CHAPTER 2

NUMERIC VARIABLE FORGETTING-FACTOR RLS ALGORITHM FOR SECOND - ORDER VOLTERRA-FILTERING

2.1 Introduction

The development of nonlinear-filtering is motivated by the amount of published research and the widespread use of nonlinear digital filters. Many polynomial system models exist to represent the nonlinear-systems; all have their own advantages and limitations. Among them, the Volterra-system [3] is one of the most commonly used models because of its structural generality and the Weierstrass theorem, which states that any continuous real-valued function can be approximated by a polynomial function with an arbitrary small error. An adaptive-filter is a self-designing filter that uses a recursive algorithm (known as an adaptation algorithm or adaptive-filtering algorithm) to “design itself”. The algorithm starts from an initial guess based on the prior knowledge available to the system; it then refines the guess through successive iterations, which converges eventually to the optimal-solution in some statistical sense. In the literature, the Volterra-system is assumed to be time-invariant, which means that the Volterra-kernels do not change with time [44]. But, there are many time-varying communication channels, like mobile wireless channels and underwater acoustic channels, which need to be tracked or estimated by using the nonlinear polynomial adaptive-filtering. In this chapter, the nonlinear-channel paradigm is considered to be based on the second-order Volterra-series, which is used to describe the input-output relationship as

$$y(n) = h_0 + \sum_{k_1=0}^{M-1} h_1(n;k_1)x(n-k_1) + \sum_{k_1=0}^{M-1}\sum_{k_2=0}^{M-1} h_2(n;k_1,k_2)x(n-k_1)x(n-k_2) + e(n) \qquad (2.1)$$

with $h_1(n;k_1) = h_1(k_1;n)$ and $h_2(n;k_1,k_2) = h_2(k_1,k_2;n)$. where, h_0 is the constant zeroth-order Volterra-kernel, h_1, h_2 are the first-order and the second-order

Volterra-kernels respectively, M is the memory length, $x(n)$ is the input, and $e(n)$ is the measurement-noise with zero-mean and variance σ_e^2. The complexity of Volterra-filter is dependent upon the memory (M). In the general case, the degree of nonlinearity (K) of the Volterra-system is usually assumed to be time-invariant [89]; therefore the time-varying Volterra-system (TVVS) can be used to describe the general input-output relationship as

$$y(n)=h_0+\sum_{k=1}^{K}\sum_{m_1=0}^{M-1}\cdots\sum_{m_k=0}^{M-1}h_k(n;m_1,...,m_k)\prod_{i=1}^{k}x(n-m_i)+e(n) \tag{2.2}$$

where, the aim is to identify the time-varying (TV) Volterra-kernels $h_k(n;m_1,...,m_k)$ through measured $y(n)$ and $x(n)$. In our previous work [189], the channel estimation method is presented using the numeric variable forgetting-factor recursive least squares algorithm combined with the second-order polynomial time-varying channel model. When the signal experiences nonstationarity, the forgetting-factor decreases automatically to estimate the channel quickly using the extended estimation error criterion [158], [190]. However under stationary conditions, the forgetting-factor increases by increasing the memory for the accurate channel estimation. In this chapter, the NVFF-RLS algorithm is incorporated in the second-order Volterra-filter with K=2 for the nonlinear-channel estimation in a nonstationary environment.

2.2 System Paradigm

The nature of time-varying Volterra-kernel can be defined either by a stochastic-process or by a deterministic one. For the deterministic model, the periodic time-varying characteristics are reported to be appropriate. In this chapter, we choose a stochastic model for the time-varying Volterra-kernels. Since there is no knowledge about the stochastic behaviour of the TV kernels, one often finds it useful to model such a process by using the first-order Markov-model [33], [191] given by

$$\vec{h}(n+1)=\phi\vec{h}(n)+\vec{w}(n+1) \tag{2.3}$$

where, ϕ is the state transition-matrix and $\vec{w}(n)$ is the zero-mean white Gaussian process-noise vector with variance σ_w^2. After establishing the Gauss-Markov model for the time-varying kernels and the input-output relationship, the state-space equation of Volterra-system is represented by

$$\vec{h}(n) = \phi\vec{h}(n-1) + \vec{w}(n) \tag{2.4}$$

For the time-varying second-order Volterra-system as shown in Fig. 2.1, the input-output relationship is given by Eq. (2.2) with K = 2. Now, let us consider the $L \times 1$ dimensional extended filter-coefficients vector as

