FAST TRACKING RLS ALGORITHM USING NOVEL VARIABLE FORGETTING FACTOR WITH UNITY ZONE

Indexing terms: Algorithms, Time-varying systems, Adaptive systems and control

A new fast tracking recursive least squares (RLS) algorithm for time-varying systems is presented. The new algorithm is based on an innovative variable forgetting factor with a unity zone and its extra computational burden is trivial compared with the standard RLS algorithm. Fast tracking and low parameter error variance properties are verified via computer simulations.

Introduction: An important requirement of recursive estimators for adaptive control and adaptive signal processing lies in their ability to track parameter changes. From this viewpoint, the famous standard RLS algorithm which is known to have the optimal properties in stationary environments is unsuitable for nonstationary environments. Thus many attempts have been directed to the development of modified versions of the RLS algorithm to include tracking capability in time-varying environments.

Among these modified RLS algorithms, the best known is the famous standard RLS algorithm which is known to have the optimal properties in stationary environments.1-3 Also the lower the value of the forgetting factor, the higher the tracking velocity but the higher the influence of the noise, that is, the larger the parameter errors. To avoid these difficulties, the idea of a variable forgetting factor was introduced.

We present a new exponentially weighted RLS (EWRLS) algorithm using a novel variable of forgetting factor. The method presented has an excellent tracking adaptability with a low forgetting factor in the nonstationary situation and less error variance than other algorithms with a unity forgetting factor in stationary environments. The additional computational complexity is negligible compared to that of standard RLS by using simple operations.

New variable forgetting factor RLS algorithm: The EWRLS algorithm is given as

\[
\dot{\theta}(t) = \dot{\theta}(t-1) + k(t) \phi(t) \quad (1)
\]

\[
\dot{\theta}(t) = \frac{1}{\rho} \frac{1}{\lambda + \phi^T(t)P(t-1)\phi(t)} \quad (2)
\]

\[
k(t) = \frac{P(t-1)\phi(t)}{\lambda + \phi^T(t)P(t-1)\phi(t)} \quad (3)
\]

\[
P(t) = \frac{1}{\lambda} \left[ P(t-1) - k(t)\phi^T(t)P(t-1) \right] \quad P(0) > 0 \quad (4)
\]

where \( t \) is the time index in the discrete-time domain, \( \dot{\theta} \in \mathbb{R}^N \) is an unknown parameter vector to be estimated, where \( N \) is the number of the parameters, \( \phi \in \mathbb{R}^N \) is the input vector, \( y \in \mathbb{R} \) is the desired output signal, \( a \) is the \( a \ priori \) estimation error, \( P \in \mathbb{R}^{N \times N} \) is the covariance matrix and \( 0 < \lambda \leq 1 \) is the forgetting factor.

In the stationary case, we can estimate the parameters with a forgetting factor \( \lambda = 1 \). In the nonstationary case, we require \( \lambda \) to be small enough to estimate quickly the local trend of a nonstationary signal by using a finite number of recent available data. Therefore \( \lambda \) must be varied adequately to give a good tracking adaptability and low parameter error variance. If the forgetting factor is kept small when the parameters are changed abruptly, and is increased to unity appropriately so that the estimated parameter vector converges to the true value, then the algorithm has good tracking capabilities during the transient stage and fewer misadjustment errors of parameters in the steady state.

To implement this scheme, we propose a new strategy for choosing the variable forgetting factor as follows:

\[
A(t) = \lambda_{\text{max}}(1 - \lambda_{\text{min}}) \cdot 2^{100} \quad (5)
\]

\[
I(t) = -\text{NINT} \left[ \rho a^T(t) \right] \quad (6)
\]

where \( \text{NINT} \) is defined as the nearest integer to \( [ \cdot ] \), and \( \rho \) is a design parameter which controls the width of a unity zone, which will be discussed later. In eqn. 5, the minimum value of the forgetting factor is obtained when \( \alpha \) goes to infinity, and when \( \alpha \) decreases to zero the forgetting factor goes to unity at an exponential rate. This rate is controlled by the sensitivity gain \( \rho \). The additional computational burden with respect to the standard RLS algorithm is only a few arithmetic calculations and some bit shift operations in digital processors for the calculation of the Lth power of 2. Thus this new algorithm is numerically computationally efficient compared with other EWRLS algorithms. This is verified in the following computer simulations.

