Some aspects of industrial system identification

Alf J. Isaksson

ABB AB, Corporate Research, SE-721 78 Västerås, Sweden
(Alf.Isaksson@se.abb.com)

Linköping University, Department of Electrical Engineering, SE-581 83 Linköping, Sweden
(Alf.Isaksson@liu.se)

Abstract: The most important and time consuming part of an industrial application of control is the modelling. It may take 50 per cent or more of the entire project. Therefore a major challenge for a control systems supplier like ABB is to constantly try to decrease the engineering effort for modelling.

This paper discusses some different aspects of modelling and identification originating from application in many different industries such as pulp and paper, rolling mills, power plants and specialty chemicals.

Keywords: System identification, modeling, grey-box identification, data mining, process control.

1. INTRODUCTION

Basically all modern control is model-based. Regardless if it is tuning of PID controllers or application of more advanced controllers like model-predictive control (MPC), modelling is a key part of industrial application of control. Obtaining models for real industrial plants there are two main alternatives available:

- Deriving a model from first principles using laws of physics, chemistry, etc, followed by fitting some parameters to data – white-box modelling.
- Estimating an empirical model from experimental data -- black-box modelling

A third option is to combine these two techniques, i.e. starting with a physical model, but adding black-box elements to this model. This is often referred to as grey-box modelling or grey-box identification. At ABB all of these three modelling approaches are used, but it is not always entirely obvious which approach to apply in a particular project.

This paper will discuss some aspects a number of the use of system identification on industrial data. Application of system identification (in particular black-box identification) normally start by planning an identification experiment in order to collect sufficiently exciting data from the industrial plant to be modelled.

It is, however, not obvious that an experiment needs to be performed at all. In Peretzki et al. (2011) a method was presented how to find time intervals in normal operating data, stored in a history database, suitable for identification. The method is briefly described in Section 2 of this paper.

Yet another aspect is how to execute the state estimation and parameter optimization for the grey-box identification. Traditionally as described in the book by Bohlin (2006) the extended Kalman filter is used. Another alternative, however, is to use horizon estimation as presented in Isaksson et al. (2010), and also in Section 3.

When non-linear optimization is used to fit the parameters to experimental data it is still important to have good starting values for the parameters. Then it may be necessary to use insight into the problem to come up with a practical solution. One such method for estimating mechanical parameters in drive train systems was presented in Isaksson et al. (2003), and also in Section 4.

2. DATA MINING FOR SYSTEM IDENTIFICATION

Performing dedicated experiments for system identification is often a time-consuming task which may also interfere with the process operation. Nowadays, it is common to store measurement data from the plant operation (without compression) in a history database. Such data is a very useful source of information about the plant, and might contain suitable data to perform process identification. Due to the size of such databases, searching for data intervals suitable for identification is a challenging task. Preferably, this task should be supported by a data scanning algorithm that automatically searches and marks data intervals of interest.

We will here briefly describe one approach to perform such a data scanning. For more details the reader is referred to the paper by Peretzki et al. (2011). The method focuses on identification of such processes that are common in process industry, viz. low order stable plants without overshoot (i.e. with real poles) or integrating plants. In both cases process time delay (or dead time) is a very common feature. A linear black-box model structure which is rather insensitive to precise knowledge of the dead time is so-called Laguerre models, see Wahlberg (1991).

Normal operating data is in fact very rarely good enough for black-box identification. For data collected in closed-loop there are fundamental theoretical results described in, for
example Ljung (1999). From a practical point of view there are mainly two scenarios to hope for that may result in data informative enough for system identification, see Horch (2000):

- The process is operating in manual mode and the input signal $u_k$ is varied enough to be exciting the process.
- The controller is in automatic and there are enough changes in the setpoint $r_k$ to make identification possible.

