
Proceedings Paper:
Jacobs, W.R., Baldacchino, T. and Anderson, S.R. orcid.org/0000-0002-7452-5681 (2015) Sparse Bayesian identification of polynomial NARX models. In: IFAC-PapersOnLine. 17th IFAC Symposium on System Identification SYSID, 19-21 Oct 2015, Beijing, China. Elsevier BV , pp. 172-177.

https://doi.org/10.1016/j.ifacol.2015.12.120

Sparse Bayesian Identification of Polynomial NARX Models

William R. Jacobs * **Tara Baldacchino** ** **Sean R. Anderson** *

* *Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK.*
** *Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Sheffield, UK.*
{*w.jacobs,s.anderson,t.baldacchino*}*@sheffield.ac.uk.*

Abstract: In this paper a novel sparse Bayesian structure detection algorithm is introduced for the identification of nonlinear autoregressive with exogenous inputs (NARX) dynamic systems. The main advantage of this algorithm over alternatives is that parameter uncertainty is naturally incorporated, and parameter estimation by variational inference is computationally efficient, consisting of a sequence of closed form updates. The proposed framework is demonstrated through a commonly used simulated benchmark problem.

Keywords: NARX models, variational Bayes, system identification, automatic relevance determination

1. INTRODUCTION

A popular model class for the identification of nonlinear dynamic systems is the polynomial nonlinear autoregressive with exogenous inputs (NARX) model (Leontaritis and Billings, 1985). One reason for the popularity of the NARX model is that it can produce a much more compact description compared to the Volterra series class of model, especially when there are nonlinear output terms in the description (Chen and Billings, 1989). The NARX model class also possesses a linear-in-the-parameters structure that simplifies parameter estimation. The most challenging part of the system identification procedure for NARX models is structure detection: choosing a subset of terms from a superset to be included in the final model, leading to a parsimonious representation of the system under investigation.

System identification, specifically structure detection, of NARX models has received a great deal of attention since the model class was first established. A popular method is the forward regression orthogonalisation (FRO) algorithm, developed by Chen et al. (1989), which calculates an error reduction ratio based on the one step ahead model prediction. FRO has undergone several modifications over the years - see Billings (2013) and references therein. Piroddi and Spinelli (2003) adopted a method which aims to minimise the simulation error - especially advantageous when the input is not persistently exciting, while Baldacchino et al. (2012) used an expectation-maximisation (EM) algorithm for identification, which is useful in scenarios where there are missing data. Kukreja et al. (2004) developed an algorithm that quantifies parameter uncertainty using bootstrapping.

⋆ W. Jacobs is financially supported by an EPSRC scholarship (EP/K503149/1).

There is however, little precedent for the identification of NARX models within a Bayesian framework. Bayesian learning is advantageous for a number of reasons: (i) it naturally penalises overly complex models thus avoiding overfitting, (ii) it naturally captures uncertainty about the model, useful in simulation and control design, (iii) it can accurately quantify model uncertainty even for short data lengths and (iv) it uses prior distributions, which give the modeller a much greater degree of influence over the modelling process because information about the system can be incorporated into the priors (Gelman et al., 2014).

To our knowledge, the only instance of a Bayesian approach to parametric modelling of NARX models appears in Baldacchino et al. (2013). The authors of that paper use a Markov chain Monte Carlo (MCMC) sampling method in order to numerically obtain posterior distributions of both the model structure and parameters. However, sampling methods are often computationally intensive to implement because they tend to rely on large numbers of samples to accurately estimate distributions (Ninness and Henriksen, 2010).

There is therefore a need for computationally efficient Bayesian system identification techniques capable of producing parsimonious models of dynamic systems. In this work we present a novel Bayesian structure detection algorithm capable of identifying accurate and compact NARX models, which is both simple to implement and relatively computationally efficient. The NARX identification algorithm is based on variational Bayesian inference, resulting in a sequence of closed-form equations in an iterative algorithm (Bishop, 2006). Structure detection is driven by the inclusion of a sparsity inducing hyperprior, referred to as automatic relevance determination (ARD), which is used to prune redundant terms from the model (MacKay, 1995).

The paper is organized as follows. Section 2 first introduces the polynomial NARX model and then continues to present the approximate Bayesian inference problem and the structure detection algorithm. Section 3 gives a simulated example of the modelling framework.

