論文 83 - 20 - 4 - 1

HARF 알고리즘에서의 誤差 緩和 필터 除去에 관한 硏究
(A Study on Eliminating the Error-Smoothing Filter from HARF Algorithm)

辛 潤 杰* 李 鍾 珏*
(Yoon Ki Shin and Chong Kak Lee)

要 約

MRAS 超安定 出力 誤差 모델(MRAS hyperstable output-error model)을 利用한 適應 循環 필터(adaptive recursive filter)의 設計上 가장 어려운 點은 誤差 緩和 필터(error-smoothing filter)의 設計이다.

本 論文에서는 適應 循環 필터의 대표적 알고리즘인 HARF(hyperstable adaptive recursive filter) 알고리즘을 적절히 變形시킴으로써 誤差 緩和 필터를 除去시킬 수 있고, 同時에 收斂 速度도 빠른 알고리즘을 얻을 수 있음을 보였다.

Abstract

In designing the adaptive recursive filter using the MRAS hyperstable output-error identifier, the most crucial point is the design of the error-smoothing filter.

In this paper, it is shown that by modifying the HARF algorithm properly, we can obtain an algorithm which requires no error-smoothing filter and has relatively high convergence rate.

I. 序 論

適應 필터(adaptive filter)는 適應 非循環 필터[1](adaptive nonrecursive filter)와 適應 循環 필터[2~5](adaptive recursive filter)로 나누어지며, 그의 多樣한 機能으로 인하여 廣範圍한 信號處理 分野에 應用되고 있다. 一般的으로 適應 循環 필터는 適應 非循環 필터에 비해 필터의 次數가 훨씬 낮기 때문에 計算量面에서 상당히 有利하다. 그러나 適應 循環 필터는 時變 非線形 歸環(time varying nonlinear feedback)을 가지는 線形 시스템이기 때문에 適應 非循環 필터의 LMS 알고리즘[1]과 같은 簡單한 형태의 實時間的 處理가 可能한 알고리즘이 適應 循環 필터에는 없는 實情이다. 따라서 最近까지의 適應 필터에 의한 應用은, 適應 循環 필터의 計算量面에서의 利點에도 불구하고 주로 適應 非循環 필터에 依存해 왔다.

適應 循環 필터의 設計는, 制御에서 널리 利用되고 있는 MRAS(model reference adaptive system)[6]을 주로 利用해 왔으며, 이들 중 代表的인 것은 MRAS의 超安定 出力-誤差 모델(hyperstable output-error model)을 利用하여 I. D. Landau[7]에 의해 提示된 알고리즘이다. 그 후 이 알고리즘의 短點을 補完하여 實時間的 處理가 可能하도록 簡素化된 알고리즘인 HARF[2] 알고리즘이 提示되었으며, 이를 더욱 簡單히 하여 適應 非循環 필터의 LMS 알고리즘 형태로 만든 것을 SHARF(simple HARF)[3]라 한다. 그러나 이상의 適應 循環 필터 알고리즘들의 最大 難點은 誤差 緩和 필터[8], 혹은 補償器(compensator)[7]라고 불리는 필터의 係數 設定에 있다.

이 필터는 V. M. Popov의 超安定性(hyperstablity)[9]을 만족시키기 위해 挿入되는 것으로서, 이 필터의 係數를 구하기 위해서는 우리가 찾아 내고자하는 未知系

*正會員, 서울大學校 工科大學 電子工學科
(Dept. of Elec. Eng., Seoul National Univ.)
接受日字 : 1983年 1月 17日

統의 파라미터중 일부를 事前에 알 수 있어야 한다는 모순이 생기게 되었다. 이를 解決하기 위해 다시 I.D. Landau에 의해 適應 補償器(adaptive compensator)[5]를 사용한 알고리즘이 提示되었으나 알고리즘은 더욱 복잡하게 되었다.

本 論文에서는 HARF 알고리즘을 變形시켜, 誤差緩和 필터를 필요로 하지 않는 알고리즘을 提示함과 동시에 이 알고리즘의 收斂性을 證明하고자 한다. 특히 이 알고리즘은, T.A.C.M. Classen, W.F.G. Mecklenbräuker[10] 등에 의해 提示된 適應 非循環 필터의 設計原理와 一致함을 확인하였다.

Ⅱ. HARF 및 誤差 緩和 필터

아래의 式들로 表示되는 ARMA 모델의 基準 系統(reference system) 및 適應系統(adaptive system)을 고찰하여 보기로 한다.

基準 系統:

$$x_j = \sum_{i=1}^{N_A} a_i u_{j-i+1} + \sum_{i=1}^{N_B} b_i x_{j-i} = [A^T\ B^T] \begin{bmatrix} U_j \\ X_{j-1} \end{bmatrix} = W^T \theta_j \quad (1)$$

단, $U_j^T = [u_j, \cdots\cdots, u_{j-N_A+1}]$ (2)

$X_{j-1}^T = [x_{j-1}, \cdots\cdots, x_{j-N_B}]$ (3)

$A^T = [a_1, \cdots\cdots, a_{N_A}]$ (4)

$B^T = [b_1, \cdots\cdots b_{N_B}]$ (5)

$W^T = [a_1, \cdots\cdots, b_{N_A}, b_1, \cdots\cdots b_{N_B}]$ (6)

