論文 85-22-6-13

適應 Lattice 필터의 收斂特性 改善에 관한 研究

(A Study on the Improvement of the Convergence Properties of the Adaptive Lattice Filters)

白 興 基*, 李 鍾 珏**

(Heung Ki Baik and Chong Kak Lee)

要 約

本 論文에서는 適應 lattice 필터에서 反射係數를 구하는 일반적인 알고리즘인 Burg 알고리즘의 收斂速性을 解析하였으며, μ알고리즘을 Burg 알고리즘에 적용하여 Burg 알고리즘보다 收斂速度가 더 큰 새로운 適應 lattice μ알고리즘을 얻었다.

특히 Burg 알고리즘의 收斂速度가 낮을 때 本 論文에서 提案한 適應 lattice μ알고리즘의 收斂速度가 상대적으로 커짐을 컴퓨터 시뮬레이션을 통하여 확인하였다.

Abstract

The convergence properties of Burg algorithm, which is commonly used to calculate the reflection coefficients in the adaptive lattice filters, are studied in this paper.

Applying the μ algorithm to Burg algorithm, a new adaptive lattice μ algorithm is derived and it shows that the convergence speed of this algorithm is higher than that of Burg algorithm.

As a result of theoretical analysis and computer simulation, it is proved that the convergence speed of the proposed algorithm is remarkably higher than that of Burg algorithm when the convergence speed of Burg algorithm is low.

I. 序 論

서로 相關關係(correlation)가 있는 일련의 信号에 있어서는 과거의 信号値에 의하여 미래의 信号値를 豫測할 수 있다. 이러한 豫測機能은 과거의 信号値를 入力으로 하고 미래의 信号値를 出力으로 하는 필터에 의하여 수행될 수 있는데, 이러한 필터를 豫測필터(prediction filter)라 하며 특히 미래의 信号値가 과거의 信号値의 線型結合으로 표시될 때 線型豫測필터(linear prediction filter)라 한다.

線型豫測필터의 係数는 일반적으로 豫測誤差(prediction error)를 最小化하는 방향으로 適應(adaptation)시켜 얻는다. 이때 係数의 適應速度는 필터의 實現方法에 따라 큰 차이가 있다.

線型豫測필터를 실현하는 방법에는 tapped delay line 필터(이하 TDL필터라 함)를 사용하는 방법과 lattice 필터를 사용하는 방법 두가지가 있다. TDL 필터를사용하는 방법은 각 段의 係数가 서로 結合되어 있어 係数의 適應速度가 낮다. 그러나 lattice 필터의 경우에

*正會員, 全北大學校 電子工學科

(Dept. of Electron. Eng., Chonbuk National Univ.)

**正會員, 서울大學校 電子工學科

(Dept. of Electron. Eng., Seoul National Univ.)

接受日字：1985年 7月 1日

는 각 段에서의 豫測誤差가 서로 直交(orthogonal) 하기 때문에 각 段별의 係数의 最適化(optimization) 를 독립적으로 시킬 수 있으므로 適應速度를 높일 수 있다.[1,2,3] 또 필터의 最適次数를 쉽게 찾을 수 있으며 word length에 대한 感度가 낮기 때문에 音聲處理나 雜音除去등 여러 방면에 이용되고 있다.[4,5,6]

適應 lattice 필터에서 反射係数를 구하는 알고리즘은 여러 종류 있으나 그중 많이 사용되고 있는 알고리즘은 Burg 알고리즘이다.[5,7,8]

본 연구에서는 이미 提案된 μ알고리즘[9] 을 適應 lattice 알고리즘에 적용할 수 있는지의 여부를 이론적으로 검토하였으며, 이것을 이용하여 Burg 알고리즘보다 適應速度가 더 높은 lattice μ알고리즘을 提案하였으며 이들 컴퓨터 시뮬레이션을 통하여 확인하였다.

Ⅱ. 適應 Lattice 필터

1. 適應 Lattice 알고리즘

適應 lattice 필터에서 反射係数(reflection coefficient)를 구하는 데 가장 많이 사용되는 알고리즘은 Burg 알고리즘이다. Burg 알고리즘의 收斂特性을 解析하기 위하여 먼저 Burg 알고리즘을 유도하여 보자.

