Exact and approximate modeling in the behavioral setting

Ivan Markovsky

PhD defense presentation

1 February 2005

Overview

Basic problem: data $\mapsto$ model

given: data (*e.g.*, measurements of an experiment)

$$\mathscr{W} := \{ w(1), \dots, w(T) \}$$

find:
- i) a linear static model $\mathscr{B}_1$
- ii) a quadratic static model $\mathscr{B}_2$ that best fits $\mathscr{W}$
- iii) an LTI dynamic model $\mathscr{B}_3$

LTI — linear time-invariant

Basic problem: data $\mapsto$ model

- What is a model? (in particular, linear, quadratic, LTI)
- What does it mean "the model fits the data well"?
- How to measure the fitting accuracy and find optimal models?

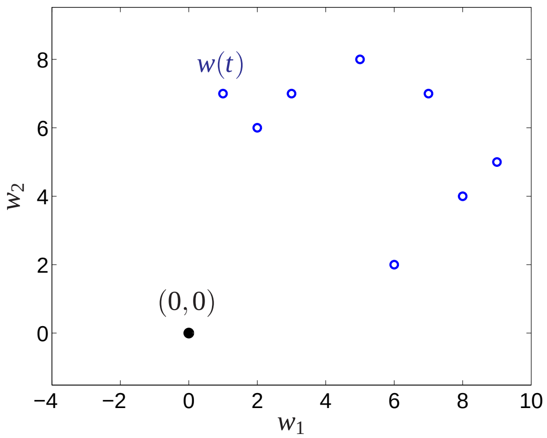
goals: find algorithms that realize the mappings

$$\mathscr{W} \mapsto \mathscr{B}_1, \quad \mathscr{W} \mapsto \mathscr{B}_2, \quad \mathscr{W} \mapsto \mathscr{B}_3, \qquad \text{with } \mathscr{B}_1, \mathscr{B}_2, \mathscr{B}_3 \text{ "optimal"}$$

implement these algorithms in a ready to use software

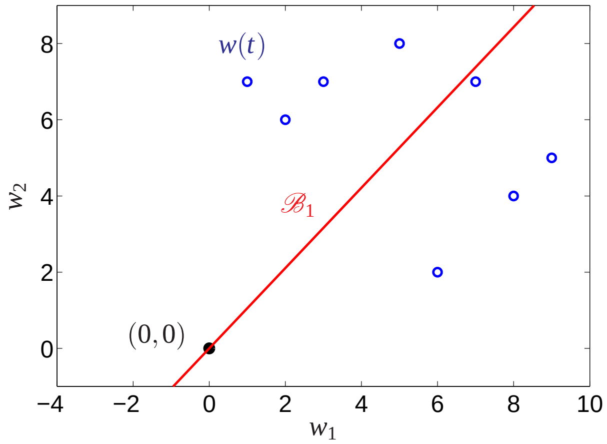
Example with 2 variables and 8 data points

$$w(1) = \begin{bmatrix} 1 \\ 7 \end{bmatrix}, \; w(2) = \begin{bmatrix} 2 \\ 6 \end{bmatrix}, \; w(3) = \begin{bmatrix} 5 \\ 8 \end{bmatrix}, \; \dots, \; w(8) = \begin{bmatrix} 8 \\ 4 \end{bmatrix}$$



