

Modified On-Line RLS Identification for Condition Monitoring †

Dr. Mazin Z. Othman¹, Shaima B. Ayoob²

^{1,2} Computer and Information Engineering Department, Electronic Engineering College, University of Mosul

e-mail: dr.mazin60@gmail.com, shaimabasil@yahoo.com

Received: 8/06/2014

Accepted: 11/09/2014

Abstract – The Recursive Least Squares (RLS) is usually utilized in control applications as in self-tuning strategy to estimate the plant discrete-time transfer function. Furthermore, it can be used as a tool to continuously monitoring the operating condition of the plant under control. However, in such applications, the RLS should be always in a “wake up” state so that it can estimate, in a few sampling time, the plant transfer function after any abrupt change in its dynamic.

In this work, two modifications to the standard RLS are presented. The first modification is called the “switching forgetting factor” while the other is called the “resetting covariance matrix”. The two modifications are applied, under LabVIEW environment, on-line to estimate the proper transfer function of a DC motor as an example to show their capabilities to monitor the motor operation. It is found that with these modifications, the RLS can estimate the plant transfer function much faster in comparison to the standard RLS algorithm.

Keywords – System identification, RLS, on-line condition monitoring.

† This paper has been presented in ECCCM-2 Conference and accredited for publication according to IJCCCE rules.

1. Introduction

The majority of industrial control systems exhibit changes in their operation with time or due to any environmental changes. The necessity of continuous on-line monitoring of their operating condition is of crucial importance. Therefore, it is required to have an approach that quickly discovers the unpredicted change in plant operation. One of the successful approaches in this field is to use the on-line system identification algorithms and in particular the Recursive Least Squares (RLS) algorithm. However, the standard RLS algorithm faces problems in dealing with sudden changes in plant dynamics.

Many researchers focused their work to overcome these problems. The RLS was modified in [1] and [2] to tackle noisy input-output data in adaptive filtering.

The modified RLS was proposed in [3] to track slow time-varying systems base on variable forgetting factor and resetting but keeping computational complexity of the standard RLS. In the field of self-tuning adaptive controllers, a modification to the RLS was introduced to improve the quality control. The so-called directional forgetting RLS was proposed in [4] which increase the complexity calculation required in the standard RLS.

The aim of this work is to modify the standard RLS algorithm taking into consideration the following points:

- No extra equations added to the standard RLS.
- Simple to implement on-line for system identification under LabVIEW environment.
- Can discover the plant dynamic changes in few sampling times.

2. The Standard RLS algorithm

For system autoregressive model with exogenous input of the form:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + e(k) \quad (1)$$

Where $y(k)$ and $u(k)$ are the output the input signals at k^{th} sampling time, respectively, $e(k)$ is a white Gaussian noise with variance σ^2 which is independent of $u(k)$, and are polynomial of time-varying system parameters.

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{n_a} z^{-n_a} \quad (2)$$

$$B(z^{-1}) = b_1 z^{-1} + \dots + b_{n_b} z^{-n_b} \quad (3)$$

Eq. (1) can be written as:

$$y(k) = x^T(k)\theta + e(k) \quad (4)$$

where $\theta(k) \in \mathbb{R}^{(n_a+n_b) \times 1}$ is a vector of unknown parameters defined by:

$$\theta(k)^T = [-a_1(k), \dots, -a_{n_a}(k), b_1(k), \dots, b_{n_b}(k)] \quad (5a)$$

and $x(k) \in \mathbb{R}^{(n_a+n_b) \times 1}$ is a regression vector partly consisting of measured input/output data as:

$$x^T(k) = [y(k-1), \dots, y(k-n_a), u(k-1), \dots, u(k-n_b)] \quad (5b)$$

The RLS algorithm is [5] :

At time step $(k+1)$

Form the vector $x(k+1)$ using the new data Evaluate the modeling error $\varepsilon \in \mathbb{R}^1$

$$\varepsilon(k+1) = y(k+1) - x^T(k+1)\theta(k) \quad (6)$$

Evaluate modeling error gain vector;

$$L \in \mathbb{R}^{(n_a+n_b) \times 1}$$

$$L(k+1) = \frac{P(k)x(k+1)}{\lambda + x^T(k+1)P(k)x(k+1)} \quad (7)$$

where λ is the forgetting factor.

Update $\Theta(k)$ to obtain $\Theta(k+1)$ by

$$\theta(k+1) = \theta(k) + L(k+1)\varepsilon(k+1) \quad (8)$$

Update the covariance matrix

$P(k) \in \mathbb{R}^{n_p \times n_p}$ to obtain $P(k+1)$ using

$$P(k+1) = \frac{1}{\lambda} [P(k) - L(k+1)x^T(k+1)P(k)] \quad (9)$$

Wait for the time step to elapse and loop back to step-1.

