cartesian_controllers: Motion, Force and Compliance Control for Robotic Manipulators

FZI Forschungszentrum Informatik, Karlsruhe

Arne Roennau, Stefan Scherzinger

Kompetenzzentrum IT-Sicherheit

DIZ DIGITALES INNOVATIONS ZENTRUM

Agenda

1. Motivation: Why Cartesian Control?
2. Control Approach
3. Applications
4. Outlook
5. Conclusion

Why Cartesian Control in Task Space?

FZI

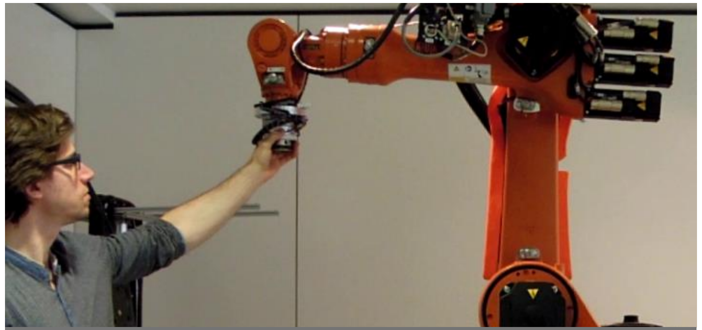
Closed loop force control

Direct teaching

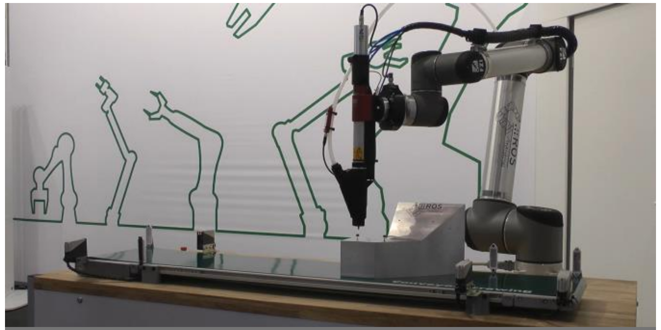
Contact-rich manipulation

…

Tele-operation



Manual guidance



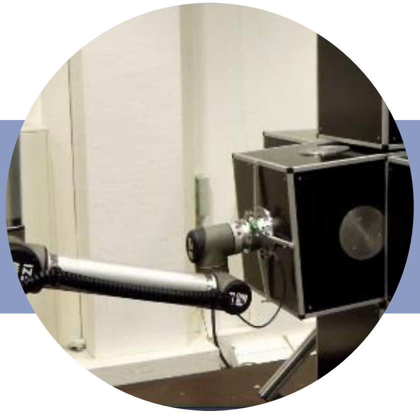
Dynamic target tracking

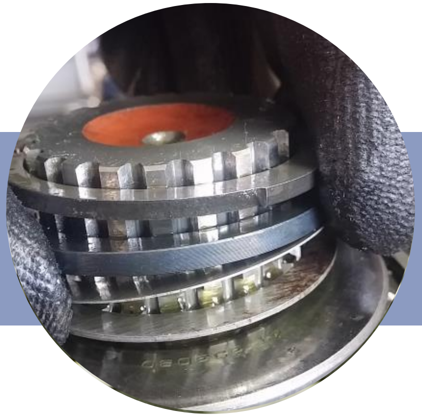
⋮

ROS Control

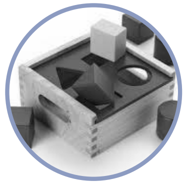
Use ROS Control with the *cartesian_controllers*

Challenges with Industrial Robots





Often contact dominated tasks



Autonomous skills

Dealing with uncertainty

Force-sensitive interaction

◄ **Goals** ►

Robot independant

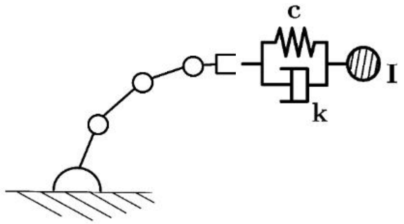
Formulation in task space

Let's go for more… Active Cartesian Compliance

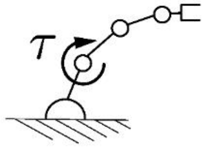
FZI

Active compliance

$$\mathbf{F} = \mathbf{c}(\mathbf{x}^d - \mathbf{x}) + \mathbf{k}(\dot{\mathbf{x}}^d - \dot{\mathbf{x}}) + \mathbf{I}(\ddot{\mathbf{x}}^d - \ddot{\mathbf{x}})$$




Impedance:

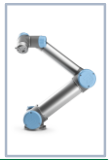


$\mathbf{x}(t) \rightarrow \mathbf{F}(t) \quad \tau(t) = \mathbf{J}^T \mathbf{F}(t)$

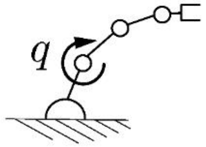
- Torque-actuated robot joints
- Joint-torque sensors







Admittance:



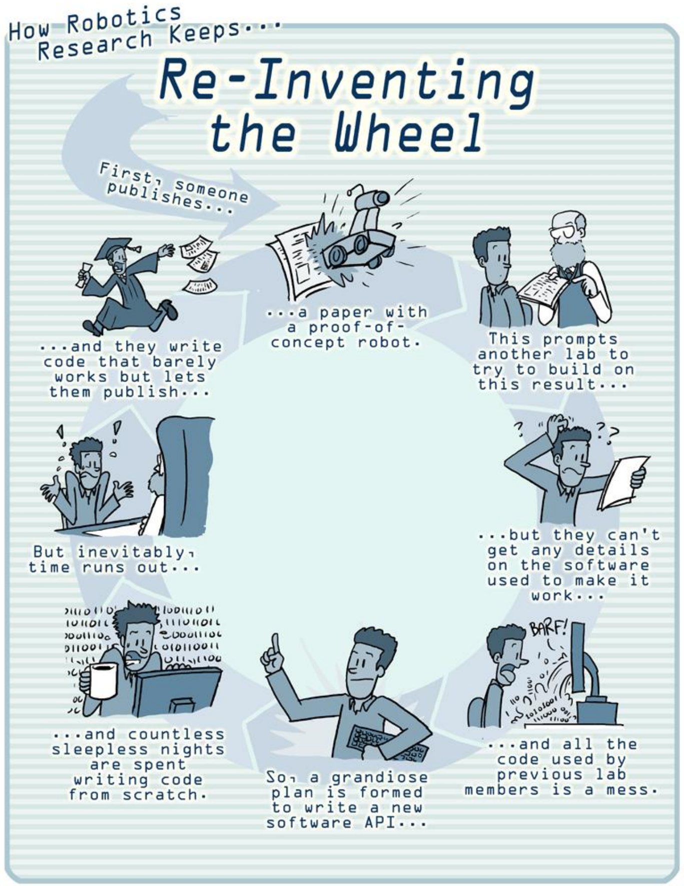
$\mathbf{F}(t) \rightarrow \mathbf{x}(t) \quad \Delta\mathbf{q} = \mathbf{J}^{-1} \Delta\mathbf{x}$

