論 文
38～3～10

슬라이딩 메모리 공분산형 환상 격자 필터 및 ARMA 모델링에의 응용

A Slidig Memory Covariance Circular Lattice Filter and Its Application to ARMA Modeling

張 英 秀*·李 哲 熙**·梁 興 錫§
(Young-Soo Jang·Chul-Heui Lee·Heung-Suk Yang)

요 약

본 논문에서는 슬라이딩 메모리 공분산형 환상 격자(sliding memory covariance circular lattice : SMC-CL)필터와 이를 이용한 효과적인 ARMA 모델링 알고리즘을 제시하였다. 먼저 일반적인 경우에 대해 기하적인 방법을 사용하여 SMC-CL필터를 유도한 뒤, ARMA 프로세스를 2채널 AR 프로세스로 변환하여 이에 SMC-CL 필터를 적용하였다. ARMA 모델링의 경우 구동입력 프로세스의 백색성때문에 SMC-CL필터의 구조가 더욱 간단해진다. 이때 ARMA 프로세스의 매개변수는 필터와 둘째 채널의 PARCOR 계수로부터 Levinson 순환식을 이용하여 구할수 있다. 컴퓨터 시뮬레이션을 통하여 제안된 알고리즘의 유효성을 보였다.

Abstract- A sliding memory covariance circular lattice(SMC-CL) filter and an efficient ARMA modeling method using the SMC-CL filter are presented. At first, SMC-CL filter is derived based on the geometric approach. Then ARMA process is converted into 2 channel AR process, and SMC-CL filter is applied to it. The structure of SMC-CL filter becomes simpler in case of ARMA modeling due to the whiteness of a driving input process. The parameters of ARMA process can be obtained by the Levinson recursions from the PARCOR coefficients of the second channel of the filter. Computer simulations are performed to show the effectiveness of the proposed algorithm.

1. Introduction

Nowadays AR and ARMA modelings are widely used in various areas of statistics, geophysics, signal processing and system theory. Often the dynamics and the characteristics of a real system can be properly described by an ARMA model, so ARMA modeling gives better performances than AR modeling. But AR modeling has been actively used in practice, while the use of ARMA modeling is somewhat limited due to the complexity of the model fit-

*正 會 員 : 서울大 工大 電氣工學科 博士課程
**正 會 員 : 서울大 大學院 電氣工學科 · 工博
§正 會 員 : 基礎電力 共同研究所 所長 · 工博
接受日字 : 1988年 10月 6日
1次修正 : 1988年 12月 1日
2次修正 : 1989年 1月20日

ting algorithms.[1]

However, some of the recent works provide relatively efficient ARMA modeling techniques. Particularly, fast algorithms for on-line ARMA modeling have been developed and they have the advantages of the computational efficiency and the good numerical properties.[2)~6)]

One of the popular techniques for on-line ARMA modeling with the fast algorithms is to convert ARMA process into a two channel AR process.In that case the hypothetical input estimation procedure is necessary because a driving white process, which is really unknown, has to be applied to the parameter estimator as input together with the observations. To overcome this difficulty the bootstrapping procedure is usually taken, but it requires considerable amount of the calculation burden in addition.[4),5)]

Lately circular lattice(CL) filter was proposed by Sakai for the prewindowed data set.[8),9)] He also showed that it can be a good alternative for ARMA modeling.[10)] Since the data flows between each channel of CL filter forms a circle, matrix operations accompanying usual multichannel processing algorithms are replaced with only simple scalar operations, and therefore the computational effort is fairly reduced.[8),9)] Of course, CL filter also preserves the advantages of lattice type algorithms.

Lattice algorithms are classified into four groups by choosing the type of the window on the data set, i.e., the autocorrelation form, the prewindowed form, the postwindowed form, and the covariance form. So far most of the works on lattice type algorithms have been done for the prewindowed form by virtue of its simplicity. But the covariance form is considered as superior to the others from the viewpoint of the accuracy, because it takes no assumption on the value of the data outside the given observations, thereby eliminating any undesirable end effects.[3),6)]

In this paper, a sliding memory covariance circular lattice(SMC-CL) filter and an efficient ARMA modeling method using SMC-CL filter are presented. First we derive SMC-CL filter using the projection operator. And then SMC-CL filter is applied to ARMA modeling after the ARMA process is converted into a two channel AR process.

2. Circular lattice filter

Here we give a brief review of CL filter. CL filter is based on Pagano's work[7)], the one to one relationship between multivariate autoregressions and scalar periodic autoregressions.

Let us consider the following d-variate p-th order AR process

$$X(t) = -\sum_{j=1}^{p} A_j X(t-j) = U(t) \qquad (2.1)$$

where $\{U(\cdot)\}$ is an uncorrelated vector sequence with zero mean and $\mathrm{cov}[U(t)]=W$. Also let us define a scalar process $Y(t)$ from $X(t)$ by

$$Y(i+td) = X_i(t) \qquad (2.2)$$

where $x_i(t)$ is i-th element of $X(t)$.

From Pagano's theorem[7)], the relation (2.2) is satisfied if $Y(t)$ is a scalar periodic AR process of period d. Thus Sakai derived CL filter by applying the technique of linear prediction to this scalar process $Y(t)$, instead of treating $X(t)$ directly.[8)] As a result, matrix operations are avoided.

Let us express the j-th order i-th channel forward and backward prediction errors by

$$\varepsilon_i^j(t) = Y(i+td) + \sum_{k=1}^{j} c_i^j(k) Y(i+td-k) \qquad (2.3)$$

$$\eta_i^j(t) = Y(i+td-j) + \sum_{k=1}^{j} d_i^j(j+1-k) Y(i+td-k+1) \qquad (2.4)$$

where $\{c_i^j(k)\}$, $\{d_i^j(k)\}$ are the coefficients of the i-th channel of the forward and backward linear predictors of order j. These predictor coefficients $\{c_i^j(k)\}$, $\{d_i^j(k)\}$ are determined so as to minimize $E\{(\epsilon_i^j(t)^2\}$, $E\{(\eta_i^j(t))^2\}$, respectively. Then we obtain the circular lattice structure

$$\varepsilon_i^{j+1}(t) = \varepsilon_i^j(t) + \alpha_i^{j+1} \eta_{i-1}^j(t) \qquad (2.5)$$

$$\eta_i^{j+1}(t) = \eta_{i-1}^j(t) + \beta_i^{j+1} \varepsilon_i^j(t) \qquad (2.6)$$

where the PARCOR coefficients $\alpha_i^{j+1}(=c_i^{j+1}(j+1))$, $\beta_i^{j+1}(j+1))$ are given by

$$\alpha_i^{j+1} = -E\{\varepsilon_i^j(t)\eta_{i-1}^j(t)\}/E\{(\eta_{i-1}^j(t))^2\} \quad (2.7)$$

$$\beta_i^{j+1} = -E\{\varepsilon_i^j(t)\eta_{i-1}^j(t)\}/E\{(\varepsilon_i^j(t))^2\} \quad (2.8)$$

However, (2.7) and (2.8) cannot be used in practical situation because the value of expectation is not given. So it is necessary to estimate α_i^{j+1}, β_i^{j+1} based on the given data sequence in least squares sense.

