



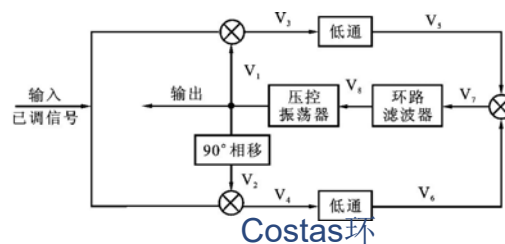
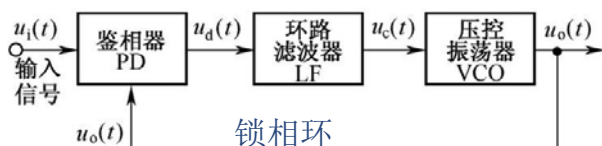
Outline

1. 通信系统模型的建立
2. 仿真中的几个重要概念
3. 链路级仿真基础知识

通信系统仿真及实现

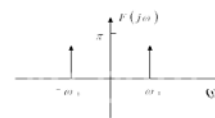
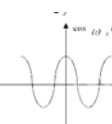


从最简单的通信链路说起



实数信号->频谱偶函数->频率搬移后形成双边带信号

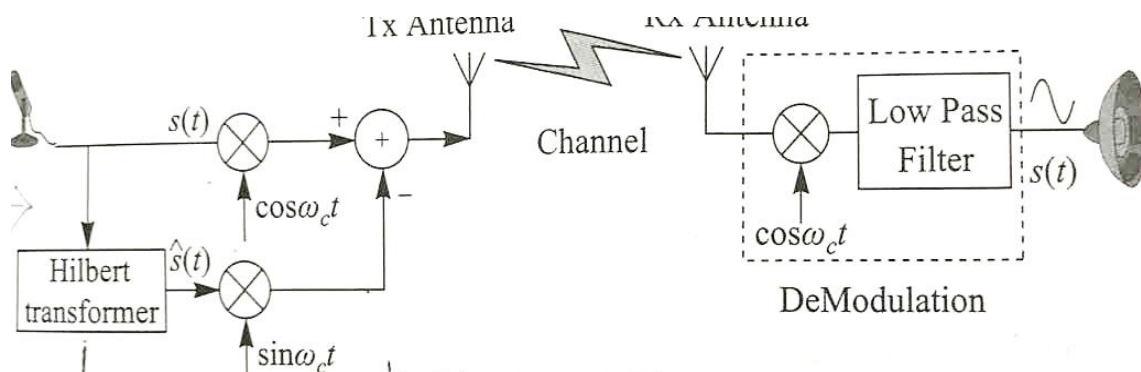
$$\cos[\omega_0 t] = \frac{1}{2} (e^{j\omega_0 t} + e^{-j\omega_0 t}) \Rightarrow \pi [\delta(\omega - \omega_0) + (\omega + \omega_0)]$$