$\theta_j^T = [U_j^T, X_{j-1}^T]$ (7)

x_j — 반복 순간 j에서의 實際 出力

u_j — 반복 순간 j에서의 入力

適應 系統(適應 필터):

$$\hat{x}_j = \sum_{i=1}^{N_A} \hat{a}_{ij} u_{j-i+1} + \sum_{i=1}^{N_B} \hat{b}_{ij} \hat{x}_{j-i} = [\hat{A}_j^T, \hat{B}_j^T] \begin{bmatrix} U_j \\ \hat{X}_{j-1} \end{bmatrix} = \hat{W}_j^T \hat{\theta}_j \quad (8)$$

$$\hat{x}_j^0 = \hat{W}_{j-1}^T \hat{\theta}_j \quad (9)$$

단, $\hat{X}_{j-1}^T = [\hat{x}_{j-1}, \cdots\cdots, \hat{x}_{j-N_B}]$ (10)

$\hat{\theta}_j^T = [U_j^T, \hat{X}_{j-1}^T]$ (11)

$\hat{A}_j^T = [\hat{a}_{1j}, \cdots\cdots, \hat{a}_{N_A j}]$ (12)

$\hat{B}_j^T = [\hat{b}_{1j}, \cdots\cdots, \hat{b}_{N_B j}]$ (13)

$\hat{x}_j$…반복 순간 j에서의 *a posteriori* 推定 出力(*a posteriori* estimated output)

$\hat{x}_j^0$…반복 순간 j에서의 *a priori* 推定出力 (*a priori* estimated output)

出力 誤差(output error):

$$a\ priori: e_j^0 = x_j - \hat{x}_j^0 \quad (14)$$

$$a\ posteriori: e_j = x_j - \hat{x}_j \quad (15)$$

誤差 緩和 필터를 通過한 誤差信號:

$$a\ priori: \nu_j^0 = e_j^0 + \sum_{i=1}^{N_C} c_i e_{j-i} \quad (16)$$

$$a\ posteriori: \nu_j = e_j + \sum_{i=1}^{N_C} c_i e_{j-i} \quad (17)$$

이상의 式들을 利用한 HARF 알고리즘은 아래와 같다.

$$\hat{W}_{j+1} = \hat{W}_j + \nu_{j+1}^0 \frac{F\hat{\theta}_{j+1}}{1 + \hat{\theta}_{j+1}^T F \hat{\theta}_{j+1}} \quad (18)$$

$$F = \begin{bmatrix} \mu_1 & & & & \\ & \ddots & & & \\ & & \mu_{N_A} & & \\ & & & \rho_1 & \\ & & & & \ddots \\ & & & & & \rho_{N_B} \end{bmatrix} \quad (19)$$

단, $|\mu_i|^{N_A} > 0$, $|\rho_i|^{N_B} > 0$

식(18)에서의 ν_{j+1}^0은, 式(16)에서의 $|c_i|^{N_C}$係數가 아래의 SPR(strictly positive realness) 조건을 만족하도록 設定되어야 한다.

$$\mathrm{Re}\left\{ \frac{1 + \sum_{i=1}^{N_C} c_i z^{-i}}{1 - \sum_{i=1}^{N_B} b_i z^{-i}} \right\} > 0, \quad |z| = 1 \quad (20)$$

$C(z) = 1 + \sum_{i=1}^{N_C} c_i z^{-i}$: 誤差 緩和 필터

즉 誤差 緩和 필터의 係數인 $|c_i|^{N_C}$를 구하기 위해서는 未知 系統의 파라미터 $|b_i|^{N_B}$를 事前에 알아야 한다는 모순이 생기게 된다. 本 論文의 趣旨는 式(20)의 SPR 條件式을 排除하는데 있다.

Ⅲ. 誤差 緩和 필터를 除去하기 위한 알고리즘의 誘導

아래의 그림 1, 그림 2는 각각 MRAS의 出力 誤差 모델[6] 및 方程式 誤差 모델[6]을 利用한 適應 系統(適應 循環 필터)의 構成을 나타낸다.

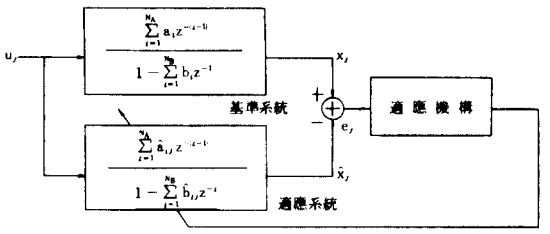

그림 1. 出力 誤差 모델 MRAS를 이용한 適應系統
Fig. 1. Abaptive system using output error model MRAS.

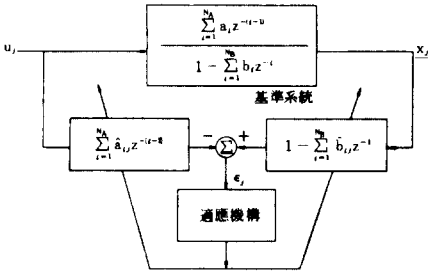

그림 2. 方程式 誤差 모델 MRAS를 이용한 適應系統
Fig. 2. Adaptive system using equation error model MRAS.

