디지털 이동통신용 RPE-LTP 음성부호화기

김선영*김진업*정종태*김영식**

목차

I. 서론
II. RPE-LTP 음성부호화기
III. 결과 및 검토

〈요약〉

세계적인 추세에 근거하여, 디지털 이동통신 음성음성부호화 방식 표준안 선정을 위해 평가 대상 방식으로 DSBC(Dynamic bit allocation Sub-Band Coding), RPE-LTP(Regular Pulse Excited Long Term Prediction), CELP(Code Excited Linear Prediction) 등을 선정한 바 있다.

본 논문에서는 이들 방식중 13 kbps RPE-LTP의 실현 및 성능평가에 관하여 다루었다. 먼저 음질에 중요한 영향을 미치는 분석/합성부호화에 근거한 파라미터 양자화 방법 그리고 채널 코딩과의 연계를 위한 비트 중요도 해석 등을 언급하였다. 끝으로 시뮬레이션 결과를 나타내었다.

I. 서론

최근 이동통신 서비스의 수요 증대로 인하여 이동통신의 디지털화가 선결되어야 할 중요
문제로 대두되고 있다. 이 경우 한정된 대역폭으로 많은 가입자를 수용하기 위해서는 가능한 늦은 전송속도로 음성을 부호화하여야 한다. 따라서 지금까지의 디지털 이동통신용 음성부호화기는 음질 및 주파수 효율 등을 고려하여 8-16kbps의 전송속도로 연구가 진행되고 있다. (표 1)은 각국에서 표준화 후보로 제시된 디지털 이동통신음성부호화기를 나타낸다.

<table>
<thead>
<tr>
<th>Type</th>
<th>Source Bit Rate</th>
<th>Gross Bit Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ericsson</td>
<td>13.0Kbps</td>
<td>19.5Kbps</td>
</tr>
<tr>
<td></td>
<td>8.7Kbps</td>
<td>13.0Kbps</td>
</tr>
<tr>
<td>AT &amp; T</td>
<td>12.0Kbps</td>
<td>16.0Kbps</td>
</tr>
<tr>
<td></td>
<td>8.0Kbps</td>
<td>16.0Kbps</td>
</tr>
<tr>
<td>Motorola</td>
<td>Celp</td>
<td>6.6Kbps</td>
</tr>
<tr>
<td></td>
<td>9.0Kbps</td>
<td></td>
</tr>
<tr>
<td>NEC</td>
<td>MPE-LPC</td>
<td>8.0Kbps</td>
</tr>
<tr>
<td></td>
<td>9.7Kbps</td>
<td></td>
</tr>
<tr>
<td>NTT</td>
<td>TC-WVQ</td>
<td>6.7Kbps</td>
</tr>
<tr>
<td></td>
<td>8.0Kbps</td>
<td></td>
</tr>
<tr>
<td>Northern Telecom</td>
<td>MPE-LPC</td>
<td>16.0Kbps</td>
</tr>
<tr>
<td></td>
<td>8.0Kbps</td>
<td></td>
</tr>
<tr>
<td>CEPT/GSM</td>
<td>RPE-LTP</td>
<td>13.0Kbps</td>
</tr>
<tr>
<td></td>
<td>22.8Kbps</td>
<td></td>
</tr>
<tr>
<td>British Telecom</td>
<td>MPE-LTP</td>
<td>9.6Kbps</td>
</tr>
</tbody>
</table>

또한 디지털 이동통신용 음성 부호화기는 다음의 조건을 고려하여 선택하여야 한다.[1, 2]

첫째, 좋은 음질을(900MHz에널로그에 비해) 유지하면서 전송속도가 낮을 것.
둘째, 채널 특성 변화에 강한 것을(랜덤오류 10^-2까지)

세계, 다양한 범위의 화자와 저장없는 통화가 가능한 점.
네째, 설계 구현시 복잡도가 낮을 것.
다섯째, 부호기로 인한 지연이 적을 것.

II. RPE-LTP 음성부호화기

1. 구성도 및 원리

RPE-LTP 음성부호화기의 원리적인 블록도는 〈그림 1〉과 같다.

음성신호는 그 특성상 두가지 상관관계를 지니고 있는데 하나는 인접 샘플간의 상관관계(Short Term Correlation)이고 또 하나는 피치간의 상관관계(Long Term Correlation)이다.