通信系统仿真及实现

BUPT

❖ 单边带调制

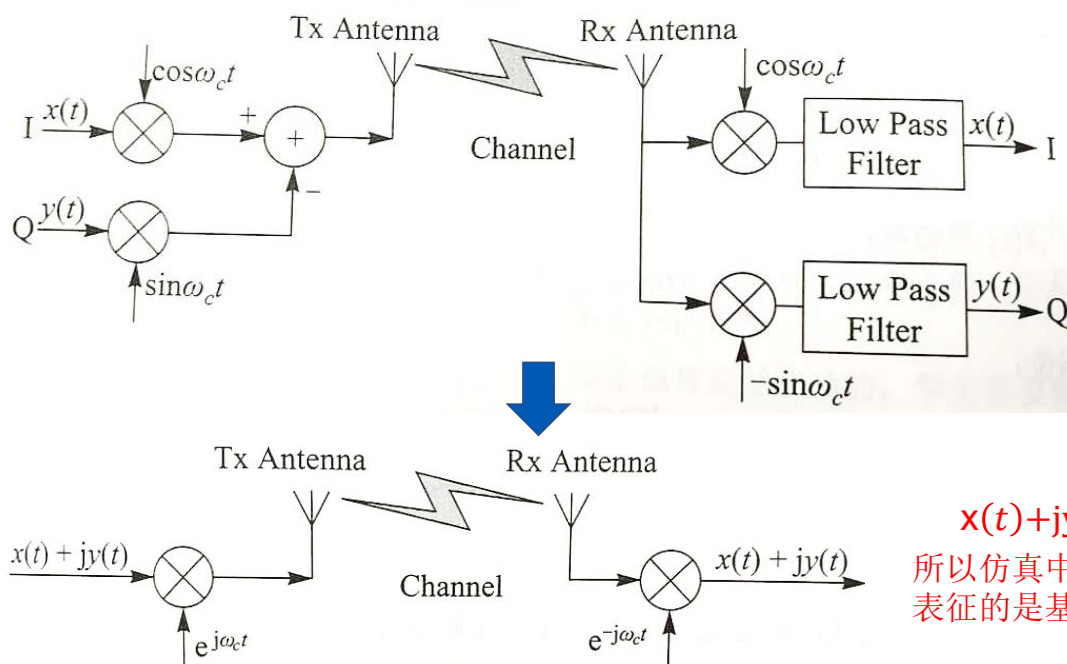


$$\begin{aligned} r(t) &= s(t)\cos(\omega_c t) - \hat{s}(t)\sin(\omega_c t) \\ &= (s(t) + j\hat{s}(t)) \cos(\omega_c t) \end{aligned}$$

通信系统仿真及实现

BUPT

❖ IQ调制



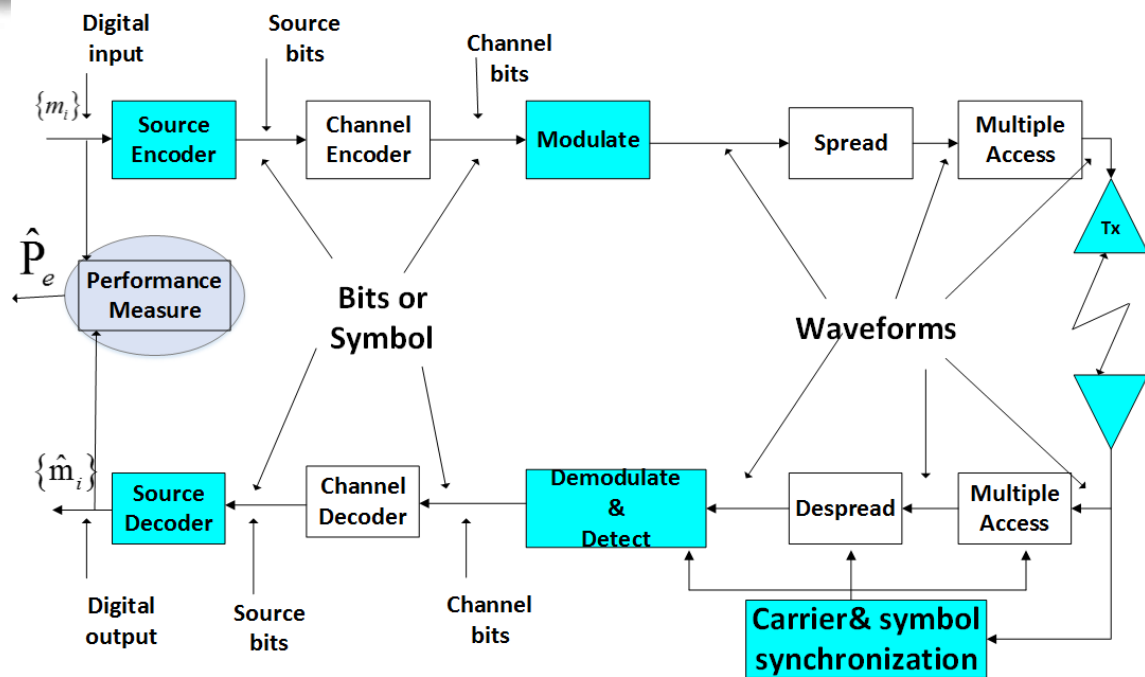
$x(t) + jy(t)$
所以仿真中的复数
表征的是基带信号

通信系统仿真及实现

BUPT



通信系统物理层模型



❖ 收发信息

- 用尽量少的能量(SNR, CNR)
- 尽量少的错误 (PER)
- 信道条件 (AWGN, 衰落信道)

