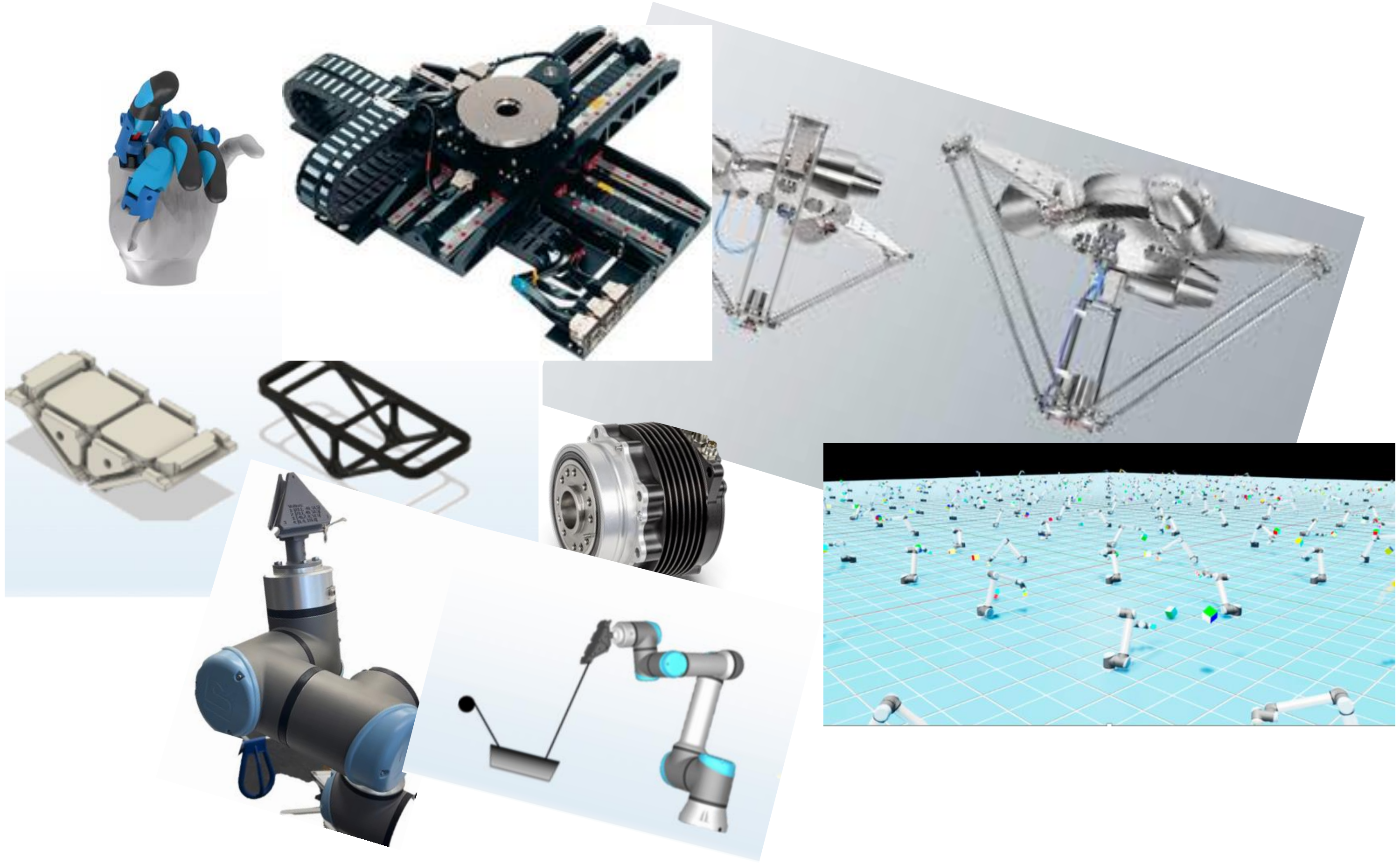
Flexible automation for robotics and manufacturing tasks

Alisa Rupenyan

alisa.rupenyan@zhaw.ch

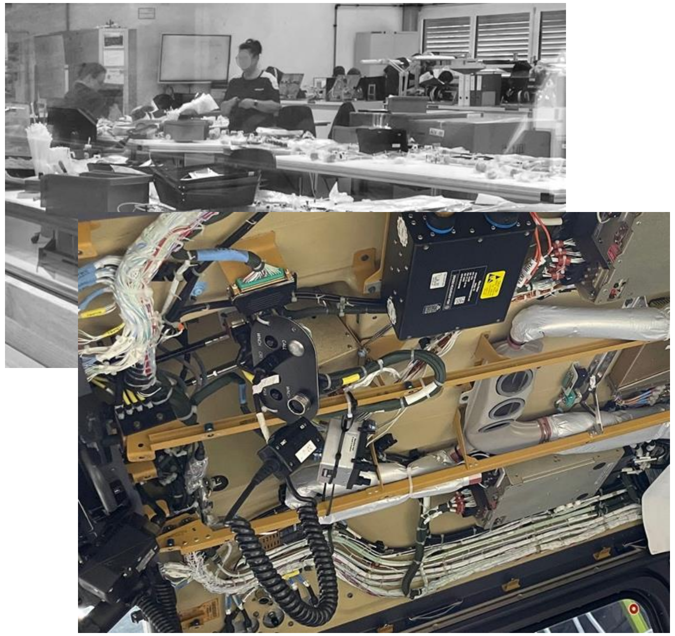
Robotics and manufacturing tasks



Automation in manufacturing

Many operations in manufacturing require understanding and distinguishing between different scenarios or components.

Cabling



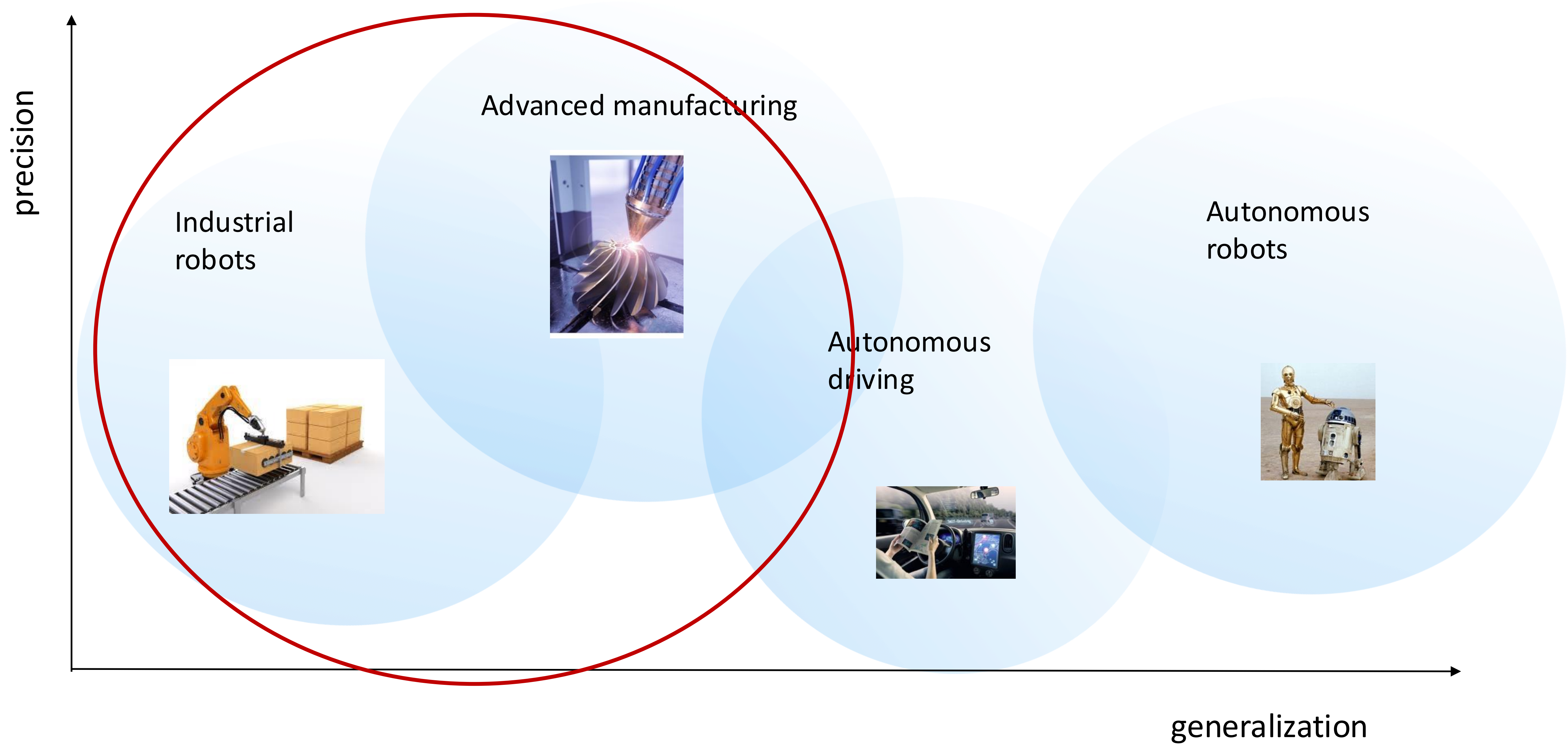
Assembly

boeing

Pick and place

amazon

Advanced manufacturing - **autonomy**



precision

Advanced manufacturing

Industrial
robots

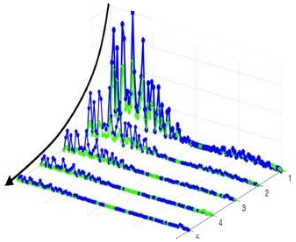
Autonomous
driving

Autonomous
robots

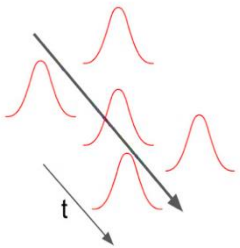
generalization

Challenge: Industrial systems change, but their controllers and planning are rigid.

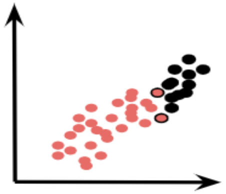
Manufacturing / industrial processes



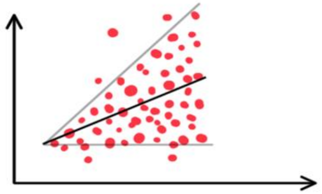
Repetitive nature



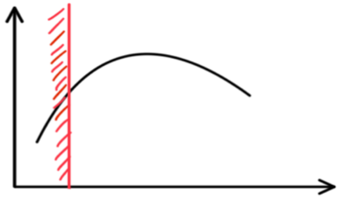
Variability



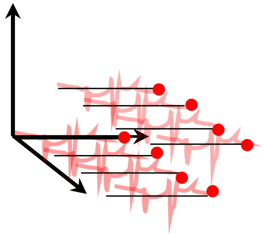
Drifts



Noise



Process and safety constraints



Sparse measurements

These characteristic make standard ML and control fail or require too much data!

