

Multi-frame AR model을 이용한 LPC 계수 양자화

Quantization of LPC Coefficients Using a Multi-frame AR-model

정원진 · 김무영

(Wonjin Jung and Moo Young Kim)

세종대학교 정보통신공학과

(접수일자: 2011년 11월 14일; 수정일자: 2011년 12월 26일; 채택일자: 2012년 1월 20일)

초 록: 음성코딩 시 성도는 Linear Predictive Coding (LPC) 계수를 이용해서 모델링 한다. 일반적으로 LPC 계수는 양자화와 선형보간 관점에서 유리한 Line Spectral Frequency (LSF) 파라미터로 변경하여 사용한다. 10차 이상의 다차원 LSF 데이터를 벡터 양자화를 이용하여 직접 코딩하게 되면 벡터 내 상관관계 (intra-frame correlation)를 모두 이용할 수 있으므로 rate-distortion 관점에서는 높은 효율을 기대할 수 있다. 하지만, 계산량과 메모리 요구량이 높아져서 실제 코딩 시스템에서는 사용할 수 없게 되므로, 차원을 나누어 압축하는 Split Vector Quantization (SVQ)이 이용된다. 또한, LSF 데이터는 과거 벡터와의 벡터 간 상관관계 (inter-frame correlation)가 높으므로, 이를 이용한 Predictive Split Vector Quantization (PSVQ)이 사용되고 있다. PSVQ는 SVQ 보다 높은 rate-distortion 성능을 보인다. 본 논문에서는 음성 저장 장치를 위한 최적의 PSVQ를 구현하기 위해서 다수의 과거 프레임 정보와의 벡터 간 상관관계 (inter-frame correlation)를 고려한 Multi-Frame AR-model 기반 SVQ (MF-AR-SVQ)를 제안하였다. 기존 PSVQ와 비교해 보았을 때, MF-AR-SVQ는 계산량과 메모리 요구량의 큰 증가 없이, 평균 spectral distortion 관점에서 약 1비트의 성능 향상을 보였다.

핵심용어: LSF, LPC, 양자화, 벡터 양자화, AR model

투고분야: 음성처리 분야(2.2)

ABSTRACT: For speech coding, a vocal tract is modeled using Linear Predictive Coding (LPC) coefficients. The LPC coefficients are typically transformed to Line Spectral Frequency (LSF) parameters which are advantageous for linear interpolation and quantization. If multidimensional LSF data are quantized directly using Vector-Quantization (VQ), high rate-distortion performance can be obtained by fully utilizing intra-frame correlation. In practice, since this direct VQ system cannot be used due to high computational complexity and memory requirement, Split VQ (SVQ) is used where a multidimensional vector is split into multiple sub-vectors for quantization. The LSF parameters also have high inter-frame correlation, and thus Predictive SVQ (PSVQ) is utilized. PSVQ provides better rate-distortion performance than SVQ. In this paper, to implement the optimal predictors in PSVQ for voice storage devices, we propose Multi-Frame AR-model based SVQ (MF-AR-SVQ) that considers the inter-frame correlations with multiple previous frames. Compared with conventional PSVQ, the proposed MF-AR-SVQ provides 1 bit gain in terms of spectral distortion without significant increase in complexity and memory requirement.

Key words: LSF, LPC, Quantization, VQ, AR model

ASK subject classification: Speech Signal Processing (2.2)

1. 서 론

효율적인 음성 신호의 압축을 위해 사람의 vocal tract 필터는 Linear Predictive Coding (LPC) 계수에 의

해 모델링 된다. 적은 비트율에서 LPC 계수를 직접 양자화하면 스펙트럼 왜곡이 심해지고 필터가 불안정해질 수 있으므로, Line Spectral Frequency (LSF)로 변경하여 양자화 하는 방식이 제안되었다^[1-9].

Vector Quantization (VQ) 방식은 입력 데이터의 차

*Corresponding author: 김무영 (mooyoung@sejong.ac.kr)
143-747 서울시 광진구 군자동 세종대학교 정보통신공학과
(전화: 02-3408-3695; 팩스: 02-3408-4330)

원 간 상관관계를 효율적으로 사용할 수 있으므로 Scalar Quantization (SQ) 방식에 비해 rate-distortion performance가 우수한 장점이 있다. 하지만, 입력 데이터의 차원이 증가할수록 계산량과 메모리 요구량이 크게 증가하게 되는 단점이 있다. 8 kHz로 샘플링된 음성신호의 표현을 위해서는 일반적으로 10차 LSF 데이터를 사용하게 되는데, 10차 LSF 데이터는 계산량과 메모리 요구량 때문에 직접 VQ를 적용하기에는 문제가 있다. 그러므로 이를 해결하기 위해 하나의 벡터를 여러 개의 부벡터로 나누어 각 부벡터를 독립적으로 양자화 하는 Split VQ (SVQ)가 제안되었다^[3,4]. 이 방법을 사용하면 계산량과 메모리 요구량은 감소하지만, 부벡터 간의 상관관계가 부분적으로 이용되지 못하는 split loss가 발생하게 되어 rate-distortion performance는 저하된다.

