

数据中心间流量的动态调度及 拥塞控制

安徽大学

指导老师：崔 杰

小组成员：房亚明、龚冉
詹志宏、张洪
王祥如

目录

CONTENTS



Part1

流量动态分级调度



Part2

拥塞控制多级反馈队列算法



Part3

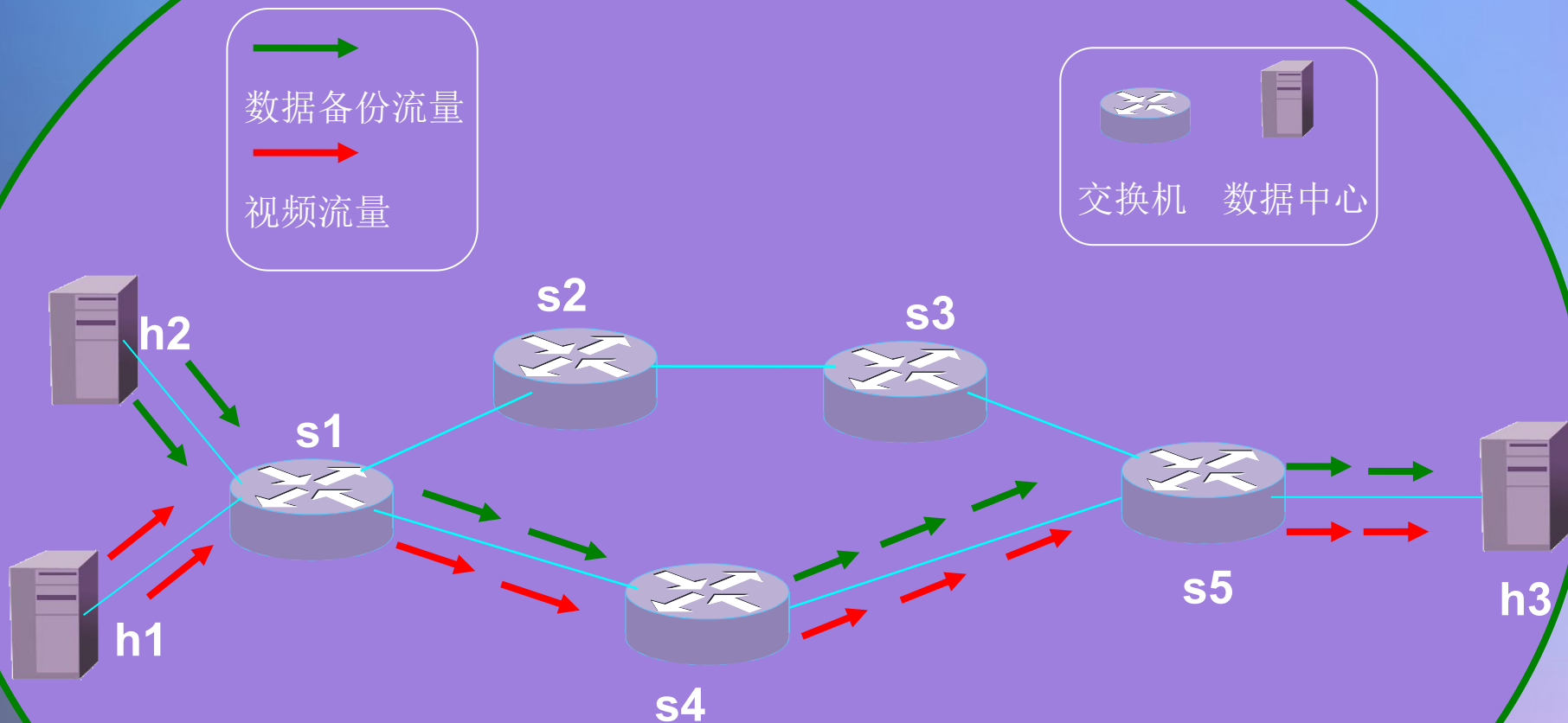
总结与展望

Part 1

流量动态分级调度

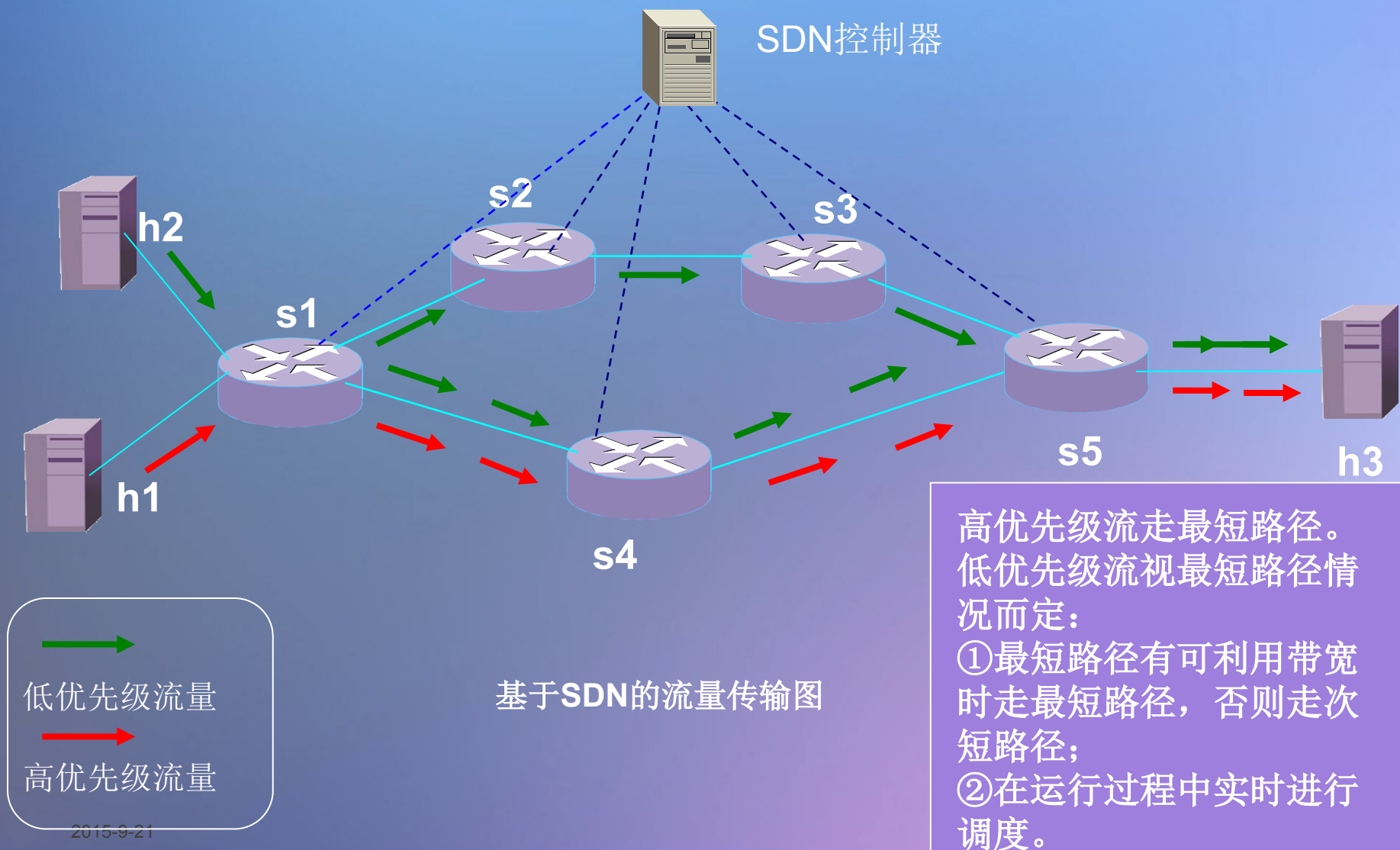
