

流完成时间实验

学号： 2021K8009929010

姓名： 贾城昊

一、 实验题目：流完成时间实验

二、 实验任务

1. 利用 fct_exp.py 脚本复现讲义上的图

每个数据点做 5 次实验,取均值

2. 调研解释图中的现象。

提示:TCP 传输、慢启动机制

三、 实验流程

1. 在 Python 脚本中设定带宽及延迟
2. 在终端中输入 `sudo python fct_exp.py`
3. 在终端中输入 `xterm h1 h2` 中启动 h1、h2 两个 host
4. 在 h2 终端输入 `dd if=/dev/zero of=file_sizeMB.dat bs=file_sizeM count=1`,其中 file_size 分别设置为 1, 10, 100 进行不同大小的实验
5. 在 h1 终端中输入 `wget http://10.0.0.2/file_sizeMB.dat` 获取主机 h2 上对应大小的文件
6. 记录每次完成传输的时间和速度,每个数据点做五次实验,取均值
7. 根据结果绘图,复现讲义上的图

四、实验结果

（一）带宽为 10Mbps 下的实验结果（延迟为 100ms）

1. 1MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|-------|-------|-------|-------|-------|-------|
| 时间(s) | 1.5 | 1.5 | 1.5 | 1.5 | 1.5 | 1.5 |
| 速度(MB/s) | 0.673 | 0.672 | 0.677 | 0.678 | 0.680 | 0.676 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/1MB.dat
--2023-09-15 18:13:06-- http://10.0.0.2/1MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：1048576 (1.0M) [application/octet-stream]
正在保存至：“1MB.dat.16”

1MB.dat.16          100%[=====>]  1.00M  673KB/s   in 1.5s
2023-09-15 18:13:08 (673 KB/s) - 已保存 “1MB.dat.16” [1048576/1048576]
```

2. 10MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 9.5 | 9.5 | 9.5 | 9.5 | 9.5 | 9.5 |
| 速度(MB/s) | 1.12 | 1.14 | 1.13 | 1.13 | 1.13 | 1.13 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/10MB.dat
--2023-09-15 18:15:27-- http://10.0.0.2/10MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
内容度：10485760 (10M) [application/octet-stream]
正在保存至：“10MB.dat.21”

10MB.dat.21      100%[=====>] 10.00M  1.12MB/s   in 9.5s
2023-09-15 18:15:37 (1.05 MB/s) - 已保存 “10MB.dat.21” [10485760/10485760]
```

3. 100MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 89 | 89 | 89 | 89 | 89 | 89 |
| 速度(MB/s) | 1.13 | 1.13 | 1.12 | 1.11 | 1.12 | 1.12 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/100MB.dat
--2023-09-15 18:22:30-- http://10.0.0.2/100MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
内容度：104857600 (100M) [application/octet-stream]
正在保存至：“100MB.dat.18”

100MB.dat.18      100%[=====>] 100.00M  1.12MB/s   in 89s
2023-09-15 18:24:00 (1.12 MB/s) - 已保存 “100MB.dat.18” [104857600/104857600]
root@Computer:~# wget http://10.0.0.2/100MB.dat
```

(二) 带宽为 50Mbps 下的实验结果（延迟为 100ms）

1. 1MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|-------|-------|-------|-------|-------|-------|
| 时间(s) | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 |
| 速度(MB/s) | 0.830 | 0.827 | 0.829 | 0.826 | 0.828 | 0.828 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/1MB.dat
--2023-09-15 18:39:17-- http://10.0.0.2/1MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：1048576 (1.0M) [application/octet-stream]
正在保存至：“1MB.dat.25”

1MB.dat.25          100%[=====>]  1.00M  828KB/s   in 1.2s
```

2. 10MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 3.0 | 2.9 | 3.0 | 2.9 | 2.9 | 2.9 |
| 速度(MB/s) | 3.38 | 3.42 | 3.39 | 3.42 | 3.40 | 3.40 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/10MB.dat
--2023-09-15 18:40:11-- http://10.0.0.2/10MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：10485760 (10M) [application/octet-stream]
正在保存至: “10MB.dat.30”

