'95 KACC (1995. 10. 23 ~ 25)

ADAPTIVE IDENTIFICATION OF VOLTERRA KERNEL OF NONLINEAR SYSTEMS

Sun Yeping* and Hiroshi Kashiwagi*

*Faculty of Engineering, Kumamoto UniversOity, Kumamoto,860, Japan

Tel: +81-96-342-3742; Fax: +81-96-342-3730; E-mail: kashiwa@gpo.kumamoto-u.ac.jp

Abstract A real time and adaptive method for obtaining Volterra kernels of a nonlinear system by use of pseudorandom M-sequences and correlation technique is proposed . The Volterra kernels are calculated real time and the obtained Volterra kernels becomes more accurate as time goes on. The simulation results show the effectiveness of this method for identifying time-varying nonlinear system.

Keywords Nonlinear systems, Identification, Correlation, M-sequence, Volterra kernel

1. INTRODUCTION

A non-linear dynamical system is, in general, described by use of Volterra series expansion, each term containing so called Volterra kernel. So the measurement of Volterra kernels becomes an important task for non-linear system identification.

Barker *et al* [1] proposed the use of pseudorandom signals, especially antisymmetric M-sequence, for obtaining 2nd-order Volterra kernels with restricted conditions .The authors[3),6)] proposed a new method for obtaining not only the linear impulse response, but also Volterra kernels of non-linear system simultaneously. We can obtain Volterra kernels of not only second order but also third order. In addition, this method is extended to make possible the real time identification of nonlinear system: that is, after the output is measured in one period of the input M-sequence, we can obtain Volterra kernels in a real time. This method for obtaining Volterra kernels of non-linear system is further developed for adaptive identification of Volterra kernels of non-linear systems.

2. PRINCIPLE OF THE METHOD

A non-linear dynamical system is, in general, described as follows.

$$y(t) = \sum_{i=1}^{\infty} \int_0^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} g_i(\tau_1, \tau_2, \cdots \tau_i) \times u(t-\tau_1)u(t-\tau_2)\cdots u(t-\tau_i)d\tau_1 d\tau_2 \cdots d\tau_i \quad (1)$$

where $u(t)$ is the input, and $y(t)$ is the output of the non-linear system, and $g_i(\tau_1, \tau_2, ...)$ is called Volterra kernel of i-th order.

When we take the crosscorrelation function between the input $u(t)$ and the output $y(t)$, we have,

$$\phi_{uy}(\tau) = \overline{u(t-\tau)y(t)} = \sum_{i=1}^{\infty} \int_0^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} g_i(\tau_1, \tau_2, \cdots \tau_i) \times \overline{u(t-\tau)u(t-\tau_1)\cdots u(t-\tau_i)}d\tau_1 d\tau_2 \cdots d\tau_i \quad (2)$$

where $\phi_{uy}(\tau)$ is the crosscorrelation function of $u(t)$ and $y(t)$ and ——— denotes time average.

The difficulty of obtaining $g_i(\tau_1, \tau_2, \cdots \tau_i)$ from $\phi_{uy}(\tau)$ is , in general, due to the difficulty of getting $(i+1)$th moment of the input $u(t)$, because the n-th moment of the signal is very difficult to obtain for actual signals.

We have shown[3)] that when we use an M-sequence as an input to the system, the n-th moment of $u(t)$ can be easily obtained by use of so-called "shift and add property" of the M-sequence. So we can obtain the Volterra kernels $g_i(\tau_1, \tau_2, \cdots \tau_i)$ from simply measuring the crosscorrelation function between the input and output of the non-linear system.

The $(i+1)$th moment of the input M-sequence $u(t)$ can be written as

$$\overline{u(t-\tau)u(t-\tau_1)u(t-\tau_2)\cdots u(t-\tau_i)} = \begin{cases} 1 & \text{(for certain } \tau) \\ -1/N & \text{(otherwise)} \end{cases} \quad (3)$$

where N is the period of the M-sequence. When we use the M-sequence with the degree greater than 10, $1/N$

is smaller than 10^{-3}. So Eq.(3) can be approximated as a set of impulses which appear at certain τ's.

