1. **举例说明 GBN 算法窗口大小取值不当会出现什么问题？**

假设发送分组序号为0/1/2/3/0/1/2/3…。

（1）.当发送窗口最大值为2^n=2^2=4时：发送方发送序号从0-3的4个帧，等待确认。接受方正确收到这4个帧，进行确认。发送方有两种情况：（1）发送序号为0-3的4个新帧（2）因确认丢失而超时，重发4个老帧。此时接受方无法判断收到的是4个新帧还是重发的4个老帧。

产生错误。

（2）.当发送窗口最大值为2^n-1=2^2-1=3时，发送方发送序号从0-2的3个帧，等待确认。接受方正确收到这3个帧，进行确认。此时发送方有两种情况：（1）收到确认，发送序号为3及0-1的3个新帧（2）因确认丢失超时，重发序号为0-2的3个老帧。收方如果收到序号为3的帧，表明这一批是新帧就接收并且对序号2确认，如果没有收到序号为3的帧，则对0-2序号的帧不予接收，并重发对序号3的确认帧。知道收到正确的帧。

没有矛盾。

1. **举例说明 SR算法窗口大小取值不当会出现什么问题？**

假设发送分组序号为0/1/2/3/0/1/2/3…。

如果窗口取值2^n=2^2=4

发送方传012，接收方收到，滑动窗口前进至3,0,1，接收方发回ack，ack丢失，发送方重传0,1,2，如果第一个序列的3丢失了，此时接收方不能判断是发送方重传的0,1,2还是发送方传了第二个重复序列的0,1。

