Data-Enabled Predictive Control

Florian Dörfler

ETH Zürich

Oberwolfach Workshop

Analysis of Data-Driven Optimal Control

Acknowledgements

Jeremy Coulson

Linbin Huang

Paul Beuchat

John Lygeros

Ivan Markovsky

Further:
Ezzat Elokda,
Daniele Alpago,
Jianzhe (Trevor) Zhen,
Saverio Bolognani,
Andrea Favato,
Paolo Carlet, &
IfA DeePC team

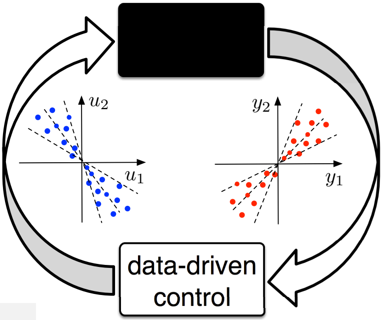
Control in a data-rich world

- ***model-based decision-making***

  data $\xrightarrow{\text{sys id}}$ model + uncertainty $\rightarrow$ control

- ever-growing trend in CS & applications:
  data-driven control by-passing models

Q: Why give up physical modeling & reliable model-based algorithms ?



Data-driven control is ***viable alternative*** when

- models are too complex to be useful
  (e.g., fluids, wind farms, & soft robotics)
- first-principle models are not conceivable
  (e.g., human-in-the-loop, biology, & perception)
- modeling & system ID is too cumbersome
  (e.g., robotics, drives, & electronics applications)

Central promise: *It is often easier to learn control policies directly from data, rather than learning a model.*

Example: PID [Åström, '73]

Abstraction reveals pros & cons

indirect (model-based) ***data-driven control***

minimize control cost (u, x)
subject to (u, x) satisfy state-space model
where x estimated from (u, y) & model
where model identified from (u^d, y^d) data

outer optimization
middle opt.
inner opt.

separation & certainty equivalence ($\rightarrow$ LQG case)

no separation ($\rightarrow$ ID-4-control)

$\rightarrow$ nested multi-level optimization problem

direct (black-box) ***data-driven control***

minimize control cost (u, y)
subject to (u, y) consistent with (u^d, y^d) data

$\rightarrow$ ***trade-offs***

modular vs. end-2-end
suboptimal (?) vs. optimal
convex vs. non-convex (?)

Additionally: all above should be min-max or $\mathbb{E}(\cdot)$ accounting for ***uncertainty*** ...

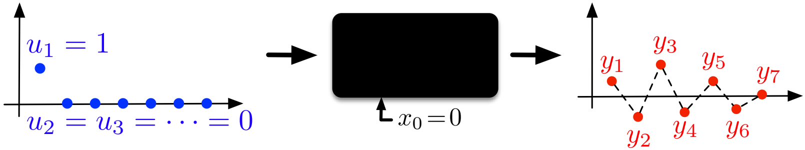
A direct approach: dictionary + MPC

① trajectory ***dictionary learning***

- motion primitives / basis functions
- theory: Koopman & Liouville
  practice: (E)DMD & particles

② ***MPC*** optimizing over dictionary span

→ huge ***theory vs. practice*** gap
→ back to basics: ***impulse response***



dynamic matrix control (Shell, 1970s): ***predictive control from raw data***

$$y_{\text{future}}(t) = \begin{bmatrix} y_1 & y_2 & y_3 & \dots \end{bmatrix} \cdot \begin{bmatrix} u_{\text{future}}(t) \\ u_{\text{future}}(t-1) \\ u_{\text{future}}(t-2) \\ \vdots \end{bmatrix}$$

today : arbitrary & finite data, … stochastic & nonlinear, … back to ID

Contents

I. Non-Parametric Data-Driven System Theory

I. Markovsky and F. Dörfler. *Identifiability in the Behavioral Setting.* [`link`]

II. Data-Enabled Predictive Control (DeePC)

J. Coulson, J. Lygeros, and F. Dörfler. *Distributionally Robust Chance Constrained Data-enabled Predictive Control*. [`arXiv:2006.01702`].

III. Bridging to System Identification

F. Dörfler., J. Coulson, and I. Markovsky. *Bridging direct & indirect data-driven control formulations via regularizations and relaxations*. [`arXiv:2101.01273`].

IV. Applications: End-to-End Automation in Energy & Robotics

L. Huang, J. Coulson, J. Lygeros, and F. Dörfler. *Decentralized Data-Enabled Predictive Control for Power System Oscillation Damping*. [`arXiv:1911.12151`].

E. Elokda, J. Coulson, P. Beuchat, J. Lygeros, and F. Dörfler. *Data-Enabled Predictive Control for Quadcopters*. [`link`].

