

2021. 1. 27 Notation tips

1) time domain indexing

analog signal: $x(t)$, $-\infty < t < \infty$ $t \in \mathbb{R}$

digitized (sampled) notation: $x[n]$ $n \in \mathbb{Z}$

Sampling theorem에 따라 엄밀하게 notation을 적으면

$x[n] = x(t_n)$, $t_n = T_s \cdot n$, T_s 는 sampling period이다.

그렇지만 speech signal processing에서는 대부분 digitized signal에서 시작하기 때문에 $x[n]$ notation도 많이 쓴다.

$t = 0, 1, 2, \dots, T$, T 는 number of total samples.

2) z-domain, frequency domain indexing

$X(z)$: z-transform of $x[n]$, z is any complex number

Fourier transform은 $z = e^{j\omega}$ 로 정의되는 special case

$$e^{j\omega} = \cos \omega + j \sin \omega, \quad |e^{j\omega}| = 1$$

즉 2차원 complex domain에서 unit circle

따라서 $x[n]$ 의 Fourier transform은 $X(e^{j\omega})$ 로 적어야

정확하지만 그냥 $X(\omega)$ 로 쓰고 암묵적으로 Fourier domain으로 통용된다

* 가끔 ω (omega) 대신 f 를 쓰기도 함

$$X(\omega) \equiv X(e^{j\omega}), \quad X(f) \equiv X(e^{j2\pi f})$$

$0 \sim 2\pi$ periodic 함 (반복)

3) discrete Fourier transform

$$\omega_k = 2\pi \frac{k}{N}, \quad N: \text{DFT 개수}$$

$X(k)$ 로 가끔 쓰기도 함.

1) LP (linear predictive) coefficients 의 문제점 (coding 관점에서)

$$A(z) \cdot X(z) = E(z) \Leftrightarrow X(z) = \frac{E(z)}{A(z)}$$

① 즉, 원래 speech를 복원하기 위해서는 IIR filter를
곱해야 (convolution in time domain) 하는데,
양측으로 전송하는 과정에서 오류가 발생하면 백산하거나,
Speech가 제대로 생성되지 않을 수 있다.

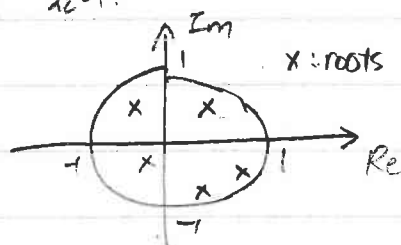
* IIR filter의 수렴조건

$A(z) = 0$ 을 만족하는

roots z^* 들이 모두

unit circle 안에 있어야

등비수열의 수렴조건을 만족한다



* 이론적으로 LP polynomial은 이를 만족한다.

$$\forall z^* \text{ for } A(z^*) = 0 \quad |z^*| < 1, \quad z^* = re^{j\theta} \Rightarrow r < 1$$

* 일반적으로 IIR filter를 설계할때 인위적으로 roots가
unit circle 안에 들어오도록 강제로 $A(z)$ 를 변환하기도 한다.
즉, $A(z) = (z - z_1^*)(z - z_2^*) \dots (z - z_M^*)$ 으로 표현하는
후에 $|z_m^*| > 1$ 의 magnitude r 을 조정할 수도 있지만
일반적이지는 않다.

$$② A(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M} \text{ 에서}$$

a_m 은 임의의 실수이다. 즉 이론적으로 $-\infty < a_m < \infty$

unbounded real number는 coding (양측)하기
매우 까다롭다.

2021. 1. 27. LPC to LSF

2) LSF (line spectral frequencies)

or equivalently LSP (line spectral pairs)

$$P(z) = A(z) + z^{-(p+1)} A(z^{-1})$$

$$Q(z) = A(z) - z^{-(p+1)} A(z^{-1})$$

$A(z)$ 로 부터 위의 두개의 polynomial을 정의할 수 있으며

두개 (pair) \Rightarrow line spectral pair (LSP)

* 특징

① $P(z)$ 와 $Q(z)$ 의 root들은 unit circle 위에 존재한다.