RPE-LTP 방식의 원리는 〈그림 2〉와 같이 LPC 분석 부분에서 음성 신호의 인접 샘플간의 상관관계를 제거하고 LTP 분석 부분에서 피치 샘플간의 관계를 다시 제거하여 백색 잡음에 가까운 잔여(residual)신호를 얻어 이들 펄스 중 원래 음성과의 차를 최소로 하는 일정 간격의 펄스열(Regular Pulse Sequences)을

43

(a) Encoder

(b) Decoder

〈그림 1〉 RPE-LTP 음성 부호화기 블록도

〈그림 2〉 RPE-LTP 음성 부호화기 송신부의 각 단계별 파형
송신하여 (그림 3)과 같이 수신측에서는 이의
역과정을 거쳐 원래 음성 신호를 재생하는 방

법이다.

이때에 인코더 부분에서는 다음과 같은 정보를
전송하여야 수신측에서 원래 신호함성이 가능
하다.
○ 음성 신호의 스펙트럼 정보
○ LTP 정보
○ 구동 신호(Excitation Signal)

2. 파라미터 양자화

가. 분석/합성 부호화

잔여 신호를 부호화하는 데 있어 만족할 만한
음질의 음을 재생하기 위해서는 기존의
RELP(Residual Excited Linear Prediction) 방
법으로는 10kbps 이하의 전송속도에서는 많은
양자화 오차가 발생하여 음질이 크게 나빠진
다.[1] 그러므로 재생된 음의 음질을 어느 정
도 유지하면서 낮은 전송 속도로 음성을 부호
화하기 위해서는 분석/합성(Analysis-by-Syn-
thesis) 부호화 기법에 의한 잔여 신호 대치 알
고리즘(Residual Substitution Algorithm)을 이
용하여 잔여신호를 다른형태로 부호화 하여야
한다. (그림 4)의 분석/합성 부호화 기법은
입력 신호와 합성된 신호의 차가 최소가 되도
록 부호화기 계수를 구하는 방법으로서 입력
음성에 잡음이 많이 섞여 있는 경우에도 음질
저하가 적게 생긴다는 장점이 있으며 16kbps
이하의 전송속도에서도 만족할만한 음질의 음
을 재생할 수 있다.
이같은 분석/합성 방식의 원리를 이용하여 송신측에서도 수신측에 전달된 파라미터와 똑 같은 즉 분석/합성 절차에 의해 양자화된 파라미터를 이용하게 된다. 따라서 (그림 1)의 송수신 구성도를 자세히 나타내면 (그림 5)와 같다. (그림 1)의 경우에 비해 각 파라미터의 분석/합성 절차가 참가된 것이다.

(a) RPE-LTP 부호화
(b) RPE-LTP 복호기

(그림 5) RPE-LTP 음성 부호화기

다음에 이들 각 부분의 원리를 상세히 다룬다.

나. 인코더 부분
1) 전처리(Pre-processing)
전처리는 Offset 보상 및 Preemphasis의 두 부분으로 구성된다. 먼저 입력 음성의 적층성 부분을 제거하기 위해 (1)식과 같은 Notch 필터를 이용한다.

\[ S_{\text{ad}}(k) = S_i(k) - S_i(k-1) - 0.999 \ S_{\text{ad}}(k-1) \quad \cdots \]

여기서 \( S_{\text{ad}} \)는 Offset Free 신호를 \( S_i \)은 입력 음성을 나타낸다. (그림 6)은 이같은 필터의 응답 특성을 나타낸다. DC부분의 Offset이 제거될 수 있다.
이같이 Offset을 제거한 다음 음성 신호 그 자체가 갖는 Spectral Rolloff 특성으로 인하여
(그림 7), 고주파부분이 잘 모델링되지 않는 단점을 제거하기 위하여 다음과 같이 1차 FIR
Preemphasis 필터를 이용하여 고주파 부분을 미리 강조(Enhance)하여 처리한 후 디코딩 단
단에서 역으로 Deemphasis을 한다.

\[ S'(k) = S_d(k) - 0.86 \cdot S_d(k-1) \]  \hspace{1cm} (2)

(그림 8)은 이같은 Preemphasis 필터의 응답특성을 나타낸다. Deemphasis의 응답특성은 이와
의 역이다.

