On learning controllers from data for polynomial systems

Claudio De Persis

Institute of Engineering and Technology
Faculty of Science and Engineering

university of groningen

joint work with Andrea Bisoffi (RUG & Politecnico Milano), Meichen Guo (RUG) and Pietro Tesi (Università di Firenze, Italy)

Polynomial Optimization and Applications in Control and Energy
CWI, Amsterdam, 18-19 November 2022
What is control?

Given a dynamical system modelled by non-homogenous ODEs

\[ \dot{x} = f_\star(x) + g_\star(x)u \]

where \( x \in \mathbb{R}^n \) (state) and \( u \in \mathbb{R}^m \) (control), we want to design a feedback control \( u = K(x) \), such that the solution of the closed-loop system

\[ \dot{x} = f_\star(x) + g_\star(x)K(x) \]

has a desired qualitative behaviour, i.e., \( x(t) \) asymptotically converges to an equilibrium of interest (the origin \( x = 0 \)).

Typically, to design \( K(x) \) the designer requires the knowledge of \( f_\star, g_\star \).

What if \( f_\star, g_\star \) are not precisely known?
Dynamical control systems

We consider systems of the form

\[ \dot{x} = f_*(x) + g_*(x)u + d \]

- $x \in \mathbb{R}^n$ (state) and $u \in \mathbb{R}^m$ (control)
- $d \in \mathbb{R}^n$ is an unmeasured perturbation
- $f_* : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is an unknown drift vector field
- $g_* : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ is an unknown (matrix of) input vector field(s)
System

\[ \dot{x} = f_*(x) + g_*(x)u + d \]

includes notable classes of systems, such as linear systems

\[ \dot{x} = A_*x + B_*u + d \]

where $A_*$, $B_*$ are (unknown) matrices.


Dealing with the control design when $f_*$, $g_*$ are generic nonlinear vector fields is a very challenging problem
A first approach is to consider a first-order Taylor’s expansion of

\[ \dot{x} = f_\star(x) + g_\star(x)u \]

around a point of interest and obtain

\[ \dot{x} = A_\star x + B_\star u + d(x, u) \]

where \( d(x, u) \) is the remainder.

1. Collect data from the nonlinear system
2. Regard the remainder as a perturbation
3. Design from data a robust controller for the linearized system
4. Apply the controller to the nonlinear system (local stability)

Considering higher-order approximations of the nonlinear system might lead to improved performance (larger DoA) \( \Rightarrow \) This requires to deal with polynomial systems

**Prior 1** In this talk, we will consider \( f_\star, g_\star \) polynomials
Why polynomial control systems?

- General nonlinear control systems can be approximated by polynomial ones via Taylor’s series expansion.

- They provide a way to connect data-driven control design with approximation theories that make use of polynomial approximations, as in e.g. polynomial kernel-based methods.

- The results can be extended to other classes of nonlinear systems, which include e.g., trigonometric functions.

- Many benchmark control systems in adaptive control are actually polynomial systems.
A vector $Z : \mathbb{R}^n \to \mathbb{R}^N$ and a matrix $W : \mathbb{R}^n \to \mathbb{R}^{M\times m}$ of monomials are known such that $f_\star(x) = A_\star Z(x)$ and $g_\star(x) = B_\star W(x)$.

The system can be written as

$$\dot{x} = A_\star Z(x) + B_\star W(x)u + d$$

where $A_\star, B_\star$ are unknown.

Technical conditions Monomials in $Z(x)$ include $x$ and have degree $\geq 1$.
A vector $Z : \mathbb{R}^n \to \mathbb{R}^N$ and a matrix $W : \mathbb{R}^n \to \mathbb{R}^{M \times m}$ of monomials are known such that $f_\star(x) = A_\star Z(x)$ and $g_\star(x) = B_\star W(x)$.

Knowledge of $Z(x)$, $W(x)$ might come from the knowledge of the physics of the system.

A “dictionary” of functions might be available from some estimation technique.

