

ヒューマンインターフェース

Report2

学籍番号 045713C:大城和也

提出日:平成 18 年 07 月 23 日 (日)

1 課題内容

音声信号に対する次の3つのスペクトルを算出，描画する SCILAB プログラム (lpc.sci) を提供します．

- Hamming 窓で窓掛けした DTF スペクトル．
- 自己相関 LPC 分析 (Levinson-Durbin 法) で得られたスペクトル．
- LPC 逆フィルタで算出された残差のスペクトル．

課題として，次の事を行ない，レポートにまとめてください．

- (1) プログラムを読解する．
- (2) 提供プログラムでは Levinson-Durbin 法を SCILAB の関数 `lev()` で実現しています．関数を使わず SCILAB 言語でプログラミングして実現しましょう．
- (3) プログラムの最後に伝達特性=0 の根を `roots()` により算出し，極周波数と帯域幅を極周波数の小さい順に表示するプログラムを追加してください．0 から 5KHz の極だけを抽出し，極周波数で昇順にソーティングしましょう．実軸に根が存在する場合もあるので，0 から 5KHz の極は次数の半分とは限りません．結果があっているか，スペクトルと極配置図により確認しましょう．
- (4) 男性音と女性音それぞれで，分析長を 100 から 300，分析次数を 8 から 20 程度に変化させ，スペクトルがどう変化するかを調べ，その結果について考察しましょう．
- (5) `pre_emp` を 1.0 に設定し，各スペクトルがどう変化するかを調べ，なぜそうなるか考察しましょう．

2 プログラム解読

初期化部分

```
fft_len=512;           // FFT のサンプル数
start=5000;           // 分析位置
len=300;              // 分析長
order=14;             // 分析次数
pre_emp=0.0;          // プリエンファシス
print(%io(2),'start= '); start = read(%io(1),1,1);
print(%io(2),start,len,fft_len,order);

x=loadwave('1.wav'); // 1.wav というファイルの読み込み
```

ここでは，ファイルの読み込みや初期値の設定などの初期化の作業が行われている．

Hamming 窓掛け&プリエンファシス

```
// Hamming 窓掛け&プリエンファシス
for i=1:len
    win(i) = 0.54 - 0.46 * cos(2 * %pi * i / len);
    x1(i) = (x(i+start)-pre_emp*x(i-1+start)) * win(i);
end
for i=len+1:fft_len // ゼロ詰め
    x1(i) = 0;
    win(i) = 0;
end
```

最初の for 文によってハミング窓掛けを行なっている．ハミング窓は次式をもって作成される．

$$win(i) = 0.54 - 0.46 \times \cos\left(\frac{2 \times \pi \times i}{len}\right)$$

ハミング窓を作成して次文で適応させている．窓掛けは

$$x(i) = x(i + start) \times win(i)$$

とすることで行なっている．なお，プログラム中ではプリエンファシスを用いた処理も行なっているが，提供されたプログラムでは pre_emp が 0 となっているのでこの状態では使わない事を意味している．

次の for 文では指定した場所からの分析長に入らなかった値を全て 0 として処理している．

DFT 対数スペクトル

```
// DFT 対数スペクトルの算出
fft_spc=20*log10(abs(fft(x1,-1)));
```

$x(1), x(2), \dots, x(i)$ を入力としてフーリエ変換を行い，abs() で絶対値にする．また $20 \times \log_{10}()$ として dB の値を算出し，fft_spc として保存しておく．この時の式は以下のように示される．

$$fft_spc = 20 \log_{10} \left(\sum_{n=1}^N x1(n) e^{j2\pi(k-1)\left(\frac{n-1}{N}\right)} \right)$$

自己相関

```
// 自己相関関数の算出
j=1;
for i=0:order
    r(j)=0;
    for n=1:len-i
        r(j)=r(j)+x1(n)*x1(n+i);
    end
    j=j+1;
end
```

自己相関を求めている．自己相関の関数は次式によって求められる．

$$r(i) = \sum_{t=0}^{N-1-i} x1(t)x1(t+i)$$

LPC 分析

```
// Levinson 算法による LPC 分析
[ar,sigma2,rc]=lev(r); // sigma2=r(0)+_sum_{i=1}^order(r(i)*a(i))
```

Levinson 算法 (自己相関法) によって LPC 分析を行なっている . Levinson は scilab にある関数 lev() を使う事で実現している . これによって出力される ar,sigma2,rc はそれぞれ , LPC 係数 , 予測誤差 , PARCOR 係数を示している .