2) LPC 분석 및 Short-Term 필터링

분석/합성 부호화 방식에서는 음성 생성 모델에서와 같이 성도를 시변 디지털 필터로 모
델링하여 필터 계수를 구하는 과정이 필요한데, 일반적으로 음성생성 모델은 All Pole 시변
디지털 필터로 나타내는데, 필터 계수들은 〈그
림 8)과 같은 선행 예측법에 의하여 구한다. 이때에 입력 신호로부터 필터 계수들을 계산하는 과정은 LPC분석이라 한다.

선행 예측법이란 처리되어지 않은 데이터 블록안에서 시간 \( n \)에서의 표본값 \( S(n) \)을 \( p \)개의 과거 표본값 \( S(n-k) \) (\( k=1, 2, \ldots, P \))로 예측하여 식 (3)에서 예측 오차 에너지 \( E \)가 최소가 되도록 최적의 선행 예측 계수로 구하는 방법을 말한다.

\[
E = \sum_{n=1}^{N} [s(n) - \sum_{k=1}^{P} a_k s(n-k)]^2 \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (3)
\]

여기서 \( S(n) \)은 시간 \( n \)에서의 음성 신호값, \( a_k \)는 선행 예측 계수, \( p \)는 선행 예측 차수, \( N \)은 한 프레임내에서 처리되어지는 음성 데이터 수이 다. 음성 신호는 시간에 따라 특성이 천천히 변화하므로 10~20ms마다 선행예측 계수들을 다시 구하여 전송하여야 한다. 그러나, 선행 예측 계수는 계수값의 변동범위가 크므로 PARCOR(PARTial CORrelation)계수를 부호화하는 것이 유리하다. PARCOR계수의 절대값은 항상 1보다 작은 값을 가지므로 선행 예측 계수 보다 적은 비트로 부호화할 수 있고, 비트제한에 의해 야기될 수 있는 합성 필터의 불안정성을 제거할 수 있다.

선행 예측 필터의 차수 즉 PARCOR 계수의 갯수에 따라 음질의 변화가 나타나는데 (그림 10)과 같다. (그림 10)로부터 \( P=8 \)이상인 경우에는 음질의 변화가 거의 없음을 알 수 있다.
입력 신호로부터 PARCOR계수를 구하려면 먼저 자기상관 함수 \( R(i) \) \((i=0, 1, \ldots, P)\)를 구하여야 하며 이 방법에는 일반적으로 Auto-correlation 및 Covariance 방법이 있다. 후자의 경우는 안정도가 문제가 되므로 전자의 경우를 택하였다. 또한 Autocorrelation 방법에 의해 PARCOR계수를 구하는 알고리즘으로는 (그림 11)의 훈련도와 같은 Schur-Recursion 방법을 사용하였다.

(그림 9) 선형 예측 부분의 구성

(그림 10) 예측 필터의 차수에 따른 음질의 변화
\[ n = 1 \]

\[ ACF = 0 \]

\[ K(9-i) = ACF(i) : i = 1, 7 \]
\[ P(j) = ACF(j) : j = 0, 8 \]

\[ P(o) < |P(1)| \]

\[ r(n) = |P(1)| / P(0) \]

\[ P(1) > 0 \]

\[ r(n) = r(-n) \]

\[ n = 8 \]

\[ P(0) = P(0) + P(1) \times r(n) \]

\[ m = 1 \]

\[ P(m) = P(1+m) + r(n) \times K(9-m) \]
\[ K(9-m) = K(9-m) + r(n) \times P(1+m) \]

\[ n = n + 1 \]
\[ m = 8 - n \]
\[ m = m + 1 \]

Transformation
\[ r \rightarrow LAR \]

End

(그림 11) Schur-Recursion 방법에 의한 LPC 분석
또한, PARCOR 계수를 직접 부호화하는 것보다 LAR(Logic Area Ratio)로 변환한 후, 부호화하는 것이 양자화 에러에 대한 스펙트럼 민감도를 줄일 수 있다. PARCOR 계수의 LAR 변환은 다음식에 의한다.