2. METHODS

2.1 The polynomial NARX model representation

Discrete-time input-output dynamic systems can be efficiently represented as a polynomial NARX model Leontaritis and Billings (1985). The system is described by some unknown nonlinear function, $f(.)$, of lagged system inputs, $\mathbf{u} = [u_1, u_2, \ldots, u_N]^T$ and outputs, $\mathbf{y} = [y_1, y_2, \ldots, y_N]^T$

$$y_t = f(\boldsymbol{\phi}_t) + e_t, \tag{1}$$

where

$$\boldsymbol{\phi}_t = [y_{t-1}, ..., y_{t-n_y}, u_{t-1}, ..., u_{t-n_y}]. \tag{2}$$

e_t are independently sampled from an iid white noise sequence and are assumed to be zero-mean and variance σ^2. n_y and n_u represent the maximum dynamic order of the output and input terms respectively.

The nonlinear function $f(.)$ can be decomposed into a sum of weighted basis functions such that

$$f(\boldsymbol{\phi}_t) = \sum_{j=1}^{M} \theta_j \phi_t^j, \tag{3}$$

where M is the number of terms in the model, θ_j is the jth model parameter and ϕ_t^j is the j'th polynomial basis function. Hence, in matrix form the polynomial NARX model is defined as

$$\mathbf{y} = \Phi\boldsymbol{\theta} + \mathbf{e}, \tag{4}$$

such that

$$\Phi = [\phi_1; \phi_2; \ldots; \phi_N], \tag{5}$$

$$\theta = [\theta_1, \theta_2, \ldots, \theta_{M,}]^T, \tag{6}$$

The structure detection task is defined as the selection of a subset of M basis functions from the complete set, denoted $\mathcal{M}$, that forms a parsimonious description of the system dynamics. Parameter estimation is greatly simplified by the linear in the parameters structure of the NARX model allowing the use of least squares based techniques.

2.2 Bayesian linear regression

The posterior distribution over the parameters of the NARX model defined by equation (4) is given by Bayes theorem as

$$p(\boldsymbol{\theta}, \tau|\mathbf{y}) = \frac{p(\mathbf{y}|\Phi, \boldsymbol{\theta}, \tau)p(\boldsymbol{\theta}, \tau)}{p(\mathbf{y})}, \tag{7}$$

where $p(\mathbf{y}|\Phi, \theta, \tau)$ is the likelihood function, $p(\theta, \tau)$ is the joint prior distribution over the model parameters and precision, $\tau = 1/\sigma^2$, and $p(\mathbf{y})$ is the model evidence or marginal likelihood.

Under the assumption of Gaussian noise the likelihood function for the data, $\mathbf{y}$, can be written

$$p(\mathbf{y}|\Phi, \boldsymbol{\theta}, \tau) = \prod_t p(y_t|\boldsymbol{\phi}_t, \boldsymbol{\theta}, \tau) \tag{8}$$

$$= \prod_t \mathcal{N}(y_t|\boldsymbol{\phi}_t\boldsymbol{\theta}, \tau^{-1}) \tag{9}$$

where $\mathcal{N}$ represents the Gaussian distribution.

2.3 Priors and Automatic Relevance Determination

Bayesian estimation incorporates prior knowledge of the system under investigation into the modelling process through the choice of prior distributions of the model parameters. Modern Bayesian estimation can use non-informative prior distributions in order to force the inference of the posterior distribution to be driven by the data. In this work a sparsity inducing hyper-prior is introduced which will be the mechanism through which the model structure detection is driven.

The conjugate normal gamma distribution is chosen as the prior over θ and τ such that

$$p(\boldsymbol{\theta}, \tau|\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\theta}|0, (\tau A)^{-1})\mathrm{Gam}(\tau|a_0, b_0), \tag{10}$$

where A is a matrix with diagonal elements $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_M)$, which is a vector of hyperparameters that are independently associated with each parameter of the model and Gam is the Gamma distribution.

The hyperparameter, $\boldsymbol{\alpha}$, is assigned the hyperprior

$$p(\boldsymbol{\alpha}) = \prod_{j=0}^{M} \mathrm{Gam}(\alpha_j|c_0, d_0). \tag{11}$$

The hyperparameter A effectively determines the extent to which each of the the model parameters is allowed to move from zero. Hence the sparsity of the inferred model parameters can be directly controlled by the choice of hyperparameter $p(\boldsymbol{\alpha})$. The value of α_j provides a measure of how relevant the corresponding input (basis function) is to the model, $\alpha_j^{-1} = 0$ implies that the corresponding basis function is not relevant to forming the distribution of the output.