$z^* = e^{j\omega^*} \Rightarrow$ 주파수값들인 ω^* 로 표현된다.

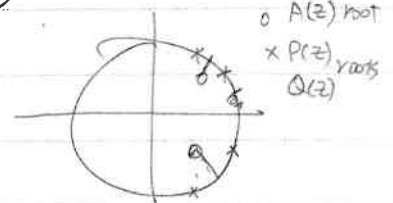
\Rightarrow line spectral frequencies (LSF)

② 두개의 LSF가 가까워수록

Spectral peak (resonance) frequency

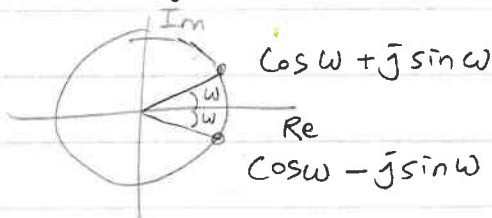
에 가깝다 (formant freq에 가깝다)

$\Rightarrow A(z)$ 의 root가 unit circle에 가까워지기 때문



③ $P(z)$, $Q(z)$ 의 root들은 $\omega = \pi$ 에 대하여 complex conjugate로 존재하기 때문에 절반만 있으면 된다.

즉, ω 가 root이면 $-(2\pi - \omega)$ 도 root이다.



$A(z)$ 의 order가 M 이면 $P(z)$, $Q(z)$ 는 $M/2$ 의 root를

표현 가능하며, 따라서 $\# \text{LSFs} = M$ (LPC order)

* LPC order가 홀수가 되면 어떻게 될지 모르겠다. 2쌍 짝수 M만 쓴다.

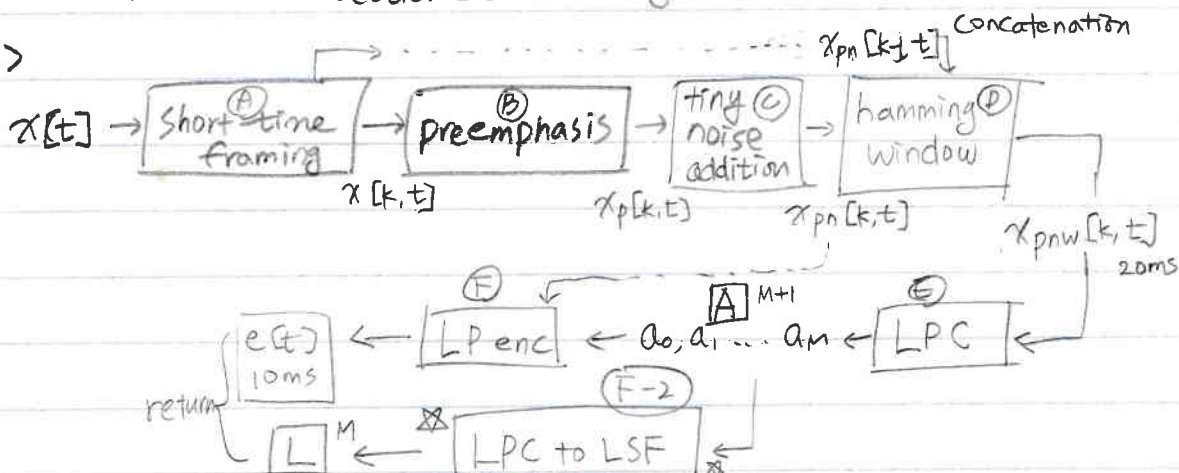
④ 위의 특징에 따라 $\text{LSF} \in [0, \pi]$ 이고 uniform 하게 분포한다.