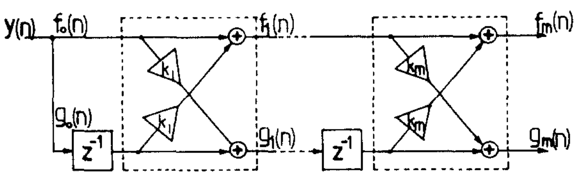


그림 1. Lattice 예측필터

Fig. 1. Lattice prediction filter.

그림 1은 lattice 필터의 기본구조이다. 여기서 다음 식이 얻어진다.

$$f_0(n) = g_0(n) = y(n) \quad (1)$$

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1) \quad (2)$$

$$g_m(n) = K_m f_{m-1}(n) + g_{m-1}(n-1) \quad (3)$$

여기서 $y(n)$은 入力이고 $f_m(n)$, $g_m(n)$, K_m은 각각 m次段에서의 順方向豫測誤差(forward prediction error), 逆方向豫測誤差(backward prediction error), 反射係数(reflection coefficient)이다.

反射係数 k_m을 구하는 방법은 여러가지 있으나 일반적으로 다음과 같이 定義되는 平均自乘誤差 J_m을 最小化하여 얻는다.[5]

$$\begin{aligned} J_m &= E[e_m^2(n)] \\ &= E[(1-r) f_m^2(n) + r g_m^2(n)] \\ &= K_m^2[(1-r) G_{m-1}(n-1) + rF_{m-1}(n)] + 2K_m H_{m-1}(n) + [(1-r) F_{m-1}(n) + rG_{m-1}(n-1)] \end{aligned} \quad (4)$$

단 $F_m(n) = E[f_m^2(n)]$ (5)

$G_m(n) = E[g_m^2(n)]$ (6)

$H_m(n) = E[f_m(n)\, g_m(n-1)]$ (7)

식(4)를 보면 平均自乘誤差 J_m이 反射係数 K_m의 2次式으로 표시되므로 最適係数(optimal coefficient) K_m^*는 다음과 같이 기울기(gradient) ∇_m을 0으로 놓고 풀면 얻어진다.[10]

$$\begin{aligned} \nabla_m &= \frac{\partial J_m}{\partial K_m} \\ &= 2K_m[(1-r) G_{m-1}(n-1) + rF_{m-1}(n)] + 2H_{m-1}(n) \\ &= 0 \end{aligned} \quad (8)$$

$$K_m^* = \frac{-H_{m-1}(n)}{(1-r) G_{m-1}(n-1) + rF_{m-1}(n)} \quad (9)$$

따라서 J_m의 最小値 $J_{m,min}$은 다음과 같이 된다.

$$J_{m,min} = K_m^* H_{m-1}(n) + [(1-r) F_{m-1}(n) + rG_{m-1}(n-1)] \quad (10)$$

(9)에 의해 K_m^*을 구하는 데는 많은 計算量이 필요하므로 steepest descent 방법을 이용하여 다음과 같이 循環的(recursive)으로 K_m을 구한다.

$$K_m(n+1) = K_m(n) + \mu[-\nabla_m(n)] \quad (11)$$

단 $$\nabla_m(n) = \frac{\partial J_m(n)}{\partial K_m(n)} \quad (12)$$

$\nabla_m(n)$ 역시 計算量이 많이 필요하므로 다음과 같이 瞬時値 $\hat{\nabla}_m(n)$으로 近似化하면 알고리즘 Ⅰ이 얻어진다.[10,11]

$$\begin{aligned} \hat{\nabla}_m(n) &= \frac{\partial e_m^2(n)}{\partial K_m(n)} \\ &= 2[f_m(n)\, g_{m-1}(n-1) + g_m(n)\, f_{m-1}(n)] \end{aligned} \quad (13)$$

Algorithm Ⅰ

$$\begin{cases} K_m(n+1) = K_m(n) + \mu[-\hat{\nabla}_m(n)] & (14) \\ \hat{\nabla}_m(n) = 2[f_m(n)\, g_{m-1}(n-1) + g_m(n)\, f_{m-1}(n)] & (15) \end{cases}$$

알고리즘 Ⅰ에서 r=0.5로 놓으면 Burg 알고리즘이 된다.[7]

2. 適應 Lattice 알고리즘의 收斂特性

Burg 알고리즘인 알고리즘 Ⅰ의 收斂特性을 解析하여 보자: 식(4)와 식(10)을 이용하면 平均自乘誤差 J_m을 다음과 같이 나타낼 수 있다.

