(article starts on next page)

NONLINEAR INSTRUMENT VARIABLE METHODS
BASED ON LOCAL LINEAR MODELS

Astrid Lundgren and Jonas Sjöberg

Department of Machine and Vehicle Systems,
Chalmers University of Technology, 412 96 Göteborg, Sweden,
Email: astrid.lundgren@me.chalmers.se,
jonas.sjoberg@me.chalmers.se

Abstract: The aim of the present study is to derive nonlinear instrument variable methods by using local linear models. Two algorithms to estimate consistent local ARX-models of the system order are presented. A local ARX-model with a regressor of higher order than the system is simulated to estimate an approximately noise-free data set. In the first algorithm this approximately noise-free data is used as estimation data to a local ARX-model of the system order. The second algorithm uses the simulated data as instrument in a local instrument variable method. The algorithms are demonstrated on both simulated and laboratory data. *Copyright ©2004 IFAC*

Keywords: nonlinear, system identification, IV-methods, local linear models

1. INTRODUCTION

In linear system identification, the advantage of the ARX-model structure is that the prediction can be expressed as linear regression, ie,

$$\hat{y}(t|\theta) = \varphi^T(t)\theta \qquad (1)$$

where

$$\theta = [a_1 \ldots a_{n_a} \; b_0 \ldots b_{n_b}]^T \qquad (2)$$

is the parameter vector, and the regressor

$$\varphi(t) = [-y(t-1) \cdots - y(t-n_a)$$
$$u(t-n_k) \ldots u(t-n_k-n_b+1)]^T \qquad (3)$$

is made of n_a lagged old outputs and n_b lagged old inputs with delay n_k samples. If the parameter estimate is defined as the minimum of the sum of squared errors, the parameter estimate can be expressed as a closed expression

$$\hat{\theta}_N = \left[\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\varphi^T(t) \right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t)y(t) \qquad (4)$$

where N is the size of the estimation data set. This means that the estimate can be calculated very rapidly

and there are no problems with local minima. The drawback with the ARX model is that there is no independently parameterized noise model, which implies that the estimate of the plant model will normally be biased depending on the color of the noise. This means that the plant estimate is not consistent. In transfer functions, the ARX model is described by

$$y(t) = \frac{B(q)}{A(q)}u(t) + \frac{1}{A(q)}e(t) \qquad (5)$$

where $B(q)$ and $A(q)$ are polynomials in the shift operator $qu(t) = u(t+1)$ parameterized by θ, (2). It is the common denominator $A(q)$ in (5) which causes the inconsistency.

There are two common remedies to the inconsistency problem in linear identification. The first, and probably the most common, is to introduce an independent noise model, for example the output error (OE) model and the Box-Jenkin's (BJ) model, or a partly independent noise model as in the ARMAX model. For these models, the parameter estimate has to be calculated with an iterative search for the minimum of a criterion.

This is more time consuming and there is also a risk to be stack in a local minimum.

The second remedy of the inconsistency is to use an instrument-variable (IV) method to estimate the plant parameters. The main idea of IV methods is to choose instruments that are uncorrelated with the noise and that replace the measured output in the regressor. For a background on linear identification see eg (Ljung, 1999; Söderström and Stoica, 1989).

In nonlinear system identification, it is possible to formulate nonlinear counterparts of the common linear model structures, see eg, (Ljung, 1999; Sjöberg *et al.*, 1995). The most commonly used model structure is the nonlinear ARX model (NARX) which consists of a nonlinear mapping of the regressor (1), ie, prediction is described by

$$\hat{y}(t|\theta) = g(\theta, \varphi(t)) \qquad (6)$$

where $g(\cdot, \cdot)$ is the nonlinear mapping. However the NARX has similar consistency problems as its linear counterpart and it is possible to define NOE, NBJ, and NARMAX models to counteract the problem, see, eg, (Ljung, 1999; Sjöberg *et al.*, 1995; Nørgaard *et al.*, 2000; Chen and Billings, 1992). In principle, these model structures solve the inconsistency problem, but there are some drawbacks. First, stability of the prediction model has to be monitored during the iterative search for the parameter estimate. Secondly, a simultaneous search for nonlinear plant and noise models increases search space so that many more model structures have to be investigated. So far, at least to the authors' knowledge, there are no IV identification methods available for nonlinear systems. The reason for this is that IV methods rely on model structures described as linear regression.