그림 1, 2를 利用하여 本 論文에서 提示하고자 하는 알고리즘은 각각 아래와 같다.

$$\hat{W}_{j+1}=\hat{W}_j+e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} \tag{21}$$

$$\hat{W}_{j+1}=\hat{W}_j+\epsilon^0_{j+1}\frac{\theta_{j+1}}{\|\theta_{j+1}\|^2} \tag{22}$$

a priori 方程式 誤差 :

$$\epsilon^0_{j+1}=x_{j+1}-\hat{W}_j^T\theta_{j+1}=(W-\hat{W}_j)^T\theta_{j+1} \tag{23}$$

a posteriori 方程式 誤差 :

$$\epsilon_{j+1}=(W-\hat{W}_{j+1})^T\theta_{j+1} \tag{24}$$

그외의 각 벡터, 行列, 스칼라는 Ⅱ章의 式 (2)에서 式(13)에 걸쳐 定義된 것과 同一하다. 알고리즘(21)은 그림 1의 出力 誤差 모델을, 알고리즘(22)는 그림 2의 方程式 誤差 모델을 각각 利用한 結果이다. 이 두 알고리즘은 誤差 緩和 필터를 사용하고 있지 않음을 알 수 있다.

1. 알고리즘 (21)의 誘導 및 收斂性에 대한 證明

HARF 알고리즘(式(18))을 變形시킴으로써 誤差 緩和 필터가 필요없는 알고리즘(21)을 誘導할 수 있음을 증명하고자 한다. HARF 알고리즘을 좀더 一般的 形態로 擴張시키면 아래와 같이 된다.

$$\hat{W}_{j+1}=\hat{W}_j+\nu^0_{j+1}\frac{F\hat{\theta}_{j+1}}{\lambda+\hat{\theta}^T_{j+1}F\hat{\theta}_{j+1}} \tag{25}$$

HARF 알고리즘을 얻기 위해서는 λ를 1로 하면 된다.

式 (16), (17), (25)를 利用하여 ν_{j+1}을 구하기 위해, 먼저 式(25)로 부터 e_{j+1}을 구한다.

$$\begin{aligned}e_{j+1}&=x_{j+1}-\hat{x}_{j+1}\\&=W^T\theta_{j+1}-\hat{W}^T_{j+1}\hat{\theta}_{j+1}\\&=W^T\hat{\theta}_{j+1}-\hat{W}^T_{j+1}\hat{\theta}_{j+1}+W^T(\theta_{j+1}-\hat{\theta}_{j+1})\\&=(W-\hat{W}_{j+1})^T\hat{\theta}_{j+1}+B^T(X_j-\hat{X}_j)\\&=(W-\hat{W}_{j+1})^T\hat{\theta}_{j+1}+B^TE_j\end{aligned} \tag{27}$$

단, $E_j=[e_j,\cdots\cdots,e_{j+1-N_B}]^T$ (28)

다음에 式(16)을 利用하여 ν^0_{j+1}을 구한다.

$$\begin{aligned}\nu^0_{j+1}&=e^0_{j+1}+\sum_{i=1}^{N_C}c_i(x_{j-i+1}-\hat{x}_{j-i+1})\\&=x_{j+1}-\hat{x}^0_{j+1}+\sum_{i=1}^{N_C}c_i(x_{j-i+1}-\hat{x}_{j-i+1})\\&=x_{j+1}-\hat{x}_{j+1}+\sum_{i=1}^{N_C}c_i(x_{j-i+1}-\hat{x}_{j-i+1})\\&\quad+(\hat{x}_{j+1}-\hat{x}^0_{j+1})\\&=\nu_{j+1}+\hat{W}^T_{j+1}\hat{\theta}_{j+1}-\hat{W}^T_j\hat{\theta}_{j+1}\\&=\nu_{j+1}+(\hat{W}_{j+1}-\hat{W}_j)^T\hat{\theta}_{j+1}\end{aligned} \tag{29}$$

式 (25)로부터

$$\hat{W}_{j+1}-\hat{W}_j=\nu^0_{j+1}\frac{F\hat{\theta}_{j+1}}{\lambda+\hat{\theta}^T_{j+1}F\hat{\theta}_{j+1}} \tag{30}$$

$$\therefore\ \nu^0_{j+1}=\nu_{j+1}+\nu^0_{j+1}\frac{\hat{\theta}^T_{j+1}F^T\hat{\theta}_{j+1}}{\lambda+\hat{\theta}^T_{j+1}F\hat{\theta}_{j+1}} \tag{31}$$

$F^T=F$ 이므로,

$$\nu_{j+1}=\nu^0_{j+1}\frac{\lambda}{\lambda+\hat{\theta}^T_{j+1}F\hat{\theta}_{j+1}} \tag{32}$$

따라서 알고리즘 (25)에서 $\lambda=0$이면

$$\nu_{j+1}=e_{j+1}+\sum_{i=1}^{N_C}C_ie_{j-i+1}=0 \tag{33}$$

式 (33)은 모든 j에 대해 항상 成立해야 하므로 *a posteriori* 出力 誤差 e_j는 항상 0이다.

즉, $e_j = 0$; 모든 j에 대해 (34)

$$\therefore \nu^0_{j+1} = e^0_{j+1} + \sum_{i=1}^{N_C} c_i e_{j-i+1} = e^0_{j+1} \quad (35)$$

즉, 誤差 緩和 필터 $C_1(z) = 1 + \sum_{i=1}^{N_C} c_i z^{-i}$가 제거됨을 알 수 있다. $\lambda = 0$일때, 알고리즘(25)는 아래와 같이 된다.

$$\hat{W}_{j+1} = \hat{W}_j + e^0_{j+1} \frac{F\hat{\theta}_{j+1}}{\hat{\theta}^T_{j+1} F \hat{\theta}_{j+1}} \quad (36)$$

알고리즘 (36)의 對角線 行列 F의 모든 인자들을 같게 하면 알고리즘 (21)을 얻을 수 있다. 또한 式 (34)에 의해 式 (27)은 아래와 같이 된다.

