

AMR 음성 압축기를 위한 효율적인 코드북 검색 방법

이도윤, 박호중

광운대학교 전자공학과

Efficient Codebook Search Method for AMR Speech Codec

Doyoon Lee and Hochong Park

Dept. of Electronics Engineering

Kwangwoon University

E-mail : hcpark@mail.gwu.ac.kr

요약문

ACELP 구조의 음성 압축기는 우수한 음질을 제공하지만 최적의 코드 벡터를 구하기 위한 계산량이 상당히 많은 단점이 있다. 이를 해결하기 위해서 본 논문에서는 AMR 음성 압축기의 코드북을 매우 효율적으로 검색하는 새로운 방법을 제안한다. 제안하는 코드북 검색 방법은 완전 순차적인 검색 방법을 사용하여 대략적인 코드 벡터를 구하고, 코드 벡터의 각 펄스들의 중요도를 계산하여 중요도가 낮은 펄스를 새로운 펄스로 교환하는 펄스 교환 과정을 수행하여 코드 벡터의 성능을 향상시키는 방법을 사용한다. 또한, AMR 음성 압축기의 구조에 맞도록 트랙별로 이동하면서 순차적으로 코드북을 검색하여 다수의 대략적인 코드벡터를 찾은 후, 각 코드 벡터에 대하여 펄스 교환 과정을 수행하여 최적의 코드 벡터를 구한다. 제안한 코드북 검색 방법을 AMR 음성 압축기의 모든 모드에 적용하여 코드북 검색을 위한 계산량과 성능을 측정하였으며, 모든 모드에 대하여 매우 적은 계산량으로 동등한 성능을 가지는 것을 확인하였다.

1. 서론

사람의 음성은 우리에게 가장 자연스러운 통신 수단으로 사용되어 왔다[2]. 이런 음성 신호를 보다 적은 양의 Bit로 표현함으로써 데이터의 양을 줄이기 위해 음성 압축기가 개발되었고, 현재와 같이 이동 통신 기술

에 대한 수요가 폭발적으로 증가함에 따라 알고리즘에 대한 연구도 활발히 진행되고 있다. 또한 최근에 인터넷 사용자가 급증하면서 음성위주의 통신망과 데이터망의 통합이 이루어짐에 따라, 음성 압축 알고리즘도 다양한 통신망을 통해 양질의 음성 정보를 제공할 수 있는 방법이 요구되고 있다.

현재 디지털 이동 통신 시스템에서 사용되는 표준 음성 압축기들은 CELP(Code Excited Linear Prediction) 구조에 기반을 두고 있으며, 현재 IMT-2000과 인터넷망을 이용한 음성 신호의 전송 및 저장을 위한 표준 압축기들은 ACELP(Algebraic CELP) 구조를 가지는 음성 압축기를 채택하고 있다. ACELP 방식의 코드북은 펄스 구조를 가지며 정해진 수의 펄스만으로 여기신호를 만든다. 대표적인 음성 압축기로는 AMR, G.729, G.723.1, EVRC등이 있다.

AMR은 ACELP 구조에 기본을 두고 있으며, 12.2kbit/s에서 4.75kbit/s까지 8개의 전송률을 가짐으로써 MR-ACELP(Multi-Rate ACELP)로 언급되기도 한다[1]. ACELP 구조의 음성 압축기는 우수한 음질을 제공하지만, 최적의 코드 벡터를 구하기 위한 계산량이 상당히 많은 단점이 있다. 이를 해결하기 위해 본 논문에서는 AMR 음성 압축기의 코드북을 매우 효율적으로 검색하는 새로운 방법을 제안한다. 부호화 과정에서 가장 많은 계산량을 차지하는 고정 코드북 검색에서의 계산량을 줄임으로써, 하나의 DSP Chip에 다채널을 효율적으로 구현하여 시스템 용량을 증가시키고 동작시의 소비전력을 줄여 단말기의 사용시간을 늘릴 수 있는 장점이 있다.