[`click here`] for related publications

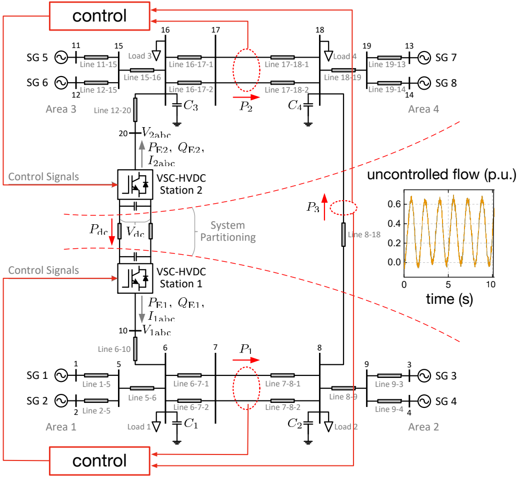
Preview

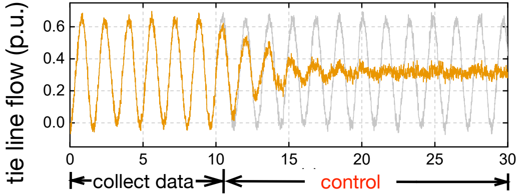
complex 4-area power ***system***:

large ($n = 208$), few sensors (8), nonlinear, noisy, stiff, input constraints, & decentralized control specifications

control objective: oscillation damping $\boxed{\text{without model}}$

(grid has many owners, models are proprietary, operation in flux, ...)





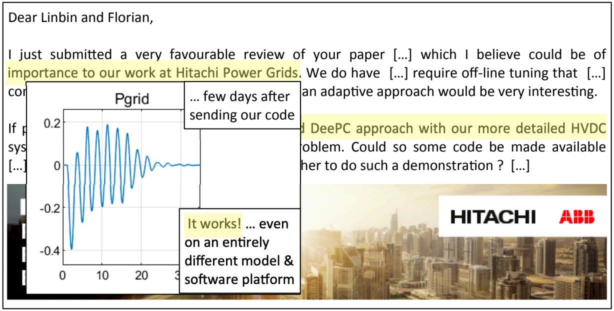
seek a method that ***works reliably***, can be ***efficiently*** implemented, & ***certifiable***

$\rightarrow$ automating ourselves

Reality check: magic or hoax ?

surely, nobody would put apply such a **shaky data-driven method**

- on the **world's most complex engineered system** (the electric grid),
- using the **world's biggest actuators** (Gigawatt-sized HVDC links),
- and subject to **real-time, safety, & stability constraints** . . . right?

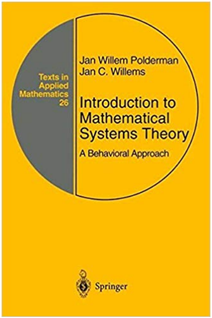


at least someone believes that DeePC is practically useful . . .

Behavioral view on LTI systems

Definition: A discrete-time ***dynamical system*** is a 3-tuple $(\mathbb{Z}_{\geq 0}, \mathbb{W}, \mathscr{B})$ where

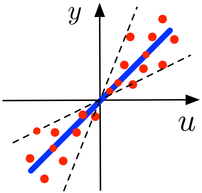
$$\left.\begin{array}{ll} \text{(i)} & \mathbb{Z}_{\geq 0} \text{ is the discrete-time axis,} \\ \text{(ii)} & \mathbb{W} \text{ is a signal space, and} \\ \text{(iii)} & \mathscr{B} \subseteq \mathbb{W}^{\mathbb{Z}_{\geq 0}} \text{ is the behavior.} \end{array}\right\} \; \mathscr{B} \text{ is the set of all trajectories}$$



Definition: The dynamical system $(\mathbb{Z}_{\geq 0}, \mathbb{W}, \mathscr{B})$ is

(i) ***linear*** if $\mathbb{W}$ is a vector space & $\mathscr{B}$ is a subspace of $\mathbb{W}^{\mathbb{Z}_{\geq 0}}$

(ii) and ***time-invariant*** if $\mathscr{B} \subseteq \sigma\mathscr{B}$, where $\sigma w_t = w_{t+1}$.



LTI system $=$ shift-invariant subspace of trajectory space

LTI systems and matrix time series

foundation of state-space subspace system ID & signal recovery algorithms

$\big(u(t), y(t)\big)$ satisfy recursive ***difference equation***

$$b_0 u_t + b_1 u_{t+1} + \ldots + b_n u_{t+n} + a_0 y_t + a_1 y_{t+1} + \ldots + a_n y_{t+n} = 0$$

(ARX / kernel representation)

$\Longleftarrow$

under assumptions

$\Longrightarrow$

$\begin{bmatrix} 0 & b_0 & a_0 & b_1 & a_1 & \ldots & b_n & a_n & 0 \end{bmatrix}$ in left nullspace of ***trajectory matrix*** (collected data)

$$\mathscr{H}_T\begin{pmatrix} u^d \\ y^d \end{pmatrix} = \begin{bmatrix} \begin{pmatrix} u^d_{1,1} \\ y^d_{1,1} \end{pmatrix} & \begin{pmatrix} u^d_{1,2} \\ y^d_{1,2} \end{pmatrix} & \begin{pmatrix} u^d_{1,3} \\ y^d_{1,3} \end{pmatrix} & \cdots \\ \begin{pmatrix} u^d_{2,1} \\ y^d_{2,1} \end{pmatrix} & \begin{pmatrix} u^d_{2,2} \\ y^d_{2,2} \end{pmatrix} & \begin{pmatrix} u^d_{2,3} \\ y^d_{2,3} \end{pmatrix} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \begin{pmatrix} u^d_{T,1} \\ y^d_{T,1} \end{pmatrix} & \begin{pmatrix} u^d_{T,2} \\ y^d_{T,2} \end{pmatrix} & \begin{pmatrix} u^d_{T,3} \\ y^d_{T,3} \end{pmatrix} & \cdots \end{bmatrix}$$

where $y^d_{t,i}$ is tth sample from ith experiment

Fundamental Lemma [Willems et al. '05], [Markovsky & Dörfler '20]

