Modeling and Estimation for Control

Lessons Handout

Lesson	Topic
1	**Introduction to Modeling**
	Systems and models, examples of models, models for systems and signals.
	PHYSICAL MODELING
2	**Principles of Physical Modeling**
	The phases of modeling, the mining ventilation problem example, structuring the problem, setting up the basic equations, forming the state-space models, simplified models.
3	**Some Basic Relationships in Physics**
	Electrical circuits, mechanical translation, mechanical rotation, flow systems, thermal systems, some observations.
4	**Bond Graphs:**
	Physical domains and power conjugate variables, physical model structure and bond graphs, energy storage and physical state, free energy dissipation, ideal transformations and gyrations, ideal sources, Kirchhoff's laws, junctions and the network structure, bond graph modeling of electrical networks, bond graph modeling of mechanical systems, examples.
	SIMULATION
5	**Computer-Aided Modeling**
	Computer algebra and its applications to modeling, analytical solutions, algebraic modeling, automatic translation of bond graphs to equations, numerical methods - a short glance.
6	**Modeling and Simulation in Scilab**
	Types of models and simulation tools for: ordinary differential equations, boundary value problems, difference equations, differential algebraic equations, hybrid systems.
	SYSTEM IDENTIFICATION
7	**Experiment Design for System Identification:**
	Basics of system identification, from continuous dynamics to sampled signals, disturbance modeling, signal spectra, choice of sampling interval and presampling filters.
8	**Non-parametric Identification:**
	Transient-response and correlation analysis, frequency-response/Fourier/spectral analysis, estimating the disturbance spectrum.
9	**Parameter Estimation in Linear Models:**
	Linear models, basic principle of parameter estimation, minimizing prediction errors, linear regressions and least squares, properties of prediction error minimization estimates.
10	**System Identification Principles and Model Validation**
	Experiments and data collection, informative experiments, input design for open-loop experiments, identification in closed-loop, choice of the model structure, model validation, residual analysis.
11	**Nonlinear Black-box Identification**
	Nonlinear state-space models, nonlinear black-box models: basic principles, parameters estimation with Gauss-Newton stochastic gradient algorithm, temperature profile identification in tokamak plasmas
	TOWARDS PROCESS SUPERVISION
12	**Recursive Estimation Methods**
	Recursive least-squares algorithm, IV method, prediction-error methods and pseudolinear regressions, Choice of updating step

Dr. Emmanuel WITRANT

Control-oriented modeling and system identification

Outline

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

Course goal

To teach systematic methods for building mathematical models of dynamical systems based on physical principles and measured data.

Main objectives:

- build mathematical models of technical systems from first principles
- use the most powerful tools for modeling and simulation
- construct mathematical models from measured data

Class overview

1	Introduction to modeling
	Physical Modeling
2	Principles of physical modeling
3	Some basic relationships in physics
4	Bond graphs
	Simulation
5	Computer-aided modeling
6	Modeling and simulation in Scilab
	System Identification
7	Experiment design for system identification
8	Non-parametric identification
9	Parameter estimation in linear models
10	System identification principles and model validation
11	Nonlinear black-box identification
	Towards process supervision
12	Recursive estimation methods

Grading policy

- Homeworks: 30 %, each due at the beginning of the next class. You can interact to find the solution but each homework has to be unique! otherwise, 0 FOR BOTH identical copies
- Final Exam: 70 %

Material

- Lecture notes from 2E1282 *Modeling of Dynamical Systems, Automatic Control*, School of Electrical Engineering, KTH, Sweden.
- L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
- S. Campbell, J-P. Chancelier and R. Nikoukhah, *Modeling and Simulation in Scilab/Scicos*, Springer, 2005.
- S. Stramigioli, *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-free Approach*, Springer, LNCIS 266, 2001.
- L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.

Class website

- Go to:
`http://www.gipsa-lab.fr/MiSCIT/courses/courses_MME.php`
or Google “MiSCIT” then go to “Courses”, “Modeling” and “Modeling and system identification”
- at the bottom of the page, click “Restricted access area” and enter with:
    - login: MiSCIT_student
    - password: ******

Modeling and estimation for control

Lecture 1: Introduction to modeling

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

September 4, 2014.

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

Systems and Models

Systems and experiments

- *System:* object or collection of objects we want to study.
- *Experiment:* investigate the system properties / verify theoretical results, BUT
  - too expensive, i.e. one day operation on Tore Supra;
  - too dangerous, i.e. nuclear plant;
  - system does not exist, i.e. wings in airplane design.

$\Rightarrow$ Need for *models*

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

What is a model?

- Tool to answer questions about the process without experiment / action-reaction.
- Different classes:
  1. *Mental models:* intuition and experience (i.e. car driving, industrial process in operator's mind);
  2. *Verbal models:* behavior in different conditions described by words (e.g. If ... then ...);
  3. *Physical models:* try to imitate the system (i.e. house esthetic or boat hydrodynamics);
  4. *Mathematical models:* relationship between observed quantities described as mathematical relationships (i.e. most law in nature).
     Generally described by *differential algebraic equations*:

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t), d(t))\\ 0 &= g(x(t), u(t), d(t))\end{aligned}$$

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

Models and simulation

- models $\rightarrow$ used to calculate or decide how the system would have reacted (analytically);
- *Simulation:* numerical experiment = inexpensive and safe way to experiment with the system;
- simulation value depends completely on the model quality.

How to build models?

- Two sources of knowledge:
  - *collected experience:* laws of nature, generations of scientists, literature;
  - from the *system itself:* observation.
- Two areas of knowledge:
  - *domain of expertise:* understanding the application and mastering the relevant facts $\rightarrow$ mathematical model;
  - *knowledge engineer:* practice in a usable and explicit model $\rightarrow$ knowledge-based model.

- Two different principles for model construction:
  - *physical modeling:* break the system into subsystems described by laws of nature or generally recognized relationships;
  - *identification:* observation to fit the model properties to those of the system (often used as a complement).

How to verify models?

- Need for confidence in the results and prediction, obtained by *verifying* or *validating* the model: model vs. system.
- Domain of validity: qualitative statements (most verbal models), quantitative predictions. Limited for all models.

⇒ Hazardous to model outside the validated area.

⇒ Models and simulations can never replace observations and experiments - but they constitute an important and useful complement.

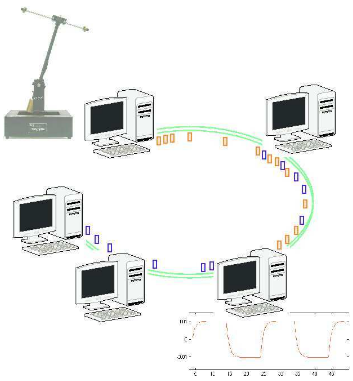
Different types of mathematical models

- Deterministic - Stochastic: exact relationships vs. stochastic variables/processes;
- Static - Dynamic: direct, instantaneous link (algebraic relationships) vs. depend also on earlier applied signals (differential/difference equations);
- Continuous - Discrete time: differential equation vs. sampled signal;
- Distributed - Lumped: events dispersed over the space (distributed parameter model → partial differential equation PDE) vs. finite number of changing variables (ordinary diff. eqn. ODE);
- Change oriented - Discrete event driven: continuous changes (Newtonian paradigm) vs. (random) event-based influences (i.e. manufacture, buffer. . .)

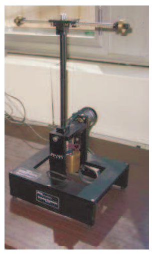
Examples of Models

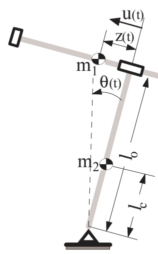
Networked control of the inverted pendulum [Springer'07]

- Objective: test control laws for control over networks.



- Physical model and abstraction:





- Mathematical model
  - from physics:

$$\begin{bmatrix} m_1 & m_1 l_0 \\ m_1 l_0 & \bar{J} + m_1 z^2 \end{bmatrix} \begin{bmatrix} \ddot{z} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & -m_1 z \dot{\theta} \\ 2 m_1 z \dot{\theta} & 0 \end{bmatrix} \begin{bmatrix} \dot{z} \\ \dot{\theta} \end{bmatrix}$$
$$+ \begin{bmatrix} -m_1 \sin\theta \\ -(m_1 l_0 + m_2 l_c)\sin\theta - m_1 z \cos\theta \end{bmatrix} g = \begin{bmatrix} 1 \\ 0 \end{bmatrix} u,$$

- input/ouput representation ($x \doteq [z,\ \dot{z},\ \theta,\ \dot{\theta}]^T$):

$$
\begin{cases}
\dot{x}_1 & = & x_2, \\
\dot{x}_2 & = & \dfrac{u}{m_1} - l_0 \dot{x}_4 + x_1 x_4^2 + g \sin(x_3), \\
\dot{x}_3 & = & x_4, \\
\dot{x}_4 & = & \dfrac{1}{J_0(x_1) - m_1 l_0^2} \left[g \left(m_2 l_c \sin(x_3) + m_1 x_1 \cos(x_3) \right) \right. \\
 & & \left. - m_1 \left(l_0 x_4 + 2 x_2 \right) x_1 x_4 + - l_0\, u \right], \\
J_0(x_1) & = & \bar{J} + m_1 x_1^2, \\
y & = & \{x_1,\ x_2\}
\end{cases}
$$

- Exercise: derive this state-space representation

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

- Fluid-flow model for the network [Misra et al. 2000, Hollot and Chait 2001]: TCP with proportional active queue management (AQM) set the window size W and queue length q variations as

$$
\begin{aligned}
\frac{dW_i(t)}{dt} &= \frac{1}{R_i(t)} - \frac{W_i(t)}{2}\frac{W_i(t - R_i(t))}{R_i(t - R_i(t))} p_i(t), \\
\frac{dq(t)}{dt} &= -C_r + \sum_{i=1}^{N} \frac{W_i(t)}{R_i(t)}, \quad q(t_0) = q_0,
\end{aligned}
$$

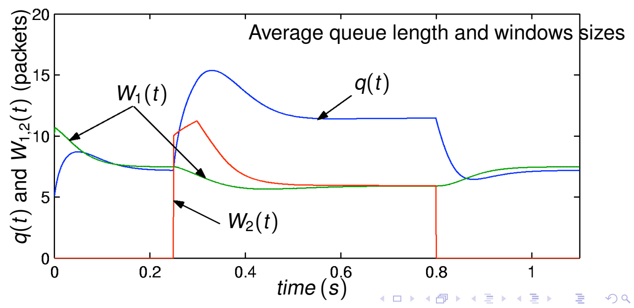
where $R_i(t) \doteq \dfrac{q(t)}{C_r} + T_{pi}$ is the round trip time, C_r the link capacity, $p_i(t) = K_p q(t - R_i(t))$ the packet discard function and T_{pi} the constant propagation delay. The average time-delay is $\tau_i = \frac{1}{2} R_i(t)$

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

E.g. network with 2 TCP flows:

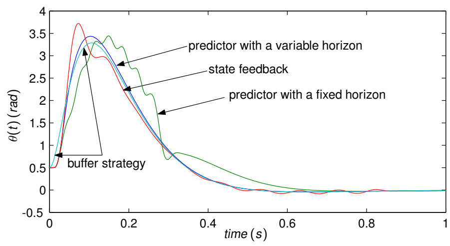
$$
\begin{aligned}
\frac{dW_{1,2}(t)}{dt} &= \frac{1}{R_{1,2}(t)} - \frac{W_{1,2}(t)}{2}\frac{W_{1,2}(t - R_{1,2}(t))}{R_{1,2}(t - R_{1,2}(t))} p_{1,2}(t) \\
\frac{dq(t)}{dt} &= -300 + \sum_{i=1}^{2} \frac{W_i(t)}{R_i(t)}, \quad q(0) = 5 \\
\tau(t) &= R_1(t)/2
\end{aligned}
$$

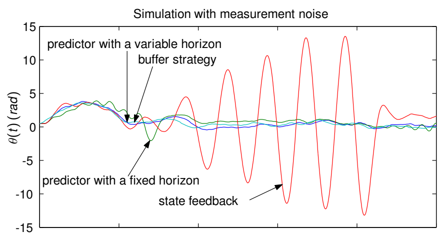
Behavior of the network internal states.



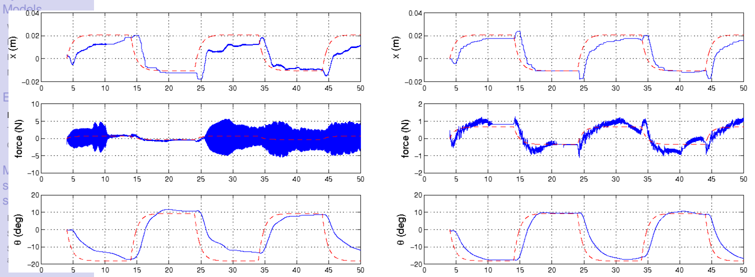
Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

- Compare different control laws: in simulation





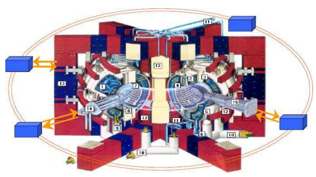
On the inverted pendulum experiment:



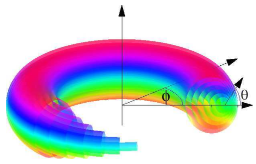
(a) Predictive control with fixed horizon.

(b) Predictive control with time-varying horizon.

Thermonuclear Fusion with Tore Supra tokamak



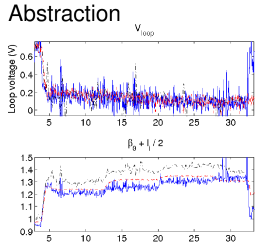
Physical model



Abstraction

$$\frac{\partial \psi}{\partial t} = \eta_{/\!/}(x,t)\left[\frac{1}{\mu_0 a^2}\frac{\partial^2 \psi}{\partial x^2} + \frac{1}{\mu_0 a^2 x}\frac{\partial \psi}{\partial x}\right.$$
$$\left. + R_0 j_{bs}(x,t) + R_0 j_{ni}(x,t)\right]$$
$$j_\phi(x,t) = -\frac{1}{\mu_0 R_0 a^2 x}\frac{\partial}{\partial x}\left[x\frac{\partial \psi}{\partial x}\right]$$

Mathematical model [PPCF 2007]



Experimental results

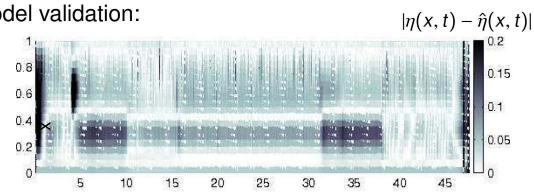
Identification of temperature profiles [CDC 2011]

- Parameter-dependant first-order dynamics:

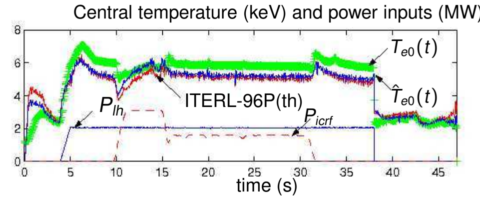
$$\left\{\begin{array}{rcl} \tau_{th}(t) & = & e^{\vartheta_{t0}} I_p^{\vartheta_{t1}} B_{\phi 0}^{\vartheta_{t2}} \bar{n}_e^{\vartheta_{t3}} P_{tot}^{\vartheta_{t4}} \\ \dfrac{dW}{dt} & = & P_{tot} - \dfrac{1}{\tau_{th}} W, \quad W(0) = P_{tot}(0)\tau_{th}(0) \\ \hat{T}_{e0}(t) & = & \mathcal{A} W \end{array}\right.$$

$\rightarrow$ "free" parameters ϑ_i determined from a *sufficiently rich* set of experimental data.

- Model validation:



Central temperature (keV) and power inputs (MW)



Comparison of the model with a shot not included in the database.

Conclusion: all models are approximate!

- A model captures only some aspects of a system:
  - Important to know which aspects are modelled and which are not;
  - Make sure that model is valid for intended purpose;
  - "If the map does not agree with reality, trust reality".
- All-encompasing models often a bad idea:
  - Large and complex hard to gain insight;
  - Cumbersome and slow to manipulate.
- Good models are simple, yet capture the essentials!

Models for Systems and Signals

Types of models

- System models (differential / difference equations) and signal models (external signals / disturbances).
- Block diagram models: *logical* decomposition of the functions and mutual influences (interactions, information flows), not unique. Related to verbal models.
- Simulation models: related to program languages.

Input, output and disturbance signals

- Constants (system or design parameters) vs. variables or signals;
- Outputs: signals whose behavior is our primary interest, typically denoted by $y_1(t),\ y_2(t),\ \dots,\ y_p(t)$.
- External signals: signals and variables that influence other variables in the system but are not influenced by the system:
  - *input* or *control signal:* we can use it to influence the system $u_1(t),\ u_2(t),\ \dots,\ u_m(t)$;
  - *disturbances:* we cannot influence or choose $w_1(t),\ w_2(t),\ \dots,\ w_r(t)$.
- Internal variables: other model variables.

Differential equations

- Either directly relate inputs u to outputs y:
$$g(y^{(n)}(t),\ y^{(n-1)}(t),\ \dots,\ y(t),\ u^{(m)}(t),\ u^{(m-1)}(t),\ \dots,\ u(t)) = 0$$
where $y^{(k)}(t) = d^k y(t)/dt^k$ and $g(\cdot)$ is an arbitrary, vector-valued, nonlinear function.
- or introduce a number of internal variables related by first order DE
$$\dot{x}(t) = f(x(t),\ u(t))$$
with x, f and u are vector-valued, nonlinear functions, i.e.
$$\begin{aligned}\dot{x}_1(t) &= f_1(x_1(t), \dots, x_n(t),\ u_1(t), \dots, u_m(t))\\ \dot{x}_2(t) &= f_2(x_1(t), \dots, x_n(t),\ u_1(t), \dots, u_m(t))\\ &\vdots\\ \dot{x}_n(t) &= f_n(x_1(t), \dots, x_n(t),\ u_1(t), \dots, u_m(t))\end{aligned}$$

The outputs are then calculated from $x_i(t)$ and $u_i(t)$ from:

$$y(t) = h(x(t), u(t))$$

- Corresponding discrete time equations:

$$\begin{aligned} x(t+1) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned}$$

The concept of state and state-space models

Definitions:

- State at t_0: with this information and $u(t)$, $t \geq t_0$, we can compute $y(t)$.
- State: information that has to be stored and updated during the simulation in order to calculate the output.
- State-space model (continuous time):

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned}$$

$u(t)$: input, an m-dimensional column vector
$y(t)$: output, a p-dimensional column vector
$x(t)$: state, an n-dimensional column vector
$\rightarrow n^{th}$ order model, unique solution if $f(x, u)$ continuously differentiable, $u(t)$ piecewise continuous and $x(t_0) = x_0$ exists.

- *State-space model (discrete time:)*

$$\begin{aligned} x(t_{k+1}) &= f(x(t_k), u(t_k)), \quad k = 0, 1, 2, \ldots \\ y(t_k) &= h(x(t_k), u(t_k)) \end{aligned}$$

where $u(t_k) \in \mathbb{R}^m$, $y(t_k) \in \mathbb{R}^p$, $x(t_k) \in \mathbb{R}^n$.
$\rightarrow n^{th}$ order model, unique solution if the initial value $x(t_0) = x_0$ exists.

Linear models:

- if $f(x, u)$ and $h(x, u)$ are linear functions of x and u:

$$\begin{aligned} f(x, u) &= Ax + Bu \\ h(x, u) &= Cx + Du \end{aligned}$$

with $A : n \times n$, $B : n \times m$, $C : p \times n$ and $D : p \times m$.
- if the matrices are independent of time, the system is linear and time-invariant.

Stationary solutions, static relationships and linearization

Stationary points: Given a system

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned}$$

a solution (x_0, u_0) such that $0 = f(x_0, u_0)$ is called a stationary point (singular point or equilibrium).
At a stationary point, the system is at rest: $x(0) = x_0$, $u(t) = u_0$ for $t \geq 0 \Rightarrow x(t) = x_0$ for all $t \geq 0$.
Stability: suppose that $x(t_0) = x_0$ gives a stationary solution, what happens for $x(t_0) = x_1$? The system is

- asymptotically stable if any solution $x(t)$ close enough to x_0 converges to x_0 as $t \rightarrow \infty$;
- globally asymptotically stable if *all* solutions $x(t)$ with $u(t) = u_0$ converge to x_0 as $t \rightarrow \infty$.

Static relationships:

- for asymptotically stable stationary point (x_0, u_0), the output converges to $y_0 = h(x_0, u_0)$. Since x_0 depends implicitly on u_0,

$$y_0 = h(x(u_0), u_0) = g(u_0)$$

  Here, $g(u_0)$ describes the *stationary relation* between u_0 and y_0.
- Consider a small change in the input level from u_0 to $u_1 = u_0 + \delta u_0$, the stationary output will be

$$y_1 = g(u_1) = g(u_0 + \delta u_0) \approx g(u_0) + g'(u_0)\delta u_0 = y_0 + g'(u_0)\delta u_0.$$

  Here $g'(u_0) : p \times m$ describes how the stationary output varies locally with the input → *static gain*.

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

Linearization:

- system behavior in the neighborhood of a stationary solution (x_0, u_0);
- consider small deviations $\Delta x(t) = x(t) - x_0$, $\Delta u(t) = u(t) - u_0$ and $\Delta y(t) = y(t) - y_0$, then

$$\begin{aligned} \dot{\Delta x} &= A\Delta x + B\Delta u \\ \Delta y &= C\Delta x + D\Delta u \end{aligned}$$

  where A, B, C and D are partial derivative matrices of $f(x(t), u(t))$ and $h(x(t), u(t))$, i.e.

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_0, u_0) & \dots & \frac{\partial f_1}{\partial x_n}(x_0, u_0) \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1}(x_0, u_0) & \dots & \frac{\partial f_n}{\partial x_n}(x_0, u_0) \end{bmatrix};$$

- Exercise: prove it

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

- important and useful tool <u>but</u>
  - only for local properties;
  - quantitative accuracy difficult to estimate → complement with simulations of the original nonlinear system.

Modeling and estimation for control
E. Witrant
Systems and Models
What is a model?
How to build models?
How to verify models?
Mathematical models
Examples
Inverted pendulum
Tore Supra
Conclusions
Models for systems and signals
Differential equations
State-space models
Stationary, stability and linearization
Conclusions
Homework

Example

From lecture notes by K.J. Åström, LTH

Model of bicycle dynamics:

$$\frac{d^2\theta}{dt^2} = \frac{mgl}{J_p}\sin\theta + \frac{mlV_0^2\cos\theta}{bJ_p}\left(\tan\beta + \frac{a}{V_0\cos^2\beta}\frac{d\beta}{dt}\right)$$

where θ is the vertical tilt and β is front wheel angle (control).
⇒ Hard to gain insight from nonlinear model...

Linearized dynamics (around $\theta = \beta = \dot{\beta} = 0$):

$$\frac{d^2\theta}{dt^2} = \frac{mgl}{J_p}\theta + \frac{mlV_0^2}{bJ_p}\left(\beta + \frac{a}{V_0}\frac{d\beta}{dt}\right)$$

has transfer function

$$G(s) = \frac{mlV_0^2}{bJ_p} \times \frac{1 + \frac{a}{V_0}s}{s^2 - \frac{mgl}{J_p}}.$$

Gain proportional to V_0^2:

- more control authority at high speeds.

Unstable pole at $\sqrt{\frac{mgl}{J_p}} \approx \sqrt{g/l}$:

- slower when l is large;
- easier to ride a full size bike than a childrens bike.

Conclusions

- Classes of models
- Preliminary questions according to your goal and main process behavior
- Some background on dynamical systems

Homework 1

Consider the inverted pendulum dynamics:

$$
\begin{cases}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= \dfrac{u}{m_1} - l_0\dot{x}_4 + x_1 x_4^2 + g\sin(x_3), \\
\dot{x}_3 &= x_4, \\
\dot{x}_4 &= \dfrac{1}{J_0(x_1) - m_1 l_0^2}\left[g\left(m_2 l_c \sin(x_3) + m_1 x_1 \cos(x_3)\right)\right. \\
& \quad \left. -m_1\left(l_0 x_4 + 2x_2\right) x_1 x_4 + -l_0\, u\right], \\
J_0(x_1) &= \bar{J} + m_1 x_1^2, \\
y &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
\end{cases}
$$

where

Parameter name	Value	Meaning
m_1	0.213 kg	Mass of the horizontal rod.
m_2	1.785 kg	Mass of the vertical rod.
l_0	0.33 m	Length of the vertical rod.
l_c	−0.029 m	Vertical rod c.g. position.
g	9.807 $\frac{m}{s^2}$	Gravity acceleration.
$\bar{J}$	0.055 Nm^2	Nominal momentum of inertia.

Analyze the system dynamics by:

1. defining the set of equilibrium points;
2. linearizing the proposed model at a "zero input force" equilibrium;
3. writing the transfer function: analytically from the initial (second order) physical equations and numerically from the state-space model;
4. interpreting the resulting equations.

References

1. L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
2. V. Misra, W.-B. Gong, and D. Towsley, Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED, SIGCOMM 2000.
3. C. Hollot and Y. Chait, Nonlinear stability analysis for a class of TCP/AQM networks, CDC 2001.
4. EW, D. Georges, C. Canudas de Wit and M. Alamir, On the use of State Predictors in Networked Control Systems, LNCS Springer, 2007.
5. EW, E. Joffrin, S. Brémond, G. Giruzzi, D. Mazon, O. Barana et P. Moreau, A control-oriented model of the current profile in Tokamak plasma, IOP PPCF 2007.
6. EW and S. Brémond, Shape Identification for Distributed Parameter Systems and Temperature Profiles in Tokamaks, CDC 2011.

Université Joseph Fourier GRENOBLE — UFR de Physique

Modeling and Estimation for Control
Physical Modeling

Lecture 2: Principles of physical modeling

Emmanuel WITRANT
emmanuel.witrant@univ-grenoble-alpes.fr

September 5, 2017.

The Three Phases of Modeling

"Successful modeling is based as much on a good feeling for the problem and common sense as on the formal aspects that can be taught"

1. Structuring the problem

- divide the system into subsystems, determine causes and effects, important variables and interactions;
- intended use of the model?
- results in block diagram or similar description;
- needs understanding and intuition;
- where complexity and degree of approximation are determined.

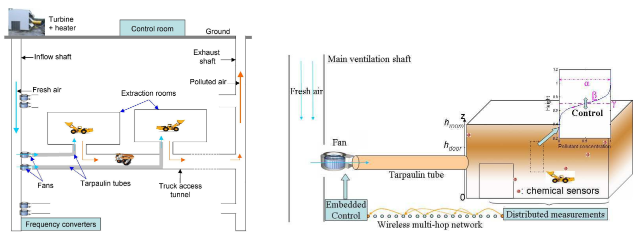
2. Setting up the Basic Equations

- "fill in" the blocks using the laws of nature and basic physical equations;
- introduce approximations and idealizations to avoid too complicated expressions;
- lack of basic equations → new hypotheses and innovative thinking.

3. Forming the State-Space Models

- formal step aiming at suitable organization of the equations/relationships;
- provides a suitable model for analysis and simulation;
- computer algebra can be helpful;
- for simulation: state-space models for subsystems along with interconnections.

Example: the Mining Ventilation Problem [IJRNC'11]



Objectives:

- propose a new automation strategy to minimize the fans energy consumption, based on distributed sensing capabilities: wireless sensor network;
- investigate design issues and the influence of sensors location;
- find the optimal control strategy that satisfies safety constraints.

Outline

Phase 1: Structuring the problem

Ask the good questions:

- What signals are of interest (outputs)?
- Which quantities are important to describe what happens in the system?
- Of these quantities, which are exogenous and which should be regarded as internal variables?
- What quantities are approximately time invariant and should be regarded as constants?
- What variables affect certain other variables?
- Which relationships are static and which are dynamic?

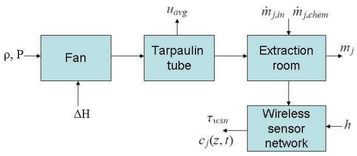
General tips:

- often need experimental results to assist these steps (i.e. time constants and influences);
- the intended use determines the complexity;
- use model to get insights, and insights to correct the model;
- work with several models in parallel, that can have different complexity and be used to answer different questions;
- for complex systems, first divide the system into subsystems, and the subsystems into blocs.

Example: for the mining ventilation problem

- Inputs to the system:
  - ρ: air density in vertical shaft;
  - P: air pressure in vertical shaft;
  - ΔH: variation of pressure produced by the fan;
  - $\dot{m}_{j,in}$: incoming pollutant mass rate due to the engines;
  - $\dot{m}_{j,chem}$: mass variation due to chemical reactions between components;
  - h: time-varying number of hops in WSN.
- Outputs from the system:
  - $c_j(z,t)$ pollutants (CO_x or NO_x) volume concentration profiles, where $z \in [0; h_{room}]$ is the height in the extraction room;
  - u_{avg} is the average velocity of the fluid in the tarpaulin tube;
  - m_j pollutant mass in the room;
  - τ_{wsn} delay due to the distributed measurements and wireless transmission between the extraction room and the fan.

- Division into subsystems:
  - fan / tarpaulin tube / extraction room / wireless sensor network.
- Corresponding block diagram:



Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Phase 2: Setting up the Basic Equations

Main principles:

- formulate quantitative I/O relationships;
- use knowledge of mechanics, physics, economics, . . .
- well-established laws, experimental curves (data sheets) or crude approximations;
- Highly problem dependent!

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Two groups of relationships:

1. Conservation laws: relate quantities of the same kind, i.e.
   - $P_{in} - P_{out}$ = stored energy / unit time;
   - inflow rate - outflow rate = stored volume / t;
   - input mass flow rate - output mass flow rate = stored mass / t;
   - nodes and loops from Kirchhoff's laws.
2. Constitutive relationships: relate quantities of different kinds (i.e. voltage - current, level - outflow, pressure drop - flow)
   - material, component or bloc in the system;
   - static relationships;
   - relate physical to engineering relationships;
   - always approximate.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

How to proceed?

- write down the conservation laws for the block/subsystem;
- use suitable constitutive relationships to express the conservation laws in the model variables. Calculate the dimensions as a check.

Mining ventilation example (i.e. extraction room):

- Conservation law - conservation of mass for chemical species j:

$$\dot{m}_j(t) = \dot{m}_{j,in}(t) - \dot{m}_{j,out}(t) - \dot{m}_{j,chem}(t)$$

- Constitutive relationship - relate the mass to concentration profile:

$$\begin{aligned} m_j(t) &= S_{room} \int_0^{h_{room}} c_j(z,t)dz \\ &= S_{room} \left[\int_0^{h_{door}} c_j(z,t)dz + \alpha_j(t)\Delta h \right], \end{aligned}$$

and hypothesis on the shape (e.g. sigmoid):

$$c_j(z,t) = \frac{\alpha_j(t)}{1 + e^{-\beta_j(t)(z-\gamma_j(t))}}.$$

Phase 3: Forming the State-Space Models

Straightforward recipe:

1. choose a set of state variables (memory of what has happened, i.e. storage variables);
2. express the time derivative of each state as a function of states and inputs;
3. express the outputs as functions of the state and inputs.

Examples of stored quantities:

- position of a mass / tank level (stored potential energy);
- velocity of a mass (stored kinetic energy);
- charge of capacitor (stored electrical field energy);
- current through inductor (stored magnetic energy);
- temperature (stored thermal energy);
- internal variables from step 2.

Make separate models

for the subsystems and diagram interconnections → modularity and error modeling diagnostic.

Extraction room model:

- Shape parameters α, β and γ chosen as the state: $x(t) = [\alpha, \beta, \gamma]^T$;
- Time derivative from mass conservation:

$$E_j \begin{bmatrix} \dot{\alpha}_j(t) \\ \dot{\beta}_j(t) \\ \dot{\gamma}_j(t) \end{bmatrix} = \dot{m}_{j,in}(t) - B_j\, u_{fan}(t - \tau_{tarp}) - D_{jk}, \text{ with}$$

$$\begin{aligned} E_j &\doteq S_{room} \left(V_{int} \begin{bmatrix} \vdots & \vdots & \vdots \\ \frac{\partial C_{j,i}}{\partial \alpha_j} & \frac{\partial C_{j,i}}{\partial \beta_j} & \frac{\partial C_{j,i}}{\partial \gamma_j} \\ \vdots & \vdots & \vdots \end{bmatrix} + \begin{bmatrix} \Delta h \\ 0 \\ 0 \end{bmatrix}^T \right) \\ B_j &\doteq \frac{1}{h_{door}} V_{int} \begin{bmatrix} \vdots \\ C_{j,i} \\ \vdots \end{bmatrix} \times S_{tarp}\, \nu,\ D_{jk} = S_{room} \left[V_{int} \begin{bmatrix} \vdots \\ \eta_{jk,i}\, C_{j,i}\, C_{k,i} \\ \vdots \end{bmatrix} + \eta_{jk} \alpha_j \alpha_k \Delta h \right] \end{aligned}$$

Number of state variables:

- sufficient if derivatives described by state and inputs;
- harder to determine unnecessary states;
- linear models → rank of matrices;
- when used in simulation, the only disadvantage is related to unnessary computations.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Simplified models

Even if a relatively good level of precision can be achieved, the model has to be manageable for our purpose.

Model simplification:

- reduced number of variables;
- easily computable;
- linear rather than nonlinear;
- tradeoff between complexity and accuracy;
- balance between the approximations;
- three kinds:
  1. small effects are neglected - approximate relationships are used;
  2. separation of time constants;
  3. aggregation of state variables.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Small effects are neglected - approximate relationships are used:

- i.e. compressibility, friction, air drag → amplitude of the resonance effects / energy losses?
- based on physical intuition and insights together with practice;
- depends on the desired accuracy;
- linear vs. nonlinear: make experiments and tabulate the results.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Separation of time constants:

- May have different orders of magnitude, i.e. for Tokamaks:

Alfvén time (MHD instabilities)	10^{-6} s
density diffusion time	$0.1 - 1$ s
heat diffusion time	0.1s-1s (3.4 s for ITER)
resistive diffusion time	few seconds ($100 - 3000$ s for ITER)

- Advices:
  - concentrate on phenomena whose time constants match the intended use;
  - approximate subsystems that have considerably faster dynamics with static relationships;
  - variables of subsystems whose dynamics are appreciably slower are approximated as constants.
- Two important advantages:
  1. reduce model order by ignoring very fast and very slow dynamics;
  2. by giving the model time constants that are on the same order of magnitude (i.e. $\tau_{max}/\tau_{min} \leq 10 - 100$), we get simpler simulations (avoid *stiffness!*). E.g. $A = [0,\ 1; -1000\ -1001]$
- When different time-scales, use different models.

