Time-varying parameter estimation with application to trajectory tracking

K. Bousson
Avionics and Control Laboratory, Department of Aerospace Sciences, University of Beira Interior, Covilhã, Portugal

Abstract
Purpose – This paper is concerned with an online parameter estimation algorithm for nonlinear uncertain time-varying systems for which no stochastic information is available.
Design/methodology/approach – The estimation procedure, called nonlinear learning rate adaptation (NLRA), computes an individual adaptive learning rate for each parameter instead of using a single adaptive learning rate for all the parameters as done in stochastic approximation, each individual learning rate being controlled by a meta-learning rate rule for the sake of minimizing the measurement prediction error. The method does not require stochastic information about the system model and the measurement noise covariance matrices contrarily to the Kalman filtering. Numerical results about aircraft navigation trajectory tracking show that the method is able to estimate reliably time-varying parameters even in presence of measurement noise.
Findings – The proposed algorithm is practically insensitive to changes in the meta-learning rate. Therefore, the performance of the method is stable with respect to the tuning parameter of the algorithm.
Practical implications – The proposed NLRA method may be adopted for recursive parameter estimation of uncertain systems when no stochastic information is available. It may also be used for process regulation and dynamic system stabilization in feedback control applications.
Originality/value – Provides a method for fast and practical computation of parameter estimates without requiring to know the model and measurement noise covariance matrices contrarily to existing stochastic estimation methods.

Keywords Programming and algorithm theory, Estimation, Adaptive system theory

Paper type Research paper

1. Introduction

Parameters of a dynamic system are said to be time-varying if these parameters may change across time instead of remaining constant as the system runs. Uncertain systems are those systems whose dynamics are not well known, and/or the measurements of their variables may be endowed with significant noise. Therefore, online parameter estimation (Bar-Shalom and Li, 1993) is required when dealing with uncertain systems having time-varying parameters in the context of supervision and control. In aerospace applications, for instance, aerodynamic parameter estimation needs to be processed for a given vehicle from flight test data (Bousson, 2002, 2001; Morelli and Klein, 1996; Linse and Stengel, 1993) for the sake of performance and stability analysis, and handling qualities assessment. The interest of estimating dynamical system parameters online is manifold: for instance, the supervision and control strategies may be improved by the possibility of using the estimated parameters to predict some state or output variables, adapting the control parameters according to these predictions, or estimating the performances of the system so that they can be enhanced efficiently.

There are two main groups of methods for coping with time-varying parameter estimation: the gradient descent (Fletcher, 2000; Bertsekas, 1997; Roll et al., 2005) and the Kalman filtering-based methods (Kalman, 1960; Kalman and Bucy, 1961; Aykan et al., 2005; Park and Kee, 2006). Nonlinear applications using gradient descent methods are limited because they require line search at each iteration, which is prohibitive when the system to be dealt with has a fast dynamics. Kalman filtering is known to be an optimal estimator in least-squares sense for linear systems. As to nonlinear systems, it is not optimal due to the linearization process. In Jang et al. (1986), it is proved that nonlinear optimization methods have better performance than the extended Kalman filtering for parameter and state estimation of nonlinear dynamic systems. Meanwhile, the work by Sutton (1992a, b) on linear networks sheds the light on the relationship between these two groups, and it can be shown that the problem of sequentially updating the learning rates in gradient descent algorithms and that of updating the process noise covariances in Kalman filtering are equivalent. It is known that the application of the Kalman filtering algorithm may be inefficient, even in the linear case, if the stochastic behavior of the system is not well understood, mainly if the process noise covariance matrix is wrongly chosen. What then if the covariance matrix is not known at all?

Adaptive filtering approaches cope with time variation of system parameters and lack of well known stochastic information about the dynamic model and the measurement noise. One of such approaches is the recursive least-square (RLS) algorithm (Fletcher, 2000) that has the best average...
performance for linear systems. However, RLS (as well as the Kalman filtering method), requires $O(p^2)$ computations per iteration, and the same order of magnitude for the memory, where $p$ is the number of the model parameters to be estimated. Sutton (1992a, b) proposed an adaptive filtering method for linear systems that outperforms least-squares methods and Kalman filtering method when the model and measurement noise are not known. That method is based on a previous learning rate adaptation approach by Jacobs (1988), and requires $O(p)$ computations per iteration, and the same order of magnitude for the memory.

