Lecture 5

Summary and Practical Aspects

The System Identification Procedure

1. Collect data (*experiment design*, $\mathcal{X}$). If possible choose the input signal such that the data become maximally informative. Reduce the influence of noise.

2. Choose the *model structure* ($\mathcal{M}$). Use priori knowledge and engineering intuition. Most important and most difficult step. (Do not estimate what you already know)

3. *Identification method* ($\mathcal{I}$). Determine the best model in the model structure (find optimal $\boldsymbol{\theta}$ using *e.g.*, the least squares method).

4. *Model validation.* Is the model good enough? Good is subjective, and depends on the purpose with the model.

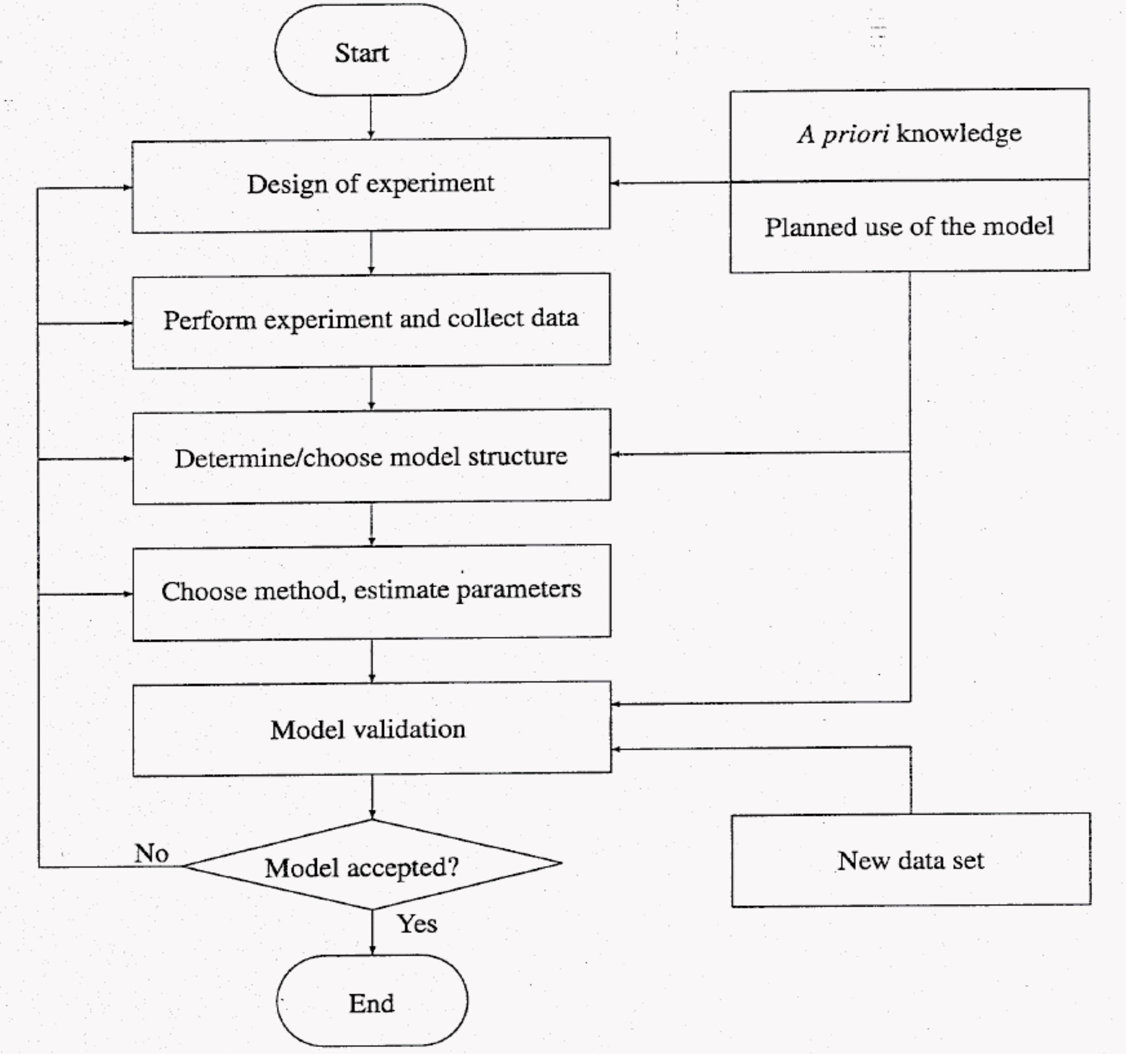


Figure 2: Schematic flowchart of system identification

Experiment Design

- Choice of input signal.
- Choice of sampling period.
- What signals to measure, and what type of sensors to use.
- How much data is needed.
- Experimental conditions.
  - Feedback in the data?
  - Test for linearity.
  - Test for time-invariance.

Choice of Input Signal

- Signal amplitude
  - Sufficiently small to ensure that we remain in the linear region of the system.
  - Sufficiently large to ensure that we have good excitation.
- Spectral range. The input should have most of its energy in the interesting frequency regions (depends on the application).
- *Persistently exciting* of a sufficient order! $\Rightarrow$ Required to assure consistency of parametric models.
- Physical limitations.