$$e_{j+1} = (W - \hat{W}_{j+1})^T \hat{\theta}_{j+1} = 0 \quad (37)$$

그러나 式 (37)의 結果를 살펴보면, 알고리즘 (21)이 파라미터 識別(parameter identification)機能을 가지기 위해서는 $\hat{W}_j$가 반드시 W에 收斂함을 立證할 필요가 있음을 알 수 있다.

e^0_{j+1}은 定義에 의해 아래와 같이 表示된다.

$$\begin{aligned} e^0_{j+1} &= x_{j+1} - \hat{x}^0_{j+1} \\ &= W^T\theta_{j+1} - \hat{W}^T_j \hat{\theta}_{j+1} \\ &= W^T\theta_{j+1} - \hat{W}^T_j\hat{\theta}_{j+1} + W^T\hat{\theta}_{j+1} + W^T\hat{\theta}_{j+1} \\ &= W^T\hat{\theta}_{j+1} - \hat{W}^T_j\hat{\theta}_{j+1} - W^T\hat{\theta}_{j+1} + W^T\hat{\theta}_{j+1} \\ &= (W - \hat{W}_j)^T\hat{\theta}_{j+1} + B^T(X_j - \hat{X}_j) \\ &= (W - \hat{W}_j)^T\hat{\theta}_{j+1} + B^T E_j \end{aligned} \quad (38)$$

式 (34)에 의해 式 (38)의 벡터 E_j는 제거된다.

$$\begin{aligned} \therefore e^0_{j+1} &= (W - \hat{W}_j)^T\hat{\theta}_{j+1} \\ &= d^T_j\hat{\theta}_{j+1} \end{aligned} \quad (39)$$

$d_j = W - \hat{W}_j$ ····· 파라미터 誤差 벡터 (40)

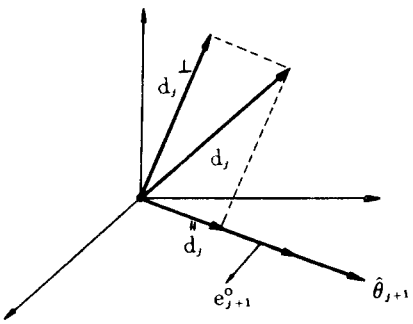

그림 3. 벡터 d_j의 成分 分解

Fig. 3. Vector decomposition of d_j.

式 (39)를 벡터 空間에 圖示하면 그림 3과 같다.

그림 3에서 벡터 d_j를 벡터 $\hat{\theta}_{j+1}$에 각각 垂直, 水平한 成分으로 나누면 式 (36)은 아래와 같이 된다.

$$\begin{aligned} e^0_{j+1} &= d^T_j\hat{\theta}_{j+1} \\ &= (d^\perp_j + d^{//}_j)^T\hat{\theta}_{j+1} \\ &= d^{//T}_j\hat{\theta}_{j+1} \end{aligned} \quad (41)$$

$$\therefore d^{//}_j = e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} \quad (42)$$

파라미터 誤差 벡터 d_j가 0으로 收斂하기 위해서는 파라미터 誤差 벡터 d_j로부터 $d^{//}_j$를 계속 빼나가면 된다.

즉,

$$\begin{aligned} d_{j+1} &= d_j - d^{//}_j \\ &= d_j - e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} \end{aligned} \quad (43)$$

$d_{j+1} = W - \hat{W}_{j+1}$이므로,

$$\therefore \hat{W}_{j+1} = \hat{W}_j + e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} \quad (21)$$

즉 式 (41), (42), (43)에 의해서도 알고리즘 (21)이 誘導됨을 알 수 있다. 式 (43)으로 부터, 파라미터 誤差 벡터 d_j의 絶對値 自乘에 대한 循環式을 다음과 같이 얻을 수 있다.

$$\|d_{j+1}\|^2 = \left(d_j - e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2}\right)^T\left(d_j - e^0_{j+1}\frac{\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2}\right)$$

$$= \|d_j\|^2 - 2e^0_{j+1}\frac{d^T_j\hat{\theta}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} + \frac{\hat{\theta}^T_{j+1}\hat{\theta}_{j+1}e^{0^2}_{j+1}}{\|\hat{\theta}_{j+1}\|^4}$$

$$= \|d_j\|^2 - \frac{e^{0^2}_{j+1}}{\|\hat{\theta}_{j+1}\|^2} < \|d_j\|^2 \quad (44)$$

따라서 反復(iteration)이 계속됨에 따라, 알고리즘 (21)의 파라미터 誤差 벡터는 반드시 0에 收斂하게 된다.

2. 方程式 誤差 모델을 利用한 알고리즘 (22)의 誘導 및 收斂性 證明

그림 2에서 반복 순간 j+1에서의 ***a priori*** 方程式 誤差는 아래와 같이 定義된다.

$$\epsilon^0_{j+1} = (x_{j+1} - \sum_{i=1}^{N_B}\hat{b}_{j,i}x_{j-i}) - \sum_{i=1}^{N_A}\hat{a}_{j,i}u_{j-i+1}$$

$$= x_{j+1} - \hat{A}^T_j U_{j+1} - \hat{B}^T_j X_j$$

$$= x_{j+1} - [\hat{A}_j^T, \hat{B}_j^T] \begin{bmatrix} U_{j+1} \\ X_j \end{bmatrix}$$

$$= x_{j+1} - \hat{W}_j^T \theta_{j+1}$$

$$= W^T \theta_{j+1} - \hat{W}_j^T \theta_{j+1}$$

$$= d_j^T \theta_{j+1} \qquad (45)$$

式 (45)를 벡터 空間에 表示하면 아래의 그림 4와 같다.