This contribution suggests two IV-related estimation methods for nonlinear system identification. Both methods are based on local linear models where a subset of the data is used in the linear regression step (1). This makes it possible to combine ideas from linear IV methods with nonlinear system identification methods based on local models. So far these local models have only been used to define NARX models. See, eg, (Atkeson *et al.*, 1997; Stenman, 1999; Bontempi *et al.*, 1999).

The paper is organized as follows: the first give a background on linear methods to make consistent estimates and the theory of local linear models. The second section discusses how to combine theory from linear IV-methods with local linear models. In the third part the parameter estimation algorithms are demonstrated on simulated and laboratory data.

2. BACKGROUND

This section gives a brief background of the relevant linear identification theory and the local linear technique used.

2.1 Inconsistency of linear regression models

Assume that the estimation data has been generated by

$$y(t) = \varphi^T(t)\theta_0 + e(t) \qquad (7)$$

where θ_0 is the true parameter vector and $e(t)$ is a disturbance signal. The parameter estimate, (4), then becomes

$$\hat{\theta}_N = \left[\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\varphi^T(t) \right]^{-1} \times$$
$$\frac{1}{N} \left(\sum_{t=1}^{N} \varphi(t)\varphi^T(t)\theta_0 + \sum_{t=1}^{N} \varphi(t)e(t) \right) \quad (8)$$

which limits θ_0 only if

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \varphi(t)e(t) = 0 \qquad (9)$$

This holds only if the data is described by (7), or equivalently by (5), ie, if the disturbance is described by the plant denominator, $1/A(q)$. If this assumptions does not hold, the estimate (8) gives a biased estimate of the plant $B(q)/A(q)$. How this is handled with IV method is described next.

2.1.1. Linear Instrumental-Variable-methods
The IV-methods are extensions of the linear regression and avoid the problem of inconsistent estimate by introducing an instrument variable, $\zeta(t)$, in the linear regression (4)

$$\hat{\theta}_N^{IV} = \left[\frac{1}{N} \sum_{t=1}^{N} \zeta(t)\varphi^T(t) \right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \zeta(t)y(t) \quad (10)$$

where the instrument variable is chosen as a function of the input $u(t)$ and the output $y(t)$, with the following properties

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \zeta(t)\varphi^T(t) \text{ nonsingular} \qquad (11)$$

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \zeta(t)e(t) = 0 \qquad (12)$$

With these properties, (10) is shown to be consistent by taking the same steps as in (8).

Many studies have been done on IV-methods of which early examples are (Reiersol, 1941; Wong and Polak, 1967). The methods can also be found in textbooks on system identification e.g. (Ljung, 1999; Söderström and Stoica, 1989).

2.1.2. Two-step ARX estimation
An alternative to the IV-methods to obtain a consistent plant of correct order is the following two-step estimation. The first step is to estimate a high order ARX-model. This will give the $A(q)$ polynomial a sufficient number of parameters so that both the denominators of the plant and noise parts can be modelled, and the extra

parameters in the $B(q)$ polynomial are then used to cancel the poles which are only needed in the noise part. If the order of the model tends towards infinity, consistency is obtained (Ljung and Whalberg, 1992). In a second step the high order model is simulated using the input signal from the estimation data. The simulated output

$$y_s(t) = \frac{\hat{B}(q)}{\hat{A}(q)} u(t) \qquad (13)$$

is an estimate of the noise-free output from the system. Finally a low order ARX-model is estimated using the simulated outputs and the inputs from the estimation data. Since the model is estimated from approximately noise free-data, (9) holds and the parameter estimate is consistent, see for example (Pandya, 1974).

2.2 Local linear models

A local model means that a subset of the estimation data of k data points around the point in the regressor space φ_0 one wants to predict or simulate is chosen. From the k data in the subset, a model is estimated by minimizing the criterion

$$\hat{\theta} = \min \arg_{\theta} \sum_{t=1}^{k} (y(t) - (\varphi(t) - \varphi_0)^T w_t \theta_{ab} - \hat{y})^2 \qquad (14)$$

where $\hat{\theta} = [\hat{y} \ \hat{\theta}_{ab}^T]^T = [\hat{y} \ a_1...a_{n_a} \ b_0...b_{n_b}]^T$ and w_t are weights on the data in $\varphi(t)$, giving higher weight on data close to φ_0. Note that the estimate of y at φ_0, denoted $\hat{y}$, becomes one of the parameters in (14). The estimate the parameters $\hat{\theta}$ can according to (14) be formulated as