As a consequence, the method is treating these two cases separately, and the method can be summarized in the following steps with gradually more and more computations involved:

1. Find intervals of data where the control is in manual and automatic operation, respectively. Treat each such interval separately.
2. Screen each interval for time instants where, for manual control, the control signal $u_k$ is changing (or $r_k$ in case of automatic). Mark this as a potential starting point of an interval with informative data.
3. Now search for time instants where the measured process output $y_k$ has a significant variation.
4. Filter the input $u_k$ through Laguerre filters up to order $n$ (use e.g. $n=10$). If the controlled loop is known to be integrating (e.g. level control) first integrate the input signal $u_k$.
5. Recursively form a regression matrix $R_k$ using all the Laguerre filter outputs, and check its condition number.
6. If the condition number of $R_k$ is low enough estimated the parameters $\theta$ of the Laguerre model, and check via a chi-squared test if they are significantly non-zero. If so mark this as a tentatively informative segment of data.
7. Repeat 3-6 for every new data point in the interval and as long as all are fulfilled the interval is still informative. When any of 3-6 fails mark as end point of interval.

The method was tested on data from 211 control loops provided by the specialty chemicals company Perstorp. The loops have considerably different dynamics but most of them can be modeled as a first order model and a delay, and it is not hard to tell beforehand which ones that will have an integrator. The delays can vary up to 10 min. The data is mainly from closed loop operation, but there is also open loop data. The database contained data from nearly 37 months of operation, sampled every 15s, in total almost 1.1G samples which requires 6.7GB of memory.

The proposed method was applied to these data, taking approximately 1.5h to process it all. In total, about 1.5% of data was found to be useful for process identification.

3. GREY-BOX IDENTIFICATION

What primarily triggers ABB’s interest in the area of grey-box identification is modelling for nonlinear model predictive control. While linear model predictive control has long been an established industrial area, nonlinear MPC has only more recently found industrial applications, see for example Franke and Vogelbacher (2006) and Pettersson et. al. (2006).

In general, a white-box model results in a set of non-linear Differential and Algebraic Equations (DAE)

$$\begin{align*}
0 &= f(x(t);u(t);u(t)) \\
\dot{x}(t) &= h(x(t);u(t))
\end{align*}$$

where $y(t)$ denotes the measured process variables, and $u(t)$ the manipulated variable, i.e. the output of the MPC. Finally the internal variable $x(t)$ is what is usually referred to as the state of the system. A black-box model on the other hand, is typically linear, and most often also discrete in time

$$\begin{align*}
x_{k+1} &= Ax_k + Bu_k \\
y_k &= Cx_k
\end{align*}$$

Here the integer $k$ denotes the $k$:th time index for which using the data intervals selected by the new method were found very similar to those obtained during the identification experiments. Figures 1 and 2 present two examples of intervals found in open/closed loop operation. They also illustrate the quantities computed as part of the algorithm.
For the state estimation the optimization target is to obtain the best estimate of the internal variable \( x \) using knowledge of \( y \) and \( u \), to be used as starting point for the forward optimization. This can be done using a Kalman filter (for an old classic see Anderson and Moore (1979)) -- or if the model is nonlinear an extended Kalman filter -- where a stochastic modeling of process and measurement noises is applied. A Kalman filter is a recursive method, meaning that it takes only the most recent values of \( y_k \) and \( u_k \) to update the previous estimate \( \hat{x}_{k-1} \) to obtain the new one \( \hat{x}_k \). Hence it does not actually solve an optimization problem on-line. Kalman filtering is done in a statistical framework by adding process noise and measurement noise to the discrete-time state space system given in the previous sub-section:

\[
\begin{align*}
    x_{k+1} & = Ax_k + Bu_k + w_k \\
    y_k & = Cx_k + v_k
\end{align*}
\]

where \( w_k \) and \( v_k \) are white Gaussian noises with covariance matrices \( Q \) and \( R \) respectively. Since we want to use certain quantities in the calculation later, the complete set of Kalman filter equations is given below:

\[
\begin{align*}
    S_k &= CP_kC^T + R \\
    K_k &= P_{k|k-1}C^T S_k^{-1} \\
    \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k (y_k - C\hat{x}_{k|k-1}) \\
    P_{k|k} &= (I - K_k C)P_{k|k-1} \\
    \tilde{x}_{k+1|k} &= A\tilde{x}_{k|k} + B u_k \\
    P_{k+1|k} &= AP_{k|k}A^T + Q
\end{align*}
\] (1)

