IDENTIFICATION OF CONTINUOUS-TIME SYSTEMS WITH UNKNOWN TIME DELAYS BY GLOBAL NONLINEAR LEAST-SQUARES METHOD

Hideto Iemura, Zi-Jiang Yang, Shunshoku Kanae, and Kiyoshi Wada

Abstract: This paper considers the identification problem of continuous-time systems with unknown time delays from sampled input-output data. By using a digital prefilter, an approximated discrete-time estimation model is first derived, in which the system parameters remain in their original form and the time delays need not be an integral multiple of sampling period. Then an iterative separable nonlinear least-squares (SEPNLS) method which estimates the time delays and transfer function parameters separably is derived. Furthermore, we propose an iterative global SEPNLS method to avoid convergence to a local minimum of the SEPNLS method by using of stochastic global-optimization techniques. Simulational results show that the global SEPNLS method is able to converge to global estimate.

Keywords: Identification, continuous-time system, time-delay, nonlinear least-squares method, stochastic approximation with convolution smoothing.

1. INTRODUCTION

Since many practical systems such as thermal processes, chemical processes, biological processes and metallurgical processes etc., have inherent time delay, the problem of identifying such a system is of great importance for system analysis and control design.

There have been some typical approaches to identification of continuous-time models with unknown delay. One approach is based on the nonlinear estimation method like nonlinear least-squares (LS) method that searches for the optimum by using a gradient-following technique. In Gawthrop et al. (1989) and Tuch et al. (1994), some variations of pure continuous-time on-line nonlinear LS methods were studied. A major problem of nonlinear estimation methods is that the estimates by such methods may be stuck at local minima. Therefore, the results may be sensitive to the initial values.

This paper considers the identification problem of continuous-time systems with unknown time delays from sampled input-output data. An iterative SEPNLS method which estimates the time delays and transfer function parameters separably is derived. Furthermore, we propose an iterative global SEPNLS method to address the problem of convergence to a local minimum of the SEPNLS method by using of stochastic global-optimization techniques. In particular, we apply stochastic approximation with convolution smoothing (SAS) to the SEPNLS method. This results in the global SEPNLS.

For the single-input single-output (SISO) systems, since there exists only one nonlinear parameter (the time delay) in the estimation problem, the problem of initial setting is considered to be relatively simple. Several trials are enough. For multi-
input single-output (MISO) systems with multiple time delays, the problem is much more difficult. In this study, SEPNLS and global SEPNLS methods are derived for of MISO systems where the time delays in the individual input channels may differ each other. It is easy to extend these methods to the case of multi-input multi-output (MIMO) systems. Simulational results show that the global SEPNLS method is able to converge to global estimates.

2. STATEMENT OF THE PROBLEM

Consider the strictly stable MISO continuous-time system with unknown time delays governed by the following dynamical equation:

\[ \sum_{i=0}^{n} a_i p^{-i} x(t) = \sum_{j=1}^{r} \sum_{i=1}^{m_j} b_{ij} p^{-m_j-i} u_j(t - \tau_j) \]  \hspace{1cm} (1)

where 0 < \tau_j \leq T is the sampling period.

Practically the discrete-time measurement of the output variable is corrupted by a stochastic measurement noise.

\[ y(k) = x(k) + v(k) \]  \hspace{1cm} (2)

where \( y(k) \), \( x(k) \), \( v(k) \) denote \( y(kT) \), \( x(kT) \), \( v(kT) \) respectively.

Our goal is to identify the time delays and the system parameters from sampled data of the inputs and the noisy output.

3. APPROXIMATED DISCRETE-TIME ESTIMATION MODEL

To avoid direct signal derivatives, we introduce a low-pass pre-filter \( Q(p) \) as

\[ Q(p) = \frac{1}{(a + 1)p} \]  \hspace{1cm} (4)

where \( a \) is the time constant which determines the pass-band of \( Q(p) \).