If the assumption is not satisfied (and $W(x) = 1$), then we regard the discrepancy

$$f_\star(x) - A_\star Z(x)$$

as a neglected nonlinearity $d(x)$ and write the system as

$$x^+ = A_\star Z(x) + B_\star u + d(x)$$
Data collection

**Experiment** We run an experiment on the true system

\[ \dot{x} = A_x Z(x) + B_x W(x) u + d \]

during which

- An open-loop control \( u \) over a finite time interval \( \mathcal{I} \) over which a solution \( x(t) \) exists is applied
- Perturbation \( d \) affects the dynamics
- \( T \) triples of samples

\[
\mathcal{D} := \{ \dot{x}(t_k), x(t_k), u(t_k) \}_{k=0}^{T-1}
\]

at sampling times \( t_k \in \mathcal{I} \) are collected (possibly during multiple experiments)

- Vectors \( Z(x(t_k)) \) and \( W(x(t_k)) u(t_k) \), \( k = 0, 1, \ldots, T - 1 \) are computed

At each sampling time \( t_k \)

\[ \dot{x}(t_k) = A_x Z(x(t_k)) + B_x W(x(t_k)) u(t_k) + d(t_k) \quad k = 0, 1, \ldots, T - 1 \]
A control problem

Problem Based on the dataset $\mathbb{D}$ design a state feedback controller

$$u = K(x), \quad K : \mathbb{R}^n \to \mathbb{R}^m$$

that makes the origin a globally asymptotically stable equilibrium for

$$\dot{x} = A_x Z(x) + B_x W(x) K(x)$$

- An equilibrium is **stable** if solutions that start close to it remain close (in an $\varepsilon - \delta$ sense)

- An equilibrium is **globally asymptotically stable** if it is stable and any solutions eventually converge to it

Example $\dot{x} = x^2 + u$ is globally asymptotically stabilized by

$$K(x) = -x^2 - x$$
Consider the dataset
\[ \mathcal{D} = \{ \dot{x}(t_k), x(t_k), u(t_k) \}_{k=0}^{T}, \quad \dot{x}(t_k) = A_x Z(x(t_k)) + B_x W(x(t_k)) u(t_k) + d(t_k), \quad k = 0, \ldots, T - 1 \]

and note that it satisfies the matrix identity
\[
\begin{bmatrix}
\dot{x}(t_0) & \dot{x}(t_1) & \ldots & \dot{x}(t_{T-1})
\end{bmatrix} = A_x \underbrace{\begin{bmatrix}
Z(x(t_0)) & Z(x(t_1)) & \ldots & Z(x(t_{T-1}))
\end{bmatrix}}_{X_1} \\
+ \quad B_x \underbrace{\begin{bmatrix}
W(x(t_0)) u(t_0) & W(x(t_1)) u(t_1) & \ldots & W(x(t_{T-1})) u(t_{T-1})
\end{bmatrix}}_{U_0} \\
+ \underbrace{\begin{bmatrix}
d(t_0) & d(t_1) & \ldots & d(t_{T-1})
\end{bmatrix}}_{D_0}
\]

\[ X_1 = A_x Z_0 + B_x U_0 + D_0 \]
Disturbance model

\[ \mathcal{D} := \{ D \in \mathbb{R}^{n \times T} : DD^\top \preceq R_D R_D^\top \text{ for known } R_D \} \]

We will consider:

**Assumpt. (Noise model correctness)** \( D_0 \in \mathcal{D} \)

**Example** \( |d(t_k)| \leq \delta, \ k = 0, 1, \ldots, T - 1 \implies R_D = \delta \sqrt{T} I_n \)

**Consistency set**

\[ \mathcal{C} := \{(A, B) : X_1 = AZ_0 + B \overline{U}_0 + D \text{ for some } D \in \mathcal{D}\} \]

**Assumpt. (Quality of data)** \( \overline{W}_0 := \begin{bmatrix} \overline{U}_0 \\ Z_0 \end{bmatrix} \) has full row rank.

**Note** Implies \( \mathcal{C} \) bounded.
Ellipsoidal description of $C$

Under the quality of data assumption, the set $C$ takes a special form

**Lemma** It holds that

$$C = \left\{ \Delta \in \mathbb{R}^{M+N\times n} : \Delta = -\Theta^{-1} V + \Theta^{-1/2} SL^{1/2}, \|S\| \leq 1 \right\}$$

where:

- $\Delta := \begin{bmatrix} B & A \end{bmatrix}^T$
- $\Theta := \overline{W}_0 \overline{W}_0^T$
- $V := -\overline{W}_0 X_1^T$
- $L := V^T \Theta^{-1} V - \Xi$
- $\Xi := X_1 X_1^T - R_D R_D^T$

The centre is $-\Theta^{-1} V$ and the size depends on $L$.