通信系统仿真及实现

BUPT



Outline

1. 通信系统模型的建立
2. 仿真中的几个重要概念
3. 链路级仿真基础知识

通信系统仿真及实现



链路级仿真vs系统级仿真

❖ 尺度不同

- 链路级：点到点通信
- 系统级：整个区域通信

❖ 关注的颗粒度不同

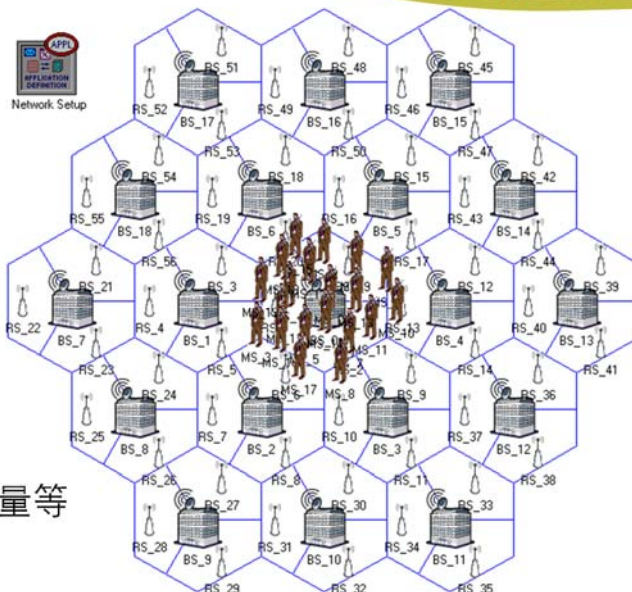
- 链路级：bit的仿真
- 系统级：packet的仿真

❖ 仿真结果收集不同

- 链路级：BER、BLER等
- 系统级：误包率、时延、吞吐量等

❖ 所用软件

- 链路级：matlab/simulink/labview/C
- 系统级：C/NS3/opnet



❖ 静态仿真Snapshot

- 撒点->计算SINR->TP映射



Monte Carlo仿真

❖ 在通信系统的计算机仿真中需要用到具有特定分布的各种随机数（或随机序列）

❖ 使用随机数产生器的仿真技术被称为Monte Carlo仿真

- Monte Carlo：城市名，位于摩纳哥，是世界著名的赌城
- Monte Carlo仿真来源于赌博。其基本思想是通过仿真用Monte Carlo技术来估计系统的某个参数，如误比特率（BER）

❖ 所谓随机数，就是相应分布的随机变量的抽样值。大量的抽样值，应服从它来自的那个母体（随机变量）的分布



Monte Carlo仿真



- ❖ 相对频率与概率：根据概率论，我们感兴趣的某个随机事件A（如二进制数字通信中“0”出错的概率）出现的概率可用N次随机试验中A发生的相对频率来近似，有

$$\Pr(A) = \lim_{N \rightarrow \infty} \frac{N_A}{N}$$

- ❖ 对于有限值N，我们有事件A的估计值

$$\hat{\Pr}(A) = \frac{N_A}{N}$$

- Monte Carlo估计的理论依据

通信系统仿真及实现

BUPT



❖ 高斯随机变量

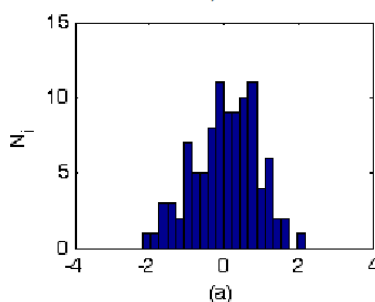
```
subplot(2,2,1)
x = randn(1,100); hist(x,20)
ylabel('N_i'); xlabel('(a)')
```

```
subplot(2,2,2)
x = randn(1,100); hist(x,5)
ylabel('N_i'); xlabel('(b)')
```