Multiplying both sides of (1) by \( Q(p) \) and using the bilinear transformation based on the block-pulse functions (Jiang & Schaufelberger, 1992), we can obtain the following approximated discrete-time estimation model of the original system (Sagara et al., 1993; Yang et al., 1997):

\[ \xi_{0g}(k) + \sum_{i=1}^{n} a_i \xi_{ig}(k) = \sum_{j=1}^{r} \sum_{i=1}^{m_i} b_{ji} \xi_{(n-m_i+i)a_j}(k - \tau_j) + r(k) \]  \hspace{1cm} (5)

where

\[ r(k) = \sum_{i=0}^{n} a_i \xi_{ie}(k) \]  \hspace{1cm} (6)

and

\[ \xi_{ia_j}(k) = \left( \frac{T}{2} \right)^i (1 + z^{-1})^i (1 - z^{-1})^{n-i} \left[ a(1 - z^{-1}) + \frac{T}{2}(1 + z^{-1}) \right]^{m_j} \tilde{y}_j(k) \]  \hspace{1cm} (7)

where \( z^{-1} \) is the backward shift operator, \( \tilde{v}(k) = (1 + z^{-1})v(k)/2 \) and \( \tilde{y}(k) = (1 + z^{-1})y(k)/2 \).

\[ \tilde{\tau}_j \] in (5) is given by

\[ \tilde{\tau}_j = \tau_j/T = l + \Delta/T \]  \hspace{1cm} (8)

where \( 0 \leq \Delta < T \) and \( l \) is a non-negative integer.

**Remark 1:** Usually it is desirable that the time-delay \( \tau_j \) is an integral multiple of the sampling period for identification of common discrete-time models, whereas our approximated discrete-time estimation model does not require this restriction. In the case of \( \Delta \neq 0 \), we can get \( \xi_{(n-m_j+i)a_j}(k - \tilde{\tau}_j) \) by linear interpolation between \( \xi_{(n-m_j+i)a_j}(k-l) \) and \( \xi_{(n-m_j+i)a_j}(k-l-1) \). (5) can be written in vector form:

\[ \xi_{0g}(k) = \varphi_T(k, \tau) \theta + r(k) \]

\[ \varphi_T(k, \tau) = [-\varphi_{b}^T(k), \varphi_{a_j}^T(k - \tilde{\tau}_j), \cdots, \varphi_{a_j}^T(k - \tilde{\tau}_j)] \]

\[ \varphi_{b}^T(k) = [\xi_{1b}(k), \cdots, \xi_{nb}(k)] \]

\[ \varphi_{a_j}^T(k - \tilde{\tau}_j) = [\xi_{(n-m_j+i)a_j}(k - \tilde{\tau}_j), \cdots, \xi_{na_j}(k - \tilde{\tau}_j)] \]

\[ \theta^T = [a_1^T, b_1^T, \cdots, b_{m_j}^T] \]

\[ \tau^T = [\tau_1, \cdots, \tau_r] \]

\[ a^T = [a_1, \cdots, a_n] \]

\[ b_j^T = [b_{j1}, \cdots, b_{jm_j}] \]
4. SEPNLS METHOD

Given a fixed set of filtered input-output data \{\xi_0(k), \phi^T(k, \theta, \tau), \ldots, \phi^T(k, \theta, \tau)\}_{k=0}^N, the offline parameter estimates are defined as the minimizing arguments of the following LS criterion

\[
V_N(\theta, \tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \frac{1}{2} \varepsilon^2(k, \theta, \tau) \tag{10}
\]

\[
epsilon(k, \theta, \tau) = \xi_0(k) - \phi^T(k, \theta, \tau) \theta
\]
such that

\[
\left[\theta^T, \tau^T\right]^T_N = \arg \min_{\theta, \tau} V_N(\theta, \tau) \tag{11}
\]

The SEPNLS method estimates the time delay vector \(\tau\) and the linear parameter vector \(\theta\) in a separable manner.

When the time delays are known, the linear parameters can be estimated by the linear LS method as

\[
\hat{\theta}_N(\tau) = R^{-1}(N, \tau)f(N, \tau)
\]

\[
R(N, \tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \phi(k, \tau)\phi^T(k, \tau)
\]

\[
f(N, \tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \phi(k, \tau)\xi_0(k)
\]

Then the LS criterion \(V_N(\theta, \tau)\) becomes the following so that the time delays can be estimated separably.

\[
\hat{V}_N(\tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \frac{1}{2} \varepsilon^2(k, \tau) \tag{13}
\]

where

\[
\varepsilon(k, \tau) = \xi_0(k) - \phi^T(k, \tau)R^{-1}(N, \tau)f(N, \tau) \tag{14}
\]

A nonlinear LS problem is called separable if one set of parameters enter linearly and another set nonlinearly in the model for parameter estimation (Ruhe & Wedin, 1980; Ngia, 2001). The time delay vector \(\tau\) and the linear parameter vector \(\theta\) can be estimated separably according to the following theorem. See Ruhe & Wedin (1980), Ngia (2001) for proof and more detailed explanations.

