

# AF 协作通信下误码率的研究

王思薇

(东南大学, 江苏 南京, 210000)

**摘要:** 本文设计了简化的中继放大转发 (AF) 协作通信系统, 并比较协作系统与非协作系统的通信精确度。建立了 Rayleigh 信道模型, 采用 MRC 合并方式实现最大合并比, 设计了基于网络最优路径的判决解码功能, 讨论了中继放大转发协作通信的误码率问题。通过仿真 AF 协作通信过程, 得到一定精度下误码率分析结果。

**关键词:** 放大转发; 协作系统; 误码率; 最大合并比; 判决解码

## Research on Bit Error Rate of Relay Amplified and Forwarded Cooperative Communication

Wang Siwei

(Southeast University, Nanjing, 210000, China)

**Abstract:** This paper designs a simplified relay amplifying and forwarding (AF) cooperative communication system and compares the communication accuracy between the collaborative system and the non-cooperative system. The Rayleigh channel model is established, and the maximum merge ratio is realized by MRC merge method. The decision decoding function based on network optimal path is designed. The error rate of relay amplifying and forwarding cooperative communication is discussed. Through the simulation AF cooperative communication process, the error rate analysis result under certain precision is obtained.

**Key words:** Amplify-and-Forward; Cooperative Communication; Symbol error rate; Maximum combination ratio; Decision decoding

随着用户对无线通信的需求增强, 协作通信将成为未来的发展趋势<sup>[1]</sup>。协作通信的基本思想是在多用户通信环境中, 使用单副天线的各临近移动用户可按照一定方式共享彼此的天线协同发送, 从而产生一种类似多天线发送的虚拟环境, 获得空间分集增益, 提高系统传输性能。协作通信思想通过用户间彼此共享天线, 互为通信中继, 实现虚拟发射分集, 从而为 MIMO 的实用提供了一个可行的思路。协作通信的核心问题是中继节点的协作协议。中继技术解决无线通信覆盖扩展和速率提高两大挑战, 1979 年 Cover 和 Gamal 关于中继信道的信息论特性的研究[2]。分析了一个三节点系统的容量, 并提出该系统可被分解为一个广播信道和一个多址信道。Sendonaris 等人[3]进一步在两个用户协作的场景下分析了信道容量和中断概率等性能[4]。上述工作集中在信息论方向, 在技术应用方面还面临不少挑战。

**作者简介:** 王思薇, (1998-), 女, 本科生, E-mail:

[213160676@seu.edu.cn](mailto:213160676@seu.edu.cn)

根据中继节点对源节点信息处理方式的不同可分为放大转发(AF)、解码转发(DF)、编码协作(CC)、空时编码协作(STCC)和网络编码协作(NCC)等多种方式。

AF 协作方式, 有的文献也称为前向放大、非再生中继。在 AF 协议下, 中继采用模拟处理, 不对接收的信号进行解调和解码, 而是直接将收到的带有噪声的信号进行模拟处理, 然后发送给接收端。AF 作为一种非再生中继, 本质上是一种模拟信号的处理方式。这种方式也最简单, 而且由于目的节点可接收到两路独立的衰落信号, AF 可获得满分集增益, 性能良好。由于中继节点在放大信号的同时也放大了源-中继信道引入的噪声, 因此 AF 方式存在着噪声传播效应。由于复杂度低中继节点处理信号的时间短, AF 协作方式被广泛应用。

本文通过 MATLAB 仿真, 来验证协作系统对通信精确度的改善, 并分析不同信道情况下的非协作系统与协作系统 (AF) 的表现, 研究二者的实际性能与所面临的主要问题。

# 1 系统模型

## 1.1 单中继协作通信系统模型

在中继信道中有 3 个节点:源节点(Source node)、中继节点(Relay node)和目标节点(Destination node),如图 1 所示。中继采用 AF 协作方式。协作通信分为两个阶段:在第一阶段源节点发送广播信息并由中继节点和目标节点接收。在第二阶段中继将第一阶段接收到的信号放大转发给目标节点。

