

LSF에서 LPC 계수를 구하는 개선된 알고리즘

김향진, 이기태, 함영희, 김형준, 임재윤

제주대학교 통신공학과

전화 : (064) 754-3635 / 팩스 : (064) 755-3610

The Revised Transform Algorithm from LSF to LPC

Kim hyang jin, Lee ki tae, Ham young hee, Kim hyoung jun, Lim jae yun

Dept. of Telecommunication Engineering Cheju University

Abstract

This paper proposes the LSF or LSP that is the method of using to transfer the speech parameters after processed the speech to LPC, which is digital coding transferring efficiently, for the best quality and the lowest bit rate of parameters. The new revised transform algorithm between LSF and LPC coefficients is proposed.

The proposed algorithm eliminates all multiplications, computes fewer operations, and reduces memory buffer sizes.

I. 서론

음성전송 시 모두 영점으로 이루어진 필터를 모델링하여 그 계수를 전송하며 이를 복원 시 음성을 합성할 때 모두 극점으로 이루어진 필터에 의한 안정적인 수행을 한다.

LPC(Linear Prediction Coding)는 음성 처리 시스템에 쓰이는 매우 효율적인 분석 기술이다. 음성을 코딩하고 합성 시스템을 구성하기 위해 LPC파라미터를 Autocorrelation, Covariance, Lattice 등등의 방법으로 얻는다. 이 LPC를 통하여 음성을 분석하고 적은 bit로 전송하여 합성할 수 있다.[4]

LPC의 계수는 양자화 하여야 할 값의 동적 영역을 알 수 없기에 많은 비트율이 쓰이게 되

며, 음성 파라미터 전송 시 매우 비효율적이다. 이 동적 영역의 제한을 위한 하나의 방법으로 LSF(Line Spectrum Frequency)기법이 사용된다. LSF는 LPC 필터 시스템 함수에서 비 선형적인 가공에 의해 양자화 하여 전송하는 파라미터의 동적 영역을 작은 영역으로 제한하여 효율적으로 전송하는 기법이다. 주파수 영역으로 바꾼 LSF의 동적 영역은 $0 \leq \omega \leq \pi$ 로 한정되어 있다. 전송하여야 하는 값의 동적 영역이 작은 범위에서 한정적이라는 것은 그 만큼 작은 영역에서 양자화가 이루어져 적은 채널 용량으로 양질의 음성 파라미터 전송이 이루어짐을 의미한다.[5]

LSF는 음성의 선형 예측 코딩(LPC)에 사용되는 분석 및 합성에서 대안의 파라미터이다. 본 논문에서는 직접 형태의 예측 계수와 LSF와의 변환 기법에 대한 개선된 방법을 제안한다. 분석을 위한 시스템 다항식은 단위 원 위에서 서로 엇갈려있는 근을 갖는 두 개의 짝수 차수의 대칭 다항식으로 변환된다. 단위 원 위의 이들 각 다항식의 응답은 Chebyshev 다항식의 확장으로 표현된다.[3][1][6] LSF는 실 함수의 실 근을 찾는 알고리즘에 의해 구해진 서로 엇갈려있는 근을 이용하여 찾을 수 있다. 방법은 매우 엄밀하며, 삼각 테이블의 저장 또는 삼각 함수의 계산을 피하는 형태로 사용할 수 있다. 예측 계수로의 LSF 복원은 Chebyshev의 확장인 근 인자의 표현을 통하여 유도된 효율적인 알고리즘을 사용한다.

II.LPC에서 LSF로의 변환알고리즘

우선 Minimum Phase LPC 다항식에서 시작한다.

$$A(z) = 1 - \sum_{k=1}^m a(k)z^{-k} \quad (2.1)$$

여기에서 m 은 LPC 필터의 차수이며, $a(k)$ 는 필터의 계수이다.