SVQ의 성능을 올리기 위한 방법 중에는 Switched SVQ (SSVQ)^[5,6], Multi-Stage VQ (MSVQ)^[7], Predictive SVQ (PSVQ)^[8,9] 등이 있다. 그중 Differential Pulse Code Modulation (DPCM) 개념을 사용하는 PSVQ는 현재 frame과 이전 frame 간의 차이값을 양자화하는 방법이다. 이 방법을 사용하면 LSF 데이터의 inter-frame 상관관계를 고려하기 때문에 SVQ보다 성능이 좋아지게 된다. 과거 frame과의 차이값을 양자화 할 때, 과거값에 대한 가중치는 autoregressive (AR) 계수를 이용하는 것이 최적의 방법이라고 알려져 있다^[8]. 관측 가능한 과거 frame의 개수가 늘어날수록 현재 frame의 예측 성능이 점점 증가하지만, 채널 에러에 더욱 민감할 뿐만 아니라 계산량 문제도 있다고 알려져 있기 때문에 과거 하나의 frame 정보에 대한 상관관계를 사용하는 것이 일반적이었다^[13].

하지만, 채널 에러가 자주 발생하지 않는 응용처에 대해서는 과거 다수의 frame 정보를 이용하는 multi-frame AR model이 성능을 올리는데 효과적이다. 따라서, 본 논문에서는 multi-frame AR model에 대해서 rate-distortion performance 관점과 complexity 관점에서 보다 구체적으로 살펴보고자 한다. 국가 재난 시스템 등 콜센터에서는 대량의 음성 통화 내용을 저장하고 있으며 이 경우에는 채널 에러의 영향은 미미하다. 이 밖에 Text-To-Speech (TTS) 데이터베이스의 압축, Tapeless Answering Device (TAD), Voice Recorder

등에서도 채널 에러가 거의 발생하지 않으므로, intra-frame 상관관계만을 이용하는 SVQ보다는 inter-frame 상관관계를 이용하는 multi-frame AR model을 사용하는 방식이 바람직하다 할 수 있다.

II장에서는 VQ의 계산량 문제를 고려하여 설계된 SVQ와 inter-frame 상관관계를 이용하는 PSVQ에 대해 설명하고, 본 논문에서 제안하는 Multi-Frame AR-model 기반의 SVQ (MF-AR-SVQ)를 설명한다. III장에서는 기존의 SVQ 및 PSVQ와 제안하는 MF-AR-SVQ의 성능을 spectral distortion (SD) 관점에서 비교 분석하였고, IV장에서는 결론을 맺도록 하겠다.

II. 본 론

2.1 기존 양자화 방식

일반적인 데이터 압축 기술은 코드북 학습을 위해서 Generalized Lloyd Algorithm (GLA)을 사용한다^[10]. 이 알고리즘은 N 개의 centroid들을 임의로 분포시키고, 입력 데이터 각각에 대해서 N 개의 centroid들과의 왜곡을 비교하여 왜곡이 가장 적은 centroid 값으로 매핑함으로써 각각의 centroid 값에 대한 보로노이 영역을 설정한다. 각 보로노이 영역 내의 입력 데이터들의 평균을 구하여 평균 값을 새로운 centroid로 설정한다. 위 과정을 반복하여 최적의 centroid를 찾는 알고리즘이 GLA이다. GLA에서는 N 개의 centroid들을 임의로 분포시키는데 비해, Linde-Buzo-Gray (LBG) 알고리즘의 경우에는 전체 데이터의 평균값을 1개의 centroid로 선언하고, centroid를 2의 지승으로 분할하면서 GLA를 수행하게 된다^[11].

VQ는 차원이 높아질수록 계산량과 메모리 요구량이 급격히 증가하기 때문에 적절하게 차원을 나눠서 각각의 부벡터를 VQ 하게 되는데 그 방법을 SVQ라 한다^[2]. SVQ를 사용하면 VQ 보다 계산량은 줄어들지만, 나뉜 부벡터 간의 상관관계를 고려하지 않으므로 차원 전체를 VQ 하는 방법보다 성능이 떨어지게 된다. 이러한 손실이 있어도 계산량과 메모리 요구량의 한계 때문에 전체 차원을 한 번에 VQ 하는 방법보다는 부벡터로 나눠서 VQ하는 SVQ가 널리 이용되고 있다.