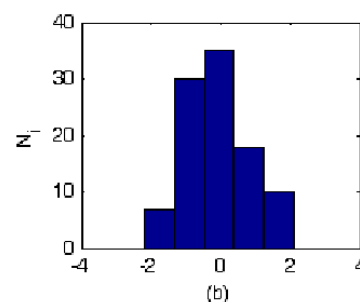
```
subplot(2,2,3)
x = randn(1,1000); hist(x,50)
ylabel('N_i'); xlabel('(c)')
```

```
subplot(2,2,4)
x = randn(1,100000); hist(x,50)
ylabel('N_i'); xlabel('(d)')
```

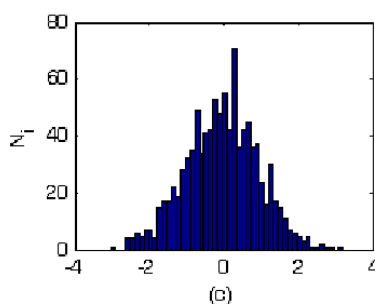
N=100,B=20



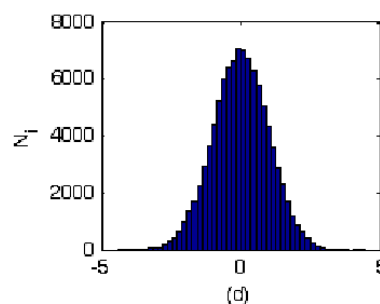
N=100,B=5



N=1000,B=50



N=100000,B=50



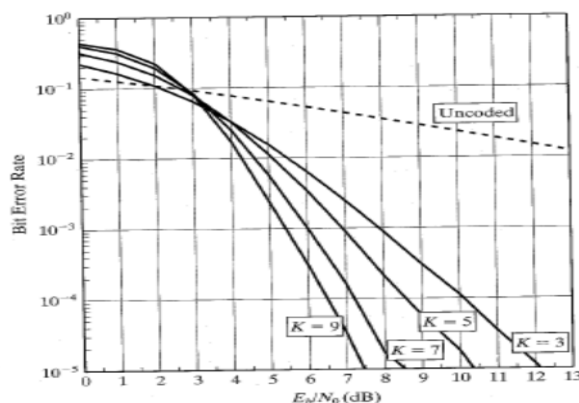
通信系统仿真及实现

BUPT



BER/BLER/FER

- ❖ RawBER
- ❖ BER——Bit Error Rate
- ❖ BLER——Block Error Rate
- ❖ FER——Frame Error Rate
- ❖ PER——Packet Error Rate



通信系统仿真及实现

BUPT



BER-SNR

- ❖ 通过仿真，我们希望获得误比特率（BER）与信噪比（SNR）之间的变化曲线。
- ❖ 在很多情况下，我们用每个比特的平均能量与噪声的单边功率谱密度之比 E_b/N_0 作为 SNR 的度量值。
- ❖ 根据信噪比（SNR）的定义，SNR 是信号平均功率（S）与噪声平均功率（N）之比，

$$S = \frac{E_s}{T_s} = E_s R_s = E_s R_b / \log_2 M = E_b R_b$$

$$N = \sigma_n^2 = N_0 B$$

其中， E_s 为每个符号的能量， T_s 为每个符号的周期， M 为每个符号的进制数， B 为信道带宽。

因此信噪比（SNR）与 E_b/N_0 的关系为

$$SNR = \frac{S}{N} = \frac{E_b R_b}{N_0 B} = \frac{E_b}{N_0} \cdot \frac{R_b}{B}$$