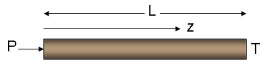
Aggregation of state variables:

To merge several similar variables into one state variable: often average or total value.

- i.e. infinite number of points in the extraction room → 3 shape parameters, trace gas transport in firns;
- hierarchy of models with different amount of aggregation, i.e. economics: investments / private and government / each sector of economy / thousand state variables;
- partial differential equations (PDE) reduced to ordinary differential equations (ODE) by difference approximation of spatial variables.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Example: Heat conduction in a rod



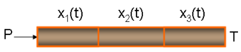
- input: power in the heat source P;
- output: temperature at the other endpoint T;
- heat equation: $\frac{\partial}{\partial t}x(z,t) = a\frac{\partial^2}{\partial z^2}x(z,t)$ where $x(z,t)$ is the temperature at time t at the distance z from the left end point and a is the heat conductivity coefficient of the metal;
- hypothesis: no losses to the environment;
- at the end points: $a\frac{\partial}{\partial z}x(0,t) = P(t), \quad x(L,t) = T(t)$
- requires to know the whole function $x(z, t_1)$, $0 \le z \le L$, to determine $T(t)$, $t \ge t_1$, → *infinite dimensional system.*

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Example: Heat conduction in a rod (2)

Aggregation of state variables: approximate for simulation

- divide the rode ($x(z,t)$, $0 \le z \le L/3$, aggregated into $x_1(t)$ etc.) and assume homogeneous temperature in each part



- conservation of energy for part 1:

$$\begin{aligned}
\frac{d}{dt}(\text{heat stored in part 1}) &= (\text{power in}) - (\text{power out to part 2}) \\
\frac{d}{dt}(C \cdot x_1(t)) &= P - K(x_1(t) - x_2(t))
\end{aligned}$$

C: heat capacity of each part, K: heat transfer

- similarly:

$$\begin{aligned}
\frac{d}{dt}(C \cdot x_2(t)) &= K(x_1(t) - x_2(t)) - K(x_2(t) - x_3(t)) \\
\frac{d}{dt}(C \cdot x_3(t)) &= K(x_2(t) - x_3(t)) \\
T(t) &= x_3(t)
\end{aligned}$$

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

- Rearrange the equations to obtain the linear state-space model:

$$\begin{aligned}
\dot{x}(t) &= \frac{K}{C}\begin{pmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{pmatrix} x + \frac{1}{C}\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} P \\
y(t) &= \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} x(t)
\end{aligned}$$

- Conclusions: essentially the same as using finite difference approximation on the space derivative (homework), a finer division would give a more accurate model.

Example: solving the air continuity in polar firns and ice cores [ACP'12]

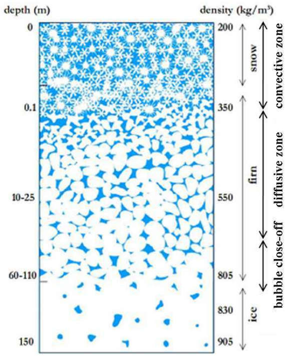
From poromechanics, firn = system composed of the ice lattice, gas connected to the surface (open pores) and gas trapped in bubbles (closed pores). Air transport is driven by:

$$\frac{\partial[\rho_{ice}(1-\epsilon)]}{\partial t} + \nabla[\rho_{ice}(1-\epsilon)\vec{v}] = 0$$

$$\frac{\partial[\rho^o_{gas} f]}{\partial t} + \nabla[\rho^o_{gas} f(\vec{v}+\vec{w}_{gas})] = -\vec{r}^{o\to c}$$

$$\frac{\partial[\rho^c_{gas}(\epsilon - f)]}{\partial t} + \nabla[\rho^c_{gas}(\epsilon - f)\vec{v}] = \vec{r}^{o\to c}$$

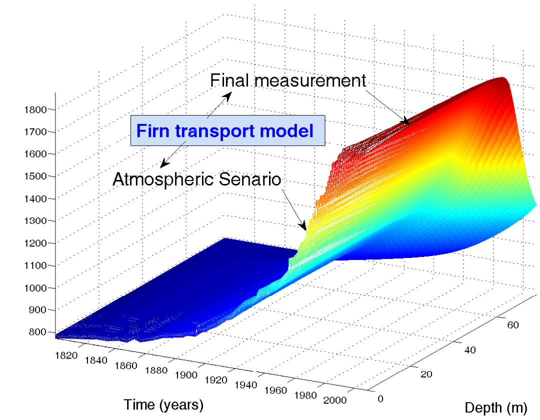
with appropriate boundary and initial conditions.



Scheme adapted from [Sowers et al.'92, Lourantou'08].

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

I.e. CH_4 transport at NEEM (Greenland)



$\Rightarrow$ Unique archive of the recent (50-100 years) anthropogenic impact. Can go much further (i.e. > 800 000 years) in ice.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

From distributed to lumped dynamics

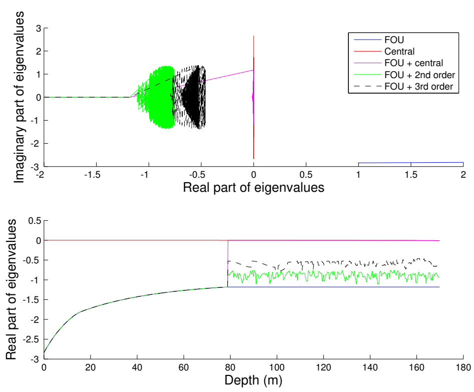
- Consider a quantity q transported in 1D by a flux $u = qv$ with a source term s ($t \in [0, T]$, $z \in [0, z_f]$):

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial z}[q\,v(z,t)] = s(z,t), \text{ with } \begin{cases} q(0,t) = 0 \\ q(x,0) = q_0(x) \end{cases}$$

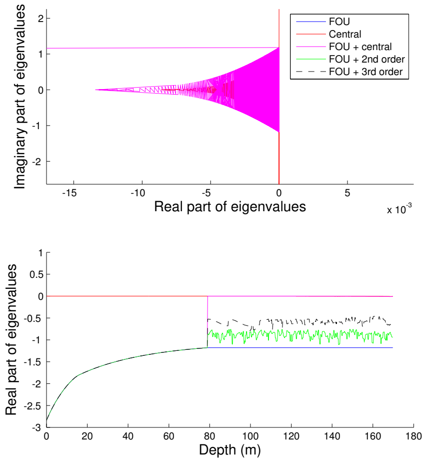
  where $s(z,t) \neq 0$ for $z < z_1 < z_f$ and $s = 0$ for $z_1 < z < z_f$.
- Approximate $\partial[qv]/\partial z$, i.e. on uniform mesh [Hirsch'07]:
  - backward difference: $(u_z)_i = \frac{u_i - u_{i-1}}{\Delta z} + \frac{\Delta z}{2}(u_{zz})_i$
  - central difference: $(u_z)_i = \frac{u_{i+1} - u_{i-1}}{2\Delta z_i} - \frac{\Delta z^2}{6}(u_{zzz})_i$
  - other second order: $(u_z)_i = \frac{u_{i+1} + 3u_i - 5u_{i-1} + u_{i-2}}{4\Delta z_i} + \frac{\Delta z^2}{12}(u_{zzz})_i - \frac{\Delta z^3}{8}(u_{zzzz})_i$
  - third order: $(u_z)_i = \frac{2u_{i+1} + 3u_i - 6u_{i-1} + u_{i-2}}{6\Delta z_i} - \frac{\Delta z^3}{12}(u_{zzzz})_i$
- Provides the computable lumped model: $dq/dt = Aq + s$
- The choice of the discretization scheme directly affects the definition of A and its eigenvalues distribution: need to check stability and precision!

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

E.g. stability: eigenvalues of A for CH_4 at NEEM with $dt \approx 1$ week



E.g. eig(A) for CH_4 at NEEM with $dt \approx 1$ week, zoom



Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Conclusions

→ Guidelines to structure the general approach for modeling

- The clarity of the model and its usage directly depends on its initial philosophy
- Prevent the temptation to avoid the documentation of “obvious steps”
- Forecasting the use of experimental knowledge and sub-model validation strategies during the modeling phases is essential

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

References

1. L. Ljung and T. Glad, Modeling of Dynamic Systems, Prentice Hall Information and System Sciences Series, 1994.
2. EW, A. D’Innocenzo, G. Sandou, F. Santucci, M. D. Di Benedetto, A. J. Isaksson, K. H. Johansson, S.-I. Niculescu, S. Olaru, E. Serra, S. Tennina and U. Tiberi, Wireless Ventilation Control for Large-Scale Systems: the Mining Industrial Case, International Journal of Robust and Nonlinear Control, vol. 20 (2), pp. 226 - 251, Jan. 2010.
3. EW, P. Martinerie, C. Hogan, J.C. Laube, K. Kawamura, E. Capron, S. A. Montzka, E.J. Dlugokencky, D. Etheridge, T. Blunier, and W.T. Sturges, A new multi-gas constrained model of trace gas non-homogeneous transport in firn: evaluation and behavior at eleven polar sites, Atmos. Chem. Phys., 12, 11465-11483, 2012.
4. C. Hirsch, Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics, 2nd Ed., Butterworth-Heinemann, 2007.

Principles of physical modeling
E. Witrant
The Phases of Modeling
1: Structuring the problem
2. Setting up the Basic Equations
3. Forming the State-Space Models
Simplified models
Firn example
Conclusions
Homework

Homework 2

Use finite differences to solve the heat conduction

$$a\frac{\partial^2}{\partial z^2}x(z,\, t) = \frac{\partial}{\partial t}x(z,\, t),\; T(t) = x(L,\, t),\; P(t) = -\frac{\partial}{\partial z}x(z,\, t)|_{z=0}.$$

1. define the discretized state $X(t) \doteq [x_1(t)\ \ldots x_i(t)\ldots x_N(t)]^T$ as a spatial discretization of $x(z,\, t)$;
2. use the central difference approximation $\frac{\partial^2 u}{\partial z^2} \approx \frac{u_{i+1}-2u_i+u_{i-1}}{\Delta z^2}$ to express $dx_i(t)/dt$ as a function of x_{i+1}, x_i and x_{i-1}, for $i = 1\ldots N$;
3. introduce the boundary conditions
   - with $\frac{\partial u}{\partial z}(0,t) \approx \frac{u_1-u_0}{\Delta z}$ to express x_0 as a function of x_1 and P, then substitute in dx_1/dt;
   - with $\frac{\partial u}{\partial z}(L,t) \approx \frac{u_{N+1}-u_N}{\Delta z}$ to express x_{N+1} as a function of x_N, then substitute in dx_N/dt (suppose that there is no heat loss: $\partial x(L,t)/\partial z = 0$);
4. write the discretized dynamics in the state-space form;
5. for $N = 3$ compare with the results obtained in class.

Modeling and Estimation for Control
Physical Modeling

Lecture 3: Some Basic Relationships in Physics

Emmanuel WITRANT
emmanuel.witrant@univ-grenoble-alpes.fr

May 17, 2018.

Introduction

- Most common relationships within a number of areas in physics.
- More general relationships become visible.

$\Rightarrow$ General modeling strategy.

Outline

Physic fundamentals

Mass conservation
For a closed system: $\Delta M = 0$.

Energy conservation
For an isolated system: $\Delta E = 0$.

1st law of thermodynamics
Heat (Q) and Work (W) are equivalent and can be exchanged.
$\Delta E = \Delta U + \Delta E_{cin} + \Delta E_{pot} = Q + W$.

Electrical Circuits

Fundamental quantities:

voltage u (volt) and current i (ampere).

Components:

Nature	Relationship (law)	Energy
Inductor (L henry)	$i(t) = \frac{1}{L}\int_0^t u(s)ds, \quad u(t) = L\frac{di(t)}{dt}$	$T(t) = \frac{1}{2}Li^2(t)$ (magnetic field E storage, J)
Capacitor (C farad)	$u(t) = \frac{1}{C}\int_0^t i(s)ds, \quad i(t) = C\frac{du(t)}{dt}$	$T(t) = \frac{1}{2}Cu^2(t)$ (electric field E storage)
Resistor (R ohm)	$u(t) = Ri(t)$	
Nonlinear resistance	$u(t) = h_1(t)i(t), \quad i(t) = h_2(t)u(t)$	$P(t) = u(t) \cdot i(t)$ (loss, in watts, $1\ W = 1\ J/s$)
Ideal rectifier	$h_2(t) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$	

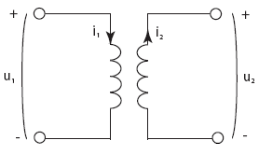
Interconnections (Kirkhhoff's laws):

$$\sum_k i_k(t) \equiv 0\ (nodes), \quad \sum_k u_k(t) \equiv 0\ (loops).$$

Ideal transformer:

transform voltage and current such that their product is constant:

$$u_1 \cdot i_1 = u_2 \cdot i_2, \quad u_1 = \alpha u_2, \quad i_1 = \frac{1}{\alpha} i_2$$



Mechanical Translation

Fundamental quantities:

force F (newton) and velocity v (m/s), 3-D vectors (suppose constant mass $\dot{m} = 0$).

Components:

Nature	Relationship (law)	Energy
Newton's force law	$v(t) = \frac{1}{m}\int_0^t F(s)ds, \quad F(t) = m\frac{dv(t)}{dt}$	$T(t) = \frac{1}{2}mv^2(t)$ (kinetic E storage)
Elastic bodies (k N/m)	$F(t) = k\int_0^t v(s)ds, \quad v(t) = \frac{1}{k}\frac{dF(t)}{dt}$	$T(t) = \frac{1}{2k}F^2(t)$ (elastic E storage)
Friction	$F(t) = h(v(t))$	
Air drag	$h(x) = cx^2 sgn(x)$	$P(t) = F(t) \cdot v(t)$ (lost as heat)
Dampers	$h(x) = \gamma x$	
Dry friction	$h(x) = \begin{cases} +\mu & \text{if } x > 0 \\ F_0 & \text{if } x = 0 \\ -\mu & \text{if } x < 0 \end{cases}$	

Interconnections:

$$\sum_k F_k(t) \equiv 0 \text{ (body at rest)}$$

$$v_1(t) = v_2(t) = \ldots = v_n(t) \text{ (interconnection point)}$$

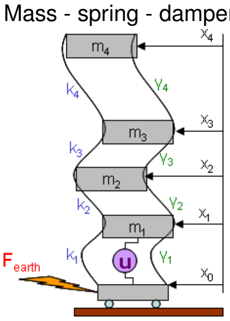
Ideal transformer:

force amplification thanks to levers:

$$\begin{aligned} F_1 \cdot v_1 &= F_2 \cdot v_2 \\ F_1 &= \alpha F_2 \\ v_1 &= \frac{1}{\alpha} v_2 \end{aligned}$$

Example: active seismic isolation control [Itagaki & Nishimura 2004]

Mass - spring - damper approximation:



$$
\begin{cases}
m_4\ddot{x}_4(t) &= \gamma_4(\dot{x}_3-\dot{x}_4)+k_4(x_3-x_4)\\
m_i\ddot{x}_i(t) &= [\gamma_i(\dot{x}_{i-1}-\dot{x}_i)+k_i(x_{i-1}-x_i)]\\
&\quad +[\gamma_{i+1}(\dot{x}_{i+1}-\dot{x}_i)\\
&\quad +k_{i+1}(x_{i+1}-x_i)], \quad i=2,3\\
m_1\ddot{x}_1(t) &= [\gamma_1(\dot{x}_0-\dot{x}_1)+k_1(x_0-x_1)]\\
&\quad +[\gamma_2(\dot{x}_2-\dot{x}_1)+k_2(x_2-x_1)]\\
&\quad +u(t)\\
m_1\ddot{x}_0(t) &= F_{earth}(t)\\
y(t) &= [\ddot{x}_0+\ddot{x}_1 \quad x_2-x_1]^T
\end{cases}
$$

Example: active seismic isolation control (2)

Experiment at UNAM (Mexico):

Mechanical Rotation

Fundamental quantities:

torque M $[N \cdot m]$ and angular velocity ω $[rad/s]$.

Components:

Nature	Relationship (law)	Energy
Inertia J $[Nm/s^2]$	$\omega(t)=\frac{1}{J}\int_0^t M(s)ds, \quad M(t)=J\frac{d\omega(t)}{dt}$	$T(t)=\frac{1}{2}J\omega^2(t)$ (rotational E storage)
Torsional stiffness k	$M(t)=k\int_0^t \omega(s)ds, \quad \omega(t)=\frac{1}{k}\frac{dM(t)}{dt}$	$T(t)=\frac{1}{2k}M^2(t)$ (torsional E storage)
Rotational friction	$M(t)=h(\omega(t))$	$P(t)=M(t)\cdot\omega(t)$

Interconnections:

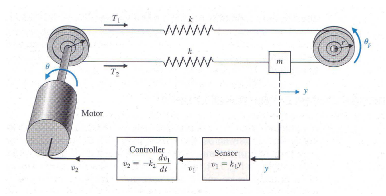
$$\sum_k M_k(t) \equiv 0 \text{ (body at rest).}$$

Ideal transformer:

a pair of gears transforms torque and angular velocity as:

$$
\begin{aligned}
M_1\cdot\omega_1 &= M_2\cdot\omega_2\\
M_1 &= \alpha M_2\\
\omega_1 &= \frac{1}{\alpha}\omega_2
\end{aligned}
$$

Example: printer belt pulley [Dorf & Bishop 2001]



$$\begin{cases} \text{Spring tension:} & T_1 = k(r\theta - r\theta_p) = k(r\theta - y) \\ \text{Spring tension:} & T_2 = k(y - r\theta) \\ \text{Newton (mass):} & T_1 - T_2 = m\dfrac{d^2y}{dt^2} \\ \text{Motor torque (resistance, } L = 0\text{):} & M_m = K_m i = \dfrac{K_m}{R} v_2 \\ \text{drives belts + disturb.:} & M_m = M + M_d \\ T \text{ drives shaft to pulleys:} & J\dfrac{d^2\theta}{dt^2} = M - h\dfrac{d\theta}{dt} - r(T_1 - T_2) \end{cases}$$

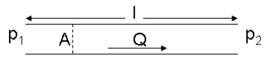
Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Flow Systems

Fundamental quantities:

for *incompressible* fluids, pressure p $[N/m^2]$ and flow Q $[m^3/s]$.

Fluid in a tube:



Pressure gradient	∇p	force	$p \cdot A$
mass	$\rho \cdot l \cdot A$	flow	$Q = v \cdot A$
inertance $[kg/m^4]$	$L_f = \rho \cdot l / A$		

Constitutive relationships (Newton: sum of forces = mass × accel.):

$$Q(t) = \frac{1}{L_f}\int_0^t \nabla p(s)ds, \quad \nabla p(t) = L_f \frac{dQ(t)}{dt} \qquad T(t) = \tfrac{1}{2}L_f Q^2(t)$$

(kinetic E storage)

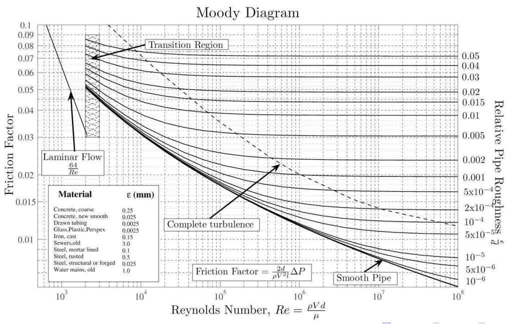
Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Fluid in a tube (2):

Pressure drop from Darcy-Weisbach's equation for a circular pipe:

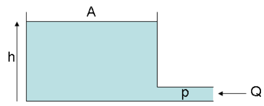
$$\frac{\partial P}{\partial x} = f\left(\frac{l}{D}\right)\left(\frac{v^2}{2g}\right)$$

Friction factor for laminar flow ($Re < 2300$): $f = \frac{64}{\mathrm{Re}}$; for turbulent flow, empirical formula or Moody Diagram:



Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Flow in a tank (i.e. no friction):



- Volume $V = \int Q dt$, $h = V/A$, and *fluid capacitance* $C_f \doteq A/\rho g$ $[m^4 s^2/kg]$.
- Constitutive relationships:

Bottom pres. $\quad \Delta p(t) = \rho \cdot g \cdot h$

$p = \rho \cdot g \cdot h + p_a \quad = \dfrac{1}{C_f}\displaystyle\int_0^t Q(s)ds \qquad T(t) = \tfrac{1}{2}C_f p^2(t)$

(potential E storage)

Flow through a section reduction:

- Pressure p, flow (hydraulic) resistance R_f, constant $\mathcal{H}$.
- Constitutive relationships:

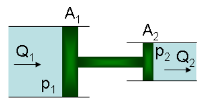
Pressure drop	$\nabla p(t) = h(Q(t))$
Darcy's law	$\nabla p(t) = R_f Q(t)$
area change	$\nabla p(t) = \mathcal{H} \cdot Q^2(t) \cdot sign(Q(t))$

Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Interconnections:

$$\sum_k Q_k(t) \equiv 0 \text{ (flows at a junction)}, \quad \sum_k p_k \equiv 0 \text{ (in a loop)}$$

Ideal transformer: piston



$$p_1 \cdot Q_1 = p_2 \cdot Q_2, \quad p_1 = \alpha p_2, \quad Q_1 = \frac{1}{\alpha} Q_2.$$

Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Thermal System

Fundamental quantities:

Temperature T $[K]$, Entropy S $[J/kg \cdot K]$ and heat flow rate $\dot{q}$ $[W]$.

3 ways to transfer heat:

- **Conduction**: Contact between 2 solids at different temperatures
- **Convection**: Propagation of heat through a fluid (gas or liquid)
- **Radiation**: 3^{rd} principle of thermodynamics : $P = \epsilon S \sigma T^4$ ($T > 0 \Rightarrow \dot{q}_{rad} > 0$)

Thermal energy of a body or Fluid: $E_{therm} = M \cdot C_p \cdot T$
Heat transported in a Flow: $\dot{q} = \dot{m} \cdot h$ (h=enthalpy)

Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

Heat Conduction

Body heating:

Fourier's law of conduction in 1D

$$k \cdot \frac{\partial T^2}{\partial x^2} = \rho \cdot C_p \cdot \frac{\partial T}{\partial t}$$
$$\dot{q}(t) = M \cdot C_p \cdot \frac{\partial T}{\partial t},$$

where k $[W/m \cdot K]$ is thermal conductivity of the body, ρ $[kg/m^3]$ and M $[kg]$ are the density and the mass of the body, and C_p $[W/(kg \cdot K)]$ is the specific Heat of the body.

Interconnections:

$$\sum_k \dot{q}_k(t) \equiv 0 \text{ (at one point)}.$$

Heat Convection

Forced convection between a flowing fluid in a pipe:

$$h \cdot S_w \cdot (T_w(t) - T(t)) = -M_w \cdot C_{p,w} \cdot \frac{dT_w(t)}{dt} = \dot{q}(t)$$

where T $[K]$ is the fluid temperature,
h $[W/m^2 \cdot K]$ is the heat transfert coefficient,
and T_w $[K]$, M_w $[kg]$, S_w $[m^2]$ $C_{p,w}$ $[J/kg \cdot K]$ are the temperature, mass, surface and specific heat of the pipe.

Interconnections:

$$\sum_k \dot{q}_k(t) \equiv 0 \text{ (at one point).}$$

Convective Heat Transfer coefficient

Correlation for Forced internal turbulent Flow:

Dittus-Boelter correlation (1930) with $10000 < Re < 120000$.

$$h = \frac{k}{D} Nu$$

where k is thermal conductivity of the bulk fluid, D is the Hydraulic diameter and Nu is the Nusselt number.

$$Nu = 0.023 \cdot Re^{0.8} \cdot Pr^n$$

with $Re = \frac{\rho \cdot v \cdot D}{\mu}$ is the Reynolds Number and Pr is the Prandtl Number. $n = 0.4$ for heating (wall hotter than the bulk fluid) and $n = 0.33$ for cooling (wall cooler than the bulk fluid). Precision is $\pm 15\%$

Thermal Systems: summary

- Conduction in 0D:
  Thermal capacity C $[J/(K \cdot s)]$
  $$T(t) = \frac{1}{C}\int_0^t \dot{q}(s)ds, \quad \dot{q}(t) = C\frac{dT(t)}{dt}$$
- Interconnections:

$$\dot{q}(t) = W\Delta T(t) \text{ (heat transfer between 2 bodies)}$$
$$\sum_k \dot{q}_k(t) \equiv 0 \text{ (at one point).}$$

where $W = hS_w$ $[J/(K \cdot s)]$.

Conclusions

Obvious similarities among the basic equations for different systems!

Some physical analogies:

System	Effort	Flow	Eff. storage	Flow stor.	Static relation
Electrical	Voltage	Current	Inductor	Capacitor	Resistor
Mechanical:					
- Translational	Force	Velocity	Body (mass)	Spring	Friction
- Rotational	Torque	Angular V.	Axis (inertia)	Torsion s.	Friction
Hydraulic	Pressure	Flow	Tube	Tank	Section
Thermal	Temperature	Heat flow rate	-	Heater	Heat transfer

Characteristics:

1. Effort variable e;
2. Flow variable f;
3. Effort storage: $f = \alpha^{-1} \cdot \int e$;
4. Flow storage: $e = \beta^{-1} \cdot \int f$;
5. Power dissipation: $P = e \cdot f$;
6. Energy storage via I.: $T = \frac{1}{2\alpha} f^2$;
7. Energy storage via C.: $T = \frac{1}{2\beta} e^2$;
8. Sum of flows equal to zero: $\sum f_i = 0$;
9. Sum of efforts (with signs) equal to zero: $\sum e_i = 0$;
10. Transformation of variables: $e_1 f_1 = e_2 f_2$.

- Note: analogies may be complete or not (i.e. thermal).

⇒ Create systematic, application-independent modeling from these analogies (next lesson).

Principles of physical modeling
E. Witrant
Physic fundamentals
Electrical Circuits
Mechanical Translation
Mechanical Rotation
Flow Systems
Thermal System
Heat Conduction
Heat Convection
Thermal Systems
Conclusions

References

1. L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
2. Noriaki Itagaki and Hidekazu Nishimura, “Gain-Scheduled Vibration Control in Consideration of Actuator Saturation”, *Proc. of the IEEE Conference on Control Applications*, pp 474- 479, vol.1, Taipei, Taiwan, Sept. 2-4, 2004.
3. R.C. Dorf and R.H. Bishop, *Modern Control Systems*, 9th Ed., Prentice Hall, 2001.

Modeling and Estimation for Control
Physical Modeling

Lecture 4: Bond Graphs

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

September 6, 2017

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Basic Concepts behind Bond Graphs [S. Stramigioli'01]

- Mathematical modeling: mathematical relations, generally without constraints or physical interpretation.
- Physical modeling: physical concepts and restrict to keep some physical laws.

Bond-graph

- satisfy 1^{st} principle of thermodynamics: energy conservation
- self-dual graphs where:

vertices = ideal physical concepts (storage or transformation of energy)

edges - power bonds - = lossless transfer of energy (i.e. water pipes, energy from one part to the other in the system)

⇒ excellent tool for describing power-consistent networks of physical systems.

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Causality

- Block diagrams: exchange of information takes place through arrows, variable x going from A to B = causal exchange of information

BUT often physically artificial and not justified, i.e. resistor

- Bond graphs: causality not considered in the modeling phase, only necessary for simulation.

Energy

- one of the most important concepts in physics
- dynamics is the direct consequence of energy exchange
- lumped physical models: system = network interconnection of basic elements which can store, dissipate or transform energy

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Outline

Physical Domains and Power Conjugate Variables

Physical domains:

- Discriminate depending on the kind of energy that a certain part of the system can store, i.e.
  - kinetic energy of a stone thrown in the air → translational mechanical
  - potential energy of a capacitor → electric domain
- Most important primal domains:
  - mechanical = mechanical potential & mechanical kinetic;
  - electromagnetic = electric potential & magnetic potential;
  - hydraulic = hydraulic potential & hydraulic kinetic;
  - thermic: only one without dual sub-domains, related to the irreversible transformation of energy to the thermal domain.

Power conjugate variables:

- Similarity among domains (cf. Lesson 3), i.e. oscillator
- In each primal domain: two special variables, power conjugate variables, whose product is dimensionally equal to power
- Efforts and flows:

Domain	Effort	Flow
Mechanical Translation	force F	velocity v
Mechanical Rotation	torque τ	angular velocity ω
Electro-magnetic	voltage v	current i
Hydraulic	pressure p	flow rate Q
Thermic	temperature T	heat flow rate $\dot{q}$

The Physical Model Structure and Bond Graphs

Energetic ports:

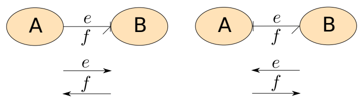
- physical modeling → atomic elements like the storage, dissipation, or transformation of energy;
- external variables = set of flows and dual vectors;
- effort-flow pairs = energetic ports since their dual product represents the energy flow through this imaginary port.

Bond graphs as a graphical language:

1. easy to draw;
2. mechanical to translate into block diagram or differential equations;
3. a few rules and it is impossible to make the common “sign mistakes” of block diagrams.

Energetic bonds:

- edges in the graph, represent the flow of energy (e.g. water pipes);
- notations: effort value above or left, flow under or right;
- rules:



1. each bond represents both an effort e and a dual flow f;
2. the half arrow gives the direction of positive power $P = e^T f$ (energy flows);
3. effort direction can be, if necessary, specified by the causal stroke & dual flow goes ALWAYS in the opposite direction (if not an element could set P independently of destination → extract infinite energy).

Network structure:

- if 2 subsystems A and B, both the effort and flow MUST be the same: interconnection constraint that specifies how A and B interact;
- more generally, interconnections and interactions are described by a set of bonds and junctions that generalize Kirchhoff's laws.

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Energy Storage and Physical State

Identical structure for physical lumped models

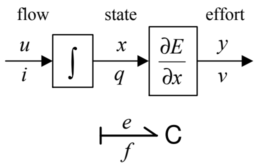
- Integral form characterized by:
  1. an input $u(t)$, always and only either effort or flow;
  2. an output $y(t)$, either flow or effort;
  3. a physical state $x(t)$;
  4. an energy function $E(x)$.
- State-space equations: $\dot{x}(t) = u(t)$, $y(t) = \dfrac{\partial E(x(t))}{\partial x}$
- Change in stored energy:

$$\dot{E} = \frac{dE}{dt} = \frac{\partial E(x)}{\partial x}^T \frac{dx}{dt} = y^T u = P_{supplied}$$

$\rightarrow$ half arrow power bonds always directed towards storage elements ($\dot{E} > 0$)!

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

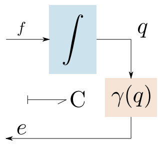
e.g. Capacitor:



Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

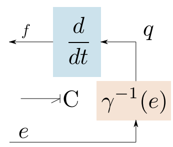
Bond graphs representations

- Depending whether u is an effort or a flow in the integral form, two dual elements:
  - C **element:** has flow input u and dual effort output y;
  - I **element:** has effort input u and dual flow output y.
- Causal representations:



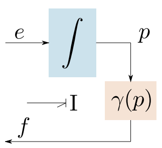
generalized displacement
$q(t) = q(t_0) + \int_{t_0}^{t} f(s)ds$

generalized potential energy $E(q)$



differential form $\rightarrow \gamma^{-1}(e)$

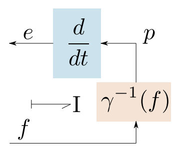
co-energy
$E^*(e) \Rightarrow \gamma^{-1}(e) = \dfrac{\partial E^*(e)}{\partial e}$



generalized momenta

$p(t) = p(t_0) + \int_{t_0}^{t} e(s)ds$

generalized kinetic energy $E(p)$



differential form $\rightarrow \gamma^{-1}(f)$

co-energy

$E^*(f) \Rightarrow \gamma^{-1}(f) = \frac{\partial E^*(f)}{\partial f}$

- Multidimensional I indicated by $\mathbb{I}$ and multidimensional C = $\mathbb{C}$.

Mechanical domain

C - Spring:

- input u = velocity v, generalized displacement $\int v = x$, stored potential energy $E(x) = \frac{1}{2}kx^2$, effort $y = \frac{\partial E}{\partial x} = kx = F$ (elastic force);
- holds for ANY properly defined energy function, which is the ONLY information characterizing an ideal storage of energy;
- e.g. nonlinear spring: $E(x) = \frac{1}{2}kx^2 + \frac{1}{4}kx^4 \Rightarrow y = F = \frac{\partial E}{\partial x} = kx + kx^3$;
- linear spring, co-energy $E^*(F) = \frac{1}{2}\frac{F^2}{k}$, $x = \gamma^{-1}(F) = \frac{\partial E^*(F)}{\partial F} = \frac{F}{k}$.

I - Effort as input, kinetic mechanical domain:

- input u = force F, $\int F = p = mv$ (momenta) by Newton's law (holds if $m(t)$)

$\Rightarrow$ proper physical state for kinetic E storage: momentum p;

- $E(p) = \frac{1}{2}\frac{p^2}{m}$, $y = v = \gamma(p) = \frac{\partial E}{\partial p} = \frac{p}{m}$;
- kinetic co-energy $E^*(v) = \frac{1}{2}mv^2$, $p = \gamma^{-1}(v) = \frac{\partial E^*(v)}{\partial v} = mv$.

Electrical domain:

- proper physical states: charge q and flux ϕ, NOT i and v;

C - Storage in electrostatic domain:

- $u = i$, physical state $\int i = q$ (generalized displacement), stored potential energy $E(q) = \frac{1}{2}\frac{q^2}{C}$ (co-energy $E^*(v) = \frac{1}{2}Cv^2$), effort $y = \frac{\partial E}{\partial q} = \frac{q}{C} = v$;
- e.g. nonlinear capacitor: $E(q) = \frac{1}{2}\frac{q^2}{C} + \frac{1}{4}\frac{q^4}{C} \Rightarrow y = v = \frac{q}{C} + \frac{q^3}{C}$.
- using co-energy, $q = \gamma^{-1}(v) = \frac{\partial E^*(v)}{\partial v} = Cv$.

I - Ideal inductor:

- $u = v$, $\int v = \phi$, $E(\phi) = \frac{1}{2}\frac{\phi^2}{L}$, where $L \doteq$ induction constant, $y = i = \frac{\phi}{L}$.

