ANALYSIS AND PARAMETER ESTIMATION OF LINEAR CONTINUOUS SYSTEMS USING LINEAR INTEGRAL FILTER

Setsuo Sagara and Zhen-Yu Zhao

Department of Electrical Engineering
Faculty of Engineering
Kyushu University
Fukuoka 812, Japan

Abstract: The problem of applying the linear integral filter in analysis and parameter estimation of linear continuous systems is discussed. A discrete-time model, which is equivalent to that obtained using the bilinear z transformation, is derived and employed to predict system output. It is shown that the output error can be controlled through the sampling interval. In order to obtain unbiased estimates, an instrumental variable (IV) method is proposed, where the instrumental variables are constituted using adaptive filtering. Some problems on implementation of the recursive IV algorithm are discussed. Both theoretical analysis and simulation study are given to illustrate the proposed methods.

1. Introduction

This paper treats the analysis and parameter estimation problems of continuous dynamic systems. The problems have received considerable attention in recent years (see e.g. [1], [2], [4]-[9]). Typical examples are the methods using orthogonal functions, Poisson moment functionals and numerical integration. It is found that the use of numerical integration has many interesting and attractive properties such as good accuracy, simplicity in calculation and convenience for computer implementation. Therefore, the subject in this paper will be concentrated on analysis and parameter estimation of continuous systems using an operation of numerical integration named the linear integral filter.

The linear integral filter was first developed by the authors to solve the initial condition problem in continuous-time model identification (CMI) [5]. Since then, it has been successfully applied in parameter estimation of various continuous systems in both deterministic and stochastic situations [6]-[8]. In this paper, we intend to apply it in analysis and parameter estimation of continuous systems. It will be shown that the analysis method using the linear integral filter of trapezoidal rule with l=1 gives satisfactory results in a very simple way. A bootstrap estimator of instrumental variable type incorporating adaptive predictor of system output is then proposed to obtain unbiased parameter estimates. The convergence property of the estimator is analysed using the ODE approach and evaluated by Monte-Carlo simulation.

In the following, we consider a continuous SISO system described by the following linear differential equation model as

$$A(s)x(t)=B(s)u(t) \qquad (1.1a)$$

$$A(s)=s^n+a_1s^{n-1}+\cdots+a_n \qquad (1.1b)$$

$$B(s)=b_1s^{n-1}+\cdots+b_n \qquad (1.1c)$$

with the initial conditions given by

$$u_0=(u(0),\cdots,u^{(n-1)}(0))^T,\quad x_0=(x(0),\cdots,x^{(n-1)}(0))^T. \qquad (1.2)$$

where $u(t)$ is the input and $x(t)$ is the output; s is the differential operator, i.e. $sx(t)=dx(t)/dt$ (loosely interpreted here as the Laplace operator).

Here we assume that the input and output signals are sampled with a sampling interval T, and the sampled output is corrupted by a zero-mean noise $v(k)$ which is independent of the input $u(k)$. The observation equation is described by

$$y(k)=x(k)+v(k) \qquad (1.3)$$

where the argument kT has been replaced by k for convenience. By the principle of superposition for linear systems, it can be assumed that $v(k)$ represents the combined effects of all unmeasurable disturbances and measurement noise affecting the process.

It is also assumed that the system under consideration is stable, i.e. all roots of $A(s)$ are in the left-hand side of the s plane, and the polynomials $A(s)$ and $B(s)$ are relatively prime.

Then the analysis problem and parameter estimation problem to be treated here are to find the values of the system output $x(k)$ at sampling instants given an input sequence $\{u(k)\}$ and the known system, and to determine the values of $\{a_i, b_i\}$ with available measurements $\{u(k), y(k)\}$, respectively.

The paper is organized as follows. In the next section, a brief review of the linear integral filter is given. In section 3, the predictor of system output, based on a discrete-time model, is presented. The theoretical analysis on the output error is taken. The procedure of parameter estimation from available measurements is illustrated in detail in section 4, where an instrumental variable method using adaptive filtering is proposed to obtain unbiased estimates. Some problems on the implementation of the IV algorithm are also discussed.