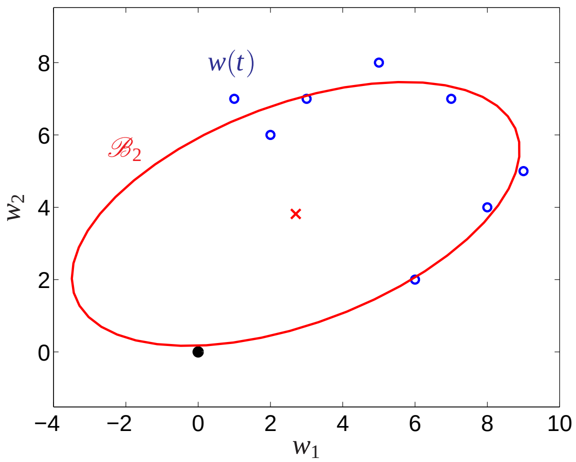
Linear static model

a (nontrivial) linear static model in $\mathbb{R}^2$ is a line through $(0,0)$



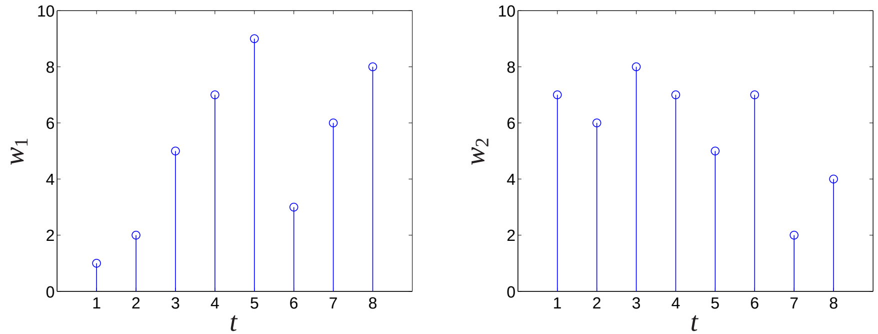
Quadratic static model

a (nondegenerate) quadratic static model in $\mathbb{R}^2$ is an ellipse



Linear dynamic model

the data $\mathscr{W}$ is viewed now as a vector time series $w = \big(w(1), \dots, w(8)\big)$
(note that in this case the ordering of the data points is important)



we look for a first order LTI model with one input

Linear dynamic model

a first order LTI model with one input can be represented by a

$\boxed{\text{scalar difference equation with one time lag}}$

$$R_0 w(t) + R_1 w(t+1) = 0, \quad \text{for } t = 1, 2, \ldots, 7, \quad \text{where } R_0, R_1 \in \mathbb{R}^{1\times 2}$$

let $R_1 =: \begin{bmatrix} Q_1 & -P_1 \end{bmatrix}$ and suppose that $P_1 \neq 0$, then

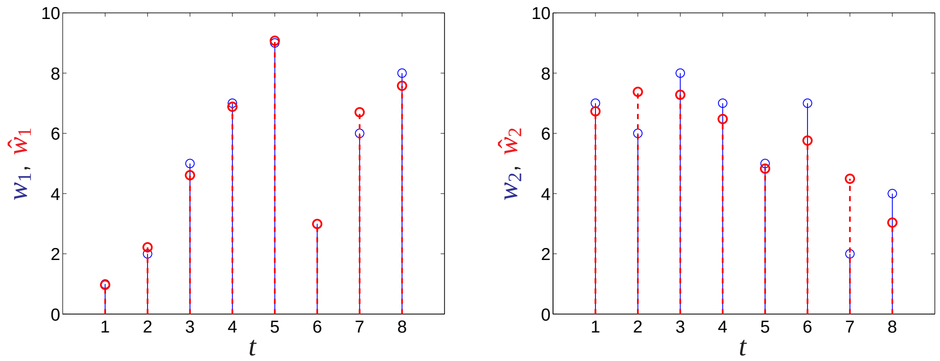
$\boxed{w_1 \text{ is an input (free) and } w_2 \text{ is an output (bound)}}$

$$w_1 \longrightarrow \boxed{\text{LTI model}} \longrightarrow w_2$$

Linear dynamic model

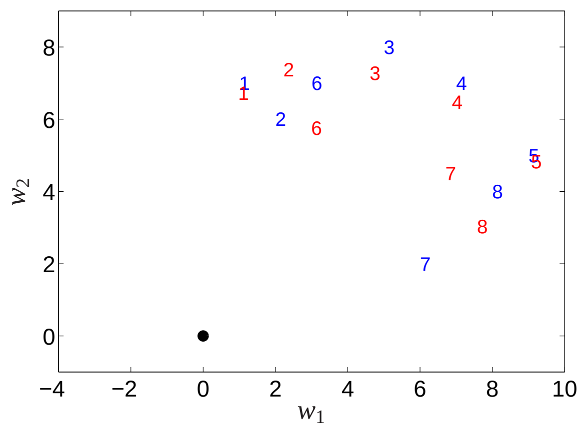
consider the model $\mathscr{B}_3 : \begin{bmatrix} 0.13 & 1.22 \end{bmatrix} w(t) - \begin{bmatrix} 0.44 & 1 \end{bmatrix} w(t+1) = 0$

data w a particular trajectory $\hat{w}$ of $\mathscr{B}_3$



Linear dynamic model

the data w and the trajectory $\hat{w}$ of $\mathscr{B}_3$ visualized in the plane



Summary

- A model is a subset of the data space.   (behavior of the model)

  linear static model: subspace of $\mathbb{R}^{\mathtt{w}}$, $\mathtt{w} := \dim\big(w(t)\big)$

  quadratic static model: hyperbola, parabola, or ellipsoid in $\mathbb{R}^{\mathtt{w}}$

  finite dim. LTI model: shift-invariant closed subspace of $(\mathbb{R}^{\mathtt{w}})^{\mathbb{Z}}$

next

- What does it mean "the model fits the data well"?