$$\hat{\theta} = \left[\frac{1}{k} \sum_{t=1}^{k} \bar{\varphi}(t) w_t \bar{\varphi}^T(t) \right]^{-1} \frac{1}{k} \sum_{t=1}^{k} \bar{\varphi}(t) w_t y(t) \qquad (15)$$

where

$$\bar{\varphi}(t) = \begin{pmatrix} 1 \\ \varphi(t) - \varphi_0 \end{pmatrix} \qquad (16)$$

and the prediction $\hat{y}$ is the first element in $\hat{\theta}$.

The size of k can either be a constant number or be optimized for each φ_0 by minimizing a goodness-of-fit-criterion. For examples on goodness-of-fit criteria see (Stenman, 1999). The weights, w_t, can be set by a kernel function (Atkeson *et al.*, 1997), $w_t = K(d(\varphi(t), \varphi_0))$ where $d(\varphi(t), \varphi_0)$ is a distance function. An alternative to set the weights with a kernel function is to optimize the distribution of the weights, see (Stenman, 1999) and (Roll *et al.*, 2003).

3. ALGORITHMS

3.1 First step

The first step in both of the algorithms is a local ARX-model with a regressor of higher order than the expected system order

$$\hat{\theta} = \left[\frac{1}{k} \sum_{t=1}^{k} \bar{\bar{\varphi}}(t) w_t \bar{\bar{\varphi}}^T(t) \right]^{-1} \frac{1}{k} \sum_{t=1}^{k} \bar{\bar{\varphi}}(t) w_t y(t) \qquad (17)$$

where

$$\tilde{\varphi}(t) = [-y(t-1)... - y(t-\tilde{n}_a)$$
$$u(t-n_k)...u(t-n_k-\tilde{n}_b+1)]^T \qquad (18)$$

note that $\bar{\bar{\varphi}}(t)$ is defined in analogy with (16). The simulated output from (17) is called $y_s(t)$. The simulated data forms a new set of estimation data $[y_s(t) \ u(t)]_{t=1}^{N}$

3.2 Local two-step ARX

The first algorithm is a local version of the two-step ARX estimation from section 2.2.

- The first step is according to section 3.1
- Estimate and simulate a local ARX-model of the expected system order $[n_a \ n_b \ n_k]$

$$\hat{\theta} = \left[\frac{1}{k} \sum_{t=1}^{k} \bar{\zeta}(t) w_t \bar{\zeta}^T(t) \right]^{-1} \frac{1}{k} \sum_{t=1}^{k} \bar{\zeta}(t) w_t y_s(t) \qquad (19)$$

where

$$\zeta(t) = [-y_s(t-1)... - y_s(t-n_a)$$
$$u(t-n_k)...u(t-n_k-n_b+1)]^T \qquad (20)$$

and $\tilde{n}_a > n_a$ and $\tilde{n}_b > n_b$, from the new estimation data set. Note that $\bar{\zeta}(t)$ is defined in analogy with (16).

3.3 Local IV-method

The second algorithm is a local version of the IV-method from section 2.1.1.

- The first step is according to section 3.1
- Form an instrument of the simulated outputs,

$$\zeta(t) = [-y_s(t-1)... - y_s(t-n_a)$$
$$u(t-n_k)...u(t-n_k-n_b+1)]^T \qquad (21)$$

- Estimate and simulate a local ARX-model of the order $[n_a \ n_b \ n_k]$, using the instrument $\zeta(t)$

$$\hat{\theta} = \left[\frac{1}{k} \sum_{t=1}^{k} \bar{\zeta}(t) w_t \bar{\varphi}^T(t) \right]^{-1} \frac{1}{k} \sum_{t=1}^{k} \bar{\zeta}(t) w_t y(t) \qquad (22)$$

where

$$\varphi(t) = [-y(t-1)... - y(t-n_a)$$
$$u(t-n_k)...u(t-n_k-n_b+1)]^T \qquad (23)$$

4. EXAMPLES

4.1 Simulated data

Three simple examples on simulated data will be used to compare the local two-step ARX-model, the local

