实验四　LPC 编解码实验

【实验目的】

1、了解语音信号 LPC 编码的原理

2、掌握 LPC 编解码的步骤流程

3、根据原理能编程实现 LPC 编码和解码的计算

【实验要求】

1、编程要求：编写一段 MATLAB 程序

2、实现功能：实现语音信号的 LPC 编码和解码

【实验原理】

线性预测编码（linear predictive coding, LPC）是运用于音频信号处理与语音处理的压缩编码方式，根据线性预测模型的信息表示数字语音信号谱包络。它是最有效的语音分析技术之一，也是低位速下编码方法高质量语音最有用的方法之一，能够提供非常精确的语音参数预测。线性预测编码通过估计共振峰剔除它们在语音信号中的作用，估计保留的蜂鸣音强度与频率来分析语音信号；同时，使用蜂鸣参数与残余信号生成源信号，使用共振峰生成表示声道的滤波器，源信号经过滤波器的处理来逆向合成语音信号。由于语音信号随着时间变化，这个过程是在一段段的语音信号帧上进行处理的，通常每秒 30 到 50 帧就能对可理解的信号进行很好的压缩。

线性预测编码通常用于语音的重新合成，它是电话公司使用的声音压缩格式，如 GSM 标准就在使用 LPC 编码格式。它还用作安全无线通信中的格式，在安全的无线通信中，声音必须进行数字化、加密然后通过狭窄的语音信道传输。

线性预测分析的基本思想是：由于语音样点之间存在相关性，所以可以用过去的样点值来预测现在或将来的样点值，即一个语音抽样可以用过去若干个语音抽样或它们的线性组合来逼近。通过使实现语音抽样与线性预测抽样之间的误差在某个准则（通常为最小均方误差准则）下达到最小值来决定一组预测系数。这一组预测系数就反映了语音信号的特性，可以作为语音信号的特征参数用于语音合成和语音识别等。

下面对 LPC 的基本原理和计算过程作一个介绍。

1、 线性预测基本原理

线性预测分析对语音的产生过程有一个基本的假设，即认为语音是由一个激励信号通过一个滤波器（响应函数）得到。通过对声道模型的研究，可以认为系统的传递函数符合全极点数字滤波器的形式。系统的传递函数如下:

$$H(z)=\frac{G}{1-\sum_{k=1}^{p}a_k z^{-k}} \tag{4-1}$$

其中，p 是极点个数即模型结束，G 是幅值因子，a_k 是模型系数，由 p 和 a_k 决定了声道特性，描述了说话人的特征。对于一个线性预测系统，采样点的输出 $s(n)$ 可以用前面 p 个样本的线性组合来表示，定义系统输出的估计值为：

$$\tilde{s}(n)=\sum_{i=1}^{p}a_i s(n-i) \tag{4-2}$$

其中系数 a_i 为预测系数，p 为预测阶数。预测误差表示如下：

$$e(n)=s(n)-\tilde{s}(n)=s(n)-\sum_{i=1}^{p}a_i s(n-i) \tag{4-3}$$

定义短时预测均方误差为：

$$E_n=\sum_n e^2(n)=\sum_{i=1}^{n}[s(n)-\sum_{i=1}^{p}a_i s(n-i)]^2 \tag{4-4}$$

显然，均方误差越接近于 0，预测的准确度在均方误差最小的意义上为最佳。因此，应满足 E_n 对于各系数 a_i 的偏微分为 0，这样计算可得到线性预测的标准方程为：

$$\sum_n s(n)s(n-j)=\sum_{i=1}^{p}a_i s(n-i)s(n-j),1\le j\le p \tag{4-5}$$

可定义：

$$\phi(j,i)=\sum_n s(n-j)s(n-i) \tag{4-6}$$

则上式可简写为：

$$\sum_{i=1}^{p}a_i\phi(j,i)=\phi(j,0) \tag{4-7}$$

求解含有 p 个未知数的方程组可得到各预测系数，利用上式，可得最小均方误差为：

$$E_n=\sum_n s^2(n)-\sum_{i=1}^{p}a_i\sum_n s(n)s(n-i) \tag{4-8}$$

因此，最小误差由一个固定分量和一个依赖于预测系数的分量组成，求解最佳预测系数，需要首先计算 $\phi(i,j)$，然后可按上式求出 a_i 。

2、 线性预测系数的计算

一般来说，可以采用莱文逊-杜宾（Levinson-Durbin）递归算法来求解线性预测系数，假设 $s(n)$ 在[0,N-1]外等于 0，即 $s(n)$ 经过有限长度的窗处理。