- Motion-actuated robot joints
- End effector force-torque sensor

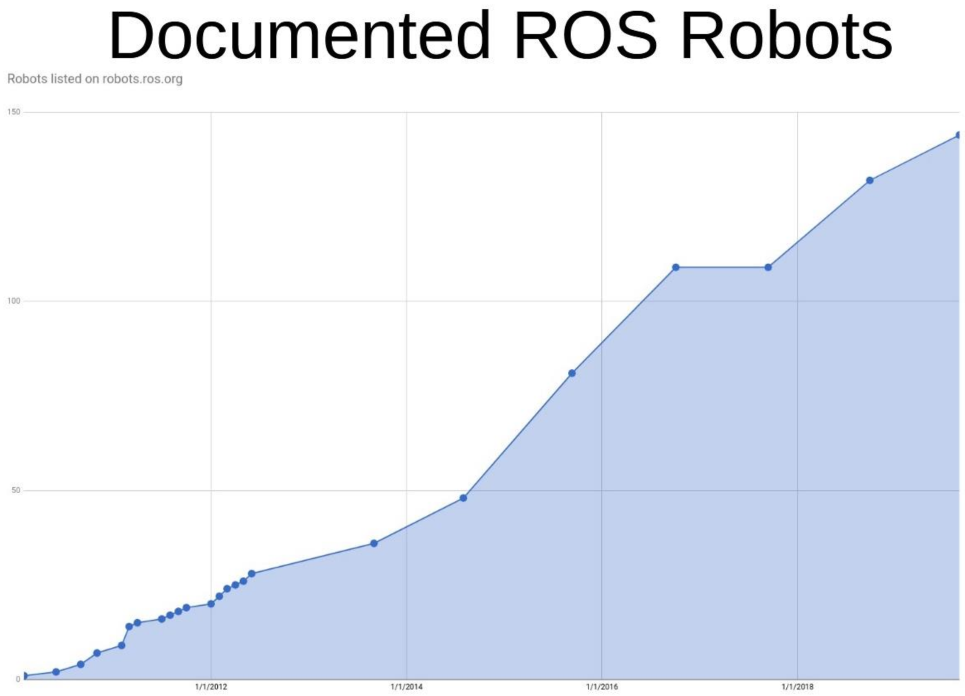
Agenda

Motivation: Why Cartesian Control?	1	4	Outlook
Control Approach	2	5	Conclusion
Applications	3		

The ROS framework… reuse it! make it available!



willowgarage.com



The number of different types of robots available to the community with ROS drivers.

Source: Ken Conley, Tully Foote, wiki.ros.org/Robots, 2017 changed over to robots.ros.org

wiki.ros.org/Metrics

The setting within ROS Control

FZI

You have:

- Joint position/velocity streaming interface
- ROS control HW abstraction
- Library of flexible controllers
- Applications with intuitive end effector control

Robot

⋮

↔

ROS control HW abstraction

- VelocityJointInterface
- PositionJointInterface
- EffortJointInterface

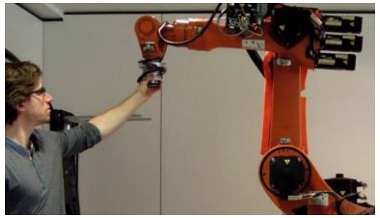
↔

cartesian_controllers

- cartesian_motion_controller
- cartesian_force_controller
- cartesian_compliance_controller
- cartesian_controller_handle

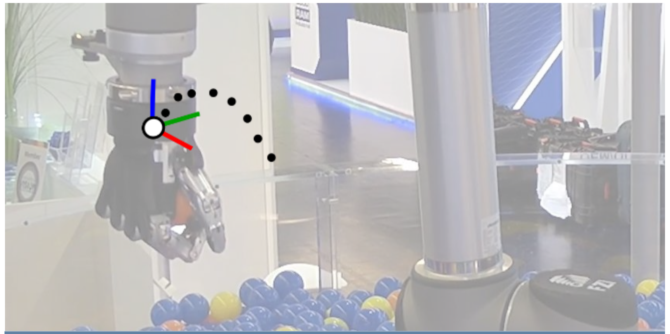
↔

User application



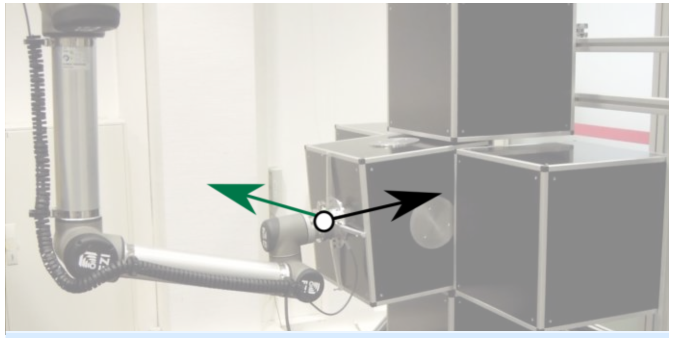
⋮

Three main controllers in *cartesian_controllers*



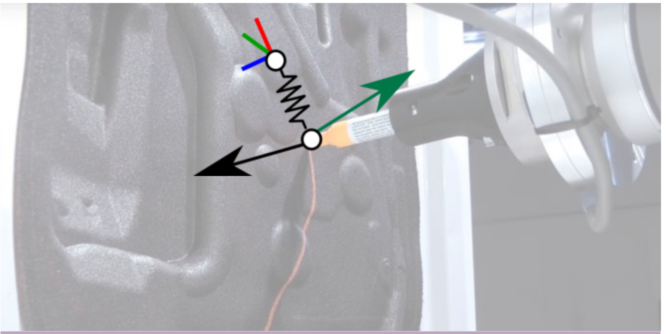
cartesian_motion_controller

- You want to follow a moving target
- The targets might be sparsely sampled
- You prefer smoothness over accuracy



cartesian_force_controller

- You want to control the robot with a wrench in contacts
- You have a wrist FT sensor



cartesian_compliance_controller

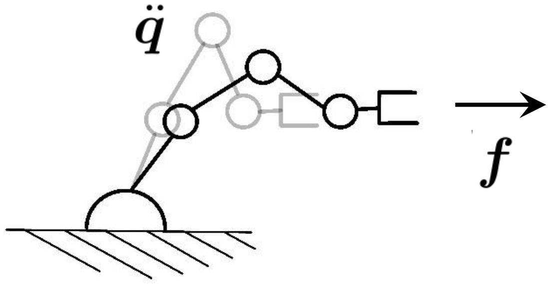
- You want to follow a moving target
- You want to react to external disturbances
- You have a wrist FT sensor

Our Approach

- Instantaneous joint motion from Cartesian error

Rigid body dynamics

$$\boldsymbol{\tau} = \boldsymbol{H}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{G}(\boldsymbol{q})$$



Simplification

$$\ddot{\boldsymbol{q}} = \boldsymbol{H}^{-1}(\boldsymbol{J}^T \boldsymbol{f} - \underbrace{\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})}_{=\mathbf{0}} - \underbrace{\boldsymbol{G}(\boldsymbol{q})}_{=\mathbf{0}})$$