$$\begin{aligned}\vec{h}(n) = [&h_1(n;0), h_1(n;1), \ \ldots\ldots \ , h_1(n;M-1), h_2(n;0,0), h_2(n;0,1), \ \ldots\ldots, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ , h_2(n;0,M-1), h_2(n;1,1), \ \ldots\ldots \ , h_2(n;M-1,M-1)]^T\end{aligned}$$

where, $(.)^T$ is the matrix transpose operator. The $L \times 1$ dimensional extended input-signal vector for the SOVF with zero-mean and variance $\sigma_x^2 = 1/L$ is defined as

$$\begin{aligned}\vec{x}(n) = [&x(n), x(n-1), \ldots\ldots, x(n-M+1), \ldots\ldots, x^2(n), x(n)x(n-1), \ldots\ldots, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots x(n)x(n-M+1), \ldots\ldots, x^2(n-1), \ldots\ldots, x^2(n-M+1)]^T\end{aligned}$$

Further, we can rewrite Eq. (2.2) compactly as

$$y(n) = \vec{x}^T(n)\vec{h}(n) + e(n) \tag{2.5}$$

Utilizing the fact that the Volterra-kernels are symmetrical, the value of coefficient $h_k(n; m_1, \ldots, m_k)$ is kept unchanged for any of the possible k! permutations of $m_1, m_2, \ldots, m_k$. Therefore, the Volterra-kernel remains time-invariant under different permutations of its argument. Eqs. (2.4) and (2.5) represent the time-varying Volterra-system in terms of the stochastic dynamic-system, which are governed by the Gauss-Markov model. For the mathematical analysis, the estimated Volterra-kernel vector may be represented by

$$\vec{h}'(n) = [h_1'(n;0), h_1'(n;1), \ \ldots \ , h_1'(n;M-1), h_2'(n;0,0), h_2'(n;0,1), \ldots..,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ , h_2'(n;0,M-1), h_2'(n;1,1), \ \ldots \ , h_2'(n;M-1,M-1)]^T$$

Therefore, the estimated received signal vector is denoted by

$$y'(n) = \vec{x}^T(n)\vec{h}'(n) \tag{2.6}$$

Hence, the estimation-error in the signal reception is calculated by using $v(n) = y(n) - y'(n)$ with zero-mean and variance σ_v^2.

$$v(n) = \left(\vec{h}(n) - \vec{h}'(n)\right)^T \vec{x}(n) + e(n) \tag{2.7}$$

After updating the filter-coefficients by using Eq. (2.4), the estimated error in the received signal is

$$v(n) = \left(\phi\vec{h}(n-1) - \vec{w}(n) - \vec{h}'(n)\right)^T \vec{x}(n) + e(n) \tag{2.8}$$

Under optimum conditions, it may be assumed that the estimated Volterra-kernel is given by

$$\vec{h}'(n) \simeq \phi\vec{h}(n-1) \tag{2.9}$$

Therefore, the estimated error in received signal from Eq. (2.8) results in

$$v(n) \simeq \vec{w}(n)^T \vec{x}(n) + e(n) \tag{2.10}$$

where, the variance of estimation-error in the received signal may be calculated as

$$\sigma_v^2 = \sigma_e^2\left(1 + \frac{\sigma_w^2}{\sigma_e^2}\right) \tag{2.11}$$

It is clear from the above equation that if $\frac{\sigma_w^2}{\sigma_e^2} << 1$, then $\sigma_v^2 \simeq \sigma_e^2$.

Therefore from Eq. (2.11), it is apparent that the noise introduced in signal reception due to the imperfect channel estimation is almost equivalent to the measurement-noise. Utilizing this property [189], the system-identification model based on the presented NVFF-RLS algorithm uses time-variable forgetting-factor λ (n) to update the channel state after each sample point, which is incorporated in the first-order and second-order Volterra-filter in the next section.

