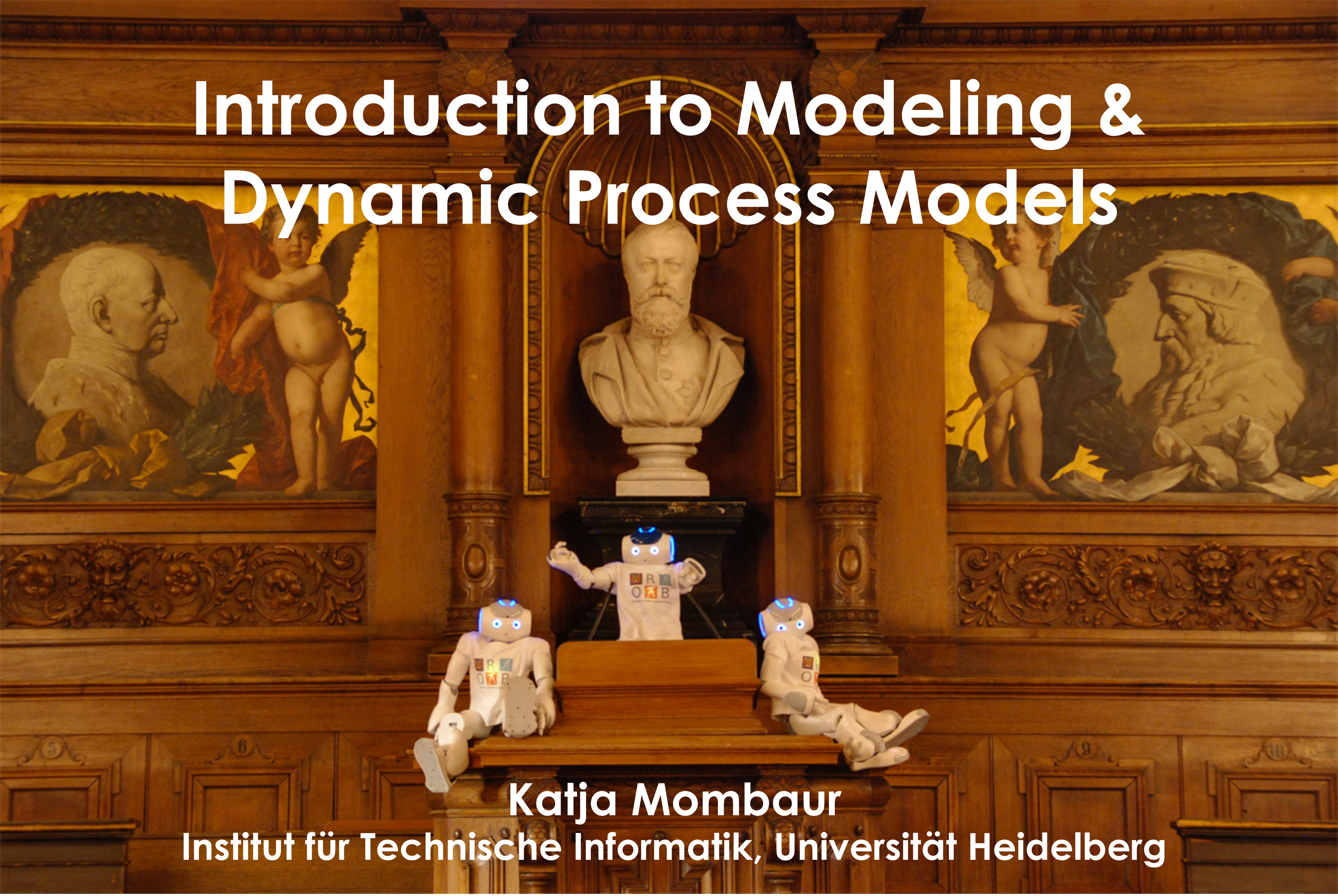
Introduction to Modeling & Dynamic Process Models



Katja Mombaur
Institut für Technische Informatik, Universität Heidelberg

Robotics 2 - 29 April 2019

What you will learn in this lecture

- Introduction to modeling
  - What is the goal of modeling?
  - Are there correct and incorrect models?
  - Introduction to the basics to human and robot models
- Basics of mathematical process models:
  - Differential equations of processes (ODE, DAE)
  - Boundary conditions
  - Multi-phase models
  - Discontinuities
  - Objective functions (cost functions)

Reality

Model

$$
\begin{aligned}
\ddot{r}_{x,1} &= \ddot{r}_{x,H} + c_T(\cos\phi_1\ddot{\phi}_1 - \sin\phi_1\dot{\phi}_1^2) \\
\ddot{r}_{y,1} &= \ddot{r}_{y,H} + c_T(\sin\phi_1\ddot{\phi}_1 + \cos\phi_1\dot{\phi}_1^2) \\[1em]
\ddot{r}_{x,2} &= \ddot{r}_{x,H} + l_T(\cos\phi_1\ddot{\phi}_1 - \sin\phi_1\dot{\phi}_1^2) + c_S(\cos\phi_2\ddot{\phi}_2 - \sin\phi_2\dot{\phi}_2^2) - w_S(\sin\phi_2\ddot{\phi}_2 - \cos\phi_2\dot{\phi}_2^2) \\
\ddot{r}_{y,2} &= \ddot{r}_{y,H} + l_T(\sin\phi_1\ddot{\phi}_1 + \cos\phi_1\dot{\phi}_1^2) + c_S(\sin\phi_2\ddot{\phi}_2 + \cos\phi_2\dot{\phi}_2^2) + w_S(\cos\phi_2\ddot{\phi}_2 - \sin\phi_2\dot{\phi}_2^2) \\[1em]
\ddot{r}_{x,3} &= \ddot{r}_{x,H} + c_T(+\cos\phi_3\ddot{\phi}_3 - \sin\phi_3\dot{\phi}_3^2) \\
\ddot{r}_{y,3} &= \ddot{r}_{y,H} + c_T(+\sin\phi_3\ddot{\phi}_3 + \cos\phi_3\dot{\phi}_3^2) \\[1em]
\ddot{r}_{x,4} &= \ddot{r}_{x,H} + l_T(\cos\phi_3\ddot{\phi}_3 - \sin\phi_3\dot{\phi}_3^2) + c_S(\cos\phi_4\ddot{\phi}_4 - \sin\phi_4\dot{\phi}_4^2) - w_S(\sin\phi_4\ddot{\phi}_4 - \cos\phi_4\dot{\phi}_4^2) \\
\ddot{r}_{y,4} &= \ddot{r}_{y,H} + l_T(\sin\phi_3\ddot{\phi}_3 + \cos\phi_3\dot{\phi}_3^2) + c_S(\sin\phi_4\ddot{\phi}_4 + \cos\phi_4\dot{\phi}_4^2) + w_S(\cos\phi_4\ddot{\phi}_4 - \sin\phi_4\dot{\phi}_4^2) \\[1em]
\phi_4 &\equiv \phi_3 \\
\dot{\phi}_4 &\equiv \dot{\phi}_3 \\
\ddot{\phi}_4 &\equiv \ddot{\phi}_3 \\[1em]
(\ddot{\phi}_2 &\equiv \ddot{\phi}_1 \quad \text{during second phase})
\end{aligned}
$$

What is the goal of modeling?

What do the terms **modeling / models** mean?

- **Our definition (Scientific Computing, Numerical Analysis, etc.) :**
  - **Models** are a quantitative / mathematical description of reality $\Rightarrow$ equations
  - **Modeling** is the process of developing a model starting from a real “thing“

- Many other disciplines use the term Model but mean something completely different

model in architecture

wind channel model of an airplane

verbal description

Bla bla bla bla
blabla blabla
blablablabla bla
bla bla bla bla

- In the process of modeling there are always errors involved since models are always simplifications of reality (see lecture Numerik 0: modeling error)

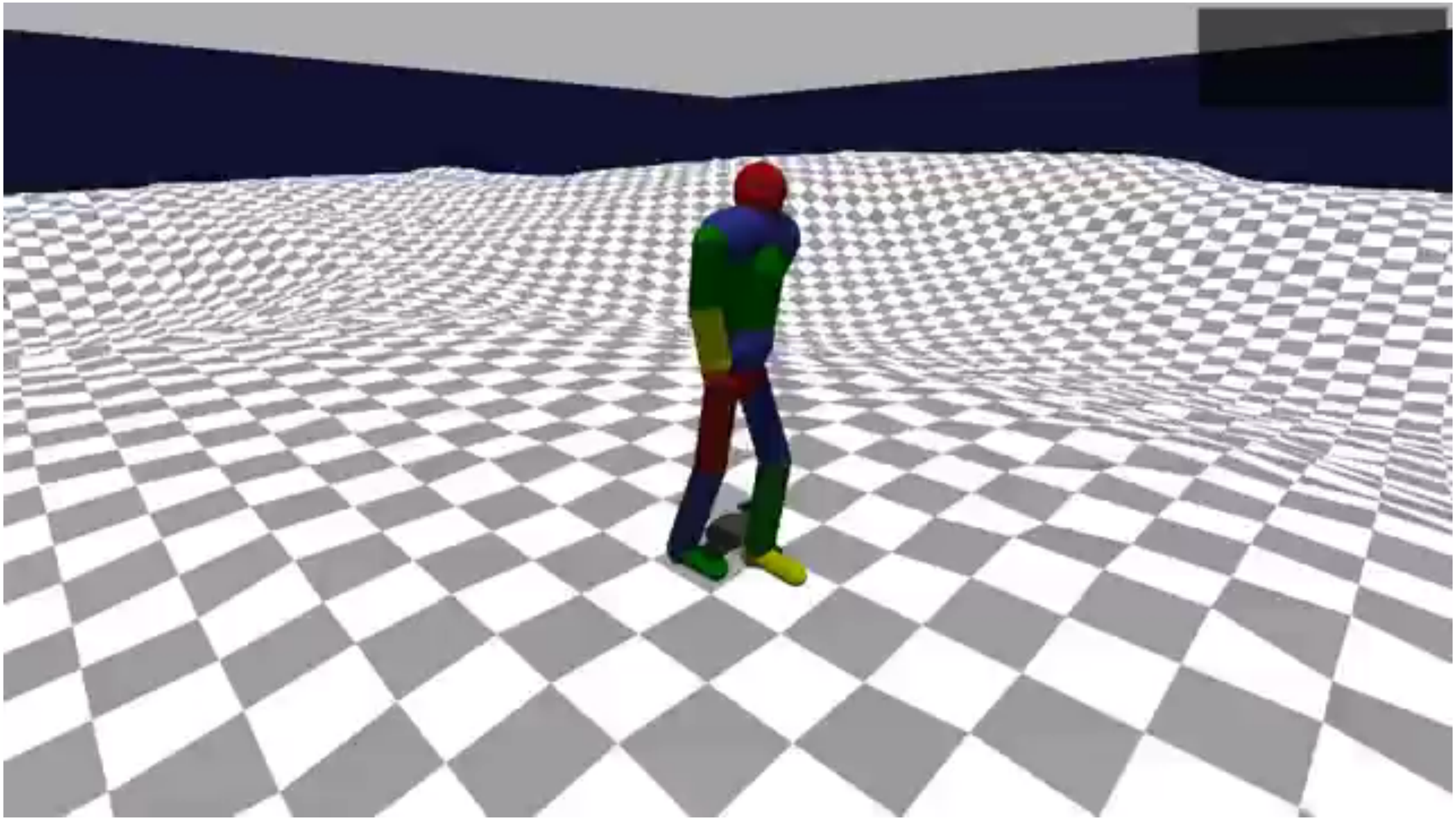
The process of modeling is not unique!