LPC 対数スペクトル

```
// LPC 対数スペクトルの算出
a(1)=1;
for i=1:order
    a(i+1)=ar(i);
end
for i=order+1:fft_len-1 // ゼロ詰め
    a(i+1)=0;
end
ar_spc=-20*log10(abs(fft(a,-1)))+10*log10(sigma2);
```

a の初期値を 1 として次からの値には LPC 係数をコピーする .

DFT 対数スペクトルの時と同じ様に , デシベル単位へと変換を行っている (ただし符号はマイナスとする) , この値に誤差のデシベルと足し合わせて LPC の対数スペクトルとしている .

残差のスペクトル

```
// 残差のスペクトルの算出
for n=1:len
    res(n)=x(start+n);
    for i=1:order
        res(n)=res(n)+ar(i)*x(start+n-i);
    end
end
for n=len+1:fft_len
    res(n)=0;
end
res_spc=20*log10(abs(fft(res,-1)));
```

LPC の残差予測は以下の式になる .

$$res(t) = x(t) + \sum_{i=1}^I ar(i)x(t-i)$$

出力関係

```
// スペクトルの描画
xset('window',1); xbas(1);
rect=[1,min(fft_spc),fft_len/2,max(fft_spc)];
tics=[2,4,2,4];
plotframe(rect,tics,[%f,%f],[ 'LPC', 'Freq.', 'Amp. [dB]'],[0,0,1,0,0.5]);
n=1:fft_len/2;
plot2d(n,fft_spc(n),1,"000");
plot2d(n,ar_spc(n),2,"000");
plot2d(n,res_spc(n),3,"000");

// 極を算出し描画
HAR=poly(a(1:order+1),'z','coeff');
pp=roots(HAR);
for i=1:order
    pp(i)=1/pp(i);
end
x=0:0.1:2*pi;
// xbas();
rect=[-1,-1,1,1];
tics=[2,5,2,5];
plotframe(rect,tics,[%f,%f],[ "Unit Circle", "Re.", "Im."],[0.25,0.5,0.5,0.5]);
plot2d(cos(x),sin(x),1,"000");
ra=real(pp); ia=imag(pp);
plot2d(ra,ia,-3);
xgrid();
```

ここでは、算出したスペクトルを図として出力する処理が行なわれている。

3 Levinson-Durbin プログラム

Levinson-Durbin の式は以下ようになる。

$$\begin{aligned} ar(i:k) &= ar(i:k-1) - rc(k)ar(k-i:i-1) \quad i=2, \dots, k \\ &= ar_i^{(k)} \\ rc(k) &= \frac{ak1(k-1)}{\sigma_{2k-1}} \\ \sigma_{2k} &= (1 - rc(k)^2) \times \sigma_{2k-1} \\ ak1(k) &= r(k) + \sum_{i=2}^k ar(i:k)r(k+1-i) \end{aligned}$$

これらをプログラムとしたのが以下である。

今回、提供されていたプログラムにおける `lev()` 関数はこのプログラムと置き換えることが可能。

Levinson プログラム

```
rsi= max(size(r));  
r=matrix(r,1,rsi);  
ar=-r(2)/r(1);  
rc(1)=ar;  
sigma2=(1-ar*conj(ar))*r(1);  
for k=2:rsi-1,  
    ak1(k) = -(r(k+1)+ar(1:k-1)*r(k:-1:2)')/sigma2;  
    rc(k) = ak1(k);  
    ak1(1:k-1) = ar(1:k-1)+ak1(k)*conj(ar(k-1:-1:1));  
    sigma2=(1-ak1(k)*conj(ak1(k)))*sigma2;  
    ar=ak1;  
end,
```

4 追加のプログラム

追加部分

```
k=1;  
for i=1:order  
    if imag(pp(i)) >= 0 then  
        pp2(k)=pp(i);  
        k=k+1;  
    end  
end  
  
//バブルソート  
k=size(pp2);  
i=1;  
while i<=(k(1)-1)  
    j=k(1);  
    while j>i  
        if real(pp2(j))<real(pp2(j-1)) then  
            l=pp2(j);  
            pp2(j)=pp2(j-1);  
            pp2(j-1)=l;  
        end  
        j=j-1;  
    end  
    i=i+1;  
end  
  
//表示  
disp(pp2);
```

これを追加する事により，実行結果に以下が追加された．

実行結果

```
--> disp(pp2);  
!  
! - 0.9065070 + 0.3228199i !  
! - 0.5025580 + 0.8361921i !  
! - 0.1697517 + 0.9047809i !  
! - 0.0729275 + 0.5971578i !  
! 0.7198840 + 0.6617157i !  
! 0.7932980 + 0.5245589i !  
! 0.9048426 + 0.1661712i !
```