Having defined all the priors of the model the parameter posterior is reformulated to include all the unknowns

$$p(\boldsymbol{\theta}, \boldsymbol{\alpha}, \tau|\mathbf{y}) = \frac{p(\mathbf{y}|\Phi, \boldsymbol{\theta}, \boldsymbol{\alpha}, \tau)p(\boldsymbol{\theta}, \boldsymbol{\alpha}, \tau)}{p(\mathbf{y})}. \tag{12}$$

2.4 Variational inference

Due to the introduction of the hyperprior (11) we are unable to compute the posterior (12) because the model evidence $p(\mathbf{y})$, in the denominator, is intractable. In order to overcome this difficulty an approximate learning algorithm is required to train the model. Here we use variational Bayesian linear regression although other Bayesian model fitting methods, such as relevance vector machines (Tipping, 2001) could be used.

An outline of the variational framework only is presented here for brevity. The interested reader is referred to Bishop (2006) for a fuller description. Variational inference relies on the assumption that the posterior $p(\boldsymbol{\theta}, \boldsymbol{\alpha}, \tau|\mathbf{y})$ can be approximated by the the variational distribution $q(\boldsymbol{\theta}, \boldsymbol{\alpha}, \tau)$ which factors into $q(\boldsymbol{\theta}, \boldsymbol{\alpha})q(\tau)$.

Given that the approximation is a good one, it can be shown, Bishop (2006), that the variational posteriors can be found by maximisation of the variational lower bound

$$\begin{aligned}\mathcal{L}(Q) = &\iiint q(\boldsymbol{\theta},\tau,\boldsymbol{\alpha}) \\ &\ln\frac{P(\mathbf{y}|\Phi,\boldsymbol{\theta},\tau)P(\boldsymbol{\theta},\tau|\boldsymbol{\alpha})P(\boldsymbol{\alpha})}{q(\boldsymbol{\theta},\tau,\boldsymbol{\alpha})}\mathrm{d}\boldsymbol{\theta}\mathrm{d}\tau\mathrm{d}\boldsymbol{\alpha} \quad (13)\\ &\leq \ln p(\mathbf{y}),\end{aligned}$$

where $P(\mathbf{y})$ is the model evidence and the assumption has been made

Using the calculus of variations we can obtain the following variational posteriors, see Drugowitsch (2013) for derivations,

$$q(\boldsymbol{\theta},\tau) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_K,\tau^{-1}V_K)\mathrm{Gam}(\tau|a_K,b_K), \quad (14)$$

$$q(\boldsymbol{\alpha}) = \prod_{j=1}^{m}\mathrm{Gam}(\alpha_j|c_K,d_K), \quad (15)$$

where

$$\begin{aligned}V_K^{-1} &= \mathbf{E}_\alpha[A] + \Phi^T\Phi,\\ \theta_K &= V_K\Phi^T\mathbf{y},\\ a_K &= a_0 + \frac{N}{2},\\ b_K &= b_0 + \frac{1}{2}((\mathbf{y}-\Phi\boldsymbol{\theta}_K)^T(\mathbf{y}-\Phi\boldsymbol{\theta}_K)\\ &\quad +\boldsymbol{\theta}^T\mathbf{E}_\alpha(A)\boldsymbol{\theta}_K, \quad (16)\\ c_K &= c_0 + \frac{1}{2},\\ d_{K,j} &= d_0 + \frac{1}{2}\mathbf{E}_{\theta,\alpha}[\tau\theta_j^2],\\ \mathbf{E}_{\theta,\alpha}[\tau\theta_j^2] &= \theta_j^2\frac{a_K}{b_K} + (V_K)_{jj},\end{aligned}$$

where $\mathbf{E}_\alpha[A]$ is a diagonal matrix with elements $\mathbf{E}_\alpha[A] = \mathrm{diag}(c_K/d_{K,1},\ldots,c_K/d_{K,M})$ and the subscript K indicates the current iteration over k. The distribution (14) is updated using the expectation of the statistics of (15) and visa-versa in an iterative algorithm that is terminated when the change in the lower bound $\mathcal{L}(Q)$ is less than the threshold $T_{\mathcal{L}(Q)}$, see Algorithm 1. The method is hence analogous to the EM algorithm within a Bayesian context. The variational lower bound is given by

$$\begin{aligned}\mathcal{L}(Q) = &-\frac{N}{2}\ln 2\pi - \frac{1}{2}\sum_n\left(\frac{a_K}{b_K}(y_n-\boldsymbol{\phi}_t\boldsymbol{\theta}_K)^2 + \boldsymbol{\phi}_t^TV_K\boldsymbol{\phi}_t\right)\\ &+\frac{1}{2}\ln|V_K| + \frac{M}{2} - \ln\Gamma(a_0) + a_0\ln b_0 - b_0\frac{a_K}{b_K} + a_K\\ &+\sum_j(c_0\ln d_0 - \ln\Gamma(c_0) + \ln\Gamma(c_K) - c_K\ln d_{K,j}). \quad (17)\end{aligned}$$