↓

Virtual dynamics

$$\ddot{\boldsymbol{q}} = \boldsymbol{H}^{-1}\boldsymbol{J}^T \boldsymbol{f}$$

Admittance:

- Directly to joint space!
- Common wrench interface

Our Approach

- Instantaneous joint motion from Cartesian error

Virtual plant dynamics

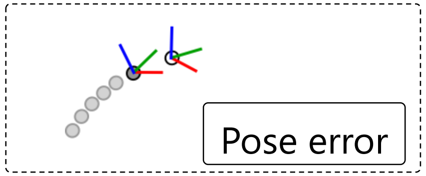
$$\ddot{q} = \boldsymbol{H}^{-1}\boldsymbol{J}^T\boldsymbol{f}$$

↓

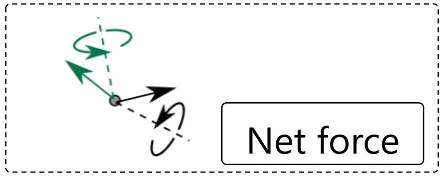
Cartesian error

$$\epsilon =$$

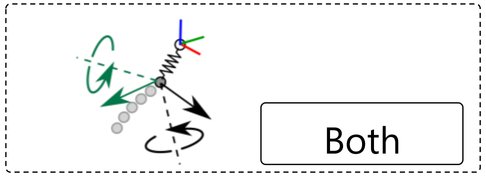
cartesian_motion_controller



cartesian_force_controller



cartesian_compliance_controller



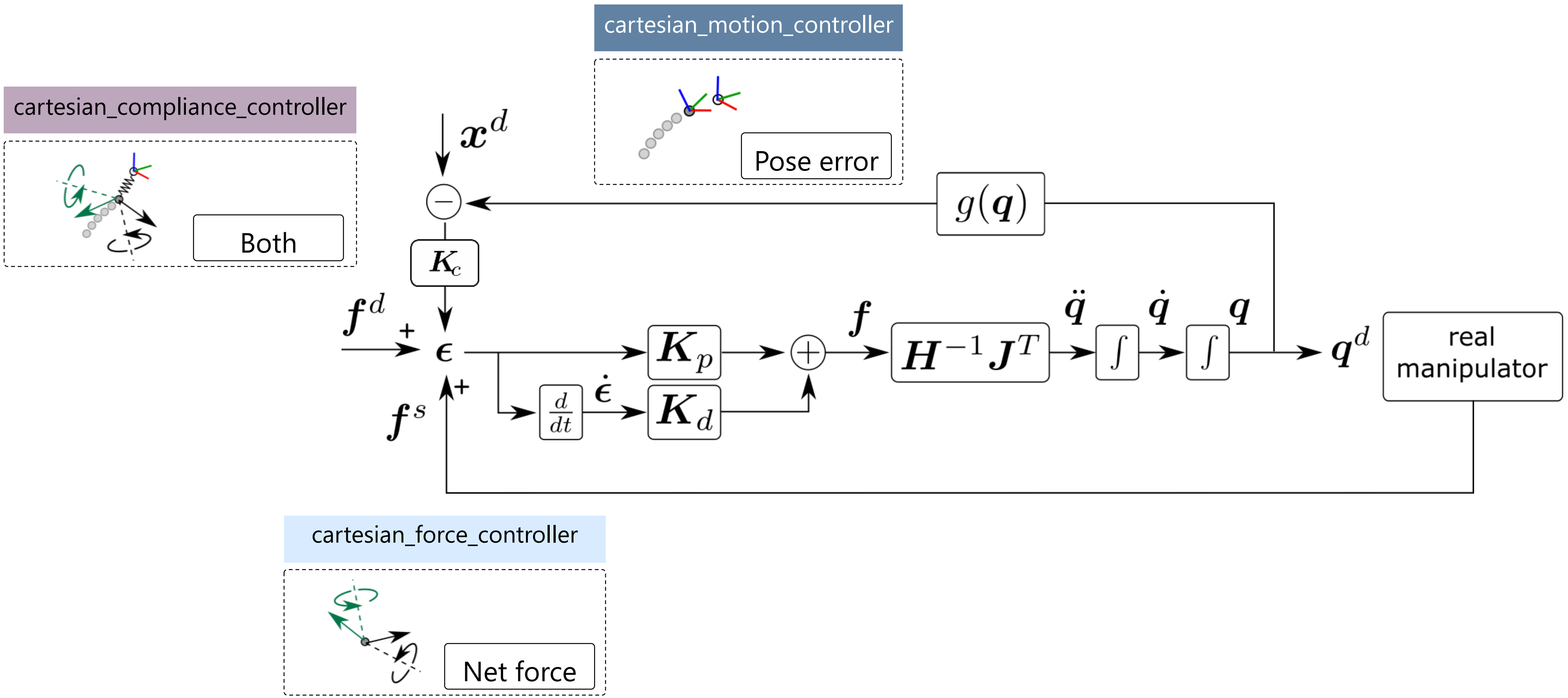
Wrench representation

$$\boldsymbol{f} = \boldsymbol{K}_p\epsilon + \boldsymbol{K}_d\dot{\epsilon}$$

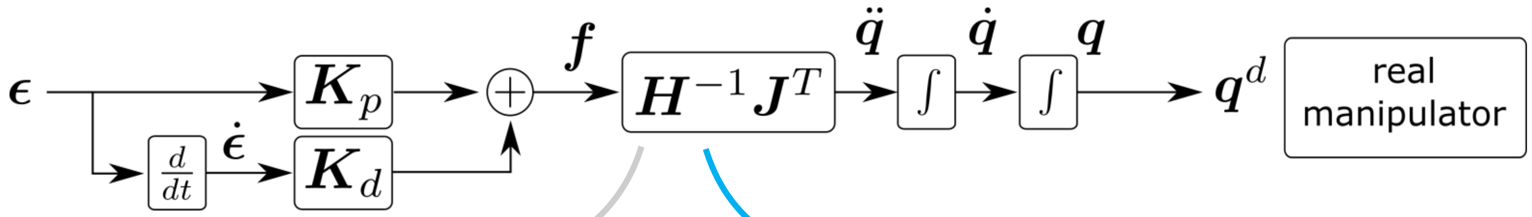
Instantaneous acceleration

$$\ddot{q} = \boldsymbol{H}^{-1}\boldsymbol{J}^T(\boldsymbol{K}_p\epsilon + \boldsymbol{K}_d\dot{\epsilon})$$

The control loop



The control loop



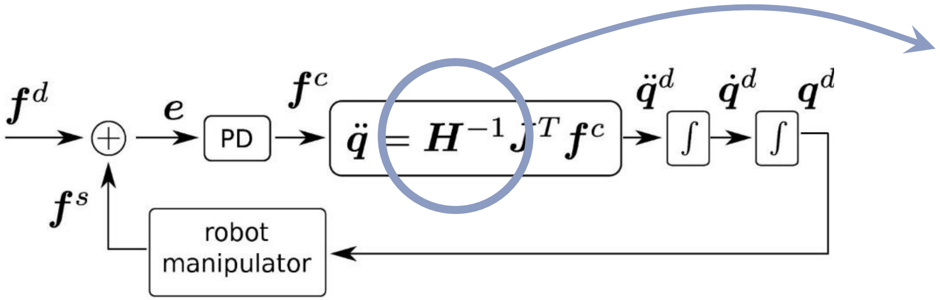
Real dynamics

- Hard to obtain
- No benefit on motion-actuated robots!