Eq.(3) is due to the so-called shift and add property of the M-sequence; that is, for any integer $k_{i1}, k_{i2}, \cdots k_{i,i-1}$ (suppose $k_{i1} < k_{i2} < \cdots, k_{ii}$), there exists a unique k_{ii} (mod N) such that

$$u(t)u(t+k_{i1})u(t+k_{i2})\cdots u(t+k_{i,i-1}) = u(t+k_{ii}) \quad (4)$$

Note that when $k_{ij} (j = 1,2,...i)$ satisfy Eq.(4), then $2^p k_{ij}$ also satisfy Eq.(4) for any integer p. Therefore Eq.(3) becomes unity when

$$\tau_1 = \tau - k_{i1}, \tau_2 = \tau - k_{i2}, \cdots \tau_i = \tau - k_{ii} \quad (5)$$

Therefore Eq.(2) becomes

$$\phi_{uy}(\tau) = \sum_{i=1}^{\infty} g_i(\tau - k_{i1}, \tau - k_{i2}, \cdots \tau - k_{ii}) \quad (6)$$

Since $g_i(\tau_1, \tau_2, \cdots \tau_i)$ is zero when any of τ_i is smaller than zero, each $g_i(\tau - k_{i1}, \tau - k_{i2}, \cdots \tau - k_{ii})$ in Eq.(6) appear in the crosscorrelation function $\phi_{uy}(\tau)$ when $\tau > k_{ii}$. If the k_{ii} of i-th Volterra kernel g_i are sufficiently apart from each other(say, more than $50\Delta t$, where Δt is the time increment of the measurement time), we can obtain each Volterra kernel $g_i(\tau - k_{i1}, \tau - k_{i2}, \cdots, \tau - k_{ii})$ from Eq.(6). Volterra kernels $g_i(\tau_1, \tau_2, \cdots \tau_i)$ are obtained as a set of crosssections along 45 degree lines in $(\tau_1, \tau_2, \cdots, \tau_i)$ space. In order for this to be realized, we have selected some suitable M-sequences.

Table 1 and Table 2 show some of k_{i2} and k_{i3}, respectively, when $f(x) = 1515155$ is used for generating the input M-sequence. In these tables, ds shows distance from the nearest $k_{i,i}$.

3. REAL TIME METHOD FOR OBTAINING VOLTERRA KERNELS

As is explained in section 2, the i-th Volterra kernels $g_i(\tau - k_{i1}, \tau - k_{i2}, \cdots, \tau - k_{ii})$ appears in the crosscorrelation $\phi_{uy}(\tau)$ when $\tau > k_{ii}$. Therefore, in order to obtain $g_i(\tau - k_{i1}, \tau - k_{i2}, \cdots, \tau - k_{ii})$, it is enough that we obtain $\phi_{uy}(\tau)$ for those τ's which are in the range of $k_{ii} < \tau < k_{ii} + 50$. That is, we don't have to calculate $\phi_{uy}(\tau)$ for all τ, but only for those τ in a necessary range.

And one more important point of our real time identification method is that we can obtain the delayed version $u(t - \tau)$ of input M-sequence $u(t)$ directly without using any delay elements. This is due to the property of M-sequence. For example, when we denote the delay operator as x and we would like to obtain the d-delayed version of M-sequence, x^d mod $f(x)$ is first calculated,

$$x^d = r_0x^0 + r_1x^1 + r_2x^2 + \cdots + r_{n-1}x^{n-1} \quad (7)$$

where $f(x)$ is the characteristic polynomial for generating M-sequence of n-th order.

Table 1: For 2nd order Volterra kernels

n = 18 $\quad f(x) = 1515155$

df=$k_{2,2} - k_{1,2}$

df	$(k_{1,2}, k_{2,2}) - ds$
1	(253039, 253040)- 93
2	(243935, 243937)- 668
3	(256101, 256104)- 212
4	(225727, 225731)- 34
5	(97638, 97643)- 572
6	(250059, 250065)- 471
7	(159826, 159833)- 664
8	(189311, 189319)- 68
9	(124205, 124214)- 250
10	(195276, 195286)- 377

Table 2: For 3rd order Volterra kernels

n = 18 $\quad f(x) = 1515155$

df1=$k_{3,3} - k_{1,3}$ df2=$k_{3,3} - k_{2,3}$

DF(df1,df2)	$(k_{1,3}, k_{2,3}, k_{3,3}) - ds$
(2, 1)	(3062, 3063, 3064)- 220
(3, 1)	(245587, 245589, 245590)- 230
(3, 2)	(185486, 185487, 185489)- 263
(4, 1)	(95570, 95573, 95574)- 286
(4, 2)	(6124, 6126, 6128)- 237
(4, 3)	(185222, 185223, 185226)- 263
(5, 1)	(188548, 188552, 188553)- 297
(5, 2)	(118126, 118129, 118131)- 381
(5, 3)	(209335, 209337, 209340)- 831
(5, 4)	(248649, 248650, 248654)- 226

Then the coefficients r_i's in Eq.(7) determine which stage of the M-sequence generating shift register should be added modulo 2 for obtaining $u(\tau - d)$. The schematic diagram for obtaining the delayed version of $u(\tau)$ is shown in Fig.1.

