Hochschule
München (HM)
University of
Applied Sciences

Laboratory for
Mechatronic
and Renewable
Energy Systems
(LMRES)

Lecture series on “System identification” (a proposal)

Christoph M. Hackl
christoph.hackl@hm.edu

26.09.2024 – Haus der bayerischen Landwirtschaft, Herrsching

Outline

Outline

Outline

Introduction: Motivation

Output of DeepAI Figure Generator when prompted with “Improving PhD seminar, more fun, more input, more interest”

Introduction: Lecture series on “System identification”

Possible agenda

Date	Topic
26.09.2024	**Part I: System identification in general and by (Recursive) Least Squares**
??.??.202?	**Part II: System identification by Artificial Neural Networks**
??.??.202?	**Part III: System identification by Extended Observers**
??.??.202?	**Part IV: ...**

Introduction: Lecture series on “System identification”

Time schedule and learning goals

Lecture series: “Part I: System identification by (Recursive) Least Squares”

- $\approx$ 20 min of lecture
- $\approx$ 10 min of tutorial (e.g., using Matlab/Simulink)
- $\approx$ 1-2 hours of homework / self-study (if of interest)

Teaching/learning goals:

After this lecture series you will be able to:

- understand and apply the required mathematical background(s);
- understand, apply and implement data set preparation;
- understand, apply, implement and analyze different system identification methods;
- understand, apply and adjust tuning parameters of different algorithms;

Outline

Introduction: Notation

List of symbols (#1)

Symbol	Explanation
$\mathbb{N}$	$:= \{1, 2, 3, \ldots\}$, set of natural numbers
$\mathbb{R}$	$:= (-\infty, \infty)$, set of real numbers
$\mathbb{C}$	$:= x + \jmath y$, set of complex numbers with $x, y \in \mathbb{R}$ and $\sqrt{\jmath} = -1$
$\boldsymbol{v}$	$:= \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} \in \mathbb{R}^n$, (column) vector with $v_i \in \mathbb{R}$ for all $i \in \{1, \ldots, n\}$
$\boldsymbol{v}^\top$	$:= \begin{pmatrix} v_1, & \ldots, & v_n \end{pmatrix}$, transposed (row) vector of $\boldsymbol{v} \in \mathbb{R}^n$
$\boldsymbol{0}_n$	$:= \begin{pmatrix} 0, & \ldots, & 0 \end{pmatrix}^\top \in \mathbb{R}^n$, column vector of n zeros
$\boldsymbol{1}_n$	$:= \begin{pmatrix} 1, & \ldots, & 1 \end{pmatrix}^\top \in \mathbb{R}^n$, column vector of n ones
$\lVert\boldsymbol{v}\rVert$	$:= \lVert\boldsymbol{v}\rVert_2 := \sqrt{\boldsymbol{v}^\top \boldsymbol{v}} = \sqrt{v_1^2 + \cdots + v_n^2}$, the Euclidean norm (or 2-norm) of $\boldsymbol{x} \in \mathbb{R}^n$
$\mathrm{diag}\{a_1, \ldots, a_n\}$	$:= \begin{bmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_n \end{bmatrix} \in \mathbb{R}^{n \times n}$, diagonal matrix with elements $a_1, \ldots, a_n \in \mathbb{R}$
$\boldsymbol{I}_n$	$:= \mathrm{diag}\{1, \ldots, 1\} \in \mathbb{R}^{n \times n}$, identity (or unit) matrix
$\boldsymbol{O}_{n \times m}$	$:= \begin{bmatrix} 0 & \ldots & 0 \\ \vdots & & \vdots \\ 0 & \ldots & 0 \end{bmatrix} \in \mathbb{R}^{n \times m}$, zero matrix

Introduction: Notation

List of symbols (#2)

Symbol	Explanation
$\boldsymbol{A}$	$:= \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \in \mathbb{R}^{n \times m}$, matrix with $a_{kl} \in \mathbb{R}$, $k \in \{1, \ldots, n\}$ & $l \in \{1, \ldots, m\}$

In the following, particularly important symbols for system identification:

$\boldsymbol{x}$	$:= (x_1, \ldots, x_m)^\top \in \mathbb{R}^m$, input data vector (e.g., measurements of system input(s))
$\boldsymbol{y}$	$:= (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$, output data vector (e.g., measurements of sys. output(s))
$\widehat{\boldsymbol{y}}$	$:= (\widehat{y}_1, \ldots, \widehat{y}_n)^\top \in \mathbb{R}^n$, estimated output data vector of system identification
$\boldsymbol{\theta}$	$:= \left(\theta_1, \ldots, \theta_p\right)^\top \in \mathbb{R}^p$, system parameter vector
$\widehat{\boldsymbol{\theta}}$	$:= \left(\widehat{\theta}_1, \ldots, \widehat{\theta}_{\widehat{p}}\right)^\top \in \mathbb{R}^{\widehat{p}}$, estimated parameter vector (with possibly $\widehat{p} \neq p$!)