In the next section, we present a recursive least squares SMC-CL filter.

3. Sliding memory covariance circular lattice filter

The covariance type algorithms obtain an estimate based only upon available data, thereby excluding the end effects caused by assuming the data is zero outside the observation interval, as is the case in the prewindowed type filters. Thus they are often used where an accurate estimate is desired given relatively short data.

So a sliding memory covariance circular lattice filter is presented in this section. Unlike the growing memory covariance type algorithms, the sliding memory covariance type algorithms compute least squares estimates based on the data contained in a window of fixed length which slides across the sequence of data one by one. Past data samples outside the window are therefore totally forgotten. As a result, it has the ability to track the time varying parameters of nonstationary process.

Here the geometric approach is employed to derive the SMC-CL filter, which is an efficient tool for the derivation of lattice type recursive least squares algorithms. By the projection theorem, the least squares estimation can be interpreted as the orthogonal projection of a new data vector on the subspace spanned by the past data vectors.[3]

Let us consider a d-variate AR process $X(t)$. Assuming a window of length M, at time t ket and bra vectors are defined respectively by

$$|x_i>_t \triangleq [x_i(t-M+1)\cdots\cdots x_i(t)] \quad (3.1)$$
$$\langle x_i|_t \triangleq |x_i\rangle_t'$$

where $|x_i\rangle_t'$ denotes the transpose of $|x_i\rangle_t$, and $|x_i\rangle_t$ lies in Hilbert space $H_t(=R^M)$. Also a shift operator s^{-1} is defined by

$$|s^{-1}x_1>_t = |x_d>_{t-1}$$
$$|s^{-1}x_i>_t = |x_{i-1}>_t,\ i=2,3,\cdots,d \quad (3.2)$$

Let $|Y_{i,1,n}\rangle_t$ be a data matrix of dimension n related to i-th channel, and $Y_{i,1,n}\rangle_t$. That is,

$$|Y_{i,1,n}>_t = [|s^{-1}x_i>_t,\ \cdots\cdots,\ |s^{-n}x_i>_t]$$
$$Y_{i,1,n,t} = \text{subspace spanned by}$$
$$|s^{-1}x_i>_t,\ \cdots,\ |s^{-n}x_i>_t \quad (3.3)$$

From (2.2) and (2.3), the prediction error vector of $|x_i\rangle_t$ is given by

$$|\varepsilon_i^n>_t = |x_i>_t + |Y_{i,1,n}>_t <I\ |c_i^n>_t \quad (3.4)$$

where $c_i^j>_t$ is the predictor coefficient vector defined by

$$|c_i^n> \triangleq [c_i^n(1,t),\ c_i^n(2,t),\ \cdots,\ c_i^n(n,t)]' \quad (3.5)$$

Each channel of p-th order SMC-CL filter minimizes the sum of squares of prediction errors defined by

$$\sigma_i^n(t) \triangleq \sum_{k=t-M+1}^{t} (\varepsilon_i^n(k))^2 = <\varepsilon_i^n|\varepsilon_i^n>_t \quad (3.6)$$

Then the least squares solution of predictor coefficients is obtained by

$$|c_i^n>_t = -<Y_{i,1,n}|Y_{i,1,n}>_t^{-1}<Y_{i,1,n}|x_i>_t \quad (3.7)$$

Substituting (3.7) into (3.4),

$$|\varepsilon_i^n>_t = |x_i>_t = |Y_{i,1,n}>_t<Y_{i,1,n}|Y_{i,1,n}>_t^{-1} <Y_{i,1,n}|x_i>_t$$
$$= (I - |Y_{i,1,n}><Y_{i,1,n}|Y_{i,1,n}>_t^{-1} <Y_{i,1,n}|_t)\ |x_i>_t \quad (3.8)$$

Thus the projection operator $P_{Yi,1,n,t}$ and the orthogonal projection operator $P_{Yi,1,n,t}^{\perp}$ are defined as follows.

$$P_{Y_{i,1,n,t}} \triangleq |Y_{i,1,n}>_t <Y_{i,1,n}|Y_{i,1,n}|_t \quad (3.9)$$

$$P^{\perp}_{Y_{i,1,n,t}} \triangleq I - |Y_{i,1,n}>_t <Y_{i,1,n}|Y_{i,1,n}>_t^{-1} <Y_{i,1,n}|_t \quad (3.10)$$

These operators satisfy the symmetry and the idempotence.

$$P \cdot P = P, \quad P' = P$$
$$P^{\perp} \cdot P^{\perp} = P^{\perp}, \quad P^{\perp\prime} = P^{\perp} \quad (3.11)$$

In the same manner, the backward prediction error vector of $|s^{-n}x_i\rangle_t$ is given by

$$|\eta_i^n>_t = P^{\perp}_{Y_{i,0,n-1,t}} |s^{-n}x_i>_t \quad (2.12)$$

Here we introduce the following two pinning vectors acting as time annihilators:

$$|\pi>_t \triangleq [\, 0 \cdots\cdots 0 \ 1 \,]' \quad (3.13)$$

$$|\pi^*>_t \triangleq [\, 1, \ 0 \cdots\cdots 0 \,]' \quad (3.14)$$

Several definitions of variables essential to the derivation of the algorithm are presented in table 1.

$$Y_{i,1,n+1,t} = Y_{i,1,n,t} \oplus P_{Y_{i,1,n,t}} |s^{-n-1}x_i>_t$$
$$= Y_{i,1,n,t} \oplus |\eta_{i-1}^n>_t \quad (3.24)$$

$$Y_{i,0,n,t} = Y_{i,n,t} \oplus P_{Y_{i,1,n,t}} |x_i>_t$$
$$= Y_{i,1,n,t} \oplus |\varepsilon_i^n>_t \quad (3.25)$$

where $\oplus$ is the direct sum. Hence, from (3.8), (3.12), (3.24) and (3.25), the order update formulas for forward and backward prediction errors are given by

$$|\varepsilon_i^{n+1}>_t = |\varepsilon_i^n>_t - |\eta_{i-1}^n>_t <\eta_{i-1}^n|\eta_{i-1}^n>_t^{-1} <\eta_{i-1}^n|\varepsilon_i^n>_t \quad (3.26)$$

$$|\eta_i^{n+1}>_t = |\eta_{i-1}^n>_t - |\varepsilon_i^n>_t <\varepsilon_i^n|\varepsilon_i^n>_t^{-1} <\varepsilon_i^n|\eta_{i-1}^n>_t \quad (3.27)$$

In the above recursions, we need $\eta_0^n>_t$ for i=1. From (3.2) and (3.12)

$$|\eta_0^n>_t = |\eta_d^n>_{t-1} \quad (3.28)$$

Taking the last element of (3.26) and (3.27), we have

Table 1 Summary of the definitions of the filter variables.

