Modeling from data: Physics + System Identification

plus a little Philosophy
Key Ideas

Models should be in a useful form
  Linear system difference equations are good for design
    Linear system plus memoryless nonlinearities OK too
    Interconnections of simple components

Simplicity is a major goal
  Occam’s razor, parsimony or even simpler
    Complexity hurts us downstream

Approximation is a necessity and is desirable
  No exact match is possible

Characterize model performance in a sensible way
  Try to reflect the ultimate model usage
Prediction error modeling

\[ P(z) + H(z) + y(t) - e(t) \]

Prediction error formalism for model selection

Good model = small prediction error

Lots of existing (linear) theory - statistics
Lots of good (linear) software - matlab toolbox

Especially useful if we want the model for prediction

What about for control?
Prediction error methods PEM

A good model predicts the plant system output well

Need to test this outside the current application

Extrapolation and not just repetition

Changing experimental conditions

Input signal

Feedback control

Acid tests to determine two things

The best model fit to the data

The quality of the fit to the data
Prediction error - some math

model

\[ y_t = \hat{P}(z)u_t + \hat{H}(z)n_t \]

\[ \hat{y}_{t+1|t} = \hat{H}(z)^{-1}\hat{P}(z)u_t + [1 - \hat{H}(z)^{-1}]y_t \]

Associated predictor

Leads to the prediction error frequency-domain formula

Changing the input spectrum alters the prediction task

For control design want inputs similar to eventual controlled system

Circular problem

Sometimes prediction needs to be formulated without an input

Example coming up in combustion instability modeling

Different formulation

Similar ideas
Affecting PEM model fits

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} \left\{ L(z)[y_k - \hat{y}_k|k-1] \right\}^2 = \\
\frac{1}{2\pi} \int_{-\pi}^{\pi} \left\{ \left| P(e^{j\omega} - \hat{P}(e^{j\omega}, \theta) \right|^2 \Phi_u(\omega) + \left| H(e^{j\omega}) \right|^2 \right\} \frac{\left| L(e^{j\omega}) \right|^2}{\left| \hat{H}(e^{j\omega}, \theta) \right|^2} d\omega
\]

Big effects on model fit over frequency

Input spectrum
  Feedback controller
    Correlation between input and output
Data filter
Disturbance model
  Assumed known or estimated
Model structure
Closed-loop PEM formulae

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} \{ L(z)[y_k - \hat{y}_k|_{k-1}] \}^2 = \]

\[
\frac{1}{2\pi} \int_{-\pi}^{\pi} \left\{ \frac{|P(e^{j\omega}) - \hat{P}(e^{j\omega}, \theta)|^2}{|1 + P(e^{j\omega})C(e^{j\omega})|^2} |C(e^{j\omega})|^2 \Phi_r(\omega) + \frac{|1 + \hat{P}(e^{j\omega}, \theta)C(e^{j\omega})|^2}{|1 + P(e^{j\omega})C(e^{j\omega})|^2} |H(e^{j\omega})|^2 \right\} \frac{|L(e^{j\omega})|^2}{|\hat{H}(e^{j\omega}, \theta)|^2} d\omega
\]

Accounts for the correlation between input and output
More complicated than open-loop PEM formula
But still comprehensible
The controller is even more evident
Connects to robust control criteria
Closed-loop modeling and control

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} \left\{ L(z) [y_k - \hat{y}_{k|k-1}] \right\}^2 = \\
\frac{1}{2\pi} \int_{-\pi}^{\pi} \left\{ \frac{\left| P(e^{j\omega}) - \hat{P}(e^{j\omega}, \theta) \right|^2}{|1 + P(e^{j\omega})C(e^{j\omega})|^2} |C(e^{j\omega})|^2 \Phi_r(\omega) \right. \\
+ \left. \frac{\left| 1 + \hat{P}(e^{j\omega}, \theta)C(e^{j\omega}) \right|^2}{|1 + P(e^{j\omega})C(e^{j\omega})|^2} |H(e^{j\omega})|^2 \right\} \frac{|L(e^{j\omega})|^2}{|\hat{H}(e^{j\omega}, \theta)|^2} d\omega
\]

\[
\frac{|P(e^{j\omega}) - P(e^{j\omega})|}{P(e^{j\omega})} \times \frac{C(e^{j\omega})P(e^{j\omega})}{1 + C(e^{j\omega})P(e^{j\omega})} < 1
\]

The main issue is to understand which controller is \( C(z) \)

Current controller for identification, next controller for control

APC04 Vancouver
Acid tests of models

Falsificationism

Propose a new experiment to test the model

Corroboration or invalidation

Hypothesis testing approach

Model invalidation

Poor prediction

Strongly correlated residuals/errors

Systematic errors

Statistical tests

Karl Popper
Building models - some Philosophy

What can you do if the model fails?
Modify it to perform better
Deductive reasoning to include Physics
New model structure
Inductive reasoning fits the model to data

**Deduction:**
deriving conclusions from general or universal principles
Adjusting model structure to accommodate new experiments
Determining model structure from Physics

**Induction:**
deriving general conclusions from specific examples
“Let the data speak for themselves”
Fitting models and parameters to experimental data
Combustion instability modeling

Jet engines and gas turbines

Lean combustion yields economic and environmental benefits

Limited by appearance of limit cycling at low fuel-to-air ratios

Benefits are lost

Build a model for control of the combustion instability

Alternating deductive and inductive stages

Stressful experimental tests of models’ predictive powers
Experimental set-up

- Perforated Plate
- Side Wall
- Orifice Plate
- Air Flow
- Main Fuel
- Plenum
- Heat Release Rate Sensor
- Pressure Sensor
- Premixing Nozzle
- Combustor

Heated Air
Experimental data

Highly periodic
Not very informative for modeling

Harmonics at 210Hz, 420Hz and 630Hz

Non-harmonic component at 720Hz

Nonlinear phenomena
Combustion chamber acoustics meet heat release rate function
Improved fidelity with model development
Parsimonious model adjustments
Model development

**Peracchio & Proscia**
- First-order acoustics and model fit to data
  - Incapable of explaining multiple frequencies

**More complex Physics**
- Third-order acoustics and model fit to data
  - Corroboration simulation test passed
  - Invalidated at multiple operating points
    - 740Hz frequency changes with fuel-to-air ratio

**More simple Physics - another phenomenon included**
- Variable delay with fuel-to-air ratio and fit
  - Multiple-operating-point corroboration test passed
Message

Modeling involves a number of processes
   Deduction, induction, testing
     Much of this might be classified as “Prejudice”
     This embodies our understanding of the process
      I call this “Idiot testing” Does the model make sense?

Modeling for use in control design has a special set of prejudices
   Extraordinary simplicity
     Control systems operate over only a couple of decades of frequency
   Stability properties are important
     Unlike when modeling for prediction

We could really model well if we know what the final controller was