- Modeling does not follow as clearly defined rules as many other things in mathematics
- During modeling many - non unique - decisions have been taken.
- A model is not clearly wrong or false
- The choice of the model depends on the question you are asking: a given model may be good to answer one question but not useful at all to answer another one
- A model of a mechanical system is "rather wrong" or not suited if obviously physical laws are violated and that hinders the purpose of the study the model was made for.
- A computer model is always a compromise between creating a good picture of reality and keeping computational time acceptable

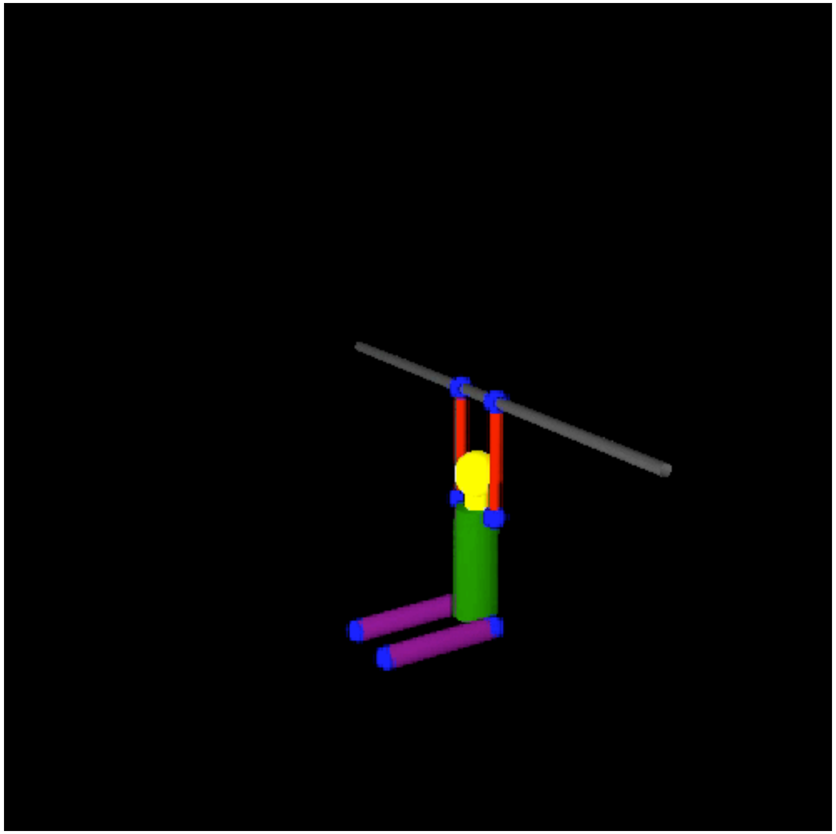
Example for a rather wrong model

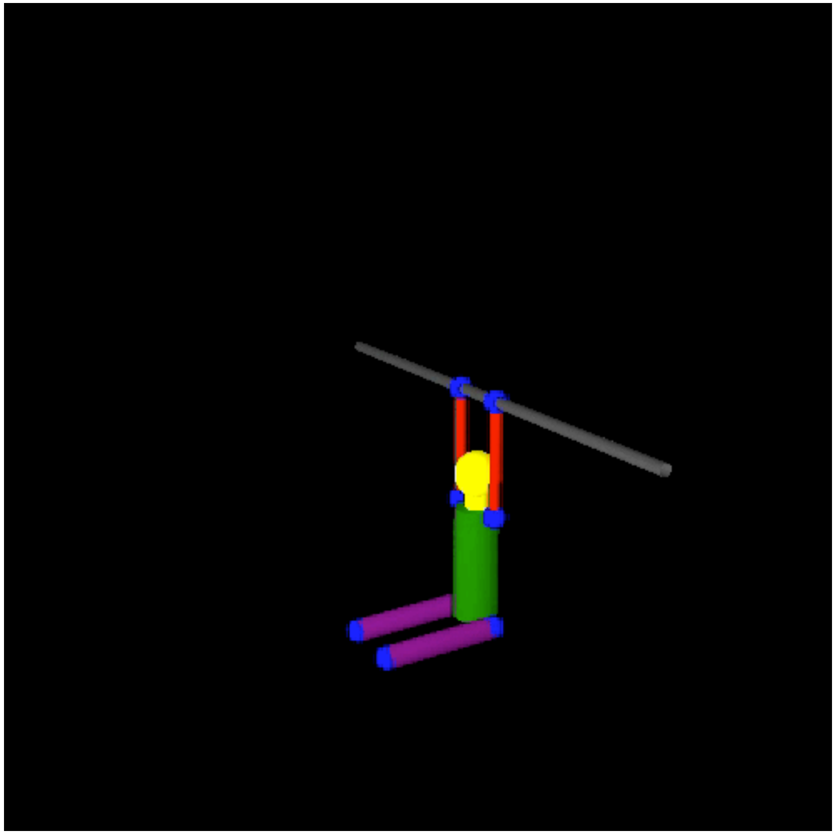
Youtube

Example for a rather wrong model



Simple model, but with correct physics





There is no unique way of formulating models of robots and humans, but …

- Models always should be physically realistic
- Depending on the question different levels of details (no. of segments etc.) should be considered
- Also depending on the question, and different aspects should be emphasized,
  - E.g. modeling fingers may be important for gasping motions of humans and robots, but not for whole-body motions like walking and running

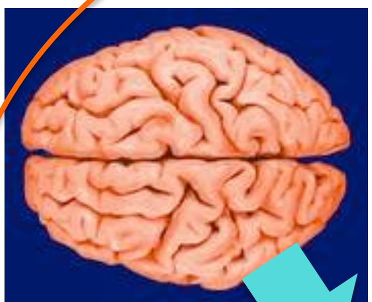
UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

As we saw yesterday, many different aspects can be considerd in human models ....

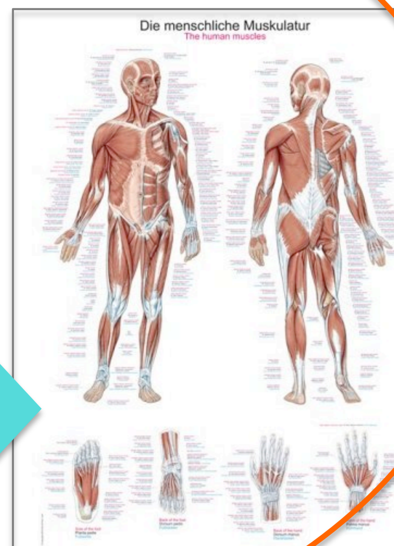
Biomechanics 1

- Human neuro-musculo-skeletal system

Brain & nervous system



Actuators



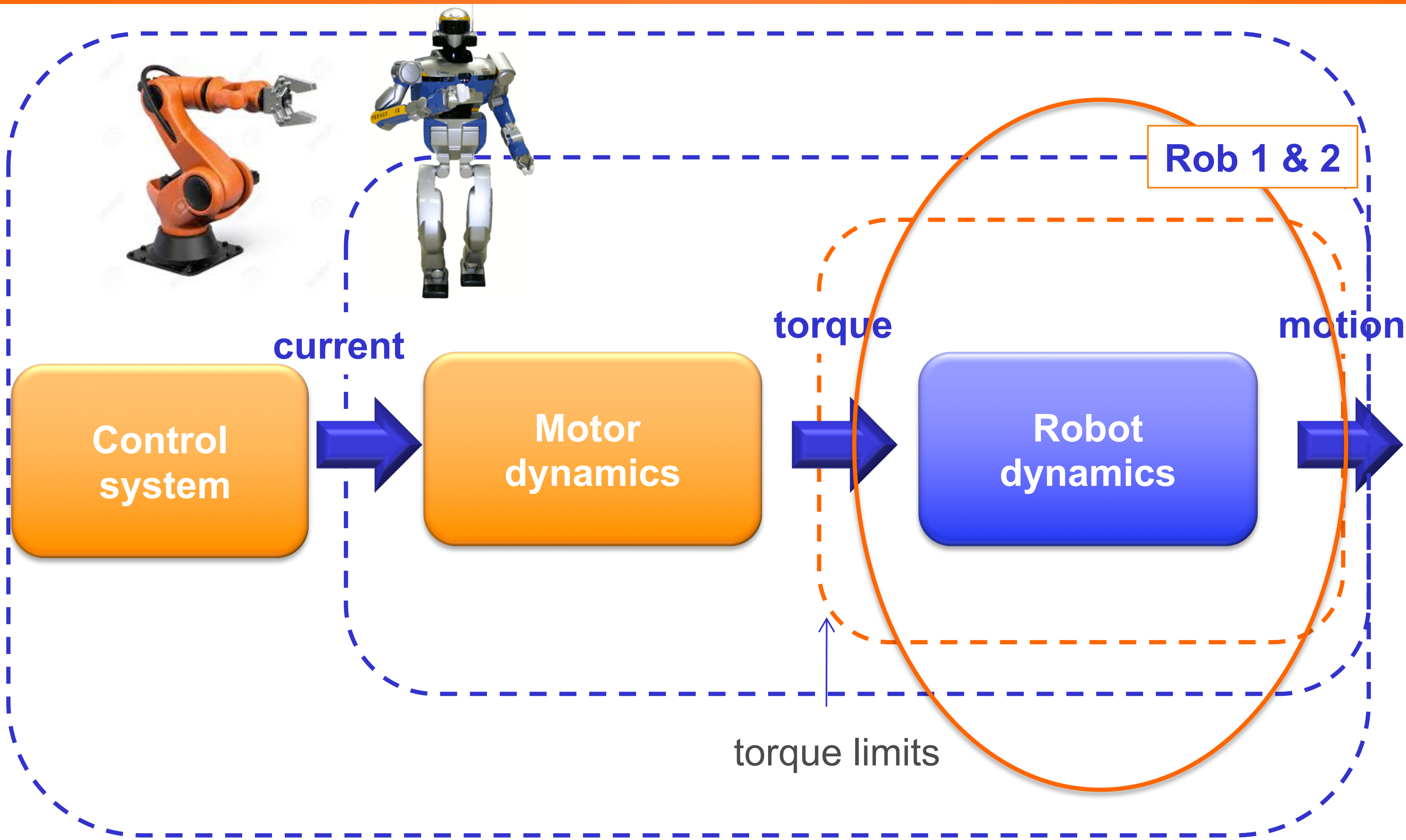
Rob 1 & 2

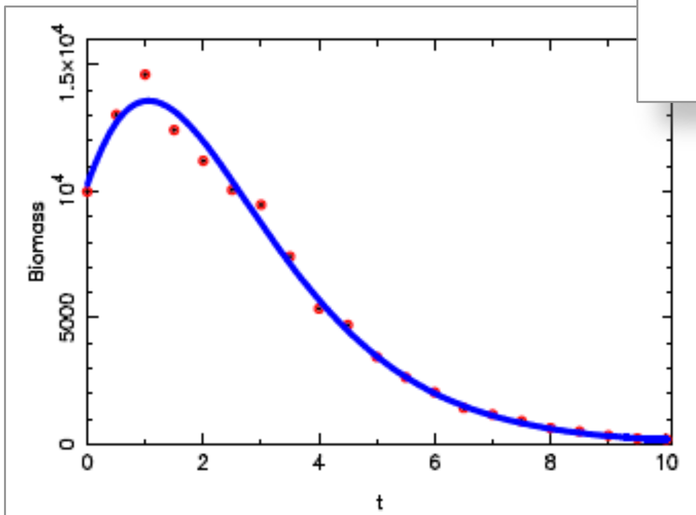
Passive mechanics

NEUROSCIENCE

BIOMECHANICS

… as well as in robot models





$$
\begin{aligned}
m \cdot \ddot{p} &= F_{ges} \\
\ddot{\theta} &= \frac{F_\theta(\theta, \dot{\theta}, \phi, \dot{\phi}, \psi)}{rm} + \sin\theta \cdot \cos\theta \cdot \dot{\phi}^2 \\
\ddot{\phi} &= \frac{F_\phi(\theta, \dot{\theta}, \phi, \dot{\phi}, \psi)}{rm \cdot \sin\phi} - 2 \cdot \cot\theta \cdot \dot{\phi}\dot{\theta}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\mathrm{d}[S]}{\mathrm{d}t} &= -k_1[E][S] + k_{-1}[C] \\
\frac{\mathrm{d}[C]}{\mathrm{d}t} &= k_1[E][S] - k_{-1}[C] - k_2[C] \\
\frac{\mathrm{d}[P]}{\mathrm{d}t} &= k_2[C]
\end{aligned}
$$