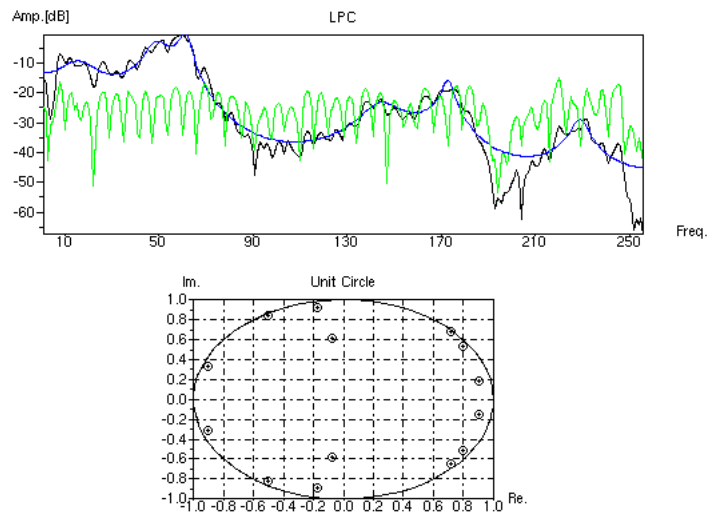


図 1: 実行結果

5 分析長および分析次数の考察

今回は分析位置を 4000 と固定し、男性音、女性音の分析長を 100,200,300 , 分析次数を 8,14,20 の組み合わせを出力した . 以下の図 2 ~ 図 19 がそれぞれの結果となっている .

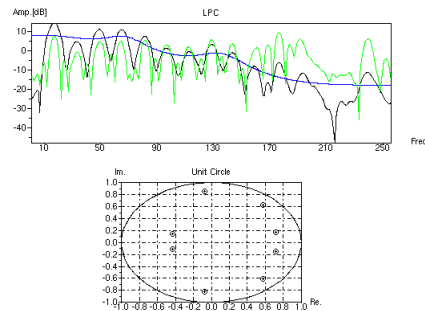
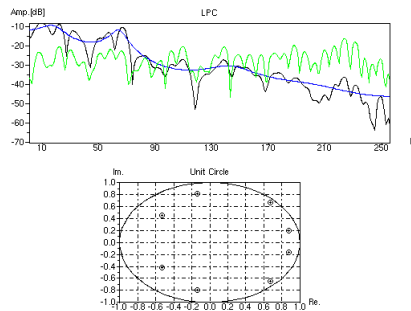


図 2: 男性音 分析長 100 分析次数 8

図 3: 女性音 分析長 100 分析次数 8

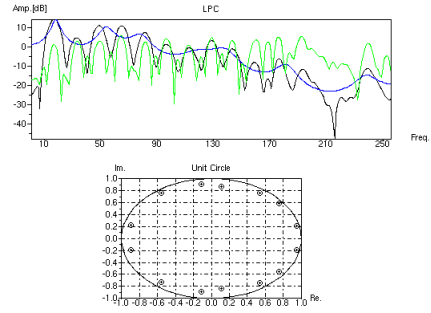
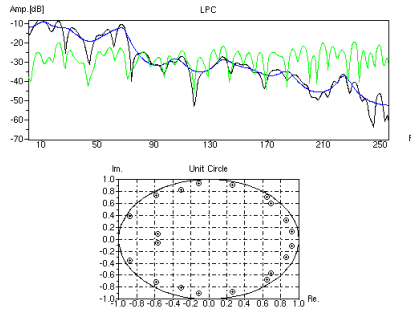


图 4: 男性音 分析長 100 分析次数 14 图 5: 女性音 分析長 100 分析次数 14

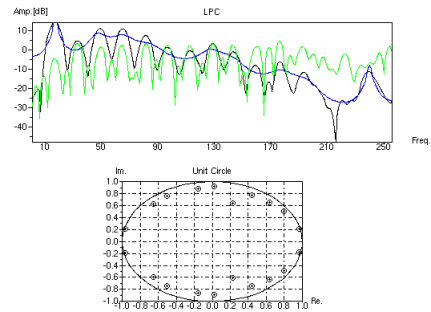
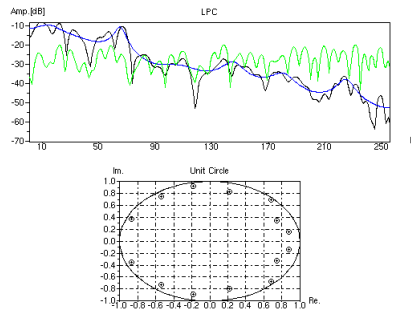


