第5章

第5章 离散时间系统的相位、结构

5.5 谱分解、反卷积及系统辨识

线性相位FIR系统的零点有镜像对称关系;

IIR全通系统的极点和零点成镜像对称关系;

IIR系统有最小相位、最大相位、混合相位系统之分，在具有相同幅频响应的滤波器集合中，零点有镜像对称关系。

由零极点图做滤波器的频率响应。

例5.4.3，三个因果稳定的系统，极点相同，零点分别在单位圆内、单位圆外，以单位圆为镜像对称。

一、谱分解

1. 具有相同幅频响应的FIR系统

最小相位系统->最大相位系统

$$H_{\min}(z)=(1-az^{-1})(1-bz^{-1}),0<a,b<1$$
$$H_{\max}(z)=(1-az)(1-bz),0<a,b<1$$
$$\begin{aligned}H_{\min}(z)H_{\min}(z^{-1})&=(1-az^{-1})(1-bz^{-1})(1-az)(1-bz)\\&=H_{\max}(z)H_{\max}(z^{-1})\end{aligned}$$

混合相位系统

$$H_{\mathrm{mix1}}(z)=(1-az)(1-bz^{-1}),0<a,b<1$$
$$H_{\mathrm{mix2}}(z)=(1-az^{-1})(1-bz),0<a,b<1$$
$$\begin{aligned}H_{\mathrm{mix1}}(z)H_{\mathrm{mix1}}(z^{-1})&=H_{\mathrm{mix2}}(z)H_{\mathrm{mix2}}(z^{-1})\\&=H_{min}(z)H_{\min}(z^{-1})\end{aligned}$$

2. 线性相位FIR系统$P(z)$的谱分解

令：$P(z) = H(z)H(z^{-1})$　　显然，$P(z)$具有线性相位

将一个转移函数的极－零点重新分配，得到两个转移函数，这一过程（或方法）称为“谱分解”。

将具有线性相位系统的转移函数作分解，在一定条件下可分成两个具有相同幅频响应的子系统。

条件：单位圆上的零点：没有，或偶数倍重零点

谱分解的一般方法

一个系统是最小相位系统，另一个系统必然是最大相位系统，这样，两个系统都有着相同的幅频响应。

另外一种分解方法

得到两个混合系统，目的是保证它们都具有线性相位。二者的幅频响应？

3. IIR系统的谱分解

谱分解的目的是想得到因果的、符合某种要求的系统，这在信号建模、系统辨识、特殊滤波器的设计应用中经常用到。

平稳随机信号的模型？模型存在性原理

例. 令序列$p(n)$为

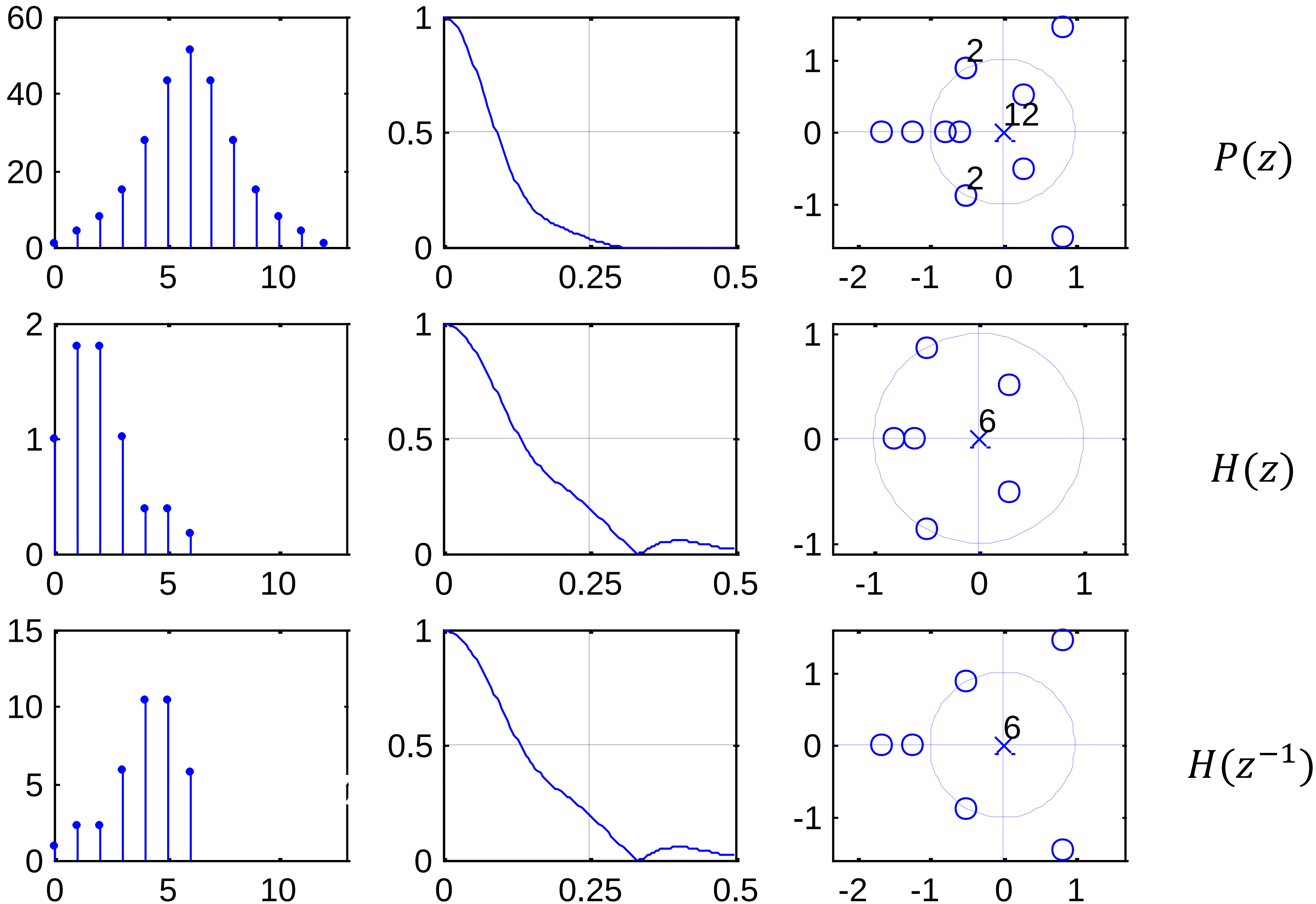
$p(n)$={1.0000, 4.0500, 8.1000, 14.9956, 27.7248, 43.2996, 51.1831, 43.2996, 27.7248, 14.9956, 8.1000, 4.0500, 1.0000}

显然，该系统具有线性相位，共有12个零点：

$$-0.8, \quad -1/0.8, \quad -0.6, \quad -1/0.6,$$

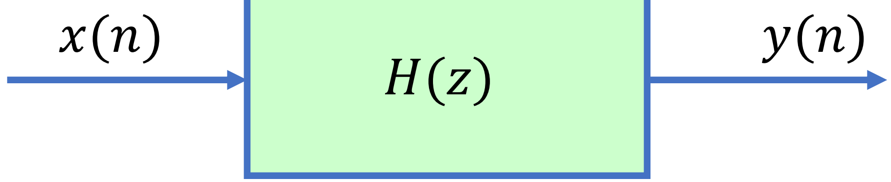
$$e^{\pm j2\pi/3}, \quad e^{\pm j2\pi/3}$$

$$0.6e^{\pm j\pi/3}, \quad e^{\pm j\pi/3}/0.6$$



对$P(z)$作谱分解的结果，分解后的两个系统具有相同的幅频响应。

二、反卷积及系统辨识



若$x(n)$和$h(n)$已知，求$y(n)$，正问题;