Outline

Introduction: References and additional materials

A few useful references

Besides the lecture series slides, following references may be of help:

[1] T. Söderström, and P. Stoica *System identification*. Prentice Hall International, New York, 1989.

[2] D. Schröder and M. Buss, *Intelligente Verfahren – Identifikation und Regelung nichtlinearer Systeme*. Springer-Verlag, 2017 (https://link.springer.com/book/10.1007/978-3-662-55327-5).

[3] R. Isermann and M. Münchhof, *Identification of Dynamic Systems: An Introduction with Applications*. Springer-Verlag, Berlin Heidelberg 2011 (https://link.springer.com/book/10.1007/978-3-540-78879-9).

[4] L. Ljung, *System Identification: Theory for the User*. Prentice Hall, Inc., New Jersey, 1999.

Outline

2 System identification
- Problem statement
- Principle solution approach
- Typical challenges
- Optimization methods

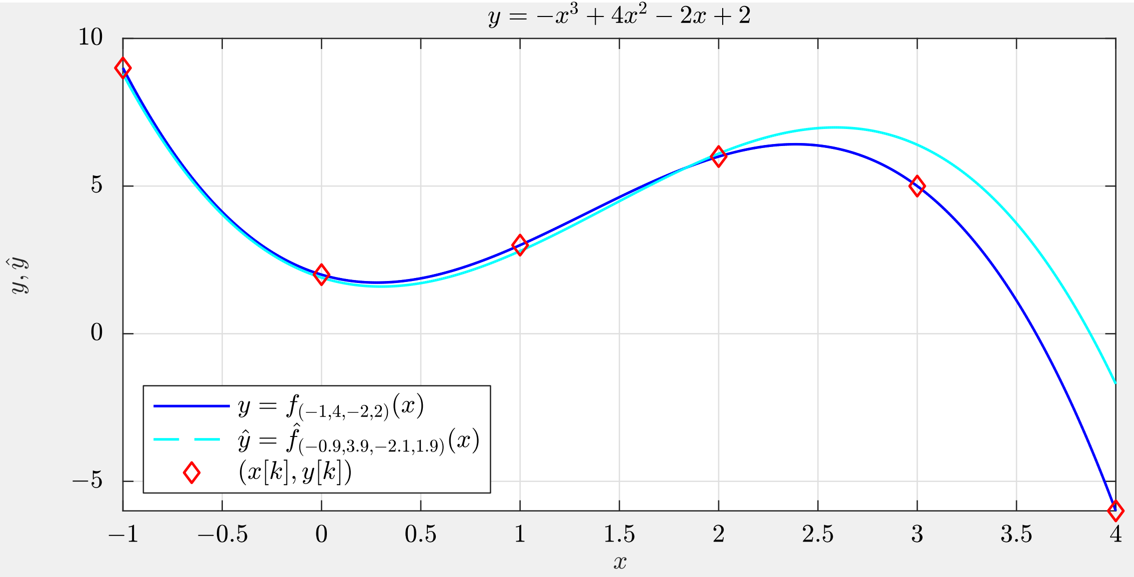
System identification

Output of DeepAI Figure Generator when prompted with “System identification of physical systems”

Outline

System identification: Problem statement

Examplary problem illustration



System identification: Problem statement

Formal description

The unknown function $\boldsymbol{f}_{\boldsymbol{\theta}} : \mathbb{R}^m \to \mathbb{R}^n$ with unknown parametrization $\boldsymbol{\theta} \in \mathbb{R}^p$, $p \in \mathbb{N}$, mapping

$$\boldsymbol{x} := (x_1, \ldots, x_m)^\top \mapsto \boldsymbol{y} := (y_1, \ldots, y_n)^\top := \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) \tag{1}$$

shall be approximated by a known function $\widehat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}} : \mathbb{R}^m \to \mathbb{R}^n$, with unknown parametrization $\widehat{\boldsymbol{\theta}} \in \mathbb{R}^{\hat{p}}$, $\hat{p} \in \mathbb{N}$, mapping

$$\boldsymbol{x} := (x_1, \ldots, x_m)^\top \mapsto \widehat{\boldsymbol{y}} := (\widehat{y}_1, \ldots, \widehat{y}_n)^\top := \widehat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}) \approx \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}). \tag{2}$$

Example

Consider
(a) $y = f_{\boldsymbol{\theta}}(x) = \theta_1 x^2 + \theta_2$ and $\widehat{y} = \widehat{f}_{\hat{\boldsymbol{\theta}}}(x) = \widehat{\theta}_1 x^2 + \widehat{\theta}_2 x + \widehat{\theta}_3$ or
(b) $y = f_{\boldsymbol{\theta}}(x_1, x_2) = \theta_1 x_1^2 + \theta_2 x_2 + \theta_3$ and $\widehat{y} = \widehat{f}_{\hat{\boldsymbol{\theta}}}(x_1, x_2) = \widehat{\theta}_1 x_1^3 + \widehat{\theta}_2 x_1^2 + \widehat{\theta}_3 x_2^2 + \widehat{\theta}_4$.
What can we observe? Which problems may arise?

