Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Modeling and Control of Legged Robots
T~~6~~: Trajectory Generation

7

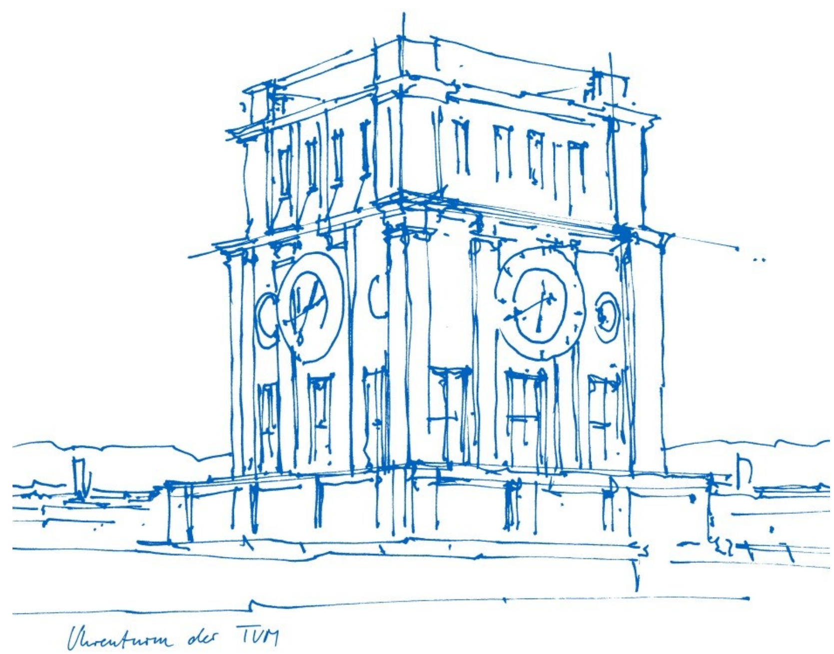
Simon Armleder

Technical University of Munich

Chair for Cognitive Systems

Prof. Dr. Gordon Cheng

20. Juni 2024

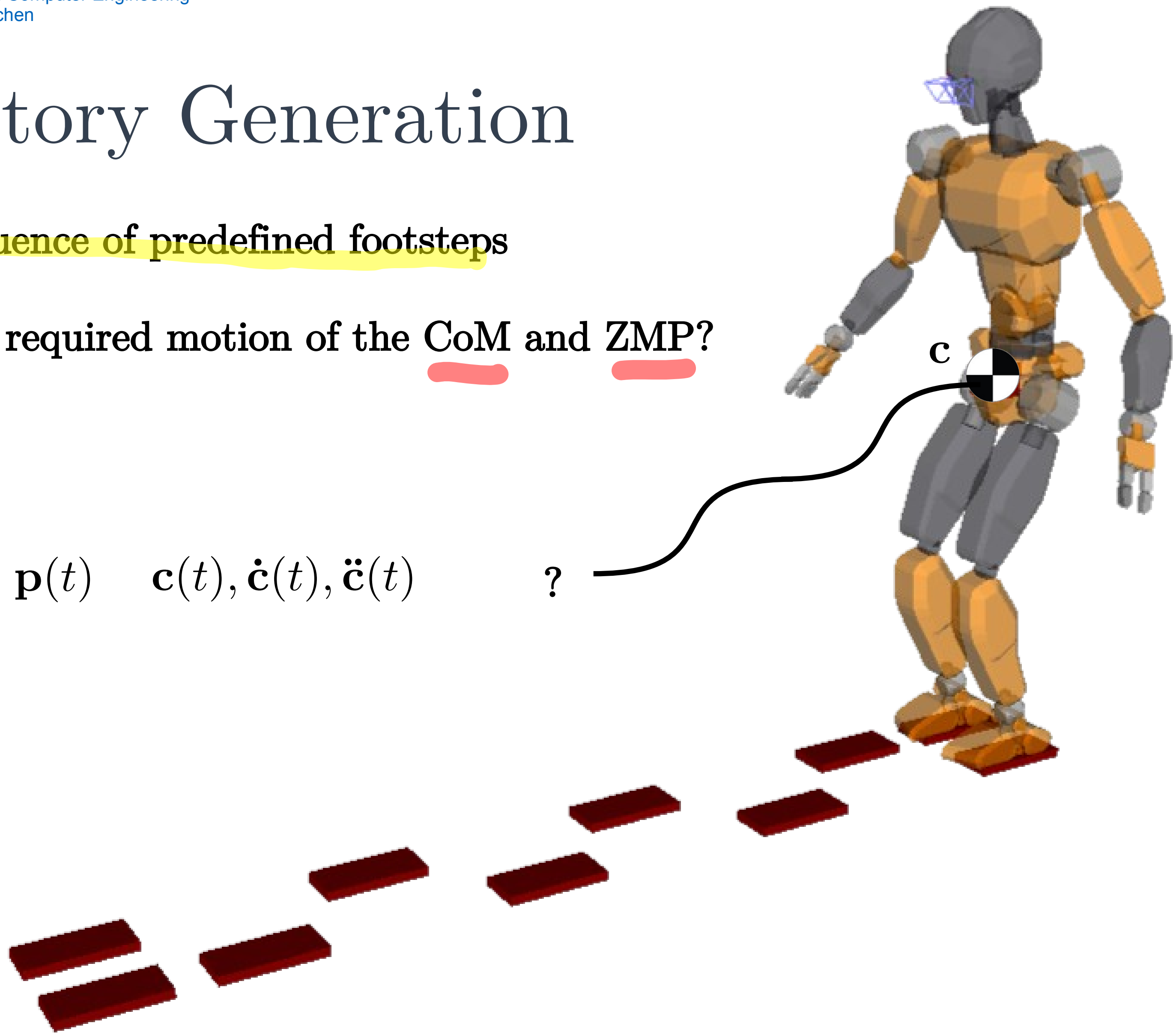


Trajectory Generation

Given a sequence of predefined footsteps

