

# 计算机网络大作业书面报告

陈誉博	<b>2014011058</b>
慕思成	<b>2014011057</b>
祝 放	<b>2014011061</b>
孙国伦	<b>2014011044</b>
张广滨	<b>2014011062</b>

**2016 年 12 月 25 日**

## 目录

实验目的 .....	3
任务分工 .....	3
实验步骤 .....	3
数据收集 .....	3
问题 1:.....	4
问题 2.....	6
问题 3.....	7
问题 4.....	10
问题 5.....	54
实验总结 .....	55
文件清单 .....	55

## 实验目的

用 **windump** 收集某个主机或者路由器所连接的某个物理网络上的 **traffic**，然后用 **excel** 对 **traffic** 的具体特征进行分析。熟悉掌握 **windump** 的使用方法，同时对 **ip** 分组的协议比例，数据分布和部分控制特征、端口特性进行进一步的认识。

## 任务分工

陈誉博	编写程序来解析数据报，统计数据报个数及其特性
孙国伦	通过 <b>windump</b> 抓包，对原始数据进行收集和整理
祝放	对 <b>traffic</b> 中 <b>Output</b> 部分的数据给出 1-5 问的图表
慕思成	对 <b>traffic</b> 中 <b>Input</b> 部分的数据给出 1-5 问的图表
张广滨	帮助绘图，实验数据整理，实验报告撰写

## 实验步骤

### 数据收集

1. 收集时间：2016 年 12 月 11 日 18:50-19:50 星期日，

2. 地点：紫荆 1 号楼

3. 端口号：ip: 59.66.138.32

4. Mac: F0:76:1C:1F:44:8F

5. 使用 **windump** 采集网络 **traffic** 原始数据：

**Windump -i2 -x -n -s 200 -t -w windump.dat ether host F0:76:1C:1F:44:8F**

6. 期间打开过熊猫 TV、斗鱼 TV、清华邮箱、微信、STEAM、战网、腾讯游戏中心，之后 **ctrl+c** 结束抓包。

## 问题 1:

给出 IP 分组携带不同协议的载荷的饼图，分别按分组数和总数据量进行统计

答：由 IPv4 的数据包格式可知，第十个字节表示所使用的协议。具体的对应关系如图 1-1：

十进制	十六进制	关键字	协议
0	0x00	HOPOPT	IPv6 逐跳选项
1	0x01	ICMP	互联网控制消息协议
2	0x02	IGMP	因特网组管理协议
3	0x03	GGP	网关对网关协议
4	0x04	IPv4	IPv4 (封装)
5	0x05	ST	因特网流协议
6	0x06	TCP	传输控制协议
7	0x07	CBT	有核树组播路由协议
8	0x08	EGP	外部网关协议
9	0x09	IGP	内部网关协议 (任意私有内部网关 (用于思科的 IGRP))
10	0x0A	BBN-RCC-MON	BBN RCC 监视
11	0x0B	NVP-II	网络语音协议
12	0x0C	PUP	Xerox PUP
13	0x0D	ARGUS	ARGUS
14	0x0E	EMCON	EMCON
15	0x0F	XNET	Cross Net Debugger
16	0x10	CHAOS	Chaos
17	0x11	UDP	用户数据报协议
18	0x12	MUX	Multiplexing
19	0x13	DCN-MEAS	DCN Measurement Subsystems
20	0x14	HMP	Host Monitoring Protocol