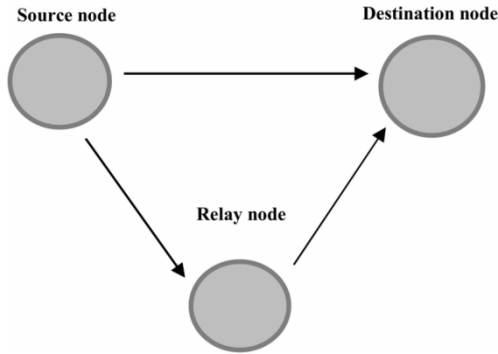


图 1 单中继协作系统模型

源端广播过程中,源节点以广播的方式向周围发送信号 $x_s$  (程序中表示为 BitsTx), 其中一路直接发送到目标节点, 一路发送到中继节点。经过信道后, 中继节点接收到信号 $y_{s,r}$  (程序中表示为 Y12) :

$$y_{s,r} = \sqrt{P_s} h_{s,r} x_s + n_{s,r} \quad (1.1)$$

目标节点 D 接收到信号 $y_{s,d}$  (程序中表示为 Y1d) :

$$y_{s,d} = \sqrt{P_s} h_{s,d} x_s + n_{s,d} \quad (1.2)$$

其中,  $P_s$ 为源发送信号的功率,  $n_{s,r}$ 为源节点与中继节点间信道噪声。

中继放大过程中, 中继节点直接将收到的来自源节点的信号 $y_{s,r}$ 以系数  $\beta$  进行功率放大, 然后转发给目标节点。AF 方式可以看成是具有两个发射端的重复码, 唯一不同的是中继节点将自身接收到的噪声信号也放大并发送到目标节点。目标节点通过合并两路信号, 对源节点的发送信号进行估计。

为保证中继节点功率受限, 放大系数 $\beta$ 应满

足:

$$\beta = \sqrt{\frac{P_s}{P_s |h_{s,r}|^2 + N_0}} \quad (1.3)$$

可见  $\beta$  取决于信道的衰落系数 $h_{s,r}$ 、源发送信号的功率 $P_s$ 和噪声功率 $N_0$ 。那么, 中继放大后的信号 $y_{AF}$  (程序中表示为 BitR2d) :

$$y_{AF} = \beta y_{s,r} \quad (1.4)$$

目标节点接收的来自中继的信号 $y_{r,d}$  (程序中表示为 Y2d) :

$$y_{r,d} = \sqrt{P_r} h_{r,d} y_{AF} + n_{r,d} \quad (1.5)$$

其中  $P_r$ 为中继节点发送信号的功率。

目标节点将接收到的来自源节点的直传信号 $y_{s,d}$ 和来自中继节点的信号 $y_{r,d}$ , 按照 MRC 方案进行合并, 得到信号  $y$  (程序中表示为 Rd) :

$$y = a_1 y_{s,d} + a_2 y_{r,d} \quad (1.6)$$

其中 $a_1$ 、 $a_2$ 分别为目标节点接收到的来自源、来自中继信号的加权系数, 为:

$$a_1 = \frac{\sqrt{P_s} h_{s,d}^*}{N_0} \quad (1.7)$$

$$a_2 = \frac{\beta \sqrt{P_r} h_{s,r}^* h_{r,d}^*}{(\beta^2 |h_{r,d}|^2 + 1) N_0} \quad (1.8)$$

式中,  $P_s$ 和 $P_r$ 分别为协作情况下源节点和中继节点的发送功率,  $h_{s,d}^*$ 、 $h_{s,r}^*$ 与 $h_{r,d}^*$ 分别为源节点与目标节点、源节点与中继节点、中继节点与目标节点之间的瑞利衰落信道系数的复共轭,  $N_0$ 为噪声功率。

## 1.2 简化的 AF 协作通信系统

程序结构流程图如图 2 所示:

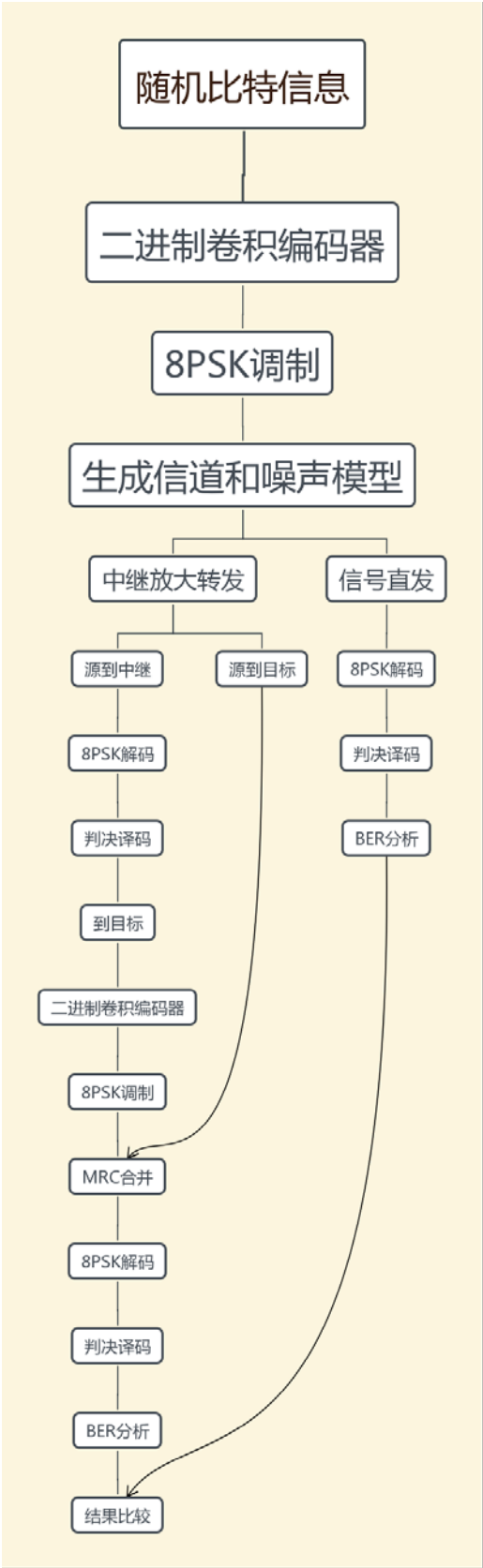


图 2 程序结构流程图

一如上图所示，程序共进行若干次互不相关的演算，以确保数据的真实性。每次演算，会生成一串长度为  $L$  的随机比特信号。这串信号进入一个预先设定好生成矩阵和位数的二进制卷积编码器，以获得随即比特信息的编码信息。

编码信息再经过 8PSK 调制，每个符号对三位比特进行调制。最终讲调制完成的信号送入假定的 Rayleigh 信道。自此可以将处理好的信号以 AF 协作的方式向各个节点传递。一方面需要将信号直发，所采集的数据作为比较组与协作通信获取的信号进行比较；另一方面，调至完成的信号将进行 AF 协作通信。

协作通信中，一部分信号由源节点发送至中继节点，另一部分由源发送到目标节点。中继节点获得的信号，将先进行 8PSK 解调，获得编码信息。编码信息再经过判决译码器进行解码。随后将解码信息发送至目标节点时再进行卷积编码和 8PSK 调制。最终在目标节点，中继节点发送的解码信息和由源节点直接送入目标节点的信息进行 MRC 合并。为了在接收端进行 BER 分析，还需将合并后的信息进行同样的解码和判决译码操作。最终讲目标节点处理好的信息与一开始的随即比特信息进行误码比较，将若干次误码数累加得到最终相对准确的误码率。

非协作通信中，也就是信号直发中，将接收端收到的信号直接进行 8PSK 解码和判决译码。最终将处理好的信号和一开始的随机比特信息进行误码比较，得到非协作通信的误码率。

最终在不同的信噪比之下进行若干次实验，得到了协作通信和非协作通信中，误码率与信噪比之间的关系。将关系曲线绘制成折线图，以便分析。

## 2 算法分析

### 2.1 二进制卷积编码器

卷积码是通信编码中纠错编码的一种，它可以将原来的信息码打乱，这时尽管出现大面积突发性错误，这些可以通过解交织器来进行分散，从而将大面积的错误较为平均地分散到不同的码段，利于信道纠错的实现。卷积编码器即为实现卷积码编码的电路。