다음으로 대칭적(Symmetric)인 다항식과 비대칭적(Antisymmetric)인 다항식으로 구성한다.

$$\begin{aligned} F_1(z) &= A(z) + z^{-(m+1)}A(z^{-1}), \\ F_2(z) &= A(z) - z^{-(m+1)}A(z^{-1}) \end{aligned} \quad (2.2)$$

이 보조 다항식의 근은 LSF를 말한다. 이 다항식에서 구해지는 근 LSF는 Minimum Phase LPC다항식이라면 위치가 단위 원 위라는 것이 Soong and Juang[2]에 의해 보여졌다.

세 번째로, $z = \pm 1$ 에서의 근을 제거한다. 이는 (2.2)의 다항식의 근이 이 위치를 제외하고는 모두 공액 복소 쌍으로 존재하여 단위 원의 상 위 부분만 고려하여도 된다는 이점이 있기 때문이다.

$$\begin{aligned} G_1(z) &= \frac{F_1(z)}{1+z^{-1}} \quad \text{and} \\ G_2(z) &= \frac{F_2(z)}{1-z^{-1}} \quad m \text{ even} \end{aligned} \quad (2.3)$$

식(2.2)와 식(2.3)을 이용하여 풀어쓰면,

$$G_1(z) = \prod_{i=2,4,\dots,m} (1 - 2z^{-1} \cos \omega_i + z^{-2}) \quad (2.4.a)$$

$$G_2(z) = \prod_{i=1,\dots,m-1} (1 - 2z^{-1} \cos \omega_i + z^{-2}) \quad (2.4.b)$$

여기에서 $G_1(z), G_2(z)$ 를 Chebyshev 다항식과 근의 위치가 단위 원이라는 점을 이용하여 x 의 다항식으로 고친다.

$$x = \cos \omega \quad (2.5)$$

$$T_m(x) = \cos(m \arccos(x)) \quad (2.6)$$

$$T_m(x) = 2xT_{m-1}(x) - T_{m-2}(x) \quad (2.7)$$

$G_1(x), G_2(x)$ 에서 근 x_i 를 계산한다.

다섯 번째로 LSF를 구한다.

III.LSF에서 LPC로의 변환알고리즘

먼저 Kabal and Ramachandran[3]의 알고리즘을 살펴보면 다음과 같다.

전송된 LSF ω_i 를 다음과 같이 바꾼다.

$$x_i = \cos \omega_i \quad (3.1)$$

$$G_1'(x) = \prod_{k=0}^{k-1} 2(x - x_{2k}), \quad (3.2)$$

$$G_2'(x) = \prod_{k=0}^{k-1} 2(x - x_{2k+1})$$

식(3.2)는 근 x_i 의 인수분해 형태로 표현된 다항식 $G_1'(x), G_2'(x)$ 이다. 이를 확장함으로써 다항식의 계수를 구한다. 이 수행을 하기 위해 근 x_i 를 하나씩 대입하면서, 식(3.3)을 이용하여 Chebyshev 다항식의 형태로 바꾼다.

$$\begin{aligned} Y_{N+1}(x) &= 2(x - x_r)Y_N(x) \\ &= \sum_{k=-1}^{N+1} [c_{k-1} - 2x_r c_k + c_{k+1}] T_k(x) \end{aligned} \quad (3.3)$$

$T_n(x) = \cos n\omega$ 를 이용하여, cosine의 항목으로 바꾸고, $G_1'(\omega), G_2'(\omega)$ 다항식을 구한다.

$$\begin{aligned} G_1(\omega) &= G_1'(\omega) \times e^{-j\omega p}, \\ G_2(\omega) &= G_2'(\omega) \times e^{-j\omega p} \end{aligned} \quad (3.4)$$

그리고, Chebyshev의 정의 식(3.5)을 이용하여, $G_1(z), G_2(z)$ 의 계수를 구한다.

$$2 \cos k\omega = z^k + z^{-k} \quad (3.5)$$

다음으로, $G_1(z), G_2(z)$ 에 $z = \pm 1$ 에서의 근을 포함시켜, $F_1(z), F_2(z)$ 의 계수를 구한다. 그런 후에 다음의 식으로 LPC계수 $a(k)$ 를 구한다.

$$A(z) = \frac{1}{2} \{F_1(z) + F_2(z)\} \quad (3.6)$$

이 알고리즘은 Soong and Juang[2]에 비해 많이 개선된 것이지만, 본 논문에서는 LPC에서 LSF로 변환할 때 쓰였던 정의 식 (2.4.a)와 (2.4.b)에 직접 대입하고 계수의 규칙성을 이용하여 더욱 적은 연산을 요하는 알고리즘으로 개선하였다. 즉 근 x_i 를 이용하고, Chebyshev 다항

식 형태로 바꾸는 (3.2)와 (3.4)를 직접 (2.4)를 이용함으로 연산 과정을 개선한다.