Variable	Definition	Meaning	
$\varepsilon_i^n(t)$	$\langle\pi\|\varepsilon_i^n\rangle_t$	forward prediction error of $x_i(t)$	(3.15)
$\varepsilon_i^{*n}(t)$	$\langle\pi^*\|\varepsilon_i^n\rangle_t$	forward prediction error of $x_i(t-M+1)$	(3.16)
$\eta_i^n(t)$	$\langle\pi\|\eta_i^n\rangle_t$	backward prediction error of $s^{-n}x_i(t)$	(3.17)
$\eta_i^{*n}(t)$	$\langle\pi^*\|\eta_i^n\rangle_t$	backward prediction error of $s^{-n}x_i(t-M+1)$	(3.18)
$\sigma_i^n(t)$	$\langle\varepsilon_i^n\|\varepsilon_i^n\rangle_t$	$\sum_{k=t-M+1}^{t}(\varepsilon_i^n(k))^2$	(3.19)
$\tau_i^n(t)$	$\langle\eta_i^n\|\eta_i^n\rangle_t$	$\sum_{k=t-M+1}^{t}(\eta_i^n(k))^2$	(3.20)
$\triangle_i^n(t)$	$\langle\varepsilon_i^n\|\eta_{i-1}^n\rangle_t$	$\sum_{k=t-M+1}^{t}(\varepsilon_i^n(k)\,\eta_{i-1}^n(k))^2$	(3.21)
$\alpha_i^{n+1}(t)$	$-\triangle_i^n(t)/\tau_{i-1}^n(t)$	forward PARCOR coefficient	(3,22)
$\beta_i^{n+1}(t)$	$-\triangle_i^n(t)/\sigma_i^n(t)$	backward PARCOR coefficient	(3.23)

To get the order update recursion, we first take the orthogonal subspace decomposition:

$$\varepsilon_i^{n+1}(t) = <\pi|\varepsilon_i^{n+1}> = \varepsilon_i^n(t) + \alpha_i^{n+1}(t)\,\eta_{i-1}^n(t) \quad (3.29)$$

$$\eta_t^{n+1}(t)=\langle\pi|\eta_t^{n+1}\rangle=n_{t-1}^{n}(t)+ +\beta_t^{n+1}(t)\,\varepsilon_t^{n}(t) \qquad (3.30)$$

and we also get from (3.28)

$$\eta_0^{n}(t)=\eta_d^{n}(t-1) \qquad (3.31)$$

The relations (3.29) through (3.31) require the update recursions for $\sigma_t^n(t)$, $\tau_{t-1}^n(t)$ and $\triangle_t^n(t)$ to calculate the time-dependent forward and backward PARCOR coefficients $\alpha_t^{n+1}(t)$, $\beta_t^{n+1}(t)$. The computation of these variables entails the time update. However, in contrast with the prewindiowd form, SMC type filter needs backward time update as well as forward time update in order to reject the influence of the data outside the window.

Time update consists of two procedures: first, the backward time update to exclude the influence of the past data is carried out, and then the forward time update to obtain the new information from the current data.

$$\triangle_t^n(t-1)=\langle\bar{x}_t\,P_{Y_{t,1,n,t-1}}|\bar{x}_t s^{-n-1}\rangle_{t-1} +\langle\varepsilon_t^n|\pi^*\rangle_{t-1}\langle\pi^*|\eta\rangle/\cos^2\theta_{t,n}^*(t-1)$$
$$=\widetilde{\triangle}_t^n(t-1)+\varepsilon_t^{*n}(t-1)\,\eta_{t-1}^{*n}(t-1)/\cos^2\theta_{t,n}^*(t-1) \qquad (3.35)$$

where $\widetilde{\sigma}_t^n(t-1)\triangleq\langle\bar{x}_t|P^{\perp}_{\bar{Y}_{t,1,n,t-1}}|s^{-n-1}\bar{x}_t\rangle_{t-1}$, $\widetilde{\tau}_{t-1}^n(t-1)\triangleq\langle s^{-n-1}x_t|P^{\perp}_{\bar{Y}_{t,1,n,t-1}}|s^{-n-1}\bar{x}_t\rangle_{t-1}$, $\widetilde{\triangle}_t^n(t-1)=\langle\bar{x}_t|P_{\bar{Y}_{t,1,n,t-1}}|s^{-n-1}\bar{x}_t\rangle_{t-1}$.

Thus we get

$$\widetilde{\sigma}_t^n(t-1)=\sigma_t^n(t-1)-\varepsilon_t^{*n}(t-1)^2/\cos^2\theta_{t,n}^*(t-1) \qquad (3.36)$$

$$\widetilde{\tau}_{t-1}^n(t-1)=\tau_{t-1}^n(t-1)-\eta_{t-1}^{*n}(t-1)^2//\cos^2\theta_{t,n}^*(t-1) \qquad (3.37)$$

$$\widetilde{\triangle}_t^n(t-1)=\triangle_t^n(t-1)-\varepsilon_t^{*n}(t-1)\,\eta_{t-1}^{*n}(t-1)/\cos^2\theta_{t,n}^*(t-1) \qquad (3.38)$$

From 3), for any vector $|u\rangle_t$, $|v\rangle_t\in H_t$, the general backward time update formula is given by

$$<u|P^{\perp}_{Y_{t,1,n,t}}|v>_t=<\bar{u}|P^{\perp}_{\bar{Y}_{t,1,n,t}}|\bar{v}>_t +<uP^{\perp}_{Y_{t,1,n,t}}|\pi^*>_t<\pi^*|P_{Y_{t,1,n,t}}|v>_t/\cos^2\theta_{t,n}^*(t) \qquad (3.32)$$

where $|\tilde{v}>_t\triangleq[0\,v(t-M+2)\cdots v(t)]$, $|\bar{Y}_{t,1,n}>_t\triangleq[|s^{-1}\bar{x}_t>_t\cdots|s^{-p}\bar{x}_t>_t]$, and $\cos^2\theta_{t,n}^*(t)\triangleq<\pi^*|P_{Y_{t,1,n,t}}|\pi^*>_t$ Here $\theta^*_{1,n}(t)$ is the angle between $|\bar{Y}_{t,1,n}>_t$ and $|Y_{t,1,n}>_t$, and $\cos^2\theta_{1,n}(t)$ can be interpreted as a measure of the influence of the past data.