- How to measure the fitting accuracy and find optimal models?

Fitting accuracy (static case)

consider a given model $\mathscr{B} \subseteq \mathbb{R}^{\mathtt{w}}$ and data $\mathscr{W} = \{w(1),\dots,w(T)\}$

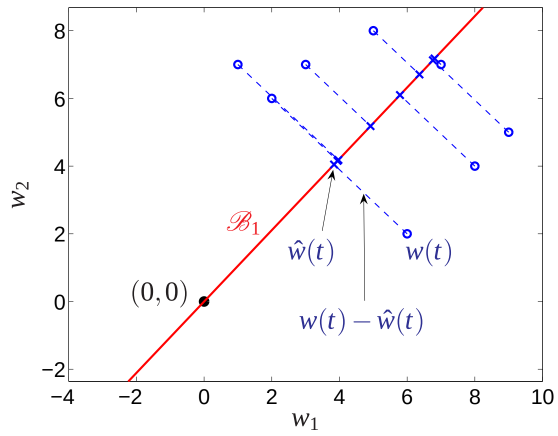
the misfit (w.r.t. to the norm $\|\cdot\|$) between $\mathscr{B}$ and $\mathscr{W}$ is defined as

$$M(\mathscr{W},\mathscr{B}) := \min_{\hat{w}(1),\dots,\hat{w}(T)\in\mathscr{B}} \sqrt{\sum_{t=1}^{T} \|w(t)-\hat{w}(t)\|^2}$$

the model $\mathscr{B}$ fits the data $\mathscr{W}$ "well" if the misfit $M(\mathscr{W},\mathscr{B})$ is "small"

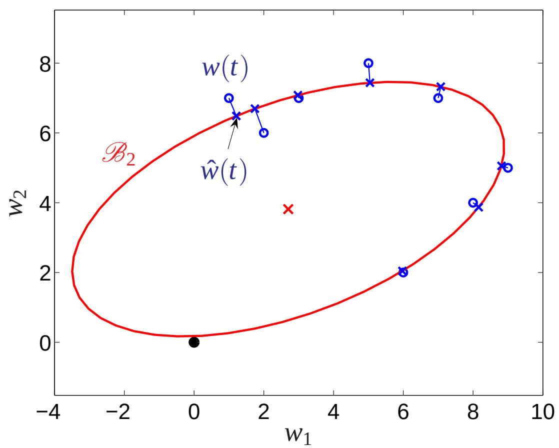
note: $M(\mathscr{W},\mathscr{B}) = 0 \iff \mathscr{B}$ is an exact model for $\mathscr{W}$

Example: linear static model



$$M(\mathscr{W},\mathscr{B}_1) = \min_{\hat{w}(1),\dots,\hat{w}(8)\in\mathscr{B}_1} \sqrt{\sum_{t=1}^{8} \|w(t)-\hat{w}(t)\|^2} = 7.8865$$

Example: quadratic static model



$$M(\mathscr{W},\mathscr{B}_2) = \min_{\hat{w}(1),\dots,\hat{w}(8)\in\mathscr{B}_2} \sqrt{\sum_{t=1}^{8} \|w(t)-\hat{w}(t)\|^2} = 1.1719$$

Fitting accuracy (dynamic case)

consider a given model $\mathscr{B} \subseteq (\mathbb{R}^{\mathtt{w}})^T$ and data $w = \big(w(1),\dots,w(T)\big)$