The quantity $\mathcal{L}(Q)$ can be used as a measure of model performance without risk of over-fitting since it provides a lower bound for $p(\mathbf{y})$.

The predictive density is obtained by marginalising the product of the likelihood and posterior distributions with respect to $\boldsymbol{\theta}$ and τ. In order to achieve this, the posterior density $p(\boldsymbol{\theta},\tau,\boldsymbol{\alpha}|\mathbf{y})$ is approximated by the variational posterior $q(\boldsymbol{\theta},\tau)q(\boldsymbol{\alpha})$ given by equation (14) such that

$$\begin{aligned}p(y_{t'}|\Phi) &= \iint p(y_{t'}|\boldsymbol{\phi}_{t'},\boldsymbol{\theta},\tau)p(\boldsymbol{\theta},\tau,\boldsymbol{\alpha}|\mathbf{y})\mathrm{d}\boldsymbol{\theta}\mathrm{d}\tau\mathrm{d}\alpha\\ &= \iiint p(y_{t'}|\boldsymbol{\phi}_{t'},\boldsymbol{\theta},\tau)q(\boldsymbol{\theta},\tau)q(\boldsymbol{\alpha})\mathrm{d}\boldsymbol{\theta}\mathrm{d}\tau\mathrm{d}\alpha \quad (18)\\ &= \mathrm{St}\left(y_{t'}|\boldsymbol{\phi}_{t'}\boldsymbol{\theta}_K,(1+\boldsymbol{\phi}_t^TV_K\boldsymbol{\phi}_t)^{-1}\frac{a_K}{b_K},2a_K\right)\end{aligned}$$

where $t' = t+1$ and St is the Student-t distribution. The above steps make use of standard results in convolving probability distributions. $\boldsymbol{\alpha}$ does not appear in the predictive distribution because it integrates to unity since it does not appear in the likelihood. The mean of the Student-t distribution is $\boldsymbol{\phi}_{t'}\boldsymbol{\theta}_K$ and the predictive variance is $(1+\boldsymbol{\phi}_t^TV_K\boldsymbol{\phi}_t)b_K/(a_K-1)$.

2.5 Structure detection

In order to achieve parsimonious structure detection at each iteration (denoting this iteration number by i) the variational inference procedure, described in the previous section, is performed for different model structures.

At the $i'th$ iteration, a new model structure $\mathcal{M}_i$ is determined by calculating the ARD values associated with each parameter of the current model structure $\mathcal{M}_{i-1}$ expressed as

$$ARD_j = \{(c_K/d_{K,j})^{-1}\}_{j-1}^M \quad (19)$$

Basis functions corresponding to ARD values that fall below some pre-defined value, T^i_{ARD}, are pruned from the model, where

$$T^i_{ARD} = \frac{(\max(ARD^i) - \min(ARD^i)}{r} \quad (20)$$

and r is a tuning parameter. Small values of r result in pruning more terms at each iteration, whilst larger values of r tend to retain more terms in the model at each iteration, hence increasing computation time.

The structure detection algorithm (Algorithm 1) will converge to a model of size $M = 1$ in a finite number of iterations because the algorithm is guaranteed to prune at least one term from the superset of model terms at each iteration. The choice of optimal model structure is simply the model that corresponds to the maximum of the variational lower bound $L(Q)_i$.

2.6 Model performance

The performance of the NARX models identified using Algorithm 1 is evaluated using an independent validation data set consisting of new unseen input data and the performance is measured by the fit to the real data. The MSPE (Mean square prediction error) given by

$$\mathrm{MSPE} = \frac{1}{N}\sum_t(y_t - \hat{y}_t)^2 \quad (21)$$

is used as a measure of performance.