Bisoffi, De Persis, Tesi. “Data-driven control via Petersen’s lemma”. Automatica, 2022
Recap so far

**Prior 1** Nonlinear polynomial input-affine systems

\[ \dot{x} = f_*(x) + g_*(x)u + d \]

**Prior 2** Known dictionaries of functions \(Z(x), W(x)\)

\[ f_*(x) = A_* Z(x) \quad g_*(x) = B_* W(x) \]

**Data set** \(T\)-long data set \(\{\dot{x}(t_k), x(t_k), u(t_k)\}_{k=0}^{T-1}\) satisfying

\[ \dot{x}(t_k) = A_* Z(x(t_k)) + B_* W(x(t_k))u(t_k) + d(t_k) \quad k = 0, 1, \ldots, T - 1 \]

or

\[ X_1 = A_* Z_0 + B_* U_0 + D_0 \]

**Prior 3** Perturbation model & model correctness

\[ D_0 \in \mathcal{D}_e := \{ D \in \mathbb{R}^{n \times T} : DD^\top \preceq R_D R_D^\top \text{ for known } R_D \} \]

where \(D_0 = [d(t_0) \quad d(t_1) \quad \ldots \quad d(t_{T-1})] \).
Control synthesis
Towards data-driven control design for nonlinear systems

Data enable the replacement of the unknown system

\[ \dot{x} = A_\star Z(x) + B_\star W(x)u \]

with the uncertain system

\[ \dot{x} = AZ(x) + BW(x)u \quad \text{with} \quad (A, B) \in C \]

How to design a controller for this data-dependent but uncertain representation? We resort to

▷ Lyapunov stability theory
▷ A matrix elimination lemma to deal with the uncertainty
Lyapunov stability theory

We look for

- a control function $K(x)$
- a $C^1$ function $V : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ such that $V(0) = 0$, $V(x) > 0$ for all $x \neq 0$,
  it is radially unbounded, it decreases along the solutions of the closed-loop system, i.e. for some $\lambda > 0$

\[
\frac{\partial V}{\partial x} (AZ(x) + BW(x)K(x)) \leq -\lambda V(x)
\]

\[
\dot{V}(x(t)) \leq -\lambda V(x(t)) \Rightarrow V(x(t)) \leq e^{-\lambda t} V(x(0))
\]
A matrix elimination lemma – Petersen’s lemma

Consider matrices \( G, M \neq 0, N \neq 0 \). Then

\[
G + MSN + N^T S^T M^T \preceq 0 \quad \forall S : S^T S \preceq I_n
\]

\[\uparrow\]

\[
\exists \varepsilon > 0 : G + \varepsilon MM^T + \varepsilon^{-1} N^T N \preceq 0
\]

**Proof.** \( \uparrow \) follows from a completion of the squares argument.

\( \downarrow \) Difficult – see*

---

* Petersen, Hollot. A Riccati equation approach to the stabilization of uncertain linear systems, Automatica, 1986
Design choices

▷ Consider the Lyapunov function

\[ V(x) = Z(x)^\top P^{-1} Z(x) \quad \text{with } P > 0 \text{ a decision variable} \]

▷ Consider the control function

\[ K(x) = Y(x) P^{-1} Z(x) \quad \text{with } Y(x) \text{ a decision variable} \]

to obtain the closed-loop system

\[
\dot{x} = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix} P^{-1} Z(x) \quad \text{with } (A, B) \in C
\]