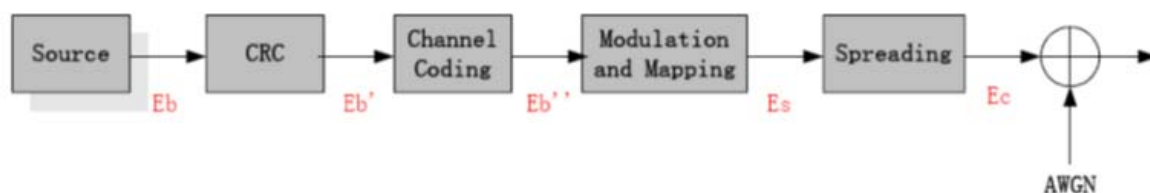
通信系统仿真及实现

BUPT



SNR与 E_b/N_0

物理链路发送端的简化流程



信源单位输出随机的二进比特信息流，假设1比特的能量为 E_b 。

首先进行循环冗余编码（CRC），假设CRC后的1比特的能量为 E_b' ，有：

$$E_b' = \eta_1 * E_b$$

式中 η_1 为CRC检错编码的效率， $\eta_1 = \frac{L_I}{L_I + L_{CRC}}$

其中 L_I 为进行CRC的比特块的长度， L_{CRC} 为CRC检错编码的尾比特数。



SNR与 E_b/N_0

CRC检错编码后的比特流进行**信道的纠错编码**，假设信道编码后的比特能量为 E_b'' ，而信道编码的效率为 η_2 （一般为1/3或1/2），则有：

$$E_b'' = \eta_2 * E_b' = \eta_1 * \eta_2 * E_b$$

接下来进行**调制**和**符号映射**。如果采用M元信号的调制方式，那么调制后一个符号的能量为：

$$E_s = \log_2^M * E_b'' = \log_2^M * \eta_1 * \eta_2 * E_b$$

最后，符号经过**扩频处理单元**后送入通道。如果采用的扩频因子为SF，那么一个码片的能量 E_c 为

$$E_c = E_s / SF = \log_2^M * \eta_1 * \eta_2 * E_b / SF$$



SNR与Eb/N₀的换算关系

如果码片的周期为 T_c ，那么信道中信号的功率为：

$$P_S = \frac{E_C}{T_c} = \frac{\log_2^M * \eta_1 * \eta_2 * E_b}{T_c * SF}$$

如果信道为加性高斯白噪声，其单边功率谱密度为 N_0 ，信道的采样周期为 T_{sam} ，那么噪声的功率为：

$$P_N = \frac{N_0}{T_{sam}} = \frac{N_0}{T_{sam}}$$

则接收端的信噪比为

$$SNR = \frac{P_S}{P_N} = \frac{\log_2^M * \eta_1 * \eta_2 * T_{sam}}{T_c * SF} * \frac{E_b}{N_0}$$

这样就得到了SNR和Eb/N₀间的关系。



SNR与Eb/N₀

例：

一般情况下， η_1 近似为1，不考虑信道编码的影响，而且抽样周期和码片周期相同，上式简化为：

$$SNR = \frac{P_S}{P_N} = \frac{\log_2^M}{SF} * \frac{E_b}{N_0}$$

若扩频因子为16，QPSK调制，则 $SNR = \frac{P_S}{P_N} = \frac{2}{16} * \frac{E_b}{N_0}$

换算成dB，有 $SNR_{dB} = 10\log_{10} \frac{P_S}{P_N} = 10\log 2 - 10\log 16 + \left(\frac{E_b}{N_0} \right)_{dB}$