What is the required motion of the CoM and ZMP?

$\mathbf{p}(t) \quad \mathbf{c}(t), \dot{\mathbf{c}}(t), \ddot{\mathbf{c}}(t)$



Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Trajectory Generation

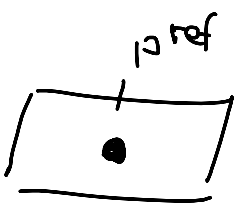
Given a sequence of predefined footsteps

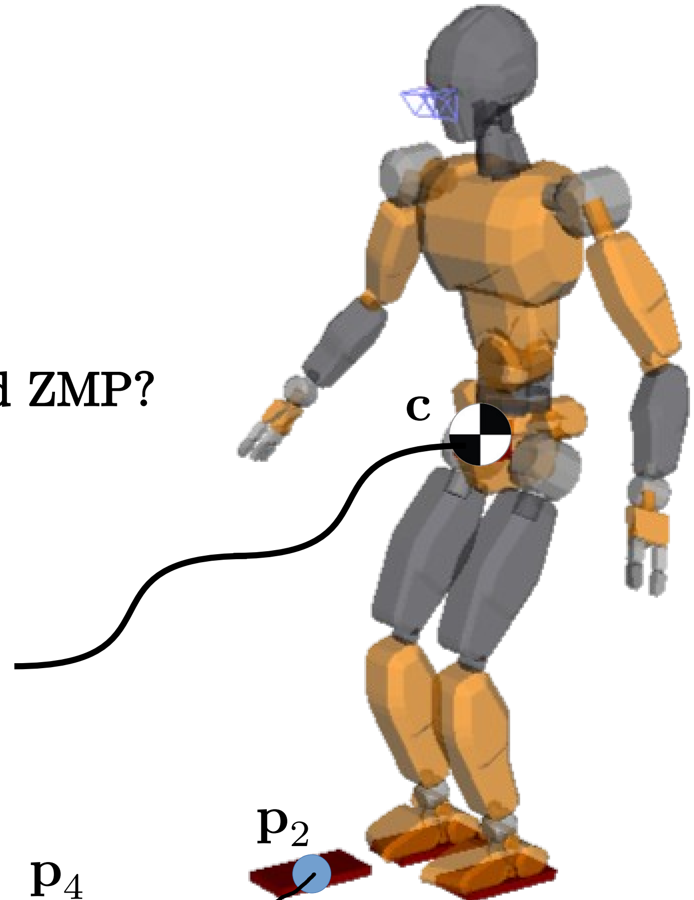
What is the required motion of the CoM and ZMP?