Introduction to Dynamic Process Modeling

Dynamic Process models*

There are different types of dynamic processes:

- Deterministic or stochastic
- Discrete time points or continuous in time
- Discrete or continuous state variables

In the context of this lecture, we will consider models of motions of mechanical systems which in general can be described as **deterministic, time and state continuous processes** that can be described by systems of **ordinary or differential algebraic equations**

* Here the word "dynamic" is not used in the mechanical sense that we will discuss later in this lecture, but in the sense of "depending on time"

Other types of process models

- **Discrete-time systems**

$$x_{k+1} = f(x_k, u_k), \quad k = 0, 1, \ldots$$

with states $x_k \in X$ and controls $u_k \in U$ (cont. or discrete sets)

- **Games (like chess)**

discrete-time (separate moves) und finite state space (due to 64 separate squares), with an opponent

- **Markov chains:**

discrete-time, finite state space; stochastic process which is described by a probabilistic transition depending on the present state of the system

$$P(x_{k+1} | x_k, u_k), \quad k = 0, 1, \ldots$$

Process model with explicit differential equation(s)

- First order system of Ordinary differential equation (ODE):

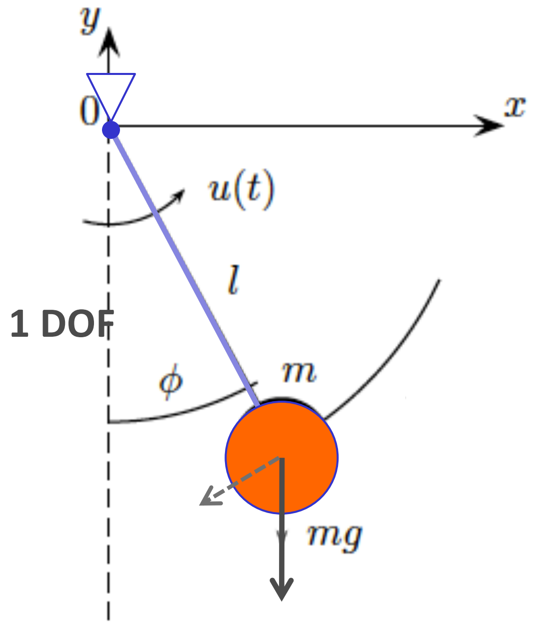
$$\dot{x} = f(t, x(t), u(t), p)$$

Example: Robot

- Time: $t \in [t_0, t_\mathrm{f}]$
- States: $x(\cdot) : [t_0, t_\mathrm{f}] \mapsto \mathbb{R}^{n_x}$ → (Positions, velocities)
- Controls $u(\cdot) : [t_0, t_\mathrm{f}] \mapsto \mathbb{R}^{n_u}$ → (Joint torques)
- Parameters $p \in \mathbb{R}^{n_p}$ → (Masses, inertia, length etc.)

Example of a mechanical system: A simple pendulum

- Generation of the equations of motion; transformation from 2nd order to 1st order:
- Oscillation is a pure rotation about point 0*



$$\theta_0 \cdot \ddot{\phi} = \sum M_0$$

$$ml^2 \ddot{\phi} = u(t) - mg \cdot \sin\phi \cdot l$$

$$\ddot{\phi} = \frac{u(t)}{ml^2} - \frac{g}{l} \sin\phi$$

$$\underbrace{\begin{bmatrix} \dot{\phi} \\ \dot{\omega} \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} \omega \\ \frac{u(t)}{ml^2} - \frac{g}{l} \sin\phi \end{bmatrix}}_{f(t,x(t),u(t),p)}$$

*details on how to set up mechanical models: see Robotics 1

Differential algebraic equations (DAE)

- System of ordinary differential equations is augmented by n_z algebraic equations

  $g_i(\cdot)$ and algebraic variables $z(t) : [t_0, t_f] \mapsto \mathbb{R}^{n_z}$

$$
\begin{aligned}
\dot{x}(t) &= f(t, x(t), z(t), u(t), p) \longleftarrow \text{Differential equations} \\
0 &= g(t, x(t), z(t), u(t), p) \longleftarrow \text{Algebraic equations}
\end{aligned}
$$

- The derivative of the algebraic equations with respect to time is also zero:

$$
0 = \frac{\mathrm{d}g(t, x(t), z(t), u(t), p)}{\mathrm{d}t} = \frac{\partial g}{\partial t} + \frac{\partial g}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial g}{\partial z}\frac{\mathrm{d}z}{\mathrm{d}t} + \frac{\partial g}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}t}
$$

- If $\frac{\partial g}{\partial z} \in \mathbb{R}^{n_z \times n_z}$ can be inverted, it holds:

$$
\frac{\mathrm{d}z}{\mathrm{d}t} = -\frac{\partial g}{\partial z}^{-1}\left(\frac{\partial g}{\partial t} + \frac{\partial g}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial g}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}t}\right)
$$

➡ Transformation of whole system to ODE is possible

Differential algebraic equations (DAE)

- If $\frac{\partial g}{\partial z} \in \mathbb{R}^{n_z \times n_z}$ is not invertible, higher order derivatives of the equation h have to be generated.

- **Definition: (Differential) Index of a DAE**

  The differential index of a DAE is equal to the order of the derivatives (with respect to time) of the algebraic constraints that must be computed in order to transform the DAE into and ODE of the form

$$\dot{\begin{pmatrix} x \\ z \end{pmatrix}} = \dot{y} = F(\dots)$$

- If the system only needs to be differentiated once (as on the previous slide):

  DAE of index 1

Boundary conditions - general

- The general form of boundary conditions

$$r(x(t_0), x(t_1), \ldots, x(t_{\text{f}}), \quad p) = 0, \quad r \in \mathbb{R}^{n_r}$$

includes the following cases

  - Initial value constraints
  - End value constraints
  - Other decoupled constraints (only one point in time is concerned)
  - Periodicity constraints
  - Other coupled constraints (concerning relations between two or more points)

Boundary conditions – Initial values

- Initial conditions:

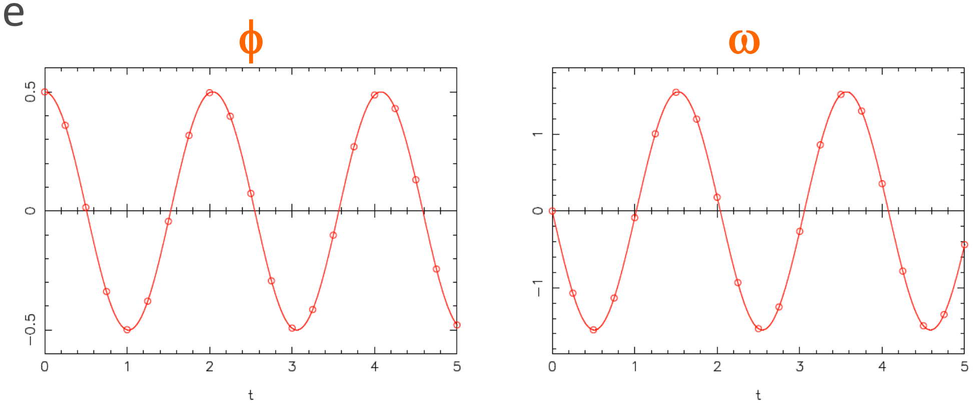
$$x(t_0) - x_0 \quad = \quad 0 \qquad \text{Fixed initial values}$$

$$x(t_0) - x_0(\quad p) \quad = \quad 0 \qquad \text{Parameter dependent initial values}$$

- Pendulum example

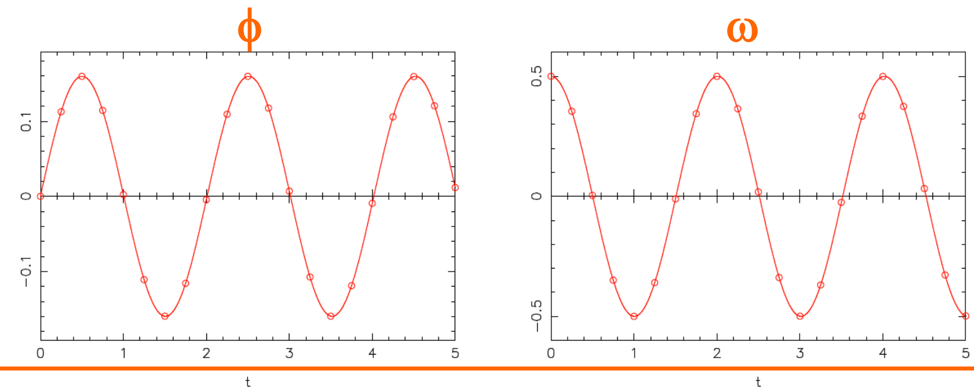
$\Phi_0 = 0.5$
$\omega_0 = 0.0$

$u = 0.0$



$\Phi_0 = 0.0$
$\omega_0 = 0.5$

$u = 0.0$



Decoupled boundary conditions

- Only concern one point (initial point, final point, „special event point“ etc.)

- Other examples:
  - Final conditions: $$x(t_f) - x_f = 0$$
  - Linear decoupled constraints $$Ax(t_0) = c_1$$ $$Bx(t_f) = c_2$$
  - Phase switching condition $$s(x(t_s), p) = 0$$

Coupled boundary conditions

- Coupled boundary conditions establish a relationship between the states at different time points

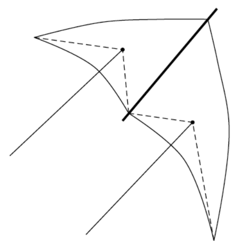
- Examples:
  - General 2 point boundary condition $r(x(t_0), x(t_f), p) = 0$
  - Linear 2 point boundary condition $Ax(t_0) + Bx(t_\mathrm{f}) = c$
  - Periodicity constraints (see next slide) $x(t_0) - x(t_\mathrm{f}) = 0$

Periodicity constraints

- **General form** of periodicity constraints

$$x(t_0) - x(t_\mathrm{f}) = 0 \qquad \text{with free} \quad t_\mathrm{f} \quad \text{and free} \quad x(t_0)$$