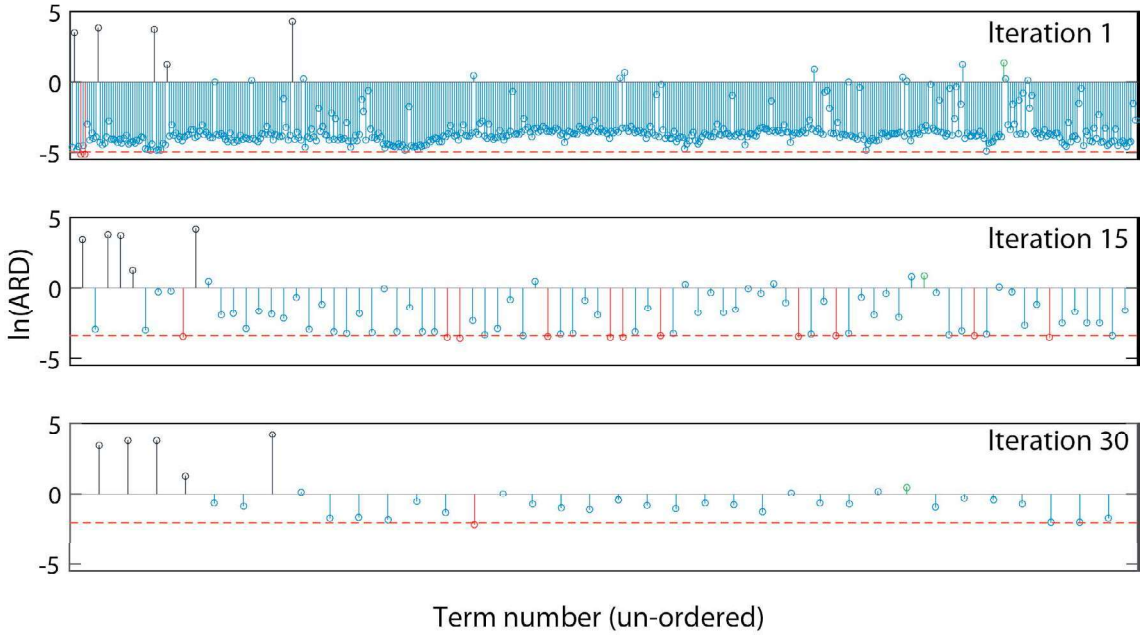

Fig. 1. ARD values for the NARX model at iteration 1,15 and 30. The correct model terms given in equation (22) are coloured black, a competing term ($y_{t-4}^2 u_{t-1} u_{t-4}$) is shown in green. The threshold for pruning terms is indicated by the dashed red line and pruned terms are also coloured red.

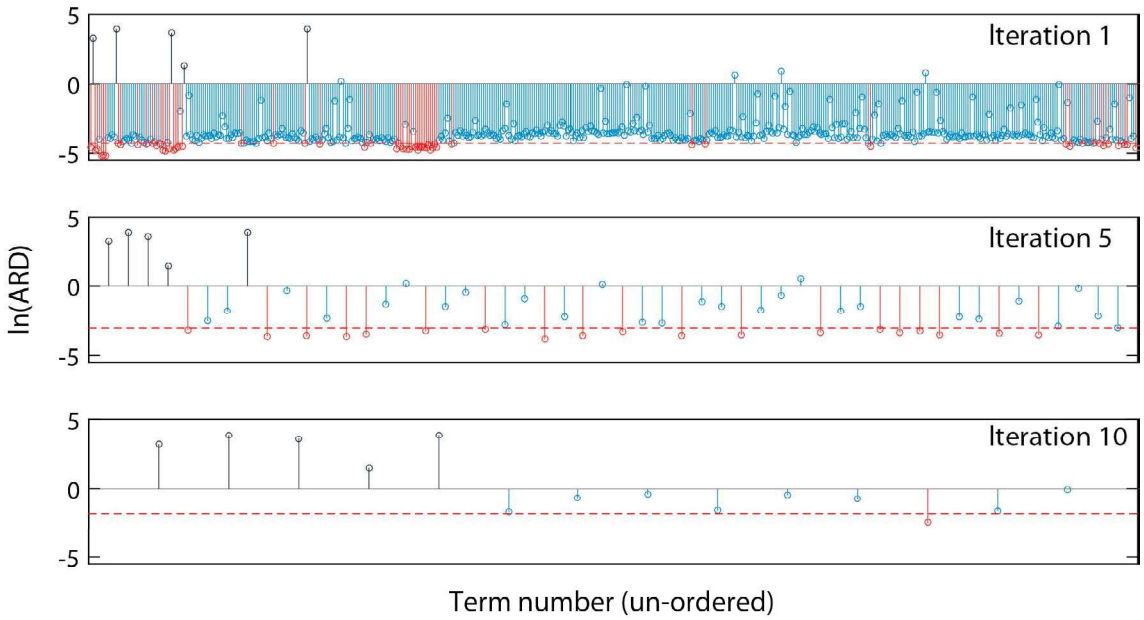

Fig. 2. ARD values for the NARX model at iteration 1,5 and 10 with $r = 10$. The correct model terms given in equation (22) are coloured black. The threshold for pruning terms is indicated by the dashed red line and pruned terms are also coloured red.