Traffic Dynamic Hierarchy Schedule

背景



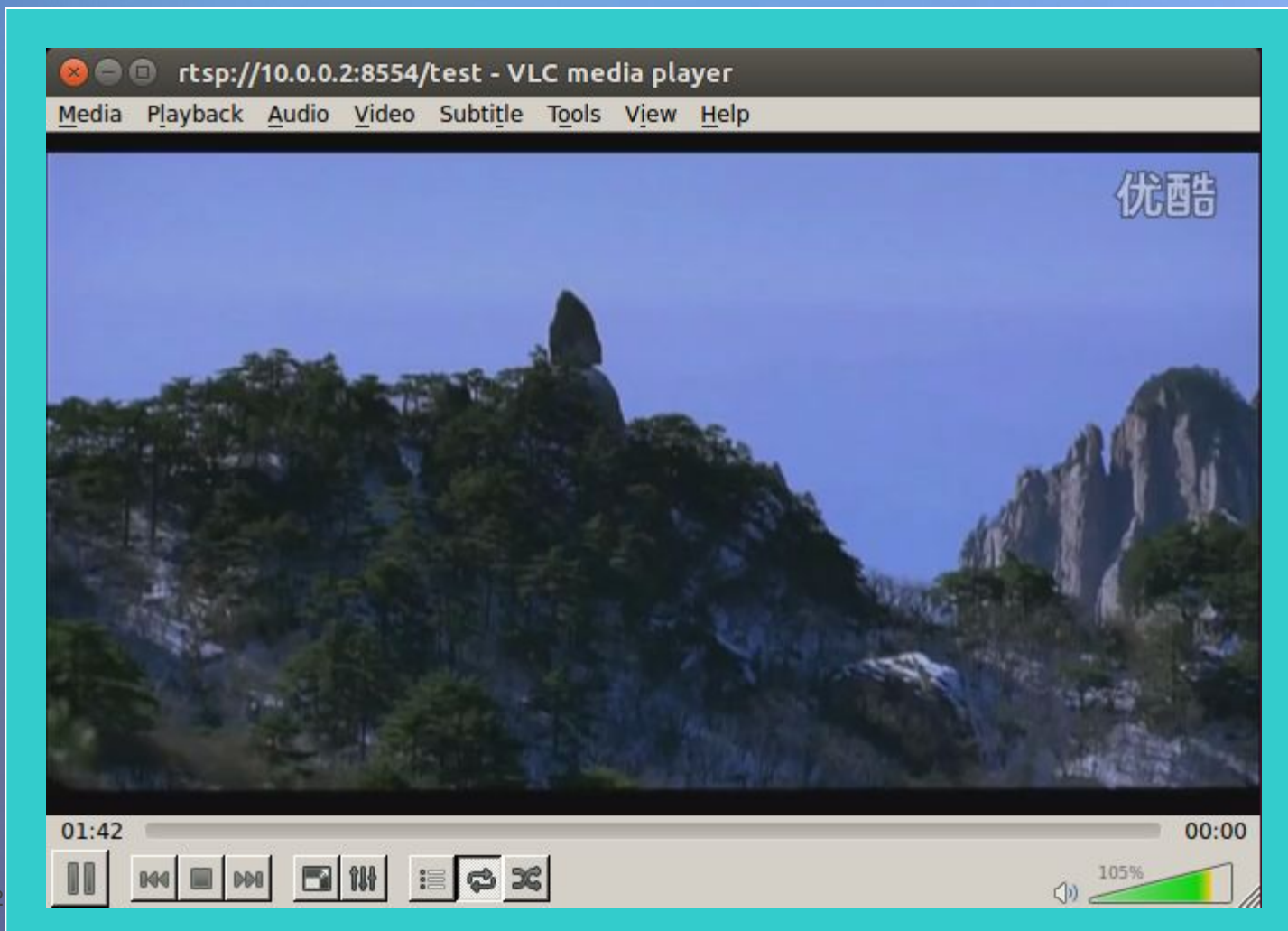
传统模式下流量传输图