IV-method and a local ARX-model of the system order. The influence of the number of data and model order in section 3.1 will be illustrated as well as the variance and bias parts of the estimation error of the simulated output. In all the examples data have been generated by the function

$$y_0(t) = 0.5u(t-1)(1-\sin(y_0(t-1)))$$
$$y(t) = y_0(t) + e(t) \quad (24)$$

with a system order of $[n_a\ n_b\ n_k] = [1\ 1\ 1]$ and where $\{u(t)\}_{t=1}^N$ are randomly generated numbers equally distributed between -2.5 and 2.5 and $\{e(t)\}_{t=1}^N$ are Gaussian white noise with zero mean and unity variance. The noise free output $\{y_0(t)\}_{t=1}^N$ is used as validation data.

4.1.1. Example I, convergence as a function of the number of data   The aim of example I is to demonstrate how the simulated output converges to the true output for increasing number of data. The model order
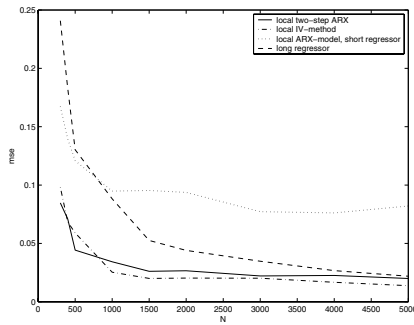

Fig. 1. On the *x*-axis the number of estimation data and on the *y*-axis is the mean squared error between the simulated outputs and the validation output. The solid line is the local two-step ARX, the dashed/dotted line is the local IV-method, the dotted line is the local model of the system order and the dashed line is the local model of the order $[5\ 5\ 1]$.

in section 3.1 is chosen to $[\tilde{n}_a\ \tilde{n}_b\ n_k] = [5\ 5\ 1]$. The number of estimation data is varied from 300 to 5000. The mean squared error between the simulated output and the validation output is calculated for each simulation.

In figure 1 the mean squared errors calculated for the local two-step ARX-model, the local IV-method and a local model of the system order is plotted versus the number of data points. The mean squared error for the local model in section 3.1 is also plotted. The performance of this model is of interest since its simulated output is the estimation data in the second step of the local two-step ARX (19) and is the instrument in the local IV-method (22). Figure 1 shows that both the local two-step ARX-model and the local IV-method give smaller mean squared errors for larger amounts of estimation data and that they both give smaller mean

squared error than the local model of order $[n_a\ n_b\ n_k]$. It can also be seen that mean squared error of the local two-step ARX converges to (17) faster than the local IV-method.

4.1.2. Example II, bias error and variance error
The aim of example II is to show how the bias and variance parts of the estimate error are affected by using the local two-step ARX-model and the local IV-method compared to a local model of the system order. A data set of 1000 points from (24) is used. The regressor in section 3.1 has the order $[\tilde{n}_a\ \tilde{n}_b\ n_k] = [5\ 5\ 1]$. The results of the two-step algorithms are compared with a local model of the system order. The calculations are run 200 times with different noise realizations $\{e(t)\}_{t=1}^N$. To make it easier to illustrate the result, two outputs in the data set have been chosen: output number $t = 600$ and output number $t = 700$, which are plotted against each other in figure 2. Figure 2 also shows the mean of the 200 outputs and the noise-free value of the outputs. The bias error has been calculated as

$$\text{bias} = ((\frac{1}{200}\sum_{i=1}^{200}(y_s^i(600)) - y_0(600))^2 +$$
$$+ (\frac{1}{200}\sum_{i=1}^{200}(y_s^i(700)) - y_0(700))^2)^{1/2} \quad (25)$$

and the variance error as

$$\text{variance} = \text{variance}(y_s(600)) + \text{variance}(y_s(700))$$
$$(26)$$

Figure 2 shows that the local two-step ARX-model, figure 2(a), and the local ARX-model of the system order, figure 2(c), give approximately the same variance error of the simulated output, but the bias error is, as expected greater for the local ARX-model of the system order. The local IV-method on the other hand gives a greater variance of the simulated outputs, but has the smallest bias, figure 2(b).