卷积码拥有良好的纠错性能，是一种被广泛应用于移动通信的信道编码系统。一个 $(n,k,m)$ 卷积码编码器由  $k$  个输入，具有  $m$  阶存储的  $n$  个输出的线形时序电路实现。通常， $n$  和  $k$  是较小的整数，且看  $k < n$ ，但  $m$  比较大。如图 3 所示。

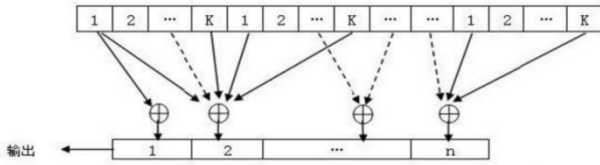


图 3 卷积编码示意图

在本程序中，设定每个时钟周期进入编码器的比特数为  $k_0=2$ ，卷积码的生成矩阵为  $g=[0\ 0\ 1\ 0\ 1\ 0\ 0\ 1; 0\ 0\ 0\ 0\ 0\ 0\ 1; 1\ 0\ 0\ 0\ 0\ 0\ 1]$

每次进行卷积编码之前，程序会检测所设定的生成矩阵的大小是否符合要求：

$$\text{rem}(\text{size}(g, 2), k_0) > 0 \quad (2.1)$$

同时检测是否有额外的零需要补充：

$$\text{rem}(\text{length}(\text{input}), k_0) > 0 \quad (2.2)$$

计算  $l$  和  $n_0$ ：

$$l = \frac{\text{size}(g, 2)}{k_0}; \quad (2.3)$$

$$n_0 = \text{size}(g, 1); \quad (2.4)$$

补充额外的零比特：

$$u = [\text{zeros}(\text{size}(1: (l - 1) * k_0)), \text{input}, \text{zeros}(\text{size}(1: (l - 1) * k_0))]; \quad (2.5)$$

生成编码器在各个时钟的卷积输出内容：

$$ul = u(l * k_0: -1: 1); \quad (2.6)$$

$$uu = \text{reshape}(ul, l * k_0, n + l - 1); \quad (2.7)$$

$$\text{output} = \text{reshape}(\text{rem}(g * uu, 2), 1, n_0 * (l + n - 1)); \quad (2.8)$$

### 2.2 判决译码器

判决译码与卷积编码相对应，由相应算法完成解码功能。程序中  $G$  与  $g$  相同，是一个  $n$  行  $L*k$  列矩阵，它的每一行决定了从移位寄存器到输入码字的连接方式。

$\text{survivor\_state}$  是一个矩阵，它显示了通过网格的最优路径，这个矩阵通过一个单独的函数  $\text{metric}(x,y)$  给出，这个算法既能用于硬判决译码也能用于软判决译码。

同样的判决译码器会检验  $G$  的维数：

$$\text{rem}(\text{size}(G, 2), k_0) > 0 \quad (2.9)$$

随后产生和初始化产生状态转移矩阵，输出矩阵和输入矩阵以开始非尾信道和尾信道输出的解码，最后从最优路径产生解码输出，由段得到状态序列，再由状态序列从  $\text{input}$  矩阵中得到该段的输出。

### 2.3 8PSK 算法

#### 2.3.1 8PSK 调制

"8PSK"中的"PSK"表示使用移相键控方式，移相键控是调相的一种形式，用于表达一系列离散的状态。因为 8PSK 拥有 8 种状态，所以 8PSK 每个符号(symbol)可以编码 3 个比特(bits)。8PSK 提供了更高的数据吞吐容量。

8PSK 信号的调制主要包括对每输入的三个比特信号映射到同相支路和正交支路上的相应电平，并对两路的电平分别进行滤波，最后再将滤波所得的同相支路和正交支路的信号调制到频带上去。其