多数情况如此

卷积

若$y(n)$和$h(n)$已知，求$x(n)$，逆问题;

心电逆问题，脑电逆问题

已知系统和输出，求源

反卷积

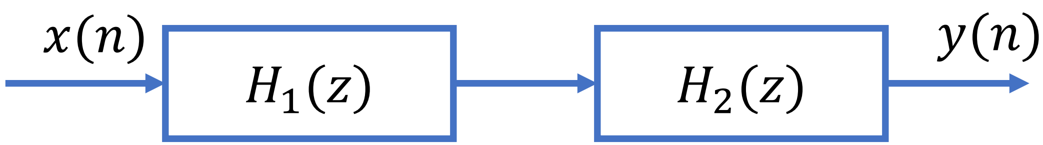
若$x(n)$和$y(n)$已知，求$h(n)$，逆问题；

矿物勘探、地球物理 等领域

已知输入输出，**求系统**

系统辨识

由输出求**输入**和**系统**这两种情况都要用到"反卷积"、"系统辨识" 和"逆系统"的概念。



如果 $$h_1(n) * h_2(n) = \delta(n)$$

$$H_1(z)H_2(z) = 1$$

互为
逆系统

最小相位系统与逆系统

$$H_1(z)=\frac{B(z)}{A(z)} \quad\Longrightarrow\quad H_2(z)=\frac{A(z)}{B(z)}$$

最小相位系统

稳定的充要条件

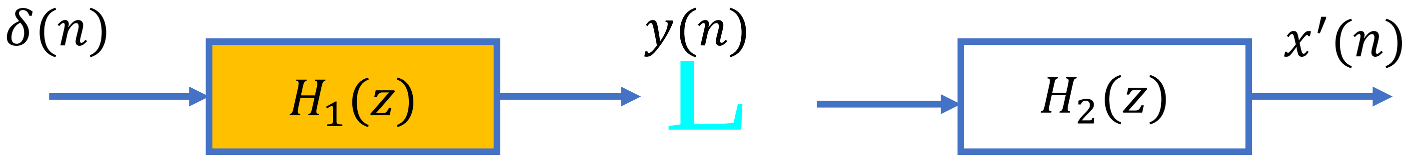
（1）若系统输入、输出已知，希望求系统（系统辨识）

$x(n)$ → $H_1(z)$ → $y(n)$ → $H_2(z)$ → $x'(n)$

调整$H_2(z)$的参数，使$x'(n)$接近等于$x(n)$，则

$$H_1(z)=\frac{1}{H_2(z)}$$

（2）若系统输入未知，输出已知，希望求系统（系统辨识）



调整$H_2(z)$的参数，使$x'(n)$接近等于$\delta(n)$，则

$$H_1(z) = \frac{1}{H_2(z)}$$

（3）若系统输出已知，再知道输入或系统，欲求另一个，可采用反卷积的方法，包括系统辨识

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

$$= \sum_{k=0}^{n} x(k)h(n-k), \quad n \geq 0$$

$\boldsymbol{y(n), h(n) \Rightarrow x(n)}$**，求输入**

$$y(0) = x(0)h(0)$$
$$x(0) = y(0)/h(0)$$

$$y(1) = x(0)h(1) + x(1)h(0)$$
$$x(1) = [y(1) - x(0)h(1)/h(0)$$

依次递推

$$y(n) = x(n)h(0) + \sum_{k=0}^{n-1} x(k)h(n-k)$$

$$x(n) = \frac{\left[y(n) - \sum_{k=0}^{n-1} x(k)h(n-k)\right]}{h(0)} \quad n \geq 1$$

反卷积

与

deconv.m

系统辨识

$\boldsymbol{y(n), x(n) \Rightarrow h(n)}$**，系统辨识**

$$h(n) = \frac{\left[y(n) - \sum_{k=0}^{n-1} h(k)x(n-k)\right]}{x(0)}, \quad n \geq 1$$

(4) 系统辨识 之 频域方法

$$x(n) \longrightarrow \boxed{H(z)} \longrightarrow y(n)$$

$$r_x(m) = \sum_{n=0}^{\infty} x(n)x(n+m)$$

$$r_y(m) = \sum_{n=0}^{\infty} y(n)y(n+m)$$

$$r_{xy}(m) = \sum_{n=0}^{\infty} x(n)y(n+m)$$

$$\begin{aligned} P_y(e^{\mathrm{j}\omega}) &= Y^*(e^{\mathrm{j}\omega})Y(e^{\mathrm{j}\omega}) \\ &= X^*(e^{\mathrm{j}\omega})H^*(e^{\mathrm{j}\omega})X(e^{\mathrm{j}\omega})H(e^{\mathrm{j}\omega}) \end{aligned}$$

$$P_y(e^{j\omega}) = \left|H(e^{j\omega})\right|^2 P_x(e^{j\omega})$$

确定性信号通过LTI系统

自相关和互相关函数
能量信号，定义…
功率信号，定义…

DTFT 时域相关定理
自相关，…
互相关，…
相关与卷积的关系

思考：
功率信号自相关函数的DTFT（功率谱）与信号DTFT的模平方的关系？
回顾DTFT性质：
Parseval 定理
Wiener-Khinchin定理

Wiener-Khinchin定理

功率信号自相关函数定义为

$$r_x(m) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x(n)x(n+m)$$

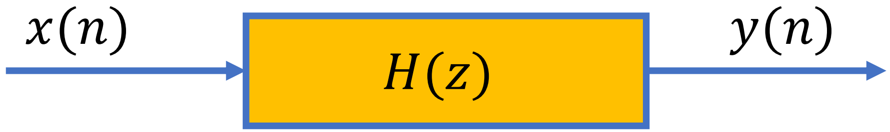
自相关函数的DTFT

$$\sum_{m=-\infty}^{\infty} r_x(m) e^{-j\omega m} = \lim_{N\to\infty} \frac{\left|X_{2N}(e^{j\omega})\right|^2}{2N+1} = \underbrace{P_x(e^{j\omega})}_{\text{信号的功率谱}}$$

信号功率 $P_x = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})\, d\omega$

$$X_{2N}(n) = \begin{cases} x(n) & |n| \le N \\ 0 & |n| > N \end{cases}$$

系统辨识



$$\begin{cases} P_y(e^{j\omega}) = \left|H(e^{j\omega})\right|^2 P_x(e^{j\omega}) \\ P_{xy}(e^{j\omega}) = H(e^{j\omega})P_x(e^{j\omega}) \end{cases}$$

$$\begin{cases} P_y(z) = H(z)H(z^{-1})P_x(z) \\ P_{xy}(z) = H(z)P_x(z) \end{cases}$$

