通信系统仿真
第2章 蒙特卡罗方法04

何晨光
哈尔滨工业大学
电子与信息工程学院

Communication Research Center

6 半解析方法

通过的上面的学习，我们发现蒙特卡罗的方法是完全通用的，而且只要能够用数值算法（数字信号处理，DSP）来定义或者至少是近似模拟系统各构建模块的仿真模型，就可以采用这种方法。

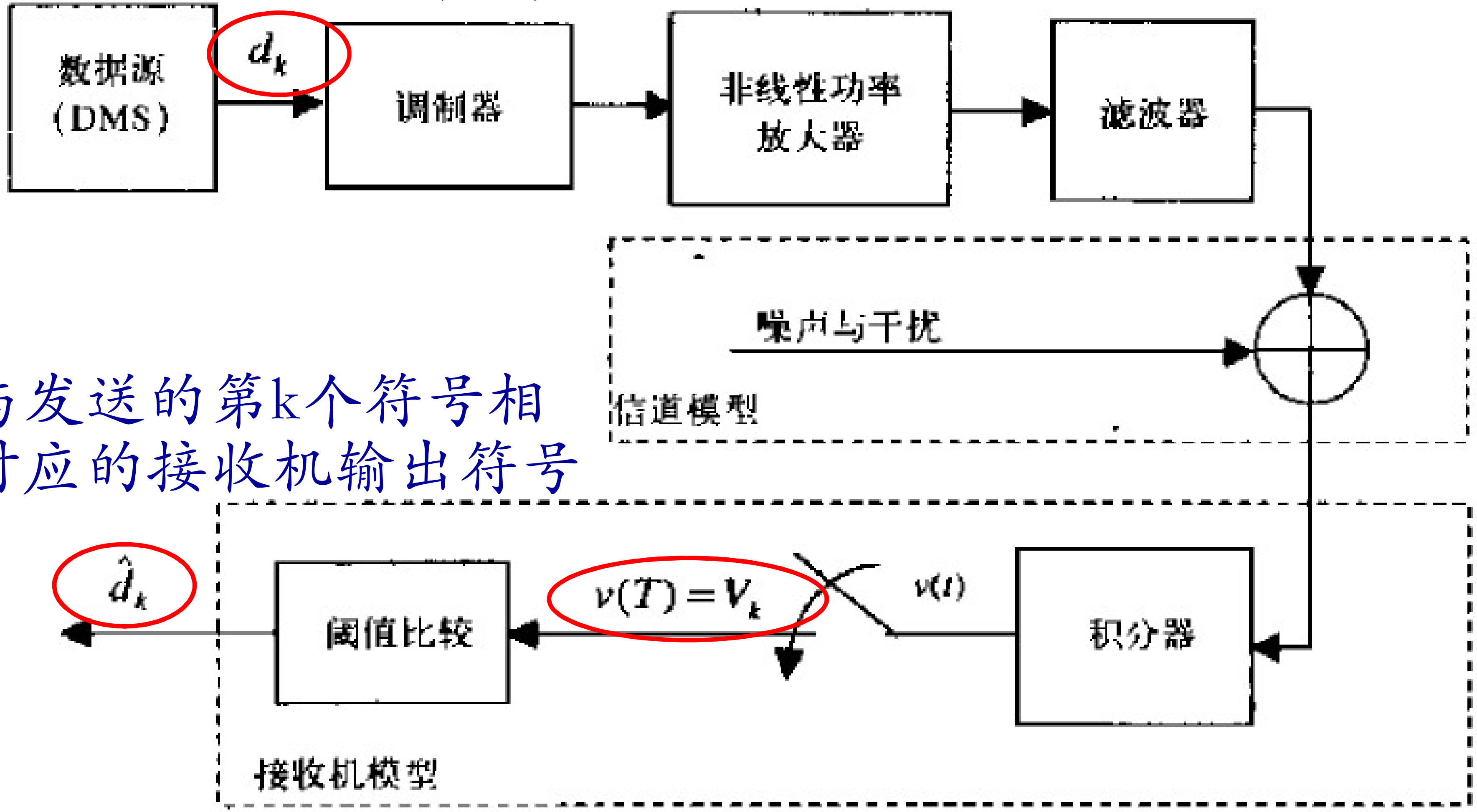
但是，应用蒙特卡罗方法所付出的代价是仿真所需要的运行时间长。如果系统和信道的模型非常复杂，误比特率又比较低，所需要的运行时间就会很长，导致蒙特卡罗方法在除最重要的仿真之外的几乎所有应用场合中都不实用。

6 半解析方法

半解析方法可以代替单纯的蒙特卡罗方法，将仿真和分析以一定的方式结合使用，从而可以迅速地得到误比特率的估计。

因为是快速仿真方法，半解析仿真方法可以在分析知识的运用程度和仿真运行时间之间做出折中。

发送的第k个符号



与发送的第k个符号相对应的接收机输出符号

V_k是关于第k个发送符号的判决统计量，接收机通过比较V_k值和阈值T来做出判决

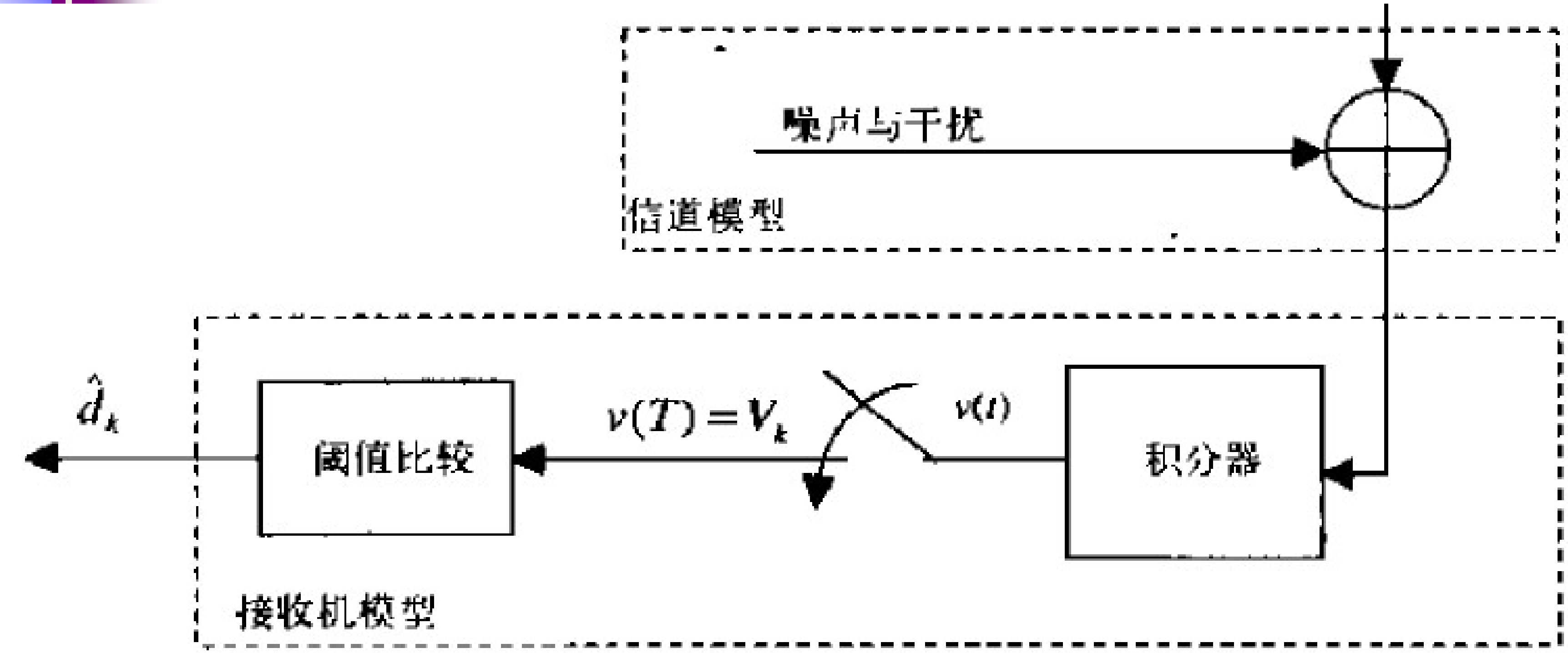
6 半解析方法

判决统计量V_k是三个分量的函数:

$$V_k = f(S_k, D_k, N_k)$$

S_k是由发射信号引起的分量;
D_k是由来自系统因素产生的失真，如由于滤波器或多径引起的ISI;
N_k是由信道扰动（如噪声和干扰）引起的。

通过蒙特卡罗仿真可以确定S_k和D_k，表示噪声影响的N_k分量可以用解析的方法处理。只要V_k中噪声分量的概率密度函数可以用解析的方法确定，就可以对系统使用半解析的仿真方法。