System identification: Problem statement

Formal description: Best-case and worst-case results

The unknown function $\boldsymbol{f}_{\boldsymbol{\theta}} \colon \mathbb{R}^m \to \mathbb{R}^n$ with unknown parametrization $\boldsymbol{\theta} \in \mathbb{R}^p$, $p \in \mathbb{N}$, mapping

$$\boldsymbol{x} := (x_1, \ldots, x_m)^\top \mapsto \boldsymbol{y} := (y_1, \ldots, y_n)^\top := \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) \tag{3}$$

shall be approximated by a known function $\widehat{\boldsymbol{f}}_{\widehat{\boldsymbol{\theta}}} \colon \mathbb{R}^m \to \mathbb{R}^n$, with unknown parametrization $\widehat{\boldsymbol{\theta}} \in \mathbb{R}^{\hat{p}}$, $\hat{p} \in \mathbb{N}$, mapping

$$\boldsymbol{x} := (x_1, \ldots, x_m)^\top \mapsto \widehat{\boldsymbol{y}} := (\widehat{y}_1, \ldots, \widehat{y}_n)^\top := \widehat{\boldsymbol{f}}_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}) \approx \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}). \tag{4}$$

Remark 1

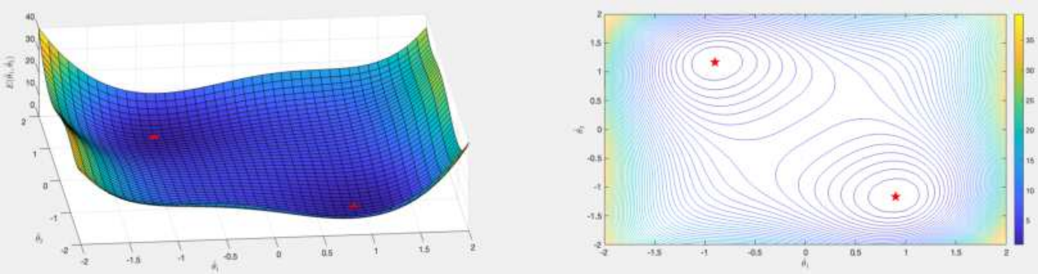
Best-case: (motivates for structered ANNs, e.g. flux linkage prototype functions)
- $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \widehat{\boldsymbol{f}}_{\boldsymbol{\theta}}(\boldsymbol{x})$ (functions equal for same parametrization) and
- $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}$ (parametrizations equal, including $p = \widehat{p}$).

Worst-case: (usual case to be dealt with!)
- $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) \neq \widehat{\boldsymbol{f}}_{\boldsymbol{\theta}}(\boldsymbol{x})$ (functions not equal for same parametrization) and/or
- $\boldsymbol{\theta} \neq \widehat{\boldsymbol{\theta}}$ (parametrizations not equal, including $p \neq \widehat{p}$).

Outline

System identification: Principle solution approach

Illustration of exemplary solution



Approximation error vector

$$e := y - \widehat{y} = y - \widehat{f}_{\widehat{\theta}}(x) \in \mathbb{R}^n \tag{5}$$

Error surface (e.g., squared error or Mean Squared Error (MSE))

$$E\colon \mathbb{R}^{\widehat{p}} \to \mathbb{R}, \quad \widehat{\theta} \mapsto E(\widehat{\theta}) := E(\widehat{\theta}, y, x) := \tfrac{1}{2} e^\top e = \tfrac{1}{2}\left[y^\top y - 2 y^\top \widehat{f}_{\widehat{\theta}}(x) + \widehat{f}_{\widehat{\theta}}(x)^\top \widehat{f}_{\widehat{\theta}}(x)\right] \tag{6}$$