图 6: 男性音 分析長 100 分析次数 20 图 7: 女性音 分析長 100 分析次数 20

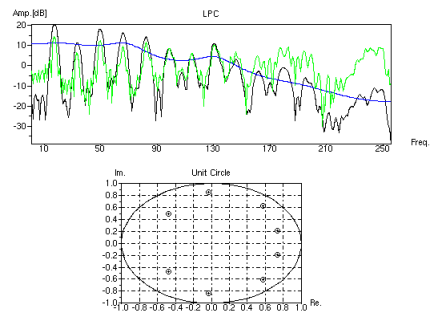
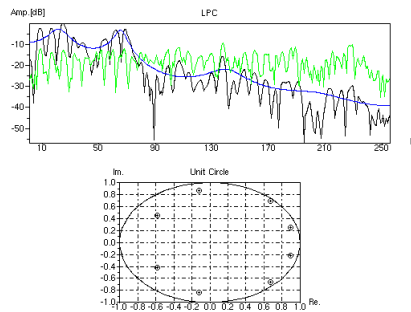


图 8: 男性音 分析長 200 分析次数 8 图 9: 女性音 分析長 200 分析次数 8

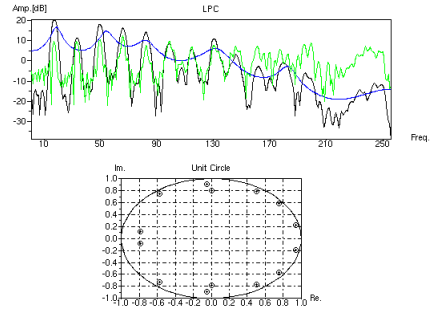
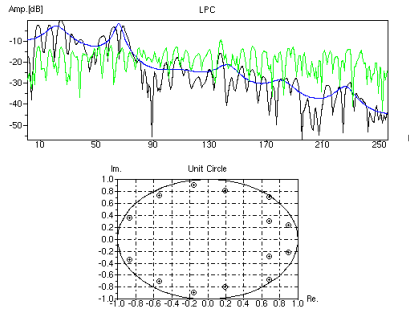


图 10: 男性音 分析長 200 分析次数 14 图 11: 女性音 分析長 200 分析次数 14

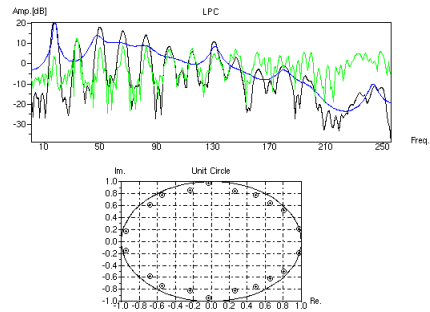
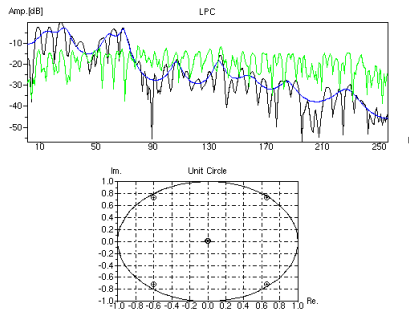


图 12: 男性音 分析長 200 分析次数 20 图 13: 女性音 分析長 200 分析次数 20

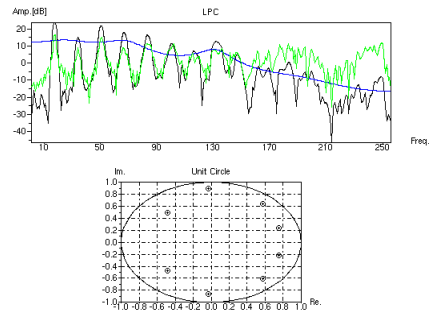
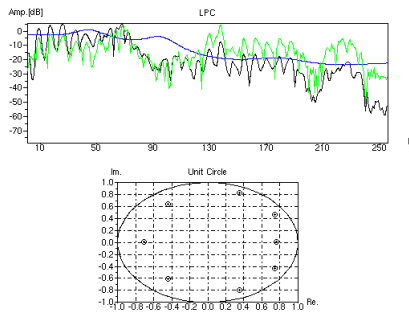


