Addendum to Chapter 3 'Model Structure Detection and Parameter Estimation'

3.3.1.2 Variants of the FROLS Algorithm

The Latest Iterative FROLS Algorithm

3.3.1.2.1. Introduction

The FROLS algorithm, which is also known as the OLS (Orthogonal Least Squares) or the OFR (Orthogonal Forward Regression) algorithm, determines the model structure of nonlinear systems based on the ERR (Error Reduction Ratio) criterion without any a priori knowledge given an initial model set. Although the classic algorithm is always very efficient, OLS can sometimes produce a suboptimal solution under some extreme circumstances (Billings & Wei, 2007; Mao & Billings, 1997; Piroddi & Spinelli, 2003; Sherstinsky & Picard, 1996). This problem is most noticeable when the systems are not persistently excited. An iterative forward regression orthogonal least squares (iFROLS） algorithm has recently been proposed to improve the suboptimal problem where a small modification to the term selection procedure has been made to significantly improve the classic OLS algorithm without any significant increase in computational cost (Y. Guo, L. Z. Guo, S. A. Billings, & H.-L. Wei, 2015a; Y. Guo, L. Z. Guo, S. A. Billings, & H. L. Wei, 2015b). Contrary to the earlier approaches (Billings, 2013; Billings & Wei, 2007; Li, Peng, & Bai, 2006; Mao & Billings, 1997; Wei & Billings, 2008), the iFROLS enhances the classic OLS under a revised but purely OLS-ERR framework without making any major conceptual changes.

3.3.1.2.2. Iterative forward regression orthogonal least squares algorithm

The iFROLS algorithm comprises two steps. At the first step, classic OLS is used to produce an initial suboptimal model set. An additional iteration of the OLS algorithm is then applied in the second step using a subset of the terms from the first suboptimal model set as the starting point of a global search. The iFROLS algorithm is summarised in the following steps.

Step 1: Preset a small tolerance ρ (as defined in Section 3.3.1.1) and apply the standard OLS algorithm on the whole term dictionary $D=\{\mathbf{p}_1,\mathbf{p}_2,\cdots,\mathbf{p}_M\}$ to produce a model consisting of suboptimal term set $P_s=\left\{\mathbf{p}_{s_1} \quad \mathbf{p}_{s_2} \quad \cdots \quad \mathbf{p}_{s_{M_0}}\right\}$;

Step 2: Adjust the tolerance ρ by a small amendment $\Delta\rho$ as the new tolerance $\rho-\Delta\rho$;

Step 3: Select a subset P_{pre} of p terms in $\left\{p_j\right\}$, where $j=s_1,s_2,\cdots,s_{M_0}$, that is, $P_{pre}\subset P_s$, as the first p terms in a model and calculate the ERR's for each term in P_{pre};

Step 4: Search the remaining terms in the model using the OLS algorithm on the dictionary excluding the terms in P_{pre} (that is, the set $D\setminus P_{pre}$), as the $(p+1)th$, $(p+2)th$, … term in the model;

Step 5: Stop the search when the model meets the stop criterion $1-\sum ERR_i<\rho-\Delta\rho$

Step 6: Repeat Step 3-5 for different subsets P_{pre} 's of P_s and obtain a collection of suboptimal models;

Step 7: Compare the obtained suboptimal models and choose the best one as the final model P_{op}.

For the NARMAX model with noise terms, an additional process can be added to the algorithm for the identification of the noise model as shown in the reference (Guo et al., 2015a).

Remarks:

The subset P_{pre} is selected as a combination of p terms in P_s. There are a total number of $\binom{M_0}{p}$ combinations. All the combinations are evaluated in Step 3-6 and $\binom{M_0}{p}$ candidate models are produced and compared to obtain the final model. In most cases, choosing $p=1$ or 2 is sufficient to obtain the optimal model.

3.3.1.2.3. Test examples

The iFROLS algorithm is tested using two benchmark examples. While the literature is full of examples where the classical OLS algorithm works extremely well, each example below has been deliberately chosen from the small number of past results where the standard OLS has been shown to give non-ideal results.

A3.1. Example 1

This example is taken from (Mao & Billings, 1997). System (3.a.1) has been widely used as a benchmark example for the study of variations of OLS algorithms and for comparisons of OLS with other algorithms (Baldacchino, Anderson, & Kadirkamanathan, 2013). This example shows that the iFROLS can produce an optimum solution even when some correct terms are not selected in the first OLS step.