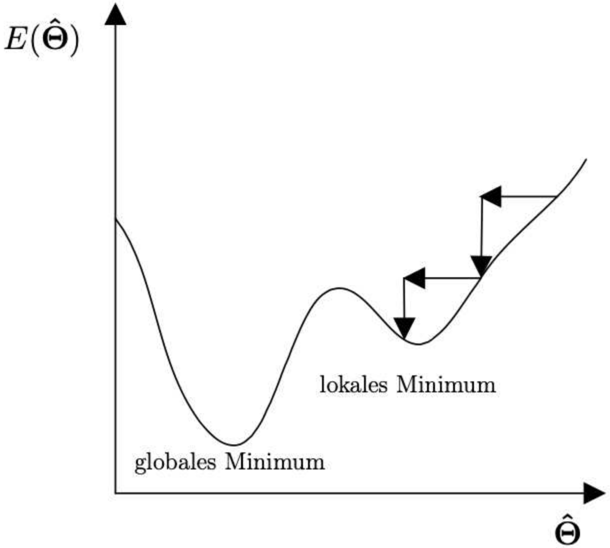
Goal: Solve optimization problem to find minimizer (★)

$$\widehat{\theta}^\star := \arg\min_{\widehat{\theta}} E(\widehat{\theta}) \tag{7}$$

Outline

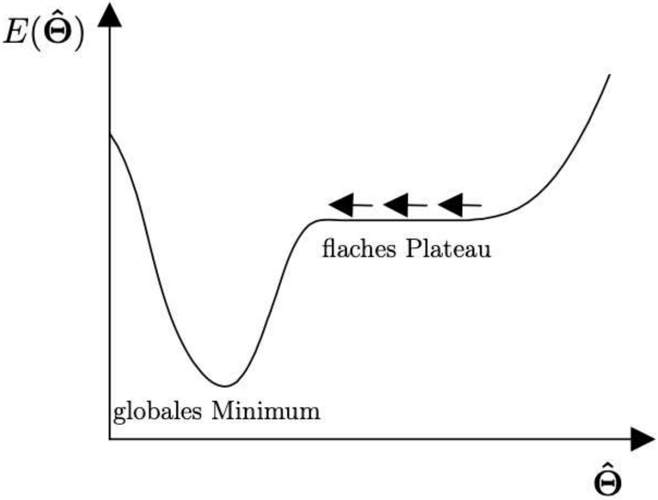
System identification: Typical challenges

"Wrong" initialization $\Rightarrow$ global minmum not reached [1, Abb. 4.12])



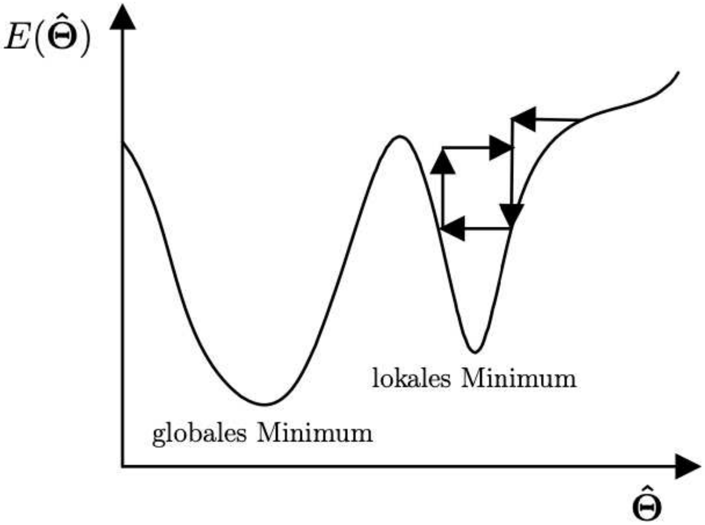
System identification: Typical challenges

Flat plateaus $\Rightarrow$ no progress [1, Abb. 4.13]



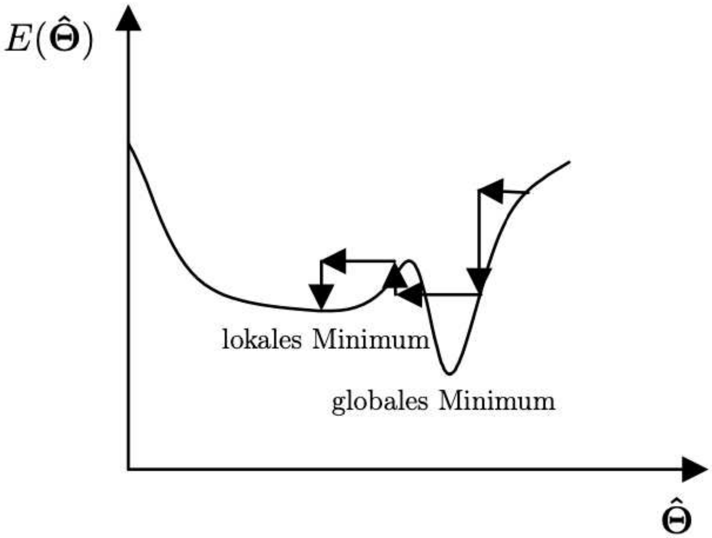
System identification: Typical challenges