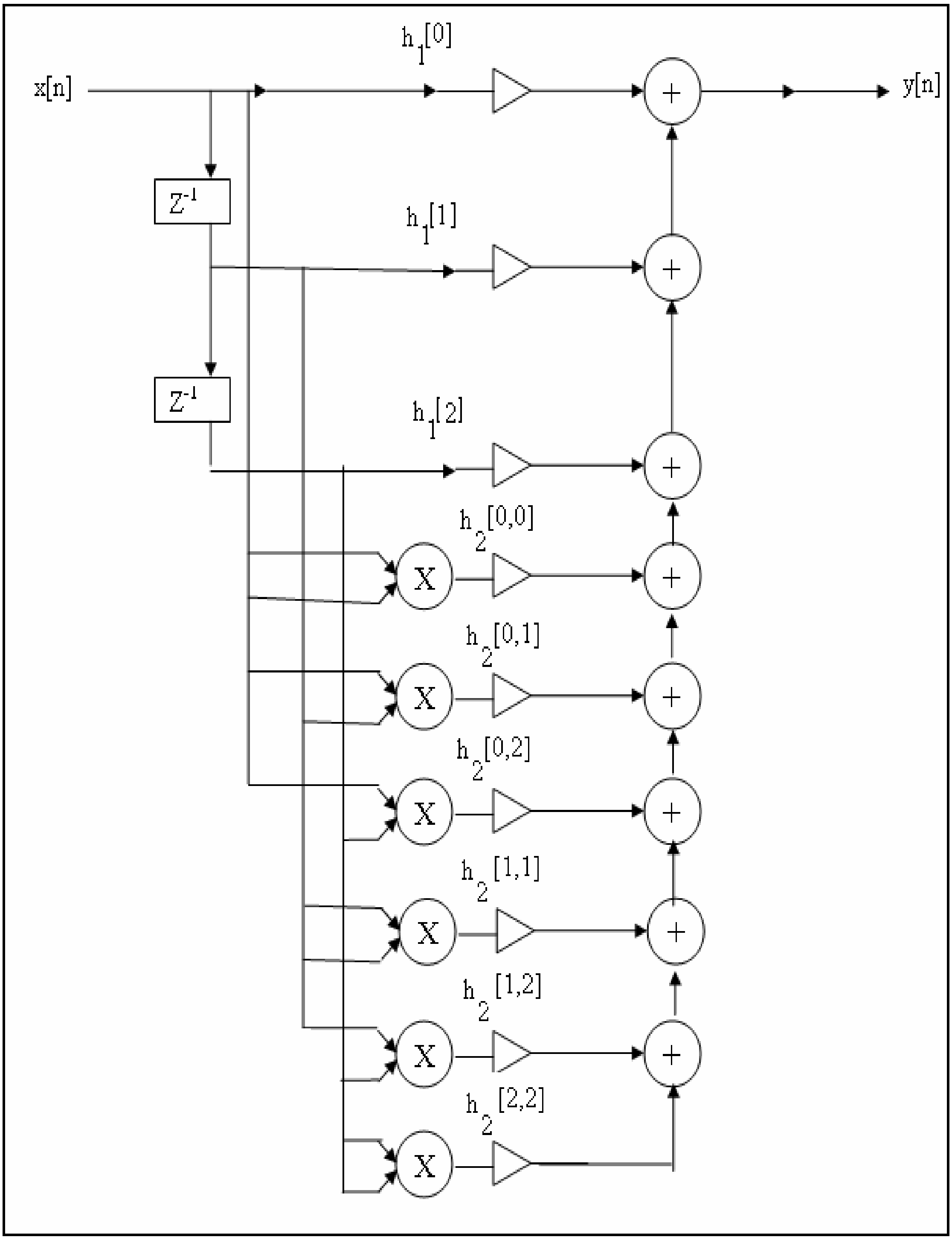

Fig. 2.1: Second-order Volterra-system with delay elements

2.3 Numeric Variable Forgetting-Factor RLS Algorithm

The technique presented in this chapter allows the variation of the forgetting-factor during the operation of RLS adaptive-filter structures. The prewindowed and the growing memory covariance algorithms in the transversal and the lattice structures were presented by Toplis and Pasupathy in [156]. Each of these algorithms simplifies to their original form, when the forgetting-factor is not undergoing any transition. The VFF algorithms allow the user to have greater flexibility in controlling the tradeoff between the lag-noise (lag-misadjustment [106]) and the channel estimation noise. This translates into improved tracking performance of the adaptive-filter operating in a nonstationary environment under the optimal parametric conditions [192].

However in the stationary case, the Volterra-kernels can be estimated with the fixed forgetting-factor with a high degree of accuracy. The estimation-error is observed to be large for the small value of the forgetting-factor, though the convergence-rate is fast due to the limited amount of available data. For the high value of the forgetting-factor, the estimation-error is less, though the convergence-rate is slow. The adaptation speed depends upon the asymptotic memory length [193] according to the following relation

$$N = 1/(1-\lambda) \tag{2.12}$$

Therefore, the forgetting-factor can also be calculated using the Eq. (2.12) as $\lambda = 1-(1/N)$. The memory lengths corresponding to $\lambda_{\max}$ and $\lambda_{\min}$ are denoted by $N_{\max}$ and $N_{\min}$ respectively. The polynomial channel model based NVFF-RLS algorithm uses forgetting-factor λ (n) to update the channel state after each sample point [189]. On the basis of the extended estimation-error criterion, the proposed algorithm for a second-order Volterra-filter is implemented as follows, such that the extended estimation-error is determined by