信道是加性高斯白噪声信道，那么从噪声注入点到判决统计量V_k的那一点之间的系统是线性的。这是由于高斯随机过程的任何线性变换还是高斯过程。所以当信道噪声是高斯时，判决统计量V_k也是高斯随机变量，其均值由S_k和D_k确定。

6 半解析方法 基本原理

以AWGN信道环境下的BPSK系统为例

假设系统具有全响应信号，暂时不考虑发送滤波器。（全响应信号就是信号的能量完全限制在一个符号周期内的系统。这样，通过在一个符号周期内对接收信号进行积分，匹配滤波器接收器就可以得到全部的发送符号能量）

给定发送的二进制是0或1的条件下，V_k是高斯的。所以在$d_k=0$和$d_k=1$的条件下，判决统计量V_k的概率密度函数为

$$f_v(v|d_k=0)=\frac{1}{\sqrt{2\pi}\sigma_v}\exp[-\frac{(v-v_1)^2}{2\sigma_v^2}]$$

$$f_v(v|d_k=1)=\frac{1}{\sqrt{2\pi}\sigma_v}\exp[-\frac{(v-v_2)^2}{2\sigma_v^2}]$$

6 半解析方法 基本原理

$$f_v(v|d_k=0)=\frac{1}{\sqrt{2\pi}\sigma_v}\exp[-\frac{(v-v_1)^2}{2\sigma_v^2}]$$

$$f_v(v|d_k=1)=\frac{1}{\sqrt{2\pi}\sigma_v}\exp[-\frac{(v-v_2)^2}{2\sigma_v^2}]$$

v1和v2分别是d_k=0和d_k=1的条件下，随机变量V_k的均值。在给定d_k=0的条件下差错概率为

$$\Pr[Error|d_k=0]=\int_T^{\infty} f_V(v|d_k=0)dv$$

其中T是判决阈值。在给定d_k=1的条件下差错概率为

$$\Pr[Error|d_k=1]=\int_{-\infty}^{T} f_V(v|d_k=1)dv$$

6 半解析方法 基本原理

如果dk=0和dk=1以等概率发送，则最阈值T位于两个条件概率密度函数相同的点上。此时，两个条件差错概率相同，总的差错概率为

也就是

$$P_E = \frac{1}{2}\Pr[Error|d_k = 0] + \frac{1}{2}\Pr[Error|d_k = 1]$$

$$P_E = \Pr[Error|d_k = 0] = \int_T^{\infty} f_V(v|d_k = 0)dv$$

在AWGN环境下，概率通常用高斯Q函数来表示。于是有

$$P_E = Q(\frac{v_1}{\sigma_v})$$

其中

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} \exp[-\frac{t^2}{2}]dt$$

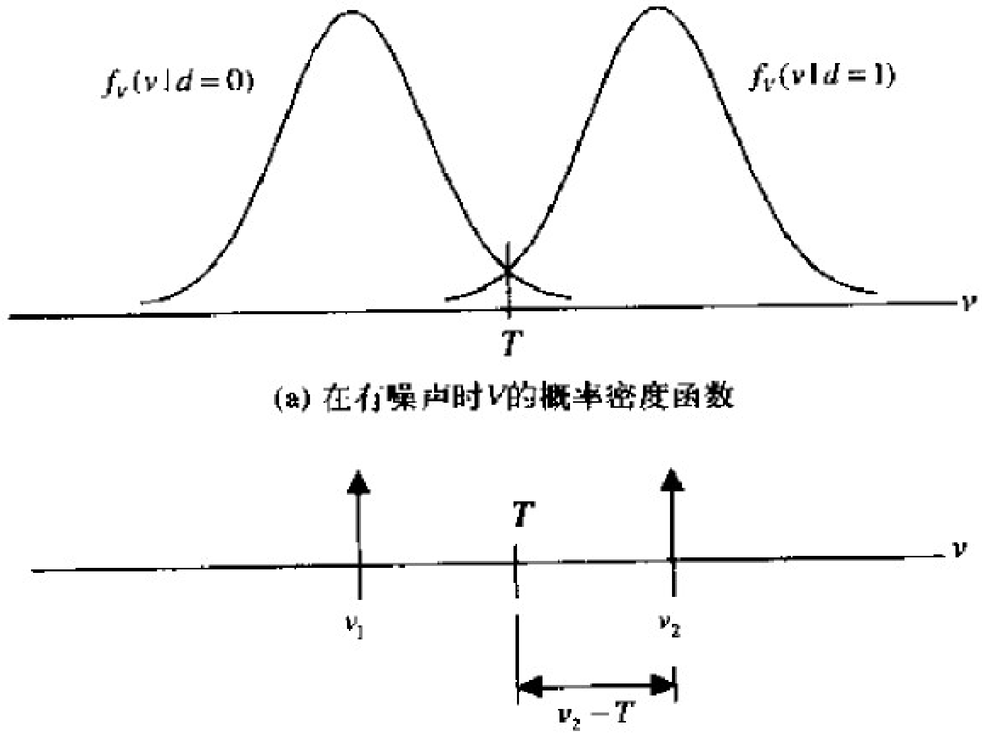
6 半解析方法 基本原理

注意在信号能量相等的情况下，阈值T为零，所以v2=-v1。因此确定了v1和σ_n ，就完全确定了系统的误比特率。

为了确定差错概率，我们只要开发仿真程序来估计v1和 σ_n即可，不需要使用蒙特卡罗方法对差错发生的次数进行计数。

6 半解析方法 基本原理

通过一个无噪声的仿真来确定v1值。如果去掉信道的噪声，下图（a）中的两个pdf曲线将退化成冲激函数（ $\sigma_n = 0$ ），如图（b）所示。每个冲激函数都有单位面积，冲激函数的位置确定了v1和v2。



(a) 在有噪声时V的概率密度函数

(b) 在无噪声时V的概率密度函数

6 半解析方法 基本原理

简单的仿真噪声通过的那部分系统，就可以确定σ_n值。

考虑系统使用积分-清楚符号检测器来建模的接收机。假设这部分系统具有传递函数H(f)，假设输入到匹配滤波器接收机中的白噪声具有双边带功率谱密度N0/2，随机变量V_k的方差为

$$\sigma_n^2 = \frac{N_0}{2}\int_{-\infty}^{\infty}\left|H(f)\right|^2 df = N_0\int_0^{\infty}\left|H(f)\right|^2 df$$