LSF 추정/coding 오류는 LSF 값에 상관없이 linear 하다

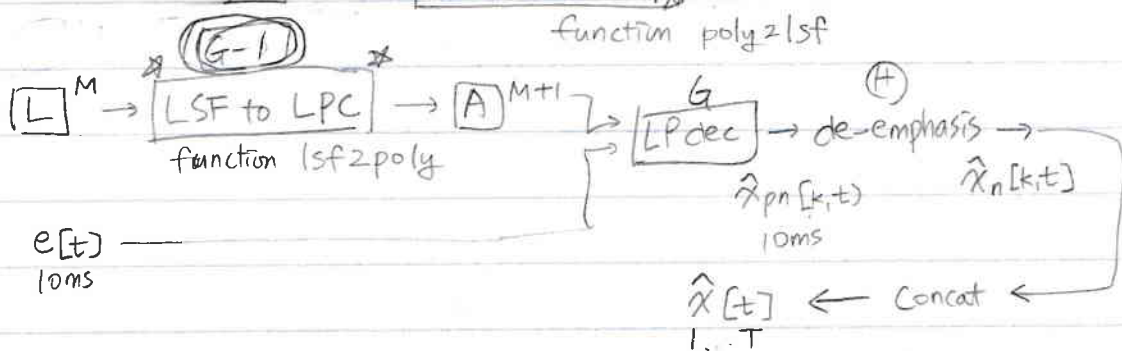
\Rightarrow coding에 매우 유리함.

2021. 1. 27. Vocoder - LSF design

encoder >



decoder >



(F-2) LPC to LSF : $M+1 \Rightarrow M$
 M -th order LP polynomial $A(z)$ (or equivalently LP coefficients $a[0] \dots a[M]$)
 을 M 개의 LSF로 변환한다.

→ method "poly2lsf" 를 검색하여 적절한 것을 찾는다.

찾아보면 LPC, vocoder 관련 tool에 있다.

python> def poly2lsf(a): return lsf
 encoder의 출력은 $(\frac{e[t]}{10ms}, \frac{lsf[n]}{M}) * K$