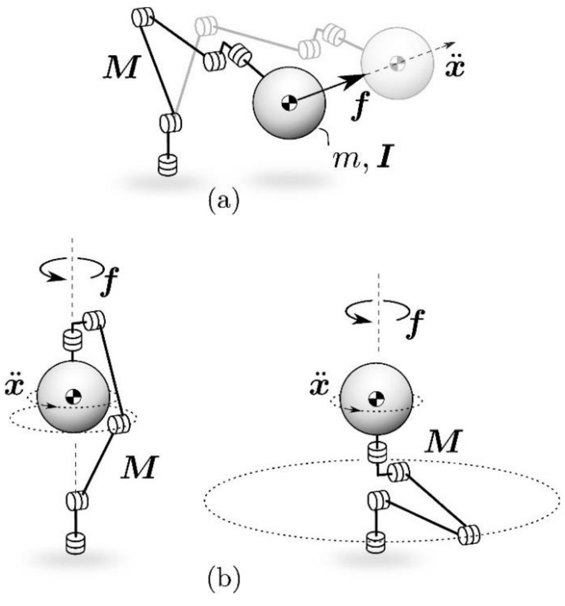
Virtual (unrealistic) dynamics

- As long as it solves IK …
- Is there something better than realistic values?

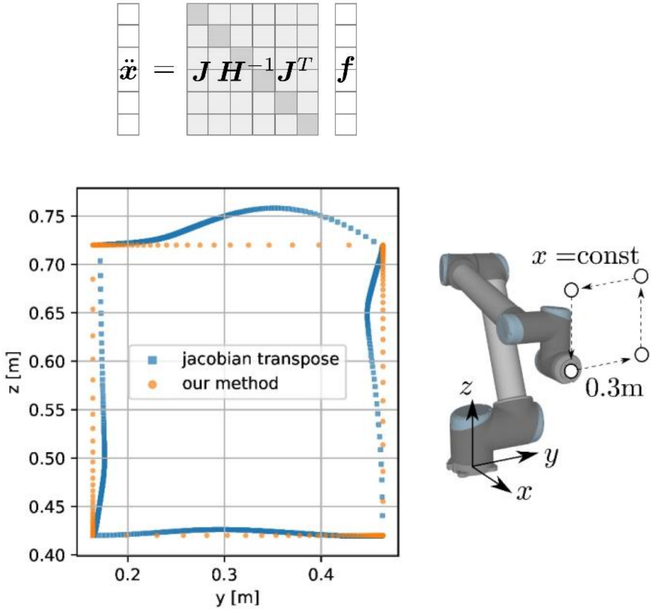
Task space linearization



Closed-Loop control scheme
(on dynamics-conditioned twin)

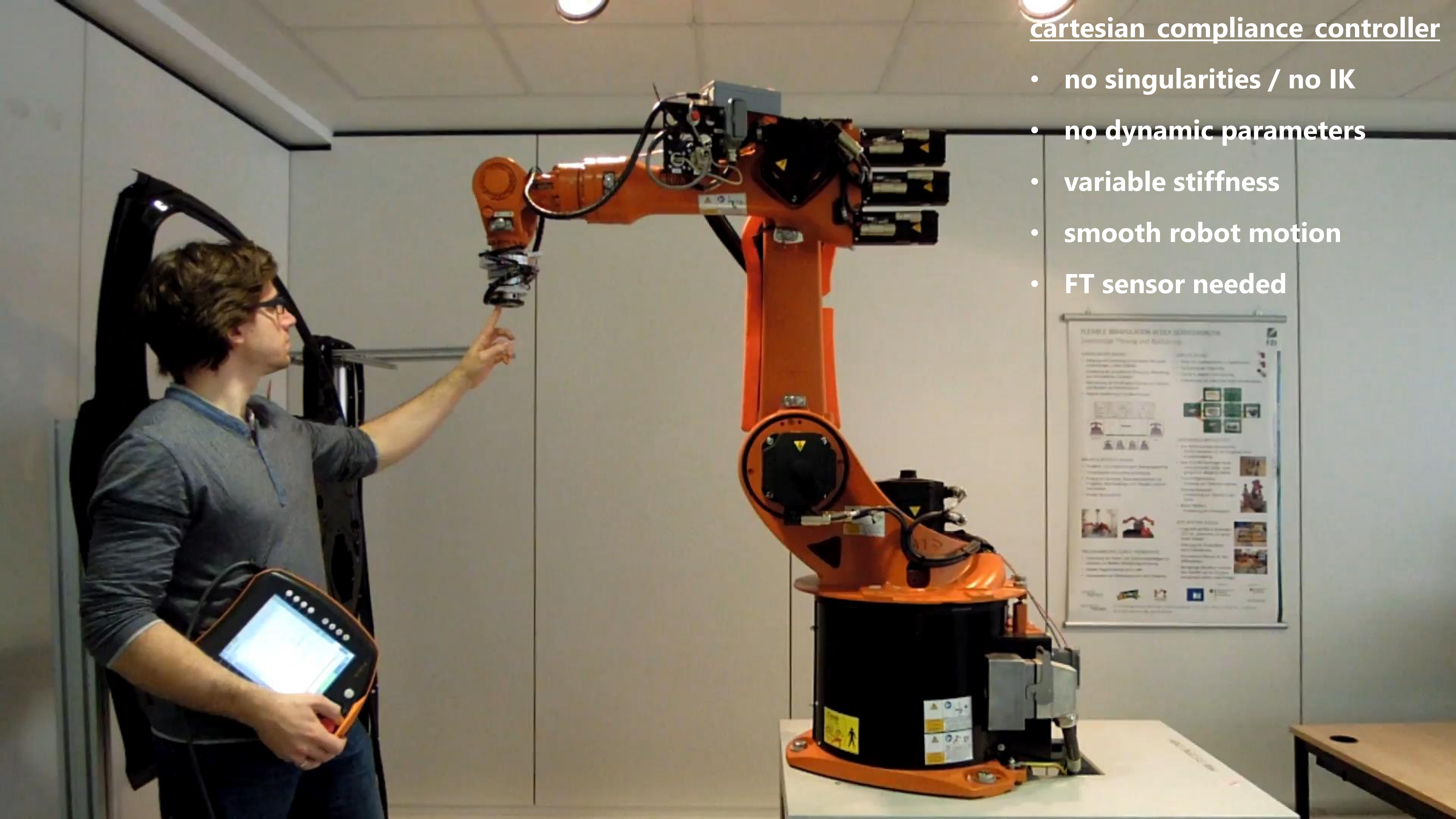


Dynamics-conditioned twin



Scherzinger et al, *Inverse Kinematics with Forward Dynamics Solvers for Sampled Motion Tracking*, IEEE ICAR 2019 (to appear)


cartesian force controller
• no singularities / no IK
• no dynamic parameters
• smooth robot motion
• FT sensor needed