$$Z(n) = \frac{1}{Q}\sum_{q=0}^{Q-1} \left|v(n-q)\right|^2 \tag{2.13}$$

The value of Q must be kept smaller than the minimum asymptotic memory-length. If the process-noise σ_w^2 is very small in comparison to the measurement-noise σ_e^2, then it can be shown using Eq. (2.11) that $\sigma_v^2 \simeq \sigma_e^2$. Consequently, the extended estimation-error criterion [189] may be defined as

$$N(n) = \frac{\sigma_v^2 N_{\max}}{Z(n)} \simeq \frac{\sigma_e^2 N_{\max}}{Z(n)} \tag{2.14}$$

Therefore, the NVFF is calculated by using Eqs. (2.13) and (2.14) as

$$\lambda(n) = 1 - (N(n))^{-1} \tag{2.15}$$

In contrast to the stationary case, a smaller value of the forgetting-factor in the nonstationary environment seems to be more beneficial; which is bounded by $\lambda_{\min} < \lambda(n) < \lambda_{\max}$. Subsequently, we initialize the algorithm by setting

$$h'(0) = 0$$

$$\vec{P}(0) = \delta^{-1} I$$

$$\vec{\Pi}(n) = \vec{P}(n-1)\vec{x}(n)$$

$$\vec{k}(n) = \frac{\vec{\Pi}(n)}{\lambda(n)\sigma_e^2 + \vec{x}(n)^H \vec{\Pi}(n)} \tag{2.16}$$

$$\xi(n) = y(n) - \vec{h}'(n-1)^H \vec{x}(n)$$

$$\vec{h}'(n) = \vec{h}'(n-1) + \vec{k}(n)\xi(n)^* \tag{2.17}$$

$$\vec{P}(n) = \lambda^{-1}\vec{P}(n-1) - \lambda^{-1}\vec{k}(n)\vec{x}(n)^H \vec{P}(n-1) \tag{2.18}$$

where, $(.)^*$ and $(.)^H$ are the conjugate operator and the Hermitian transpose matrix operator. Eq. (2.17) provides the *a posteriori* estimate of $h(n)$. When the signal experiences nonstationarity, the value of the NVFF decreases to improve the tracking performance. But, the value of forgetting-factor can only vary between $\lambda_{\min}$ and $\lambda_{\max}$. The presented algorithm may also be incorporated easily in the higher order Volterra-filters to tackle the nonlinear parameter estimation.

2.4 Simulation Results

In the following simulations, we consider a second-order time-varying Volterra-system (TVVS), in which the tap-coefficients follow the Gauss-Markov model. The proposed system consists of the linear and quadratic-kernels for the first-order and second-order TVVS with memory length M = 3. We mainly investigate TVVS identification using LMS and RLS algorithms in a time-varying environment using MATLAB software. For the results presented in Fig.2.2 and Fig. 2.3, the state transition-matrix of the linear kernel is chosen to be

$$\phi = \begin{bmatrix} -1.31 & -0.43 & 2.25 \\ -0.88 & -0.53 & 2.03 \\ -0.53 & -1.43 & 2.65 \end{bmatrix} \tag{2.19}$$

Similarly, for the results presented in Fig. 2.4 and Fig. 2.5 in case of the quadratic-kernel, the state transition-matrix is chosen to be

$$\phi = \begin{bmatrix} 2.71 & -7.19 & -0.04 & 0.72 & -0.82 & 4.27 \\ 1.6 & -4.19 & -0.09 & 0.36 & -0.52 & 2.67 \\ -0.39 & 1.27 & 0.17 & -0.57 & -0.47 & 0.25 \\ 1.99 & -5.67 & -0.19 & 0.84 & -0.11 & 2.81 \\ 2.38 & -5.27 & -0.26 & 0.77 & -0.91 & 3.02 \\ 0.42 & -0.46 & -0.15 & -0.26 & -0.58 & 1.14 \end{bmatrix} \tag{2. 20}$$

However, Eqs. (2.19) and (2.20) make the state transition-matrix stable, ensuring the stationarity of $h(n)$ [169]. The presented results are based on the ensemble average of 2500 independent simulation runs for the different channel realizations. We assume averaging factor Q = 5, measurement-noise $\sigma_e^2 = 0.01$, and also consider the process-noise variance