본 논문에서의 알고리즘은 다음과 같다. 식 (2.4)를 다시 써 보면 다음과 같고,

$$G_1(z) = \prod_{i=2,4,\dots,m} (1 - z^{-1} \cos \omega_i + z^{-2}), \quad (3.7)$$

$$G_2(z) = \prod_{i=1,\dots,m-1} (1 - z^{-1} \cos \omega_i + z^{-2})$$

식 (3.7)을 다음 식으로 풀어 쓸 수 있다.

$$G_1(z) = \sum_{n=0}^k (b_n + b_{n-1}x_{2k} + b_{n-2})z^{-n}, \quad (3.8)$$

$$G_2(z) = \sum_{n=0}^{k-1} (b_n + b_{n-1}x_{2k+1} + b_{n-2})z^{-n}$$

식 (3.8)에 x_i 를 직접 대입하여 다항식을 확장 하게 되면, $g_1(n) = g_1(m-n)$ 임을 알 수 있다. $g_2(n)$ 도 마찬가지이다. 그러므로, 계수를 구할 때 LPC 차수 m 까지 구하지 않고, $m/2$ 인 p 까지만 구하고, 이를 앞의 식에 적용시켜 모든 계수를 구할 수 있다.

식 (3.8)에 x_i 를 직접 대입하여 계수를 구하므로써, Kabal 알고리즘에서 x 의 다항식으로 고친 후에 Chebyshev의 다항식으로 변환하는 과정을 한 단계로 줄여, Kabal의 알고리즘에서 곱셈 및 덧셈의 연산 숫자를 개선한다.

그림 1.에서 이 단계를 거쳐 구하여 진 다항식 (3.8)의 근의 위치를 도시하였다. 그림에서 근의 위치가 공액 복소 쌍으로 나타남을 알 수 있다.

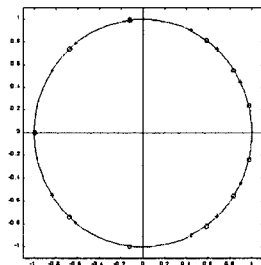


그림 1. $G_1(z), G_2(z)$ 근의 위치
Fig 1. The Root Position of $G_1(z), G_2(z)$

다음으로, $G_1(z), G_2(z)$ 에 $z = \pm 1$ 에서의 극점을 포함시켜, LPC계수 $a(k)$ 를 구한다.

$$A(z) = \frac{1}{2} \{G_1(z) + G_2(z) + z^{-1}(G_1(z) - G_2(z))\} \quad (3.9)$$

그림 2.에서 (3.9) 다항식의 근의 위치를 도시 하였다.

그림 2.에서 살펴보듯이 LPC 다항식의 근들의 위치가 모두 단위 원안에 위치하여 Minimum Phase 다항식임을 확인 할 수 있다.

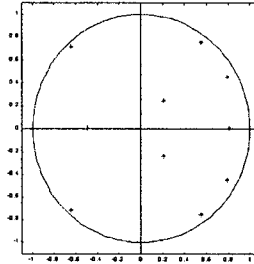


그림 2. $A(z)$ 근의 위치
Fig 2. The Root Position of $A(z)$

IV. 알고리즘 흐름도 및 수행결과

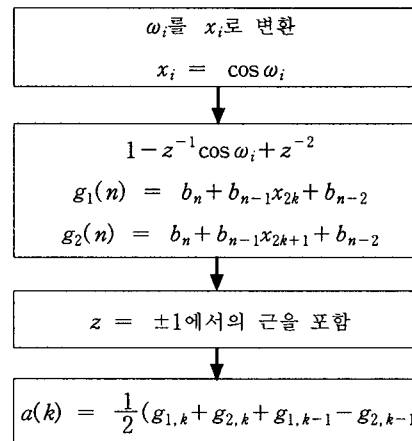


그림 3. LSF에서 LPC의 변환 알고리즘
Fig 3. LSF to LPC Transform Algorithm

그림 3.에서 LSF에서 LPC로의 변환 알고리즘을 도시하였는데 여기에서 살펴볼 수 있듯이 식 (3.7)을 직접 이용하였다.

