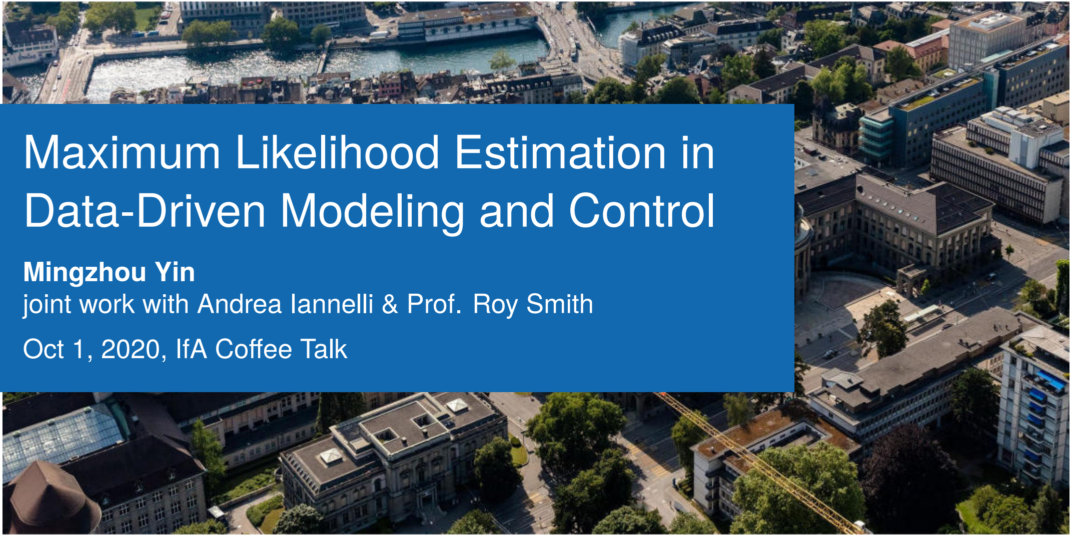
ETH zürich

Automatic Control Laboratory



Maximum Likelihood Estimation in Data-Driven Modeling and Control

Mingzhou Yin
joint work with Andrea Iannelli & Prof. Roy Smith

Oct 1, 2020, IfA Coffee Talk

Outline

Introduction

- Data-driven method as an input-output mapping
- Problem with noisy data in data-driven modelling

Methodology

- Maximum likelihood estimation for the optimal model
- Signal matrix model with an efficient algorithm
- Preconditioning for big data

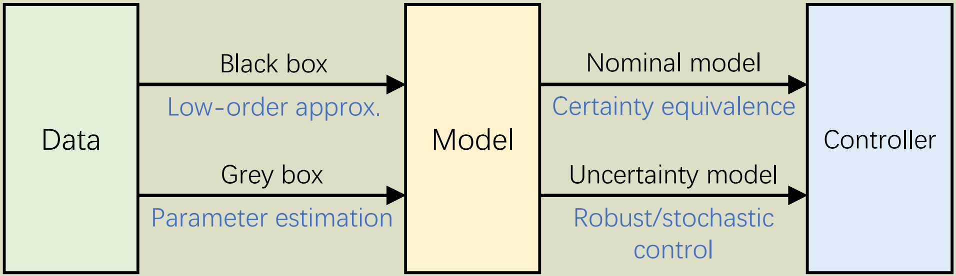
New model in action

- Impulse response estimation
- Data-driven predictive control

Data-driven in control

- Most control applications are data-driven
- ... but were restricted by control design tools $\rightarrow$ model

Paradigm of system identification



Unleash the power of data-driven

- Challenge: more complex systems

Problems	**Good things**
Modelling is hard	Data is big
An underlying low-order model?	Need for low-order models?

- Model-free? Low-order model $\rightarrow$ input-output mapping
- **Willems' fundamental lemma**[1]: one long input-output trajectory can characterize all trajectories of a shorter horizon.
- ... via signal matrices

[1] Willems, 2005

In a world without noise...

Input-output mapping based on Willems' fundamental lemma[2]

Given: input-output data $(u_i^d, y_i^d)_{i=0}^{N-1}$.