\[
LAR(i) = \log \left( \frac{1+r(i)}{1-r(i)} \right)
\]  \hspace{1cm} \text{(4)}

Companding 특성을 고려하여 위의 식을 근사화하는 방법에는 선형 근사화 방법이 가장 많이 사용된다. 이 근사화 방법에 의해 5등분으로 근사화하면 LAR 계수들은 다음과 같다.

\[
LAR(i) = \begin{cases} 
0 & |r(i)| < 0.675 \\
\text{sign}(r(i)) \times [2 \cdot |r(i)| - 0.675], & 0.675 \leq |r(i)| < 0.95 \\
\text{sign}(r(i)) \times [8 \cdot |r(i)| - 0.375], & 0.95 \leq |r(i)| < 1.0 
\end{cases}
\]  \hspace{1cm} \text{(5)}

따라서 Log 연산 및 나눗셈을 곱셈 및 비교 연산으로 바꿀 수 있다. 역으로 \(r(i)\)의 변환은 다음식에 의한다.

\[
r(i) = \begin{cases} 
LAR & |LAR(i)| < 0.675 \\
\text{sign}[LAR(i)] \times [0.5 \times |LAR(i)| + 0.3375], & 0.675 \leq |LAR(i)| < 1.225 \\
\text{sign}[LAR(i)] \times [0.125 \times |LAR(i)| + 0.796875], & 1.225 \leq |LAR(i)| < 1.625 
\end{cases}
\]  \hspace{1cm} \text{(6)}

(그림 12)는 석에서 정의된 LAR 계수들의 확률밀도 함수를 나타낸다. PARCOR 계수와 비교해 볼 때 변동범위(Dynamic Range) 및 분포가 다르다. 이에 근거하여 LAR은 다음과 같이 양자화된다.
 그림 12  LAR 계수들의 확률밀도함수

(a) LAR(1)의 확률밀도함수 (b) LAR(2)의 확률밀도함수
(c) LAR(3)의 확률밀도함수 (d) LAR(4)의 확률밀도함수
(e) LAR(5)의 확률밀도함수 (f) LAR(6)의 확률밀도함수
(g) LAR(7)의 확률밀도함수 (h) LAR(8)의 확률밀도함수

\[ LAR_c(i) = \text{NINT}\{A(i) \cdot LAR(i) + B(i)\} \ldots (7) \]

여기서 \( \text{NINT}(z) = \text{INT}\{z + \text{sign}(z) \cdot 0.5\} \)이고 이때 계수는 \(<\text{표 2}>\)와 같다.
〈표 2〉 LAR(i)의 양자화

<table>
<thead>
<tr>
<th>LARi</th>
<th>A(i)</th>
<th>B(i)</th>
<th>min LARc(i)</th>
<th>max LARc(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>0</td>
<td>-32</td>
<td>31</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>0</td>
<td>-32</td>
<td>31</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>4</td>
<td>-16</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>-5</td>
<td>-16</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>13.637</td>
<td>0.184</td>
<td>-8</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>-3.5</td>
<td>-8</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>8.334</td>
<td>-0.666</td>
<td>-4</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>8.824</td>
<td>-2.235</td>
<td>-4</td>
<td>3</td>
</tr>
</tbody>
</table>

역으로 디코딩 과정은 다음에 의한다.

\[ LAR^c(i) = (LAR_c(i) - B(i)) / A(i) \]  \hspace{1cm} (8)

이때 필터 계수값의 감각스런 변화에 따른 프레임 경계 부근에서의 과도(Transient) 응답을 방지하기 위해 LAR 계수들은〈표 3〉과 같이 선형보간(Interpolation)한다. 이 방법은 프레임 데이터의 일부 점에서 처리하는 Overlap 방법과 같은 효과를 나타내며 동시에 계산량을 줄일 수 있는 방법이다.