Note that:

$P(0)$ is usually set to be

$$P(0) = \mu I_{n_p \times n_p} \quad (\mu = 10 \text{ or } 100)$$

$\Theta(k)$ is usually initialized according to a prior knowledge about the system to be monitored.

Indeed, the choice of the forgetting factor plays an important role in the convergence of the RLS. Small values of the forgetting factor cause increase the ability of the RLS to track the variation in plant parameters, but make the algorithm sensitive to noise and vice versa. The other factor which is found to be an effective in increase or decrease the smoothness of the steady-state convergence to the true plant parameters is the covariance matrix. The trace of the covariance matrix reflects the RLS wakefulness and consequently discovers the plant parameter change as soon as it occurs.

3. The Modified RLS

The proposed modifications on the RLS relies on changing the forgetting factor or the covariance matrix each time an indication exist for an abrupt variation in plant parameters. In this work, it is found that the modeling error of Eq. (6) is a fast indicator which points out that something happens to the plant under monitoring.

3.1. The RLS with resetting covariance matrix

The first proposed modification to the RLS is to reset the covariance matrix according to the following condition:

$$P(k) = \begin{cases} \mu I & |\varepsilon| > \delta \\ \text{updated according to Eq. (9)} & |\varepsilon| \leq \delta \end{cases} \quad (10)$$

Here, δ is a modeling error threshold used to indicate that sudden changes happened to the plant parameters. Therefore, the covariance matrix, which is considered as an RLS memory that correlates the past input and output data, is reset to its starting condition in order to co-operate with the new condition immediately. The modeling error threshold δ should not be selected either too small so the RLS would be very sensitive to the noisy input/output data nor too big so the RLS would be in a deep sleep and not consider any changes in plant parameters. However, once the resetting process is activated, a predefined period ($N \cdot T_o$) is considered to avoid continuous covariance matrix resetting after an abrupt change is happened where T_o is the sampling time. It is found that suitable value for N is taken to be (50-100).

3.2. The RLS with switching forgetting factor Units

Use The second proposed modification to the RLS is to assign two values to the

forgetting factor according to the following condition:

$$\lambda(k) = \begin{cases} \beta & |\varepsilon| > \delta \\ 1 & |\varepsilon| \leq \delta \end{cases} \quad (11)$$

In Eq. (11), $0 < \beta < 1$ is a switching value to which the forgetting factor is switch to in the case of indicating an abrupt change in plant parameters.

It should not be selected either too small so the RLS would be very sensitive to the noisy input/output data nor too big so the RLS would be unable to track sudden and unexpected changes in plant parameters. After many simulation tests, it is found that β can have value of 0.8 or 0.9.

4. The Experimental Results

In order to illustrate the LabVIEW implementation of the on-line estimation and condition monitoring, a laboratory size DC motor [6] [7] is taken as a plant under consideration as shown in Fig. (1). Firstly, the DC motor is modeled as a first order discrete-time transfer function. Figure (2) reflects the LabVIEW implementation of Eqs. (6, 7, 8, and 9) for first order model. Figures (3) and (4) reflect the LabVIEW implementation of the RLS for first and second order DC motor models. The validation test, illustrated in Figures (5) and (6), indicates that the second order model is the best to represent the motor dynamics.

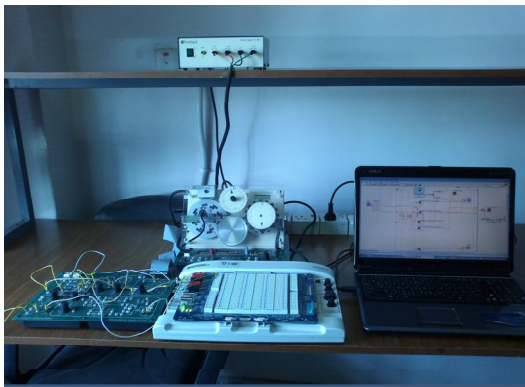
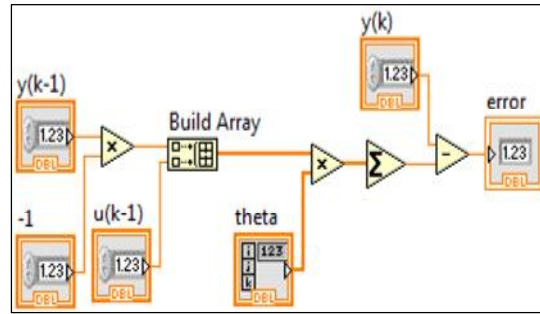
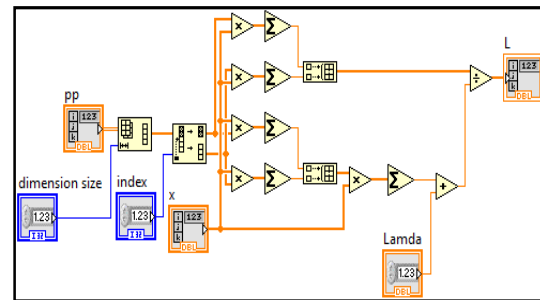