其中噪声带宽定义为

$$B_N = \int_0^{\infty}\left|H(f)\right|^2 df$$

这也是接收机的等价噪声带宽。于是有差错概率为

$$P_E = Q\left(\frac{v_1}{\sqrt{N_0 B_N}}\right)$$

6 半解析方法 基本原理

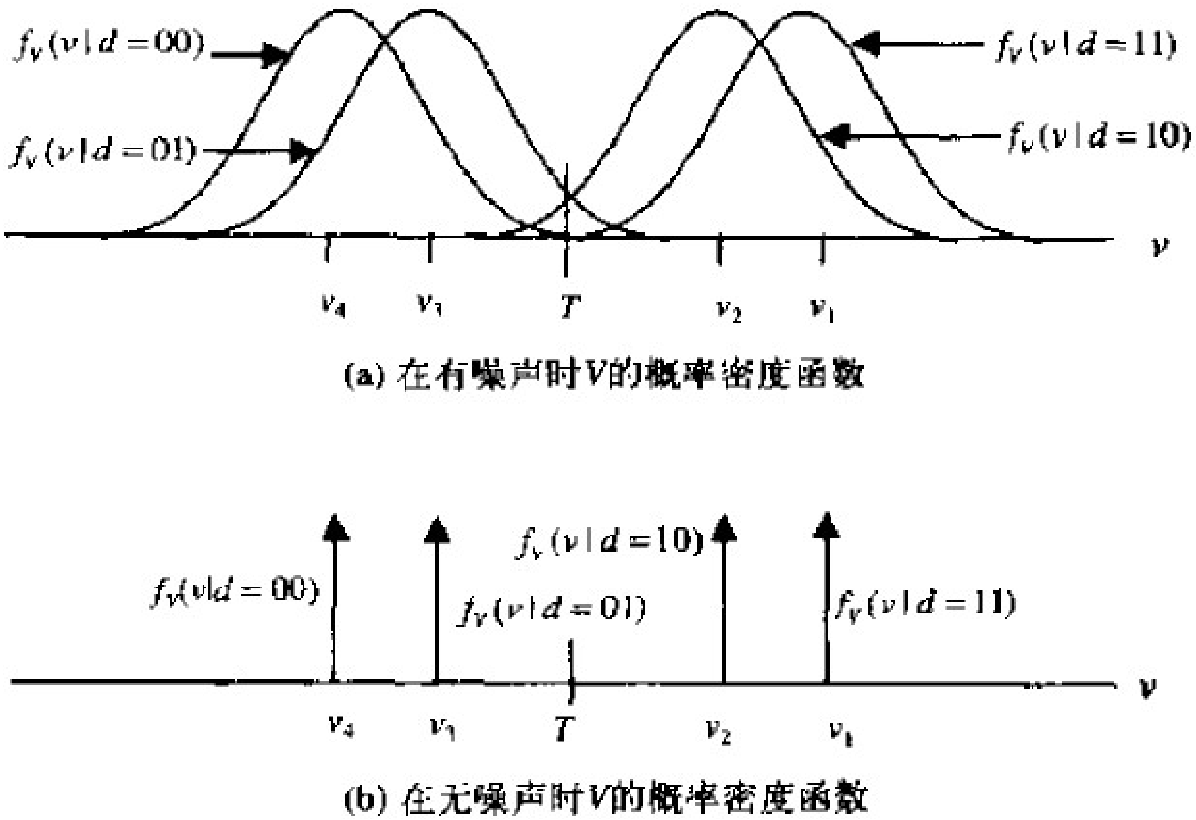
截至目前，我们是在假设没有发射滤波器的情况下。

那么尽管系统是加性高斯白噪声的，非线性放大器仍可能会影响系统的性能，因为非线性放大器会影响发送信号的形状，而这又会影响v1的值。

下面考虑有发射滤波器的情况。这个滤波器的影响是，在时间上将所发送符号的能量扩展到发送符号周期之外，从而引起码间干扰。

6 半解析方法 基本原理

如果滤波器的记忆长度是两个符号周期，则与发送符号相对应的差错概率不仅依赖于当前的符号，而且和前面发送的一个符号有关。因此在计算差错概率时将牵涉到四个概率密度函数，如下图所示。



6 半解析方法 基本原理

如果滤波器的记忆长度是两个符号周期，则与发送符号相对应的差错概率不仅依赖于当前的符号，而且和前面发送的一个符号有关。因此在计算差错概率时将牵涉到四个概率密度函数，如下图所示。

和前面的方法相同，执行无噪声的仿真可以得出v1,v2,v3和v4。系统的差错概率为

$$P_E = \frac{1}{2}\sum_{i=1}^{4} Q(\frac{v_i}{\sigma_v})$$

那么，我们可以很容易的将结果扩展到记忆长度为M个符号的情况。

6 半解析方法 等效噪声源

判决统计量Vk是三个分量的函数：

$$V_k = f(S_k, D_k, N_k)$$

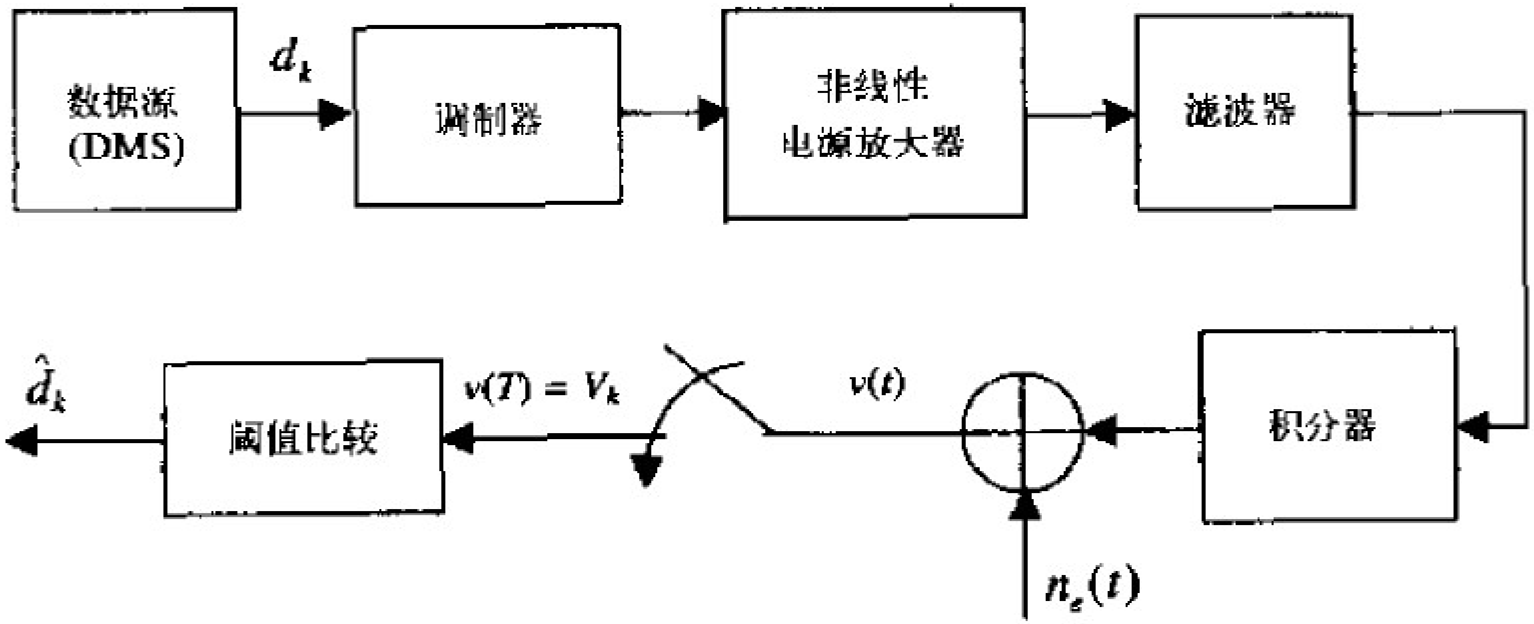
如果执行无噪声的仿真，则所得的统计量只是Sk和Dk的函数。在此统计量上需要加上一个方差由

$$\sigma_n^2 = \frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2\,df = N_0\int_0^{\infty}|H(f)|^2\,df$$