〈표 3〉 LAR 파라미터의 인터플레이션

<table>
<thead>
<tr>
<th>k</th>
<th>LAR*(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0~12</td>
<td>0.75 + LAR*-i(i) + 0.25<em>LAR</em>(i)</td>
</tr>
<tr>
<td>13~26</td>
<td>0.5 + LAR*-i(i) + 0.25<em>LAR</em>-i(i)</td>
</tr>
<tr>
<td>27~39</td>
<td>0.25 + LAR*-i(i) + 0.75<em>LAR</em>-i(i)</td>
</tr>
<tr>
<td>40~159</td>
<td>LAR*-i(i)</td>
</tr>
</tbody>
</table>

3) LTP(Long Term Predictiton)

ST(Short Term)필터링 후의 잔여 신호에는 아직 더 제거해야 할 피치 샘플간의 상관관계가 존재하므로 다음과 같은 피치 예측기를 이용하여 이러한 상관관계를 제거할 수 있다.

\[ LTP(z) = 1 - \beta z^{-1} \]  \hspace{1cm} (9)

여기서 \( \beta \)는 LTP이득, \( \alpha \)는 입력 음성의 피치를 나타낸다. 피치 예측 방법은 Open Loop 및 Closed Loop방법으로 대분되며 유럽의 GSM에서는〈그림 13〉과 같은 Open Loop 방법을 권고하고 있다.

\[
\begin{align*}
r(n) + & = d(n) \\
\beta z^{-a} & \\
\end{align*}
\]

〈그림 13〉 개루프에 의한 피치예측 방법

Open Loop방법에서는 LTP파라미터는 LTP 후의 잔여 신호의 MSE(Mean Square Error)를 최소로 함으로써 결정된다. 즉 \( \alpha \)값의 변화에 따라 식 0의 \( E_a \)가 최소가 되는 \( \alpha \) 및 \( \beta \)를 선택하면 된다.

\[
E_a = \sum_{n=0}^{N-1} d(n) = \sum_{n=0}^{N-1} (r(n) - \beta r(n-a))^2 \hspace{1cm} (10)
\]

여기서 \( r(n) \)은 LPC잔여 신호로 다음과 같다.

\[
r(n) = s(n) - \sum_{k=1}^{P} a_n s(n-k) \hspace{1cm} (11)
\]

00식을 최소로 하려면 다음항을 최대로 하여 된다.

\[
C_a = \left[ \sum_{n=0}^{N-1} r(n) r(n-a) \right]^{-1} \sum_{n=0}^{N-1} [r(n-a)]^2 \hspace{1cm} (12)
\]

임의는 잔여신호 \( r(n) \)과 그의 이전 신호 \( r(n \)
이들 식 (14)와 같이 4개의 그룹으로 나누어
\[ x_m(i) = x(j^*40 + m + 3^*i) \]

여기서 필수위치 \( i = 0, \ldots, 12 \), 그리드 위치 \( m = 0, \ldots, 3 \) \( j \)는 부프레임 인덱스 0, \( \ldots \), 3이다.

다음과 같이 에너지가 가장 큰 일정간격의 필스 열(Regular Pulse Sequence)을 선택하여 전송한다.

\[ E_{\text{M-Max}} \sum_{i=1}^{13} x_m(i) \quad m = 0, 1, 2, 3 \]

여기서 최적 그리드 위치 \( M \)은 2비트로 부호화 한다. 이와같이 선택된 일정간격 필스열은 각 필스열을 블럭 최대값으로 나눈 다음 블럭 적응 PCM방식으로 양자화하여 전송하는데 이때 블럭 최대값 \( X_{\text{max}} \)는 (표 4)와 같이 6비트 Logarithmic 양자화로 양자화하여 전송한다.
다. 디코더 부분

Decoding은 다음과 같은 4부분으로 구성된다.

1) RPE Decoding
2) LTP
3) ST 합성 필터
4) Postprocessing(Deemphasis)

위의 과정은 Encoding 분석/합성 부분에서 연급하였으므로 생략한다. 이때 합성 필터는 다음과 같다.