框图如图所示：

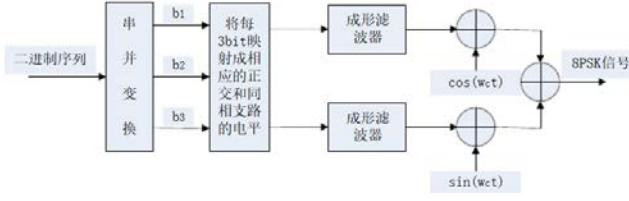


图 4 8PSK 调制框图

在程序中 8PSK 每个符号(symbol)即状态,由两个参数产生,分别为:

$$aa = \frac{\tan\left(\frac{3}{8}\pi\right)}{\sqrt{6}}; \quad (2.10)$$

$$bb = \frac{\tan\left(\frac{1}{8}\pi\right)}{\sqrt{6}}; \quad (2.11)$$

状态选择则取决于预调制比特信号在重新排列后的十进制信息。

### 2.3.1 8PSK 解调

在程序中,获取调至信息的实部和虚部:

$$rr = \text{real}(\text{mod\_data}(i)); \quad (2.12)$$

$$ii = \text{real}(\text{mod\_data}(i)); \quad (2.13)$$

经由大小关系等比较,可以解调出原始信号的比特信息。由于是每个 8PSK 符号对 3 位比特进行编码。所以解码出的结果分八种情况。

### 2.4 MRC 合并

最大比合并是分集合并技术中的最优选择,相对于选择合并和等增益合并可以获得最好的性能,性能提升是由 Array Gain 带来的更高的信噪比,进而带来更好的误码率特性决定的。

AF 模式下最大比合并加权系数的决定因素与中继节点的放大系数有关(在程序中由 G 表示)。AF 由于是对接收信号的放大,所以对于 Relay node 到 Destination node 的信号加权系数还与 Source node 到 Relay node 信道的参数、Source 到 Relay node 信道的信号和噪声功率相关。

本程序主要计算了中继节点的放大系数,以确定最大合并加权系数:

$$H12 = \text{RayleighCH}(); \quad (2.14)$$

$$P = \text{mean}(|\text{BitsTX}(:)|^2); \quad (2.15)$$

$$\text{sig} = 10^{-\frac{\text{snr}}{20}}; \quad (2.16)$$

$$G = \sqrt{\frac{P}{P * |H12|^2 + 2\text{sig}^2}}; \quad (2.17)$$

## 3 误码率

### 3.1 协作通信下理论误码率分析

假定所传输的信号功率归一化,采用 MRC 合并方式,则此单中继模型下的 AF 协作通信系统的理论信噪比为:

$$\gamma = \gamma_1 + \gamma_2 \quad (3.1)$$

其中:

$$\gamma_1 = \frac{|a_1 \sqrt{P_s} h_{s,d}|^2}{|a_1|^2 N_0} = \frac{P_s |h_{s,d}|^2}{N_0} \quad (3.2)$$

和

$$\begin{aligned} \gamma_2 &= \frac{\left| a_2 \frac{\sqrt{P_r}}{\sqrt{P_r |h_{s,r}|^2 + N_0}} \sqrt{P_r} h_{r,d} h_{s,r} \right|^2}{N_0' |a_2|^2} \\ &= \frac{\frac{P_r^2}{P_r |h_{s,r}|^2 + N_0} |h_{s,r}|^2 |h_{r,d}|^2}{\left( \frac{P_r |h_{r,d}|^2}{P_r |h_{s,r}|^2 + N_0} + 1 \right) N_0} \\ &= \frac{1}{N_0} \frac{P_r^2 |h_{s,r}|^2 |h_{r,d}|^2}{P_r |h_{s,r}|^2 + P_r |h_{r,d}|^2 + N_0} \end{aligned} \quad (3.3)$$

对于 BPSK 系统,当发送“1”符号和发送“0”符号概率相等时,最佳判决门限  $b^*=0$ ,则系统的总误码率  $P_e$  为:

$$P_e = P(0)P(0/1) + P(1)P(1/)$$

$$0) = \frac{1}{2} \operatorname{erfc}(\sqrt{\gamma}) \quad (3.4)$$

其中,  $P(0/1)$  为符号“0”被判为符号“1”的概率。

在大信噪比 ( $\gamma \gg 1$ ) 条件下, 式(3.4)可近似表示为:

$$P_e \approx \frac{1}{2\sqrt{\pi\gamma}} e^{-\gamma} \quad (3.5)$$

可见, 无论是 AF 还是 DF 协作通信系统中通过中继产生分集, 其信噪比大于非协作通信系统信噪比 ( $\gamma \gg \frac{P_s |h_{s,d}|^2}{N_0}$ ), 因而其误码率得以降低。