LSF를 LPC로 변환시의 알고리즘을 식(3.7)으로 직접 대입으로 전체 연산의 과정이 줄어들

고, 연산에서 요구되어지는 곱셈 및 덧셈의 연산이 현저히 줄어들었다. 이런 알고리즘을 이용하여 연산 시간의 감소를 가져와 음성 처리 시간의 감소를 얻었다.

이를 표 1.에서 전에 제안된 Kabal의 알고리즘과 비교해 보았다.

표 1. Kabal 알고리즘과 개선된 알고리즘에서의 $G_1(z)$ 의 계수 연산 숫자

Table 1. G_1 'S COEFFICIENTS OPERATIONS COUNT FOR THE KABAL ALGORITHM AND THE REVISED ALGORITHM

Order	[3]'s-multiplication	[3]'s-add	new-multiplication	new-add
4	3	12	1	3
6	7	29	3	8
8	12	51	6	15
10	18	78	10	24
12	25	109	15	35

본 논문에서는 홀수 차수의 LPC 필터는 고려치 않고, 많이 이용되고 있는 짝수 차수에 대한 개선 알고리즘이다.

이 표를 살펴보면 가장 많이 이용되고 있는 10차를 예를 들면 곱셈 연산이 18에서 10개로 줄고, 덧셈 연산이 78개에서 24개로 줄었음을 확인할 수 있다.

V. 결론

본 논문에서는 LSF에서 LPC로 변환하는 효율적인 알고리즘에 대해 제안하였다. 기존의 방법에 비해 곱셈과 덧셈의 연산 과정을 줄였으며, 10차 LPC인 경우 곱셈은 40%, 덧셈은 1/3정도로 줄일 수 있어 본 알고리즘의 유용성을 보였다. 본 알고리즘은 CELP등 최근에 저 비트 실시간 음성처리 시스템에 유용하게 사용되리라 사료된다. 추후 연구 방향으로는 DSP상에서 처리 시 계수들 간의 연관성을 고려한 병렬처리를 수행함으로써 실시간 처리 효율을 높일 수 있는 방법을 고려하는 것이다.

참고 문헌

- [1] Chung-Hsien Wu and Jau-Hung Chen, "A novel two-level method for the computation of the lsp frequencies using a decimation-in-degree algorithm," *IEEE Trans, on Speech and Audio Processing*, vol.5, no.2 pp.106~115, March 1997
- [2] F.K. Soong and B.H.Juang. "Line spectrum pair(LSP) and speech data compression," *Proc. IEEE int. Conf, on Acoustics, Speech, and Signal Processing*, vol.1 pp.1.10.1~1.10.4, May 1984
- [3] Peter Kabal and Ravi Prakash Ramachandran, "The Computation of Line Spectral Frequencies Using Chebyshev Polynomials," *IEEE Trans, on Acoustics, Speech, and Signal Processing*, VOL. ASSP.34, No.6, December 1986
- [4] H. Wakita, "Linear prediction voice synthesizers: Line spectrum pairs(LSP) is the newest of several techniques," *Speech Technol*, Fall 1981
- [5] G. S. Kang and L. J. Fransen, "Application of line spectrum pairs to low bit rate speech encoders," in *Proc. Int. Conf. Acoust. , Speech, Signal Processing*, Tempa, FL. pp. 7.3.1~7.3.4 April 1985
- [6] L. Fox and I. B. Parker, *Chebyshev Polynomials in Numerical Analysis*, London, England: Oxford University Press, 1968