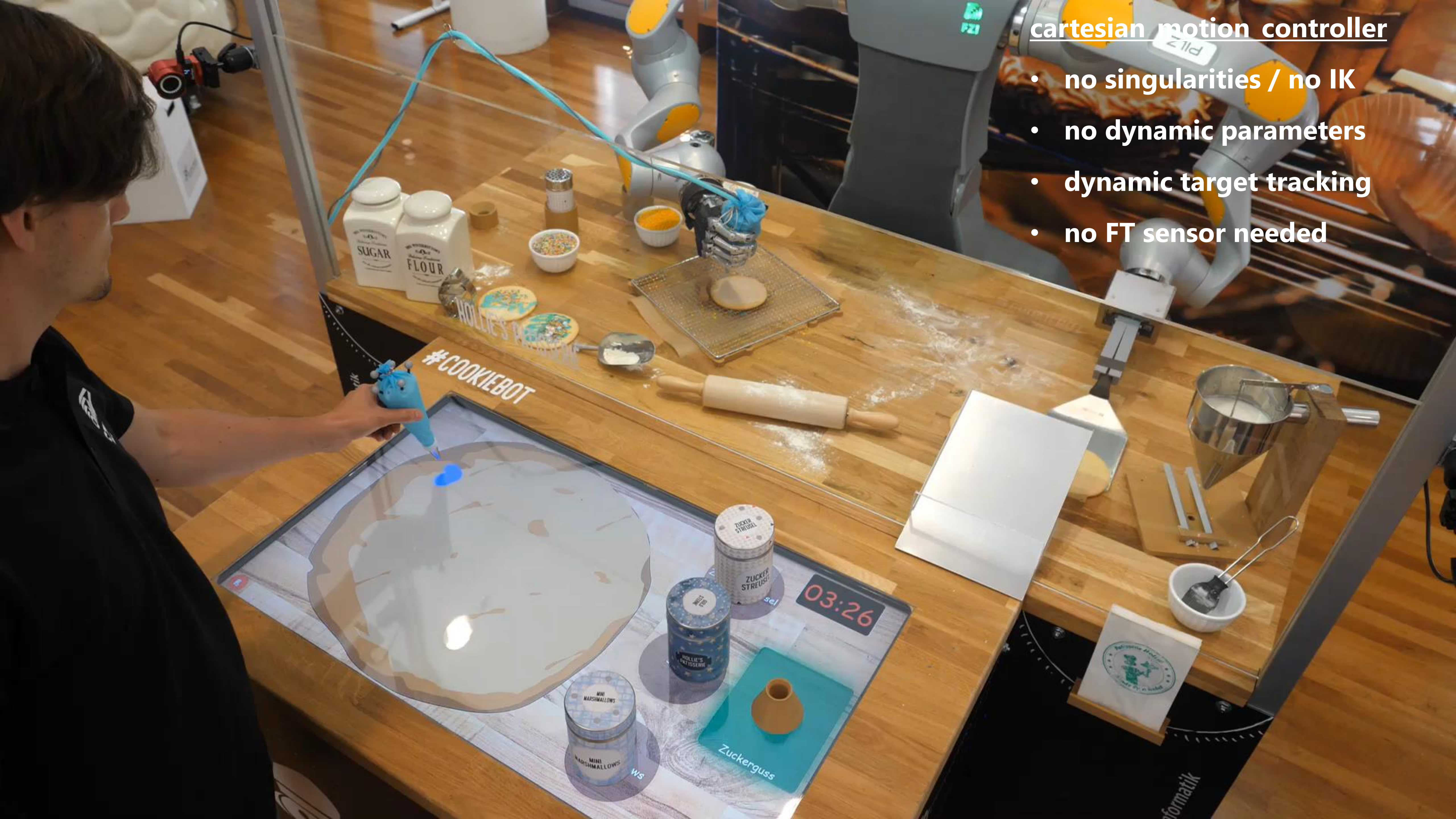
cartesian_compliance_controller

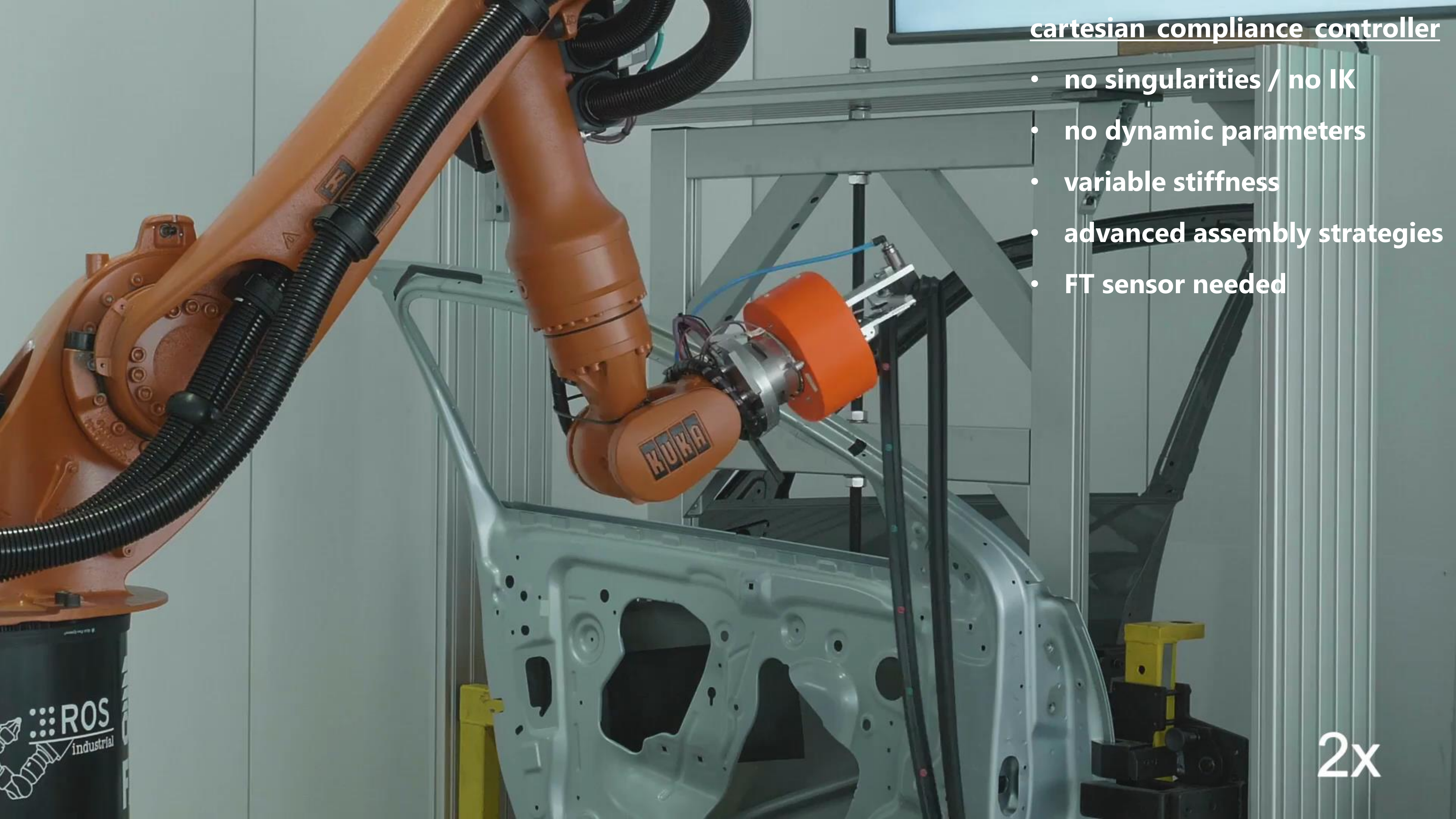
- no singularities / no IK
- no dynamic parameters
- variable stiffness
- smooth robot motion
- FT sensor needed


cartesian motion controller
• no singularities / no IK
• no dynamic parameters
• linear cartesian motions
• dynamic target tracking
• no FT sensor needed
KUKA
Powered by FZI
SCHUNK
2x