谱分解，得到$H(z)$

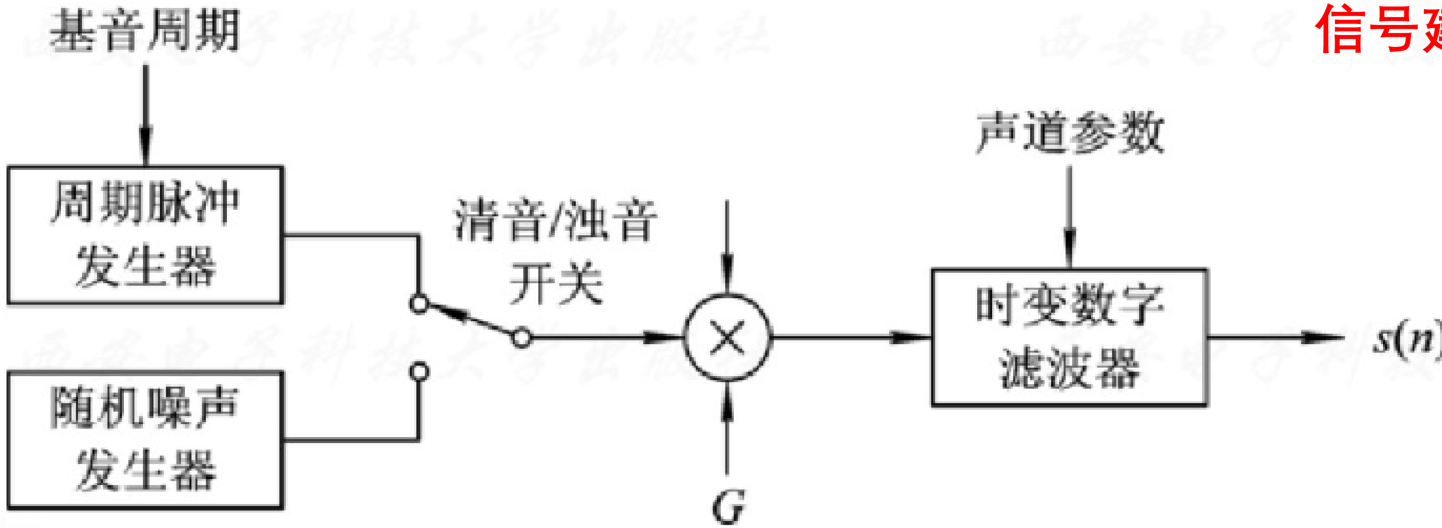
假定输入信号的功率是一平的谱

$$let \quad P_x(z) = 1/k$$

$$\Rightarrow \begin{cases} KP_y(z) = \underline{H(z)}H(z^{-1}) \\ KP_{xy}(z) = H(z) \end{cases}$$

若输入信号是确定性信号，它是？一般的，设平稳随机信号通过LTI系统，这时输入信号是白噪声。

信号建模



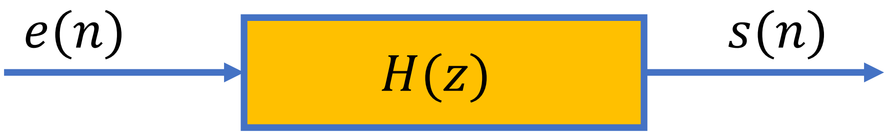
一般情况下，极点个数取8～12个，零点个数取3～5个，在采样率为8 kHz或10 kHz时，$H(z)$在10～20 ms范围内可以很好地反映语音信号的特征。

根据随机过程理论，一个零点可以用若干极点来近似。因此，适当选取极点个数p，可以用全极点模型即AR(p)过程来表达语音信号，即

$$H(z)=\frac{G}{1-\sum_{i=1}^{p}a_i z^{-i}}$$

简化的语音信号数字模型的传递函数，包括声门激励模型、嘴唇辐射模型和声道调制模型。

LPC



$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + e(n)$$

根据采集的语音信号，首先获得声道模型参数，然后获取激励信息。

首先获取声道模型参数。可以理解为声道系统建模，为了简单化，使用指定阶次的全极点模型，则待求解参数为分母多项式系数。

然后执行逆滤波获得激励信息。将全极点模型倒过来得到全零点模型，即FIR系统，再执行逆滤波。

若是语音信号压缩编解码，则将全极点IIR系数、激励信息进行编码、发送，接收方可以重建语音。

线性预测分析的基本原理

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + e(n)$$

- 线性预测分析的基本思想是：用过去p个样点值来预测现在或未来的样点值：

$$\hat{s}(n) = \sum_{i=1}^{p} a_i s(n-i)$$

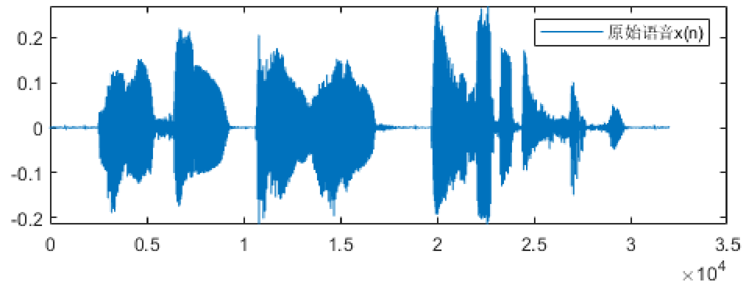
一步预测，自噪声不参与预测

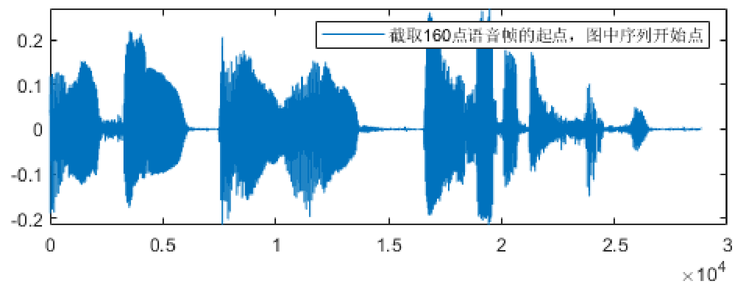
- 预测误差ε(n)为：

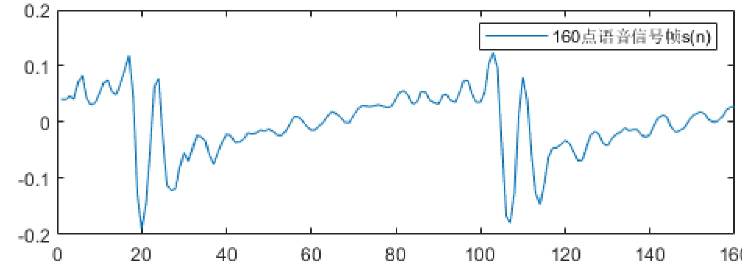
$$\varepsilon(n) = s(n) - \hat{s}(n) = s(n) - \sum_{i=1}^{p} a_i s(n-i)$$

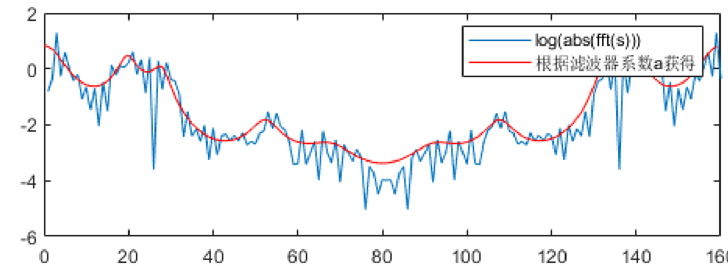
- 这样就可以通过在某个准则下使预测误差ε(n)达到最小值的方法来决定惟一的一组线性预测系数a_i（i=1，2，…，p）。