### 표 4 최대값의 양자화

<table>
<thead>
<tr>
<th>xmax</th>
<th>x' max</th>
<th>xmaxc</th>
<th>xmax</th>
<th>x' max</th>
<th>xmaxc</th>
</tr>
</thead>
<tbody>
<tr>
<td>0~31</td>
<td>31</td>
<td>0</td>
<td>048~2303</td>
<td>2303</td>
<td>32</td>
</tr>
<tr>
<td>32~63</td>
<td>63</td>
<td>1</td>
<td>2304~2559</td>
<td>2559</td>
<td>33</td>
</tr>
<tr>
<td>64~95</td>
<td>95</td>
<td>2</td>
<td>2560~2815</td>
<td>2815</td>
<td>34</td>
</tr>
<tr>
<td>96~127</td>
<td>127</td>
<td>3</td>
<td>2816~3071</td>
<td>3071</td>
<td>35</td>
</tr>
<tr>
<td>128~159</td>
<td>159</td>
<td>4</td>
<td>3072~3327</td>
<td>3327</td>
<td>36</td>
</tr>
<tr>
<td>160~191</td>
<td>191</td>
<td>5</td>
<td>3328~3583</td>
<td>3583</td>
<td>37</td>
</tr>
<tr>
<td>192~223</td>
<td>223</td>
<td>6</td>
<td>3584~3839</td>
<td>3839</td>
<td>38</td>
</tr>
<tr>
<td>224~255</td>
<td>255</td>
<td>7</td>
<td>3840~4095</td>
<td>4095</td>
<td>39</td>
</tr>
<tr>
<td>256~287</td>
<td>287</td>
<td>8</td>
<td>4095~4607</td>
<td>4607</td>
<td>40</td>
</tr>
<tr>
<td>288~319</td>
<td>319</td>
<td>9</td>
<td>4608~5119</td>
<td>5119</td>
<td>41</td>
</tr>
<tr>
<td>320~351</td>
<td>351</td>
<td>10</td>
<td>5120~5631</td>
<td>5631</td>
<td>42</td>
</tr>
<tr>
<td>352~383</td>
<td>383</td>
<td>11</td>
<td>5632~6143</td>
<td>6143</td>
<td>43</td>
</tr>
<tr>
<td>384~415</td>
<td>415</td>
<td>12</td>
<td>6144~6655</td>
<td>6655</td>
<td>44</td>
</tr>
<tr>
<td>416~447</td>
<td>447</td>
<td>13</td>
<td>6656~7167</td>
<td>7167</td>
<td>45</td>
</tr>
<tr>
<td>448~479</td>
<td>479</td>
<td>14</td>
<td>7168~7679</td>
<td>7679</td>
<td>46</td>
</tr>
<tr>
<td>480~511</td>
<td>511</td>
<td>15</td>
<td>7680~8191</td>
<td>8191</td>
<td>47</td>
</tr>
<tr>
<td>512~575</td>
<td>575</td>
<td>16</td>
<td>8192~9215</td>
<td>9215</td>
<td>48</td>
</tr>
<tr>
<td>576~639</td>
<td>639</td>
<td>17</td>
<td>9216~10239</td>
<td>10239</td>
<td>49</td>
</tr>
<tr>
<td>640~703</td>
<td>703</td>
<td>18</td>
<td>10240~11263</td>
<td>11263</td>
<td>50</td>
</tr>
<tr>
<td>704~767</td>
<td>767</td>
<td>19</td>
<td>11264~12287</td>
<td>12287</td>
<td>51</td>
</tr>
<tr>
<td>768~831</td>
<td>831</td>
<td>20</td>
<td>12288~13311</td>
<td>13311</td>
<td>52</td>
</tr>
<tr>
<td>832~895</td>
<td>895</td>
<td>21</td>
<td>13312~14335</td>
<td>14335</td>
<td>53</td>
</tr>
<tr>
<td>896~959</td>
<td>959</td>
<td>22</td>
<td>14336~15359</td>
<td>15359</td>
<td>54</td>
</tr>
<tr>
<td>960~1023</td>
<td>1023</td>
<td>23</td>
<td>15360~16383</td>
<td>16383</td>
<td>55</td>
</tr>
<tr>
<td>1024~1151</td>
<td>1151</td>
<td>24</td>
<td>16384~18431</td>
<td>18431</td>
<td>56</td>
</tr>
<tr>
<td>1152~1297</td>
<td>1279</td>
<td>25</td>
<td>18432~10479</td>
<td>20479</td>
<td>57</td>
</tr>
<tr>
<td>1280~1407</td>
<td>1407</td>
<td>26</td>
<td>10480~11527</td>
<td>22527</td>
<td>58</td>
</tr>
<tr>
<td>1408~1535</td>
<td>1535</td>
<td>27</td>
<td>22528~24575</td>
<td>54575</td>
<td>59</td>
</tr>
<tr>
<td>1536~1663</td>
<td>1663</td>
<td>28</td>
<td>24576~26623</td>
<td>26623</td>
<td>60</td>
</tr>
<tr>
<td>1664~1719</td>
<td>1791</td>
<td>29</td>
<td>26624~28671</td>
<td>28671</td>
<td>61</td>
</tr>
<tr>
<td>1792~1919</td>
<td>1919</td>
<td>30</td>
<td>28672~30719</td>
<td>30719</td>
<td>62</td>
</tr>
<tr>
<td>1920~2047</td>
<td>2047</td>
<td>31</td>
<td>30720~32767</td>
<td>32767</td>
<td>63</td>
</tr>
</tbody>
</table>
3. 시뮬레이션