推导出的随机变量Nk。

6 半解析方法 等效噪声源

随机变量Nk视作来自下图的等价噪声源ne(t)的一个样本。该噪声源包含了反映在积分-清除检测器的积分器输出中的，有噪声、干扰和其它信道损伤所产生的总影响。



6 半解析方法 等效噪声源

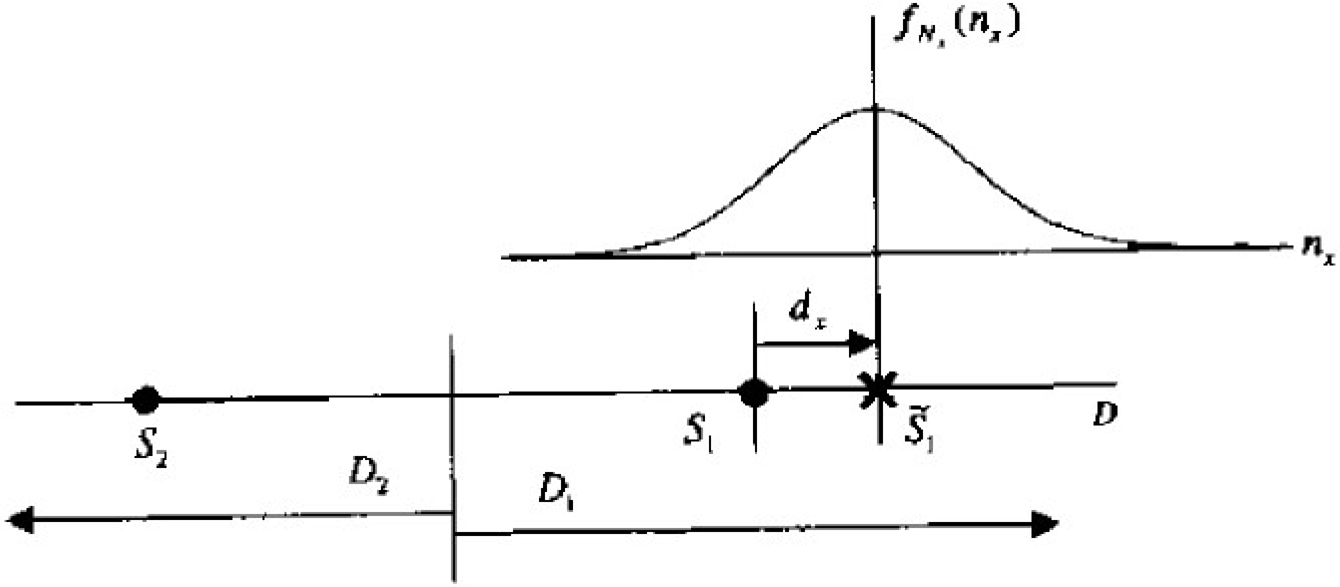
随机变量N_k视作来自下图的等价噪声源$n_e(t)$的一个样本。该噪声源包含了反映在积分-清除检测器的积分器输出中的，有噪声、干扰和其它信道损伤所产生的总影响。

如果信道是白噪声，则可用冲激响应，或等价地用下式所定义的传递函数，来把信道噪声变换到积分器的输出中。

$$\sigma_n^2 = \frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2\,df = N_0\int_0^{\infty}|H(f)|^2\,df$$

例8 PSK系统误比特率半解析估计方法

如下图所示的信号星座。



我们假设发送的是S1而接收到的是 $\tilde{S}_1$ 。由于存在符号间干扰、非线性等因素，S1与 $\tilde{S}_1$ 会不相同。那么在发送S1的条件下，条件差错概率为

$$\Pr[Error|S_1]=\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma_n}\exp[-\frac{(n-\tilde{S}_1)^2}{2\sigma_n^2}]dn$$

$$\Pr[Error|S_1] = \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_n} \exp[-\frac{(n-\tilde{S}_1)^2}{2\sigma_n^2}]dn$$

$$\Pr[Error|S_1] = Q(\frac{\tilde{S}_1}{\sigma_n})$$

由蒙特克罗仿真求上面Q函数中的两个参数，就可以确定条件误比特率。在确定方差时，可由仿真的冲激响应h(n)求出B_N值。

假设S_k是N比特长的仿真序列中的第k个发送比特。对于每一个k值，$1 \le k \le N$ ，S_k是S_1或者S_2。条件误比特率为

$$\Pr[Error|S_k] = Q(\frac{\tilde{S}_k}{\sigma_n})$$

通过对整个N比特序列做平均，所得到的总体误比特率为

$$P_E = \frac{1}{N}\sum_{k=1}^{N} Q(\frac{\tilde{S}_k}{\sigma_n})$$

例8 PSK系统误比特率半解析估计方法

主函数

```
NN = 256;                                  % number of symbols
tb = 1;                                    % bit file
p0 = 1;                                    % power
fs = 16;                                   % samples/symbol
ebn0db = [0:1:8];                          % Eb/No vector in dB
[bt,at] = butter(5,2/fs);                  % transmitter filter parameters
x = random_binary(NN,fs);                  % establish PSK signal
y1 = x;                                    % save signal
y2a = y1*sqrt(p0);                         % scale amplitude
y2 = filter(bt,at,y2a);                    % transmitter output
br = ones(1,fs); br = br/fs; ar = 1;       % matched filter parameters
y = filter(br,ar,y2);                      % matched filter output
```

（代码见c219.m）

例8 PSK系统误比特率半解析估计方法

buffer

butter函数是求Butterworth数字滤波器的系数，在求出系数后对信号进行滤波时用filter函数。

设计滤波器就是设计滤波器系数[B,A]。
[B,A] = BUTTER(N,Wn,'high') ---用来设计高通滤波器
[B,A] = BUTTER(N,Wn,'low') --低通滤波器
[B,A] = BUTTER(N,Wn)--带通滤波器

N是滤波器的阶数，整数
Wn的确定跟采样频率Fs有关

例8 PSK系统误比特率半解析估计方法

buffer

对于原始信号x，采样频率Fs=1000Hz，设计一个8阶、通带为100-200Hz的带通滤波器：

```
[b,a]=butter(8,[0.2 0.4])
[b,a]= butter(8,[100/(1000/2) 200/(1000/2) ])
```

例8 PSK系统误比特率半解析估计方法

```
a = [1 2];
b = [2 3];
x = [1 2 3 4 5 6];
y = filter(b, a, x)
```

得： y =2 3 6 5 12 3

下面给出具体的计算过程如下：

```
a(1)y(1) = b(1)x(1); %可以求出y(1)
a(1)y(2) = b(1)x(2)+b(2)x(1)-a(2)y(1); %可以由y(1)求出y(2)
a(1)y(3) = b(1)x(3)+b(2)x(2)-a(2)y(2); %可以由y(2)求出y(3)
a(1)y(4) = b(1)x(4)+b(2)x(3)-a(2)y(3); %可以由y(3)求出y(4)
a(1)y(5) = b(1)x(5)+b(2)x(4)-a(2)y(4); %可以由y(4)求出y(5)
a(1)y(6) = b(1)x(6)+b(2)x(5)-a(2)y(5); %可以由y(5)求出y(6)
```

例8 PSK系统误比特率半解析估计方法

```
%
[cor lags] = vxcorr(x,y);                 % compute crosscorrelation
[cmax nmax] = max(abs(cor));              % maximum of crosscorrelation
timelag = lags(nmax);                     % lag at max crosscorrelation
theta = angle(cor(nmax));                 % determine angle
y = y*exp(-i*theta);                      % derotate
%
% Noise BW calibration.
%
hh = impz(br,ar);                         % receiver impulse response
nbw = (fs/2)*sum(hh.^2);                  % noise bandwidth
```

$$B_N = \int_0^{\infty} |H(f)|^2 \, df$$

（代码见c219.m）

例8 PSK系统误比特率半解析估计方法

```
index = (10*fs+8:fs:(NN-10)*fs+8);
xx = x(index);
yy = y(index-timelag+1);
eb = tb*sum(abs(y2).^2)/length(y2);
eb = eb/2;
[peideal,pesystem] = psk_berest(xx,yy,ebn0db,eb,tb,nbw);
semilogy(ebn0db,pesystem,'ro-',ebn0db,peideal); grid;
xlabel('E_b/N_0 (dB)'); ylabel('Bit Error Rate')
legend('System Under Study','AWGN Reference',0)
% End of script file.
```