**Theorem 1.** Let \(\hat{\theta}_N(\tau) = R^{-1}(N, \tau)f(N, \tau)\) denotes one solution of the LS criterion (10). Then

\[
\left[\theta^T, \tau^T\right]^T_N = \arg \min_{\theta, \tau} V_N(\theta, \tau) = \arg \min_{\tau} \hat{V}_N(\tau) \tag{15}
\]

Then the estimate of time delays can be obtained as

\[
\hat{\tau}_N = \arg \min_{\tau} \hat{V}_N(\tau) \tag{16}
\]

through the following iterative search algorithm.

\[
\hat{\tau}^{(j+1)}_N = \hat{\tau}^{(j)}_N - \mu^{(j)} \left[\hat{R}^{-1}_N(\hat{\tau}^{(j)}_N)\right]^{-1} \hat{V}^T_N(\hat{\tau}^{(j)}_N) \tag{17}
\]

where \(\mu^{(j)}\) is the step-size which assures that \(\hat{V}_N(\tau)\) decreases and that \(\hat{\tau}_N\) stays in a pre-specified interval, i.e.,

\[
\left[\hat{\tau}^{(j+1)}_N\right]_i \in \Omega, \quad \left[\hat{\tau}^{(j+1)}_N\right]_i - \left[\hat{\tau}^{(j)}_N\right]_i \leq 0
\]

where \(\cdot\) denotes the \(i\)th element of a vector. And \(\hat{V}^T_N(\tau)\) and \(\hat{R}^{-1}_N(\tau)\) are respectively the gradient and the estimate of the Hessian of the LS criterion:

\[
\hat{V}^T_N(\tau) = -\frac{1}{N - k_a} \sum_{k = k_a + 1}^N \psi(k, \tau)\varepsilon(k, \tau)
\]

\[
\hat{R}^{-1}_N(\tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \psi(k, \tau)\psi^T(k, \tau)
\]

\[
\psi(k, \tau) \text{ can be obtained through tedious but straightforward calculations as follows.}
\]

\[
\phi(k, \tau)_j = -\frac{\partial \varepsilon(k, \tau)_j}{\partial \tau_j} = \phi^T_j(k, \tau)R^{-1}(N, \tau)f(N, \tau) + \psi^T(k, \tau)R^{-1}(N, \tau)f(N, \tau)
\]

\[
-\psi^T(k, \tau)R^{-1}(N, \tau)(R_{\tau j}(N, \tau) + R_k(N, \tau)R_{\tau j}(N, \tau)R_{\tau j}(N, \tau)R^{-1}(N, \tau)f(N, \tau)
\]

where

\[
R_{\tau j}(N, \tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \phi_{\tau j}(k, \tau)\xi_0(k)
\]

\[
f_{\tau j}(N, \tau) = \frac{1}{N - k_a} \sum_{k = k_a + 1}^N \phi_{\tau j}(k, \tau)\xi_0(k)
\]

\[
\phi_{\tau j}(k, \tau) = \frac{\partial \phi^T(k, \tau)\phi^T(k, \tau)}{\partial \tau_j}
\]

\[
= [01_{\times m}, 01_{\times m}, \ldots, 01_{\times m}, \phi^T(k - \bar{\tau}_j), 01_{\times m}, \ldots, 01_{\times m}]^T
\]

\[
\phi^T_{\tau j}(k - \bar{\tau}_j) = [-\xi_{(n-m)}]_{\tau j}(k - \bar{\tau}_j), -\xi_{(n-m+1)}]_{\tau j}(k - \bar{\tau}_j), \ldots, -\xi_{(n-1-m)}]_{\tau j}(k - \bar{\tau}_j)
\]

\[
(20)
\]

The SEPNLS method can be summarized as follows.

(1) Let \(j = 0\). Set the initial estimate \(\hat{\tau}^{(0)}_N\) and the considerable upper bound of time delays \(\tau\).

(2) Perform the following.