Choice of Sampling Period

Data: $\{u(kh), y(kh)\}_{k=1}^{N}$ where h is the sampling interval.

- Undersampling. Cannot pick up the essential dynamic $\Rightarrow$ Poor accuracy. Also, *aliasing*!
- Do not forget the anti-aliasing filter!
- Oversampling. Consecutive samples contain almost the same information $\Rightarrow$ Poor excitation $\Rightarrow$ Numerical and identifiability problems.

Determination of Model Structure

- Linear versus nonlinear, static versus dynamic, ...
- Algorithm complexity.
- Computational time and power.
- Depends on the application. Simple or more sophisticated model.

Static Models

Typical examples:

- Trends and non-zero means
- Cyclic components and harmonics

Model:

$$y(t) = \boldsymbol{\varphi}^T(t)\boldsymbol{\theta}$$

where $\boldsymbol{\varphi}(t)$ is deterministic (does not depend on old values of $y(t)$).

Ex: $y(t) = \boldsymbol{\varphi}^T(t)\boldsymbol{\theta}$, $\boldsymbol{\varphi}^T(t) = [1 \ t \ t^2]$.

Dynamic Models

General models:

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t)$$

$$y(t) = \boldsymbol{\varphi}^T(t)\boldsymbol{\theta} + v(t)$$

where $\boldsymbol{\varphi}(t)$ depends on old values of $y(t)$.

- Typical models: ARX, ARMAX, OE.
- The models have a certain dynamic range and are valid around a particular "working point".

In Practice

Data contain both static and dynamic components:

$$y(t) = y_s(t) + y_d(t)$$

where $y_s(t)$ is a static component and $y_d(t)$ is a dynamic component.

- Remove the static component before estimating the dynamic model. For example: detrend the data.
- Estimate the static component.
- Handling non-zero means.

Identification Methods

- Nonparametric Methods
  - Transient response, frequency response, spectral analysis.
  - Gives basic information about the system, and is useful for validation
- Parametric Methods: Static and dynamic cases
  - Least squares methods, instrumental variable methods, prediction error methods
  - Good accuracy. Easy to use for, *e.g.*, control
- On-line or off-line methods.

Least Squares Methods

System:

$$y(t) = \boldsymbol{\varphi}^T(t)\boldsymbol{\theta} + v(t), \quad t = 1, \ldots, N$$
$$\boldsymbol{Y} = \boldsymbol{\Phi}\boldsymbol{\theta} + \boldsymbol{v}$$

where $v(t)$ is a disturbance and $E\boldsymbol{v} = 0$, $E\boldsymbol{v}\boldsymbol{v}^T = \boldsymbol{R}$.

Estimate:

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{Y} = \Big[\sum_{t=1}^{N}\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^T(t)\Big]^{-1}\Big[\sum_{t=1}^{N}\boldsymbol{\varphi}(t)y(t)\Big]$$

Static case: Here $\varphi(t)$ is deterministic.

- Generally consistent estimates
- For *finite* value of N we have:

$$E\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_0$$

$$\operatorname{cov}\hat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{R}\boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}$$

- Can be extended to include the weighted least squares and the BLUE.

Dynamic case: $\varphi(t)$ depends on old values of $y(t)$.

- Consistent estimates if $v(t) = e(t)$ is white noise! $(Ee^2(t) = \lambda^2)$
- Asymptotically $(N \to \infty)$ it holds $(v(t) = e(t))$

$$\operatorname{cov}\hat{\boldsymbol{\theta}} = \lambda^2\left[E\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^T(t)\right]^{-1}$$

Instrumental variable methods

System:

$$y(t) = \boldsymbol{\varphi}^T(t)\boldsymbol{\theta} + v(t), \quad t = 1, \ldots, N$$

where $v(t)$ is a disturbance with $Ev(t) = 0$.

Estimate: Modify the least squares solution. We get:

$$\hat{\boldsymbol{\theta}} = \Big[\sum_{t=1}^{N} \boldsymbol{z}(t)\boldsymbol{\varphi}^T(t)\Big]^{-1}\Big[\sum_{t=1}^{N} \boldsymbol{z}(t)y(t)\Big]$$

where $\boldsymbol{z}(t)$ is the vector of instruments.

Results:

- Consistent estimate if:

$$
\begin{aligned}
&E\boldsymbol{z}(t)\boldsymbol{\varphi}^T(t) \quad \text{has full rank} \\
&E\boldsymbol{z}(t)v(t) = 0
\end{aligned}
$$

- The basic IV can be extended to include filtering and weighting.
- In general bad accuracy. Can be improved by, for instance, appropriate filtering.

Prediction error methods

Idea: Model the noise as well. General methodology applicable to a broad range of models.

The following choices have to be made:

- Choice of model structure. Ex: ARMAX, OE.
- Choice of predictor $\hat{y}(t|t-1, \boldsymbol{\theta})$.
- Choice of criterion function. Ex: $V(\boldsymbol{\theta}) = \frac{1}{N}\sum \varepsilon^2(t, \boldsymbol{\theta})$.