线性预测分析方法可用于语音信号分析、基于模型的谱估计。全极点模型有快速算法。利用了平稳随机信号的模型存在性原理。



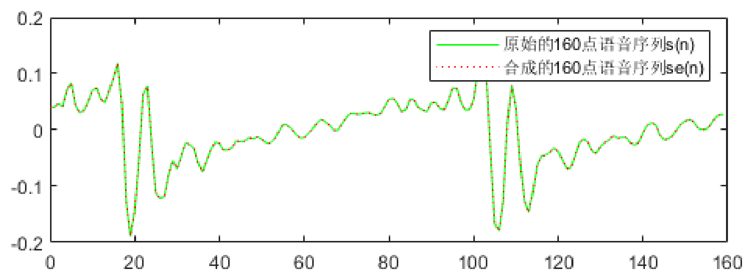


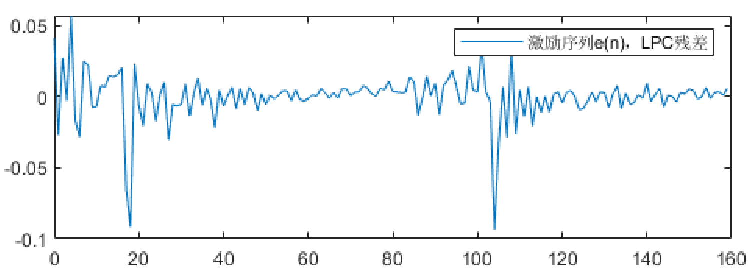




从有效语音信号中获取一帧做实验

```
[a, g] = lpc(s, p);
e = filter(a, 1, s);
se = filter(1, a, e);
```





提取10阶全极点IIR系统声道参数a_i，采用lpc函数；

然后，执行10阶全零点FIR逆滤波，得到激励$e(n)$；

接下来，激励$e(n)$通过全极点IIR，合成语音信号。

思考：合成信号$se(n)$与原始信号$s(n)$如此完美！？

5.6 系统的信号流图与结构

任何线性时不变集总参数离散系统都可用下列三种形式之一来表述其输入输出关系:

差分方程、卷积公式、系统函数

这三种方法是等价的，从任何一个都能推导出其它两个。

即使用同一类表示方法，也存在着许多不同的等价的算法结构，滤波器的工程实现要用计算机的硬件或软件。

不同的算法要求不同的实现结构，也会影响系统的某些实际性能。

滤波器实现中需要考虑许多问题，如：
(1) 计算的效率，即完成整个滤波所需要的乘法和加法次数;
(2) 需要的存储量;
(3) 滤波器系数的量化影响;
(4) 运算中的舍入和截断误差、饱和和溢出;

不同的算法在满足上述的要求方面是有很大差别的；先前一个经验：研究DFT的FFT算法，同样的DFT运算，改变计算结构对提高计算效率和节省存储量有多大作用。

方框图与信号流图：没有本质区别，后者是前者的简化表示；用节点和支路（有向线段）代替方框图中的三种基本单元（相加器、数乘器、延时单元）。

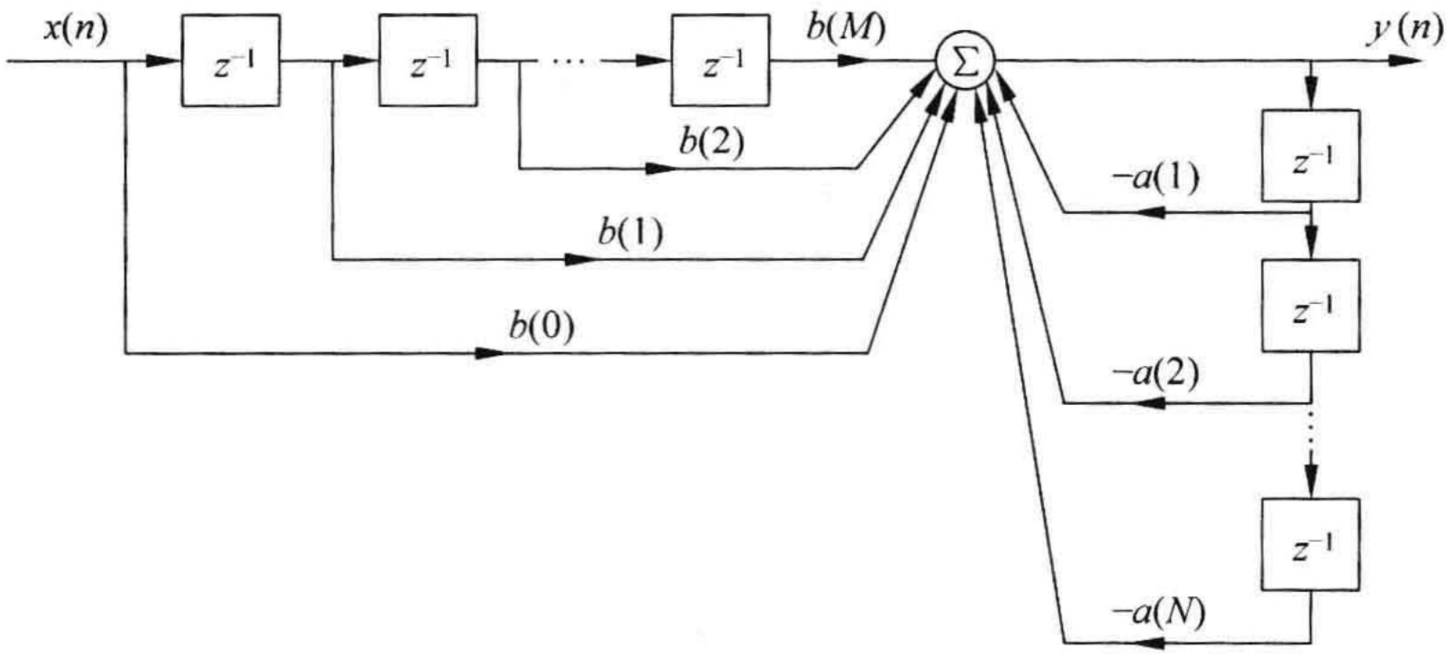
一、IIR 系统的信号流图与结构

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{r=0}^{M} b_r x(n-r)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{r=0}^{M} b_r z^{-r}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