본 논문에서 언급하는 PSVQ는 이전 frame에 가중

치를 주지 않고 현재 frame과의 차를 양자화 하는 방법을 소개한다. PSVQ는 다음 식과 같이 i 번째 frame의 LSF 데이터와 $i-1$ 번째 frame의 LSF 데이터의 차이인 잔차 e_i 를 양자화하는 방식이다^[8,9].

$$e_i = X_i - X_{i-1} \quad (1)$$

i 번째 frame과 직전 frame의 상관관계를 제거한 e_i 값의 엔트로피가 제거되기 전 초기 X_i 값의 엔트로피보다 낮기 때문에, LSF 데이터의 inter-frame correlation을 양자화에 이용하는 PSVQ가 이러한 상관관계를 이용하지 않는 SVQ보다 코딩효율이 높아지게 된다.

2.2 Multi-Frame AR-model based SVQ (MF-AR-SVQ)

과거 frame의 LSF 데이터를 이용하여 현재 frame의 LSF 데이터와의 상관관계를 제거할 때, (1)과 같이 과거 값을 직접 이용하는 것 보다는 AR 계수를 통해 현재 값을 예측하여 양자화에 적용하는 것이 더 바람직하다. 즉, MF-AR-SVQ에서는 과거 multiple-frames에서 구한 LSF 데이터로부터 현재 frame의 LSF 데이터를 예측하여 그 잔차만을 양자화하게 된다.

X_i 를 i 번째 frame의 LSF 데이터라고 할 때, 과거 P 개의 LSF 데이터로부터 예측된 i 번째 frame의 LSF 데이터인 \tilde{X}_i 는 다음과 같이 표현할 수 있다.

$$\tilde{X}_i = - \sum_{j=1}^P X_{i-j} \cdot a_j \quad (2)$$

여기서 $\{a_j\}_{j=1, \dots, P}$ 는 P 차 AR 계수를 나타낸다. 이 P 차 AR 계수를 이용하면 현재 frame과 다수의 과거 P 개 frame간의 상관관계를 추정할 수 있게 된다.

원신호 X_i 와 AR 계수를 이용하여 예측된 신호 \tilde{X}_i 의 오차신호는 다음과 같다.

$$\begin{aligned} E_i &= X_i + \sum_{j=1}^P X_{i-j} \cdot a_j \\ &= a^{(P)T} X_{i:i-P}^{(P+1)} \end{aligned} \quad (3)$$

이 식에서 T 는 행렬의 전치행렬이고, $a^{(P)}$ 는 AR 계수를 나타낸 $(P+1) \times 1$ 행렬 $[1, a_1, \dots, a_P]^T$ 이다. 그리고 $X_{i:i-P}^{(P+1)}$ 는 현재 frame과 과거 P 개 frame의 LSF 데이터를 나타낸 $(P+1) \times 1$ 행렬 $[X_i, X_{i-1}, \dots, X_{i-P}]^T$ 이다. (3)의 오차신호를 이용해서 Mean-Squared Error 값을 구한 에러분산 값은 다음 식을 통해 구할 수 있다.

$$\begin{aligned} \sigma_P^2 &= E[(a^{(P)T} X_{i:i-P}^{(P+1)})^2] \\ &= a^{(P)T} R^{(P)} a^{(P)} \end{aligned} \quad (4)$$

여기서 $R^{(P)}$ 는 LSF 데이터의 covariance matrix로써 다음과 같이 표현할 수 있다.

$$R^{(P)} = E \begin{bmatrix} X_i X_i & X_i X_{i-1} & & & \\ X_{i-1} X_i & X_{i-1} X_{i-1} & & & \\ & & \ddots & & \\ & & & X_{i-P} X_{i-P} & \\ & & & & \ddots \end{bmatrix} \quad (5)$$

식 (4)의 에러분산 σ_P^2 값을 최소화 시키는 P 차 AR 계수 $a^{(P)}$ 는 Lagrange multiplier를 이용하여 다음 식으로 정리가 가능하며, Levinson 알고리즘을 이용하면 효율적으로 $a^{(P)}$ 와 σ_P^2 를 구할 수 있다^[12].

$$R^{(P)} a^{(P)} = \begin{bmatrix} \sigma_P^2 \\ 0^P \end{bmatrix} \quad (6)$$

이 식에서 0^P 는 P 개의 0이 있는 것을 의미한다.