Non-convergence (e.g., oscillations) [1, Abb. 4.14]



System identification: Typical challenges

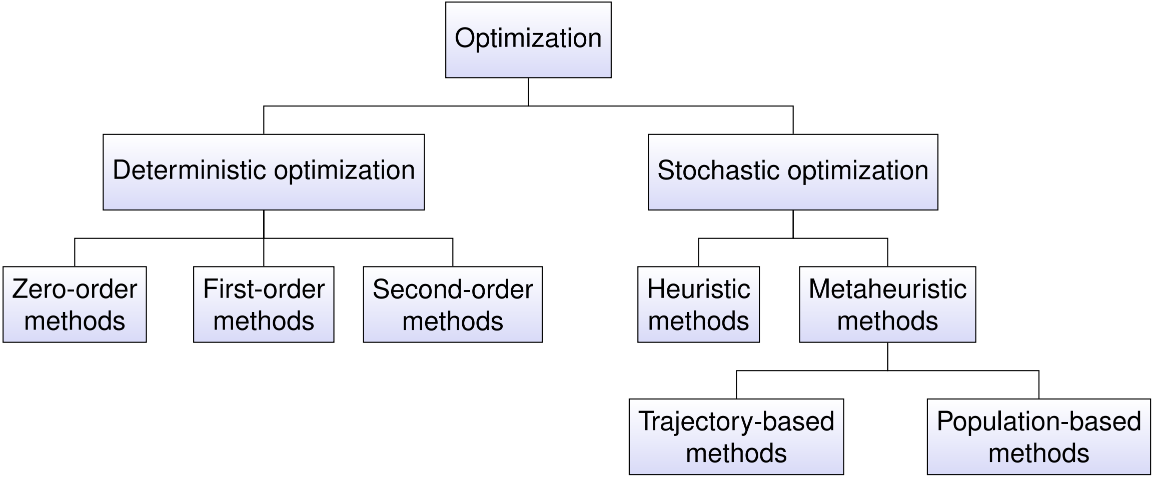
Convergence to local minimum [1, Abb. 4.15]



Outline

System identification: Optimization methods

Tree-like illustration



For more details, see [1, Chapter 10 & 11] or [2, Chapter 3] or https://en.wikipedia.org/wiki/Global_optimization.

System identification: Optimization methods

Optimality conditions for deterministic optimization approaches

Necessary condition:

$$\mathbf{0}_{\hat{p}} \overset{!}{=} \frac{\mathrm{d}E(\hat{\boldsymbol{\theta}})}{\mathrm{d}\hat{\boldsymbol{\theta}}} = -\frac{\mathrm{d}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})^\top}{\mathrm{d}\hat{\boldsymbol{\theta}}}\boldsymbol{y} + \frac{1}{2}\left[\frac{\mathrm{d}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})^\top}{\mathrm{d}\hat{\boldsymbol{\theta}}}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}) + \hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})^\top \frac{\mathrm{d}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\right]$$

$$= -\frac{\mathrm{d}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})^\top}{\mathrm{d}\hat{\boldsymbol{\theta}}}\left(\boldsymbol{y} - \hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})\right)$$

$$= -\underbrace{\frac{\mathrm{d}\hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})^\top}{\mathrm{d}\hat{\boldsymbol{\theta}}}}_{=:\boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top \in \mathbb{R}^{\hat{p}\times n}} \boldsymbol{e} =: \boldsymbol{g}_E \in \mathbb{R}^{\hat{p}} \text{ (gradient of } E \text{ / Jacobian of } \hat{\boldsymbol{f}}) \tag{8}$$

Sufficient condition:

$$0 \overset{!}{<} \frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\frac{\mathrm{d}E(\hat{\boldsymbol{\theta}})}{\mathrm{d}\hat{\boldsymbol{\theta}}} = -\frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\left[\boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top\left(\boldsymbol{y} - \hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})\right)\right] = -\frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\left[\boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top\boldsymbol{e}\right] - \boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top\frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\left[\boldsymbol{y} - \hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})\right]$$

$$= -\frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{\theta}}^\top}\left[\boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top\boldsymbol{e}\right] + \boldsymbol{J}_{\hat{\boldsymbol{f}}}^\top\boldsymbol{J}_{\hat{\boldsymbol{f}}} =: \boldsymbol{H}_E \in \mathbb{R}^{\hat{p}\times\hat{p}} \quad \text{(Hessian of } E) \tag{9}$$