$$\begin{aligned} J_m &= J_{m,min} + (K_m^2 - K_m^{*2})[(1-r) G_{m-1}(n-1) + rF_{m-1}(n)] + 2(K_m - K_m^*) H_{m-1}(n) \\ &= J_{m,min} + (K_m - K_m^*)^2[(1-r) G_{m-1}(n-1) + rF_{m-1}(n)] \end{aligned} \quad (16)$$

식(16)에서 기울기(gradient) ∇_m은 다음과 같이 된다.

$$\nabla_m = 2(K_m - K_m^*)[(1-r) G_{m-1}(n-1) + rF_{m-1}(n)] \quad (17)$$

여기서 $K_m - K_m^* = L_m$이라 놓으면 식(11)은 다음과 같이 된다.

$$K_m(n+1)-K_m^*=K_m(n)-K_m^*+\mu[-\nabla_m(n)] \quad (18)$$

$$L_m(n+1)=L_m(n)[1-2\mu\{(1-r)G_{m-1}(n-1)+rF_{m-1}(n)\}] \quad (19)$$

K_m이 K_m^*으로 收斂하기 위해서는 L_m이 0으로 收斂해야 한다. 따라서 K_m이 K_m^*으로 收斂하기 위해서는 다음식이 성립해야 한다.

$$|1-2\mu\{(1-r)G_{m-1}(n-1)+rF_{m-1}(n)\}|<1 \quad (20)$$

$$0<\mu<\frac{1}{(1-r)G_{m-1}(n-1)+rF_{m-1}(n)} \quad (21)$$

식(21)이 알고리즘 I 이 收斂하기 위한 조건이다. 만일 入力信号가 定常(stationary)이라고 가정하면 $F_{m-1}(n)$, $G_{m-1}(n-1)$은 n과 무관한 常数가 되어 식(19)는 等比数列의 형태가 되며, 이 때의 公比를 r_m이라 하면 r_m은 다음과 같이 된다.

$$r_m=1-2\mu[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)] \quad (22)$$

일반적으로 指数函数는 等比数列을 나타내는데 적합하다. 만일 반복주기(iteration cycle)를 단위시간(unit time)으로 잡고 收斂速度가 낮다고(slow adaptation) 가정하면 反射係数 K_m의 時定数 τ_m은 다음과 같이 구해진다.

$$\begin{aligned} r_m &= \exp\left(-\frac{1}{\tau_m}\right) \\ &= 1-\frac{1}{\tau_m}+\frac{1}{2!\,\tau_m^2}+\cdots\cdots \\ &= 1-2\mu[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)] \\ &\simeq 1-\frac{1}{\tau_m} \end{aligned} \quad (23)$$

$$\tau_m=\frac{1}{2\mu[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)]} \quad (24)$$

만일 入力信号가 定常(stationary)이라고 가정하면 時定数 τ_m은 일정한 값을 갖는 常数가 되어 入力信号의 통계치를 알 수 있는 경우 反射係数의 收斂速度를 알 수 있다.

식(16)을 다시 쓰면 다음과 같이 된다.

$$\begin{aligned} J_m(n) &= J_{m,min}+L_m^2(n)[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)] \\ &= J_{m,min}+L_m^2(o)\,(r_m^2)^n[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)] \end{aligned} \quad (25)$$

따라서 平均自乘誤差 J_m의 learning curve의 時定数는 다음과 같이 된다.

$$\begin{aligned} \tau_{m,mse} &= \frac{1}{2}Z_m \\ &= \frac{1}{4\mu[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)]} \end{aligned} \quad (26)$$

식(24)를 보면 反射係数의 收斂速度가 入力信号의 分散(variance)과 관계있음을 알 수 있다. 따라서 μ값을 分散으로 나누어 入力信号의 통계치와 관계가 없도록 定規化(normalize)하면 다음과 같은 알고리즘이 얻어진다.

$$K_m(n+1)=K_m(n)+\frac{\mu}{\sigma_m(n)}[-\nabla_m(n)] \quad (27)$$

$$\sigma_m(n)=[(1-r)G_{m-1}(n-1)+rF_{m-1}(n)] \quad (28)$$

위 식에서 $\nabla_m(n)$, $\sigma_m(n)$을 다음과 같이 $\hat{\nabla}_m(n)$, $\hat{\sigma}_m(n)$으로 近似化하면 Griffith가 提案한 알고리즘 II가 얻어진다.

