

# 基于UDP服务设计可靠传输协议并编程实现之3-4（性能对比测试）

## 实验内容

基于给定的实验测试环境，通过改变延迟时间和丢包率，完成下面3组性能对比实验：

- 1. 停等机制与滑动窗口机制性能对比
- 2. 滑动窗口机制中不同窗口大小对性能的影响
- 3. 有拥塞控制和无拥塞控制的性能比较。

## 本实验注意事项

严格控制变量，每次之改变同一个变量，保持其它变量不变。

## 停等机制(3-1)与滑动窗口机制(3-2)性能对比

代码中超时设置为200ms，路由器中设置延时分别为0、30、60、90、120ms，丢包率设置为0%、5%、10%、15%。每次测试均发送文件1.jpg。

测得数据如下，每格中时间单位为ms，吞吐率单位为kpbs。

### 停等机制测试结果

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间333、吞吐率43.57	时间5842、吞吐率2.4	时间9642、吞吐率1.50	时间13773、吞吐率1.05	时间17799、吞吐率0.81
5	时间2483、吞吐率5.84	时间9007、吞吐率1.61	时间13013、吞吐率1.11	时间16724、吞吐率0.87	时间20568、吞吐率0.70
10	时间4356、吞吐率3.33	时间10928、吞吐率1.33	时间14621、吞吐率0.99	时间18493、吞吐率0.78	时间22321、吞吐率0.65
15	时间6999、吞吐率2.07	时间13418、吞吐率1.08	时间17390、吞吐率0.83	时间20993、吞吐率0.69	时间24635、吞吐率0.62

### 滑动窗口机制测试结果

滑动窗口大小设置为8

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间98、吞吐率148.07	时间2322、吞吐率6.80	时间4872、吞吐率3.37	时间6987、吞吐率2.40	时间9098、吞吐率1.70
5	时间2844、吞吐率5.10	时间5766、吞吐率2.81	时间7542、吞吐率2.19	时间9863、吞吐率1.66	时间11367、吞吐率1.35
10	时间5134、吞吐率3.20	时间8645、吞吐率1.96	时间11298、吞吐率1.28	时间18644、吞吐率0.86	时间21683、吞吐率0.77
15	时间7435、吞吐率2.11	时间10375、吞吐率1.59	时间15233、吞吐率0.98	时间20114、吞吐率0.70	时间25322、吞吐率0.59

## 结论

结合测试结果看出

- 在较小丢包率的情况下（丢包率小于等于5%时），使用滑动窗口机制比使用停等机制有较好的加速比，大约两倍左右。这是因为丢包较小时，滑动窗口采用了并行发送多个数据包的方式，大大优化了停等机制中每发一个报都要等一个ACK的情况。
- 在较大丢包率的情况下（丢包率大于5%时），使用滑动窗口机制相对于停等机制的加速比明显降低，甚至有的时候性能不如停等机制。这是由于滑动窗口在丢包时需要重传数据包并等待重传数据包的ACK收到之后才能将窗口移动，因此定长滑动窗口受丢包率的影响较大。

## 滑动窗口机制(3-2)中不同窗口大小对性能的影响

滑动窗口固定大小分别设置为8、16。路由器中设置延时分别为0、30、60、90、120ms，丢包率设置为0%、5%、10%、15%。每次测试均发送文件1.jpg。

测得数据如下，每格中时间单位为ms，吞吐率单位为kpbs。

滑动窗口大小设置为8

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间98、吞吐率157.50	时间2322、吞吐率7.45	时间4722、吞吐率3.56	时间7196、吞吐率2.23	时间9261、吞吐率1.74
5	时间2782、吞吐率5.85	时间5592、吞吐率2.72	时间7522、吞吐率2.21	时间9915、吞吐率1.62	时间11520、吞吐率1.33
10	时间5390、吞吐率3.12	时间8519、吞吐率1.97	时间11092、吞吐率1.35	时间18254、吞吐率0.92	时间20473、吞吐率0.76
15	时间7029、吞吐率2.25	时间11283、吞吐率1.40	时间16194、吞吐率0.93	时间21630、吞吐率0.71	时间25970、吞吐率0.59