본 논문의 구성은 제 2장에서 AMR 음성 압축기에

대해 설명하고, 제 3장에서 코드북의 계산량을 감소시키기 위해 제안한 방법을 설명하고, 제 4장에서 성능 비교를 하고, 제 5장에서 결론을 맺는다.

2. AMR 음성 압축기

AMR은 제 3세대 이동 통신인 IMT-2000에서 표준으로 채택된 음성 압축기이다. ACELP에 기본을 두고 있으며, 주어진 채널 환경에서 최적의 음질을 유지하기 위해 4.75kbit/s에서 12.2kbit/s까지 8개의 모드로 구성된다. AMR은 통신 채널에 따라 소스 코딩과 채널 코딩의 Bit 할당을 가변 하여 통화 품질을 향상시켰다. 채널 당 할당된 Bit 수는 항상 고정되기 때문에 채널상의 오류가 많을 경우에는 채널 코딩에 많은 Bit를 할당하고, 소스 코딩에는 상대적으로 적은 Bit를 할당한다. 채널상의 오류가 적을 경우에는 이와 반대로 Bit를 할당함으로써 고품질의 통신 서비스를 제공한다. 20ms의 프레임 크기를 가지고, 4개의 부프레임(5ms)을 가진다. 모드별 할당되는 비트 수는 표 1과 같다. AMR의 입력은 8KHz로 샘플링된 16Bit Linear PCM을 13Bit Linear PCM으로 변환한 신호이다.

Mode (kbit/s)	LSP set	Pitch delay	Pitch Gain	Algebraic codebook	Codebook gain	Total
12.2	38	30	16	140	20	244
10.2	26	26		124	28	204
7.95	27	28	16	68	20	159
7.40	26	26		68	28	148
6.70	26	24		56	28	134
5.90	26	24		44	24	118
5.15	23	20		36	24	103
4.75	23	20		36	16	95

표 1. AMR의 모드별 Bit 할당

2.1 LP(Linear Prediction) 분석

입력신호는 전 처리 과정을 거친 후, 두 개의 다른 크기를 갖는 해밍 윈도우로 구성된 두 가지의 30ms 버 대칭 윈도우를 통과시킨다. 12.2kbit/s 모드에서는 프레임마다 선형 예측을 두 번 실행하는데, lookahead는 사용되지 않으며, Levinson-Durbin 알고리즘을 이용해 계수를 구한다. 나머지 모드는 프레임마다 한 번의 선형 예측을 수행하는데, 5ms lookahead가 사용된다. 계산된 LPC 계수는 Cosine 영역에서 Chebyshev 다항식을 이용하여 양자화(Quantization) 및 보간(Interpolation) 특성이 좋은 LSP(Line Spectral Pair)로

변환한 후 양자화 한다. 12.2kbit/s 모드는 LSP 2Set씩 묶어서 SMQ(Split Matrix Quantization)을 사용하는데 5개의 Submatrix로 나누어 7, 8, 8+1, 8, 6Bit로 Jointly 양자화 한다. 나머지 모드에서는 3, 3, 4의 Subvector로 나누어서 양자화 한다.

2.2 적응 코드북(Adaptive Codebook)

적응 코드북 검색은 계산량을 줄이기 위해, 고정 코드북 신호를 0으로 하고 AbS(Analysis-by-Synthesis) 방식에 의해 수행된다. 피치 검색을 간단하게 수행하기 위해서 개루프(Open-loop) 검색을 통해 정수 지연값을 찾은 후, 이 값의 주변에서 세밀하게 폐루프(Closed-loop) 검색을 수행한다. 4.75kbit/s와 5.15kbit/s 모드는 프레임 당 한 번, 나머지 모드는 두 번의 개루프 검색을 수행한다. 폐루프 검색은 서브 프레임 단위로 수행되고, 12.2kbit/s 모드는 1/6, 나머지 모드는 1/3의 분해능을 사용한다.

