Data-enabled iterative learning control:
A zero-sum game design for time-scale-varying tasks

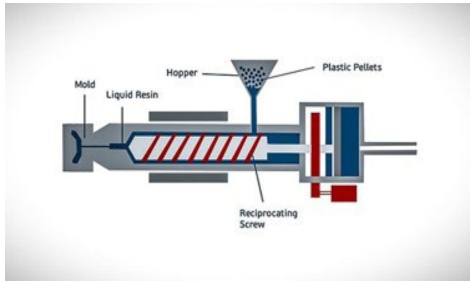
Zhihe Zhuang
Rodrigo A. González
Hongfeng Tao
Wojciech Paszke
Tom Oomen

江南大學
JIANGNAN UNIVERSITY
TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF ZIELONA GÓRA

November 14, 2025

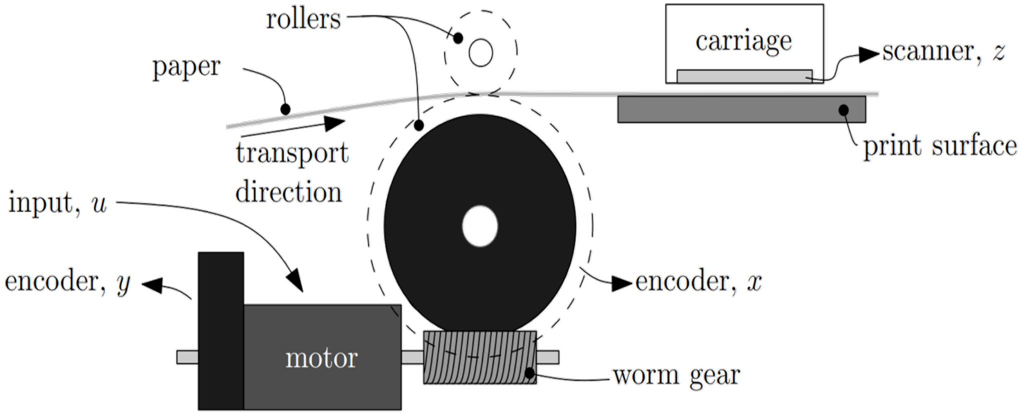
Research background & Motivation

Repetitive systems



Repetitive tracking tasks

Varying tasks in ILC
(iterative learning control)



Adaptability Flexibility VS. **High precision Low cost**

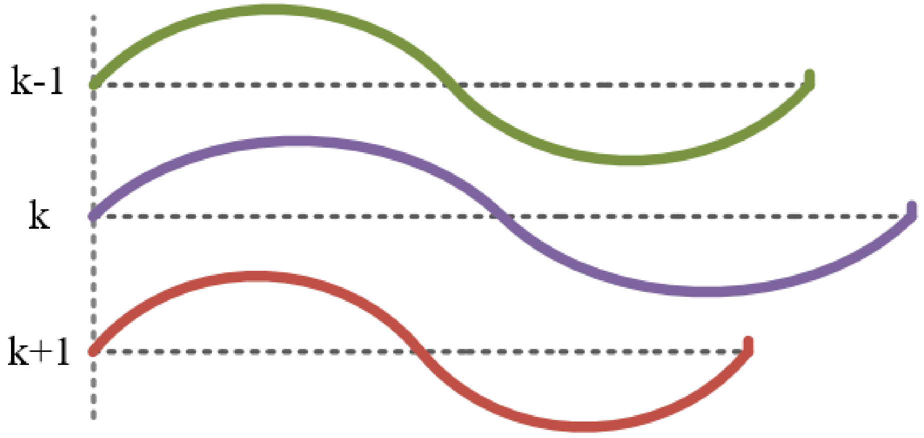
ILC for varying tasks

Goal: Balance between task flexibility and high precision & low cost

Contents

JIANGNAN UNIVERSITY

Contents



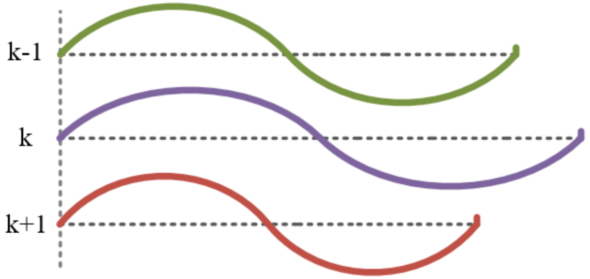
Task definition

JIANGNAN UNIVERSITY

Time-scale-varying tasks

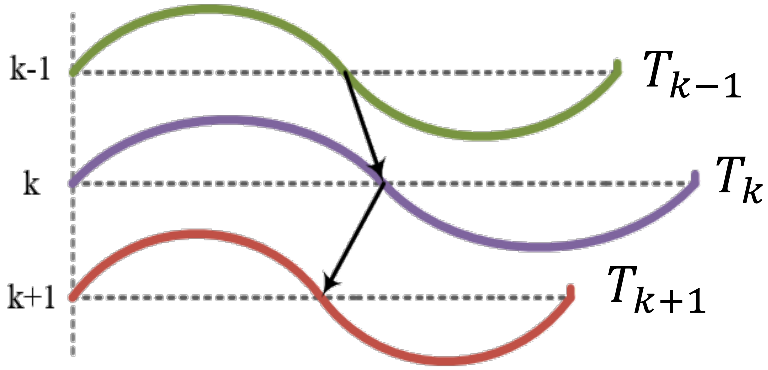
same "shape"

Different time scales

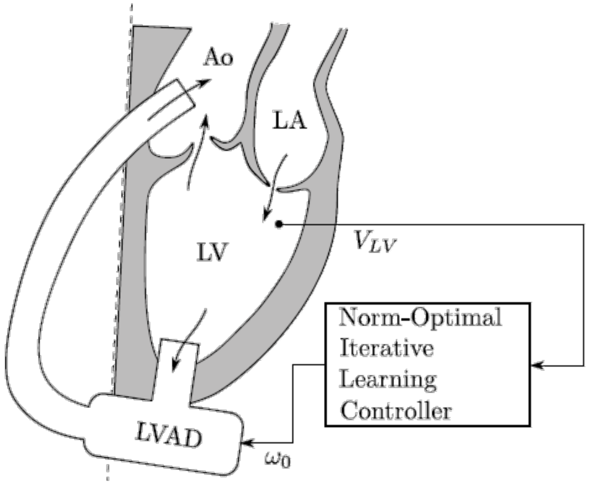


Definition

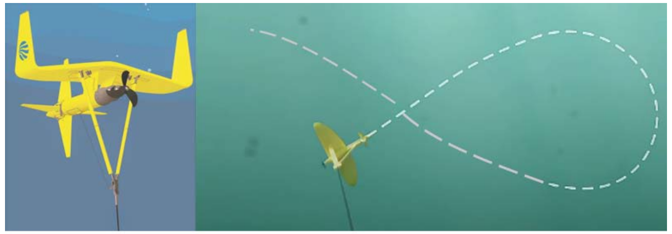
1. **Bijection** between different time scales
2. **Unique** mapping