From (3.11), (3.19)-(3.21), (3.32) and $\langle P^{\perp}v|Pu\rangle=0$,

$$\sigma_t^n(t-1)=\langle x_t|P^{\perp}_{\bar{Y}_{t,1,n,t-1}}|x_t\rangle_{t-1}+\langle\varepsilon_t^n|\pi^*\rangle_{t-1}\langle\pi^*|\varepsilon_t^n\rangle_{t-1}/\cos^2\theta_{t,n}^*(t-1)$$
$$=\widetilde{\sigma}_t^n(t-1)+\varepsilon_t^{*n}(t-1)^2/\cos^2\theta_{t,n}^*(t-1) \qquad (3.33)$$

$$\tau_{t-1}^n(t-1)=\langle s^{-n-1}\bar{x}_t|P^{\perp}_{Y_{t,1,n,t-1}}|s^{-n-1}\bar{x}_t\rangle_{t-1} +\langle\eta_{t-1}^n|\pi^*\rangle_{t-1}\langle\pi^*|\eta_{t-1}^n\rangle_{t-1}/\cos^2\theta_{t,n}^*(t-1)$$
$$==\widetilde{\tau}_{t-1}^n(t-1)+\eta_{t-1}^{*n}(t-1)^2/\cos^2\theta_{t,n}^*(t-1) \qquad (3.34)$$

To compute the above update formulas, the values of $\varepsilon_t^{*n}(t-1)$, $\eta_{t-1}^{*n}(t-1)$, and $\cos^2\theta_{t,n}^*(t-1)$ are needed. Taking the first element of (3.26) and (3.27),

$$\varepsilon_t^{*n+1}(t)=\varepsilon_t^{*n}(t)+\alpha_t^{n+1}(t)\,\eta_{t-1}^{*n}(t) \qquad (3.39)$$

$$\eta_t^{*n+1}(t)=\eta_{t-1}^{*n}(t)+\beta_t^{n+1}(t)\,\varepsilon_t^{*n}(t) \qquad (3.40)$$

Also from (3.24) and the definition of $\cos^2\theta_{t,n}^*(t)$,

$$\cos^2\theta_{t,n+1}^*(t)=\cos^2\theta_{t,n}^*(t)-(\eta_{t-1}^{*n}(t))^2/\tau_{t-1}^n(t) \qquad (3.41)$$

Now we proceed to forward time update to compute $\sigma_t^n(t)$, $\tau_{t-1}^n(t)$ and $\triangle_t^n(t)$. From 3), the general forward time update formula is given by

$$\langle u|P^{\perp}_{Y_{t,1,n,t}}|v\rangle_t=\langle\tilde{u}|P^{\perp}_{\bar{Y}_{t,1,n,t}}|v\rangle_t +\langle u|P^{\perp}_{Y_{t,1,n,t}}|\pi\rangle_t\langle\pi|P^{\perp}_{Y_{t,1,n,t}}|v\rangle_t/\cos^2\theta_{t,n}(t) \qquad (3.42)$$

where $|\tilde{v}>_t\triangleq[v(t-M+1)\cdots v(t-1)\;0]'$, $|\bar{Y}_{t,1,n}>_t\triangleq[|s^{-1}\tilde{x}_t>_t\cdots|s^{-n}\tilde{x}_t>_t]$ and $\cos^2\theta_{1,n}(t)=\langle\pi|P_{Y_{1,1,n,t}}|\pi\rangle_t$. Here $\theta_{1,n}(t)$ is the angle between

$|Y_{1,1,n}\rangle_t$ and $|Y_{1,1,n}\rangle_t$, and $\cos^2\theta_{1,n}(t)$ can be interpreted as a measure o the new information contained in the current data.

From (3.42) and $<\tilde{x}_i|P^{\perp}_{Y_{i,1,n,t}}|s^{-n-1}\tilde{x}_i>_t=<\bar{x}_i|P^{\perp}_{Y_{i,1,n,t-1}}|s^{-n-1}\bar{x}_i>_{t-1}$, we have

$$\sigma^n_i(t)=\langle\tilde{x}_i|P^{\perp}_{Y_{i,1,n,t}}|\tilde{x}_i\rangle_t+\langle\varepsilon^n_i|\pi\rangle_t\langle\pi|\varepsilon^n_i\rangle_t/\cos^2\theta_{i,n}(t)$$
$$=\tilde{\sigma}^n_i(t-1)+\varepsilon^n_i(t)^2/\cos^2\theta_{i,n}(t) \quad (3.43)$$

$$\tau^n_{i-1}(t)=\langle s^{-n-1}\tilde{x}_i|P^{\perp}_{Y_{i,1,n,t}}|s^{-n-1}\tilde{x}_i\rangle_t+\langle\eta^n_{i-1}|\pi\rangle_t\langle\pi|\eta^n_{i-1}\rangle_t/\cos^2\theta_{i,n}(t) \quad (3.44)$$
$$=\tilde{\tau}^n_{i-1}(t-1)+\eta^n_{i-1}(t)^2/\cos^2\theta_{i,n}(t)$$

$$\triangle^n_i(t)=\langle\tilde{x}_i|P^{\perp}_{Y_{i,1,n,t}}|s^{-n-1}\tilde{x}_i\rangle_t+\langle x_i|P_{Y_{i,1,n,t}}|\pi\rangle_t\langle\pi|P_{Y_{i,1,n,t}}|s^{-n-1}x_i\rangle_t/\cos^2\theta_{i,n}(t)$$
$$=\tilde{\triangle}^n_i(t-1)+\varepsilon^n_i(t)\eta^n_{i-1}(t)/\cos^2\theta_{i,n}(t) \quad (3.45)$$

Table 2 SMC-CL Filter algorithm

INITIAILZE :