2.3 고정 코드북(Fixed Codebook)

고정 코드북 검색도 적응 코드북과 마찬가지로 AbS 방법을 이용해서 코드북 인덱스와 이득을 구한다. 각각 가중된 음성 신호와 각각 가중된 합성 신호 사이의 평균 제곱 오차(Mean Square Error)를 최소화시키는 값을 찾아 수행한다. 이것은 다음의 A_k 값을 구해 이를 최대로 하는 펄스의 위치와 크기를 선택함으로써 고정 코드북이 정해진다.

$$A_k = \frac{(C_k)^2}{E_{DK}} = \frac{(d^t c_k)^2}{c_k^t \Phi c_k} \quad (1)$$

여기서, c_k 는 인덱스 k에서의 고정 코드 벡터이다. d 는 목표 신호와 충격 응답 함수의 자기상관함수, Φ 는 충격 응답 함수의 자기상관함수, d^t 와 c_k^t 는 각각 d 와 c_k 의 전치행렬이다.

고정 코드북은 여러 개의 펄스로 이루어진 고정 대수 코드북이며, 모드별 트랙과 펄스 수는 표 2와 같다.

Mode (kbit/s)	# of Track	# of Pulse	Total Bits
12.2	5	10	35
10.2	4	8	31
7.95			
7.40	4	4	17
6.70	3	3	14
5.90	2	2	11
5.15			
4.75	2	2	9

표 2. AMR 고정 코드북 구조

ACELP 구조의 음성 압축기는 각각의 펄스의 크기가 1과 -1로 이루어져 있고, 펄스 위치를 제외한 나머지는 모두 0값을 가지므로 고속 연산이 가능하다. 펄스의 수가 많아질수록 고정 코드북의 경우의 수가 많아져 음질이 좋아지기는 하지만, 코드북 검색을 위한 계산량이 많아진다는 단점이 있다.

2.4 디코더(Decoder) 구조

음성 신호 복원 시에는 코드북 이득과 인덱스로 여기 신호를 만들고, 피치 주기와 이득을 이용해서 피치를 복원하며, 복원된 신호와 LPC 합성 필터를 사용해서 우리가 원하는 신호를 만들어 낸다. 합성된 신호는 인간의 청각적 특성을 이용한 Postfilter를 사용해서 음질을 개선시킨다.

3. 제안한 코드북 검색 방법

3.1 AMR 표준의 코드북 검색 방법

AMR 음성 압축기의 고정 코드북 검색 방법은 크게 두 가지로 나뉜다. 12.2kbit/s와 10.2kbit/s 모드는 한 트랙에 두 개씩의 펄스로 구성되는데, nested loop 방식으로 펄스를 두 개씩 쌍으로 묶어 검색하며 각각의 펄스의 쌍들은 순차적으로 검색한다. 나머지 모드는 첫 번째 펄스의 루프 안에서 나머지 펄스들을 순차적으로 검색하는 nested loop를 형성하여 코드 벡터를 구한다. 또한 5.90kbit/s, 5.15kbit/s와 4.75kbit/s 모드를 제외한 나머지 모드에서는 음질의 향상을 위해 트랙에 따라 펄스를 옮기면서 위의 과정을 수행하여 다수의 코드 벡터를 구한 후, 그 중에서 식 (1)의 A_k 를 최대로 하는 최적의 코드 벡터를 구하는 방식을 사용한다.

3.2 제안한 방법

제안한 코드북 검색 방법은 펄스 교환 방법[3]에 기초한다. 펄스 교환 방법은 첫 단계에서 완전 순차 검색 방법을 이용하여 대략적인 코드 벡터를 매우 빠르게 찾는다. 이 방법에 의해 코드북 검색 시에 계산량을 많이 차지하는 nested loop를 완전히 제거함으로써 많은 계산량을 줄일 수 있다. 두 번째 단계에서는 성능의 향상을 위해, 펄스의 중요도를 결정하고 중요도에 따라 중요도가 낮은 펄스를 새로운 펄스로 교환하는 펄스 교환 과정을 수행함으로써 최적의 코드 벡터를 구한다.