(a) Compute

\[
\Delta \hat{\tau}^{(j+1)}_N = -\hat{R}^{-1}_N(\hat{\tau}^{(j)}_N)\hat{V}^T_N(\hat{\tau}^{(j)}_N)
\]

(b) Compute

\[
\hat{\tau}^{(j+1)}_N = \hat{\tau}^{(j)}_N + \Delta \hat{\tau}^{(j+1)}_N
\]

(c) Check if \(0 \leq \hat{\tau}^{(j+1)}_N \leq |\bar{\tau}|.\) If not, let

\[
\Delta \hat{\tau}^{(j+1)}_N = 0.5 \Delta \hat{\tau}^{(j+1)}_N
\]

and go back to (b).
Remark 2: Notice that $\mu^{(j)}$ in equation (17) is chosen to guarantee $\tau_N^{(j+1)} \in \Omega_{\tau}$ and that the criterion (13) decreases at each iteration such that

$$\tilde{V}_N\left(\hat{\tau}_N^{(j+1)}\right) \leq \tilde{V}_N\left(\hat{\tau}_N^{(j)}\right).$$

Typically one starts with $\mu^{(1)} = 1$, and test if these requirements are met. If not, let $\mu^{(j)} = 0.5\mu^{(j)}$, and recalculate $\hat{\tau}_N^{(j+1)}$. This process continues iteratively until the requirements are satisfied (Ngia, 2001).

Finally, by substituting $\hat{\tau}_N$ into (12), the linear parameter vector $\hat{\theta}$ can be estimated by the linear LS method (12).

5. GLOBAL SEPNLS METHOD

SAS is a global-optimization algorithm for minimizing a nonconvex function

$$\min_{\tau \in \Omega_{\tau}} \tilde{V}_N(\tau)$$  \quad (21)

The smoothing process represents the convolution of $\tilde{V}_N(\tau)$ with a smoothing function $h(\eta, \beta)$, where $\eta \in R^p$ is a random vector used to perturb $\tau$, and $\beta$ controls the degree of smoothing. This smoothed functional, described in Rubinstein (1981), is given by

$$\hat{V}_N(\tau, \beta) = \int_{-\infty}^{\infty} h(\eta, \beta) \tilde{V}_N(\tau - \eta) d\eta$$

$$= \int_{-\infty}^{\infty} h(\tau - \eta, \beta) \tilde{V}_N(\eta) d\eta$$  \quad (22)

which represents an averaged version of $\tilde{V}_N(\tau)$ weighted by $h(\cdot, \beta)$. The objective of convolution smoothing is to smooth the nonconvex objective function by convolving it with a noise probability density function (pdf). To yield a properly-smoothed functional $\hat{V}_N(\tau, \beta)$, the kernel functional $h(\eta, \beta)$ must have the following properties (Rubinstein, 1981):

1. $h(\eta, \beta) = (1/\beta^p)h(\eta/\beta)$ is piecewise differentiable with respect to $\beta$;
2. $\lim_{\beta \to 0} h(\eta, \beta) = \delta(\eta)$; ($\delta(\eta)$ is the Dirac delta function);
3. $\lim_{\beta \to 0} \hat{V}_N(\tau, \beta) = \tilde{V}_N(\tau)$;
4. $h(\eta, \beta)$ is a pdf.

One of the possible choices for $h(\eta)$ is a Gaussian pdf (Rubinstein, 1981), which leads to the following expression for $h(\eta, \beta)$:

$$h(\eta, \beta) = \frac{1}{(2\pi)^{p/2} \beta^p} \exp \left[ -\frac{1}{2} \sum_{i=1}^{p} \left( \frac{\eta_i}{\beta} \right)^2 \right]$$  \quad (23)

Under these conditions, we can rewrite (22) as the expectation with respect to $\eta$

$$\hat{V}_N(\tau, \beta) = E[\tilde{V}_N(\tau - \eta)]$$  \quad (24)

In our case, $h(\eta, \beta)$ will be the sampled values of its pdf, which is convolved with the original objective function for smoothing. Gaussian, uniform, and Cauchy distributions satisfy the above properties. Throughout this paper, we will use the Gaussian distribution.

The value of $\beta$ plays a dominant role in the smoothing process by controlling the variance of $h(\eta, \beta)$; see properties 2 and 3. Furthermore, property 3 states that to avoid convergence to a local minimum, $\beta$ has to be large at the start of the optimization process and is then reduced to approximately zero as the global minimum is reached. Therefore, there will be a set of smoothed functionals $\hat{V}_N(\tau, \beta^{(j)}) j = 1, 2, \ldots$ that are to be evaluated for different values of $\beta$ before the global optimum point is reached.

Our objective now is to solve the following SAS optimization problem: Minimize the smoothed functional $\hat{V}_N(\tau, \beta)$ with $\beta \to 0$ as $\tau \to \tau^*$, where $\tau^*$ is the global minimizer of the original function $\tilde{V}_N(\tau)$ (Styblinski & Tang, 1990).