Examples



Left ventricular assist device (LVAD)



Marine hydrokinetic energy system

M. Ketelhut et al. 2019
Control Engineering Practice.

M. Cobb et al. 2022
IEEE Transactions on Control Systems Technology.

- **Library-based learning control,**

J. Xu 1998, *IEEE Transactions on Automatic Control.*

- **ILC with basis function,**

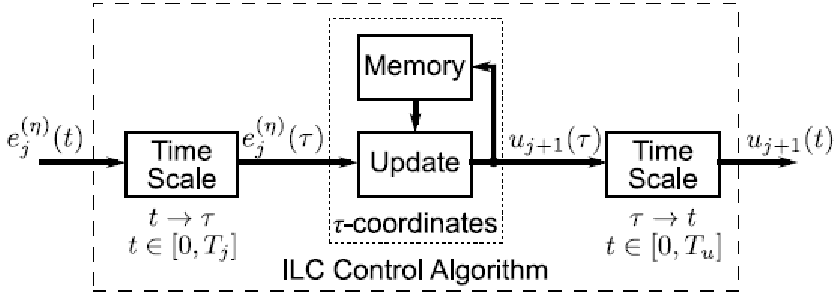
Bolder &Oomen 2015, *IEEE Transactions on Control Systems Technology.*

- **Path optimization ILC,**

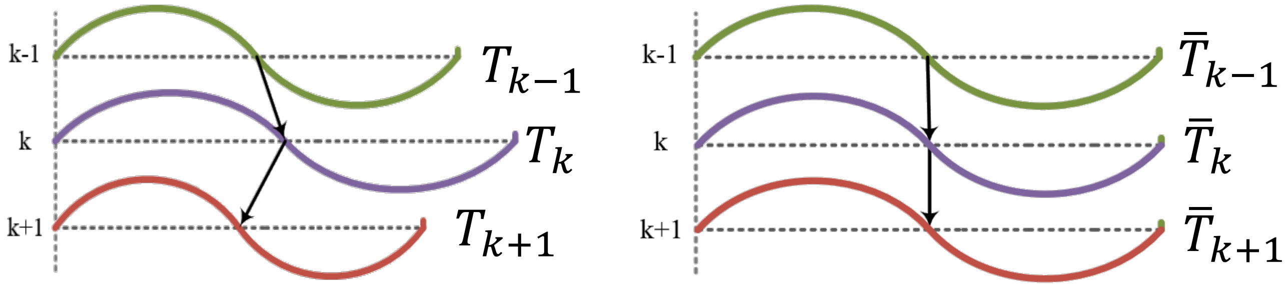
Cobb et al. 2021, *IEEE Transactions on Control Systems Technology.*

Failure of corresponding learning

Normalization



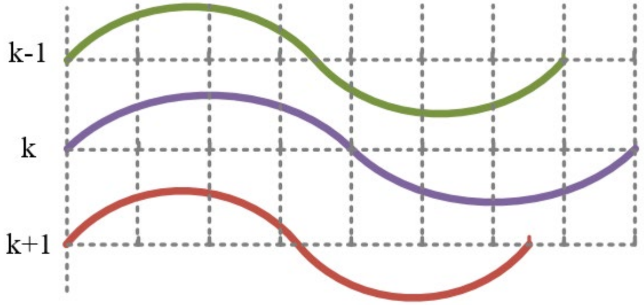
M. Cobb et al. 2022
IEEE Transactions on Control Systems Technology.

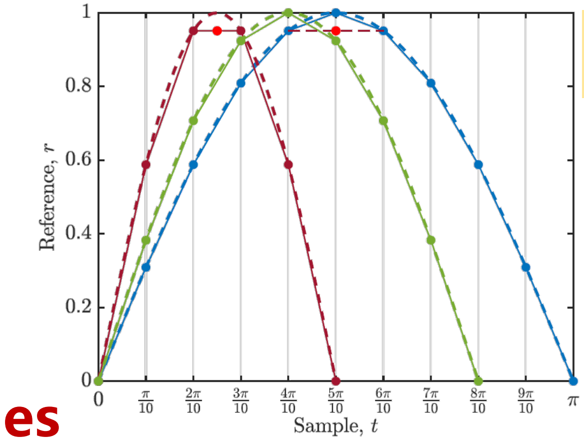


Bijection in continuous-time domain

Discrete-time ILC

with sampling behavior?





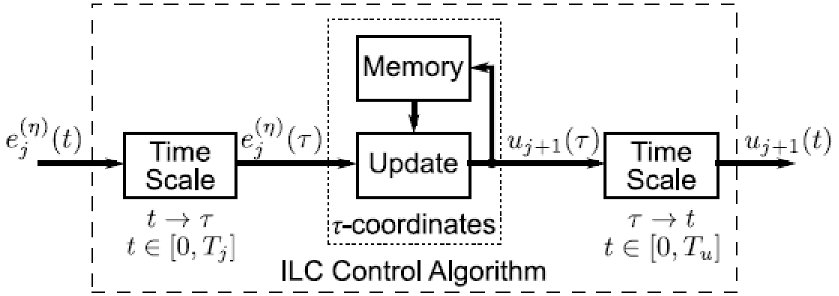
Sine example

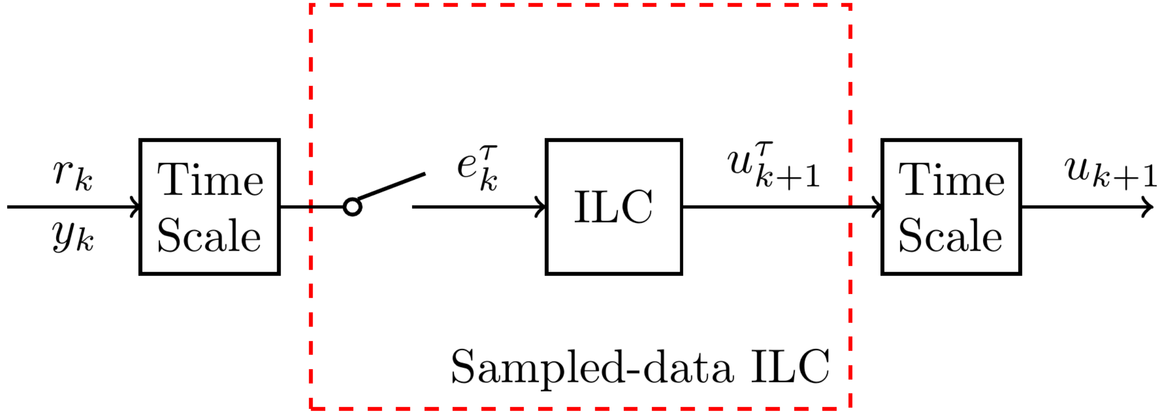
Issues

1. **Varying set points at a sampling instant**
2. **Missing information for learning**