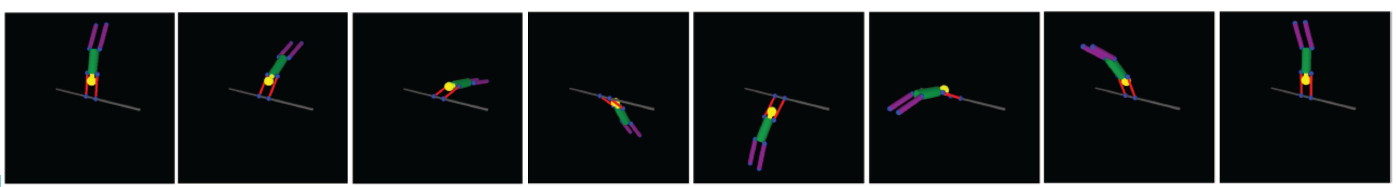
- **Example 1:** Kite moving along an infinity shaped curved



$$x(t_0) = x(t_\mathrm{f})$$

States: Kite position (spherical coord.) + velocity

- **Example 2**: High bar gymnast – periodic giant turns



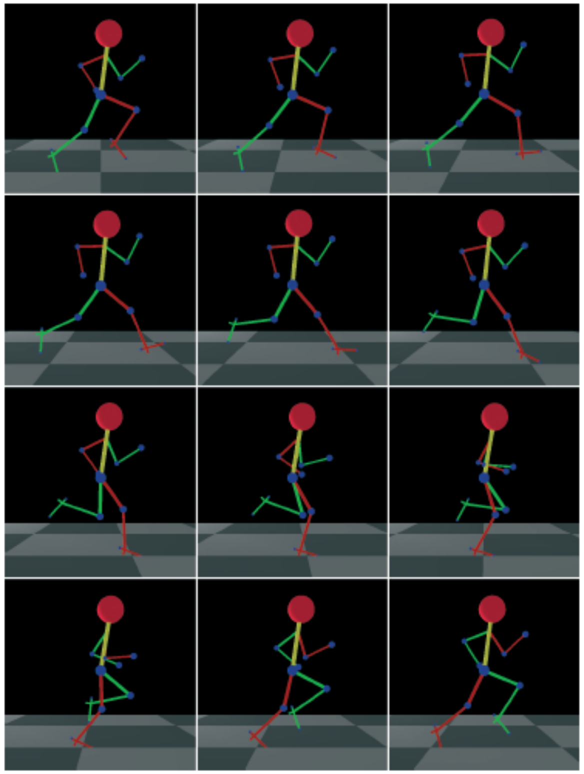
$$x(t_0) = x(t_\mathrm{f})$$

States: Position (3 Angles – at bar, shoulder and hip) + velocities

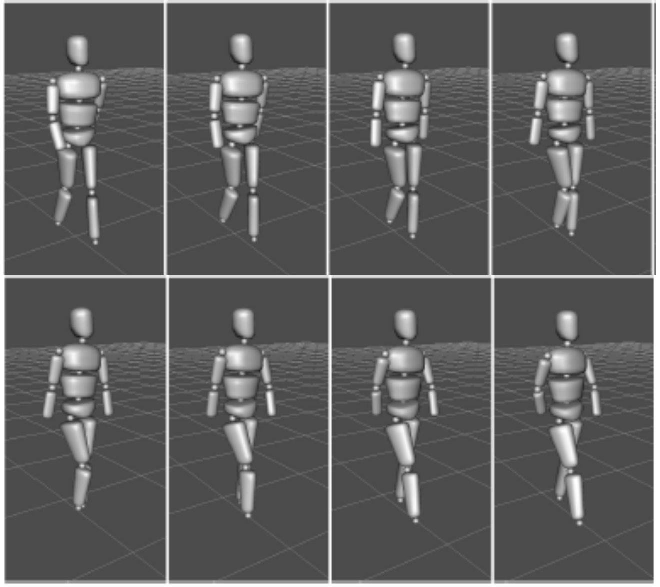
Periodicity constraints

- Example 3: walking and running motions

$$x(t_0) = x(t_\mathrm{f})$$



States: Positions (Pelvis translation and rotation, joint angles) + velocities



- Physical walking cycle: 2 steps
- Often periodicity is imposed over one step with a reflection of sides

Box constraints / simple bounds

- Simple upper and lower limits on all variables

$$x_{\min} \le x(t) \le x_{\max}$$

$$u_{\min} \le u(t) \le u_{\max}$$

$$p_{\min} \le p \le p_{\max}$$

$$T_{\min} \le T = t_{\mathrm{f}} - t_0 \le T_{\max}$$

Further constraints

- There also can be more complex inequality constraints :
  - of continuous type

$$c(t, x(t), u(t), p) \ \geq 0$$

  - or of discrete type (point-wise inequality constraints):

$$r_u(x(t_0), x(t_1), \ldots, x(t_\mathrm{f}), p) \geq 0$$

Multi-phase models

- Multi-phase models have M phases which are characterized by potentially different model equa¨

$$\dot{x}_i(t) = f_i(x_i(t), z_i(t), u_i(t), q_i, p)$$

$$t \in [t_i, t_{i+1}], \quad i = 0, \ldots, M-1$$

- In the general case, neither the duration nor the order of phases is known. Here, we generally assume that the order of phases is known.

- The phases can differ in:
  - The dimensions of the state, control and parameter vectors as well as the meaning of the entries
  - the phase durations
  - The right hand sides of the differential equations

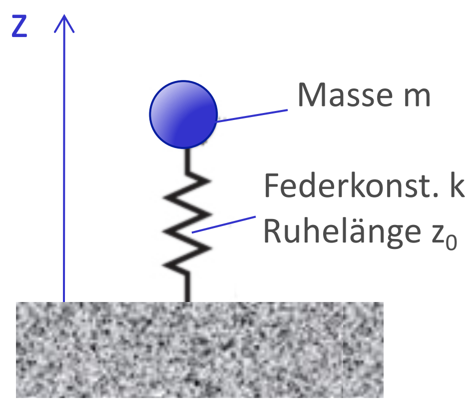
- Phase boundaries t_i are either fixed or are implicitly defined by so-called switching conditions:

$$s_i(x(t_i), p) = 0$$

Example for a multi-phase problem

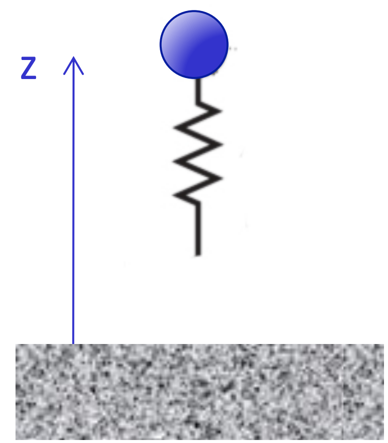
- 1D SLIP (mass spring system, hopping in place), 2 phases

Contact phase:



Schaltbedingung

Flight phase:



$$m\ddot{z} = k(z_0 - z) - mg$$

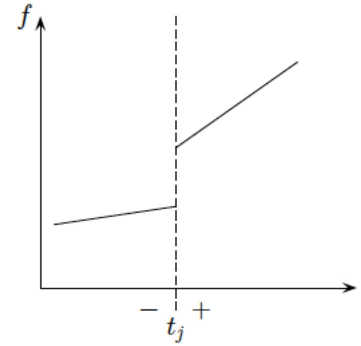
$$m\ddot{z} = -mg$$

- Switching condition between contact and flight phase (in both directions: take-off and touch down)

$$z = z_0$$

Discontinuities

- Discontinuities are instantaneous changes
  - In the right hand side: $$f_i(t_i^+) \neq f_{i-1}(t_i^-)$$
  - In the state variables: $$x_i(t_i^+) \neq x_{i-1}(t_i^-)$$



i.e.. $$x_i(t_i^+) = x_{i-1}(t_i^-) + \Delta x(t_i^-)$$

or $$x_i(t_i^+) = J(x_{i-1}(t_i^-), u(t_i^-), p)$$

- Multi-phase systems with discontinuities in the state variables are also called **hybrid systems** (combination of continuous and discrete phases)

Example for discontinuities: Hockey puck

- **Reflection of the puck at the boards:**

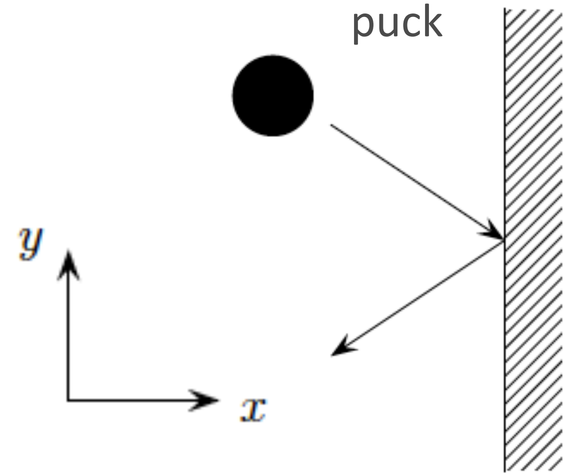
Assumption: fully elastic impact
(no energy loss)

Discontinuity of the velocity in
x direction

$$v_x(t_i^+) = -v_x(t_i^-)$$

Continuous velocity in v direction

$$v_y(t_i^+) = v_y(t_i^-)$$



Identical equations of motion before and after impact

Alternative assumption: **(partly) inelastic impact:**

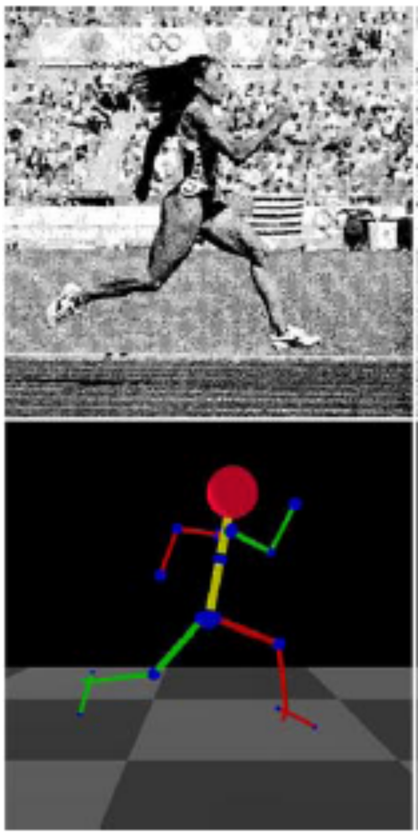
$$v_x(t_i^+) = -\gamma v_x(t_i^-), \; v_y^+ = \gamma v_y^-, \; \gamma \in [0, 1[$$

Extreme case: Fully inelastic impact: $\gamma = 0$ puck remains at board

Examples for discontinuities: running motions

- **Flight phase** | **Touch down** | **Contact phase**

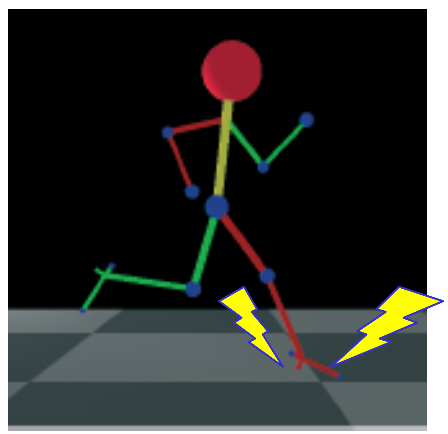
Flight phase



Equations of motion for flight

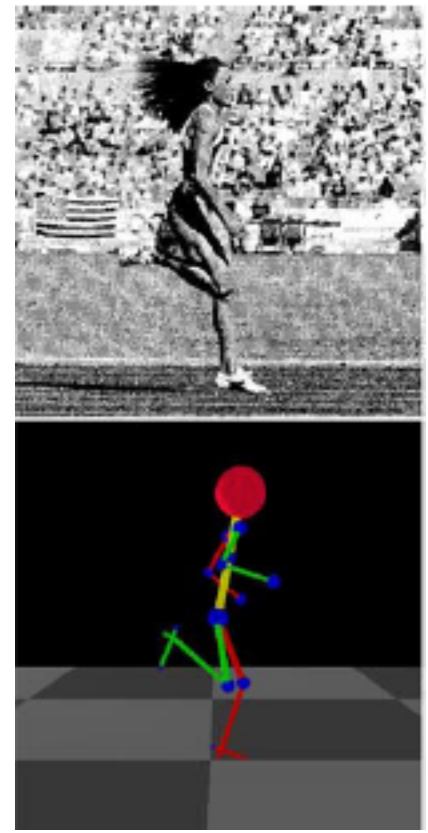
Touch down

Can be modeled as inelastic impact



occurs at foot height zero

Contact phase



Equations of motion for contact

Objective functions

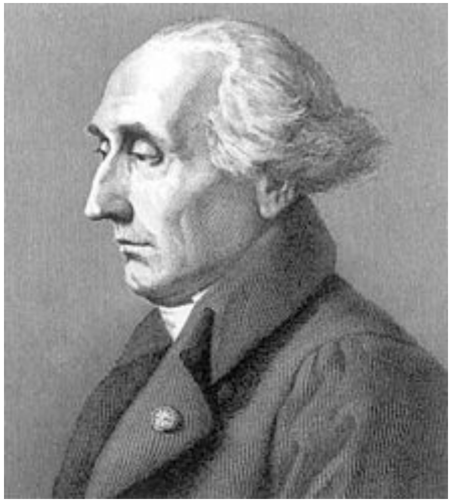
- An **objective function** is the function that is optimized in an optimization or optimal control problem (also called cost function)