Energy storage:

- Generalized states:

Domain	Gen. momentum ($\int e$)	Gen. displacement ($\int f$)
Mech. Translational	momentum p	displacement x
Mech. Rotational	ang. momentum m	ang. displacement θ
Electromagnetic	flux linkage ϕ	charge q
Hydraulic	pressure mom. P_p	volume V
Thermic	NON EXISTENT	entropy E

- Storage elements:
  1. what are the real physical states?
  2. energy function provides the equation;
  3. argument → what physical ideal element it represents;
  4. the only ideal physical elements to which a state is associated are energy storage;
  5. in bond graphs, the power bond connected to a storage element must always be directed toward the element.

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Duality

- 2 storage / physical domain but thermal (generalized potential and kinetic energy storage) = dual;
- one major concept in physics: oscillations if interconnected dual elements, e.g. spring-mass or capacitor-inductor;
- thermal domain does NOT have both = irreversibility of energy transformation due to a lack of "symmetry".

Extra supporting states

- states without physical energy;
- e.g. position of a mass translating by itself: physical state p, position $x = \int v = p/m$ but if the measurement is x and not v:
$$\begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} = \begin{pmatrix} 0 \\ p/m \end{pmatrix} + \begin{pmatrix} u \\ 0 \end{pmatrix}, \quad y = x$$

⇒ total state $(p,\ x)^T$, physical state p, supporting state x needed for analysis without associated physical energy.

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Free Energy Dissipation

Principle:

- irreversible transformation, e.g. mechanical or electrical → thermal;
- dissipation of energy is transformation (1^{st} principle of thermodynamics);
- dissipation of free-energy (math.: Legendre transformation of energy with respect to entropy), e.g. ideal electrical resistors or mechanical dampers;
- ideal dissipator characterized by a purely statical (no-states) effort/flow relation: $e = Z(f)$ (Impedance form) or $f = Y(e)$ (Admittance form) for which $Z(f)f < 0$ or $eY(e) < 0$ (energy flowing toward the element)

$$\xrightharpoonup[f]{e} \mathrm{R} \quad : r$$

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Electrical domain

- Ohm's law: $u = Ri$ and $i = u/R$;
- causally invertible;
- r: constant R of a linear element ($r = R$).

Mechanical domain

- viscous damping coefficient b: $F = bv$ and $v = F/b$, $r = b$.

Ideal Transformations and Gyrations

Electrical domain

- elements with two power ports = two power bonds;
- ideal, power continuous, two port elements: power flowing from one port (input bond) ≡ one flowing out from other port (output bond) ⇒ cannot store energy inside.
- e.g. ideal transformer:
  - input and output bonds with positive power flow in and out;
  - external variables: (e_{in}, f_{in}) = power flowing in from input port and (e_{out}, f_{out}) = power flowing out from other port;
  - power continuity: $P_{in} = e_{in}^T f_{in} = e_{out}^T f_{out} = P_{out}$
- linear relation between one of the external variable on one port to one of the external variables on the other port;
- flow-flow → ideal transformers, flow-effort → ideal gyrators

Ideal Transformers

$$\xrightarrow[f_{in}]{e_{in}} \underset{n}{\overset{}{\mathrm{TF}}} \xrightarrow[f_{out}]{e_{out}} \qquad \xrightarrow[f_{in}]{e_{in}} \underset{\substack{\uparrow \\ n}}{\mathrm{MTF}} \xrightarrow[f_{out}]{e_{out}}$$

- relation: linear between flows and dependent linear between efforts;
- characterizing equation: $f_{out} = nf_{in}$ where n: linear constant characterizing the transformer
- power constraint: $e_{in} = ne_{out}$ ⇔ $e_{out} = \frac{1}{n}e_{in}$
  ⇒ if 2 ports belong to same domain and $n < 1$, $e_{in} < e_{out}$ but $f_{in} > f_{out}$.
- e.g. gearbox of a bicyle: e_{in} = torque applied on pedal axis and f_{in} = angular velocity around the pedals, (e_{out}, f_{out}) on the back wheel;
- n relates the efforts in one way and also the flows in the other way;
- if n variable: modulated TF (extra arrow).

Ideal Gyrators

$$\xrightarrow[f_{in}]{e_{in}} \underset{n}{\mathrm{GY}} \xrightarrow[f_{out}]{e_{out}} \qquad \xrightarrow[f_{in}]{e_{in}} \underset{\substack{\uparrow \\ n}}{\mathrm{MGY}} \xrightarrow[f_{out}]{e_{out}}$$

- linear constant between effort of output port and flow of input port: $e_{out} = nf_{in}$;
- power constraint: $e_{in} = nf_{out}$ ⇔ $f_{out} = \frac{1}{n}e_{in}$;
- e.g. gyrative effect of a DC motor (electrical power flows in and mechanical power flows out): out torque $\tau = Ki$, power continuity → $u = K\omega$ (e.m.f.):

$$\textit{Electrical domain} \quad \begin{pmatrix} i \\ u \end{pmatrix} \quad \begin{matrix} \rightarrow \\ \leftarrow \end{matrix} \quad \begin{pmatrix} \tau \\ \omega \end{pmatrix} \quad \textit{Rotational domain}$$

- if n variable: modulated gyrator.

Multi-bonds

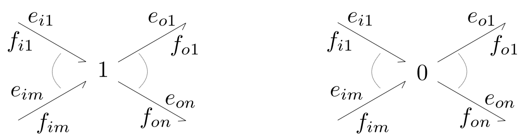
- characteristic constant → matrix, if variable → modulated transformer or gyrator;
- Transformers:
  - $\mathbb{TF}$, $\mathbb{MTF}$;
  - $f_2 = Nf_1 \Rightarrow e_1 = N^T e_2$ (using $e_1^T f_1 = e_2^T f_2$);
- Gyrators:
  - $\mathbb{GY}$, $\mathbb{MGY}$, $\mathbb{SGY}$;
  - $e_2 = Nf_1 \Rightarrow e_1 = N^T f_2$;
  - $e = Sf$ with $S = -S^T = \begin{bmatrix} 0 & -N^T \\ N & 0 \end{bmatrix}$
  - if N = identity matrix: symplectic gyrator $\mathbb{SGY}$ (algebraic relationship, can be used to dualize $\mathbb{C}$ into $\mathbb{I}$).

Ideal Sources

$$\rightarrow f : \quad \mathrm{S_f} \overset{e}{\underset{f}{\rightharpoonup}} \qquad\qquad \rightarrow e : \quad \mathrm{S_e} \overset{e}{\underset{f}{\rightharpoonup}}$$

- Supply energy: ideal flow source and ideal effort source.
- Only elements from which the power bond direction goes out: $P_{source} = e^T f$.
- Supply a certain effort or flow independently of the value of their dual flow and effort.
- e.g. ideal voltage and current source in the electrical domain

Kirchhoff's Laws, Junctions and the Network Structure



- How we place the bricks with respect to each other determines the energy flows and dynamics
- Generalization of Kirchhoff's laws, network structure → constraints between efforts and flows
- Two basic BG structures: **1** junctions = flow junctions and **0** junctions = effort junctions
- Any number of attached bonds
- Power continuous (in = out)

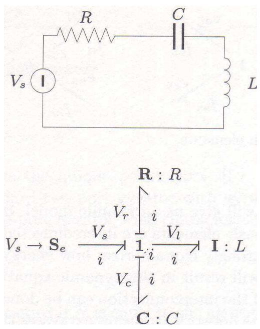
1-junctions:

- flow junction: all connected bonds are constrained to have the same flow values;
- causality: only one bond sets the in flow and all other bonds use it (strokes constraint);
- equations:

$$f_{i1} = \cdots = f_{im} = f_{o1} = \cdots = f_{on} \quad \text{(flow equation)},$$
$$\sum_{k=1}^{m} e_{ik} = \sum_{k=1}^{n} e_{ok} \quad \text{(effort equation)};$$

- Kirchhoff's law for a mesh in electrical networks: same current and the algebraic potential sum = 0;

Electrical example:



- same current → flow junction

- all bonds point to R, C and I and source bond point out → all signs are automatically correct;
- I (integral causality) "sets" the junction current (mesh) and other elements have this current as input and voltages as outputs;
- complete dynamics described by:
  - effort equation: $V_s = V_r + V_c + V_l$
  - I element: $\dot{\phi} = V_l$ and $i = \phi/L$
  - q element: $\dot{q} = i$ and $V_c = q/C$
  - R element: $V_r = Ri$

0-junctions:

- effort junction: all connected bonds constrained to have same efforts;
- causality: only one bond sets e_{in} and all other bonds use it;
- equations:

$$e_{i1} = \cdots = e_{im} = e_{o1} = \cdots = e_{on} \quad \text{(effort equation)},$$
$$\sum_{k=1}^{m} f_{ik} = \sum_{k=1}^{n} f_{ok} \quad \text{(flow equation)};$$

- Kirchhoff's law for a node: algebraic current sum = 0.

Effort difference:

$$\begin{tikzcd} & e_1 - e_2 : 0 & \\ e_1 : 0 \arrow[r, "e_1", "f"'] & 1 \arrow[u, "e_1 - e_2", "f"'] \arrow[r, "e_2", "f"'] & 0 : e_2 \end{tikzcd}$$

- need the difference of two efforts to specify power consistent interconnection with other elements;
- all flows are the same and

$$\sum_{k=1}^{m} e_{ik} = \sum_{k=1}^{n} e_{ok} \Rightarrow e_1 = e_2 + e_3 \Leftrightarrow e_3 = e_1 - e_2.$$

Flow difference:

$$\begin{tikzcd} & 1 : f_1 - f_2 & \\ f_1 : 1 \arrow[r, "e", "f_1"'] & 0 \arrow[u, "e", "f_1 - f_2"'] \arrow[r, "e", "f_2"'] & 1 : f_2 \end{tikzcd}$$

- need the difference of two flows to specify power consistent interconnection with other elements;
- all efforts are the same and

$$\sum_{k=1}^{m} f_{ik} = \sum_{k=1}^{n} f_{ok} \Rightarrow f_1 = f_2 + f_3 \Leftrightarrow f_3 = f_1 - f_2.$$

Bond Graph Modeling of Electrical Networks

Algorithm:

1. for each node draw a 0-junction which corresponds to the node potential;
2. for each bipole connected between two nodes, use effort difference where a bipole is attached and connect the ideal element to the 0-junction representing the difference.
3. choose a reference ($v = 0$) and attach an effort source equal to zero to the corresponding 0-junction.
4. simplify:
   - eliminate any junction with only 2 attached bonds and have the same continuing direction (one in and one out);
   - fuse 1 and 0-junctions that are connected through a single-bond;
   - eliminate all junctions after the 0 reference source that do not add any additional constraint.

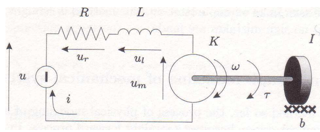
Bond Graph Modeling of Mechanical Systems

Algorithm:

1. for each moving mass draw a 1-junction = mass velocity;
2. add an additional 1-junction for inertial reference with an attached $S_f = 0$;
3. for each inertia attach a corresponding I element to the one junction corresponding to its velocity;
4. for each damper or spring: flow difference for Δv attach to the 1-junction;
5. simplify the graph by:
   - eliminating all junctions with only two bonds in the same continuing direction;
   - fuse 1 and 0-junctions connected through a single-bond;
   - eliminate all the junctions after the reference source which do not add any additional constraints.

Examples

DC motor example



- 6 interconnected lumps:
  - 2 storage elements with corresponding physical states (ϕ, p): ideal inductor L and rotational inertia $I \to 2$ states and order 2 model;
  - 2 dissipative elements: the resistor R and the friction b;
  - 1 gyration effect K;
  - an ideal voltage source u.

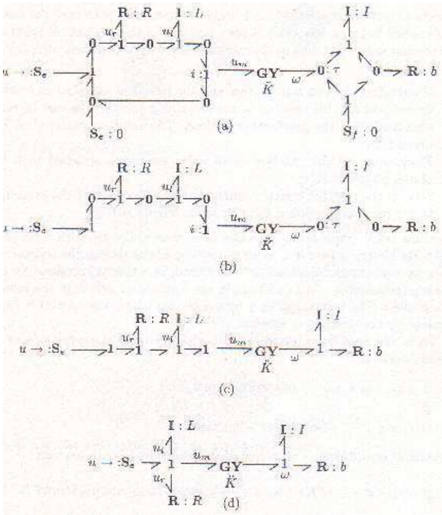
- Elements equations:
  - storage elements and physical states:

$$\text{Inertia} \quad \begin{cases} \dot{p} = \tau_I \\ \omega = \dfrac{\partial E_I}{\partial p} = \dfrac{\partial}{\partial p}\left(\dfrac{1}{2I}p^2\right) = \dfrac{p}{I} \end{cases}$$

$$\text{Inductor} \quad \begin{cases} \dot{\phi} = u_l \\ i = \dfrac{\partial E_L}{\partial \phi} = \dfrac{\partial}{\partial \phi}\left(\dfrac{1}{2L}\phi^2\right) = \dfrac{\phi}{L} \end{cases}$$

  - dissipation (linear): $u_r = Ri$ and $\tau_b = b\omega$ (dissipating torque);
  - gyration equations: $\tau = Ki$ and $u_m = K\omega$

- Network interconnection:
  - use previous algorithms to describe the electrical and mechanical parts;
  - introduce the gyrator to connect the two domains → inter-domain element;
  - (a) Preliminary diagram drawing:
    - 0-junctions of electrical to indicate the connection points of the bipoles;
    - mechanical: 1-junctions = angular rotation of the wheel and reference inertial frame (source);
    - gyrator = relation from flow i to effort $\tau \Rightarrow$ 1 to 0 junction;
    - torque applied between the wheel and ground.
  - simplifications:
    - (b) eliminate the two zero sources and attached junctions;
    - (c) eliminate any junction with only two bonds attached to it;
    - (d) mix all the possible directly communicating junctions of the same type.

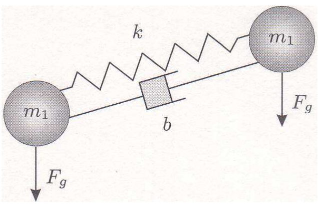


Intuitively:

- electrical part = series connection source, resistor, inductor and electrical gyrator side → 1-junction;
- mechanical part: only the velocity w is present, the motor applies a torque to the wheel, but part of it is "stolen" by the dissipating element.
- final equations ⇒ LTI state-space form:

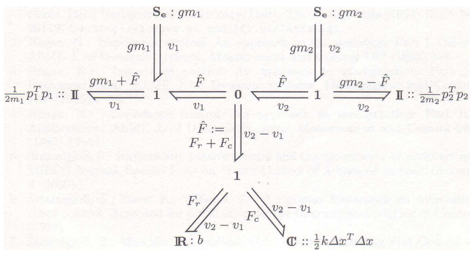
$$\begin{aligned}
\dot{p} &= \tau_I = \tau - \tau_b = Ki - b\omega = \frac{K}{L}\phi - \frac{b}{I}p,\\
\dot{\phi} &= u_I = -u_m - u_r + u = -\frac{K}{I}p - \frac{R}{L}\phi + u\\
\frac{d}{dt}\begin{pmatrix} p \\ \phi \end{pmatrix} &= \underbrace{\begin{pmatrix} -b/I & K/L \\ -K/I & -R/L \end{pmatrix}}_{A}\begin{pmatrix} p \\ \phi \end{pmatrix} + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{B}u\\
y \doteq \omega &= \underbrace{\begin{pmatrix} 1/I & 0 \end{pmatrix}}_{C}\begin{pmatrix} p \\ \phi \end{pmatrix}
\end{aligned}$$

Multidimensional example



- two point masses connected by an elastic translational spring and a damper;

- bond graph:



- note: all bonds attached to 1-junction have the same flows and all attached to 0-junction the same effort;
- ":: $E(q)$" = energy function, q = energy variable (p_1, p_2) for $\mathbb{I}$ and position diff. Δx for elastic;
- ideal source → constant force = gravitation for each mass;
- ": b" for dissipative element indicates $F_r = b(v_2 - v_I)$.

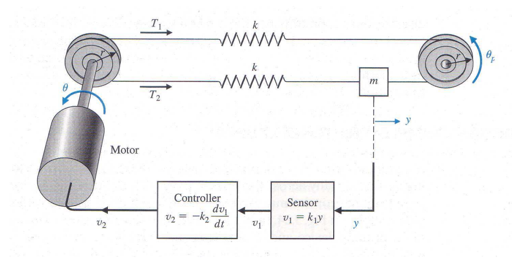
Conclusions

Bond graphs:

- Provide a systematic approach to multiphysics modeling
- Based on the fundamental laws of energy conservation
- Fundamental theory = port-Hamiltonian systems
- Used in industry with dedicated numerical solvers (e.g. 20-Sim)
- Needs practice!

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Homework 3



Draw the bond graph model of the printer belt pulley problem introduced in Lesson 3 and check that you obtain the same equations.

Bond Graphs
E. Witrant
Power Conjugate Variables
Structure and Bond Graphs
Storage and state
Energy Dissipation
Transformations and Gyrations
Ideal sources
Junctions
Electrical Networks
Mechanical Systems
Examples
Conclusions

Reference

1. S. Stramigioli, *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-free Approach*, Springer, LNCIS 266, 2001.
2. L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.

Modeling and Estimation for Control

Simulation

Lecture 5: Computer-aided modeling

Emmanuel WITRANT
`emmanuel.witrant@ujf-grenoble.fr`

April 22, 2014

Introduction

$$
\begin{aligned}
\dot{x} &= f(x, u) \\
y &= h(x, u)
\end{aligned}
$$

- Can contain complex calculations.
- Computer assistance?
  - Computer algebra.
  - 2 systematic ways to state-space: algebraic and bond graphs.
- Numerical limitations.

Outline

Computer Algebra

- Methods for manipulating mathematical formulas ($\neq$ numerical calculations).
- Numerous softwares: Macsyma, Maple, Reduce, Axiom, Mathematica...
- Examples of capabilities:
  - Algebraic expressions: $(x+y)^2 = x^2 + 2xy + y^2$
  - Factorizations: $x^3 - y^3 = (x-y)(x^2 + xy + y^2)$
  - Symbolic differentiation

$$\frac{\partial}{\partial z}(x^2 z + \sin yz + a \tan z) = x^2 + y \cos yz + \frac{a}{1+z^2}$$

  - Symbolic integration

$$\int \sqrt{1+x^2}dx = \frac{1}{2}(\text{arc sinh} x + x\sqrt{x^2+1})$$

Analytical Solutions

- May have partial interesting results, i.e.

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, x_2)\\ \dot{x}_2 &= f_2(x_1, x_2)\end{aligned}$$

solution algorithm generates $F(x_1, x_2) = C$ if possible, continue from this to

$$\begin{aligned}x_1 &= \phi_1(t)\\ x_2 &= \phi_2(t).\end{aligned}$$

F is called the *integral* of the system, geometrically = path in $x_1 - x_2$ plane, but do not have velocity information.

Example: the pendulum

$$\begin{aligned}\dot{\theta} &= \omega\\ \dot{\omega} &= -\frac{g}{l}\sin\theta\end{aligned}$$

has integral $\frac{1}{2}\omega^2 - \frac{g}{l}\cos\theta = C$ which represents the energy (kinetic + potential) of the system.

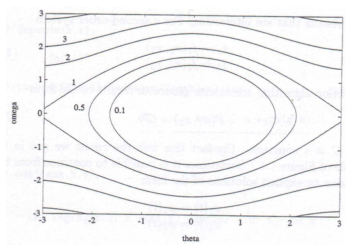


Figure: Pendulum trajectories in $\theta - \omega$ plane

Algebraic Modeling

→ Transform the equations into a convenient form.

Introduction of state variables for higher-order differential equations:

- Consider
$$F(y, \dot{y}, \ldots, y^{n-1}, y^n; u) = 0,$$
- introduce the variables
$$x_1 = y,\ x_2 = \dot{y},\ \ldots,\ x_n = y^{n-1},$$
- we get
$$\dot{x}_1 = x_2, \quad ,\dot{x}_2 = x_3, \quad \ldots \quad ,\dot{x}_{n-1} = x_n$$
$$F(x_1, x_2, \ldots, x_n, \dot{x}_n; u) = 0$$

→ state-space description provided $\dot{x}_n$ can be solved for the last equation.

Example

- Let
$$y^{(3)^2} - \dot{y}^2 y^4 - 1 = 0.$$
- With $x_1 = y$, $x_2 = \dot{y}$, $x_3 = \ddot{y}$, we get
$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= x_3\\ \dot{x}_3^2 - x_2^2 x_1^4 - 1 &= 0\end{aligned}$$
- The last equation can be solved for $\dot{x}_3$ and gives
$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= x_3\\ \dot{x}_3 &= \pm\sqrt{x_2^2 x_1^4 + 1}\end{aligned}$$

Note: 2 cases if we don't know the sign of $y^{(3)} = \dot{x}_3$ from physical context.

Systems of higher-order differential equations:

- two higher-order differential equations in 2 variables

$$\begin{aligned} F(y, \dot{y}, \ldots, y^{n-1}, y^n; v, \dot{v}, \ldots, v^{m-1}; u) &= 0 \\ G(y, \dot{y}, \ldots, y^{n-1}; v, \dot{v}, \ldots, v^{m-1}, v^m; u) &= 0 \end{aligned}$$

- introduce the variables

$$x_1 = y,\ x_2 = \dot{y},\ \ldots,\ x_n = y^{n-1},$$
$$x_{n+1} = v,\ x_{n+2} = \dot{v},\ \ldots,\ x_{n+m} = v^{m-1},$$

- we get

$$\begin{aligned} &\dot{x}_1 = x_2, \quad \dot{x}_2 = x_3, \quad \ldots, \quad \dot{x}_{n-1} = x_n \\ &F(x_1, x_2, \ldots, x_n, \dot{x}_n; x_{n+1}, \ldots, x_{n+m}; u) = 0 \\ &\dot{x}_{n+1} = x_{n+2}, \quad \ldots, \quad \dot{x}_{n+m-1} = x_{n+m} \\ &G(x_1, x_2, \ldots, x_n; x_{n+1}, \ldots, x_{n+m}, \dot{x}_{n+m}; u) = 0 \end{aligned}$$

$\Rightarrow$ state-space description if $\dot{x}_n$ and $\dot{x}_{n+m}$ can be solved in F and G.

- Example:

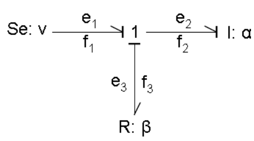
$$\begin{aligned} \ddot{y} + \ddot{v} + \dot{y}\dot{v} &= 0 \quad (1) \\ \frac{y^2}{2} + \frac{v^2}{2} - 1 &= 0 \quad (2) \end{aligned}$$

Problem: highest order derivatives in same equation

- Solution:
  - differentiate (2) twice gives (3);
  - (1)$\times v$-(3) =(4);
  - (4)$\times v^2$ & $v\dot{v}$ eliminated with (3) gives (5);
  - eliminate v thanks to (2) $\rightarrow$ eq. in y only.
- Can be generalized to an arbitrary number of equations provided all equations are polynomial in the variables and their derivatives.

An Automatic Translation of Bond Graphs to Equations

From a simple example:



- Introduce the state $x = \alpha f_2$ for I: $\dot{x} = e_2$;
- imagine a list of equations with e_i and f_i computed from v and x, $e_1 = v$ first and $f_1 = f_2$ last (or $f_1 = f_3$);

$$\begin{aligned} e_1 &= v \\ &\vdots \\ f_1 &= f_2 \end{aligned}$$

1) from I element: $f_2 = x/\alpha$, dual $e_2 = \dot{x}_2 = e_1 - e_3$ (junction output) $\rightarrow$ second to last so that e_1 and e_3 are calculated before:

$$\begin{aligned} e_1 &= v \\ f_2 &= \frac{1}{\alpha}x \\ &\vdots \\ \dot{x} &= e_2 = e_1 - e_3 \\ f_1 &= f_2 \end{aligned}$$

2) What variables are defined by first 2 equation? Junction $\rightarrow$ flows and R:

$$\begin{aligned} e_1 &= v \\ f_2 &= \frac{1}{\alpha}x \\ f_3 &= f_2 \\ e_3 &= \beta f_3 \\ \dot{x} &= e_2 = e_1 - e_3 \\ f_1 &= f_2 \end{aligned}$$

$\Rightarrow$ starting from v and x, all variables evaluated in proper order.

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- successive substitutions gives a compact state-space description:

$$\dot{x} = e_1 - e_3 = e_1 - \beta f_3 = e_1 - \beta f_2 = e_1 - \frac{\beta}{\alpha}x = v - \frac{\beta}{\alpha}x$$

$\rightarrow$ choose 2 lists, forward and backward, instead of one.

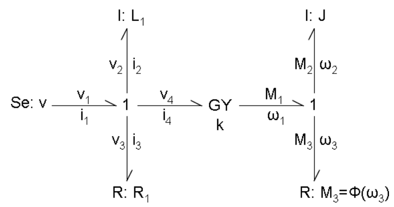
Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Algorithms for Equation Sorting

1. Choose a source and write its input in forward list and the equation of its dual in backward list.
2. From adjacent bonds, if some variable is defined in terms of already calculated variables, write its equation in the forward list and the equation of the other bond variable in the backward list, as far as possible.
3. Repeat 1 and 2 until all sources have been treated.
4. Choose an I element and write the equation $f_i = \frac{1}{\alpha_i}x_i$ in forward list and $\dot{x}_i = e_i = \ldots$ in backward list.
5. Do the analogy of step 2.
6. Repeat 4 and 5 until all I elements have been processed.
7. Do the analogy of steps 4, 5, and 6 for all C elements ($e_i = \frac{1}{\beta_i}x_i$ to forward list and $\dot{x}_i = f_i$ backward list.
8. Reverse the order of the backward list and put it after the forward list.

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Example: DC motor



- State variables:

$$x_1 = \int^t v_2 d\tau = L_1 i_2, \quad \int^t M^2 d\tau = J\omega_2$$

- Create the list:

Step	Forward list	Backward list
1	$v_1 = v$	$i_1 = i_2$
2	$i_2 = \frac{1}{L_1}x_1$	$\dot{x}_1 = v_2 = v_1 - v_3 - v_4$
2	$i_3 = i_2$	$v_3 = R_1 i_3$
2	$i_4 = i_2$	$v_4 = k\omega_1$
2	$M_1 = ki_4$	$\omega_1 = \omega_2$
4	$\omega_2 = \frac{1}{J}x_2$	$\dot{x}_2 = M_2 = M_1 - M_3$
5	$\omega_3 = \omega_2$	$M_3 = \phi(\omega_3)$

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- Reverse backward list after forward list:

$$\begin{aligned}
v_1 &= v \\
i_2 &= \frac{1}{L_1}x_1 \\
i_3 &= i_2 \\
i_4 &= i_2 \\
M_1 &= ki_4 \\
\omega_2 &= \frac{1}{J}x_2 \\
\omega_3 &= \omega_2 \\
M_3 &= \phi(\omega_3) \\
\dot{x}_2 &= M_2 = M_1 - M_3 \\
\omega_1 &= \omega_2 \\
v_4 &= k\omega_1 \\
v_3 &= R_1 i_3 \\
\dot{x}_1 &= v_2 = v_1 - v_3 - v_4 \\
i_1 &= i_2
\end{aligned}$$

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- Eliminating all variables that are not states gives:

$$\begin{aligned}
\dot{x}_1 &= v - \frac{R_1}{L_1}x_1 - \frac{k}{J}x_2 \\
\dot{x}_2 &= \frac{k}{L_1}x_1 - \phi(x_2/J)
\end{aligned}$$

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Numerical Methods

Physical model → state-space equations → scaling (same order of magnitude to avoid numerical problems) → impact of discretization in simulation.

Basis of Numerical Methods:

- Consider the state-space model

$$\dot{x} = f(x(t), u(t))$$

where $x \in \mathbb{R}^n$. If fixed input $u(t) = \bar{u}(t)$, u is a time variation and

$$\begin{aligned}
\dot{x} &= f(t, x(t)) \\
x(0) &= x_0
\end{aligned}$$

we want an approximation of x at $0 < t_1 < t_2 < \cdots < t_f$ → $x_1, x_2, x_3, \ldots$ approximate $x(t_1), x(t_2), x(t_3), \ldots$

- Simplest algorithm: difference ratio = *Euler's method*:

$$\frac{x_{n+1} - x_n}{h} \approx \dot{x}(t_n) = f(t_n, x_n), \text{ where } h = t_{n+1} - t_n$$

$$\Rightarrow x_{n+1} = x_n + h \cdot f(t_n, x_n)$$

more generally

$$x_{n+1} = G(t, x_{n-k+1}, x_{n-k+2}, \ldots, x_n, x_{n+1})$$

where k is the number of utilized previous steps $\rightarrow$ *k-step method*. If x_{n+1} not in G: *explicit* method (i.e. Euler), otherwise *implicit*.

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Firn example: gas in open pores (1)

Impact of the convection term discretization on the trace gases mixing ratio at NEEM (EU hole)

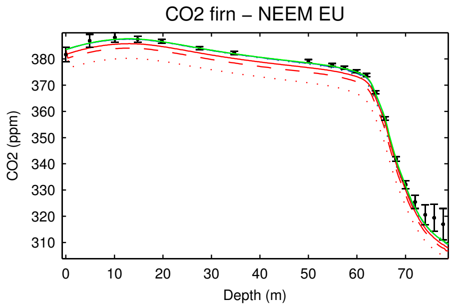


Figure: For 100 ('$\cdots$'), 200 ('- - -') and 395 ('—') depth levels ($\Delta z \approx$ 0.8, 0.4 and 0.2 m, respectively): Lax-Wendroff (blue, reference), central (red) and first order upwind (green).

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Firn example: gas in open pores (2)

Impact of time discretization on the trace gases mixing ratio at NEEM (EU hole, $\Delta z = 0.2$ m and a zoom on specific region)

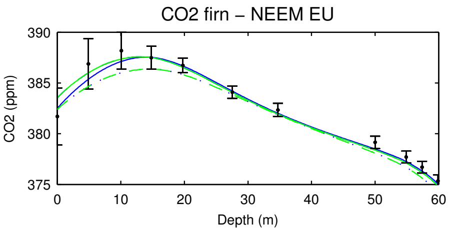


Figure: Explicit with a sampling time t_s=15 minutes (red), implicit (blue) with $t_s = 1$ day ('—'), 1 week ('– – –') and 1 month ('- - -'), and implicit-explicit (green) with $t_s = 1$ week ('—') and 1 month ('– – –').

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Firn example: gas in open pores (3)

Averaged simulation time per gas associated with the proposed time-discretization schemes for NEEM EU (1800 to 2008, full close-off depth at 78.8 m, 12 gases, left) and South Pole 1995 (1500 to 1995, full close-off depth at 123 m), obtained on a *PC* laptop equipped with the processor i5 540 m (2.53 Ghz, 3 Mo):

Method	t_s	Δz [a]	**Simulation time** [a]
Implicit	1 day	0.2 m	4.02 / 22.25 s
Implicit	1 week	0.2 m	0.63 / 3.91 s
Implicit	1 month	0.2 m	0.26 / 1.48 s
Explicit	15 min	0.2 m	5.09 / 29.45 min
Explicit	30 min	0.4 / 0.61 m	24.39 s / 1.34 min
Explicit	1 h	0.8 / 1.23 m	7.19 s / 12.13 s
Imp-explicit [b]	1 week	0.2 m	0.63 s / 3.77 s
Imp-explicit [b]	1 month	0.2 m	0.27 s / 1.48 s

[a]: NEEM EU / South Pole; [b]: Crank-Nicholson.

- Accuracy determined by the global error

$$E_n = x(t_n) - x_n$$

but hard to compute → one-step (provided exact previous steps), local error

$$e_n = x(t_n) - z_n, \quad z_n = G(t, x(t_{n-k}), x(t_{n-k+1}), \dots, z_n)$$

i.e. for Euler ($x_{n+1} \approx x_n + h \cdot f(t_n, x_n)$)

$$\begin{aligned} e_{n+1} &= x(t_{n+1}) - z_{n+1} = x(t_{n+1}) - x(t_n) - h \cdot f(t_n, x(t_n)) \\ &= \frac{h^2}{2}\ddot{x}(\zeta), \quad \text{for } t_n < \zeta < t_{n+1} \end{aligned}$$

Note (Taylor):
$x(t_{n+1}) = x(t_n) + h \cdot f(t_n, x(t_n)) + \frac{h^2}{2} \cdot f'(t_n, x(t_n)) + O(3)$
→ local error proportional to h^2 and global error proportional to h (number of steps proportional to h^{-1}).
If local error $O(h^{k+1})$, k is the order of accuracy.

Computer-aided modeling
E. Witrant

Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- Stability is also crucial. i.e.

$$\begin{aligned} \dot{x} &= \lambda x, \quad \lambda \in \mathbb{C} \\ x(0) &= 1 \end{aligned}$$

with Euler: $x_{n+1} = x_n + h\lambda x_n = (1 + h\lambda)x_n$ has solution

$$x_n = (1 + h\lambda)^n.$$

It implies that

$$\begin{aligned} x_n &\to 0 \quad \text{if} \quad |1 + h\lambda| < 1 \\ |x_n| &\to \infty \quad \text{if} \quad |1 + h\lambda| > 1 \end{aligned}$$

stable if $R_e[\lambda] < 0$ AND $|1 + h\lambda| < 1$ (h small enough)
→ the stability of the DE does not necessarily coincides with the one of the numerical scheme!

Computer-aided modeling
E. Witrant

Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

The Runge-Kutta Methods:

Consider the integral form

$$x(t_{n+1}) = x(t_n) + \int_{t_n}^{t_{n+1}} f(\tau, x(\tau))d\tau$$

with central approximation

$$x_{n+1} = x_n + h \cdot f(t_n + \frac{h}{2}, x(t_n + \frac{h}{2}))$$

and (Euler) $x(t_n + \frac{h}{2}) \approx x_n + \frac{h}{2} f(t_n, x_n)$. Consequently, we have the simplest Runge-Kutta algorithm

$$\begin{aligned} k_1 &= f(t_n, x_n), \\ k_2 &= f(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_1), \\ x_{n+1} &= x_n + hk_2. \end{aligned}$$

Local error $x(t_{n+1}) - x_{n+1} = O(h^3)$ → 1 order of magnitude more accurate than Euler.

Computer-aided modeling
E. Witrant

Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- General form:

$$\begin{aligned} k_1 &= f(t_n, x_n), \\ k_2 &= f(t_n + c_2 h, x_n + ha_{21}k_1, \\ k_3 &= f(t_n + c_3 h, x_n + h(a_{31}k_1 + a_{32}k_2)), \\ &\vdots \\ k_s &= f(t_n + c_s h, x_n + h(a_{s1}k_1 + \dots + a_{s,s-1}k_{s-1})), \\ x_{n+1} &= x_n + h(b_1 k_1 + \dots + b_s k_s), \end{aligned}$$

where s, c_i, b_i and a_{ij} chosen to obtain the desired order of accuracy p, calculation complexity or other criterion → family of Runge-Kutta methods.

