Tutorial on the behavioral approach to data-driven system theory

Ivan Markovsky

Premise: familiarity with classical approach

why is a different approach needed?

how is the behavioral approach different?

what new does it bring?

Thesis: behavioral approach has added value

In the classical approach, a system is an input-output map

$$\text{input} \longrightarrow \boxed{\text{system}} \longrightarrow \text{output}$$

the input *causes* the output

the system is a *signal processor*

the system is defined by *equations*

Outline

Classical vs behavioral approaches

Data-driven interpolation and approximation

Convex relaxations and empirical validation

Outline

Classical vs behavioral approaches

Data-driven interpolation and approximation

Convex relaxations and empirical validation

Why is a different approach needed?

input/output maps assume zero *initial conditions*

modeling from first principles leads to *relations*

interconnection of systems is *variables sharing*

Why is a different approach needed?

input/output maps assume zero *initial conditions*

- without input, what is a signal processor processing?
- initial conditions can be added as an afterthought

modeling from first principles leads to *relations*

interconnection of systems is *variables sharing*

Why is a different approach needed?

input/output maps assume zero *initial conditions*

modeling from first principles leads to *relations*

- *e.g.*, ideal gas law: $PV = cMT$

  (P — pressure, V — volume, M — mass, T — temperature, c — constant)

interconnection of systems is *variables sharing*

Why is a different approach needed?

input/output maps assume zero *initial conditions*

modeling from first principles leads to *relations*

interconnection of systems is *variables sharing*

- mechanical systems: position and velocity
- electrical systems: potential and current
- hydraulic systems: pressure and flow

The behavioral approach was put forward by Jan C. Willems in the 1980's

3-part, 70-page, Automatica paper:

Part I. Finite dimensional linear time invariant systems
Part II. Exact modelling
Part III. Approximate modelling

From Time Series to Linear System—
Part I. Finite Dimensional Linear Time Invariant Systems*

JAN C. WILLEMS†

Dynamical systems are defined in terms of their behaviour, and input/output systems appear as particular representations. Finite dimensional linear time invariant systems are characterized by the fact that their behaviour is a linear shift invariant complete (equivalently closed) subspace of $(\mathbb{R}^q)^{\mathbb{Z}}$ *or* $(\mathbb{R}^q)^{\mathbb{Z}_+}$.

“Good definition should formalize sensible intuition” J.C. Willems

“I was not going to use the classical format where a definition is given first, followed by illustrative examples. I wanted this to go the other way around: show how examples lead to definitions.”

some of the examples he used:

- Newton’s second law
- Maxwell’s equations
- the first and second laws of thermodynamics

How is the behavioral approach *different* from the classical one?

dynamical system $\mathscr{B}$ is a set of signals w

$$
\begin{aligned}
w \in \mathscr{B} \quad &\leftrightarrow \quad \text{"} w \text{ is trajectory of } \mathscr{B}\text{"} \\
&\leftrightarrow \quad \text{"} \mathscr{B} \text{ is exact model for } w\text{"}
\end{aligned}
$$

no inputs and outputs, no causality, no equations

the system is detached from its *representations*

properties and problems are separated from methods

How is the behavioral approach *similar* to the classical one?

input/output partitioning $w = \Pi \begin{bmatrix} u \\ y \end{bmatrix}$ and representations can be derived from $\mathscr{B}$, *e.g.*,

$$\mathscr{B} = \left\{ w = \Pi \begin{bmatrix} u \\ y \end{bmatrix} \in (\mathbb{R}^q)^{\mathbb{N}} \mid \exists\, x \in (\mathbb{R}^n)^{\mathbb{N}},\ \begin{bmatrix} \sigma x \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \right\}$$