- **There are different types of objective functions**

  1. **Lagrange type objective function :**

$$\min \Phi_L = \min \int_{t_0}^{t_f} L(t, x(t), u(t), p)dt$$

Integral form



Examples: Energy, efficiency, …Integral over difference to reference values, integrals over forces and torques,

Joseph-Louis de Lagrange (* 1736 in Turin als Giuseppe Lodovico Lagrangia; † 10. April 1813 in Paris) was an intalian mathematician and astronomer

Objective functions

2. Mayer type objective function

$$\min \Phi_E = \min E(t_f, x(t_f), p)$$

Depends only on values at the end of the interval

Examples:

- Total time / duration
- Total distance

Christian Gustav Adolf Mayer (1839 – 1907)
German mathematician,
student and professor in Heidelberg

Different types of objective functions are equivalent

- Lagrange term can be transformed into Mayer term

$$\min \int_{t_0}^{t_f} L(t, x(t), u(t), p)\,dt \quad \Rightarrow \quad \begin{array}{l} \dot{x}_{n+1} = L(t, x(t), u(t), p) \\ \\ \min \quad x_{n+1}(t_f) \end{array}$$

- Mayer term can be transformed into Lagrange term

$$\min E(t_f, x(t_f), p) \quad \Rightarrow \quad \min \int_{t_0}^{t_f} \dot{E}(t, x(t), p)\,dt$$

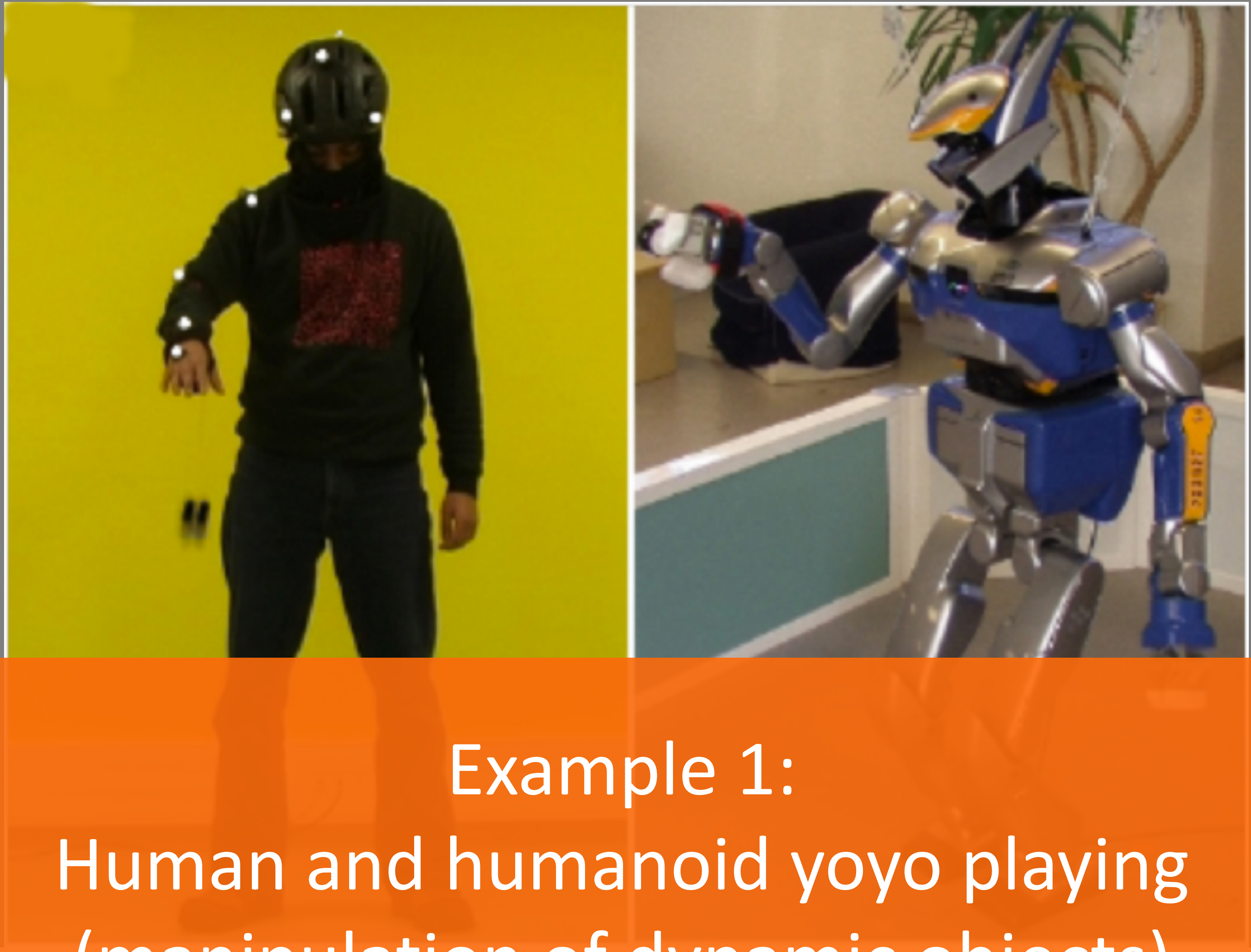
- However, in many cases it is convenient to be able to work with both forms.
  In the optimal control code MUSCOD (s. later in computer exercises) both types can be formulated without the necessity to do transformations

Objective functions

3. **Bolza type**
   Combination of Mayer und Lagrange type objective functions

$$\Phi = E(t_\mathrm{f}, x(t_\mathrm{f}), \quad p) + \int_{t_0}^{t_\mathrm{f}} L(t, x(t), u(t), \quad p)\mathrm{d}t$$

Oskar Bolza (1857 in Bergzabern, † 1942 in Freiburg / Br.)
was a German mathematician
Studied partly in Heidelberg

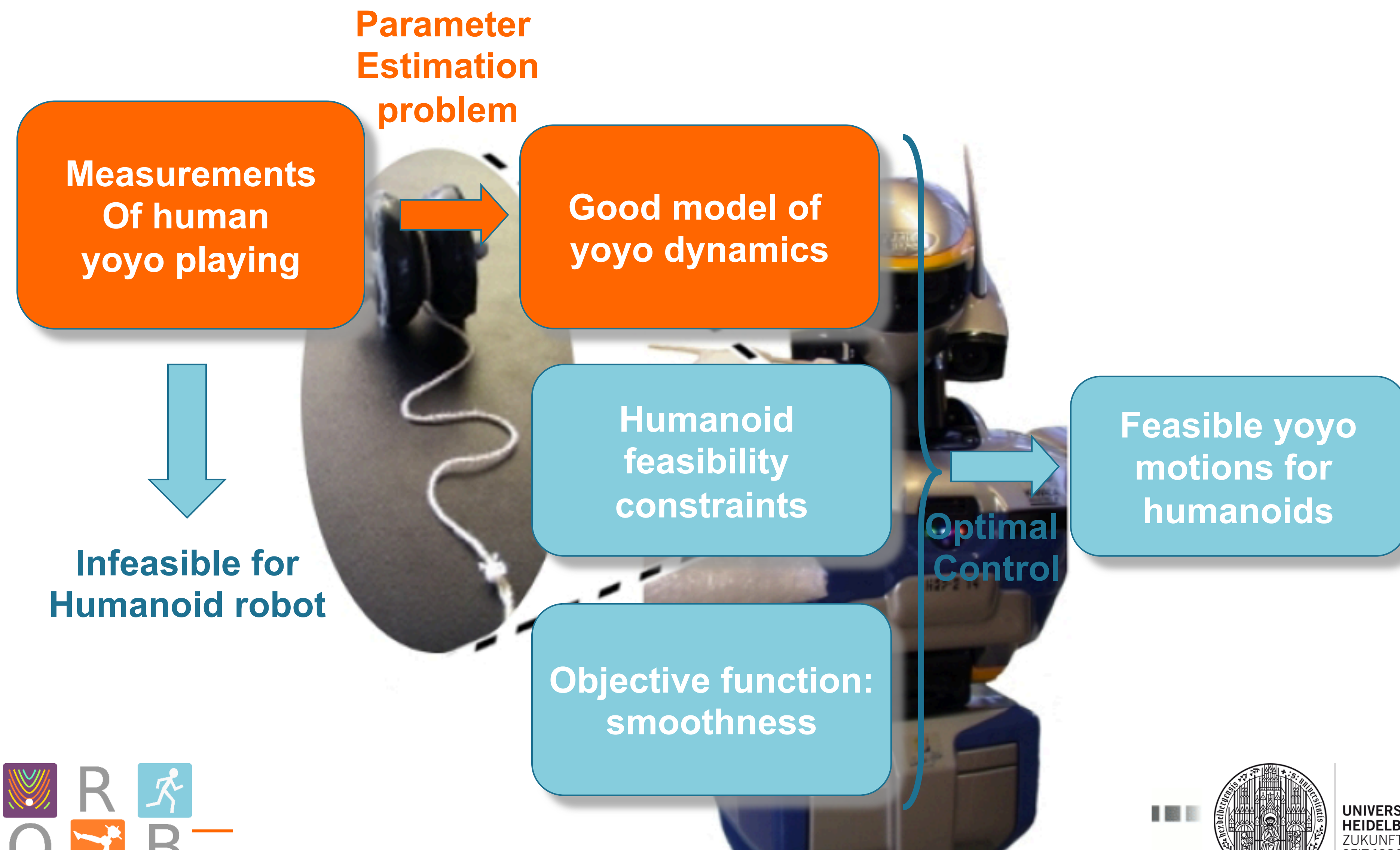


Example 1:
Human and humanoid yoyo playing
(manipulation of dynamic objects)