基于SDN的动态分级调度



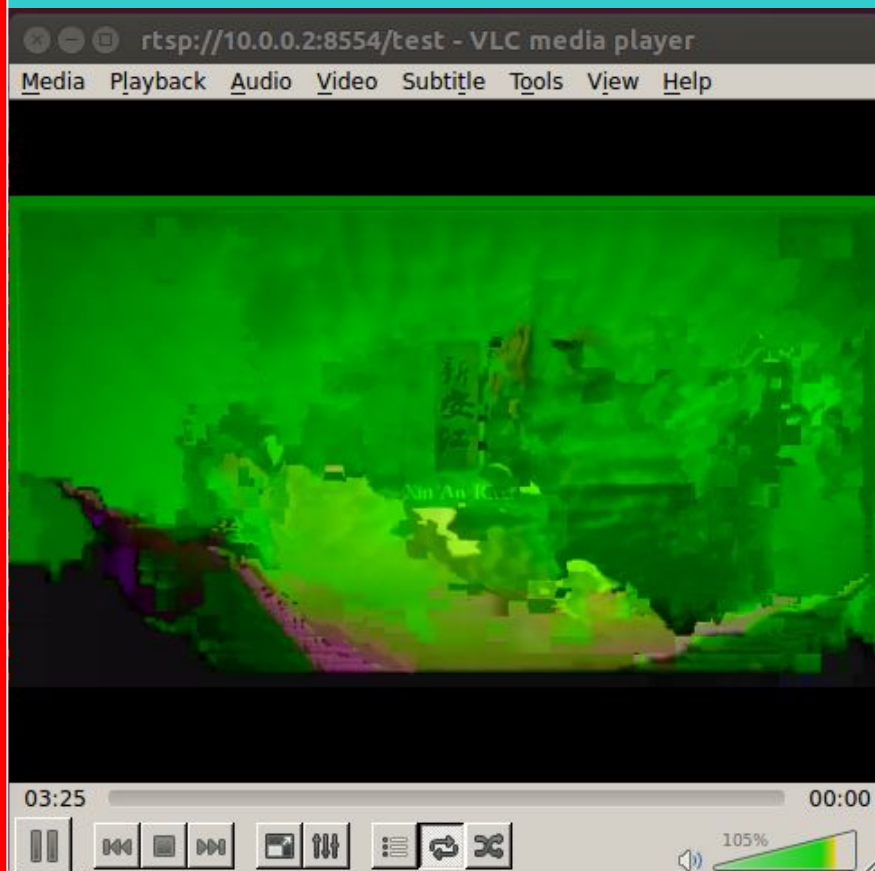
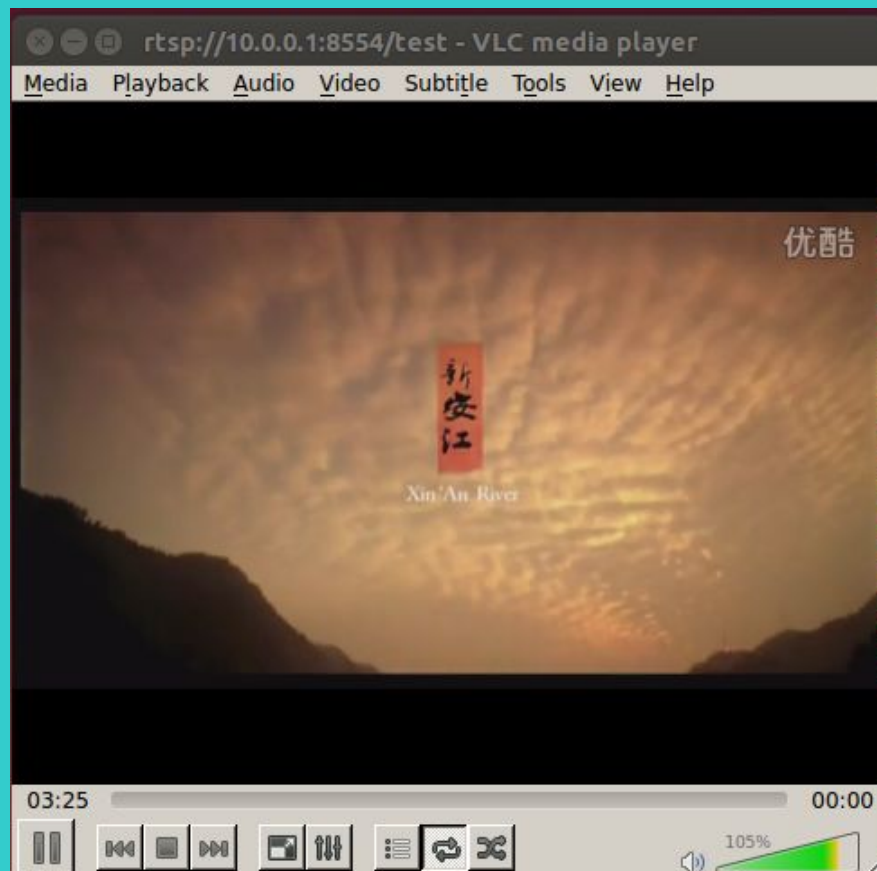
实验演示