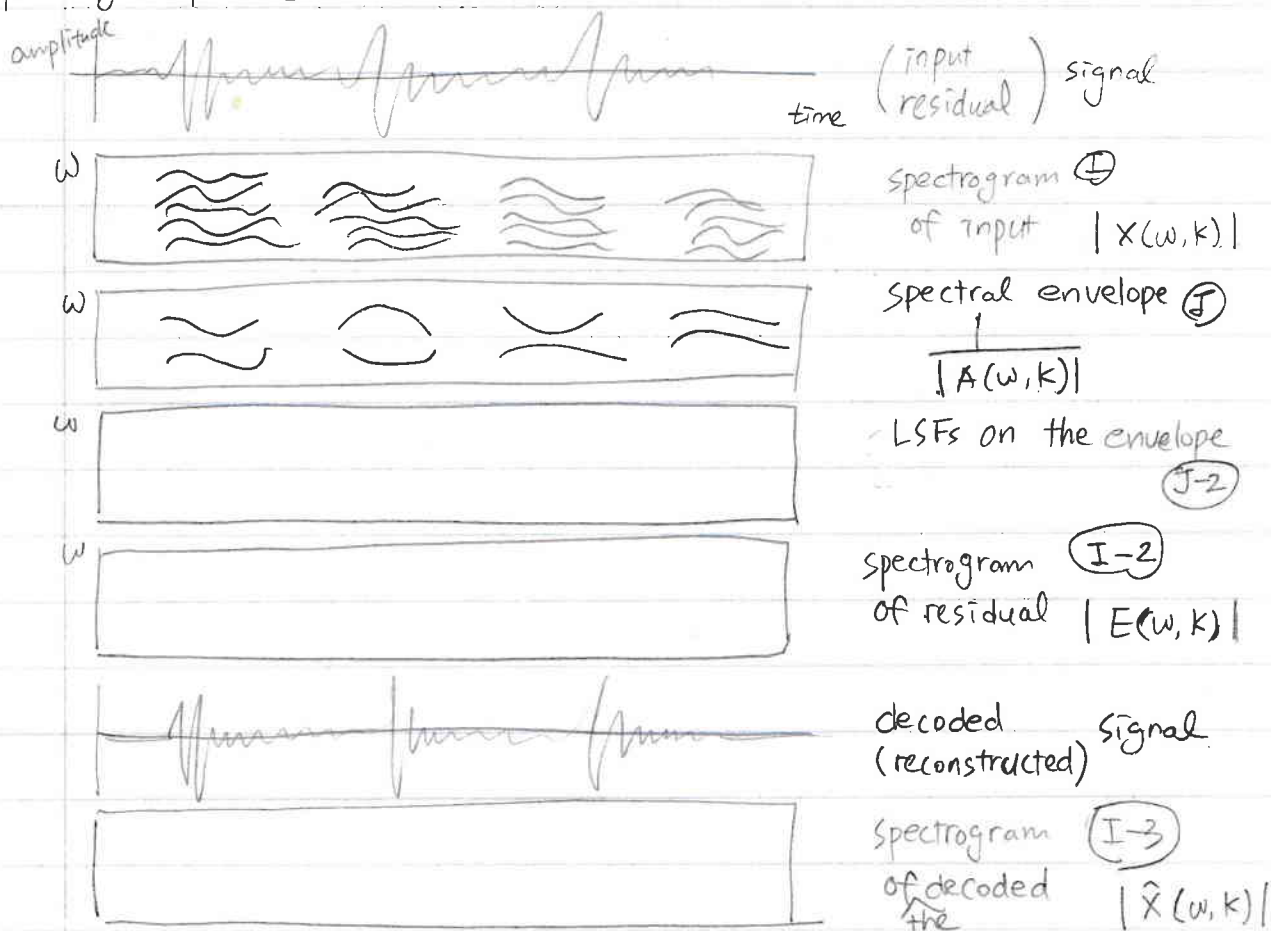
(G-1) LSF to LPC : $M \Rightarrow M+1$

→ method "lsf2poly" 찾아서 사용

<python> def lsf2poly(lsf): return a
 decoder의 입력은 $(e[t], lsf[n]) * K \Rightarrow$ 출력 $\hat{x}(t) (t=1 \dots T)$

2021. 1. 27. Vocoder-LSF design : plotting

< plotting implementation >



(I-1), (I-2): 1-D array 에서 STFT spectrum 을 구해서 그려다.

(I-3) pyplot. spectrogram 을 쓰면 된다 log/dB scale

(J): A 는 2-D array of size $M \times K$

for each $A[:, k]$ of length M , do FFT with $NFFT = 20ms$ (320 points) 16kHz

$\Rightarrow 0 \sim M-1$ 만 값이 있고 나머지는 0 으로 채운다.

hamming window 는 곱할 필요없다.

\Rightarrow smoothed Spectrum 이 그려짐.

* Spectrum 은 $0 \sim \pi$ 만 그려다. 전반만 그려야 하고, 0과 π 를 모두 포함하기 위해 전반+1이 필요함 $\left(\frac{NFFT+2}{2} \right)$ use np.fft.

2021. 1. 27. Vocoder-LSF design : VQ

- J-2 LSF를 $|A(\omega)|$ 위에 pyplot. plot 함수를 이용하여 그린다.
그리고 LSF와 spectral peak 등과의 관계를 본다.
* 양측의 범위를 잘 맞춰야 한다.

[Note] 예제로 주어진 출력그림을 참고하여 작성한다

2) Efficient LSF coding by k-means vector quantization >

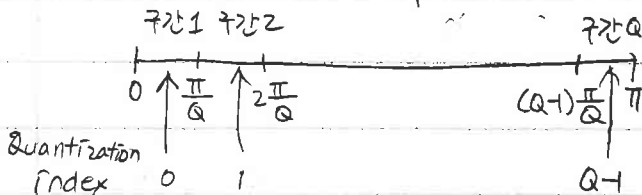
$lsf \in [0, \pi]$ real value인 lsf를 점진적으로 압축한다
 \Rightarrow lsf를 integer index로 변환.