Assume: finite-dimensional LTI, controllability, sufficient persistency of excitation, upper bound on system order L_0.

Define: Hankel signal matrices

$$U_p = \begin{bmatrix} u_0^d & u_1^d & \cdots & u_{M-1}^d \\ \vdots & \vdots & \ddots & \vdots \\ u_{L_0-1}^d & u_{L_0}^d & \cdots & u_{M+L_0-2}^d \end{bmatrix}, U_f = \begin{bmatrix} u_{L_0}^d & u_{L_0+1}^d & \cdots & u_{M+L_0-1}^d \\ \vdots & \vdots & \ddots & \vdots \\ u_{L-1}^d & u_L^d & \cdots & u_{N-1}^d \end{bmatrix},$$

and similarly for Y_p and Y_f.

[2] Markovsky, 2005

In a world without noise...

Input-output mapping based on Willems' fundamental lemma *(cont'd)*

Input: past trajectory $\mathbf{u}_{\text{ini}} = (u_i)_{i=-L_0}^{-1}$, $\mathbf{y}_{\text{ini}} = (y_i)_{i=-L_0}^{-1}$, input trajectory $\mathbf{u} = (u_i)_{i=0}^{L'-1}$, $L' = L - L_0$.

- Solve the linear system $\begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{y}_{\text{ini}} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} U_p \\ Y_p \\ U_f \end{bmatrix} g \ (\star)$ for g.

Output: output trajectory $\mathbf{y} = Y_f g$.

- A 'model' of $\mathbf{y} = f(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}})$?
- ... but implicit & overparametrized
- As predictor in MPC setup, **DeePC**[3]

[3] Coulson, 2019

... until noise ruins everything

- The linear system $(\star)$ is underdetermined
- In **noise-free** case, still well-defined due to

$$\text{rank}\,(H) := \text{rank}\left(\begin{bmatrix} U_p \\ U_f \\ Y_p \\ Y_f \end{bmatrix}\right) \leq L_0 + n_u L$$

- ... a core identity in behavioral system theory (and subspace identification)
- But when signal matrices are **noisy**,

$$H \,:\, \text{full row rank a.s.}$$

- **Consequence:** $\forall\, \mathbf{y},\, \exists\, g,\, (\star)$ is satisfied

Remedies

- **Subspace identification**[4]:
  (structured) low-rank approximation of H
- **Least-norm solution**[5]:

$$g_{\text{pinv}} = \begin{bmatrix} U_p \\ Y_p \\ U_f \end{bmatrix}^{\dagger} \begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{y}_{\text{ini}} \\ \mathbf{u} \end{bmatrix}$$

- **Regularization** (regularized DeePC)

$$\text{min } J_{\text{ctr}}(\mathbf{u}, \mathbf{y}) + \lambda_g \|g\|_p^p + \lambda_y \|\hat{\mathbf{y}}_{\text{ini}} - \mathbf{y}_{\text{ini}}\|_p^p$$

Problems

- Not so 'data-driven'
- Computationally hard
- Sub-optimal
- Biased towards control objective
- Hyperparameter tuning

[4] Moonen, 1989
[5] Sedghizadeh, 2018

Road to a better remedy

- **One central question**: what is the 'optimal' solution to $(\star)$ with noise?
- **Difficulty**: errors-in-variables structure
  - Noise on both sides: $\begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{y}_{\text{ini}} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} U_p \\ Y_p \\ U_f \end{bmatrix} g$
- **Our approach**: maximum likelihood estimation
- **Core idea**: find the g that optimizes the likelihood of observing both the **predicted output trajectory** $\mathbf{y}$ and the **residual of** $(\star)$.

A little boring derivation...