With access to more computational power, a much newer and increasingly popular approach is to use so-called moving horizon estimation (MHE) instead, see Rao et. al. (2001). Then the process and measurement noise introduced are used as slack variables in the optimization. If the model is nonlinear these slack variables are usually introduced in a discretized version of the model:

\[
\begin{align*}
    x_{k+1} &= g(x_k, u_k) + w_k \\
    y_k &= h(x_k, u_k) + v_k
\end{align*}
\]

Moving horizon estimation then corresponds to minimizing

\[
\begin{align*}
    \min \, V &= (x_{k-M+1} - \hat{x}_{k-M+1})^T P^{-1} (x_{k-M+1} - \hat{x}_{k-M+1}) \\
    &+ \sum_{n=k-M+1}^{k} w_n^T Q^{-1} w_n + v_n^T R^{-1} v_n
\end{align*}
\] (2)

with respect to all states \( x_n \) within the horizon and possibly subject to constraints as, for example, \( x_{\text{min}} < x_n < x_{\text{max}} \). Here \( P, Q \) and \( R \) are weight matrices used for tuning of the estimator, which have a similar interpretation and importance as the estimate and noise covariance matrices in Kalman filtering. As indicated in its name, the optimization for moving horizon estimation is typically done over a horizon \([t-(M-1)T_s; t]\) of data, where \( t \) is the current measurement time. Since this time interval is in the past, we assume access to historic values of the applied manipulated variables \( u_k \) and the measured process variables \( y_k \). The first penalty term in the criterion is called the arrival cost. It is to create a link from one optimization window to the next, where \( \hat{x}_{k-M+1} \) denotes the estimate for this particular time instant from the optimization run at the previous cycle. Figure 3 tries to illustrate the MHE optimization which is a weighted sum of the vertical bars in the upper and lower plots.

Notice that in the optimization problems described above, the model (nonlinear or linear) should be considered as an equality constraint. We will not go into how these optimization problems are solved. Let us just point out that depending on the objective function and constraints (most importantly the model) different type of optimization problems result, leading to different type of optimization solvers needed. For example, a quadratic objective together with linear constraints corresponds to a quadratic programming (QP) problem, whereas a nonlinear objective or nonlinear model yields a nonlinear programming (NLP) problem which of course is much more difficult to solve. The latter case, when combined with the forward optimization, is usually referred to as a nonlinear MPC problem (NMPC).

### 3.1 Parameter estimation

Even if physical modelling is used there are often parameters whose numerical values are unknown, and therefore need to be estimated from data. Furthermore it is typically not easy to know at what level of detail to perform the physical modelling. A systematic procedure to gradually build up nonlinear models using collected process data is usually referred to as grey-box identification. One such procedure is presented in the book by Bohlin (2006), with which also follows a Matlab based software called MoCaVa. The software was also presented in Bohlin and Isaksson (2003), and contains modules for:

- Preparation of data
- Simulation -- To solve the (nonlinear) differential equations for fixed parameters and known inputs
- Calibration -- to start with a root model and extend model gradually
- Validation -- to test the model for a specific purpose

The part of the procedure of particular interest in this paper is the Calibration which includes the steps:

- Start with the root model
- For each extension make a fit to data
- Compute the statistical risk of rejecting the previous model

The goal of the calibration can be formulated as: "Find the simplest model that cannot be falsified using the available data". For the parameter fit, MoCaVa uses a maximum-likelihood criterion based on the extended Kalman filter

\[
\min_{\theta} V_{MLE} = \sum_{k=1}^{N} ((y_k - \hat{y}_k)^T S_k^{-1} (y_k - \hat{y}_k) + \log \det(S_k)) \tag{3}
\]

where \( S_k \) is the same as given by (1). Then for the computation of the statistical risk the fact that \( V_{MLE} \) is (asymptotically) chi-squared distributed is utilized.

