A Robust Variable Forgetting Factor Recursive Least-Squares Algorithm for System Identification

Constantin Paleologu, *Member, IEEE*, Jacob Benesty, *Senior Member, IEEE*, and Silviu Ciochină, *Member, IEEE*

Abstract—The performance of the recursive least-squares (RLS) algorithm is governed by the forgetting factor. This parameter leads to a compromise between 1) the tracking capabilities and 2) the misadjustment and stability. In this letter, a variable forgetting factor RLS (VFF-RLS) algorithm is proposed for system identification. In general, the output of the unknown system is corrupted by a noise-like signal. This signal should be recovered in the error signal of the adaptive filter after this one converges to the true solution. This condition is used to control the value of the forgetting factor. The simulation results indicate the good performance and the robustness of the proposed algorithm.

Index Terms—Adaptive filtering, echo cancellation, recursive least-squares (RLS).

I. INTRODUCTION

THE recursive least-squares (RLS) algorithm is one of the most popular adaptive filters [1], [2]. As compared to the least-mean-square (LMS) algorithm, the RLS offers a superior convergence rate, especially for highly correlated input signals. The price to pay for this is an increase in the computational complexity. The RLS algorithm belongs to the Kalman filters family [3], and many adaptive algorithms (including the LMS) can be seen as approximations of it.

The performance of RLS-type algorithms in terms of convergence rate, tracking, misadjustment, and stability depends on the forgetting factor. The classical RLS algorithm uses a constant forgetting factor $(0 < \lambda \leq 1)$ and needs to compromise between the previous performance criteria. When the forgetting factor is very close to one, the algorithm achieves low misadjustment and good stability, but its tracking capabilities are reduced. A smaller value of the forgetting factor improves the tracking but increases the misadjustment, and it could affect the stability of the algorithm. Motivated by these aspects, a number of variable forgetting factor RLS (VFF-RLS) algorithms have been developed (see [4], [5], and references therein). The performance and the applicability of these methods for system identification depend on several factors such as 1) the ability of detecting the changes of the system, 2) the level and the character of the noise that usually corrupts the output of the unknown system, and 3) complexity and stability issues.

C. Paleologu and S. Ciochină are with the Telecommunications Department, University Politehnica of Bucharest, Bucharest, Romania (e-mail: pale@comm.pub.ro; silviu@comm.pub.ro).

J. Benesty is with the Universite du Quebec, INRS-EMT, Montreal, QC H5A 1K6, Canada (e-mail: benesty@emt.inrs.ca).

It should be mentioned that in the system identification context, when the output of the unknown system is corrupted by another signal (which is usually an additive noise), the goal of the adaptive filter is not to make the error signal goes to zero, because this will introduce noise in the adaptive filter. The objective instead is to recover the "corrupting signal" in the error signal of the adaptive filter after this one converges to the true solution as explained in Sections II and III. Based on this condition, a new VFF-RLS algorithm is derived in this letter. The mechanism that controls the forgetting factor is very simple and not expensive in terms of complexity. Moreover, the proposed algorithm has good tracking capabilities, a stable behavior, and it is very robust against different variations of the "corrupting signal."

II. EFFECTS OF THE RLS FORGETTING FACTOR FOR SYSTEM IDENTIFICATION

In a system identification context, the desired signal for the adaptive filter is

$$d(n) = y(n) + v(n) = \mathbf{h}^T \mathbf{x}(n) + v(n) \qquad (1)$$

where $y(n) = \mathbf{h}^T \mathbf{x}(n)$ is the output of an unknown system defined by the vector $\mathbf{h}$ of length L, $\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-L+1)]^T$ is the input signal vector, and $v(n)$ is the system noise (which is assumed to be independent of the input signal). The superscript T denotes transposition.

Our objective is to identify the unknown system $\mathbf{h}$ using an adaptive filter defined by the vector $\hat{\mathbf{h}}(n)$ (it is assumed that the system and the adaptive filter have the same length). The adaptive filter is driven by the error signal

$$e(n) = d(n) - \hat{y}(n) = d(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n) \qquad (2)$$

where $\hat{y}(n) = \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n)$ is the output of the adaptive filter at time n, which is computed using the adaptive filter at time $n-1$, i.e., $\hat{\mathbf{h}}(n-1)$. In the presence of the system noise $v(n)$, the requirement of the application is not to make the error signal goes to zero, because this will introduce noise in the estimator $\hat{\mathbf{h}}(n)$. The correct goal is to extract the signal $v(n)$ from the mixture $y(n) + v(n)$, i.e., to recover $v(n)$ in the error signal $e(n)$.