2. Linear Integral Filter

Define a multiple integral of $f(t)$ j times as

$$I_j f(t) = \underbrace{\int_{t-lT}^{t} \cdots \int_{t-lT}^{t}}_{j \text{ times}} f(t) \mathrm{d}t^j \qquad j=0,1,\cdots,n \qquad (2.1)$$

where $I_0 f(t) \triangleq f(t)$ and l is a factor which determines the data length of the linear integral filter. Then the linear integral filter for handling time derivatives is given by the following theorem [6], [8].

Theorem 1. Let $f^{(j)}(t)$ be the jth derivative of $f(t)$ w.r.t. t and $f^{(0)}(t) \triangleq f(t)$. Then the multiple integral of $f^{(j)}(t)$ defined as (2.1) can be approximately calculated as follows

$$I_n f^{(j)}(t) \doteq \mathcal{G}_j f(t) = \sum_{i=0}^{nl} p_i^j q^{-i} f(t) \qquad j=0,1,\cdots,n \qquad (2.2)$$

where the polynomial $\mathcal{G}_j$ is given by

$$\mathcal{G}_j = (1-q^{-l})^j (f_0 + f_1 q^{-1} + \cdots + f_l q^{-l})^{n-j} \qquad j=0,1,\cdots,n. \qquad (2.3)$$

Here q^{-1} is a unit-delay operator, i.e. $q^{-1}f(t)=f(t-T)$ and the coefficients f_i, $i=0,1,\cdots,l$ are determined by integrating rules of numerical integration.

For system analysis, the linear integral filter of trapezoidal rule with $l=1$ is used. In this case, we have the following relation between the truncation error of numerical integration and the sampling interval [15].

Theorem 2. The linear integral filter of trapezoidal rule with $l=1$ is given by

$$I^*_n f^{(j)}(t) = \mathcal{G}^*_j f(t) + O(T^{n+2})$$

$$= f_0^{n-j} (1-q^{-1})^j (1+q^{-1})^{n-j} f(t) + O(T^{n+2}) \qquad j=0,1,\cdots,n \qquad (2.4)$$

where $f_0 = T/2$, and $O(x)$ implies that $\lim_{x\to 0} O(x)/x = c \neq 0$.

Since both input and output signals that we treat here are in sampled data form, we give the LIF in the discrete form as follows

$$I_n f^{(j)}(k) \doteq \mathcal{G}_j f(k) = \sum_{i=0}^{nl} p_i^j f(k-i) \qquad (2.5)$$

$$I^*_n f^{(j)}(k) = \mathcal{G}^*_j f(k) + O(T^{n+2}). \qquad (2.6)$$

3. Analysis of linear continuous systems

In this section, the main results on system analysis will be presented. For details, the reader is referred to [15].

Let us first derive a discrete-time model for predicting the system output in a simple way. Integrating (1.1a) n times over the time interval $[t-T, t)$, using (2.6), we get

$$\mathcal{G}^*_n x(k) + \sum_{i=1}^{n} a_i \mathcal{G}^*_{n-i} x(k) = \sum_{i=1}^{n} b_i \mathcal{G}^*_{n-i} u(k) + \varepsilon(k) \qquad (3.1)$$

Here some real constant M_0 exists such that

$$|\varepsilon(k)| < M_0 T^{n+2}. \qquad (3.2)$$

Define

$$\alpha(q^{-1}) = \alpha_0 + \alpha_1 q^{-1} + \cdots + \alpha_n q^{-n} = \sum_{i=0}^{n} a_i \mathcal{G}^*_{n-i} \qquad (\alpha_0 = 1)$$

$$\beta(q^{-1}) = \beta_0 + \beta_1 q^{-1} + \cdots + \beta_n q^{-n} = \sum_{i=1}^{n} b_i \mathcal{G}^*_{n-i}. \qquad (3.3)$$

Then we have the following result.