Now instead, having an MPC implementation based on moving horizon estimation, a valid question is whether one cannot use the same optimization code for the parameter estimation? Because the identification is made on a collected set of input and output data, an on-line identification would rather correspond to Horizon Estimation, since no moving window will be applied.

The answer is that an ML criterion may be formed using horizon estimation, but in addition to the criterion (2) it will contain a bias correction term which (somewhat surprisingly) is exactly the same one as in (3), i.e the ML-criterion becomes

\[
\min_{s_k, \theta} V_{ML} = \sum_{k=1}^{N} v_k^T R^{-1} v_k + w_k^T Q^{-1} w_k + \log \det(S_k) \tag{4}
\]

See Isaksson et al. (2010) for a detailed derivation of this expression.

A common approach is to not include any process noise \( w_k \) when performing the parameter estimation (see e.g. Zavala (2008)), and only use the first term in (4) with fixed or no weighting matrix. This is, in the system identification literature (e.g. Ljung (1999)), known as Output Error (OE) and will also lead to unbiased estimates. Notice, however, that if we want to include the measurement covariance \( R \) as a free estimation parameter, then the bias correction term is indeed necessary, otherwise there is nothing preventing the covariance estimate from tending to infinity.

### 3.2 Monte-Carlo Simulations

In this section we will illustrate the theoretical result of the previous section by simulating a simple first order (linear) process for many different noise realizations. The chosen example is given by

\[
x_{k+1} = ax_k + bu_k + w_k \\
y_k = x_k + v_k
\]

where both \( w_k \) and \( v_k \) are zero mean Gaussian noises with unit variance.

The particular example \( a = 0.7 \) and \( b = 0.3 \) was simulated for 100 different noise realizations (one such simulation is shown in Figure 2) and the optimal parameter estimates where found using three different criteria:

**HE** Horizon estimation, i.e.

\[
\min_{x_k, a, b} V_{HE} = \sum_{k=1}^{N} w_k^2 + v_k^2 \\
x_{k+1} = ax_k + bu_k + w_k \\
y_k = x_k + v_k
\]

**ML** Horizon estimation with the bias correction:

\[
\min_{s_k, a, b} V_{ML} = \sum_{k=1}^{N} v_k^2 + w_k^2 + \log(s_k) \\
x_{k+1} = ax_k + bu_k + w_k \\
y_k = x_k + v_k \\
p_{k+1} = a^2 p_k^2 / (p_k + 1) + 1 \\
s_k = p_k + 1
\]

**OE** Output error, i.e. estimating the parameters from a pure simulation model

\[
\min_{a, b} V_{OE} = \sum_{k=1}^{N} (\hat{y}_k - y_k)^2 \\
\hat{y}_{k+1} = a\hat{x}_k + bu_k
\]

The parameter estimates are shown in Figure 4, where each cross corresponds to one estimate for one realization. From the figure it is clear that pure Horizon Estimation leads to significant bias in the parameter estimates, while both Maximum-Likelihood and Output Error are unbiased. However, as expected, ML clearly has much lower covariance of the parameter estimates.

![Figure 4](image-url). Monte-Carlo simulation comparing three different identification methods for first-order system.

### 3.3 Estimation of non-linear models

While the derivation and simulations above are for linear systems, the end goal is of course to use this new optimization approach for identification of non-linear models. For the case without process noise the maximum-likelihood has a very straight-forward solution, since \( S_k = R \), by solving the optimization problem (e.g. using IPOPT):
The complete non-linear case with process noise is, however, still an open research problem which we are now studying as part of a large European research project called MODRIO.

4. IDENTIFICATION OF DRIVE TRAIN SYSTEMS

Previous section discussed the concept of model calibration for non-linear systems. To be able to initialize such a calibration, initial values of the unknown parameters are needed. Very rarely is a zero parameter value a good starting point in a non-linear model. This section discusses this by an example for identification of mechanical parameters in drive systems. Many industrial drives consist of motor, flexible shafts, gear-boxes and load, which form a multi-mass system. Backlash is introduced due to existence of elements like gear-boxes and flexible couplings. In this case, the system model becomes non-linear.