Available prior knowledge: already available data and information (digital twins, expert data, …)

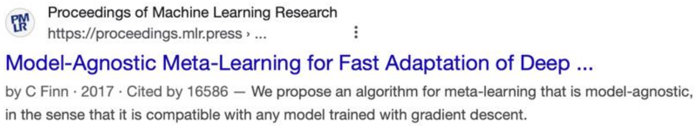
Our goal: intelligent adaptive systems that handle these challenges autonomously and are data-efficient.

via

- Optimization
- Self-tuning algorithms
- Adaptation and task/skill transfer

Dealing with variability

"Learn to learn"



Proceedings of Machine Learning Research
https://proceedings.mlr.press › ...

Model-Agnostic Meta-Learning for Fast Adaptation of Deep ...

by C Finn · 2017 · Cited by 16586 — We propose an algorithm for meta-learning that is model-agnostic, in the sense that it is compatible with any model trained with gradient descent.

Meta-learning: learner is trained on source tasks so that it quickly adapts to an unseen task.

Common skills are learned from source tasks that enable quick learning in an unseen task.

Meta-learning based framework for control

Meta-learning: direct control: policy learning. ; indirect control: model learning

pre-train the Neural SSM (or the Q-function) with abundant data from source systems

- fine-tuning it on the target system with less data.
- Solves the bi-level optimization problem efficiently (implicit gradient)
- Enables Sim-2-Real transfer.

Benefits

- Rapid adaptation to new systems with limited data
- Maintains high control performance

Mpc of uncertain nonlinear systems with meta-learning for fast adaptation of neural predictive models,
Jiaqi Yan, Ankush Chakrabarty, Alisa Rupenyan, and John Lygeros
In *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE* IEEE, 2024.

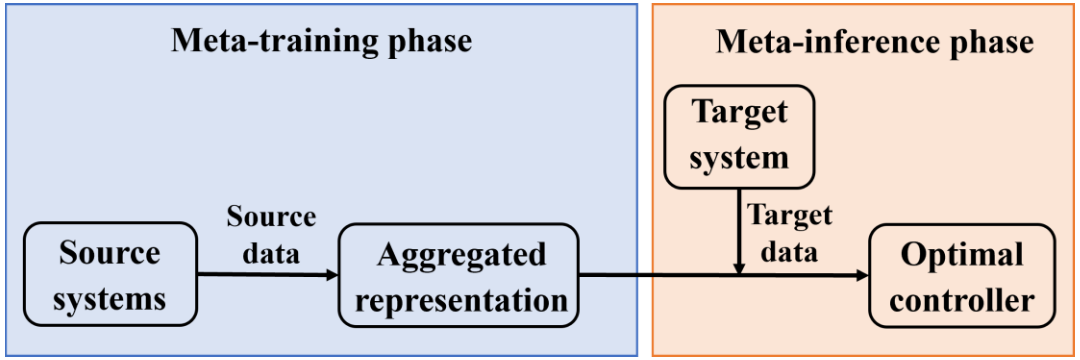
Meta-Learning for Rapid Adaptation in Reference Tracking of Uncertain Nonlinear Systems
Jiaqi Yan, Ankush Chakrabarty, John Lygeros, Alisa Rupenyan
Submitted, IEEE Trans. on Automation, Science, and Engineering, 2026

Meta-learning for control: indirect and direct

Indirect: learn the model (MPC)

Many trajectories from simulations

One trajectory from the new robot



Meta-training phase

Source systems

Source data

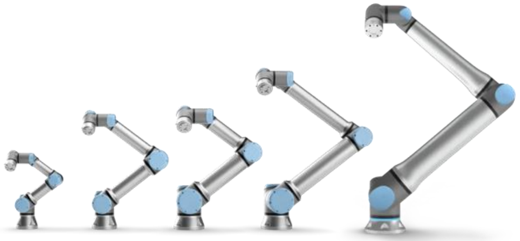
Aggregated representation

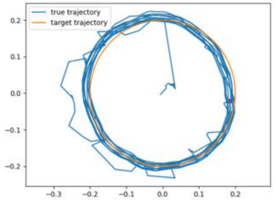
Meta-inference phase

Target system

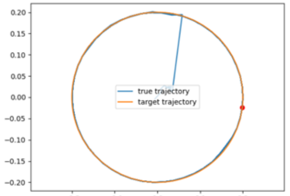
Target data

Optimal controller



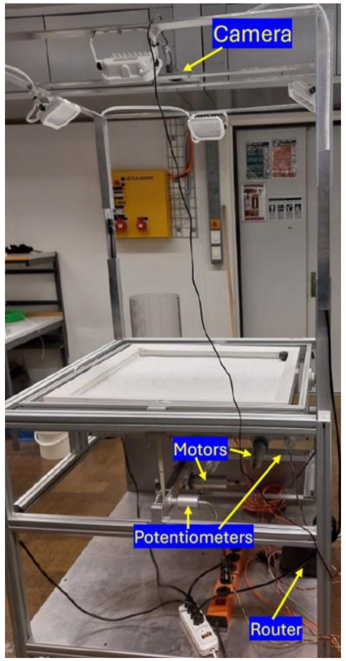


No prior

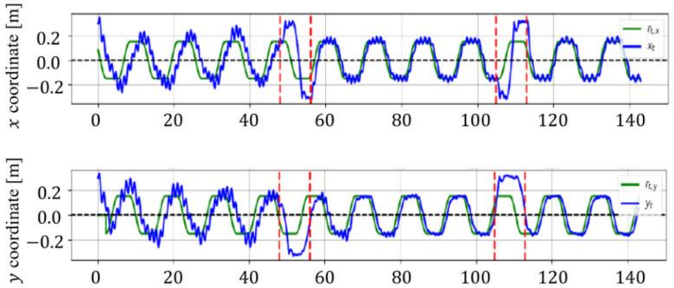


Meta-learning prior

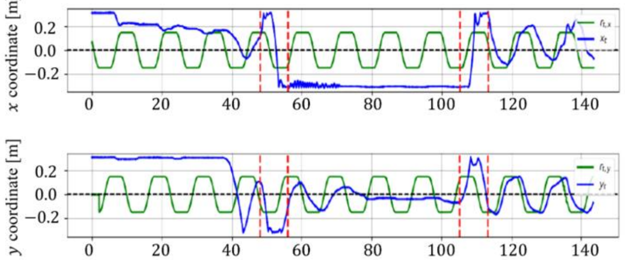
Direct: learn the policy (DQN) — ball-on-plate, sim-to-real



Meta-learning



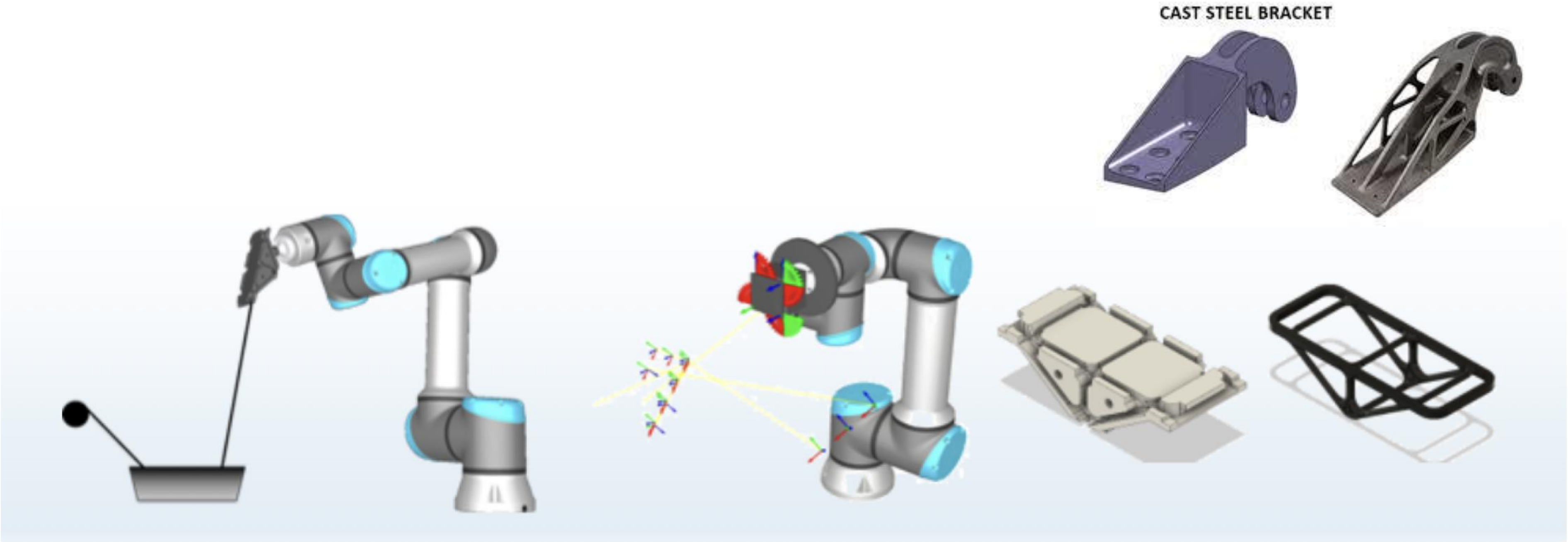
No meta-learning

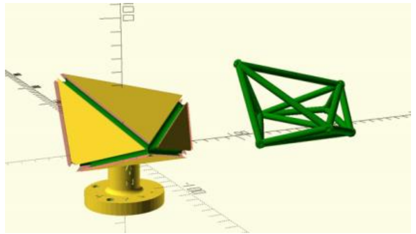


After pre-training with meta-learning:

73.6% reduction in mean tracking cost
vs. training from scratch on hardware
(under review)