- A classic method sets $s = p = 4$ with

$$\begin{aligned} &c_2 = c_3 = \frac{1}{2},\ c_4 = 1,\ a_{21} = a_{32} = \frac{1}{2},\ a_{43} = 1, \\ &b_1 = b_4 = \frac{1}{6},\ b_2 = b_3 = \frac{2}{6},\ (others = 0) \end{aligned}$$

Adams' Methods:

- Family of multistep methods

$$x_n = x_{n-1} + \sum_{j=0}^{k} \beta_j f_{n-j}, \quad f_i = f(t_i, x_i)$$

where β_j chosen such that the order of accuracy is as high as possible. If $\beta_0 = 0$: explicit form (accuracy $k+1$), Adams-Bashforth, while $\beta_0 \neq 0$: implicit form (accuracy k), Adams-Moulton.

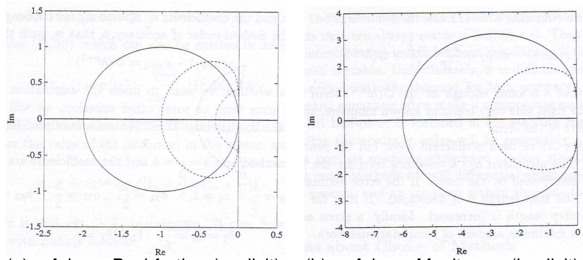
- Simplest explicit forms:

$$\begin{aligned}
k = 1: \; x_n &= x_{n-1} + f_{n-1}h \\
k = 2: \; x_n &= x_{n-1} + (3f_{n-1} - f_{n-2})\frac{h}{2} \\
k = 3: \; x_n &= x_{n-1} + (23f_{n-1} - 16f_{n-2} + 5f_{n-3})\frac{h}{12} \\
k = 4: \; x_n &= x_{n-1} + (55f_{n-1} - 59f_{n-2} + 37f_{n-3} - 9f_{n-4})\frac{h}{24}
\end{aligned}$$

- Simplest implicit forms:

$$\begin{aligned}
k = 1: \; x_n &= x_{n-1} + f_n h \\
k = 2: \; x_n &= x_{n-1} + (f_n + f_{n-1})h/2 \\
k = 3: \; x_n &= x_{n-1} + (5f_n + 8f_{n-1} - f_{n-2})h/12 \\
k = 4: \; x_n &= x_{n-1} + (9f_n + 19f_{n-1} - 5f_{n-2} + f_{n-3})h/24
\end{aligned}$$

- Why more complicated implicit methods?



(a) Adams-Bashforth (explicit), $k = 1$ (–), $k = 2$ (– –) and $k = 3$ (· · ·). (b) Adams-Moulton (implicit), $k = 2$ (–) and $k = 3$ (– –).

$\Rightarrow$ Larger stability regions. Note: $\nearrow k \searrow$ stability.

Variable Step Length:

- Fixed steps often inefficient $\rightarrow$ large steps when slow changes & small steps when rapid changes.
- Automatic adjustment based on local error approximation, i.e. assume a local error

$$x(t_{n+1}) - x_{n+1} = Ch^{p+1} + O(h^{p+2})$$

where C depends on the solution (unknown). If 2 steps of length h, we have approximately (errors are added)

$$x(t_{n+2}) - x_{n+2} = 2Ch^{p+1} + O(h^{p+2}) \qquad (1)$$

$\tilde{x} \doteq$ value computed for a step of length $2h$ from t_n to t_{n+2}:

$$\begin{aligned}
x(t_{n+2}) - \tilde{x} &= C(2h)^{p+1} + O(h^{p+2}) \qquad (2) \\
(2) - (1): \; x_{n+2} - \tilde{x} &= 2Ch^{p+1}(2^p - 1) + O(h^{p+2}) \qquad (3) \\
C \text{ from (3) in (1)}: \; x(t_{n+2}) - x_{n+2} &= \frac{x_{n+2} - \tilde{x}}{2^p - 1} + O(h^{p+2})
\end{aligned}$$

Previous result:

$$x(t_{n+2}) - x_{n+2} = \frac{x_{n+2} - \tilde{x}}{2^p - 1} + O(h^{p+2})$$

Assume $O(h^{p+2})$ negligible $\rightarrow$ known estimate of the error.

- The estimate can be used in several ways, in general:
  $\searrow h$ if error $>$ tolerance,
  $\nearrow h$ if error $\ll$ tolerance.
  Ideally, a given accuracy is obtained with minimum computational load.
- Crucial issue for embedded control and large-scale plants. Most of the time, use existing softwares/libraries.

Stiff differential equations:

- Both fast and slow components and large difference between the time constants, i.e.

$$\dot{x} = \begin{pmatrix} -10001 & -10000 \\ 1 & 0 \end{pmatrix} x$$
$$x(0) = \begin{pmatrix} 2 \\ -1.0001 \end{pmatrix}$$

has solution

$$x_1 = e^{-t} + e^{-10000t}$$
$$x_2 = -e^{-t} - 0.0001e^{-10000t}.$$

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

- Problem: in simulation, start with very small step to follow the fast term (i.e. $e^{-10000t}$), which soon goes to zero: solution only characterized by slow term. BUT $\nearrow h$ implies stability problems (i.e. $-10\,000 \cdot h$ within stability region).

$\Rightarrow$ use methods that are always stable: compromise with accuracy (implicit in general).

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Comments about Choice of Methods:

- Runge-Kutta most effective for low complexity (computational work) while Adams better for high complexity;
- methods for stiff problems - may be - ineffective for nonstiff problems;
- problem dependent.

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

Conclusions

- First step:
  - from the physical model (high order or bond graphs), write the system in a state-space form
  - investigate the behavior of the continuous dynamics, e.g. nonlinearities, time-delays, time constants of the linearized dynamics ...
- Second step:
  - discretize the dynamics to get computable difference equations
  - check on the impact of the discretization step
  - advanced methods with predictor/corrector schemes
- From experience:
  - hand-made discretizations are often more tractable
  - when doing “equivalent” model transformations, they are more equivalent in the discretized framework

Homework 4

a. Write the state-space description for:

- Example 1:

$$\begin{aligned} \ddot{y} + \dot{v}^2 + y &= 0 \\ \dot{y}^2 + \ddot{v} + vy &= 0 \end{aligned}$$

- Example 2:

$$\begin{aligned} \ddot{y} + v^3 + \dot{v}^2 + y &= 0 \\ \dot{y}^2 + \ddot{v} + vy &= 0 \end{aligned}$$

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

b. Numerical methods

TABLE 8.6.3 Exact Solution of $y'' - 10\pi^2 y = 0$, $y(0) = 1$, $y'(0) = -\sqrt{10}\pi$ and Numerical Solution Using the Runge–Kutta Method with $h = 0.01$.

t	y Numerical	y Exact
0.0	1.0	1.0
0.25	8.3439×10^{-2}	8.3438×10^{-2}
0.5	6.9631×10^{-3}	6.9620×10^{-3}
0.75	5.9390×10^{-4}	5.8089×10^{-4}
1.0	2.0437×10^{-4}	4.8469×10^{-5}
1.5	2.2394×10^{-2}	3.3744×10^{-7}
2.0	3.2166	2.3492×10^{-9}
2.5	4.6202×10^{2}	1.6356×10^{-11}
3.0	6.6363×10^{4}	1.1386×10^{-13}
3.5	9.5322×10^{6}	7.9272×10^{-16}
4.0	1.3692×10^{9}	5.5189×10^{-18}
4.5	1.9667×10^{11}	3.8422×10^{-20}
5.0	2.8249×10^{13}	2.6750×10^{-22}

Consider the differential equation

$$\begin{aligned} &y''(t) - 10\pi^2 y(t) = 0 \\ &y(0) = 1,\ \dot{y}(0) = -\sqrt{10}\pi \end{aligned}$$

1. Write this equation in state-space form.
2. Compute the eigenvalues.
3. Explain the difference between exact and numerical difference expressed in Table 8.6.3.

Computer-aided modeling
E. Witrant
Computer Algebra
Analytical Solutions
Algebraic Modeling
Automatic Bond Graphs Translation
Numerical Methods
Conclusions
Homework

References

1. L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
2. W.E. Boyce and R.C. Di Prima, *Elementary Differential Equations and Boundary Value Problems*, 6^{th} edition, John Wiley & Sons, Inc., 1997.

Modeling and Estimation for Control

Simulation

Lecture 6: Simulation with Scilab

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

April 22, 2014

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Models for simulation

- Choose the appropriate simulation tool/function depending on the class of model
- I.e. *Scilab* provides a wide array of tools for different models.
- Can use abbreviated commands and defaults parameters.
- Important to know appropriate tools, how the algorithms are set up and how to face difficulties.

Simulation tools

Three forms:

1. primary tools used by knowledgeable users on challenging problems;
2. simplified version easier to use and for simpler problems;
3. special cases occurring in specific areas of science and engineering.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Outline

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Ordinary differential equations (ODEs)

$$\dot{y} \;=\; f(t, y), \quad y(t_0) = y_0$$

where y, f vector valued, and $t \in \mathbb{R}$.

- Higher order models can always be transformed into 1^{st} order and directly simulated in *Scilab*, except *Boundary value problems*.
- Unique solution if f and $\partial f/\partial y$ continuous.
- The most continuous derivatives of $f(t, y)$ exist, the more derivatives y has. In simulation, accuracy obtained from error estimates that are based on derivatives.
- Controlled differential equation (DE):

  $$\dot{y} = f(t, y, u(t))$$

  y has only one more derivative than u → may create problems for piecewise continuous inputs.

Simulating ODEs: simplest call

`y=ode(y0,t0,t,f)`

- t_0, y_0, $f(t,\ y)$ → default method and error tolerance, adjust step size;
- many more solutions than needed: specify also final time vector `t`;
- returns `y`$= [y(t_0),\ y(t_1), \ldots,\ y(t_n)]$;
- online function definition, i.e. $f(t,\ y) = -y + \sin(t)$

```
function ydot = f(t,y)
    ydot=-y+sin(t)
endfunction
```

- interface to ode solvers like ODEPACK.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating ODEs: advanced call

```
[y,w,iw]=ode([type],y0,t0,t [,rtol [,atol]],f [,jac]
 ... [w,iw])
```

- "type": `lsoda` (default, automatically selects between nonstiff predictor-corrector Adams and stiff backward difference formula BDF), `adams`, `stiff` (BDF), `rk` (adaptive Runge-Kutta of order 4), `rkf` (RK 45, highly accurate), `fix` (simpler rkf), `root` (lsodar with root finding), `discrete`.
- "rtol, atol": real constants or vectors, set absolute and relative tolerance on y: $\epsilon_y(i) = rtol(i) * |y(i)| + atol(i)$, computational time vs. accuracy.
- "jac": external, analytic Jacobian (for BDF and implicit) `J=jac(t,y)`.
- "w,iw": real vectors for storing information returned by integration routine.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating ODEs: more options

```
odeoptions[itask,tcrit,h0,hmax,hmin,jactyp,mxstep,
maxordn,maxords,ixpr,ml,mu]
```

- sets computation strategy, critical time, step size and bounds, how nonlinear equations are solved, number of steps, max. nonstiff and stiff order, half-bandwidths of banded Jacobian.
- computational time and accuracy can vary greatly with the method.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating ODEs: Implicit differential equations

- $A(t, y)\dot{y} = g(t, y)$, $y(t_0) = y_0$. If A not invertible $\forall(t, y)$ of interest → implicit DAE, if invertible → linearly implicit DE or index-zero DAE.
- Better to consider directly than inverting A (more efficient and reliable integration).

```
y=impl([type],y0,ydot0,t0,t [,atol, [rtol]],res,adda
... [,jac])
```

→ requires also $\dot{y}(t_0)$ and to compute the residuals $(g(t, y) - A(t, y)\dot{y})$ as: `r=res(t,y,ydot)`

Simulating ODEs: Linear systems

- number of specialized functions for

$$\begin{aligned} \dot{x} &= Ax + Bu, \quad x(0) = x_0, \\ y &= Cx + Du. \end{aligned}$$

- `[sl]=syslin(dom,A,B,C [,D [,x0] ])` defines a continuous or discrete (`dom`) state-space system, system values recovered using `[A,B,C,D]=abcd(sl);`
- `[y [,x]]=csim(u,t,sl,[x0])` → simulation (time response) of linear system.

Simulating ODEs: Root finding

- to simulate a DE up to the time something happens;
- `y,rd[,w,iw]=ode(`root",y0,t0,t [,rtol [,atol]],f [,jac],ng,g [,w,iw])" integrate ODE f until $g(t, y) = 0$;
- iteratively reduces the last step to find surface crossing.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Boundary value problems (BVPs)

- DE with information given at 2 or more times:

$$\begin{aligned} \dot{y} &= f(t, y), \ t_0 \le t \le t_f, \\ 0 &= B(y(t_0), \ y(t_f)). \end{aligned}$$

  If y is n-dimensional → n boundaries.
- More complicated than initial value problems (cf. Optimization class), where local algorithm move from one point to the next.
- BVP: need more global algorithm with full t interval → much larger system of equations.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating BVPs: Numerous methods

1. shooting methods: take given IC then guess the rest and adjust by integrating the full interval → easy to program but not reliable on long intervals and stiff problems;
2. multiple shooting: breaks time interval into subinterval and shoot over these;
3. discretize the DE and solve the large discrete system, i.e. Euler with step h on $\dot{y} = f(t, y)$, $t_0 \le t \le t_f$, $0 = B(y(t_0), y(t_f))$ gives:

$$\begin{aligned} y_{i+1} - y_i - f(t_0 + ih, y_i) &= 0, \quad i = 0, \ldots, N-1, \\ B(y_0, y_N) &= 0. \end{aligned}$$

usually with more complicated methods than Euler but large system of (nonlinear) DE → BVP solver has to deal with numerical problems and need Jacobian-like information.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating BVPs: COLNEW

Scilab uses Fortran COLNEW code in `bvode`, which assumes that the BVP is of the form

$$\begin{aligned} \frac{d^{m_i} u_i}{dx^{m_i}} &= f_i\left(x, u(x), \frac{du}{dx}, \ldots, \frac{d^{m_i-1} u}{dx^{m_i-1}}\right), \ 1 \le i \le n_c, \\ g_i\left(\zeta_j, u(\zeta_j), \ldots, \frac{d^{m_*} u}{dx^{m_*}}\right) &= 0, \ j = 1, \ldots, m_*, \end{aligned}$$

where ζ_j are x where BC hold and $a_L \le x \le a_R$.
Let $m_* = m_1 + m_2 + \cdots + m_{n_c}$, $z(u) = \left[u, \frac{du}{dx}, \ldots, \frac{d^{m_*} u}{dx^{m_*}}\right]$, then

$$\begin{aligned} \frac{d^{m_i} u_i}{dx^{m_i}} &= f_i\left(x, z(u(x))\right), \ 1 \le i \le n_c, a_L \le x \le a_R \\ g_i\left(\zeta_j, z(u(\zeta_j))\right) &= 0, \ j = 1, \ldots, m_*, \end{aligned}$$

`bvode` starts with initial mesh, solve NL system and iteratively refines the mesh.

Simulating BVPs: COLNEW implementation

```
[z]=bvode(points,ncomp,m,aleft,aright,zeta,ipar,ltol,
...tol,fixpnt,...fsub1,dfsub1,gsub1,dgsub1,guess1)
```

- solution `z` evaluated at the given `points` for `ncomp`≤ 20 DE;
- we have to provide bounds (`aleft`,`aright`) for u, BCs and numerical properties of the model.

Simulating BVPs: Example - optimal control

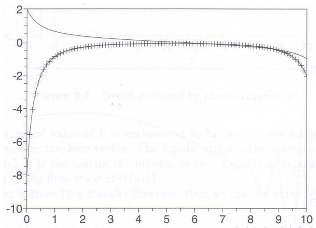
Necessary conditions: consider the NL controlled system

$$\dot{y} = y^2 + v, \quad J(y,u) = \int_0^{10} 10v^2 + y^2\, dt$$

Find $v : y(0) = 2 \rightarrow y(10) = -1$, while min J. NC found from Hamiltonian and give the BV DAE

$$\begin{array}{rcl} \dot{y} & = & y^2 + v, \\ \dot{\lambda} & = & -2y - 2\lambda y, \\ 0 & = & 20v + \lambda, \\ y(0) & = & 2,\ y(10) = -1. \end{array} \Rightarrow BVP : \left\{ \begin{array}{rcl} \dot{y} & = & y^2 - \lambda/20, \\ \dot{\lambda} & = & -2y - 2\lambda y, \\ y(0) & = & 2,\ y(10) = -1. \end{array} \right.$$

Ready to be solved by `bvode`, which gives:



Difference equations

- Discrete-time values or values changing only at discrete times, for discrete processes or because of isolated observations.
- Integer variable k and sequence $y(k)$ that solves

$$y(k+1) = f(k, y(k)), \quad y(k_0) = y_0,$$

or with time sequence t_k, $k \geq k_0$:

$$z(t_{k+1}) = g(t_k, z(t_k)), \quad z(t_{k0}) = z_0.$$

If evenly spaced events $t_{k+1} - t_k = h = cst$:

$$\begin{array}{rcl} v(k+1) & = & g(w(k), v(k)), \quad v(k_0) = v_0, \\ w(k+1) & = & w(k) + h, \quad w(k_0) = t_{k_0} \end{array}$$

Difference equations (2)

- Solution existence simpler than DE: $y(k)$ computed recursively from $y(k_0)$ as long as $(k, y(k)) \in \mathcal{D}_f$.
- Note: uniqueness theorem for DE (if 2 solutions start at the same time but with different y_0 and if continuity of f, f_y holds, then they never intersect) not true for difference equations.
- Can always be written as 1^{st} order difference equations.

Simulating difference equations

1. Easier because no choice about time step and no error from derivatives approximations → only function evaluation and roundoff errors.
2. First order $y(k+1) = f(k, y(k))$, $y(k_0) = y_0$, evaluated by `y=ode(discrete",y0,k0,kvect,f)"` where `kvect` = evaluation times.
3. *Linear systems*

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \quad x(0) = x_0, \\ y(k) &= Cx(k) + Du(k), \end{aligned}$$

- `[X]=ltitr(A,B,U,[x0])` or `[xf,X]=ltitr(A,B,U,[x0]);`
- If given by a transfer function `[y]=rtitr(Num,Den,u [,up,yp])` where `[,up,yp]` are past values, if any;
- Time response obtained using `[y [,x]]=flts(u,sl [,x0])`.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Differential algebraic equations (DAEs)

- Most physical models are differential + algebraic (DAEs):

$$F(t, y, \dot{y}) = 0$$

→ rewrite as ODE or simpler DAE, or simulate the DAE directly.

- Theory much more complex than ODEs: ∃ solutions only for certain IC, called *consistent* IC, i.e.

$$\begin{aligned} \dot{y}_1 &= y_1 - \cos(y_2) + t, \\ 0 &= y_1^3 + y_2 + e^t, \end{aligned}$$

requires $y_1(t_0)^3 + y_2(t_0) + e^{t_0} = 0$.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Differential algebreic equations (2)

- Structure → *index* definition (≥ 0, 0 for ODE). Index-one DAE in *Scilab*: $F(t, y, \dot{y}) = 0$ with $\{F_{\dot{y}}, F_y\}$ is an index-one matrix pencil along solutions and $F_{\dot{y}}$ has constant rank:
  1. *implicit semiexplicit:*

$$\begin{aligned} F_1(t, y_1, y_2, \dot{y}_1) &= 0 \\ F_2(t, y_1, y_2) &= 0 \end{aligned}$$

  where $\partial F_1/\partial \dot{y}_1$ and $\partial F_2/\partial y_2$ nonsingular, y_1 is the differential variable and y_2 the algebraic one;
  2. *semiexplicit:*

$$\begin{aligned} \dot{y}_1 &= F_1(t, y_1, y_2) \\ 0 &= F_2(t, y_1, y_2) \end{aligned}$$

  with $\partial F_2/\partial y_2$ nonsingular.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating DAEs

- Need information on both $y(t_0)$ and $\dot{y}(t_0)$ to uniquely determine the solution and start integration, i.e. $\tan(\dot{y}) = -y + g(t)$ → family of DE $\dot{y} = \tan^{-1}(-y + g) + n\pi$. Sometimes approximate value of $\dot{y}(t_0)$ or none at all.
- Scilab uses backward differentiation formulas (BDF), i.e. backward Euler on $F(t, y, \dot{y}) = 0$ gives

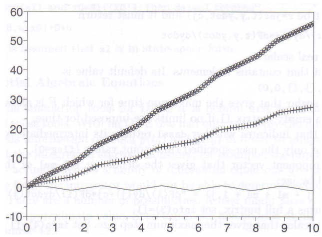
$$F\left(t_{n+1}, y_{n+1}, \frac{y_{n+1} - y_n}{h}\right) = 0$$

→ given y_n, iterative resolution using the Jacobian w.r.t. y_{n+1}: $F_y + \frac{1}{h} F_{y'}$.

- based on DASSL code (for nonlinear fully implicit index-one DAEs): `[r [,hd]]=dassl(x0,t0,t [,atol,[rtol]],res [,jac]` where `x0` is `y0 [ydot0]`, `res` returns the residue `r=g(t,y,ydot)` and `info` sets computation properties.

Example $\tan(\dot{y}) = -y + 10t\cos(3t)$, $y(0) = 0$

$\dot{y}(0) = n\pi$ is a consistent IC → compare $\dot{y}(0) = \{0, \pi, 2\pi\}$



implicit ODEs and fully implicit DAEs can have multiple nearby roots → integrators must ensure no jump on another solution when making a step (conservatism in the step size choice).

DAEs and root-finding:

`[r,nn,[,hd]]=dasrt(x0,t0,t [,atol,[rtol]] ...,res [,jac],ng, surf [,info] [,hd])`: just add intersection surface.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Hybrid systems

- Mixture of continuous- and discrete-time events.
- When an event (discrete variable change) occurs: change in DE, state dimension, IC (initialization problem)...
- Interfere with error control of integrators.
- Handled in *Scilab* and more particularly *Scicos*.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Simulating Hybrid systems

- Continuous variable y_c and discrete variable y_d (piecewise constant on $[t_k, t_{k+1}[$):

$$\begin{aligned} \dot{y}_c(t) &= f_0(t, y_c(t), y_d(t)),\ t \in [t_k, t_{k+1}[\\ y_d(t_{k+1}) &= f_1(t, y_c(t_{k+1}), y_d(t_k)) \text{ at } t = t_{k+1} \end{aligned}$$

  i.e. sampled data system (u is a control function):

$$\begin{aligned} \dot{y}_c(t) &= f_0(t, y_c(t), u(t)),\ t \in [t_k, t_{k+1}[, \\ u(t_{k+1}) &= f_1(t, y_c(t_{k+1}), u(t_k)) \text{ at } t = t_{k+1}. \end{aligned}$$

- `yt=odedc(y0,nd,stdel,t0,t,f)`, where `y0=[y0c;y0d]`, `stdel=[h, delta]` with `delta`=delay/h, `yp=f(t,yc,yd,flag)`.

Computer-aided modeling
E. Witrant
Ordinary differential equations
Simulating ODEs
advanced call
more options
Implicit diff eq
Linear systems
Boundary value problems
Simulating BVPs
COLNEW
optimal control
Difference equations
Simulation
Differential algebraic equations
Simulation
Example
Hybrid systems
Simulation
Conclusions

Conclusions

- Implementing the equations needs some dedicated thinking
- Need to understand the expected results prior to computation
- Trade-off:
  - computation time vs. precision
  - mathematical simplicity vs. code efficiency
- Particularly challenging for real-time modeling
- A code aims to be transmitted to other people: the structure and comments have to be clear!

References

1. S. Campbell, J-P. Chancelier and R. Nikoukhah, *Modeling and Simulation in Scilab/Scicos*, Springer, 2005.
2. Scilab website: *http://www.scilab.org*.

Modeling and Estimation for Control
System Identification

Lecture 7: Signals for System Identification

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

April 23, 2014

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

Basics of System Identification

System identification = use of data in modeling

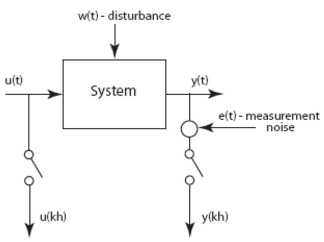
- Include experimental data in modeling work.
- Used to find constants or complete the model.
- Based on system variables: inputs, outputs and possibly disturbances.

$\rightarrow$ understand how the system works, describe partial systems and compute values of the constants.

- Three different ways to use identification for modeling:
  1. make simple experiments to facilitate problem structuration (phase 1);
  2. describe I/O relationships independently of physical insights (often linear);
  3. use data to determine unknown parameters in physical models: *tailor-made models*.

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

Estimate system from measurements of $u(t)$ and $y(t)$



Many issues:

- choice of sampling frequency, input signal (experiment conditions);
- what class of models, how to model disturbances?
- estimating model parameters from sampled, finite and noisy data.

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

Outline

$\Rightarrow$ Introduction to signal analysis and processing

From Continuous Dynamics to Sampled Signals

Continuous-time signals and systems

Continuous-time signal	$y(t)$
Fourier transform	$Y(\omega) = \int_{-\infty}^{\infty} y(t)e^{-i\omega t}dt$
Laplace transform	$Y(s) = \int_{-\infty}^{\infty} y(t)e^{-st}dt$
Linear system	$y(t) = g * u(t)$
	$Y(\omega) = G(\omega)U(\omega)$
	$Y(s) = G(s)U(s)$

Derivation operator $p \times u(t) = \dot{u}(t)$ works as s-variable, but in time domain.

Example (0 IC)

$$\begin{aligned} y(t) &= 0.5\dot{u}(t) + u(t) \\ y(t) &= (0.5p + 1)u(t) \\ Y(s) &= (0.5s + 1)U(s) \end{aligned}$$

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

Discrete-time signals and systems

Discrete-time signal	$y(kh)$
Fourier transform	$Y^{(h)}(\omega) = h\sum_{k=-\infty}^{\infty} y(kh)e^{-i\omega kh}$
z-transform	$Y(z) = \sum_{k=-\infty}^{\infty} y(kh)z^{-k}$
Linear system	$y(kh) = g * u(kh)$
	$Y^{(h)}(\omega) = G_d(e^{i\omega h})U^{(h)}(\omega)$
	$Y(z) = G_d(z)U(z)$

Shift operator $q \times u(kh) = u(kh + h)$ works as z-variable, but in time-domain.

Example (0 IC)

$$\begin{aligned} y(kh) &= 0.5u(kh) + u(kh - h) \\ y(kh) &= (0.5 + q^{-1})u(kh) \\ Y(z) &= (0.5 + z^{-1})U(z) \end{aligned}$$

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

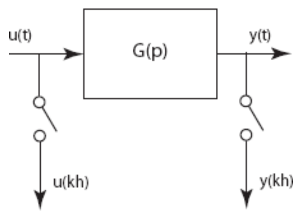
Sampled systems

Continuous-time linear system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

$\Rightarrow G(s) = C(sI - A)^{-1}B + D.$

Assume that we sample the inputs and outputs of the system



Relation between sampled inputs $u[k]$ and outputs $y[k]$?

Signals for System Identification
E. Witrant
Basics of System Identification
Sampled Signals
Discrete-time
Sampled systems
Sampling
Disturbance Modeling
Discrete-time stochastic processes
Some background
Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum
Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect
Conclusions

Sampled systems (2)

Systems with piecewise constant input:

- Exact relation possible if $u(t)$ is constant over each sampling interval.
- Solving the system equations over one sampling interval gives

$$\begin{aligned} x[k+1] &= A_d x[k] + B_d u[k] \\ y[k] &= Cx[k] + Du[k] \\ G_d(z) &= C(zI - A_d)^{-1}B_d + D \end{aligned}$$

where $A_d = e^{Ah}$ and $B_d = \int_0^h e^{As}Bds$.

Sampled systems (3)

Example: sampling of scalar system

- Continuous-time dynamics

$$\dot{x}(t) = ax(t) + bu(t)$$

- Assuming that the input $u(t)$ is constant over a sampling interval

$$x[k+1] = a_d x[k] + b_d u[k]$$

where $a_d = e^{ah}$ and $b_d = \int_0^h e^{as} b ds = \frac{b}{a}(e^{ah} - 1)$.
- Note: continuous-time poles in $s = a$, discrete-time poles in $z = e^{ah}$.

Sampled systems (4)

Frequency-domain analysis of sampling

- Transfer function of sampled system

$$G_d(z) = C(zI - A_d)^{-1} B_d + D$$

produces same output as $G(s)$ at sampling intervals.
- However, frequency responses are not the same! One has

$$|G(i\omega) - G_d(e^{i\omega h})| \leq \omega h \int_0^\infty |g(\tau)| d\tau$$

where $g(\tau)$ is the impulse response for $G(s)$.

$\Rightarrow$ Good match at low frequencies ($\omega < 0.1\omega_s$) $\Rightarrow$ choose sampling frequency > 10× system bandwidth.

Sampling of general systems

- For more general systems,
  - nonlinear dynamics, or
  - linear systems where input is not piecewise constant

  conversion from continuous-time to discrete-time is not trivial.
- Simple approach: approximate time-derivative with finite difference:

$$p \approx \frac{1 - q^{-1}}{h} \quad \text{Euler backward}$$

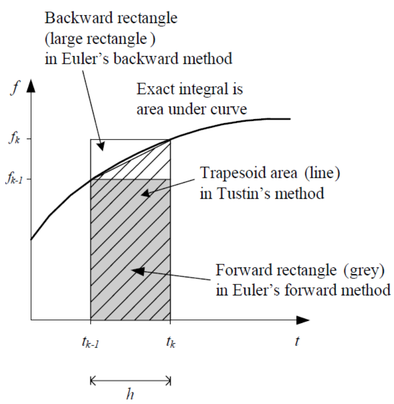
$$p \approx \frac{q - 1}{h} \quad \text{Euler forward}$$

$$p \approx \frac{2}{h} \times \frac{q-1}{q+1} \quad \text{Tustin's approximation (typical for linear systems)}$$

...

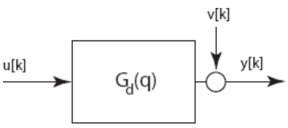
- I.e. write $x(t_k) = x(t_k - 1) + \int_{t_k-1}^{t_k} f(\tau)\, d\tau$ and find the previous transformations using different integral approximations

Numerical approximations of the integral [F. Haugen'05]



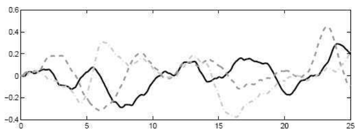
Disturbance Modeling

- Discrete-time set-up:



$\Rightarrow$ Estimate G_d from measurements $y[k]$ and $u[k]$. The effect of disturbances is crucial, need for a disturbance model!

- Basic observations:



  - disturbances are different from time to time
  - some characteristics (e.g., frequency content) persist

- Can be captured by describing disturbances as filtered white noise $v(k) = H(q)e(k)$

Discrete-time stochastic processes

- Discrete-time stochastic process: an infinite sequence $\{v(k,\theta)\}$ whose values depend on a random variable θ
- To each fixed value θ^* of θ, the sequence $\{v(k,\theta^*)\}$ depends only on k and is called a realization of the stochastic process
- For a discrete-time stochastic process $v[k]$, we define its

  Expected or mean value $\quad m_v(k) = E_\theta\{v[k]\}$

  Auto-correlation function $\quad R_v(k,l) = E_\theta\{v[k+l]v[k]\}$

  and say that $v[k]$ is

  stationary — if m_v and R_v are independent of k

  ergodic — if m_v and R_v can be computed from a single realization

Some background

- define the real random variable e the possible outcomes of unpredictable experiment;
- define $f_e(x)$ the probability density function:

$$P(a \le e < b) = \int_a^b f_e(x)dx$$

- the expectation is

$$E_e = \int_{\mathbb{R}} x f_e(x)dx \quad \text{or (discrete)} \quad E_e = \sum x_i P[X = x_i]$$

- the covariance matrix is

$$\begin{aligned} Cov(e,y) &= E[(e-E(e))(y-E(y))^T] = E(ey) - E(e)E(y) \\ &= \sum_{i,j}(e_i - E(e))(y_j - E(y))P[e = e_i, y = y_j] \quad \text{(discrete)} \end{aligned}$$

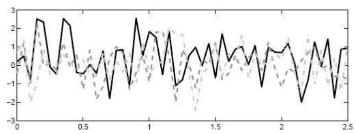
Some background (2)

White noise:

- A stochastic process $e[k]$ is called *white noise*

if $m_e = 0$ and

$$R_e(k,l) = \begin{cases} \sigma^2 & \text{if } l = 0 \\ 0 & \text{otherwise} \end{cases}$$

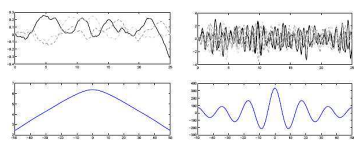


Unpredictable sequence!

Signals and auto-correlation function (ACF)

- Different realizations may look very different.
- Still, qualitative properties captured as:

- slowly varying ACF $\leftrightarrow$ slowly varying process;
- quickly varying ACF $\leftrightarrow$ quickly varying process.



- Close to white noise if $R(l) \to 0$ rapidly as $|l|$ grows.

Some background (3)

Properties of the auto-correlation function [Wikipedia]

- Symmetry: ACF is even ($R_f(-l) = R_f(l)$ if $f \in \mathbb{R}$) or Hermitian (conjugate transpose, $R_f(-l) = R_f^*(l)$ if $f \in \mathbb{C}$)
- Peak at the origin ($|R_f(l)| \leq R_f(0)$) and the ACF of a periodic function is periodic with the same period (dirac at 0 if white noise)
- $\sum$ uncorrelated functions (0 cross-correlation $\forall l$) = $\sum$ ACF of each function
- Estimate: for discrete process $\{X_1, X_2, \ldots, X_n\}$ with known mean μ and variance σ:

$$R(l) \approx \frac{1}{(n-l)\sigma^2} \sum_{t=1}^{n-l} (X_t - \mu)(X_{t+k} - \mu), \quad \forall l < n \in \mathbb{N}^+$$

  - unbiased if true μ and σ
  - biased estimate if sample mean and variance are used
  - can split the data set to separate the μ and σ estimates from the ACF estimate

Signal Spectra

A common framework for deterministic and stochastic signals

- Signals typically described as stochastic processes with deterministic components (det. inputs vs. stoch. disturbances).
- For a linear system with additive disturbance $e(t)$ (sequence of independent random variables with $m_e(t) = 0$ and variances σ^2)

$$y(t) = G(q)u(t) + H(q)e(t)$$

  we have that

$$Ey(t) = G(q)u(t)$$

  so $y(t)$ is not a stationary process.