观察：实现本系统，需要一个加法器，$N+M$个乘法器，$N+M$个延迟器。

1. IIR系统直接I型实现的信号流图



方框图

若将上图作一改造，可大量节约延迟器

$$Y(z)=\frac{\sum_{r=0}^{M} b_r z^{-r}}{1+\sum_{k=1}^{N} a_k z^{-k}} X(z)$$

$$\boxed{W(z)=\frac{X(z)}{1+\sum_{k=1}^{N} a_k z^{-k}}}$$

则：

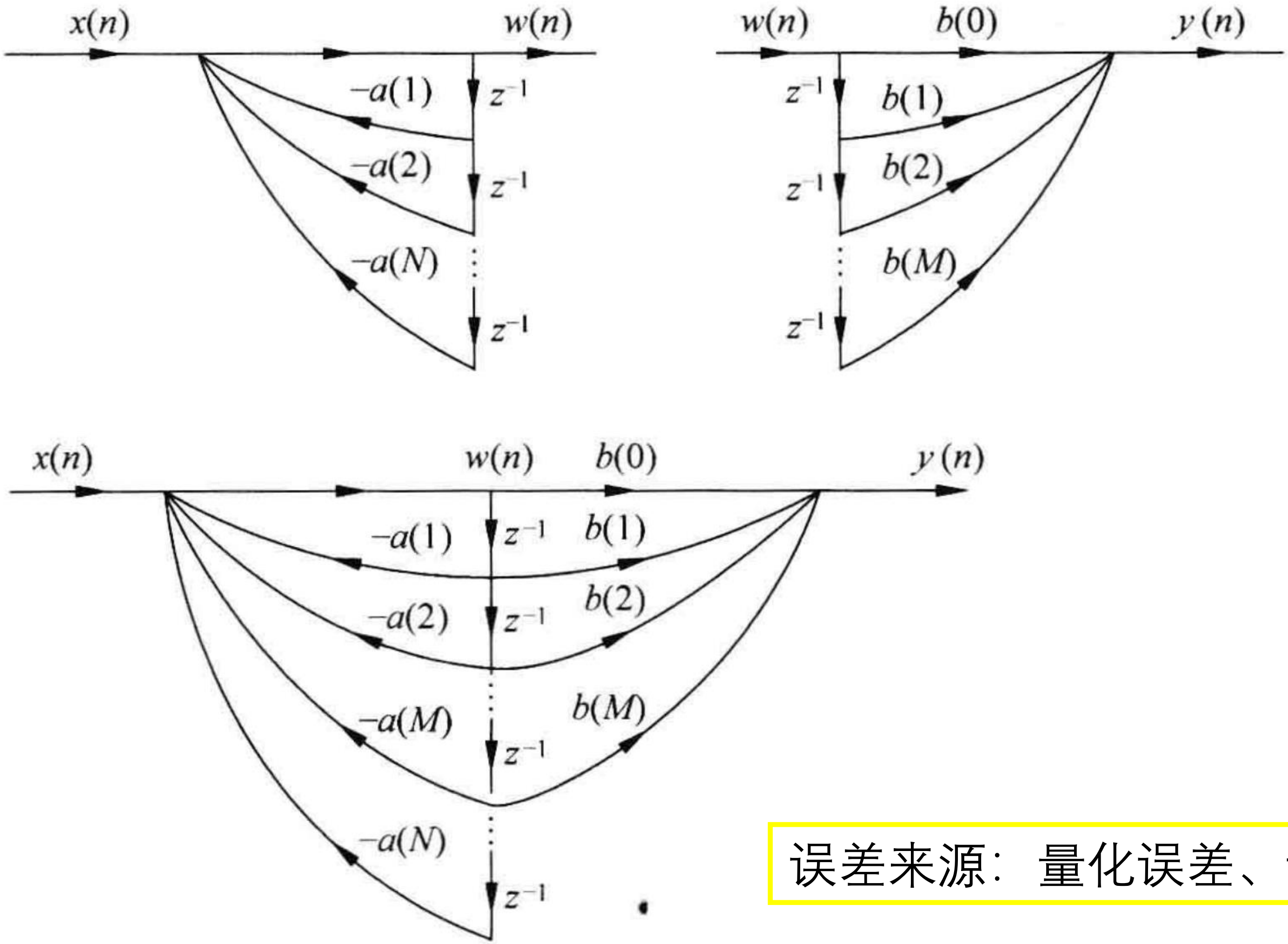
$$Y(z)=W(z)\sum_{r=0}^{M} b_r z^{-r}$$

及

$$w(n)=-\sum_{k=1}^{N} a_k w(n-k)+x(n)$$

$$y(n)=\sum_{r=0}^{M} b_r w(n-r)$$

2. 直接II型实现



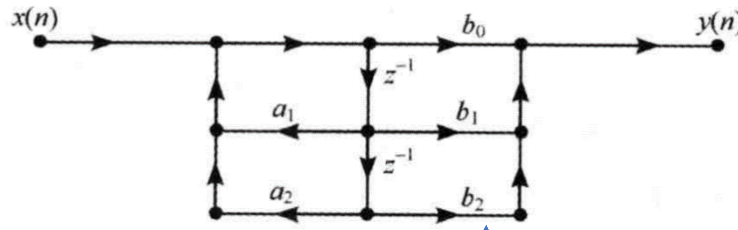
信号流图

优点：比直接I型少用了M个延时单元。假定$N>M$。

缺点：系数对滤波器性能的控制作用不明显；极点对系数变化较灵敏，在有限精度下，容易出现不稳定或产生较大误差；累积误差较大。

误差来源：量化误差、计算误差、存储。

3. 级联实现



$$H(z)=\frac{\sum_{r=0}^{M} b_r z^{-r}}{1+\sum_{k=1}^{N} a_k z^{-k}}=\prod_{k=1}^{N/2} H_k(z)$$

双二阶环节

$$H_k(z)=\frac{1+\beta_{k,1}z^{-1}+\beta_{k,2}z^{-2}}{1+a_{k,1}z^{-1}+a_{k,2}z^{-2}},\quad k=1,\cdots,\frac{N}{2}$$

$$x(n)\Rightarrow \boxed{H_1(z)} \Rightarrow\cdots\Rightarrow \boxed{H_{N/2}(z)} \Rightarrow y(n)$$

$$y(n)=(\cdots((x*h_1)*h_2)*\cdots*h_{N/2})$$

优点：滤波器的第k对极点、零点可以单独调整，因此便于实现滤波器的极零点，便于调整滤波器的频率响应的性能。与直接型相比，累积误差较小。

缺点：各级联子系统的次序、相互间的配合存在优化问题（级联子系统间要考虑信号的放大、或缩小，要考虑防溢出、信噪比）。

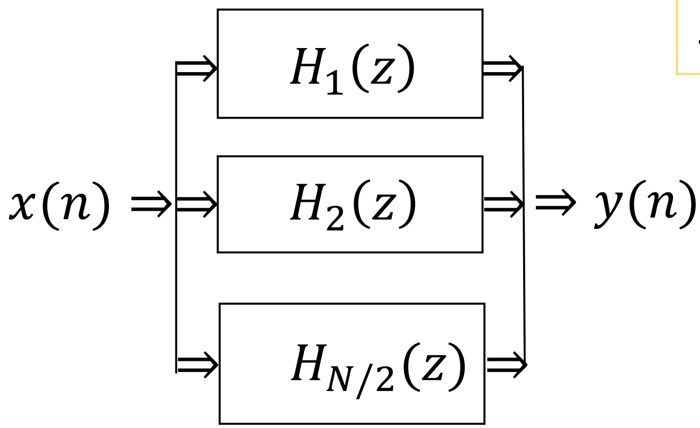
系统函数$H(z)$的分子分母分别因式分解成二阶子系统。每个子系统都以直接形式实现，整个系统函数由二阶环节的级联实现。

4. 并联实现

$$H(z)=\sum_{k=1}^{N/2} H_k(z)$$

优点：每一个子系统都是独立的、是对误差最不敏感的形式；可实现并行计算。

缺点：不能像级联型那样能单独调整零点的位置。



$$y(n)=x(n)*h_1(n)+x(n)*h_2(n)+\cdots+x(n)*h_{N/2}(n)$$

把系统函数$H(z)$用部分分式展开，合并共轭项，成为实系数二阶子系统的和。整个系统函数以子系统的并联网络实现。

在数字信号处理中，由于表示“数”的字长总是有限的，这就必然带来误差。对一个离散系统，这些误差主要包括如下3个方面：

- 模拟信号抽样时的量化误差，相当于引人一个误差序列$e(n)$；$e(n)$在系统中传递，最后出现在输出端；
- 系统的系数也要量化，量化就必然产生误差，该误差一定会影响系统的性能；
- 系统中加、减和乘法运算将产生舍入误差。