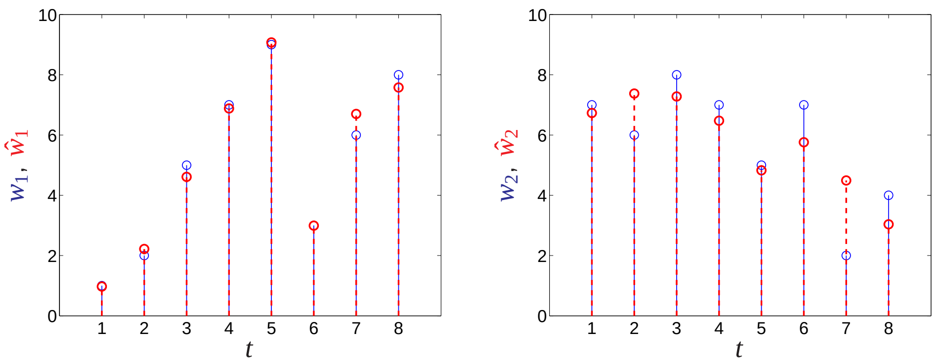
misfit (w.r.t. to the norm $\|\cdot\|$) between $\mathscr{B}$ and w is defined as

$$M(w,\mathscr{B}) := \min_{\hat{w}\in\mathscr{B}} \|w-\hat{w}\|$$

the model $\mathscr{B}$ fits the data w "well" if the misfit $M(w,\mathscr{B})$ is "small"

note: $M(w,\mathscr{B}) = 0 \iff \mathscr{B}$ is an exact model for w

Example: linear dynamic model



$$M(\mathscr{W}, \mathscr{B}_3) = \min_{\hat{w} \in \mathscr{B}_3} \|w - \hat{w}\| = 3.5144$$

Optimal approximate model

$\mathscr{M}$ — given model class, in the example

i) all lines in $\mathbb{R}^2$ passing through $(0,0)$
ii) all ellipses in $\mathbb{R}^2$
iii) all first order LTI systems with one input

find the model $\mathscr{B}^*$ in $\mathscr{M}$ that best fits the data

$$\mathscr{B}^* := \arg \min_{\mathscr{B} \in \mathscr{M}} M(\mathscr{W}, \mathscr{B})$$

the models $\mathscr{B}_1$, $\mathscr{B}_2$, and $\mathscr{B}_3$ are optimal; they are computed by algorithms and software that treat the general case

Summary

- the model $\mathscr{B}$ fits the data $\mathscr{W}$ "well" if the misfit $M(\mathscr{W}, \mathscr{B})$ is small
- $M(\mathscr{W}, \mathscr{B})$ is a quantitative measure of the model quality
- $\mathscr{B}^* = \arg\min_{\mathscr{B} \in \mathscr{M}} M(\mathscr{W}, \mathscr{B})$ is an optimal model for $\mathscr{W}$ in $\mathscr{M}$

next

- find algorithms for the computation of $\mathscr{B}^*$

Approximation problems $AX \approx B$

many classical approximation problems are of the type:

$\boxed{\text{given } A \text{ and } B\text{, solve for } X\text{, an overdetermined system } AX \approx B}$

typically there is no exact solution $\rightsquigarrow$ basic idea: modify A and B

$\boxed{A + \Delta A =: \hat{A},\ B + \Delta B =: \hat{B},\ \text{so that } \hat{A}X = \hat{B} \text{ is solvable}}$

in addition, preserve the structure (if any) of $\begin{bmatrix} A & B \end{bmatrix}$ in $\begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix}$

typical structures in A and B are block-Hankel and block-Toeplitz

Examples of static approximation problems

in static approximation problems $AX \approx B$, A and B are unstructured

the modification of A or B might be forbidden, *i.e.*, $\Delta A = 0$ or $\Delta B = 0$
in this case, we say that A or B is fixed (exact)

classical examples:

1. Least squares — A fixed, B unstructured
2. Data least squares — A unstructured, B fixed
3. Total least squares — A and B unstructured (line fitting model $\mathscr{B}_1$)

Examples of dynamic approximation problems

4. Finite Impulse Response system identification

   A block-Toeplitz (blocks #inputs$\times$#outputs), B unstructured

5. Impulse response approximation

   $\begin{bmatrix} A & B \end{bmatrix}$ block-Hankel, block size: #inputs$\times$#outputs

6. Global total least squares (diff. eqn. fitting, model $\mathscr{B}_3$)

   $\begin{bmatrix} A & B \end{bmatrix}$ block-Hankel, block size: #time series$\times$#variables