Table 3 shows example of r_i's in Eq.(7).

Fig. 2 shows the configuration of real time identification scheme for non-linear systems.

In order to obtain x^d mod $f(x)$, it would be advantageous to use the fast method for division over GF(2)[3].

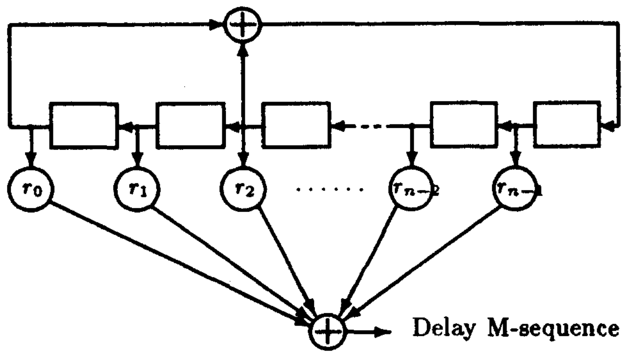

Fig 1: Schematic diagram for obtaining delayed M-sequence ($\oplus$ denotes addition mod 2)

4. ADAPTIVE METHOD FOR OBTAINING VOLTERRA KERNELS

Eq.(2) is written in a discrete form as follows.

$$\phi_{uy}(k\Delta t) = \frac{1}{N}\sum_{i=1}^{N} u(\overline{i-k}\Delta t)y(i\Delta t) \tag{8}$$

where Δt is the time increment, and $N = 2^n - 1$, period of the input M-sequence.

Summation in Eq.(8) is devided into, say, m parts and summed up into m different registers R_j's, as follows.

$$\phi_{uy}(k\Delta t) = \frac{1}{N}\sum_{i=1}^{m} R_i$$

$$R_j = \sum_{i=N_{j-1}+1}^{i=N_j} u(\overline{i-k}\Delta t)y(i\Delta t)$$

where $N = N_1 + N_2 + \cdots + N_m$.

When the m-th register R_m is added N_m times, then each register R_i is shifted to left as $R_j \Leftarrow R_{j+1}$, thus newly obtained information about Volterra kernels are replaced with the older contents.

So even if the system is a time-varying nonlinear system, we can obtain the updated Volterra kernels in a real time when the time-variation of the system is slow.

5. APPLICATIONS

This method of Volterra kernel identification is applied to several actual non-linear systems. The one of the applications is to the dead-band type non-linear system as shown in Fig.3.

We have compared the actual output of the non-linear system with the estimated output calculated from measured Volterra kernels as shown in Fig.4, and got a good agreement between them, showing the validity of the method for non-linear system identification.

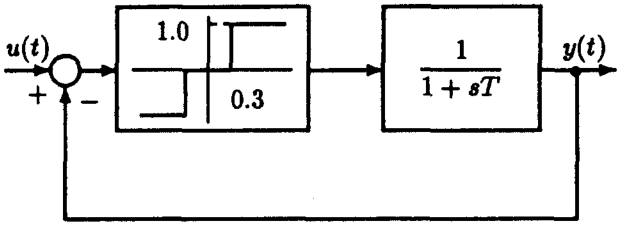

Fig 3: A non-linear system having feedback loop

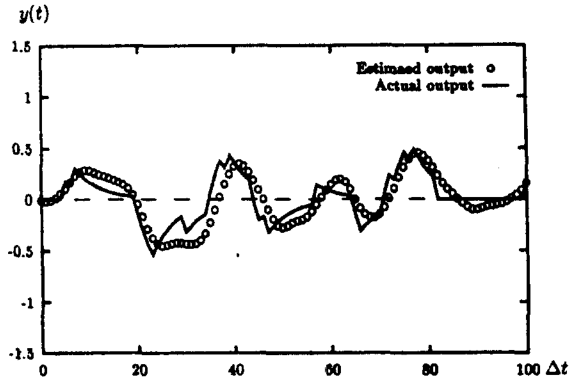

Fig 4: Comparison of the actual output with the estimated one for Fig.3

6. CONCLUSION

A method for obtaining Volterra kernels of non-linear system by use of pseudorandom M-sequence is further developed to make possible the adaptive and real time identification of Volterra kernels.

The obtained Volterra kernels are updated in a prescribed time to obtain the new information on the system. Thus we can obtain all crosssections of Volterra kernels up to 3rd order in adaptive way.

This method for obtaining Volterra kernel is simulated on the computer for non-linear systems having up to third order Volterra kernels. The results of simulation show a good agreement with the theoretical considerations. The method is also applied to some actual non-linear system having dead-band type non-linear element.The results show that this method is effective for non-linear system identification.