（代码见c219.m）

例8 PSK系统误比特率半解析估计方法

BER子函数

```
function [peideal,pesystem] = psk_berest(xx,yy,ebn0db,eb,tb,nbw)
```

输出：

peideal 理想Pe

pesystem 系统Pe

输入：

xx 参考（理想）输入信号

yy 滤波器的输出

ebn0db 信噪比

eb bit能量

tb 比特宽度

nbw 等效噪声带宽

（代码见psk_berest.m）

例8 PSK系统误比特率半解析估计方法

```
% For comparision purposes, set the noise BW of the ideal
% receiver (integrate and dump) to be equal to rs/2.
%
nbwideal = 1/(2*tb);
for m=1:length(ebn0db)
    peideal(m) = 0.0; pesystem(m) = 0.0;          % initialize
    %
    % Find n0 and the variance of the noise.
    %
    ebn0(m) = 10^(ebn0db(m)/10);                  % dB to linear
    n0 = eb/ebn0(m);                              % noise power
    sigma = sqrt(n0*nbw*2);                       % variance
    sigma1 = sqrt(n0*nbwideal*2);                 % variance of ideal
```

理想的等效噪声带宽等于1/2采样频率

$$B_N=\int_0^{\infty}|H(f)|^2\,df \qquad \sigma_n^2=\frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2\,df=N_0\int_0^{\infty}|H(f)|^2\,df$$

例8 PSK系统误比特率半解析估计方法

对于理想信号来说，需要乘以一个平均因子，使理想的输入 和 在接收端的输入具有相同的能量

```
%%
nx = length(xx);
    b = sqrt(2*eb/tb)/sqrt(sum(abs(xx).^2)/nx);
    d1 = b*abs(xx);
    d3 = abs(yy);
    peideal(m) = sum(q(d1/sigma1));
    pesystem(m) = sum(q(d3/sigma));
end
peideal = peideal/nx;
pesystem = pesystem/nx;
% End of function file.
```

$$P_E = \frac{1}{N}\sum_{k=1}^{N} Q\left(\frac{S_k}{\sigma_n}\right)$$

（代码见psk_berest.m）

例8 PSK系统误比特率半解析估计方法

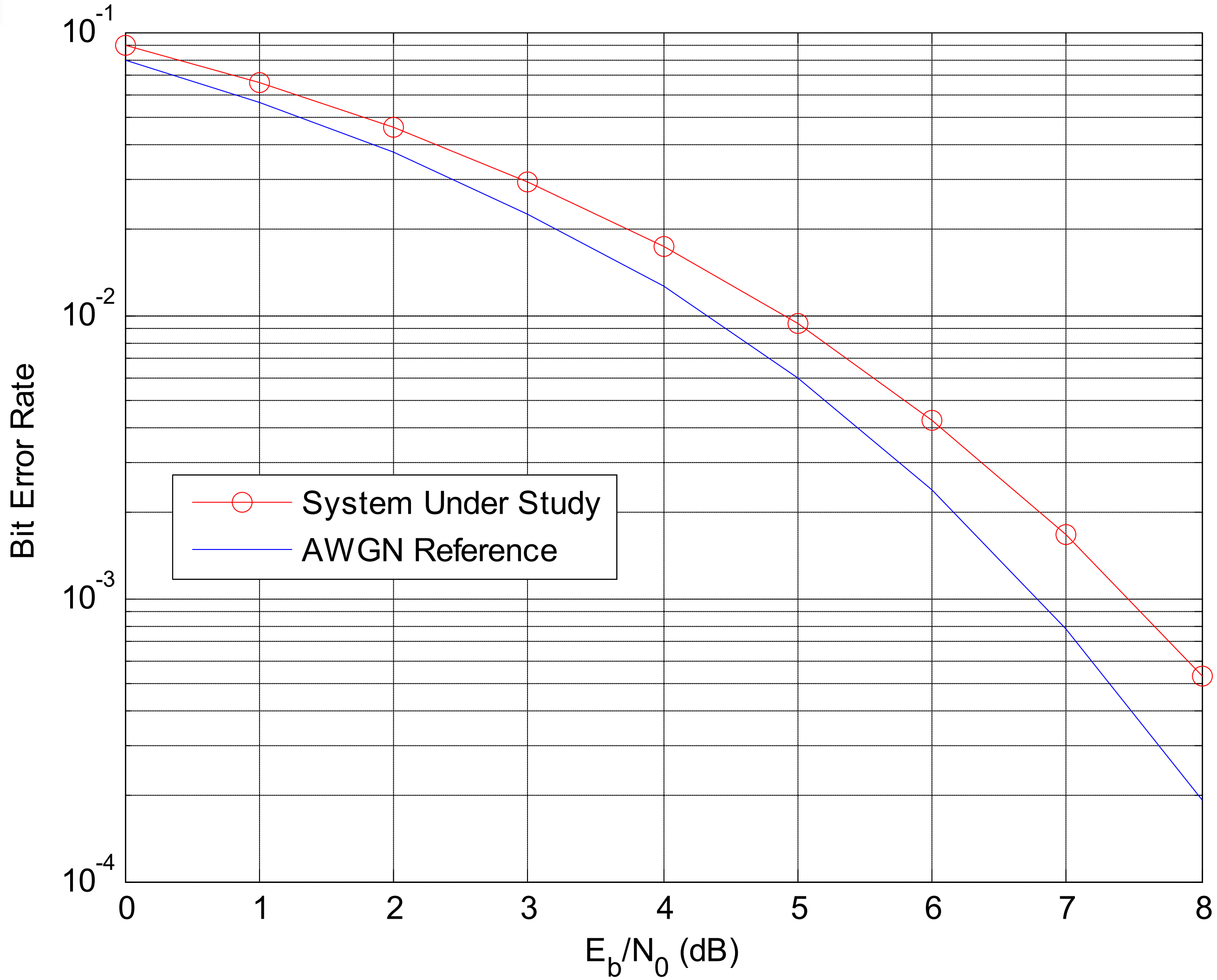
Q函数

```
function y=q(x)
y = 0.5*erfc(x/sqrt(2));
% End function file.
```

$$Q(x)=\frac{1}{\sqrt{2\pi}}\int_x^{\infty}\exp[-\frac{t^2}{2}]dt$$

（代码见 q.m）

例8 PSK系统误比特率半解析估计方法



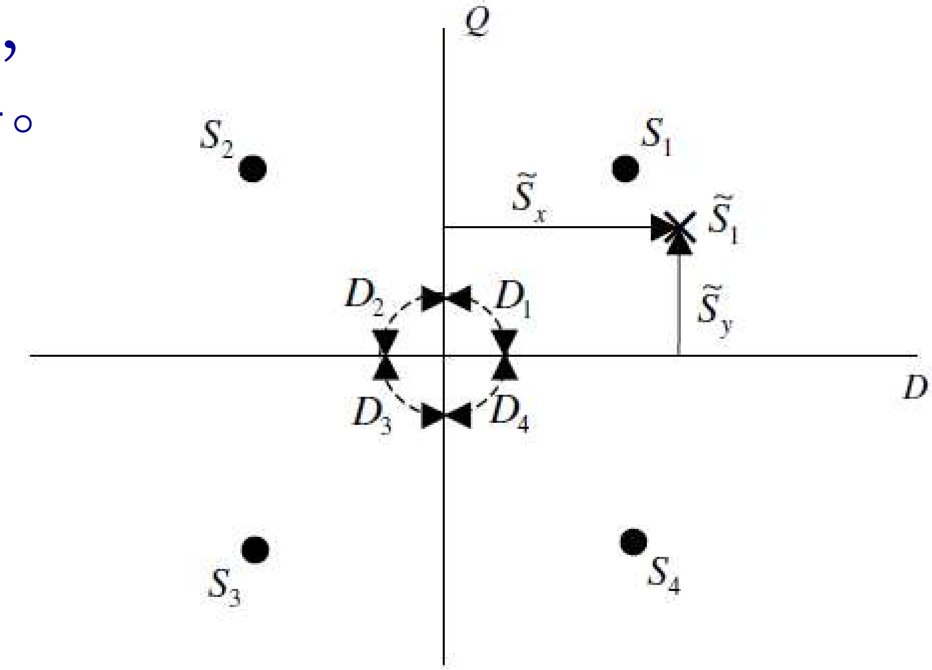
例9 QPSK系统误比特率半解析估计方法

信号空间中QPSK点是用符号而不是比特来定义的。所以我们利用半解析仿真计算的是QPSK系统的符号差错概率，一旦确定了符号差错概率，利用解析的方法就可以把符号差错概率转化为比特差错概率。对于BPSK系统，符号差错概率和比特差错概率是相等的。