Autonomous Robotic Winding

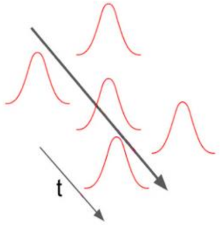




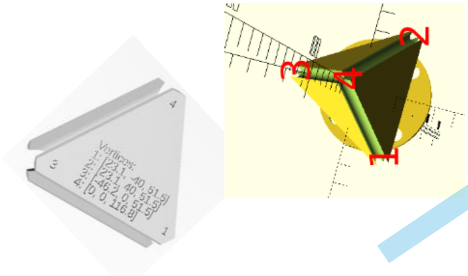
Different core geometry

↕

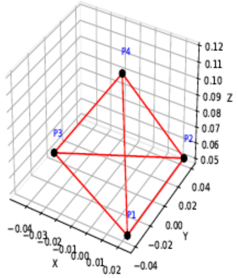
different control parameters



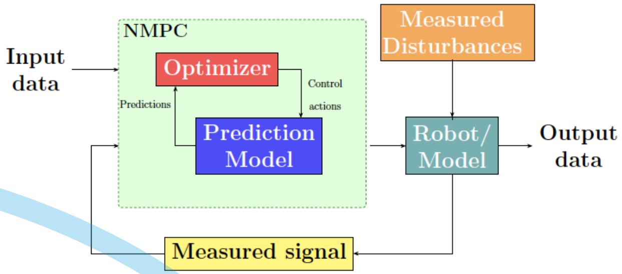
Robotic winding: learning pipeline



Core geometry



Path planning


Input data

NMPC

Optimizer

Predictions

Control actions

Prediction Model

Measured Disturbances

Robot/ Model

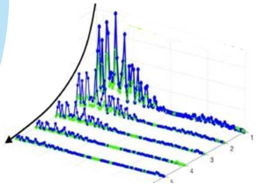
Output data

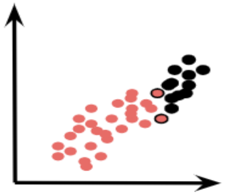
Measured signal

Predictive control

Control parameters to tune

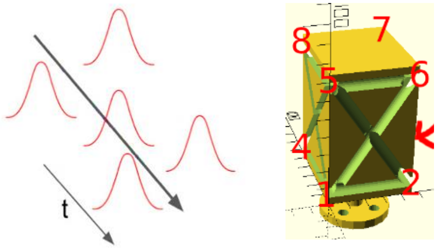
Iterative self-tuning algorithm



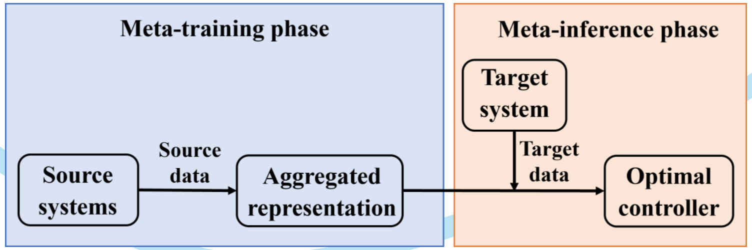


10 winding iterations

Optimized control parameters



New Core geometry


Meta-training phase

Source systems

Source data

Aggregated representation

Meta-inference phase

Target system

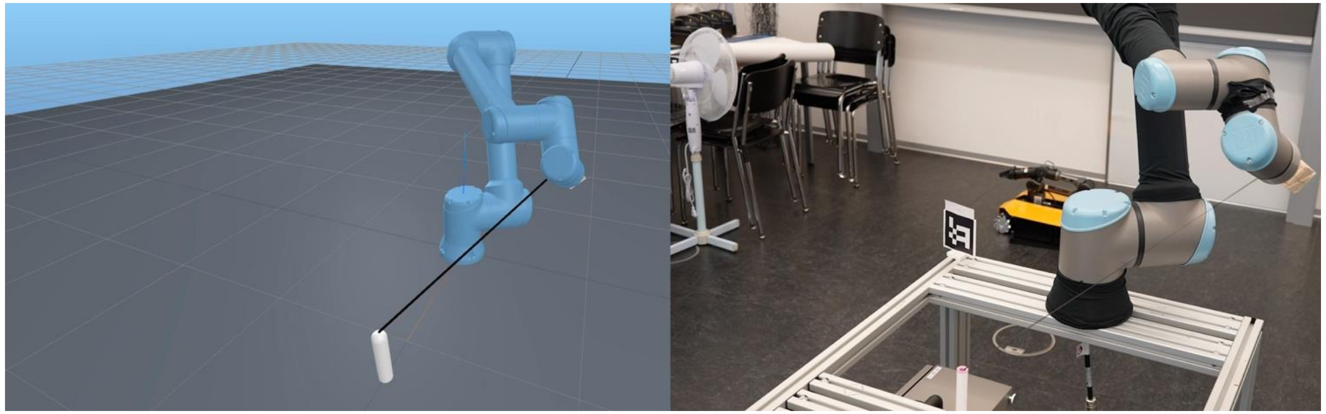
Target data

Optimal controller

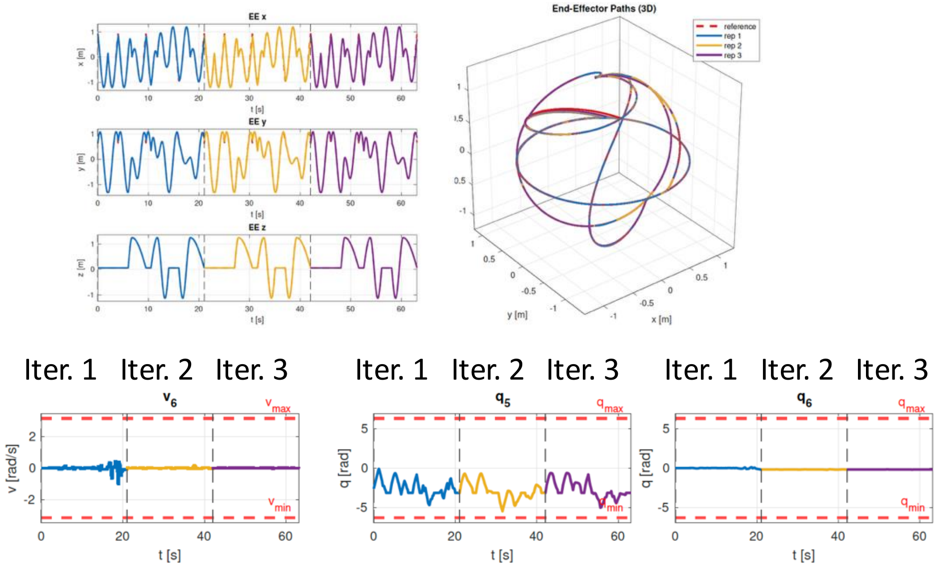
Meta-learning

Pre-trained model

Robotic winding with self tuning



Iterative **self-tuning** of control parameters



Motion planning: in simulation

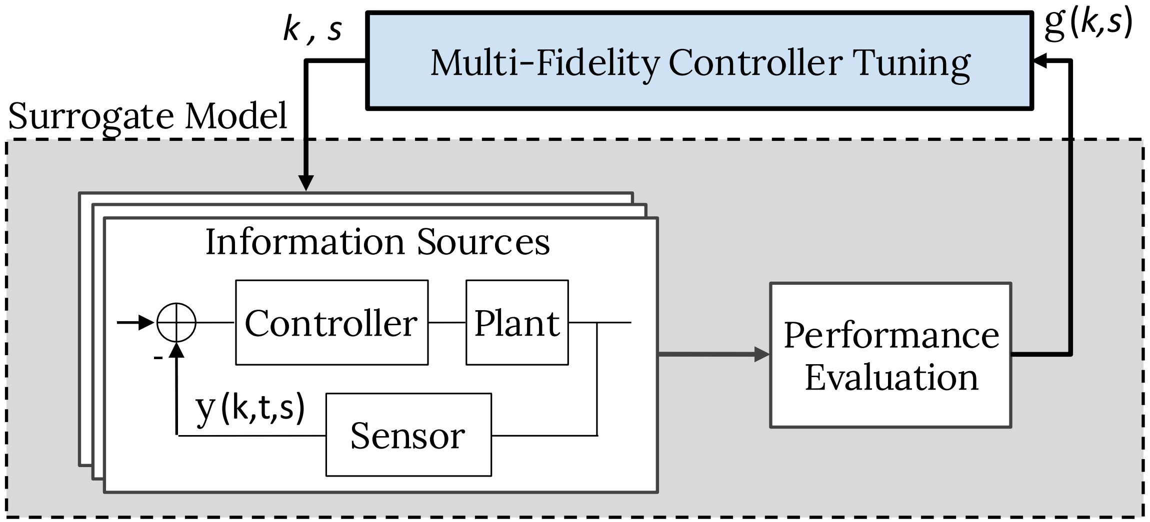
Two approaches: - Graph learning-based
- Optimization-based

Metric	Iteration 1	Iteration 3	Overall Change
RMSE (mm)	59.06	53.348	-9.67%
MAE (mm)	22.968	19.972	-13.04%
RMS (jerk)	75.842	63.052	-16.87%
Control Efforts	248.36	200.13	-19.43%
Control Sat.	0.807	0.682	-15.49%

10-20% improvement in all metrics

Iterative Tuning of Nonlinear Model Predictive Control for Robotic Manufacturing Tasks
D Ingole, V Bhend, SG Murali, O Doebrich, A Rupenyan, 2026 IFAC World Congress

Multi-fidelity optimization of control parameters using Digital Twins



$$k^* := \arg\min_{k\in\mathcal{K}} \hat{\mathrm{g}}(k, s=1)$$

Use available models and DTs to optimize parameters efficiently and to adapt to changes in the system

Guided optimization approach

Performance metric

$$\hat{\mathrm{g}}(k,s) := \mathrm{w}^{\mathsf{T}}\mathrm{h}(\mathrm{y}(k,t,s)) = \sum_{i=1}^{n_{\mathrm{h}}} w_i h_i(\mathrm{y}(k,t,s))$$

$$\mathrm{g}(z) = \hat{\mathrm{g}}(z) + \eta$$

controller gains

time

information source fidelity parameter

Dataset (all inf. sources)

$$\mathcal{D} \leftarrow \mathcal{D} \cup \{[k',s'], \mathrm{g_c}(k',s')\}$$

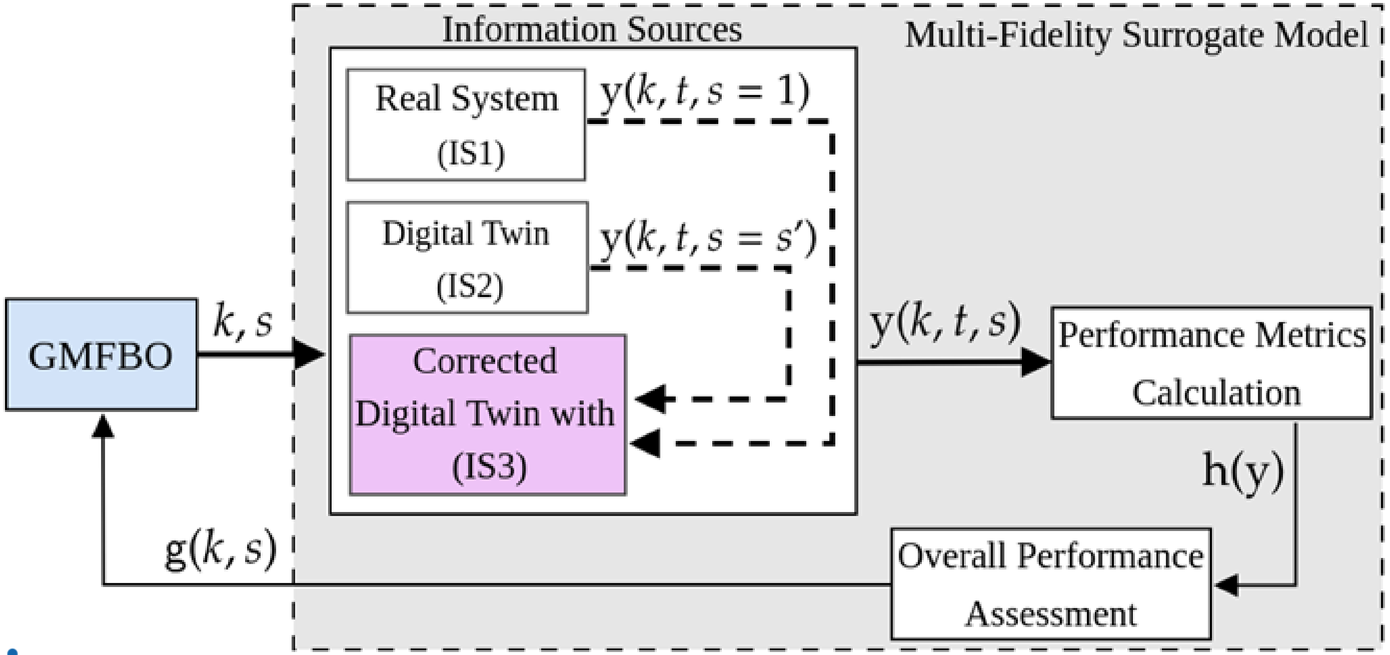
Sanity check condition for IS3

$$\bar{\sigma}_{\mathrm{c}} \ll \alpha := \sqrt{\frac{1}{T}\sum_{t=0}^{T} \mathrm{y}^*(t)} \in \mathbb{R}$$