3. NUMERICAL EXAMPLE

3.1 A nonlinear benchmark example

In order to investigate the performance of Algorithm 1 we use the following test system, which has previously been used for benchmarking in Mao and Billings (1997), Piroddi and Spinelli (2003) and Baldacchino et al. (2013), particularly because it is designed so that the standard FRO algorithm fails on this problem by selecting incorrect terms,

$$\begin{aligned} y_t = &-0.5y_{t-2} + 0.7y_{t-1}u_{t-1} + 0.6u_{t-2}^2 0.2y_{t-1}^3 \\ &- 0.7y_{t-2}u_{t-2}^2 + e_t. \end{aligned} \tag{22}$$

where the input signal, u_t, is a uniformly distributed white noise sequence in the range $(-1, 1)$ and the noise e_t is normally distributed white noise. The system was used to generate three sets of $N = 1000$ input-output data samples with signal to noise ratio (SNR) $\approx$ 5dB, 10dB and 20dB.

Algorithm 1 was applied to perform structure detection on the nonlinear system given in (22). A superset of basis functions was considered with dynamic order $n_u = n_y = 4$

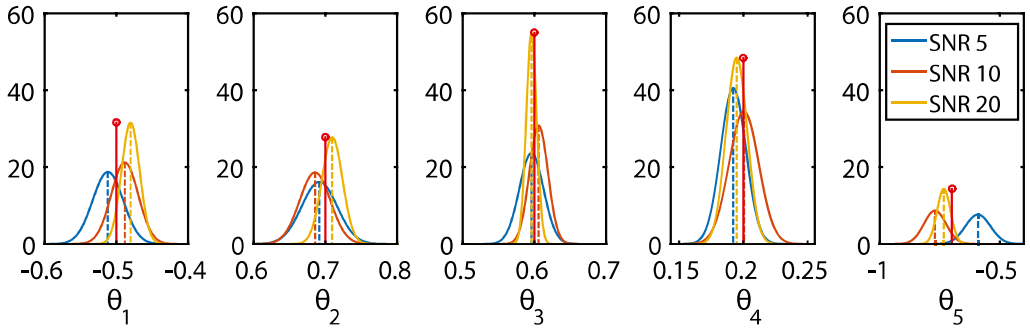

Fig. 3. Estimated parameter distributions for each term identified by the ARD algorithm at SNR $\approx 5, 10$ and 20. The red line indicates the true value and the dashed lines indicate the mean of the Gaussian distribution. Parameter values are given in the same order as for the system given by equation (22).

Table 1. Comparison of true and estimated parameters with assosiated standard deviations for the structure detection of the benchmark model given by equation (22) using the ARD algorithm.

Regressor	True value	SNR ≈ 5	SNR ≈ 10	SNR ≈ 20
y_{t-2}	-0.5	-0.5118 ± 0.0213	-0.4883 ± 0.0189	-0.4802 ± 0.0127
$y_{t-1}u_{t-1}$	0.7	0.6918 ± 0.0247	0.6856 ± 0.0215	0.7096 ± 0.0144
u^2_{t-2}	0.6	0.5960 ± 0.0169	0.6061 ± 0.0130	0.5961 ± 0.0073
y^3_{t-1}	0.2	0.1921 ± 0.0098	0.2004 ± 0.0116	0.1948 ± 0.0082
$y_{t-2}u^2_{t-2}$	-0.7	-0.5911 ± 0.0522	-0.7697 ± 0.0460	-0.7330 ± 0.0279