由于QPSK的信号星座图有四个点而不是两个，并且信号空间是二维而不是一维，因此QPSK半解析估计器和PSK估计器的不同之处在于前者必须为正交信道增加一维。

例9 QPSK系统误比特率半解析估计方法

如图所示的星座图中，发送的信号点记做S1,S2,S3,S4，相应的有判决D1,D2,D3,D4。



假设发送的是S_1，接收的无噪声信号记做$\tilde{S}_1$，由于存在符号间干扰和失真

$$\tilde{S}_1 \neq S_1$$

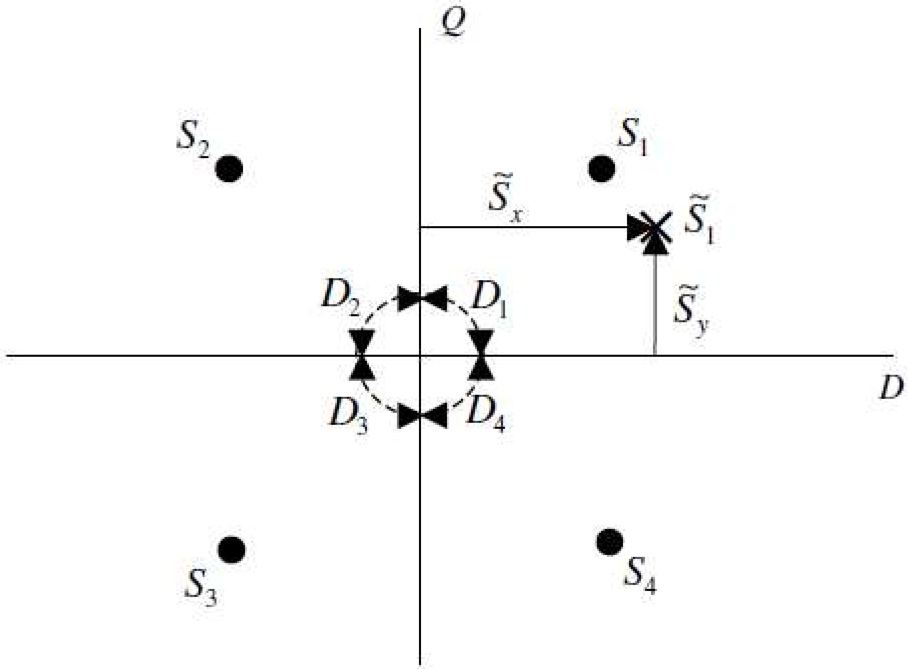
因此仿真考虑了符号间干扰的影响而没有考虑噪声的影响，所以半解析仿真所确定的是 $\tilde{S}_1$ 而不是 S_1 。

例9 QPSK系统误比特率半解析估计方法

$\tilde{S}_1$ 的同相和正交分量分别用 $\tilde{S}_X$ 和 $\tilde{S}_Y$ 表示，其中

$$\tilde{S}_X = \mathrm{Re}(\tilde{S}_1)$$

$$\tilde{S}_Y = \mathrm{Im}(\tilde{S}_1)$$



当考虑噪声时，分别在 $\tilde{S}_X$ 和 $\tilde{S}_Y$ 上加入n_x和n_y。此时，在发送S_1的条件下

如果 $(\tilde{S}_X + n_X, \tilde{S}_Y + n_Y) \in D_1$，则作出了正确的判决；

如果 $(\tilde{S}_X + n_X, \tilde{S}_Y + n_Y) \notin D_1$，则发生了差错。由于是半解析仿真估计器，噪声的影响用解析方法做了处理，所以没有出现在图中。

例9 QPSK系统误比特率半解析估计方法

给定信号空间中接收到的（无噪声）信号是 $\mathring{S}_1$ ，现在要确定会导致差错的噪声分量n_x和n_y。

这个问题很类似与前面PSK系统的例子，主要的差别在于现在所考虑的是二维信号空间而不是一维。

我们假设同相和正交噪声分量不相关并且是联合高斯分布的，因此在发送S_1接收 $\mathring{S}_1$ 情况下，发生差错的条件概率密度是

$$\Pr[Error|S_1]=\iint_{(\overset{+}{S}_X+n_X,\overset{+}{S}_Y+n_Y)\notin D_1}\frac{1}{\sqrt{2\pi}\sigma_n\sigma_n}\exp[-\frac{(n_x-\overset{+}{S}_x)^2}{2\sigma_n^2}-\frac{(n_y-\overset{+}{S}_y)^2}{2\sigma_n^2}]dn_xdn_y$$

例9 QPSK系统误比特率半解析估计方法

$$\Pr[Error|S_1]=\iint_{(\hat{S}_X+n_X,\hat{S}_Y+n_Y)\notin D_1}\frac{1}{\sqrt{2\pi\sigma_n\sigma_n}}\exp[-\frac{(n_x-\hat{S}_x)^2}{2\sigma_n^2}-\frac{(n_y-\hat{S}_y)^2}{2\sigma_n^2}]dn_x dn_y$$

简化：

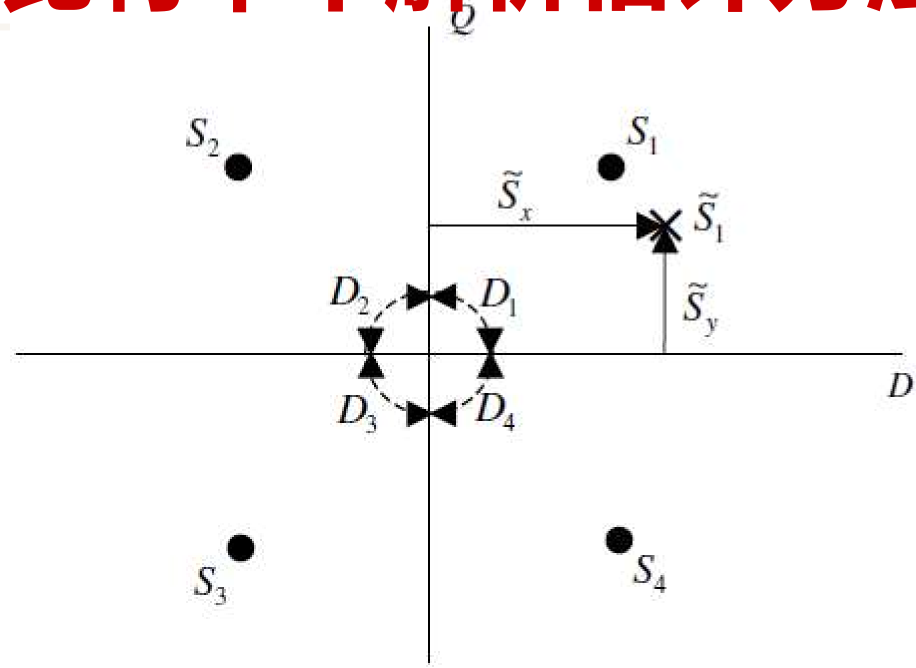
$$f_{n_X}(n_X|\hat{S}_x,\sigma_n)=\frac{1}{\sqrt{2\pi\sigma_n}}\exp[-\frac{(n_x-\hat{S}_x)^2}{2\sigma_n^2}]$$

$$f_{n_Y}(n_Y|\hat{S}_y,\sigma_n)=\frac{1}{\sqrt{2\pi\sigma_n}}\exp[-\frac{(n_Y-\hat{S}_Y)^2}{2\sigma_n^2}]$$

可得：

$$\Pr[Error|S_1]=\iint_{(\hat{S}_X+n_X,\hat{S}_Y+n_Y)\notin D_1}f_{n_X}(n_X|\hat{S}_x,\sigma_n)f_{n_Y}(n_Y|\hat{S}_y,\sigma_n)dn_x dn_y$$