In the case of the RLS algorithm, the normal equations are $\mathbf{\Phi}(n)\hat{\mathbf{h}}(n) = \boldsymbol{\theta}(n)$, where $\mathbf{\Phi}(n) = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i)$ and $\boldsymbol{\theta}(n) = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)d(i)$; the parameter λ is the forgetting factor of the algorithm. According to (1), the normal equations become

$$\sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i)\hat{\mathbf{h}}(n)$$

$$= \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)y(i) + \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)v(i). \qquad (3)$$

For a value of λ very close to one and for a value of n high enough, it may be assumed that

$$\frac{1}{n}\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)v(i) \cong E\left\{\mathbf{x}(n)v(n)\right\} = 0 \qquad (4)$$

where $E\{\cdot\}$ denotes mathematical expectation. Consequently, taking (3) into account

$$\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i)\hat{\mathbf{h}}(n) \cong \sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)y(i)$$
$$= \sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i)\mathbf{h} \quad (5)$$

thus $\hat{\mathbf{h}}(n) \cong \mathbf{h}$, and $e(n) \cong v(n)$. For a smaller value of the forgetting factor, so that $\lambda^k \ll 1$ for $k \geq n_0$, it can be assumed that $\sum_{i=1}^{n}\lambda^{n-i}\{\bullet\} \cong \sum_{i=n-n_0+1}^{n}\lambda^{n-i}\{\bullet\}$. According to the orthogonality theorem [1], the normal equations become $\sum_{i=n-n_0+1}^{n}\lambda^{n-i}\mathbf{x}(i)e(i) = \mathbf{0}_{L\times 1}$. This is a homogeneous set of L equations with n_0 unknown parameters, $e(i)$. In the case when $n_0 < L$, this set of equations has the unique solution $e(i) = 0$, for $i = n - n_0 + 1, \ldots, n$, leading to $\hat{y}(n) = y(n) + v(n)$. Consequently, there is a "leakage" of $v(n)$ into the output of the adaptive filter. In this situation, the signal $v(n)$ is cancelled; even if the error signal is $e(n) = 0$, this does not lead to a correct solution from the system identification point of view. A small value of λ or a high value of L intensifies this phenomenon.

Summarizing, for a low value of λ, the output of the adaptive system is $\hat{y}(n) \cong y(n) + v(n)$, while $\lambda \cong 1$ leads to $\hat{y}(n) \cong y(n)$. Apparently, for a system identification application, a value of λ very close to one is desired; but in this case, even if the initial convergence rate of the algorithm is satisfactory, the tracking capabilities suffer a lot. In order to provide fast tracking, a lower value of λ is desired. On the other hand, taking into account the previous aspects, a low value of λ is not proper in the steady-state. Consequently, a VFF-RLS algorithm (which could provide both fast tracking and low misadjustment) can be a more appropriate solution, in order to deal with these aspects.

III. PROPOSED VFF-RLS ALGORITHM

The following relations define the RLS algorithm:

$$e(n) = d(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n) \qquad (6)$$

$$\mathbf{k}(n) = \frac{\mathbf{P}(n-1)\mathbf{x}(n)}{\lambda + \mathbf{x}^T(n)\mathbf{P}(n-1)\mathbf{x}(n)} \qquad (7)$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mathbf{k}(n)e(n) \qquad (8)$$

$$\mathbf{P}(n) = \frac{1}{\lambda}\left[\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{x}^T(n)\mathbf{P}(n-1)\right] \qquad (9)$$

where $\mathbf{k}(n)$ is the Kalman gain vector and $\mathbf{P}(n)$ is the inverse of the input correlation matrix $\mathbf{\Phi}(n)$.