Agenda

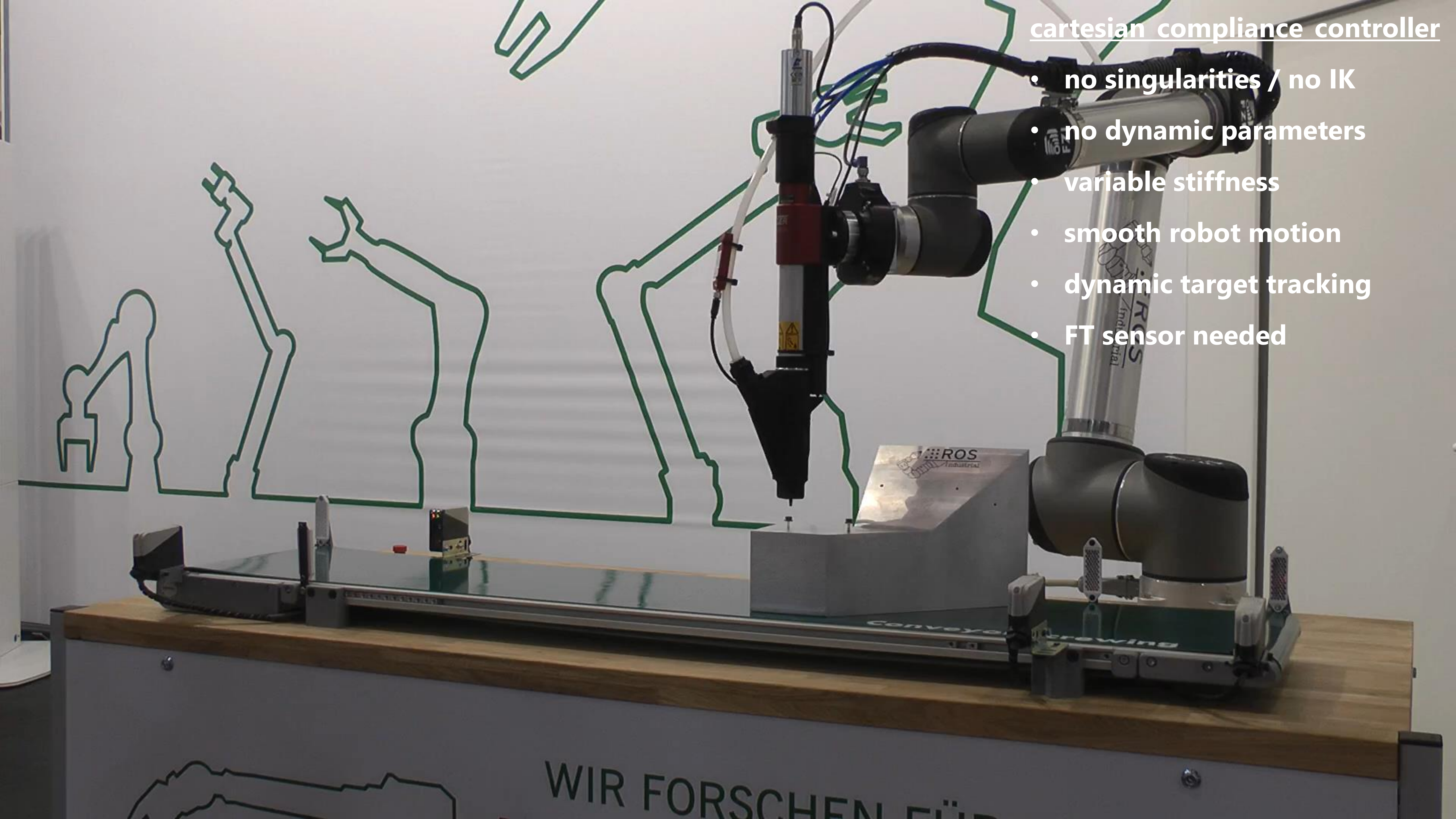
Motivation: Why Cartesian Control?	1	4	Outlook
Control Approach	2	5	Conclusion
Applications	3		