$\sigma_w^2 << 0.01$. By invoking Eq. (2.11), it may be inferred that $\sigma_v^2 \simeq \sigma_e^2$. Therefore, we can incorporate Eq. (2.14) to calculate $N(n)$, which in turn leads to updating $\lambda(n)$ at every iteration. The performance of the adaptive channel estimators are compared on the basis of MMSE criterion, which is calculated by using the following formula

$$J(n) = E\left[\left|h(n) - h'(n)\right|^2\right] \tag{2.21}$$

where, $E(.)$ is the expectation or ensemble average operator.

$$J(n) = \frac{\sum_{j=1}^{2500}\left[\left|h(n,j) - h'(n,j)\right|^2\right]}{2500} \tag{2.22}$$

The performance of the RLS algorithm depends upon the convergence-rate, tracking, misadjustment and stability, which vary according to the value of forgetting-factor. In Fig. 2.2 and Fig. 2.4, the channel tracking performance of the first-order and second-order Volterra-systems using the NVFF-RLS algorithm is compared with the tracking performance of the conventional fixed forgetting-factor RLS algorithm and the Kalman-filtering algorithm. For this simulation, the values of $\lambda_{\max}$ and $\lambda_{\min}$ are chosen 0.975 and 0.75 respectively. For the simulation of the fixed forgetting-factor RLS algorithm, the value of λ is chosen to be 0.975 and the value of δ is fixed at 10^{-5}. The actual channel coefficient is denoted as True. These results show the lag-misadjustment between the conventional fixed forgetting-factor RLS and the proposed algorithm. It may be inferred from these simulation results that the channel tracking performance of the NVFF-RLS algorithm is better than that of the conventional RLS algorithm for the Volterra-systems, though both appear to be inferior to the Kalman-filtering algorithm based approach.

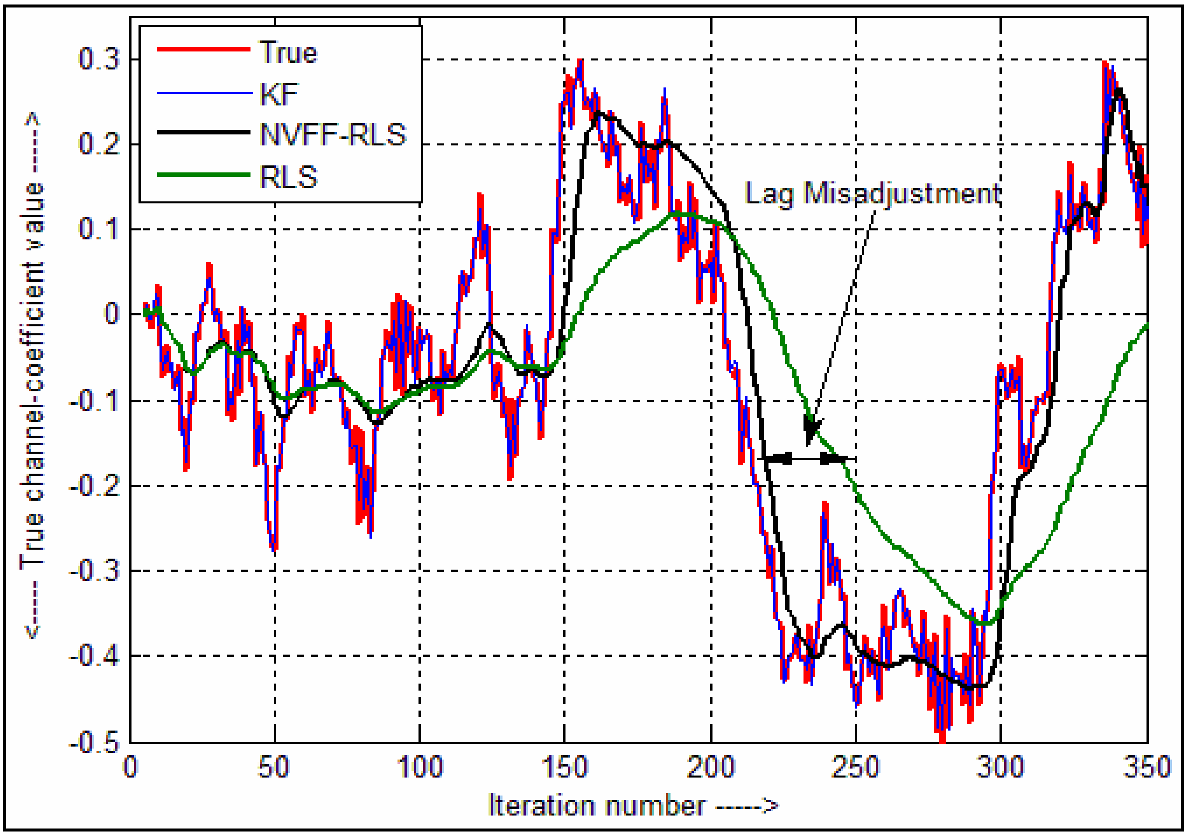

Fig. 2.2: Tracking performance of simulated algorithms for first-order Volterra-system