Algorithm II

$$\left[\begin{aligned} & K_m(n+1)=K_m(n)+\frac{\mu}{\hat{\sigma}_m(n)}[-\hat{\nabla}_m(n)] && (29) \\ & \hat{\nabla}_m(n)=2[f_m(n)\,g_{m-1}(n-1)+g_m(n)\,f_{m-1}(n)] && (30) \\ & \hat{\sigma}_m(n)=(1-\beta)\,\hat{\sigma}_m(n-1)+\beta[(1-r)g_{m-1}^2(n-1)+rf_{m-1}^2(n)] && (31) \end{aligned}\right.$$

식(27)과 식(28)로 주어지는 알고리즘에서 反射係数 K_m이 收斂하기 위한 조건, 反射係数의 時定数, learning curve의 時定数는 각각 다음과 같다.

$$0<\mu<1 \quad (32)$$

$$\tau_m=\frac{1}{2\mu} \quad (33)$$

$$\tau_{m,mse}=\frac{1}{4\mu} \quad (34)$$

III. 適應 Lattice 필터에의 μ알고리즘의 適用

1. Lattice μ알고리즘의 誘導

[9]에서 提案된 μ알고리즘은 適應利得 μ를 변화시켜 收斂速度를 증가시키는 알고리즘이다. 이 때 μ값은 平均自乘誤差 J_m을 最小化하는 방향으로 변화시킨다.[9]

이 μ알고리즘을 lattice 필터에 적용하여 보자. 식(4)와 식(11)에서 J_m은 K_m의 2次式으로 표시되고 K_m과 μ는 1次관계에 있으므로 J_m은 μ의 2次式으로 표시된다. 따라서 steepest descent 방법을 적용하여 平均自乘誤差 J_m을 最小化하는 방향으로 μ값을 변화시키면 다음과 같은 식이 얻어진다.[9]

$$K_m(n+1)=K_m(n)+\mu_m(n)[-\nabla_m(n)] \quad (35)$$

$$\mu_m(n)=\mu_m(n-1)+\rho[-\nabla_m'(n)] \quad (36)$$

단 $$\nabla_m(n)=\frac{\partial J_m}{\partial K_m(n)} \quad (37)$$

$$\nabla_m'(n)=\frac{\partial J_m}{\partial \mu_m(n-1)} \quad (38)$$

이 때 식(38)의 $\nabla_m'(n)$은 다음과 같이 주어진다.

$$\begin{aligned} \nabla_m'(n) &= \frac{\partial J_m}{\partial \mu_m(n-1)} \\ &= \frac{\partial J_m}{\partial K_m(n)}\cdot\frac{\partial K_m(n)}{\partial \mu_m(n-1)} \\ &= -\nabla_m(n)\,\nabla_m(n-1) \end{aligned} \quad (39)$$

식(39)를 식(36)에 대입하면 다음 식이 얻어진다.

$$K_m(n+1)=K_m(n)+\mu_m(n)[-\nabla_m(n)] \quad (40)$$

$$\mu_m(n)=\mu_m(n-1)+\rho\nabla_m(n)\ \nabla_m(n-1) \qquad (41)$$

위 식에서 알고리즘 I 과 같이 $\nabla_m(n)$을 $\hat{\nabla}_m(n)$으로 近似化하면 다음과 같은 알고리즘 I′ 가 얻어진다.

Algorithm I′

$$\begin{cases} K_m(n+1)=K_m(n)+\mu_m(n)[-\hat{\nabla}_m(n)] & (42)\\ \mu_m(n)=\mu_m(n-1)+\rho\hat{\nabla}_m(n)\ \hat{\nabla}_m(n-1) & (43)\\ \hat{\nabla}_m(n)=2[f_m(n)\ g_{m-1}(n-1) \\ \qquad +g_m(n)\ f_{m-1}(n)] & (44)\end{cases}$$

이 때 알고리즘 I′ 가 收斂하기 위한 조건은 다음과 같다.[9]

$$\mu_m(n)>0\ ,\ m=1,2,3,\cdots\cdots \qquad (45)$$

$$0<\mu_m(0)<\frac{1}{(1-r)G_{m-1}(n-1)+rF_{m-1}(n)} \qquad (46)$$

마찬가지로 알고리즘 II 에 μ알고리즘을 적용하면 다음과 같은 알고리즘 II 가 얻어진다.