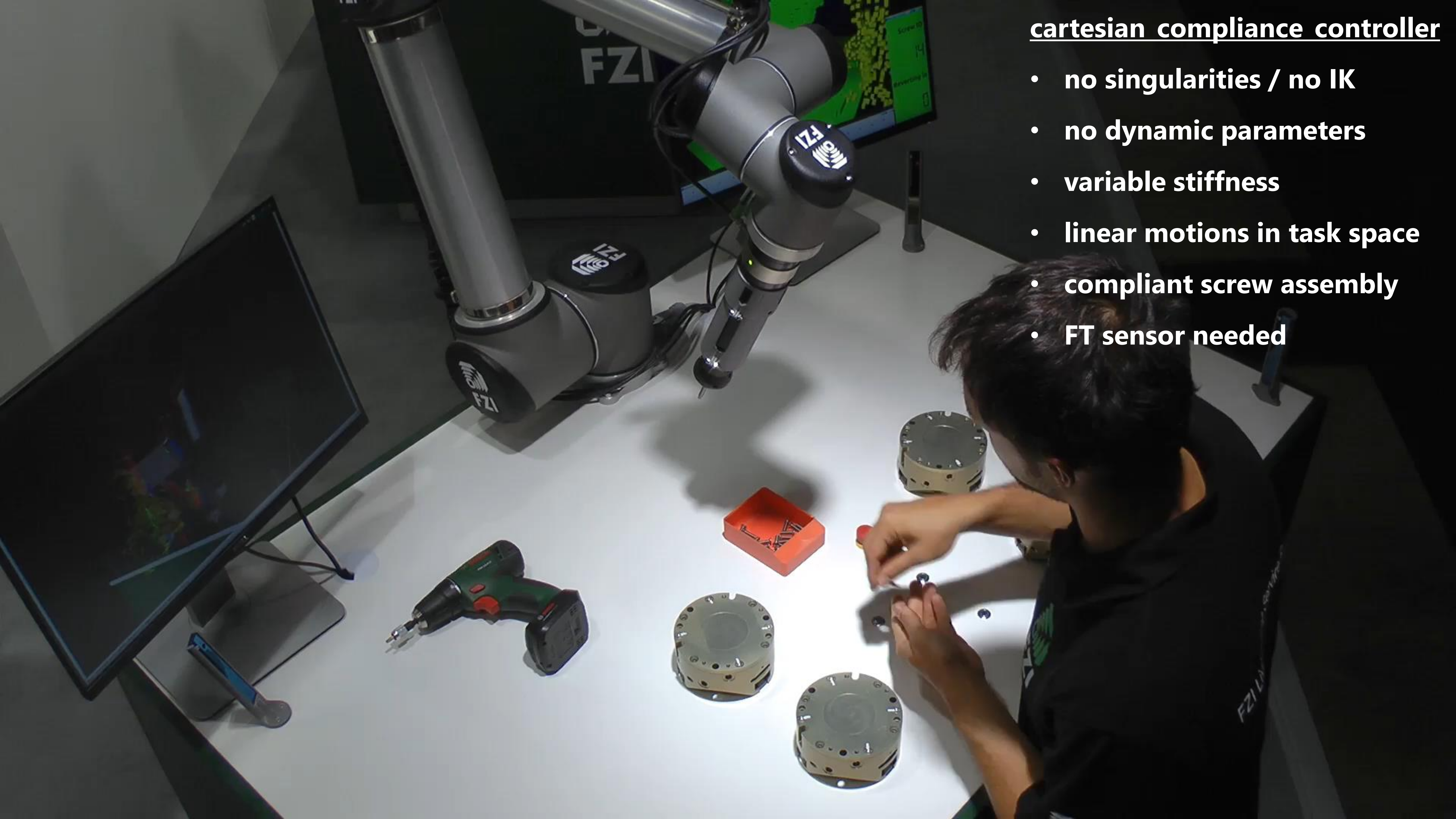

cartesian motion controller
• no singularities / no IK
• no dynamic parameters
• dynamic target tracking
• no FT sensor needed
SUGAR
FLOUR
#COOKIEBOT
03:26
Zuckerguss


cartesian compliance controller
• no singularities / no IK
• no dynamic parameters
• variable stiffness
• advanced assembly strategies
• FT sensor needed
2x
KUKA
ROS
industrial

cartesian_compliance_controller

- no singularities / no IK
- no dynamic parameters
- variable stiffness
- smooth robot motion
- dynamic target tracking
- FT sensor needed





<u>cartesian_compliance_controller</u>

- no singularities / no IK
- no dynamic parameters
- variable stiffness
- linear motions in task space
- compliant screw assembly
- FT sensor needed

Agenda

1 — Motivation: Why Cartesian Control?

2 — Control Approach

3 — Applications

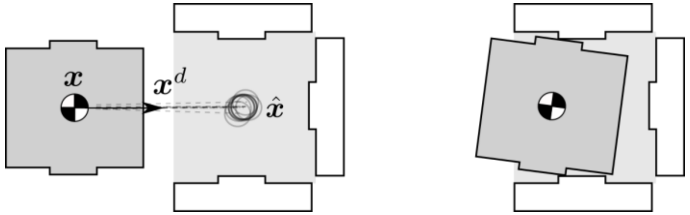
4 — Outlook

5 — Conclusion

Satellite assembly

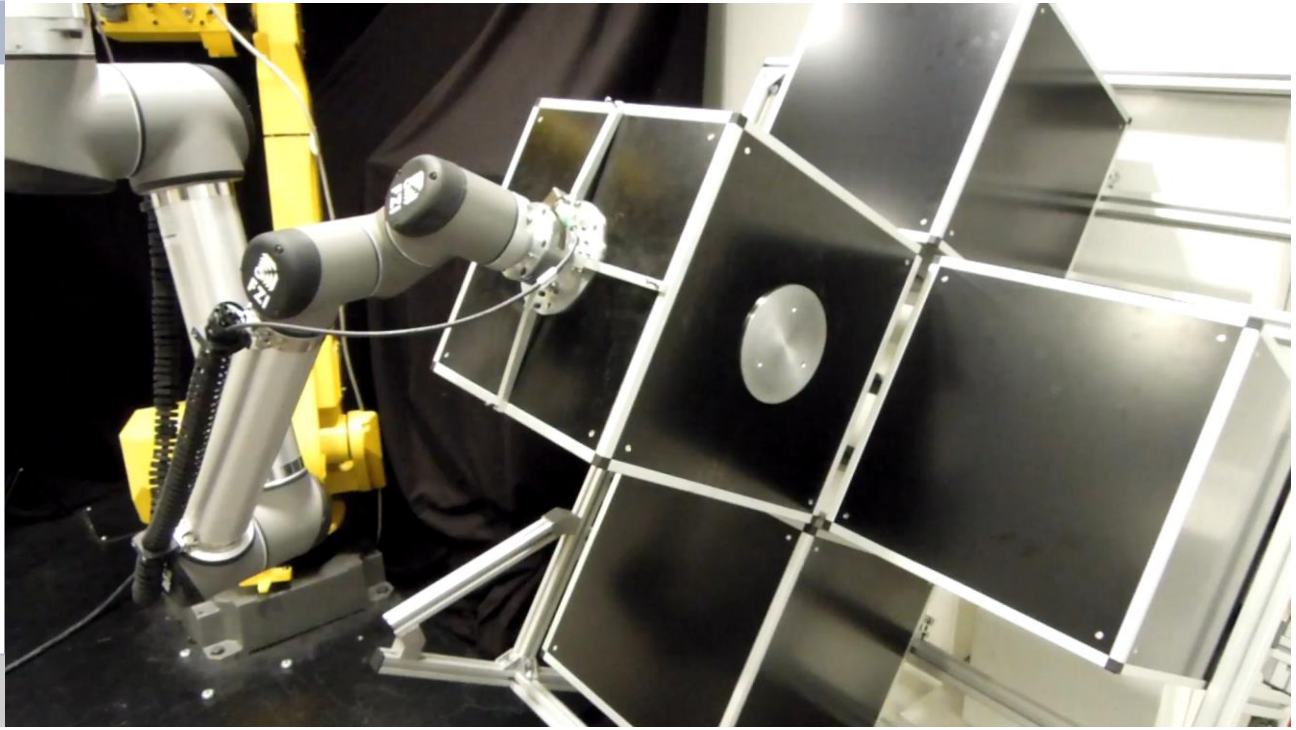
INDUSTRIAL SETTINGS

- Contact-dominated assembly tasks
- Object poses with uncertainty
- More relevant for complex insertion