Given: data $\begin{pmatrix} u_i^d \\ y_i^d \end{pmatrix} \in \mathbb{R}^{m+p}$ & LTI complexity parameters $\begin{cases} \text{lag } \ell \\ \text{order } n \end{cases}$

set of all T-length trajectories $=$

$$\left\{ (u,y) \in \mathbb{R}^{(m+p)T} : \exists x \in \mathbb{R}^n \text{ s.t. } x^+ = Ax + Bu\,,\; y = Cx + Du \right\}$$

parametric state-space model

$$= \quad \text{colspan} \begin{bmatrix} \begin{pmatrix} u_{1,1}^d \\ y_{1,1}^d \end{pmatrix} & \begin{pmatrix} u_{1,2}^d \\ y_{1,2}^d \end{pmatrix} & \begin{pmatrix} u_{1,3}^d \\ y_{1,3}^d \end{pmatrix} & \cdots \\ \begin{pmatrix} u_{2,1}^d \\ y_{2,1}^d \end{pmatrix} & \begin{pmatrix} u_{2,2}^d \\ y_{2,2}^d \end{pmatrix} & \begin{pmatrix} u_{2,3}^d \\ y_{2,3}^d \end{pmatrix} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \begin{pmatrix} u_{T,1}^d \\ y_{T,1}^d \end{pmatrix} & \begin{pmatrix} u_{T,2}^d \\ y_{T,2}^d \end{pmatrix} & \begin{pmatrix} u_{T,3}^d \\ y_{T,3}^d \end{pmatrix} & \cdots \end{bmatrix}$$

non-parametric model from raw data

if and only if the trajectory matrix has rank $m \cdot T + n$ for all $T > \ell$

$$\underbrace{\begin{array}{l}\text{set of all } T\text{-length trajectories } = \\ \Big\{ (u,y) \in \mathbb{R}^{(m+p)T} : \exists x \in \mathbb{R}^n \text{ s.t.} \\ x^+ = Ax + Bu \,,\; y = Cx + Du \Big\}\end{array}}_{\text{parametric state-space model}} \quad = \quad \underbrace{\text{colspan} \begin{bmatrix} \begin{pmatrix} u^d_{1,1} \\ y^d_{1,1} \end{pmatrix} & \begin{pmatrix} u^d_{1,2} \\ y^d_{1,2} \end{pmatrix} & \begin{pmatrix} u^d_{1,3} \\ y^d_{1,3} \end{pmatrix} & \cdots \\ \begin{pmatrix} u^d_{2,1} \\ y^d_{2,1} \end{pmatrix} & \begin{pmatrix} u^d_{2,2} \\ y^d_{2,2} \end{pmatrix} & \begin{pmatrix} u^d_{2,3} \\ y^d_{2,3} \end{pmatrix} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \begin{pmatrix} u^d_{T,1} \\ y^d_{T,1} \end{pmatrix} & \begin{pmatrix} u^d_{T,2} \\ y^d_{T,2} \end{pmatrix} & \begin{pmatrix} u^d_{T,3} \\ y^d_{T,3} \end{pmatrix} & \cdots \end{bmatrix}}_{\text{non-parametric model from raw data}}$$

all trajectories constructible from finitely many previous trajectories

- can also use other ***matrix data structures***: (mosaic) Hankel, Page, . . .
- ***novelty (?)***: motion primitives, (E)DMD, dictionary learning, subspace system id, . . . all implicitly rely on this equivalence $\rightarrow$ c.f. “fundamental”
- ***standing on the shoulders of giants***: classic Willems’ result was only “if” & required further assumptions: Hankel, persistency of excitation, controllability

A note on persistency of excitation

Jan C. Willems[a], Paolo Rapisarda[b], Ivan Markovsky[a,*], Bart L.M. De Moor[a]

[a]ESAT, SCD/SISTA, K.U. Leuven, Kasteelpark Arenberg 10, B 3001 Leuven, Heverlee, Belgium
[b]Department of Mathematics, University of Maastricht, 6200 MD Maastricht, The Netherlands

Received 3 June 2004; accepted 7 September 2004
Available online 30 November 2004

Control from matrix time series data

A note on persistency of excitation

Jan C. Willems[a], Paolo Rapisarda[b], Ivan Markovsky[a,*], Bart L.M. De Moor[a]

[a]*ESAT, SCD/SISTA, K.U. Leuven, Kasteelpark Arenberg 10, B 3001 Leuven, Heverlee, Belgium*
[b]*Department of Mathematics, University of Maastricht, 6200 MD Maastricht, The Netherlands*

Received 3 June 2004; accepted 7 September 2004
Available online 30 November 2004

We are all writing merely the dramatic corollaries …

implicit & stochastic

$\rightarrow$ Markovsky & ourselves

explicit & deterministic

$\rightarrow$ Groningen: Persis, Camlibel, …

$\rightarrow$ ***lots of recent momentum*** ($\sim$ 1 ArXiv / week) with contributions by Scherer, Allgöwer, Matni, Pappas, Fischer, Pasqualetti, Jones, Goulart, Mesbahi, …