The purpose of the present paper is to propose an online parameter estimation method for nonlinear uncertain systems when no stochastic information about the system uncertainties is available. The method requires $O(p)$ computations per iteration and the same order of magnitude for the storage memory as well. The paper focuses on the suitability of the adaptive learning rate techniques for nonlinear time-varying parametric estimation. It is organized as follows. Section 2 states the problem to be solved. In Section 3 is summarized the nonlinear recursive least-squares (NRLS) based estimation method that will be compared with our method. Section 4 elaborates on the proposed method. The efficiency of the proposed method is demonstrated on a navigation trajectory tracking problem and compared with the NRLS method in Section 5. Finally, conclusion and discussions about the method are given in Section 6.

2. Problem statement

Consider a system whose dynamics is described by the following discrete-time model:

$$y_k = f(x_k, \theta_k) + \eta_k$$  \hspace{1cm} (1)

where $x_k \in \mathbb{R}^q$ is the input variable, $y_k \in \mathbb{R}^n$ the output variable, $\theta_k \in \mathbb{R}^p$ a parameter vector, and $\eta_k$ the measurement noise. Function $f = (f_1, \ldots, f_n)^T$ is a vector of $n$ nonlinear functions that are twice differentiable with respect to the parameter vector. The parameter $\theta_k$ is assumed to vary over time according to a random walk:

$$\theta_{k+1} = \theta_k + \xi_k$$  \hspace{1cm} (2)

where $\xi_k$ is the parameter noise vector.

The problem to be solved is to find an estimate $\hat{\theta}_k$ of the parameter vector $\theta_k$ across-time such that the discrepancy between the actual output $y_k$ and the predicted output $\hat{y}_k = f(x_k, \hat{\theta}_k)$ be as small as possible as $k$ increases. In other words, we are interested in predicting the output of the filter, and therefore we define the output prediction error vector as:

$$e_k = y_k - \hat{y}_k = f(x_k, \theta_k) - f(x_k, \hat{\theta}_k)$$  \hspace{1cm} (3)

3. The nonlinear recursive least square

In the least-squares framework, it is assumed that the actual output and parameter vectors are random variables according to equations (1) and (2), respectively. Particularly, both $\eta_k$ and $\xi_k$ are assumed to be Gaussian with mean zero, uncorrelated one from another, with covariance matrices $I$ and $\mu I$, respectively, ($\mu > 0$ and $I$ is the identity matrix), but unfortunately with the value for $\mu$ being unknown. Under these assumptions, the least-square approach searches for a minimizer of the following cost function at each time step $k$:

$$C_k(\theta) = \sum_{j=0}^k \|e_j\|^2$$  \hspace{1cm} (4)

The minimizing solution of that cost function is well-known (Fletcher, 2000) and is given by the NRLS algorithm:

$$\hat{\theta}_{k+1} = \hat{\theta}_k + K_k(y_k - f(x_k, \hat{\theta}_k)), \hat{\theta}_0 = 0$$  \hspace{1cm} (5)

with:

$$K_k = P_k G_k^T (I + G_k^T P_k G_k)^{-1}$$  \hspace{1cm} (6)

where $G_k$ is the Jacobian matrix of $f$ at $\theta = \hat{\theta}_k$, that is:

$$G_k = \left(\frac{\partial f(x_k, \theta)}{\partial \theta}\right)_{\theta = \hat{\theta}_k}, \text{ and } P_k \text{satisfies the Riccati recursion:}$$

$$P_{k+1} = P_k - P_k G_k G_k^T P_k (I + G_k^T P_k G_k)^{-1}, \quad P_0 = \mu I$$  \hspace{1cm} (7)

When function $f$ is linear, it is known that the RLS solution provided by equations (5)-(7) has the best average performance in the least square sense when the actual value of parameter $\mu$, ($0 < \mu < 2$), is used. However, when parameter $\mu$ is wrongly set or function $f$ is nonlinear, the RLS filtering procedure may diverge. Moreover, the performance of the RLS method is sensitive to the free parameter $\mu$. Therefore, there is a need, on one hand, to devise a filtering method that can deal more efficiently with nonlinear systems and at the same time does not require stochastic information about the measurement and model noise, and on the other hand, its performance should be much less sensitive to its free parameter than in case of the RLS method. Such a method is proposed in the following section.