思考：直接实现、级联实现和并联实现，那一种实现方式对上述误差最不敏感？

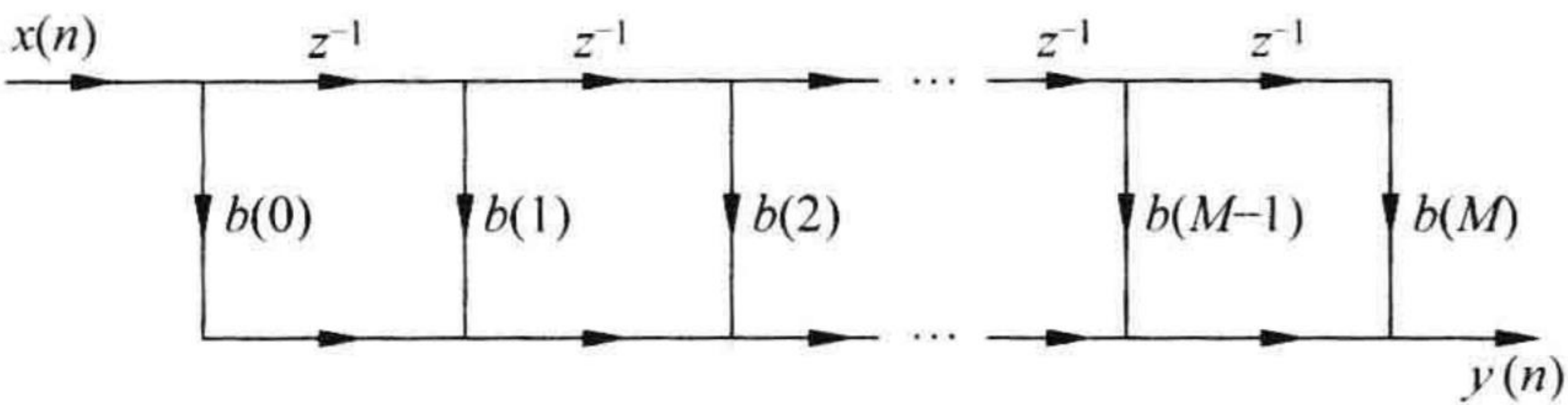
二、FIR 系统的结构

1. 直接实现和级联实现

直接实现

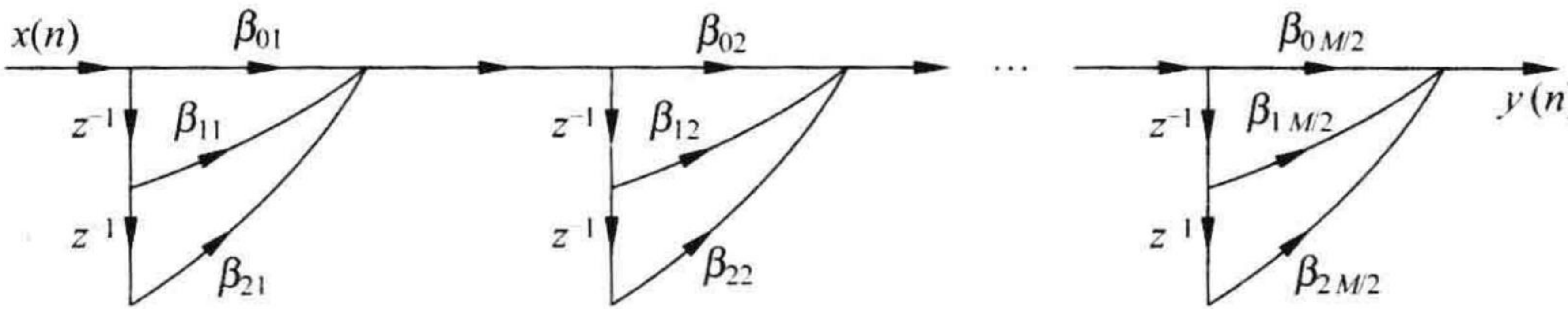
$$Y(z) = X(z)[b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_M z^{-M}]$$