$\tilde{\triangle}^0_i(-1)=\eta^0_d(-1)=\eta^{*0}_d(-1)=0$

$\tilde{\tau}^0_i(-1)=\tilde{\sigma}^0_i(-1)=$ small positive value

FOR t = 0 TO T

FOR i = 1 TO d

$\varepsilon^0_i(t)=\eta^0_i(t)=x_i(t)$

$\varepsilon^{*0}_i(t)=\eta^{*0}_i(t)=x_i(t-M+1)$

$\cos^2\theta_{i,0}(t)=\cos^2\theta^*_{i,0}(t)=1$

FOR n = 0 TO min{p,t}−1

(IF i = 1, $\eta^n_0(t)=\eta^n_d(t-1)$, $\eta^{*n}_0(t)=\eta^{*n}_d(t-1)$)

$\triangle^n_i(t)=\tilde{\triangle}^n_i(t-1)+\varepsilon^n_i(t)\eta^n_{i-1}(t)/\cos^2\theta_{i,n}(t)$

$\sigma^n_i(t)=\tilde{\sigma}^n_i(t-1)+\varepsilon^n_i(t)^2/\cos^2\theta_{i,n}(t)$

$\tau^n_{i-1}(t)=\tilde{\tau}^n_{i-1}(t-1)+\eta^n_{i-1}(t)^2/\cos^2\theta_{i,n}(t)$

$\alpha^{n+1}_i(t)=-\triangle^n_i(t)/\tau^n_{i-1}(t)$

$\beta^{n+1}_i(t)=-\triangle^n_i(t)/\sigma^n_i(t)$

$\varepsilon^{n+1}_i(t)=\varepsilon^n_i(t)+\alpha^{n+1}_i(t)\eta^n_{i-1}(t)$

$\eta^{n+1}_i(t)=\eta^n_{i-1}(t)+\beta^{n+1}_i(t)\varepsilon^n_i(t)$

$\cos^2\theta_{i,n+1}(t)=\cos^2\theta_{i,n}(t)-\eta^n_{i-1}(t)^2/\tau^n_{i-1}(t)$

$\varepsilon^{*n+1}_i(t)=\varepsilon^{*n}_i(t)+\alpha^{n+1}_i(t)\eta^{*n}_{i-1}(t)$

$\eta^{*n+1}_i(t)=\eta^{*n}_{i-1}(t)+\beta^{n+1}_i(t)\varepsilon^{*n}_i(t)$

$\tilde{\triangle}^n_i(t)=\triangle^n_i(t)-\varepsilon^{*n}_i(t)\eta^{*n}_{i-1}(t)/\cos^2\theta^*_{i,n}(t)$

$\tilde{\sigma}^n_i(t)=\sigma^n_i(t)-\varepsilon^{*n}_i(t)^2/\cos^2\theta^*_{i,n}(t)$

$\tilde{\tau}^n_{i-1}(t)=\tau^n_{i-1}(t)-\eta^{*n}_{i-1}(t)^2/\cos^2\theta^*_{i,n}(t)$

$\cos^2\theta^*_{i,n+1}(t)=\cos^2\theta^*_{i,n}(t)-\eta^{*n}_{i-1}(t)^2/\tau^n_{i-1}(t)$

Also from(3.24) and the definition of $\cos^2\theta_{1,n}(t)$, the recursion for $\cos^2\theta_{1,n}(t)$ is given by

$$\cos^2\theta_{i,n+1}(t)=\cos^2\theta_{i,n}(t)-(\eta^n_{i-1}(t))^2/\tau^n_{i-1}(t) \quad (3.46)$$

The resultant SMC-CL filter algorithm is summarized in Table 2. SMC-CL filter consists of two prewindowed type CL filter with identical time varying coefficients:one is for obtaining the new information from the current data, and the other is for eliminating the influence of the past data. Thus the computational burden increases more or less, and SMC-CL filter requires that all data samples in the sliding window to be stored. Each channel of the filter runs in parallel.

Because the influence of one data is completely removed after M step in SMC-CL filter, the initial conditions are the same as the prewindowed CL filter, and SMC-CL filter becomes identical with the prewindowed CL filter, when t<M.

4. ARMA modeling using SMC-CL filter

Like other lattice filters, CL filter can be used for ARMA modeling[10]. Here we apply SMC-CL filter to ARMA modeling and induce an efficient feature of the filter.

Consider the ARMA(p,p) process

$$y(t)=-\sum_{i=1}^{p}a_i y(t-i)+\sum_{j=0}^{p}b_j e(t-j) \quad (4.1)$$

where $b_0=1$, and $\{e(t)\}$ is a white sequence with $\sim N(0,1)$. If we convert $y(t)$ into a 2 channel AR process $X(t)$ defined by $X(t)=[e(t)\ y(t)]$, we can now apply SMC-CL filter to $X(t)$ with $d=2$ instead of treating $y(t)$.

The following special properties due to the whiteness of $e(t)$ are still satisfied in SMC-CL filter as well as in the prewindowed CL filter.[10]

$$E\{\varepsilon^n_1(t)\eta^n_2(t-1)\}=E\{e(t)\eta^n_2(t-1)\}=0$$
$$\alpha^{n+1}_1=\beta^{n+1}_1=0$$
$$\alpha^1_2=-E\{\varepsilon^0_2(t)\eta^0_1(t)\}/E\{\eta^0_1(t)^2\}$$
$$=-E\{y(t)e(t)\}/E\{e(t)^2\}=-1$$
$$\text{for } n=0,\ \cdots,2p-1$$

$$\varepsilon_1^{n+1}(t)=\varepsilon_1^n(t)=e(t)$$
$$\eta_1^{n+1}(t)=\eta_2^n(t-1) \qquad (4.2)$$

That is, the PARCOR coefficients and the prediction errors of the first channel need not be calculated. Furthermore, from (3.39), (3.40) and (4.2), we know SMC-CL filter satisfies the properities (4.3) in addition.

$$\tau_1^{n+1}(t)=\tau_2^n(t-1)$$
$$\varepsilon_1^{*n+1}(t)=\varepsilon_1^{*n}(t)=e(t-M+1) \quad \text{for } n=0,\cdots,2p-1 \qquad (4.3)$$
$$\eta_1^{*n+1}(t)=\eta_2^{*n}(t-1)$$

So the filter structure becomes quite simple in consequence. However, we cannot use SMC-CL algorithm directly, and the hypothetical input estimation is necessary, for at time t we must know the value of e(t) which is not available. But we note from (4.1) that e(t) is considered to be an estimation error of y(t) based on {y(t-1), e(t-1), · · ·, y(t-p), e(t-p)}.