$\rightarrow$ more classic ***subspace predictive control*** (De Moor) literature

Prediction & estimation [Markovsky & Rapisarda '08]

Problem: predict future output $y \in \mathbb{R}^{p \cdot T_{\text{future}}}$ based on

- initial trajectory $\text{col}(u_{\text{ini}}, y_{\text{ini}}) \in \mathbb{R}^{(m+p) \cdot T_{\text{ini}}}$ $\rightarrow$ to estimate initial x_{ini}
- input signal $u \in \mathbb{R}^{m \cdot T_{\text{future}}}$ $\rightarrow$ to predict forward
- past data $\text{col}(u^{\text{d}}, y^{\text{d}}) \in \mathscr{B}_{T_{\text{data}}}$ $\rightarrow$ to form trajectory matrix

Solution: given u & $\text{col}(u_{\text{ini}}, y_{\text{ini}}) \rightarrow$ compute $\boldsymbol{g}$ & y from

$$\left[\begin{array}{cccc}
u^{\text{d}}_{1,1} & u^{\text{d}}_{2,1} & u^{\text{d}}_{3,1} & \cdots \\
\vdots & \vdots & \vdots & \vdots \\
u^{\text{d}}_{1,T_{\text{ini}}} & u^{\text{d}}_{2,T_{\text{ini}}} & u^{\text{d}}_{3,T_{\text{ini}}} & \cdots \\
\hline
y^{\text{d}}_{1,1} & y^{\text{d}}_{2,1} & y^{\text{d}}_{3,1} & \cdots \\
\vdots & \vdots & \vdots & \vdots \\
y^{\text{d}}_{1,T_{\text{ini}}} & y^{\text{d}}_{2,T_{\text{ini}}} & y^{\text{d}}_{3,T_{\text{ini}}} & \cdots \\
\hline
u^{\text{d}}_{1,T_{\text{ini}}+1} & u^{\text{d}}_{2,T_{\text{ini}}+1} & u^{\text{d}}_{3,T_{\text{ini}}+1} & \cdots \\
\vdots & \vdots & \vdots & \vdots \\
u^{\text{d}}_{1,T_{\text{ini}}+T_{\text{future}}} & u^{\text{d}}_{2,T_{\text{ini}}+T_{\text{future}}} & u^{\text{d}}_{3,T_{\text{ini}}+T_{\text{future}}} & \cdots \\
\hline
y^{\text{d}}_{1,T_{\text{ini}}+1} & y^{\text{d}}_{2,T_{\text{ini}}+1} & y^{\text{d}}_{3,T_{\text{ini}}+1} & \cdots \\
\vdots & \vdots & \vdots & \vdots \\
y^{\text{d}}_{1,T_{\text{ini}}+T_{\text{future}}} & y^{\text{d}}_{2,T_{\text{ini}}+T_{\text{future}}} & y^{\text{d}}_{3,T_{\text{ini}}+T_{\text{future}}} & \cdots
\end{array}\right] \boldsymbol{g} = \mathscr{H}_{T_{\text{ini}}+T_{\text{future}}}\begin{pmatrix} u^{\text{d}} \\ y^{\text{d}} \end{pmatrix} \boldsymbol{g} = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}$$

$\Rightarrow$ observability condition: if $T_{\text{ini}} \geq$ lag of system, then y is ***unique***

Output **M**odel **P**redictive **C**ontrol

The canonical receding-horizon ***MPC optimization problem***:

$$
\begin{aligned}
\underset{u,x,y}{\text{minimize}} \quad & \sum_{k=1}^{T_{\text{future}}} \|y_k - r_k\|_Q^2 + \|u_k\|_R^2 \\
\text{subject to} \quad & x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& y_k = Cx_k + Du_k, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{-T_{\text{ini}} - 1, \ldots, 0\}, \\
& y_k = Cx_k + Du_k, \quad \forall k \in \{-T_{\text{ini}} - 1, \ldots, 0\}, \\
& u_k \in \mathcal{U}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& y_k \in \mathcal{Y}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}
\end{aligned}
$$

quadratic cost with $R \succ 0, Q \succeq 0$ & ref. r

model for **prediction** over $k \in [1, T_{\text{future}}]$

model for **estimation** (many variations)

hard operational or safety **constraints**

For a deterministic LTI plant and an exact model of the plant,
MPC is the ***gold standard of control***: safe, optimal, tracking, ...

Data-Enabled Predictive Control

DeePC uses trajectory matrix for receding-horizon prediction/estimation:

$$
\begin{aligned}
\underset{g,u,y}{\text{minimize}} \quad & \sum_{k=1}^{T_{\text{future}}} \|y_k - r_k\|_Q^2 + \|u_k\|_R^2 \\
\text{subject to} \quad & \mathscr{H}\begin{pmatrix} u^{\mathsf{d}} \\ y^{\mathsf{d}} \end{pmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}, \\
& u_k \in \mathcal{U}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& y_k \in \mathcal{Y}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}
\end{aligned}
$$

quadratic cost with $R \succ 0, Q \succeq 0$ & ref. r

non-parametric model for **prediction** and **estimation**

hard operational or safety **constraints**

- past $T_{\text{ini}} \geq$ lag samples $(u_{\text{ini}}, y_{\text{ini}})$ for x_{ini} estimation — updated **online**
- trajectory matrix $\mathscr{H}_{T_{\text{ini}}+T_{\text{future}}}\begin{pmatrix} u^{\mathsf{d}} \\ y^{\mathsf{d}} \end{pmatrix}$ from past data — collected **offline** (could be adapted online)

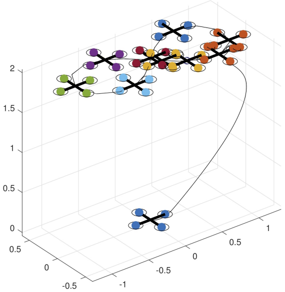
(drop $T_{\text{ini}} + T_{\text{future}}$ subscript from now on)

Consistency for LTI Systems

Theorem: Consider ***DeePC & MPC optimization problems***. If the rank condition holds (= rich data), then ***the feasible sets coincide***.