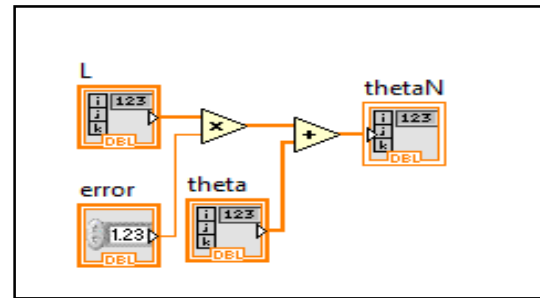
Fig. 1. Practical side of on-line condition monitoring for DC motor



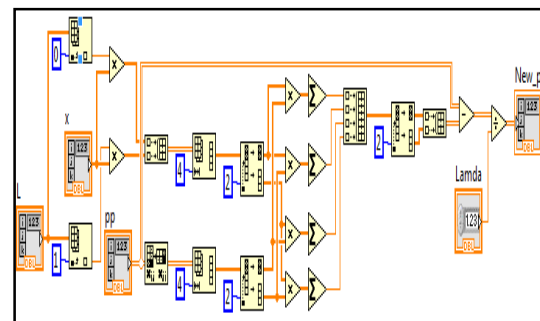
a) Calculating the error Eq. (6)



b) Calculating L(k) Eq. (7)



c) Update theta Eq. (8)



d) Update the covariance matrix Eq. (9)

Fig. 2. LabVIEW implementation of Equations (6-9)

Mazin Z. Othman and Shaima B. Ayoub.