Motion capture of human yoyo playing

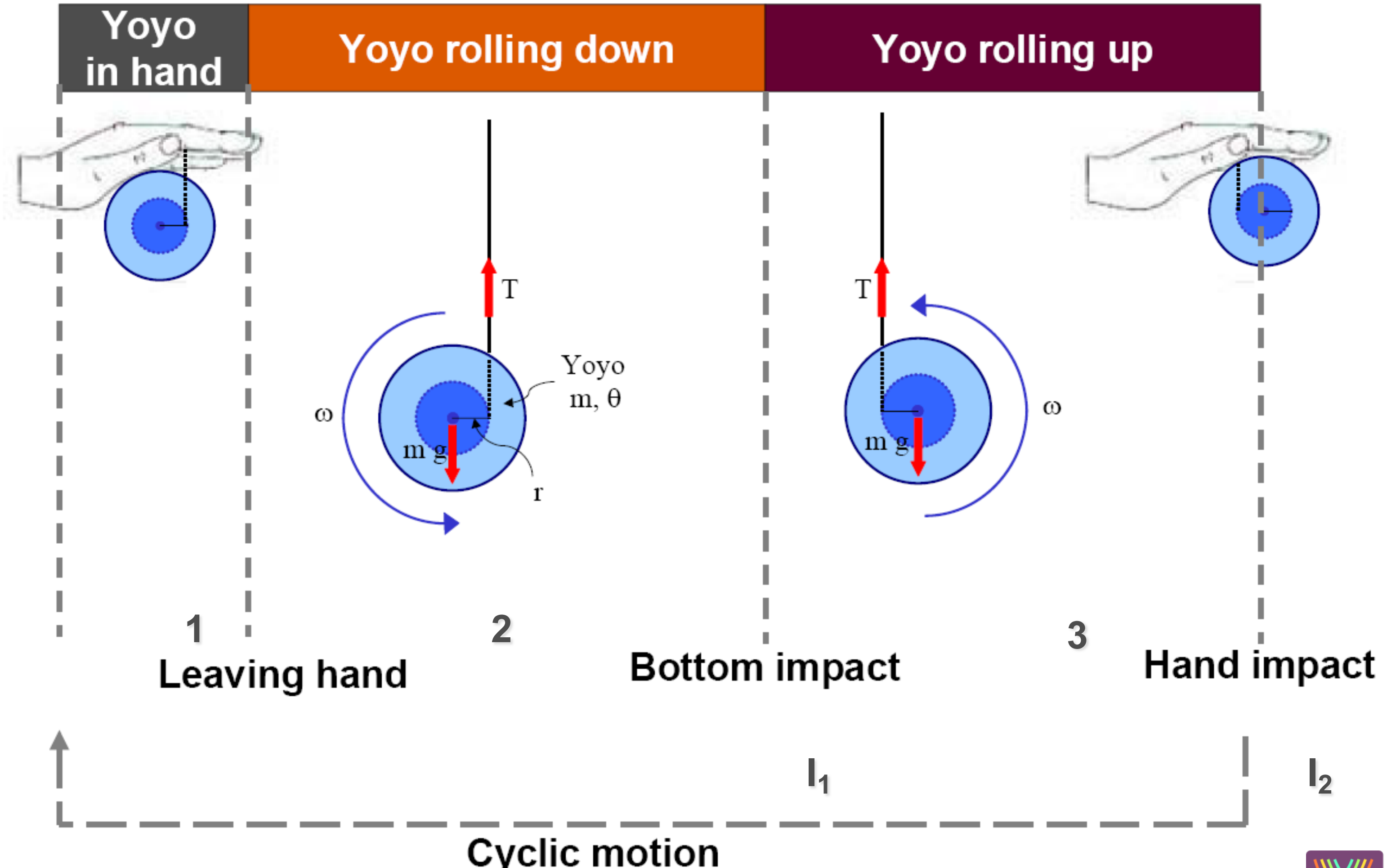
Joint work with Manish Sreenivasa (at that time at LAAS)

From human to humanoid yoyo playing



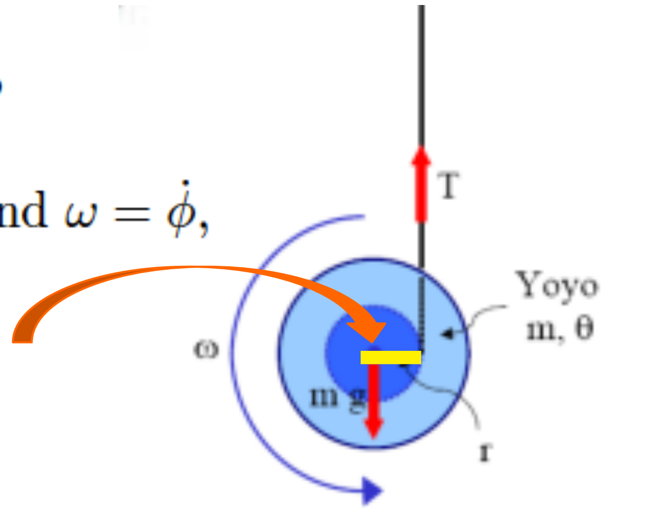
HRP-2 playing yoyo

Yoyo Model


Yoyo in hand
Yoyo rolling down
Yoyo rolling up
T
Yoyo
m, θ
ω
m g
r
T
ω
m g
1
Leaving hand
2
Bottom impact
3
Hand impact
I_1
I_2
Cyclic motion

Variables and objective function of the yoyo

- **State variables**
  - Hand height, yoyo height rotation angle h, z, ϕ
  - Correspond. velocities $v_h = \dot{h}$, $v_z = \dot{z}$, and $\omega = \dot{\phi}$,
  - Radial distance at which string acts on yoyo r.



- **Control variables (inputs)**
  - Hand acceleration a_h

- **Objective function**

$$\min_{x(\cdot),u(\cdot),p} \sum_{k=1}^{m} ||W(x_{pos}(t_k) - \hat{x}_{pos}(t_k))||_2^2 \quad \text{bzw.} \quad \min_{x(\cdot),u(\cdot),p,\tau} \int_0^T a_h^2 dt.$$

Hybrid dynamic model of the yoyo

Eqns of motion of phase 1 (yoyo in hand):

$$\begin{aligned} \dot h &= v_h \\ \dot z &= v_z \\ \dot\phi &= \omega \\ \dot v_h &= a_h \end{aligned} \qquad ✪ \qquad \begin{aligned} \dot v_z &= a_h \\ \dot\omega &= 0 \\ \dot r &= 0 \end{aligned}$$

Eqns of motions phase 2 (yoyo rolling down)

✪ &

$$\begin{aligned} \dot v_z + \dot\omega r &= a_h - \omega \dot r \\ \Theta\dot\omega - mr\dot v_z &= mgr - \gamma r\omega \\ \dot r &= -\rho\,\omega \end{aligned}$$

Eqns of motions phase 3 (yoyo rolling up)

✪ &

$$\begin{aligned} \dot v_z - \dot\omega r &= a_h + \omega \dot r \\ \Theta\dot\omega + mr\dot v_z &= -mgr - \gamma r\omega \\ \dot r &= \rho\,\omega \end{aligned}$$

- Initial conditions:

$$\begin{aligned} z(\tau_0) &= h(\tau_0) + \Delta_h \\ v_z(\tau_0) &= v_h(\tau_0) \\ \phi(\tau_0) &= \omega(\tau_0) = 0. \end{aligned}$$

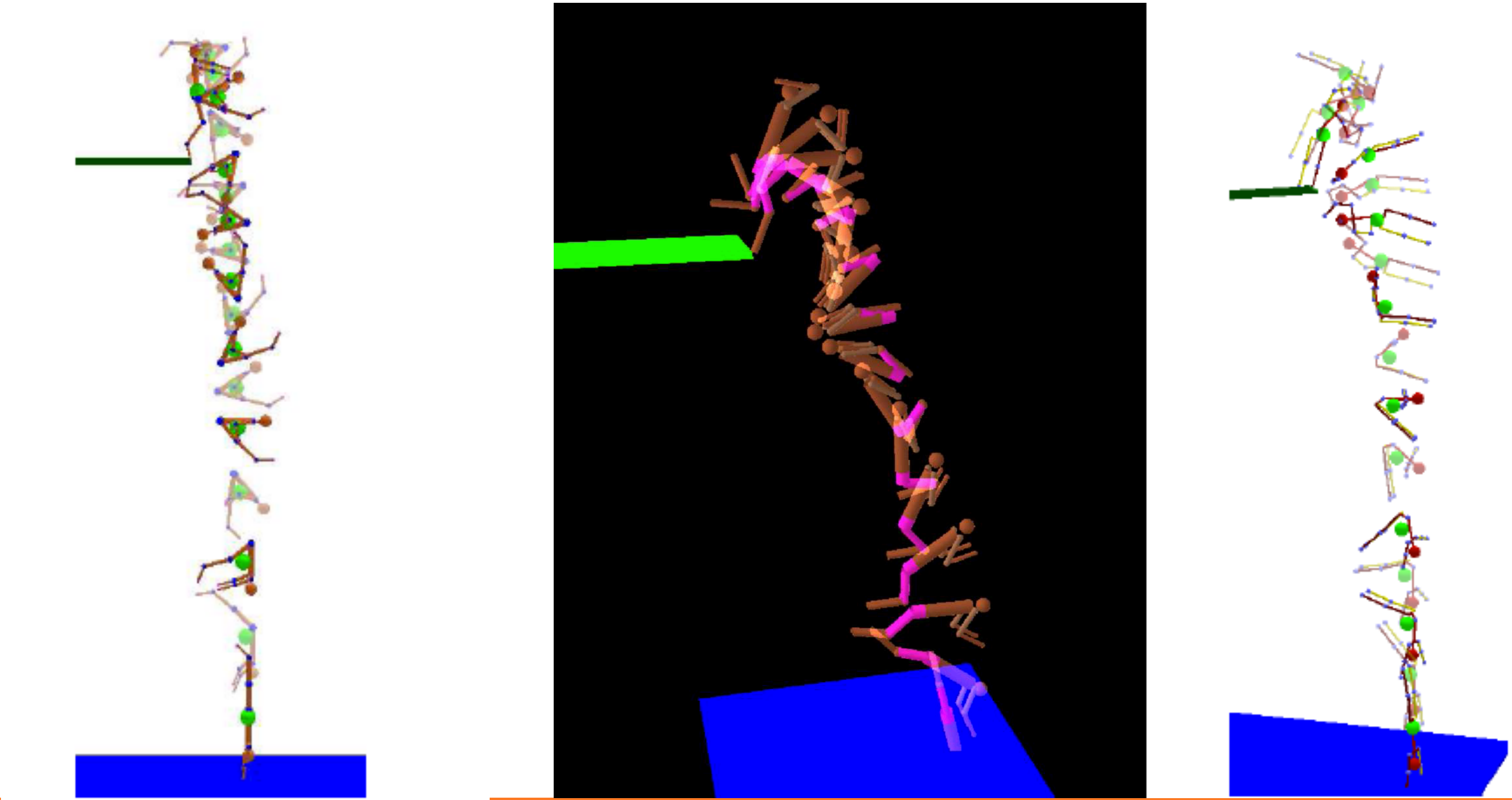
- Yoyo leaving hand at $\tau_1 = 0.1 \text{ s}$
- Bottom impact at $l = l_{max}$

$$\begin{aligned} \omega(\tau_2^+) &= \beta\omega(\tau_2^-) \quad \text{with } \beta < 1 \\ v_z(\tau_2^+) &= \begin{cases} v_h(\tau_2^+) + \omega(\tau_2^+) r & \text{if } (a_h(\tau_2^-) \geq b_{a_h}) \\ v_h(\tau_2^+) & \text{else} \end{cases} \end{aligned}$$

- Hand impact at $l = h - z - \Delta_h = 0$
- Periodicity constraints:

$$(h(T), z(T), v_h(T), r(T)) = (h(0), z(0), v_h(0), r(0))$$

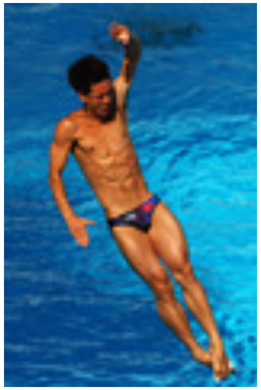
(implicit periodicity of v_z and ω)



Example 2:
Modeling and optimal control of human platform diving

Human platform diving

Thesis work of Jens Koschorreck



How to formulate and diving motions as dynamic processes and solve them as optimal control problem?

Some diving basics

- 6 types of jumps (FINA regulations):
  - Forward: jump forward, rotate forward*
  - Backward: jump backward, rotate backward*
  - Reverse: jump forward, rotate backward*
  - Inward: jump backward, rotate forward*
  - Any of the above with twists
  - All jumps starting in handstand

*somersault rot.

- Different positions during jump....