图 14: 男性音 分析長 300 分析次数 8 图 15: 女性音 分析長 300 分析次数 8

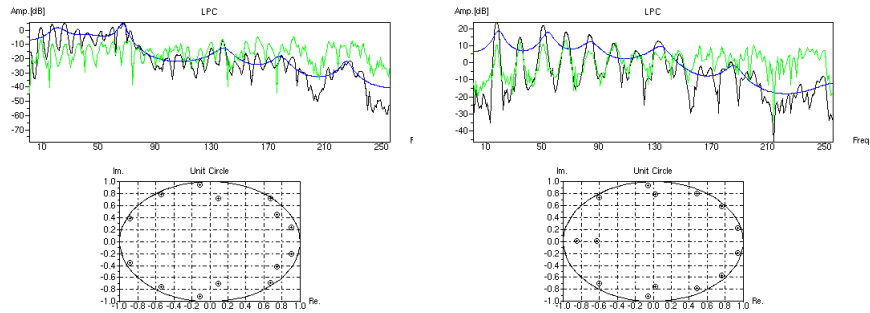


図 16: 男性音 分析長 300 分析次数 14 図 17: 女性音 分析長 300 分析次数 14

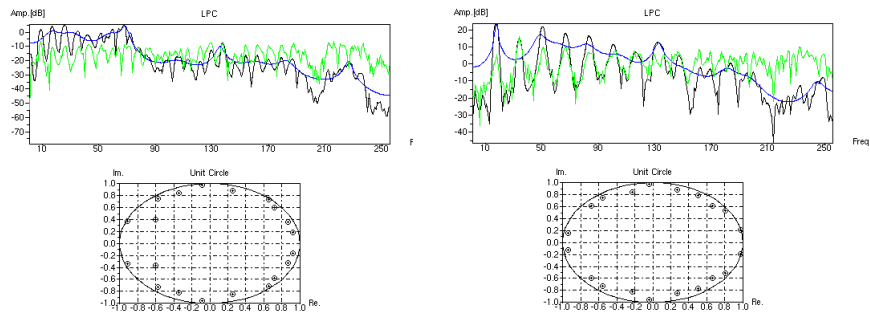


図 18: 男性音 分析長 300 分析次数 20 図 19: 女性音 分析長 300 分析次数 20

5.1 考察

分析長を上げる事によって DFT スペクトルの振動が細かくなり、帯域幅が狭まっているのが見て取れる。また、同様な事が残差予測スペクトルに見られる。LPC スペクトルは分析次数によって変化している事がわかる。分析次数を増やす事によって極(ホルマント)が増加している。男性音と女性音を比べてみると男性音の帯域幅が女性音の帯域幅と比べて狭く、人間の音声の特徴を分析している事がわかる。

6 プリエンファシスの考察

プリエンファシスの値を 1.0 にすることによってハミング窓掛けが適応されるようになった。そのために、図 21 は図 20 と比べて、始めの部分で急激に落ち込んでいる。また、プリエンファシスを用いた計算によって高周波数の時の値が若干だが高くなっている。

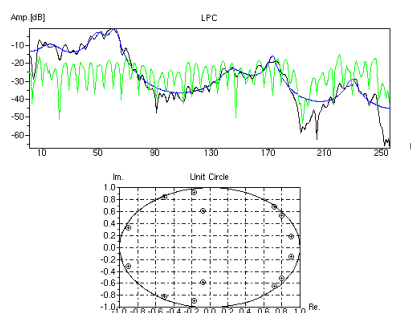


図 20: pre_emp 0.0

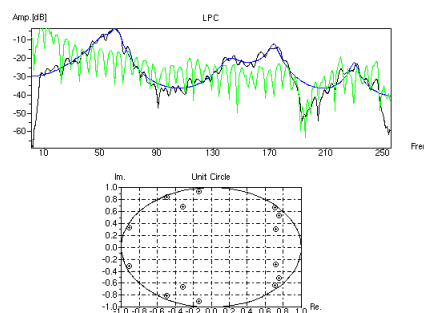


図 21: pre_emp 1.0

参考文献

- [1] フーリエ変換と窓関数,
”http://www.orixrentec.co.jp/tmsite/know/know_mado.html”
- [2] デジタル信号処理,
”<http://laputa.cs.shinshu-u.ac.jp/~yizawa/InfSys1/basic/index.htm>”
- [3] SCILAB,
”<http://www.cc.u-ryukyu.ac.jp/~funaki/scilab.html>”
- [4] 信号処理～第2部 確率的な信号の処理～, 横田 構成