Time-scale transformation scheme

Modified scheme



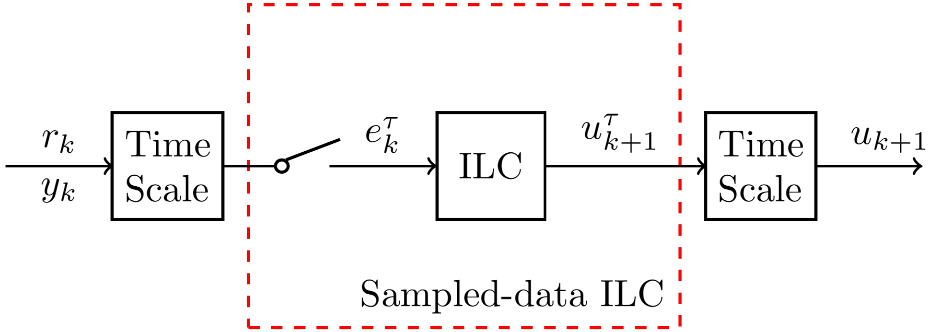


$[0, T_k] \longrightarrow [0,1]$

Continuous-time reference ⟷ **Discrete-time ILC input**

$[0,1] \longrightarrow [0, N_k]$

ILC problem



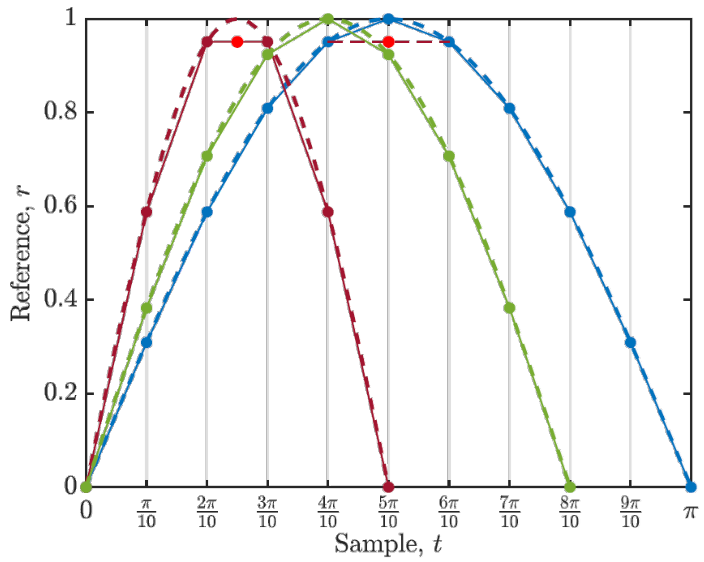
Sampled-data ILC problem

$$u_{k+1} = \arg\min_{u} J_{k+1}^S(u, r^c, y_{k+1}^c)$$

suitable sampling time

Discrete-time ILC problem

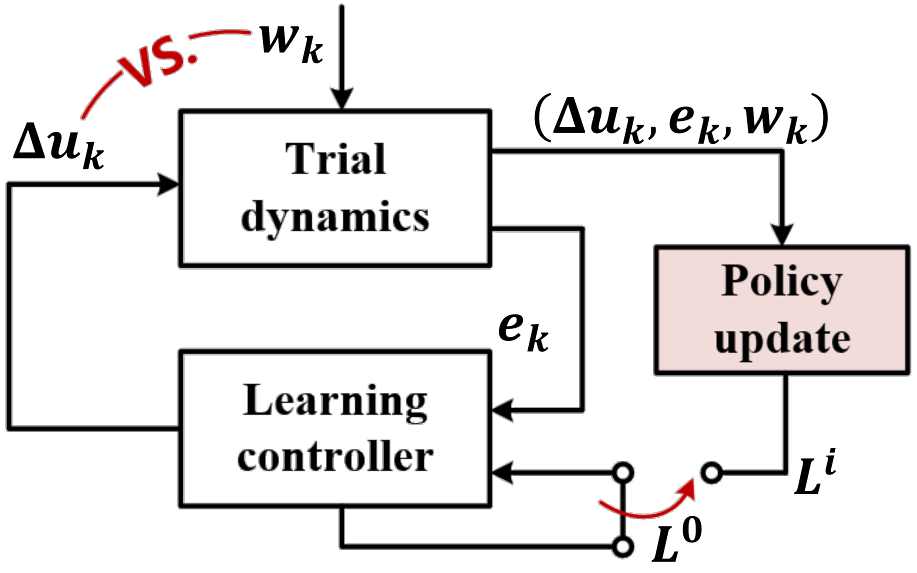
$$u_{k+1}^\tau = \arg\min_{u^\tau} J_{k+1}(u^\tau, e^\tau)$$



ILC problem:

1. define a suitable **cost function**
2. solve the **optimization** problem
3. Ensure **trial convergence**
4. Handle **trial-varying disturbances**

Contents



VS.
w_k
Δu_k
$(\Delta u_k, e_k, w_k)$
Trial dynamics
Policy update
e_k
Learning controller
L^i
L^0

System dynamics

JIANGNAN UNIVERSITY

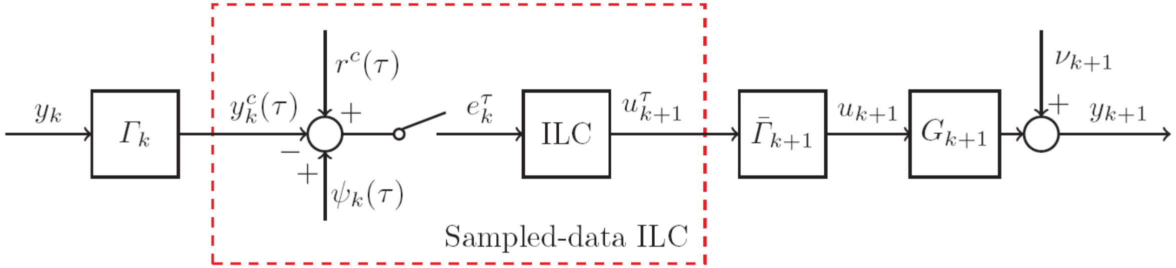
Data-driven representation

$$y_k = G_k u_k + v_k$$

y_k: Output data sequence
u_k: Input data sequence
G_k: Data transfer matrix
v_k: External disturbance

+

Modified time-scale transformation scheme



Trial error dynamics

with trial-varying disturbances

$$e_{k+1} = e_k - G\Delta u_{k+1} + \boxed{w_{k+1}}$$

1. **Process disturbance ψ_k from varying time scales**
2. **External unknown noise v_k**

Finite energy assumption:

$$\sum_{k=0}^{\infty} w_k^\top w_k < \infty$$

Issue

Why trial dynamics?