The parameter $e(n)$ from (6) is the *a priori* error, since it is computed using the adaptive filter at time $n-1$. The *a posteriori* error is defined as

$$\varepsilon(n) = d(n) - \hat{\mathbf{h}}^T(n)\mathbf{x}(n). \qquad (10)$$

Consequently, using (6) and (8) in (10), it results

$$\varepsilon(n) = e(n)\left[1 - \mathbf{x}^T(n)\mathbf{k}(n)\right]. \qquad (11)$$

According to the problem statement, it is desirable to recover the system noise in the error signal. Consequently, it can be imposed that $E\{\varepsilon^2(n)\} = \sigma_v^2$ [6], where $E\{v^2(n)\} = \sigma_v^2$ is the power of the system noise. Using the previous condition in (11) and taking (7) into account, it results

$$E\left\{\left[1 - \frac{q(n)}{\lambda(n) + q(n)}\right]^2\right\} = \frac{\sigma_v^2}{\sigma_e^2(n)} \qquad (12)$$

where $q(n) = \mathbf{x}^T(n)\mathbf{P}(n-1)\mathbf{x}(n)$ and $E\{e^2(n)\} = \sigma_e^2(n)$ is the power of the *a priori* error signal. In (12), we assumed that the input and error signals are uncorrelated, which is true when the adaptive filter has started to converge to the true solution. We also assume that the forgetting factor is deterministic and time dependent. By solving the quadratic equation (12), it results a variable forgetting factor

$$\lambda(n) = \frac{\sigma_q(n)\sigma_v}{\sigma_e(n) - \sigma_v} \qquad (13)$$

where $E\{q^2(n)\} = \sigma_q^2(n)$. In practice, the power estimates are computed using

$$\hat{\sigma}_e^2(n) = \alpha\hat{\sigma}_e^2(n-1) + (1-\alpha)e^2(n) \qquad (14)$$

$$\hat{\sigma}_q^2(n) = \alpha\hat{\sigma}_q^2(n-1) + (1-\alpha)q^2(n) \qquad (15)$$

where $\alpha = 1 - 1/(K_\alpha L)$ is a weighting factor, with $K_\alpha \geq 2$. From practical reasons, the power of the noise can be estimated from $e(n)$ using a longer exponential window [5], i.e.,

$$\hat{\sigma}_v^2(n) = \beta\hat{\sigma}_v^2(n-1) + (1-\beta)e^2(n) \qquad (16)$$

with $\beta = 1 - 1/(K_\beta L)$ and $K_\beta > K_\alpha$.

Theoretically, $\sigma_e(n) \geq \sigma_v$ in (13). As compared to the LMS algorithm (where there is the gradient noise, so that $\sigma_e(n) \geq \sigma_v$), an RLS algorithm with $\lambda(n) \cong 1$ leads to $\sigma_e(n) \cong \sigma_v$. In practice (since power estimates are used), several situations have to be prevented in (13). Apparently, when $\hat{\sigma}_e(n) \leq \hat{\sigma}_v(n)$, it could be set $\lambda(n) = \lambda_{\max}$, where $\lambda_{\max}$ is very close to or equal to one. However, this could be a limitation, because in the steady-state of the algorithm, $\hat{\sigma}_e(n)$ varies "around" $\hat{\sigma}_v(n)$. A more reasonable solution is to impose that $\lambda(n) = \lambda_{\max}$ when

$$\hat{\sigma}_e(n) \leq \gamma\hat{\sigma}_v(n) \qquad (17)$$

with $1 < \gamma \leq 2$. Otherwise, the forgetting factor of the proposed VFF-RLS algorithm is evaluated as

$$\lambda(n) = \min\left\{\frac{\hat{\sigma}_q(n)\hat{\sigma}_v(n)}{\xi + |\hat{\sigma}_e(n) - \hat{\sigma}_v(n)|}, \lambda_{\max}\right\}. \qquad (18)$$

The small positive constant ξ prevents the division by zero. Before the algorithm converges or when there is an abrupt change of the system, $\hat{\sigma}_e(n)$ is large as compared to $\hat{\sigma}_v(n)$; thus, the parameter $\lambda(n)$ from (18) goes to lower values, providing fast convergence and tracking. When the algorithm converges to the steady-state solution, $\hat{\sigma}_e(n) \cong \hat{\sigma}_v(n)$ [so that, condition (17) is fulfilled] and $\lambda(n)$ goes to $\lambda_{\max}$, providing low misadjustment.