Quasi-Stationary Signals (QSS)

Definition: $s(t)$ is QSS if

1. $Es(t) = m_s(t)$, $|m_s(t)| \leq C$, $\forall t$ (bounded mean)
2. $Es(t)s(r) = R_s(t, r)$, $|R_s(t, r)| \leq C$, and the following limit exists

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} R_s(t, t-\tau) = R_s(\tau), \forall \tau \quad \text{(bounded autocor.)}$$

where E is with respect to the stochastic components of $s(t)$.

Quasi-Stationary Signals (2)

- If $\{s(t)\}$ deterministic then $\{s(t)\}$ is a bounded sequence such that

$$R_s(\tau) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} s(t)s(t-\tau)$$

  exists (E has no effect).
- If $\{s(t)\}$ stationary, the bounds are trivially satisfied and $R_s(\tau)$ do not depend on t.
- Two signals $\{s(t)\}$ and $\{w(t)\}$ are jointly quasi-stationary if both QSS and if the *cross-covariance*

$$R_{sw}(\tau) = \bar{E}s(t)w(t-\tau), \text{ with } \bar{E}f(t) \doteq \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} Ef(t), \text{exists.}$$

- Uncorrelated signals if $R_{sw}(\tau) \equiv 0$.

Definition of Spectra

- Power spectrum of $\{s(t)\}$ (freq. content of stoch. process, always real):

$$\phi_s(\omega) = \sum_{\tau=-\infty}^{\infty} R_s(\tau) e^{-i\tau\omega}$$

  e.g. for white noise $\phi_s(\omega) = \sigma^2$: same power at all frequencies.

- Cross-spectrum between $\{w(t)\}$ and $\{s(t)\}$ (measures how two processes co-vary, in general complex):

$$\phi_{sw}(\omega) = \sum_{\tau=-\infty}^{\infty} R_{sw}(\tau) e^{-i\tau\omega}$$

  $\Re(\phi_{sw}) \rightarrow$ *cospectrum*, $\Im(\phi_{sw}) \rightarrow$ *quadrature spectrum*, $\arg(\phi_{sw}) \rightarrow$ *phase spectrum*, $|\phi_{sw}| \rightarrow$ *amplitude spectrum*.

Definition of Spectra (2)

- From the definition of inverse Fourier transform:

$$\bar{E}s^2(t) = R_s(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \phi_s(\omega) d\omega$$

- Example (stationary stochastic process): for the process $v(t) = H(q)e(t)$, the spectrum is $\phi_v(\omega) = \sigma^2 |H(e^{i\omega})|^2$.
- Example (spectrum of a mixed det. and stoch. signal): for the signal

$$s(t) = u(t) + v(t),$$

  where $\{u(t)\}$ deterministic and $\{v(t)\}$ stationary stochastic process, the spectrum is $\phi_s(\omega) = \phi_u(\omega) + \phi_v(\omega)$.

Transformation of spectrum by linear systems

- **Theorem:** Let $\{w(t)\}$ QSS with spectrum $\phi_w(\omega)$, $G(q)$ stable and $s(t) = G(q)w(t)$. Then $\{s(t)\}$ is also QSS and

$$\begin{aligned} \phi_s(\omega) &= |G(e^{i\omega})|^2 \phi_w(\omega) \\ \phi_{sw}(\omega) &= G(e^{i\omega})\phi_w(\omega) \end{aligned}$$

- **Corollary:** Let $y(t)$ given by

$$y(t) = G(q)u(t) + H(q)e(t)$$

  where $\{u(t)\}$ QSS, deterministic with spectrum $\phi_u(\omega)$, and $\{e(t)\}$ white noise with variance σ^2.
  Let G and H be stable filters, then $\{y(t)\}$ is QSS and

$$\begin{aligned} \phi_y(\omega) &= |G(e^{i\omega})|^2 \phi_u(\omega) + \sigma^2 |H(e^{i\omega})|^2 \\ \phi_{yu}(\omega) &= G(e^{i\omega})\phi_u(\omega) \end{aligned}$$

$\Rightarrow$ We can use filtered white noise to model the character of disturbances!

Spectral factorization

- The previous theorem describes spectrum as real-valued rational functions of $e^{i\omega}$ from transfer functions $G(q)$ and $H(q)$.
  In practice: given a spectrum $\phi_v(\omega)$, can we find $H(q)$ s.t. $v(t) = H(q)e(t)$ has this spectrum and $e(t)$ is white noise? Exact conditions in [Wiener 1949] & [Rozanov 1967]
- Spectral factorization: suppose that $\phi_v(\omega) > 0$ is a rational function of $\cos(\omega)$ (or $e^{i\omega}$), then there exists a monic rational function (leading coef. = 1) of z, $H(z)$, without poles or zeros on or outside the unit circle, such that:

$$\phi_v(\omega) = \sigma^2 |H(e^{i\omega})|^2$$

Spectral factorization (SF): ARMA process example

If a stationary process $\{v(t)\}$ has a rational spectrum $\phi_v(\omega)$, we can represent it as $v(t) = H(q)e(t)$, where

$$H(q) = \frac{C(q)}{A(q)} = \frac{1 + c_1 q^{-1} + \cdots + c_{n_c} q^{-n_c}}{1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}}.$$

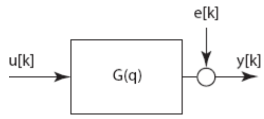
We may write the ARMA model:

$$\begin{aligned} v(t) + a_1 v(t-1) + \cdots + a_{n_a} v(t-n_a) \\ = e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c) \end{aligned}$$

- if $n_c = 0$, autoregressive (AR) model:
  $v(t) + a_1 v(t-1) + \cdots + a_{n_a} v(t-n_a) = e(t)$,
- if $n_a = 0$, moving average (MA) model:
  $v(t) = e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c)$.

⇒ SF provides a representation of disturbances in the standard form $v = H(q)e$ from information about its spectrum only.

Filtering and spectrum



- Consider the general set-up with $u(k)$ and $e(k)$ uncorrelated:

$$\begin{aligned} \phi_y(\omega) &= |G(e^{i\omega})|^2 \phi_u(\omega) + \phi_e(\omega) \\ \phi_{yu}(\omega) &= G(e^{i\omega}) \phi_u(\omega) \end{aligned}$$

- Note:
  - power spectrum additive if signals are uncorrelated
  - cross correlation can be used to get rid of disturbances

Choice of Sampling Interval and Presampling Filters

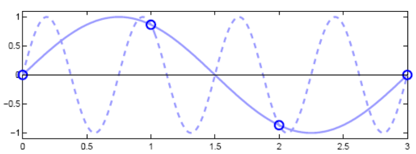
Sampling is inherent to computer-based data-acquisition systems → select (equidistant) sampling instances to minimize information losses.

Aliasing

Suppose $s(t)$ with sampling interval T: $s_k = s(kT)$, $k = 1, 2, \ldots$, sampling frequency $\omega_s = 2\pi/T$ and *Nyquist (folding) frequency* $\omega_N = \omega_s/2$.
If sinusoid with $|\omega| > \omega_N$, $\exists\, \bar{\omega}$, $-\omega_N \leq \bar{\omega} \leq \omega_N$, such that

$$\begin{aligned} \cos \omega kT &= \cos \bar{\omega} kT \\ \sin \omega kT &= \sin \bar{\omega} kT \end{aligned}$$

$$k = 0, 1, \ldots$$



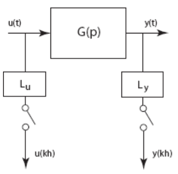
Aliasing (2)

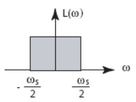
⇒ Alias phenomenon: part of the signal with $\omega > \omega_N$ interpreted as lower frequency ↔ spectrum of sampled signal is a superposition (folding) of original spectrum:

$$\phi_s^{(T)}(\omega) = \sum_{r=-\infty}^{\infty} \phi_s^c(\omega + r\omega_s)$$

where ϕ_s^c and $\phi_s^{(T)}$ correspond to continuous-time and sampled signals.
To avoid aliasing: choose ω_s so that $\phi_s^c(\omega)$ is zero outside $(-\omega_s/2, \omega_s/2)$. This implies $\phi_s^{(T)}(\omega) = \phi_s^c(\omega)$.

Antialiasing presampling filters





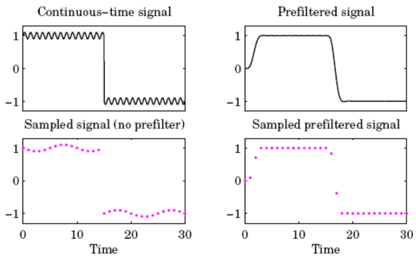
- We loose signals above ω_N, do not let folding effect distort the interesting part of spectrum below ω_N → presampling filters $\kappa(p)$: $s_F(t) = \kappa(p)s(t) \Rightarrow \phi^c_{sF}(\omega) = |\kappa(i\omega)|^2\phi^c_s(\omega)$
- Ideally, $\kappa(i\omega)$ s.t. $\begin{cases} |\kappa(i\omega)| = 1, & |\omega| \leq \omega_N \\ |\kappa(i\omega)| = 0, & |\omega| > \omega_N \end{cases}$ which means that $s^F_k = s_F(kT)$ would have spectrum

$$\phi^{(T)}_{sF}(\omega) = \phi^c_s(\omega), \quad -\omega_N \leq \omega < \omega_N$$

Antialiasing presampling filters (2)

⇒ Sampled spectrum without alias thanks to antialiasing filter, which should always be applied before sampling if significant energy above ω_N.

- Example - Continuous-time signal: square wave plus high-frequency sinusoidal



Noise-reduction effect of antialiasing (AA) filters

- Typically, signal = useful part $m(t)$ + disturbances $v(t)$ (more broadband, e.g. noise): choose ω_s such that most of the useful spectrum below ω_N. AA filters cuts away HF.
- Suppose $s(t) = m(t) + v(t)$ and sampled, prefiltered signal $s^F_k = m^F_k + v^F_k$, $s^F_k = s_F(kT)$. Noise variance:

$$E(v^F_k)^2 = \int_{-\omega_N}^{\omega_N} \phi^{(T)}_{v_F}(\omega)d\omega = \sum_{r=-\infty}^{\infty} \int_{-\omega_N}^{\omega_N} \phi^c_{v_F}(\omega + r\omega_s)d\omega$$

→ noise effects from HF folded into region $[-\omega_N, \omega_N]$ and introduce noise power. Eliminating HF noise by AA filter, variance of v^F_k is thus reduced by

$$\sum_{r\neq 0} \int_{-\omega_N}^{\omega_N} \phi^c_v(\omega + r\omega_s)d\omega = \int_{|\omega|>\omega_N} \phi^c_v(\omega)d\omega$$

compared to no presampling filter.

⇒ ↘ noise if spectrum with energy above ω_N.

Conclusions

- First step to modeling and identification = data acquisition
- Implies computer-based processing and sampled signal
- Models including both deterministic and stochastic components
- Characterize the spectrum for analysis and processing
- Prepare experimental signal prior to the identification phase

Homework

Spectrum of a sinusoid function:

$$u(t) = A\cos(\omega_0 t)$$

1. Show that $u(t)$ is a quasi-stationary signal by computing the bound $R_u(\tau)$.
2. Show that the power spectrum $\phi_u(\omega)$ is composed of two Dirak δ functions.

Hint - you may wish to use the identities:

$$\cos\theta + \cos\phi = 2\cos\left(\frac{\theta+\phi}{2}\right)\cos\left(\frac{\theta-\phi}{2}\right)$$

$$\cos(\omega_0\tau) = \frac{1}{2}\left(e^{i\omega_0\tau} + e^{-i\omega_0\tau}\right)$$

$$\delta(x) = \frac{1}{2\pi}\sum_{n=-\infty}^{\infty} e^{inx}$$

Signals for System Identification

E. Witrant

Basics of System Identification

Sampled Signals
Discrete-time
Sampled systems
Sampling

Disturbance Modeling
Discrete-time stochastic processes
Some background

Signal Spectra
Quasi-Stationary Signals
Definition of Spectra
Transformation by linear systems
Spectral factorization
Filtering and spectrum

Sampling Interval and Filters
Aliasing
Antialiasing filters
Noise-reduction effect

Conclusions

References

1. L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.
2. L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
3. Finn Haugen, *Discrete-time signals and systems*, TechTeach, 2005. `http://techteach.no/publications/discretetime_signals_systems/discrete.pdf`

Modeling and Estimation for Control
System Identification

Lecture 8: Non-parametric Identification

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

September 20, 2017

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Class goal

Linear time-invariant model

- described by transfer functions or impulse responses
- determine such functions directly, without restricting the set of models
- non-parametric: do not explicitly employ finite-dimensional parameter vector in the search
- focus on determining $G(q)$ from input to output

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Outline

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Time-domain methods

Impulse-response analysis

Consider the system:

(input $u(t)$, output $y(t)$ and additive disturbance $v(t)$)

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t-k) + v(t) = G_0(q)u(t) + v(t)$$

- pulse $u(t) = \begin{cases} \alpha, & t = 0 \\ 0, & t \neq 0 \end{cases} \rightarrow y(t) = \alpha g_0(t) + v(t)$, by def of G_0 and impulse response $\{g_0(t)\}$
- if $v(t)$ small, then the estimate is $\hat{g}(t) = y(t)/\alpha$ and error $\epsilon(t) = v(t)/\alpha$ from experiment with pulse input.
- Drawbacks:
  - most physical processes do not allow pulse inputs s.t. $\epsilon(t)$ negligible
  - nonlinear effects may be emphasized

Step-response analysis

Similarly,

$$u(t) = \begin{cases} \alpha, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

- $y(t) = \alpha \sum_{k=1}^{t} g_0(k) + v(t)$
- $\hat{g}(t) = \dfrac{y(t) - y(t-1)}{\alpha}$ and $\epsilon(t) = \dfrac{v(t) - v(t-1)}{\alpha}$
- results in
  - large errors in most practical application
  - sufficient accuracy for control variables, i.e. time delay, static gain, dominant time-constants
  - simple regulators tuning (Ziegler-Nichols rule, 1942)
  - graphical parameter determination (Rake, 1980)

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Correlation analysis

Consider again:

$$y(t) = \sum_{k=1}^{\infty} g_0(k) u(t-k) + v(t)$$

- If u is QSS with $\bar{E}u(t)u(t-\tau) = R_u(\tau)$ and $\bar{E}u(t)v(t-\tau) = 0$ (OL) then

$$\bar{E}y(t)u(t-\tau) = R_{yu}(\tau) = \sum_{k=1}^{\infty} g_0(k) R_u(k-\tau)$$

- If u is a white noise s.t. $R_u(\tau) = \alpha\delta_{\tau 0}$ then $g_0(\tau) = R_{yu}(\tau)/\alpha$
  - An estimate of the impulse response is obtained from an estimate of $R_{yu}(\tau)$

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Example: N measurements

$$\hat{R}_{yu}^N(\tau) = \frac{1}{N} \sum_{t=\tau}^{N} y(t) u(t-\tau)$$

if $u \neq$ white noise,

- estimate $\hat{R}_u^N(\tau) = \dfrac{1}{N} \sum_{t=\tau}^{N} u(t) u(t-\tau)$
- solve $\hat{R}_{yu}^N(\tau) = \dfrac{1}{N} \sum_{k=1}^{N} \hat{g}(k) \hat{R}_u^N(k-\tau)$ for $\hat{g}(k)$
- if possible, set u such that $\hat{R}_u^N$ and $\hat{R}_{yu}^N$ are easy to solve (typically done by commercial solvers).

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Frequency-response analysis

Sine-wave testing

- physically, $G(z)$ is such that $G(e^{i\omega})$ describes what happened to a sinusoid
- if $u(t) = \alpha \cos \omega t$, $t = 0, 1, 2, \ldots$ then

$$y(t) = \alpha |G_0(e^{i\omega})| \cos(\omega t + \phi) + v(t) + \text{ transient}$$

  where $\phi = \arg G_0(e^{i\omega})$
  - $G_0(e^{i\omega})$ determined as:
    - from $u(t)$, get the amplitude and phase shift of $y(t)$
    - deduce the estimate $\hat{G}_N(e^{i\omega})$
    - repeat for frequencies within the interesting band
- known as frequency analysis
- drawback: $|G_0(e^{i\omega})|$ and ϕ difficult to determine accurately when $v(t)$ is important

Frequency analysis by the correlation method

- since $y(t)$ of known freq., correlate it out from noise
- define sums

$$I_C(N) \doteq \frac{1}{N}\sum_{t=1}^{N} y(t)\cos\omega t \quad \text{and} \quad I_S(N) \doteq \frac{1}{N}\sum_{t=1}^{N} y(t)\sin\omega t$$

- based on previous $y(t)$ (ignore transients and $\cos(a+b)$)

$$I_C(N) = \frac{\alpha}{2}|G_0(e^{i\omega})|\cos(\phi) + \underbrace{\alpha|G_0(e^{i\omega})|\frac{1}{2}\frac{1}{N}\sum_{t=1}^{N}\cos(2\omega t+\phi)}_{\to 0 \text{ as } N\to\infty}$$
$$+ \underbrace{\frac{1}{N}\sum_{t=1}^{N} v(t)\cos(\omega t)}_{\to 0 \text{ as } N\to\infty \text{ if } v(t) \text{ DN contain } \omega}$$

- if $\{v(t)\}$ is a stat. stoch. process s.t. $\sum_0^\infty \tau|R_v(\tau)| < \infty$ then the 3rd term variance decays like $1/N$

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

- similarly,

$$I_S(N) = -\frac{\alpha}{2}|G_0(e^{i\omega})|\sin(\phi) + \alpha|G_0(e^{i\omega})|\frac{1}{2}\frac{1}{N}\sum_{t=1}^{N}\sin(2\omega t+\phi)$$
$$+ \frac{1}{N}\sum_{t=1}^{N} v(t)\sin(\omega t) \approx -\frac{\alpha}{2}|G_0(e^{i\omega})|\sin(\phi)$$

- and we get the estimates

$$|\hat{G}_N(e^{i\omega})| = \frac{\sqrt{I_C^2(N)+I_S^2(N)}}{\alpha/2}, \quad \hat{\phi}_N = \arg \hat{G}_N(e^{i\omega}) = -\arctan\frac{I_S(N)}{I_C(N)}$$

- repeat over the freq. of interest (commercial soft.)

(+) Bode plot easily obtained and focus on spec. freq.

(-) many industrial processes DN admit sin inputs & long experimentation

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Relationship to Fourier analysis

Consider the discrete Fourier transform $Y_N(\omega) = \frac{1}{\sqrt{N}}\sum_{t=1}^{N} y(t)e^{-i\omega t}$ and I_C & I_S, which gives

$$I_C(N) - i\, I_S(N) = \frac{1}{\sqrt{N}}Y_N(\omega)$$

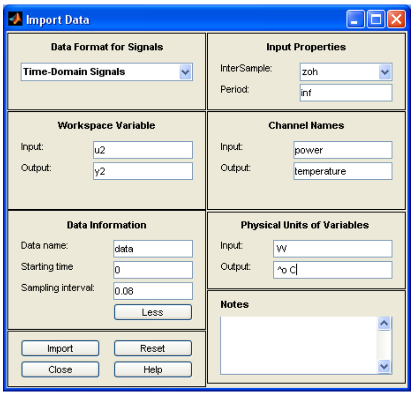
- from the periodogram (signal power at frequency ω) of $u(t) = \alpha\cos\omega t$, $U_N(\omega) = \sqrt{N}\alpha/2$ if $\omega = 2\pi r/N$ for some $r \in \mathbb{N}$
- then $\hat{G}_N(e^{i\omega}) = \frac{\sqrt{N}Y_N(\omega)}{N\alpha/2} = \frac{Y_N(\omega)}{U_N(\omega)}$
- ω is precisely the input frequency
- provides a most reasonable estimate.

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

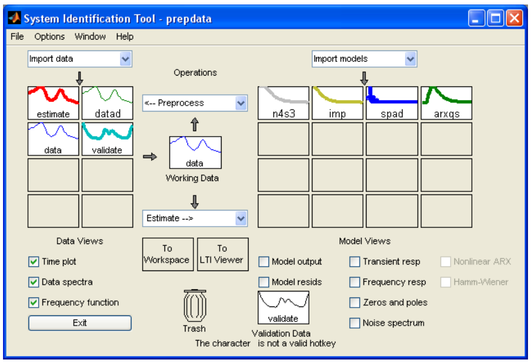
Commercial software example

In practice, you may use *Matlab Identification toolbox*® to

- import the data in a GUI

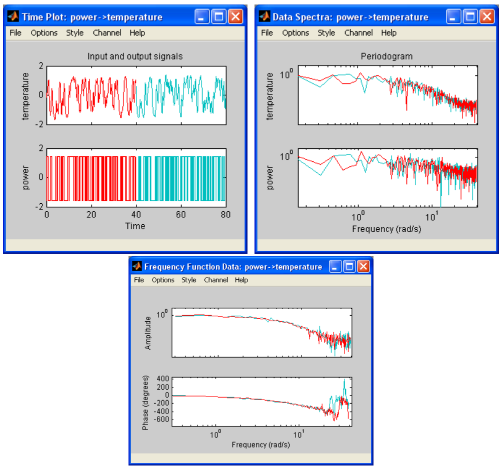


- pre-process it (remove mean, pre-filter, separate estimation from validation, etc.)



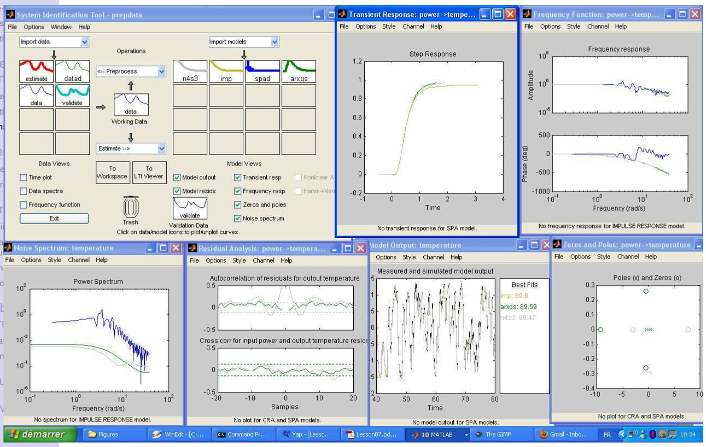
Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

- analyse the signals



Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

- get multiple models of desired order and compare the outputs



Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Fourier analysis

Empirical transfer-function estimate

Extend previous estimate to multifrequency inputs

$$\hat{\hat{G}}_N(e^{i\omega}) = \frac{Y_N(\omega)}{U_N(\omega)} \text{ with } (Y/U)_N(\omega) = \frac{1}{\sqrt{N}}\sum_{t=1}^{N}(y/u)(t)e^{-i\omega t}$$

$\hat{\hat{G}}_N$ = ETFE, since no other assumption than linearity

- original data of $2N$ numbers $y(t)$, $u(t)$, $t = 1 \dots N$ condensed into N numbers (essential points/2)

$$Re\hat{\hat{G}}_N(e^{2\pi ik/N}),\ Im\hat{\hat{G}}_N(e^{2\pi ik/N}),\ k = 0, 1, \dots, \frac{N}{2} - 1$$

$\rightarrow$ modest model reduction

- approx. solves the convolution (using Fourier techniques)

$$y(t) = \sum_{k=1}^{N} g_0(k)u(t-k), \quad t = 1, 2, \dots, N$$

Properties of the ETFE

If the input is periodic:

- the ETFE is defined only for a fixed number of frequencies
- at these frequencies the ETFE is unbiased and its variance decays like $1/N$

If the input is a realization of a stochastic process:

- the periodogram $|U_N(\omega)|^2$ is an erratic function of ω, which fluctuates around $\phi_u(\omega)$
- the ETFE is an asymptotically unbiased estimate of the TF at increasingly (with N) many frequencies
- the ETFE variance do not $\searrow$ as $N \nearrow$ and is given as the noise to signal ratio at the considered freq.
- the estimates at different frequencies are uncorrelated

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Conclusions on ETFE

- increasingly good quality for periodic signals but no improvement otherwise as $N \nearrow$
- very crude estimate in most practical cases
- due to uncorrelated information per estimated parameter

$\Rightarrow$ relate the system behavior at one frequency to another

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Spectral analysis

Smoothing the ETFE

Assumption: the true transfer function $G_0(e^{i\omega})$ is a smooth function of ω. Consequences:

- $G_0(e^{i\omega})$ supposed constant over

$$\frac{2\pi k_1}{N} = \omega_0 - \Delta\omega < \omega < \omega_0 + \Delta\omega = \frac{2\pi k_2}{N}$$

- the best way (min. var.) to estimate this cst is a weighted average of $G_0(e^{i\omega})$ on the previous freq., each measurement weighted by its inverse variance:
  - for large N, we can use Riemann sums and introduce the weights $W_\gamma(\zeta) = \begin{cases} 1, & |\zeta| < \Delta\omega \\ 0, & |\zeta| > \Delta\omega \end{cases}$
- after some cooking and simplifications,

$$\hat{G}_N(e^{i\omega_0}) = \frac{\int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)|U_N(\zeta)|^2 \hat{\hat{G}}_N(e^{i\zeta})d\zeta}{\int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)|U_N(\zeta)|^2 d\zeta}$$

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Connection with the Blackman-Turkey procedure

Noticing that as $N \to \infty$

$$\int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)|U_N(\zeta)|^2 d\zeta \to \int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)\phi_u(\zeta)d\zeta$$

supposing $\int_{-\pi}^{\pi} W_\gamma(\zeta)d\zeta = 1$ then

$$\begin{aligned} \hat{\phi}_u^N(\omega_0) &= \int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)|U_N(\zeta)|^2 d\zeta \\ \hat{\phi}_{yu}^N(\omega_0) &= \int_{-\pi}^{\pi} W_\gamma(\zeta-\omega_0)Y_N(\zeta)\bar{U}_N(\zeta)d\zeta \\ \hat{G}_N(e^{i\omega_0}) &= \frac{\hat{\phi}_{yu}^N(\omega_0)}{\hat{\phi}_u^N(\omega_0)} \end{aligned}$$

$\to$ ratio of cross spectrum by input spectrum (smoothed periodograms proposed by $B\&T$)

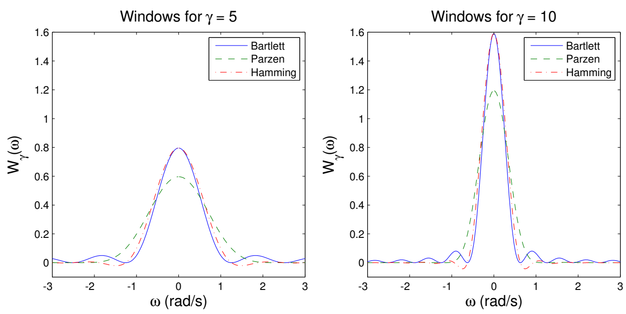
Weighting function $W_\gamma(\zeta)$: the frequency window

- "Wide" fw → weight many different frequencies, small variance of $\hat{G}_N(e^{i\omega_0})$ but far from ω_0 = bias
- γ ($\sim$ width^{-1}) = trade-off between bias and variance
- Width and amplitude:
  $M(\gamma) \doteq \int_{-\pi}^{\pi} \zeta^2 W_\gamma(\zeta) d\zeta$ and $\bar{W}(\gamma) \doteq 2\pi \int_{-\pi}^{\pi} W_\gamma^2(\zeta) d\zeta$
- Typical windows for spectral analysis:

	$2\pi W_\gamma(\omega)$	$M(\gamma)$	$\bar{W}(\gamma)$
Bartlett	$\frac{1}{\gamma}\left(\frac{\sin \gamma\omega/2}{\sin \omega/2}\right)^2$	$\frac{2.78}{\gamma}$	0.67γ
Parzen	$\frac{4(2+\cos\omega)}{\gamma^3}\left(\frac{\sin \gamma\omega/4}{\sin \omega/2}\right)^4$	$\frac{12}{\gamma^2}$	0.54γ
Hamming	$\frac{1}{2}D_\gamma(\omega)+\frac{1}{4}D_\gamma(\omega-\pi/\gamma)+\frac{1}{4}D_\gamma(\omega+\pi/\gamma)$, where $D_\gamma(\omega) \doteq \frac{\sin(\gamma+1/2)\omega}{\sin \omega/2}$	$\frac{\pi^2}{2\gamma^2}$	0.75γ

- good approx. for $\gamma \geq 5$, as $\gamma \nearrow M(\gamma) \searrow$ and $\bar{W}(\gamma) \nearrow$

- Example: $\gamma = 5$ vs. $\gamma = 10$



Asymptotic properties of the smoothed estimate

- The estimates $Re\hat{G}_N(e^{i\omega})$ and $Im\hat{G}_N(e^{i\omega})$ are asymptotically uncorrelated and of known variance
- $\hat{G}_N(e^{i\omega})$ at $\neq$ freq. are asymptotically uncorrelated
- γ that min. the mean square estimate (MSE) is

$$\gamma_{opt} = \left(\frac{4M^2|R(\omega)|^2\phi_u(\omega)}{\bar{W}\phi_v(\omega)}\right)^{1/5} \cdot N^{1/5}$$

  → frequency window more narrow when more data available, and leads to $MSE \sim C \cdot N^{-4/5}$
- typically, start with $\gamma = N/20$ and compute $\hat{G}_N(e^{i\omega})$ for various values of γ, $\nearrow \gamma \searrow$ bias $\nearrow$ variance (more details)

Example

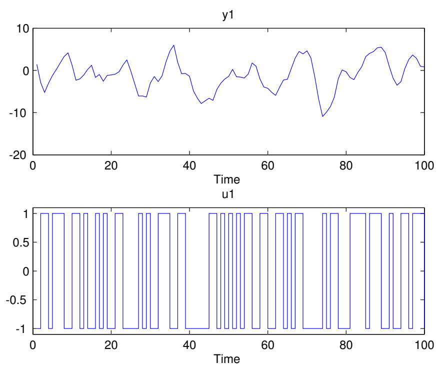
- Consider the system

$$y(t) - 1.5y(t-1) + 0.7y(t-2) = u(t-1) + 0.5u(t-2) + e(t)$$

where $e(t)$ is a white noise with variance 1 and $u(t)$ a pseudo-random binary signal (PRBS), over 1000 samples.

```
% Construct the polynomial
m0=poly2th([1 -1.5 0.7],[0 1 0.5]);
% Generate pseudorandom, binary signal
u=idinput(1000,'prbs');
% Normally distributed random numbers
e=randn(1000,1);
% Simulate and plot the output
y=idsim([u e],m0);
z=[y u]; idplot(z,[101:200])
```

- we get the inputs and ouputs



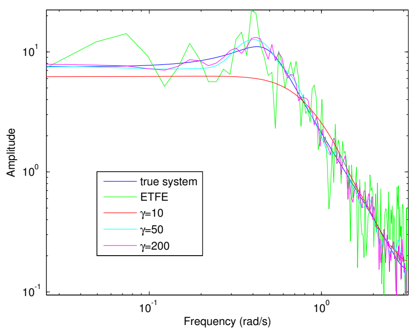
Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

- The ETFE and smoothing thanks to Hamming window ($\gamma = 10$, 50, 200) are obtained as

```
% Compute the ETFE
ghh=etfe(z);[om,ghha]=getff(ghh);
% Performs spectral analysis
g10=spa(z,10);[om,g10a]=getff(g10);
g50=spa(z,50);[om,g50a]=getff(g50);
g200=spa(z,200);[om,g200a]=getff(g200);
g0=th2ff(m0);[om,g0a]=getff(g0);
bodeplot(g0,ghh,g10,g50,g200,'a');
```

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

- we get the ETFE and estimates



$\Rightarrow$ $\gamma = 50$ seems a good choice

Nonparametric identification
E. Witrant
Time-domain methods
Impulse-response
Step-response
Correlation
Frequency-response
Sine-wave testing
Correlation method
Relationship to Fourier
Fourier
ETFE definition
ETFE properties
Spectral
Smoothing the ETFE
Blackman-Turkey procedure
Frequency window
Asymptotic properties
Disturbance spectrum
Residual spectrum
Coherency spectrum
Conclusions
Homework

Estimating the disturbance spectrum

$$y(t) = G_0(q)u(t) + v(t)$$

Estimating spectra

- Ideally, $\phi_v(\omega)$ given as (if $v(t)$ measurable):

$$\hat{\phi}_v^N(\omega) = \int_{-\pi}^{\pi} W_\gamma(\zeta - \omega)|V_N(\zeta)|^2 d\zeta$$

- Bias: $E\,\hat{\phi}_v^N(\omega) - \phi_v(\omega) = \frac{1}{2}M(\gamma)\phi_v''(\omega) + \underbrace{O(C_1(\gamma))}_{\gamma\to\infty} + \underbrace{O(\sqrt{1/N})}_{N\to\infty}$
- Variance : $\text{Var}\,\hat{\phi}_v^N(\omega) = \frac{\bar{W}(\gamma)}{N}\phi_v^2(\omega) + \underbrace{O(1/N)}_{N\to\infty}$
- Estimates at $\neq$ freq. are uncorrelated

The residual spectrum

- $v(t)$ not measurable $\rightarrow$ given the estimate $\hat{G}_N$

$$\hat{v}(t) = y(t) - \hat{G}_N(q)u(t)$$

gives

$$\hat{\phi}_v^N(\omega) = \int_{-\pi}^{\pi} W_\gamma(\zeta - \omega)|Y_N(\zeta) - \hat{G}_N(e^{i\zeta})U_N(\zeta)|^2 d\zeta$$

- After simplifications: $\hat{\phi}_v^N(\omega) = \hat{\phi}_y^N(\omega) - \frac{|\hat{\phi}_{yu}^N(\omega)|^2}{\hat{\phi}_u^N(\omega)}$
- Asymptotically uncorrelated with $\hat{G}_N$

Coherency spectrum

- Defined as

$$\hat{\kappa}_{yu}^N(\omega) \doteq \sqrt{\frac{|\hat{\phi}_{yu}^N(\omega)|^2}{\hat{\phi}_y^N(\omega)\hat{\phi}_u^N(\omega)}} \rightarrow \hat{\phi}_v^N(\omega) = \hat{\phi}_y^N(\omega)[1 - (\hat{\kappa}_{yu}^N(\omega))^2]$$

- $\kappa_{yu}(\omega)$ is the *coherency spectrum*, i.e. freq. dependent corr. btw I/O
- if 1 at a given ω, perfect corr. $\leftrightarrow$ no noise.