- **Noise model** (example): No input noise,

$$y_i^d = y_i^{d,0} + w_i^d,\ (w_i^d)_{i=0}^{N-1} \sim \mathcal{N}(0, \sigma^2 \mathbb{I}),$$

$$\mathbf{y}_{\mathsf{ini}} = \mathbf{y}_{\mathsf{ini}}^0 + \mathbf{w}_p,\ \mathbf{w}_p \sim \mathcal{N}(0, \sigma_p^2 \mathbb{I}).$$

- Define

$$\hat{\mathbf{y}} = \begin{bmatrix} \epsilon_y \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} Y_p g - \mathbf{y}_{\mathsf{ini}} \\ Y_f g \end{bmatrix} = \left(g^\mathsf{T} \otimes \mathbb{I}\right) \mathsf{vec}\left(\begin{bmatrix} Y_p \\ Y_f \end{bmatrix}\right) - \begin{bmatrix} \mathbf{y}_{\mathsf{ini}} \\ \mathbf{0} \end{bmatrix}.$$

- Then,

$$\hat{\mathbf{y}} | g \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ Y_f^0 g \end{bmatrix}, \Sigma_y\right), \Sigma_y = \left(g^\mathsf{T} \otimes \mathbb{I}\right) \mathsf{cov}\left[\mathsf{vec}\left(\begin{bmatrix} Y_p \\ Y_f \end{bmatrix}\right)\right] (g \otimes \mathbb{I}) + \begin{bmatrix} \sigma_p^2 \mathbb{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

The maximum likelihood estimator

$$\underset{g \in \mathcal{G}}{\text{minimize}} \quad \underbrace{\text{logdet}(\Sigma_y(g))}_{\text{Uncertainty of prediction}} + \underbrace{\begin{bmatrix} Y_p g - \mathbf{y}_{\text{ini}} \\ \mathbf{0} \end{bmatrix}^{\mathsf{T}} \Sigma_y^{-1}(g) \begin{bmatrix} Y_p g - \mathbf{y}_{\text{ini}} \\ \mathbf{0} \end{bmatrix}}_{\text{Deviation from past output measurements}}$$

- $\mathcal{G} = \left\{ g \in \mathbb{R}^M \,\middle|\, \begin{bmatrix} U_p \\ U_f \end{bmatrix} g = \begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{u} \end{bmatrix} \right\}$ is the parameter space.
- **Signal matrix model**: $\mathbf{y} = Y_f \, g^*$

Approximation

- Approximate Σ_y with its diagonal part
  - Holds exactly if Page signal matrices are used, i.e.,

$$\begin{bmatrix} Y_p \\ Y_f \end{bmatrix} = \begin{bmatrix} y_0^d & y_L^d & \cdots & y_{(M-1)L}^d \\ \vdots & \vdots & \ddots & \vdots \\ y_{L-1}^d & y_{2L-1}^d & \cdots & y_{ML-1}^d \end{bmatrix}$$

  - ... but worse data efficiency

$$\underset{g \in \mathcal{G}}{\text{minimize}} \; L' \log\left(\|g\|_2^2\right) + L_0 \log\left(\sigma^2 \|g\|_2^2 + \sigma_p^2\right) + \frac{1}{\sigma^2 \|g\|_2^2 + \sigma_p^2} \|Y_p g - \mathbf{y}_{\text{ini}}\|_2^2 .$$

An iterative algorithm of signal matrix model

- Solve the approximated problem by SQP

$$
\begin{aligned}
g^{k+1} = \quad & \underset{g}{\text{argmin}} \quad \lambda(g^k)\,\|g\|_2^2 + \|Y_p g - \mathbf{y}_{\text{ini}}\|_2^2 \\
& \text{subject to} \quad \begin{bmatrix} U_p \\ U_f \end{bmatrix} g = \begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{u} \end{bmatrix},
\end{aligned}
$$

where

$$
\lambda(g^k) = \frac{L'\sigma_p^2}{\|g^k\|_2^2} + L\sigma^2.
$$

- Closed-form solution in the form of

$$
g^{k+1} = \mathcal{P}\left(\lambda(g^k)\right)\mathbf{y}_{\text{ini}} + \mathcal{Q}\left(\lambda(g^k)\right)\begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \mathbf{u} \end{bmatrix}. \quad \text{(CL)}
$$

Algorithm 1 Signal matrix model

Given: $U_p, U_f, Y_p, Y_f, \sigma, \sigma_p, \epsilon$.
Input: $\mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}}, \mathbf{u}$.
$k \leftarrow 0$, $g^0 \leftarrow g_{\text{pinv}}$
repeat
    Calculate g^{k+1} with (CL).
    $k \leftarrow k + 1$
until $\left\|g^k - g^{k-1}\right\| < \epsilon \left\|g^{k-1}\right\|$
Output: $g_{\text{SMM}} = g^k$, $\mathbf{y} = Y_f g^k$.