Zero-sum game design

Cost design via “Learning from the delayed rewards”

$$J_{k+1}(\Delta u_{k+1}) = \sum_{j=k}^{\infty} \rho^{j-k}\left(e_{j+1}^{\top} Q e_{j+1} + \Delta u_{j+1}^{\top} R \Delta u_{j+1} - \gamma^2 w_{j+1}^{\top} w_{j+1}\right)$$

Discounted factor

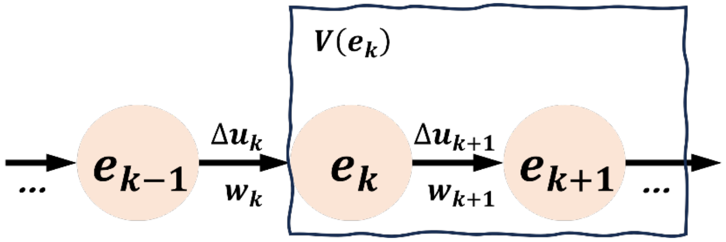
Disturbance attenuation goal:

$$\sum_{k=0}^{\infty}\left(e_k^{\top} Q e_k + \Delta u_k^{\top} R \Delta u_k\right) \leq \gamma^2 \sum_{k=0}^{\infty} w_k^{\top} w_k$$

Reinforcement learning perspective

$$J_{k+1}(\Delta u_{k+1}) \Rightarrow V(e_k) = \underbrace{c(e_k, \Delta u_{k+1}, w_{k+1})}_{\text{current state}} + \underbrace{\rho V(e_{k+1})}_{\text{future cost}}$$

Bellman equation



Zero-sum game design

$$e_{k+1} = e_k - G\Delta u_{k+1} + w_{k+1}$$

Player 1 Δu_k

Minimize $V(e_k)$

Player 2 w_k

Maximize $V(e_k)$

Min-max optimization problem

$$\min_{\Delta u_{k+1}} \max_{w_{k+1}} V(e_k)$$

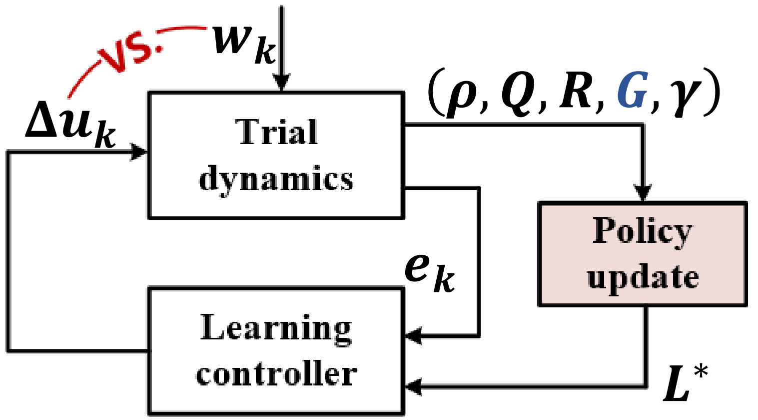
$$\text{s.t. } e_{k+1} = e_k - G\Delta u_{k+1} + w_{k+1}$$

Model-based solution

JIANGNAN UNIVERSITY

If known G

$$V(e_k) = c(e_k, \Delta u_{k+1}, w_{k+1}) + \rho V(e_{k+1})$$



$$\Delta u_{j+1} = L_u^* e_k$$
$$w_{j+1} = L_w^* e_k$$

$$L_u^* = \left[R + G^\top PG + G^\top P\left(\gamma^2 I - P\right)^{-1} PG\right]^{-1} \cdot \left[G^\top P + G^\top P\left(\gamma^2 I - P\right)^{-1} P\right]$$

$$L_w^* = \left[\gamma^2 I - P + PG(R + G^\top PG)^{-1} G^\top P\right]^{-1} \cdot \left[P - PG(R + G^\top PG)^{-1} G^\top P\right]$$

Saddle point

$$\min_{\Delta u_{k+1}} \max_{w_{k+1}} V(e_k)$$
$$s.t.\ e_{k+1} = e_k - G\Delta u_{k+1} + w_{k+1}$$

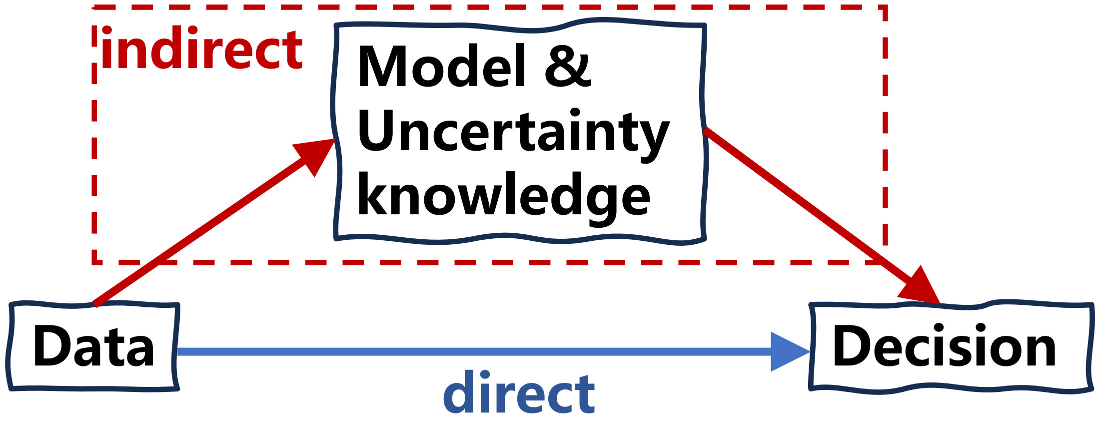
Discounted game algebraic Riccati equation (DGARE)

$$P = \rho P + Q - \rho\begin{bmatrix} -PG & P \end{bmatrix} \begin{bmatrix} R + G^\top PG & -G^\top P \\ -PG & P - \gamma^2 I \end{bmatrix}^{-1} \begin{bmatrix} -G^\top P \\ P \end{bmatrix}$$