Thus we estimate e(t) by the joint process estimation of y(t) based on {y(t-1), e(t-1), · · · ·, y(t-p), e(t-p)} whose orthogonal basis is $\{\eta_2^0(t-1), \eta_2^1(t-1), \cdots, \eta_2^{2p-1}(t-1)\}$.

$$\hat{e}(t)\triangleq\varepsilon_y^{2p}(t)=y(t)+\gamma^1\eta_2^0(t-1)+\cdots+\gamma^{2p}\eta_2^{2p-1}(t-1) \qquad (4.4)$$

where

$$\gamma^n=-E\{\varepsilon_y^{n-1}(t)\eta_2^{n-1}(t-1)\}/E\{(\eta_2^{n-1}(t-1))^2\} \qquad (4.5)$$

Computing (4.4) and (4.5) recursively from the data entails the least squares joint process estimation algorithm by the same procedure in section 3. The resultant recursion formulas are

$$\triangle_y^n(t)=\tilde{\triangle}_y^n(t-1)+\varepsilon_y^n(t)\eta_2^n(t-1)/\cos^2\theta_{1,n}(t) \qquad (4.6)$$

$$\gamma^{n+1}(t)=-\triangle_y^n(t)/\tau_2^n(t-1) \qquad (4.7)$$

$$\varepsilon_y^{n+1}(t)=\varepsilon_y^n(t)+\gamma^{n+1}(t)\eta_2^n(t-1) \qquad (4.8)$$

$$\cos^2\theta_{1,n+1}(t)=\cos^2\theta_{1,n}(t)-\eta_2^n(t-1)^2/\tau_2^n(t-1) \qquad (4.9)$$

$$\varepsilon_y^{*n+1}(t)=\varepsilon_y^{*n}(t)+\gamma^{n+1}(t)\eta_2^{*n}(t-1) \qquad (4.10)$$

$$\tilde{\triangle}_y^n(t)=\triangle_y^n(t)-\varepsilon_y^{*n}(t)\eta_2^{*n}(t-1)/\cos^2\theta_{1,n}^*(t) \qquad (4.11)$$

$$\cos^2\theta_{1,n+1}^*(t)=\cos^2\theta_{1,n}^*(t)-\eta_2^{*n}(t-1)^2/\tau_2^n(t-1) \qquad (4.12)$$

where the variable $\eta_2^n(t-1)$, $\eta_2^{*n}(t-1)$ and $\tau_2^n(t-1)$ are already calculated in SMC-CL filter, so the recursions (4.6)-(4.12) run well without any trouble.

Thus we can proceed ARMA modeling with two step procedure: at first step we compute $\varepsilon_y^{2p}(t)$, the estimate of e(t), using (4.6)-(4.12), then the parameters of the SMC-CL filter are computed using y(t) and $\varepsilon_y^{2p}(t)$ as the second step. This modeling scheme is simplified more or less by the following relations.

Let us express the two linear spaces spanned by n variables {e(t), y(t-1), · · ·} and n-1 variables {y(t-1), e(t-1), · · ·} as $F_{2,1,n}$, $F_{1,1,n-1}$ respectively.By the orthogonal subspace decomposition,

$$F_{2,1,n}=F_{1,1,n-1}\oplus e(t) \qquad (4.13)$$

Then

$$\varepsilon_2^n(t)=y(t)-E\{y(t)\mid F_{2,1,n}\}$$
$$=y(t)-E\{y(t)\mid F_{1,1,n-1}\}-E\{y(t)\mid e(t)\}$$
$$=\varepsilon_y^{n-1}(t)-e(t) \qquad (4.14)$$

In the same manner, $\varepsilon_2^{*n}(t)$ is given by

$$\varepsilon_2^{*n}(t)=\varepsilon_y^{*n-1}(t)-e(t-M+1) \qquad (4.15)$$

Also from (4.2) and (4.14),

$$\alpha_2^{n+1}=-E\{\varepsilon_2^n(t)\eta_1^n(t)\}/E\{\eta_1^2(t)^2\}$$
$$=-E\{\eta_2^{n-1}(t-1)(\varepsilon_y^{n-1}(t)-e(t))\}/E\{\eta_2^{n-1}(t-1)^2\}=\gamma^n \qquad (4.16)$$

Though the relations (4.2) and (4.14)-(4.15) are valid in the strict sense only when the estimated input $\hat{e}(t)$ is the actual $e(t)$, they work well under certain conditions, and are often used in system identification.

The resultant SMC-CL filter algorithm for ARMA modeling is summarized in Table 3,and its feature is shown in Fig.1.

It is easy to obtain the parameters of an ARMA process y(t) given by (4.1) from the filter variables in Table 3.

Let Y(t) be a scalar periodic AR process corresponding to X(t).

$$Y(i+td)=x_i(t),\quad i=1,\ 2. \tag{4.17}$$

Since each channel of SMC-CL filter is a lattice type predictor, the forward and backward prediction errors of the filter can be rewritten as

$$\varepsilon_i^n(t)=Y(i+td)+\sum_{j=1}^{n} c_i^n(j,t)\ Y(i+td-j)\quad =1,\ 2. \tag{4.18}$$

$$\eta_i^n(t)=Y(i+td-n)+\sum_{j=1}^{n} d_i^n(n+1-j,t)\,Y(i+td+1-j)\quad i=1,\ 2 \tag{4.19}$$

where $\{c_i^n(j,t)\}$ and $\{d_i^n(j,t)\}$ are the parameters of the forward and backward linear predictors of order n. Also the PARCOR cofficients $\alpha_i^n(t)$, $\beta_i^n(t)$ are equivalent to the n-th parameters of order n.[3),8)] That is, $\alpha_i^{n+1}(t)=c_i^{n+1}(n+1,t)$ and $\beta_i^{n+1}(t)=d_i^{n+1}(n+1,t)$. Therefore $\{c_i^{n+1}(j,t)\}$, $\{d_i^{n+1}(j,t)\}$ can be calculated by the well-known Levinson recursions together with the cyclic property.[3),8)]

$$c_i^{n+1}(n+1,t)=\alpha_i^{n+1}(t) \tag{4.20}$$

$$d_i^{n+1}(n+1,t)=\beta_i^{n+1}(t) \tag{4.21}$$

$$c_i^{n+1}(j,t)=c_i^n(j,t)+\alpha_i^{n+1}(t)\,d_{i-1}^n(n+1-j,t)\quad j=1,\cdots\cdots,n \tag{4.22}$$

$$d_i^{n+1}(j,t)=d_{i-1}^n(j,t)+\beta_i^{n+1}(t)\,c_i^n(n+1-j,t)\quad j=1,\cdots\cdots,n \tag{4.23}$$