We will here assume that for many applications a two-mass model will be sufficient. The following linear model of an uncertain two-mass system without backlash was presented in (Nordin, 2000):

\[
J_m \dot{\omega}_m = -c_m \omega_m - T_e + T_m - T_0 \\
J_l \dot{\theta}_l = -c_l \omega_l + T_e
\]

with transmitted shaft torque

\[
T_e = k_s (\omega_m - \omega_l)
\]

where the system input is \( T_m \), the motor torque, and the measured output is \( \omega_m \), the motor speed. The other two signals in the model are \( \theta_l \) and \( \omega_l \), the angle difference between motor and load, and \( \omega_m \), the load speed.

The parameters to estimate are:

- \( T_0 \): the idle torque, Nm
- \( J_m \): the motor moment of inertia, kgm\(^2\)
- \( c_m \): the viscous motor friction, Nm/(rad/s)
- \( J_l \): the load moment of inertia, kgm\(^2\)
- \( c_l \): the viscous load friction, Nm/(rad/s)
- \( k_s \): the shaft elasticity, Nm/rad
- \( c_s \): the shaft damping, Nm/(rad/s)

The suggested approach is to fit parameters directly in the model (5)-(8) by solving the system of differential equations and forming a least-squares criterion by comparing the computed motor speed to the measured one.

There are, however, at least two things that complicate matters. Firstly, we need to somehow obtain initial estimates of the seven parameters listed above. Secondly, when there is significant backlash in the system we need to introduce a model for this as well and estimate a gap parameter.

In an attempt to circumvent these two complications an approach based on two dedicated dynamic experiments was tested:

1) Perform a tailored experiment that guarantees that no gap openings are encountered. Then use black-box identification of a general third order transfer function to find initial mechanical parameters.
2) Fine tune the mechanical parameters on the same data using a physical model.
3) Perform another experiment that with certainty contains gap openings. Use this data set to estimate the gap size, while keeping the other model parameters fixed.

These steps are described in some more detail in the subsequent sections.

4.1 Black-box identification

The uncompensated transfer function of the open loop, \( P_{um}(s) \), from \( T_m \) to \( \omega_m \) is given by:

\[
P_{um}(s) = \frac{J_m s^2 + (c_m + c_l) s + k_s}{d(s)}
\]

with

\[
d(s) = J_m J_l s^3 + (J_l (c_m + c_l) + J_m (c_l + c_i)) s^2 + ((J_l + J_m) k_s + c_m c_l + c_l c_i) s + (c_m + c_l) k_s
\]

The static gain

\[
\frac{1}{c_m + c_l}
\]

is very difficult to estimate from dynamic experiments, since it is so large that for short data records the system acts almost like an integrating system.

Neglecting the inertia dampings gives a model with pure integration of the form

\[
G(s) = \frac{b_2 s^2 + b_1 s + b_0}{s(s^2 + a_2 s + a_1)}
\]

where

\[
b_2 = \frac{1}{J_m}
\]

\[
b_1 = \frac{c_i}{J_m J_l}
\]

\[
b_0 = \frac{k_s}{J_m J_l}
\]

\[
a_2 = \frac{c_l (J_m + J_l)}{J_m J_l}
\]
The idea is now to estimate a black-box model and solve for the \( a \) and \( b \) parameters. Notice that there are now five equations but only four unknown parameters. Several alternatives are therefore possible (including solving the overdetermined system of equations in a least-squares sense). We have tried them all and in the end settled for using equations (9), (11), (12) and (13). Hence the equation (10) is excluded in the solution.

It now remains to find initial estimates for the inertia dampings \( c_1 \) and \( c_m \), since they were excluded in the black-box identification. Notice also that since mean values are typically subtracted from the signals before black-box identification no estimate of the idle torque \( T_0 \) was obtained either.

To solve these two remaining problems we suggest introduction of yet another dedicated experiment to identify the static gain. This is described in detail in Isaksson et al. (2003).

