PLS를 이용한 AR, ARMA 모델의 차수 결정을 위한 격자 알고리즘

○김현표* 정찬수** 양흥석*

* 서울대 전기과 ** 숭실대 전기과

Lattice Algorithms for Order Determination of AR,ARMA Models by PLS

Kim Hyun-pyo* Chung Chan-soo** Yang Heung-suk*

* Dept. of Electrical Eng. Seoul National Univ. ** Dept. of Electrical Eng. Soongsil Univ.

ABSTRACT

In this paper, a new method for determining the order of AR, ARMA processes based on PLS (predictive least square) principle is proposed. This method using modified lattice algorithm which has additional step is amenable to on-line or adaptive operation and is more accurate than any other method.

Some computer simulations are presented to show the efficiency of the proposed algorithms.

I. 서 론

시스템의 제어, 음성 및 영상 신호처리, 스펙트럼 추정에 이르기까지 주어진 시스템의 관측된 데이타로부터 시스템의 특성을 분석하는 과정이 필요하다. 이러한 과정은 시스템의 특성을 가장 잘 표현하는 모델의 차수 (order)를 결정하는 문제와 적당한 계수 추정방법을 선택하여 모델링한 시스템의 계수를 추정하는 문제로 구분할 수 있다. 차수 결정은 계수를 추정하기 이전에 선행되어야 할 문제로서 기존의 차수 결정방법에는 Akaike[1]에 의해 제안된 AIC (Akaike information criterion), Rissanen[2]등에 의해 제안된 MDL (minimum description length), Wellstead [3]등의 IDR (instrumental determinant ratio), Chow[4]등의 상관행렬의 특이성 (singularity)을 검사하는 방법 등이 있다. 기존의 방법들은 시행착오에 근거한 방법들이 많으며 주로 점근적 결과 (asymptotic results) 에 근거하고 있으므로 많은 데이타가 필요하며 따라서 많은 양의 계산을 필요로 한다. 또한 가기 consistency, 정확성의 문제, 온-라인으로 변경할 수 있는 제약을 안고 있다. [5, 6]

최근에는 기존의 방법들의 문제점을 해결하기 위한 방안의 하나로 Rissanen에 의해 연구된 PLS (predictive least square) principle을 이용하여 AR 모델의 차수를 결정하는 방법이 Wax[6]에 의해 제안되었다. 수리적으로 좋은 특성을 가지고 하드 웨어 구현에도 유리한 격자 (lattice) 형태의 알고리즘을 이용한 이 방법은 계산상의 간편성이 있는 전치 윈도우 방식 (pre-windowed type) 을 사용하고 있다.

본 논문에서는 AR 모델의 경우, 윈도우 밖의 데이타에 대한 가정이 완화된 공분산 방식 (covariance type) 의 격자 알고리즘[7-12]을 변형하여 적은 양의 데이타로도 기존의 방식들보다 정확히 모델의 차수를 결정할 수 있는 방법, 그리고 ARMA 모델의 경우 2 채널 AR모델로 변환시켜[10] 모델의 차수를 결정할 수 있는 방법을 제안한다.

II절에서는 PLS principle에 대한 내용을 살펴 보고, III절에서는 AR 모델의 경우, 공분산 방식의 격자

제시하고, IV절에서는 ARMA 모델에 대한 차수 결정 알고리즘을 제시하고, V절에서는 시뮬레이션 결과를, 그리고 VI절에서는 결론을 싣는다.

II. PLS principle

다음과 같은 이산계 프로세스 $\{ y(t),\ 0 \le t \le N-1 \}$ 을 고려하자. 과거의 데이타 $\{ y(0), \cdots, y(t-1) \}$ 로 부터 $y(t)$를 예측했을때 차수 k인 AR 모델의 실 예측 오차 (true prediction error)는

$$\hat{e}(t \mid k) = y(t) - \sum_{i=1}^{k} \hat{a}(t-1,k,i)\, y(t-i) \qquad (1)$$

여기에서 $\{ \hat{a}(t-1,k,i) \}$는 오차의 제곱을 최소화하는 예측자의 계수이다.

