

实验 6 ALOHA

一、实验目的

加深对纯 ALOHA 和 Slotted ALOHA 分析的理解并通过仿真观察随机接入现象、验证理论结果。

二、实验环境

所提供的参考代码使用的是 Matlab，任意版本均可，也可自选其它编程语言。

三、实验原理

考虑随机接入系统共有 N 个用户，假设每个用户的数据帧长 b (bits)、无碰撞时的信息发送速率 R (bps)、帧到达率 λ (帧/s) 均相同，且每个用户的帧到达过程服从独立同分布的泊松过程。每一帧在无碰撞时的发送时间为 $T_{fr} = b/R$ (s)。

1、纯 ALOHA

纯 ALOHA 中每个用户在帧到达时刻即立即发送，脆弱时间为 $2T_{fr}$ ，则每一帧成功传输的概率为：

$$P_s = e^{-2N\lambda T_{fr}}$$

则归一化信息吞吐速率 ρ 为：

$$\rho = \frac{P_s N \lambda b}{R} = \frac{N \lambda b}{R} e^{-2N\lambda T_{fr}}$$

定义归一化总业务量 $G = \frac{N \lambda b}{R} = N \lambda T_{fr}$ ，则有：

$$\rho = G e^{-2G}$$

不难得出在 $G = 0.5$ 时， ρ 取得最大值 $\frac{1}{2e}$

2、Slotted ALOHA

ALOHA 协议中， $2T_{fr}$ 的脆弱时间过长，因此考虑将时间划分为长度为 T_{fr} 的时隙(slots)，并规定用户只能在每个时隙的起始时刻开始发送。

若有多于一帧在同一时隙传输，则会发生冲突。Slotted ALOHA 的脆弱时间为 T_{fr} ，降低为纯 ALOHA 的一半。

类似纯 ALOHA 的推导过程，某一帧成功传输要求脆弱时间内没有其他帧发送：

$$P_s = e^{-N\lambda T_{fr}} = e^{-G}$$

则可得归一化信息吞吐速率 ρ 为:

$$\rho = Ge^{-G}$$

不难得出在 $G = 1$ 时, ρ 取得最大值 $\frac{1}{e}$ 。

四、实验内容

用户的帧到达过程可以有两种设置, 设置 1 是和实验原理部分中的归一化信息吞吐速率结论完全对应, 即帧到达过程是泊松过程, 每个用户的帧到达时间间隔独立同分布地服从于均值为 $\frac{1}{\lambda}$ 的指数分布。但这种帧到达过程会存在某一用户在发送某一帧的持续时间 T_{fr} 内有一新的帧到达该用户的情况, 理论计算过程将该情况视为冲突产生。

帧到达过程设置 2 为每个用户的帧到达时间间隔为 T_{fr} 加上一个独立同分布的均值为 $\frac{1}{\lambda}$ 的指数分布随机变量, 这样就避免了上述冲突情况, 但这种设置会使得每个用户的帧到达过程不再是泊松过程, 且真实的帧到达率不再是 λ , 需要对 λ 进行修正。

参考代码已完整给出了两种帧到达过程设置下的仿真过程, 请理解代码并设置合适的参数运行, 以完成下面任务。

(1)、设置用户数 N 和单用户的帧到达率 λ , 分别绘制出纯 ALOHA 和 Slotted ALOHA 的理论计算的 $\rho - G$ 曲线和在上述两种帧到达过程设置下的仿真 $\rho - G$ 曲线(设置 2 下先不修正 λ)。观察结果, 并解释仿真结果和理论结果有差距的原因。

(2)、在(1)的绘图结果的基础上多绘制一条仿真 $\rho - G$ 曲线: 在帧到达过程设置 2 下修正 λ 使之更接近于真实的到达率, 观察结果。代码中给出了一种修正方法, 解释它的原理。

(3)、改变用户数 N , 重做(2), 观察总结 N 对仿真结果的影响, 并解释。