- threshold on max acceptable uncertainty in the predictions of IS3 << than ref. signal magnitude

Relative mismatch metric for IS2

$$\hat{e}_{\mathrm{IS2}} := \frac{1}{N_{\mathrm{c}}}\sum_{i=1}^{N_{\mathrm{c}}} \frac{|\mathrm{g_c}(k_i) - \mathrm{g}(k_i,s')|}{|\mathrm{g_c}(k_i)|}$$



Optimization setup

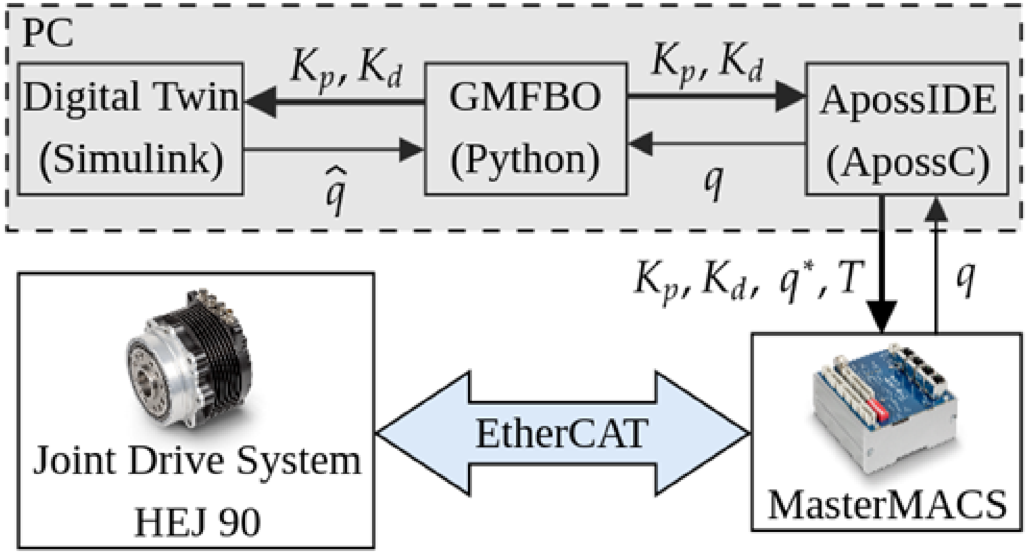
- Controller parameters feasible set

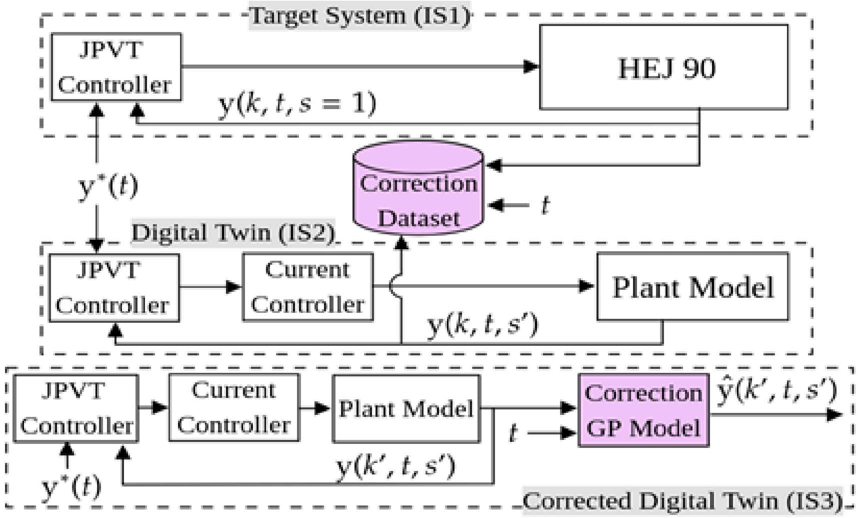
$$\mathcal{K}=\{(K_p,K_d)\mid K_p\in[K_{p_{\min}},K_{p_{\max}}],K_d\in[K_{d_{\min}},K_{d_{\max}}]\}$$

- Weighted sum of performance metrics

$$g(z)=w_1\cdot O_s+w_2\cdot T_{tr}+w_3\cdot T_r+w_4\cdot T_s$$

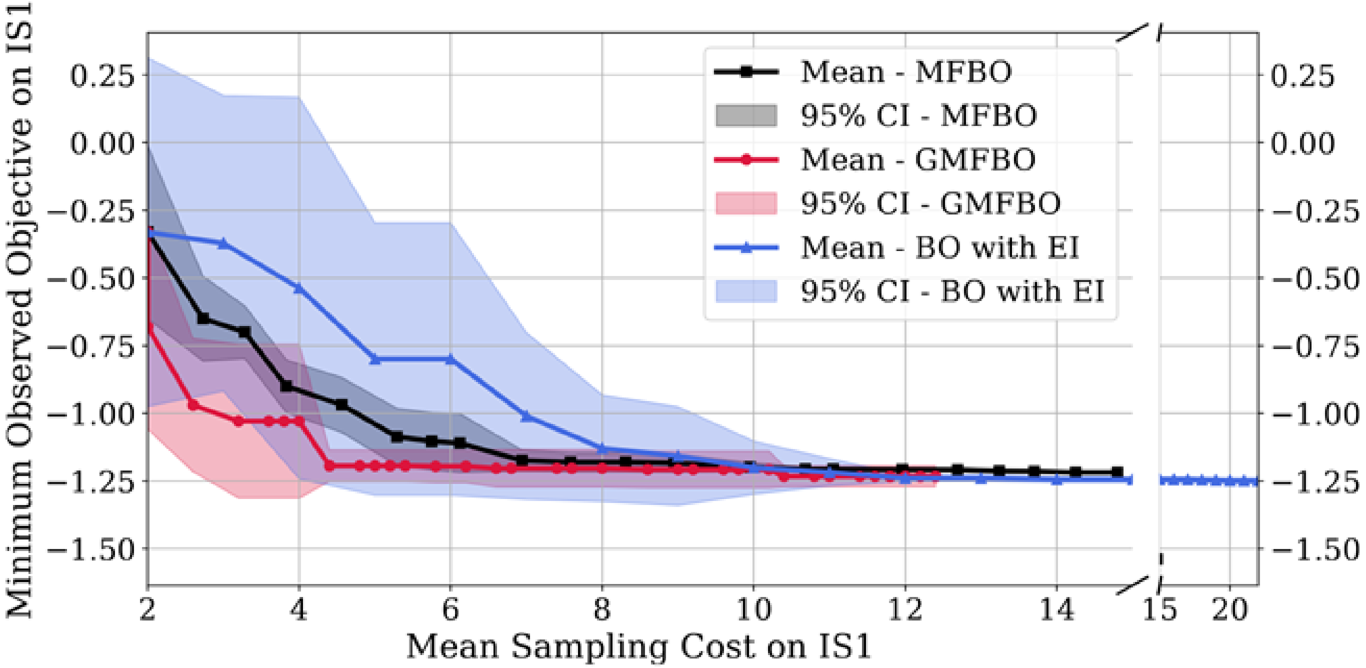
N0s=1=2, N0
s=10,





Results - Real System

- GMFBO improves tuning efficiency at least 28% on real system

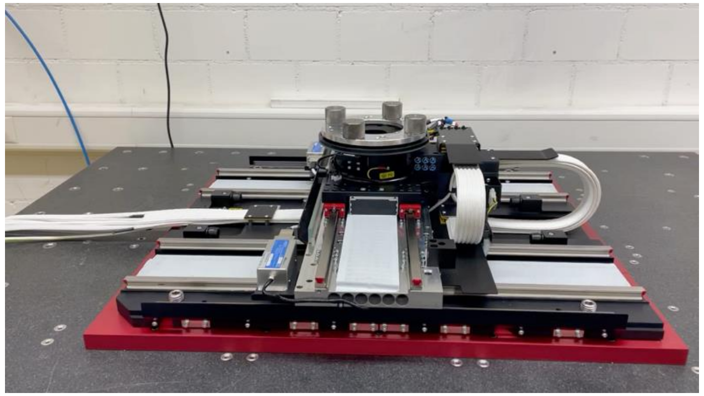


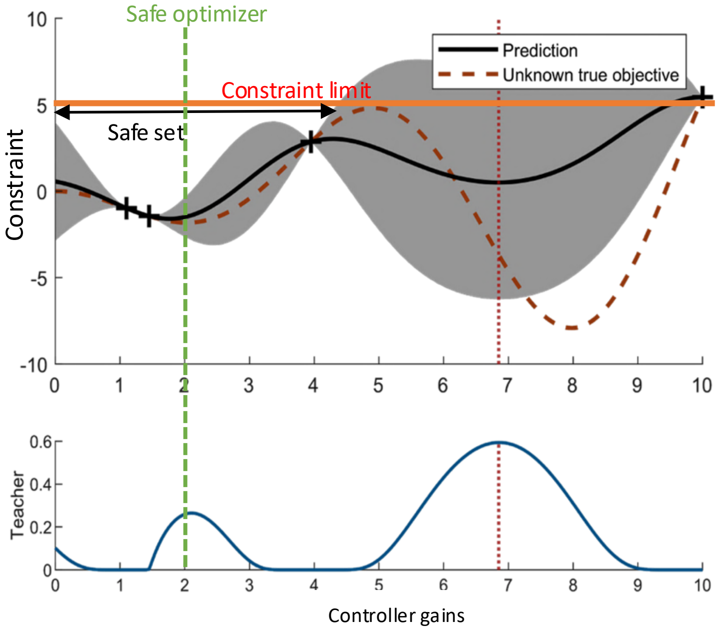
Guided multi-fidelity Bayesian optimization for data-driven controller tuning with digital twins
M Nobar, J Keller, A Forino, J Lygeros, A Rupenyan IEEE Robotics and Automation Letters 2026