Outline

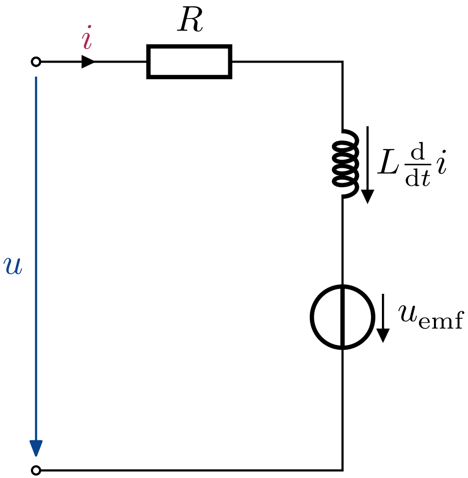
Outline

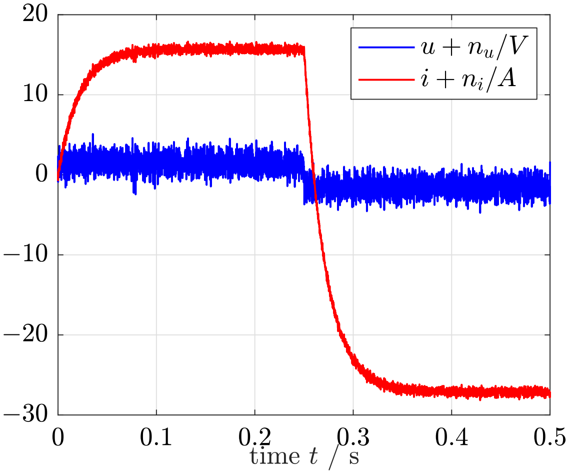
3 System identification by Least Squares

- Motivating example
- Algorithm steps
- Tutorial

System identification by Least Squares: Motivating example

Electrical circuit of RL system with back-emf and measured data



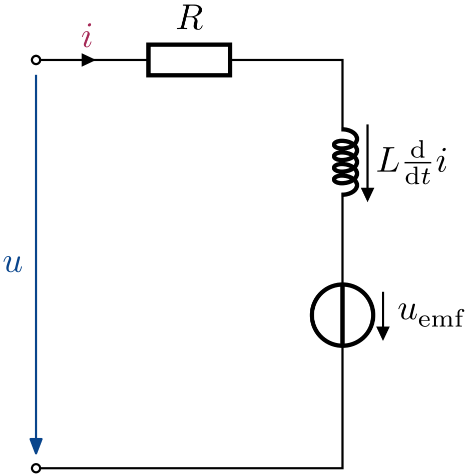


$\Longrightarrow$ How can we estimate R, L and u_{emf} from measured data?

Outline

System identification by Least Squares: Algorithm steps

1. Step: Modeling (GreyBox approach)



Kirchhoff's voltage law gives

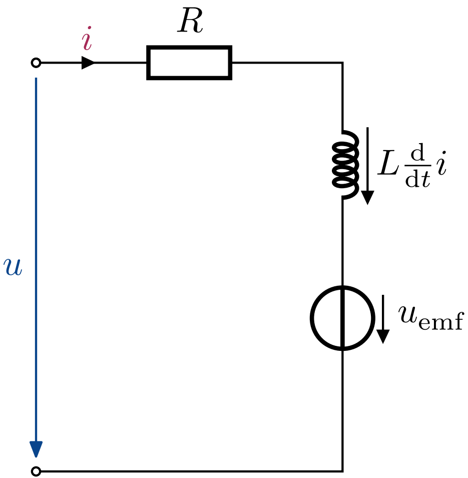
$$u = R\,i + L\frac{\mathrm{d}}{\mathrm{d}t}i + u_{\mathrm{emf}} \Leftrightarrow \frac{\mathrm{d}}{\mathrm{d}t}i = \frac{1}{L}\big(u - R\,i - u_{\mathrm{emf}}\big)$$

Descretization (using Euler forward, i.e. $\frac{\mathrm{d}}{\mathrm{d}t}i \approx \frac{i[k+1]-i[k]}{T_{\mathrm{s}}}$ with sampling time T_{s} and $i[k] = i(kT_{\mathrm{s}})$), yields

$$\boxed{\frac{i[k+1]-i[k]}{T_{\mathrm{s}}} = \frac{1}{L}\big(u[k] - R\,i[k] - u_{\mathrm{emf}}[k]\big)}$$

System identification by Least Squares: Algorithm steps

2. Step: Rewriting in form $\boldsymbol{y} = \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})$ [here: $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{\theta}$ possible, not always the case!]