Theorem 3. $\alpha(q^{-1})$ is stable if $A(s)$ is stable. That is, if s_i, z_i, $i=1,2,\cdots,n$ are roots of $A(s)$ and $\alpha(z)$ respectively, we have

$$\text{Real}(s_i) < 0 \ (i=1,2,\cdots,n) \rightarrow |z_i| > 1 \ (i=1,2,\cdots,n). \qquad (3.4)$$

Proof. The theorem can be easily proved using the bilinear z transformation that is well-known in the modern control theory.

For stable $A(s)$, $f_0 > 0$ can not be a root of $\bar{A}(s)$ that is defined by

$$\bar{A}(s) = 1 + a_1 s + \cdots + a_n s^n. \qquad (3.5)$$

It follows easily that

$$\alpha_0 = 1 + a_1 f_0 + \cdots + a_n f_0^n = \bar{A}(f_0) \neq 0. \qquad (3.6)$$

Then from (3.1) and (3.3), the system output at sampling instants can be predicted by

$$\hat{x}(k) = [(\alpha_0 - \alpha(q^{-1}))\hat{x}(k) + \beta(q^{-1})u(k)]/\alpha_0 \qquad (3.7)$$

and the output error $\Delta x(k) = x(k) - \hat{x}(k)$ satisfies

$$\lim_{k\to\infty} |\Delta x(k)| < M_1 T^{n+2} \qquad (3.8)$$

where the constant M_1 depends on the roots of $\alpha(q^{-1})$.
It follows easily from (3.8) that for the stable system, the output error can be made small by choosing appropriately small sampling interval when time proceeds sufficiently long. Therefore, (3.7) can be taken as the predictor of the system output $x(k)$ at sampling instants. To initiate the predictor, we may choose $\hat{x}(k)=x(k)$ for $k=0,1,\cdots,n-1$ when the output samples are available at first sampling instants or $\hat{x}(k)=0$ for $k=0,1,\cdots,n-1$ when no information on system output is available a priori. In the later case, the initial effect will die out gradually as time proceeds because the system under consideration is stable.
Now we give a numerical example.

Consider a second-order continuous system given by

$$(s^2 + 2.5s + 1.0)x(t) = 2.0u(t) \qquad (3.9)$$

with initial conditions $x(0)=1.0$, $\dot{x}(0)=5.0$. The system is stimulated by the input signal

$$u(t) = \sin t + 1.5\sin 1.5t + 2.5\sin 2.5t. \qquad (3.10)$$

The results are shown in Fig. 1. The curve denoted by x was obtained using a fourth-order Runge-Kutta method with the step size 0.001, while the curves denoted by $\hat{x}_1$, $\hat{x}_2$ were obtained by the present method with $T=0.01$ when initial conditions were taken as $\hat{x}(k)=x(k)$ and $\hat{x}(k)=0$ for $k=0,1$ respectively.

It can be observed that when $\hat{x}(k)=x(k)$, $k=0,1$ the present method gives very satisfactory results with less computation than the Runge-Kutta method. When $\hat{x}(k)=0$, $k=0,1$ the predicted output is approaching its true value as time proceeds. Therefore this method can be employed in adaptive control or parameter estimation where few computations and good accuracy are required. In the next section the output predictor will be used to generate instrumental variables for the adaptive IV method.

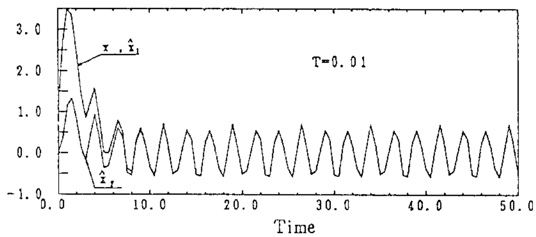

Fig.1 Predicted values of the system output.

4. Parameter estimation

This section consists of three subsections. An idealized IV estimator is presented and discussed in §4.1. In §4.2, the estimator is implemented in on-line way using adaptive filtering, and some problems are discussed. Monte-Carlo simulation analysis is given in §4.3.