Guided Bayesian optimization: Data-efficient controller tuning with digital twin
M Nobar, J Keller, A Rupenyan, M Khosravi, J Lygeros IEEE Transactions on Automation Science and Engineering 2024

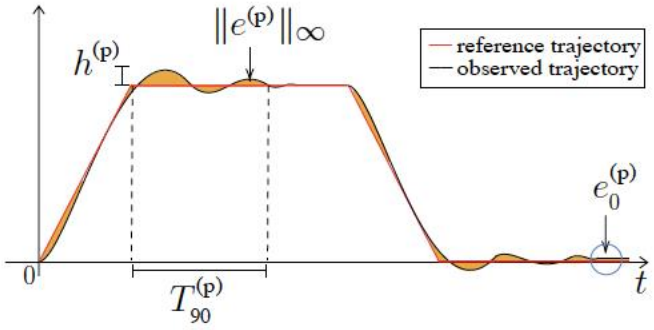
Continuous contex-dependent Bayesian Optimization

Micrometer precision



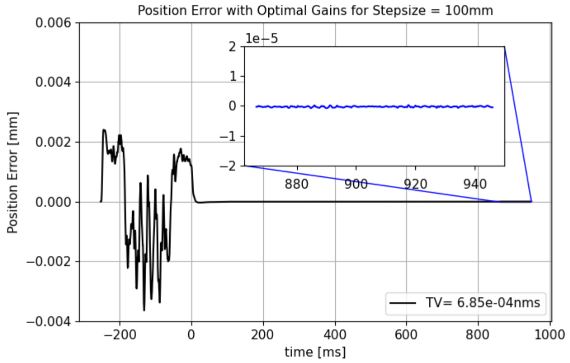


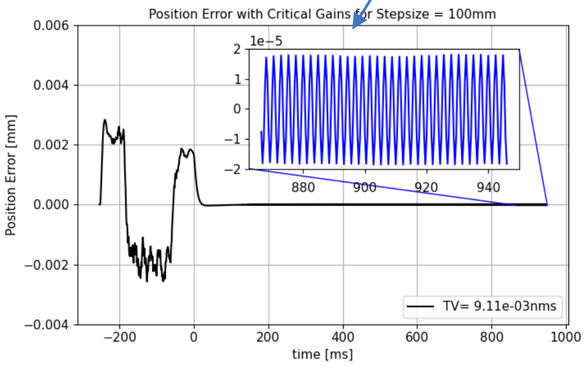
cost



$$F^{(p)} := [h^{(p)}, \bar{h}^{(p)}, T_{90}^{(p)}, \|e^{(p)}\|_\infty, e_{\mathrm{ITAE}}^{(p)}, e_{\mathrm{ss}}^{(p)}, e_0^{(p)}]^{\mathsf{T}}$$

constraint



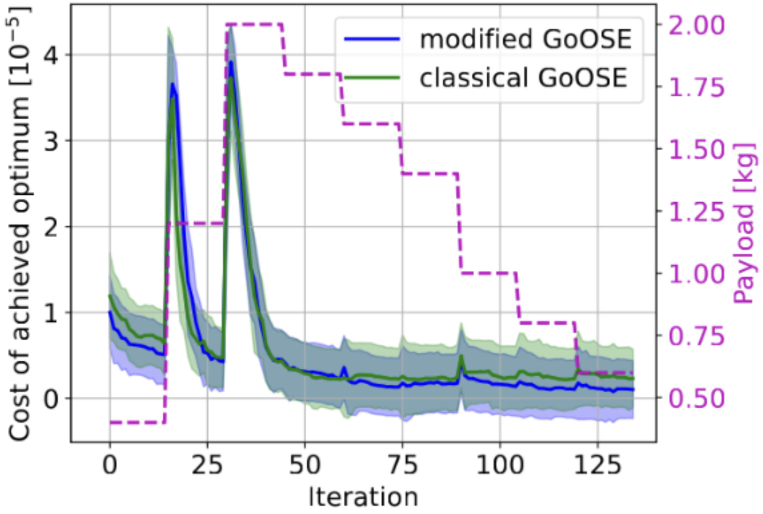


Adaptive Bayesian Optimization for High-Precision Motion Systems
Christopher Koenig, Raamadaas Krishnadas, Efe Balta, and Alisa Rupenyan, 2025, T-ASE

Results

During operation, every movement is optimized!

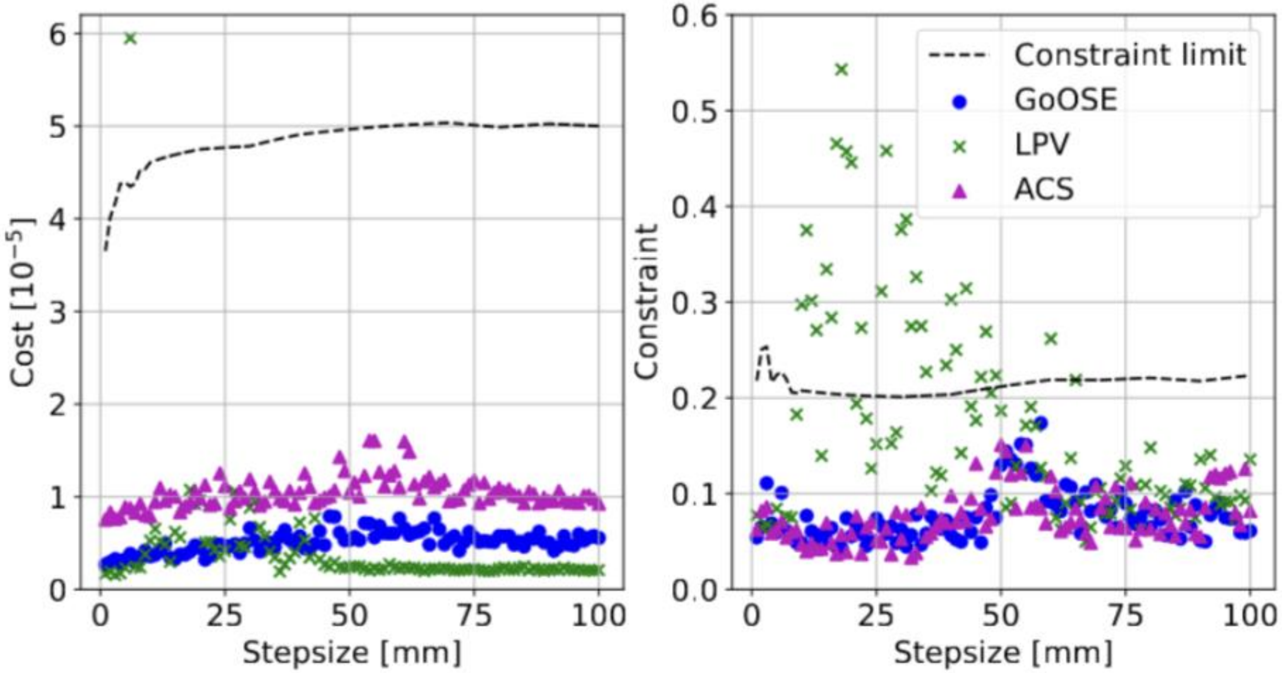
Payload adaption



20% difference in payload

Step size adation

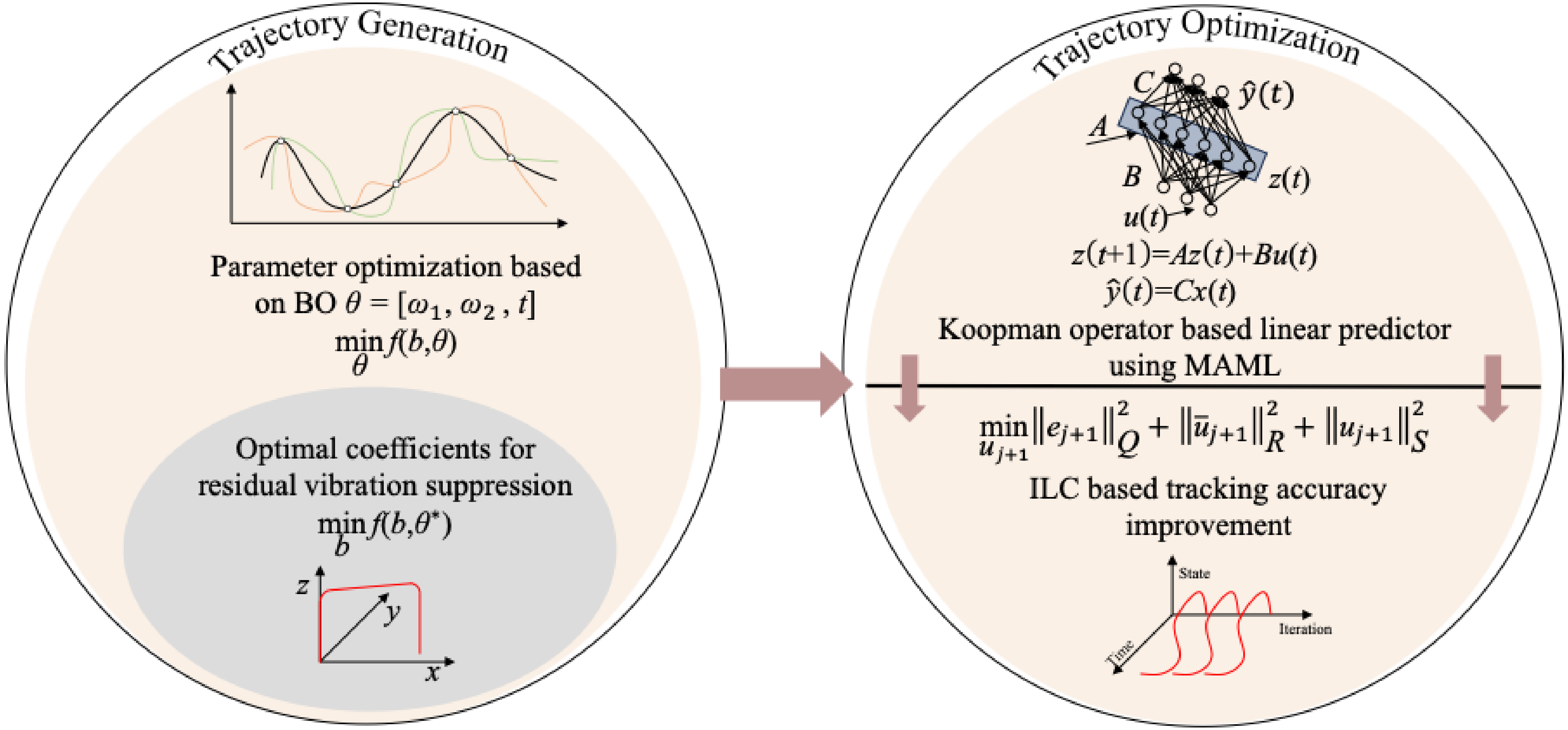
- Outperforming GoOSE, LPV (linear grid interpolation), autotuning