Estimate:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} V(\boldsymbol{\theta})$$

Results:

- In general we need to perform a numerical minimization.
- Consistent estimates (if the model covers the true system).
- In general statistically efficient estimates (Gaussian noise).
- Useful also for approximations.

On-line methods (Recursive Identification)

- In many cases an on-line estimate is required. Ex: adaptive signal processing, adaptive control.
- Tracking time-varying parameters.
- Fault detection.

Most off-line methods can be converted into on-line methods (exact or approximate).

Alternative Methods

- Subspace methods
- Nonlinear methods
- ...

Model Validation

A model is of no use unless it is validated!

- Check the residuals.
- Pole-zero cancellation.
- Cross validation.
- Parsimony principle.

Implementation Aspects

Most of the theory covered in the course is "implemented" in the *System Identification Toolbox*.

One can learn a lot by studying the toolbox. For example, type **help ident** or **iddemo**.

iddemo

1) The Graphical User Interface (ident): A guided Tour.
2) Build simple models from real laboratory process data.
3) Compare different identification methods.
4) Data and model objects in the Toolbox.
5) Dealing with multivariable systems.
6) Building structured and user-defined models.
7) Model structure determination case study.
8) How to deal with multiple experiments.
9) Spectrum estimation (Marple's test case).
10) Adaptive/Recursive algorithms.
11) Use of SIMULINK and continuous time models.
12) Case studies.

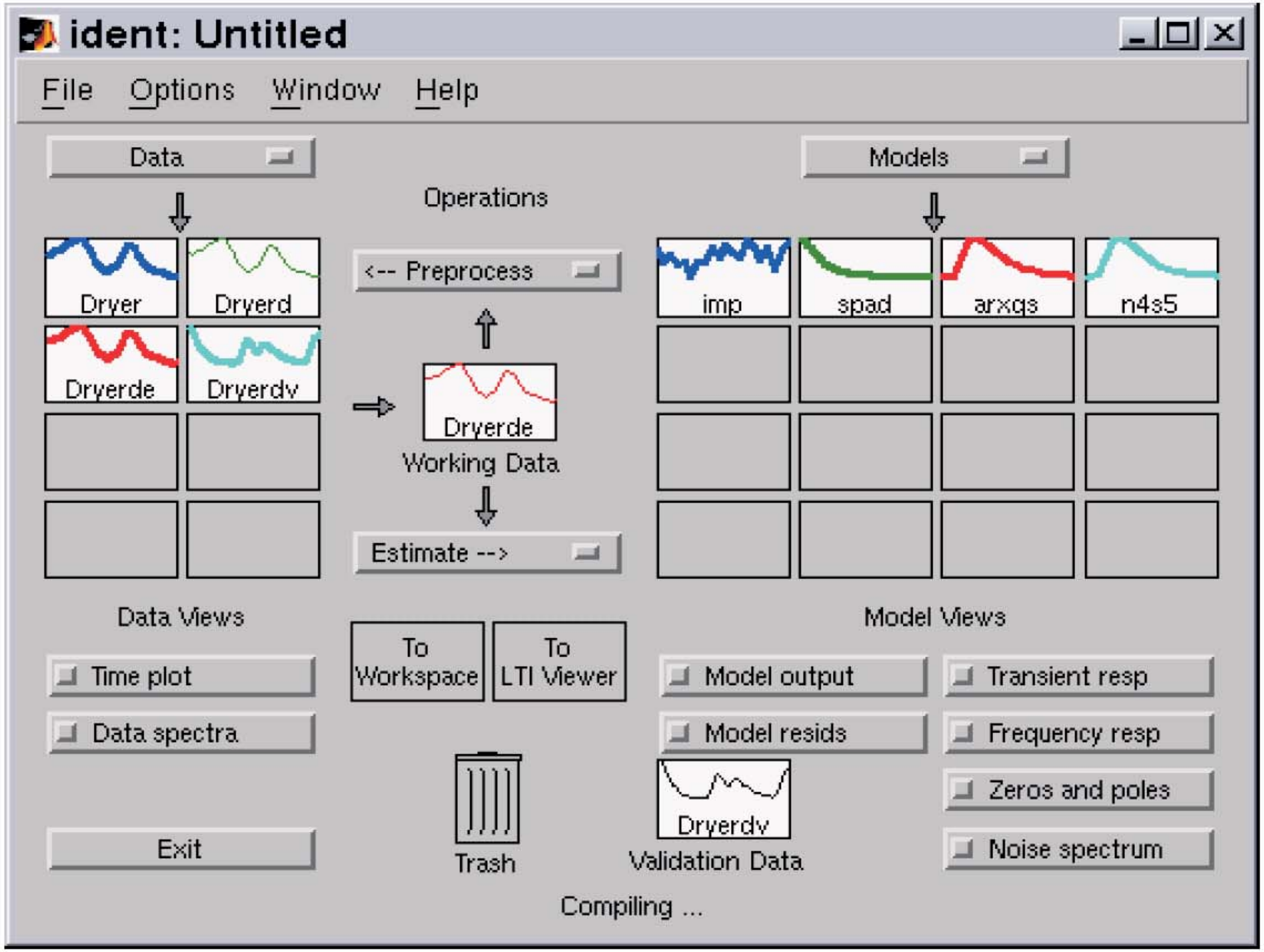


Figure 1: Graphical interface: ident

Case Study: Energizing a Transformer

In this case study we shall consider the current signal from the R-phase when a 400 kV three-phase transformer is energized. The measurements were performed by Sydkraft AB in Sweden.