4.1 Idealized IV estimator

Operating on each of the terms in (1.1a) with the linear integral filter in the discrete form (2.5) and substituting $x(k)$ with $y(k)$ in the observation equation (1.3) yields

$$\vartheta_n y(k)+\sum_{j=1}^{n} a_j\vartheta_{n-j}y(k)=\sum_{j=1}^{n} b_j\vartheta_{n-j}u(k)+e(k) \qquad (4.1)$$

where the equation error $e(k)$ is given by

$$e(k)=\vartheta_n v(k)+\sum_{j=1}^{n} a_j\vartheta_{n-j}v(k)$$
$$\triangleq(h_0+h_1q^{-1}+\cdots+h_{nl}q^{-nl})v(k)$$
$$\triangleq H(q^{-1})v(k) \qquad (4.2)$$

and

$$h_i=\sum_{j=0}^{n} a_j p_i^{n-j} \qquad a_0=1,\ i=0,1,\cdots,nl. \qquad (4.3)$$

Here introduce the following notations

$$\theta^*=(a_1\ \cdots\ a_n\ b_1\ \cdots\ b_n)^T$$
$$\hat\theta=(\hat a_1\ \cdots\ \hat a_n\ \hat b_1\ \cdots\ \hat b_n)^T$$
$$\varphi(k)=(-\vartheta_{n-1}y(k)\ \cdots\ -\vartheta_0 y(k)\ \vartheta_{n-1}u(k)\ \cdots\ \vartheta_0 u(k))^T$$
$$\overline{\varphi}(k)=(-\vartheta_{n-1}x(k)\ \cdots\ -\vartheta_0 x(k)\ \vartheta_{n-1}u(k)\ \cdots\ \vartheta_0 u(k))^T \qquad (4.4)$$

θ^* and $\hat\theta$ denote the vector of true parameters and the vector of estimated parameters respectively. Note that the elements of $\overline{\varphi}(k)$ are composed of the input signal $u(k)$ and the noise-free output signal $x(k)$, and thus they are independent of $v(k)$ and also $e(k)$. It is, therefore, possible to use $\overline{\varphi}(k)$ as an instrumental variable. In this case, the idealized IV estimator can be written as follows

$$\hat\theta_N=[\sum_{k=1}^{N}\overline{\varphi}(k)\varphi(k)^T]^{-1}\sum_{k=1}^{N}\overline{\varphi}(k)\vartheta_n y(k) \qquad (4.5)$$

where N represents the number of data. It is easy to find that

$$E[\overline{\varphi}(k)(\vartheta_n y(k)-\varphi(k)^T\theta^*)]$$
$$=E[\overline{\varphi}(k)H(q^{-1})v(k)]=0 \qquad (4.6)$$

which implies that the IV estimates given by (4.5) is asymptotically consistent with probability one (w.p.1) provided the inverse exists [7].
This kind of choice for instrumental variables is very well-known and popular in discrete-time model identification (DMI) (see e.g. [12], [13]). However, this choice is not possible in practice, because the true parameters are required. Instead it is naturally suggested, as often used in DMI, to implement the estimator using the "boot-strapping" technique in on-line or iterative manner [11]. In this way, $\overline{\varphi}(k)$ is substituted with $\tilde\varphi(k,\hat\theta)$ which is defined as

$$\tilde\varphi(k,\hat\theta)=(-\vartheta_{n-1}\tilde x(k)\ \cdots\ -\vartheta_0\tilde x(k)\ \vartheta_{n-1}u(k)\ \cdots\ \vartheta_0 u(k))^T \qquad (4.7)$$

where the signal $\tilde x(k)$ is obtained by filtering the input signal according to

$$\tilde x(k)=\frac{\hat\beta(q^{-1})}{\hat\alpha(q^{-1})}u(k) \qquad (4.8)$$

and

$$\hat\alpha(q^{-1})=\sum_{i=0}^{n}\hat a_i\vartheta^*_{n-i} \qquad (\hat a_0=1)$$
$$\hat\beta(q^{-1})=\sum_{i=1}^{n}\hat b_i\vartheta^*_{n-i} \qquad (4.9)$$

(see (3.3)).