다음표는 RPE-LTP의 비트합당을 나타낸다. 따라서 전체전송 속도는 13kbps가 된다.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>RPE-LTP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame Size</td>
<td>20msec</td>
</tr>
<tr>
<td>Sub-Frame Size</td>
<td>5msec</td>
</tr>
<tr>
<td>Sampling Rate</td>
<td>8KHz</td>
</tr>
<tr>
<td>8 LAR'S</td>
<td>32bit</td>
</tr>
<tr>
<td>4 LTP Coeff.</td>
<td>8bit</td>
</tr>
<tr>
<td>4 LTP Delays</td>
<td>28bit</td>
</tr>
<tr>
<td>4 RPE Grid</td>
<td>8bit</td>
</tr>
<tr>
<td>4 Block Max.</td>
<td>24bit</td>
</tr>
<tr>
<td>52 RPE Samples</td>
<td>156bit</td>
</tr>
<tr>
<td>Gross Bit</td>
<td>260bit</td>
</tr>
<tr>
<td>Bit Rates</td>
<td>13.0Kbps</td>
</tr>
</tbody>
</table>

다음 (그림 15)는 13kbps RPE-LTP 방식의 Floating 및 Integer 시뮬레이션 결과를 나타낸다. Integer 시뮬레이션을 하는 이유는 고정소수점 연산 디지털 신호처리기를 이용하여 음성부호화기 알고리즘을 구현할 경우 유한한 레저스터 길이로 인하여 여러변의 연산을 할 경우에 발생되는 Truncation 또는 반올림 오차의 누적에 의한 오버플로우 및 언더플로우 등을 고려하여 신호처리기가 행하는 몇 가지 기본연산(예, 16비트 가감산, 32 비트 가감산, 16비트 승산 계산, 절대치 연산 등...)을 그대로 시뮬레이션에서 행함으로써 실제 H/W로 구현할 경우와 같은 출력이 얻어지므로 유리하다.

4. 비트 중요도 해석

추출된 음성 정보의 각 파라미터들은 2진 부호화하여 전송하여야 하는데 이때 전송 채널의 영향 특히 이동통신에서는 다중경로에 의한 헤이딩등의 영향으로 비트에러가 발생할 확률이 높다. 따라서 이들 파라미터들이 채널의 영향을 받았을 때 정보전달에 결정적인 영향을 미치게 된다. 이같은 에러 발병시의 영향 (Error Sensitivity)은 각 파라미터에 따라 그리고 그 파라미터의 개별 비트별로 다르게 나타난다. 따라서 중요한 영향을 미치는 정보비트들은 채널 코딩시 상대적으로 많은 보호비트(Protection Bit)로써 코딩하고 에러가 발생하여도 정보전달에는 그다지 영향을 미치지 않을 경우에는 그대로 전송하는 방법을 사용하면 효율적인 정보전송이 가능하다. 이같은 입
의 비트 에러 발생시의 영향에 대한 분석은 매 프레임마다 음성 정보 비트들의 에러 감도를 구하여 이의 평균치를 구하면 된다.