Algorithm 1: NARX Structure detection

Set $T_{\mathcal{L}(Q)}, T_{\text{ARD}}, a_0, b_0, c_0, d_0,$
Initialise model structure to all basis functions $\mathcal{M}_0 = \{\Phi^j\}_{j=1}^{M}$
$i = 0$
while $M > 1$
  $i = i + 1$
  $k = 1$
  while $\mathcal{L}(Q)_k - \mathcal{L}(Q)_{k-1} \leq T_{\mathcal{L}(Q)}$
    k = k+1
    update parameter estimates for model $\mathcal{M}_{i-1}$ using (16),
    calculate $\mathcal{L}(Q)_k$ via equation (17)
  end while
  Set $\mathcal{L}(Q)_i = \mathcal{L}(Q)_k$
  Calculate T^i_{ARD} via equation (20)
  Initialise pruning terms set, $\mathcal{M}^- = \emptyset$,
  for $j = 1 : |\mathcal{M}_{i-1}|$
    if $ARD_j \leq T^i_{\text{ARD}}$
      collect terms to prune, $\mathcal{M}^- = \mathcal{M}^- \cup \Phi^j$,
    end if
  end for
Set current model structure to $\mathcal{M}_i = \mathcal{M}_{i-1} \setminus \mathcal{M}^-$
end while
Set optimal model $\mathcal{M}^* = \mathcal{M}_{i^*}$ where $i^* = \arg\max_i \mathcal{L}(Q)_i$

and polynomial order $n_p = 4$ providing a set containing $\mathcal{M} = 494$ terms in which to search. Terms were removed from the model that fell below a threshold ARD value, T_{ARD}, given by equation (20) with the resolution $r = 50$, this value was chosen as it provided an acceptable trade off between algorithm speed and accuracy. Uninformative priors were used by choosing hyperparameters for $\text{Gam}(\tau|a_0, b_0)$ and $\text{Gam}(\alpha_i|c_0, d_0)$ as: $a_0 = c_0 = 1 \times 10^{-2}$ and $b_0 = d_0 = 1 \times 10^{-3}$, these are common choices for setting uninformative priors. The number of iterations over the variational update equations (16) was set to 200, this value provided a satisfactory convergence of the variational lower bound, $\mathcal{L}(Q)$. The algorithm was run until all but one term was pruned from the model. The final

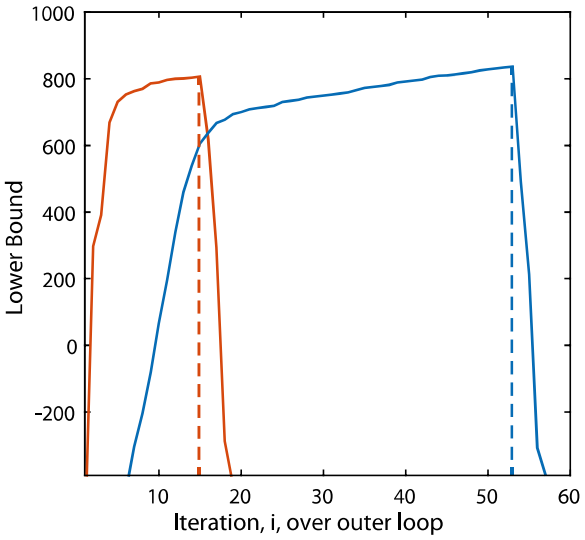

Fig. 4. Variational lower bound against iteration number of the outer loop in Algorithm 1 for $r = 10$ (Red) and $r = 50$ (Blue). The vertical dashed lines indicate the largest value of the lower bound and hence the chosen model.

model $\mathcal{M}^*$ is given by the model that corresponds to the largest value of the variational lower bound, $\mathcal{L}(Q)_i$. The time for the algorithm to run was 83 seconds (Intel i5 3470@3.20GHz,4GBRAM).

The algorithm correctly identified the structure of the system system for all three levels of noise, the final model terms and the estimated parameters are given in Table 1. The progression of the algorithm for SNR $\approx$ 20 dB can be seen in Figure 1, where the ARD values have been plotted on a ln scale for clarity. After the first iteration all the correct model terms, shown in black, are

automatically assigned high ARD values indicating a high relevance to the data. However, some surplus terms are also assigned comparably large values, most notable the term $y_{t-4}^2 u_{t-1} u_{t-4}$ (shown in green) is assigned a value higher than that given to one of the correct model terms. As the algorithm advances the least relevant terms (those associated ARD values falling below a specified threshold indicated by the dashed red line) are pruned from the model (coloured red), the remaining non model terms are observed to gradually decrease in relevance. Posterior distributions over the parameters are calculated for all three noise levels, their means are within one standard deviation of the true parameter values, see Figure 3.