차수 k인 PLS criterion 을 다음의 식 (2)와 같이 나타낼때

$$PLS(k) = \frac{1}{N} \sum_{t=1}^{N} \hat{e}^{2}(t \mid k) \qquad (2)$$

모델의 최대 가능 차수를 M이라 하면

$$\min_{k \le M} PLS(k) \qquad (3)$$

식 (3)을 만족하는 k 의 값이 모델의 차수로 결정된다. [6]

모델의 차수를 결정하기 위해서는 매 시간, 매 차수에서의 실 예측 오차가 계산되어야 한다. 그러기 위해서는 식 (1)에서 보여지듯이 예측자의 계수에 대한 식 (4)와 같은 계산이 먼저 필요.하다.

$$\underline{\hat{a}}(t-1,k) = \left(\sum_{i=0}^{t-1} \underline{y}(i-1,k)\, \underline{y}^{T}(i-1,k) \right)^{-1} \sum_{i=0}^{t-1} \underline{y}(i-1,k)\, y(i) \qquad (4)$$

여기에서

$$\underline{\hat{a}}(t-1,k) = [\, \hat{a}(t-1,k,1), \cdots, \hat{a}(t-1,k,k)]^{T}$$

$$\underline{y}(i-1,k) = [\, y(i-1), \cdots, y(i-k)]^{T}$$

이와 같은 계산과정은 III절에서 언급할 격자 필터 (lattice filter) [7-12]를 이용함으로써 효율적인 계산이 가능해진다.

III. AR 모델의 차수 결정

이산계 프로세스 $\{ y(0), \cdots, y(N-1) \}$ 주어졌을때 n 차 전향 예측오차 (forward prediction error) 및 후향 예측 오차 (backward prediction error) 는 각각 다음과 같이 표현된다.

$$e(t \mid n) = y(t) - \sum_{i=1}^{n} a(t,n,i)\, y(t-i) \qquad (5)$$

$$r(t \mid n) = y(t-n) - \sum_{i=0}^{n-1} b(t,n,i+1)\, y(t-i) \qquad (6)$$

t 시간 n 차의 전향 및 후향 예측오차는 격자 필터의 경우, (t-1) 시간, (n-1) 차의 값만으로 update 가 가능하다. 따라서 데이타가 증가함에 따라 식 (4)의 계수에 대한 매 시간, 매 차수에서의 계산이 필요없이 격자필터에서 update 되는 예측오차로 부터 식 (1)로 주어지는 실 예측오차에 대한 효율적인 계산이 가능해진다. [9, 13]

기존의 차수 결정 방식들이 주로 데이타가 증가될때마다 증가된 데이타를 포함한 기존의 데이타 전체에 대하여 새로이 차수결정을 위한 계산을 해야 하는 batch 처리 방식인데 비하여 격자 알고리즘을 이용할 경우 새로운 데이타의 증가에 따라 단지 과거의 데이타 전체에 대한 정보가 있는 전 시간 , 전 차수에 대한 값만으로 예측 오차의 update 및 차수의 결정이 이루어지므로 온-라인이 가능해지며 데이타의 증가에 추가되는 계산량을 현저히 줄일 수 있다.

본 논문에서 제안한 공분산 방식의 차수 결정 격자 알고리즘은 윈도우 밖의 데이타에 대한 가정이 없기 때문에 Wax[6]가 제안한 전치 윈도우 방식에서의 윈도우 밖의 데이타의 가정 때문에 발생되는 불필요한 end effects 를 없앨 수 있다. 따라서 AR 모델의 경우, 전치 윈도우 방식에 비해 약간의 계산량을 증가시키더라도 본 논문에서 제안한 공분산 방식의 차수 결정 격자 알고리즘이 적은 데이타로도 보다 정확히 차수를 결정할 수 있다. 이때의 알고리즘이 표 1.에 있다.

표 1. 공분산방식의 차수 결정 격자알고리즘

Table 1. Order determination lattice algorithm of covariance type

for $t=0$ to $N-1$ do

$$e^{*}(0 \mid n) = \begin{cases} y(0), & \text{for } n=0 \\ 0, & \text{for } n>0 \end{cases}$$

$$e(t \mid 0) = r(t \mid 0) = y(t)$$

$$R^{e}(t \mid 0) = R^{r}(t \mid 0) = R^{e}(t-1 \mid 0) + y^{2}(t)$$

$$\gamma(t \mid -1) = 0$$

for $n=1$ to $\min(t, M)$ do

$$\Delta(t \mid n) = \Delta(t-1 \mid n) + \frac{e(1,t \mid n-1)\, r(t-1 \mid n-1)}{1 - \gamma(1, t-1 \mid n-1)}$$