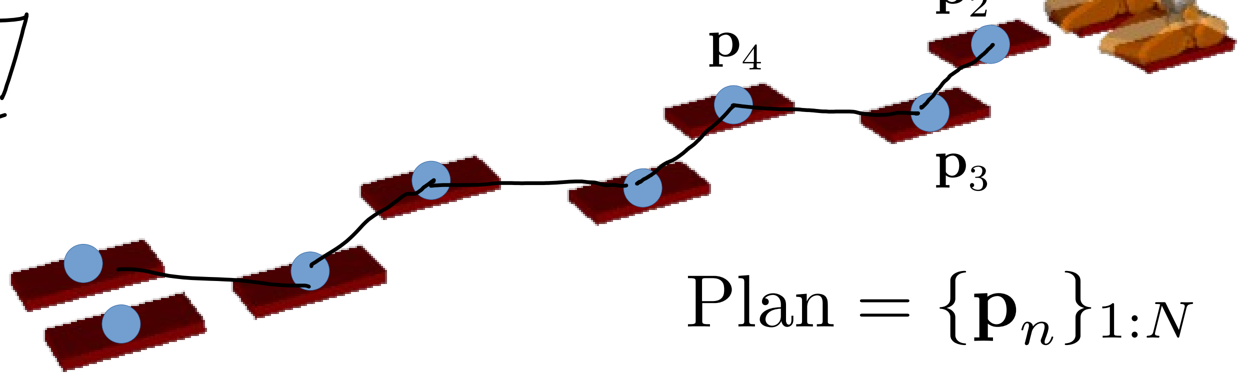
Remember:

ZMP needs to stay within Support polygon

→ ZMP reference are the footstep positions







$$\text{Plan} = \{\mathbf{p}_n\}_{1:N}$$

Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

ZMP Equation (sec. order eq.)

$$\ddot{c}_{xy} = \frac{g}{h}\left(c_{xy} - p_{xy}\right)$$

Dyn.

CoM pos

ZMP pos

Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

ZMP Equation (sec. order eq.)

$$\ddot{c}_{xy} = \frac{g}{h}\left(c_{xy} - p_{xy}\right)$$

ΔT step size

Can be analytically integrated in time to obtain:

pos

vel

$$\underbrace{\begin{bmatrix} c_{xy}^{+} \\ \dot{c}_{xy}^{+} \end{bmatrix}}_{\text{com next step}} = \begin{bmatrix} \cosh(\omega\Delta T) & \omega^{-1}\sinh(\omega\Delta T) \\ \omega\sinh(\omega\Delta T) & \cosh(\omega\Delta T) \end{bmatrix} \underbrace{\begin{bmatrix} c_{xy}^{-} \\ \dot{c}_{xy}^{-} \end{bmatrix}}_{\text{com prev. step.}} + \begin{bmatrix} 1 - \cosh(\omega\Delta T) \\ -\omega\sinh(\omega\Delta T) \end{bmatrix} p_{xy}$$

Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

ZMP Equation (sec. order eq.)