Conclusions

Nonparametric identification

- direct estimate of transient or frequency response
- valuable initially to provide the model structure (relations between variables, static relations, dominant time-constants ...)
- spectral analysis for frequency fonctions, Fourier = special case (wide lag window)
- essential user influence = γ: trade-off between frequency resolution vs. variability
- reasonable γ gives dominant frequency properties

Homework

1. Download the User's guide for the System Identification Toolbox^{TM}
`http://www.mathworks.com/access/helpdesk/help/pdf_doc/ident/ident.p`
Suppose that you have some data set with inputs $u \in \mathbb{R}^{1\times N_t}$ and outputs $y \in \mathbb{R}^{N_y\times N_t}$ for which you wish to build a model: find the functions in the system identification toolbox that would allow you to perform all the computations done in class.
2. Follow the Matlab example *Estimating Transfer Function Models for a Heat Exchanger*: perform and analyse all the proposed functions.

References

- L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.
- L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
- O. Hinton, *Digital Signal Processing*, EEE305 class material, Chapter 6 - Describing Random Sequences, `http://www.staff.ncl.ac.uk/oliver.hinton/eee305/Chapter6.pdf`

Université Joseph Fourier GRENOBLE — UFR de Physique

SYSTEM IDENTIFICATION

Lecture 9: Parameter Estimation in Linear Models

Emmanuel WITRANT
emmanuel.witrant@gipsa-lab.grenoble-inp.fr

September 25, 2017

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

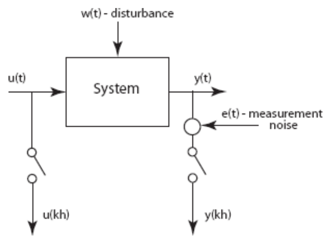
Class goal

Today, you should be able to

- distinguish between common model structures used in identification
- estimate model parameters using the prediction-error method
- calculate the optimal parameters for ARX models using least-squares
- estimate bias and variance of estimates from model and input signal properties

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

System identification

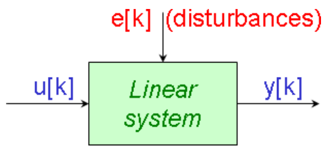


Many issues:

- **Les. 7** choice of sampling frequency, input signal (experiment conditions), pre-filtering;
- **Les. 8** non parametric models, from finite and noisy data, how to model disturbances?
- **Today** what class of models? estimating model parameters from processed data.

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

System identification via parameter estimation



Need to fix model structure before trying to estimate parameters

- system vs. disturbance model
- model order (degrees of transfer function polynomials)

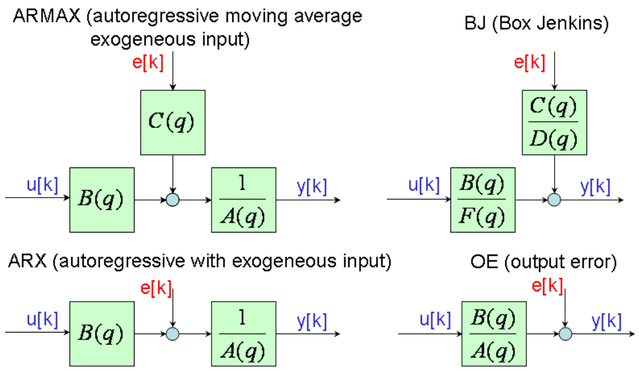
Outline

Parameter estimation in linear models
E. Witrant

Linear models

Model structures

Many model structures commonly used (BJ includes all others as special cases)



Parameter estimation in linear models
E. Witrant

Transfer function parameterizations

The transfer functions $G(q)$ and $H(q)$ in the linear model

$$y[k] = G(q;\theta)u[k] + H(q;\theta)e[k]$$

will be parameterized as (i.e. BJ)

$$G(q;\theta) \doteq q^{-n_k}\frac{b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}}{1 + f_1 q^{-1} + \cdots + f_{n_f} q^{-n_f}}$$

$$H(q;\theta) \doteq \frac{1 + c_1 q^{-1} + \cdots + c_{n_c} q^{-n_c}}{1 + d_1 q^{-1} + \cdots + d_{n_d} q^{-n_d}}$$

where the parameter vector θ contains the coefficients $\{b_k\}$, $\{f_k\}$, $\{c_k\}$, $\{d_k\}$.
Note: n_k determines dead-time, n_b, n_f, n_c, n_d order of transfer function polynomials.

Parameter estimation in linear models
E. Witrant

Model order selection from physical insight

Physical insights often help to determine the right model order:

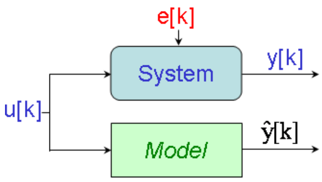
$$y[k] = q^{-n_k}\frac{b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}}{1 + f_1 q^{-1} + \cdots + f_{n_f} q^{-n_f}}u[k] + H(q;\theta)e[k]$$

If system sampled with first-order hold (input pw. cst, $1 - q^{-1}$),

- n_f equals the number of poles of continuous-time system
- if system has no delay and no direct term, then $n_b = n_f$, $n_k = 1$
- if system has no delay but direct term, then $n_b = n_f + 1$, $n_k = 0$
- if continuous system has time delay τ, then $n_k = [\tau/h] + 1$

Note: n_b does not depend on number of continuous-time zeros! i.e. compare Euler vs. Tustin discretization

Basic principle of parameter estimation



- For given parameters θ, the model predicts that the system output should be $\hat{y}[t;\theta]$
- Determine θ so that $\hat{y}[t;\theta]$ matches observed output $y[t]$ "as closely as possible"
- To solve the parameter estimation problem, note that:
  1. $\hat{y}[t;\theta]$ depends on the disturbance model
  2. "as closely as possible" needs a mathematical formulation

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

One step-ahead prediction

Consider LTI $y(t) = G(q)u(t) + H(q)e(t)$ and undisturbed output $y^* = G^*u^*$. Suppose that $H(q)$ is monic ($h(0) = 1$, i.e. $1 + cq^{-1}$ for moving average), the disturbance is

$$v(t) = H(q)e(t) = \sum_{k=0}^{\infty} h(k)e(t-k) = e(t) + \underbrace{\sum_{k=1}^{\infty} h(k)e(t-k)}_{m(t-1),\ \text{known at } t-1}$$

Since $e(t)$ white noise (0 mean), the conditional expectations (expected value of a real random variable with respect to a conditional probability distribution) are:

$$\begin{aligned}
\hat{v}(t|t-1) &= m(t-1) = (H(q)-1)e(t) = (1-H^{-1}(q))v(t)\\
\Rightarrow \hat{y}(t|t-1) &= G(q)u(t) + \hat{v}(t|t-1)\\
&= G(q)u(t) + (1-H^{-1}(q))(y(t)-G(q)u(t))\\
&= \left[1-H^{-1}(q)\right]y(t) + H^{-1}(q)G(q)u(t)
\end{aligned}$$

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Parameter estimation methods

Consider the particular model structure $\mathcal{M}$ parameterized using $\theta \in \mathcal{D}_{\mathcal{M}} \subset \mathbb{R}^d$: $\mathcal{M}^* = \{\mathcal{M}(\theta)|\theta \in \mathcal{D}_{\mathcal{M}}\}$

- each model can predict future outputs:

  $$\mathcal{M}(\theta) : \hat{y}(t|\theta) = W_y(q,\theta)y(t) + W_u(q,\theta)u(t)$$

  i.e. one step-ahead prediction of

  $$y(t) = G(q,\theta)u(t) + H(q,\theta)e(t) :$$

  $W_y(q,\theta) = [1 - H^{-1}(q,\theta)]$, $W_u(q,\theta) = H^{-1}(q,\theta)G(q,\theta)$ (multiply by H^{-1} to make e white noise),
- or nonlinear filter $\mathcal{M}(\theta) : \hat{y}(t|\theta) = g(t, Z^{t-1};\theta)$ where $Z^N \doteq [y(1), u(1), \ldots, y(N), u(N)]$ contains the past information.

$\Rightarrow$ Determine the map $Z^N \to \hat{\theta}_N \in \mathcal{D}_{\mathcal{M}}$ = parameter estimation method

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Evaluating the candidate models

Given a specific model $\mathcal{M}(\theta_*)$, we want to evaluate the prediction error

$$\epsilon(t,\theta_*) = y(t) - \hat{y}(t|\theta_*)$$

computed for $t = 1, 2, \ldots, N$ when Z^N is known.

- "Good model" = small ϵ when applied to observed data,
- "good" prediction performance multiply defined, guiding principle:
  Based on Z^t we can compute the prediction error $\epsilon(t,\theta)$. At time $t = N$, select $\hat{\theta}_N$ such that $\epsilon(t,\hat{\theta}_N)$, $t = 1, 2, \ldots, N$, becomes as small as possible.
- How to qualify "small":
  1. scalar-valued norm or criterion function measuring the size of ϵ;
  2. $\epsilon(t,\hat{\theta}_N)$ uncorrelated with given data ("projections" are 0).

Minimizing prediction errors

1. Get $\hat{y}(t|\theta_*)$ from the model to compute $\epsilon(t,\theta_*) = y(t) - \hat{y}(t|\theta_*)$. Ex.: calculate ϵ
2. Filter $\epsilon \in \mathbb{R}^N$ with a stable linear filter $L(q)$: $\epsilon_F(t,\theta) = L(q)\epsilon(t,\theta), \quad 1 \le t \le N$
3. Use the norm ($l(\cdot) > 0$ scalar-valued)

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N} l(\epsilon_F(t,\theta))$$

4. Estimate $\hat{\theta}_N$ by minimization

$$\hat{\theta}_N = \hat{\theta}_N(Z^N) = \arg\min_{\theta\in\mathcal{D}_\mathcal{M}} V_N(\theta, Z^N)$$

$\Rightarrow$ Prediction-error estimation methods (PEM), defined depending on $l(\cdot)$ and prefilter $L(q)$.

Parameter estimation in linear models
E. Witrant

Choice of L

Extra freedom for non-momentary properties of ϵ

- same as filtering I/O data prior to identification
- L acts on HF disturbances or slow drift terms, as frequency weighting
- note that the filtered error is

$$\epsilon_F(t,\theta) = L(q)\epsilon(t,\theta) = \left[L^{-1}(q)H(q,\theta)\right]^{-1}[y(t) - G(q,\theta)]$$

$\Rightarrow$ filtering is same as changing the noise model to $\bar{H}_L(q,\theta) = L^{-1}(q)H(q,\theta)$

Parameter estimation in linear models
E. Witrant

Choice of l

- quadratic norm $l(\epsilon)$ is first candidate
- other choices for robustness constraints
- may be parameterized as $l(\epsilon,\theta)$, independently of model parametrization

$$\theta = \begin{bmatrix} \theta' \\ \alpha \end{bmatrix} : l(\epsilon(t,\theta),\theta) = l(\epsilon(t,\theta'),\alpha)$$

Parameter estimation in linear models
E. Witrant

Multivariable systems

Quadratic criterion:

$$l(\epsilon) = \frac{1}{2}\epsilon^T\Lambda^{-1}\epsilon$$

with weight $\Lambda \ge 0 \in \mathbb{R}^{p\times p}$

- Define, instead of l, the $p\times p$ matrix

$$Q_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}\epsilon(t,\theta)\epsilon^T(t,\theta)$$

- and the scalar-valued function

$$V_N(\theta, Z^N) = h(Q_N(\theta, Z^N))$$

with $h(Q) = \frac{1}{2}\mathrm{tr}(Q\Lambda^{-1})$.

Linear regressions and least squares

Linear regressions

Employ predictor architecture (linear in *theta*)

$$\hat{y}(t|\theta) = \phi^T(t)\theta + \mu(t)$$

where ϕ is the regression vector, i.e. for ARX

$$\begin{aligned} y(t) &+ a_1 y(t-1) + \ldots + a_{n_a} y(t-n_a) \\ &= b_1 u(t-1) + \ldots + b_{n_b} u(t-n_b) + e(t), \\ \Rightarrow \phi(t) &= [-y(t-1)\ -y(t-2)\ \ldots -y(t-n_a) \\ &\quad u(t-1) \ldots u(t-n_b)]^T \end{aligned}$$

and $\mu(t)$ a known data-dependent vector (take $\mu(t) = 0$ in the following).

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Least-squares criterion

The prediction error becomes $\epsilon(t,\theta) = y(t) - \phi^T(t)\theta$ and the criterion function (with $L(q) = 1$ and $l(\epsilon) = \frac{1}{2}\epsilon^2$)

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \frac{1}{2}\left\| y(t) - \phi^T(t)\theta \right\|_2^2$$

$\therefore$ least-squares criterion for linear regression. Can be minimized analytically (1^{st} order condition) with

$$\theta_N^{LS} = \arg\min V_N(\theta, Z^N) = \underbrace{\left[\frac{1}{N}\sum_{t=1}^{N}\phi(t)\phi^T(t)\right]^{-1}}_{R(N)^{-1}\in\mathbb{R}^{d\times d}} \underbrace{\frac{1}{N}\sum_{t=1}^{N}\phi(t)y(t)}_{f(N)\in\mathbb{R}^d}$$

the least-squares estimate (LSE). [Exercise: proove this result]

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Example: parameter estimation in ARX models

Estimate the model parameters a and b in the ARX model

$$y(k) = ay(k-1) + bu(k-1) + e(k)$$

from $\{y(k)\}$, $\{u(k)\}$ for $k = 0, \ldots, N$.
$\Rightarrow$ find θ_N^{LS}!

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Solution

- $\theta = [a\ b]^T$ and $\phi(t) = [y(t-1)\ u(t-1)]^T$
- The optimization problem is solved with

$$R(N) = \frac{1}{N}\sum_{t=1}^{N}\begin{bmatrix} y^2(t-1) & y(t-1)u(t-1) \\ y(t-1)u(t-1) & u^2(t-1) \end{bmatrix}$$

and

$$f(N) = \frac{1}{N}\sum_{t=1}^{N}\begin{bmatrix} y(t-1)y(t) \\ u(t-1)y(t) \end{bmatrix}$$

- Note: estimate computed using covariances of $u(t)$, $y(t)$ (cf. correlation analysis).

[Exercise:] Find R^{-1} for $N = 2$. Remember:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

The Inverted Correlation Matrix

- The determinant of the correlation matrix will equal 1.0 only if all correlations equal 0. Otherwise the determinant will be less than 1.
- The determinant is related to the volume of the space occupied by the swarm of data points represented by standard scores on the measures involved.
- When the measures are uncorrelated, this space is a sphere with a volume of 1.
- When the measures are correlated, the space occupied becomes an ellipsoid whose volume is less than 1.

refs: `https://www.quora.com/What-does-the-determinant-of-the-correlation-matrix-represent`, `http://www.tulane.edu/~PsycStat/dunlap/Psyc613/RI2.html`

Parameter estimation in linear models
E. Witrant
Linear models: Model structures, TF parameterizations, From physical insights
Parameter estimation: Prediction, Estimation methods, Evaluating models
Minimizing pred error: Choice of L, Choice of l, Multivariable systems
Linear reg. & LS: LS criterion, Properties of the LSE, Multivariable LS, LS for state-space, General models
PEM properties: Convergence, Variance, Identifiability
Conclusions

Properties of the LSE

Consider the observed data $y(t) = \phi^T(t)\theta_0 + v_0(t)$, θ_0 being the *true* value:

$$\lim_{N\to\infty} \theta_N^{LS} - \theta_0 = \lim_{N\to\infty} R(N)^{-1}\frac{1}{N}\sum_{t=1}^{N}\phi(t)v_0(t) = (R^*)^{-1}f^*$$

with $R^* = \bar{E}\phi(t)\phi^T(t)$, $f^* = \bar{E}\phi(t)v_0(t)$, v_0 & ϕ QSS. Then $\theta_N^{LS} \to \theta_0$ if

- R^* non-singular (co-variance exists, decaying as $1/N$)
- $f^* = 0$, satisfied if
  1. $v_0(t)$ a sequence of independent random variables with zero mean (i.e. white noise): $v_0(t)$ indep. of what happened up to $t-1$
  2. $\{u(t)\}$ indep. of $\{v_0(t)\}$ & $n_a = 0$ (i.e. ARX) $\to \phi(t)$ depends on $u(t)$ only.

[Exercise: proove this result]

Parameter estimation in linear models
E. Witrant
Linear models: Model structures, TF parameterizations, From physical insights
Parameter estimation: Prediction, Estimation methods, Evaluating models
Minimizing pred error: Choice of L, Choice of l, Multivariable systems
Linear reg. & LS: LS criterion, Properties of the LSE, Multivariable LS, LS for state-space, General models
PEM properties: Convergence, Variance, Identifiability
Conclusions

Multivariable case

When $y(t) \in \mathbb{R}^p$

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}\left[y(t) - \phi^T(t)\theta\right]^T \Lambda^{-1}\left[y(t) - \phi^T(t)\theta\right]$$

gives the estimate

$$\begin{aligned}\theta_N^{LS} &= \arg\min V_N(\theta, Z^N)\\ &= \left[\frac{1}{N}\sum_{t=1}^{N}\phi(t)\Lambda^{-1}\phi^T(t)\right]^{-1}\frac{1}{N}\sum_{t=1}^{N}\phi(t)\Lambda^{-1}y(t)\end{aligned}$$

Key issue: proper choice of the relative weight Λ!

Parameter estimation in linear models
E. Witrant
Linear models: Model structures, TF parameterizations, From physical insights
Parameter estimation: Prediction, Estimation methods, Evaluating models
Minimizing pred error: Choice of L, Choice of l, Multivariable systems
Linear reg. & LS: LS criterion, Properties of the LSE, Multivariable LS, LS for state-space, General models
PEM properties: Convergence, Variance, Identifiability
Conclusions

LS for state-space

Consider the LTI

$$\begin{aligned}x(t+1) &= Ax(t) + Bu(t) + w(t)\\ y(t) &= Cx(t) + Du(t) + v(t)\end{aligned}$$

Set

$$Y(t) = \begin{bmatrix} x(t+1)\\ y(t)\end{bmatrix}, \Theta = \begin{bmatrix} A & B\\ C & D\end{bmatrix}, \Phi(t) = \begin{bmatrix} x(t)\\ u(t)\end{bmatrix}, E(t) = \begin{bmatrix} w(t)\\ v(t)\end{bmatrix}$$

Then $Y(t) = \Theta\Phi(t) + E(t)$ where $E(t)$ from sampled sum of squared residuals (provides cov. mat. for Kalman filter).
Problem: get $x(t)$. Essentially obtained as $x(t) = L\hat{Y}_r$ where $\hat{Y}_r$ is a r-steps ahead predictor (cf. basic subspace algorithm).

Parameter estimation in general model structures

More complicated when predictor is not linear in parameters. In general, we need to minimize $V_N(\theta) \geq 0$ using iterative numerical method, e.g.,

$$\theta^{i+1} = \theta^i - \mu^i M^i V_N'(\theta^i)$$

[Exercise: analyze the convergence of V]
Example: Newtons method uses (pseudo-Hessian)

$$M^i = \left(V_N''(\theta^i)\right)^{-1} \text{ or } \left(V_N''(\theta^i) + \alpha\right)^{-1}$$

while Gauss-Newton approximate M^i using first-order derivatives.
⇒ locally optimal, but not necessarily globally optimal.

Properties of prediction error minimization estimates

What can we say about models estimated using prediction error minimization?
Model errors have two components:

1. Bias errors: arise if model is unable to capture true system
2. Variance errors: influence of stochastic disturbances

Two properties of general prediction error methods:

1. Convergence: what happens with $\hat{\theta}_N$ as N grows?
2. Accuracy: what can we say about size of $\hat{\theta}_N - \theta_0$ as $N \nearrow$?

Convergence

- If disturbances acting on system are stochastic, then so is prediction error $\epsilon(t)$
- Under quite general conditions (even if $\epsilon(t)$ are not independent)

$$\lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \epsilon^2(t|\theta) = E\{\epsilon^2(t|\theta)\}$$

and

$$\hat{\theta}_N \to \theta^* = \arg\min_\theta E\{\epsilon^2(t|\theta)\} \text{ as } N \to \infty$$

⇒ Even if model cannot reflect reality, estimate will minimize prediction error variance! ↔ Robustness property.

Example

Assume that you try to estimate the parameter b in the model

$$\hat{y}[k] = bu[k-1] + e[k]$$

while the true system is given by

$$y[k] = u[k-1] + u[k-2] + w[k]$$

where $\{u, e, w\}$ are white noise sequences, independent of each other.
[Exercise: What will the estimate (computed using the prediction error method) converge to?]

Solution

The PEM will find the parameters that minimize the variance

$$
\begin{aligned}
E\{\epsilon^2(k)\} &= E\{(y[k]-\hat{y}[k])^2\} \\
&= E\{(u[k-1]+u[k-2]+w[k]-bu[k-1]-e[k])^2\} \\
&= E\{((1-b)u[k-1]+u[k-2])^2\}+\sigma_w^2+\sigma_e^2 \\
&= (1-b)^2\sigma_u^2+\sigma_u^2+\sigma_w^2+\sigma_e^2
\end{aligned}
$$

minimized by $b=1 \rightarrow$ asymptotic estimate.

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Convergence (2): frequency analysis

Consider the one-step ahead predictor and true system

$$
\begin{aligned}
\hat{y}(t) &= [1-H_*^{-1}(q,\theta)]y(t)+H_*^{-1}(q,\theta)G(q,\theta)u(t) \\
y(t) &= G_0(q)u(t)+w(t) \\
\Rightarrow \epsilon(t,\theta) &= H_*^{-1}(q)[y(t)-G(q,\theta)u(t)] \\
&= H_*^{-1}(q)[G_0(q)-G(q,\theta)]u(t)+H_*^{-1}w(t)
\end{aligned}
$$

Looking at the spectrum and with Parseval's identity

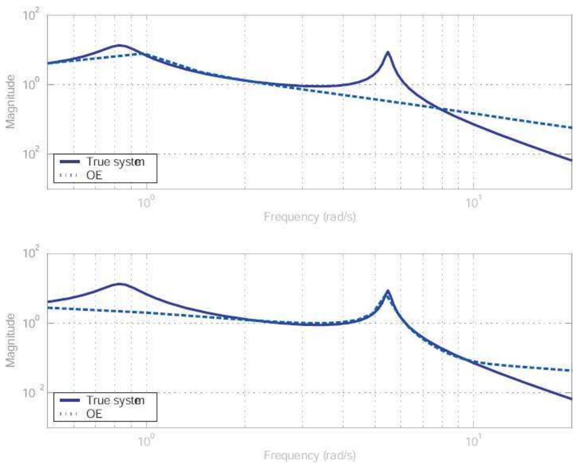
$$
\theta^* = \lim_{N\to\infty}\hat{\theta}_N = \arg\min_\theta \int_{-\pi}^{\pi} \underbrace{|G_0(e^{i\omega})-G(e^{i\omega},\theta)|^2}_{\text{made as small as possible}} \underbrace{\frac{\phi_u(\omega)}{|H_*(e^{i\omega})|^2}}_{\text{weighting function}} d\omega
$$

- good fit where $\phi_u(\omega)$ contains much energy, or $H_*(e^{i\omega})$ contains little energy
- can focus model accuracy to "important" frequency range by proper choice of $\{u\}$
- θ^* can be computed using the ETFE as G_0

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Example

Output error method using low- and high-frequency inputs



Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Estimation error variance

Supposing that $\exists\ \theta_0$ s.t.

$$
y(t)-\hat{y}(t|\theta_0)=\epsilon(t|\theta_0)=e(t)=\text{white noise with var } \lambda
$$

the estimation error variance is
$E\{(\hat{\theta}_N-\theta_0)(\hat{\theta}_N-\theta_0)^T\} \approx \frac{1}{N}\lambda\bar{R}^{-1}$, where $\bar{R}=E\{\psi(t|\theta_0)\psi(t|\theta_0)^T\}$ and $\psi(t|\theta) \doteq \frac{d}{d\theta}\hat{y}(t|\theta)$ (prediction gradient wrt θ). Then:

- the error variance $\nearrow$ with noise intensity and $\searrow$ with N
- the prediction quality is proportional to the sensitivity of $\hat{y}$ with respect to θ (componentwise)
- considering that ψ computed by min. algo., use

$$
\bar{R} \approx \frac{1}{N}\sum_{t=1}^{N}\psi(t|\hat{\theta}_N)\psi(t|\hat{\theta}_N)^T, \quad \lambda \approx \frac{1}{N}\sum_{t=1}^{N}\epsilon^2(t|\hat{\theta}_N)
$$

- $\hat{\theta}_N$ converges to a normal distribution with mean θ_0 and variance $\frac{1}{N}\lambda\bar{R}^{-1}$

Error variance (2):frequency domain characterization

The variance of the frequency response of the estimate

$$\mathbf{Var}\left\{G(e^{i\omega};\theta)\approx \frac{n}{N}\frac{\Phi_w(\omega)}{\Phi_u(\omega)}\right\}$$

- increases with number of model parameters n
- decreases with N & signal-to-noise ratio
- input frequency content influences model accuracy

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Identifiability

- Determines if the chosen parameters can be determined from the data, uniquely.
- A specific parametrization is identifiable at θ_* if

$$\hat{y}(t|\theta_*) \equiv \hat{y}(t|\theta) \text{ implies } \theta = \theta_*$$

- May not hold if
  - two $\neq \theta$ give identical I/O model properties
  - we get $\neq$ models for $\neq \theta$ but the predictions are the same due to input deficiencies

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Conclusions

- Model structure from physical insights
- Seek (next step) model prediction using measurement history
- Minimize prediction error with proper weights (filters)
- i.e. least squares: regressor & disturbance architecture $\Rightarrow$ optimization using signal covariances
- Evaluate convergence & variance as performance criteria, check identifiability

Parameter estimation in linear models
E. Witrant
Linear models
Model structures
TF parameterizations
From physical insights
Parameter estimation
Prediction
Estimation methods
Evaluating models
Minimizing pred error
Choice of L
Choice of l
Multivariable systems
Linear reg. & LS
LS criterion
Properties of the LSE
Multivariable LS
LS for state-space
General models
PEM properties
Convergence
Variance
Identifiability
Conclusions

Homework (Exam 2012)

Design an identification scheme for processes with transfer functions of the form:

1. $G_1(z^{-1}) = \dfrac{b_1 z^{-1}}{1 + a_1 z^{-1}} z^{-2}$
2. $G_2(s) = \dfrac{b_0}{(Ts+1)^2}$

e.g. identify the parameters a_i, b_i and T from N inputs and outputs measurements.
Hint: use Tustin's method to discretize G_2.

References

- L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.
- L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
- Lecture notes from 2E1282 *Modeling of Dynamical Systems, Automatic Control*, School of Electrical Engineering, KTH, Sweden.

System Identification

Lecture 10: Experiment Design and Model Validation

Emmanuel WITRANT
emmanuel.witrant@ujf-grenoble.fr

October 12, 2016

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

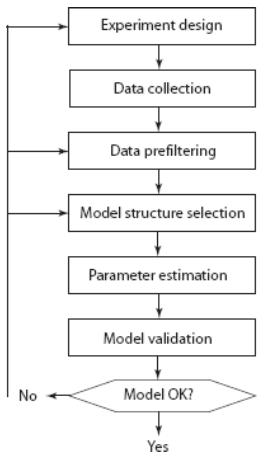
Class goal

Today, you should be able to

- use system identification as a systematic model-building tool
- do a careful experiment design/data collection to enable good model estimation
- select the appropriate model structure and model order
- validate that the estimated model is able to reproduce the observed data

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

System identification: an iterative procedure



Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Outline

1. Experiments and data collection
2. Informative experiments
3. Input design for open-loop experiments
4. Identification in closed-loop
5. Choice of the model structure
6. Model validation
7. Residual analysis

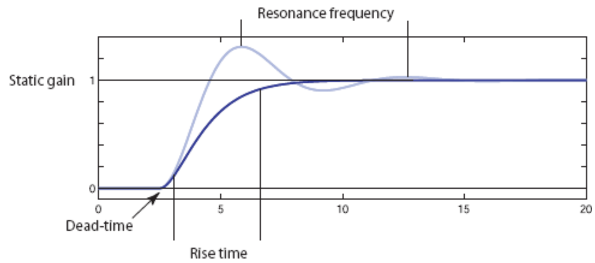
Experiments and data collection

A two-stage approach.

1. Preliminary experiments:
   - step/impulse response tests to get basic understanding of system dynamics
   - linearity, stationary gains, time delays, time constants, sampling interval
2. Data collection for model estimation:
   - carefully designed experiment to enable good model fit
   - operating point, input signal type, number of data points to collect, etc.

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Preliminary experiments: step response



Useful for obtaining qualitative information about system:

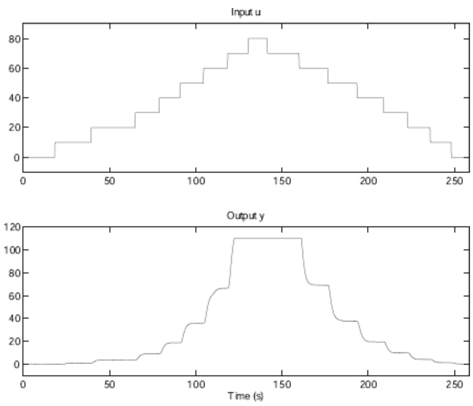
- indicates dead-times, static gain, time constants and resonances
- aids sampling time selection (rule-of-thumb: 4-10 samples per rise time)

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Tests for verifying linearity

For linear systems, response is independent of operating point,

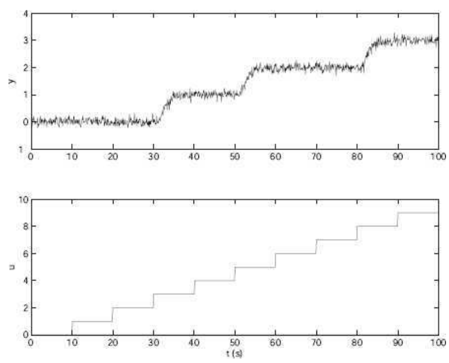
- test linearity by a sequence of step response tests for different operating points



Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Tests for detecting friction

Friction can be detected by using small step increases in input

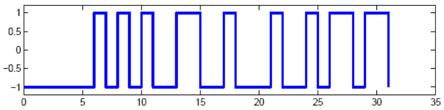


Input moves every two or three steps.

Designing experiment for model estimation

Input signal should excite all relevant frequencies

- estimated model accurate in frequency ranges where input has much energy
- good choice is often a binary sequence with random hold times (e.g., PRBS)



Trade-off in selection of signal amplitude

- large amplitude gives high signal-to-noise ratio, low parameter variance
- most systems are nonlinear for large input amplitudes

Many pitfalls if estimating a model of a system under closed-loop control!

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Informative experiments

- The data set Z^∞ is "informative enough" with respect to model set $\mathcal{M}^*$ if it allows for discremination between $2\neq$ models in the set.
- Transferred to "informative enough" experiment if it generates appropriate data set.
- Applicable to all models likely to be used.

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Open-loop experiments

Consider the set of SISO linear models

$$\mathcal{M}^* = \{G(q,\theta), H(q,\theta) | \theta \in D_{\mathcal{M}}\}$$

with the true model

$$y(t) = G_0(q)u(t) + H_0(q)e_0(t)$$

If the data are not informative with respect to $\mathcal{M}^*$ & $\theta_1 \neq \theta_2$, then

$$|\Delta G(e^{i\omega})|^2 \Phi_u(\omega) \equiv 0,$$

where $\Delta G(q) \doteq G(q,\theta_1) - G(q,\theta_2)$:

$\Rightarrow$ crucial condition on the open-loop input spectrum $\Phi_u(\omega)$

- if it implies that $\Delta G(e^{i\omega}) \equiv 0$ for two equal models, then the data is sufficiently informative with respect to $\mathcal{M}^*$

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Persistence of excitation

Def. A QSS $\{u(t)\}$ with spectrum $\Phi_u(\omega)$ is said persistently exciting of order n if, $\forall M_n(q) = m_1 q^{-1} + \ldots + m_n q^{-n}$

$$|M_n(e^{i\omega})|^2 \Phi_u(\omega) \equiv 0 \rightarrow M_n(e^{i\omega}) \equiv 0$$

Lem. In terms of covariance function $R_u(\tau)$, it means that if

$$\bar{R}_n \doteq \begin{bmatrix} R_u(0) & \ldots & R_u(n-1) \\ \vdots & \ddots & \vdots \\ R_u(n-1) & \ldots & R_u(0) \end{bmatrix}$$

then $\{u(t)\}$ *persistently exciting* $\Leftrightarrow$ $\bar{R}_n$ nonsingular.

Lec.PE If the underlying system is $y[t] = \theta^T \phi[t] + v[t]$ then $\hat{\theta}$ that makes the model $y[t] = \hat{\theta}\phi[t]$ best fit measured $\{u[t]\}$ and $\{y[t]\}$ are given by

$$\hat{\theta} = (\underbrace{\phi_N^T \phi_N}_{\bar{R}_n})^{-1} \phi_N^T y_N$$

Informative open-loop experiments

Consider a set $\mathcal{M}^*$ st.

$$G(q,\theta) = \frac{q^{-n_k}(b_1 + b_2 q^{-1} + \ldots + b_{n_b} q^{-n_b+1})}{1 + f_1 q^{-1} + \ldots + f_{n_f} q^{-n_f}}$$

then an OL experiment with an input that is persistently exciting of order $n = n_b + n_f$ is sufficiently informative with respect to $\mathcal{M}^*$.

Cor. an OL experiment is informative if the input is persistently exciting.

- the order of excitation = nb of identified parameters
- e.g. $\Phi_u(\omega) \neq 0$ at n points (n sinusoids)

Rq: immediate multivariable counterpart

$\Rightarrow$ The input should include many distinct frequencies: still a large degree of freedom!

Experiment design and model validation
E. Witrant

Input design for open-loop experiments

Three basic facts:

- asymptotic properties of the estimate (bias & variance) depend only on input spectrum, not the waveform
- limited input amplitude: $\underline{u} \leq u \leq \bar{u}$
- periodic inputs may have some advantages

Experiment design and model validation
E. Witrant

The crest factor

- cov. matrix typically inversely proportional to input power $\Rightarrow$ have as much power as possible
- physical bounds $\underline{u}, \bar{u} \rightarrow$ desired waveform property defined as crest factor; for zero-mean signal:

$$C_r^2 = \frac{\max_t u^2(t)}{\lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} u^2(t)}$$

- good waveform = small crest factor
- theoretical lower bound is 1 = binary, symmetric signals $u(t) = \pm\bar{u}$
- specific caution: do not allow validation against nonlinearities

Experiment design and model validation
E. Witrant

Common input signals

Achieve desired input spectrum with smallest crest factor: typically antagonist properties.