Data, more data!

- When abundant data are available,

$$\text{dim}(g) \approx \text{data length } N \gg \text{prediction horizon } L$$

- **Problem**: high-dimensional optimization
- ... but $2L$ independent basis vectors should be enough for everything within the prediction horizon.
- **Solution**: precondition signal matrices to make dim$(g) = 2L$

Preconditioning of signal matrices

1) SVD of signal matrices:

$$\mathrm{col}\,(U_p, U_f, Y_p, Y_f) = WSV^{\mathsf{T}} \in \mathbb{R}^{2L\times M}$$

2) Define new signal matrices:

$$\mathrm{col}\left(\tilde{U}_p, \tilde{U}_f, \tilde{Y}_p, \tilde{Y}_f\right) = WS_{2L} \in \mathbb{R}^{2L\times 2L},$$

S_{2L}: the first $2L$ columns of S.

Proposition: The signal matrix models with (U_p, Y_p, U_f, Y_f) and $(\tilde{U}_p, \tilde{Y}_p, \tilde{U}_f, \tilde{Y}_f)$ are equivalent.

Problem 1: Impulse response estimation

- Impulse response model: $y_t = \sum_{i=0}^{\infty} h_i u_{t-i}$
- Conventional approach: regression from data

$$\underbrace{\begin{bmatrix} y_0^d \\ y_1^d \\ \vdots \\ y_{N-1}^d \end{bmatrix}}_{y_N} = \underbrace{\begin{bmatrix} u_0^d & u_{-1}^d & \cdots & u_{1-n}^d \\ u_1^d & u_0^d & \cdots & u_{2-n}^d \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-1}^d & u_{N-2}^d & \cdots & u_{N-n}^d \end{bmatrix}}_{\Phi_N} \underbrace{\begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{n-1} \end{bmatrix}}_{h}$$

- **Least-squares** solution: $\hat{h} = \text{argmin} \ \|y_N - \Phi_N h\|_2^2 = \left(\Phi_N^\mathsf{T}\Phi_N\right)^{-1} \Phi_N^\mathsf{T} y_N$

An alternative approach

- **Problem**:
  1) truncation error from $(h_i)_{i=n}^{\infty}$,
  2) unknown past inputs: $\left(u_i^d\right)_{i=1-n}^{-1}$
- Biased & incorrect even without noise
- **Alternative**: **signal matrix model** with

$$\mathbf{u}_{\text{ini}} = \mathbf{0},\ \mathbf{y}_{\text{ini}} = \mathbf{0},\ \mathbf{u} = \text{col}(1, \mathbf{0}),\ \sigma_p = 0,\ L' = n.$$

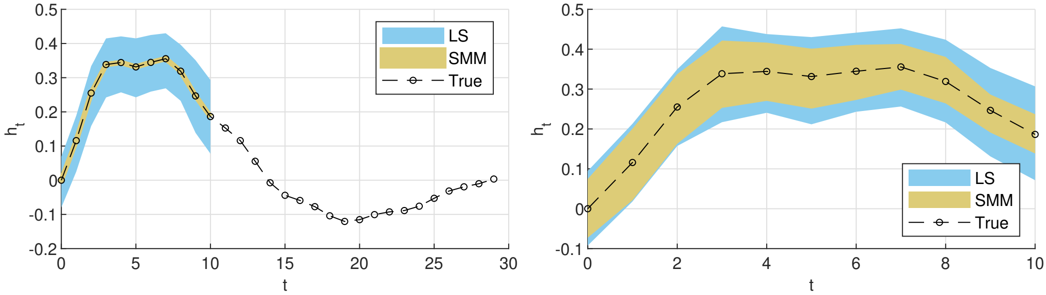
Numerical tests

- **Parameters**: $N = 50,\ L_0 = 4,\ n = L' = 11,\ \sigma^2 = 0.01,\ (u_i^d)_{i=0}^{N-1} \sim \mathcal{N}(0, \mathbb{I})$
- **Example 1**: big truncation error, known past inputs