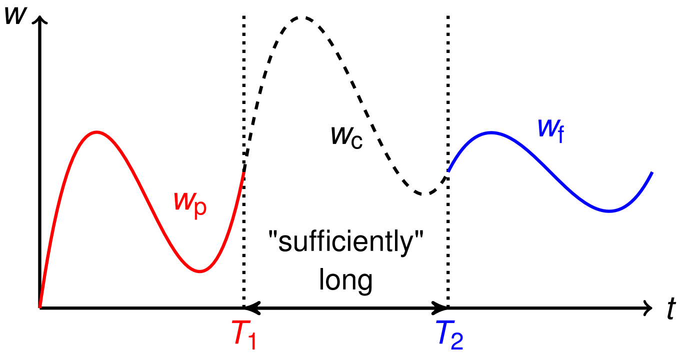
however

- given $\mathscr{B}$, an input/output partitioning is typically not unique
- also, properties and problems are defined in terms of $\mathscr{B}$
- equivalent representations define the same system

Example: what means that $\mathscr{B}$ is controllable?

controllability is the property of "patching"
any past trajectory with any future trajectory

$$w_{\mathrm{p}} \wedge w_{\mathrm{c}} \wedge w_{\mathrm{f}} \in \mathscr{B}$$



Compare with the classical definition: transfer from any initial to any terminal state

property of a state-space representation of $\mathscr{B}$

- is lack of controllability due to a “bad” choice of the state or due to an intrinsic issue with the system?
- minimal (controllable and observable) state-space representation can’t be assumed w.l.g.
- how to quantify the “distance” to uncontrollability?

does not apply to infinite dimensional system

Separating problems from solution methods

different representations $\rightsquigarrow$ different methods

- with different properties (efficiency, robustness, ...)
- their common feature is that they solve the same problem

clarifies links among methods

leads to new methods

Example: back to the controllability example

how to check controllability of an LTI system?

using state-space representation:

1. ensure minimality (in the behavioral sense)
2. perform rank test for the controllability matrix

using matrix fraction representation:

$$\mathscr{B} = \left\{ w = \Pi \begin{bmatrix} u \\ y \end{bmatrix} \in (\mathbb{R}^q)^{\mathbb{N}} \mid N(\sigma)u = D(\sigma)y \right\}$$

- facts: $\mathscr{B}$ is controllable $\iff$ N and D are co-prime
- $\rightsquigarrow$ rank test for the (generalized) Sylvester matrix

The behavioral approach is naturally suited for the "data-driven paradigm"

1940–1960	classical	SISO transfer function
1960–1980	modern	MIMO state-space
1980–2000	behavioral	the system as a set
2000–now	data-driven	using directly the data

Summary: behavioral approach

detach the system from its representations

- define properties and problems in terms of the behavior
- lead to new, more general, definitions and problems
- avoid inconsistencies of the classical approach

separate problem from solution methods

- different representations lead to different methods
- show links among different methods
- lead to new solutions

naturally suited for the “data-driven paradigm”

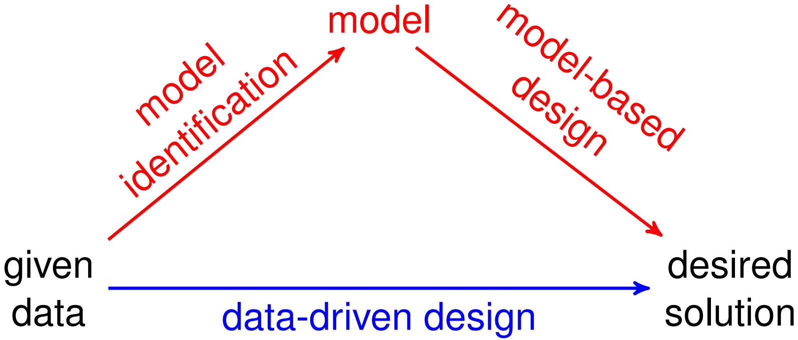
Outline

Classical vs behavioral approaches

Data-driven interpolation and approximation

Convex relaxations and empirical validation

The new “data-driven” paradigm obtains desired solution directly from given data



Data-driven does not mean model-free

data-driven problems do assume model

however, specific representation is not fixed

the methods we review are non-parametric

A dynamical system $\mathscr{B}$ is a set of signals

$\mathscr{B}$ is linear system $\quad :\Longleftrightarrow \quad$ $\mathscr{B}$ is subspace