▷ For a given \( \lambda > 0 \), look for \( P > 0 \) and \( Y(x) \) such that

\[
\dot{V}(x) + \lambda V(x) = 2 Z(x)^\top P^{-1} \frac{\partial Z(x)}{\partial x} \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix} P^{-1} Z(x) + \lambda V(x) \leq 0
\quad \forall x \text{ and } \forall (A, B) \in C
\]
If we factor out the vectors $Z(x)^\top P^{-1}$ on the left-hand side and $P^{-1}Z(x)$ on the right-hand side of

$$\dot{V}(x) + \lambda V(x)$$

we obtain

$$\dot{V}(x) + \lambda V(x) = Z(x)^\top P^{-1} M(x) P^{-1} Z(x)$$

where

$$M(x) = \frac{\partial Z(x)}{\partial x} [B \quad A] \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix} + \begin{bmatrix} W(x) Y(x) \end{bmatrix}^\top [B \quad A]^\top \frac{\partial Z(x)}{\partial x}^\top + \lambda P$$

Look for $P \succ 0$ and $Y(x)$ such that

$$M(x) \preceq 0 \quad \forall x, \quad \forall (A, B) \in \mathcal{C}$$
Bearing in mind that \((A, B) \in C\) is equivalent to

\[
\begin{bmatrix} B & A \end{bmatrix}^\top = Z_c + \Theta^{-1/2} SL^{1/2} \quad S^\top S \preceq I
\]

where \(Z_c = -\Theta^{-1} V\) for short, we write

\[
\begin{align*}
M(x) &= \frac{\partial Z(x)}{\partial x} Z_c^\top \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix} + \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix}^\top Z_c \frac{\partial Z(x)}{\partial x}^\top + \lambda P \\
&\quad + \frac{\partial Z(x)}{\partial x} L^{1/2} S^\top \Theta^{-1/2} \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix} \\
&\quad + \begin{bmatrix} W(x) Y(x) \\ P \end{bmatrix}^\top \Theta^{-1/2} S L^{1/2} \frac{\partial Z(x)}{\partial x}^\top
\end{align*}
\]
Let us single out the different parts of $M(x)$

\[
M(x) = \begin{aligned}
&\underbrace{\frac{\partial Z(x)}{\partial x}}_{\mathcal{M}(x)} Z_c^\top \begin{bmatrix} W(x)Y(x) \\ P \end{bmatrix} + \begin{bmatrix} W(x)Y(x) \\ P \end{bmatrix}^\top Z_c \frac{\partial Z(x)}{\partial x} + \lambda P + \\
&\underbrace{\left[ W(x)Y(x) \right]^\top \Theta^{-1/2} \begin{bmatrix} S \\ \mathcal{N}(x) \end{bmatrix}}_{\mathcal{G}(x)} \underbrace{L^{1/2} \frac{\partial Z(x)}{\partial x}}_{\mathcal{N}(x)} + \underbrace{\ldots}_{\mathcal{N}(x)^\top S^\top M(x)^\top}
\end{aligned}
\]

By pointwise application of Petersen's lemma

\[
M(x) \preceq 0 \quad \forall S : S^\top S \preceq I_n
\]

\[
\uparrow
\]

\[
\exists \varepsilon(x) > 0 : \mathcal{G}(x) + \varepsilon(x)\mathcal{M}(x) \mathcal{M}(x)^\top + \varepsilon(x)^{-1} \mathcal{N}(x)^\top \mathcal{N}(x) \preceq 0
\]
We explicitly write the condition

\[ \mathcal{G}(x) + \varepsilon(x) \mathcal{M}(x) \mathcal{M}(x)^\top + \varepsilon(x)^{-1} \mathcal{N}(x)^\top \mathcal{N}(x) \preceq 0 \]

to obtain

\[
\begin{bmatrix}
W(x) Y(x) \\
\quad P
\end{bmatrix}^\top Z_c \frac{\partial Z(x)}{\partial x}^\top + (\ast)^\top + \lambda P + \varepsilon(x) \begin{bmatrix}
W(x) Y(x) \\
\quad P
\end{bmatrix}^\top \Theta^{-1} \begin{bmatrix}
W(x) Y(x) \\
\quad P
\end{bmatrix} + \varepsilon(x)^{-1} \frac{\partial Z(x)}{\partial x} L \frac{\partial Z(x)}{\partial x}^\top \leq 0
\]