4.1.3. Example III, model order   The aim of example III is to demonstrate how the order of the regressor in section 3.1 will affect the estimation result. In this example two data sets from (24) are used: one of 1000 data and one of 5000. The data sets are simulated using the local two-step ARX (19) and the local IV-method (22). The order of (18), $\tilde{n}_a = \tilde{n}_b = n$, goes from 1 to 10. For each model size, the mean squared error between validation output and the simulated output is calculated. The results of the calculations are shown in figure 3. The figure shows that for the local two-step ARX-model the order of the large regressor (18) is important and that the order of the best larger model depends on the number of data. In the case when the number of data was 1000, the optimum is $[2\ 2\ 1]$ and for $N = 5000$ the optimum is $[3\ 3\ 1]$. The local IV-method seems to be robuster towards the order of the model in section 3.1.
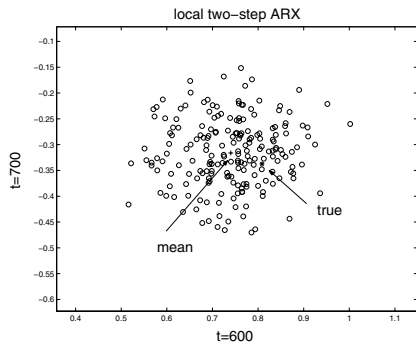
(a) Local two-step ARX, bias $= 0.0716$, variance $= 0.0127$
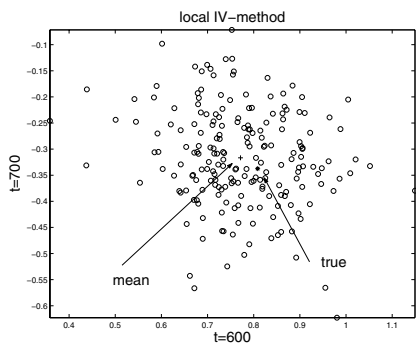
(b) Local IV-method, bias $= 0.0425$, variance $= 0.0224$

(c) Local ARX, system order, bias $= 0.1088$, variance $= 0.0135$

Fig. 2. Output 600 on the *x*-axis vs point 700 on the *y*-axis. The circles are the simulated outputs, the crosses are the mean of the simulations and the stars are the noise free outputs. The arrows are help to find the crosses and the stars.

4.2 DC-motor

The algorithms will be illustrated on laboratory data from a DC-motor with a flexible torsional load. The input is the voltage applied to the motor and the output is the angular speed of the load. The linear trend in the data is removed and a data set of 1191 data point is divided into an estimation set of the first three quarters
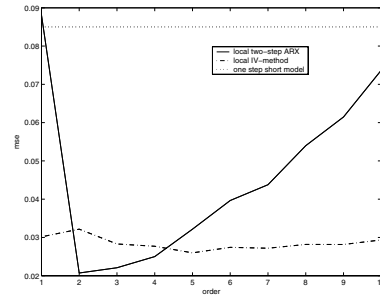
(a) Number of data 1000
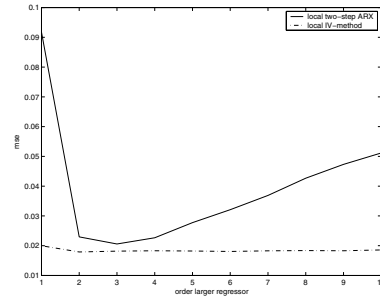
(b) Number of data 5000

Fig. 3. The mean squared error between simulated output and validation output vs the model order. The solid line is the local two-step ARX-model, the dashed line the local IV-method and the dotted line is the local model of the system order.

of the data and a validation set of the last quarter of the data, see figure 4. The data in the estimation set
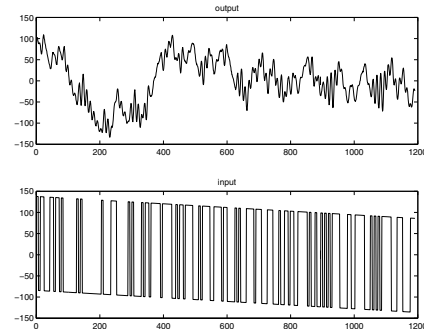

Fig. 4. Data from DC-motor. The first three quarters are used as estimation data and the last quarter as validation data.