10MB.dat.30      100%[=====>] 10.00M  3.40MB/s   in 2.9s
2023-09-15 18:40:15 (3.40 MB/s) - 已保存 “10MB.dat.30” [10485760/10485760]

```

3. 100MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 19 | 19 | 19 | 19 | 19 | 19 |
| 速度(MB/s) | 5.66 | 5.61 | 5.56 | 5.60 | 5.57 | 5.60 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/100MB.dat
--2023-09-15 18:42:40-- http://10.0.0.2/100MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：104857600 (100M) [application/octet-stream]
正在保存至: “100MB.dat.25”

100MB.dat.25      100%[=====>] 100.00M  5.57MB/s   in 19s
2023-09-15 18:42:59 (5.25 MB/s) - 已保存 “100MB.dat.25” [104857600/104857600]

```

(三) 带宽为 100Mbps 下的实验结果（延迟为 100ms）

1. 1MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|-------|-------|-------|-------|-------|-------|
| 时间(s) | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 |
| 速度(MB/s) | 0.835 | 0.836 | 0.836 | 0.834 | 0.832 | 0.835 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/1MB.dat
--2023-09-15 17:57:11-- http://10.0.0.2/1MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：1048576 (1.0M) [application/octet-stream]
正在保存至：“1MB.dat.12”

1MB.dat.12          100%[=====>]  1.00M  836KB/s   in 1.2s
2023-09-15 17:57:12 (836 KB/s) - 已保存 “1MB.dat.12” [1048576/1048576]
```

2. 10MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 2.2 | 2.2 | 2.2 | 2.2 | 2.2 | 2.2 |
| 速度(MB/s) | 5.50 | 5.54 | 5.50 | 5.52 | 5.49 | 5.51 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/10MB.dat
--2023-09-15 18:04:12-- http://10.0.0.2/10MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：10485760 (10M) [application/octet-stream]
正在保存至：“10MB.dat.20”

10MB.dat.20          100%[=====>]  10.00M  4.49MB/s   in 2.2s
2023-09-15 18:04:15 (4.49 MB/s) - 已保存 “10MB.dat.20” [10485760/10485760]
```

3. 100MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 11 | 11 | 11 | 11 | 11 | 11 |
| 速度(MB/s) | 10.8 | 10.8 | 10.3 | 10.9 | 10.9 | 10.7 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/100MB.dat
--2023-09-15 18:07:37-- http://10.0.0.2/100MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：104857600 (100M) [application/octet-stream]
正在保存至：“100MB.dat.15”

100MB.dat.15      100%[=====>] 100.00M  10.9MB/s   in 11s
2023-09-15 18:07:48 (9.49 MB/s) - 已保存 “100MB.dat.15” [104857600/104857600]
```

（四）带宽为 100Mbps 下的实验结果（延迟为 100ms）

1. 1MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|-------|-------|-------|-------|-------|-------|
| 时间(s) | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 |
| 速度(MB/s) | 0.841 | 0.841 | 0.842 | 0.840 | 0.839 | 0.840 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/1MB.dat
--2023-09-15 18:51:00-- http://10.0.0.2/1MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：1048576 (1.0M) [application/octet-stream]
正在保存至：“1MB.dat.26”

1MB.dat.26          100%[=====>]  1.00M  841KB/s   in 1.2s
2023-09-15 18:51:02 (841 KB/s) - 已保存 “1MB.dat.26” [1048576/1048576]

```

2. 10MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 1.9 | 1.9 | 1.9 | 1.9 | 1.9 | 1.9 |
| 速度(MB/s) | 5.35 | 5.35 | 5.34 | 5.29 | 5.34 | 5.33 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/10MB.dat
--2023-09-15 18:57:32-- http://10.0.0.2/10MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：10485760 (10M) [application/octet-stream]
正在保存至：“10MB.dat.36”