Algorithm II′

$$\begin{cases} K_m(n+1)=K_m(n)+\frac{\mu_m(n)}{\hat{\sigma}_m(n)}[-\hat{\nabla}_m(n)] & (47)\\ \mu_m(n)=\mu_m(n-1)+\rho\hat{\nabla}_m(n)\ \hat{\nabla}_m(n-1) & (48)\\ \hat{\nabla}_m(n)=2[f_m(n)\ g_{m-1}(n-1) \\ \qquad +g_m(n)\ f_{m-1}(n)] & (49)\\ \hat{\sigma}_m(n)=(1-\beta)\ \hat{\sigma}_m(n-1)+\beta[(1-r)\ g_{m-1}^2(n-1) \\ \qquad +rf_{m-1}^2(n)] & (50)\end{cases}$$

이제 알고리즘 I′ 와 알고리즘 II′ 를 편의상 適應 lattice μ알고리즘이라 부르기로 한다.

식(35)에서 식(41)까지를 살펴보면 平均自乘誤差가 구하고자 하는 係數의 2次式으로 표시되고, 또 gradient search 방법으로 係數를 구하는 모든 알고리즘에 μ알고리즘을 적용시킬 수 있음을 알 수 있다.

2. Lattice μ알고리즘의 高速收斂性에 관한 考察

식을 간단히 하기 위해 다음과 같이 놓자.

$$R_{m-1}(n)=(1-r)\ G_{m-1}(n-1)+rF_{m-1}(n) \qquad (51)$$

식(17)을 식(39)에 대입하면 다음과 같은 식이 얻어진다.

$$\begin{aligned}\nabla'_m(n)&=-\nabla_m(n)\ \nabla_m(n-1)\\ &=-4L_m(n)\ L_m(n-1)\ R_{m-1}(n)\ R_{m-1}(n-1)\end{aligned} \qquad (52)$$

식(52)에서 入力信号가 定常(stationary)이라고 가정하면 $R_{m-1}(n)$은 n에 관계없는 常数가 된다. 따라서 $R_{m-1}(n)=R_{m-1}(n-1)=R_{m-1}$라 놓으면 식(52)는 다음과 같이 된다.

$$\nabla'_m(n)=-4L_m(n)\ L_m(n-1)\ R_{m-1}^2 \qquad (53)$$

따라서 식(36)는 다음과 같이 된다.

$$\begin{aligned}\mu_m(n)&=\mu_m(n-1)+4\rho\ R_{m-1}^2\ L_m(n)\ L_m(n-1)\\ &=\mu_m(0)+4\rho\ R_{m-1}^2\sum_{k=0}^{n-1}L_m(k)\ L_m(k+1)\end{aligned} \qquad (54)$$

$L_m(n)$은 指数的으로 감소한다고 가정할 수 있으므로 이 때의 時定数를 τ'라 하면 다음과 같이 된다.

$$L_m(n)=L_m(o)\ \exp\left(-\frac{n}{\tau'}\right) \qquad (55)$$

식(55)를 이용하여 $\sum_{k=0}^{n-1}L_m(k+1)\ L_m(k)$의 近似値를 계산할 수 있다.

$$\begin{aligned}\sum_{k=0}^{n-1}L_m(k+1)\ L_m(k)&=\sum_{k=0}^{n-1}L_m^2(o)\ \exp\left(-\frac{2n+1}{\tau'}\right)\\ &=L_m^2(o)\frac{1-\exp\left(-\frac{2n+1}{\tau'}\right)}{1-\exp\left(-\frac{2}{\tau'}\right)}\exp\left(-\frac{1}{\tau'}\right)\\ &\approx L_m^2(o)\ \exp\left(-\frac{1}{\tau'}\right)\frac{\tau'}{2}\left[1-\exp\left(-\frac{2n+1}{\tau'}\right)\right]\end{aligned} \qquad (56)$$

따라서 식(54)는 다음과 같이 된다.

$$\mu_m(n)=\mu_m(o)+C\left[1-\exp\left(-\frac{2n+1}{\tau'}\right)\right] \qquad (57)$$

$$\text{단}\ \ C=2\rho\ R_{m-1}^2\ L_m^2(o)\ \exp\left(-\frac{1}{\tau'}\right)\tau' \qquad (58)$$

식(19)에서 기하학적 비율 r'는 다음과 같이 된다.

$$\begin{aligned}r'&=\frac{L_m(n+1)}{L_m(n)}\\ &=1-2\ R_{m-1}\ \mu_m(n)\\ &=1-2\ R_{m-1}\left[\mu_m(o)+C\left\{1-\exp\left(-\frac{2n+1}{\tau'}\right)\right\}\right]\end{aligned} \qquad (59)$$

식(59)를 보면 n이 커짐에 따라 r'의 값이 점점 감소함을 알 수 있다. 이것은 收斂速度가 증가함을 의미한다. 또 식(58)에서 ρ값이 커지면 C값이 커져 收斂速度가 더욱 증가한다.