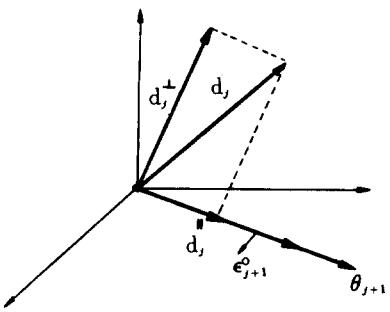

그림 4. 벡터 d_j의 成分 分解

Fig. 4. Vector decomposition of d_j.

앞의 式 (41)~(44)와 같은 過程을 되풀이 하면 아래와 같이 전개된다.

$$d_j^T \theta_{j+1} = (d_j^{\perp} + d_j'')^T \theta_{j+1}$$

$$= d_j''^T \theta_{j+1}$$

$$= \epsilon_{j+1}^o \qquad (46)$$

$$\therefore d_j'' = \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \qquad (47)$$

$$d_{j+1} = d_j - d_j''$$

$$= d_j - \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \qquad (48)$$

$$d_j = W - \hat{W}_j$$

$$\therefore \hat{W}_{j+1} = \hat{W}_j + \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \qquad (22)$$

따라서 提示된 알고리즘 (22)가 誘導되었다. 式 (48)로부터 아래와 같은 關係式을 얻을 수 있다.

$$\| d_{j+1} \|^2 = \left(d_j - \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \right)^T \left(d_j - \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \right)$$

$$= \| d_j \|^2 - \frac{\epsilon_{j+1}^{o2}}{\| \theta_{j+1} \|^2} < \| d_j \|^2 \qquad (49)$$

따라서 알고리즘 (22)의 파라미터 誤差 벡터도 항상 0으로 收斂한다. 또한 式 (22), (23), (24)로 부터 *a posteriori* 方程式 誤差 ϵ_{j+1}을 구해보면 아래와 같다.

$$\epsilon_{j+1} = W^T \theta_{j+1} - \hat{W}_j^T \theta_{j+1}$$

$$= W^T \theta_{j+1} - \left(\hat{W}_j^T + \epsilon_{j+1}^o \frac{\theta_{j+1}}{\| \theta_{j+1} \|^2} \right)^T \theta_{j+1}$$

$$= W^T \theta_{j+1} - \hat{W}_j^T \theta_{j+1} - \epsilon_{j+1}^o$$

$$= \epsilon_{j+1}^o - \epsilon_{j+1}^o$$

$$= 0 \qquad (50)$$

式 (34)와 (50)에 의해 $e_j = \epsilon_j = 0$이므로 모든 j에 대해 $\hat{\theta}_j = \theta_j$이다. 따라서 이상의 結果들을 종합하면 다음과 같다.

$$\left.\begin{aligned} e_j &= \epsilon_j = 0 \\ \hat{\theta}_j &= \theta_j \\ e_j^o &= d_{j-1}^T \hat{\theta}_j = d_{j-1}^T \theta_j = \epsilon_j^o \end{aligned}\right\} \qquad (51)$$

따라서 알고리즘 (21), (22)는 完全히 同一한 알고리즘이다. 즉 本論文에서 提示한 適應 循環 필터 設計 方法은, 出力 誤差 모델 MRAS를 利用한 設計 方法이나 방정식 오차 모델 NRAS를 이용한 설계방법이나 그 結果가 同一함을 알 수 있다. 알고리즘 (22)는 따로 $\hat{\theta}_j$를 계산할 필요가 없으므로 차후로는 알고리즘 (22)로써 이 두 알고리즘을 대신하고자 한다. 알고리즘 (22)에 대한 全體 系統 構成圖는 그림 5에 나타나 있다.

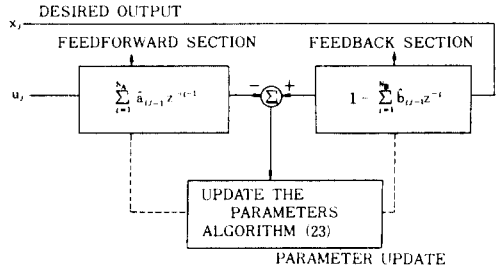

그림 5. 알고리즘 (22)의 구현

Fig. 5. Implementation of algorithm (22).

Ⅳ. 알고리즘 (22)의 收斂 速度

아래의 假定下에서 알고리즘 (22)의 平均的 收斂 速度를 式으로 表示하겠다. 또한 아래의 유도과정은 論文 [10]을 참조하였음을 밝혀 둔다.

1) 入力 U_j, 出力 x_j, 파라미터 誤差 벡터 d_j는 서

로 獨立的이다.

2) 入力 U_j, 出力 x_j는 각각 平均値가 0이며, 標準偏差가 σ_u^2, σ_x^2인 스테이셔너리 스토케스틱 프로세스(stationary stockastic process)이다.

벡터 θ_{j+1}의 自己 相關 行列 R은 아래와 같이 表示된다.

$$R = E(\theta_{j+1}\theta_{j+1}^T) = Q\Gamma Q^{-1} = Q\Gamma Q^T \qquad (52)$$

Q는 行列 R의 特性 벡터(eigen vector)들로 구성된 行列이며 Γ는 特性値(eigen value)들로 구성된 對角線 行列이다. 동시에 R은 對稱 行列이므로 $Q^{-1} = Q^T$이다.