The application of this technique to the SEPNLS method requires a gradient operation on the functional $\hat{V}_N(\tau, \beta)$, i.e., $\hat{V}'_N(\tau, \beta)$. As described in (Styblinski & Tang, 1990; Rubinstein 1981), if $h(\eta, \beta)$ is a Gaussian distribution, when only the gradient of $\tilde{V}_N(\cdot)$ is known, then the unbiased gradient estimate of the smoothed functional can be expressed as

$$\hat{V}'_N(\tau, \beta) = \frac{1}{M} \sum_{i=1}^{M} \hat{V}'_N(\tau - \beta \eta_i)$$  \quad (25)

In (25) $\eta_i$ are sampled with the pdf $h(\eta)$. Substituting $M = 1$ in (25) one obtains the one-sample gradient estimator usually used in the stochastic approximation algorithms (Styblinski & Tang, 1990).

$$\hat{V}'_N(\tau, \beta) = \hat{V}'_N(\tau - \beta \eta)$$  \quad (26)

In Styblinski & Tang. (1990), using $\hat{V}'_N(\tau - \beta \eta)$ in (26), SAS is applied to the normalized steepest descent method.

Edmonson et al. (1998) proposed a simplification that involves expressing the gradient $\tau - \beta \eta$ as a Taylor series around the operating point:

$$\hat{V}'_N(\tau - \beta \eta) = \hat{V}'_N(\tau) - \beta \hat{V}'_N(\tau) \eta + \cdots$$  \quad (27)
Additonal, \( \hat{V}_N'(\tau) \) in the above equation is approximated as an identity matrix and only the first two terms of the Taylor series are kept. Then \( \hat{V}_N'(\tau - \beta \eta) \) is used to modify the least mean square (LMS) algorithm.

To extend the idea in Edmonson et al. (1998) to our SEPNLS method, we replace \( \hat{V}_N'(\tau) \) by the estimate of the Hessian of LS criterion, i.e., \( \hat{R}_N(\tau) \), such that

\[
\hat{V}_N'(\tau - \beta \eta) \approx \hat{V}_N'(\tau) - \beta \hat{R}_N(\tau) \eta
\]  

(28)

Replacing \( \hat{V}_N'(\tau) \) in (17) by \( \hat{V}_N'(\tau - \beta \eta) \), we obtain the following result.

\[
\hat{\tau}_N^{(j+1)} = \hat{\tau}_N^{(j)} - \mu^{(j)} \left[ \hat{R}_N(\tau_N^{(j)}) \right]^{-1} \hat{V}_N(\hat{\tau}_N^{(j)}) + \mu^{(j)} \beta^{(j)} \eta
\]  

(29)

This is our global SEPNLS method which modifies the SEPNLS method with an addition of a stochastic perturbation term.

Remark 3: As suggested in Styblinski & Tang (1990), \( \beta \) has to be chosen large at the start of the iterations and is then decreased to approximately zero as the global minimum is reached. And in Edmonson et al. (1998), the sequence of \( \beta^{(j)} \) is chosen as a discrete exponentially decaying function of iteration number \( j \). However, in both works \( \beta \) are chosen by trial and errors. And we have not found in the literature any reliable policy telling us how to determine reliable and efficient values of \( \beta \). In this paper, however, based empirical studies, we recommend the following choice:

\[ \beta^{(j)} = \beta_0 \hat{V}_N(\hat{\tau}_N^{(j)}) \]

where \( \beta_0 \) is a sufficiently large positive constant. It can be understood that if \( \hat{V}_N(\hat{\tau}_N^{(j)}) \) is far from the global minimum, \( \beta^{(j)} \) is large, and if it becomes near the global minimum, \( \beta^{(j)} \) becomes small. Finally, it should be mentioned here that the results are not sensitive to the constant \( \beta_0 \).

The overall algorithm global SEPNLS method can be summarized as follows.