$$\ddot{c}_{xy} = \frac{g}{h}\left(c_{xy} - p_{xy}\right)$$

Can be analytically integrated in time to obtain:

$$\begin{bmatrix} c_{xy}^{+} \\ \dot{c}_{xy}^{+} \end{bmatrix} = \begin{bmatrix} \cosh\left(\omega \Delta T\right) & \omega^{-1}\sinh\left(\omega \Delta T\right) \\ \omega\sinh\left(\omega \Delta T\right) & \cosh\left(\omega \Delta T\right) \end{bmatrix} \begin{bmatrix} c_{xy}^{-} \\ \dot{c}_{xy}^{-} \end{bmatrix} + \begin{bmatrix} 1 - \cosh\left(\omega \Delta T\right) \\ -\omega\sinh\left(\omega \Delta T\right) \end{bmatrix} p_{xy}$$

Which gives the update: (with discretization step ΔT)

$$\mathbf{x}^{+} = \mathbf{A}\mathbf{x}^{-} + \mathbf{B}u$$

– input is zmp

com state

constan

Matrix

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

ZMP Equation (sec. order eq.)

$$\ddot{c}_{xy} = \frac{g}{h}\left(c_{xy} - p_{xy}\right)$$

Can be analytically integrated in time to obtain:

$$\begin{bmatrix} c_{xy}^{+} \\ \dot{c}_{xy}^{+} \end{bmatrix} = \begin{bmatrix} \cosh\left(\omega \Delta T\right) & \omega^{-1}\sinh\left(\omega \Delta T\right) \\ \omega\sinh\left(\omega \Delta T\right) & \cosh\left(\omega \Delta T\right) \end{bmatrix} \begin{bmatrix} c_{xy}^{-} \\ \dot{c}_{xy}^{-} \end{bmatrix} + \begin{bmatrix} 1 - \cosh\left(\omega \Delta T\right) \\ -\omega\sinh\left(\omega \Delta T\right) \end{bmatrix} p_{xy}$$

Which gives the update: (with discretization step ΔT)

$$\mathbf{x}^{+} = \mathbf{A}\mathbf{x}^{-} + \mathbf{B}u$$

Control input is ZMP

State CoM

Trajectory Generation

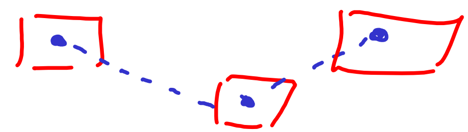
Can be formulated as Model Predictive Control (MPC)

Reference Trajectory

ZMP Positions: $\{\mathbf{p}_t\}_{1:T}$ are the footsteps centers (remain in Support Polygon)

CoM Positions: $\{\mathbf{c}_t\}_{1:T}$ (if known)

CoM Velocity: $\{\dot{\mathbf{c}}_t\}_{1:T}$ (if known)



Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

Horizon length

Problem:

$$\min_{\mathbf{c}[],\dot{\mathbf{c}}[],\mathbf{p}[]} \sum_{k}^{K} \underbrace{\frac{\beta}{2}\left\|c_k - c_k^{ref}\right\|^2}_{\text{com pos}} + \underbrace{\frac{\gamma}{2}\left\|\dot{c}_k - \dot{c}_k^{ref}\right\|^2}_{\text{com vel}} + \underbrace{\frac{\alpha}{2}\left\|u_k - u_k^{ref}\right\|^2}_{\text{zmp pos}}$$

subject to

α, β, γ are user defined gains

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

Problem:

$$\min_{\mathbf{c}[],\dot{\mathbf{c}}[],\mathbf{p}[]} \sum_k \frac{\beta}{2} \left\| c_k - c_k^{ref} \right\|^2 + \frac{\gamma}{2} \left\| \dot{c}_k - \dot{c}_k^{ref} \right\|^2 + \frac{\alpha}{2} \left\| u_k - u_k^{ref} \right\|^2$$

subject to

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$ Equality: Respect the discretized Dynamics