한편, (5)를 계산하는 방법으로는 (7)과 같은 autocorrelation 방법과 (8)과 같은 covariance 방법이 있다.

본 논문에서는, 위 두 방법 중에서 window w_i 를 사용하는 autocorrelation 방법을 통해 AR 계수를 구한다.

$$\begin{aligned} R_{ij} &= \frac{1}{N} \sum_{n=-\infty}^{\infty} w_{n-i} X_{n-i} w_{n-j} X_{n-j} \\ w_i &= 0 \text{ for } \forall i \in \dots, -1, 0, N+1, N+2, \dots \end{aligned} \quad (7)$$

$$C_{ij} = \frac{1}{N-P} \sum_{n=P+1}^N X_{n-i} X_{n-j} \quad (8)$$

Autocorrelation 방법은 prediction gain 관점에서는

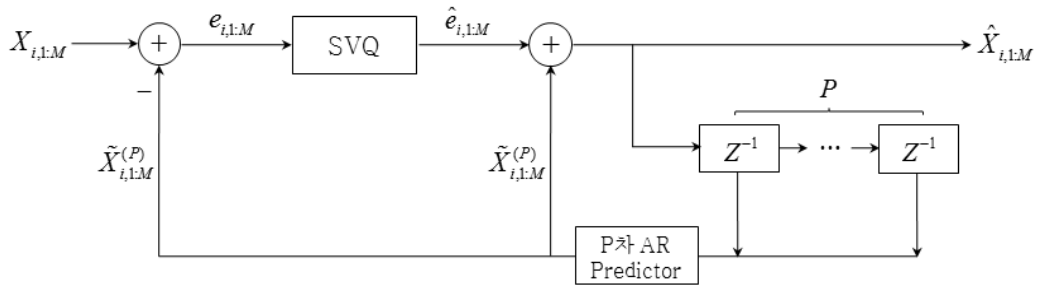


그림 1. MF-AR-SVQ 구조

Fig. 1. Blockdiagram of MF-AR-SVQ.

covariance 방법에 비해서 성능이 떨어지지만 stable 한 필터 계수를 찾을 수 있다는 장점이 있다.

그림 1은 P 차 AR 계수를 이용하여 LSF 데이터를 코딩하는 MF-AR-SVQ의 전체적인 블록도를 나타내고 있다. 전체 order가 M 인 LSF 데이터의 i 번째 frame- m 번째 order LSF 데이터는 과거 P 개 frame의 LSF 데이터들을 이용해서 다음과 같이 예측할 수 있다.

$$\tilde{X}_{i,m} = - \sum_{j=1}^P \hat{X}_{i-j,m} \cdot a_{j,m}, \quad 1 \leq m \leq M \quad (9)$$

Decoder 단에서는 과거 frame의 original LSF 데이터들을 정확히 예측할 수 없으므로, (9) 식에서는 (2) 식과 달리 quantized LSF 데이터인 $\{\hat{X}_{i-1,m}, \dots, \hat{X}_{i-P,m}\}$ 을 사용하였다. P 값을 1로 한 Single Frame AR-SVQ (SF-AR-SVQ)의 경우에는 PSVQ와 같이 과거값 하나만을 보게 된다. 따라서, SF-AR-SVQ와 PSVQ의 성능 차이를 비교하면, AR 계수를 사용한 가중치가 과거 frame에 어떤 영향을 미치는지 확인할 수 있다. 또한, AR-model의 차수 P 를 높이면, 과거 값을 그만큼 많이 보기 때문에 에러분산 값이 감소한다.

또한, (9) 식에서 $\{a_{j,m}\}_{1 \leq j \leq P, 1 \leq m \leq M}$ 은 m 번째 order LSF 데이터에 대한 j 번째 AR 계수를 의미한다. 이 값은 학습 데이터를 통해 구해지며 실제 코더에 적용 시에는 고정된 값이 사용되게 된다. 만약 학습 시 미리 구한 AR 계수를 고정해서 사용하는 것이 아니고 frame마다 적응적으로 구하고자 한다면, AR 계수를 전송하기 위한 별도의 비트가 필요할 뿐만 아니라 AR 계수의 추정을 위한 추가적인 계산량이 필요하다. 별도의 비트 전송 없이 적응적인 AR 계수를

decoder 단에 전달하기 위해서는 과거 frame의 양자화된 LSF 데이터들을 이용한 backward-adaptive AR 계수 추정 방법이 사용가능하나, 이 또한 추가적인 계산량이 매 frame 요구되므로 본 논문에서는 별도로 구현하지 않았다.