4.2 On-line implementation

For simplicity, only the on-line estimator will be discussed here. In this case, the estimates are updated at each sampling instant k by the following algorithm [3]

$$\hat\theta(k)=\hat\theta(k-1)+L(k)\varepsilon(k)$$
$$L(k)=\frac{P(k-1)\zeta(k)}{1+\varphi(k)^T P(k-1)\zeta(k)}$$
$$P(k)=P(k-1)-L(k)\varphi(k)^T P(k-1)$$
$$\varepsilon(k)=\vartheta_n y(k)-\varphi(k)^T\hat\theta(k-1). \qquad (4.10)$$

When the vector $\zeta(k)$ is replaced by the vector $\tilde\varphi(k,\hat\theta(k-1))$, the above algorithm is the required RIV algorithm discussed above, while it becomes the well-known RLS algorithm when the vector $\varphi(k)$ substitutes the vector $\zeta(k)$. It has been shown that the LS estimates so-obtained are always asymptotically biased [6], [14].

The convergence analysis of the above RIV algorithm can be carried out using the ordinary differential equation (ODE) approach [10]. It is easy to verify that

$$E[\tilde\varphi(k,\theta^*)(\vartheta_n y(k)-\varphi(k)^T\theta^*)]$$
$$=E[\overline{\varphi}(k)H(q^{-1})v(k)]=0 \qquad (4.11)$$

and the matrix

$$E[\tilde{\varphi}(k,\theta^*)\varphi(k)^T]^{-1}\frac{\partial}{\partial\theta}E[\tilde{\varphi}(k,\hat{\theta})(\vartheta_n y(k)-\varphi(k)^T\hat{\theta})]|_{\hat{\theta}=\theta^*}$$
$$=E[\overline{\varphi}(k)\varphi(k)^T]^{-1}E[\frac{\partial}{\partial\theta}\tilde{\varphi}(k,\hat{\theta})|_{\hat{\theta}=\theta^*}H(q^{-1})v(k)-\tilde{\varphi}(k,\theta^*)\varphi(k)^T]$$
$$=E[\overline{\varphi}(k)\varphi(k)^T]^{-1}(-E[\overline{\varphi}(k)\varphi(k)^T])$$
$$=-I \tag{4.12}$$
has all eigenvalues in the left half plane, where $\tilde{\varphi}(k,\theta^*)=\overline{\varphi}(k)$ has been used. Therefore, from (4.11) and (4.12), we can conclude that

Theorem 4. The RIV estimates generated by (4.10) is locally convergent to the vector θ^* of true parameters.

In practical use of the on-line algorithm (4.10), the following aspects should be considered.

(1) Setting initial conditions $\hat{\theta}(0)$ and $P(0)$. For the RLS algorithm, $\hat{\theta}(0)=0$ and $P(0)=c^2I$ are often taken where c is a large number. But in the above RIV algorithm, $\hat{\theta}(0)=0$ can not be taken because the adaptive filtering scheme (4.8), (4.9) is used. One easy-to-use way is to take an LS estimate as the initial values of the RIV algorithm, that is, the on-line algorithm (4.10) runs with $\zeta(k)=\varphi(k)$ for some first iterations, and then is switched to the IV algorithm with $\zeta(k)=\tilde{\varphi}(k,\hat{\theta}(k-1))$.

(2) Filtering the estimates before they are used in (4.8), (4.9). In order to improve the function of the adaptive filtering scheme, certain modification is needed, i.e. $\hat{\theta}(k)$ is substituted with $\hat{\theta}_F(k)$ which is given by

$$\hat{\theta}_F(k)=F(q^{-1})\hat{\theta}(k) \tag{4.13}$$

where $F(q^{-1})$ may be a low pass filter, or $F(q^{-1})=q^{-\tau}$. Here the mean estimates

$$\hat{\theta}_{min}(k)=\frac{1}{m}(1+q^{-1}+\cdots+q^{-m})\hat{\theta}(k) \tag{4.14}$$

is taken as $\hat{\theta}_F(k)$.