$s(n)$ 的自相关函数为：

$$r(j)=\sum_{n=-\infty}^{\infty}s(n)s(n-j)=\sum_{n=0}^{N-1}s(n)s(n-j) \tag{4-9}$$

由上可知，$\phi(j,i)$ 即为 $r(j-i)$。$r(j-i)$ 只与 j 和 i 的相对大小有关，与 i,j 的取值无关，因此：

$$\phi(j,i)=r(|j-i|) \tag{4-10}$$

这是一个含 Toeplitz 矩阵的 Yule-Walker 方程。利用 Toeplitz 矩阵的性质可得到求解 Yule-Walker 方程的高效解法—杜宾算法。其迭代过程如下：

(1) 计算自相关系数 $r_n(j)$；

(2) $E^{(0)}=r_n(0)$；

(3) $i=1$；

(4) 开始按公式 $k_i=\dfrac{r_n(i)-\sum_{j=1}^{i-1}a_j^{(i-1)}r_n(i-j)}{E^{(i-1)}}$ 进行递推计算；

(5) 使 $i=i+1, i>p$，则算法结束退出，否则返回(4)，按上式进行递推。

递推的最终解为：

$$a_j=a_j^{(p)}, 1\le j\le p \tag{4-11}$$

$$E^{(p)}=r_n(0)\prod_{i=1}^{p}(1-\mathrm{k}_i^{2}) \tag{4-12}$$

通过 LPC 分析，由若干帧语音可得到若干组 LPC 参数，每组参数形成一个描绘该帧语音特征的矢量，即 LPC 矢量。

LPC 参数是模拟人的发声器官的，是一种基于语音合成的参数模型，每段声管对应一个 LPC 模型的极点。一般情况下，极点个数在 12-16 之间，就可以足够清晰地描述语音信号的特征了。选择 p=12 可以对绝大多数的语音信号声道模型取得足够近似的逼近。p 值选择过大虽可以改善逼近效果，但会增大计算量，且可能增添一些不必要的细节，如在用声道模型谱进行共振峰分析时效果变差等。

【实验步骤】

1、 为了显示方便，分别编写编码、解码函数程序 encode1.m、decode1.m。encode1.m 对输入的原模拟信号进行线性预测编码，decode1.m 对生成的 LPC 码进行解码恢复出原信号。具体函数定义如下。

函数格式：[ipitch,irms,irc]= encode1(pitcha,rms,kk)

输入参数：pitcha 是基音频率，rms 表示均方误差，kk 表示预测系数。

输出参数：ipitch 为已编码的基音频率，irms 为量化的均方误差，irc 为量化的预测系数。

函数格式：[voice,pitcha,rms,kk]=decodel(ipitch,irms,irc,rcn)

输入参数：ipitch 为已编码的基音频率，irms 为量化的均方误差，irc 为量化的预测系数，rcn 为系数阶数。

输出参数：voice 为半帧音频信号，pitcha 为解码得到的基音频率，rms 为得到的均方误差，kk 为预测系数。

2、编程实现对信号的编码和解码过程，波形如图 4-1 所示。

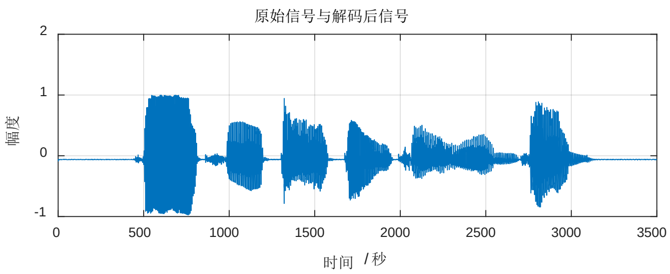

图 4-1 LPC 编解码对比图

【思考题】

编程比较不同的线性预测阶数对 LPC 编解码后语音恢复的影响。