- Filtered Gaussian white noise (WN): any spectrum with appropriate filter, use off-line non-causal filters (e.g. Kaiser & Reed, 1977) to eliminate the transients (theoretically unbounded)
- Random binary signals (RBS): generate with a filtered zero-mean Gaussian noise and take the sign. $C_r = 1$, problem: filter change spectrum
- Pseudo-Random Binary Signal (PRBS): periodic, deterministic signal with white noise properties. Advantages with respect to RBS:
  - cov. matrix can be analytically inverted
  - secured second order properties when whole periods
  - not straightforward to generate uncorrelated PRBS
  - work with integer number of periods to have full PRBS advantages $\rightarrow$ limited by experimental length

Common input signals (2)

- Low-pass filtering by increasing the clock period: to get more low-frequency, filter PRBS (no B) and take P samples over one period:

$$u(t) = \frac{1}{P}(e(t) + \ldots + e(t-P+1))$$

- Multi-sines: sum of sinusoids $u(t) = \sum_{k=1}^{d} a_k \cos(\omega_k t + \phi_k)$
- Chirp signals or swept sinusoids: sin. with freq. that changes continuously over certain band $\Omega : \omega_1 \leq \omega \leq \omega_2$ and time period $0 \leq t \leq M$

$$u(t) = A\cos\left(\omega_1 t + (\omega_2 - \omega_1)t^2/(2M)\right)$$

instantaneous frequency (d/dt): $\omega_i = \omega_1 + \frac{t}{M}(\omega_2 - \omega_1)$. Good control over excited freq. and same crest as sin. but induces freq. outside Ω.

Periodic inputs

Some guidelines:

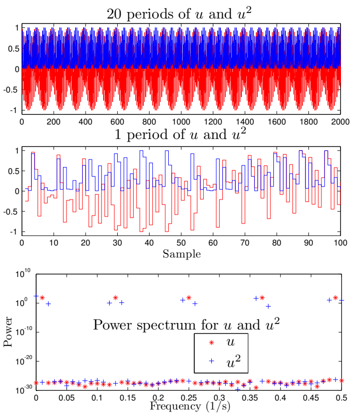
- generate PRBS over one full period, $M = 2^n - 1$ and repeat it
- for multi-sine of period M, choose ω_k from DFT-grid (density functional theory) $\omega_l = 2\pi l/M$, $l = 0, 1, \ldots, M-1$
- for chirp of period M, choose $\omega_{1,2} = 2\pi k_{1,2}/M$

Advantages and drawbacks:

- period $M \rightarrow M$ distinct frequencies in spectrum, persistent excitation of (at most) order M
- when K periods of length M ($N = KM$), average outputs over the periods and select one to work with (↘ data to handle, signal to noise ration improved by K)
- allows noise estimation: removing transients, differences in output responses over ≠ periods attributed to noise
- when model estimated in Fourier transformed data, no leakage when forming FT

Example: input consisting of five sinusoids

```
u = idinput([100 1 20],'sine',
    [],[],[5 10 1]);
% u = idinput(N,type,band,levels)
% [u,freqs] = idinput(N,'sine',
%    band,levels,sinedata)
% N = [P nu M] gives a periodic
%    input with nu channels,
%    each of length M*P and
%    periodic with period P.
% sinedata = [No_of_Sinusoids,
%    No_of_Trials, Grid_Skip]
u = iddata([],u,1,'per',100);
u2 = u.u.^2;
u2 = iddata([],u2,1,'per',100);
```



Spectrum of u vs. u^2: frequency splitting (the square having spectral support at other frequencies) reveals the nonlinearity involved.

Identification in closed-loop

Identification under output feedback necessary if unstable plant, or controlled for safety/production, or inherent feedback mechanisms.

Basic good news: prediction error method provides good estimate regardless of CL if

- the data is informative
- the model sets contains the true system

Some fallacies:

- CL experiment may be non-informative even if persistent input, associated with too simple regulators
- direct spectral analysis gives erroneous results
- corr. analysis gives biased estimate, since $\bar{E}u(t)v(t-\tau) \neq 0$
- OEM do not give consistent G when the additive noise not white

Example: proportional feedback

Consider the first-order model and feedback

$$y(t) + ay(t-1) = bu(t-1) + e(t), \quad u(t) = -fy(t)$$

then

$$y(t) + (a + bf)y(t-1) = e(t)$$

$\Rightarrow$ all models $\hat{a} = a + \gamma f$, $\hat{b} = b - \gamma$ where γ is an arbitrary scalar give the same I/O description: even if $u(t)$ is persistently exciting, the experimental condition is not informative enough.

Some guidelines

- The CL experiment is informative $\Leftrightarrow$ reference $r(t)$ is persistently exciting in

$$\begin{aligned} y(t) &= G_0(q)u(t) + H_0(q)e(t) \\ u(t) &= r(t) - F_y(q)y(t) \end{aligned}$$

- Non linear, time-varying or complex (high-order) regulators yield informative enough experiments in general
- A switch between regulators, e.g.

$$u(t) = -F_1(q)y(t) \text{ and } u(t) = -F_2(q)y(t),$$
$$\text{s.t. } F_1(e^{i\omega}) \neq F_2(e^{i\omega});\ \forall \omega$$

  achieves informative experiments
- Feedback allows to inject more input in certain freq ranges without increasing output power.

Choice of the model structure

1. Start with non-parametric estimates (correlation analysis, spectral estimation)
   - give information about model order and important frequency regions
2. Prefilter I/O data to emphasize important frequency ranges
3. Begin with ARX models
4. Select model orders via
   - cross-validation (simulate & compare with new data)
   - Akaike's Information Criterion, i.e., pick the model that minimizes

$$\left(1 + 2\frac{d}{N}\right)\sum_{t=1}^{N} \epsilon[t;\theta]^2$$

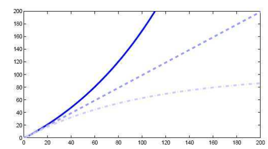
   where d = nb estimated parameters in the model

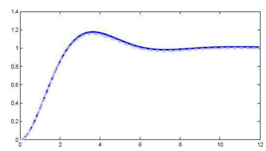
Model validation

Parameter estimation $\rightarrow$ "best model" in chosen structure, but "good enough"?

- sufficient agreement with observed data
- appropriate for intended purpose
- closeness to the "true system"

Example: $G(s) = \dfrac{1}{(s+1)(s+a)}$ has O- & CL responses for $a = \{-0.01, 0, 0.01\}$



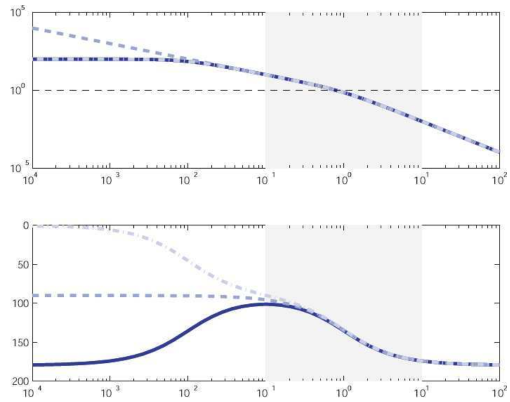


Insufficient for OL prediction, good enough for CL control!

Validation

- with respect to purpose: regulator design, prediction or simulation → test on specific problem, may be limited to do exhaustively (cost, safety)
- feasibility of physical parameters: estimated values and variance compared with prior knowledge. can also check sensitivity for identifiability
- consistency of I/O behavior:
  - Bode's diagrams for ≠ models & spectral analysis
  - by simulation for NL models
- with respect to data: verify that observations behave according to modeling assumptions
  1. Compare model simulation/prediction with real data
  2. Compare estimated models frequency response and spectral analysis estimate
  3. Perform statistical tests on prediction errors

Example: Bode plot for CL control



Different low-frequency behavior, similar responses around cross-over frequency

Model reduction

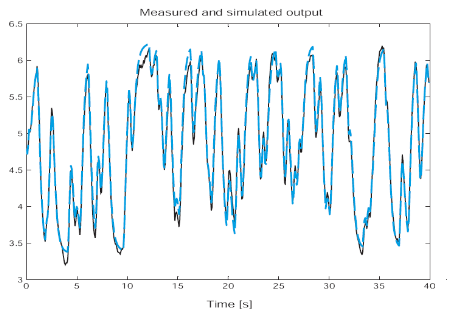
- Original model unnecessarily complex if I/O properties not much affected by model reduction
- Conserve spectrum/eigenvalues
- Numerical issues associated with matrix conditioning (e.g. plasma in optimization class)

Parameter confidence interval

- Compare estimate with corresponding estimated standard deviation
- If 0∈ confidence interval, the corresponding parameter may be removed
- Usually interesting if related to a physical property (model order or time-delay)
- If standard dev. are all large, information matrix close to singular and typically too large order

Simulation and prediction

- Split data into two parts; one for estimation and one for validation.
- Apply input signal in validation data set to estimated model
- Compare simulated output with output stored in validation data set.



Residual analysis

- Analyze the data not reproduced by model = *residual*

$$\epsilon(t) = \epsilon(t, \hat{\theta}_N) = y(t) - \hat{y}(t|\hat{\theta}_N)$$

- e.g. if we fit the parameters of the model

$$y(t) = G(q, \theta)u(t) + H(q, \theta)e(t)$$

to data, the residuals

$$\epsilon(t) = H(q, \theta)^{-1}\left[y(t) - G(q, \theta)u(t)\right]$$

represent a disturbance that explains mismatch between model and observed data.
- If the model is correct, the residuals should be:
  - white, and
  - uncorrelated with u

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Statistical model validation

- Pragmatic viewpoint: basic statistics from

$$S_1 = \max_t |\epsilon(t)|, \quad S_2^2 = \frac{1}{N}\sum_{t=1}^{N}\epsilon^2(t)$$

likely to hold for future data = invariance assumption (ϵ do not depend on something likely to change or on a particular input in Z^N)

$\Rightarrow$ Study covariance

$$\hat{R}_{\epsilon u}^N(\tau) = \frac{1}{N}\sum_{t=1}^{N}\epsilon(t)u(t-\tau), \quad \hat{R}_{\epsilon}^N(\tau) = \frac{1}{N}\sum_{t=1}^{N}\epsilon(t)\epsilon(t-\tau)$$

  - $\hat{R}_{\epsilon u}^N(\tau)$: if small, $S_{1,2}$ likely to be relevant for other inputs, otherwise, remaining traces of $y(t)$ not in $\mathcal{M}$
  - $\hat{R}_{\epsilon}^N(\tau)$: if not small for $\tau \neq 0$, part of $\epsilon(t)$ could have been predicted $\Rightarrow$ $y(t)$ could be better predicted

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Whiteness test

- Suppose that ϵ is a white noise with zero mean and variance λ, then

$$\frac{N}{\lambda^2}\sum_{\tau=1}^{M}\left(\hat{R}_{\epsilon}^N(\tau)\right)^2 = \frac{N}{\left(\hat{R}_{\epsilon}^N(\tau)\right)^2}\sum_{\tau=1}^{M}\left(\hat{R}_{\epsilon}^N(\tau)\right)^2 \doteq \zeta_{N,M}$$

should be asymptotically $\chi^2(M)$-distributed (independency test), e.g. if $\zeta_{N,M} < \chi^2_\alpha(M)$, the α level of $\chi^2(M)$
- Simplified rule: autocorrelation function $\sqrt{N}\hat{R}_{\epsilon}^N(\tau)$ lies within a 95% confidence region around zero $\rightarrow$ large components indicate unmodelled dynamics
- Similarly, independency if $\sqrt{N}\hat{R}_{\epsilon u}^N(\tau)$ within 95% confidence region around zero:
  - large components indicate unmodelled dynamics
  - $\hat{R}_{\epsilon u}^N(\tau)$ nonzero for $\tau < 0$ (non-causality) indicates the presence of feedback

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

Conclusions

System identification: an iterative procedure in several steps

- Experiment design
  - preliminary experiments detect basic system behavior
  - carefully designed experiment enable good model estimation (choice of sampling interval, anti-alias filters, input signal)
- Examination and prefiltering of data
  - remove outliers and trends
- Model structure selection
- Model validation
  - cross-validation and residual tests

Homework (Exam 2014)

You wish to obtain a model from an experimental process which allows you to perform all the desired tests and sequences of inputs.

1. Which preliminary experiments should be carried to get a preliminary idea of the system properties before the identification?
2. Suppose that you wish to evaluate the matching between the identified model:

$$G(q,\theta) = \frac{q^{-2}(b_1 + b_2 q^{-1} + b_3 q^{-2})}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3} + f_4 q^{-4} + f_5 q^{-5}}$$

   and your measured signals:
   1. which property has to be verified by your input signal?
   2. write the algorithm that would allow you to check this property.
3. If a local feedback controller is set on the experiment, how would you proceed to get valid measurements?

Experiment design and model validation
E. Witrant
Experiments and data collection
Preliminary experiments
Designing for estimation
Informative experiments
Open-loop experiments
Persistence of excitation
Input design
Crest factor
Common input
Periodic inputs
CL Identif
Guidelines
Model structure
Model validation
Residual analysis
Statistical validation
Homework

References

- L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.
- L. Ljung and T. Glad, *Modeling of Dynamic Systems*, Prentice Hall Information and System Sciences Series, 1994.
- Lecture notes from 2E1282 *Modeling of Dynamical Systems, Automatic Control*, School of Electrical Engineering, KTH, Sweden.

UFR de Physique

System Identification

Lecture 11: Nonlinear Black-box Identification

Emmanuel WITRANT
emmanuel.witrant@univ-grenoble-alpes.fr

October 10, 2017

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Motivation

Linear systems limited when considering:

- Physical models
- large parameter variability
- complex systems

Today's concerns:

- generic classes of models
- black box: neural networks and Artificial Intelligence
- parameter estimation for NL models: back on nonlinear programming

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Outline

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Nonlinear State-space Models

- General model set:

$$\begin{aligned} x(t+1) &= f(t, x(t), u(t), w(t); \theta) \\ y(t) &= h(t, x(t), u(t), v(t); \theta) \end{aligned}$$

  Nonlinear prediction → no finite-dimensional solution except specific cases: approximations
- Predictor obtained from simulation model (noise-free)

$$\begin{aligned} x(t+1, \theta) &= f(t, x(t, \theta), u(t), 0; \theta) \Leftrightarrow \frac{d}{dt}x(t, \theta) = f(\cdot) \\ \hat{y}(t|\theta) &= h(t, x(t, \theta), u(t), 0; \theta) \end{aligned}$$

- Include known physical parts of the model, but unmodeled dynamics that can still have a strong impact on the system

→ black-box components.

Nonlinear Black-box Models: Basic Principles

Model = mapping from past data Z^{t-1} to the space of output

$$\hat{y}(t|\theta) = g(Z^{t-1}, \theta)$$

→ seek parameterizations (parameters θ) of g that are flexible and cover "all kinds of reasonable behavior" ≡ nonlinear black-box model structure.

Nonlinear Black-box Identification
E. Witrant

Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

A structure for the general mapping: Regressors

Express g as a concatenation of two mappings:

- $\phi(t) = \phi(Z^{t-1})$: takes past observation into regression vector ϕ (components = regressors), or $\phi(t) = \phi(Z^{t-1}, \theta)$;
- $g(\phi(t), \theta)$: maps ϕ into space of outputs.

Two partial problems:

1. How to choose $\phi(t)$ from past I/O? Typically, using only measured quantities, i.e. NFIR (Nonlinear Finite Impulse Response) and NARX.
2. How to choose the nonlinear mapping $g(\phi, \theta)$ from regressor to output space?

Nonlinear Black-box Identification
E. Witrant

Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Basic features of function expansions and basis functions

- Focus on $g(\phi(t), \theta) : \mathbb{R}^d \to \mathbb{R}^p$, $\phi \in \mathbb{R}^d$, $y \in \mathbb{R}^p$.
- Parametrized function as family of function expansions

$$g(\phi, \theta) = \sum_{k=1}^{n} \alpha_k g_k(\phi),\ \theta = [\alpha_1\ \ldots\ \alpha_n]^T$$

  g_k referred as basis functions, provides a unified framework for most NL black-box model structures.
- How to choose g_k? Typically
  - all g_k formed from one "mother basis function" $\kappa(x)$;
  - $\kappa(x)$ depends on a scalar variable x;
  - g_k are dilated (scaled) and translated versions of κ, i.e. if $d = 1$ (scalar case)

$$g_k(\phi) = g_k(\phi, \beta_k, \gamma_k) = \kappa(\beta_k(\phi - \gamma_k))$$

  where β_k = dilatation and γ_k = translation.

Nonlinear Black-box Identification
E. Witrant

Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Scalar examples

- Fourier series: $\kappa(x) = \cos(x)$, g are Fourier series expansion, with β_k as frequencies and γ_k as phases.
- Piece-wise continuous functions: κ as unit interval indicator function

$$\kappa(x) = \begin{cases} 1 & \text{for } 0 \le x < 1 \\ 0 & \text{else} \end{cases}$$

  and $\gamma_k = k\Delta$, $\beta_k = 1/\Delta$, $\alpha_k = f(k\Delta)$: give a piece-wise constant approximation ∀ f over intervals of length Δ. Similar version with Gaussian bell $\kappa(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$.
- Piece-wise continuous functions - variant -: κ as unit step function

$$\kappa(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x > 0 \end{cases}$$

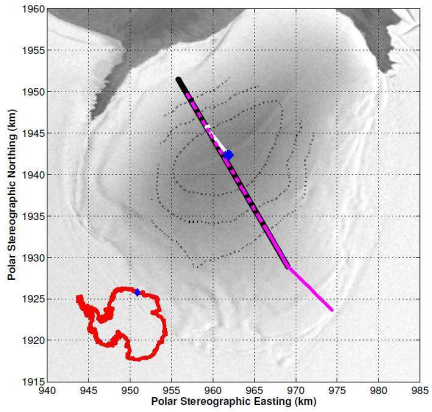
  Similar result with sigmoid function $\kappa(x) = \frac{1}{1+e^{-x}}$

Classification of single-variable basis functions

- local basis functions, with significant variations in local environment (i.e. presented piece-wise continuous functions);
- global basis functions, with significant variations over the whole real axis (i.e. Fourier, Voltera, Legendre polynomials).

Example: accumulation rate in Antarctica

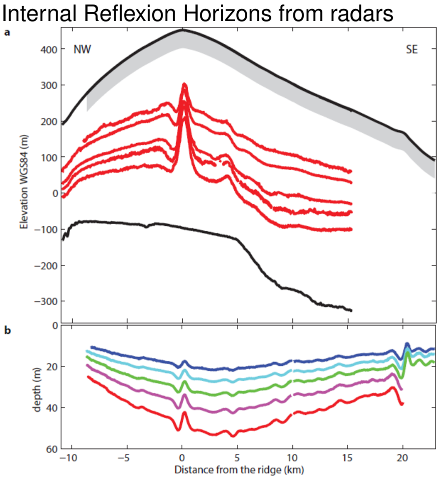
D. Callens, R. Drews, E. Witrant, M. Philippe, F. Pattyn: Temporally stable surface mass balance asymmetry across an icerise derived from radar internal reflection horizons through inverse modeling, Journal of Glaciology, 62(233) 525-534, 2016.



Map of Derwael Ice Rise

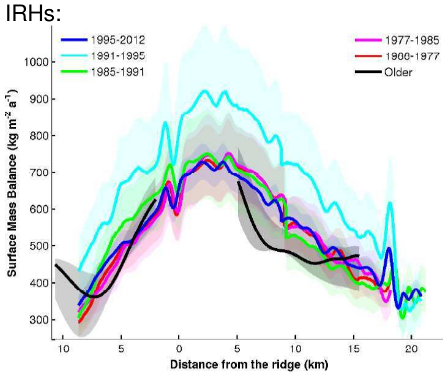
Example: accumulation rate in Antarctica (2)

Internal Reflexion Horizons from radars



Example: accumulation rate in Antarctica (3)

Spatial distribution of the SMB across the DIR inferred (inverse problem with Legendre polynomials) from younger and deeper IRHs:



→ asymmetric distribution related to orographic uplift of air masses which induces an increase of precipitation on the upwind side and a deficit on the downwind side (NW).

Construction of multi-variable basis functions ($\phi \in \mathbb{R}^d$, $d > 1$)

1. **Tensor product.** Product of the single-variable function, applied to each component of ϕ:

$$g_k(\phi) = g_k(\phi, \beta_k, \gamma_k) = \prod_{j=1}^{d} \kappa(\beta_k^j(\phi_j - \gamma_k^j))$$

2. **Radial construction.** Value depend only on ϕ's distance from a given center point

$$g_k(\phi) = g_k(\phi, \beta_k, \gamma_k) = \kappa\left(\|\phi - \gamma_k\|_{\beta_k}\right)$$

where $\|\cdot\|_{\beta_k}$ is any chosen norm, i.e. quadratic: $\|\phi\|_{\beta_k}^2 = \phi^T \beta_k \phi$ with $\beta_k > 0$ matrix.

3. **Ridge construction.** Value depend only on ϕ's distance from a given hyperplane (cst $\forall \phi$ in hyperplane)

$$g_k(\phi) = g_k(\phi, \beta_k, \gamma_k) = \kappa(\beta_k^T(\phi - \gamma_k))$$

Approximation issues

- For any of the described choices, the resulting model becomes

$$g(\phi, \theta) = \sum_{k=1}^{n} \alpha_k \kappa(\beta_k(\phi - \gamma_k))$$

- Fully determined by $\kappa(x)$ and the basis functions expansion on a vector ϕ.
- Parametrization in terms of θ characterized by three parameters: coordinates α, scale or dilatation β, location γ. Note: linear regression for fixed scale and location.
- Accuracy [Juditsky et al., 1995]: *for almost any choice of $\kappa(x)$ (except polynomial), we can approximate any "reasonable" function $g_0(\phi)$ (true system) arbitrarily well with n large enough.*

Approximation issues (2)

Efficiency [Barron 1993]:

1. if β and γ allowed to depend on the function g_0 then n much less than if β_k, γ_k fixed a priori;
2. for local, radial approach, necessary n to achieve a degree of approximation d of s times differentiable function:

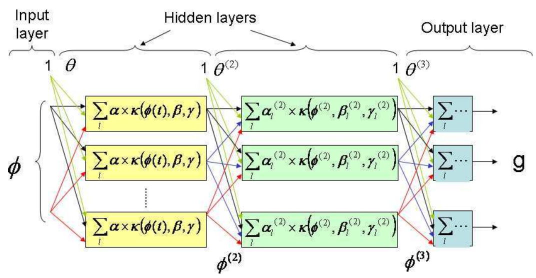
$$n \sim \frac{1}{\delta^{(d/s)}}, \ \delta \ll 1$$

$\rightarrow$ increases exponentially with the number of regressors = curse of dimensionality.

Networks for nonlinear black-box structures

Basis function expansions often referred to as networks.

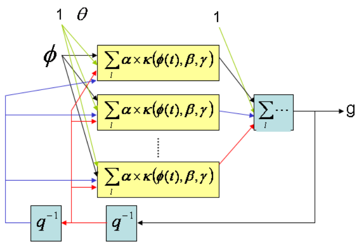
- **Multi-layer networks:**



Instead of taking a linear combination of regressors, treat as new regressors and introduce another "layer" of basis functions forming a second expansion, e.g. two-hidden layers network

Networks for nonlinear black-box structures (2)

- **Recurrent networks.** When some regressors at t are outputs from previous time instants $\phi_k(t) = g(\phi(t-k), \theta)$.



Estimation aspects

Asymptotic properties and basic algorithms are the same as the other model structures!

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Parameters estimation with *Gauss-Newton* stochastic gradient algorithm

$\Rightarrow$ A possible solution to determine the optimal parameters of each layer.

Problem description

Consider n_o system outputs $y \in \mathbb{R}^{n_m \times n_o}$, with n_m measurements for each output, and a model output $\hat{y} \in \mathbb{R}^{n_m \times n_o}$.

Objective: determine the optimal set of model parameters θ which minimizes the quadratic cost function

$$J(\theta) \doteq \frac{1}{n_m} \sum_{i=1}^{n_m} \|y(i) - \hat{y}(\theta, i)\|_2^2$$

Output error variance is minimized for $\theta^* = \arg\min_\theta J(\theta)$.

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Stochastic descent algorithm

Based on the sensitivity of $\hat{y}(\theta, i)$ with respect to θ

$$S(\theta, i) \doteq \frac{\partial \hat{y}}{\partial \theta} = \left[\frac{\partial \hat{y}}{\partial \theta_1}, \ldots, \frac{\partial \hat{y}}{\partial \theta_{n_v}}\right]^T,$$

the gradient of the cost function writes as

$$\nabla_\theta J(\theta) = -\frac{2}{n_m} \sum_{i=1}^{n_m} S(\theta, i)(y(i) - \hat{y}(\theta, i))$$

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Stochastic descent algorithm (2)

θ^* obtained by moving along the steepest slope $-\nabla_\theta J(\theta)$ with a step η, which as to ensure that

$$\theta^{l+1} = \theta^l - \eta^l \nabla_\theta J(\theta^l)$$

converges to θ^*, where $l \doteq$ algorithm iteration index. η^l chosen according to Gauss-Newton's method as

$$\eta^l \doteq (\Psi_\theta J(\theta^l) + \upsilon I)^{-1},$$

where $\upsilon > 0$ is a constant introduced to ensure strict positiveness and $\Psi_\theta J(\theta^l)$ is the pseudo-Hessian, obtained using Gauss-Newton approximation

$$\Psi_\theta J(\theta^l) = \frac{2}{n_m} \sum_{i=1}^{n_m} S(\theta^l, i) S(\theta^l, i)^T$$

Stochastic descent algorithm (3)

Consider dynamical systems modeled as ($t \in [0, T]$)

$$\begin{cases} \dfrac{dx_m}{dt} = f_m(x_m(t), u(t), \theta), \quad x_m(t_0) = x_{m0} \\ \hat{y}(t) = g_m(x_m(t), u(t), \theta) \end{cases}$$

x_m is the model state and $f_m(\cdot) \in C^1$, then

$$S(\theta, t) = \frac{\partial g_m}{\partial x_m}\frac{\partial x_m}{\partial \theta} + \frac{\partial g_m}{\partial \theta}$$

where the state sensitivity $\dfrac{\partial x_m}{\partial \theta}$ obtained by solving the ODE

$$\frac{d}{dt}\left[\frac{\partial x_m}{\partial \theta}\right] = \frac{\partial f_m}{\partial x_m}\frac{\partial x_m}{\partial \theta} + \frac{\partial f_m}{\partial \theta}$$

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Assumptions

- n_i independent system inputs $u = \{u_1, \ldots, u_{n_i}\} \in \mathbb{R}^{n_m \times n_i}$, available during the optimal parameter search process.
- The set $\{y, u\}$ corresponds to historic data and J is the data variance.
- The set of n_m measurements is large enough and well chosen (sufficiently rich input) to be considered as generators of persistent excitation to ensure that the resulting model represents the physical phenomenon accurately within the bounds of u.

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

For black-box models

Consider the nonlinear black-box structure

$$g(\phi, \theta) = \sum_{k=1}^{n} \alpha_k \kappa(\beta_k(\phi - \gamma_k))$$

To find the gradient $\nabla_\theta J(\theta)$ we just need to compute

$$\begin{aligned} \frac{\partial}{\partial \alpha}[\alpha\kappa(\beta(\phi-\gamma))] &= \kappa(\beta(\phi-\gamma)) \\ \frac{\partial}{\partial \beta}[\alpha\kappa(\beta(\phi-\gamma))] &= \alpha\frac{\partial}{\partial \beta}[\kappa(\beta(\phi-\gamma))]\phi \\ \frac{\partial}{\partial \gamma}[\alpha\kappa(\beta(\phi-\gamma))] &= -\alpha\frac{\partial}{\partial \gamma}[\kappa(\beta(\phi-\gamma))] \end{aligned}$$

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

Example: sigmoid functions family

$$\kappa_j = \frac{1}{1 + e^{-\beta_j(x-\gamma_j)}}$$

The sensitivity function is set with

$$\frac{\partial \hat{y}}{\partial \alpha_j} = \frac{1}{1 + e^{-\beta_j(x-\gamma_j)}}, \quad \frac{\partial \hat{y}}{\partial \beta_j} = \frac{\alpha_j e^{-\beta_j(x-\gamma_j)}(x-\gamma_j)}{(1 + e^{-\beta_j(x-\gamma_j)})^2},$$
$$\frac{\partial \hat{y}}{\partial \gamma_j} = -\frac{\alpha_j e^{-\beta_j(x-\gamma_j)}\beta_j}{(1 + e^{-\beta_j(x-\gamma_j)})^2}.$$

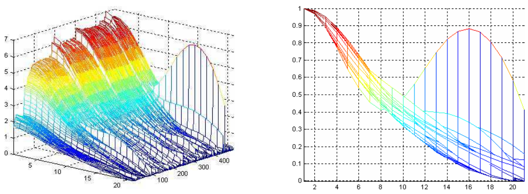
Notes:

- any continuous function can be arbitrarily well approximated using a superposition of sigmoid functions [Cybenko, 1989]
- nonlinear function $\Rightarrow$ nonlinear optimization problem

Temperature profile identification in tokamak plasmas

⇒ Parameter dependant identification of nonlinear distributed systems

- Grey-box modeling,
- 3-hidden layers approach: spatial distribution, steady-state and transient behaviour,
- Stochastic descent method with direct differentiation.



Identification method: TS Temperature profile (L-mode)

Physical model: $$\frac{3}{2}\frac{\partial nT}{\partial t} = \nabla\left(n\chi\nabla T\right) + S_T$$

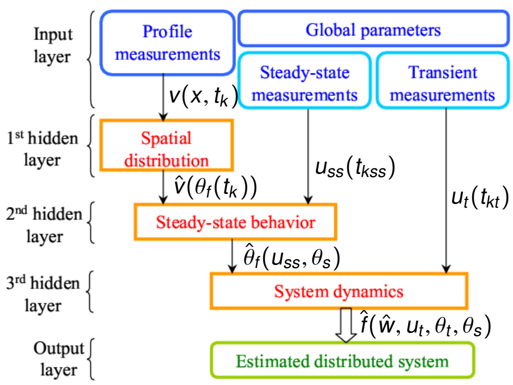
- Input: normalized profile $v(x,t) = \frac{T_e(x,t)}{T_{e0}(t)}$

1. $\hat{v}(x,t) = \frac{\alpha}{1+e^{-\beta(x-\gamma)}}, \quad \Rightarrow \theta_f = \{\alpha,\beta,\gamma\}$

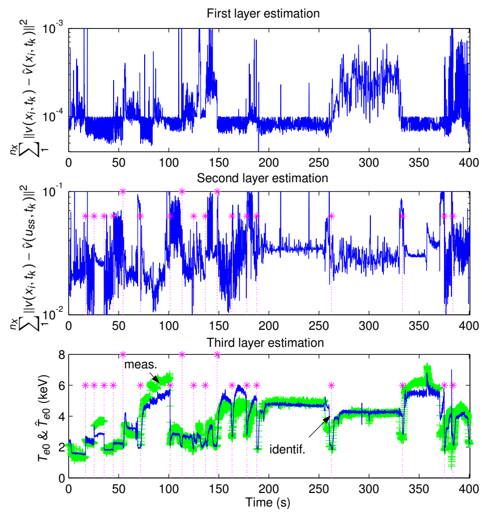
2. $$\begin{cases} \alpha_{lh} &= e^{\vartheta_{s\alpha 0}} I_p^{\vartheta_{s\alpha 1}} B_{\phi 0}^{\vartheta_{s\alpha 2}} N_{//}^{\vartheta_{s\alpha 3}} \left(1+\frac{P_{icrf}}{P_{tot}}\right)^{\vartheta_{s\alpha 4}} \\ \beta_{lh} &= -e^{\vartheta_{s\beta 0}} I_p^{\vartheta_{s\beta 1}} B_{\phi 0}^{\vartheta_{s\beta 2}} \bar{n}_e^{\vartheta_{s\beta 3}} N_{//}^{\vartheta_{s\beta 4}} \\ \gamma_{lh} &= e^{\vartheta_{s\gamma 0}} I_p^{\vartheta_{s\gamma 1}} B_{\phi 0}^{\vartheta_{s\gamma 2}} N_{//}^{\vartheta_{s\gamma 3}} \left(1+\frac{P_{icrf}}{P_{tot}}\right)^{\vartheta_{s\gamma 4}} \left(1+\frac{P_{icrf}}{P_{tot}}\right)^{\vartheta_{s\gamma 5}} \end{cases} \Rightarrow \theta_s = \{\vartheta_{s\alpha\, i}, \vartheta_{s\beta\, i}, \vartheta_{s\gamma\, i}\}$$

3. $$\begin{cases} \tau_{th}(t) &= e^{\vartheta_{t0}} I_p^{\vartheta_{t1}} B_{\phi 0}^{\vartheta_{t2}} \bar{n}_e^{\vartheta_{t3}} P_{tot}^{\vartheta_{t4}} \\ \frac{dW}{dt} &= P_{tot} - \frac{1}{\tau_{th}} W, \quad W(0) = P_{tot}(0)\tau_{th}(0) \\ \hat{T}_{e0}(t) &= \mathcal{A} W \end{cases} \Rightarrow \theta_t = \{\vartheta_{t,i}\}$$

Identification method (2)

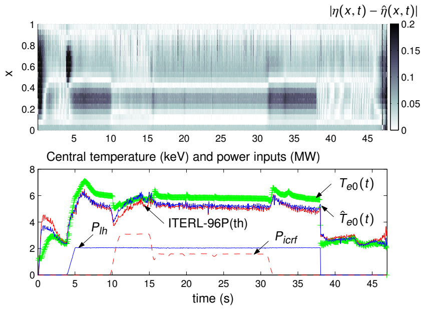


Results (19 shots - 9500 measurements)



Results (2)

Test case:



Conclusions

- Development similar to linear models
- Predictor → nonlinear function of past observations
- Unstructured black-box models much more demanding
- Clearly identify nonlinearities prior to identification: semi-physical models give the regressor
- Define sub-models that can be analyzed independently

Homework 5

1. Comment and write down the equations corresponding to the algorithm “fitTe sig.m” below.
2. How should the script be modified to use Gaussian fitting curves?

```
function [P,Jf] = fitTe_sig(TE,xval,P)
% TE = input temperature profile
% xval = location of the measurement
% P = Initial Conditions on design parameters
[np,nm] = size(TE); % number of profiles/measurements
xval = xval'; y = TE';
ni = 1000;        % number of iterations
nv = 3;                 % number of design parameters
nu = .1*eye(nv);        % conditioning parameter
J = zeros(ni,1);        % cost function
for j = 1:ni            % for each iteration
    GJ = zeros(nv,1);   % Gradient J
    DP2(:,j) = P;       % design parameters evolution recording
```



```
    y_est = zeros(nm,1); % model output (estimated system output)
    sigmoid = P(1)./(1+exp(-P(2).*(xval-P(3)))); % source terms
    y_est = y_est + sigmoid;
    dsigmoid_e = sigmoid.^2./P(1).*exp(-P(2).*(xval-P(3)));
S = [sigmoid./P(1)  (xval-P(3)).*dsigmoid_e -P(2).*dsigmoid_e];
                        % Sensitivity function dy/dK
    PJ = S'*S;          % pseudo-hessian \psi
    for i=1:np          % for each profile
        error = y(:,i)-y_est;
    % difference between the reference (0 in our case) and model
        for k = 1:nv    % for each design parameter
            GJt(k) = error'*S(:,k);
        end
        GJ = GJ + GJt';
        J(j) = J(j) + (error'*error)/np;
    end
    GJ = -2/np.*GJ;
    PJ = 2.*PJ;
    alpha = inv(PJ + nu);
    P = P - alpha*GJ;   % this is the veriation law for K
```

```
    if J(j) < 1*1e-4
 % if the cost function is sufficiently small we are happy and
 %  get out!
        TE_est = y_est';
        break
    end
end
Jf = J(j);
```

Nonlinear Black-box Identification
E. Witrant
Nonlinear State-space Models
Nonlinear Black-box Models
Regressors
Function expansions and basis functions
Multi-variable basis functions
Approximation issues
Networks
Parameters estimation with Gauss-Newton
Stochastic descent
Assumptions
For black-box models
Identification in tokamaks
Identification method
Results
Conclusions

References

1. L. Ljung, “System Identification: Theory for the User”, 2^{nd} Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.
2. E. Witrant and S. Brémond, ”Shape Identification for Distributed Parameter Systems and Temperature Profiles in Tokamaks”, Proc. of 50th IEEE Conference on Decision and Control, Orlando, USA, December 12-15, 2011.