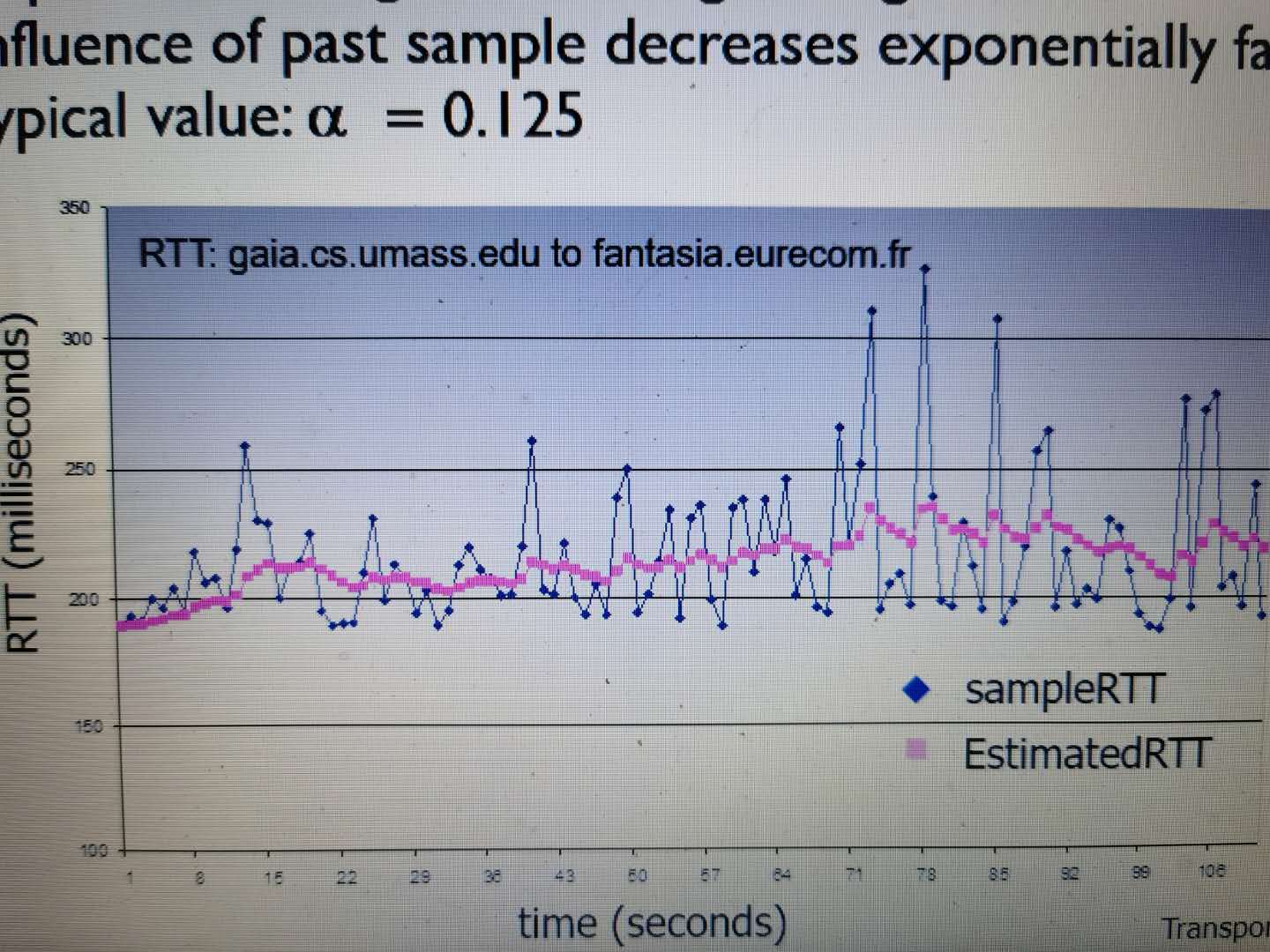
产生错误。

SR算法窗口取值为2^(n-1)=2

发送方传0,1，接收方收到，滑动窗口移动至2,3，发回ack，ack丢失，发送方重传0,1，接收方看到发送方发0，1,与当前窗口不符，判断重传

没有矛盾。

1. **如下图，取连续六个点的SampleRTT（采样）值，计算EstimatedRTT（估计） 以及 TimeoutInterval。**



**迭代算法：**EstimatedRTT= (1-x)\*EstimatedRTT+ x\*SampleRTT

其中SampleRTT为上一个数据的实际往返时间，x的值一般默认取0.125。（x仿真取值）