Recall discretized Grey Box model

$$\frac{i[k+1]-i[k]}{T_{\mathrm{s}}} = \frac{1}{L}\big(u[k] - R\,i[k] - u_{\mathrm{emf}}[k]\big)$$

Define generic system parameters (to be estimated)

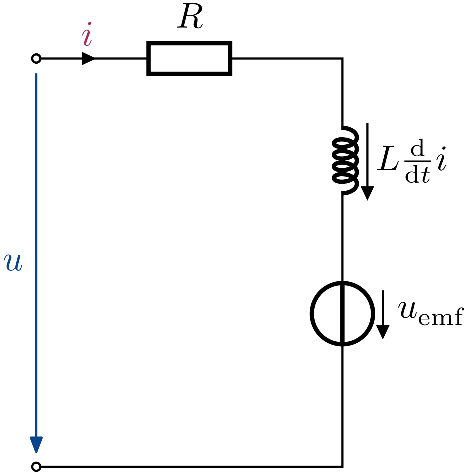
$$\boldsymbol{\theta} := (\theta_1, \theta_2, \theta_3)^\top = \left(\tfrac{1}{L}, \tfrac{R}{L}, \tfrac{u_{\mathrm{emf}}[k]}{L}\right)^\top \in \mathbb{R}^{p=3}$$

then Grey Box model may be rewritten as

$$\underbrace{\frac{i[k+1]-i[k]}{T_{\mathrm{s}}}}_{=: y[k]} = \underbrace{\big(u[k], \quad -i[k], \quad -1\big)}_{=: \boldsymbol{x}[k]^\top} \boldsymbol{\theta}$$

System identification by Least Squares: Algorithm steps

3. Step: Selecting approximation function $\hat{\boldsymbol{y}} = \hat{\boldsymbol{f}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x})$ [$= \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})$ with $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} \in \mathbb{R}^3$] and adjusting dimensions



Approximation function with adjusted dimension

$$\hat{\boldsymbol{y}}[b] = \boldsymbol{X}[b]\hat{\boldsymbol{\theta}}[b]$$

As we have $p = \hat{p} = 3$, we need $b \geqslant 3$, i.e.

$$\hat{\boldsymbol{y}}[b] := \begin{pmatrix} \hat{y}[k] \\ \hat{y}[k+1] \\ \vdots \\ \hat{y}[k+b] \end{pmatrix} \in \mathbb{R}^b$$

and

$$\boldsymbol{X}[b] := \begin{bmatrix} \boldsymbol{x}[k]^\top \\ \boldsymbol{x}[k+1]^\top \\ \vdots \\ \boldsymbol{x}[k+b]^\top \end{bmatrix} \in \mathbb{R}^{b \times \hat{p}}$$

System identification by Least Squares: Algorithm steps

4. Step: Defining error vector and deriving error surface

Approximation error (per batch b; recall (5))

$$\boldsymbol{e}[b] := \boldsymbol{y}[b] - \widehat{\boldsymbol{y}}[b] = \boldsymbol{y}[b] - \widehat{\boldsymbol{f}}_{\widehat{\boldsymbol{\theta}}[b]}(\boldsymbol{x}[b]) = \boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b] \in \mathbb{R}^{b} \tag{10}$$

Error surface (e.g., squared error or Mean Squared Error (MSE); recall (6))

$$\begin{aligned} E(\widehat{\boldsymbol{\theta}}[b]) &:= E(\widehat{\boldsymbol{\theta}}[b], \boldsymbol{y}[b], \boldsymbol{X}[b]) \\ &:= \tfrac{1}{2}\|\boldsymbol{e}[b]\|^2 = \tfrac{1}{2}\boldsymbol{e}[b]^\top \boldsymbol{e}[b] = \tfrac{1}{2}\big(\boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\big)^\top \big(\boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\big) \\ &= \tfrac{1}{2}\Big(\boldsymbol{y}[b]^\top \boldsymbol{y}[b] - 2\boldsymbol{y}[b]^\top \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b] + \widehat{\boldsymbol{\theta}}[b]^\top \boldsymbol{X}[b]^\top \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\Big) \end{aligned} \tag{11}$$

Goal: Solve optimization problem to find minimizer (recall (7))

$$\widehat{\boldsymbol{\theta}}^\star := \arg\min_{\widehat{\boldsymbol{\theta}}} E(\widehat{\boldsymbol{\theta}}) \tag{12}$$