IV. 컴퓨터 시뮬레이션 및 檢討

시뮬레이션에 사용된 入力信号 $y(n)$은 다음과 같은 all pole 필터 $1/A(z)$에 白色 Gaussian 雜音을 통과시킨 후 分散(variance)이 1이 되도록 定規化(normalize)하여 얻는다.

$$\begin{aligned}A(z)=1&-2.73542\ z^{-1}+3.50115\ z^{-2}\\ &-2.41871\ z^{-3}+0.81451\ z^{-4}\end{aligned} \qquad (60)$$

10회의 시뮬레이션을 통하여 얻은 最適反射係数(optimal reflection coefficient)의 平均値는 다음과 같다.

$K_1^*=-0.87008$

$K_2^*=0.78646$

$K_3^*=-0.71777$

$K_4^*=0.82258$

그림 2～그림 6 은 모두 10회의 시뮬레이션 결과를 평균한 그림이다.

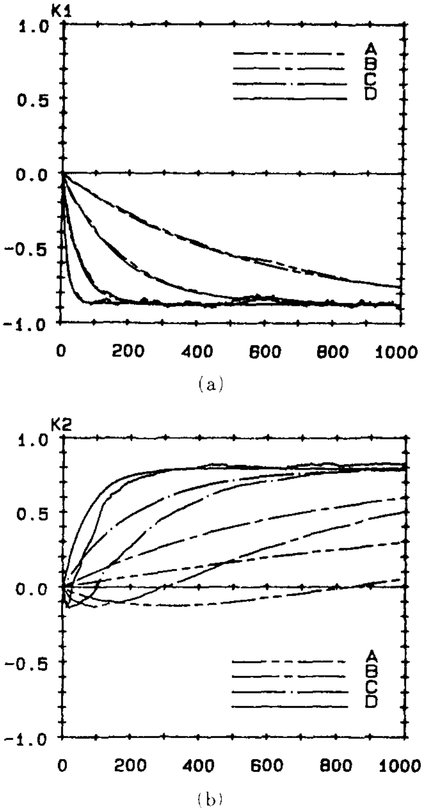


그림 2. 알고리즘 I 에서의 反射係數(理論値와 實驗値의 비교)

A : $\mu=0.001$, B : $\mu=0.003$, C : $\mu=0.01$, D : $\mu=0.03$

Fig. 2. Reflection coefficients in algorithm I (comparision between practical value and theoretical value).

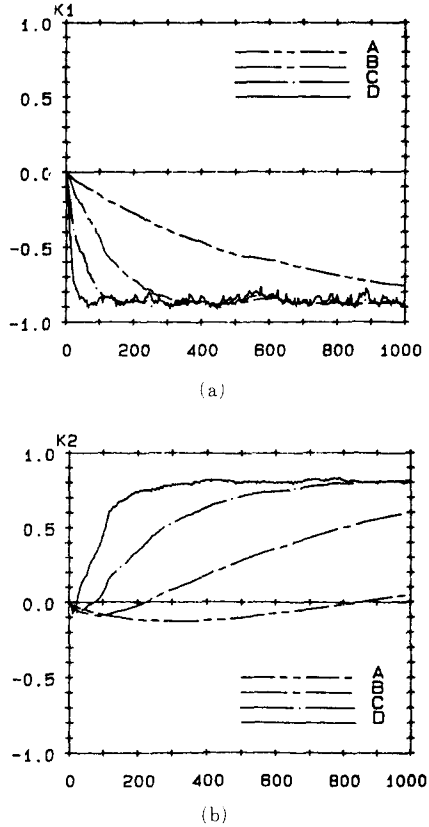


그림 3. 알고리즘 I′ 에서의 反射係數

$\mu_i(o)=0.001$, $i=1, 2$

A : $\rho=0$, B : $\rho=0.00001$, C : $\rho=0.0001$, D : $\rho=0.001$

Fig. 3. Reflection coefficients in algorithm I′.