α, β, γ are user defined gains

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

Problem:

$$\min_{\mathbf{c}[],\dot{\mathbf{c}}[],\mathbf{p}[]} \sum_k \frac{\beta}{2} \left\| c_k - c_k^{ref} \right\|^2 + \frac{\gamma}{2} \left\| \dot{c}_k - \dot{c}_k^{ref} \right\|^2 + \frac{\alpha}{2} \left\| u_k - u_k^{ref} \right\|^2$$

subject to

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

Equality: Respect the discretized Dynamics

$$\mathbf{x}_0 = \mathbf{x}_{\text{init}}$$
$$\mathbf{x}_N = \mathbf{x}_{\text{terminal}}$$

Initial and terminal conditions

α, β, γ are user defined gains

Trajectory Generation

Can be formulated as Model Predictive Control (MPC)

Problem:

$$\min_{\mathbf{c}[],\dot{\mathbf{c}}[],\mathbf{p}[]} \sum_k \frac{\beta}{2} \left\| c_k - c_k^{ref} \right\|^2 + \frac{\gamma}{2} \left\| \dot{c}_k - \dot{c}_k^{ref} \right\|^2 + \frac{\alpha}{2} \left\| u_k - u_k^{ref} \right\|^2$$

subject to

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

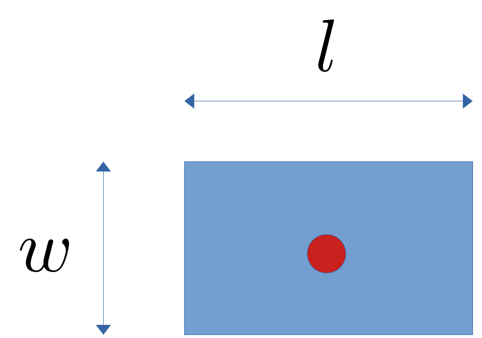
$$\mathbf{x}_0 = \mathbf{x}_{\text{init}}$$