## 4 仿真与分析

### 4.1 仿真过程与结果

程序首先设定循环次数为 1000, 以获得较为精确的误码率信息。设定随机比特流信息长度位 65 以及信噪比由 0 至 25, 步长为 1。

首先建立信道模型和噪声模型, 以建立基本的协作通信系统。在信道处理中对随机比特流信息进行二进制卷积编码和 8PSK 调制, 在中继节点进行 8PSK 解调和判决译码。同时在中继节点处进行中继放大和 MRC 合并。最终合并信号经过信道处理后送至目标节点, 并进行 8PSK 解调和判决译码。

目标节点所得到的信号与初始随机信号进行误码分析, 同时直发信号也进行误码分析, 绘制图形如图 5 所示:

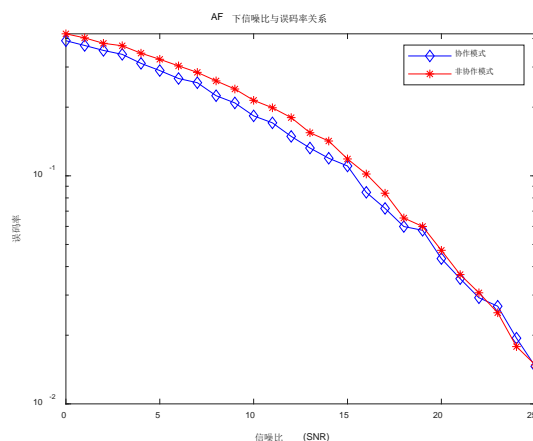


图 5 误码率曲线

### 4.2 实际误码率分析

由图可以看出, 在信噪比相同的情况下, 协作模式的误码率整体低于非协作模式的误码率, 符合理论结果。

文献[5]在具有 AWGN 噪声的 Rayleigh 衰落环境下进行 MATLAB 仿真, 得到的结果与本程序结果基本一致, 可验证其可靠性。

## 5 结论

本文分析了中继放大转发模式下协作通信与非协作通信不同的信噪比对误码率的影响, 通过建立简易模型, 得到一定精度下误码率的不同结果。

为了建立中继转发放大模型, 建立了 Rayleigh 信道模型, 采用 MRC 合并方式实现最大合并比, 设计了基于网络最优路径的判决解码功能, 仿真了 AF 协作通信过程, 得到了不同情况下的误码率曲线对比, 通过分析可知, 中继转发放大模式下的协作通信基本可有效降低通信过程中的误码率。

## 参考文献:

- [1] Wang Jian-Rong *et al* 2015 *Chinese Phys. B* **24** 060101.
- [2] Cover T, Gamal A E L. Capacity theorems for the relay channel[J]. *IEEE Transactions on information theory*, 1979, 25(5): 572-584.
- [3] Sendonaris A, Erkip E, Aazhang B. Increasing uplink capacity via user cooperation diversity[C]//*Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on*. IEEE, 1998: 156.
- [4] Sendonaris A, Erkip E, Aazhang B. User cooperation diversity. Part I. System description[J]. *IEEE Transactions on communications*, 2003, 51(11): 1927-1938.
- [5] Mohammad Amin Maleki Sadr, Mahmoud Ahmadian-Attari, Rouhollah Amiri, Vahid Veysi Sabegh, "Worst-Case Jamming Attack and Optimum Defense Strategy in Cooperative Relay Networks", *Control Systems Letters IEEE*, vol. 3, no. 1, pp. 7-12, 2019.