$$= 3 - 12 + \left(\frac{E_b}{N_0} \right)_{dB}$$

$$= -9 + \left(\frac{E_b}{N_0} \right)_{dB}$$



Outline

1. 通信系统模型的建立

2. 仿真中的几个重要概念

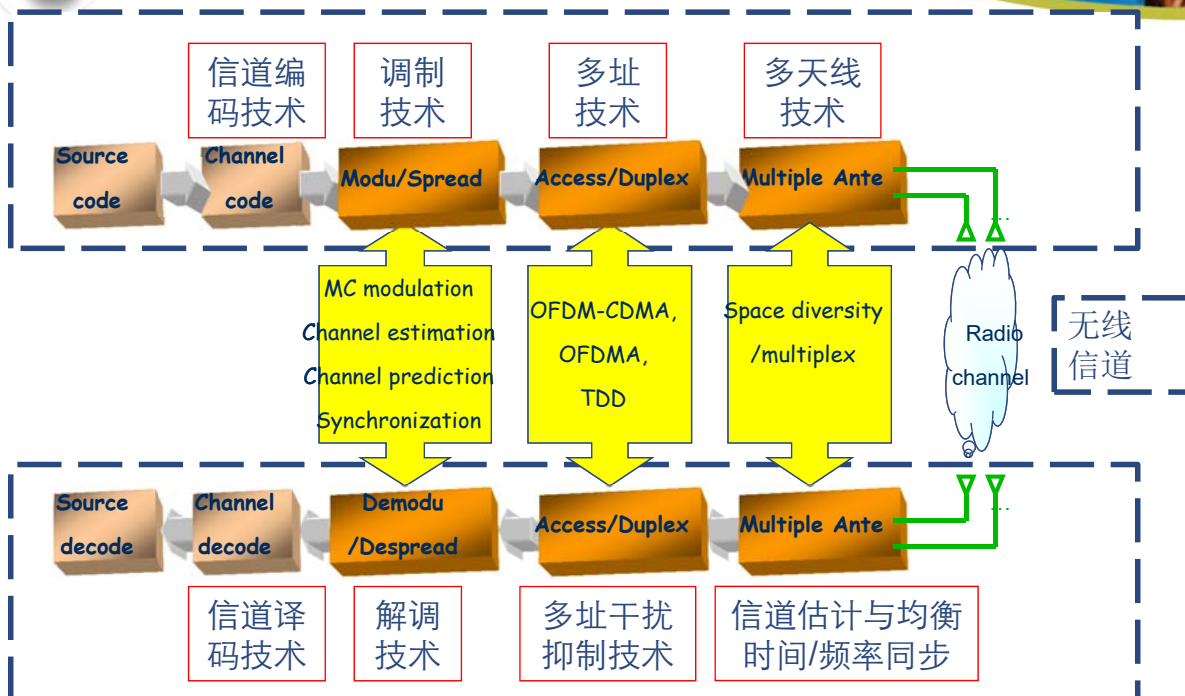
3. 链路级仿真基础知识

- 链路仿真的系统模型
- 仿真软件的基本使用
- 仿真结果的记录与处理
- 仿真误差分析方法

通信系统仿真及实现



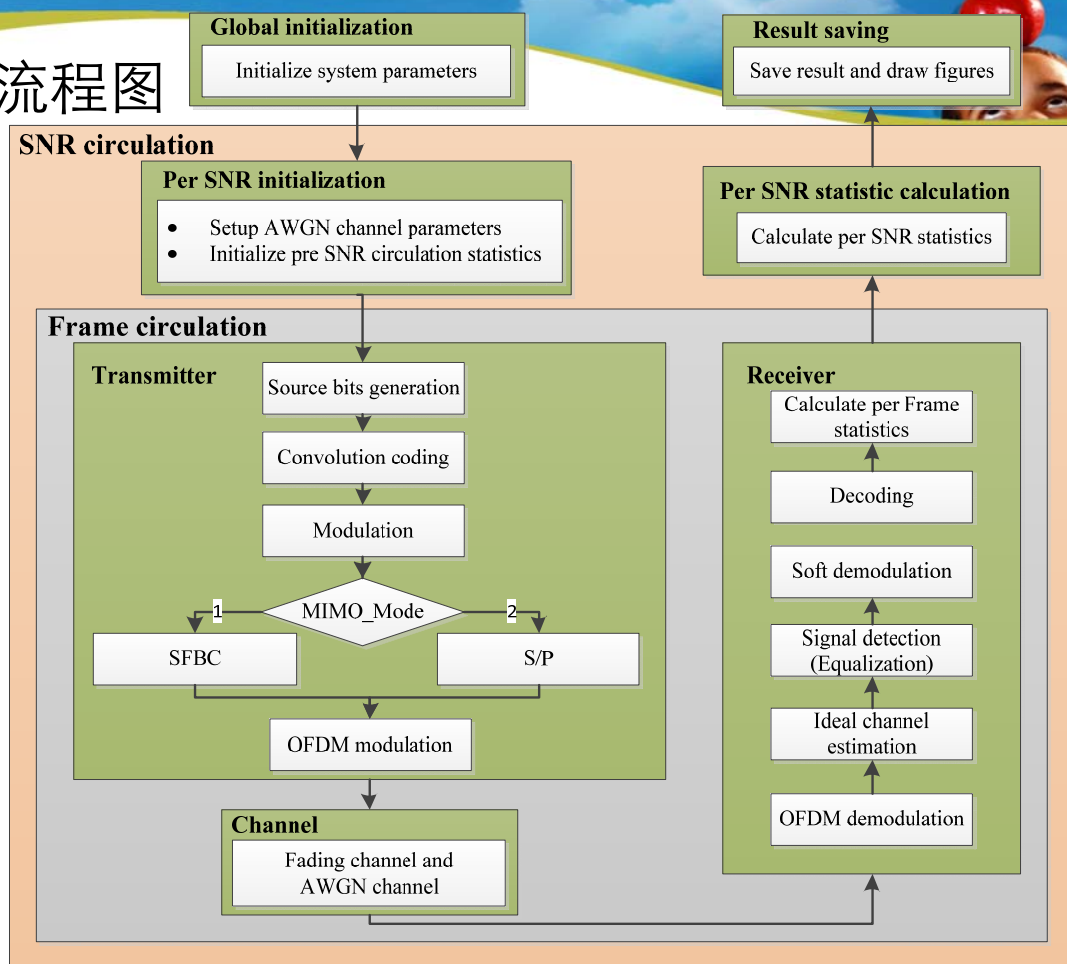
链路仿真的系统模型



通信系统仿真及实现

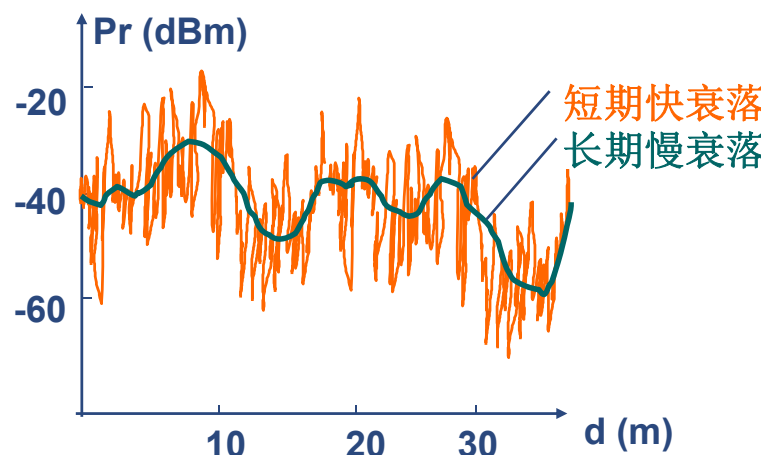
BUPT

程序流程图



❖ 关于信道的说明

- 链路级仿真中 大多数情况不考虑大尺度衰落的建模
 - 因为大尺度衰落描述与位置相关的能量损耗，可以用功率控制等方式克服
- 重点刻画小尺度衰落
 - 对性能影响大
 - 体现各种技术的差异
- 噪声和干扰需要建模





简介

链路级仿真、验证各种算法的主要仿真软件



软件界面：

- 主窗口：
 - current directory
 - command window
 - workspace
 - command history
- Help窗口