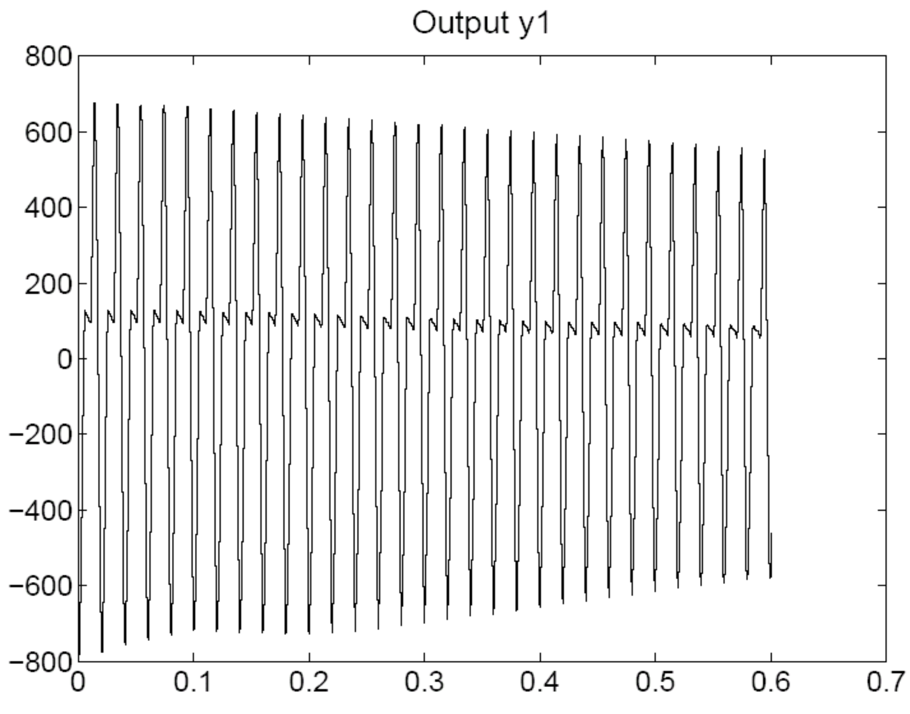
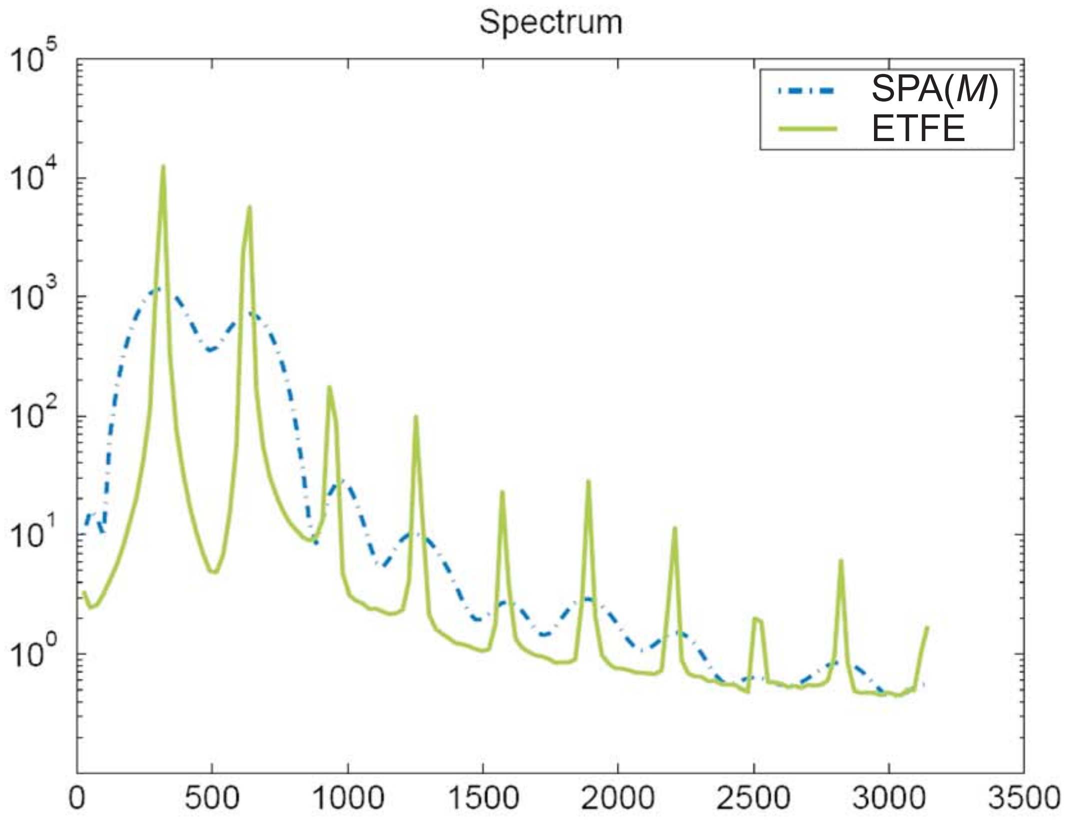