## 实验程序

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%12/17/2018      Wang Siwei
% 一个具有差错检测的 AF 仿真程序
% 并非原创，编解码部分有所借鉴（来自 Pudn）
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
echo off;clear all;close all;clc;
tic
N=1000;
L=65;    % 一帧长度
BerSnrTable=zeros(20,5);
for snr=0:25
    %snr = snrpot+snrpot;
    BerSnrTable(snr+1,1) = snr;
    sig=1/sqrt(10^(snr/10));
    %temp=0;%
    %temp1=0;%
    for i=1:N
        BitsTx = floor(rand(1,L)*2);
        P=mean(abs(BitsTx(:)).^2);
        %BitsTxcrc=CrcEncode(BitsTx);%
        BitsTxcnv=cnv( BitsTx);%确定二进制
        卷积编码器的输出序列

Mod8Tx=mod_8psk(BitsTxcnv);%8PSK 调制
M=length(Mod8Tx);%M=36
% 以下为假设信道模型和噪声模型，
% 由于本次仿真重点不在于此，所以做以下简化
        H1d=RayleighCH();%RayleighCH 信
        道模型
        H12=RayleighCH();
        H2d=RayleighCH();
        Z1d=randn(1,M)+j*randn(1,M);
        Z12=randn(1,M)+j*randn(1,M);
        Z2d=randn(1,M)+j*randn(1,M);

        %user2 接收并解码

```

```

        Y1d=H1d.*Mod8Tx+sig*Z1d;
        Y12=H12.*Mod8Tx+sig*Z12;
        R12=conj(H12).*Y12;
        BitR12=demod_8psk(R12);%8PSK 解
        调
        BitR12viterbi=viterbi(BitR12);% 硬判
        决译码和软判决译码，输出矩阵
        BitR12viterbi=BitR12viterbi(1:length(BitR12viterbi)-
        1);
        %[BitR12decrc,error]=CrcDecode(Bit
        R12viterbi);%CrcDecode.m 是网上找的开源代码，但
        是没用上
        %error=0,正确解码    error=1, 错误
        解码
        %非协作情况
        %if(error==1)%
            R1d=conj(H1d).*Y1d;
            BitR1d=demod_8psk(R1d);
            BitR1dviterbi=viterbi(BitR1d);

            BitR1dviterbi=BitR1dviterbi(1:length(BitR1dviterbi)-
            1);

            [Num1,Ber1]
            =
            symerr( BitR1dviterbi,BitsTx);

            BerSnrTable(snr+1,4)=BerSnrTable(snr+1,4)+Num1;

            %BitR1ddecrc=CrcDecode(BitR1
            dviterbi);%
            %[Num,Ber]
            =
            symerr(BitR1ddecrc,BitsTx);%
            %BerSnrTable(snr+1,2)=BerSnrTa
            ble(snr+1,2)+Num;%
            %end%
            %协作情况
            %if(error==0)%
                G=sqrt(P/(P*abs(H12).^2+2*sig*sig));
                Bits2d=BitR12viterbi;
                %Bits2dcrc=CrcEncode(Bits2d);

                %CRC 编码输出

```

```

        Bits2dcnv=cnv(Bits2d);% 确定二
进制卷积编码器的输出序列
        Mod8_2d=mod_8psk(Bits2dcnv);

Y2d=H2d.*G*(Mod8_2d+sig*Z12)+sig*Z2d;
        %最大合并比在此处的简化形式

Rd=conj(H2d).*Y2d+conj(H1d).*Y1d;
        BitRd=demod_8psk(Rd);
        BitRdviterbi=viterbi(BitRd);

BitRdviterbi=BitRdviterbi(1:length(BitRdviterbi)-1);
        %BitRddecr=CrcDecode(BitRdvi
terbi);%%CRC 编码输出
        [Num,Ber] =
symerr( BitRdviterbi,BitsTx);%求得误比特率

BerSnrTable(snr+1,2)=BerSnrTable(snr+1,2)+Num;%
误码数累加
        %temp=temp+1;%
        %end %
    end

BerSnrTable(snr+1,3)=BerSnrTable(snr+1,2)/(L*N);

BerSnrTable(snr+1,5)=BerSnrTable(snr+1,4)/(L*N);%
求出误码率
        %BerSnrTable(snr+1,4)=temp;%
    end
    semilogy(BerSnrTable(:,1),BerSnrTable(:,5),'bD-
',BerSnrTable(:,1),BerSnrTable(:,3),'r*-');
    xlabel('信噪比(SNR)');ylabel('误码率');title('AF
下信噪比与误码率关系');
    legend('协作模式','非协作模式')
    %figure
    %semilogy(BerSnrTable(:,1),BerSnrTable(:,4),'g*
-');
    time_of_sim = toc
    echo on;

```