$\mathscr{B}$ is time-invariant $\quad :\Longleftrightarrow \quad$ $\sigma \mathscr{B} = \mathscr{B}$

$(\sigma w)(t) := w(t+1)$ — shift operator

$\sigma \mathscr{B} := \{ \sigma w \mid w \in \mathscr{B} \}$

"good definition should formalize sensible intuition"

The set of linear time-invariant systems $\mathscr{L}$ has structure characterized by set of integers

the dimension of $\mathscr{B} \in \mathscr{L}$ is determined by

$\mathbf{m}(\mathscr{B})$ — number of inputs

$\mathbf{n}(\mathscr{B})$ — order ($=$ minimal state dimension)

$\boldsymbol{\ell}(\mathscr{B})$ — lag ($=$ observability index)

J.C. Willems, From time series to linear systems.
Part I, Finite dimensional linear time invariant systems, Automatica, 22(561–580), 1986

$\mathscr{B}_1$ less complex than $\mathscr{B}_2 \quad \iff \quad \mathscr{B}_1 \subset \mathscr{B}_2$

in the LTI case, complexity $\leftrightarrow$ dimension

complexity: (# inputs, order, lag)

$$\mathbf{c}(\mathscr{B}) := \big(\mathbf{m}(\mathscr{B}), \mathbf{n}(\mathscr{B}), \boldsymbol{\ell}(\mathscr{B})\big)$$

$\mathscr{L}_c$ — bounded complexity LTI model class

Data-driven representation (infinite horizon)

data: exact infinite trajectory w_{d} of $\mathscr{B} \in \mathscr{L}$

define $\quad \widehat{\mathscr{B}} := \mathrm{span}\{\, w_{\mathrm{d}}, \sigma w_{\mathrm{d}}, \sigma^2 w_{\mathrm{d}}, \dots \}$

identifiability condition: $\quad \mathscr{B} = \widehat{\mathscr{B}}$

Data-driven representation (finite horizon)

restriction of w and $\mathscr{B}$ to finite interval $[1, L]$

$$w|_L := \big(w(1), \dots, w(L)\big), \quad \mathscr{B}|_L := \{\, w|_L \mid w \in \mathscr{B} \,\}$$

for $w_\mathrm{d} = \big(w_\mathrm{d}(1), \dots, w_\mathrm{d}(T)\big)$ and $1 \leq L \leq T$

$$\mathscr{H}_L(w_\mathrm{d}) := \Big[(\sigma^0 w_\mathrm{d})|_L \quad (\sigma^1 w_\mathrm{d})|_L \quad \cdots \quad (\sigma^{T-L} w_\mathrm{d})|_L\Big]$$

define $\quad \widehat{\mathscr{B}}|_L := \operatorname{image} \mathscr{H}_L(w_\mathrm{d})$

Conditions for informativity of the data

$\mathscr{B}|_L = \text{image}\, \mathscr{H}_L(w_\text{d})$ if and only if

$$\text{rank}\, \mathscr{H}_L(w_\text{d}) = L\mathbf{m}(\mathscr{B}) + \mathbf{n}(\mathscr{B}) \qquad \text{(GPE)}$$

I. Markovsky and F. Dörfler, Identifiability in the Behavioral Setting, TAC, 2023

sufficient conditions (input design perspective):

1. $w_\text{d} = \begin{bmatrix} u_\text{d} \\ y_\text{d} \end{bmatrix}$
2. $\mathscr{B}$ controllable
3. $\mathscr{H}_{L+\mathbf{n}(\mathscr{B})}(u_\text{d})$ full row rank (PE)

J.C. Willems et al., A note on persistency of excitation
Systems & Control Letters, (54)325–329, 2005