10MB.dat.36          100%[=====>]  10.00M  5.34MB/s   in 1.9s
2023-09-15 18:57:35 (5.34 MB/s) - 已保存 “10MB.dat.36” [10485760/10485760]

```

3. 100MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----|---|---|---|---|---|----|
|----|---|---|---|---|---|----|

| | | | | | | |
|----------|------|------|------|------|------|------|
| 时间(s) | 4.9 | 4.9 | 4.9 | 4.9 | 4.9 | 4.9 |
| 速度(MB/s) | 27.1 | 27.2 | 24.6 | 26.8 | 25.7 | 26.3 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/100MB.dat
--2023-09-15 19:00:29-- http://10.0.0.2/100MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：104857600 (100M) [application/octet-stream]
正在保存至：“100MB.dat.35”

100MB.dat.35      100%[=====>] 100.00M  25.7MB/s   in 4.9s
2023-09-15 19:00:34 (20.5 MB/s) - 已保存 “100MB.dat.35” [104857600/104857600]

```

（五）带宽为 1Gbps 下的实验结果（延迟为 100ms）

1. 1MB 数据包传输结果

进行 5 次实验，结果如下：

| | | | | | | |
|----------|-------|-------|-------|-------|-------|-------|
| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
| 时间(s) | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 | 1.2 |
| 速度(MB/s) | 0.841 | 0.848 | 0.843 | 0.842 | 0.848 | 0.834 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/1MB.dat
--2023-09-15 19:04:20-- http://10.0.0.2/1MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：1048576 (1.0M) [application/octet-stream]
正在保存至：“1MB.dat.31”

1MB.dat.31      100%[=====>]  1.00M   841KB/s   in 1.2s
2023-09-15 19:04:22 (841 KB/s) - 已保存 “1MB.dat.31” [1048576/1048576]

```

2. 10MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 1.8 | 1.8 | 1.8 | 1.8 | 1.8 | 1.8 |
| 速度(MB/s) | 5.42 | 5.42 | 5.42 | 5.42 | 5.43 | 5.42 |

其中一次传输结果如下：

```
root@Computer:~# wget http://10.0.0.2/10MB.dat
--2023-09-15 19:04:41-- http://10.0.0.2/10MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：10485760 (10M) [application/octet-stream]
正在保存至：“10MB.dat.39”

10MB.dat.39          100%[=====>] 10.00M  5.42MB/s   in 1.8s
2023-09-15 19:04:43 (5.42 MB/s) - 已保存 “10MB.dat.39” [10485760/10485760]
```

3. 100MB 数据包传输结果

进行 5 次实验，结果如下：

| 序号 | 1 | 2 | 3 | 4 | 5 | 均值 |
|----------|------|------|------|------|------|------|
| 时间(s) | 4.1 | 4.1 | 4.2 | 4.1 | 4.1 | 4.1 |
| 速度(MB/s) | 24.2 | 24.3 | 24.8 | 24.5 | 24.5 | 24.5 |

其中一次传输结果如下：

```

root@Computer:~# wget http://10.0.0.2/100MB.dat
--2023-09-15 19:05:25-- http://10.0.0.2/100MB.dat
正在连接 10.0.0.2:80... 已连接。
已发出 HTTP 请求，正在等待回... 200 OK
长度：104857600 (100M) [application/octet-stream]
正在保存至：“100MB.dat.40”