본 논문에서 제안하는 MF-AR-SVQ 방법은 학습 시 AR 계수를 따로 추출하여 실제 코더에서는 고정된 값을 이용하기 때문에 과거값을 하나 이상 봤을 때에 적용되는 가중치에 대한 계산량은 큰 차이가 없다. 즉, (9) 식을 통해 예측된 LSF 데이터를 계산하는 과정은 코드북 탐색 과정에 비해서 추가되는 계산량이 미미하므로 MF-AR-SVQ와 PSVQ의 계산량은 유사하다고 할 수 있다. 또한, 동일한 비트를 할당했을 경우에 MF-AR-SVQ와 PSVQ에 필요한 메모리 요구량은 동일하므로, 추가적인 메모리 요구량도 없다고 할 수 있다.

P 차 AR 계수는 (4) 식에서 에러분산을 최소화하여 계산되기 때문에 AR 차수 P 를 증가시키면 에러분산 값이 감소하게 된다. 결국 잔차 $\{e_{i,m}^{(P)}\}_{1 \leq m \leq M}$ 의 양자화 효율은 AR 차수 P 를 증가시켜 높일 수 있다.

III. 실험 및 토의

제안된 방식을 평가하기 위해서 TIMIT 데이터베이스를 8 kHz 샘플링 주파수로 다운 샘플링하여 사용하였다. 우선, AR 계수의 추출과 코드북 학습을 위해서 TIMIT 데이터베이스에서 10차 LSF 데이터를 1,200,000개 추출하였다. 10차 LSF 벡터는 3, 3, 4 차원으로 나누어서 독립적으로 양자화하였다. 성능평가에 이용된 LSF 데이터는 학습 시 사용한 데이터와서

로 다른 600,000개의 LSF 데이터를 TIMIT 데이터를 통해 추출 후 이용하였다.

양자화기의 성능은 다음 식과 같이 Spectral Distortion (SD)을 사용하여 비교 및 평가하였다. M 차 LSF 벡터 a 를 \hat{a} 로 양자화 함으로써 측정하는 SD는

$$SD(a, \hat{a}) = \frac{\beta}{2\pi} \int_{-\pi}^{\pi} (\ln(|A(w)|^2) - \ln(|\hat{A}(w)|^2))^2 dw \quad (10)$$

이다. 이 식에서 $\beta = (10/\ln(10))^2$ 이고, $|A(w)|^2$ 과 $|\hat{A}(w)|^2$ 는 original LSF 데이터와 quantized LSF 데이터를 통해 구해진 파워 스펙트럴 envelope를 의미한다. 실험에 쓰인 SD 계산의 주파수의 범위는 50 ~ 4000 Hz로 설정하였다.

본 논문에서는 10차 LSF 벡터 1,200,000개를 학습 데이터로 사용하여 P 차 AR 계수를 각 차원에 대해서 독립적으로 추정하였다. AR 계수는 성능평가 시 고정되어 사용되므로 별도의 비트를 전송할 필요가 없다. 10차 LSF 벡터의 각 차원별 데이터에 대해서 rectangular window를 사용하여 (4)의 에러분산을 최소화하는 P 차 AR 계수를 독립적으로 추정하였으며, 따라서 총 $10P$ 개의 AR 계수를 구할 수 있었다.

그림 2는 PSVQ, SF-AR-SVQ ($P=1$), MF-AR-SVQ ($P=5$), MF-AR-SVQ ($P=10$) 알고리즘에 대해서 27, 28, 29, 30 비트를 할당 하면서 평균 SD 성능을 평가한 결과이다.

3, 3, 4 차원으로 나눈 부벡터 별 양자화 비트는 27 비트의 경우 9, 9, 9 비트, 28 비트의 경우 9, 9, 10 비트, 29 비트의 경우 9, 10, 10비트, 30비트의 경우 10, 10, 10 비트가 각각 할당 되었다. 모든 비트 할당에 대해서 MF-AR-SVQ ($P=10$)가 가장 우수한 평균 SD 성능을 보였고, SF-AR-SVQ도 PSVQ 보다는 우수한 성능을 보였다.