Efficient safe learning for controller tuning with experimental validation M Zagorowska, C König, H Yu, EC Balta, A Rupenyan, J Lygeros, Engineering Applications of Artificial Intelligence, 2025

Adaptive Bayesian Optimization for High-Precision Motion Systems, Christopher König, Raamadaas Krishnadas, Efe Balta, and Alisa Rupenyan, T-ASE, 2025

Data-driven pick-and-place trajectory generation and optimization for industrial robots



Trajectory Generation

Parameter optimization based on BO $\theta = [\omega_1, \omega_2, t]$

$$\min_{\theta} f(b,\theta)$$

Optimal coefficients for residual vibration suppression

$$\min_{b} f(b,\theta^*)$$

Trajectory Optimization

$$z(t+1)=Az(t)+Bu(t)$$
$$\hat{y}(t)=Cx(t)$$

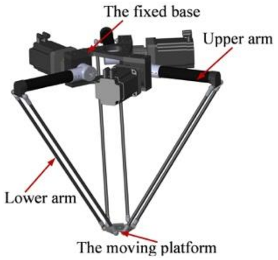
Koopman operator based linear predictor using MAML

$$\min_{u_{j+1}} \|e_{j+1}\|_Q^2 + \|\bar{u}_{j+1}\|_R^2 + \|u_{j+1}\|_S^2$$

ILC based tracking accuracy improvement

Trajectory generation for Pick and Place (PaP) tasks

Acceleration profile $J_\omega(b) = \int_{\omega_1}^{\omega_2} \left| \int_{t_1}^{t_2} a(t) e^{-j\omega t} dt \right|^2 d\omega$



$$\min_{\theta \in \Theta} J(\theta) = w_1 J_1(\theta) + w_2 J_2(\theta)$$

vibrations

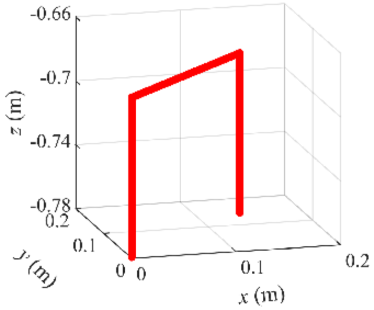
traversal time

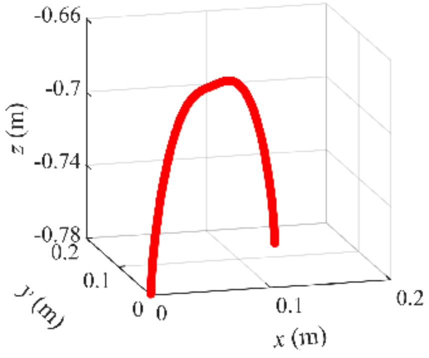
$\theta = [\omega_1, \omega_2, t_r]^T$ the natural resonant frequencies of the delta robot are not known

$$J_1(\theta) = \sum_{k=1}^{3} \sqrt{\frac{1}{N} \sum_{i=1}^{N} rv_k(i)^2} \text{ (vibration RMS)}$$

$$J_2(\theta) = 2(t_v - t_r \Delta t) + t_h \text{ (traversal time)}$$

Different than trajectory tracking optimization!



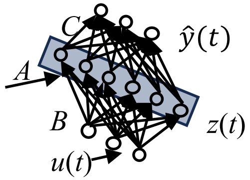


- Frequency optimal PaP trajectory
- Bayesian Optimization (BO) -based parameter tuning → QP $\min_b \frac{1}{2} b^T K b$ (s.t. boundary and kinematic constraints)

  Generate the trajectory with the optimal coefficients b $p(t) = b_0 + b_1 t + b_2 t^2 + \cdots + b_{11} t^{11}$

Trajectory optimization

System model: DT



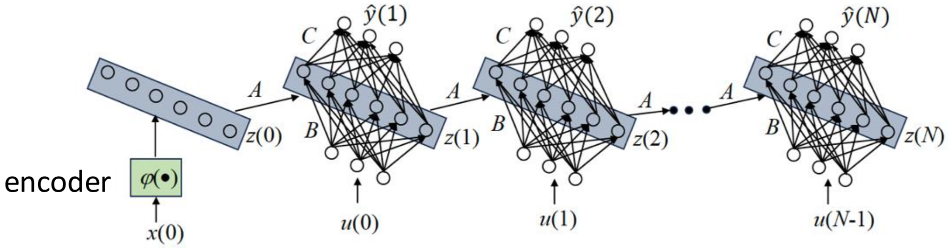
$$z(t+1)=Az(t)+Bu(t)$$
$$\hat{y}(t)=Cx(t)$$

Koopman operator based linear predictor

$$loss(A,B,C)=\sum_{i=1}^{n_d}\frac{1}{Tn_d}\left\|Y_i-\hat{Y}_i\right\|^2$$

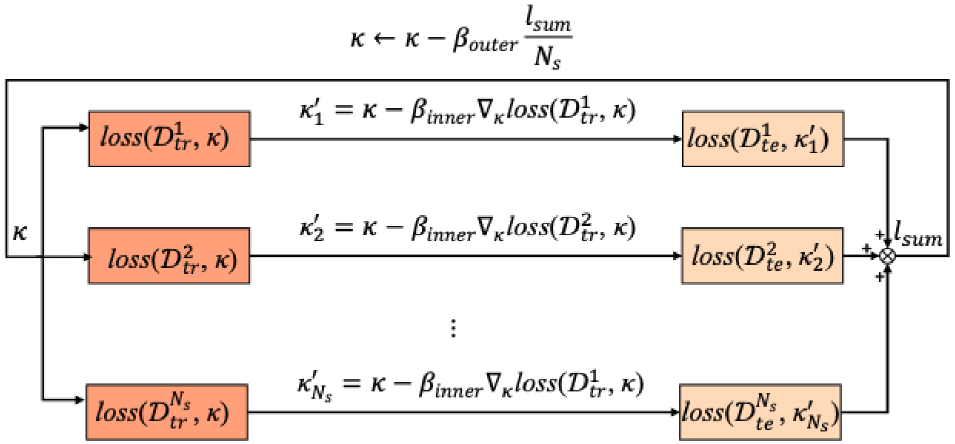
T : length of trajectories; n_d : number of trajectories

RNN (encoder)



$$z(0)=\varphi(x(0)) := [\varphi_1(x(0)), \cdots, \varphi_{n_z}(x(0))]^T$$

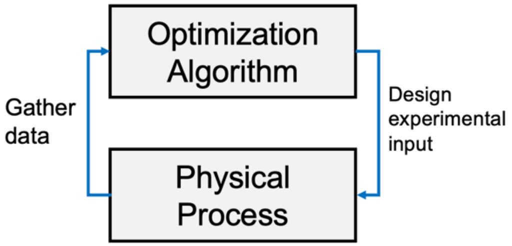
Trained via **meta-learning**
(data from **simulated** UR robots, different masses and lengths of links, friction)



Inference: 1 trajectory from the real robot (UR5)

Data-driven pick-and-place trajectory generation and optimization for industrial robots

Trajectory generation and optimization (BO)



$$\begin{aligned} z(t+1) &= Az(t)+Bu(t)\\ \hat{y}(t) &= Cz(t)\\ z(0) &= \varphi(x(0)) := [\varphi_1(x(0)),\cdots,\varphi_{n_z}(x(0))]^T \end{aligned}$$

$$loss(A,B,C) = \sum_{i=1}^{n_d} \frac{1}{Tn_d} \left\| Y_i - \hat{Y}_i \right\|^2$$

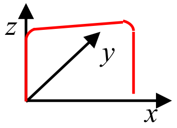
Trained via **meta-learning**
(data from multiple simulated systems)

$$\hat{Y}_j = Hz_j(0) + DU_j$$

For iteration j

$$D = \begin{bmatrix} CB & & \\ \vdots & \ddots & \\ CA^{T-1}B & \cdots & CB \end{bmatrix}, \quad H = \begin{bmatrix} CA \\ \vdots \\ CA^T \end{bmatrix}$$

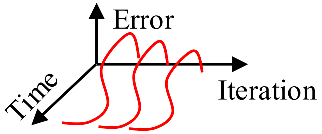
Generated reference trajectory

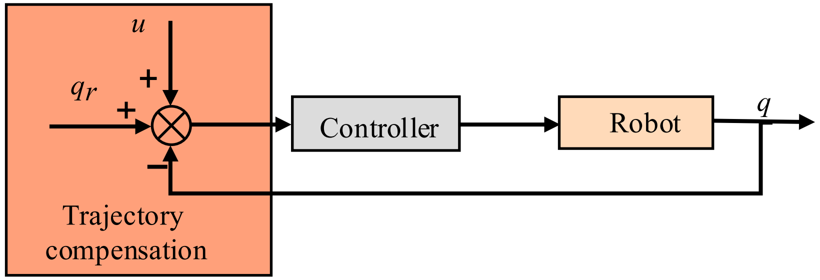


Iterative learning trajectory optimization

$$\min_{u_{j+1}} \left\| e_{j+1} \right\|_Q^2 + \left\| \bar{u}_{j+1} \right\|_R^2 + \left\| u_{j+1} \right\|_S^2$$