System identification by Least Squares: Algorithm steps

5. Step: Evaluating necessary and sufficient condition

Evaluating necessary condition per batch b yields (recall (8)):

$$
\begin{aligned}
\mathbf{0}_{\hat{p}} \overset{!}{=} \frac{\mathrm{d}E(\widehat{\boldsymbol{\theta}}[b])}{\mathrm{d}\widehat{\boldsymbol{\theta}}[b]} &= -\frac{\mathrm{d}(\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b])^\top}{\mathrm{d}\widehat{\boldsymbol{\theta}}[b]}\boldsymbol{y}[b] + \frac{1}{2}\left[\frac{\mathrm{d}(\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b])^\top}{\mathrm{d}\widehat{\boldsymbol{\theta}}[b]}\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b] + \widehat{\boldsymbol{\theta}}[b]^\top\boldsymbol{X}[b]^\top\frac{\mathrm{d}(\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b])}{\mathrm{d}\widehat{\boldsymbol{\theta}}[b]^\top}\right] \\
&= -\frac{\mathrm{d}(\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b])^\top}{\mathrm{d}\widehat{\boldsymbol{\theta}}[b]}\Big(\boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\Big) = -\boldsymbol{X}[b]^\top\Big(\boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\Big) \\
\Longrightarrow \qquad \mathbf{0}_{\hat{p}} &= -\boldsymbol{X}[b]^\top\boldsymbol{y}[b] + \boldsymbol{X}[b]^\top\boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b] \Longleftrightarrow \boxed{\widehat{\boldsymbol{\theta}}[b] = \big(\boldsymbol{X}[b]^\top\boldsymbol{X}[b]\big)^{-1}\boldsymbol{X}[b]^\top\boldsymbol{y}[b]}
\end{aligned}
\tag{13}
$$

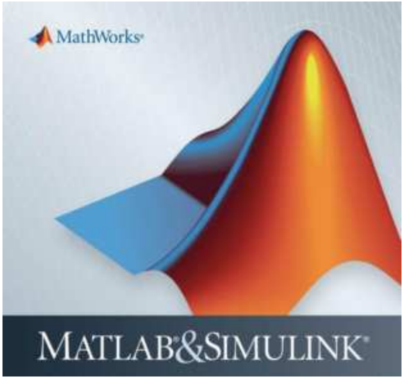
Evaluating sufficient condition per batch b yields (recall (9)):

$$
\frac{\mathrm{d}}{\mathrm{d}\widehat{\boldsymbol{\theta}}^\top}\frac{\mathrm{d}E(\widehat{\boldsymbol{\theta}})}{\mathrm{d}\widehat{\boldsymbol{\theta}}} = \frac{\mathrm{d}}{\mathrm{d}\widehat{\boldsymbol{\theta}}^\top}\left[-\boldsymbol{X}[b]^\top\Big(\boldsymbol{y}[b] - \boldsymbol{X}[b]\widehat{\boldsymbol{\theta}}[b]\Big)\right] = \boxed{\boldsymbol{X}[b]^\top\boldsymbol{X}[b] \overset{?}{>} 0}
\tag{14}
$$

Outline

System identification by Least Squares: Tutorial

Implement Least Squares method for system identifcation of RL circuit with constant back-emf



Tutorial:

- Download provided ZIP file from `https://www.dropbox.com/scl/fi/7cpnqxr568lqgroljlyb3/SystemIdentification_RL.zip?rlkey=24ayi27uacpb3kslwy1la7ve2&dl=0`
- In Matlab open m-file `SystemIdentification_RL_LeastSquares_Template.m`
- Implement *Least Squares* algorithm, i.e. (13)
- Extract the estimated physical parameters $\widehat{R}$, $\widehat{L}$ and $\widehat{u}_{\mathrm{emf}}$ from the estimated generic system parameters $\widehat{\theta}_1$, $\widehat{\theta}_2$ and $\widehat{\theta}_3$
- Play around with batch size b (when do you obtain good/best estimation results?)
- What happens when you use the noisy data?
- **Homework:** Implement *Recursive Least Squares* algorithm

Outline

4 Conclusion

Conclusion

To take home

System identification is a (very) **powerful tool** (but one needs to understand fundamentals):

- **Key problems** are
  - finding proper approximation function (including function type and parametrization)
  - solving the underlying optimization problem (avoiding reaching local minima, plateaus, etc.)
- **Applications** are widespread (almost anywhere useful)
  - **offline estimation** almost always feasible
  - **online estimation** not always feasible (e.g., depends on identification method)
- **User skills and hardware requirements** are
  - **experience** (data set generation, selection of topology, approximation functions, etc.)
  - **physical understanding** (imho good models are crucial for proper (i.e., grey box) model design)
  - **computational power** (in particular for matrix inversion studies; online capability, e.g. for recursive least squares (see home work))
  - **high storage capabilities** (at least for a large amount of measured)
- ... next lecture series will present other identification approaches.

Conclusion

Feedback on lecture series

... see interactive PPTX slides!

References I

[1] Dierk Schröder and Martin Buss. *Intelligente Verfahren: Identifikation und Regelung nichtlinearer Systeme*. Springer-Verlag, Berlin, 2017.

[2] Charu C. Aggarwal. *Neural Networks and Deep Learning: A Textbook*. Springer International Publishing, Cham, 2018.