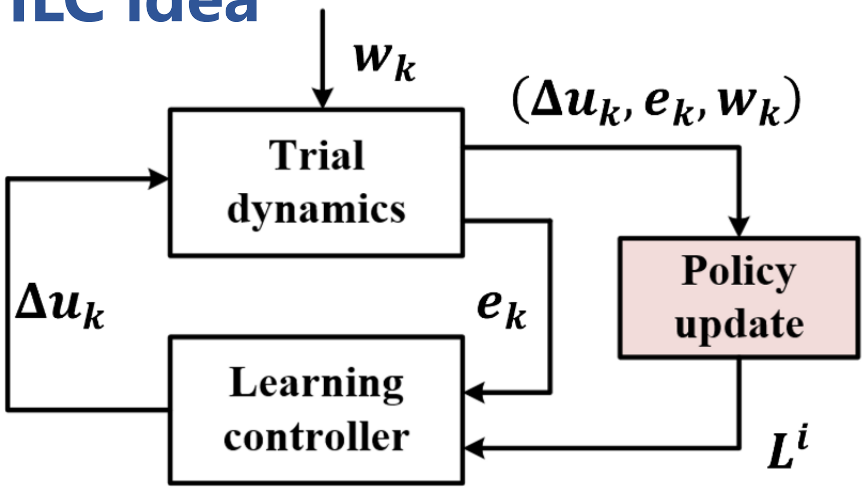
Robust optimal policy

Data-based solution

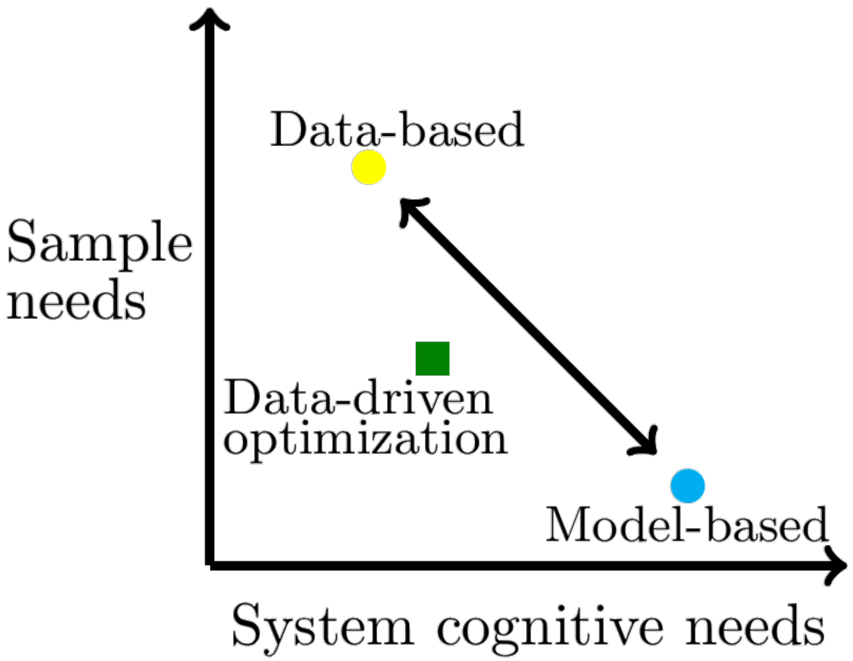
Data-driven control



Standard ILC idea



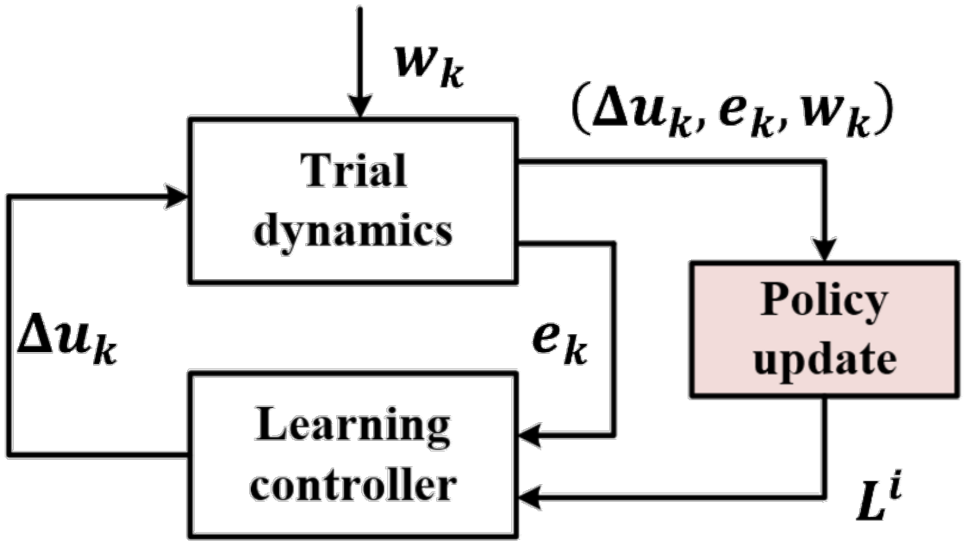
Purpose

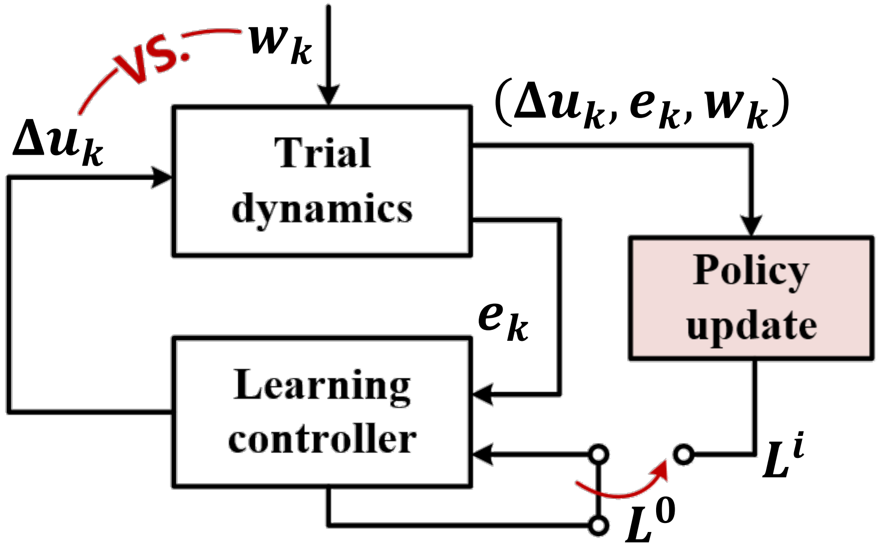


1. **Sample efficiency improvement**
2. **Disturbance attenuation**

Off-policy ILC

On-policy vs. Off-policy

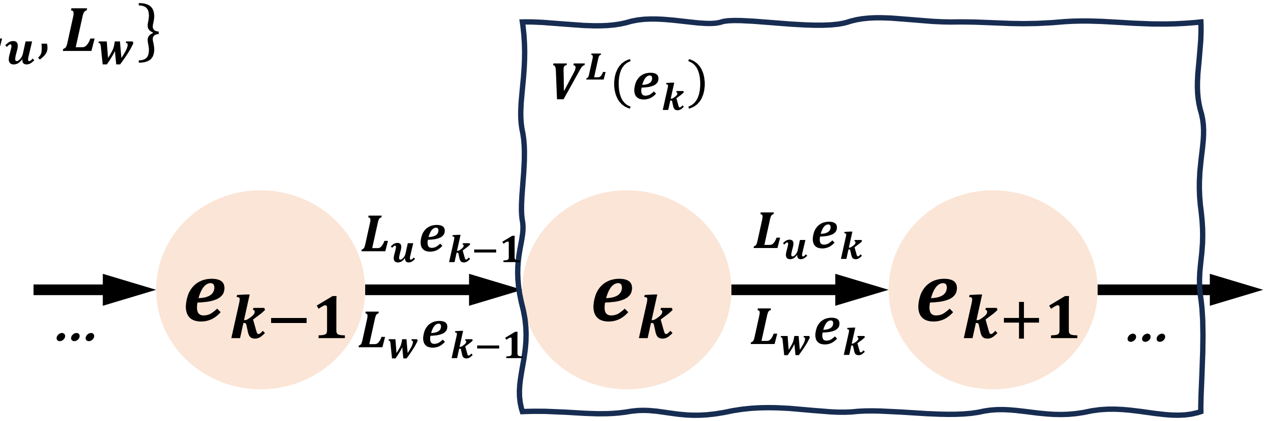




Data-based solution

Value function if given policy $L = \{L_u, L_w\}$

$$V^L(e_k) = \sum_{j=k}^{\infty} \rho^{j-k} c(e_j, L_u e_j, L_w e_j)$$



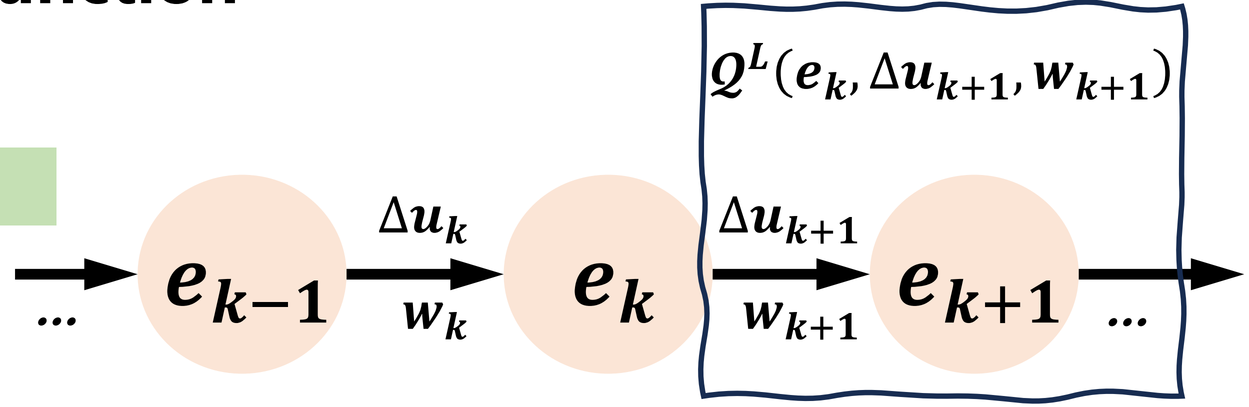
$$Q^L(e_k, L_u e_k, L_w e_k) = V^L(e_k)$$

Define the Q-function: state-action function

state

action

$$Q^L(e_k, \Delta u_{k+1}, w_{k+1})$$
$$= c(e_k, \Delta u_{k+1}, w_{k+1}) + \rho V^L(e_{k+1})$$



Data-based solution

Goal: find the target policy $L = \{L_u, L_w\}$