4. The proposed estimation method

A reasonable loss function $J_k$ to be considered at each discrete time $k$ for computing the estimate $\hat{\theta}_k$ may be defined as:

$$J_k(\theta) = \frac{1}{2} \|y_k - f(x_k, \theta)\|^2$$  \hspace{1cm} (8)

where $\|\cdot\|$ denotes the Euclidean norm, $y_k = (y_k^T, \ldots, y_k^T)^T$ is the actual measurement vector.

4.1 Update equations

Since, function $f$ is supposed to be differentiable with respect to $\theta = (\theta^0, \ldots, \theta^m)^T$, $J_k$ can be minimized by gradient descent according to the following scheme:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - P_k \nabla J_k(\hat{\theta}_k)$$  \hspace{1cm} (9)

where $\hat{\theta}_k$ is the estimate of the parameter vector $\theta$ at iteration $k$ (corresponding to time step $t_k$), $P_k$ is a matrix to be described below, and $\nabla J_k(\hat{\theta}_k)$ the gradient vector of the loss function $J_k$ at $\theta = \hat{\theta}_k$.

The above equation is equivalent to:

$$\hat{\theta}_{k+1} = \hat{\theta}_k + K_k(y_k - f(x_k, \hat{\theta}_k))$$  \hspace{1cm} (10)

with:

$$K_k = P_k G_k^T$$  \hspace{1cm} (11)

$$G_k = \left(\frac{\partial f(x_k, \theta)}{\partial \theta}\right)_{\theta = \hat{\theta}_k}$$  \hspace{1cm} (12)
where \( G_k \) is the Jacobian matrix of \( f(x_k, \theta) \) with respect to \( \theta \) at \( \theta = \hat{\theta}_k \).

Such an approach, void of prior stochastic information, was proposed in (Sutton, 1992a, b) for single output linear systems. The idea was to approximate the estimation error covariance matrix \( P_k \) by a diagonal matrix with positive diagonal elements \( P^\mu_k \) (at iteration \( k \)) defined as:

\[
P^\mu_k = \sigma(r_k^i), \quad i = 1, \ldots, p
\]

(13)

where, for the sake of minimizing the prediction error, \( r_k^i \) is updated by the steepest descent algorithm with a constant meta-stepsize or by the least-mean-square rule. Although the basic linear version guarantees convergence, one should notice that any nonlinear extension may give rise to \( r_k^i \) with high positive values when equation (13) is used. Therefore, the iterate given by equation (9) may not converge when \( f \) is nonlinear and if matrix \( P_k \) is computed through equation (13). Indeed, each component \( P^\mu_k \) operates as a stepsize in a steepest descent minimization scheme (equation (9)); therefore, for \( r_k^i > 0 \), it comes that \( P^\mu_k > 1 \), hence divergence or oscillations of the sequence provided by equation (9) may occur when function /is nonlinear and the search process starts relatively far away from the actual value of the parameter vector. Therefore, we propose, in the sequel, a systematic adaptive learning rate rule in the framework of nonlinear parameter identification.

We search for a diagonal matrix \( P_k \) with positive diagonal elements \( P^\mu_k \) (at iteration \( k \)) or equivalently at time \( t_k \) defined as:

\[
P^\mu_k = \sigma(r_k^i), \quad i = 1, \ldots, p
\]

(14)

where function \( \sigma \) is defined as:

\[
\sigma(r) = \frac{1}{1 + e^{-r}}
\]

(15)

One should noticed that \( \sigma \) is an increasing function on \( \mathbb{R} \) and that for any \( r < 0 \), \( 0 < \sigma(r) < 1 \). Therefore, each diagonal element \( P^\mu_k \) of matrix \( P_k \) is guaranteed to be positive and less than 1.