IV. SIMULATION RESULTS

The performance of the proposed algorithm is evaluated in the context of echo cancellation (i.e., system identification). The "unknown" system is the first impulse response from ITU-T G168 Recommendation [7]; it has 64 coefficients. The same length is used for the adaptive filter. The output of the unknown system is corrupted by a white Gaussian noise
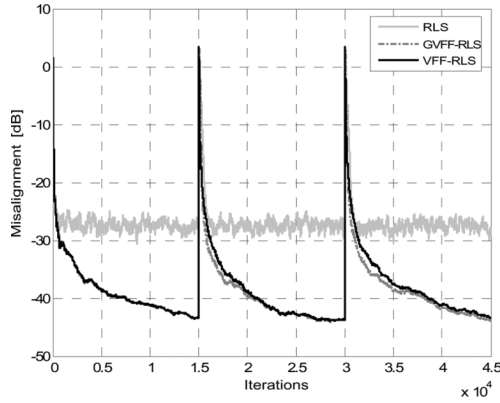
Fig. 1. Misalignment of the RLS algorithm using $\lambda = 1 - 1/(3L)$, with $L = 64$, and misalignment of the GVFF-RLS (with $\mu = 0.1$) and VFF-RLS algorithms. The impulse response changes at iterations 15 000 and 30 000. The input signal is a white Gaussian noise, $K_\alpha = 2$, $K_\beta = 5K_\alpha$, $\lambda_{\max} = 0.999999$, $\gamma = 1.5$, and SNR = 20 dB.

Fig. 2. Evolution in time of $\lambda(n)$ of the (a) GVFF-RLS algorithm and (b) VFF-RLS algorithm. Evolution in time of the condition number of the input signal covariance matrix of the (c) GVFF-RLS algorithm and (d) VFF-RLS algorithm. Other conditions are the same as in Fig. 1.

with 20 dB signal-to-noise ratio (SNR). Three types of input signal are used, i.e., 1) a white Gaussian noise, 2) an AR(1) process generated by filtering a white Gaussian noise through a first-order system $1/(1 - 0.9z^{-1})$, and 3) a speech signal. Two abrupt changes of the system are introduced at iterations 15 000 and 30 000, by shifting the impulse response to the right by four samples each time.

The proposed VFF-RLS algorithm is compared with the gradient-based VFF-RLS (GVFF-RLS) algorithm from [5], and with the classical RLS algorithm [using $\lambda = 1 - 1/(3L)$ when the input signal is white Gaussian noise or AR(1) process, and $\lambda = 1 - 1/(10L)$ for the speech input]. The GVFF-RLS algorithm uses a gradient-based control for the forgetting factor, according to the update

$$\lambda(n) = \left[\lambda(n-1) - \frac{\mu}{1 - \lambda(n-1)} \cdot \frac{\partial \hat{\sigma}_e^2(n)}{\partial \lambda} \right]_{\lambda_{\min}}^{\lambda_{\max}} \quad (19)$$

where μ is the step-size, $\lambda_{\min}$ is a lower bound limit that ensures the stability [see (33) from [5]], and $\lambda_{\max}$ is an upper bound limit desired in the steady-state. The gradient term $\partial \hat{\sigma}_e^2(n)/\partial \lambda$ is also evaluated in a recursive manner [see (40) from [5]]. In the experiments, the value of the step-size is set to $\mu = 0.1$. For a fair comparison, the upper bound limit of the forgetting factor for both VFF-RLS algorithms is $\lambda_{\max} = 0.999999$. Nevertheless, in the case of the proposed VFF-RLS algorithm, this limit can be chosen equal to one. The exponential windows use $K_\alpha = 2$ and $K_\beta = 5K_\alpha$ when the input signal is a white Gaussian noise or AR(1) process, and $K_\alpha = 6$ and $K_\beta = 3K_\alpha$ for the speech input. The proposed VFF-RLS algorithm uses $\gamma = 1.5$ and $\xi = 10^{-8}$. The measure of performance is the normalized misalignment (in dB), defined as $20\log_{10}(\|\mathbf{h} - \hat{\mathbf{h}}(n)\|_2/\|\mathbf{h}\|_2)$, where $\|\cdot\|_2$ denotes the l_2 norm. Also, both VFF-RLS algorithms are analyzed in terms of the condition number of the input signal covariance matrix. A recursive estimate of this parameter is evaluated according to the method proposed in [8]. This gives insightful information about the potential stability problem of the algorithms.