1. 当最短路径有可利用带宽时，低优先级流走最短路径。实验效果图如下所示：



实验演示

2. 最短路径上有高优先级流，没有可利用带宽时，低优先级流为高优先级流让出最短路径。实验效果图如下所示：



实验演示

3. 当高优先级流退出时，低优先级流被调度到最短路径。实验效果图如下所示：



Part 2

拥塞控制多级反馈队列算法

Congestion Control Multilevel Feedback Queue Algorithm

传统拥塞控制队列

01 FIFO
先入先出队列

02 PQ
优先级队列

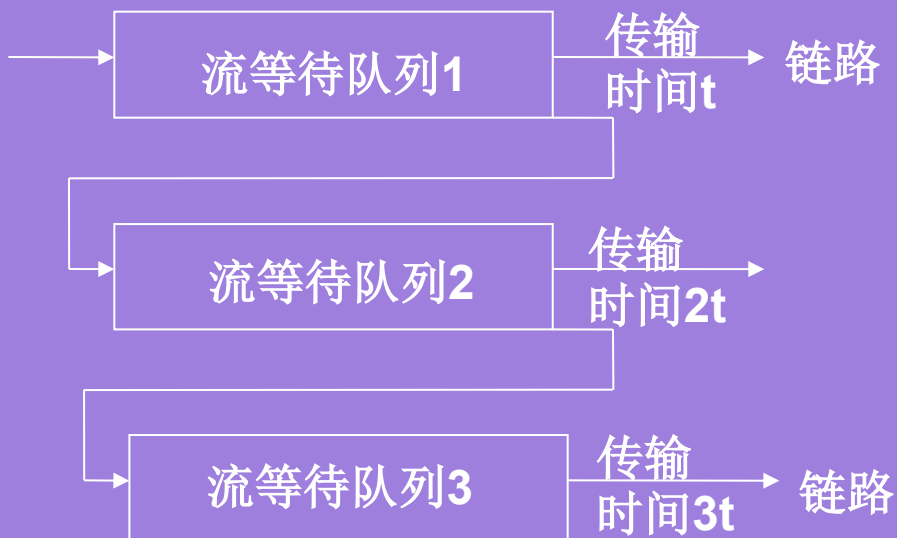
03 CQ
用户定制队列

04 WFQ
加权公平队列

传统方案的不足：

- ① 需要对每个数据包进行复杂的队列控制
- ② 需要提前设置好策略，缺乏灵活性
- ③ 不能解决某些流量长时间占有队列问题

拥塞控制多级反馈队列算法



(1) 首先将等待传输的流放入队列1的末尾，按FCFS原则排队调度。

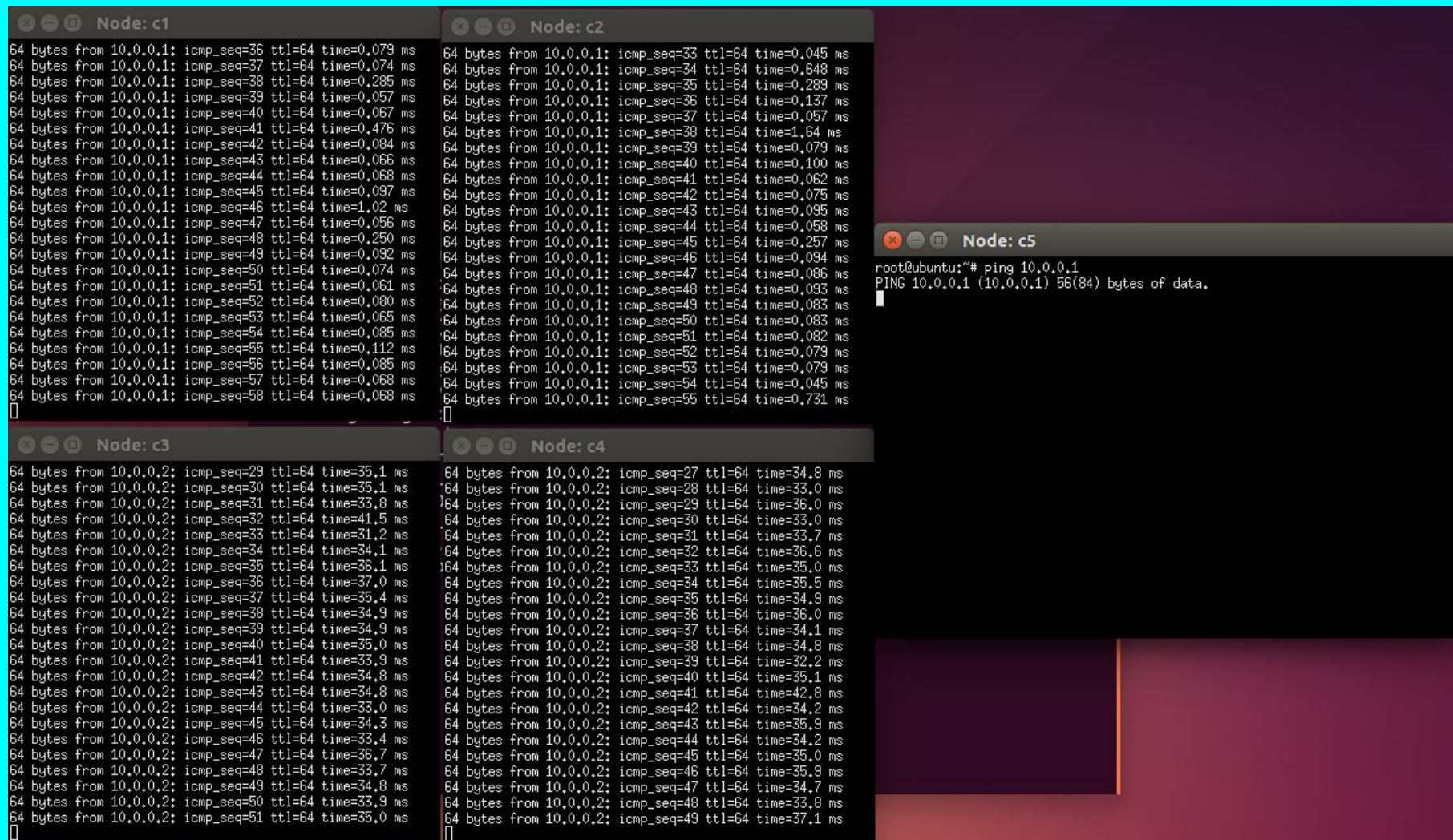
(2) 当队列1中的流被调度到链路进行传输时，其所拥有的链路传输时间为 t 。若其在时间 t 内未完成传输，将其调度到队列2的末尾，以此类推。