 - 其他方法： help + function_name
- 编译窗口: Editor
 - 保存文件 file.m (可直接用记事本打开)
 - 自动保存文件 file.asv

22

通信系统仿真及实现



基本文件结构

- ❖ 一个完整的程序由主函数和其它外部调用函数组成。为了方便整理这些不同作用的函数，我们一般把它们放在不同的文件夹里，并给其一定的命名
- ❖ 一般包括： main, Tx, channel, Rx, Result,

在主函数开始的时候会调用这些外部文件。

23

通信系统仿真及实现



基本文件结构

❖ 文件夹的命名

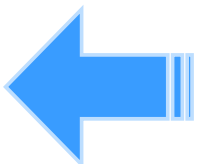
❖ 文件的命名

❖ 备注 %

- 整个文件的说明
- 程序流程的说明
- 单句命令的功能说明

❖ 外部文件夹中函数的调用

- `addpath(path,'../Tx');`
- `y = function_name(x);`



良好的编程习惯



常用函数

❖ `randn` 正态分布

❖ `randi` 均匀分布的伪随机整数 (1到imax)

`A=randi(imax,m,n)` m*n矩阵

`B=randi(imax)` 单个随机数

`C=randi([c,d],...)` 范围[c,d]

提问：如何产生300个0,1随机bit?