In the first case, the input signal is a white Gaussian noise (Figs. 1 and 2). It can be noticed that both VFF-RLS algorithms behave in a similar manner. As compared to the classical RLS
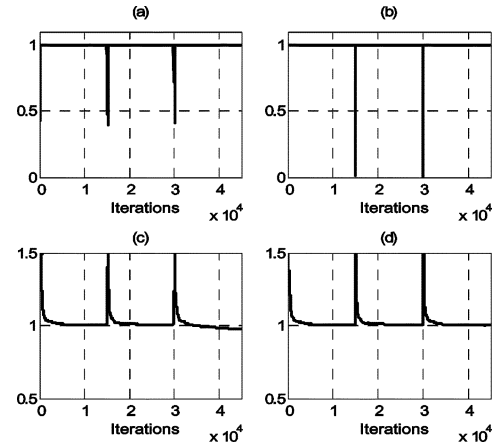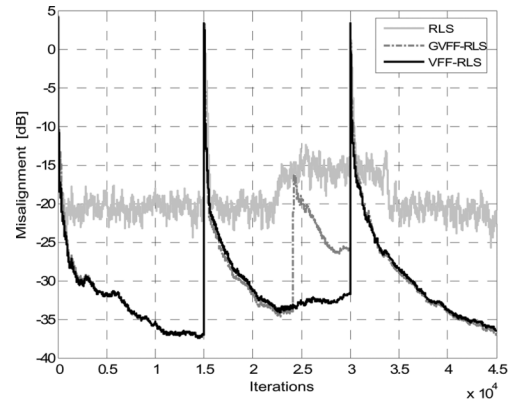

Fig. 3. Misalignment of the algorithms. The input signal is an AR(1) process. SNR decreases from 20 dB to 15 dB between iterations 22 500 and 33 750. Other conditions are the same as in Fig. 1.
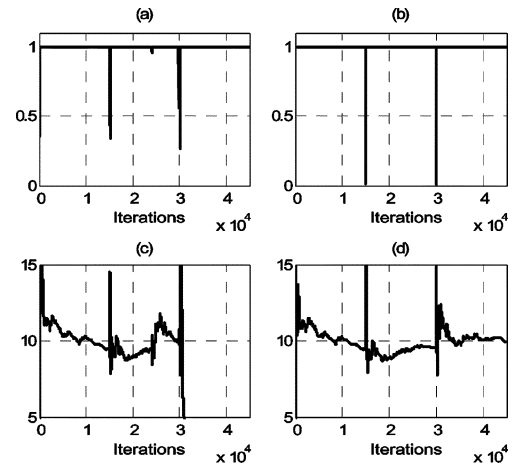

Fig. 4. Evolution in time of $\lambda(n)$ of the (a) GVFF-RLS algorithm and (b) VFF-RLS algorithm. Evolution in time of the condition number of the input signal covariance matrix of the (c) GVFF-RLS algorithm and (d) VFF-RLS algorithm. Other conditions are the same as in Fig. 3.

algorithm, they achieve a significant lower misalignment and similar tracking capabilities.

In the second set of simulations, the input signal is the AR(1) process (Figs. 3 and 4). A system noise increase is considered
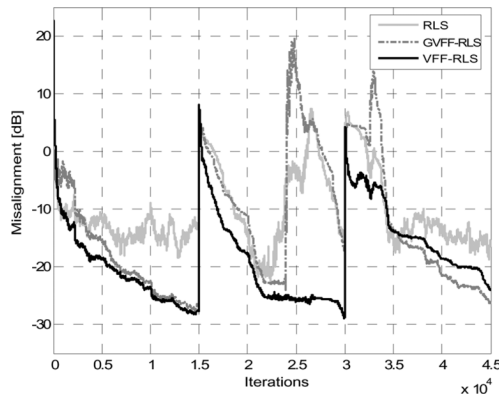
Fig. 5. Misalignment of the RLS algorithm using $\lambda = 1 - 1/(10L)$, with $L = 64$, and misalignment of the GVFF-RLS (with $\mu = 0.1$) and VFF-RLS algorithms. The input signal is speech, $K_\alpha = 6$, $K_\beta = 3K_\alpha$. A speech corrupts the output of the unknown system between iterations 22 500 and 33 750. The input signal-to-corrupting speech ratio is 15 dB. Other conditions are the same as in Fig. 1.
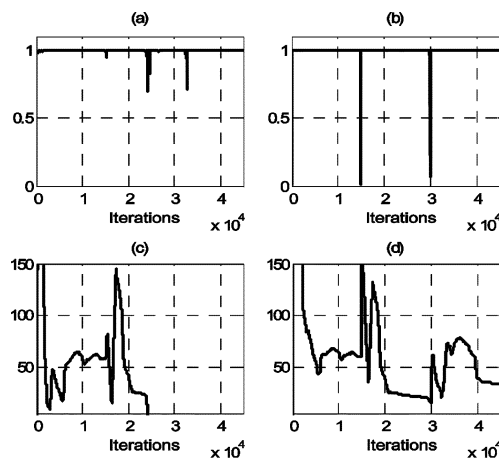