滑动窗口大小设置为16

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间105、 吞吐率 146.91	时间568、 吞吐率 26.81	时间1794、 吞吐率9.08	时间3733、 吞吐率	时间4587、 吞吐率
5	时间1567、 吞吐率 10.08	时间3826、 吞吐率4.44	时间5296、 吞吐率3.14	时间6422、 吞吐率2.45	时间7722、 吞吐率2.01
10	时间4743、 吞吐率3.24	时间8023、 吞吐率1.96	时间9826、 吞吐率1.64	时间 16232、吞 吐率0.96	时间 18993、吞 吐率0.85
15	时间6991、 吞吐率2.20	时间 11432、吞 吐率1.36	时间 15362、吞 吐率0.99	时间 19224、吞 吐率0.80	时间 24732、吞 吐率0.67

## 结论

结合测试结果看出

- 在丢包率较小时（小于等于5%时），使用更大的滑动窗口相对于较小的滑动窗口有不错的性能提升。加速比大概在1.5到2.0之间。分析滑动窗口机制，此时丢包率较小，使用更大的滑动窗口一次能发出更多的包，而没有太多的丢包导致的重传，所以能起到较好的加速效果。
- 在丢包率较大时（大于5%时），使用更大的滑动窗口相对于较小的滑动窗口性能优势明显减弱，加速比随路由器延时的增加逐渐减小到1.0左右，甚至在有几次测试中，性能不如更较小滑动窗口。分析重传机制，当有较多丢包时，滑动窗口需要累积确认ACK、重传失序未确认的数据包，当这些失序的数据包ACK收到后才能移动窗口，因此有较多的性能损失。尤其是当路由延时设置较大时，性能损失更明显。

## 有拥塞控制(3-3)和无拥塞控制(3-2)的性能比较

为控制变量，将设为无拥塞控制的窗口长度统一设置为8。

定长滑动窗口大小（设置为8）

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间89、吞吐率181.96	时间2554、吞吐率6.62	时间4385、吞吐率3.59	时间7685、吞吐率2.08	时间8198、吞吐率1.89
5	时间2560、吞吐率5.94	时间4352、吞吐率3.64	时间6788、吞吐率2.44	时间10849、吞吐率1.46	时间12503、吞吐率1.20
10	时间5647、吞吐率2.94	时间7781、吞吐率1.97	时间10169、吞吐率1.52	时间16780、吞吐率0.98	时间23851、吞吐率0.68
15	时间8178、吞吐率1.97	时间11412、吞吐率1.42	时间16756、吞吐率1.00	时间21125、吞吐率0.73	时间26854、吞吐率0.59

### 拥塞控制

miss(%)\路由延时(ms)	0	30	60	90	120
0	时间935、吞吐率16.37	时间2595、吞吐率5.97	时间4695、吞吐率3.32	时间6843、吞吐率2.36	时间8780、吞吐率1.79
5	时间1954、吞吐率8.44	时间5692、吞吐率2.71	时间6234、吞吐率2.49	时间8593、吞吐率1.78	时间11583、吞吐率1.43
10	时间3986、吞吐率4.03	时间6849、吞吐率2.47	时间8649、吞吐率1.78	时间12045、吞吐率1.40	时间14755、吞吐率1.04
15	时间5047、吞吐率3.25	时间9249、吞吐率1.63	时间12330、吞吐率1.30	时间14832、吞吐率1.12	时间18241、吞吐率0.85

## 结论

分析实验结果，可以发现

- 丢包率较低时：由于拥塞控制采用了可变长窗口，本地网络状况等原因，此机制在发送的时候速度是变化的。拥塞控制机制性能比非拥塞控制机制性能稍逊，随着延时增加，差距逐渐减小
- 丢包率较高时。拥塞控制是可变长窗口，受丢包率的影响比定长窗口小，而非拥塞控制受路由延时的影响比拥塞控制的大，所以此时拥塞控制性能比非拥塞控制性能稍好。加速比为1.5倍左右，这一加速比随路由延时的增加变化不大。