본 논문에서 제안된 방식은 먼저 AMR 음성 압축기의 구조에 맞도록 트랙을 이동하면서 완전 순차 검색 방법으로 대략적인 다수의 코드 벡터를 매우 빠르게 찾은 후, 중요도에 따라 펄스 교환 과정을 수행하여 최적

의 코드 벡터를 찾는 방식이다. 새로 구한 코드 벡터는 펄스 교환 전의 코드 벡터보다 A_k 값이 클 경우에만 선택이 되므로, 교환 전의 코드 벡터보다 우수한 성능을 갖는다. 펄스 교환을 몇 번 반복 수행하면 성능이 향상된 최적의 코드 벡터를 찾을 수 있다. 그림 1에 대략적인 블록도를 나타내었다.

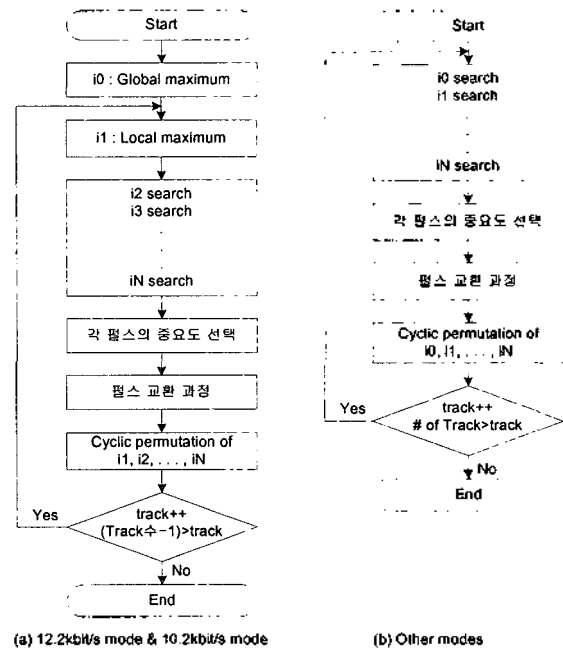


그림 1. 제안된 코드북 검색 방법의 블록도

12.2kbit/s 모드에서 코드북 검색을 위한 계산량을 비교하면, AMR 표준의 경우 펄스의 트랙이 네 번 바뀌고 한 트랙에 펄스가 위치할 수 있는 위치가 8개가 있으며 두 개의 펄스의 쌍으로 묶인 nested loop가 네 번 있으므로 $4 \times (8 \times 8 \times 4 + 1) = 1028$ 의 계산량이 필요하다. 반면 제안된 방법의 계산량은 다음과 같다. 완전 순차 검색 방법의 계산량 $8 \times 8 + 1 = 65$, 중요도에 따라 새로운 펄스를 선택하는 펄스 교환 과정의 계산량은 $(8 + 8) \times (\text{펄스 교환 회수})$ 이다. 따라서 펄스 교환을 한 번만 수행할 경우의 계산량은 네 번의 트랙 이동이 있으므로 $4 \times (65 + 16 + 1) = 328$ 이 된다. 이는 트랙의 이동이 없이 펄스 교환 방법을 적용했을 때보다 4배로 계산량이 증가된 수치이지만, AMR 표준의 코드북 검색 계산량의 1/3로 매우 적은 계산량을 가진다.

4. 성능 평가

제안한 방법을 AMR 음성 압축기의 모든 모드에 적용하여 코드북 검색을 위한 계산량과 주관적, 객관적

성능을 측정하였다. 실험 데이터로는 한국어 남성과 여성, 영어 남성과 여성 4개의 음성으로 각각 2분장을 가지고 측정하였다. 계산량 측정은 식 (1)의 A_k 를 계산하는 회수로서 측정하였고, 모드별 코드북 검색의 계산량 비교는 표 3에 정리되어 있다.