(3) 仅当队列1空闲时才调度队列2中的流。队列2空闲时才调度队列3中的流。

- 直接对流进行控制，实现简单
- 对流进行多级调度，能满足各种类型流的需求
- 能够防止某些流恶意长时间占用带宽

实验演示

1. 刚产生拥塞时，c5处于等待状态。实验效果图如下所示：



```
Node: c1
64 bytes from 10.0.0.1: icmp_seq=36 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=37 ttl=64 time=0.074 ms
64 bytes from 10.0.0.1: icmp_seq=38 ttl=64 time=0.285 ms
64 bytes from 10.0.0.1: icmp_seq=39 ttl=64 time=0.057 ms
64 bytes from 10.0.0.1: icmp_seq=40 ttl=64 time=0.067 ms
64 bytes from 10.0.0.1: icmp_seq=41 ttl=64 time=0.476 ms
64 bytes from 10.0.0.1: icmp_seq=42 ttl=64 time=0.084 ms
64 bytes from 10.0.0.1: icmp_seq=43 ttl=64 time=0.066 ms
64 bytes from 10.0.0.1: icmp_seq=44 ttl=64 time=0.068 ms
64 bytes from 10.0.0.1: icmp_seq=45 ttl=64 time=0.097 ms
64 bytes from 10.0.0.1: icmp_seq=46 ttl=64 time=1.02 ms
64 bytes from 10.0.0.1: icmp_seq=47 ttl=64 time=0.056 ms
64 bytes from 10.0.0.1: icmp_seq=48 ttl=64 time=0.250 ms
64 bytes from 10.0.0.1: icmp_seq=49 ttl=64 time=0.092 ms
64 bytes from 10.0.0.1: icmp_seq=50 ttl=64 time=0.074 ms
64 bytes from 10.0.0.1: icmp_seq=51 ttl=64 time=0.061 ms
64 bytes from 10.0.0.1: icmp_seq=52 ttl=64 time=0.080 ms
64 bytes from 10.0.0.1: icmp_seq=53 ttl=64 time=0.065 ms
64 bytes from 10.0.0.1: icmp_seq=54 ttl=64 time=0.085 ms
64 bytes from 10.0.0.1: icmp_seq=55 ttl=64 time=0.112 ms
64 bytes from 10.0.0.1: icmp_seq=56 ttl=64 time=0.085 ms
64 bytes from 10.0.0.1: icmp_seq=57 ttl=64 time=0.068 ms
64 bytes from 10.0.0.1: icmp_seq=58 ttl=64 time=0.068 ms

Node: c2
64 bytes from 10.0.0.1: icmp_seq=33 ttl=64 time=0.045 ms
64 bytes from 10.0.0.1: icmp_seq=34 ttl=64 time=0.648 ms
64 bytes from 10.0.0.1: icmp_seq=35 ttl=64 time=0.289 ms
64 bytes from 10.0.0.1: icmp_seq=36 ttl=64 time=0.137 ms
64 bytes from 10.0.0.1: icmp_seq=37 ttl=64 time=0.057 ms
64 bytes from 10.0.0.1: icmp_seq=38 ttl=64 time=1.64 ms
64 bytes from 10.0.0.1: icmp_seq=39 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=40 ttl=64 time=0.100 ms
64 bytes from 10.0.0.1: icmp_seq=41 ttl=64 time=0.062 ms
64 bytes from 10.0.0.1: icmp_seq=42 ttl=64 time=0.075 ms
64 bytes from 10.0.0.1: icmp_seq=43 ttl=64 time=0.095 ms
64 bytes from 10.0.0.1: icmp_seq=44 ttl=64 time=0.058 ms
64 bytes from 10.0.0.1: icmp_seq=45 ttl=64 time=0.257 ms
64 bytes from 10.0.0.1: icmp_seq=46 ttl=64 time=0.094 ms
64 bytes from 10.0.0.1: icmp_seq=47 ttl=64 time=0.086 ms
64 bytes from 10.0.0.1: icmp_seq=48 ttl=64 time=0.093 ms
64 bytes from 10.0.0.1: icmp_seq=49 ttl=64 time=0.083 ms
64 bytes from 10.0.0.1: icmp_seq=50 ttl=64 time=0.083 ms
64 bytes from 10.0.0.1: icmp_seq=51 ttl=64 time=0.082 ms
64 bytes from 10.0.0.1: icmp_seq=52 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=53 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=54 ttl=64 time=0.045 ms
64 bytes from 10.0.0.1: icmp_seq=55 ttl=64 time=0.731 ms

Node: c3
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=35.1 ms
64 bytes from 10.0.0.2: icmp_seq=30 ttl=64 time=35.1 ms
64 bytes from 10.0.0.2: icmp_seq=31 ttl=64 time=33.8 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=41.5 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=31.2 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=34.1 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=36.1 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=37.0 ms
64 bytes from 10.0.0.2: icmp_seq=37 ttl=64 time=35.4 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=35.0 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=33.9 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=33.0 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=34.3 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=33.4 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=36.7 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=33.7 ms
64 bytes from 10.0.0.2: icmp_seq=49 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=33.9 ms
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=35.0 ms

Node: c4
64 bytes from 10.0.0.2: icmp_seq=27 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=28 ttl=64 time=33.0 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=36.0 ms
64 bytes from 10.0.0.2: icmp_seq=30 ttl=64 time=33.0 ms
64 bytes from 10.0.0.2: icmp_seq=31 ttl=64 time=33.7 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=36.6 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=35.0 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=35.5 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=36.0 ms
64 bytes from 10.0.0.2: icmp_seq=37 ttl=64 time=34.1 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=32.2 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=35.1 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=42.8 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=34.2 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=35.9 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=34.2 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=35.0 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=35.9 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=34.7 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=33.8 ms
64 bytes from 10.0.0.2: icmp_seq=49 ttl=64 time=37.1 ms

Node: c5
root@ubuntu:~# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
```