**偏差：**DevRTT = (1-β)\*DevRTT + β\*|SampleRTT - EstimatedRTT|

其中β=0.25（推荐值）

**超时：**TimeoutInterval = EstimatedRTT + 4\*DevRTT

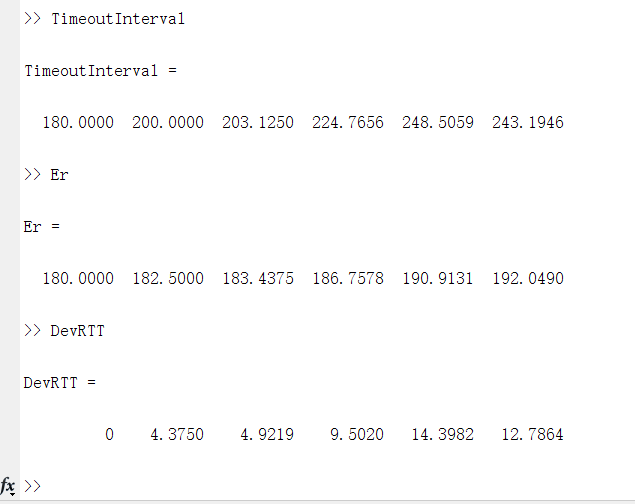
令EstimatedRTT初始值为SampleRTT，DevRTT初始值为0

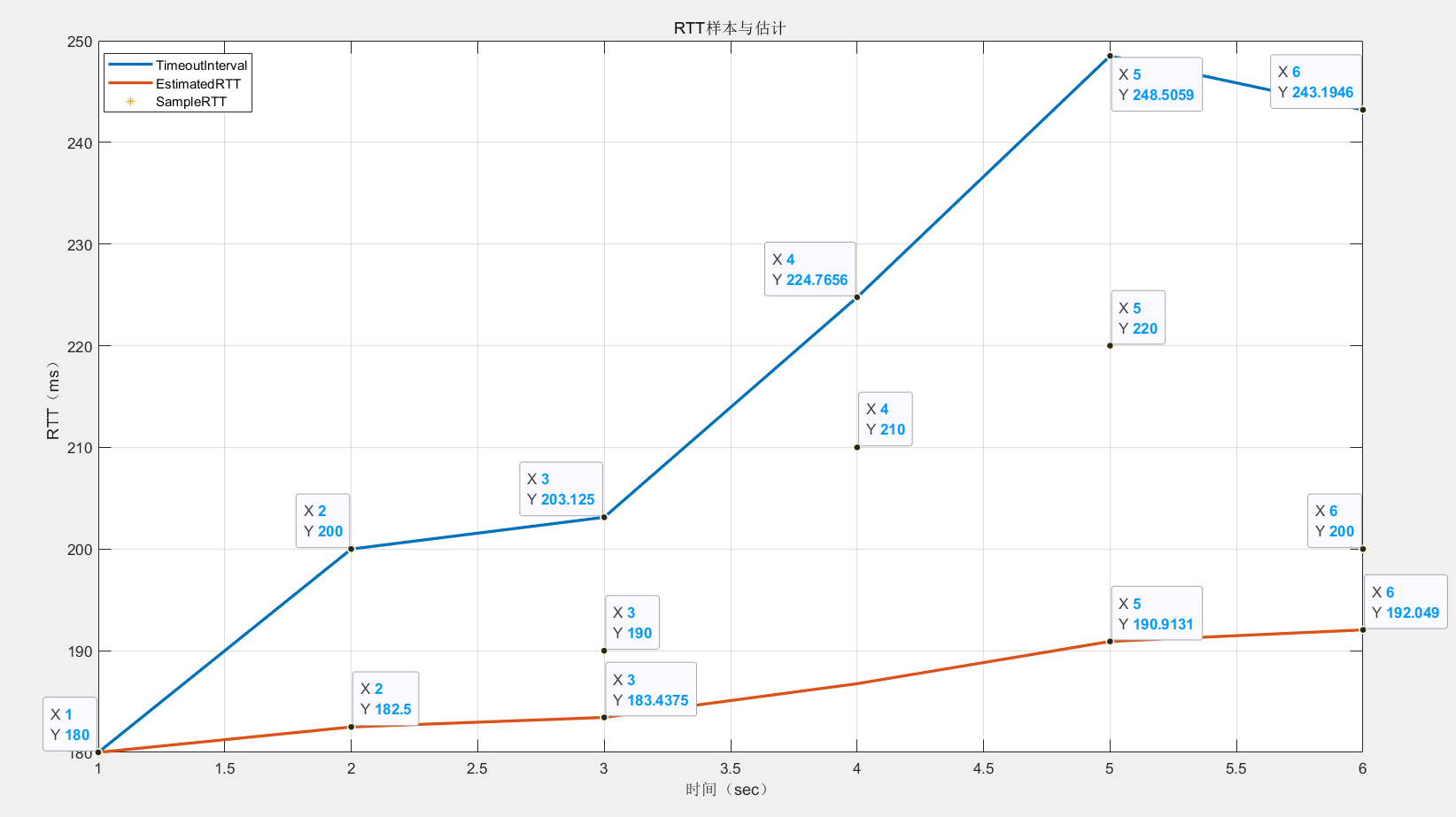
假设1，2，3，4，5，6 sec的samplertt的取值为180，200，190，210，220，200