Mode (kbit/s)	AMR 표준	적용된 방식	
		계산량(비율)	필스교환회수
12.2	1028	464(45.1%)	3
10.2	1204	448(37.2%)	3
7.95	1600	268(16.8%)	2
7.40	1600	268(16.8%)	2
6.70	1632	216(13.2%)	1
5.90	576	198(34.4%)	1
5.15	144	56(38.9%)	1
4.75	144	56(38.9%)	1

표 3. 모드별 고정 코드북 검색의 계산량 비교

객관적 성능 측정을 위해서 AMR 표준과 제안된 방법의 Segmental SNR(segSNR) 평균값을 측정하여 비교하였고, 표 4에 정리하였다.

모드 (kbit/s)	AMR 표준	한국어		영어	
		여성	남성	여성	남성
12.2	AMR 표준	15.385	15.512	14.479	13.213
	적용된방식	15.565	15.638	14.683	13.294
10.2	AMR 표준	14.455	14.565	12.370	12.371
	적용된방식	14.418	14.650	12.446	12.44
7.95	AMR 표준	11.985	12.168	11.251	9.684
	적용된방식	12.036	12.168	11.353	9.740
7.40	AMR 표준	11.797	11.937	10.945	9.512
	적용된방식	11.933	11.973	11.039	9.581
6.70	AMR 표준	11.096	10.896	10.100	8.834
	적용된방식	11.157	10.908	10.151	8.875
5.90	AMR 표준	10.048	9.782	8.869	7.746
	적용된방식	10.058	9.807	8.878	7.871
5.15	AMR 표준	8.936	9.068	8.309	7.097
	적용된방식	9.008	9.111	8.436	7.102
4.75	AMR 표준	8.531	8.470	7.799	6.738
	적용된방식	8.605	8.585	7.819	6.783

표 4. 모드별 segSNR 비교(단위 : dB)

주관적인 음질 평가는 영어 남성과 여성, 한국어 남성과 여성의 4가지로 AMR 표준의 신호와 제안된 코드북 검색 방법을 사용한 AMR의 신호를 임의의 순서로 들려주고 어느 쪽 음질이 더 좋은지 결정하는 방법으로

실시하였다. 주관적인 음질 평가 결과는 표 5에 정리하였다.

모드	A is better	No difference	B is better
12.2 kbit/s	8.1%	74.4%	17.5%
10.2 kbit/s	8.1%	73.1%	18.8%
7.95 kbit/s	7.5%	75.0%	17.5%
7.40 kbit/s	5.0%	83.8%	11.2%
6.70 kbit/s	11.3%	80.0%	8.1%
5.90 kbit/s	5.0%	82.5%	12.5%
5.15 kbit/s	17.5%	75.0%	7.5%
4.75 kbit/s	6.3%	81.2%	12.5%

표 5. 주관적 음질 평가(A:제안된 방식, B:AMR 표준)

5. 결론

본 논문에서 AMR의 고정 코드북을 보다 효율적으로 검색하여 계산량을 줄이는 방법을 제안하였다. AMR의 구조에 맞도록 트래클을 이동하면서 완전 순차 검색 방법에 의해 대략적인 다수의 코드 벡터를 구한 후, 중요도에 따라 펄스 교환 과정을 수행하여 최적의 코드 벡터를 구한다. 음성 압축기의 성능을 다양한 음성 신호에 대하여 객관적 방법과 주관적 방법을 사용하여 측정한 결과, 모든 모드에 대하여 매우 적은 계산량으로 동일한 성능을 가지는 것을 확인하였다.

참고 문헌

- [1] ETSI EN 301 703, "Digital cellular telecommunication system(Phase 2+); Adaptive Multi-Rate (AMR); Speech processing function", 1999.
- [2] 한진수, *음성신호처리*, 오성미디어, 2000.
- [3] Hochong Park, "Efficient codebook search method for EVRC speech codec", *IEEE Signal Processing Letters*, vol.1, no.1, pp.2-3, January, 2000.
- [4] Panos E. Papamichalis, *Practical Approaches To Speech Coding*, Prentice Hall, 1987.