한 프레임 $T_b(20\text{ms})$의 원래 음성 $S(t)$를 $N$ ($260$)비트로 부호화 한 후, 이를 재구성한 음성을 $r(t)$라 하고, $N$비트들에 대해 $2^{N-1}$경우로 오류를 발생 시킨 후 재구성한 음성을 $r_e(t)$라 하자. 이때 $i$번째 프레임의 신호 및 잡음 전력은 다음과 같다.

$$ P^{(i)} = \frac{1}{T_b} \int \mathbf{P}_{r_e(t)} [s(t)]^2 dt \quad \text{(06)} $$

$$ n^{(i)} = \frac{1}{T_b} \int \mathbf{P}_{r_e(t)} [s(t) - r(t)]^2 dt $$

$$ + \sum_{e=1}^{P} \mathbf{P}_{r_e(t)} \mathbf{P}_{r_e(t)} [(s(t) - r_e(t))^2 + 2(s(t) - r_e(t))^2 $$

$$ (r(t) - r_e(t))] \quad \text{(07)} $$

채널 에러 확률이 $P_e$일 경우, $j$번째 프레임의 신호대 잡음비는 식 (18)과 같고 $J$개의 프레임에 대하여 평균한 신호대 잡음비는 식 (19)과 같이 표현된다.

$$ \text{SEGSNR}^{(j)} = 10\log_{10} \frac{P^{(j)}}{n^{(j)}} \quad \text{(18)} $$

$$ \text{SEGSNR} = \frac{1}{J} \sum_{j=1}^{J} 10\log_{10} \frac{P^{(j)}}{n^{(j)}} \quad \text{(19)} $$

그런데 식 (19)의 SEGSR는 $2^{N-1}$에러 패턴에 대한 에러를 각 비트에 에러 패턴 $N$의 경우로 줄여 해석할 때 음성 정보의 각 비트에 대한 에러 영향은 첫번째 비트부터 $N$번째 비트까지 한 비트씩 에러를 발생한 후, 그 비트에 대한 에러 영향을 구하면 된다. 즉, $i$번째 비트에 에차가 발생한 경우 잡음전력은 식 (20)과 같이 표현할 수 있다.

$$ r^{(i)} = \frac{1}{T_b} \int \mathbf{P}_{r_e(t)} [s(t) - r(t)]^2 dt $$

$$ + \frac{1}{T_b} \int \mathbf{P}_{r_e(t)} [(s(t) - r_e(t))^2 + 2(s(t) - r_e(t))] $$

$$ (r(t) - r_e(t))] \quad \text{(20)} $$

다음은 이상의 과정에 의해 $13\text{kbps}$ RPE-LTP방식에 관한 음성정보 비트의 에러 감도를 분석한 결과이다. 먼저 프레임 형태는 다음과 같다.

<table>
<thead>
<tr>
<th>36</th>
<th>92</th>
<th>260</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 6 5 5 4 4 3 3 7 2 2 6 3</td>
<td>Subframe 2</td>
<td>Subframe 4</td>
</tr>
<tr>
<td>LAR</td>
<td>LTP LTP Grid Block Rpe-Pulses</td>
<td></td>
</tr>
<tr>
<td>Lag Gain Position Max #1 ~ #13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36bit</td>
<td>56bit Sub frame 1</td>
<td></td>
</tr>
</tbody>
</table>

〈그림 16〉 RPE-LTP 음성 부호기의 프레임 형태

〈그림 17〉은 앞의 프레임에 근거한 RRE—LTP 음성 부호기의 비트 중요도를 분석한 결과이다. 이로부터 블록 최대치 및 LAR이 중요한 정보임을 알 수 있다. 따라서 채널코딩과 연계할 경우 이들 정보에 대해 보호를 더 많이 해줘야 한다.

〈그림 17〉 13 kbps RPE—LTP음성 부호기의
비트 중요도
III. 결과 및 검토

디지털 이동 통신용 음성 부호화 방식 선정을 목적으로 13kbps RPE-LTP방식의 이론 및 시뮬레이션 결과를 다루었다. H/W 구현에 관한 내용은 연속된 논문으로 별도로 제출할 것이다. 또한 이동통신 단말기의 전력소실을 방지하기 위한 방안으로 DTX(Discontinuous Transmission) 기능 부가에 관한 연구도 병행할 예정이다. 차후 다른 부호화 방식 DSBC, CELP(특히 북미 표준방식인 VSELP)등과의 음질 비교도 할 예정이다. 또한 앞으로의 Half-Rate 부호화를 위해 6.5kbps 이하의 방식도 연구되어야 할 것이다.

참고 문헌