Ertt1 = 0.875\*180+0.125\*180 = 180

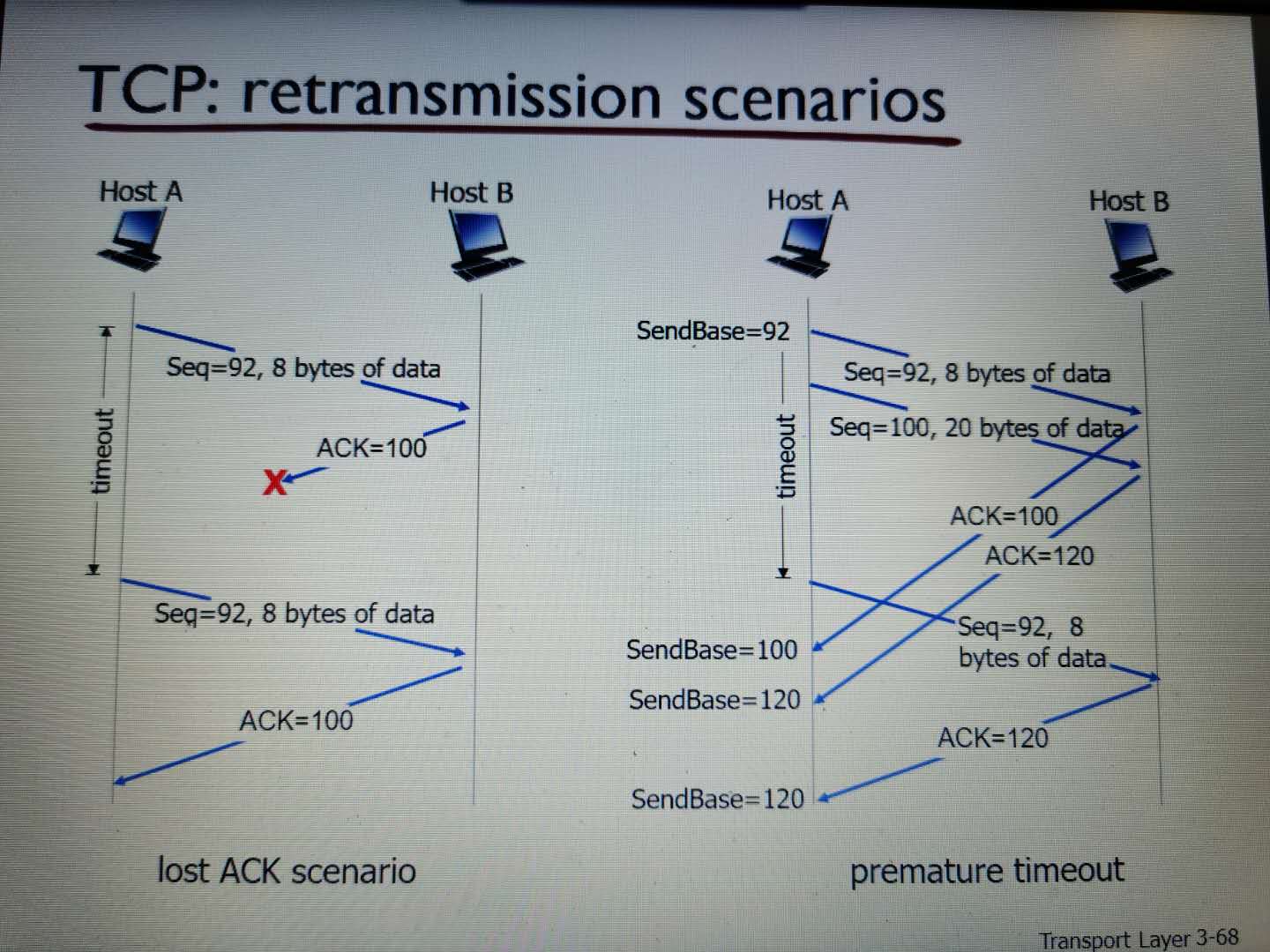
……

利用matlab计算结果为：





1. **如下图解释TCP重发过程。**



第一个图 发送方未收到ack100 重新发送8bytes（92开始） 但是接收方收到后丢弃并发送ack100，如此重复（发送方发100以后字节）

第二个图 发送方未收到ack100 超时重新发送8bytes（92开始） 但是接收方收到后丢弃并发送ack120，如此重复（发送方发120以后字节）