In order to demonstrate the effect of the choice of the tuning parameter r, the algorithm was run again using the same data (SNR$\approx 20dB$) and initialisation but choosing $r = 10$. It can be noted that the pruning of terms is much less conservative in this case, see Figure 2. Although the correct structure is still identified, the figure indicates that the choice of r should be treated with caution as reducing this value more could result in the pruning of desirable model terms. The advantage of choosing a small r is seen in the number of pruning iterations completed by the algorithm and hence computation time, see Figure 4.

Table 2. Comparison of true and estimated parameters for the structure detection of the benchmark model given by equation (22) using the FRO algorithm.

Regressor	ERR	True value	Estimated parameter
$y_{t-4}u_{t-2}^2$	0.3587		0.0086
$y_{t-1}u_{t-1}$	0.1351	0.7	0.6884
u_{t-2}^4	0.1217		−0.0085
y_{t-2}	0.2316	−0.5	−0.5064
u_{t-2}^2	0.0331	0.6	0.6010
$y_{t-2}u_{t-2}^2$	0.0267	−0.7	−0.6900
y_{t-1}^3	0.0239	0.2	0.1940

The structure detection task was performed again using the FRO algorithm, see Table 2. The FRO algorithm predicts more terms than the generating system, a real system is not governed by true polynomial basis functions and so a method which models the dynamics of the system in a parsimonious way is desired. The run time for the FRO algorithm using the same computer was 3374.98 seconds,indicating that a significant speed up is achieved with our algorithm.

4. SUMMARY

The benchmark problem tackled in Section 3.1 demonstrates the applicability of the ARD structure detection framework developed in this work. The correct model structure is identified in a situation where standard FRO techniques fail. Although more sophisticated additions to the FRO scheme as well as simulation based algorithms have achieved success in this area, these methods tend to be computationally intensive and do not naturally describe the uncertainty in the model parameters. The novel algorithm presented here has distinct advantages in this respect. To further this investigation we need to consider the effects of how uncertainty in the parameters effects the prediction.

The main contribution of the algorithm presented in this work is the quantification of the model uncertainty within a Bayesian framework that is simple to implement and computationally efficient. The simplicity of the method derives from the closed form nature of the variational update equations, removing the necessity of using sampling methods that are required by many Bayesian algorithms. In this case the steps in the iterative identification algorithm are closed form, which leads to relatively fast computation times.

REFERENCES

Baldacchino, T., Anderson, S.R., and Kadirkamanathan, V. (2012). Structure detection and parameter estimation for NARX models in a unified EM framework. *Automatica*, 48(5), 857–865.

Baldacchino, T., Anderson, S.R., and Kadirkamanathan, V. (2013). Computational system identification for bayesian NARMAX modelling. *Automatica*, 49(9), 2641–2651.

Billings, S.A. (2013). *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. Wiley.

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Chen, S. and Billings, S.A. (1989). Representations of non-linear systems: the NARMAX model. *International Journal of Control*, 49(3), 1013–1032.

Chen, S., Billings, S.A., and Luo, W. (1989). Orthogonal least squares methods and their application to non-linear system identification. *International Journal of Control*, 50(5), 1873–1896.

Drugowitsch, J. (2013). Variational bayesian inference for linear and logistic regression.

Gelman, A., Carlin, J., Stern, H.S., Dunson, D.B., Vehtari, A., and Rubin., D.B. (2014). *Bayesian data analysis. Third Edition*. CRC Press.

Kukreja , S.L., Galiana, H.L., and Kearney, R.E. (2004). A bootstrap method for structure detection of NARMAX models. *International Journal of Control*, 77(2), 132–143.

Leontaritis, I.J. and Billings, S.A. (1985). Input-output parametric models for non-linear systems part i: deterministic non-linear systems. *International Journal of Control*, 41(2), 303–328.

MacKay, D.J.C. (1995). Probable networks and plausible predicitons - a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6, 469–505.

Mao, K.Z. and Billings, S.A. (1997). Algorithms for minimal model structure detection in nonlinear dynamic system identification. *International Journal of Control*, 68(2), 311–330.

Ninness, B. and Henriksen, S. (2010). Bayesian system identification via markov chain monte carlo techniques. *Automatica*, 46(1), 40–51.

Piroddi, L. and Spinelli, W. (2003). An identification algorithm for polynomial NARX models based on simulation error minimization. *International Journal of Control*, 76(17), 1767–1781.

Tipping, M.E. (2001). Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 1, 211–244.