Université Joseph Fourier GRENOBLE

UFR de Physique

System Identification

Lecture 12: Recursive Estimation Methods

Emmanuel WITRANT
`emmanuel.witrant@ujf-grenoble.fr`

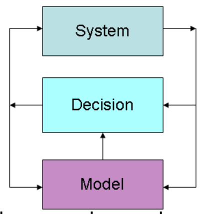
November 14, 2017

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Motivation

On-line model when the system is in operation to take decision about the system, i.e.

- Which input should be applied next?
- How to tune the filter parameters?
- What are the best predictions of next outputs?
- Has a failure occurred? Of what type?



$\Rightarrow$ Adaptive methods (control, filtering, signal processing and prediction).

Recursive estimation methods

- completed in one sampling interval to keep up with information flow;
- carry on their estimate of parameter variance;
- also competitive for off-line situations.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Overview

- General mapping of data set to parameter space $\hat{\theta}_t = F(t, Z^t)$ may involve an unknown large amount of calculation for F.
- Recursive algorithm format:
  - $X(t)$: information state

$$\begin{aligned} X(t) &= H(t, X(t-1), y(t), u(t)) \\ \hat{\theta}_t &= h(X(t)) \end{aligned}$$

  - H & h: explicit expressions involving limited calculations

  $\Rightarrow$ $\hat{\theta}_t$ evaluated during a sampling interval
- small information content in latest measurements pair (γ_t and μ_t):

$$\begin{aligned} \hat{\theta}_t &= \hat{\theta}_{t-1} + \gamma_t Q_\theta(X(t), y(t), u(t)) \\ X(t) &= X(t-1) + \mu_t Q_X(X(t), y(t), u(t)) \end{aligned}$$

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Outline

The Recursive Least-Squares Algorithm

Weighted LS criterion

$$\hat{\theta}_t = \arg\min_{\theta} \sum_{k=1}^{t} \beta(t,k)\left[y(k) - \phi^T(k)\theta\right]^2$$

where ϕ is the regressor, has solution

$$\begin{cases} \hat{\theta}_t &= \bar{R}^{-1}(t) f(t) \\ \bar{R}(t) &= \sum_{k=1}^{t} \beta(t,k)\phi(k)\phi^T(k) \\ f(t) &= \sum_{k=1}^{t} \beta(t,k)\phi(k)y(k) \end{cases}$$

Z^t and $\hat{\theta}_{t-1}$ cannot be directly used, even if closely related to $\hat{\theta}_t$.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Recursive algorithm

- Suppose the weighting sequence properties

$$\left.\begin{array}{rcl} \beta(t,k) &=& \lambda(t)\beta(t-1,k), \quad 0 \le k \le t-1 \\ \beta(t,t) &=& 1 \end{array}\right\} \Rightarrow \quad \beta(t,k) = \prod_{k+1}^{t} \lambda(j)$$

where $\lambda(t)$ is the forgetting factor. It implies that

$$\begin{array}{rcl} \bar{R}(t) &=& \lambda(t)\bar{R}(t-1) + \phi(t)\phi^T(t) \\ f(t) &=& \lambda(t) f(t-1) + \phi(t) y(t) \\ \Rightarrow \hat{\theta}_t &=& \bar{R}^{-1}(t) f(t) = \hat{\theta}_{t-1} + \bar{R}^{-1}(t)\phi(t)\left[y(t) - \phi^T(t)\hat{\theta}_{t-1}\right] \end{array}$$

Exercise: prove it

- At $(t-1)$ we only need to store the information vector $X(t-1) = [\hat{\theta}_{t-1}, \bar{R}(t-1)]$.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Efficient matrix inversion

To avoid inverting $\bar{R}(t)$ at each step, introduce $P(t) = \bar{R}^{-1}(t)$ and the matrix inversion lemma

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B + C^{-1}]^{-1}DA^{-1}$$

with $A = \lambda(t)\bar{R}(t-1)$, $B = D^T = \phi(t)$ and $C = 1$ to obtain

$$\begin{cases} \hat{\theta}(t) &= \hat{\theta}(t-1) + L(t)\left[y(t) - \phi^T(t)\hat{\theta}(t-1)\right] \\ L(t) &= \dfrac{P(t-1)\phi(t)}{\lambda(t) + \phi^T(t)P(t-1)\phi(t)} \\ P(t) &= \dfrac{1}{\lambda(t)}\left[P(t-1) - L(t)\phi^T(t)P(t-1)\right] \end{cases}$$

Note: we used $\hat{\theta}(t)$ instead of $\hat{\theta}_t$ to account for slight differences due to the IC.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Normalized gain version

To bring out the influence of $\bar{R}$ & $\lambda(t)$ on $\hat{\theta}_{t-1}$, normalize as

$$R(t) = \gamma(t)\bar{R}(t), \quad \gamma(t) = \left[\sum_{k=1}^{t}\beta(t,k)\right]^{-1} \Rightarrow \frac{1}{\gamma(t)} = \frac{\lambda(t)}{\gamma(t-1)} + 1$$

It implies that

$$\begin{array}{rcl} \hat{\theta}_t &=& \hat{\theta}_{t-1} + \bar{R}^{-1}(t)\phi(t)\left[y(t) - \phi^T(t)\hat{\theta}_{t-1}\right] \\ \bar{R}(t) &=& \lambda(t)\bar{R}(t-1) + \phi(t)\phi^T(t) \end{array}$$

$$\text{becomes} \begin{cases} \epsilon(t) &= y(t) - \phi^T(t)\hat{\theta}(t-1) \\ \hat{\theta}(t) &= \hat{\theta}(t-1) + \gamma(t)R^{-1}\phi(t)\epsilon(t) \\ R(t) &= R(t-1) + \gamma(t)\left[\phi(t)\phi^T(t) - R(t-1)\right] \end{cases}$$

- $R(t)$: weighted arithmetic mean of $\phi(t)\phi^T(t)$;
- $\epsilon(t)$: prediction error according to current model;
- $\gamma(t)$: updating step size or gain of the algorithm.

Initial conditions

- Ideally, $\bar{R}(0)=0$, $\hat{\theta}_0=\theta_I$ but cannot be used ($\bar{R}^{-1}$) $\rightarrow$ initialize when $\bar{R}(t_0)$ invertible: spare t_0 samples s.t.

$$\begin{cases} P^{-1}(t_0)=\bar{R}(t_0) & = \displaystyle\sum_{k=1}^{t_0}\beta(t_0,k)\phi(k)\phi^T(k) \\ \hat{\theta}_0 & = P(t_0)\displaystyle\sum_{k=1}^{t_0}\beta(t_0,k)\phi(k)y(k) \end{cases}$$

- Simpler alternative: use $P(0)=P_0$ and $\hat{\theta}(0)=\theta_I$, which gives

$$\hat{\theta}(t)=\left[\beta(t,0)P_0^{-1}+\sum_{k=1}^{t}\beta(t,k)\phi(k)\phi^T(k)\right]^{-1}\left[\beta(t,0)P_0^{-1}\theta_I+\sum_{k=1}^{t}\beta(t,k)\phi(k)y(k)\right]$$

If P_0 and t large, insignificant difference.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Weighted multivariable case

$$\hat{\theta}_t=\arg\min_{\theta}\frac{1}{2}\sum_{k=1}^{t}\beta(t,k)\left[y(k)-\phi^T(k)\theta\right]^T\Lambda_k^{-1}\left[y(k)-\phi^T(k)\theta\right]$$

gives, similarly to the scalar case

$$\begin{cases} \hat{\theta}(t) & = \hat{\theta}(t-1)+L(t)\left[y(t)-\phi^T(t)\hat{\theta}(t-1)\right] \\ L(t) & = P(t-1)\phi(t)\left[\lambda(t)\Lambda_t+\phi^T(t)P(t-1)\phi(t)\right]^{-1} \\ P(t) & = \dfrac{1}{\lambda(t)}\left[P(t-1)-L(t)\phi^T(t)P(t-1)\right] \end{cases}$$

and (normalized gain) $\begin{cases} \epsilon(t) & = y(t)-\phi^T(t)\hat{\theta}(t-1) \\ \hat{\theta}(t) & = \hat{\theta}(t-1)+\gamma(t)R^{-1}\phi(t)\Lambda_t^{-1}\epsilon(t) \\ R(t) & = R(t-1)+\gamma(t)\left[\phi(t)\Lambda_t^{-1}\phi^T(t)-R(t-1)\right] \end{cases}$

Note: can also be used for the scalar case with weighted norm

$$\beta(t,k)=\alpha_k\prod_{k+1}^{t}\lambda(j),$$ where the scalar α_k corresponds to Λ_k^{-1}

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Kalman filter interpretation

- The Kalman filter for $\begin{cases} x(t+1) & = F(t)x(t)+w(t) \\ y(t) & = H(t)x(t)+v(t) \end{cases}$ is

$$\begin{cases} \hat{x}(t+1) & = F(t)\hat{x}(t)+K(t)[y(t)-H(t)\hat{x}(t)], \quad \hat{x}(0)=x_0, \\ K(t) & = [F(t)P(t)H^T(t)+R_{12}(t)][H(t)P(t)H^T(t)+R_2(t)]^{-1} \\ P(t+1) & = F(t)P(t)F^T(t)+R_1(t)-K(t)[H(t)P(t)H^T(t)+R_2(t)]K^T(t), \\ & \quad P(0)=\Pi_0. \end{cases}$$

with $R_1(t)=Ew(t)w^T(t)$, $R_{12}(t)=Ew(t)v^T(t)$, $R_2(t)=Ev(t)v^T(t)$

- The linear regression model $\hat{y}(t|\theta)=\phi^T(t)\theta$ can be expressed as $\begin{cases} \theta(t+1) & = I\theta(t)+0, \quad (\equiv\theta) \\ y(t) & = \phi^T(t)\theta(t)+v(t) \end{cases}$

Corresponding KF: ($\Lambda_t \doteq R_2(t)$)
$$\begin{cases} \theta(t+1) & = \theta(t)+K(t)[y(t)-\phi^T(t)\theta(t)], \\ K(t) & = P(t)\phi(t)[\phi^T(t)P(t)\phi(t)+\Lambda_t]^{-1}, \\ P(t+1) & = P(t)-K(t)[\phi^T(t)P(t)\phi(t)+\Lambda_t]K^T(t). \end{cases}$$

= exactly the multivariable case formulation if $\lambda(t)\equiv 1$!

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Resulting practical hints

- if $v(t)$ is white and Gaussian, then the posteriori distribution of $\theta(t)$, given Z^{t-1}, is Gaussian with mean value $\hat{\theta}(t)$ and covariance $P(t)$;
- IC: $\hat{\theta}(0)$ is the mean and $P(0)$ the covariance of the prior distribution $\rightarrow$ $\hat{\theta}(0)$ is our guess before seing the data and $P(0)$ reflects our confidence in that guess;
- the natural choice for $|\Lambda_t|$ is the error noise covariance matrix. If (scalar) $\alpha_t^{-1}=Ev^2(t)$ is time-varying, use $\beta(k,k)=\alpha_k$ in weighted criterion.

Time-varying systems

- Adaptive methods and recursive algorithms: time-varying system properties $\Rightarrow$ track these variations.

$\Rightarrow$ Assign less weight to older measurements: choose $\lambda(j) < 1$, i.e. if $\lambda(j) \equiv \lambda$, then $\beta(t,k) = \lambda^{t-k}$ and old measurements are exponentially discounted: λ is the forgetting factor. Consequently $\gamma(t) \equiv \gamma = 1 - \lambda$

- OR have the parameter vector vary like random walk

$$\theta(t+1) = \theta(t) + w(t), \quad Ew(t)w^T(t) = R_1(t)$$

with w white Gaussian and $Ev^2(t) = R_2(t)$.

Kalman filter gives conditional expectation and covariance of $\hat{\theta}$ as:

$$\begin{cases} \hat{\theta}(t) &= \hat{\theta}(t-1) + L(t)\left[y(t) - \phi^T(t)\hat{\theta}(t-1)\right] \\ L(t) &= \dfrac{P(t-1)\phi(t)}{R_2(t) + \phi^T(t)P(t-1)\phi(t)} \\ P(t) &= P(t-1) - L(t)\phi^T(t)P(t-1) + R_1(t) \end{cases}$$

$\Rightarrow R_1(t)$ prevents $L(t)$ from tending to zero.

Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Example: parametrization of the plasma resistivity profile

Consider the time and space dependant $\eta(x,t)$ (shot 35109), approximated with the scaling law

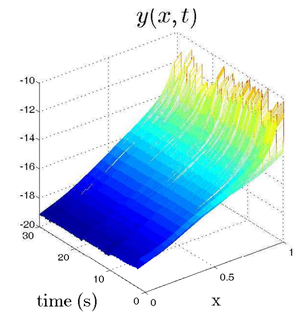
$$\hat{\eta}(x,t,\theta(t)) \doteq e^{\theta_1} e^{\theta_2 x} e^{\theta_3 x^2} \dots e^{\theta_{N_\theta} x^{N_\theta - 1}}$$

where $x \in \mathbb{R}^{N_x}$ and $\theta = \theta(t) \in \mathbb{R}^{N_\theta}$, then

- the data is processed as $y(x,t) = \ln \eta(x,t)$
- the model is parameterized as

$$\hat{y}(t,\theta) \doteq \ln \hat{\eta}(x,t,\theta(t))$$

$$= \underbrace{[1\ x\ x^2\ \dots x^{N_\theta - 1}]}_{\Phi^T \in \mathbb{R}^{N_x \times N_\theta}} \underbrace{\begin{bmatrix} \theta_1(t) \\ \theta_2(t) \\ \vdots \\ \theta_{N_\theta}(t) \end{bmatrix}}_{\theta(t)}$$



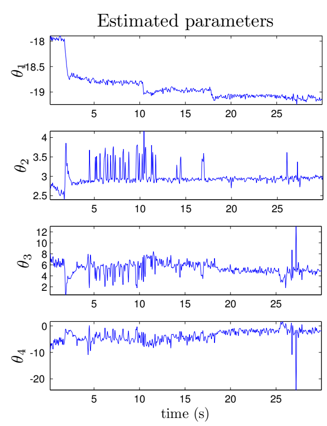
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

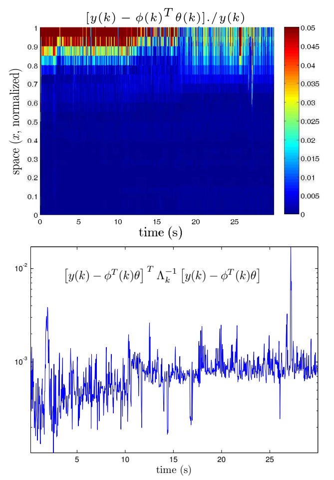
Example (2): code for multivariable case

```
lambda = .5;                          % forgetting factor
weight = exp(-x.^2./.07);             % gaussian distribution
Lambda = inv(diag(weight));           % gauss. weight
P = inv(lambda.*Phi*Phi');            % initial P
for j = 1:Nt                          % time loop
    y_t = y(:,j);                     % acquire measurement
    epsilon = y_t - Phi'*Theta;       % prediction err.
    Theta_r(:,j) = Theta;             % store param. at t
    L = P*Phi*inv(lambda.*Lambda + Phi'*P*Phi);
    P = (P - L*Phi'*P)./lambda;       % update P
    Theta = Theta + L*epsilon;        % update Theta
    y_est(:,j) = Phi'*Theta;          % record estimation
    cost(j) = epsilon'*inv(Lambda)*epsilon;
end
```

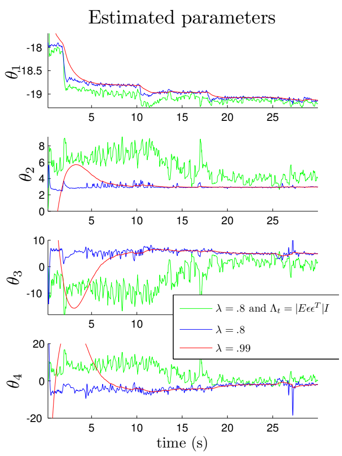

Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

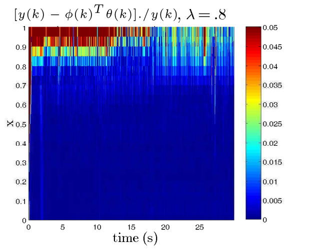
Simulation results

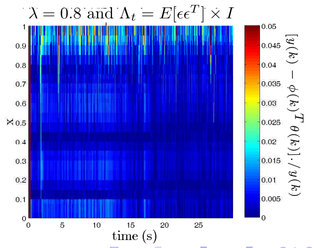




Simulation results (2): effect of λ and Λ_t







Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

The Recursive IV Method

Instrumental Variables (IV)

- Linear regression model: $\hat{y}(t|\theta) = \phi^T(t)\theta$

$$\Rightarrow \quad \hat{\theta}_N^{LS} = sol\left\{\frac{1}{N}\sum_{t=1}^{N}\phi(t)[y(t)-\phi^T(t)\theta] = 0\right\}$$

- Actual data: $y(t) = \phi^T(t)\theta_0 + v_0(t)$. LSE $\hat{\theta}_N \nrightarrow \theta_0$ typically, because of the correlation between $v_0(t)$ and $\phi(t)$: introduce a general correlation vector $\zeta(t)$, which elements are called the instruments or instrumental variables.
- IV estimation:

$$\hat{\theta}_N^{IV} = sol\left\{\frac{1}{N}\sum_{t=1}^{N}\zeta(t)[y(t)-\phi^T(t)\theta] = 0\right\} = \left[\frac{1}{N}\sum_{t=1}^{N}\zeta(t)\phi^T(t)\right]^{-1}\frac{1}{N}\sum_{t=1}^{N}\zeta(t)y(t)$$

$$\text{Requires} \begin{cases} \bar{E}\zeta(t)\phi^T(t) \text{ nonsingular} & \text{IV cor. with } \phi, \\ \bar{E}\zeta(t)v_0(t) = 0 & \text{IV not cor. with noise} \end{cases}$$

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Choice of Instruments: i.e. ARX

Supposing the true system:

$$y(t) \quad + \quad a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) + v(t)$$

Choose the IV similar to the previous model, while ensuring the correlation constraints:

$$\zeta(t) = K(q)[-x(t-1) \ \ldots \ -x(t-n_a) \ u(t-1) \ \ldots \ u(t-n_b)]^T,$$

where K is a filter and $N(q)x(t) = M(q)u(t)$ (i.e. N, M from LS estimated model and $K(q) = 1$ for open-loop).
$\Rightarrow \zeta$ obtained from filtered past inputs: $\zeta(t) = \zeta(t, u^{t-1})$

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Recursive IV method

- Rewrite the IV method as

$$\hat{\theta}_N^{IV} = \bar{R}^{-1}(t)f(t), \text{ with } \bar{R}(t) = \sum_{k=1}^{N}\beta(t,k)\zeta(k)\phi^T(k), \ f(t) = \sum_{k=1}^{N}\beta(t,k)\zeta(k)y(k)$$

which implies that

$$\begin{cases} \hat{\theta}(t) & = & \hat{\theta}(t-1) + L(t)\left[y(t) - \phi^T(t)\hat{\theta}(t-1)\right] \\ L(t) & = & \dfrac{P(t-1)\zeta(t)}{\lambda(t) + \phi^T(t)P(t-1)\zeta(t)} \\ P(t) & = & \dfrac{1}{\lambda(t)}\left[P(t-1) - L(t)\phi^T(t)P(t-1)\right] \end{cases}$$

- Asymptotic behavior: same as off-line counterpart except for the initial condition issue.
- Choice of the IV (i.e. model-dependant): $\zeta(t,\theta) = K_u(q,\theta)u(t)$ with K_u a linear filter and $\zeta(t,\theta) : \{x(t,\theta), u(t)\}$ with $A(q,\theta)x(t,\theta) = B(q,\theta)u(t)$.

Recursive Prediction-Error Methods

Weighted prediction-error criterion

$$V_t(\theta, Z^t) = \gamma(t)\frac{1}{2}\sum_{k=1}^{t}\beta(t,k)\epsilon^2(k,\theta),$$

with γ, β as defined above ($\beta(t,k) = \lambda(t)\beta(t-1,k)$, $\beta(t,t) = 1$). Note that $\sum_{k=1}^{t}\gamma(t)\beta(t,k) = 1$ and the gradient w.r.t. θ obeys (with $\epsilon(k,\theta) = y(k) - \hat{y}(k,\theta)$ and $\psi \doteq \partial\hat{y}/\partial\theta$):

$$\begin{aligned}
\nabla_\theta V_t(\theta, Z^t) &= -\gamma(t)\sum_{k=1}^{t}\beta(t,k)\psi(k,\theta)\epsilon(k,\theta),\\
&= \gamma(t)\left[\lambda(t)\frac{1}{\gamma(t-1)}\nabla_\theta V_{t-1}(\theta, Z^{t-1}) - \psi(t,\theta)\epsilon(t,\theta)\right]\\
&= \nabla_\theta V_{t-1}(\theta, Z^{t-1}) + \gamma(t)\left[-\psi(t,\theta)\epsilon(t,\theta) - \nabla_\theta V_{t-1}(\theta, Z^{t-1})\right]
\end{aligned}$$

since $\lambda(t)\gamma(t)/\gamma(t-1) = 1 - \gamma(t)$.

Recursive method

General search algorithm (i^{th} iteration of min. and Z^t data):

$$\hat{\theta}_t^{(i)} = \hat{\theta}_t^{(i-1)} - \mu_t^{(i)}\left[R_t^{(i)}\right]^{-1}\nabla_\theta V_t(\hat{\theta}_t^{(i-1)}, Z^t),$$

Suppose one more data point collected at each iteration:

$$\hat{\theta}(t) = \hat{\theta}(t-1) - \mu_t^{(t)}\left[R(t)\right]^{-1}\nabla_\theta V_t(\hat{\theta}(t-1), Z^t),$$

where $\hat{\theta}(t) = \hat{\theta}_t^{(t)}$ and $R(t) = R_t^{(t)}$. Make the induction assumption that $\hat{\theta}(t-1)$ minimized $V_{t-1}(\theta, Z^{t-1})$:

$$\nabla_\theta V_{t-1}(\hat{\theta}(t-1), Z^{t-1}) = 0$$

$$\Rightarrow \nabla_\theta V_t(\hat{\theta}(t-1), Z^t) = -\gamma(t)\psi(t,\hat{\theta}(t-1))\epsilon(t,\hat{\theta}(t-1))$$

along with $\mu(t) = 1$, it gives

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \gamma(t)R^{-1}(t)\psi(t,\hat{\theta}(t-1))\epsilon(t,\hat{\theta}(t-1))$$

Recursive method (2)

- **Problem:** $\psi(t,\hat{\theta}(t-1))$, $\epsilon(t,\hat{\theta}(t-1))$ derived from $\hat{y}(t,\hat{\theta}(t-1))$ & $\hat{y}(t,\theta)$ requires the knowledge of all the data Z^{t-1}, and consequently cannot be computed recursively.
- **Assumption:** at k, replace θ by the currently available estimate $\hat{\theta}(k)$ and denote the approximation of $\psi(t,\hat{\theta}(t-1))$ and $\hat{y}(t,\hat{\theta}(t-1))$ by $\psi(t)$ and $\hat{y}(t)$.

Ex.1 **Finite LPV:**
$$\left\{\begin{array}{rcl}
\xi(t+1,\theta) &=& A(\theta)\xi(t,\theta) + B(\theta)\begin{bmatrix} y(t)\\ u(t)\end{bmatrix}\\
\begin{bmatrix}\hat{y}(t|\theta)\\ \psi(t,\theta)\end{bmatrix} &=& C(\theta)\xi(t,\theta)
\end{array}\right.$$

$$\approx\left\{\begin{array}{rcl}
\xi(t+1) &=& A(\hat{\theta}(t))\xi(t) + B(\hat{\theta}(t))\begin{bmatrix} y(t)\\ u(t)\end{bmatrix}\\
\begin{bmatrix}\hat{y}(t)\\ \psi(t)\end{bmatrix} &=& C(\hat{\theta}(t-1))\xi(t).
\end{array}\right.$$

Ex.2: **Gauss-Newton**

$$\frac{\partial^2 V_N(\theta, Z^N)}{\partial\theta^2} \approx \frac{1}{N}\sum_{1}^{N}\psi(t,\theta)\psi^T(t,\theta) \doteq H_N(\theta),\ \&\ R_N^{(i)} = H_N(\hat{\theta}_N^{(i)}),$$

with the proposed approximation suggests that

$$R(t) = \gamma(t)\sum_{k=1}^{t}\beta(t,k)\psi(k)\psi^T(k).$$

Final recursive scheme

$$\left\{\begin{array}{rcl}
\epsilon(t) &=& y(t) - \hat{y}(t)\\
\hat{\theta}(t) &=& \hat{\theta}(t-1) + \gamma(t)R^{-1}(t)\psi(t)\epsilon(t)\\
R(t) &=& R(t-1) + \gamma(t)\left[\psi(t)\psi^T(t) - R(t-1)\right]
\end{array}\right.$$

Together with $R(t)$ from Gauss-Newton example $\rightarrow$ recursive Gauss-Newton prediction-error algorithm.

Family of recursive prediction-error methods (RPEM)

- Wide family of methods depending on the underlying model structure & choice of $R(t)$.
- **Example:** linear regression $\hat{y}(t|\theta) = \phi^T(t)\theta$ gives $\psi(t,\theta) = \psi(t) = \phi(t)$, the RLS method. Gradient variant ($R(t) = I$) on the same structure:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \gamma(t)\phi(t)\epsilon(t)$$

where the gain $\gamma(t)$ is a given sequence or normalized as $\gamma(t) = \gamma'(t)/|\phi(t)|^2$ widely used in adaptive signal processing and called LMS (least mean squares).

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Example: recursive maximum likelihood

Consider the ARMAX model

$$\begin{aligned} y(t) &+ a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) \\ &+ e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c) \end{aligned}$$

and define $\theta \doteq [a_1 \ \dots \ a_{n_a} \ b_1 \ \dots \ b_{n_b} \ c_1 \ \dots \ c_{n_c}]^T$. Introduce the vector

$$\phi(t,\theta) = [-y(t-1) \ \dots \ -y(t-n_a) \ u(t-1) \ \dots \ u(t-n_b) \ \epsilon(t-1,\theta) \ \dots \ \epsilon(t-n_c,\theta)]^T,$$

$$\Rightarrow \begin{cases} \hat{y}(t|\theta) = \phi^T(t,\theta)\theta, \quad \epsilon(t,\theta) = y(t) - \hat{y}(t|\theta) \\ \psi(t,\theta) + c_1\psi(t-1,\theta) + \cdots + c_{n_c}\psi(t-n_c,\theta) = \phi(t,\theta) \end{cases}$$

The previous simplifying assumption implies that

$$\begin{aligned} \bar{\epsilon}(t) &= y(t) - \phi^T(t)\hat{\theta}(t) \\ \phi(t) &= [-y(t-1) \ \dots \ -y(t-n_a) \ u(t-1) \ \dots \ u(t-n_b) \ \bar{\epsilon}(t-1,\theta) \ \dots \ \bar{\epsilon}(t-n_c,\theta)]^T \\ \hat{y}(t) &= \phi^T(t)\hat{\theta}(t-1); \quad \epsilon(t) = y(t) - \hat{y}(t) \\ \psi(t) &+ \hat{c}_1(t-1)\psi(t-1) + \cdots + \hat{c}_{n_c}\psi(t-n_c) = \phi(t) \end{aligned}$$

and the algorithm becomes $\hat{\theta}(t) = \hat{\theta}(t-1) + \gamma(t)R^{-1}(t)\psi(t)\epsilon(t)$
$\Rightarrow$ Recursive Maximum Likelihood (RML) scheme.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Recursive Pseudolinear Regressions

Very similar to *Recursive Prediction-Error Methods* except that the gradient is replaced by the regressor:

$$\begin{cases} \hat{y}(t) &= \phi^T(t)\hat{\theta}(t-1) \\ \epsilon(t) &= y(t) - \hat{y}(t) \\ \hat{\theta}(t) &= \hat{\theta}(t-1) + \gamma(t)R^{-1}(t)\phi(t)\epsilon(t) \\ R(t) &= R(t-1) + \gamma(t)\left[\phi(t)\phi^T(t) - R(t-1)\right] \end{cases}$$

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Choice of Updating Step

How to determine the update direction and length of the step ($\gamma(t)R^{-1}(t)$)?

Update direction

1. Gauss-Newton: $R(t)$ approximates the Hessian

$$R(t) = R(t-1) + \gamma(t)\left[\psi(t)\psi^T(t) - R(t-1)\right]$$

2. Gradient: $R(t)$ is a scaled identity $R(t) = |\psi(t)|^2 \cdot I$ or

$$R(t) = R(t-1) + \gamma(t)\left[|\psi(t)|^2 \cdot I - R(t-1)\right]$$

$\rightarrow$ trade-off between convergence rate (Gauss-Newton, d^2 operations) and algorithm complexity (gradient, d operations)

Update step: adaptation gain

Two ways to cope with time-varying aspects:

1. select appropriate forgetting profile $\beta(t,k)$ or suitable gain $\gamma(t)$, equivalent as

$$\beta(t,k) = \prod_{j=k+1}^{t} \lambda(j) = \frac{\gamma(k)}{\gamma(t)} \prod_{j=k+1}^{t} (1-\gamma(j)),$$

$$\lambda(t) = \frac{\gamma(t-1)}{\gamma(t)}(1-\gamma(t)) \Leftrightarrow \gamma(t) = \left[1 + \frac{\lambda(t)}{\gamma(t-1)}\right]^{-1};$$

2. introduce covariance matrix $R_1(t)$ for parameters change per sample: $\nearrow P(t)$ and consequently the gain vector $L(t)$.

$\rightarrow$ trade-off between tracking ability and noise sensitivity.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Choice of forgetting factors $\lambda(t)$

- Select the forgetting profile $\beta(t,k)$ so that the criterion keeps the relevant measurements for the current properties.
- For "quasi-stationary" systems, constant factor $\lambda(t) \equiv \lambda$ slightly < 1:

$$\beta(t,k) = \lambda^{t-k} = e^{(t-k)\ln\lambda} \approx e^{-(t-k)(1-\lambda)}$$

$\Rightarrow$ measurements older than memory time constant $t-k = \frac{1}{1-\lambda}$ included with a weight $< e^{-1} \approx 36\%$ (good if the system remains approximately constant over $t-k$ samples). Typically, $\lambda \in [0.98,\ 0.995]$.

- If the system undergoes abrupt and sudden changes, choose adaptive $\lambda(t)$: $\searrow$ temporary if abrupt change ("cut off" past measurements).

$\rightarrow$ trade-off between tracking alertness and noise sensitivity.

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Conclusions

- Instruments for most adaptation schemes
- Derived from off-line methods by setting a new iteration when a new observation is performed
- Same results off/on line for specific cases (RLS, RIV) but data not maximally utilized
- Asymptotic properties of RPEM for constant systems are the same as off-line: the previous analysis hold
- 2 new important quantities: update direction and gains
- Can be applied to both "deterministic" and "stochastic" systems

Recursive Estimation Methods
E. Witrant
Recursive LS
Recursive algorithm
Matrix inversion
Normalized gain
Initial conditions
Multivariable case
Kalman filter
Time-varying systems
Example
IV Method
Choice of Instruments
Recursive IV method
Recursive PEM
Recursive method
Final recursive scheme
Family of RPEM
Example
Recursive Pseudolinear Regressions
Updating Step
Adaptation gain
Forgetting factors
Conclusions
Homework

Homework 6

1. Apply RPEM to a first-order ARMA model

$$y(t) + ay(t-1) = e(t) + ce(t-1).$$

Derive an explicit expression for the difference

$$\hat{y}(t) - \hat{y}(t|\hat{\theta}(t-1)).$$

Discuss when this difference will be small.

2. Consider $\gamma(t) = \left[\sum_{k=1}^{t} \beta(t,k)\right]^{-1}$ and $\beta(t,k)$ defined by

$$\left.\begin{array}{rcl} \beta(t,k) &=& \lambda(t)\beta(t-1,k), \quad 0 \le k \le t-1 \\ \beta(t,t) &=& 1 \end{array}\right\}$$

Show that $\beta(t,k) = \dfrac{\gamma(k)}{\gamma(t)} \displaystyle\prod_{j=k+1}^{t} [1-\gamma(j)]$

Reference

1. L. Ljung, “System Identification: Theory for the User”, 2^{nd} Edition, Information and System Sciences, (Upper Saddle River, NJ: PTR Prentice Hall), 1999.