AR 계수의 차수 P 에 따라서 MF-AR-SVQ의 평균 SD 성능이 우수해지는 이유는 차수가 높아질수록 더 많은 과거 frame을 참조하여 현재 frame을 예측하므로, 식 (4)에서 구한 에러분산 값이 줄어들기 때문이다. 그림 3에서는 참조하는 과거 frame의 갯수 P 를 변화시키면서, 즉 AR 계수의 차수를 변화시키면서 MF-AR-SVQ의 에러 분산값을 관찰해 보았다. 이전

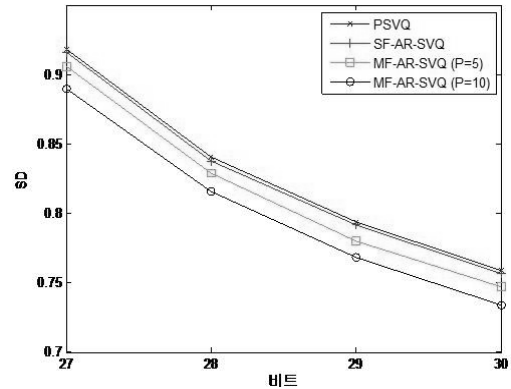


그림 2. PSVQ(x), SF-AR-SVQ(+), MF-AR-SVQ ($P=5$) (□), MF-AR-SVQ ($P=10$) (o)의 평균 SD 성능
Fig. 2. Average SD performance of PSVQ(x), SF-AR-SVQ(+), MF-AR-SVQ ($P=5$) (□) and MF-AR-SVQ ($P=10$) (o).

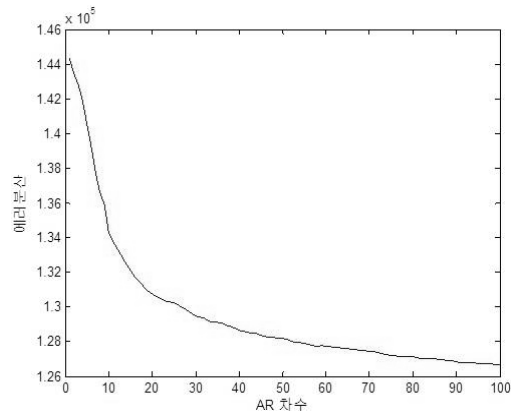


그림 3. AR 계수의 차수 변화에 따른 ($P=1-100$) MF-AR-SVQ의 에러 분산값
Fig. 3. Error variance of MF-AR-SVQ according to the order of AR coefficients ($P=1-100$).

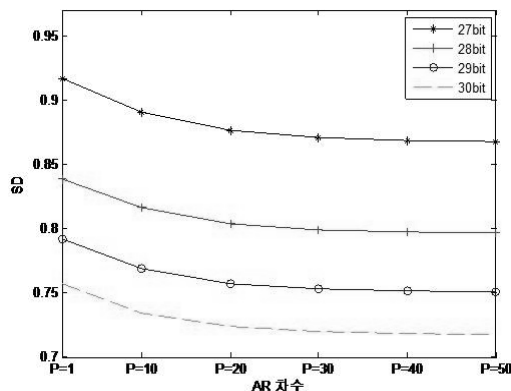


그림 4. AR 계수의 차수 변화에 따른 ($P=1-50$) MF-AR-SVQ의 평균 SD 성능
Fig. 4. Average SD performance of MF-AR-SVQ according to the order of AR coefficients ($P=1-50$).

표 1. SVQ, PSVQ, SF-AR-SVQ, MF-AR-SVQ의 평균 SD (dB)와 SD outlier percentage (%) 성능
Table 1. Average SD (dB) and outlier percentage (%) performance of SVQ, PSVQ, SF-AR-SVQ, MF-AR-SVQ.