$$H(z) = \sum_{n=0}^{M} b_n z^{-n}$$



级联实现

$$H(z)=\prod_{k=1}^{L}(\beta_{0k}+\beta_{1k}z^{-1}+\beta_{2k}z^{-2})$$

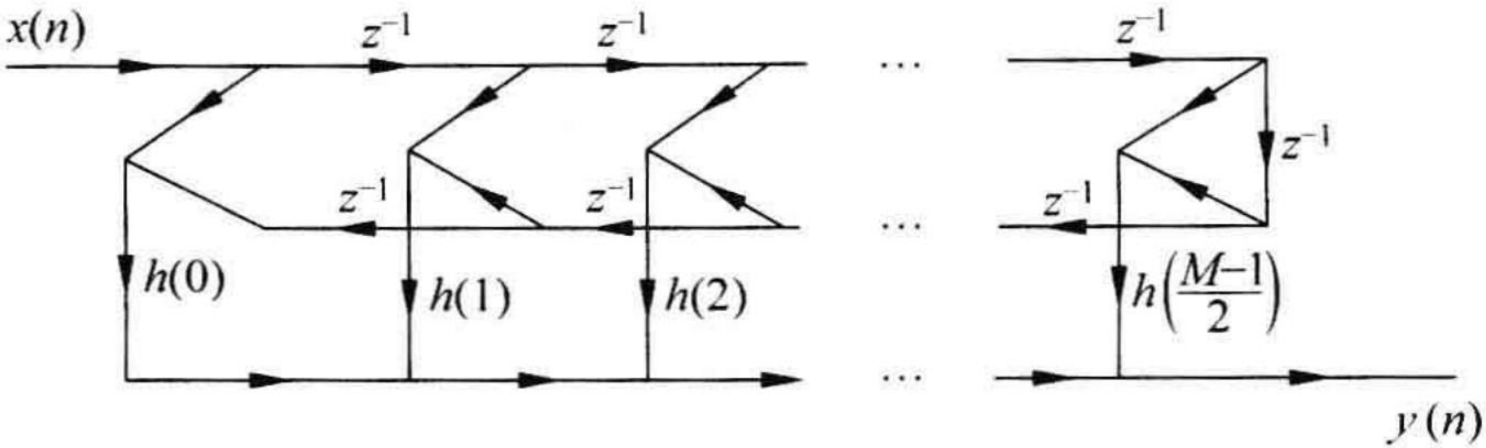


特点：每一节控制一对零点，但所需乘法次数比直接型多。

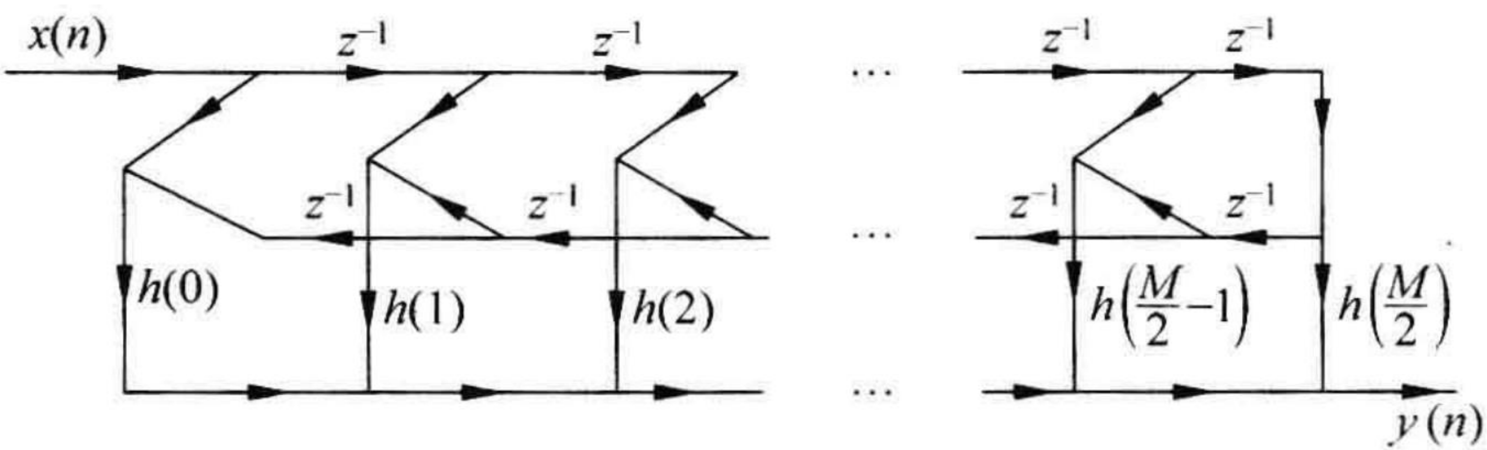
2. 具有线性相位的FIR系统的结构

$$h(n) = h(N-1-n)$$

乘法量? 减少一半



M: odd



M: even

该结构的优点：比直接形式少用$M/2$个乘法。系数和乘法都减少一半。

该结构理论的推导：系数具有对称性，可以是对称结构，也可以是反对称结构。

用该类型滤波器时，求解系统输出的程序设计上，取操作数的特点？

线性相位结构在本质上仍然是直接形式，它只是缩减了乘法计算量。因此，传递函数多项式的组成形式上，线性相位结构等于直接形式。

3. 频率抽样实现

$$H(z)=\sum_{n=0}^{N-1}h(n)z^{-n},\quad H(k)=\sum_{n=0}^{N-1}h(n){W_N}^{nk}$$

思路：用DFT系数$H(k)$表示系统函数$H(z)$

$$H(z)=\sum_{n=0}^{N-1}[\frac{1}{N}\sum_{k=0}^{N-1}H(k){W_N}^{-nk}]z^{-n}$$
$$=\frac{1}{N}\sum_{k=0}^{N-1}H(k)\frac{1-z^{-N}}{1-e^{\mathrm{j}\frac{2\pi}{N}k}z^{-1}}$$
$$=\frac{1-z^{-N}}{N}\sum_{k=0}^{N-1}\frac{H(k)}{1-e^{\mathrm{j}\frac{2\pi}{N}k}z^{-1}}$$

$$H(z)=\frac{1-z^{-N}}{N}\sum_{k=0}^{N-1}\frac{H(k)}{1-e^{\mathrm{j}\frac{2\pi}{N}k}z^{-1}}$$

令:

$$H_1(z)=\frac{1-z^{-N}}{N}$$

梳状滤波器

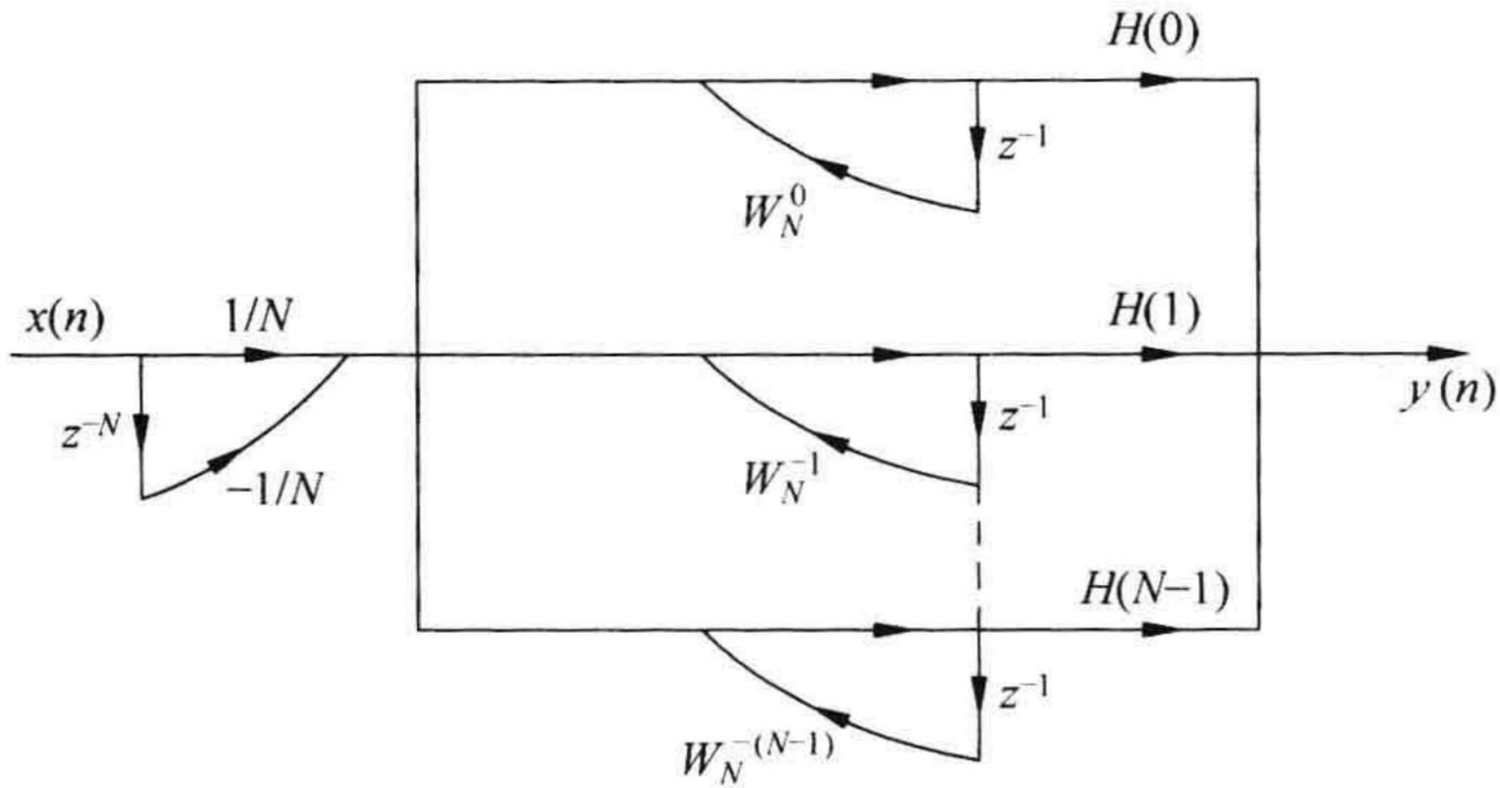
$$H_{2,k}(z)=\frac{H(k)}{1-e^{\mathrm{j}\frac{2\pi}{N}k}z^{-1}}$$

N个一阶IIR系统

则:

$$H(z)=H_1(z)\sum_{k=0}^{N-1}H_{2,k}(z)$$

可按上述级联方式得到系统的信号流图:



该结构一方面反映了 Z 变换、DTFT、DFT之间的关系，另一方面，给出了FIR 滤波器设计的一种有效方法。

讨论：

每一条从输入到输出的支路，有多少个零点?