$$G_1(z) = \frac{0.1159(z^3 + 0.5z)}{z^4 - 2.2z^3 + 2.42z^2 - 1.87z + 0.7225}$$

- **Example 2**: negligible truncation error, unknown past inputs

$$G_2(z) = \frac{0.9183z}{z^2 + 0.24z + 0.36}.$$



Noise-free $\qquad$ $\sigma^2 = 0.01$

Figure: Impulse response estimation of **Example 1**.

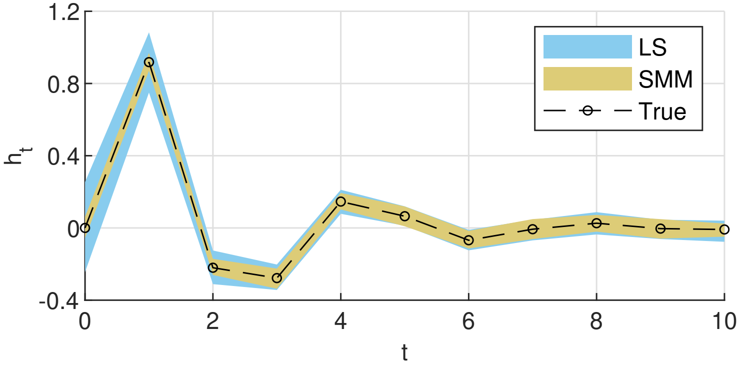


Figure: Impulse response estimation of **Example 2**.

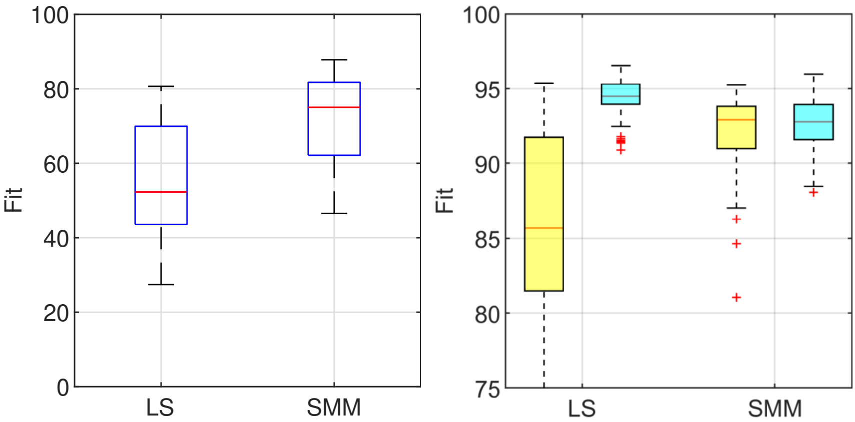


Figure: Model fitting. Yellow : unknown past inputs, cyan : known past inputs.

Problem 2: Data-driven predictive control

- **Receding horizon control** structure:

$$\underset{\mathbf{u},\mathbf{y}}{\text{minimize}} \quad \underbrace{\sum_{k=0}^{L'-1} \left(\|y_k - r_{t+k}\|_Q^2 + \|u_k\|_R^2 \right)}_{J_{\text{ctr}}(\mathbf{u},\mathbf{y})}$$

$$\text{subject to} \quad \mathbf{y} = f(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}}), \mathbf{u} \in \mathcal{U}, \mathbf{y} \in \mathcal{Y}$$

  $f(\cdot)$: data-driven input-output mapping

- **Signal matrix model** as the predictor:

$$f_{\text{SMM}}(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}}) = Y_f \, g_{\text{SMM}}(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}})$$

- **Problem**: implicit constraint

Signal matrix model predictive control (SMM-PC)