$$K^{e}(t \mid n) = \Delta(t \mid n) \,/\, R^{e}(1, t \mid n-1)$$

$$K^{r}(t \mid n) = \Delta(t \mid n) \,/\, R^{r}(t-1 \mid n-1)$$

$$e(t \mid n) = e(1, t \mid n-1) - K^{r}(t \mid n)\, r(t-1 \mid n-1)$$

$$r(t \mid n) = r(t-1 \mid n-1) - K^{e}(t \mid n)\, e(1, t \mid n-1)$$

$$R^{e}(t \mid n) = R^{e}(1, t \mid n-1) - K^{r}(t \mid n)\, \Delta(t \mid n)$$

$$R^{r}(t \mid n) = R^{r}(t-1 \mid n-1) - K^{e}(t \mid n)\, \Delta(t \mid n)$$

$$e^{*}(t \mid n) = e^{*}(t \mid n) - \frac{e(t \mid n)\, \alpha(t-1 \mid n)}{1 - \gamma(t-1 \mid n)}$$

$$e(1, t \mid n) = e(t \mid n) + \frac{e^{*}(t \mid n)\, \alpha(t-1 \mid n)}{1 - \gamma^{*}(t-1 \mid n)}$$

$$R^{e}(1, t \mid n) = R^{e}(t \mid n) - \frac{e^{*2}(t \mid n)}{1 - \gamma^{*}(t-1 \mid n)}$$

$$\hat{e}(t \mid n) = e(t \mid n) \,/\, (1 - \gamma(t-1 \mid n))$$

if $n < N$ then

$$\gamma(t \mid n+1) = \gamma(t-1 \mid n) + e^{2}(t \mid n) / R^{e}(t \mid n)$$

$$\gamma^{*}(t \mid n+1) = \gamma^{*}(t-1 \mid n) + e^{*2}(t \mid n) / R^{e}(t \mid n)$$

$$\alpha(t \mid n+1) = \alpha(t-1 \mid n) + \frac{e^{*}(t \mid n)\, e(t \mid n)}{R^{e}(t \mid n)}$$

$$\gamma(1, t \mid n) = \gamma(t \mid n+1) - r^{2}(t \mid n) / R^{r}(t \mid n)$$

$$\text{determine } k : \min_{k \le M} \frac{1}{N} \sum_{t=0}^{N-1} \hat{e}(t \mid k)$$

IV. ARMA 모델의 차수 결정

편의상 $y(t)$, $u(t)$ 가 scalar 이고 AR, MA 의 차수가 같은 다음과 같은 ARMA 모델을 고려하자.

$$y(t) = \sum_{i=1}^{n} a_i\, y(t-i) + \sum_{i=0}^{n} b_i\, u(t-i) \tag{7}$$

벡터 프로세스 $w(t)$ 를 식 (8)과 같이 잡으면

$$w(t) = \begin{bmatrix} y(t) \\ u(t) \end{bmatrix} \tag{8}$$

식 (7)과 같은 ARMA 모델을 2 채널의 AR 모델로 변환할 수 있다. [10, 11]

이 경우 주어진 시스템의 입력 $u(t)$ 를 알 수 있는 시스템 재어 문제의 경우에는 필터의 입력을 $y(t)$, $u(t)$ 2 채널로 하여 AR 차수 결정 격자 알고리즘을 이용할 수 있다. 여기에서 복잡한 행렬 연산의 번거로움을 줄이기 위해 계산량이 다소 적은 전치 윈도우 방식의 차수 결정 격자 알고리즘을 표 2.에 제시하였다.

표 2. 2 채널 AR 격자필터를 이용한 ARMA 모델의 차수 결정 알고리즘 (전치 윈도우 방식)

Table 2. Order determination algorithm of ARMA model using 2 channel AR lattice filter (pre-windowed type)

for $t=0$ to $N-1$ do

$$e(t \mid 0) = r(t \mid 0) = w(t) = \begin{bmatrix} y(t) \\ u(t) \end{bmatrix}$$

$$R^{e}(t \mid 0) = R^{r}(t \mid 0) = \lambda R^{e}(t-1 \mid 0) + w(t) w(t)^{T}$$

$$\gamma(t \mid -1) = 0$$

for $n=0$ to $\min(t, M)-1$ do

$$\Delta(t \mid n+1) = \Delta(t-1 \mid n+1) + r(t-1 \mid n)\, e^{T}(t \mid n) \cdot \frac{1}{1 - \gamma(t-1 \mid n-1)}$$