Now we iterate the above recursions from n=1 up to n=2p+1, and compare $\{c_2^{2p+1}(t)\}$ with (4.1), then the parameters of ARMA process y(t) are obtained by

$$c_2^{2p+1}(2i,t)=a_i(t)\quad i=1,\cdots\cdots,p \tag{4.24}$$

$$c_2^{2p+1}(2i+1,t)=-b_i(t)\quad i=1,\cdots\cdots,p \tag{4.25}$$

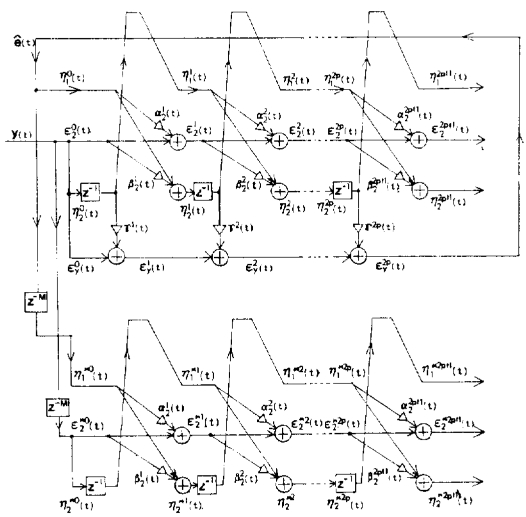

Fig.1 SMC-CL filter for ARMA modeling.

Table 3 SMC-CL filter algorithm for ARMA modeling.

INITIALIZE :

$\widetilde{\Delta}_2^0(-1)=\widetilde{\Delta}_y^0(-1)=\eta_2^0(-1)=\eta_2^{*0}(-1)=0$

$\tilde{\tau}_2^0(-1)=\tilde{\sigma}_2^0(-1)=$ small positive value

For t=0 to T

INPUT ESTIMATION PART :

$\varepsilon_y^0(t)=y(t),\quad \varepsilon_y^{*0}(t)=y(t-M+1)$

$\cos^2\theta_{1,0}(t)=\cos^2\theta_{1,0}^*(t)=1$

For n=0 to min{2p, 2t}−1

$\Delta_y^n(t)=\widetilde{\Delta}_y^n(t-1)+\varepsilon_y^n(t)\,\eta_2^n(t-1)/\cos^2\theta_{1,n}(t)$

$\tau_2^n(t-1)=\tilde{\tau}_2^n(t-2)+\eta_2^n(t-1)^2/\cos^2\theta_{1,n}(t)$

$\gamma^{n+1}(t)=-\Delta_y^n(t)/\tau_2^n(t-1)$

$\varepsilon_y^{n+1}(t)=\varepsilon_y^n(t)+\gamma^{n+1}(t)\,\eta_2^n(t-1)$

$\cos^2\theta_{1,n+1}(t)=\cos^2\theta_{1,n}(t)-\eta_2^n(t-1)^2/\tau_2^n(t-1)$

$$\varepsilon_y^{*n+1}(t)=\varepsilon_y^{*n}(t)+\gamma^{n+1}(t)\,\eta_2^{*n}(t-1)$$
$$\tilde{\triangle}_y^n(t)=\triangle_y^n(t)-\varepsilon_y^{*n}(t)\,\eta_2^{*n}(t-1)/\cos^2\theta_{1,n}^{*}(t)$$
$$\tilde{\tau}_2^n(t-1)=\tau_2^n(t-1)-\eta_2^{*n}(t-1)^2/\cos^2\theta_{1,n}^{*}(t)$$
$$\cos^2\theta_{1,n+1}^{*}(t)=\cos^2\theta_{1,n}^{*}(t)-\eta_2^{*n}(t-1)^2/\tau_2^n(t-1)$$

$$\hat{e}(t)=\varepsilon_y^{2p}(t)$$

SMC-CL FILTER PART:

$$\eta_1^0(t)=\hat{e}(t),\quad \varepsilon_2^0(t)=\eta_2^0(t)=y(t)$$
$$\eta_1^{*0}(t)=\hat{e}(t-m+1),\quad \varepsilon_2^{*0}(t)=\eta_2^{*0}(t)=y(t-M+1)$$
$$\tau_1^0(t)=\tau_1^0(t-1)+\hat{e}(t)^2-\hat{e}(t-M+1)^2$$
$$\cos^2\theta_{2,0}(t)=\cos^2\theta_{2,0}^{*}(t)=1$$

For n= 0 to min {2p, 2t}

FIRST CHANNEL:

$$\eta_1^{n+1}(t)=\eta_2^n(t-1)$$
$$\eta_1^{*n+1}(t)=\eta_2^{*n}(t-1)$$
$$\tau_1^{n+1}(t)=\tau_2^n(t-1)$$

SECOND CHANNEL:

$$\triangle_2^n(t)=\tilde{\triangle}_2^n(t-1)+\varepsilon_2^n(t)\,\eta_1^n(t)/\cos^2\theta_{2,n}(t)$$
$$\sigma_2^n(t)=\tilde{\sigma}_2^n(t-1)+\varepsilon_2^n(t)^2/\cos^2\theta_{2,n}(t)$$
$$\alpha_2^{n+1}(t)=\gamma^n(t)$$
$$\beta_2^{n+1}(t)=-\triangle_2^n(t)/\sigma_2^n(t)$$
$$\varepsilon_2^{n+1}(t)=\varepsilon_y^n(t)-\hat{e}(t)$$
$$\eta_2^{n+1}(t)=\eta_1^n(t)+\beta_2^{n+1}(t)\,\varepsilon_2^n(t)$$
$$\cos^2\theta_{2,n+1}(t)=\cos^2\theta_{2,n}(t)-\eta_1^n(t)^2/\tau_1^n(t)$$
$$\varepsilon_2^{*n+1}(t)=\varepsilon_y^{*n}(t)-\hat{e}(t-M+1)$$
$$\eta_2^{*n+1}(t)=\eta_1^{*n}(t)+\beta_2^{n+1}(t)\,\varepsilon_2^{*n}(t)$$
$$\tilde{\triangle}_2^n(t)=\triangle_2^n(t)-\varepsilon_2^{*n}(t)\,\eta_1^{*n}(t)/\cos^2\theta_{2,n}^{*}(t)$$
$$\tilde{\sigma}_2^n(t)=\sigma_2^n(t)-\varepsilon_2^{*n}(t)^2/\cos^2\theta_{2,n}^{*}(t)$$
$$\cos^2\theta_{2,n+1}^{*}(t)=\cos^2\theta_{2,n}(t)-\eta_1^{*n}(t)^2/\tau_1^n(t)$$

5. Simulation results

The theoretical analysis of the convergence characteristics and the numerical properties of lattice type algorithms is a difficult problem because the relations between the filter variables are highly nonlinear to reduce the computational burdens. Therefore, as an alternative, computer simulations are performed to show the effectiveness of the proposed algorithm. Two examples are taken; one is an ARMA process and the other is a sinusoidal data case.