式 (46), (48), (52)로 부터 아래와 같은 關係式을 얻을 수 있다.

$$d_{j+1} = d_j - \varepsilon_{j+1}^0 \frac{\theta_{j+1}}{\|\theta_{j+1}\|^2}$$

$$= d_j - \frac{\theta_{j+1}\theta_{j+1}^T d_j}{\|\theta_{j+1}\|^2} \qquad (53)$$

式 (53)의 양변에 $\|\theta_{j+1}\|^2 Q^{-1}$ 行列을 곱한 다음, 양변 각 항의 平均値를 취하면 다음과 같다.

$$(N_A\sigma_u^2 + N_B\sigma_x^2)\ E(Q^{-1}d_{j+1})$$
$$= (N_A\sigma_u^2 + N_B\sigma_x^2)\cdot E(Q^{-1}d_j)$$
$$- Q^{-1}\cdot E(\theta_{j+1}\theta_{j+1}^T)\cdot E(d_j)$$
$$= (N_A\sigma_u^2 + N_B\sigma_x^2)\cdot E(Q^{-1}d_j)$$
$$- Q^{-1}\cdot Q\Gamma Q^{-1}\cdot E(d_j)$$
$$= (N_A\sigma_u^2 + N_B\sigma_x^2)\cdot E(Q^{-1}d_j)$$
$$- \Gamma\cdot E(Q^{-1}d_j) \qquad (54)$$

$$\therefore D_{j+1} = \left(I - \frac{\Gamma}{N_A\sigma_u^2 + N_B\sigma_x^2}\right) D_j \qquad (55)$$

단, $D_j = E(Q^{-1}d_j)$ (56)

벡터 D_j의 i번째 인자를 D_j^i로 표시하면, D_j^i, $i = 1, 2, \cdots\cdots, N_A + N_B$의 收斂 速度는 아래와 같이 近似化할 수 있다.

$$D_{j+1}^i = \left(1 - \frac{\gamma_i}{N_A\sigma_u^2 + N_B\sigma_x^2}\right) D_j^i$$

$$= \left(1 - \frac{\gamma_i}{N_A\sigma_u^2 + N_B\sigma_x^2}\right)^{j+1} D_0^i \qquad (57)$$

단 λ_i는 對角線 行列 Γ의 i번째 인자이다. D_{j+1}^i을 時定數 τ_i를 가지는 指數 函數로 近似化할 수 있다고 假定한다.[1]

$$\therefore \frac{D_{j+1}^i}{D_j^i} = 1 - \frac{\gamma_i}{N_A\sigma_u^2 + N_B\sigma_x^2}$$

$$\fallingdotseq \exp\left(-\frac{1}{\tau_i}\right)$$

$$= 1 - \frac{1}{\tau_i} + \frac{1}{2!\tau_i^2} - \frac{1}{3!\tau_i^3} + \cdots\cdots \qquad (58)$$

보통의 收斂 速度에서는 $\tau_i \gg 1$이다.

$$\therefore \tau_i \fallingdotseq \frac{N_A\sigma_u^2 + N_B\sigma_x^2}{\gamma_i} \qquad (59)$$

즉 變形된 파라미터 誤差 벡터 $Q^{-1}d_j$의 i번째 인자는 平均的으로 時定數 $\tau_i = \frac{N_A\sigma_u^2 + N_B\sigma_x^2}{\gamma_i}$을 가지고 指數的으로 減少함을 알 수 있으며, 同時에 필터의 分子, 分母의 次數 N_A, N_B가 각각 增加할수록 時定數가 커져 收斂 速度가 減少함을 알 수 있다. 물론 本論文의 알고리즘에서는 收斂因子(convergence factor)가 나타나 있지 않으므로 式(58)처럼 가정하는 근거가 희박하나, 여러 차례의 實驗 結果, 필터의 次數가 어느 정도 증가하면 알고리즘(23)의 收斂 速度가 충분히 느려짐을 관찰하였으며, 이를 근거로하여 위의 收斂式들이 유도되었음을 밝혀 두고자 한다.

V. 電算 시뮬레이션

첫째로, 基準系統으로서

$$H(z) = \frac{0.057}{1 - 1.645Z^{-1} + 0.9025Z^{-2}} \quad 및$$

$$H(z) = \frac{1}{1 - 1.7Z^{-1} + 0.7225Z^{-2}}$$

두가지의 AR 모델을 設定하여 SHARF 알고리즘 및 알고리즘 (22)의, 각 모델에 대한 파라미터 抽出能力을 比較하였으며, 同時에 誤差 緩和 필터가 SHARF 알고리즘의 收斂 狀態에 미치는 영향도 考察하였다. 이 結果는 그림 7, 그림 8에 잘 나타나 있다. 각 모델의 入力 信號는 平均이 0이며 標準偏差가 0.5인 가우시안 랜덤 信號로 하였다. 이 신호는 FACOM 기종에 내장되어 있는 패케지 서브루틴인 NORRNS를 사용하여 발생시켰다. SHARF 알고리즘은, HARF 알고리즘 (18), (19)에서 $\{\mu_j\}^{N_A}$, $\{\rho_j\}^{N_B}$의 값을 충분히 작게 設定했을 때 아래와 같이 近似的으로 구해지는 알고리즘이다.

$$\hat{W}_{j+1} \fallingdotseq \hat{W}_j + \nu_{j+1}^0 F\hat{\theta}_{j+1} \qquad (60)$$

단, $\hat{\theta}_{j+1}^T F \hat{\theta}_{j+1} \ll 1$ (61)

전체적인 전산처리 흐름도는 그림 6에서 나타나 있다.