- **Observation**: $\|g\|_2^2$ doesn't change much at each time step.
- ... and signal matrix model is only iterative w.r.t. $\|g\|_2^2$.
- **Solution**: warm-starting from g at previous step & one iteration

$$
\begin{aligned}
\underset{\mathbf{u},\mathbf{y}}{\text{minimize}} \quad & J_{\text{ctr}}(\mathbf{u},\mathbf{y}) \\
\text{subject to} \quad & \mathbf{y} = Y_f\, g^t, \\
& g^t = \mathcal{P}(g^{t-1})\,\mathbf{y}_{\text{ini}} + \mathcal{Q}(g^{t-1})\,\tilde{\mathbf{u}}, \\
& \mathbf{u} \in \mathcal{U}, \mathbf{y} \in \mathcal{Y}.
\end{aligned}
$$

Methods with other remedies

- **Subspace predictive control** (least-norm solution):

$$f_{\text{Sub}}(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}}) = Y_f \, g_{\text{pinv}}(\mathbf{u}; \mathbf{u}_{\text{ini}}, \mathbf{y}_{\text{ini}})$$

- **Regularized DeePC**:

$$\begin{aligned} \underset{\mathbf{u},\mathbf{y},g,\hat{\mathbf{y}}_{\text{ini}}}{\text{minimize}} \quad & J_{\text{ctr}}(\mathbf{u}, \mathbf{y}) + \lambda_g \, \|g\|_p^p + \lambda_y \, \|\hat{\mathbf{y}}_{\text{ini}} - \mathbf{y}_{\text{ini}}\|_p^p \\ \text{subject to} \quad & \begin{bmatrix} U_p \\ Y_p \\ U_f \\ Y_f \end{bmatrix} g = \begin{bmatrix} \mathbf{u}_{\text{ini}} \\ \hat{\mathbf{y}}_{\text{ini}} \\ \mathbf{u} \\ \mathbf{y} \end{bmatrix}, \mathbf{u} \in \mathcal{U}, \mathbf{y} \in \mathcal{Y}, \end{aligned}$$

Signal matrix model & regularized DeePC

- **Resemblance** in objective function:

$$\text{minimize} \;\; \lambda(g^k)\,\|g\|_2^2 + \|Y_p g - \mathbf{y}_{\text{ini}}\|_2^2 \qquad \text{(SMM)}$$

$$\text{minimize} \;\; J_{\text{ctr}}(\mathbf{u},\mathbf{y}) + \lambda_g\,\|g\|_p^p + \lambda_y\,\|Y_p g - \mathbf{y}_{\text{ini}}\|_p^p \qquad \text{(DeePC)}$$

- **Differences**:
  1) Parameter estimation is isolated from control performance
  2) Hyperparameter tuning is simplified to noise level estimation

Numerical tests

- **Parameters**:

$$N = 200,\ L_0 = 4,\ L' = 11,\ \sigma^2 = \sigma_p^2 = 1,\ (u_i^d)_{i=0}^{N-1} \sim \mathcal{N}(0, \mathbb{I})$$

$$Q = R = 1, \mathcal{U} = \mathcal{Y} = \mathbb{R}^{L'},\ r_t = 0.5 \sin\left(\frac{\pi}{10} t\right)$$

- **System**:

$$G_1(z) = \frac{0.1159(z^3 + 0.5z)}{z^4 - 2.2z^3 + 2.42z^2 - 1.87z + 0.7225}$$

- Known noise level for SMM-PC
- Optimally tuned DeePC with an oracle
- Benchmark approach: MPC with noise-free state measurements

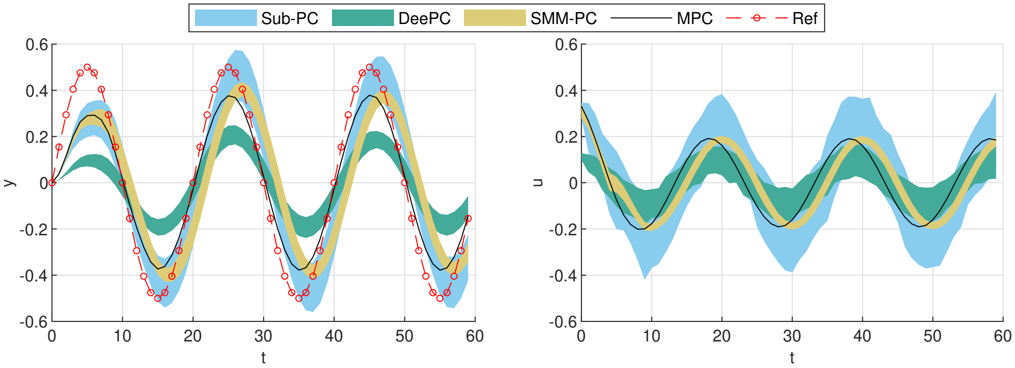


Figure: Closed-loop trajectories within one standard deviation.