Parameters \( r_k^i \) in equation (14) are updated by a gradient descent formula so as to minimize the cost function \( J_k \) according to the following scheme:

\[
r_0^i = 0, \quad r_{k+1}^i = r_k^i - \mu \left( \frac{\partial J_k(\theta)}{\partial \theta^i} \right)_{\hat{\theta}_k}, \quad k = 0, 1, \ldots
\]

(16)

where \( \mu, (0 < \mu < 1) \), is a meta-learning rate associated with the parameter vector \( \theta \).

Equation (16) is a meta-learning rule that aims at adjusting the individual learning rates expressed in equation (14). In other words, the individual learning rates \( P^\mu_k \) are learnt using equation (16) for the sake of minimizing the measurements prediction error. Hereafter, equation (16) is worked out so as to find an easily computable expression.

Let:

\[
g_k^i = \left( \frac{\partial J_k(\theta)}{\partial \theta^i} \right)_{\hat{\theta}_k} = - \sum_{j=1}^n v_j \left( \frac{\partial f_j(x_k, \theta)}{\partial \theta^i} \right)_{\hat{\theta}_k} = - e_k^i G_k^i
\]

(17)

where: \( e_k = y_k(t) - f(x_k, \hat{\theta}_k) \) is the prediction error at discrete time \( k \) and \( G_k^i \) the \( i \)th column of the Jacobian matrix \( G_k \) (equation (12)).
Therefore, equation (26) becomes:

\[
\frac{\partial \mathbf{d}_k^j}{\partial \theta_k} = \sum_{j=1}^{n} \left[ e_k^j(t) \left( \frac{\partial^2 f_i(x_k, \theta)}{\partial (\theta)^2} \right)_{\theta = \theta_k} - \left( \frac{\partial f_i(x_k, \theta)}{\partial \theta} \right)_{\theta = \theta_k} \right] \tag{30}
\]

Let \( B_k \) be the vector whose elements \( b_k^j \) are defined as:

\[
b_k^j = \left( \frac{\partial^2 f_i(x_k, \theta)}{\partial (\theta)^2} \right)_{\theta = \theta_k}, \quad j = 1, \ldots, n \tag{31}
\]

Then equation (30) may be written as:

\[
\frac{\partial \mathbf{d}_k^j}{\partial \theta_k} = e_k^j B_k - \|G_k^*\|^2 \tag{32}
\]

However, for slowly varying parameter vector \( \theta \) (as is usually the case), the following approximation is valid:

\[
\frac{\partial \mathbf{d}_k^j}{\partial \theta_k} = \frac{d_k^j - d_{k-1}^j}{\theta_k - \theta_{k-1}} \approx \frac{1}{p_k^*} \left( \frac{d_k^j}{d_{k-1}^j} - 1 \right) \tag{33}
\]

Therefore, equation (26) becomes:

\[
\mathbf{s}_{k+1} = \mathbf{s}_k \left( 1 + p_k^* \right) + p_k^* \mathbf{u}_k^* \mathbf{u}_k^* \tag{34}
\]

with:

\[
p_k^* = \frac{\mathbf{p}_k^*}{\mathbf{p}_k^* - 1} \left( \frac{d_k^j}{d_{k-1}^j} - 1 \right) \tag{35}
\]

### 4.2 Estimation algorithm

The calculations developed above are summarized below as the updating procedure for parameter estimation:

**Step 0.** Set \( k = 1 \); for each \( i = 1, \ldots, p \) set \( 0 < \mu < 1 \), \( \mathbf{s}_0 = 0 \), \( \mathbf{r}_0 = 0 \) as desired.

**Step 1.** While measurements are available \( \mathbf{y}_k \), do:

- Compute \( G_k \) using equation (12);
- Compute \( P_k \) using equation (14);
- Compute \( K_k \) using equation (11);
- Compute \( \theta_{k+1} \) using equation (10);
- For each \( i = 1, \ldots, p \):
  - Compute \( d_k^i \) using equation (22);
  - Update \( \mathbf{s}_{k+1} \) using equation (34);
  - Update \( \mathbf{r}_{k+1} \) using equation (19);
- End-For Each
- \( k = k + 1 \).