Corollary: closed-loop behaviors under DeePC and MPC coincide.

Aerial robotics case study :



Thus, most of ***MPC carries over to DeePC*** . . . in the ***nominal case***
c.f. stability certificate [Berberich et al. '19]

Beyond LTI: what about noise, corrupted data, & nonlinearities ?

. . . playing ***certainty-equivalence fails*** $\rightarrow$ need robustified approach

Noisy real-time measurements

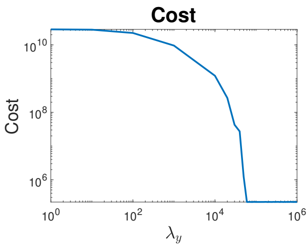
$$\begin{aligned}
\underset{g,u,y}{\text{minimize}} \quad & \sum_{k=1}^{T_{\text{future}}} \|y_k - r_k\|_Q^2 + \|u_k\|_R^2 + \lambda_y \|\sigma_{\text{ini}}\|_p \\
\text{subject to} \quad & \mathscr{H}\begin{pmatrix} u^{\mathsf{d}} \\ y^{\mathsf{d}} \end{pmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ \sigma_{\text{ini}} \\ 0 \\ 0 \end{bmatrix}, \\
& u_k \in \mathcal{U}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& y_k \in \mathcal{Y}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}
\end{aligned}$$

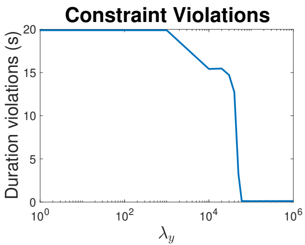
Solution: add **ℓ_p-slack** σ_{ini} to ensure feasibility

$\rightarrow$ moving-horizon least-square filter

$\rightarrow$ for $\lambda_y \gg 1$: constraint is slack only if infeasible

c.f. ***sensitivity analysis*** over randomized sims





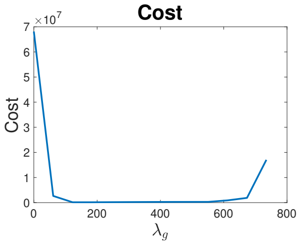
Trajectory matrix corrupted by noise

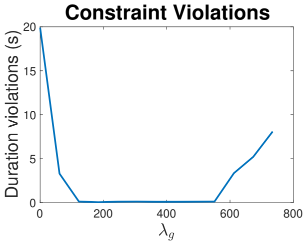
$$\begin{aligned}
\underset{g,u,y}{\text{minimize}} \quad & \sum_{k=1}^{T_{\text{future}}} \|y_k - r_k\|_Q^2 + \|u_k\|_R^2 + \lambda_g \|g\|_1 \\
\text{subject to} \quad & \mathscr{H}\begin{pmatrix} u^{\mathsf{d}} \\ y^{\mathsf{d}} \end{pmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}, \\
& u_k \in \mathcal{U}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}, \\
& y_k \in \mathcal{Y}, \quad \forall k \in \{1, \ldots, T_{\text{future}}\}
\end{aligned}$$

Solution : add a ℓ_1-**penalty** on g

intuition: ℓ_1 sparsely selects {trajectory matrix columns} = {motion primitives} $\sim$ low-order basis

c.f. ***sensitivity analysis*** over randomized sims



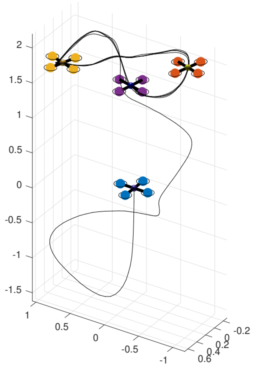


Towards nonlinear systems

Idea : lift nonlinear system to large/∞-dimensional bi-/linear system
$\rightarrow$ Carleman, Volterra, Fliess, Koopman, Sturm-Liouville methods
$\rightarrow$ nonlinear dynamics can be approximated by LTI on finite horizon