REFERENCES

1) Barker,H.A.*et al* (1972) Performance of antisymmetric pseudorandom signal in the measurement of second-order Volterra kernels by crosscorrelation, Proc. IEE, **119**, 3, 353-362

2) Kashiwagi,H. and T.Moriuchi (1981) A Fast Method for Arithmetic Modulo a Polynomial over GF(2) and its Applications, Proc. 8'th IFAC held in Kyoto, Japan,pp.II-124-129

3) Kashiwagi,H., Sun Yeping and E. Nishiyama (1993a) Identification of Volterra Kernels of Nonlinear Systems by use of M-sequence, Proc. 1993 KACC held in Seoul, Korea, 150-154

4) Kashiwagi,H., Sun Yeping and E. Nishiyama (1993b) Identification of 2nd and 3rd Volterra Kernels of Non-linear Systems, Proc. '93 APCCM held in Kunming, China, 15-19

5) Kashiwagi,H. and Sun Yeping (1995a) A Method for Identifying Volterra Kernels of Nonlinear System, Trans. SICE, **31**, 8, 1054-1060

6) Sun Yeping and Kashiwagi,H. (1995b) Comparison of Volterra Kernel Method with Describing Function for Nonlinear System Identification, Proc. APCCM'95 held in Wuhan, China, 28-31

7) Sun Yeping and Kashiwagi,H. (1995c) A Real Time Method for Non-linear System Identification, Proc. ICAUTO'95 hold in Indore, India in 1995

Table 3: Power of x in $x^{k_{i,2}}$ mod $f(x)$, when $f(x)$=1515155

k_{i2}	Power of x in $x^{k_{i2}}$ mod $f(x)$											
253040	17	8	1	16	10	4	14	11	5	2	0	
243937	16	13	10	7	4	1	15	9	3			
256104	17	11	15	9	5	16	12	10	6	4	3	1
225731	17	15	9	3	1							
97643	17	4	1	10	11	5						
250065	17	2	14	11	8	5	9	3	1			
159833	17	14	13	11	7	5						
189319	17	8	13	1	14	12	11	6	5	2	0	
124214	16	11	5	13	12	10	7	4	1			
195286	17	15	9	14	8	5	2	16	12	6	3	1

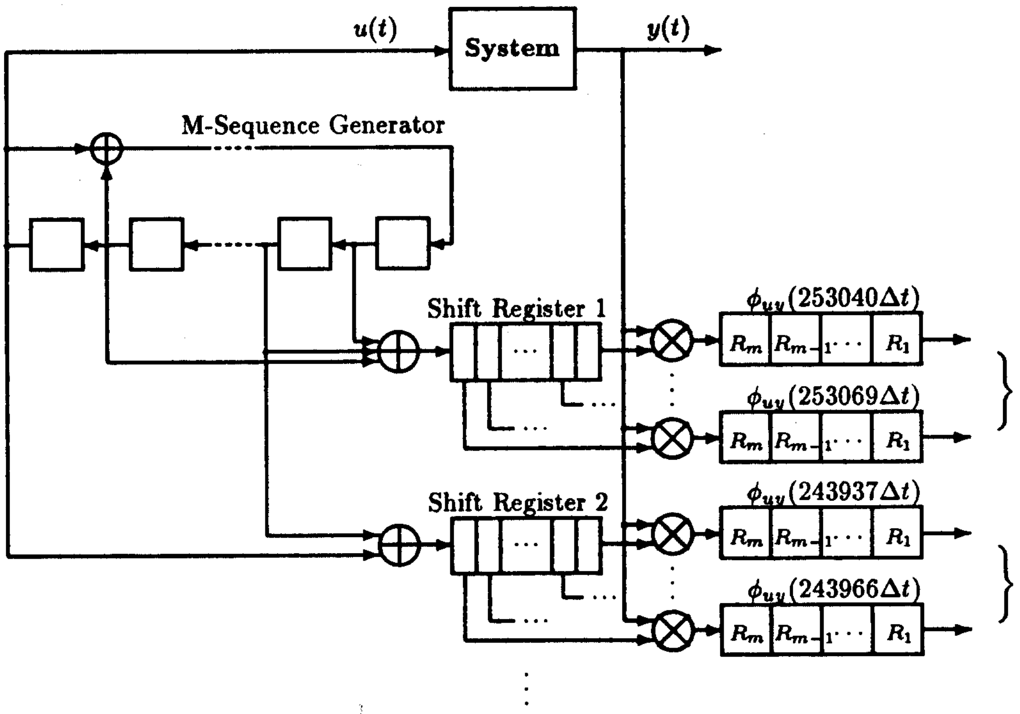

Fig 2: Schematic diagram for real time identification of non-linear system