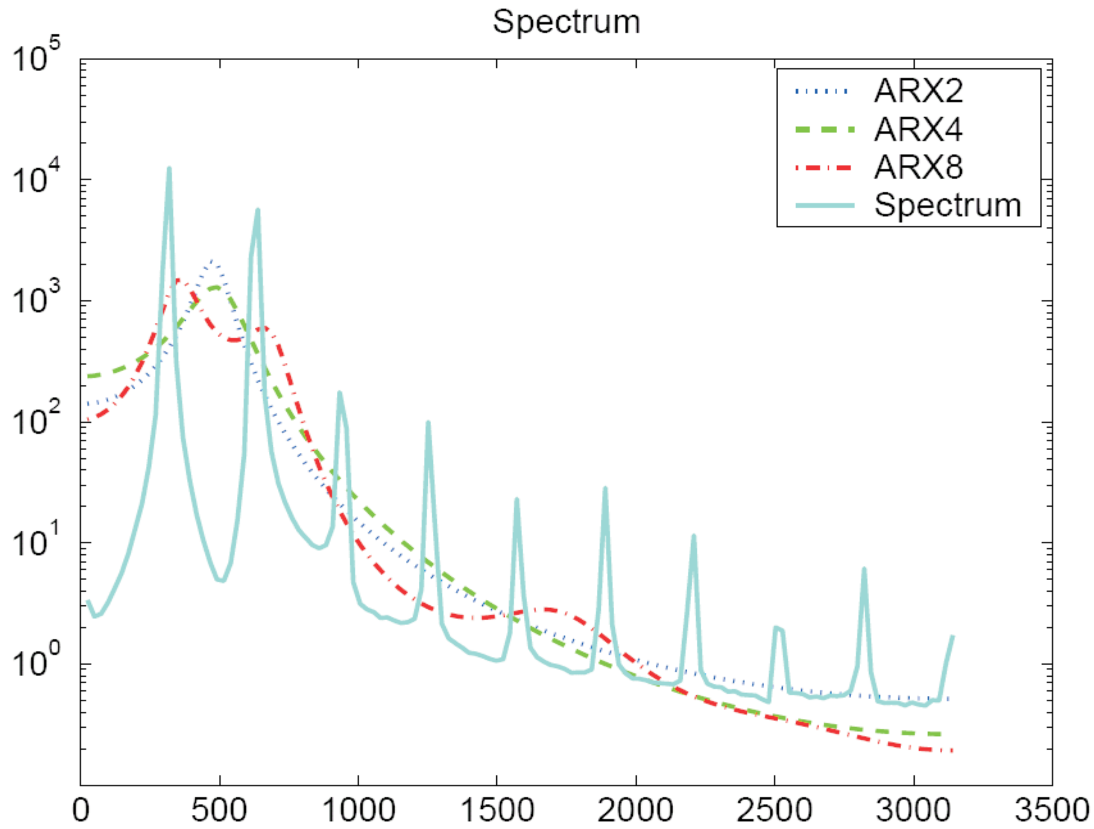
Figure 2: Data

Let us check the spectrum of the signal:



We see that a very large lag window will be required to see all the fine resonances of the signal. Standard spectral analysis does not work well.

Let us instead compute spectra by parametric AR-methods. Models of 2nd 4th and 8th order are considered:



We see that the parametric spectra are not capable of picking up the harmonics. The reason is that the AR-models attach too much attention to the higher frequencies, which are difficult to model.

Let us check all orders up to 30:

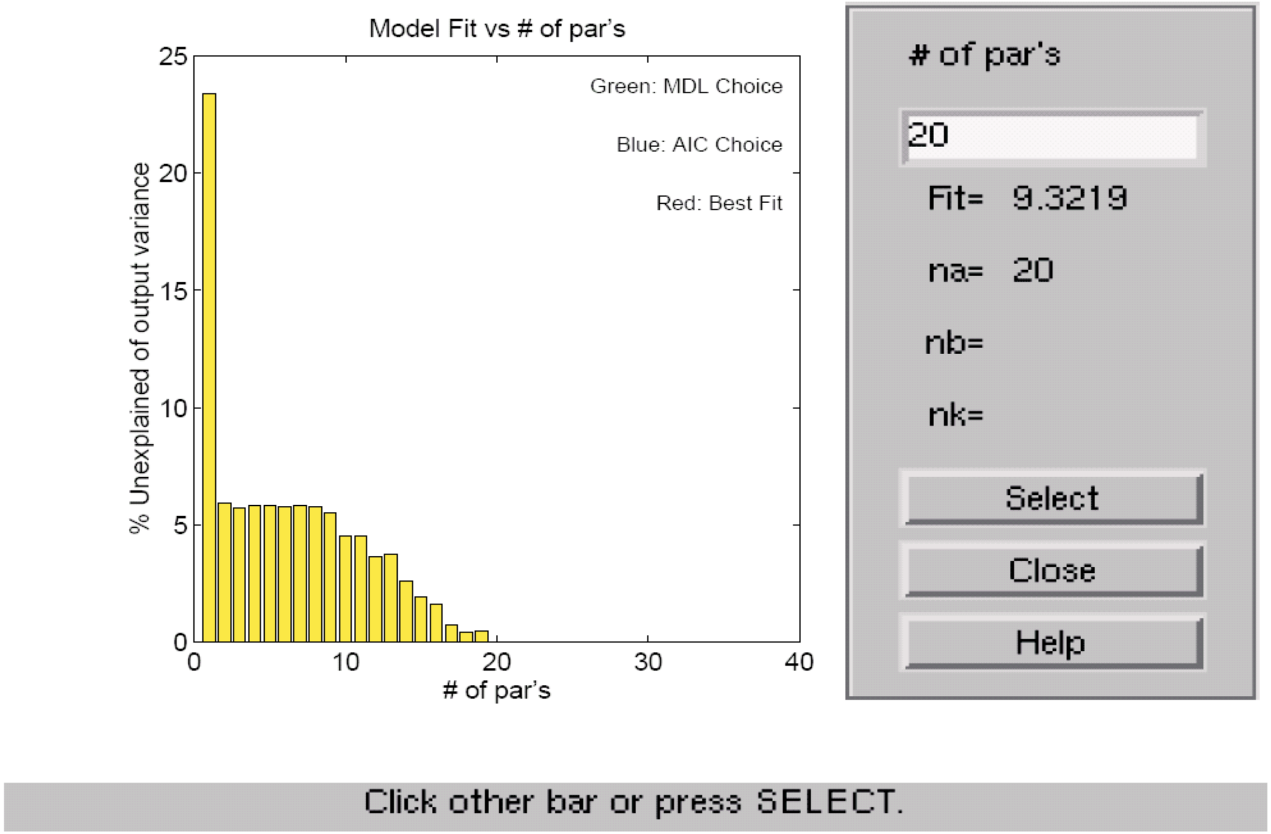
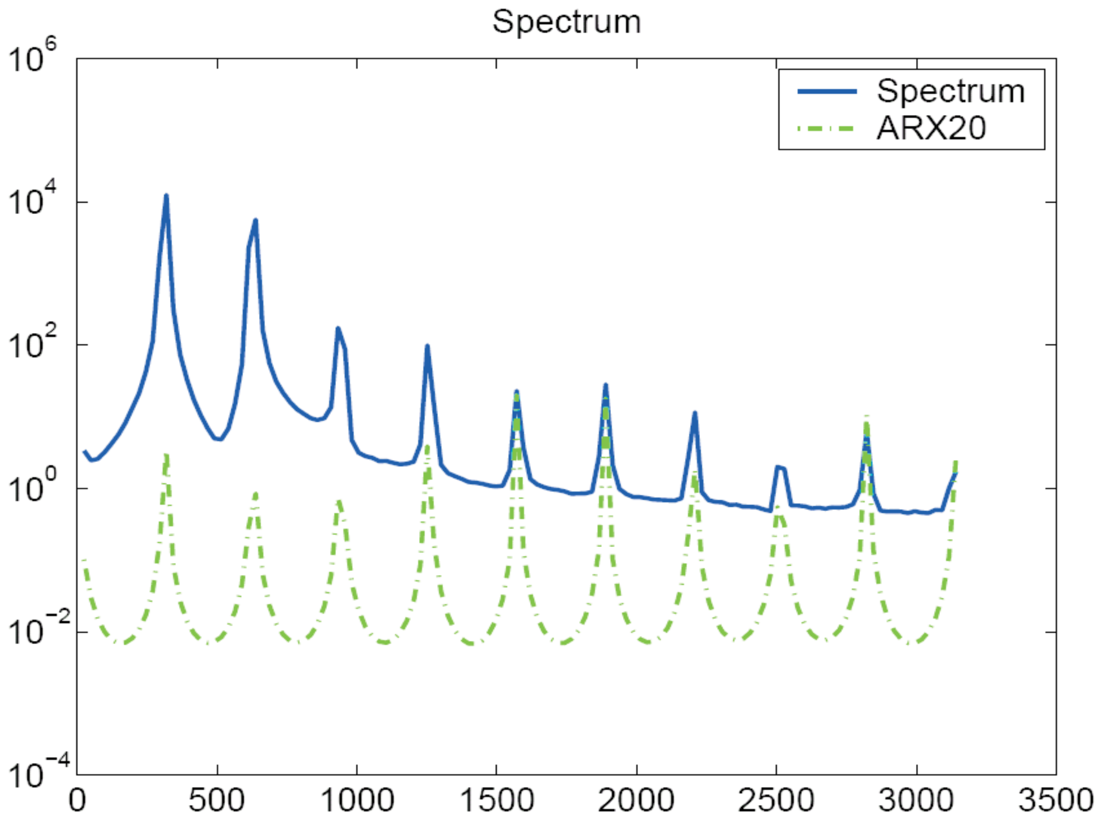