By pointwise Schur complement, the latter is equivalent to

\[
\begin{bmatrix}
W(x) Y(x) \\
\quad P
\end{bmatrix}^\top Z_c \frac{\partial Z(x)}{\partial x}^\top + (\ast)^\top + \varepsilon(x)^{-1} \frac{\partial Z(x)}{\partial x} L \frac{\partial Z(x)}{\partial x}^\top + \lambda P \begin{bmatrix}
W(x) Y(x) \\
\quad P
\end{bmatrix}^\top - \varepsilon(x)^{-1} \Theta \leq 0
\]
or, setting $\mu(x) := \varepsilon^{-1}(x)$,

$$
- \begin{bmatrix}
W(x)Y(x) & Z_c \frac{\partial Z(x)}{\partial x} \\
\text{P} & \end{bmatrix}^T \\
\begin{bmatrix}
W(x)Y(x) \\
\text{P}
\end{bmatrix} + (\star)^T + \mu(x) \frac{\partial Z(x)}{\partial x} L \frac{\partial Z(x)}{\partial x}^T + \lambda \text{P}
$$

$$
\begin{bmatrix}
W(x)Y(x) \\
\text{P} \\
-\mu(x)\Theta
\end{bmatrix}^T \\
= : \mathcal{P}(x)
$$

If we relax the positive semi-definiteness requirement for the matrix above to the requirement that the matrix is an SOS polynomial matrix, we obtain a convex test to determine a controller and a Lyapunov function from data.
Main result

Assume

Prior 1-2 \[ \dot{x} = f_*(x) + g_*(x)u = A_*Z(x) + B_*W(x)u \] polynomial

Prior 3 \[ D_0 \in D := \{ D \in \mathbb{R}^{n \times T} : DD^\top \preceq R_D R_D^\top \text{ for known } R_D \} \]

If there exist \( P \succ 0 \), polynomial matrix \( Y(x) \) and scalar polynomial \( \mu(x) > 0 \) such that

\[
\mathcal{P}(x) := -\begin{bmatrix}
W(x)Y(x) \\ P
\end{bmatrix}^\top Z_c \frac{\partial Z(x)}{\partial x}^\top + (\ast)^\top + \mu(x) \frac{\partial Z(x)}{\partial x} L \frac{\partial Z(x)}{\partial x}^\top + \lambda P \begin{bmatrix}
W(x)Y(x) \\ P
\end{bmatrix}^\top
\]

is an SOS matrix, then

\[
u = Y(x)P^{-1}Z(x) \quad \text{globally asymptotically stabilizer}
\]

\[
V(x) = Z(x)^\top P^{-1}Z(x) \quad \text{Lyapunov function}
\]
End-to-end method: data $\rightarrow$ SOS $\rightarrow$ controller with certificate

The polynomial matrix has dimensions $r \times r$ with $r = M + 2N$ and degree $2\deg$ determined by the decision variables $P \in S^{N \times N}$, $Y(x) \in \Pi_{m,N}$, $\mu(x) \in \Pi$.

Checking whether or not it is an SOS matrix involves finding a matrix $\Lambda \succeq 0$ such that

$$P(x) = (\zeta(x) \otimes I_r)^\top \Lambda (\zeta(x) \otimes I_r)$$

where $\zeta(x)$ is the vector of all monomials of degree less than or equal to $\deg$. This check is performed by software tools.
An extension

Safe control Given a subset (safe set) of the state space defined by polynomial constraints, design a controller that maintains the state in the safe set for all $t$

Caveat

- If $Z(x), W(x)$ include too many unnecessary monomials, the uncertainty introduced by the noise can give raise to systems that are substantially different from the ground-truth one (because of dense $A, B$)

- Priors such as knowledge of the system’s physics to guide the choice of $Z(x), W(x)$ might lead to a dramatic reduction of the complexity.

- Inclusion of as many priors as possible and increased quality of data are essential factors to successfully carry out control synthesis.

- To cope with the size of SOS programs $\Rightarrow$ scalable alternatives to SOS synthesis.
Final remarks

- Data-driven control design method for polynomial systems expressible via “basis” functions
- Simple end-to-end criterion or “formula” (Data $\Rightarrow$ SOS $\Rightarrow$ controller)

Additional results

- In the presence of remainders $\Rightarrow$ local results with RoA estimation

Outlook

- SOS approaches have potentials, in spite of limitations (wise choice of basis functions, numerical complexity)
- Main question how to exploit recent advances in polynomial optimization to improve current data-driven design methods?

Thank you!

c.de.persis@rug.nl