그림 2 는 알고리즘 I 에서 μ값을 0.001, 0.003, 0.01, 0.03으로 변화시켰을 때의 反射係數 K_1, K_2 를 나타낸 것이다. K_1의 경우 식(24)로 주어지는 時定數에 의한 理論値와 實驗値가 잘 일치함을 알 수 있다. K_2의 경우 理論値와 實驗値는 상당한 차이를 나타낸다. 이것은 식(24)에 의한 時定數의 계산이 入力(前段의 出力)이 定常(stationary)이라고 가정했을 때만 성립하기 때문이다. 다시 말하면 前段의 反射係數가 이미 最適値로 收斂해서 고정된 값을 갖고 있다고 가정했을 때만 성립하기 때문이다. $\mu=0.03$의 경우 K_1이 最適値 K_1^*로 상당히 빨리 收斂해 있기 때문에 K_2의 경우 理論値와 實驗値가 어느정도 비슷함을 알 수 있다.

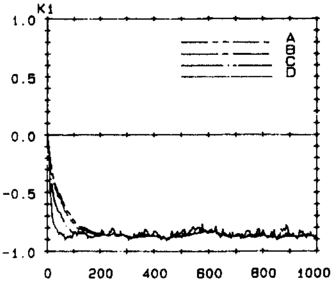
그림 3 은 알고리즘 I′ 에서 ρ값을 0, 0.00001, 0.0001, 0.001로 변화시켰을 때 反射係數 K_1과 K_2를 나타낸 것이다. $\rho=0$일 때 알고리즘 I′ 는 알고리즘 I 와 같게 되며, ρ값이 커질수록 收斂速度가 커짐을 알 수 있다.

그림 4 도 알고리즘 I′ 에서 ρ값을 변화시켰을 때 反射係數 K_1과 K_2를 나타낸 것으로 ρ값이 커질수록 收斂速度가 커짐을 알 수 있다.

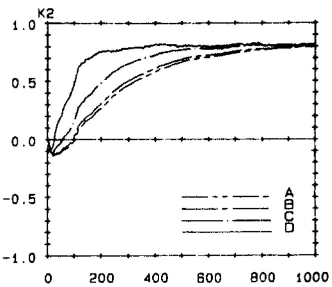
그림 3 과 그림 4 를 비교해 보면 lattice 알고리즘(Berg 알고리즘)에서 μ값이 작을 때, 즉 收斂速度가 낮을 때 lattice μ알고리즘의 收斂速度가 상대적으로 빨라짐을 알 수 있다.

그림 5 와 그림 6 은 알고리즘 II′ 에서 ρ값을 변화시켰을 때의 反射係數를 나타낸 것으로, ρ값이 커질수록 收斂速度가 커지며 lattice 알고리즘의 收斂速度가 낮을 때 lattice μ알고리즘의 收斂速度가 상대적으로 커짐을 알 수 있다.

따라서 入力信号에 대한 통계치가 전혀 알려져 있지 않은 경우 필터의 安定性을 위하여 μ값을 작게 취하여 收斂速度가 낮게되는 데 이럴 때 lattice μ알고리즘을 사용하면 收斂速度를 크게 할 수 있다.

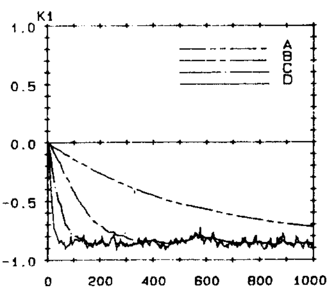


(a)

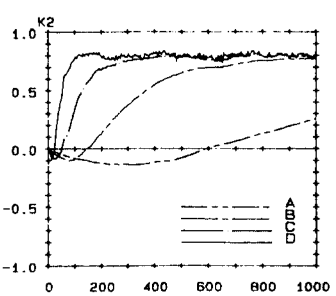


(b)

그림 4. 알고리즘 I′ 에서의 反射係數
$\mu_i(o) = 0.01$, $i = 1, 2$
A : $\rho = 0$, B : $\rho = 0.00001$, C : $\rho = 0.0001$, D : $\rho = 0.001$
Fig. 4. Reflection coefficients in algorithm I′.