100MB.dat.40          100%[=====>] 100.00M  24.5MB/s   in 4.1s
2023-09-15 19:05:29 (24.5 MB/s) - 已保存 “100MB.dat.40” [104857600/104857600]

```

（六）延迟为 10ms 时，1MB 10MB 100MB 数据包在不同带宽下的传输情况

下面每个数据均是五次测试取平均值后的结果，鉴于篇幅原因，原始数据在此不做具体展示

1. 1MB 数据包

| 带宽(Mbps) | 10 | 50 | 100 | 500 | 1000 |
|----------|------|------|------|------|------|
| 时间(s) | 0.9 | 0.2 | 0.2 | 0.2 | 0.2 |
| 速度(MB/s) | 1.11 | 4.33 | 5.89 | 5.96 | 6.22 |

2. 10MB 数据包

| 带宽(Mbps) | 10 | 50 | 100 | 500 | 1000 |
|----------|------|------|------|------|------|
| 时间(s) | 8.9 | 1.9 | 1.0 | 0.4 | 0.4 |
| 速度(MB/s) | 1.12 | 5.22 | 9.93 | 24.4 | 25.0 |

3. 100MB 数据包

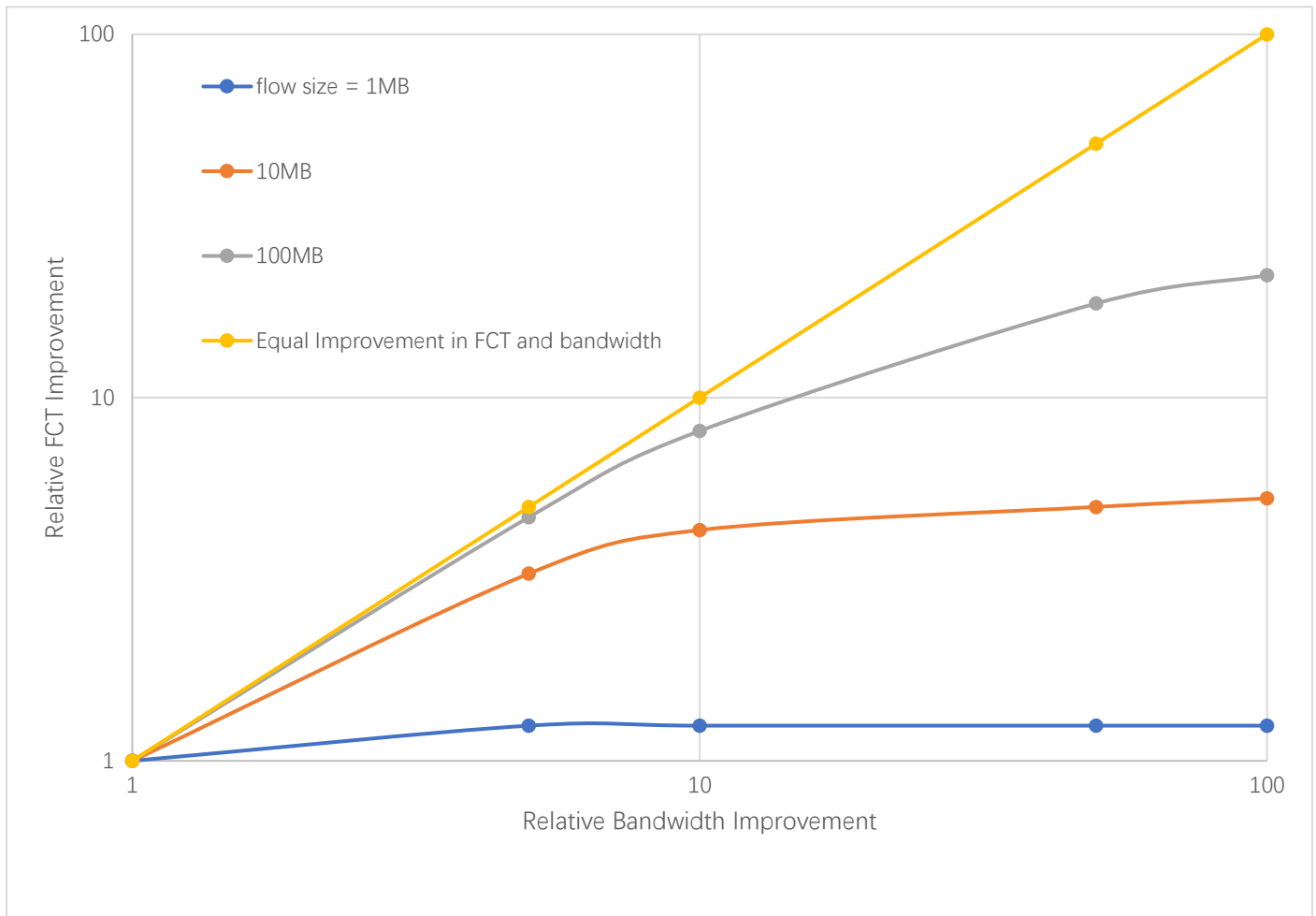
| | | | | | |
|----------|------|------|------|------|------|
| 带宽(Mbps) | 10 | 50 | 100 | 500 | 1000 |
| 时间(s) | 89 | 18 | 9.5 | 2.2 | 1.5 |
| 速度(MB/s) | 1.12 | 5.51 | 10.9 | 46.5 | 67.2 |

从上面三个表可以看出可以看到,在相同条件,延迟为 10ms 时的传输速度明显快于延迟为 100ms 时的传输速度,在带宽较大时更为明显,这点将在稍后的分析中解释。

五、 结果分析

1. 绘图：

要复现讲义上的图,首先要使用对数坐标系,横纵坐标是带宽和 FCT 的改进,对于纵坐标,我们将最长的耗时作为分子,每个耗时的数据作为分母,以此作为纵坐标来体现 FCT 的改进(可以理解为每个点的传输速率和最慢速率的比值的倒数) ,横坐标则是每个带宽和最小带宽的比值,这样定义的横纵坐标可以用来表示带宽的改进和 FCT 的改进,绘图如下:



可以看出，该图与讲义上要求复现的图的结果基本一致

下面是上图以及改变延迟所得到的结果中出现的几个现象的总结：

1. 当带宽到达 50Mbps 时，1MB 的包速率不再发生变化（时间耗时均为 1.2s）。
2. 当带宽一定时,数据包越大,网络传输速率越高,数据包越小,网络传输速率越低
3. 当数据包大小不变时,网络传输速率不会随着带宽线性增加（特别的当带宽达到 500Mbps 后,传输速率基本不再增加）。
4. 在改变延迟的实验中发现,减小延迟可以显著增加高带宽、大数据包的传输速率,同时根据实验中观察到的现象,在数据包开始传输的一段时间内速度是较慢的,而这段时间的长短与延迟成正相关。

下面是本人进行的一些调研，以及对上述现象的解释：

六、 调研以及对结果解释

1. TCP 传输

TCP 协议会将应用层的数据流分割成适当长度的报文段，最大传输段大小 (MSS)通常受该计算机连接的网路的数据链路层的最大传送单元 (MTU) 限制。而 TCP 的传输速率是由其阻塞算法决定的,TCP 拥塞算法缓慢地探测网路的可用带宽,增加传输速率直到检测到分组丢失，然后指数地降低传输速率。

当数据包大小不变带宽增加时,该算法会增加传输速率直至分组丢失,而降低传输速率时指数级的,因此速率并不会随着带宽的增加而线性增加。

同时,对数据进行分组也会造成丢包、排队、阻塞等问题,这也会影响到传输速率的增长。

2. 慢启动机制

慢启动是 TCP 使用的一种阻塞控制机制。慢启动也叫做指数增长期。慢启动是指每次 TCP 接收窗口收到确认时都会增长。增加的大小就是已确认段的数目。这种情况一直保持到要么没有收到一些段,要么窗口大小到达预先定义的阈值。如果发生丢失事件, TCP 就认为这是网络阻塞,就会采取措施减轻网络拥挤。一旦发生丢失事件或者到达阈值, TCP 就会进入线性增长阶段。这时,每经过一个 RTT 窗口增长一个段。

由于 TCP 连接会随着时间进行自我调谐,起初会限制连接的最大速度,如果数据传输成功,会随着时间的推移提高传输速度,这就是 TCP 的慢启动机制。

这样就解释了在带宽较高时,小数据包没有达到期待的网速的问题。在慢启动阶段, TCP 预留的窗口大小会随着每接受到一个段而指数级增长,对于数据包大小较小的包,在窗

口还没有达到带宽的阈值时可能传输就已经结束了,因此此时测得的传输速率会明显小于对应带宽的最大速率。

七、 实验总结

这次实验使用了实际的数据,给我带来的直观且深刻的印象。加深了我对 TCP 协议的认识,也让我进一步熟悉了实验环境的使用,同时对影响网速的因素有了浅显的认识。