7. Output error identification

   A fixed, B block-Hankel, block size: #time series$\times$#outputs

LTI model fitting $\rightsquigarrow$ block-Hankel structure

consider the vector difference equation

$$R_0 w(t) + R_1 w(t+1) + \cdots + R_l w(t+l) = 0$$

for $t = 1, \ldots, T-l$, it is equivalent to the system of equations

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_l \end{bmatrix} \underbrace{\begin{bmatrix} w(1) & w(2) & w(3) & \cdots & w(T-l) \\ w(2) & w(3) & w(4) & \cdots & w(T-l+1) \\ w(3) & w(4) & w(5) & \cdots & w(T-l+1) \\ \vdots & \vdots & \vdots & & \vdots \\ w(l+1) & w(l+2) & w(l+3) & \cdots & w(T) \end{bmatrix}}_{\text{block-Hankel structured matrix}} = 0$$

Unification

static problems — unstructured
dynamic problems — block-Toeplitz/Hankel structure

question: How to unify these approximation problems?

answer: the right formalization turns out to be what is called

the structured total least squares (STLS) problem

STLS—tool for approximation by static and dynamic linear models

($\mathscr{B}_1$ and $\mathscr{B}_3$ but not $\mathscr{B}_2$ are computed by solving STLS problems)

Structured total least squares

structure specification $\mathscr{S}$: parameters $\mapsto$ structured matrices

STLS problem: given structure $\mathscr{S}$, parameter p, and rank n, find

$$\hat{p}_{\text{stls}} = \arg\min_{\hat{p}} \|p - \hat{p}\| \quad \text{subject to} \quad \operatorname{rank}\big(\mathscr{S}(\hat{p})\big) \leq n$$

perturb p as little as necessary, so that the perturbed structured matrix $\mathscr{S}(\hat{p})$ becomes rank deficient with rank at most n

$$\operatorname{rank}\big(\mathscr{S}(\hat{p})\big) \leq n \quad \iff \quad \exists\, X \in \mathbb{R}^{n\times\bullet} \text{ such that } \mathscr{S}(\hat{p}) \begin{bmatrix} X \\ -I \end{bmatrix} = 0$$

Efficient computation

double minimization problem

$$\min_X \left(\min_{\hat{p}} \|p - \hat{p}\| \quad \text{subject to} \quad \mathscr{S}(\hat{p}) \begin{bmatrix} X \\ -I \end{bmatrix} = 0 \right)$$

minimizing analytically over p gives the equivalent problem

$$\min_X \Big(\mathscr{S}(p) \left[\begin{smallmatrix} X \\ -I \end{smallmatrix}\right]\Big)^{\top} \Gamma^{-1}(X) \Big(\mathscr{S}(p) \left[\begin{smallmatrix} X \\ -I \end{smallmatrix}\right]\Big)$$

$\Gamma(X)$ is block-banded and Toeplitz for a large class of structure specifications $\mathscr{S}$ that includes in particular all examples listed before

the structure of Γ allows efficient cost function and gradient evaluation $\rightsquigarrow$ efficient local optimization algorithms

Advantages over alternative algorithms

- flexible structure specification
- easily generalized to
  - diagonal weighting in the cost function
  - regularization
- software implementation is available

recognizing the structure of Γ encapsulates core computational problem:

Cholesky factorization of block-banded and Toeplitz matrix

we use software from SLICOT in order to solve this core problem

Summary

- STLS — optimal data fitting by structured linear models
- exploiting the structure $\rightsquigarrow$ efficient algorithms for optimal modeling

Exact identification

given: a vector time series

$$w = \big(w(1),\dots,w(T)\big)$$

generated by an LTI system $\mathscr{B}$

find: the system $\mathscr{B}$ back from the data w

note: the given data is exact and the identified system fits exactly w
the time horizon T is much larger than the order n of $\mathscr{B}$

Algorithms for exact identification

1. $w \mapsto$ difference equation R
2. $w \mapsto$ impulse response H
3. $w \mapsto$ input/state/output representation (A,B,C,D)