$$Q^L(e_k, \Delta u_{k+1}, w_{k+1}) = \underbrace{\begin{bmatrix} e_k \\ \Delta u_{k+1} \\ w_{k+1} \end{bmatrix}^\top}_{z_k^\top} \Phi^L \underbrace{\begin{bmatrix} e_k \\ \Delta u_{k+1} \\ w_{k+1} \end{bmatrix}}_{z_k}, \quad \Phi^L = \begin{bmatrix} \phi_{ee} & \phi_{eu} & \phi_{ew} \\ \phi_{ue} & \phi_{uu} & \phi_{uw} \\ \phi_{we} & \phi_{wu} & \phi_{ww} \end{bmatrix}$$

unknown G

known G

$$L_u = \left(\phi_{uu} - \phi_{uw}\phi_{ww}^{-1}\phi_{wu}\right)^{-1} \cdot \left(\phi_{uw}\phi_{ww}^{-1}\phi_{we} - \phi_{ue}\right)$$

$$L_w = \left(\phi_{ww} - \phi_{wu}\phi_{uu}^{-1}\phi_{uw}\right)^{-1} \cdot \left(\phi_{wu}\phi_{uu}^{-1}\phi_{ue} - \phi_{we}\right)$$

$$\Phi^{L^*} = \begin{bmatrix} Q + \rho P & -\rho P G & \rho P \\ -\rho G^\top P & \rho(G^\top P G + R) & -\rho G^\top P \\ \rho P & -\rho P G & \rho\left(P - \gamma^2 I\right) \end{bmatrix}$$