例9 QPSK系统误比特率半解析估计方法



$$\Pr[Error|S_1] = \iint_{(\tilde{S}_X+n_X,\tilde{S}_Y+n_Y)\notin D_1} f_{n_X}(n_X|\tilde{S}_x,\sigma_n) f_{n_Y}(n_Y|\tilde{S}_y,\sigma_n)\, dn_x dn_y$$

该式的上界可以表示为：

$$\Pr[Error|S_1] < \iint_{(\tilde{S}_X+n_X,\tilde{S}_Y+n_Y)\in(D_2\cup D_3)} f_{n_X}(n_X|\tilde{S}_x,\sigma_n) f_{n_Y}(n_Y|\tilde{S}_y,\sigma_n)\, dn_x dn_y + \iint_{(\tilde{S}_X+n_X,\tilde{S}_Y+n_Y)\in(D_3\cup D_4)} f_{n_X}(n_X|\tilde{S}_x,\sigma_n) f_{n_Y}(n_Y|\tilde{S}_y,\sigma_n)\, dn_x dn_y$$

例9 QPSK系统误比特率半解析估计方法

$$\Pr[Error|S_1] < \iint_{(\tilde{s}_X+n_X,\tilde{s}_Y+n_Y)\in(D_2\cup D_3)} f_{n_X}(n_X|\tilde{s}_x,\sigma_n)f_{n_Y}(n_Y|\tilde{s}_y,\sigma_n)dn_x dn_y +$$

$$\iint_{(\tilde{s}_X+n_X,\tilde{s}_Y+n_Y)\in(D_3\cup D_4)} f_{n_X}(n_X|\tilde{s}_x,\sigma_n)f_{n_Y}(n_Y|\tilde{s}_y,\sigma_n)dn_x dn_y$$

因为判决区间D3在上式中出现了两次，所以这个公式给出的是上界。根据判决区域的定义可以得到

$$\Pr[Error|S_1] < \int_{-\infty}^{0} f_{n_X}(n_X|\tilde{s}_x,\sigma_n)dn_X \int_{-\infty}^{\infty} f_{n_Y}(n_Y|\tilde{s}_y,\sigma_n)dn_Y +$$

$$\int_{-\infty}^{\infty} f_{n_X}(n_X|\tilde{s}_x,\sigma_n)dn_X \int_{-\infty}^{0} f_{n_Y}(n_Y|\tilde{s}_y,\sigma_n)dn_Y$$

可以看出四个积分中有两个为1，所以

$$\Pr[Error|S_1] < \int_{-\infty}^{0} f_{n_X}(n_X|\tilde{s}_x,\sigma_n)dn_X + \int_{-\infty}^{0} f_{n_Y}(n_Y|\tilde{s}_y,\sigma_n)dn_Y$$

例9 QPSK系统误比特率半解析估计方法

$$\Pr[Error|S_1] < \int_{-\infty}^{0} f_{n_X}(n_X|\tilde{S}_x,\sigma_n)dn_X + \int_{-\infty}^{0} f_{n_Y}(n_Y|\tilde{S}_y,\sigma_n)dn_Y$$

利用

$$f_{n_X}(n_X|\tilde{S}_x,\sigma_n) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp[-\frac{(n_x-\tilde{S}_x)^2}{2\sigma_n^2}]$$

$$f_{n_Y}(n_Y|\tilde{S}_y,\sigma_n) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp[-\frac{(n_Y-\tilde{S}_Y)^2}{2\sigma_n^2}]$$

可以得到

$$\Pr[Error|S_1] < Q(\frac{\mathrm{Re}[\tilde{S}_1]}{\sigma_n}) + Q(\frac{\mathrm{Im}[\tilde{S}_1]}{\sigma_n})$$

例9 QPSK系统误比特率半解析估计方法

$$\Pr[Error|S_1] < Q\left(\frac{\mathrm{Re}[\tilde{S}_1]}{\sigma_n}\right) + Q\left(\frac{\mathrm{Im}[\tilde{S}_1]}{\sigma_n}\right)$$

根据对称性，四种可能发送符号中的任何一种都有相同的条件差错概率。和PSK系统一样，假设S_k是N个仿真符号序列中的第K个发送符号，可得条件误符号率的上界为

$$\Pr[Error|S_k] < Q\left(\frac{\mathrm{Re}[\tilde{S}_k]}{\sigma_n}\right) + Q\left(\frac{\mathrm{Im}[\tilde{S}_k]}{\sigma_n}\right)$$

通过对总的N个符号序列的条件差错概率做平均，得到的总的误符号率为

$$P_S < \frac{1}{N}\sum_{k=1}^{N}\left[Q\left(\frac{\mathrm{Re}[\tilde{S}_k]}{\sigma_n}\right) + Q\left(\frac{\mathrm{Im}[\tilde{S}_k]}{\sigma_n}\right)\right]$$

例9 QPSK系统误比特率半解析估计方法

主函数

（代码见c220.m）

```
NN = 256;                                   % number of symbols
tb = 0.5;                                   % bit time
p0 = 1;                                     % power
fs = 16;                                    % samples/symbol
ebn0db = [0:1:10];                          % Eb/N0 vector
[b,a] = butter(5,1/16);                     % transmitter filter parameters
%
% Establish QPSK signals
%
x = random_binary(NN,fs)+i*random_binary(NN,fs);   % QPSK signal
y1 = x;                                            % save signal
y2a = y1*sqrt(p0);                                 % scale amplitude
%
% Transmitter filter
%
y2 = filter(b,a,y2a);                       % filtered signal
%
% Matched filter
%
b = ones(1,fs); b = b/fs; a = 1;            % matched filter parameters
y = filter(b,a,y2);                         % matched filter output
```

例9 QPSK系统误比特率半解析估计方法

主函数

```
[cor lags] = vxcorr(x,y);
cmax = max(abs(cor));
nmax = find(abs(cor)==cmax);
timelag = lags(nmax);
theta = angle(cor(nmax));
y = y*exp(-i*theta);                    % derotate
%
% Noise BW calibration
%
hh = impz(b,a);                         % receiver impulse response
nbw = (fs/2)*sum(hh.^2);                % noise bandwidth
…
```

（代码见c219.m）

例9 QPSK系统误比特率半解析估计方法

主函数

size(), 获取矩阵的行数和列数

```
index = (10*fs+8:fs:(NN-10)*fs+8);
xx = x(index);
yy = y(index-timelag+1);
[n1 n2] = size(y2); ny2=n1*n2;
eb = tb*sum(sum(abs(y2).^2))/ny2;
eb = eb/2;
[peideal,pesystem] = qpsk_berest(xx,yy,ebn0db,eb,tb,nbw);
subplot(1,2,1)
yscale = 1.5*max(real(yy));
plot(yy,'+')
xlabel('Direct Sample'); ylabel('Quadrature Sample'); grid;
axis([-yscale yscale -yscale yscale])
subplot(1,2,2)
semilogy(ebn0db,peideal,ebn0db,pesystem,'ro-'); grid;
xlabel('E_b/N_0 (dB)'); ylabel('Bit Error Rate')
legend('AWGN Reference','System Under Study')
```

例9 QPSK系统误比特率半解析估计方法

子函数 （代码见qpsk_berest.m）

```
function [peideal,pesystem] = qpsk_berest(xx,yy,ebn0db,eb,tb,nbw)
nbwideal = 1/(2*tb*2);
for m=1:neb
    peideal(m) = 0.0; pesystem(m) = 0.0;          % initialize
    %
    % Find  n0 and the variance of the noise.
    %
    string1 = ['Eb/No = ',num2str(ebn0db(m))];
    disp(string1)                                  % track execution
    ebn0(m) = 10^(ebn0db(m)/10);                   % dB to linear
    n0 = eb/ebn0(m);                               % noise power
    sigma = sqrt(n0*nbw*2);                        % variance
    sigma1 = sqrt(n0*nbwideal*2);                  % variance of ideal
```

$$\sigma_n^2 = \frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2\,df = N_0\int_0^{\infty}|H(f)|^2\,df$$