Fig. 6. Evolution in time of $\lambda(n)$ of the (a) GVFF-RLS algorithm and (b) VFF-RLS algorithm. Evolution in time of the condition number of the input signal covariance matrix of the (c) GVFF-RLS algorithm and (d) VFF-RLS algorithm. Other conditions are the same as in Fig. 5.

between iterations 22 500 and 33 750, when the SNR decreases from 20 dB to 15 dB. In terms of the final misalignment and tracking, both VFF-RLS algorithms achieve similar performances, but the proposed algorithm outperforms the other algorithms when the system noise increases. The GVFF-RLS algorithm does not make a clear distinction between an abrupt change of the system and an increase of the noise [see [5, eq. (40)]]; consequently, the value of the forgetting factor decreases in both situations [Fig. 4(a)]. Moreover, this variation of the noise could become a source of numerical instability. This can be noticed from the estimation of the condition number, which becomes unstable in the case of the GVFF-RLS algorithm [Fig. 4(c)]. The proposed VFF-RLS algorithm is more robust in this situation [as far as the condition (17) is fulfilled].

Finally, the results using the speech signal are presented in Figs. 5 and 6. A small amount of speech corrupts the output of the unknown system between iterations 22 500 and 33 750, such that the input signal-to-corrupting speech ratio is 15 dB.

This can be viewed as a mild double-talk situation. In terms of robustness, the proposed VFF-RLS algorithm outperforms the other algorithms. The estimation of the condition number becomes unstable in the case of the GVFF-RLS algorithm [Fig. 5(a)]. The analysis from [5] is based on the assumptions that 1) the input signal is a zero mean stationary Gaussian sequence and 2) the system noise is an independent and identically distributed Gaussian noise. These properties are always desirable for system identification, but some deviations from these theoretical conditions (like different input signals or system noise variations) are also possible in the context of different applications.

The proposed VFF-RLS algorithm obtains good performance (in terms of tracking capabilities and misadjustment) for both stationary and nonstationary input signals. In addition, the simulation results indicate that it is robust against different system noise variations. The value of the parameter γ in (17), together with the value of the parameters α and β (used for power estimates), control this behavior. The value of these parameters can be chosen according to the specific of application. The value of the parameter γ has to compromise between tracking capabilities and robustness against system noise variation. The parameters α and β are related to the character of the input signal; e.g., for nonstationary signals like speech, higher values of these parameter are desirable (as compared to the case of stationary inputs), in order to obtain smother power estimates.

V. CONCLUSIONS

In this letter, a new VFF-RLS algorithm is proposed in the context of system identification. Its derivation is based on the objective to recover the system noise in the error signal of the adaptive filter. The proposed algorithm is simple and easy to implement in practice. The simulation results indicate the good performance of the proposed algorithm for both stationary and nonstationary input signals. Besides, as far as the condition (17) is fulfilled (which can be controlled by the parameters γ, K_α, and K_β, according to the specific of the application), the algorithm is robust against different types of system noise variations, e.g., variations of the statistics of the noise or even the presence of a nonstationary signal, like speech.

REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[2] *Adaptive Signal Processing-Applications to Real-World Problems*, J. Benesty and Y. Huang, Eds. Berlin, Germany: Springer-Verlag, 2003.
[3] A. H. Sayed and T. Kailath, "A state-space approach to adaptive RLS filtering," *IEEE Signal Process. Mag.*, vol. 11, no. 3, pp. 18–60, Jul. 1994.
[4] S. Song, J. S. Lim, S. J. Baek, and K. M. Sung, "Gauss Newton variable forgetting factor recursive least squares for time varying parameter tracking," *Electron. Lett.*, vol. 36, no. 11, pp. 988–990, May 2000.
[5] S.-H. Leung and C. F. So, "Gradient-based variable forgetting factor RLS algorithm in time-varying environments," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3141–3150, Aug. 2005.
[6] J. Benesty, H. Rey, L. R. Vega, and S. Tressens, "A nonparametric VSS NLMS algorithm," *IEEE Signal Process. Lett.*, vol. 13, no. 10, pp. 581–584, Oct. 2006.
[7] *Digital Network Echo Cancellers*, ITU-T Rec. G.168, 2002.
[8] J. Benesty and T. Gänsler, "New insights into the RLS algorithm," *EURASIP J. Appl. Signal Process.*, vol. 3, pp. 331–339, 2004.