3.a. $w \mapsto R \mapsto (A,B,C,D)$ or $w \mapsto H \mapsto (A,B,C,D)$

3.b. $w \mapsto$ observability matrix $\mapsto (A,B,C,D)$

3.c. $w \mapsto$ state sequence $\mapsto (A,B,C,D)$

Persistency of excitation

a condition for solvability of the exact identification problem

definition: the sequence $u = \big(u(1),\dots,u(T)\big)$ is

persistently exciting of order L

if the Hankel matrix

$$\mathscr{H}_L(u) := \begin{bmatrix} u(1) & u(2) & u(3) & \cdots & u(T-L+1) \\ u(2) & u(3) & u(4) & \cdots & u(T-L+2) \\ u(3) & u(4) & u(5) & \cdots & u(T-L+3) \\ \vdots & \vdots & \vdots & & \vdots \\ u(L) & u(L+1) & u(L+2) & \cdots & u(T) \end{bmatrix}$$

is of full row rank

Fundamental Lemma

Let $\mathscr{B}$ be controllable and let $w := (u,y) \in \mathscr{B}|_{[1,T]}$. Then, if u is persistently exciting of order $L+n$, where n is the order of $\mathscr{B}$,

$$\text{image}\left(\begin{bmatrix} w(1) & w(2) & w(3) & \cdots & w(T-L+1) \\ w(2) & w(3) & w(4) & \cdots & w(T-L+2) \\ w(3) & w(4) & w(5) & \cdots & w(T-L+3) \\ \vdots & \vdots & \vdots & & \vdots \\ w(L) & w(L+1) & w(L+2) & \cdots & w(T) \end{bmatrix}\right) = \mathscr{B}|_{[1,L]}$$

$\Longrightarrow$ with $L = l+1$, where l is the lag of $\mathscr{B}$, the FL gives conditions for identifiability, namely "u persistently exciting of order $l+1+n$"

$\Longrightarrow$ under the conditions of the FL, any L samples long trajectory of $\mathscr{B}$ can be obtained as $\mathscr{H}_L(w)g$, for certain g $\rightsquigarrow$ algorithms

Example $w \mapsto$ impulse response H

under the conditions of FL, there is G, such that $H = \mathscr{H}_t(y)G$

the problem reduces to the one of finding a particular G

$$\begin{bmatrix} \mathscr{H}_{l+t}(u) \\ \hline \mathscr{H}_{l+t}(y) \end{bmatrix} G = \begin{bmatrix} 0 \\ \left[\begin{smallmatrix} I \\ 0 \end{smallmatrix}\right] \\ \hline 0 \\ H \end{bmatrix} \quad \begin{array}{l} \leftarrow \; l \text{ zero samples} \\ \leftarrow \; t \text{ samples long impulse} \\ \hline \leftarrow \; l \text{ zero samples} \\ \leftarrow \; t \text{ samples impulse response} \end{array}$$

block algorithm:

1. solve the system of equations in blue for G

2. substitute G in the equations in red $\rightsquigarrow H$

Simulation example $w \mapsto$ impulse response H

$\mathscr{B}$ is of order $n = 4$, lag $l = 2$, with $m = 2$ inputs, and $p = 2$ outputs

w is a trajectory of $\mathscr{B}$ with length $T = 500$

estimation error $e = ||H - \hat{H}||_{\mathrm{F}}$ and execution time for three methods

method	error, e	time, sec.
block algorithm	10^{-14}	0.293
iterative algorithm	10^{-14}	0.066
`impulse`*	0.059	0.584

* from System Identification Toolbox of MATLAB

Summary

- deterministic subspace algorithms are implementations of the FL

  $w \mapsto$ obsv. matrix $\mapsto (A,B,C,D)$ — MOESP-type algorithms

  $w \mapsto$ state sequence $\mapsto (A,B,C,D)$ — N4SID-type algorithms

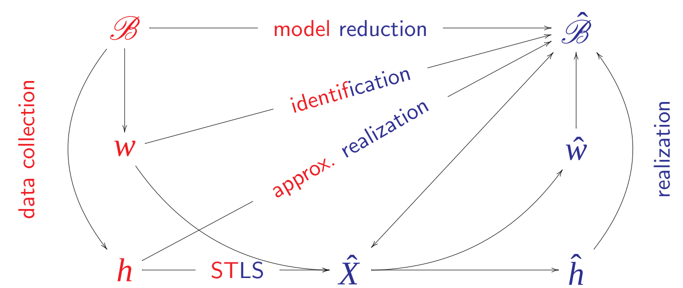
- the FL reveals the meaning of the oblique and orthogonal projections

  computation of special responses from data

- the FL gives identifiability conditions that are verifiable from w

LTI approximate modeling

$\mathscr{B}$ — "true" (high order) model	w	— observed response
	h	— observed impulse resp.
$\hat{\mathscr{B}}$ — approximate (low order) model	$\hat{w}$	— response of $\hat{\mathscr{B}}$
	$\hat{h}$	— impulse resp. of $\hat{\mathscr{B}}$