$$\gamma(t \mid n) = \gamma(t \mid n-1) + r^{T}(t \mid n)\, R^{-r}(t \mid n)\, r(t \mid n)$$

$$K^{e}(t \mid n+1) = \Delta(t \mid n+1)\, R^{-e}(t \mid n)$$

$$K^{r}(t \mid n+1) = \Delta^{T}(t \mid n+1)\, R^{-r}(t-1 \mid n)$$

$$e(t \mid n+1) = e(t \mid n) - K^{r}(t \mid n+1)\, r(t-1 \mid n)$$

$$r(t \mid n+1) = r(t-1 \mid n) - K^{e}(t \mid n+1)\, e(t \mid n)$$

$$R^{e}(t \mid n+1) = R^{e}(t \mid n) - K^{r}(t \mid n+1)\, \Delta(t \mid n+1)$$

$$R^{r}(t \mid n+1) = R^{r}(t-1 \mid n) - K^{e}(t \mid n+1)\, \Delta^{T}(t \mid n+1)$$

$$\hat{e}(t \mid n+1) = \frac{| e^{T}(t \mid n+1)\, e(t \mid n+1) |^{\frac{1}{2}}}{1 - \gamma(t-1 \mid n)}$$

$$\text{determine } k : \min_{k \le M} \frac{1}{N} \sum_{i=0}^{N-1} \hat{e}(t \mid k)$$

반면에 신호 처리 문제의 경우에는 $u(t)$를 알 수 없고 다지 주어진 데이타 시퀀스를 가상 입력 (hypothetical input)인 백색잡음으로 생성되는 ARMA 모델로 가정하고 그 차수를 결정하는 것이므로 필터의 입력에 필요한 $u(t)$에 대한 추정과정이 필요하다. $u(t)$에 대한 추정방법의 하나인 부트 스트래핑 (bootstrapping) 방식을 사용한 경우의 알고리즘을 표 3.에 제시하고 있다.

표 3. 가상 입력 추정을 통한 ARMA 모델의 차수결정 알고리즘

Table 3. Order determination algorithm of ARMA model by estimating hypothetical input

for $t=0$ to $N-1$ do

$$e(t|0) = r(t|0) = \mathbf{w}(t) = \begin{bmatrix} y(t) \\ 0 \end{bmatrix}$$

$$R^{e}(t|0) = R^{r}(t|0) = \lambda R^{e}(t-1|0) + \mathbf{w}(t)\mathbf{w}(t)^{T}$$

$$\gamma(t|-1) = 0$$

for $n=0$ to $M-1$ do

(표 2.의 n-loop 와 동일)

$$\hat{u}(t) = R^{-e(\psi)/2}(t,M)\, e^{\psi}(t,M)$$

$$e(t|0) = r(t|0) = \mathbf{w}(t) = \begin{bmatrix} y(t) \\ \hat{u}(t) \end{bmatrix}$$

for $n=0$ to $\min(t,M)-1$ do

(표 2.의 n-loop와 동일)

determine k : $\min\limits_{k \le M} \frac{1}{N} \sum\limits_{i=0}^{N-1} \hat{e}(t|k)$

V. 시뮬레이션 결과

< 예 1 >

$$y(t) = 0.93y(t-1)+e(t)$$

< 예 2 >

$$y(t) = 1.8y(t-1)-0.97y(t-2)+e(t)$$

< 예 3 >

$$y(t) = 2.2y(t-1)-1.86y(t-2)+0.592y(t-3)+e(t)$$

< 예 4 >

$$y(t) = 1.8y(t-1)-0.97y(t-2) +0.8u(t-1)-0.5u(t-2)+e(t)$$

$$u(t) = 4\sin(0.5t)+1.2\cos(0.8t)$$

$e(t)$: 백색 잡음

1 차, 2 차, 3 차 AR 모델에 대한 AIC, 전치윈도우 방식, 공분산 방식의 차수 결정 격자 알고리즘에 의한 100번의 Monte Carlo 시뮬레이션 결과가 각각 표 4, 5, 6 에 있다. 결과에서 보여지듯이 공분산 방식이 타 방식들에 비해 적은 데이타로도 더 정확히 차수를 결정함을 알 수 있다. <예 4>의 시뮬레이션 결과는 표 7 에 있다.