LPC / 비트		27	28	29	30	
SVQ	Avg.SD	1.188	1.113	1.078	1.048	
	2-4 dB	4.391	3.391	2.975	2.028	
	4 dB	0.005	0.002	0.000	0.000	
PSVQ	Avg.SD	0.918	0.841	0.794	0.759	
	2-4 dB	2.877	2.085	1.697	1.356	
	4 dB	0.026	0.011	0.015	0.010	
SF-AR-SVQ	Avg.SD	0.916	0.838	0.792	0.757	
	2-4 dB	2.766	1.996	1.563	1.305	
	4 dB	0.021	0.010	0.011	0.008	
MF-AR-SVQ	P=2	Avg.SD	0.917	0.838	0.790	0.756
		2-4 dB	2.803	2.120	1.669	1.393
		4 dB	0.026	0.013	0.007	0.005
	P=3	Avg.SD	0.911	0.835	0.787	0.752
		2-4 dB	2.709	1.961	1.594	1.304
		4 dB	0.023	0.011	0.005	0.003
	P=4	Avg.SD	0.909	0.832	0.784	0.751
		2-4 dB	2.656	1.868	1.464	1.212
		4 dB	0.021	0.010	0.010	0.005
	P=5	Avg.SD	0.904	0.827	0.779	0.746
		2-4 dB	2.547	1.829	1.420	1.199
		4 dB	0.020	0.010	0.007	0.005
	P=6	Avg.SD	0.901	0.825	0.779	0.746
		2-4 dB	2.448	1.756	1.441	1.188
		4 dB	0.018	0.011	0.005	0.003
	P=7	Avg.SD	0.899	0.823	0.776	0.742
		2-4 dB	2.358	1.751	1.393	1.134
		4 dB	0.021	0.008	0.008	0.005
	P=8	Avg.SD	0.893	0.819	0.772	0.738
		2-4 dB	2.342	1.700	1.364	1.131
		4 dB	0.018	0.008	0.005	0.005
	P=9	Avg.SD	0.892	0.817	0.770	0.736
		2-4 dB	2.280	1.674	1.278	1.079
		4 dB	0.013	0.008	0.007	0.003
	P=10	Avg.SD	0.890	0.816	0.768	0.734
		2-4 dB	2.262	1.643	1.318	1.101
		4 dB	0.016	0.010	0.007	0.008
	P=20	Avg.SD	0.876	0.804	0.757	0.724
		2-4 dB	2.009	1.496	1.152	0.932
		4 dB	0.023	0.010	0.003	0.003
P=30	Avg.SD	0.871	0.799	0.753	0.720	
	2-4 dB	1.997	1.354	1.012	0.829	
	4 dB	0.010	0.007	0.003	0.003	
P=40	Avg.SD	0.869	0.797	0.752	0.718	
	2-4 dB	1.919	1.387	1.010	0.842	
	4 dB	0.011	0.003	0.007	0.003	
P=50	Avg.SD	0.868	0.797	0.751	0.718	
	2-4 dB	1.819	1.310	0.997	0.840	
	4 dB	0.018	0.010	0.008	0.002	

frame과의 상관관계만 고려한 $P=1$ 인 경우보다 P 값이 증가할수록 에러분산 값이 지속적으로 줄어들고 있음을 볼 수 있었다.

AR 계수의 차수 P 를 1부터 50까지 변화시키면서 측정된 MF-AR-SVQ의 평균 SD 성능은 그림 4에 나타나 있다. Frame당 비트할당은 27비트부터 30비트까지 변화시키면서 성능을 관찰하였다. MF-AR-SVQ의 평균 SD 성능은 AR 계수의 차수가 증가함에 따라서 지속적으로 줄어들고 있으며, 40차 이상이 되면 일정한 값으로 수렴함을 알 수 있다.

표 1은 SVQ, PSVQ, SF-AR-SVQ, MF-AR-SVQ의 성능을 다양한 비트율에 대해서 측정된 결과이다. 과거 데이터와의 inter-frame correlation은 이용하지 않고 intra-frame correlation 만을 이용하는 SVQ에 비해서 직전 frame 데이터와의 차이를 코딩하는 PSVQ의 평균 SD 성능이 우수함을 볼 수 있다. 하지만, 2-4 dB와 4 dB 이상의 outlier percentage는 평균 SD 성능 향상에 비해서 떨어짐을 볼 수 있다. PSVQ와 비교해 볼 때, 1차 AR 계수를 사용하는 SF-AR-SVQ의 성능이 약간 증가하는 것을 볼 수 있다. AR 계수의 차수를 50차까지 증가시키면서 구현한 MF-AR-SVQ의 성능은 평균 SD 뿐만 아니라 2-4 dB와 4 dB 이상의 outlier percentage 관점에서도 성능 증가가 지속적으로 이루어짐을 알 수 있다. 50차 MF-AR-SVQ는 PSVQ나 SF-AR-SVQ와 비교하면 약 1 비트 정도의 성능 향상을 얻을 수 있음을 알 수 있다. 또한, informal listening test 시 제안하는 50차 MF-AR-SVQ 방법이 기존 PSVQ 방법 보다 우수한 성능을 보임을 확인 하였다.

IV. 결 론

효율적인 음성 신호의 압축을 위해 사람의 vocal tract 필터는 LSF 계수로 모델링 하여 양자화 된다. 음성을 녹음하여 저장하는 장치의 경우에는 채널 에러가 거의 발생하지 않으므로 LSF 계수의 intra-frame correlation과 inter-frame correlation을 모두 이용하는 PSVQ가 최적의 성능을 나타낸다. 기존의 PSVQ는 현재 frame 데이터와 직전 frame 데이터 간의 차이를 코딩하는 방식인 반면에, 본 논문에서 제안하는 MF-AR-SVQ 방식은 P 차 AR model을 이용하여 과거 P 개

frame들과 현재 frame 데이터의 상관 관계를 고려함으로써 SD 관점에서 1 비트의 성능 향상을 얻을 수 있었다. 또한, MF-AR-SVQ 방식은 학습 시에 구한 AR 계수를 고정하여 사용함으로써, PSVQ 방식과 비슷하게 계산량과 메모리 요구량 관점에서도 유사한 성능을 보인다. 향후에는 별도의 비트 전송 없이 적응적인 AR 계수를 디코더 단계 전달하기 위해서 과거 frame의 양자화된 LSF 데이터들을 이용한 backward-adaptive AR 계수 추정 방법을 구현해 보고자 한다.