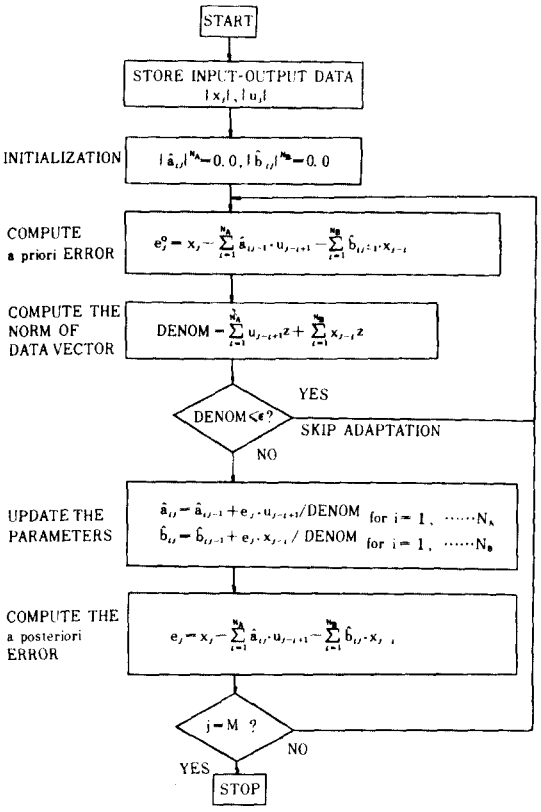

그림 6. 電算 處理 흐름도

Fig. 6. Flowchart of computer simulation.

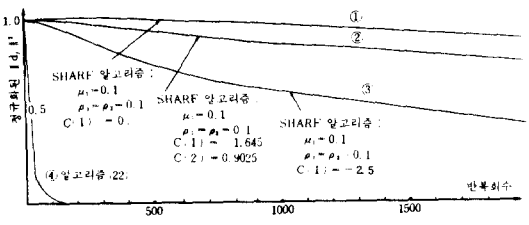

그림 7. AR 모델 $\frac{0.057}{1-1.645z^{-1}+0.9025^{-2}}$의 파라미터 抽出

Fig. 7. Parameter identification of AR model $\frac{0.057}{1-1.645z^{-1}+0.9025z^{-2}}$.

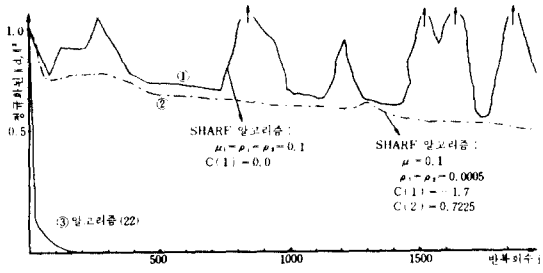

그림 8. AR 모델 $\frac{1}{1-1.7z^{-1}+0.7225z^{-2}}$의 파라미터 抽出

Fig. 8. Parameter identification of AR model $\frac{1}{1-1.7z^{-1}+0.7225z^{-2}}$.

그림 7에서 보면, SHARF 알고리즘은 誤差 緩和 필터가 없을 때(그래프 ①)나 SPR조건을 만족하는 誤差 緩和 필터 $C(z)=1-1.645z^{-1}+0.9025z^{-2}$가 播入됐을 때(그래프 ②)도 收斂 狀態가 서로 비슷하게 安定된 것을 볼 수 있다. 이 이유는 式(20)의 결과가 1이 되어 항상 0보다 크기 때문이다.(SPR 조건 만족) 이상의 사실로 SPR 조건이 SHARF 및 HARF 알고리즘의 收斂 條件에 대하여 充分條件일 뿐이지 必要條件이 아님을 알 수 있다. 誤差 緩和 필터를 $C(z)=1-2.5z^{-1}$로 하면 收斂 速度는 그래프 ①, ②에 비해 빨라진다.(그래프 ③) 그래프 ④는 알고리즘 (22)의 收斂 狀態를 나타낸 것으로써, 誤差 緩和 필터없이 빠른 收斂 速度를 나타내고 있다.

그림 8의 모델에서는 誤差 緩和 필터를 제거하면 SHARF 알고리즘의 收斂 狀態는 매우 不安해진다. 이 이유는 물론 SPR조건이 만족되지 않기 때문이다.(그래프①) 다음에 그래프 ②에서와 같이 誤差 緩和 필터 $C(z)$를 基準 系統의 分母式과 같게 하면 항상 SPR 조건이 만족되어 收斂 狀態가 安定되는 것을 알 수 있다). 그래프 ③은 알고리즘 (22)의 收斂 狀態를 나타낸다.

그림 9에서는 基準系統 函數의 次數變化에 따른 알고리즘 (22)의 收斂 狀態가 圖示되어 있다. 그림 7에서 그래프 ①~⑤는 각각 아래의 基準 系統 函數에 대한 알고리즘 (22)의 收斂 狀態를 나타낸다.