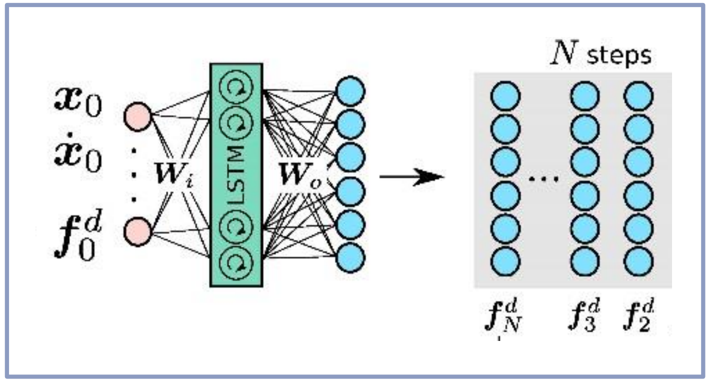
GOALS

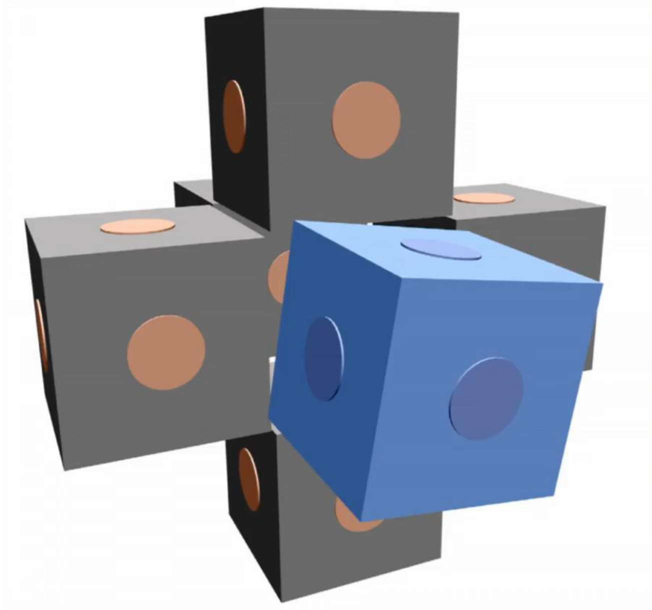
- Error correcting contact skills for autonomous execution that are transferable to different robots



Approach and Methods

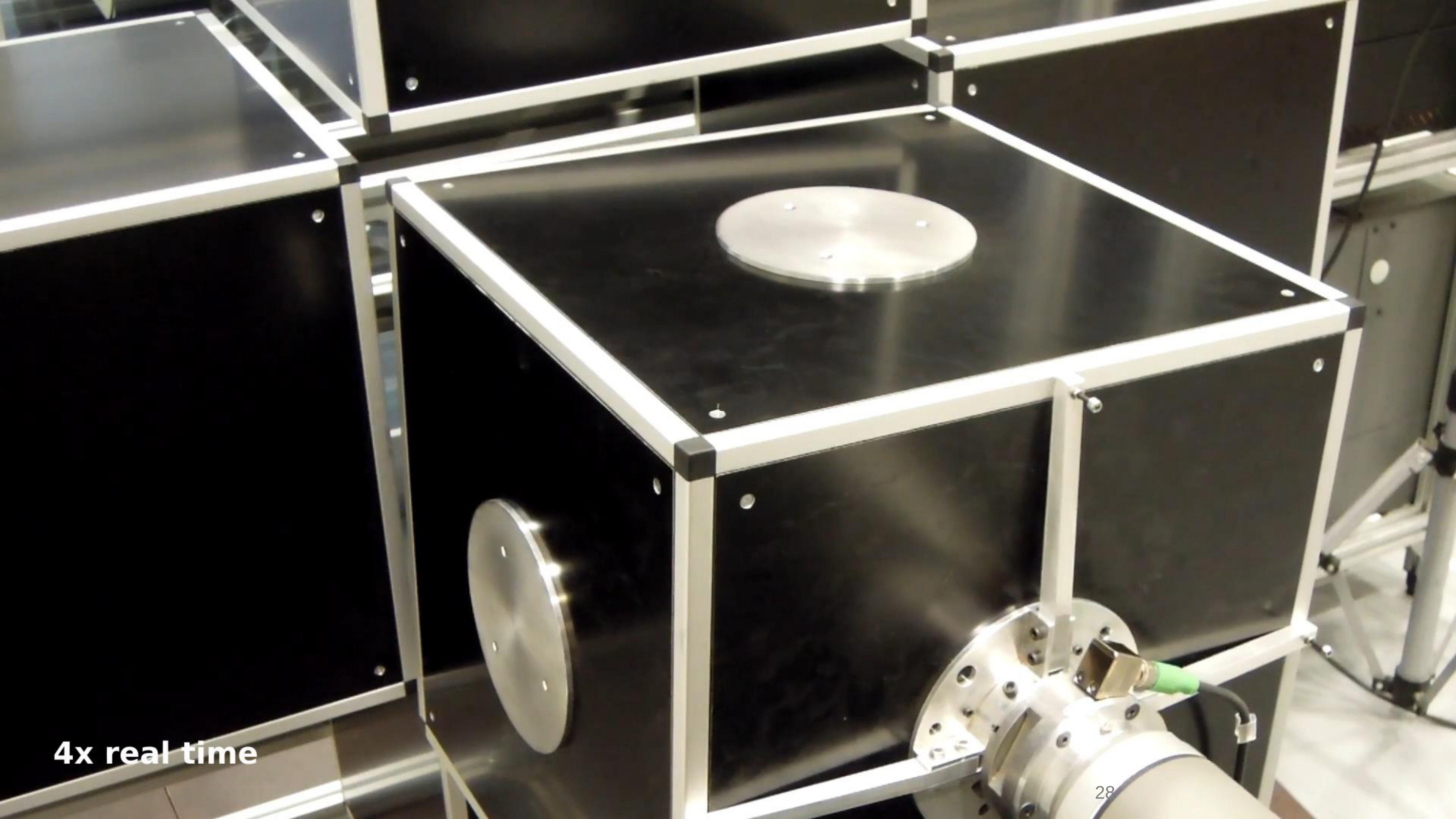
► Contact skill extraction in simulation





► Let humans solve tilting and jamming for challenging configurations

Scherzinger et al, *Contact Skill Imitation Learning for Robot-Independent Assembly Programming*, IEEE IROS 2019


4x real time

Agenda

1 — Motivation: Why Cartesian Control?
2 — Control Approach
3 — Applications
4 — Outlook
5 — Conclusion

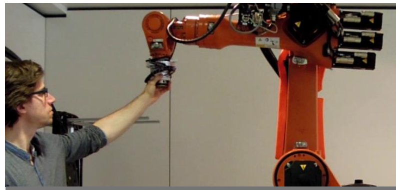
Summary

Control your robots in Cartesian task space!!!

Baseline
- Joint position/velocity streaming interface

Goal
- Application with fast, direct, task space control


Task space control


Add-on compliance


Closed loop force control


Object manipulation

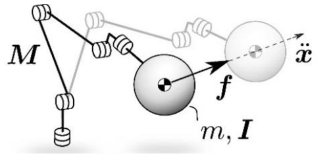
Available as OpenSource!

github.com/fzi-forschungszentrum-informatik/cartesian_controllers

ROS Control

cartesian_controllers

- cartesian_motion_controller
- cartesian_force_controller
- cartesian_compliance_controller



More details here:

Scherzinger et al., "Forward Dynamics Compliance Control: A new approach to cartesian compliance for robotic manipulators." *IEEE IROS, 2017.*

Scherzinger et al., “Inverse Kinematics with Forward Dynamics Solvers for Sampled Motion Tracking.” *IEEE ICAR, 2019.*

Scherzinger et al., “Contact Skill Imitation Learning for Robot-Independent Assembly Programming.”, *IEEE IROS, 2019.*

Further questions?...Ask us!… roennau@fzi.de scherzin@fzi.de