Consider the system

$$\begin{aligned} y(k) &= 0.2y^3(k-1) + 0.7y(k-1)u(k-1) + 0.6u^2(k-2) - 0.5y(k-2) \\ &\quad - 0.7y(k-2)u^2(k-2) + e(k) \end{aligned} \tag{3.a.1}$$

The system is excited with a uniformly distributed white noise u(k) ~ U(-1,1) and the output y(k) is disturbed by a normally distributed white noise e(k) ~ N(0,0.12). A total number of 1000 input and output data were used for the system identification.

Up to third order polynomials of the delayed inputs and outputs {y(k-1), y(k-2), y(k-3), y(k-4), u(k-1), u(k-2), u(k-3)} were used to model the nonlinear system. A total number of 120 terms were included in the term dictionary D. Applying the OLS algorithm yields a five term model which is shown in Table 1.

Table 1 Results produced by the standard OFR algorithm for system (3.a.1)

No.	Terms	ERRs	Parameter	Standard Deviation
1	y(k-4)u2(k-2)	36.2732	0.2922	0.02602
2	**y(k-1)u(k-1)**	13.7147	0.6544	0.01528
3	**u2(k-2)**	11.3488	0.5134	0.009331
4	**y(k-2)**	26.8516	-0.6743	0.01165
5	**y3(k-1)**	3.3248	0.1949	0.009847
SERR	--	91.513	--	--

Notice that the model in Table 1 includes a redundant term y(k-4)u2(k-2) but misses a correct term y(k-2)u2(k-2). The nonlinear cross correlation model validation tests (Billings & Voon, 1986; Billings & Zhu, 1994) clearly show that the model is unacceptable as a sufficient model of system (3.a.1).

Take the incorrect model in Table 1 as the starting point and apply the iteration steps from Section 3.3.1.2.2. Use each term (p=1) in the previous model as the first term and employ the OLS algorithm to select the remaining terms until SERR satisfies a tolerance where $\Delta\rho$ =0. The new iFROLS produced five different models. The results are shown in Table 2. All the five models have the same number of terms. However, models 3 and 4 give a better SERR than the other three models. Compared with system(3.a.1), both model 3 and model 4 are composed of the correct terms and give an accurate representation of the original system. The missed term y(k-2)u2(k-2) in Table 1 has now been correctly selected into the final model by the iFROLS algorithm

Table 2 Results produced by iFROLS algorithm for system (3.a.1)

	Model 1			Model 2			Model 3		
No.	Terms	ERRs	Para.	Terms	ERRs	Para.	Terms	ERRs	Para.
1	y(k-4)u2(k-2)*	36.273	0.2922	**y(k-1)u(k-1)***	13.7511	0.6544	**u2(k-2)***	24.760	0.6004
2	**y(k-1)u(k-1)**	13.715	0.6544	y(k-4)u2(k-2)	36.2367	0.2922	**y(k-2)**	48.582	-0.5124
3	**u2(k-2)**	11.349	0.5134	**u2(k-2)**	11.3488	0.5134	**y(k-1)u(k-1)**	13.985	0.6828
4	**y(k-2)**	26.852	-0.6743	**y(k-2)**	26.8516	-0.6743	**y(k-2)u2(k-2)**	3.249	-0.6683
5	**y3(k-1)**	3.325	0.1949	**y3(k-1)**	3.3248	0.1949	**y3(k-1)**	3.445	0.1983
SERR	--	91.513	--	--	91.513	--	--	94.021	
	Model 4			Model 5					
No.	Terms	ERRs	Para.	Terms	ERRs	Para.			
1	**y(k-2)***	29.893	-0.5124	**y3(k-1)***	0.9922	0.1949			
2	**u2(k-2)**	43.449	0.6004	y(k-4)u2(k-2)	37.4213	0.2922			
3	**y(k-1)u(k-1)**	13.985	0.6828	**y(k-1)u(k-1)**	15.4601	0.6544			
4	**y(k-2)u2(k-2)**	3.249	-0.6683	**y(k-2)**	11.9572	-0.6743			
5	**y3(k-1)**	3.445	0.1983	**u2(k-2)**	25.6822	0.5134			
SERR	--	94.021	--	--	91.513	--			

* represents the term that is determined first.

A3.2. Example 2

This example is taken from Piroddi and Spinelli's paper with the same parameter settings (Piroddi & Spinelli, 2003). This example will be used to show that the iFROLS algorithm can correctly identify an optimal model even when the systems are not persistently excited.