### 5. Application to 3D navigation trajectory tracking

We consider the problem of tracking the trajectory of an aircraft through the estimation of the speed, the flight path angle and the heading. Trajectory tracking (Bar-Shalom and Li, 1993; Mook and Shyu, 1992) is a major task in missile guidance, adaptive trajectory control, aerial vehicle collision avoidance in air traffic control, and system supervision. The 3D navigation equations of a flying vehicle are (Stevens and Lewis, 2004):

\[
\begin{align*}
\dot{x} &= V \cos \chi \cos \gamma + w_x, \\
\dot{y} &= V \sin \chi \cos \gamma + w_y, \\
\dot{h} &= V \sin \gamma + w_h
\end{align*}
\]

Where \( x, y, h \) are the 3D coordinates, \( V \) the speed, \( x \) the heading, and \( \gamma \) the flight path angle. The heading is the (ground-track) angle measured counterclockwise from the x-axis to the projection of the speed vector onto the xy-plane. The flight path angle is the angle between the speed vector and the local horizontal. Numbers \( w_x, w_y, w_h \) are measurement uncertainties due to sensors errors and wind disturbances. The vector \((x y h)\) is the output, and \((V \gamma \chi)^T\) the control vector but considered here as the parameter vector \( \theta \). The aim of the present application is to estimate the parameter vector online assuming it not to be measured.

#### 5.1 Modelling

Let denote the state vector and the parameter vector by \( X \) and \( \theta \), respectively, that is:

\[
X = (x \ y \ h)^T, \quad \theta = (V \ \gamma \ \chi)^T
\]

Then equations (36)-(38) may be written in the form:

\[
\dot{X} = g(X, \theta) + \mathbf{w}
\]

where: \( \mathbf{w} = (w_x, w_y, w_h)^T \), and function \( g : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3 \) is defined such that \( g(X, \theta) \) equals the righthand side of equations (36)-(38) without the additive terms \( w_x, w_y \) and \( w_h \). Then equation (39) may be transformed into an equation in form of equation (1) through discretization as described hereafter:

A numerical integration scheme (Euler method, Runge-Kutta, ...) is used to obtain a discrete model of the system of differential equations (39) in the form:

\[
X_{k+1} = f(X_k, \theta_k) + \mathbf{\eta}_k, \quad k = 0, 1, 2, \ldots
\]

where:

\[
X_k = X(t_k), \quad \theta_k = \theta(t_k)
\]

with \( t_k = t_0 + k \Delta t \), \( \Delta t = t_{k+1} - t_k \) is the integration stepsize, and \( f \) is the discretization function.

Let:

\[
Y_k = X_{k+1}, \quad k = 0, 1, 2, \ldots
\]

Hence, equations (40) and (42) taken together give an equation in form of equation (1).

#### 5.2 Estimation procedure

The actual patterns of the time-varying parameters are defined for a small size unmanned aerial vehicle as:

\[
V(t) = \begin{cases} 
25, & 0 \leq t \leq 25 \\
30, & 25 < t \leq 30 \\
30, & 30 < t \leq 60 
\end{cases} 
\]

\[
\gamma(t) = \begin{cases} 
30, & 0 \leq t \leq 20 \\
4t - 150, & 20 < t \leq 25 \\
4t - 160, & 25 < t \leq 45 \\
20, & 45 < t \leq 60 \\
10, & 65 < t \leq 15
\end{cases} 
\]

\[
\chi(t) = \begin{cases} 
6.5t - 87.5, & 15 < t \leq 35 \\
140, & 35 < t \leq 60
\end{cases} 
\]
The speed \( V \) is expressed in meter per second, the path angle \( \gamma \) and the heading \( \chi \) in degree.

As mentioned formerly, we intend to estimate these patterns online. Therefore, we first compute the measurements \( Y_k \) (by simulation) using a real-time integration scheme based on Adams predictor-corrector method (Howe, 1991) to have the discrete model equations (40) and (42). Then we corrupt \( Y_k \) with 10 percent Gaussian random noise, which made the signal-to-noise ratio 10-to-1 for each simulated output, but the patterns for \( V, \gamma \) and \( \chi \) were kept noise-free. This corrupted simulated measurement is considered as the actual measurement. Finally, the learning rate adaptation method presented in Section 4 is used to compute the estimate \( \hat{u}_{k+1} \) (equation (9)). The procedure is repeated as described in the estimation algorithm in Section 4.2 for each actual measurement obtained by simulation of equation (40) and corruption by noise injection of the simulated state vector \( Y_k \).