PE — persistency of excitation, GPE — generalized PE

Generic data-driven problem: trajectory interpolation/approximation

given: "data" trajectory $w_\mathrm{d} \in \mathscr{B}|_T$

partially specified trajectory $w|_{I_\mathrm{given}}$

($w|_{I_\mathrm{given}}$ selects the elements of w, specified by I_given)

aim: minimize over $\widehat{w}$ $\quad \| w|_{I_\mathrm{given}} - \widehat{w}|_{I_\mathrm{given}} \|$

subject to $\widehat{w} \in \mathscr{B}|_L$

$$\widehat{w} = \mathscr{H}_L(w_\mathrm{d}) \big(\mathscr{H}_L(w_\mathrm{d})|_{I_\mathrm{given}} \big)^+ w|_{I_\mathrm{given}} \qquad \text{(SOL)}$$

Special cases

simulation

- given data: initial condition and input
- to-be-found: output (exact interpolation)

smoothing

- given data: noisy trajectory
- to-be-found: ℓ_2-optimal approximation

tracking control

- given data: to-be-tracked trajectory
- to-be-found: ℓ_2-optimal approximation

Generalizations

multiple data trajectories $w_\mathrm{d}^1, \dots, w_\mathrm{d}^N$

$$\mathscr{B} = \text{image}\begin{bmatrix} \mathscr{H}_L(w_\mathrm{d}^1) & \cdots & \mathscr{H}_L(w_\mathrm{d}^N) \end{bmatrix}$$

w_d not exact / noisy

maximum-likelihood estimation
$\rightsquigarrow$ Hankel structured low-rank approximation/completion
nuclear norm and ℓ_1-norm relaxations
$\rightsquigarrow$ nonparametric, convex optimization problems

nonlinear systems

results for special classes of nonlinear systems:
Volterra, Wiener-Hammerstein, bilinear, ...

Summary: data-driven signal processing

data-driven representation

leads to general, simple, practical methods

interpolation/approximation of trajectories

simulation, filtering and control are special cases
assumes only LTI dynamics; no hyper parameters

dealing with noise and nonlinearities

nonlinear optimization
convex relaxations

Outline

Classical vs behavioral approaches

Data-driven interpolation and approximation

Convex relaxations and empirical validation

The data w_{d} being exact vs inexact / "noisy"

w_{d} exact and satisfying (GPE)

- "system theory" problems
- image $\mathscr{H}_L(w_{\text{d}})$ is nonparametric finite-horizon model
- data-driven solution $=$ model-based solution

w_{d} inexact, due to noise and/or nonlinearities

- naive approach: apply the solution (SOL) for exact data
- rigorous: assume noise model $\rightsquigarrow$ ML estimation problem
- heuristics: convex relaxations of the ML estimator

The maximum-likelihood estimation problem in the errors-in-variables setup is nonconvex

errors-in-variables setup: $w_\mathrm{d} = \overline{w}_\mathrm{d} + \widetilde{w}_\mathrm{d}$

- $\overline{w}_\mathrm{d}$ — true data, $\overline{w}_\mathrm{d} \in \mathscr{B}|_T$, $\mathscr{B} \in \mathscr{L}_c^q$
- $\widetilde{w}_\mathrm{d}$ — zero mean, white, Gaussian measurement noise

ML problem: given w_d, c, and $w|_{I_\mathrm{given}}$

$$\begin{aligned} \underset{g}{\text{minimize}} \quad & \| w|_{I_\mathrm{given}} - \mathscr{H}_L(\widehat{w}_\mathrm{d}^*)|_{I_\mathrm{given}} g \| \\ \text{subject to} \quad & \widehat{w}_\mathrm{d}^* = \arg\min_{\widehat{w}_\mathrm{d}, \widehat{\mathscr{B}}} \quad \| w_\mathrm{d} - \widehat{w}_\mathrm{d} \| \\ & \qquad \text{subject to} \quad \widehat{w}_\mathrm{d} \in \widehat{\mathscr{B}}|_T \text{ and } \widehat{\mathscr{B}} \in \mathscr{L}_c^q \end{aligned}$$