常用函数

❖ reshape

`B=reshape(A,a,b)` 列优先

❖ plot

❖ semilogy

- 结构均相同
- 但semilogy对纵轴取以10为底的对数

❖ `plot/semilogy(x,y,' 颜色+节点形状+连线')`

❖ 注意添加横纵坐标名称



常用函数

❖ find

- 避免用for循环进行遍历

例子 `index=find (A~=B)`

- 找到数组A和数组B中不相等的元素



设置断点和调试

❖ 设置断点调试

❖ 单步调试

❖ 对仿真时间的估计

- 学会用tic、toc
 - tic 表示开始秒表计时
 - toc 对应tic 表示计时结束
 - 程序运行时间是toc-tic



仿真结果的记录与处理

❖ 数据的存储与读取

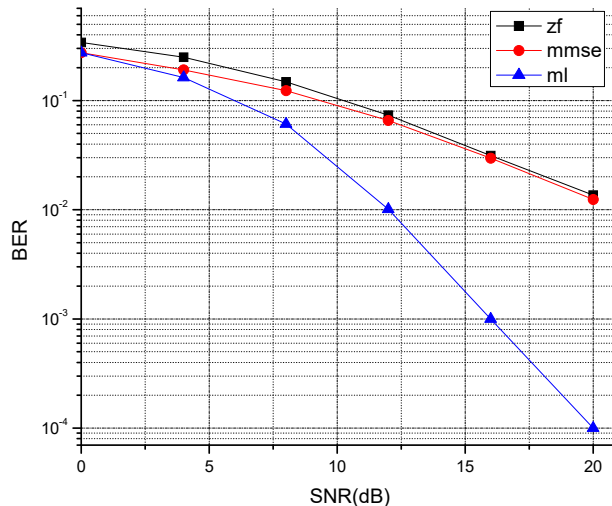
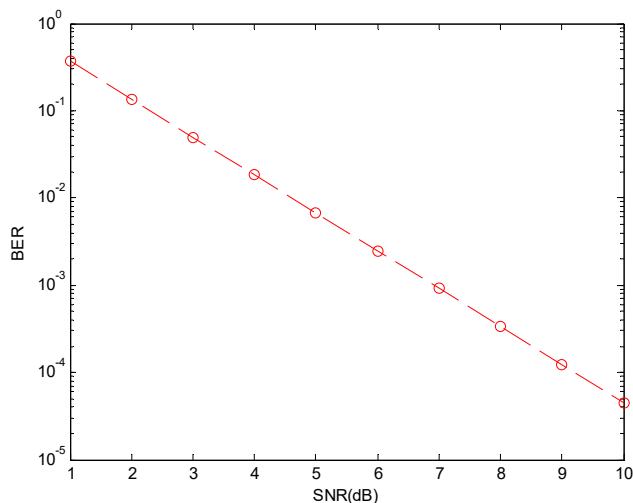
- MATLAB的数据文件 data.mat
- 存储数据的函数：save （可保存成多种文件形式）
- 读取数据的函数：load



仿真结果的记录与处理

❖ 如何提取数据画图

- 在MATLAB中直接画图：方便观察仿真结果
- 在origin里画图：适用于正式文档中
- 注意横纵坐标轴，学习编辑legend



30

通信系统仿真及实现



仿真误差分析方法

❖ 理论分析是否与仿真结果一致

❖ 如何让仿真尽快收敛

- 结果曲线平滑
- 注意横纵坐标

❖ SNR和帧数参数的设计

- 估计BER和BLER曲线的趋势
- 设置帧数使得
 - 错误bit至少出现1000个
 - 错误块至少出现100个
 - 低SNR时，仿真需要的帧数少;高SNR时，仿真需要的帧数多
 - 在曲线陡峭处，SNR的间隔设置更密

31

通信系统仿真及实现