(3) Testing stability. Since the adaptive filtering scheme is used for generating the instrumental variables, its stableness must be assured. According to Theorem 3, it is then necessary to keep the parameter estimates $\hat{\theta}(k)$ in the stable domain $\mathfrak{D}_s$={ all roots of $\hat{A}(s)$ are in the left half plane}. For this purpose, the following projection algorithm can be used instead of (4.10) [3].

Step 1. Choose a factor $0\le\mu<1$.
Step 2. Compute $\Delta\hat{\theta}(k)=L(k)\varepsilon(k)$.
Step 3. Compute $\hat{\theta}(k)=\hat{\theta}(k-1)+\Delta\hat{\theta}(k)$.
Step 4. Test if $\hat{\theta}(k)\in\mathfrak{D}_s$. If yes, go to next iteration; if not, go to step 5.
Step 5. Set $\Delta\hat{\theta}(k):=\mu\Delta\hat{\theta}(k)$ and go to step 3. (4.15)

Here the factor μ determines the reduction of the step size. The experience tells that μ=0.5 works well [3].

An implementation scheme for the above adaptive instrumental variable method is shown in Fig.2.

4.3 Monte-Carlo simulation analysis

To verify its feasibility, the above algorithm has been evaluated by applying Monte-Carlo simulation to a second-order continuous system described by

$$x^{(2)}(t)+a_1x^{(1)}(t)+a_2x(t)=b_1u^{(1)}(t)+b_2u(t)$$
$$y(k)=x(k)+v(k) \tag{4.16}$$

with system parameters given by

$$a_1=2.8,\ a_2=4.0,\ b_1=0.0,\ b_2=5.0 \tag{4.17}$$

and subject to initial conditions $x(0)=1.0$, $x^{(1)}(0)=5.0$.

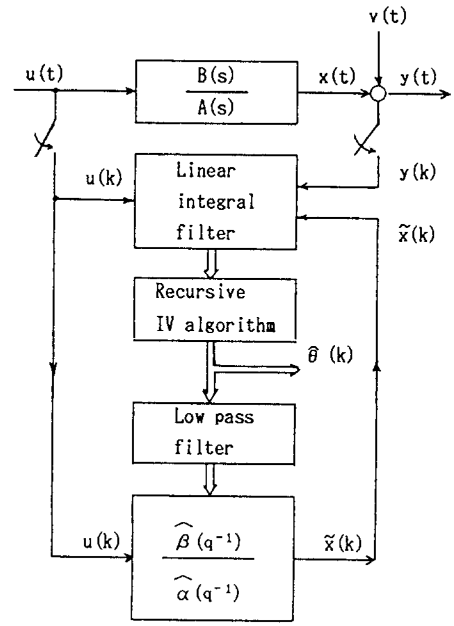

Fig.2 An implementation scheme for the adaptive IV method.

In the simulation, the system was stimulated by the input signal $u(t)$ as (3.10). The linear integral filter of trapezoidal rule with l=12 was used, and the sampling interval T=0.05 was taken. The on-line algorithm (4.15) was run with $\zeta(k)=\varphi(k)$ at first 100 iterations and then turned to the RIV algorithm with m=5 in (4.14). The following cases of the noise contained in the system output were considered.

(1) White noise:

$$v(k)=w(k) \tag{4.18}$$

(2) Moving average (MA) process noise:

$$v(k)=(1-1.5q^{-1}+0.5725q^{-2})w(k) \tag{4.19}$$

(3) Autoregressive moving average (ARMA) process noise:

$$v(k)=\frac{1-1.5q^{-1}+0.5725q^{-2}}{1-1.6q^{-1}+0.68q^{-2}}w(k) \tag{4.20}$$

where $\{w(k)\}$ was an identically and independently distributed (i.i.d.) Gaussian sequence. The variance of $w(k)$ was adjusted to obtain the desired ratio of noise to signal (N/S) defined by

$$N/S=\frac{\text{SD of } v(k)}{\text{SD of } x(k)} \tag{4.21}$$

(SD, standard deviation). Here the N/S ratio was taken 20%.

