

# 实验报告

孔静 2014K8009929022

September 28, 2017

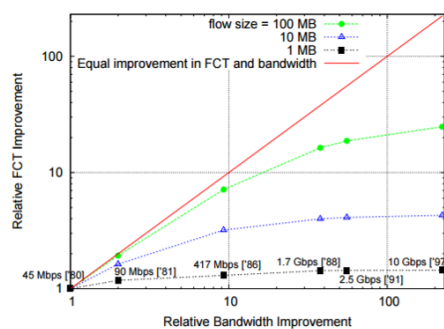
## 1 实验题目

互联网协议与传输性能实验

## 2 实验内容

### 2.1 流完成时间实验

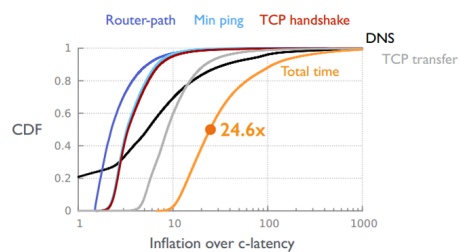
- 利用 `fct_exp.py` 脚本, 重现下图中的实验结果
- 文件大小: 1MB, 10MB, 100MB
- 带宽: 10Mbps, 50Mbps, 100Mbps, 500Mbps, 1Gbps



### 2.2 互联网传输性能实验

- 利用 `timeit.py` 脚本, 重现下图中的实验结果
- 测量 500 个网站

- DNS 解析时间, 建立连接 (handshake) 时间, 数据传输时间



### 3 实验流程

#### 3.1 流完成时间实验

- 修改 fct\_exp.py 脚本中的带宽及执行时文件大小
- 测试并记录结果

大小 \ 带宽	10	50	100	500	1000
1	0.9	0.2	0.2	0.1	0.1
10	8.8	1.8	1.0	0.4	0.3
100	88	18	9.4	2.1	1.2

#### 3.2 互联网传输性能实验

- 修改 plot\_ping\_time.py 脚本, 获取 dns, establish, download 时间

```

1 try:
2     t1 = timeit_dns(url)
3     t2 = timeit_conn(url)
4     t3 = timeit_curl(url)
5     of.write("%s %.4f %.4f %.4f\n", url, t1, t2, t3)
6 except:
7     print 'fail'
8

```

- 运行脚本获取至少 500 个网站的结果数据

#### 3.3 绘制两个实验的图

- 绘制实验 1 的 python 脚本

```

1 def plot_time():
2     bw = [1, 5, 10, 50, 100]
3     time1 = [0.9, 0.2, 0.2, 0.1, 0.1]
4     time10 = [8.8, 1.8, 1, 0.4, 0.3]
5     time100 = [88, 18, 9.4, 2.1, 1.2]
6     for x in [4, 3, 2, 1, 0]:
7         time1[x] = 1. * time1[0] / time1[x]
8         time10[x] = 1. * time10[0] / time10[x]
9         time100[x] = 1. * time100[0] / time100[x]
10    plt.plot(bw, bw, 'r', label='equal improvement in FCT and
        bandwidth')
11    plt.plot(bw, time1, 'k-s', label='1MB')
12    plt.plot(bw, time10, 'b-^', label='10MB')
13    plt.plot(bw, time100, 'g-o', label='100MB')
14    plt.legend(loc='best')
15    plt.xlabel('Relative bandwidth improvement')
16    plt.ylabel('Relative FCT improvement')
17    plt.xlim([1, 100])
18    plt.ylim([1, 100])
19    plt.xscale('log')
20    plt.yscale('log')
21    #plt.show()
22    plt.savefig('pic1.png')
23

```

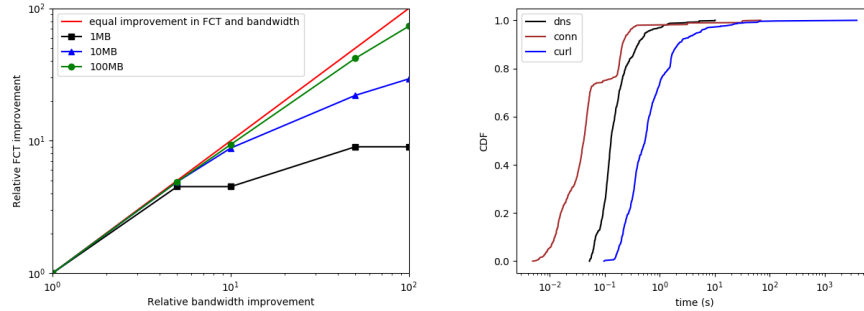
#### • 绘制实验 2 的 python 脚本

```

1 def plot_cdf(fname):
2     dns = list()
3     conn = list()
4     curl = list()
5     for line in open(fname, 'r'):
6         t1 = float(line.strip().split()[1])
7         t2 = float(line.strip().split()[2])
8         t3 = float(line.strip().split()[3])
9         dns.append(t1)
10        conn.append(t2)
11        curl.append(t3)
12    dns.sort()
13    conn.sort()
14    curl.sort()
15    p = 1. * np.arange(len(dns))/(len(dns) - 1)
16    plt.plot(dns, p, color='black', label='dns')
17    p = 1. * np.arange(len(conn))/(len(conn) - 1)
18    plt.plot(conn, p, color='brown', label='conn')
19    p = 1. * np.arange(len(curl))/(len(curl) - 1)
20    plt.plot(curl, p, color='blue', label='curl')
21    plt.xscale('log')
22    plt.legend(loc='best')
23    plt.xlabel('time (s)')
24    plt.ylabel('CDF')
25    #plt.show()
26    plt.savefig('pic2.png')
27

```

## 4 实验结果



## 5 结果分析

### 5.1 流完成时间实验

由图可知，带宽越大，文件传输速率越快，同时，文件越大，越依赖于带宽。当文件大小/带宽增大时，带宽的增长对文件传输速率的影响减小，延迟的影响增大。

图中 1MB 线出现平线，不符合规律，重新测试，发现下载速度有所增长，只是完成时间保留位数问题导致出现平线。

Figure 1: 10Mbps

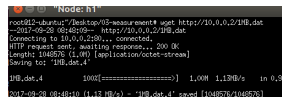


Figure 4: 500Mbps

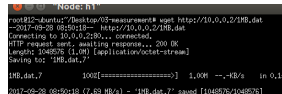


Figure 2: 50Mbps

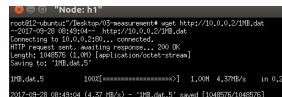


Figure 3: 100Mbps

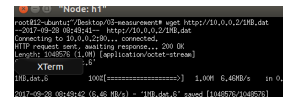
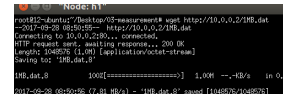


Figure 5: 1Gbps



其中 500Mbps 到 1Gbps 的增速不多，可能是因为文件太小，还未增长至最大速度便已下载完成。

### 5.2 互联网传输性能实验

由图可知，数据传输时间 > DNS 解析时间 > 建立连接 (handshake) 时间。