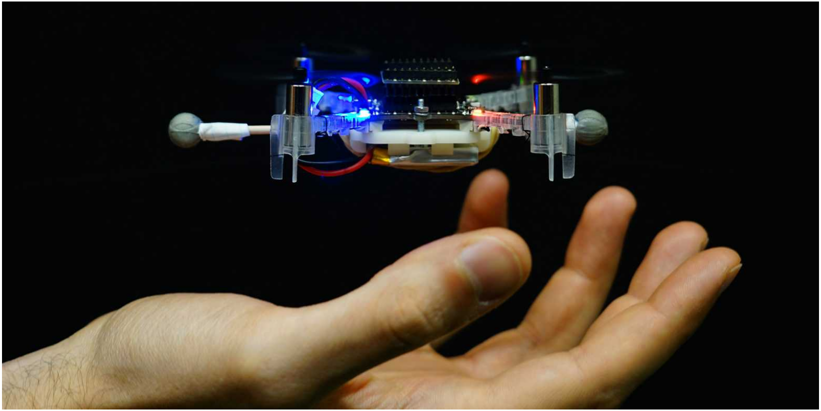
regularization singles out relevant features / basis functions in data

case study :
DeePC
+ σ_{ini} slack
+ $\|g\|_1$ regularizer
+ more columns in $\mathscr{H}\begin{pmatrix} u^{\mathrm{d}} \\ y^{\mathrm{d}} \end{pmatrix}$

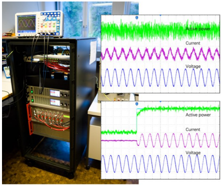


fluke
or
solid ?

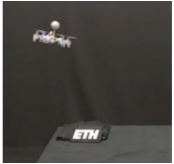
Experimental snippet



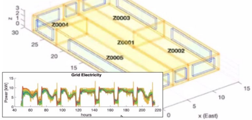
Consistent observations across case studies — more than a fluke



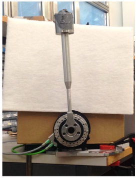
grid-connected converter

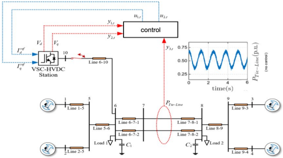


quad coptor fig-8 tracking

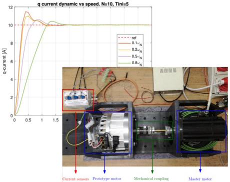


energy hub & building automation


pendulum swing up



power system oscillation damping (see later)



synchronous motor drive


robotic excavator

let’s try to put some theory behind all of this …

Distributional robust formulation [Coulson et al. '19]

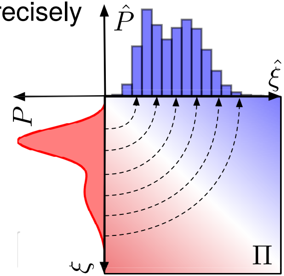
- ***problem abstraction***: $\min_{x\in\mathcal{X}}\ c\big(\widehat{\xi}, x\big) \ = \ \min_{x\in\mathcal{X}}\ \mathbb{E}_{\widehat{\mathbb{P}}}\big[c\left(\xi, x\right)\big]$

  where $\widehat{\xi}$ denotes *measured* data with *empirical distribution* $\widehat{\mathbb{P}} = \delta_{\widehat{\xi}}$

$\Rightarrow$ ***poor out-of-sample performance*** of above sample-average solution $x^\star$ for real problem: $\mathbb{E}_{\mathbb{P}}\big[c\left(\xi, x^\star\right)\big]$ where $\mathbb{P}$ is the *unknown distribution* of ξ

- ***distributionally robust*** formulation $\longrightarrow$ "$\min_{x\in\mathcal{X}} \max\ \mathbb{E}\big[c\left(\xi, x\right)\big]$"
  where $\max$ accounts for all stochastic processes (linear or nonlinear) that could have generated the data …more precisely

$$\inf_{x\in\mathcal{X}}\ \sup_{Q\in\mathbb{B}_\epsilon(\widehat{P})}\ \mathbb{E}_Q\big[c\left(\xi, x\right)\big]$$



where $\mathbb{B}_\epsilon(\widehat{P})$ is an ***ϵ-Wasserstein ball*** centered at empirical sample distribution $\widehat{P}$:

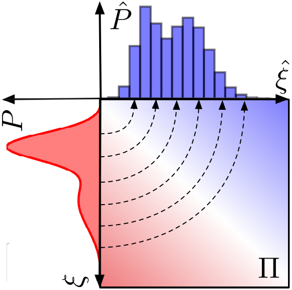
$$\mathbb{B}_\epsilon(\widehat{P}) = \left\{P \ : \ \inf_{\Pi} \int \left\|\xi - \hat{\xi}\right\|_p d\Pi \ \leq \ \epsilon\right\}$$

- ***distributionally robust*** formulation:

$$\inf_{x\in\mathcal{X}} \ \sup_{Q\in\mathbb{B}_\epsilon(\widehat{P})} \ \mathbb{E}_Q\left[c\left(\xi, x\right)\right]$$

where $\mathbb{B}_\epsilon(\widehat{P})$ is an ϵ-***Wasserstein ball*** centered at empirical sample distribution $\widehat{P}$:

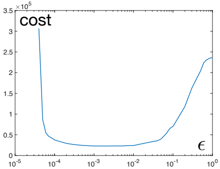
$$\mathbb{B}_\epsilon(\widehat{P}) = \left\{ P \ : \ \inf_{\Pi} \int \left\| \xi - \hat{\xi} \right\|_p d\Pi \ \leq \ \epsilon \right\}$$



Theorem : Under minor technical conditions:

$$\inf_{x\in\mathcal{X}} \ \sup_{Q\in\mathbb{B}_\epsilon(\widehat{P})} \ \mathbb{E}_Q\left[c\left(\xi, x\right)\right] \ \equiv \ \min_{x\in\mathcal{X}} \ c\left(\widehat{\xi}, x\right) \ + \ \epsilon \, \text{Lip}(c) \cdot \|x\|_p^\star$$