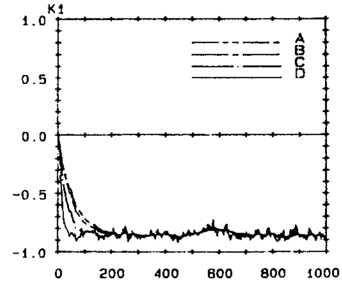


(a)

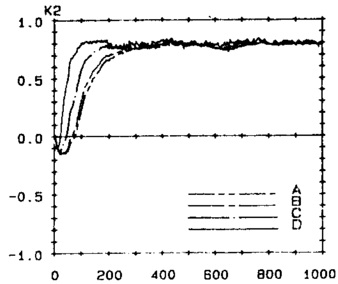


(b)

그림 5. 알고리즘 II′ 에서의 反射係數
$\mu_i(o) = 0.001$, $i = 1, 2$
A : $\rho = 0$, B: $\rho = 0.00001$, C : $\rho = 0.0001$, D : $\rho = 0.001$
Fig. 5. Reflection coefficients in algorithm II′.



(a)



(b)

그림 6. 알고리즘 II′ 에서의 反射係數
$\mu_i(o) = 0.01$, $i = 1, 2$
A : $\rho = 0$, B : $\rho = 0.00001$, C : $\rho = 0.0001$, D : $\rho = 0.001$
Fig. 6. Reflection coefficients in algorithm II′.

V. 結 論

適應 lattice 필터의 反射係數를 구하는 대표적인 알고리즘인 Burg 알고리즘의 收斂特性을 解析하였으며, 解析結果가 타당함을 컴퓨터 시뮬레이션을 통하여 확인하였다.

또 μ알고리즘을 Burg 알고리즘에 적용하여 收斂速度가 보다 큰 Burg 알고리즘의 새로운 형태인 lattice μ알고리즘을 提案하고 이 알고리즘의 高速收斂性을 解析하였으며, 특히 lattice 알고리즘의 收斂速度가 작을 때 lattice μ알고리즘의 收斂速度가 상대적으로 증가함을 컴퓨터 시뮬레이션을 통하여 확인하였다.

일반적으로 入力信号에 대한 통계치가 전혀 알려져 있는 않는 경우에는 필터의 安定性을 위하여 係數의 適應時 μ값을 작게 취하는 데, 이 때 lattice μ알고리즘을 사용하면 lattice 알고리즘에 비하여 係數의 收斂速度를 크게 증가시킬 수 있음을 알 수 있다.

參 考 文 献

[1] C.D. Gibson and S. Haykin, "Learning characteristics of adaptive lattice filtering algorithms", *IEEE Trans. on ASSP*, vol. ASSP-28, no. 6, pp. 681-691, Dec. 1980.

[2] M.L. Honig and D.G. Messerschmitt, "Convergence properties of an adaptive digital lattice filter", *IEEE Trans. on ASSP,* vol. ASSP-29, no. 3, pp. 642-653, Jun. 1981.

[3] B. Fridlander, "Lattice filters for adaptive processing", *Proc. IEEE,* vol. 70, no. 8, pp. 829-868, Aug. 1982.

[4] P.L. Chu and D.G. Messerschmitt, "Zero sensitivity properties of the digital lattice filters", *IEEE Trans. on ASSP,* vol. ASSP-31, no. 3, pp. 685-706, Jun. 1983.

[5] J.I. Makhoul and L.K. Cosell, "Adaptive lattice analysis of speech", *IEEE Trans. on ASSP,* vol. ASSP-29, no. 3, pp. 654-685, Jun. 1981.

[6] L.J. Griffiths, "an adaptive lattice structure for noise-cancelling applications," Proc. IEEE Conf. ASSP, (Tulsa. OK), pp. 87-90, Apr. 1978.

[7] J. Makhoul, "Stable and efficient lattice methods for linear prediction", *IEEE Trans. on ASSP,* vol. ASSP-25, no. 5, pp. 423-428, Oct. 1977.

[8] G.R.L. Sohie and L.H. Sibul, "Stochastic convergence properties of the adaptive gradient lattice", *IEEE Trans. on ASSP,* vol. ASSP-32, no. 1, pp. 102-107, Feb. 1984.

[9] 신윤기, 이종각, "LMS 적응필터 설계를 위한 고속수렴 알고리즘에 관한 연구", 전자공학회지, 제19권 제15호, PP. 12-19, 10월 1983년

[10] B. Widrow, J.M. Mccool, M.G. Larimore and C.R. Jonson, "Stationary and nonstationary learning characteristics of the LMS adaptive filter", *Proc. IEEE,* vol. 64, no. 8, pp. 1151-1162, Aug. 1976.

[11] G.C. Goodwin and K.S. Sin, *Adaptive Filtering Prediction and Control,* Prentice-Hall, pp. 390, 1984.