$$\mathbf{x}_N = \mathbf{x}_{\text{terminal}}$$

known

$$\underbrace{p_k - \frac{s}{2}}_{\text{known}} < u_k < p_k + \frac{s}{2}$$

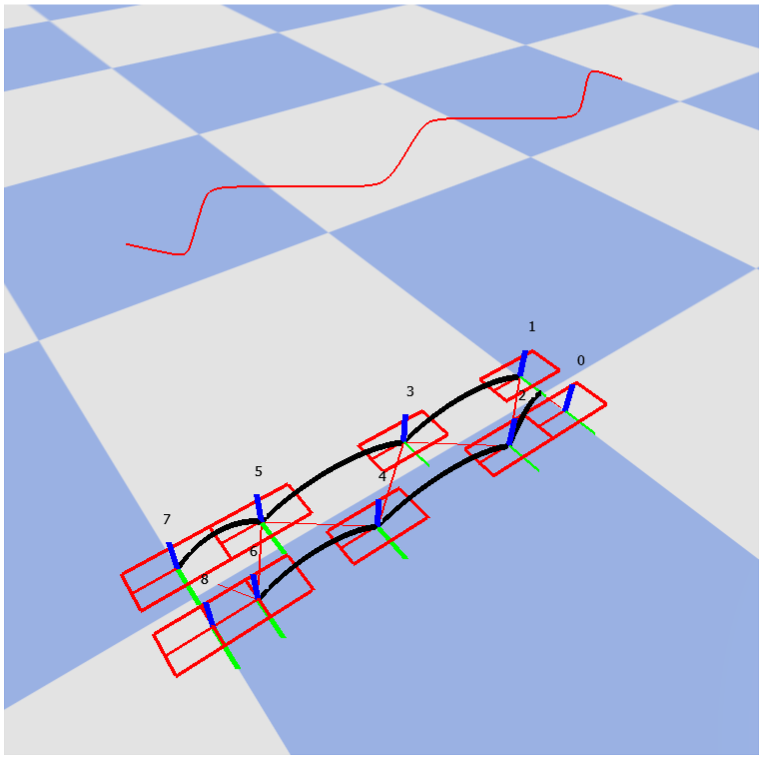
α, β, γ are user defined gains

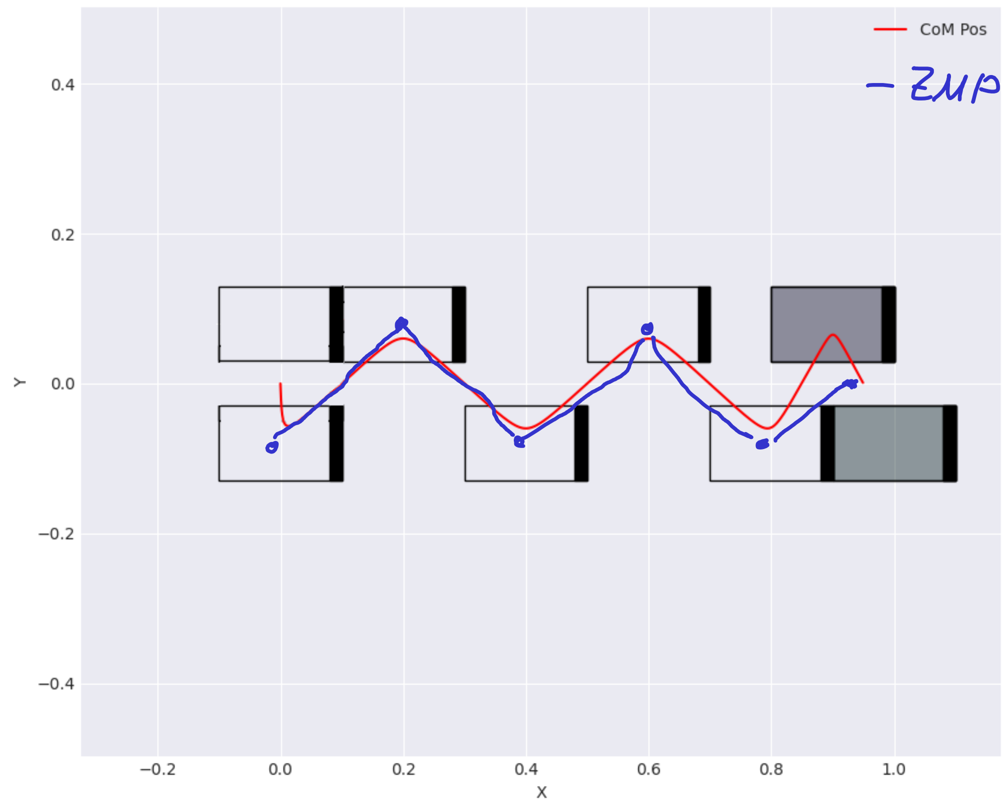


$$s = \{w, l\}$$

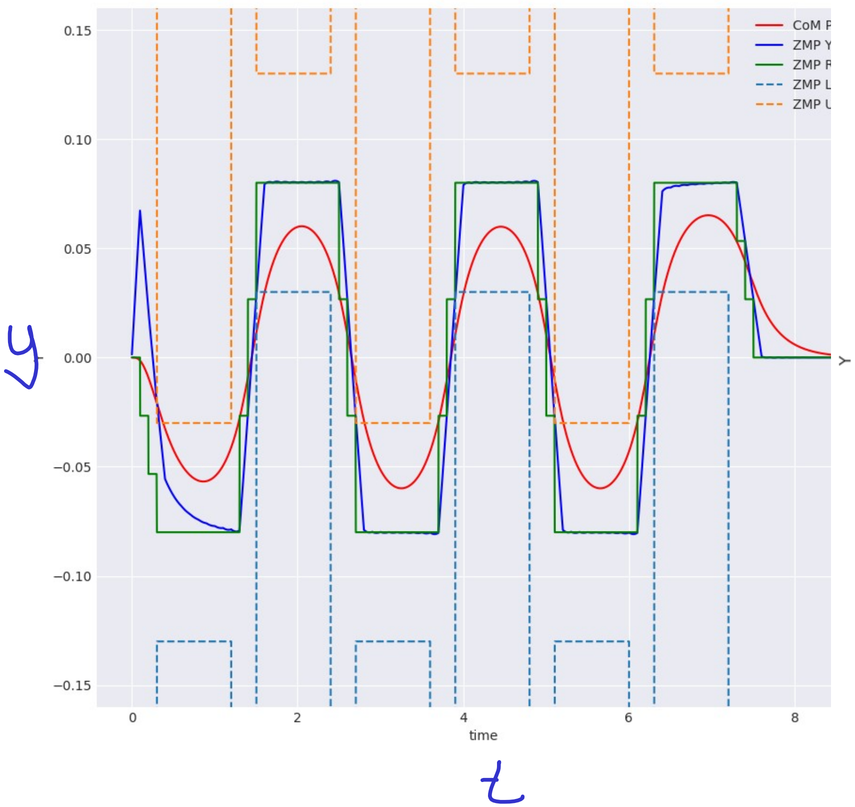
Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