ILC based tracking accuracy improvement

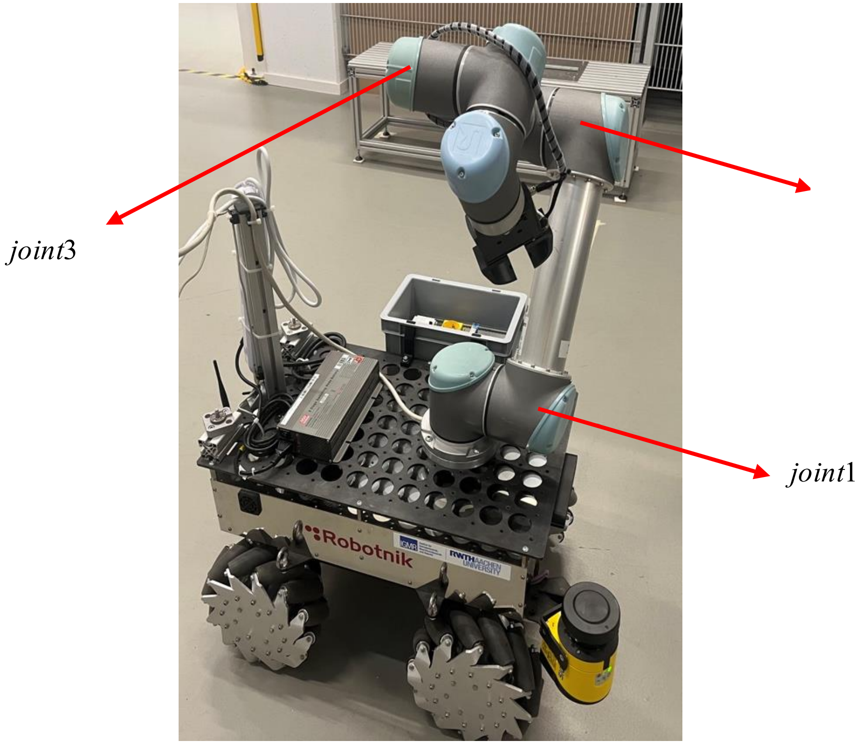




$$U_{j+1} = (D^TQD + R + S)^{-1}(D^TQe_j + D^TQDU_j + RU_j).$$

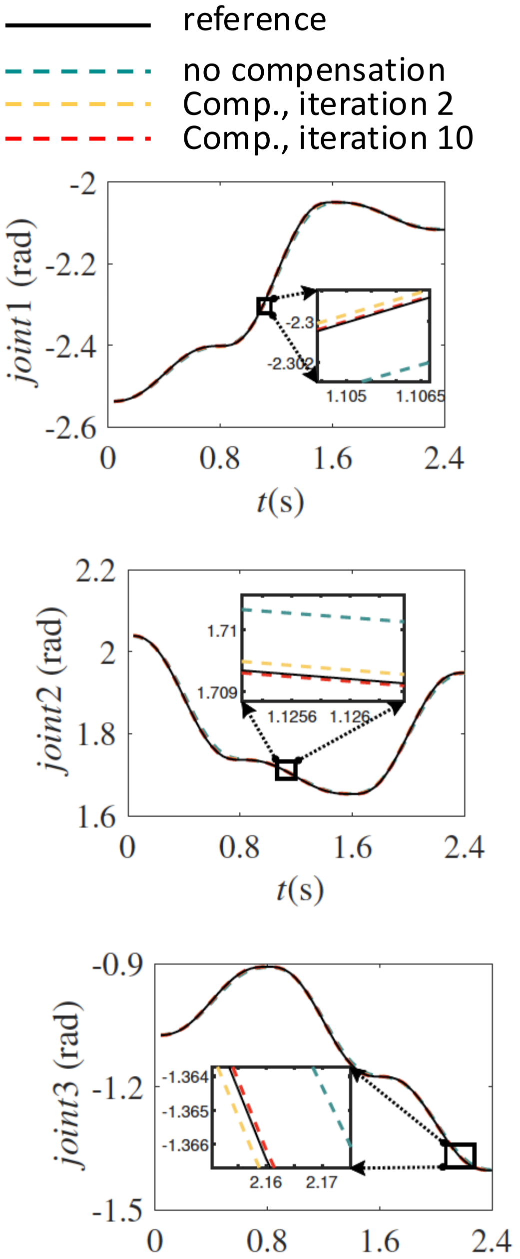
Measure the tracking error and adjust the next iteration input to minimize error.

Trajectory optimization: Experiments on the UR5 robot

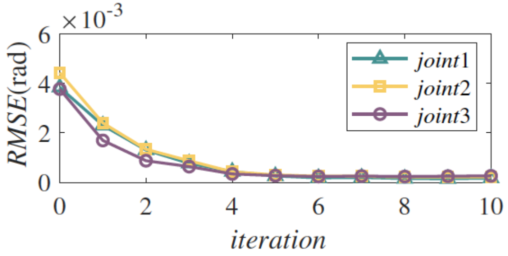


Positioning accuracy comparison. (Unit: 10^{-4}rad)

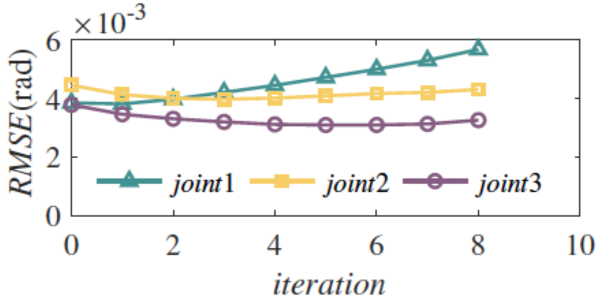
	*joint*1	*joint*2	*joint*3
without compensation	13.11	3.22	9.17
with compensation	0.82	1.82	0.34

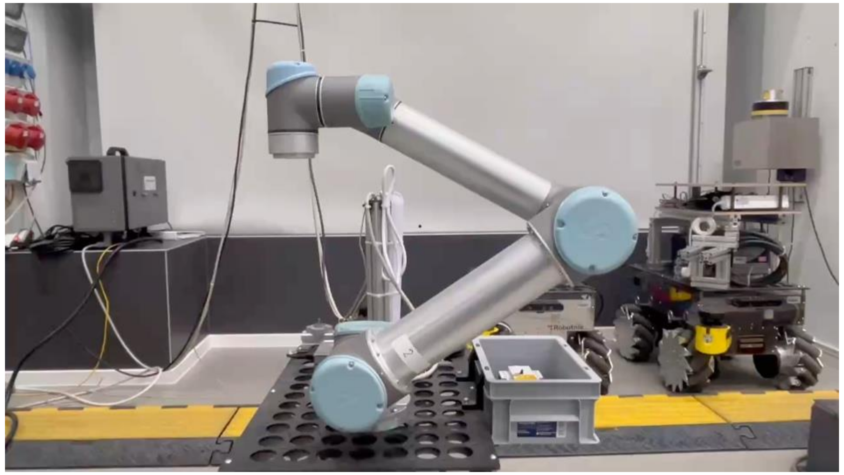


With meta-learning

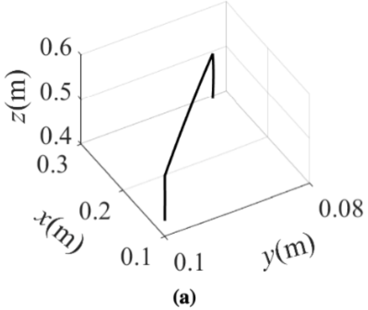


No meta-learning
(1 traj. used)





sampling time 0.04s,
traversal time 2.4s

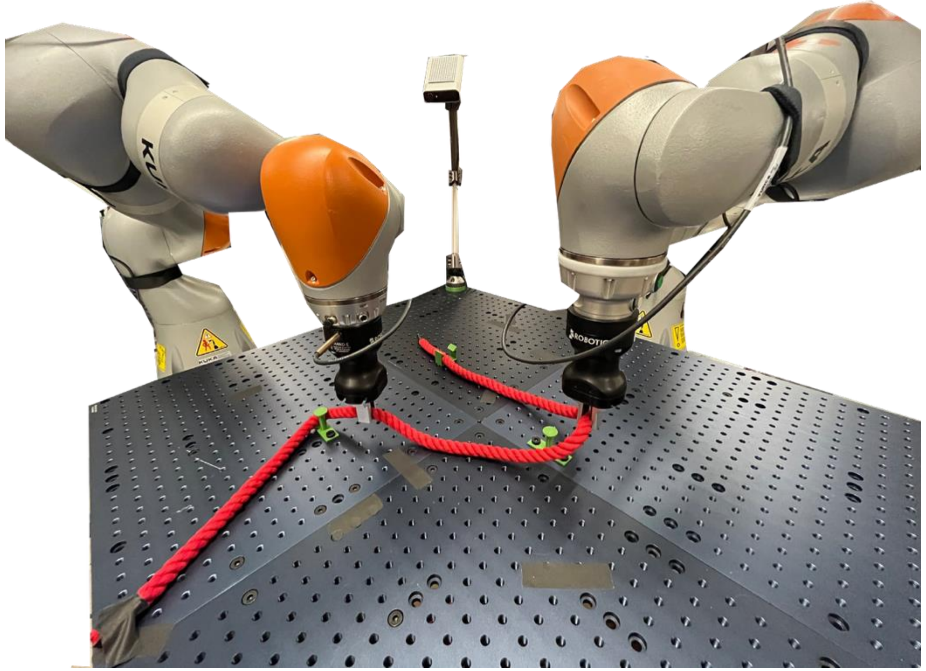


Final optimized trajectory

Mingkun Wu, Alisa Rupenyan, and Burkhard Corves. "Autogeneration and optimization of pick-and-place trajectories in robotic systems: A data-driven approach." ***Robotics and Computer-Integrated Manufacturing*** **97, 2025**