STLS as a kernel subproblem

- SVD-based methods:

  balanced model reduction, subspace identification, and Kung's alg. use the singular value decomposition in order to find a rank deficient matrix $\mathscr{H}(\hat{w})$ approximating a given full rank matrix $\mathscr{H}(w)$

  note that SVD is suboptimal in terms of the misfit criterion $\|w-\hat{w}\|_{\ell_2}^2$

- STLS-based methods:

  optimal approximation according to the misfit criterion

  need initial approximation (*e.g.*, from SVD-based method)

  iterative improvement of heuristic suboptimal solution

Data sets from DAISY

#	Data set name	T	m	p	l
1	Data of a simulation of the western basin of Lake Erie	57	5	2	1
2	Data of Ethane-ethylene destillation column	90	5	3	1
3	Data of a 120 MW power plant	200	5	3	2
4	Heating system	801	1	1	2
5	Data from an industrial dryer (Cambridge Control Ltd)	867	3	3	1
6	Data of a laboratory setup acting like a hair dryer	1000	1	1	5
7	Data of the ball-and-beam setup in SISTA	1000	1	1	2
8	Wing flutter data	1024	1	1	5
9	Data from a flexible robot arm	1024	1	1	4

Data sets from DAISY (cont.)

#	Data set name	T	m	p	l
10	Data of a glass furnace (Philips)	1247	3	6	1
11	Heat flow density through a two layer wall	1680	2	1	2
12	Simulation data of a pH neutralization process	2001	2	1	6
13	Data of a CD-player arm	2048	2	2	1
14	Data from a test setup of an industrial winding process	2500	5	2	2
15	Liquid-saturated steam heat exchanger	4000	1	1	2
16	Data from an industrial evaporator	6305	3	3	1
17	Continuous stirred tank reactor	7500	1	2	1
18	Model of a steam generator at Abbott Power Plant	9600	4	4	1

Simulation setup

the approximations obtained by the following methods are compared:

`stls` — misfit minimization method
`pem` — the prediction error method (Identification Toolbox)
`subid` — robust combined subspace algorithm

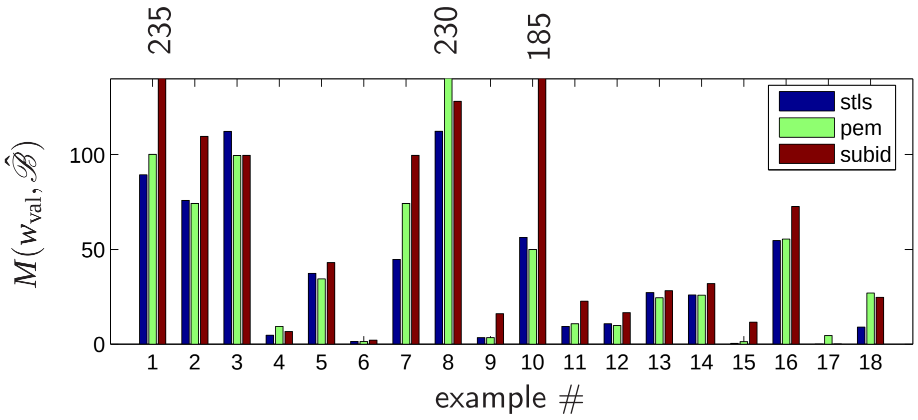
(initial approximation for `stls` and `pem` is the result of `subid`)

a model $\hat{\mathscr{B}}$ is obtained from w_{id} — the first 70% of the data w