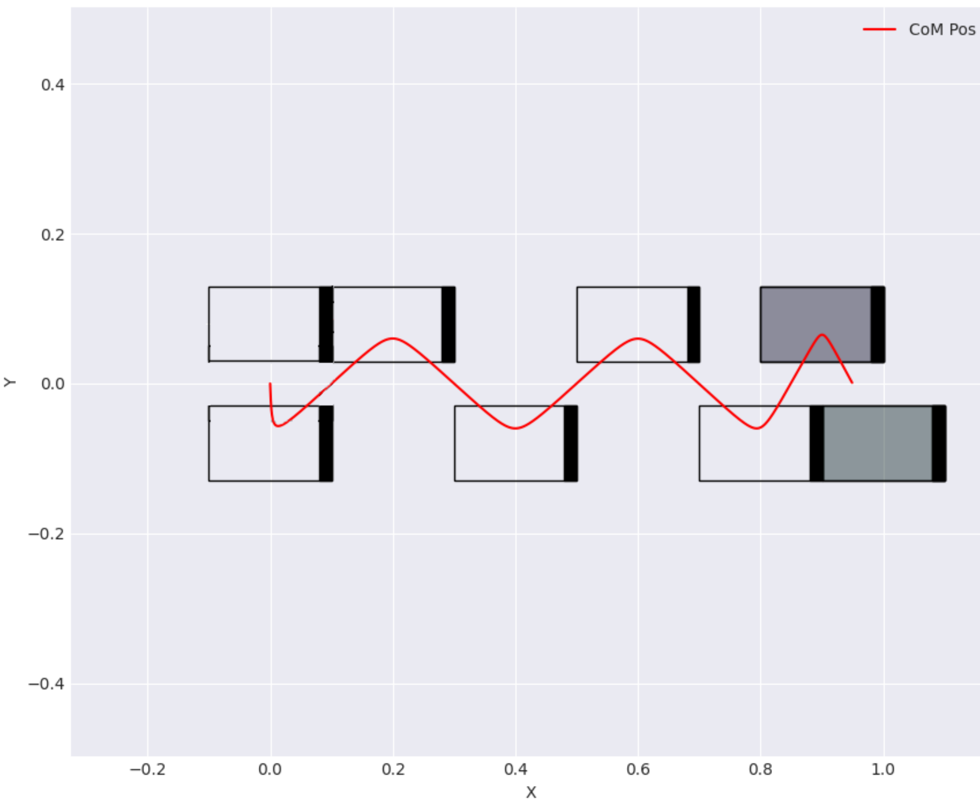
Trajectory Generation





Trajectory Generation





Continious Sys:

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

Solver: Discritized

$$x_{k+1} = x_k + \delta t \, \dot{x}_k$$

$$x_{k+1} = x_k + \delta t \, f(x_k, u_k)$$

Equality constrain
to inforce dynamics