The ML estimation problem is equivalent to Hankel structured low-rank approximation

$$\begin{aligned} \underset{g}{\text{minimize}} \quad & \| w|_{I_{\text{given}}} - \mathscr{H}_L(\widehat{w}_{\text{d}}^*)|_{I_{\text{given}}} g \| \\ \text{subject to} \quad & \widehat{w}_{\text{d}}^* = \arg\min_{\widehat{w}_{\text{d}}, \widehat{\mathscr{B}}} \quad \| w_{\text{d}} - \widehat{w}_{\text{d}} \| \\ & \qquad \text{subject to} \quad \widehat{w}_{\text{d}} \in \widehat{\mathscr{B}}|_T \text{ and } \widehat{\mathscr{B}} \in \mathscr{L}_c^q \end{aligned}$$

$$\Updownarrow$$

$$\begin{aligned} \underset{g}{\text{minimize}} \quad & \| w|_{I_{\text{given}}} - \mathscr{H}_L(\widehat{w}_{\text{d}}^*)|_{I_{\text{given}}} g \| \\ \text{subject to} \quad & \widehat{w}_{\text{d}}^* = \arg\min_{\widehat{w}_{\text{d}}} \quad \| w_{\text{d}} - \widehat{w}_{\text{d}} \| \\ & \qquad \text{subject to} \quad \operatorname{rank} \mathscr{H}_{\ell+1}(\widehat{w}_{\text{d}}) \leq (\ell+1)m + n \end{aligned}$$

Solution methods

local optimization

- choose a parametric representation of $\widehat{\mathscr{B}}(\theta)$
- optimize over $\widehat{w}$, $\widehat{w_{\mathrm{d}}}$, and θ
- depends on the initial guess

convex relaxation based on the nuclear norm

$$\begin{aligned} \text{minimize} \quad \text{over } \widehat{w}_{\mathrm{d}} \text{ and } \widehat{w} \quad & \|w|_{I_{\text{given}}} - \widehat{w}|_{I_{\text{given}}}\| + \|w_{\mathrm{d}} - \widehat{w}_{\mathrm{d}}\| \\ & + \gamma \cdot \left\| \begin{bmatrix} \mathscr{H}_{\Delta}(\widehat{w}_{\mathrm{d}}) & \mathscr{H}_{\Delta}(\widehat{w}) \end{bmatrix} \right\|_{*} \end{aligned}$$

convex relaxation based on ℓ_1-norm (LASSO)

$$\text{minimize} \quad \text{over } g \quad \|w|_{I_{\text{given}}} - \mathscr{H}_{L}(w_{\mathrm{d}})|_{I_{\text{given}}} g\| + \lambda \|g\|_1$$

Empirical validation on real-life datasets

	data set name	T	m	p
1	Air passengers data	144	0	1
2	Distillation column	90	5	3
3	pH process	2001	2	1
4	Hair dryer	1000	1	1
5	Heat flow density	1680	2	1
6	Heating system	801	1	1

G. Box, and G. Jenkins. Time Series Analysis: Forecasting and Control, Holden-Day, 1976

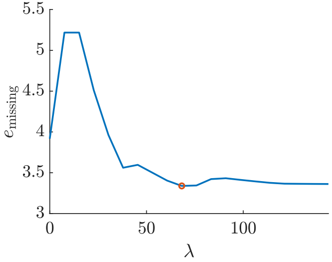
B. De Moor, et al.*DAISY: A database for identification of systems. Journal A, 38:4–5, 1997*

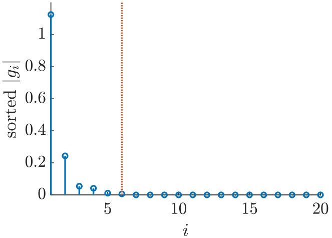
ℓ_1-norm regularization with optimized λ achieves the best performance

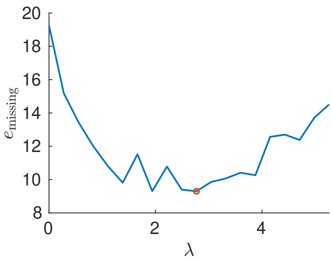
$$e_{\text{missing}} := \frac{\left\| w|_{I_{\text{missing}}} - \widehat{w}|_{I_{\text{missing}}} \right\|}{\left\| w|_{I_{\text{missing}}} \right\|} \, 100\%$$