Cor : ℓ_∞-robustness in trajectory space
$\iff$ ℓ_1-regularization of DeePC

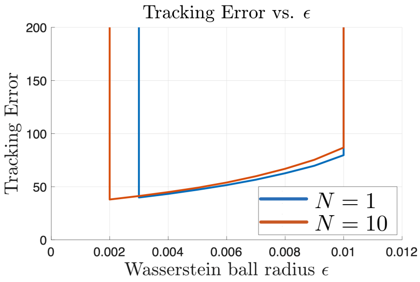


Proof builds on methods by Shafieezadeh, Esfahani, & Kuhn : problem tractable after marginalization, for discrete worst case, & with many convex conjugates.

Further ingredients

[Coulson et al. '19], [Alpago et al. '20]

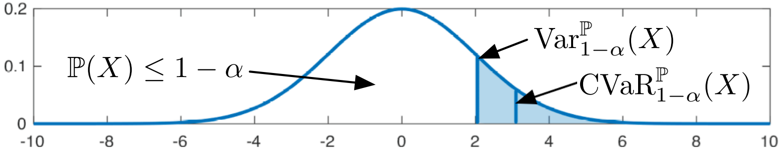
- ***average data matrix*** $\frac{1}{N}\sum_{i=1}^{N}\mathscr{H}_i(y^{\mathsf{d}})$ from multiple i.i.d. experiments
- ***measure concentration***: Wasserstein ball $\mathbb{B}_\epsilon(\widehat{P})$ includes true distribution $\mathbb{P}$ with high confidence if $\epsilon \sim 1/N^{1/\dim(\xi)}$



- recursive ***Kalman filtering*** based on explicit solution $g^\star$ as hidden state

- ***distributionally robust probabilistic constraints***

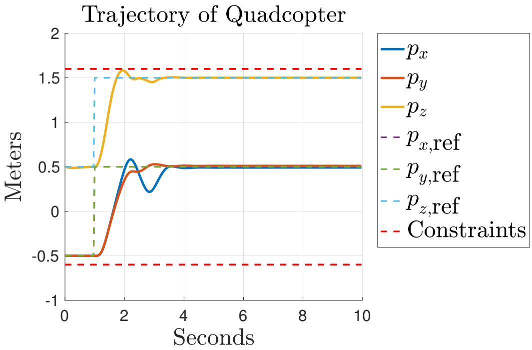
$\sup_{Q\in\mathbb{B}_\epsilon(\widehat{P})} \text{CVaR}_{1-\alpha}^{\mathbb{Q}} \quad \Leftrightarrow \quad$ averaging + regularization + tightening



All together in action

control objective
+ regularization
+ matrix predictor
+ averaging
+ CVaR constraints
+ σ_{ini} estimation slack

$\rightarrow$ DeePC works much better than it should !



main catch (no-free-lunch) : optimization problems become large
$\rightarrow$ models are compressed, de-noised, & tidied-up representations

surprise: **DeePC consistently beats models** across case studies !

$\rightarrow$ more to be understood

how does DeePC relate to sequential SysID + control ?

Recall problem abstraction

indirect* ID-based *data-driven control

minimize control cost (u, y)

subject to (u, y) satisfy parametric model

where model $\in$ argmin id cost (u^d, y^d)

subject to model $\in$ LTI(n, ℓ) class

ID

ID projects data on the set of LTI models

- with parameters (n, ℓ)
- removes noise & thus lowers variance error
- suffers bias error if plant is not LTI(n, ℓ)

direct* regularized *data-driven control

minimize control cost (u, y) + $\lambda\cdot$ regularizer

subject to (u, y) consistent with (u^d, y^d) data

- ***regularization robustifies***
  $\rightarrow$ choosing λ makes it work
- ***no projection*** on LTI(n, ℓ)
  $\rightarrow$ no de-noising & no bias

hypothesis 1: ID wins in stochastic (variance) & DeePC in nonlinear (bias) case

hypothesis 2: regularizer is somehow (???) connected to identification step

hypothesis 1 :

ID wins in stochastic (variance) &
DeePC in nonlinear (bias) case

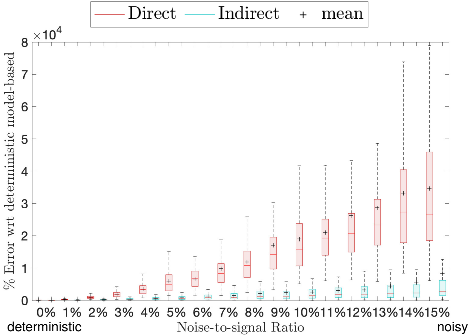
Comparison: direct vs. indirect control

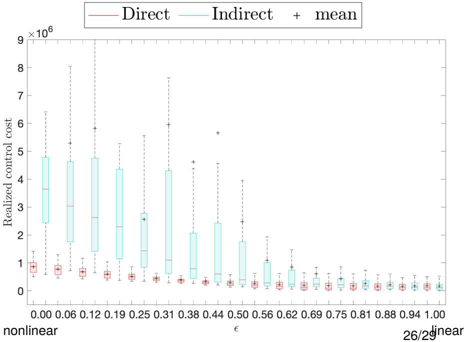
stochastic LTI case $\rightarrow$ indirect ID wins