Computer simulations: In our simulations, we compare our algorithm with the algorithm of Fortescue et al.4 and the exponential resetting and forgetting (ERF) algorithm of Salgado et al.5 because they are known to have the best performances up to now. The system used for the simulations is given by

\[
y(t) = \theta^T(t)\phi(t) + w(t) \quad (7)
\]

where \( \phi \in \mathbb{R}^N \) is an input vector, \( w \) is random noise with zero mean and variance 0.01 and \( \theta \in \mathbb{R}^N \) is the time-varying parameter vector, which is changed from

\[
\theta^T = [0.1 \ 0.2 \ 0.3 \ 0.4 \ 0.5 \ 0.6 \ 0.7 \ 0.8 \ 0.9 \ 1.0] \quad 0 \quad 0 \quad 1 \quad 3 \quad 1 \quad 1 \quad -0.7 \quad 0.1 \quad 0.9 \quad 1 \quad 5 \quad -0.2 \quad -0.3] \quad (8)
\]

at iteration time 70. The parameters used in this simulation are

\[
\rho = 5 \quad \lambda_{\text{min}} = 0.7
\]

The results obtained by the ensemble average of 100 independent executions are presented in the Figures. Fig. 1 compares the norm of the parameter error, \( |\theta - \hat{\theta}(t)| \).

The convergence speed of the proposed algorithm is similar to the algorithm of Fortescue et al. but the parameter misadjustment error in the steady state is smaller than other algorithms. This is the major advantage of our variable forgetting factor. After the weight vector tracks parameter changes sufficiently, \( \lambda \) must go to unity quickly to reduce the steady state errors. In the algorithm of Fortescue et al., the forgetting factor stays near the value of 0.8 even in the situation of reducing parameter errors after convergence. However, in our proposed algorithm, the forgetting factor is switched quickly to unity in the parameter error reduction situation. This performance is accomplished by the presence of a unity zone in which the forgetting factor becomes unity due to
small errors. If the error is small enough to allow the weighted error square \( [w^*] \) is less than 0.5, the forgetting factor goes to unity due to a rounding operation of \( N/N^2 \). Thus, as the parameter estimator converges, the estimation output error is decreased and this reduced error drives the forgetting factor to go to unity by the effect of the unity zone. With this forgetting factor, our EWRLS algorithm becomes the standard RLS algorithm and has lower parameter misadjustment errors. This is the merit of the standard RLS algorithm. This unity region is controlled by the sensitivity gain \( p \). If \( p \) is small, the width of the unity zone becomes large, and convergence speed is reduced a little. The variable forgetting factor performance is shown in Fig. 2.

The performance improvement of our algorithm is also plotted in Fig. 3, which shows the mean square value of the estimation output error.

![Variable forgetting factor](image1)

![Mean square error](image2)

**Fig. 2 Variable forgetting factor**

- Proposed
- Fortescue et al.
- ERF

**Fig. 3 Mean square error**

- Proposed
- Fortescue et al.
- ERF

Conclusions: A fast tracking RLS algorithm using a novel variable forgetting factor with a unity zone is proposed. According to the computer simulation results, the algorithm has excellent tracking capabilities in the time-varying environments and lower parametric error variance than other algorithms in the time-invariant environments. The extra computational requirement is not complicated compared with the computational requirement of the standard RLS algorithm.

D. J. PARK
B. E. JUN
J. H. KIM

Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
373-1 Kuwon-dong, Yuseong-gu, Taejon 305-701, Korea

**References**


**IMPROVING DESIGN FEEDBACK EQUALISER PERFORMANCE USING NEURAL NETWORKS**

Indexing terms: Neural networks, Equalisers

Novel equaliser structures combining traditional transversal equalisers and neural computation have been introduced for adaptive discrete-signal detection. Extensive simulations using a two-path channel model and 16QAM modulation have been run to investigate the performance characteristics of these neural equalisers. The results show that they adapt very well to changing channel conditions, including both linear multipath and nonlinear distortions. The new structures are superior when compared to the traditional equalisers with equal computational complexity, especially in difficult channels.

**Introduction**: Adaptive equalisers in the form of linear or nonlinear transversal filters have traditionally been used in digital transmission to rectify the deterioration caused by dispersive transmission media. Typical application areas are fixed radio links and mobile radio services. In these cases the transmission path may cause, in addition to linear dispersion, other unwanted effects such as nonlinear distortions or phase shifts. The conventional equalisers do not respond particularly well to these other effects.

We have suggested a new equaliser structure employing neural computation. In this approach, a neural network algorithm, called a selforganising map (SOM) algorithm, was connected with a linear transversal equaliser (LE) or with a decision feedback equaliser (DFE). Patent applications were made regarding the new equalising schemes. The analyses concentrated on quadrature amplitude modulation (QAM), particularly on 4QAM and 16QAM because they have great interest in fixed and mobile radio services. 16QAM has also been proposed for digital HDTV transmission in the USA.

This Letter gives further results of computer simulations on the dynamic behaviour of the combined SOM-DFE equaliser with 16QAM. A two-path channel model having nonlinear distortion has been used. An enhanced performance can be achieved with a relatively minor increase in the receiver complexity. This may well provide practically useful receiver structures.

**Selforganising maps**: Selforganising maps are neural networks that produce localised responses to input signals and represent the topology of the input signal space over the network. In the adaptive detection method based on the SOM the learning framework in the algorithm is a planar array of adaptive cells, each cell corresponding to a particular gridpoint in the ideal discrete signal constellation. Each cell of