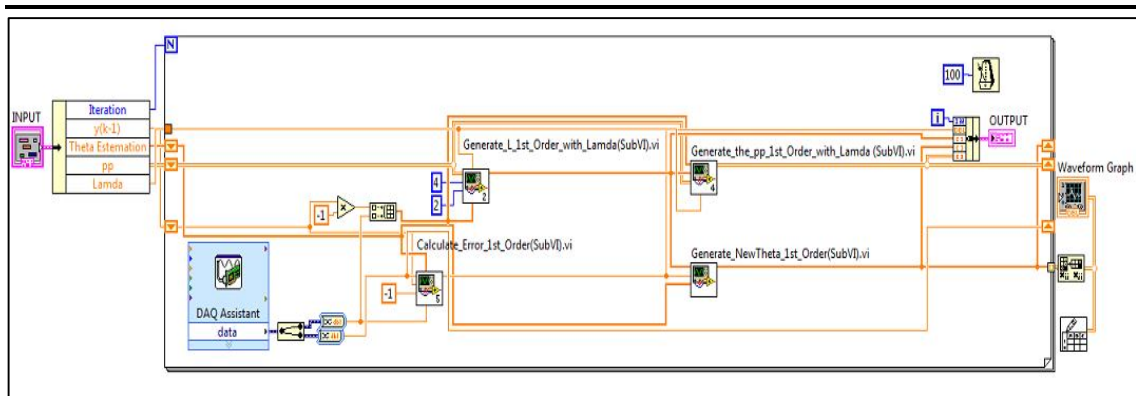


Fig. 3. LabVIEW implementation of DC motor 1st order model

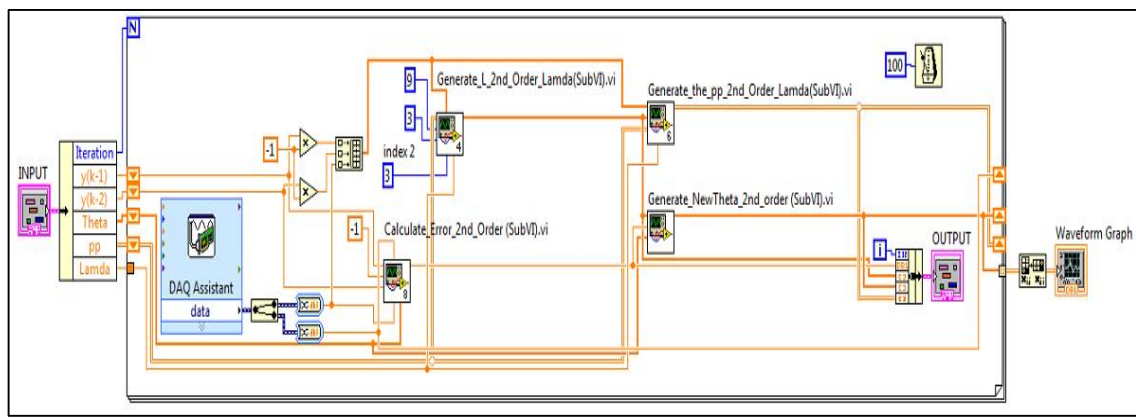


Fig. 4. LabVIEW implementation of DC motor 2nd order model

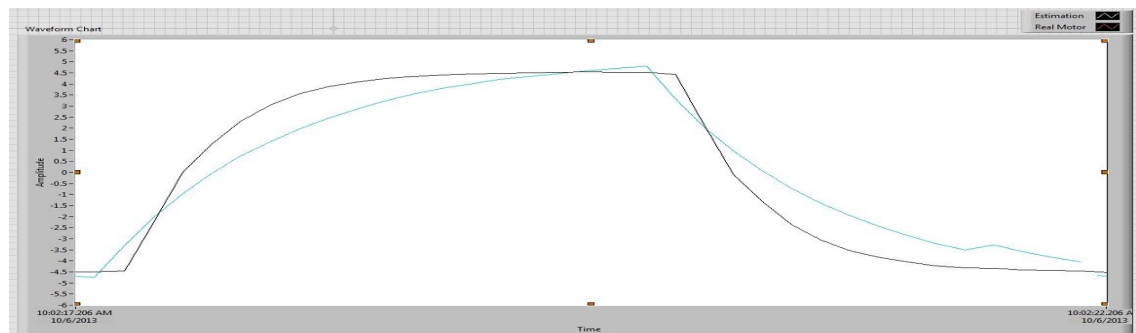


Fig. 5. The validation test for 1st order model

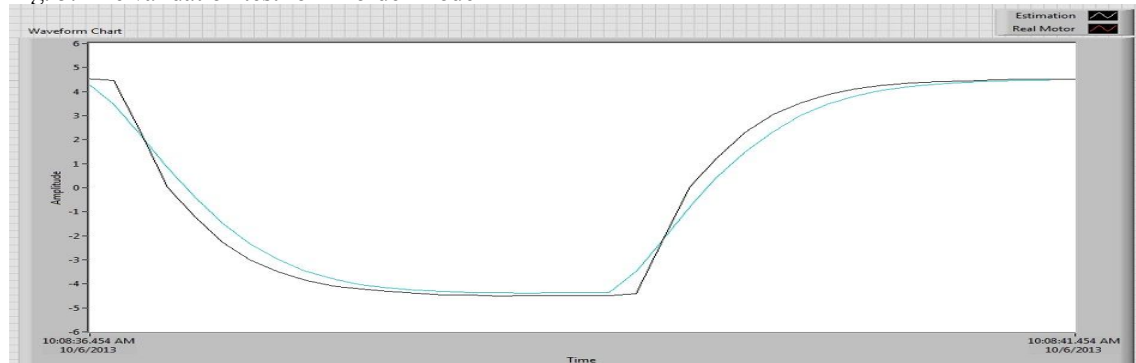
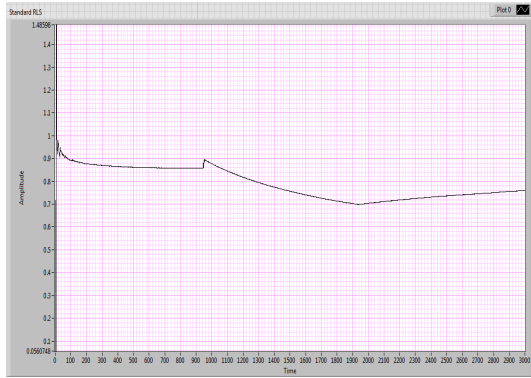
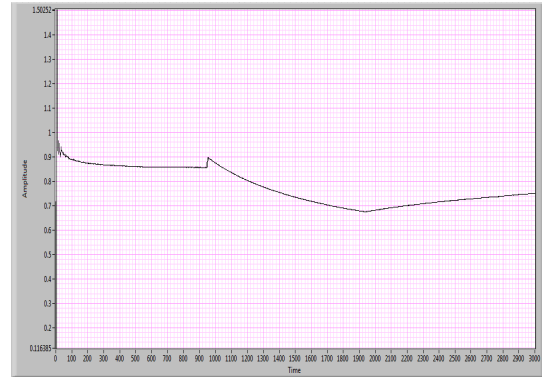