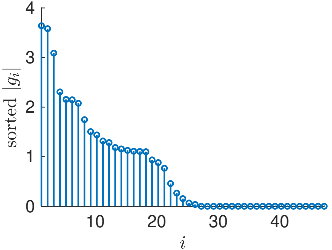
	data set name	naive	ML	LASSO
1	Air passengers data	3.9	fail	3.3
2	Distillation column	19.24	17.44	9.30
3	pH process	38.38	85.71	12.19
4	Hair dryer	12.35	8.96	7.06
5	Heat flow density	7.16	44.10	3.98
6	Heating system	0.92	1.35	0.36

Tuning of λ and sparsity of g (datasets 1, 2)

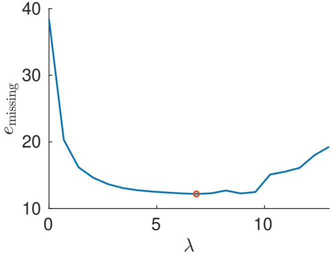


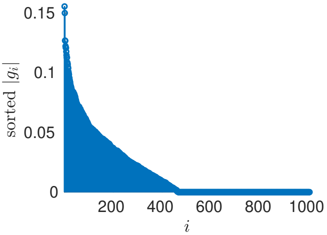


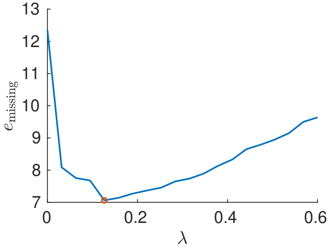


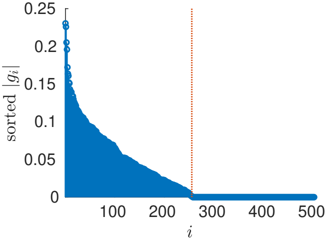


Tuning of λ and sparsity of g (datasets 3, 4)

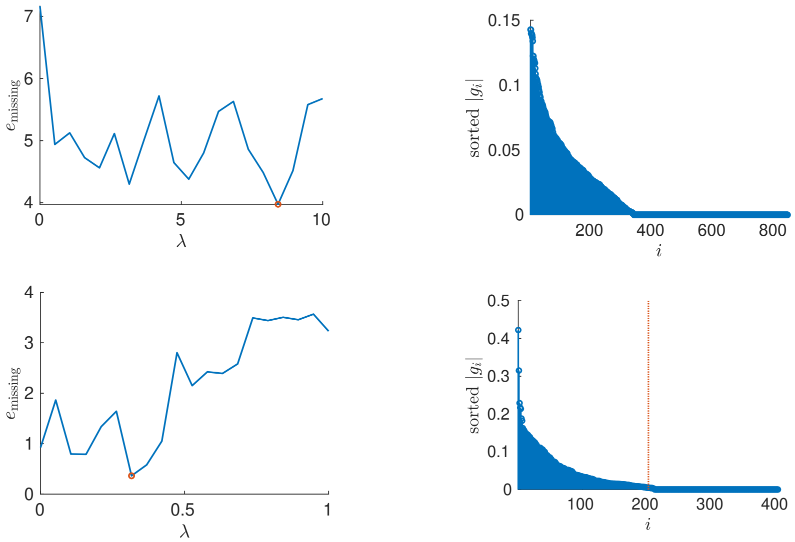








Tuning of λ and sparsity of g (datasets 5, 6)



Summary: convex relaxations

w_{d} exact $\rightsquigarrow$ system theory

- exact analytical solution
- current work: efficient real-time algorithms

w_{d} inexact $\rightsquigarrow$ nonconvex optimization

- subspace methods
- local optimization
- convex relaxations

empirical validation

- the naive approach works (surprisingly) well
- parametric local optimization is not robust
- ℓ_1-norm regularization gives the best results