① $H(z)=\frac{0.792z^{-1}+0.5z^{-2}}{1-0.916z^{-1}+0.416z^{-2}-0.104z^{-3}}$

② $H(z)=\frac{0.792z^{-1}+0.5z^{-2}}{1-1.116z^{-1}+0.5992z^{-2}-0.1872z^{-3}-0.002}$

$$③\quad H(z)=\frac{1}{1-2.41z^{-1}+2.678z^{-2}-2.355z^{-3}+2.481z^{-4}-2.0537z^{-5}+0.7865z^{-6}}$$

$$④\quad H(z)=\frac{0.4728z^{-1}-0.3703z^{-2}}{1-2.41z^{-1}+2.678z^{-2}-2.355z^{-3}+2.481z^{-4}-2.0537z^{-5}+0.7865z^{-6}}$$

$$⑤\quad H(z)=\frac{0.4728z^{-1}-0.3703z^{-2}+0.28z^{-3}}{1-2.41z^{-1}+2.678z^{-2}-2.355z^{-3}+2.481z^{-4}-2.0537z^{-5}+0.7865z^{-6}}$$

式(59)에서 時定數 τ_i는 필터의 次數와 入出力 信號의 파우어의 平均値인 σ_u^2, σ_x^2에 영향을 받고 있음을 알 수 있다. 그러나 필터 系數와 出力 파우어의 관계는 高次 ARMA 모델에서는 상당히 복잡하므로 본논문에서는 단순히 出力信號의 파우어가 크게 변하지 않는 범위내에서 필터의 係數選定은 별로 염두에 두지 않고 다만 필터 次數가 時定數에 미치는 대략적인 영향만을 고찰하고자 하였다.

따라서 ARMA 모델 ①~⑤를 선정하는데 있어서 필터 係數는 크게 고려하지 않고 필터 次數의 變化에 만 重點을 두었다.

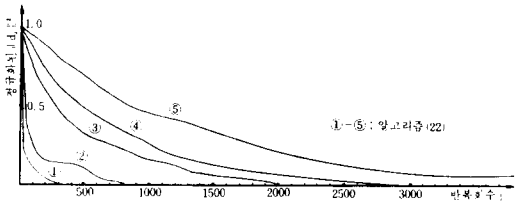

그림 9. 適應 필터의 次數 變化에 따른 알고리즘(22)의 收斂 速度 變化

Fig. 9. Variation of convergence rate of algorithm (22) according to different adaptive filter order.

그림 9의 結果를 살펴보면, 式(59)에서 알고리즘 (22)의 收斂 速度가 誘導된 바와 같이, 適應 필터의 次數가 增加할수록 그래프 ①~⑤와 같이 收斂 速度가 減少함을 알 수 있다.

Ⅵ. 結 論

HARF 알고리즘의 誤差 緩和 필터를 除去하기 위해 本論文에서 提示한 알고리즘 및 그의 長短點은 다음과 같다.

提示된 알고리즘 :

$$W_{j+1}=W_j+\epsilon_{j+1}^0\cdot\frac{\theta_j}{\|\theta_j\|^2}\qquad(22)$$

長點 :

1) 誤差緩和 필터가 필요하지 않다.

2) 모든 入力 信號에 대해 항상 收斂한다.

3) 알고리즘이 簡單하다.

4) 收斂 速度가 빠른 편이다.

短點 :

1) 適應 필터의 分子, 分母의 次數가 增加함에 따라 收斂 速度가 늦어진다.

무엇보다도 誤差 緩和 필터 除去가 알고리즘 (22)의 가장 큰 長點이다. 끝으로 이 알고리즘의 雜音 特性에 대해서 앞으로 硏究해 볼 필요가 있을 것 같다.

參 考 文 獻

[1] B. Widrow, J.M. McCool, M.G. Larimore, C.R. Johnson, "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proceedings of the IEEE*, vol. 64, no. 8, Aug. 1976.

[2] C.R. Johnson, "A convergence proof for a hyperstable adaptive recursive filter," *IEEE Trans. Information Theory*, vol. IT-25, no. 6, pp. 745-748, Nov. 1979.

[3] M.G. Larimore, J.R. Treichler, C.R. Johnson, "SHARF: An algorithm for adapting IIR digital filters," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. Assp-28, no. 4, pp. 428-440, Aug. 1980.

[4] P.L. Feintuch, "An adaptive LMS filter," *Proc. IEEE*, pp. 1622-1624, Nov. 1976.

[5] I.D. Landau, "Elimination of real positivity condition in the design of parallel MRAS," *IEEE Trans. Automatic Control*, vol. AC-23, no. 6, pp.1015-1020, Dec. 1978.

[6] I.D. Landau, "A survey of model reference adaptive technigues-theory and application," *Automatica*, vol. 10, pp. 353-379, Pergamon Press, 1974.

[7] I.D. Landau, "Unbiased recursive identifi-

cation using model reference adaptive technigues," *IEEE Trans. Automatic Control,* vol. AC-21, no. 2, pp. 194-202, Apr. 1976.

[8] C.R. Johnson, M.G. Larimore, J.R.. Treichler, B.O. Anderson, "SHARF convergence properties," *IEEE Trans. Acoustics, Speech, and Signal Processing,* vol. ASSP-29, no. 3, pp. 659-670, June, 1981.

[9] V.M. Popov, *Hyperstability of Control Systems.* Berlin Springer Verlag, 1973.

[10] T.A.C.M. Classen, W.F.G. Mecklenbrauker, "Comparison of the convergence of two algorithms for adaptive FIR digital filters," *IEEE Trans. Acoustics, Speech and Signal Processing,* vol. ASSP-29, no.3, pp. 670-678, June, 1981.