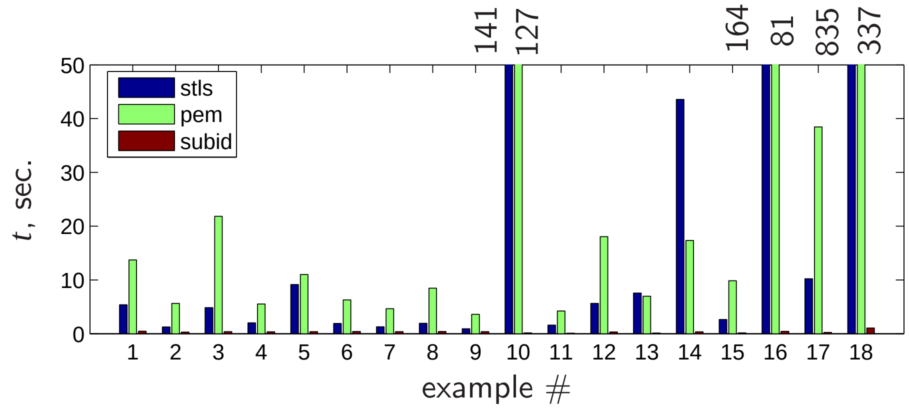
we consider output error identification, *i.e.*, the input is assumed exact

and compare the misfit $M(w_{\mathrm{val}}, \hat{\mathscr{B}})$ on the last 30% of the data w and the execution time for computing $\hat{\mathscr{B}}$

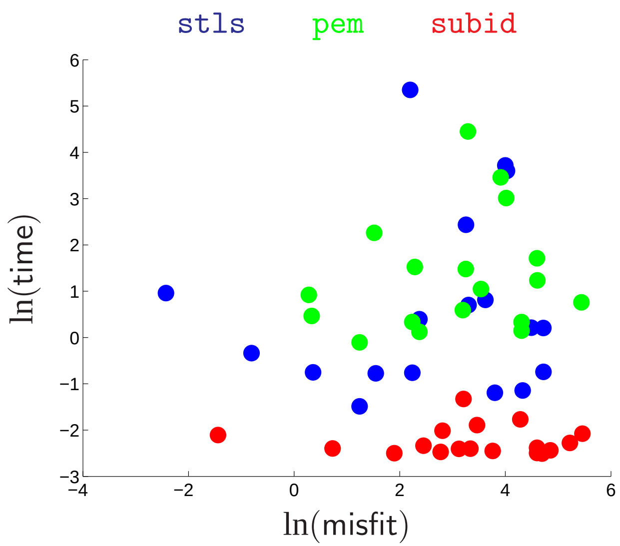
Simulation results — output error



Simulation results — execution time



Simulation results — scatter plot misfit vs time



Summary

- STLS is a kernel problem for approximate LTI modeling

  approx. realization, model reduction, system ident., *etc.*

- a single algorithm can solve a large variety of problems

- the software implementation can solve problems with

  a few thousands data points ($T < 10000$), a few outputs ($p < 10$), and a few time lags ($l < 10$)

Insights

- models are sets of allowed outcomes from a universum of outcomes

  the representation free (behavioral) setting gives a notion of equivalence

- apriori fixed input/output partition (*e.g.*, $AX = B$) $\rightsquigarrow$ "nongeneric problems"

  kernel and image representations do not suffer from this shortcoming

- a convenient repr. for LTI model is polynomial matrix in one variable

  $\rightsquigarrow$ kernel representation $\equiv$ difference equation representation

- the EIV model $\mathscr{W} = \bar{\mathscr{W}} + \tilde{\mathscr{W}}$, $\bar{\mathscr{W}} \in \bar{\mathscr{B}}$, $\tilde{\mathscr{W}} \sim \mathrm{N}(0, \sigma^2 V)$ is not as convincing starting point as the deterministic misfit $\mathscr{W} = \hat{\mathscr{W}} + \Delta\mathscr{W}$

Contributions

- new formulation and efficient solution method of the STLS problem

  software implementation and C and MATLAB

- adjusted least squares estimation of elipsoids

  suboptimal in the misfit sense but very effective and efficient

- identifiability condition and algorithms for exact identification

- balanced model identification algorithms

- equivalence of the classical and errors-in-variables Kalman filters

- application of STLS for approximate system identification

Thesis contents

Weighted total least squares	Chapter 2
Structured total least squares	Chapter 3
Fundamental matrix and ellipsoid estimation	Chapters 4 and 5
Exact system identification	Chapters 7 and 8
Errors-in-variables Kalman filtering	Chapter 9
Approximate system identification	Chapter 10

Current and planned future work

- recursive identification methods

- extend the misfit framework with unobserved (latent) variables

- find link with the prediction error methods

- algorithms for STLS problems using kernel and image representations