例9 QPSK系统误比特率半解析估计方法

对于理想信号来说，需要乘以一个平均因子，使理想的输入 和 在接收端的输入具有相同的能量

```
b = sqrt(2*eb/tb)/sqrt(sum(abs(xx).^2)/nx);
for n=1:nx
    theta = angle(xx(n));
    if (theta<0)
        theta = theta+2*pi;
    end
  %
  % Rotate x and y to the first quadrant and compute BER.
  %
    xxx(n) = b*xx(n)*exp(-i*(theta-(pi/4)));
    yyy(n) = yy(n)*exp(-i*(theta-(pi/4)));
```

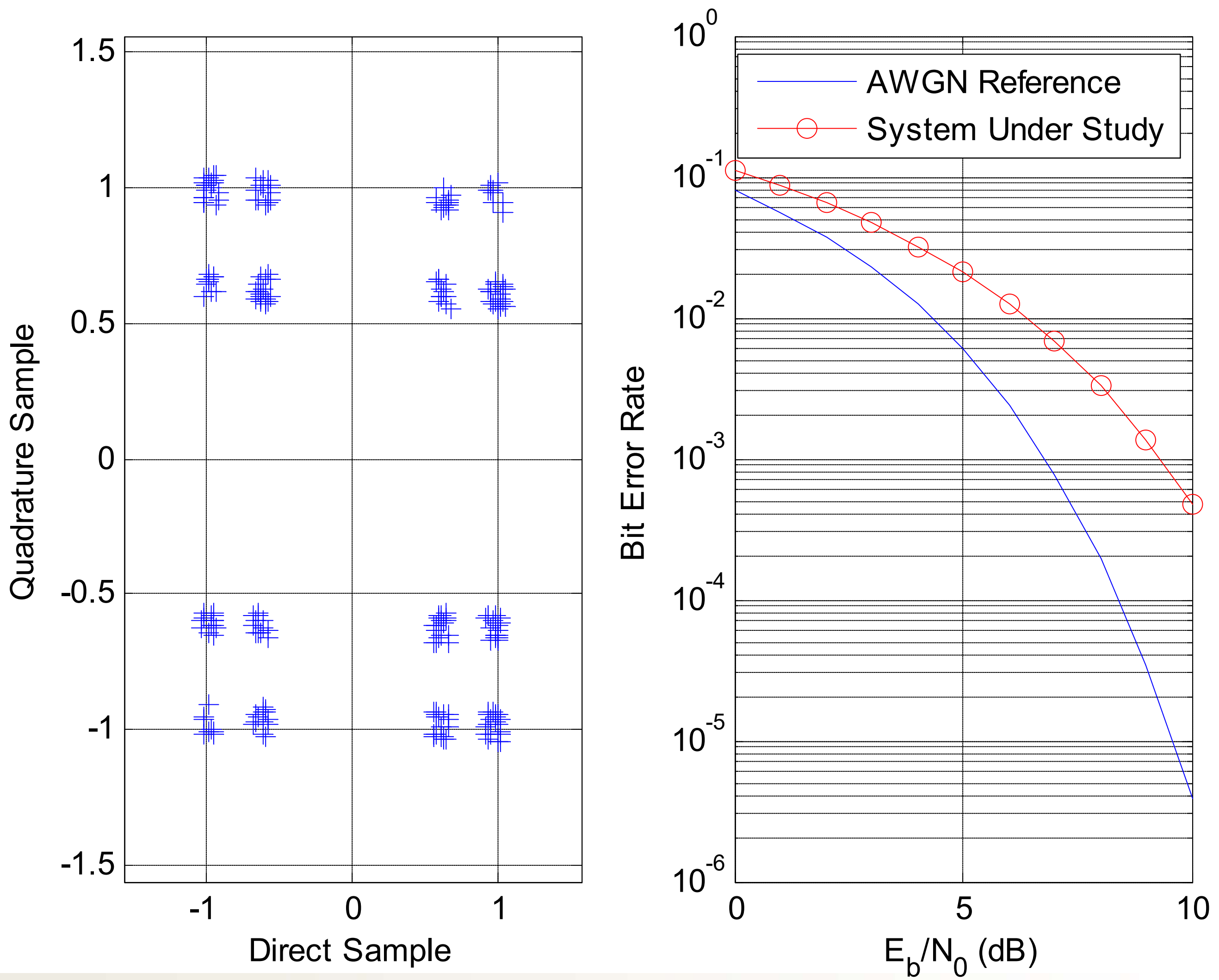
（代码见qpsk_berest.m）

例9 QPSK系统误比特率半解析估计方法

```
        d1 = real(xxx(n)); d2 = imag(xxx(n));        % reference
        d3 = real(yyy(n)); d4 = imag(yyy(n));        % system
        pe1 = q(d1/sigma1)+q(d2/sigma1);             % reference
        pe2 = q(d3/sigma)+q(d4/sigma);               % system
        peideal(m) = peideal(m)+pe1;                 % SER of reference
        pesystem(m) = pesystem(m)+pe2;               % SER of system
    end
end
peideal = (1/2)*peideal./nx;                         % convert to BER
pesystem = (1/2)*pesystem./nx;                       % convert to BER
```

（代码见qpsk_berest.m）

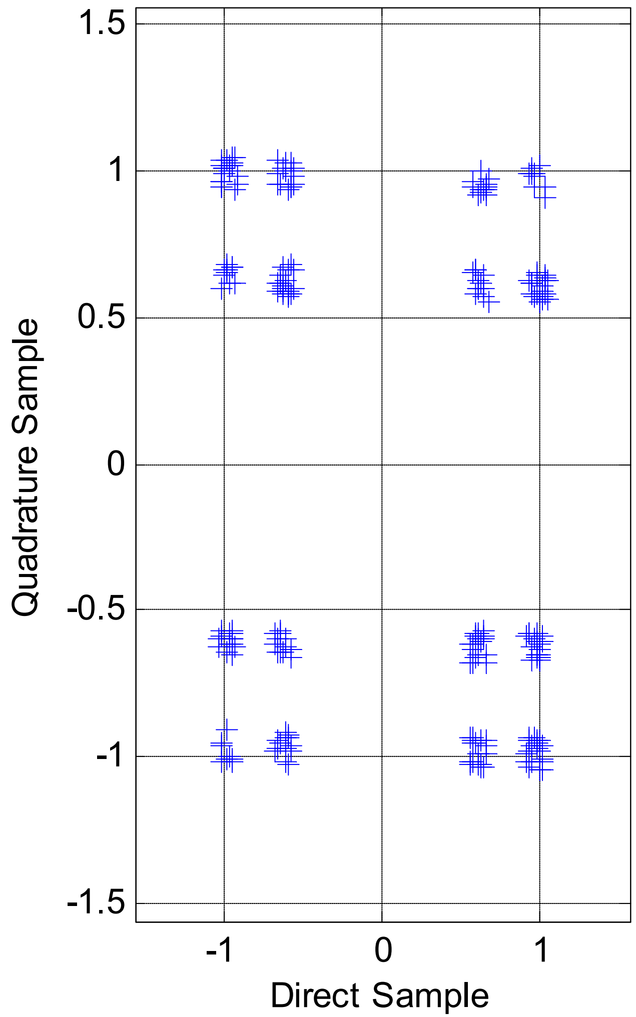
例9 QPSK系统误比特率半解析估计方法



例9 QPSK系统误比特率半解析估计方法

此图是信号星座图。注意此时接收到的信号星座图不再是跟理想QPSK系统一样只有四个点，而是由16个点组成。

假设在第一象限中的信号所代表的数据比特是00，同时假设由于ISI所造成的系统记忆长度为两个符号（当前的和前面发送的符号），因此发送00时将产生四个信号点，分别是0000,0001,0010,0011.其中每个信号点的前两位是当前的符号，后两位是先前发送的符号。



例9 QPSK系统误比特率半解析估计方法

注意第一象限中的四个点中的每一个点都是由稍稍散布着的多个点组成，这些散步是由于系统表现出来的记忆长度超过两个符号所造成的，尽管这些额外的记忆的影响很小。