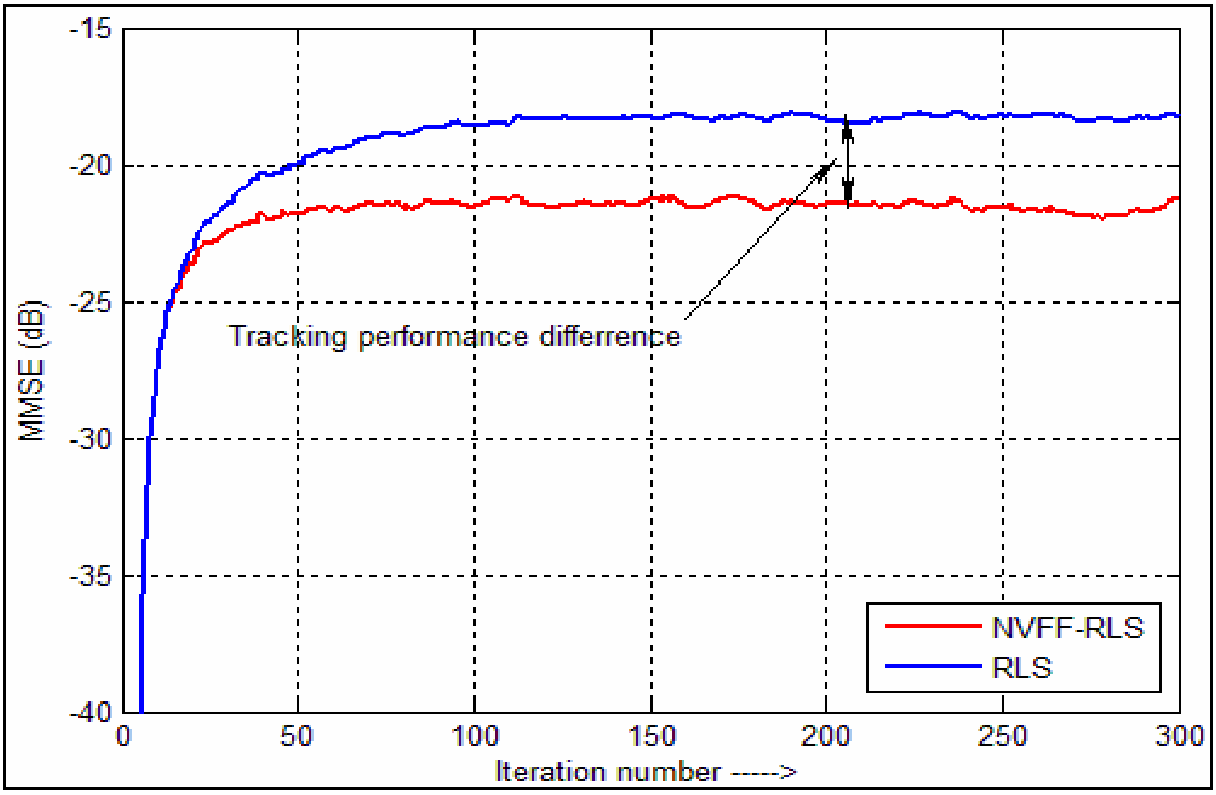

Fig. 2.3: MMSE in channel estimation for simulated algorithms for first-order Volterra-system