Differentiable Simulation for Trajectory Optimization in Deformable Objects

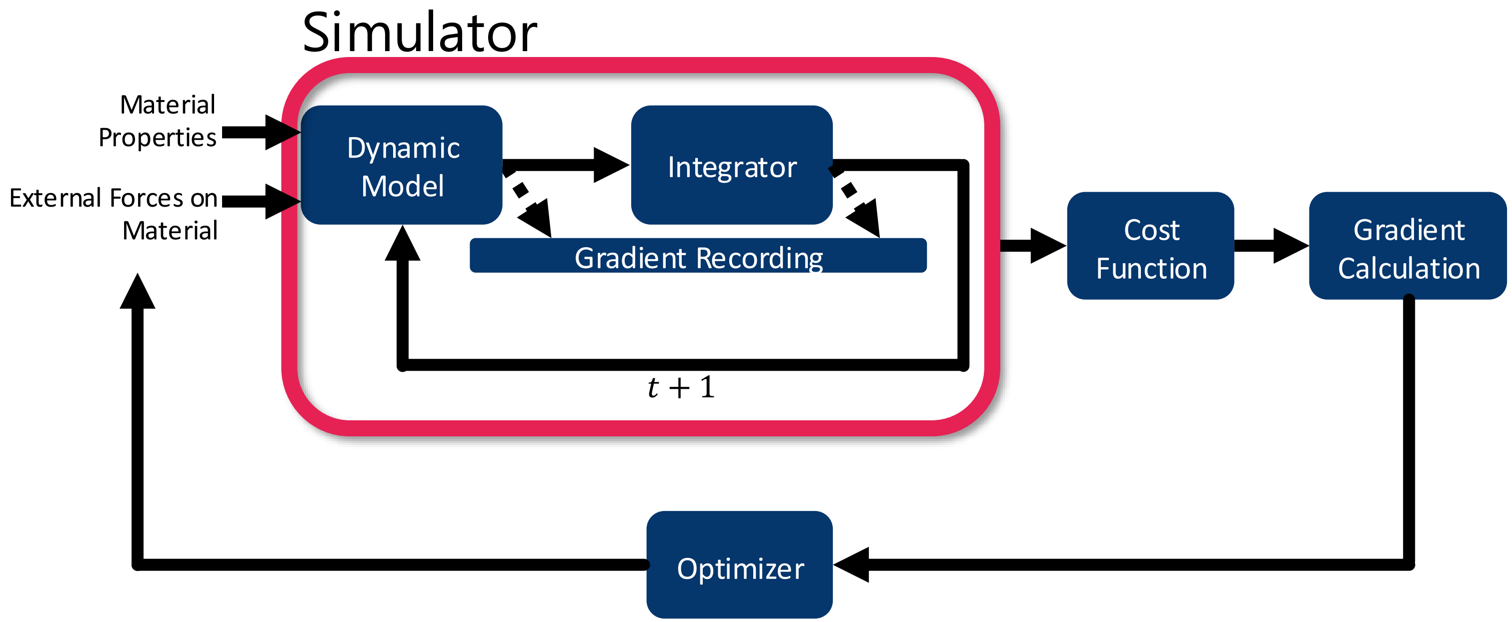
Manipulating deformable objects



Differentiable Implementation to learn and model deformable object behavior

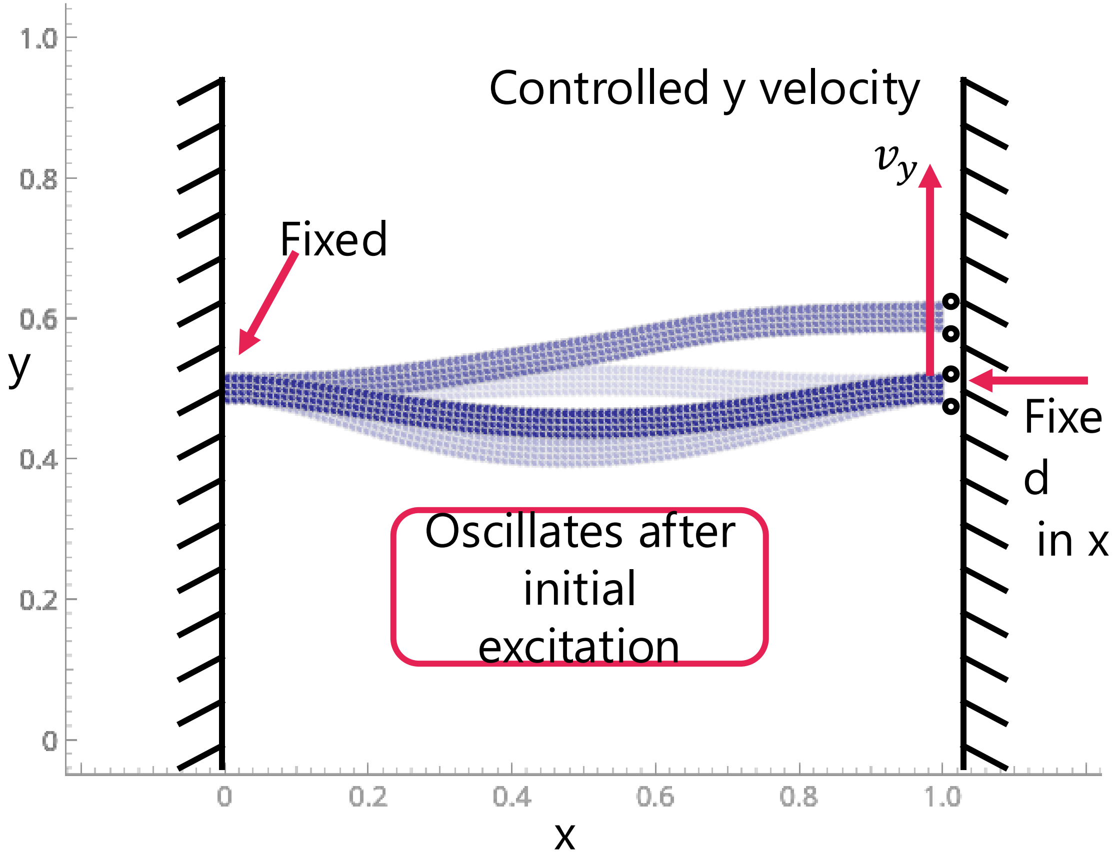
- Gradients would allow us to optimize
- Analytical differentiation of the dynamics are not possible
- With ML methods **automatic differentiation** has become accessible

→ Allows us to take the gradient through single simulation steps and entire dynamic simulations



Control Problem Formulation

- 2D simulation
- Rope like object: 1m by 4cm
- Hyperelastic material, silicon rubber like
- Damping based on strain (deformation) rate

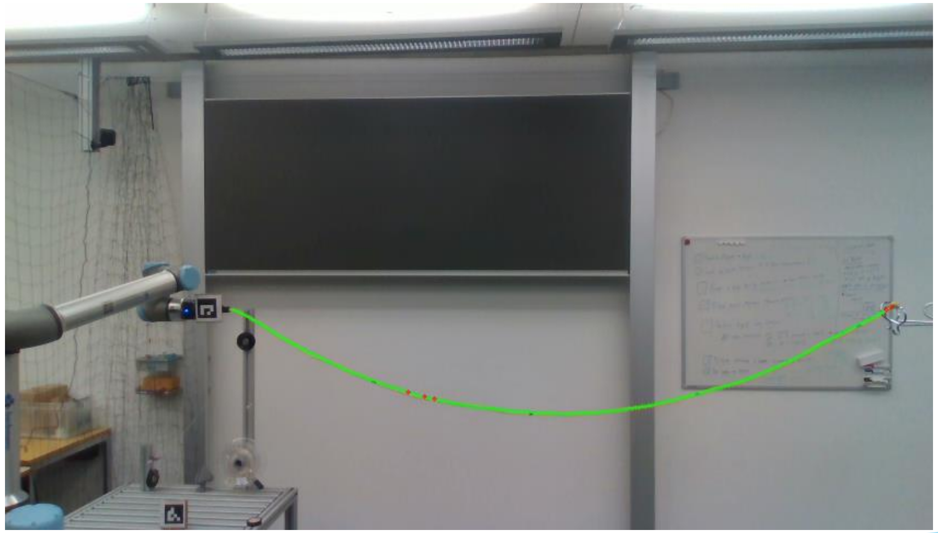


- Excite the object using a scheduled velocity profile.

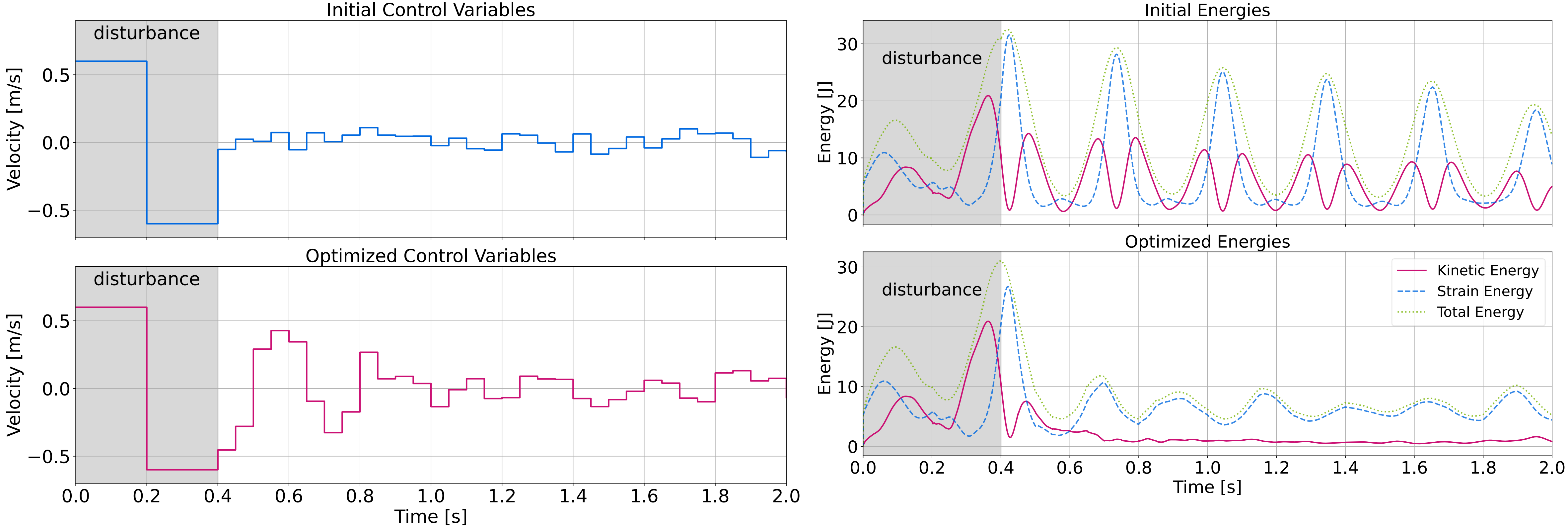
→ Elicits oscillations in the object

Goal: Find a velocity trajectory minimizing the kinetic energy in the system

- Real setup implementation in progress



Results


Initial Control Variables

Optimized Control Variables

Initial Energies

Optimized Energies

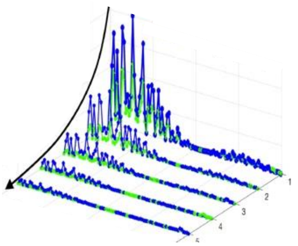
With simulator: The oscillations are damped in 0.5 seconds

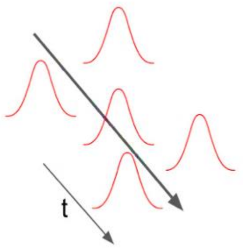
Without simulator: Damping continues after 2 seconds

Differentiable Material Point Method for the Control of Deformable Objects
D Bolliger, G Fadini, M Bambach, A Rupenyan, IFAC World Congress, 2026

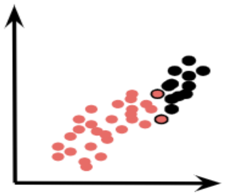
Challenge: Industrial systems change, but their controllers and planning are rigid.

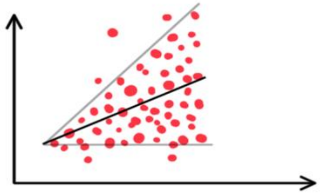
Manufacturing / industrial processes

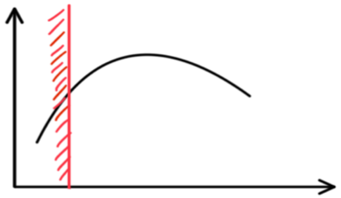

Repetitive nature

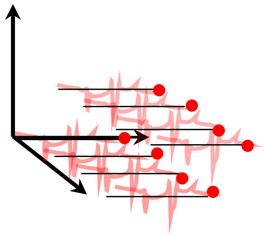


Variability


Drifts


Noise


Process and safety constraints


Sparse measurements

Iterative learning based optimization for efficient adaptation
Meta-learning to handle variability in machines and scenarios, and for continual learning
Data-driven optimization + digital twins to optimize in the presence of drifts and sparse measurements

Acknowledgements

zh aw School of Engineering
CAI Centre for Artificial Intelligence

Funding

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Innosuisse – Schweizerische Agentur für Innovationsförderung

Swiss National Science Foundation

Horizon Europe

NCCR Automation

Johann Jacob Rieter-Stiftung

Industry partners and collaborators

SIEMENS

maxon

MITSUBISHI ELECTRIC RESEARCH LABORATORIES

Collaborators

ETH zürich

inspire

NCCR Automation

n|w Fachhochschule Nordwestschweiz

RWTH AACHEN UNIVERSITY