① scalar quantization

codebook size은 지칭됨
($Q = 16, 32, 64, 128, \dots$)

lsf는 uniform, so $[0, \pi]$ 를 Q 개로 균등하게 나눈다.

M 개의 lsf를 independent 하게 quantization, 다음과 같이 간단하게



$$q(x) = \text{floor}\left(\frac{x}{\pi} \cdot Q\right)$$

($x = \pi$ 일 경우 처리 필요)

② 만약 $Q = 16 = 2^4$ 이면 4 bits가 필요

LPC 10 은 $4 \times 10 = 40$ bits 필요

* non-uniform scalar quantization: $10 = 1 + 4 + 5$

4개의 variable Q 로 quantize: $1 \times 3 + 4 \times 4 + 5 \times 3 = 34$ bits

③ $Q = 256 = 2^8 \rightarrow 8 \times 10 = 80$ bits / frame

③ Split vector quantization :

$10 = 3 + 3 + 4$, each subdimension quantized by $Q = 256$.

$\begin{matrix} \text{low-freq} & \text{mid} & \text{high} \\ \text{freq} & & \text{freq} \end{matrix}$

\Rightarrow subdimension size가 작을수록 더 자세히 모델링.

즉 Q 가 고정되어 있으면 dimension(즉 주어진 information)이 더 많아지고 압축도 줄어든다.

#bits = $3 \times 8 = 24$ bits / frame

2021. 1. 27. Vocoder-LSF design: VQ

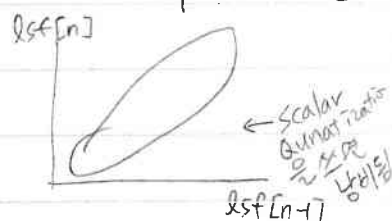
- ② * scalar / split vector quantization의 경우 각 (sub-)dimension들이 independently quantization 되기 때문에 $lsf[n-1] \leq lsf[n]$ 이 유지되도록 후처리가 필요할 수 있다. (codebook design을 잘 하면 필요없을 수도 있다)

③ full vector quantization:

$lsf[n-1] \leq lsf[n]$ 이기 때문에 전체 M-dim vector space에 있는 positive cross-correlation 값이 크다 (skewed)

학습을 통해 full-dimension VQ를 하고

그 code word를 사용하면 매우 효율적으로 압축이 가능하다



<codebook training>

- 많은 개수의 wave file을 준비함 (training set)
- LSF를 추출한다 ($G-1$, (M, K_1) , (M, K_2) ... (M, K_N))
- k-means VQ 학습 (scikit-learn 사용 가능)

input: $M \times (K_1 + K_2 + \dots + K_N)$ LSF vectors

output: $M \times Q$ codebook of size Q

$1 \times (K_1 + K_2 + \dots + K_N)$ quantization indexes of training data (optional)

algorithm: quantization error (Euclidean)가 최소가 되도록 학습

- LSF quantization ($G-2$)

input: $M \times Q$ codebook

$M \times 1$ LSF vector to quantize

output: g : integer

* Codebook size: $Q=256 \rightarrow 8 \text{ bits}$
 $Q=512 \rightarrow 9 \text{ bits}$ } 필요 비트수

page 8/8

2021. 1. 27 Vocoder-LSF design: assignment 2

<programming assignment #2>

2-1. LPC to LSF, LSF to LPC 를 구현하여 vocoder design.

2-2. Spectrogram 으로 보기.

2-3. LSF codebook 탐색 \rightarrow limit-selected 의 wav 파일들을

2-4. Codebook 과 quantization index 를 이용한

vocoder design

이용하여
탐색한다

* 추가사항 2021. 2. 25.

LSF를 codebook mapping 으로 얻은 후에

새로 LPC를 구하고 그 LPC로 residual을 다시
뽑아서 전송해야 함.

\Rightarrow mapping 이전의 LSF를 사용하면

SNR이 매우 작아짐 (소리도 왜곡)