Example 1 : ARMA parameter estimation

This example is taken to show the convergence of SMC-CL filter. The signal data are generated by ARMA(2, 2) model represented by

$$\begin{aligned} y(t) &= -a_1y(t-1)-a_2y(t-2)+e(t)+b_1e(t-1)+b_2e(t-2) \\ &= 1.5y(t-1)-0.7y(t-2)+e(t)+0.5e(t-1)-0.3e(t-2) \end{aligned}$$

where $\{e(t)\}$ is white noise sequence with $\sim N(0.1)$

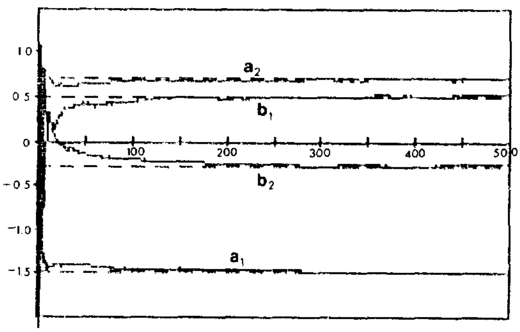

Fig.2 ARMA(2.2) parameters of example 1.

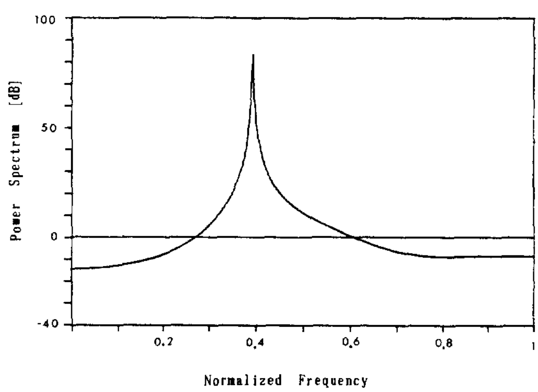

Fig.3 Power spectrum of example 2(t=350).

The simulations are performed with p=2, and results are in Fig.2. The dotted lines indicate the true ARMA parameters and the solid lines are the corresponding estimates for each parameter averaged over ten different sample paths. The estimates of ARMA parameters converge to the true values well. The convergence rate of MA parameters is slower than that of AR parameters due to the hypothetical input estimation.

Example 2: Adaptive spectral estimation

This example is taken to show the tracking performance of SMC-CL filter. In this example, the following time varying sinusoidal signal is used.

$$y(t)=\begin{cases}\sin(0.3\pi t) & t<200\\ \sin((0.3+(t-200)/1000)\pi t) & 200\leq t<300\\ \sin(0.4\pi t) & t\geq 300\end{cases}$$

The simulation is done with p=8 and M=48, and the result is depicted in Fig.3. An exact spectrum of the time varing signal is achieved after the window slides up to the Mth data from the point of signal change. That is, SMC-CL filter tracks well to the time varying parameters of the process.

6. Conclusion

The sliding window covariance circular lattice (SMC-CL) filter for ARMA modeling is presented. First SMC-CL filter is derived for the general case by the geometric approach. Since SMC-CL filter is a covariance type filter, it works well in case of relatively short data record. Also it can track to the time varying parameters by virtue of its sliding windowing action on data. As CL filters including SMC-CL filter contain only the scalar operations, SMC-CL filter is efficient in the computational aspects.

To use SMC-CL filter in ARMA modeling, ARMA process is converted to two variate AR process, and then SMC-CL filter is applied to it. The joint process estimation algorithm is also derived to estimate e(t) which is not available at time t. Due to the whiteness of e(t), SMC-CL ARMA modeling algorithm becomes simpler. The parameters of ARMA process can be obtained from the forward and backward PARCOR coefficients of the second channel of SMC-CL filter by the Levinson recursions.

SMC-CL filter may be used in fields of adaptive signal processing, system identification and spectral estimation. The quantitative analysis of stability and convergence characteristics of SMC-CL filter still remains a future work.

(본 연구는 한국과학재단의 연구비 지원으로 이루어졌읍니다)

REFERENCES

1) S.M.Kay and S.L.Marple, Jr., "Spetrum Analysis-A Modern Perspective," Proc. IEEE, vol.69, pp.1380-1419, 1981.

2) S.Haykin, Adaptive Filter Theory, Englewood Cliffs, Prentice Hall, 1986.

3) M.L.Honig and D.G.Messerschmitt, Adaptive Filters: Structures, Algorithms and Applications. Boston, Kluwer Academic Publishers, 1984.

4) B.Friedlander, "Lattice Filters for Adaptive Processing," Proc. IEEE, vol.70, pp.829-867, 1982.

5) D.T.L.Lee, B.Friedlander and M.Morf, "Recursive Ladder Algorithm for ARMA Modeling," IEEE Tr. Automat. Contr., vol.AC-27, pp.753-764, 1982.

6) B.Porat, B.Friedlander and M.Morf, "Square Root Covariance Ladder Algorithms," IEEE Tr.Automat. Contr., vol.AC-27, pp.813-829, 1982.

7) M.Pagano, "On Periodic and Multiple Autoregressions," Ann. Statist., vol.6, pp.1310-1317, 1978.

8) H.Sakai, "Circular Lattice Filtering Using Pagano's Method," IEEE Tr.Acoust., Speech and Signal Processing, vol.ASSP-30, pp.279-27, 1982.

9) T.Kawase, H.Sakai and H.Tocumaru, "Recursive Least Squares Circular Lattice and Escalator Estimation Algorithms," IEEE Tr.Acoust,, Speech and Signal Processing, vol.ASSP-31, pp.228-231, 1983.

10) H.Sakai, H.Tokumaru and A.Futamura, "ARMA Modeling Using Cicular Lattice Filter," IFAC 9th World Congress, vol.10, pp.65-69, Budapest, 1984.