The system is given as follows

$$\begin{cases} w(k) = u(k-1) + 0.5u(k-2) + 0.25u(k-1)u(k-2) - 0.3u^3(k-2) \\ y(k) = w(k) + \dfrac{1}{1-0.8z^{-1}} e(k) \end{cases} \qquad (3.a.2)$$

where *u* represents the input signal and *y* represents the observation of the output *w*. Both the input *u(k)* and the noise *e(k)* are Gaussian distributed white noise. It can be shown that the classic OLS algorithm can correctly select all the terms and produce an accurate model when the system is persistently excited. However, Piroddi and Spinelli argued that the classic OLS algorithm may incorrectly select autoregressive terms when the input signal is less rich in frequency components. Piroddi and Spinelli recommended an input which is generated by an AR process with two real poles between 0.75 and 0.9. Repeating Piroddi and Spinelli's simulation using an input signal which was generated by the following AR process.

$$u(k) = \frac{0.25}{1 - 1.6z^{-1} + 0.64z^{-2}} v(k) \qquad (3.a.3)$$

where *v(k)* is Gaussian noise *v(k) ~ N(0,1)*. The AR process has a repeat pole at 0.8 and the coefficient 0.25 is chosen to guarantee the input signal is at a reasonable level. Here the noise signal *e(k)* is a Gaussian distributed noise with a variance 0.02, that is, *e(k) ~ N(0,0.02)*. The results produced by the standard OLS algorithm are given in Table 3.

Table 3 Results produced by the standard OFR algorithm for example 2

No.	Terms	ERRs	Coefficients	Standard Deviation
1	y(k-1)	88.704	0.517832	0.01638
2	y(k-2)	3.539335	-0.02977	0.007505
3	**u^3(k-1)**	0.876307	-0.30076	0.002555
4	u^3(k-2)	4.292185	0.146574	0.00526
5	**u(k-1)**	0.633041	1.17602	0.02149
6	u^2(k-1)	1.456973	0.12567	0.003581
7	**u(k-2)**	0.059644	-0.38214	0.03271
SERR	--	99.56	--	--

Observe that two incorrect autoregressive terms were selected as correct model terms. A correct term u(k-1)u(k-2) was also missed in the identification. The iFROLS was employed to overcome the problem by searching the optimal solutions on different paths. Combinations of any two terms (p=2) in the model in Table 3 were selected as the pre-determined two terms and the remaining terms were selected in a model using the iteration process. In this example, a total number of $\binom{7}{2}=21$ models were obtained. The sum of the ERR's in the 21 models are shown in Fig 1, where the red line indicates the sum of ERR values produced by the real model. It can be observed that two of the 21 models give the maximum SERR value which is equal to the SERR produced by the correct model. The results show that both models with the maximum SERR value consist of all the correct terms in (3.a.2). That is, the optimal model has been found on two different search paths.

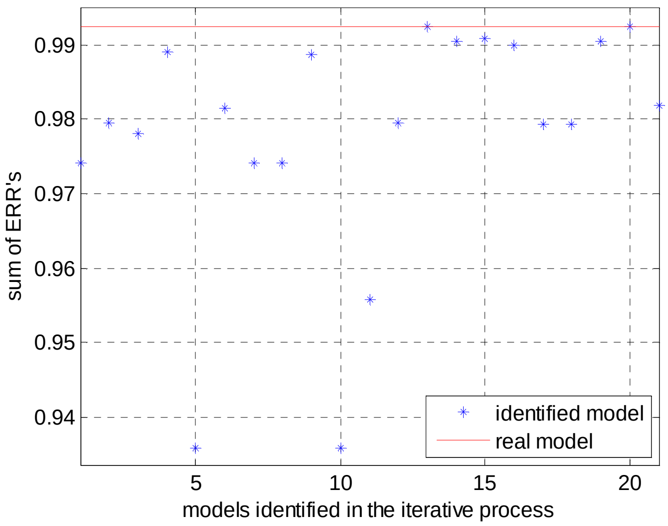

Fig 1 Sum of ERR's of terms in 21 identified model for example 1

The optimal model is given in Table 4.