### 4.2 Grey-box identification

Once initial estimates of the mechanical parameters have been obtained using the procedures described in Sections 4 and 5, fine tuning is made by fitting the mechanical parameters directly, using a least-squares criterion

\[
V = \sum (\omega_m(k) - \hat{\omega}_m(k, \theta))^2
\]

where \( \omega_m(k) \) denotes the (discrete-time) measurement of motor speed and \( \hat{\omega}_m(k, \theta) \) denotes the motor speed simulated through the model (5)-(8) at the same time instants, using the current mechanical parameter values here denoted by the vector \( \theta \).

In a pre-study the general purpose Matlab based program MoCaVa -- (Bohlin and Isaksson, 2003) -- was used to verify the feasibility of the approach. For the final tool a modified version of the Matlab toolbox Diffpar -- (Edsberg and Wedin, 1995), Edsberg and Wikström, 1995 -- has instead been used to develop a graphical user interface dedicated to the drive train problem.

### 4.3 Real data example

Real data has been collected on a test rig at ABB Process Automation, as well is from both rolling mills and paper mills. We will here show the results from one set of data from the test rig, consisting of one AC motor on the drive side, and a DC motor as load. The system should be possible to model very well using a two-mass model. From data sheets the motor inertia is given as \( J_m = 0.40 \), the load inertia \( J_l = 5.6 \) and shaft stiffness \( k_s = 3300 \) (with negligible shaft inertia). The test rig also has a rubber belt coupling, which has a similar effect as a gear play.

First the steady-state experiment is shown in Figure 5, where in the lower plot the automatically found steady-state intervals are marked. Obviously there is very little damping (i.e. the line has almost zero slope), while there is significant static friction resulting in a non-zero idle torque. These facts are also reflected in the obtained parameter estimates \( c = 0.00184 \) and \( T_0 = 15.14 \).

![Fig. 5. Steady state identification for test rig data.](image)

In Figures 6 and 7 the data and model fit for the backlash free experiment are shown. Figure 6 shows the fit after initial black-box identification and Figure 7 the fine tuned fit after linear grey-box estimation.

![Figure 6. Data and model fit of black-box identification for test rig data (data solid, simulation dashed).](image)

The estimation results for the mechanical parameters are summarized in Table 1. Notice that even though from visual inspection the fit looks very good already after the initial black-box estimation, the fine tuning in fact reduces the loss function by almost 25 %.

According to the “true” values given above both the load inertia and shaft stiffness estimates are quite acceptable, whereas the motor inertia is estimated about 50 % too large. In this case, however, it is probably fair to claim that the estimated parameters better represent the real system than the ones obtained from data sheet.
5. CONCLUSIONS

The paper has discussed some important aspects for industrial application of system identification. In Section 2 it was concluded that it may be possible to find informative data intervals in historic data, thus avoiding to perform an identification experiment. This has multiple advantages since an experiment may be time-consuming and sometimes leads to out of spec product while the experiment is on-going. Furthermore there is always the risk of tripping the process during the experiment.

For less informative data, an appealing alternative may be to estimate parameters in a physical model, since often fewer parameters need to be fitted if the process is higher order. It is then important, however, to not free up all possible parameters at once but use a calibration procedure based on statistical significance; so-called grey-box identification. To do this maximum-likelihood estimation is the natural solution. In Section 3 it was shown that Horizon Estimation can be used to form the maximum-likelihood criterion but only if a bias compensation term is added.

Finally in Section 4 it was shown via an example that tailored solutions are sometimes necessary to find the initial parameters for the grey-box identification.

ACKNOWLEDGEMENTS

The author gratefully acknowledges financial support from the Swedish Foundation for Strategic Research (SSF) -- as part of the Process Industry Centre Linköping (PIC-LI), and the Swedish Agency for Innovation Systems (VINNOVA) through the project MODRIO. The author would also like to thank a number of former and current students and colleagues, who also appear as co-authors in the cited papers.

REFERENCES