1. Let \( j = 0 \). Set \( \beta_0 \), the initial estimate \( \hat{\tau}_N^{(0)} \) and considerable upper bound of time delays \( \tau \).
2. Set \( \beta^{(0)} = \beta_0 \hat{V}_N(\hat{\tau}_N^{(0)}) \).
3. Perform the following.
   (a) Compute \( \Delta \hat{\tau}_N^{(j+1)} = -\hat{R}_N^{-1}(\hat{\tau}_N^{(j)}) \hat{V}_N(\hat{\tau}_N^{(j)}) + \beta^{(j)} \eta \)
   (b) Compute \( \hat{\tau}_N^{(j+1)} = \hat{\tau}_N^{(j)} + \Delta \hat{\tau}_N^{(j+1)} \)
   (c) Check if \( 0 \leq \hat{\tau}_N^{(j+1)} \leq \tau \). If not, let \( \Delta \hat{\tau}_N^{(j+1)} = 0.5 \Delta \hat{\tau}_N^{(j+1)} \) and go back to (b).
   (d) Check if \( \hat{V}_N(\hat{\tau}_N^{(j+1)}) \leq \hat{V}_N(\hat{\tau}_N^{(j)}) \). If not, let \( \Delta \hat{\tau}_N^{(j+1)} = 0.5 \Delta \hat{\tau}_N^{(j+1)} \) and go back to (b).
4. Terminate the algorithm if the stopping condition is satisfied. Otherwise, let \( j = j + 1 \) and go back to step 2.

Finally, by substituting \( \hat{\tau}_N \) into (12), the linear parameter vector \( \theta \) can be estimated by the linear LS method (12).

Remark 4: According to Remark 2, owing to (a) and (b) in step 3 of the global SEPNLS method, \( \hat{V}_N(\hat{\tau}_N^{(j)}) = 0,1, \cdots \) is a decreasing sequence and hence \( \hat{\tau}_N^{(j+1)} \) is not scrambled very randomly. Additionally, even if the global minimum of \( \hat{V}_N(\hat{\tau}_N^{(j)}) \) and hence \( \beta^{(j)} \) are not exactly zero, \( \hat{V}_N(\hat{\tau}_N^{(j)}) \) does converge.

6. NUMERICAL RESULTS AND CONCLUSIONS

Consider the following MISO continuous-time system:

\[
\dot{x}(t) + a_1 x(t) + a_2 x(t) = b_1 u_1(t - \tau_1) + b_2 u_2(t - \tau_2)
\]

\[
a_1 = 3.0, \quad a_2 = 4.0, \quad b_1 = 2.0, \quad b_2 = 1.0,
\]

\[
\tau_1 = 9.130, \quad \tau_2 = 2.570
\]

(30)

Each input signal is output of a zero-order hold driven by a white signal filtered by a second-order Butterworth filter

\[
L(p) = \frac{1}{(p/\omega_c)^2 + \sqrt{2}(p/\omega_c) + 1}
\]

(31)

which is discretized by the bilinear transformation. The input and output signals are sampled with a sampling period taken as \( T = 0.05 \), and \( \alpha \) in the low-pass pre-filter \( Q(p) \) is 0.4. \( \beta_0 \) is chosen as 105, as suggested in remark 3.

Firstly, the global SEPNLS method is carried out through 20 realizations of the random vector \( \eta \) in the presence of the measurement noise (NSR (noise to signal ratio) is 5%). The data length is 1000. The algorithm is implemented for 200 iterations. The initial value is set as \( \hat{\tau}_N^{(0)} = [1, 1]^T \). Convergence to the global minimum occurred at 95%. An example of the behavior of \( \beta \) and the convergency behaviour of the estimates of the time delays are shown in Figs. 1 and 2 respectively.

Secondly, for one fixed realization of \( \eta \), experiments are carried out through 20 realizations of the measurement noise, when NSR is 5%. The data length, the number of iterations and the initial values are the same as those in the first experiments.
Table 1. Estimates in the case of NSR= 5%.

<table>
<thead>
<tr>
<th></th>
<th>$\hat{\tau}_1(9.13)$</th>
<th>$\hat{\tau}_2(2.57)$</th>
<th>$\hat{a}_1(3.0)$</th>
<th>$\hat{a}_2(4.0)$</th>
<th>$b_{11}(2.0)$</th>
<th>$b_{21}(1.0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td>0.1568</td>
<td>0.7136</td>
<td>-0.0506</td>
<td>1.0330</td>
<td>0.1068</td>
<td>-0.1078</td>
</tr>
<tr>
<td></td>
<td>±0.0063</td>
<td>±0.0054</td>
<td>±0.0009</td>
<td>±0.0011</td>
<td>±0.0012</td>
<td>±0.0013</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>0.1305</td>
<td>2.5607</td>
<td>2.9570</td>
<td>1.0601</td>
<td>±0.0022</td>
<td>±0.0031</td>
</tr>
<tr>
<td></td>
<td>±0.0022</td>
<td>±0.0031</td>
<td>±0.0210</td>
<td>±0.0292</td>
<td>±0.0155</td>
<td>±0.0074</td>
</tr>
</tbody>
</table>

7. REFERENCES