5.3 Results

Online estimation of the parameter vector \( \theta = (V \gamma \chi)^T \) were performed using separately the RLS and the nonlinear learning rate adaptation (NLRA) presented in the present paper for values of the meta-learning rate \( \mu \) in interval \([0.001, 2]\). For each of these values we compute the mean position error as:

\[
e_{\text{rmse}}^\mu = \frac{1}{N} \sum_{k=1}^{N} \left\| Y_k - \hat{Y}_k^\mu \right\|^2
\]

where \( Y_k = (x_k, y_k, h_k)^T \) is the actual position of the aircraft as described in Section 5.2 (with \( x_k = x(t_k), y_k = y(t_k), \) \( h_k = h(t_k) \)), and \( \hat{Y}_k^\mu \) is the predicted (or estimated) position obtained from the estimate \( \hat{u}_k \) of the parameter vector for a given value of the meta-learning rate \( \mu \) in equation (19).

The results of the estimations are shown in Figure 1. These show that NLRA is much less sensitive to changes in
the meta-learning rate \( \mu \) than RLS, and gives rise to an estimation error much less than RLS does. Furthermore, the maximum estimation error obtained through NLRA for \( \mu \in [0.2] \) is lower than the minimum estimation error provided by RLS on the same range of \( \mu \). Although NLRA is supposed to give good results for small values of \( \mu \) (that is, \( 0 < \mu < 1 \)), it gives estimates equally good even for values of \( \mu \) between 1 and 2 (at least for the present example).

From the results shown in Figure 1, we found that the optimal value of \( \mu \) was 0.37 for NLRA and 0.74 for RLS. For the sake of comparison, Figures 2-4 show, respectively, the estimates of the speed \( V \), the flight path angle \( \gamma \), and the heading \( \chi \) corresponding to the optimal values of \( \mu \) for NLRA and RLS. We observe in all these three figures that NLRA performs better than RLS on the steady-state phases as well as on transient phases. However, there is practically no difference between NLRA and RLS during the transient phase of the heading angle (Figure 4).

Figure 5 shows the actual ground trajectory, and the trajectory estimated with NLRA. We can see that the trajectory estimate is close to the actual one.

6. Conclusion

A method for online parameter estimation is presented for nonlinear uncertain systems for which no stochastic information is available. The estimation procedure computes an individual adaptive learning rate for each parameter, each individual learning rate being controlled by a meta-learning rate rule for the sake of minimizing the measurement prediction error, without requiring stochastic information about the system model and the measurement noise covariance matrices contrarily to the Kalman filtering method. Numerical results about aircraft trajectory tracking

![Figure 1 Mean position error for values of the meta-learning rate in interval [0.001,2]](image-url)
show that the method is able to estimate efficiently time-varying parameters even in presence of measurement noise. Importantly, simulation results show that the performance of the proposed method is much less sensitive to changes in the meta-learning rate and much more accurate than RLS. Because the method is void of prior stochastic information about the model and measurement errors, it may well be an alternative to existing nonlinear (and linear) recursive parameter estimation methods mainly when information about uncertainties related to the model and the measurements is unavailable. Future work will investigate the extension of the proposed method to nonlinear state estimation and feedback control of uncertain systems.
Figure 4 Heading estimation comparison between NLRA and RLS

![Figure 4](image)

Figure 5 Actual ground trajectory, and estimated trajectory by the NLRA method

![Figure 5](image)

References


**About the author**

**K. Bousson** is a Professor in the Department of Aerospace Sciences at the University of Beira Interior in Covilhã Portugal, since February 1995. Formerly, he was a Researcher at the LAAS Laboratory of the French National Council for Scientific Research (CNRS) in Toulouse, France, from 1993 to 1995. His current research activities concern flight control, optimization, and state/parameter estimation. K. Bousson can be contacted at: klbousson@yahoo.com