Tables 2–4 show the results which were obtained from Monte Carlo simulation of 20 experiments in cases of white noise, MA process noise and ARMA process noise, respectively. The tables include the computed mean and standard deviation of each estimate. The mean normalized error (MNE) and average standard deviation (ASD) are also included for quick comparison. They are defined by

$$MNE=\frac{\|\hat{\theta}_{mean}-\theta\|^2}{\|\theta\|^2}$$

$$ASD=\frac{\Sigma\ \sigma_i}{2n}. \quad (4.22)$$

In order to demonstrate the effectiveness of the IV estimator, the LS estimates in white noise case is also presented in Table 1. It can be observed from Tables 1–2 that the LS estimates are asymptotically biased, while the IV estimates seem to be convergent to the true parameters. It can also found from Tables 2–4 that the precision of parameter estimates are affected by the type of noise. When $v(k)$ is a filtered white noise by an MA process, the effect on the precision seems to be less than that of the other two cases. Nevertheless, the precision in all cases can be improved by taking a large number of data.

5. Conclusion

The linear integral filter for handling time derivatives has been employed in the analysis and parameter estimation of linear continuous systems.

The discrete time model obtained using the linear integral filter of trapezoidal rule with l=1 is equivalent to that obtained by the well-known bilinear z transformation, and can be used to predict the system output at sampling instants. Some results on the output error are presented. It has been found that the system output can be predicted in a very simple and accurate way. Thus it can be used in adaptive control and parameter estimation, where simple analysis methods are often required.

For parameter estimation, an identification model is first derived, from which parameters of a continuous system can be determined in a direct way. The idealized estimator of instrumental variable type using the noise-free output signals is discussed, and then implemented using the "boot strapping" technique in an on-line way. The output predictor discussed in section 3 is used to generate the instrumental variables in the adaptive IV method. The convergence analysis of the proposed on-line algorithm is carried out using the ODE approach, and some problems on implementation are also discussed. The results obtained using Monte-Carlo simulation have confirmed the theoretical analysis.

REFERENCES

[1] Cheng, B. and N.S. Hsu (1982). Analysis and Parameter Estimation of Bilinear Systems via Block pulse Functions. *Int. J. Control*, 36, 1, 53–65.

[2] Liu, C.C. and Y.P. Shih (1985). Analysis and Parameter Identification of Linear Systems via Chebyshev Polynomials of the Second Kind. *Int. J. Systems Sci.* 16, 6, 753–759.

[3] Ljung, L. and T. Söderström (1983). *Theory and Practice of Recursive Identification*. The MIT Press, Cambridge, Mass.

[4] Prasada Rao, G. and K.R. Palanisamy (1983). Improved Algorithms for Parameter Identification in Continuous Systems via Walsh Functions. *IEE Proc.*, D, 130, 1, 9–16.

[5] Sagara, S. and Z.Y. Zhao (1987). An Improved Algorithm for On-line Identification in Continuous Systems via Numerical Integration. *Proceedings of the 26th SICE annual conference*, Hiroshima, Japan, II, 1077–1080.

[6] Sagara, S. and Z.Y. Zhao (1987). On-line Identification of Continuous Systems using Linear Integral Filter. *Preprints of the 19th JAACE symposium on stochastic systems theory and its applications*, Fukuoka, Japan, 41–46.

[7] Sagara, S. and Z.Y. Zhao (1988). Numerical Integration Approach to On-line Identification of Continuous Systems in Presence of Measurement Noise. *The 8th IFAC symposium on identification and system parameter estimation*, Beijing, China (to appear).

[8] Sagara, S. and Z.Y. Zhao (1988). On-line Identification of Continuous Systems using the Operation of Numerical Integration. *T. IEE Japan*, 108–C, 8.

[9] Saha, D.C. and G. Prasada Rao (1983). *Identification of Continuous Dynamical Systems – The Poisson Moment Functional (PMF) Approach.* Lecture Notes in Control and Information Sciences, Springer-Verlag, 56.