감사의 글

이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임 (No. 2011-0002502).

참고문헌

1. F. Itakura, "Line Spectrum Representation of Linear Predictive Coefficients of Speech Signal," *J. Acoust. Soc. Amer.*, vol. 57, suppl. 1, pp. S35(A), 1975.
2. 김해진, 강상원, "효율적인 LSF 양자화기를 이용한 QCELP 성능개선," *한국음향학회지*, 16권, 1호, 10-15 쪽, 1997.
3. K. K. Paliwal and B. S. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame," *IEEE Trans. Speech and Audio Proc.*, vol. 1, no. 1, pp. 3-14, 1993.
4. F. Nordin and T. Eriksson, "On split quantization of LSF parameters," *IEEE Int. Conf. Acoust. Speech and Signal Proc.*, vol. 1, pp. I-157-60, 2004.
5. S. So and K. K. Paliwal, "Switched split vector quantization of line spectral frequencies for wide-band speech coding," in *Proc. European Conf. Speech Commun. Tech (INTERSPEECH -2005)*, pp. 2705-2708, 2005.
6. S. So and K. K. Paliwal, "Efficient product code vector quantization using the switched split vector quantizer," *Digital Signal Proc.*, vol. 17, no. 1, pp. 138-171, 2007.
7. W. P. LeBlanc, B. Bhattacharya and S. A. Mahmoud, "Efficient Search and Design Procedures for Robust Multi-Stage VQ of LPC Parameters for 4 kb/s Speech Coding" *IEEE Trans. Speech Audio Proc.*, vol. 1, no. 4, pp. 373-385, 1993.
8. T. Eriksson, J. Linden and Jan Skoglund, "Interframe

- LSF Quantization for Noisy Channels," *IEEE Trans. Speech Audio Proc.*, vol. 7, no. 5, pp. 495-509, 1999.
9. S. Chatterjee and T.V. Sreenivas, "Predicting VQ Performance Bound for LSF Coding," *IEEE Signal Proc. Letter*, vol. 15, pp. 166-169, 2008.
10. M. Sabin and R. Gray, "Global convergence and empirical consistency of the generalized Lloyd algorithm," *IEEE Trans. Information Theory*, vol. 32, no. 2, pp. 148-155, 1986.
11. Y. Linde, A. Buzo and R. Gray, "An Algorithm for Vector Quantization Design," *Commun., IEEE Trans.*, vol. 28, no. 1, pp. 84-95, 1980.
12. W. B. Kleijn, *A Basis for Source Coding*, Course notes, KTH, Stockholm, 2008.
13. R. Salami, C. Laflamme, J.-P. Adoul and D. Massalux, "A Toll Quality 8 Kb/s Speech Codec for the Personal Communications System (PCS)," *IEEE Trans. Vehicular tech.*, vol. 43, no. 3, part: 1-2, pp. 808-816, Aug. 1994.

저자 약력

▶ 정 원 진 (Wonjin Jung)



2005년 3월 ~ 2011년 2월: 세종대학교
정보통신공학과, 공학사
2011년 3월 ~ 현재: 세종대학교 정보
통신공학과, 석사 과정
(관심분야) 음성코딩, 정보이론

▶ 김 무 영 (Moo Young Kim)



1989년 3월 ~ 1993년 2월: 연세대학교
전자공학과, 학사
1993년 3월 ~ 1995년 2월: 연세대학교
전자공학과, 석사
1995년 2월 ~ 2000년: 삼성종합기술원
전문연구원
2001년 1월 ~ 2004년 11월: Royal Institute
of Technology (KTH, 스웨덴) Dept.
Signals, Sensors, Systems, 박사
2004년 ~ 2005년 2월: Royal Institute
of Technology (KTH, 스웨덴) Dept.
Signals, Sensors, Systems, PostDoc
2005년 2월 ~ 2006년 8월: Ericsson
Research (스웨덴), Senior Research
Engineer
2006년 8월 ~ 현재: 세종대학교 정보통신
공학과, 부교수
(관심분야) 음성/오디오 신호처리 및 코
딩, 패턴인식, 정보이론.