- LQR control of 5th order LTI system
- Gaussian noise with varying noise to signal ratio (100 rollouts each case)
- ℓ_1-regularized DeePC, SysID via N4SID, & judicious hyper-parameters

nonlinear case $\rightarrow$ direct DeePC wins

- Lotka-Volterra + control: $x^+ = f(x, u)$
- interpolated system $x^+ = \epsilon \cdot f_{\text{linearized}}(x, u) + (1 - \epsilon) \cdot f(x, u)$
- same ID & DeePC as on the left & 100 initial x_0 rollouts for each ϵ





hypothesis 2:

regularizer is somehow (???) connected to identification step

Regularization = relaxation of bi-level ID

$$\begin{array}{ll} \text{minimize}_{u,y} & \text{control cost}(u,y) \\ \text{subject to} & (u,y) \text{ satisfy (parametric) model} \\ \text{where} & \text{model} \in \text{argmin id cost}(u^d,y^d) \\ & \text{subject to model} \in \text{LTI}(n,\ell) \end{array}$$

$$\begin{array}{ll} \text{minimize}_{u,y} & \text{control cost}(u,y) \\ \text{subject to} & (u,y) \text{ satisfy (parametric) model} \\ \text{where} & \text{model} \in \text{argmin id cost}(u^d,y^d) \\ & \text{subject to model} \in \text{LTI}(n,\ell) \end{array}$$

↓ drop non-convex LTI complexity specifications ↓

$$\begin{array}{ll} \text{minimize}_{u,y} & \text{control cost}(u,y) \\ \text{subject to} & (u,y) \text{ satisfy (parametric) model} \\ \text{where} & \text{model} \in \text{argmin id cost}(u^d,y^d) \end{array}$$

$$\begin{array}{ll} \text{minimize}_{u,y} & \text{control cost}(u,y) \\ \text{subject to} & (u,y) \text{ satisfy (parametric) model} \\ \text{where} & \text{model} \in \text{argmin id cost}(u^d,y^d) \\ & \text{subject to model} \in \text{LTI}(n,\ell) \end{array}$$

Example I: *parametric ARX predictor*

$$y = K \cdot \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \end{bmatrix}$$

where K is low-rank + least-square fit

⇒ ***ID-based*** regularizer $\left\| \ker \mathscr{H}\begin{pmatrix} u^{\text{d}} \\ y^{\text{d}} \end{pmatrix} g \right\|$

Example II: *non-parametric predictor*

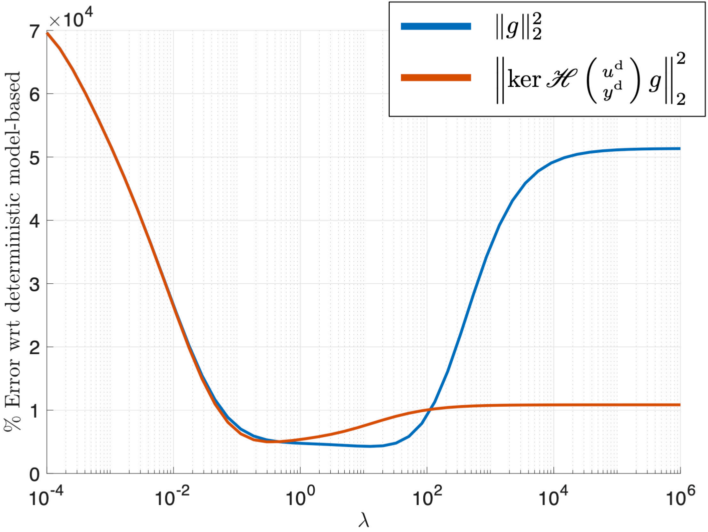
$$\begin{bmatrix} u \\ y \end{bmatrix} = \mathscr{H}\begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} g$$

where $\begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} \in \text{argmin} \left\| \begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} - \begin{pmatrix} u^{\text{d}} \\ y^{\text{d}} \end{pmatrix} \right\|$

subject to $\text{rank}\, \mathscr{H}\begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} = mL + n$

⇒ ***low-rank*** promoting regularizer $\|g\|_1$

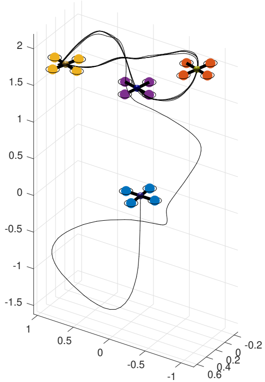
Performance of identification-induced regularizer on stochastic LTI system



Summary & conclusions

main take-aways

- matrix time series serves as predictive model
- consistent for deterministic LTI systems
- distributional robustness via regularizations
- principled derivation from ID & comparison



future work

$\rightarrow$ tighter certificates for nonlinear systems

$\rightarrow$ explicit policies & direct adaptive control

$\rightarrow$ online optimization & real-time iteration

Why have these powerful ideas not been mixed long before ?

Willems '07: *"[MPC] has perhaps too little system theory and too much brute force computation in it."*

The other side often proclaims *"behavioral systems theory is beautiful but did not prove utterly useful."*

Thanks !

Florian Dörfler

mail: dorfler@ethz.ch

[link] to homepage

[link] to related publications