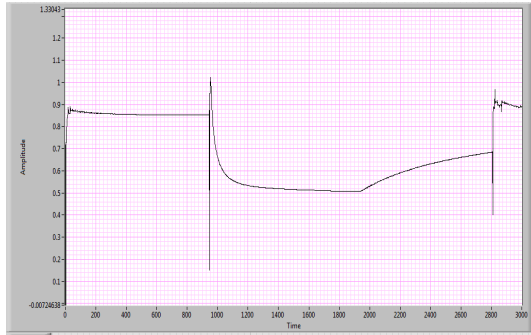
Fig. 6. The validation test for 2nd order model



(a) The standard RLS



b) The RLS with switching forgetting factor



(c) The RLS with resetting covariance matrix

Fig. 7. (a, b, c): On –line condition monitoring of DC-motor (trace of static gain)

The 1st and 2nd order models obtained on-line are:

$$G_{1st}(z) = \frac{0.150546z^{-1}}{1 + 0.853242z^{-1}} \quad (12)$$

$$G_{2nd}(z) = \frac{0.0895338z^{-1}}{1 - 1.4118z^{-1} + 0.517153z^{-2}} \quad (13)$$

In order to illustrate the benefits of the estimated parameters of Eq.(13) in the context of motor condition monitoring, the motor static gain is used as an indicator that the motor operating condition is changed. Here the value of the static gain indicated the no-load/load motor condition. The motor static gain is given by

$$G(0) = \lim_{z \rightarrow 1} G(z) \frac{b}{1+a_1+a_2} \quad (14)$$

In this work, $G(0)$ is monitored and traced while as shown in Figure (7), respectively. The motor was under on-line identification process using the standard RLS, the RLS with switching forgetting factor, and the RLS with resetting covariance matrix. Firstly, the motor was under no-load then at iteration number 1000 the motor was under full load. Afterward at iteration number 2000 the motor was turned back to no-load condition. It is clear that the RLS with resetting covariance matrix is powerful in identification of the DC motor static gain within few sampling time in comparison to the standard RLS and that with switching forgetting factor.

Table (1) shows a comparison study with Variable gain RLS (VRLS)

presented in [3] and with the RLS with directional forgetting presented in [4]. In comparison to the VRLS, the proposed modified RLS with resetting covariance matrix is more fast (less sampling time or iterations) and no flick around the estimated values if it faces an abrupt dynamic change. On the other hand, the RLS with directional forgetting which was presented in [4] suffer from complexity as it has to implement many equations within the RLS algorithm.

Table 1 Comparison between Different RLS algorithms

method	No. of iterations	Flicking	complexity
VRLS[3]	25-30	Yes	No
RLS with directional forgetting[4]	10-15	No	Yes
RLS with resetting covariance matrix (proposed)	10-15	No	No

5. Conclusions

This work presents two modifications to the standard RLS algorithm for on-line identification of discrete-time plant transfer function facing an abrupt change in its dynamics. Such identification is required to be fast and simple in order to be used further in plant condition monitoring. These modifications are the RLS with switching forgetting factor and the RLS with resetting covariance matrix. Both modifications ensure mostly the three points which are indicated in section (I). However it is found that the RLS with resetting covariance matrix perform fast (requires few sampling time) and its structure is simple. Therefore, it will be adopted next in future to be implemented with the FPGA.

References

- [1] D. Z. Feng, Z. Bao and X.D. Zhang, "Modified RLS algorithm for unbiased estimation of FIR system with input and output noise", *Electronic letters*, Vol.36, NO.3, pp 273-274, 3 Feb2000.
- [2] F. Ernst and A. Schweikard, "Prediction of motion using a modified recursive least squares algorithm", *bartz*, p175-160, 2008.
- [3] X. Yuncan and Q. Jinxin, "Modified Recursive Least Square algorithm with variable parameters and resetting for time-varying sytem", *Chinese J. Chem. eng.*, vol. 10, NO.(3) pp298-303, 2002.
- [4] V. Bobal, J. Böhm, J. Fessl and J. Mach'aček, "Digital Self-tuning Controllers Algorithms, Implementation and Applications", Springer, p318, 2005.
- [5] K. J. Astrom and B. Witten mark, "Computer controlled systems – theory and design", Prentice Hall, p.557, 1997.
- [6] FeedBack, control and instrumentation 33-100 and 33-110.
- [7] National Instrument NI ELVES 2, 2008.