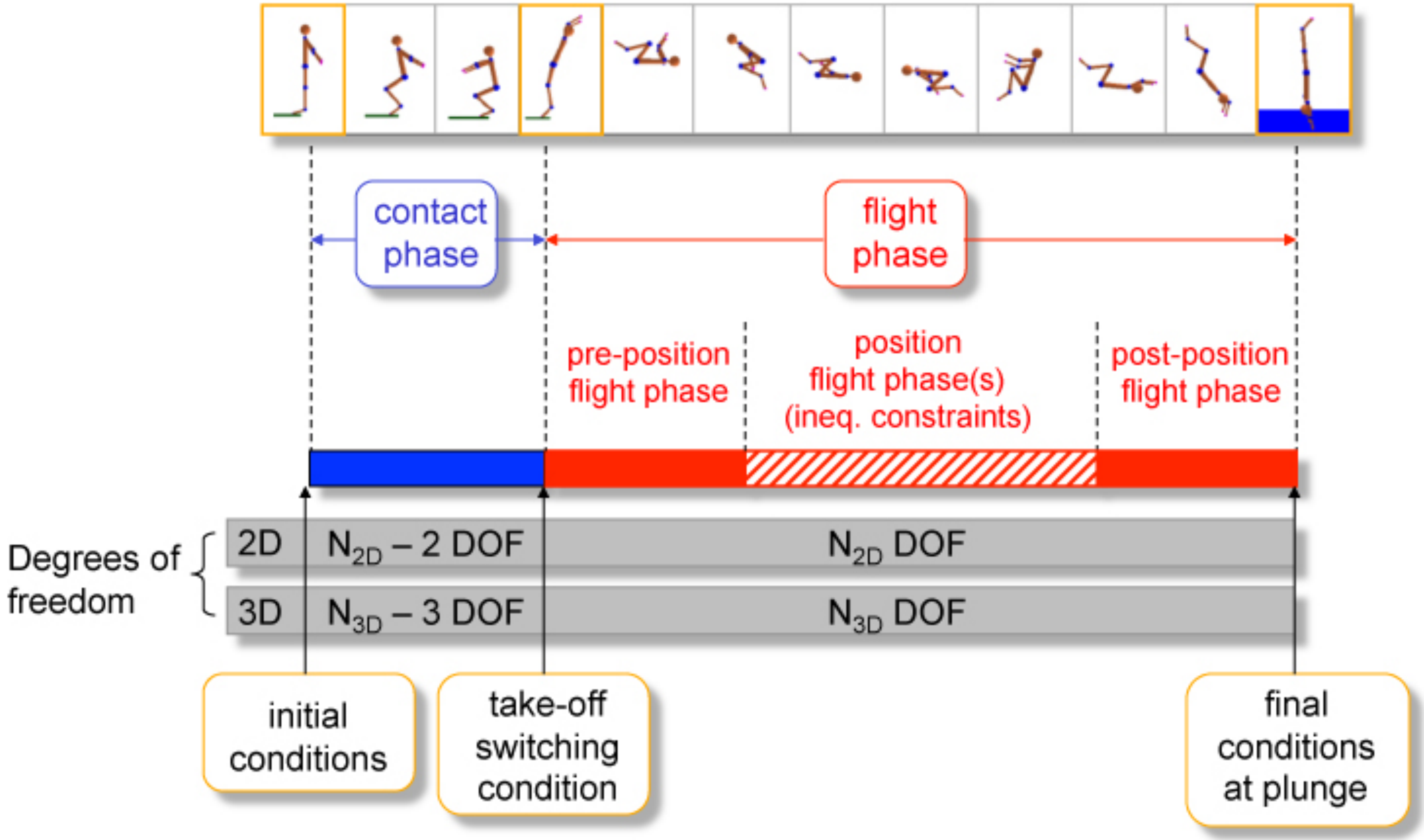
tuck

pike

straight

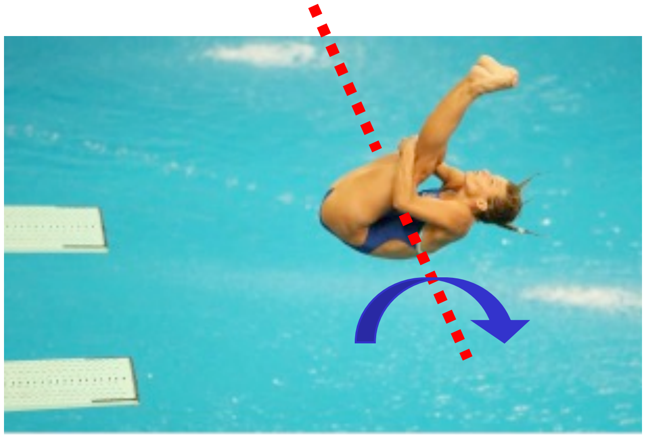
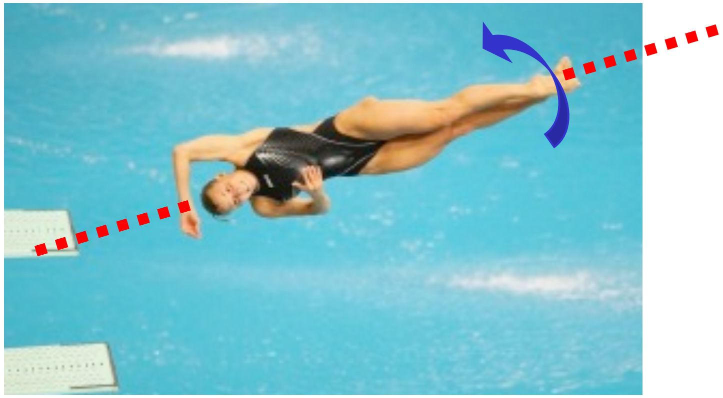
or free

The phases of motion should be in correct order



The desired number of somersault and twisting rotations should be reached

- Enforce final angles about both axes of rotation at plunch time

-  

e.g. 2.5 somersault

$$\varphi_{\text{Torso,z}}(\bar{t}_2) = -5\pi$$

e.g. 1 twisting rotation

$$\varphi_{\text{Torso,y}}\left(\bar{t}_2\right) = -2\pi$$

Formulate as end point constraints in the optimal control problem

Desired positions during the jump should be enforced

- Tuck, pike and straight positions can be described by putting tight limits on the joints



(in pure somersaults, tuck often appears naturally from the optimization, and does not have to be enforced)

- Start and end of special positions can be defined via the somersault rotation angle

Formulate as constraints of the optimal control problem

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

During flight, the motion should be as smooth and efficient as possible

- A minimization of joint torques squared usually leads to very smooth and efficient motions

$$\min \int_{\bar{t}_1}^{\bar{t}_2} \|\tau(t)\|_2^2 dt$$

(or weighted sum)

Formulate as objective function of the flight phase in the optimal control problem

Before take-off, the diver should gain as much vertical momentum as possible

- A maximization of vertical take-off velocity generates highest jumps

$$\min \left(-\dot{z}_{com} \right)$$

with $$\dot{z}_{com} = j(q, \dot{q})$$

Formulate as objective function of the contact phase
in the optimal control problem

At water entry, the diver should be completely straight, and there should be no splash

- Can be guaranteed by
  - Conditions on joint angles, such that the whole body is vertical, head down
  - Conditions on angular velocities (get them very small)

Formulate as end point constraints of the optimal control problem

Several other inequality constraints have to be considered

e.g.

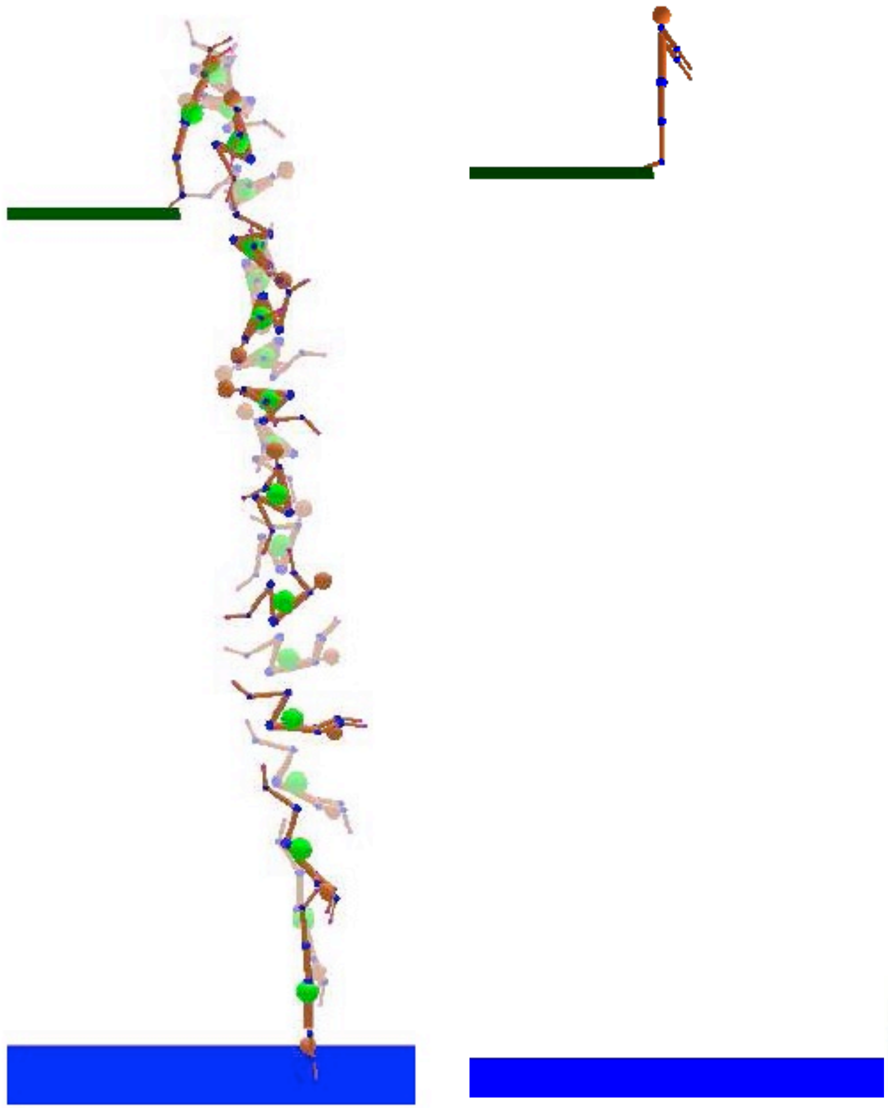
- Distance to platform or springboard during flight must be significantly > 0

- Upper and lower bounds on
  - angles
  - angular rates
  - torques
  - phase times