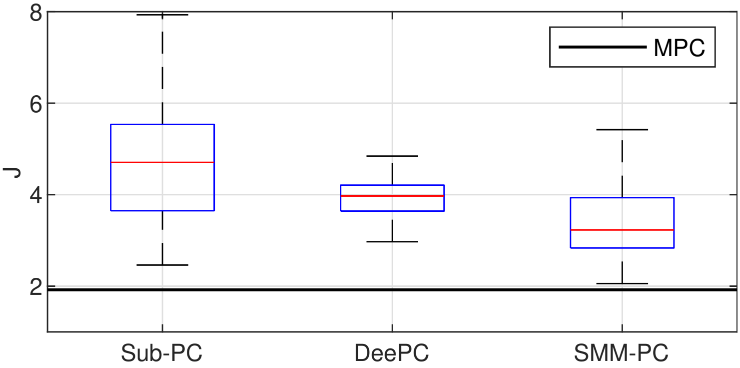


Figure: Total true stage cost $J = \sum_{k=0}^{N_c-1} \left(\|y_k^0 - r_k\|_Q^2 + \|u_k\|_R^2 \right)$.

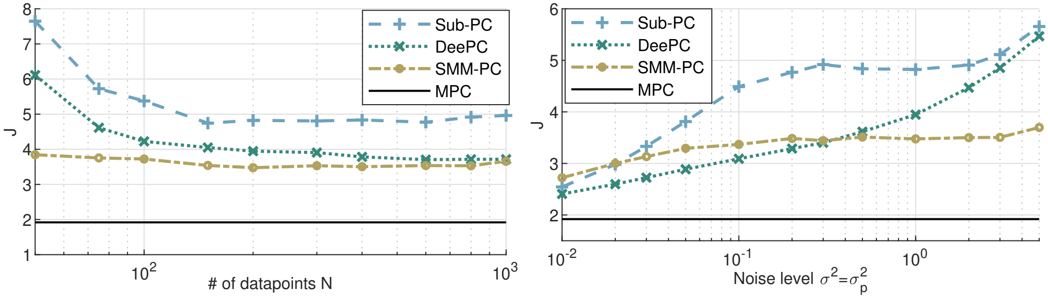


Figure: Effect of data sizes & noise levels.

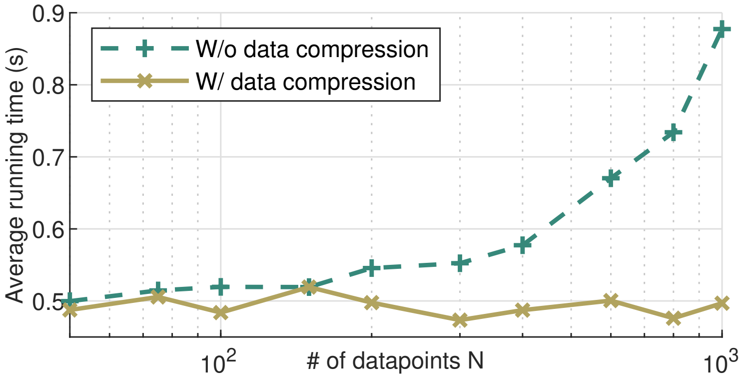


Figure: Average computing time for SMM-PC.

Conclusion & outlook

- Statistical framework for data-driven modelling with noise
- Effective in both impulse response estimation & predictive control

- Incorporating online data
- Input design with signal matrix model
- Causality & time-invariance constraints
- ...

ETH zürich

AUTOMATIC CONTROL LABORATORY

Mingzhou Yin
myin@ethz.ch