任意两条不同的支路里，有几个不同的零点?

整个系统有多少个不同的零点?

优点：该结构的系数就是滤波器在$\omega=2\pi k/N$处的响应，因此可以方便地控制滤波器的频率响应；对于窄带滤波器，大部分频率采样值为0，并联的一阶网络的个数?

缺点：结构中的系数多为复数，增加了乘法次数和存储量；极点在单位圆上时，由W因子决定，其量化将引起极点移动，从而将导致系统不稳定。

4. FIR系统的递归实现及梳状滤波器

$$h(n)=\begin{cases}\dfrac{1}{N}, & n=0,1,\ldots,N-1\\ 0, & \text{其他}\end{cases}$$

FIR 系统

$$y(n)=\sum_{k=0}^{N-1}x(n-k)h(k)=\frac{1}{N}\sum_{k=0}^{N-1}x(n-k)$$

该系统实际上是一个N点平均器。

$$H(z)=\frac{1}{N}\sum_{n=0}^{N-1}z^{-n}=\frac{1}{N}\frac{1-z^{-N}}{1-z^{-1}}$$

IIR系统 

$$H(z)=\frac{1}{N}\frac{1-z^{-N}}{1-z^{-1}}$$

IIR 实现

令 $H_1(z)=\frac{1-z^{-N}}{N}$, $H_2(z)=\frac{1}{1-z^{-1}}$

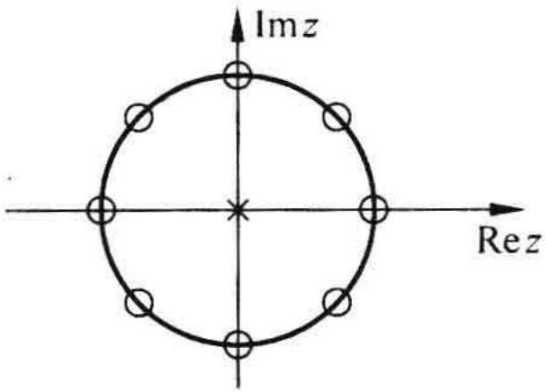
该系统可由一FIR系统和一个一阶IIR系统级联而成，极－零点抵消后，仍是一FIR系统。

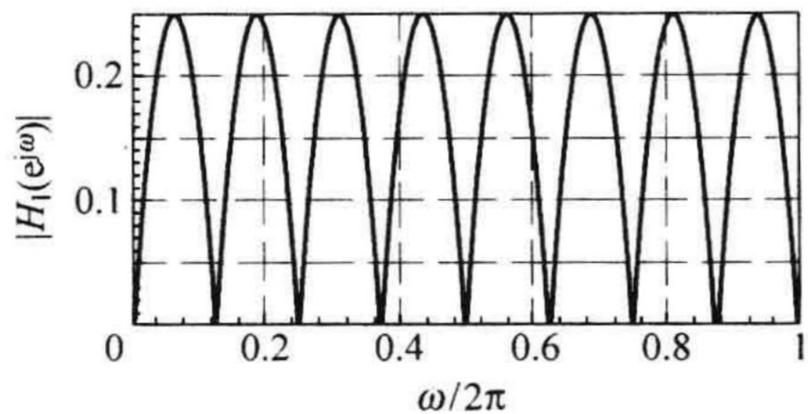
$$H_1(z)=\frac{1}{N}\prod_{k=0}^{N-1}(1-e^{j\frac{2\pi}{N}k}z^{-1}),$$

$$H_1(e^{j\omega})=j2e^{-j\omega N/2}\frac{\sin(\omega N/2)}{N}$$

$$H_1(z) = \frac{1}{N}\prod_{k=0}^{N-1}\left(1 - e^{j\frac{2\pi}{N}k} z^{-1}\right),$$

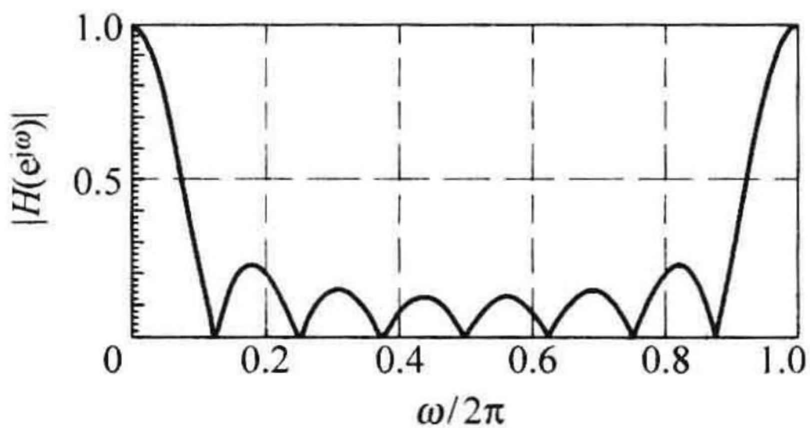
$$H_1(e^{j\omega}) = j2e^{-j\omega N/2}\frac{\sin(\omega N/2)}{N}$$





梳状滤波器

*N*点平均器



Matlab函数：tf2sos

$$H(z)=\frac{B(z)}{A(z)}=\frac{b_0+b_1z^{-1}+\cdots+b_mz^{-m}}{a_0+a_1z^{-1}+\cdots+a_mz^{-m}}$$

Convert digital filter transfer function data to second-order sections form.

$$\text{sos}=\begin{bmatrix} b_{01} & b_{11}z^{-1} & b_{21}z^{-1} & 1 & a_{11}z^{-1} & a_{21}z^{-1} \\ b_{02} & b_{12}z^{-1} & b_{22}z^{-1} & 1 & a_{12}z^{-1} & a_{22}z^{-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{0L} & b_{1L}z^{-1} & b_{2L}z^{-1} & 1 & a_{1L}z^{-1} & a_{2L}z^{-1} \end{bmatrix}$$

Syntax

[sos,g] = tf2sos(b,a)

[sos,g] = tf2sos(b,a,'*order*')

[sos,g] = tf2sos(b,a,'*order*','*scale*')

sos = tf2sos(...)

$$H(z)=g\prod_{k=1}^{L}H_k(z)=g\prod_{k=1}^{L}\frac{b_{0k}+b_{1k}z^{-1}+b_{2k}z^{-2}}{1+a_{1k}z^{-1}+a_{2k}z^{-2}}$$

Description

tf2sos converts a transfer function representation of a given digital filter to an equivalent second-order section representation.

[sos,g] = tf2sos(b,a) finds a matrix sos in second-order section form with gain g that is equivalent to the digital filter represented by transfer function coefficient vectors a and b.

Matlab函数：sosfilt

sosfilt

Second-order (biquadratic) IIR digital filtering

Syntax

y = sosfilt(sos,x)

y = sosfilt(sos,x,dim)

Description

y= sosfilt(sos,x) applies the second-order section digital filter sos to the input signal x.

y = sosfilt(sos,x,dim) operates along the dimension dim.