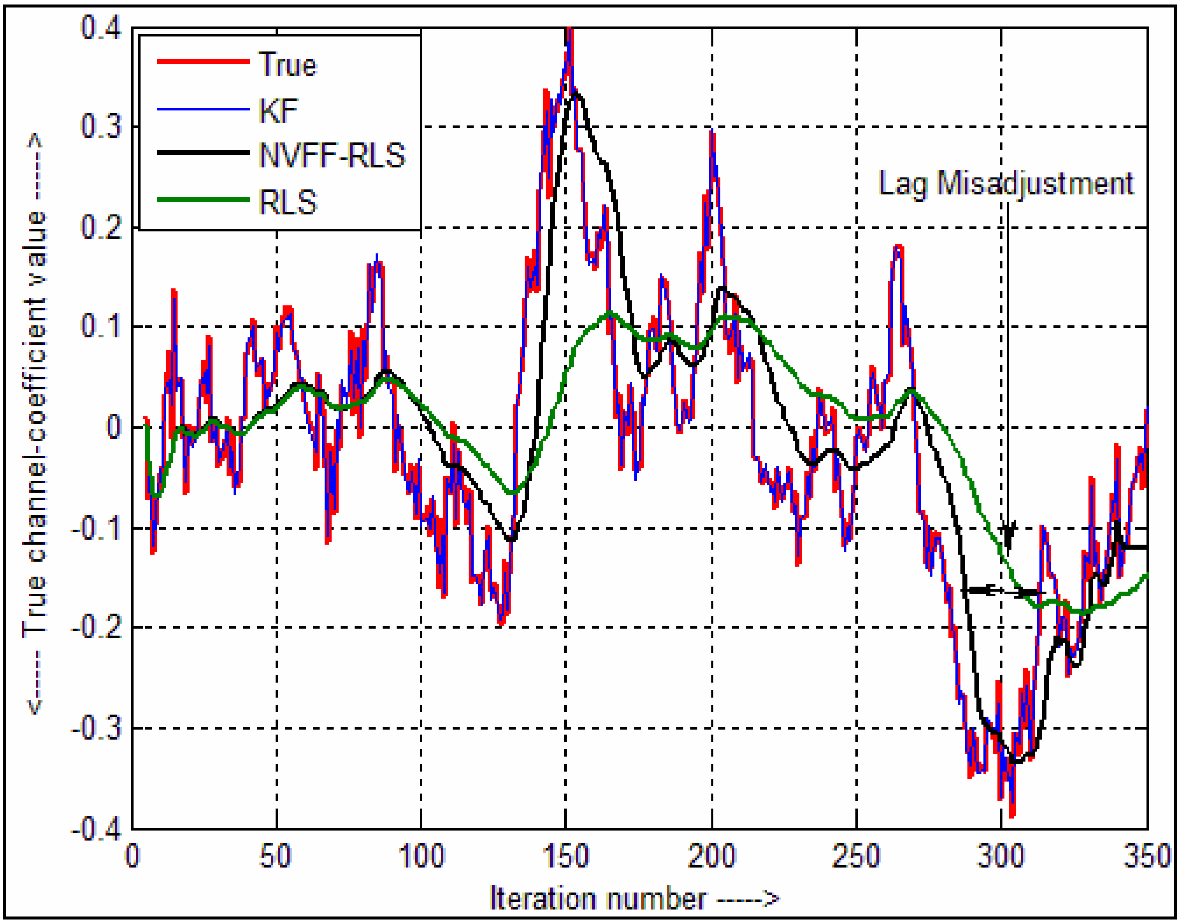

Fig. 2.4: Tracking performance of simulated algorithms for second-order Volterra-system