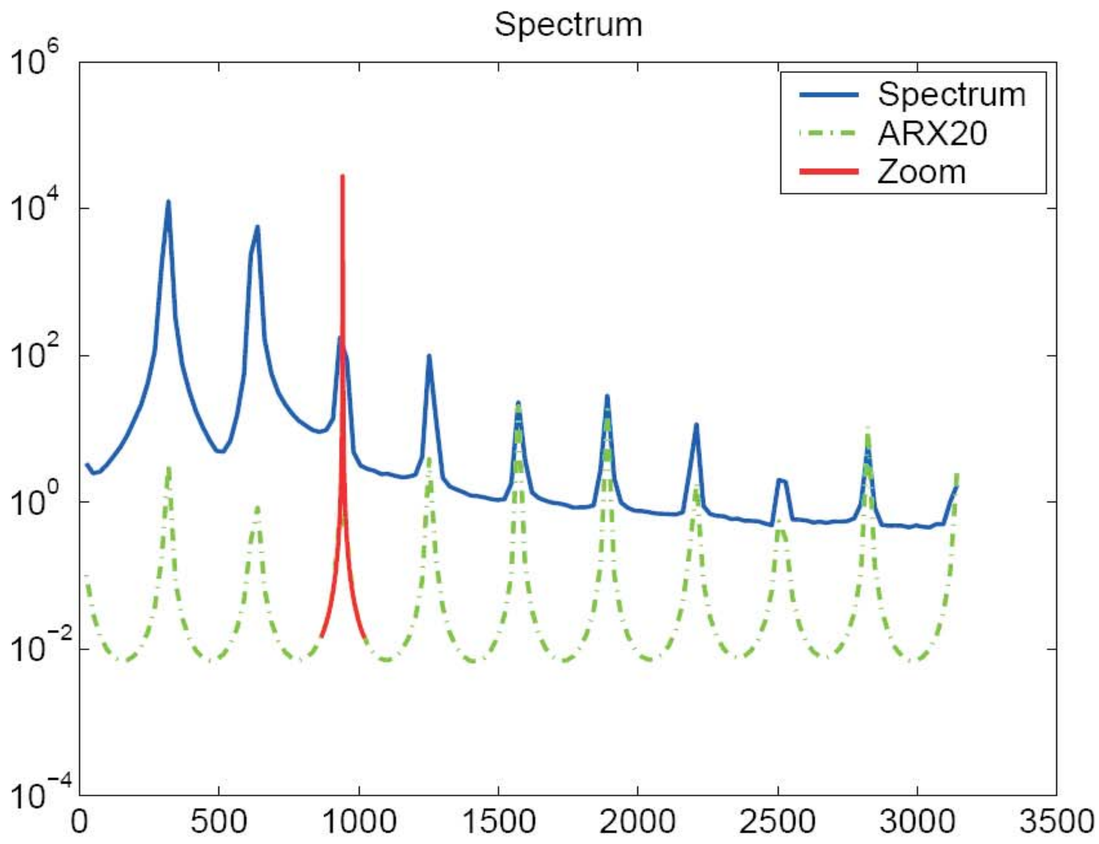
Figure 3: Model fit.

We see a dramatic drop for $n = 20$, so let's pick that order

ARX of order 20:

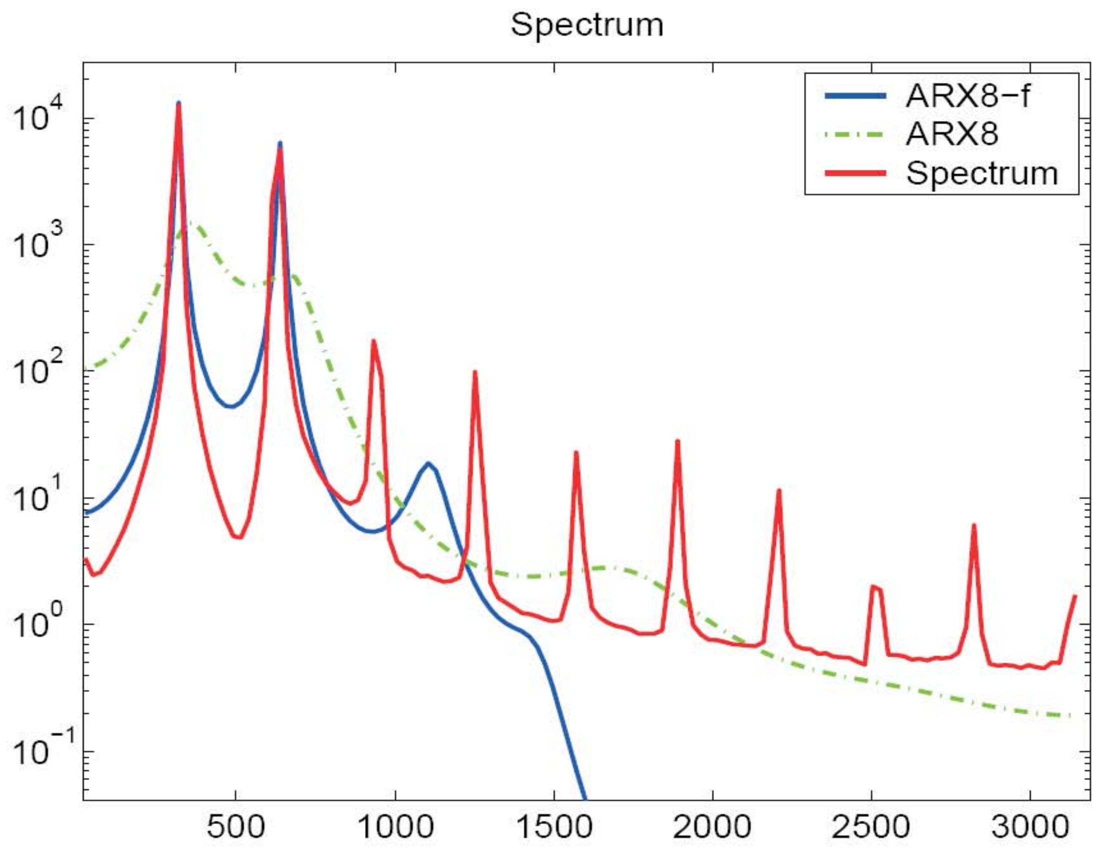


All the harmonics are now picked up, but why has the level dropped? The reason is that ARX20 contains very thin but high peaks. With the crude gitter of frequency points in the plot we simply don't see the true levels of the peaks. We can illustrate this as follows:

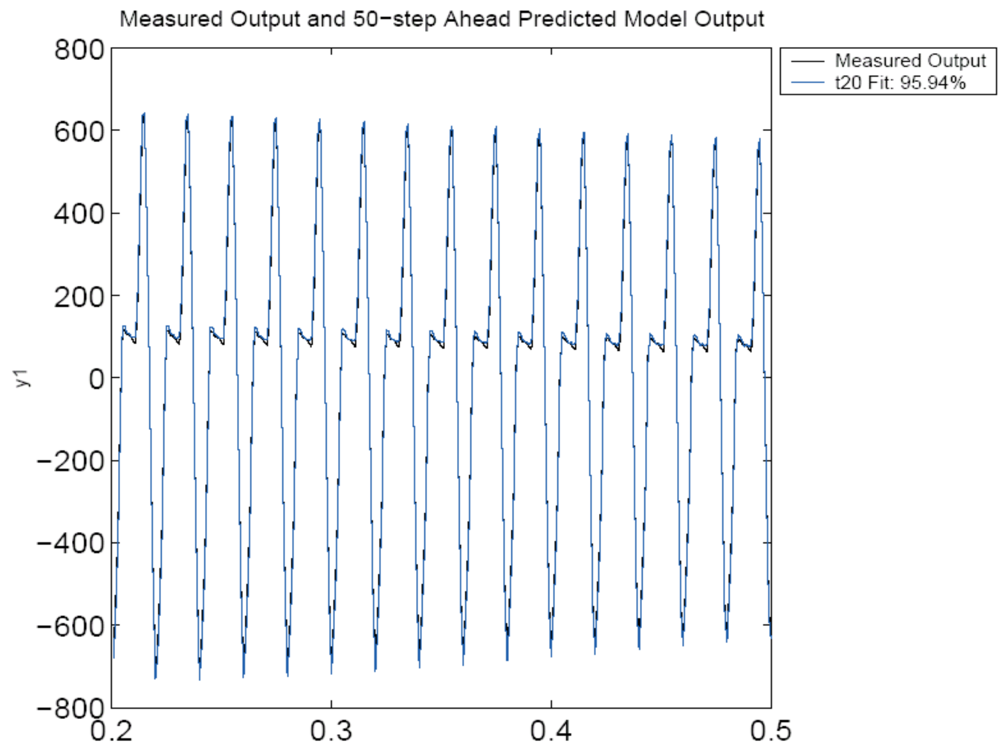


If we are primarily interested in the lower harmonics, and want to use lower order models we will have to apply prefiltering of the data. We select a fifth order Butterworth filter with cut-off frequency at 200 Hz. (This should cover the 50, 100 and 150 Hz modes).

Model the lower harmonics:



Conclusion:



For a complete model of the signal, ARX20 thus is the natural choice, both in terms of finding the harmonics and in prediction capabilities. For models in certain frequency ranges we can however do very well with lower order models, but we then have to prefilter the data accordingly.

Conclusions

- System identification is a powerful technique to model dynamic systems.
- Applications in virtually all disciplines of science.
- Implemented in *e.g.*, MATLAB.