实验演示

2. 一段时间后，c5被调度运行，c2被调入拥塞队列。实验效果图如下所示：

```
Node: c1
64 bytes from 10.0.0.1: icmp_seq=55 ttl=64 time=0.112 ms
64 bytes from 10.0.0.1: icmp_seq=56 ttl=64 time=0.085 ms
64 bytes from 10.0.0.1: icmp_seq=57 ttl=64 time=0.068 ms
64 bytes from 10.0.0.1: icmp_seq=58 ttl=64 time=0.068 ms
64 bytes from 10.0.0.1: icmp_seq=59 ttl=64 time=0.083 ms
64 bytes from 10.0.0.1: icmp_seq=60 ttl=64 time=1.01 ms
64 bytes from 10.0.0.1: icmp_seq=61 ttl=64 time=0.058 ms
64 bytes from 10.0.0.1: icmp_seq=65 ttl=64 time=0.378 ms
64 bytes from 10.0.0.1: icmp_seq=66 ttl=64 time=0.072 ms
64 bytes from 10.0.0.1: icmp_seq=67 ttl=64 time=0.078 ms
64 bytes from 10.0.0.1: icmp_seq=68 ttl=64 time=0.974 ms
64 bytes from 10.0.0.1: icmp_seq=69 ttl=64 time=0.071 ms
64 bytes from 10.0.0.1: icmp_seq=70 ttl=64 time=0.062 ms
64 bytes from 10.0.0.1: icmp_seq=71 ttl=64 time=0.541 ms
64 bytes from 10.0.0.1: icmp_seq=72 ttl=64 time=0.064 ms
64 bytes from 10.0.0.1: icmp_seq=73 ttl=64 time=0.054 ms
64 bytes from 10.0.0.1: icmp_seq=74 ttl=64 time=0.413 ms
64 bytes from 10.0.0.1: icmp_seq=75 ttl=64 time=0.060 ms
64 bytes from 10.0.0.1: icmp_seq=76 ttl=64 time=0.094 ms
64 bytes from 10.0.0.1: icmp_seq=77 ttl=64 time=0.110 ms
64 bytes from 10.0.0.1: icmp_seq=78 ttl=64 time=0.076 ms
64 bytes from 10.0.0.1: icmp_seq=79 ttl=64 time=0.061 ms
64 bytes from 10.0.0.1: icmp_seq=80 ttl=64 time=0.054 ms

Node: c2
64 bytes from 10.0.0.1: icmp_seq=39 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=40 ttl=64 time=0.100 ms
64 bytes from 10.0.0.1: icmp_seq=41 ttl=64 time=0.062 ms
64 bytes from 10.0.0.1: icmp_seq=42 ttl=64 time=0.075 ms
64 bytes from 10.0.0.1: icmp_seq=43 ttl=64 time=0.095 ms
64 bytes from 10.0.0.1: icmp_seq=44 ttl=64 time=0.058 ms
64 bytes from 10.0.0.1: icmp_seq=45 ttl=64 time=0.257 ms
64 bytes from 10.0.0.1: icmp_seq=46 ttl=64 time=0.094 ms
64 bytes from 10.0.0.1: icmp_seq=47 ttl=64 time=0.086 ms
64 bytes from 10.0.0.1: icmp_seq=48 ttl=64 time=0.093 ms
64 bytes from 10.0.0.1: icmp_seq=49 ttl=64 time=0.083 ms
64 bytes from 10.0.0.1: icmp_seq=50 ttl=64 time=0.083 ms
64 bytes from 10.0.0.1: icmp_seq=51 ttl=64 time=0.082 ms
64 bytes from 10.0.0.1: icmp_seq=52 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=53 ttl=64 time=0.079 ms
64 bytes from 10.0.0.1: icmp_seq=54 ttl=64 time=0.045 ms
64 bytes from 10.0.0.1: icmp_seq=55 ttl=64 time=0.731 ms
64 bytes from 10.0.0.1: icmp_seq=56 ttl=64 time=0.080 ms
64 bytes from 10.0.0.1: icmp_seq=57 ttl=64 time=0.110 ms
64 bytes from 10.0.0.1: icmp_seq=58 ttl=64 time=0.093 ms
64 bytes from 10.0.0.1: icmp_seq=59 ttl=64 time=0.204 ms
64 bytes from 10.0.0.1: icmp_seq=60 ttl=64 time=0.056 ms
64 bytes from 10.0.0.1: icmp_seq=61 ttl=64 time=0.095 ms

Node: c3
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=35.0 ms
64 bytes from 10.0.0.2: icmp_seq=52 ttl=64 time=35.0 ms
64 bytes from 10.0.0.2: icmp_seq=53 ttl=64 time=44.2 ms
64 bytes from 10.0.0.2: icmp_seq=54 ttl=64 time=36.0 ms
64 bytes from 10.0.0.2: icmp_seq=55 ttl=64 time=36.9 ms
64 bytes from 10.0.0.2: icmp_seq=56 ttl=64 time=35.8 ms
64 bytes from 10.0.0.2: icmp_seq=57 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=58 ttl=64 time=43.0 ms
64 bytes from 10.0.0.2: icmp_seq=59 ttl=64 time=38.4 ms
64 bytes from 10.0.0.2: icmp_seq=60 ttl=64 time=35.2 ms
64 bytes from 10.0.0.2: icmp_seq=61 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=62 ttl=64 time=35.2 ms