Table 4 Model identified using the iFROLS algorithm for example 1

No.	Terms	ERRs	Coefficients	Standard Deviation
1	**u^3(k-1)**	63.77856	-0.30218	0.001013
2	**u(k-1)**	27.00761	1.04567	0.02279
3	**u (k-1)u(k-2)**	8.084573	0.23962	0.002311
4	**u(k-2)**	0.370287	0.485927	0.02202
SERR	--	99.15	--	--

In both examples, the iFROLS algorithm produced optimal models which include all the correct terms and are of the simplest structure in a very efficient computational process. The iFROLS algorithm worked well even when the first OLS step did not give a correct model as in the first example. Moreover, in both examples, the iFROLS algorithm found optimal solutions on more than one search path. This indicates that the new algorithm is significantly robust because the iFROLS can still produce an optimal solution even when the algorithm fails on one of the parallel search paths.

3.3.1.2.4. Conclusions

Several algorithms have been proposed to enhance the OLS algorithm by introducing complicated modified or add on algorithms, but the iFROLS algorithm improves OLS under a purely OLS-ERR framework. Very little extra programming is needed to implement the new algorithm which is also highly computationally efficient. The new iFROLS improves the classic OLS in two ways: it eliminates the redundant terms in a suboptimal model to produce a more parsimonious model, and selects the correct terms to obtain an accurate system description. Another advantage of the iFROLS algorithm is that the new two-stage process does not require the initial model obtained at the first step to be an accurate model, which is different from most coarse-to-fine algorithms. That is, the iOFR algorithm can start from an incomplete model and can still produce a complete optimal model. Other new developments to the OLS algorithm includes the ultra-orthogonal least squares algorithms (Guo, Guo, Billings, & Wei, 2016).

References

Baldacchino, T., Anderson, S. R., & Kadirkamanathan, V. (2013). Computational system identification for Bayesian NARMAX modelling. *Automatica, 49*(9), 2641-2651. doi:http://dx.doi.org/10.1016/j.automatica.2013.05.023

Billings, S. A. (2013). *Nonlinear System Identification : NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*. Hoboken, New Jersey: John Wiley & Sons Ltd.

Billings, S. A., & Voon, W. S. F. (1986). Correlation based model validity tests for nonliear models. *International Journal of Control, 44*(1), 235-244.

Billings, S. A., & Wei, H.-L. (2007). Sparse Model Identification Using a Forward Orthogonal Regression Algorithm Aided by Mutual Information. *Neural Networks, IEEE Transactions on, 18*(1), 306-310. doi:10.1109/tnn.2006.886356

Billings, S. A., & Zhu, Q. M. (1994). Nonlinear model validation using correlation tests. *International Journal of Control, 60*(6), 1107-1120.

Guo, Y., Guo, L. Z., Billings, S. A., & Wei, H.-L. (2015a). An iterative orthogonal forward regression algorithm. *International Journal of Systems Science, 46*(5), 776-789. doi:10.1080/00207721.2014.981237

Guo, Y., Guo, L. Z., Billings, S. A., & Wei, H.-L. (2016). Ultra-Orthogonal Forward Regression Algorithms for the Identification of Non-Linear Dynamic Systems. *Neurocomputing, 173, Part 3*, 715-723. doi:http://dx.doi.org/10.1016/j.neucom.2015.08.022

Guo, Y., Guo, L. Z., Billings, S. A., & Wei, H. L. (2015b). Identification of non-linear systems with non-persistent excitation using an iterative forward orthogonal least squares regression algorithm. *International Journal of Modelling, Identification and Control, 23*(1), 1-7.

Li, K., Peng, J.-X., & Bai, E.-W. (2006). A two-stage algorithm for identification of nonlinear dynamic systems. *Automatica, 42*(7), 1189-1197. doi:http://dx.doi.org/10.1016/j.automatica.2006.03.004

Mao, K. Z., & Billings, S. A. (1997). Algorithms for minimal model structure detection in nonlinear dynamic system identification. *Int. J. Control, 68*(2), 311-330. doi:10.1080/002071797223631

Piroddi, L., & Spinelli, W. (2003). An identification algorithm for polynomial NARX models based on simulation error minimization. *Int. J. Control, 76*(17), 1767-1781. doi:10.1080/00207170310001635419

Sherstinsky, A., & Picard, R. W. (1996). On the efficiency of the orthogonal least squares training method for radial basis function networks. *Trans. Neur. Netw., 7*(1), 195-200. doi:10.1109/72.478404

Wei, H.-L., & Billings, S. A. (2008). Model structure selection using an integrated forward orthogonal search algorithm assisted by squared correlation and mutual information. *International Journal of Modelling, Identification and Control, 3*(4), 341-356. doi:10.1504/ijmic.2008.020543