DFT 영역 CONSTANT MODULUS ALGORITHM 의 고속 적응 알고리듬

*양윤기, **박창현

* 수원대학교 정보통신공학과, ygyang@suwon.ac.kr

** 주) 티브이로직/TVLogic Co.,Ltd, chp7509@tclogic.co.kr

ABSTRACT

본 논문에서는 DFT 를 사용하여 고속으로 필터링을 수행하는 고속 CMA인 FBCMA (frequency domain block CMA) 를 제안한다. 이는 기존의 주파수 영역 적응필터 (FBLMS: frequency domain block LMS) [4, 5] 를 CMA 에 적용한 것인데, 커다란 차이점은 CMA 가 비선형 cost function 을 사용하기 때문에 이를 바로 FBLMS 에 적용할 수 없다는 점이다. 본 논문에서는 DFT 영역에서 CMA 의 cost function 을 최소화하는 새로운 비선형 적응기법을 소개하였으며, 이를 사용하여 올바른 적응등화가 이루어짐을 모의실험을 통하여 검증하였다. CMA와 제안한 방식의 계산량은 각각 $O(N^2)$, $O(N\log N)$ 으로서, 블럭의 크기 N 이 클수록 제안된 알고리듬이 유리하다.

1. 연구 배경

최근에 CDMA 와 같은 무선망에서 간섭을 제거하여 시스템의 용량과 성능을 개선 하고자하는 움직임이 활발해지고 있다. 이중에서 CMA (constant modulus algorithm) 은 가장 많이 사용되는 블라인드 알고리듬중에 하나이다. 이는 수신단에서 등화와 carrier recovery 가 독립적으로 수행될 수 있다는 장점과 함께, 소위 Bussgang 계열의 블라인드 알고리듬 중에서 가장 강인한 수렴특성을 갖는다는 장점이 보고되고 있어 최근 20년간 다양한 관심을 가진바 있다. 그런데, 심볼률이 증가 할수록 수신단에서 실시간 구현에 따른 계산량 부담이 증가하게 된다. 또한 CDMA 와 같은 간섭이 커다란 채널에서는 필터의 길이가 상대적으로 길어야 하므로 이를 실시간으로 구현하는데 계산량 부담이 크다 [3]. 따라서, 본 논문에서는 계산량을 감소하게 하고자 일반적인 적응 필터알고리듬에서 고안된 주파수 영역 적응필터 [5, 4] 를 CMA 로 확장하고자 한다.

그런데, FBLMS 에는 목적함수가 unimodal 이고, desired 신호에 대한 정보가 입력되어야 하는데, 블라인드 알고리듬은 이 두가지가 만족될 수 없다. 따라서, 새로운 적응기법이 필요한데 본 논문에서는 DFT 영역에서 새로운 적응 방정식을 도입한 주파수 영역 블럭 CMA (frequency domain block CMA: FBCMA) 를 제안한다.

블럭의 길이와 필터의 길이가 N 일때 기존의 CMA 는 $O(N^2)$ 의 계산량이 필요한 반면, 제안하는 FBCMA 는 $O(Nlog_2N)$ 의 계산량이 필요하다. CDMA 의 경우 CMA 의 필터차수가 spreading code 의 길이만큼 사용하는 경우가 많은데, 이경우 많은 계산상의 이점이 있음을 알 수 있다 [3].

[1]본 연구는 정보통신부의 2003년 정보통신기초 기술연구지원사업(정보통신연구진흥원)으로 수행한 연구결과입니다. (03-기초-0067)

2. DFT 영역의 CMA 적응 알고리듬

그림 1 에는 FBCMA 의 블럭선도가 제시되어 있다. 본 논문에서는 주파수 영역에서 선형필터링을 구현할 수 있는 알고리듬 중에 하나인 overlap save 방식을 사용하였다. Overlap save 방식에서 시간영역에서의 필터의 길이를 M 이라하고, 시간 영역에서의 적응필터의 계수는 M 차의 FIR 필터라 할때, 이를 후반에 0 을 채워놓은 벡터를 $\mathbf{c_k} = (c_k[0], \cdots c_k[M-1], 0, \cdots, 0)$ 이라고 하자. 이제 입력의 길이 N 인 블럭 데이타를 처리한다고 하자. 그러면, 수신신호 $x[n]$ 은 이전 블럭의 후단의 $(M-1)$ 심볼을 중복하고, 새로운 L 개의 심볼을 첨가하여 전체적으로 길이가 N 인 블럭을 만든다. 이를 N-point FFT 를 수행한 주파수 영역의 수신신호

$$\mathbf{X}_k = (X[kN], X[kN+1], \cdots, X[kN+N-1])^t \tag{1}$$

는

$$\mathbf{X}_k = \mathbf{F}\mathbf{x}_k \tag{2}$$

가 되는데, $\mathbf{F}$ 는 N-point DFT 를 표현하는 $N \times N$ 크기의 행렬이다. DFT 영역에서의 길이 N의 출력신호 벡터

$$\mathbf{Y}_k = (Y[kN], Y[kN+1], \cdots, Y[kN+N-1])^t \tag{3}$$

는 $\mathbf{X}_k$ 와 DFT 영역의 N-point 계수 벡터 $\mathbf{C}_k$,

$$\mathbf{C}_k = (C_k[0], C_k[1], \cdots, C_k[N-1])^t \tag{4}$$

의 곱으로 부터 얻어지는데,

$$\mathbf{C_k} = \mathbf{F}\mathbf{c_k} \tag{5}$$

이다. DFT 영역의 출력 $Y[kN+m]$ 은

$$Y[kN+m] = C_k[m]X[kN+m], \quad m = 0, 1, \cdots, N-1 \tag{6}$$

과 같이 주어진다. 시간영역에서의 출력결과를 얻기 위해서는 다음과 같이 $\mathbf{Y}_k$ 의 N-point IDFT 를 취한다. 즉

$$\mathbf{y}_k = \mathbf{F}^{-1}\mathbf{Y}_k \tag{7}$$

인데, 여기서 $\mathbf{y}_k$ 는 시간 영역상의 출력 벡터로