64 bytes from 10.0.0.2: icmp_seq=63 ttl=64 time=36.4 ms
64 bytes from 10.0.0.2: icmp_seq=64 ttl=64 time=33.2 ms
64 bytes from 10.0.0.2: icmp_seq=65 ttl=64 time=34.5 ms
64 bytes from 10.0.0.2: icmp_seq=66 ttl=64 time=34.0 ms
64 bytes from 10.0.0.2: icmp_seq=67 ttl=64 time=33.0 ms
64 bytes from 10.0.0.2: icmp_seq=68 ttl=64 time=34.0 ms
64 bytes from 10.0.0.2: icmp_seq=69 ttl=64 time=35.2 ms
64 bytes from 10.0.0.2: icmp_seq=70 ttl=64 time=36.0 ms
64 bytes from 10.0.0.2: icmp_seq=71 ttl=64 time=33.1 ms
64 bytes from 10.0.0.2: icmp_seq=72 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=73 ttl=64 time=35.5 ms

Node: c4
64 bytes from 10.0.0.2: icmp_seq=49 ttl=64 time=37.1 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=31.4 ms
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=33.9 ms
64 bytes from 10.0.0.2: icmp_seq=52 ttl=64 time=34.9 ms
64 bytes from 10.0.0.2: icmp_seq=53 ttl=64 time=44.8 ms
64 bytes from 10.0.0.2: icmp_seq=54 ttl=64 time=35.9 ms
64 bytes from 10.0.0.2: icmp_seq=55 ttl=64 time=33.9 ms
64 bytes from 10.0.0.2: icmp_seq=56 ttl=64 time=37.3 ms
64 bytes from 10.0.0.2: icmp_seq=57 ttl=64 time=35.9 ms
64 bytes from 10.0.0.2: icmp_seq=58 ttl=64 time=33.9 ms
64 bytes from 10.0.0.2: icmp_seq=59 ttl=64 time=37.1 ms
64 bytes from 10.0.0.2: icmp_seq=60 ttl=64 time=35.1 ms
64 bytes from 10.0.0.2: icmp_seq=61 ttl=64 time=34.0 ms
64 bytes from 10.0.0.2: icmp_seq=62 ttl=64 time=35.2 ms
64 bytes from 10.0.0.2: icmp_seq=63 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=64 ttl=64 time=34.0 ms
64 bytes from 10.0.0.2: icmp_seq=65 ttl=64 time=34.8 ms
64 bytes from 10.0.0.2: icmp_seq=66 ttl=64 time=33.2 ms
64 bytes from 10.0.0.2: icmp_seq=67 ttl=64 time=37.3 ms
64 bytes from 10.0.0.2: icmp_seq=68 ttl=64 time=34.7 ms
64 bytes from 10.0.0.2: icmp_seq=69 ttl=64 time=34.1 ms
64 bytes from 10.0.0.2: icmp_seq=70 ttl=64 time=35.1 ms
64 bytes from 10.0.0.2: icmp_seq=71 ttl=64 time=34.7 ms

Node: c5
root@ubuntu:~# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=7 ttl=64 time=0.609 ms
64 bytes from 10.0.0.1: icmp_seq=8 ttl=64 time=0.076 ms
64 bytes from 10.0.0.1: icmp_seq=9 ttl=64 time=0.057 ms
64 bytes from 10.0.0.1: icmp_seq=10 ttl=64 time=0.280 ms
64 bytes from 10.0.0.1: icmp_seq=11 ttl=64 time=0.103 ms
64 bytes from 10.0.0.1: icmp_seq=12 ttl=64 time=1.59 ms
64 bytes from 10.0.0.1: icmp_seq=13 ttl=64 time=0.526 ms
64 bytes from 10.0.0.1: icmp_seq=14 ttl=64 time=0.089 ms
64 bytes from 10.0.0.1: icmp_seq=15 ttl=64 time=0.069 ms
64 bytes from 10.0.0.1: icmp_seq=16 ttl=64 time=0.338 ms
64 bytes from 10.0.0.1: icmp_seq=17 ttl=64 time=0.080 ms
64 bytes from 10.0.0.1: icmp_seq=18 ttl=64 time=0.096 ms
64 bytes from 10.0.0.1: icmp_seq=19 ttl=64 time=0.086 ms
64 bytes from 10.0.0.1: icmp_seq=20 ttl=64 time=0.085 ms
64 bytes from 10.0.0.1: icmp_seq=21 ttl=64 time=0.100 ms
64 bytes from 10.0.0.1: icmp_seq=22 ttl=64 time=0.074 ms
64 bytes from 10.0.0.1: icmp_seq=23 ttl=64 time=0.061 ms
64 bytes from 10.0.0.1: icmp_seq=24 ttl=64 time=0.085 ms
```

Part 3

总结与展望

Conclusion & Future

总结

流量动态分级调度方案：

- ◆ 实时动态进行流的路径选择和调度
- ◆ 充分利用最短路径带宽
- ◆ 高效的流量分级处理

拥塞控制多级反馈队列算法：

- ◆ 创新性将多级反馈队列算法用于拥塞控制中
- ◆ 较传统方案，实现简单
- ◆ 能满足各种类型流的需求
- ◆ 能避免流的恶意传输

展望

- ✓ 为流量提供更加细粒度的分级服务，更加灵活地进行流量调整
- ✓ 增加实验次数，找出更加合理的多级反馈队列时间划分方案
- ✓ 提供界面化的流量管理工具，提高流量管理的直观性和方便性

THANK YOU !