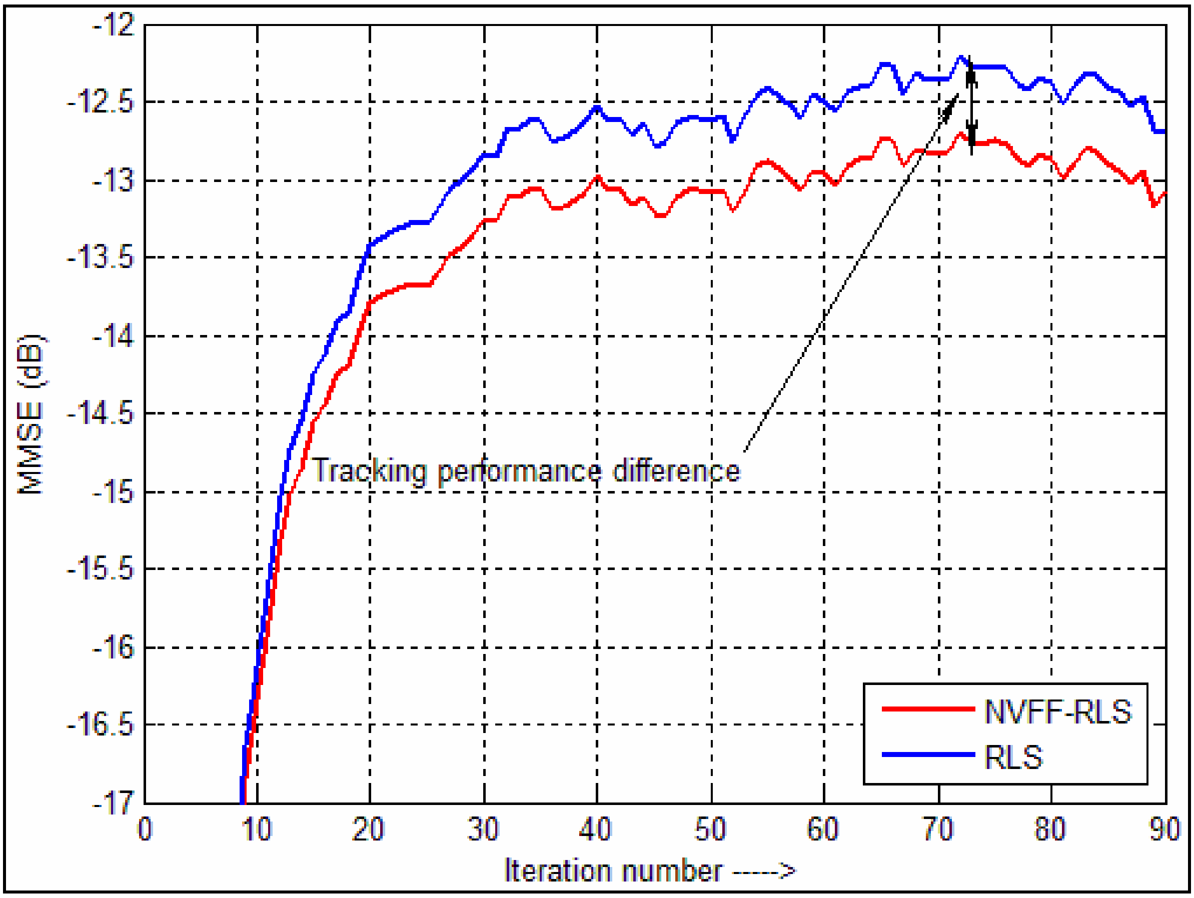

Fig. 2.5: MMSE in channel estimation for simulated algorithms for second-order Volterra-system

In Fig. 2.3 and Fig. 2.5, the MMSEs in the tracking of channel coefficients for the first-order and second-order Volterra-systems are presented respectively. Under similar time-varying environment, the NVFF-RLS algorithm performs approximately 2.5 dB better than the conventional RLS algorithm for the first-order Volterra-filtering system.

However, the tracking performance advantage is approximately 0.5 dB in the case of NVFF-RLS algorithm compared with the conventional RLS algorithm for the second-order Volterra-filtering system. This observed degradation in the performance of NVFF-RLS algorithm is due to the increased number of Volterra-kernels in the second-order systems. However, the tracking performance of the conventional RLS algorithm improves because the lag-misadjustment reduces for the second-order Volterra-filters, which is also evident from the results depicted in Fig. 2.3 and Fig. 2.5. Moreover, it is clear from the simulation results for the third-order Volterra-systems that the MMSE performance gap between the NVFF-RLS and conventional RLS algorithms is negligible.

In the above discussion, we have examined the simulations of adaptive algorithms for the first-order and second-order Volterra-systems. But with the increasing number of sources of nonlinearity, we have to look at certain adaptive algorithms or other refined adaptive algorithms that may be used to overcome the nonlinear effects. We can also apply these adaptive algorithms in the channel estimation for the latest wireless technologies like BLAST, 3G, 4G etc.

2.5 Summary of Chapter

The parameter estimation of time-varying first-order and second-order Volterra-systems is investigated using three different adaptive algorithms, namely the Kalman-filter, the fixed forgetting-factor conventional RLS algorithm, and the numeric variable forgetting-factor RLS algorithm. The Gauss-Markov model is used to define the time-variation of the kernels in a time-varying wireless environment. The parameter estimation or channel tracking performance of the NVFF-RLS algorithm supersedes the conventional fixed forgetting-factor RLS algorithm by 2.5 dB and 0.5 dB for the first-order and second-order nonlinear-systems respectively. The higher order Volterra-filters in combination with the NVFF-RLS algorithm may not prove very beneficial in tracking the time-varying wireless fading channels, but the presented nonlinear polynomial-filtering technique proves to be more computationally efficient than the available approaches of [151], [152], [153], [154], [157]. Future work includes the application of proposed adaptive polynomial-filters in the emerging fields of signal processing and communication engineering.