is simulated with the local two-step ARX-algorithm, the local IV-algorithm and a local ARX-model of order [3 2 2]. The model in the first step has the order [5 4 2] and the model in the second step has the order [3 2 2]. To compare the results the mean squared error between the output and simulated output

$$\text{mse} = \frac{1}{N} \sum_{t=1}^{N} (y_{val}(t) - y_s(t))^2 \qquad (27)$$
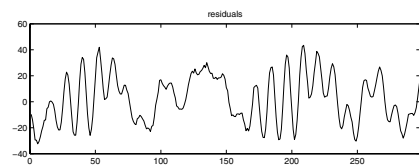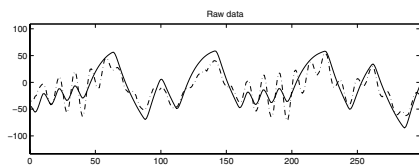
is calculated for each case. Figure 5 shows that the local IV-method preforms the best and the local two
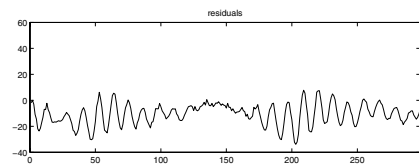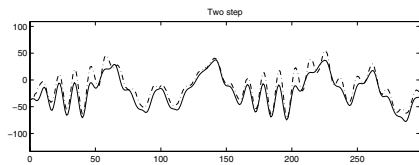
(a) Local two step ARX, mse= 176.7080
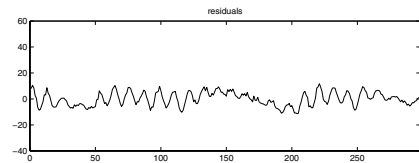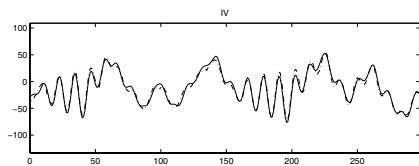
(b) Local IV, mse= 27.0949

(c) Local ARX, mse= 341.5614

Fig. 5. Simulations of data from a DC-motor and the residuals between the simulation and the validation data by the local two-step ARX-model and the local IV-method compered with a local ARX-model of order [3 2 2]. The solid line is the simulated output and the dashed/dotted line is the validation output.

step ARX simulates data well, but the local ARX-model of order [3 2 2] does not.

5. CONCLUSIONS

In this study we have proposed two parameter estimation algorithms for consistent estimates of local linear models. The estimation result's dependence on the order of the model used to simulate an approximately noise-free data set and the number of data has been illustrated in a couple of examples.

REFERENCES

Atkeson, C. G., A. W. Moore and S. Schaal (1997). Locally weighted learning. *Artificial Intelligence Review* **72**, 643–58.

Bontempi, G., M. Birattari and H. Bersini (1999). Lazy learning for local modeling and control. *International Journal of Control* **11**, 11–73.

Chen, S. and S.A. Billings (1992). Neural networks for nonlinear dynamic system modelling and identification. *Int. J. Control* **56**(2), 319–346.

Ljung, L. (1999). *System identification*. second ed.. Prentice hall.

Ljung, L. and B. Whalberg (1992). Asymtotic properties of the least-squares method for estimating transfer functions and disturbance spectra. *Adv. Appl. Prob.* **24**, 412–440.

Nørgaard, M., O. Ravn, N.K. Poulsen and L.K. Hansen (2000). *Neural Networks for Modelling and Control of Dynamic Systems*. Advanced Textbooks in Control and Signal Processing. Springer-Verlag. London. http://kalman.iau.dtu.dk/neuralnet.html.

Pandya, R. N. (1974). A class of bootstrap estimators and their relationship to the generalized two stage least squares estimaors. *IEEE transactions on automatic control* **19**, 831–835.

Reiersol, O. (1941). Confluence analysis by mean of lag moments and other methods of confluence analysis. *Econometrica* **9**, 1–23.

Roll, J., A. Nazin and L. Ljung (2003). Local modelling of nonlinear dynamic systems using direct weight optimization. In: *13th IFAC symposium on system identification*.

Sjöberg, J., Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P-Y. Glorennec, H. Hjalmarsson and A. Juditsky (1995). Nonlinear black-box modelling in system identification: a unified overview. *Automatica* **31**, 1691–1724.

Söderström, T. and P. Stoica (1989). *System Identification*. Prentice-Hall International. Hemel Hempstead, Hertfordshire.

Stenman, A. (1999). Model on Demand: Algorithms, Analysis, Applications. PhD thesis. Linköping University, Sweden.

Wong, K. Y. and E. Polak (1967). Idenitification of linear discrete time systems using the instrument variable method. *IEEE transactions on automatic control* **12**, 707–718.