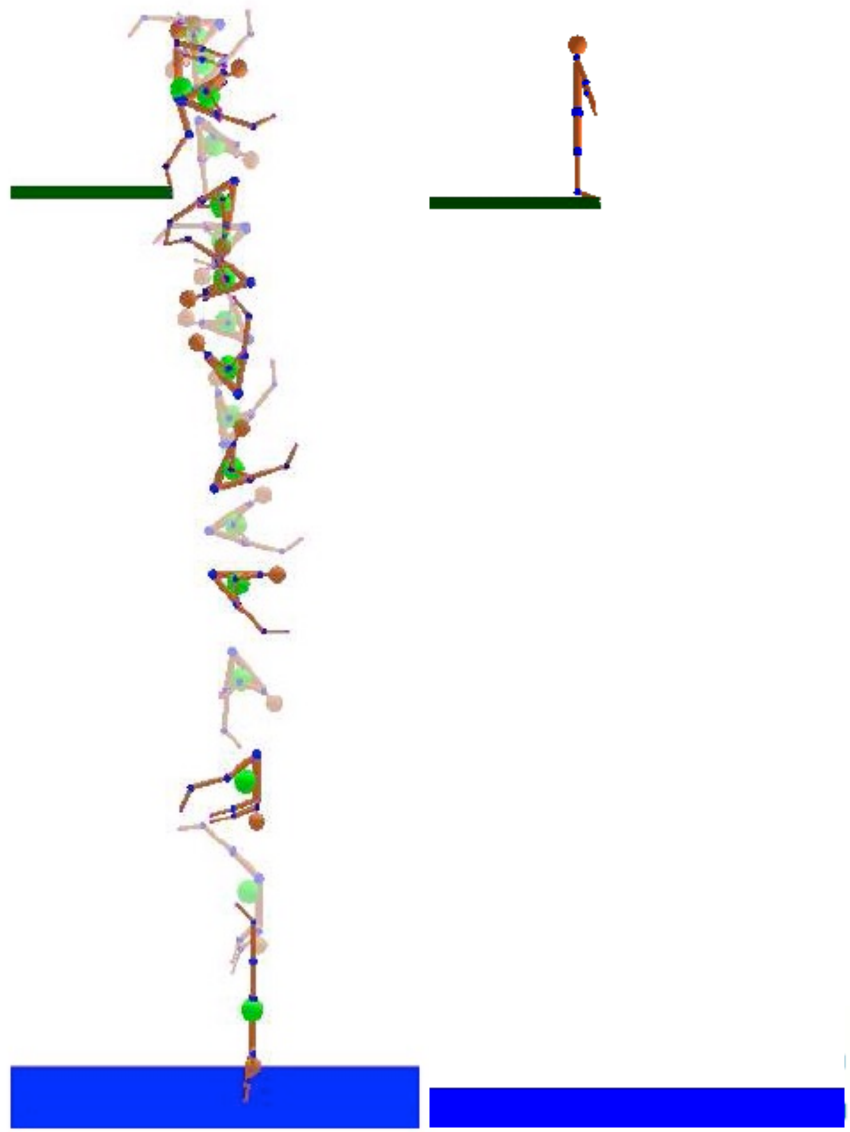
- Ground reaction force at contact phase must always point upwards

Formulate as inequality constraints of the optimal control problem

Backward 2.5 somersaults tuck



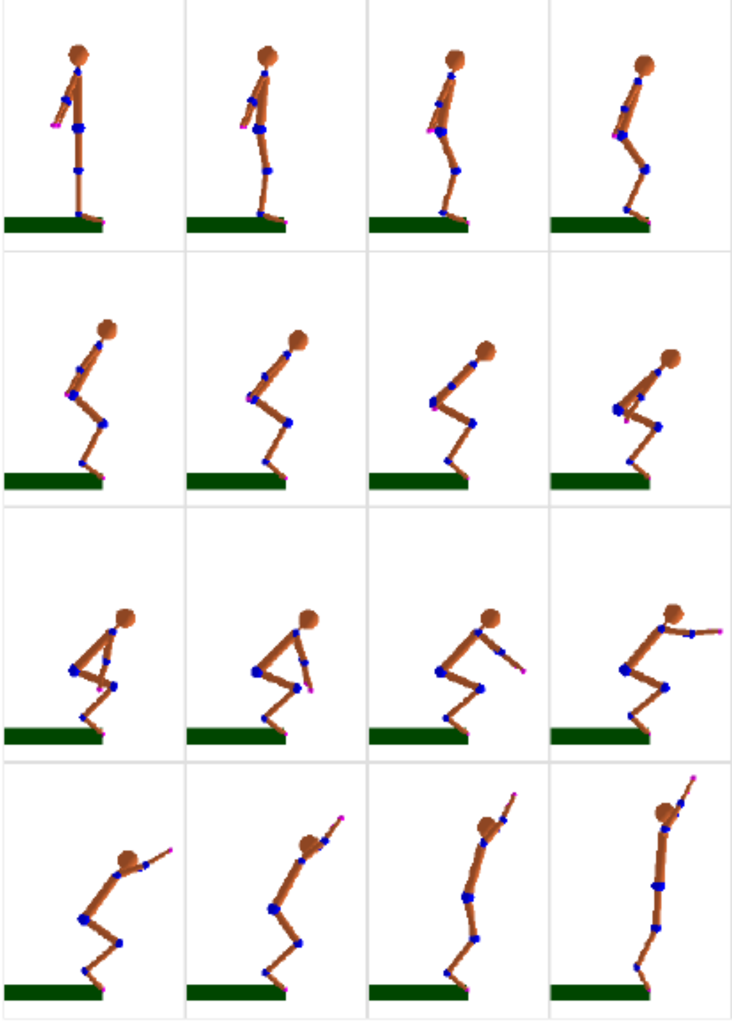
Forward 2.5 somersaults pike



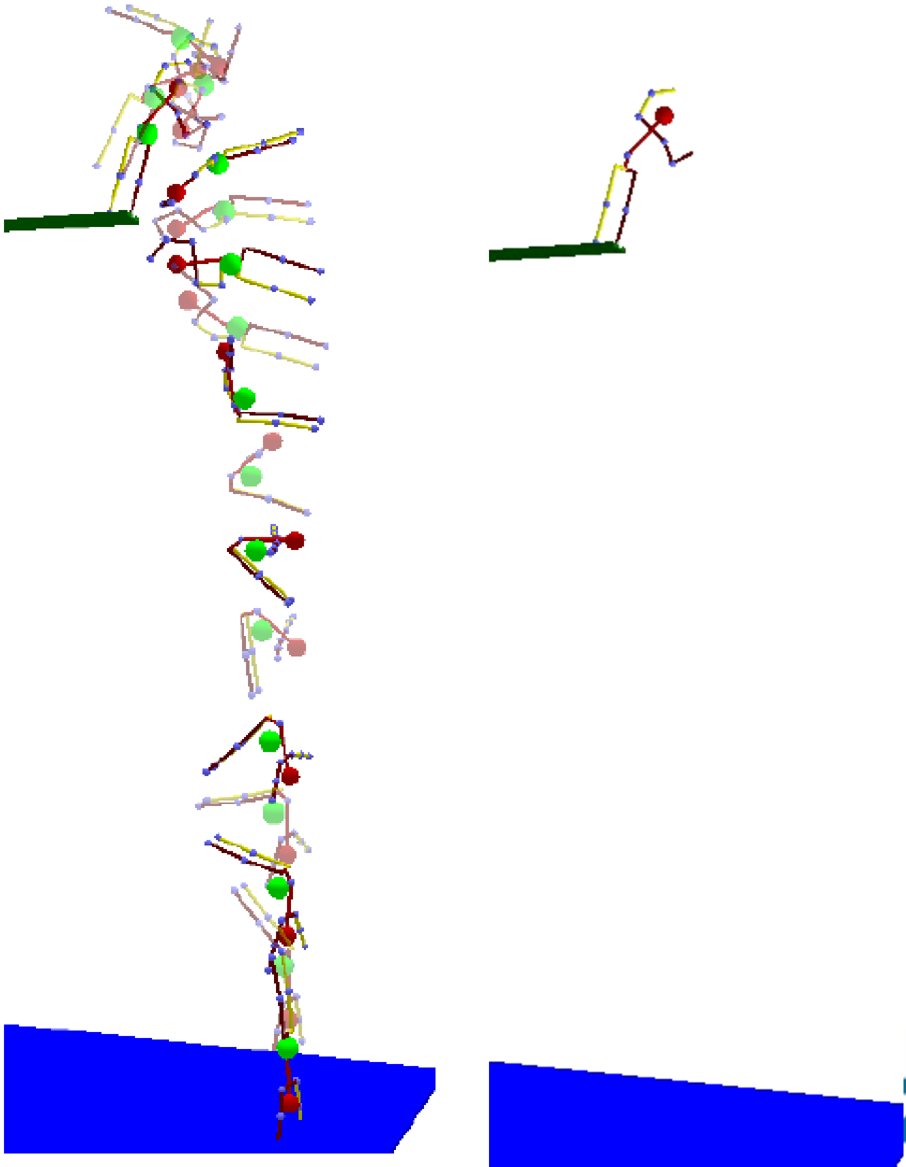
Contact phase and lift-off in detail

backward 2.5 somersault

reverse dive, straight

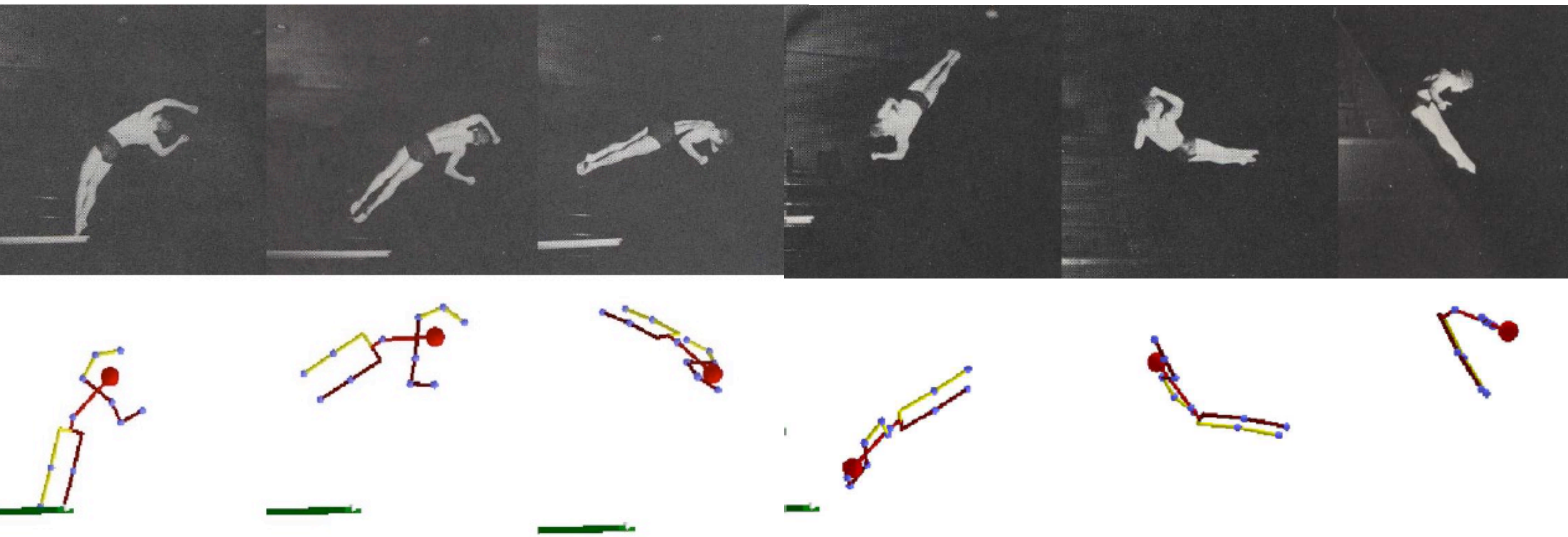


3D diving: somersaults with twists

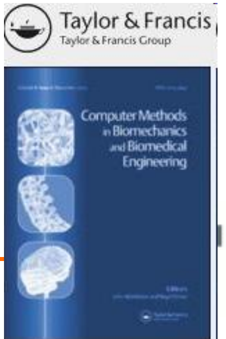


Optimal control of platform diving

- Comparison with real motions show good agreement



Koschorreck ,Mombaur, 2009



Terms that you should be familiar with after this lecture

- Modeling
- State variables
- Control variables
- Model parameters
- Different types of process models (deterministic/ stochastic, continuous / discrete in time /states)
- Models of differential equations (ODE, DAE)
- Index of a DAE
- Boundary conditions:
  - Start / end point constraints
  - Coupled / decoupled b. c.
  - Periodic b. c.
- Bounds (inequality constraints)
- Multi-phase models
- Discontinuities
- Objective function (cost function):
  - Lagrange type
  - Mayer type
  - Bolza type


Thank you very much for your attention!