Data-based solution

Solve the **Bellman equation** in a model-free way

$$Q^L(e_k, \Delta u_{k+1}, w_{k+1}) = c(e_k, \Delta u_{k+1}, w_{k+1}) + \rho Q^L(e_{k+1}, L_u e_{k+1}, L_w e_{k+1})$$

$$\Downarrow$$

$$z_k^\top \Phi^L z_k = z_k^\top \begin{bmatrix} Q & & \\ & \rho R & \\ & & \gamma^2 I \end{bmatrix} z_k + \rho \underbrace{\begin{bmatrix} e_{k+1} \\ L_u e_{k+1} \\ L_w e_{k+1} \end{bmatrix}^\top}_{\chi_k^\top} \Phi^L \underbrace{\begin{bmatrix} e_{k+1} \\ L_u e_{k+1} \\ L_w e_{k+1} \end{bmatrix}}_{\chi_k}$$

Data-based solution

Data collection

$$z_k^\top \Phi^L z_k = z_k^\top W z_k + \rho \chi_k^\top \Phi^L \chi_k$$

$$Z = \begin{bmatrix} e_{k_1} & \cdots & e_{k_\eta} \\ \Delta u_{k_1+1} & \cdots & \Delta u_{k_\eta+1} \\ w_{k_1+1} & \cdots & w_{k_\eta+1} \end{bmatrix} \qquad E_i = \begin{bmatrix} e_{k_1+1} & \cdots & e_{k_\eta+1} \\ L_u^i e_{k_1+1} & \cdots & L_u^i e_{k_\eta+1} \\ L_w^i e_{k_1+1} & \cdots & L_w^i e_{k_\eta+1} \end{bmatrix}$$

$$\begin{array}{ccc} \underbrace{\qquad}_{z_{k_1}} & & \underbrace{\qquad}_{z_{k_\eta}} \end{array} \qquad \begin{array}{ccc} \underbrace{\qquad}_{\chi_{k_1+1}^i} & & \underbrace{\qquad}_{\chi_{k_\eta+1}^i} \end{array}$$

$$Z^\top \Phi^{L^{i+1}} Z = Z^\top W Z + \rho E_i^\top \Phi^{L^{i+1}} E_i$$

Generalized Sylvester Equation: MATLAB command *dlyap*

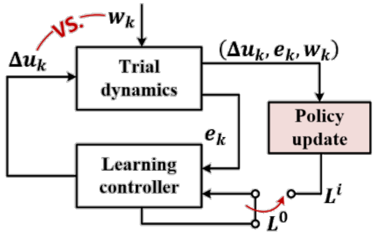
Data-based solution

Off-policy direct data-based ILC

Behavioral systems theory

LQR learning

$$H_J(u_{[0,K-1]}) = \begin{bmatrix} u_0 & \cdots & u_{K-J} \\ \vdots & \ddots & \vdots \\ u_{J-1} & \cdots & u_{K-1} \end{bmatrix}$$



Step 1 Run initial policy to collect data

Step 2 Sort the data to construct matrix

Step 3 Learn the optimal policy

Step 4 Switch to the optimal policy

Issues:

1. How many data should we collect?
2. How to ensure to learn the optimal policy?

Contents

Willems' Fundamental Lemma:
All trajectories of controllable linear systems are constructible from finitely many previous trajectories

Sample efficiency

JIANGNAN UNIVERSITY

To estimate

$$\Phi^L = \begin{bmatrix} \phi_{ee} & \phi_{eu} & \phi_{ew} \\ \phi_{ue} & \phi_{uu} & \phi_{uw} \\ \phi_{we} & \phi_{wu} & \phi_{ww} \end{bmatrix}$$

$$Z^\top \Phi^{L^{i+1}} Z = Z^\top W Z + \rho E_i^\top \Phi^{L^{i+1}} E_i$$

Generalized Sylvester equation solvable

Z, E_i **are full row rank**

How many trials needed?

Willems' Fundamental Lemma:

All trajectories of controllable linear systems are constructible from finitely many previous trajectories

Design a persistently exciting (PE) input of sufficient order

Hankel matrix with order J:

$$H_J(u_{[0,K-1]}) = \begin{bmatrix} u_0 & \cdots & u_{K-J} \\ \vdots & \ddots & \vdots \\ u_{J-1} & \cdots & u_{K-1} \end{bmatrix}$$

full row rank

Sample efficiency

How about disturbance?

$$e_{k+1} = e_k - G\Delta u_{k+1} + w_{k+1}$$

$$e_{k+1} = e_k + [-G \quad I]\,\bar{u}_k$$

$$\bar{u}_k = \begin{bmatrix} \Delta u_{k+1} \\ w_{k+1} \end{bmatrix}$$

Augmented input vector

$$\bar{u}_{[0,K-1]} = [\bar{u}_0^\top \quad \cdots \quad \bar{u}_{K-1}^\top]^\top$$

PE of sufficient order

$\bar{u}_{[0,K-1]}$ is PE of order J

$\xrightarrow{\text{Willems' Fundamental Lemma}}$

$$\begin{bmatrix} H_1(e_{[0,K-J+1]}) \\ H_J(\bar{u}_{[0,K-1]}) \end{bmatrix}$$

full row rank

$\xrightarrow{J=1}$

$$\begin{bmatrix} H_1(e_{[0,K+1]}) \\ H_1(u_{[1,K]}) \\ H_1(w_{[1,K]}) \end{bmatrix}$$

Enough data for Z

full row rank: $K \geq ?$

Convergence analysis

JIANGNAN UNIVERSITY

The policy iteration is stabilizing

$$Z^\top \Phi^{L^{i+1}} Z = Z^\top W Z + \rho E_i^\top \Phi^{L^{i+1}} E_i$$

$\chi_{k+1}^i = \Theta_i z_k$

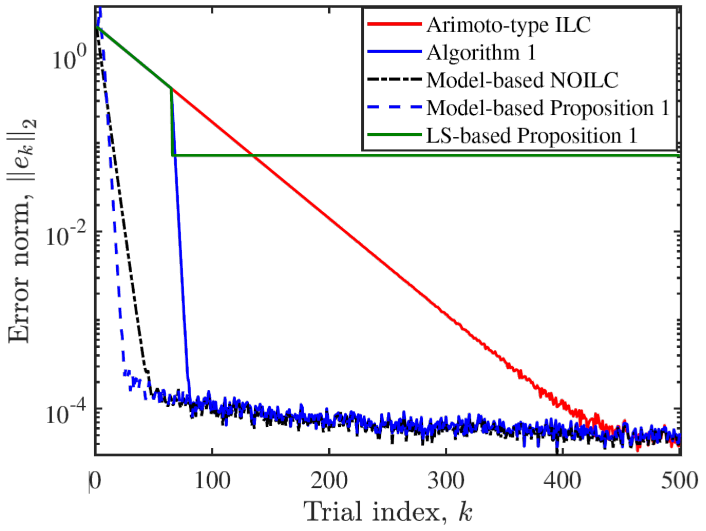
Z, E_i are full row rank

$$\Phi^{L^{i+1}} = W + \rho \Theta_i^\top \Phi^{L^{i+1}} \Theta_i$$

Model-based solution

$$\Phi^{L^*} = \begin{bmatrix} Q + \rho P & -\rho P G & \rho P \\ -\rho G^\top P & \rho(G^\top P G + R) & -\rho G^\top P \\ \rho P & -\rho P G & \rho\left(P - \gamma^2 I\right) \end{bmatrix}$$