VI. 결 론

본 논문에서는 AR, ARMA 모델에 대한 차수 결정에 관한 문제를 PLS criterion 및 계산상의 효율성이 있는 격자 알고리즘을 이용하여 기존의 방식들에 비해 보다 정확히 모델의 차수를 결정할 수 있는 방법을 제시하였다. AR 모델의 경우에는 공분산 방식의 격자 알고리즘을 이용하여 end effects 없이 적은 데이타로도 정확한 차수의 결정이 가능해졌으며, ARMA 모델의 경우 2 채널 AR 모델로의 변환을 통하여 전치 윈도우 방식의 AR 차수 결정 격자 알고리즘을 ARMA 모델로 확장시켰다.

참 고 문 헌

[1] H. Akaike, "A new look at the statistical model identification," IEEE Trans. Automat. Contr., Vol.AC-19, pp.716-723, 1974

[2] J. Rissanen , " Modeling by shortest data description," Atomatica, Vol.14, pp.465-471, 1978

[3] P. E. Wellstead, "An instrumental Product Moment Test for model order estimation," Automatica, Vol.14, pp.89-91, 1978

[4] J. C. Chow, "On estimating the orders of an autoregressive moving-average process with uncertain observations," IEEE Trans. Automat. Contr., Vol.AC-17, pp.707-709, 1972

[5] S. Haykin, Adaptive filter theory, Prentice-Hall, New Jersey, 1986

[6] M. Wax, "Order selection for AR models by predictive least squares," IEEE Trans. Acoust., Speech and Signal Processing, Vol.ASSP-36, pp.581-588, 1988

[7] B. Friedlander, "Lattice filters for adaptive processing," Proc. IEEE, Vol.70, pp.829-867, 1982

[8] B. Friedlander, "Lattice methods for spectral estimation," Proc. IEEE, Vol.70, pp.990-1017, 1982

[9] M. L. Honig and D. G. Messerschmitt, Adaptive filters:structures, algorithms and applications, Kluwer Academic Publishers, 1984

[10] D. T. L. Lee, "Canonical ladder form realizations and fast estimation algorithms," Ph. D. dissertation, Dept. Elec. Eng., Stanford Univ., Stanford, CA., Aug. 1980

[11] D. T. L. Lee, B. Friedlander and M. Morf, "Recursive ladder algorithms for ARMA Modeling," IEEE Trans. Automat. Contr., Vol.AC-27, pp.753-764, 1982

[12] D. T. L. Lee, M. Morf and B. Friedlander, "Recursive least squares ladder estimation algorithms," IEEE Trans. Circuits and Systems, Vol.CAS-28, pp.467-481, 1981

[13] J. M. Cioffi and T. Kailath, "Fast, recursive least squares transversal filters for adaptive filtering," IEEE Trans. Acoust., Speech and Signal Processing, Vol.ASSP-32, pp.304-337, 1984

표 4. < 예 1 > 의 시뮬레이션 결과

Table 4. Simulation results of example 1

k \ N	A I C		PLS < P>		PLS < C>	
	30	50	30	50	30	50
1	64	83	78	87	81	91
2	14	10	14	8	15	8
3	1	4	5	2	4	1
4	1	0	2	1	0	0
5	5	0	0	2	0	0
6	0	0	1	0	0	0
7	0	1	0	0	0	0
8	4	1	0	0	0	0
9	6	0	0	0	0	0
10	5	1	0	0	0	0

표 5. < 예 2 > 의 시뮬레이션 결과

Table 5. Simlation results of example 2

k \ N	A I C		PLS < P>		PLS < C>	
	50	100	50	100	50	100
1	0	0	0	0	0	0
2	82	83	85	91	94	96
3	9	9	9	6	6	4
4	1	1	3	3	0	0
5	3	3	2	0	0	0
6	0	1	1	0	0	0
7	2	1	0	0	0	0
8	0	2	0	0	0	0
9	1	0	0	0	0	0
10	2	0	0	0	0	0

표 6. < 예 3 >의 시뮬레이션 결과

Table 6. Simulation results of example 3

k \ N	A I C		PLS < P>		PLS < C>	
	50	100	50	100	50	100
1	0	0	0	0	0	0
2	1	0	3	0	8	0
3	81	82	87	92	92	99
4	12	10	7	8	0	1
5	2	4	2	0	0	0
6	1	1	1	0	0	0
7	1	0	0	0	0	0
8	0	3	0	0	0	0
9	0	0	0	0	0	0
10	2	0	0	0	0	0

표 7. < 예 4 > 의 시뮬레이션 결과

Table 7. Simulation results of example 4

k \ N	PLS < P >	
	50	100
1	0	0
2	89	92
3	9	8
4	2	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0