[10] Söderström, T., L. Ljung and I. Gustavsson (1978). A Theoretical Analysis of Recursive Identification Methods. *Automatica*, 14, 231–244.

[11] Stoica, P. and T. Söderström (1981). Asymptotic behavior of some bootstrap estimators. *Int. J. Control*, 33, 433–454.

[12] Wong, k. and E. Polak (1967). Identification of Linear Discrete Time Systems using the Instrumental Variable Method. *IEEE Trans on Autom. Cont.* 12, 707–718.

[13] Young, P.C. (1970). An Instrumental Variable Method for Real-time Identification of a Noisy Process. *Automatica*, 6, 1, 271–287.

[14] Zhao, Z.Y. (1988). *Numerical Integration Approach to Recursive Identification of Continuous Systems.* M.E. thesis, Dept. of Eng., Kyushu University, Japan.

[15] Zhao, Z.Y. and S. Sagara (1988). A Numerical Method for the Analysis of Continuous Systems. *Proceedings of the 27th SICE annual conference*, Narashino, Japan, I, 69–70.

Table 1. Parameter estimates using the LS algorithm in white noise case.

	True value	N=1000	N=2000	N=5000	N=7000	N=10000
a_1	2.8	2.566±.120	2.536±.125	2.527±.056	2.522±.048	2.521±.037
a_2	4.0	3.951±.183	3.957±.135	3.984±.059	4.003±.060	4.003±.045
b_1	0.0	0.063±.047	0.068±.047	0.065±.020	0.061±.021	0.061±.015
b_2	5.0	4.649±.186	4.612±.178	4.593±.081	4.578±.069	4.577±.058
MNE($\times 10^{-3}$)		3.78	4.63	5.00	5.30	5.34
ASD($\times 10^{-1}$)		1.34	1.21	0.54	0.49	0.38

Table 2. Parameter estimates using the IV algorithm in white noise case.

	True value	N=1000	N=2000	N=5000	N=7000	N=10000
a_1	2.8	2.904±.153	2.864±.121	2.849±.060	2.832±.058	2.826±.043
a_2	4.0	3.956±.184	3.952±.122	3.984±.061	3.991±.059	3.987±.043
b_1	0.0	-0.010±.045	-0.000±.040	-0.003±.019	-0.002±.020	0.000±.015
b_2	5.0	5.167±.250	5.119±.180	5.088±.092	5.058±.088	5.052±.066
MNE($\times 10^{-3}$)		0.84	0.42	0.21	0.09	0.07
ASD($\times 10^{-1}$)		1.58	1.16	0.58	0.57	0.41

Table 3. Parameter estimates using the IV algorithm in MA process noise case.

	True value	N=1000	N=2000	N=5000	N=7000	N=10000
a_1	2.8	2.820±.009	2.817±.009	2.818±.007	2.818±.007	2.818±.006
a_2	4.0	3.996±.011	3.996±.008	3.997±.005	3.998±.006	3.998±.005
b_1	0.0	0.000±.003	0.001±.002	0.000±.001	0.000±.001	0.000±.001
b_2	5.0	5.035±.012	5.032±.012	5.033±.007	5.033±.006	5.033±.006
MNE($\times 10^{-3}$)		0.03	0.03	0.03	0.03	0.03
ASD($\times 10^{-1}$)		0.09	0.08	0.05	0.05	0.04

Table 4. Parameter estimates using the IV algorithm in ARMA process noise case.

	True value	N=1000	N=2000	N=5000	N=7000	N=10000
a_1	2.8	2.915±.190	2.869±.152	2.853±.075	2.835±.075	2.828±.056
a_2	4.0	3.943±.229	3.946±.139	3.982±.074	3.990±.068	3.986±.049
b_1	0.0	-0.010±.060	-0.001±.048	-0.003±.024	-0.002±.025	0.001±.019
b_2	5.0	5.188±.309	5.128±.230	5.095±.117	5.063±.115	5.055±.086
MNE($\times 10^{-3}$)		1.06	0.49	0.25	0.11	0.08
ASD($\times 10^{-1}$)		1.97	1.42	0.72	0.71	0.52