Contents



Arimoto-type ILC
Algorithm 1
Model-based NOILC
Model-based Proposition 1
LS-based Proposition 1
10^0
10^{-2}
10^{-4}
Error norm, $\|e_k\|_2$
0
100
200
300
400
500
Trial index, k

Numerical simulation

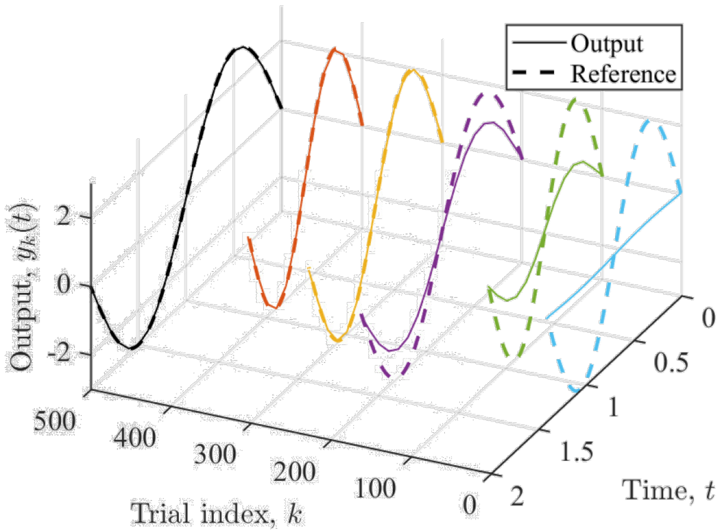
JIANGNAN UNIVERSITY

Linear time-invariant system

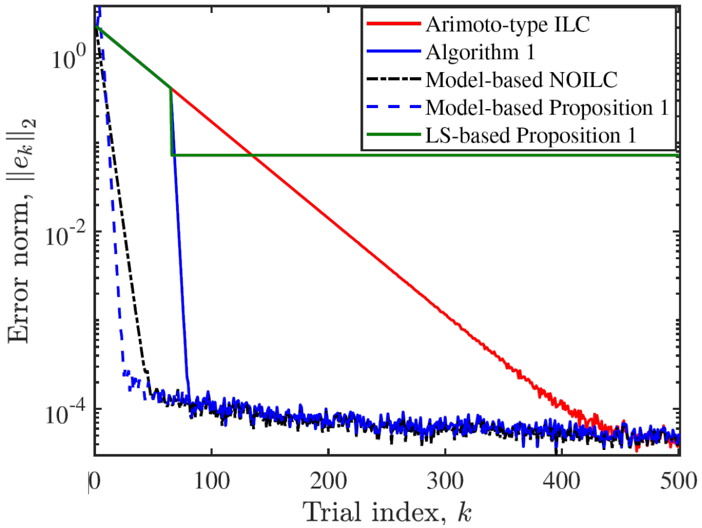
$$A = \begin{bmatrix} 0.90644 & -0.0816012 & -0.0005 \\ 0.074349 & 0.90121 & -0.000708383 \\ 0 & 0 & 0.132655 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.00150808 \\ -0.0096 \\ 0.867345 \end{bmatrix}, C = [0 \quad 0 \quad 0.5], D = [0].$$

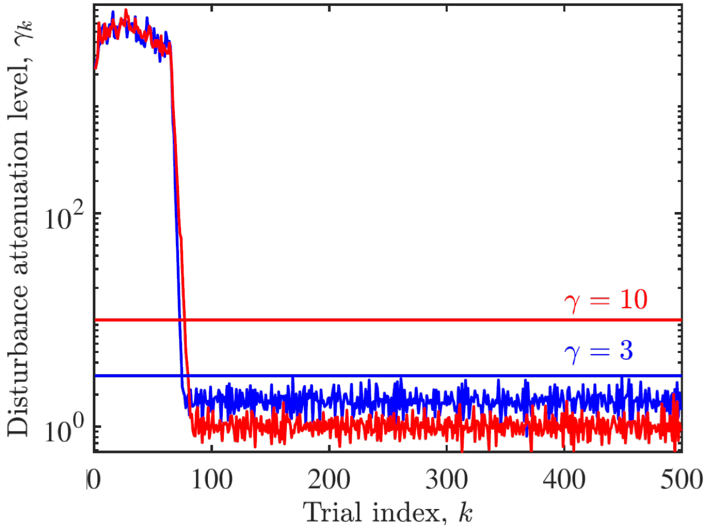
$$\gamma_k = \sqrt{\sum_{k=0}^{\infty} (e_k^\top Q e_k + \Delta u_k^\top R \Delta u_k) / \sum_{k=0}^{\infty} w_k^\top w_k}$$



Tracking outputs



Trial error convergence



Disturbance attenuation level

Conclusion and Future work

Conclusion

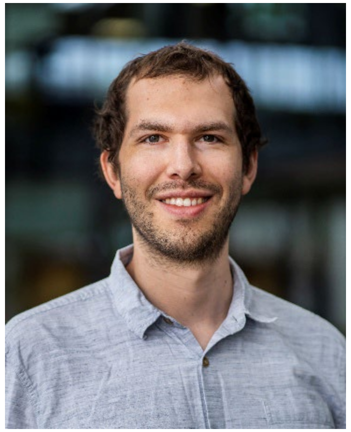
1. Discrete-time ILC for time-scale-varying tasks
2. Robust direct data-based ILC design:
    1) zero-sum game design for general trial-varying disturbances
    2) Off-policy design in the trial domain
3. Sample efficiency assessment and convergence analysis

Future work

1. Sample needs prediction
2. Extension to nonlinear systems
3. ……

Acknowledgements

JIANGNAN UNIVERSITY



EINDHOVEN UNIVERSITY OF TECHNOLOGY

Rodrigo A. Gonzalez

JIANGNAN UNIVERSITY

Hongfeng Tao

UNIVERSITY OF ZIELONA GÓRA

Wojciech Paszke

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Tom Oomen