图1-1 IP协议号列表

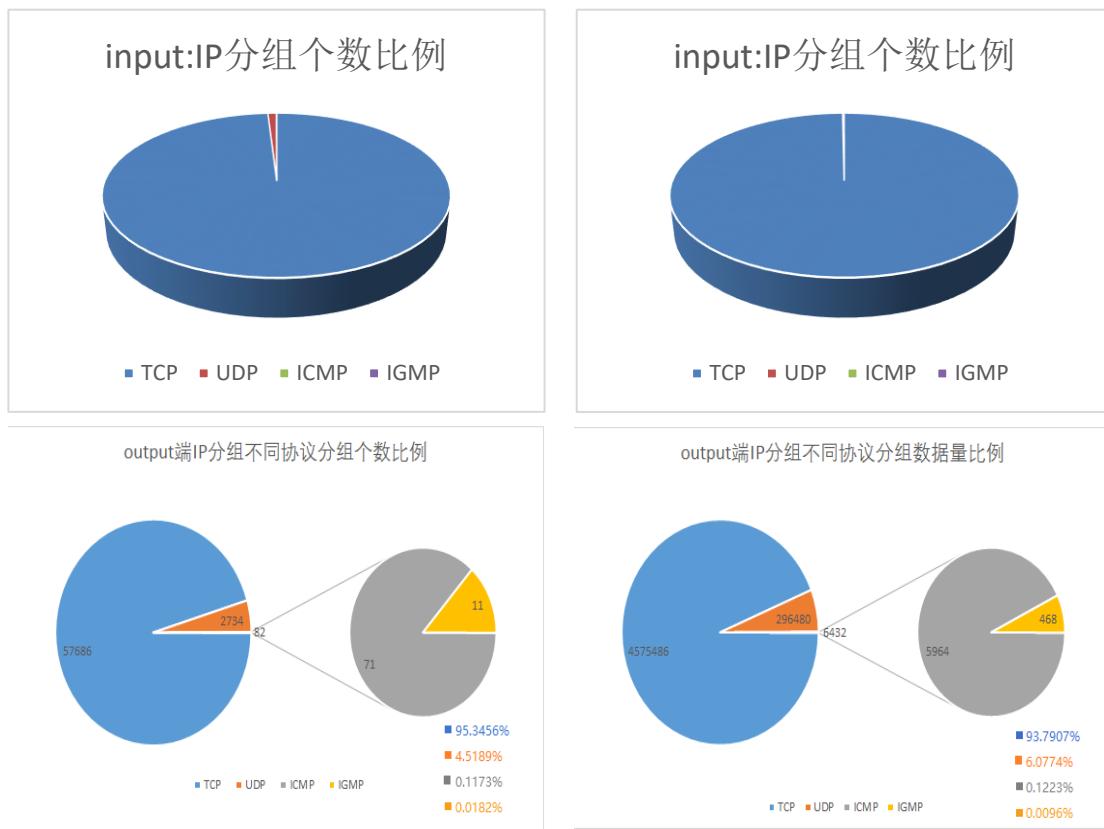
而实际中，我们收集到的数据报大部分是如下几个协议：

十进制	十六进制	关键字	协议
1	0x01	ICMP	互联网控制消息协议
2	0x02	IGMP	因特网组管理协议
6	0x06	TCP	传输控制协议
17	0x11	UDP	用户数据报协议

图1-2 收集到的IP协议号列表

通过excel对数据进行分析和整理，得到了图1-3：

图 1-3 接收和发送 IP 分组携带协议比例统计



从图 1-3 中不难看出，在 output 和 input 两个方向上，UDP 和 TCP 占据绝大多数，另外有少量的 ICMP 和 IGMP 的数据包。此外，在 input 和 output 两个方向上的数据包数量和比例都是 TCP 占有绝大多数。

## 问题 2

有多少 IP 分组是片段(fragment)? 有多少 IP 数据报被分片? 载荷为 TCP 和 UDP 的分别有多少比例的 IP 数据报被分片?

答: 对于分片的判断可以通过 IP 包头的标志位, 见图 2-1

标志

0	DF	MF
---	----	----

标志字段共 3 位, 最高位为 0, 该值必须复制到所有分组中;

不分片 (DF) 值: DF=1, 表示接收结点不能对分组分片; DF=0, 表示可以分片;

分片 (MF) 值: MF=1 表示接收的分片不是最后一个分片, MF=0 表示接收的是最后一个分片。

图2-1 IP头标志字段

### 实验结果:

#### Input和output:

片段的 IP 分组个数: 0, 被分段的 IP 数据报个数: 0, 被分段的载荷 TCP 的 IP 数据报个数: 0, 被分段的载荷 UDP 的 IP 数据报个数: 0

通过实验数据不难看出, 无论 TCP 还是 UDP, 数据几乎不存在分片。对于 TCP 来讲, TCP 是避免分片的, 因为当在 IP 层进行了分片后, 如果其中的某片数据丢失, 则需对整个数据报进行重传。因为 IP 层本身没有超时重传机制, 当来自 TCP 报文段的某一片丢失后, TCP 在超时后重发整个 TCP 报文段, 该报文段对应于一份 IP 数据报, 没有办法只重传数据报中的一个数据报片。而且如果对数据报分片的是中间路由器, 而不是起始端系统, 那么起始端系统就无法知道数据报是如何被分片的, 因此基于这种原因, TCP 是经常要避免分片的。

使用 UDP 则容易导致分片, 但是在本实验中体现不明显, 具体原因正如老师上课所说, 目前网络状态良好, 网络荷载量足够大, 一般不需要再分片处理。

### 问题 3

给出 IP 数据报长度的累积分布曲线，并分别比较载荷为 TCP 和 UDP 的 IP 数据报长度的累积分布

答：通过 excel 可以根据 IP 数据报长度画出累积分布曲线：

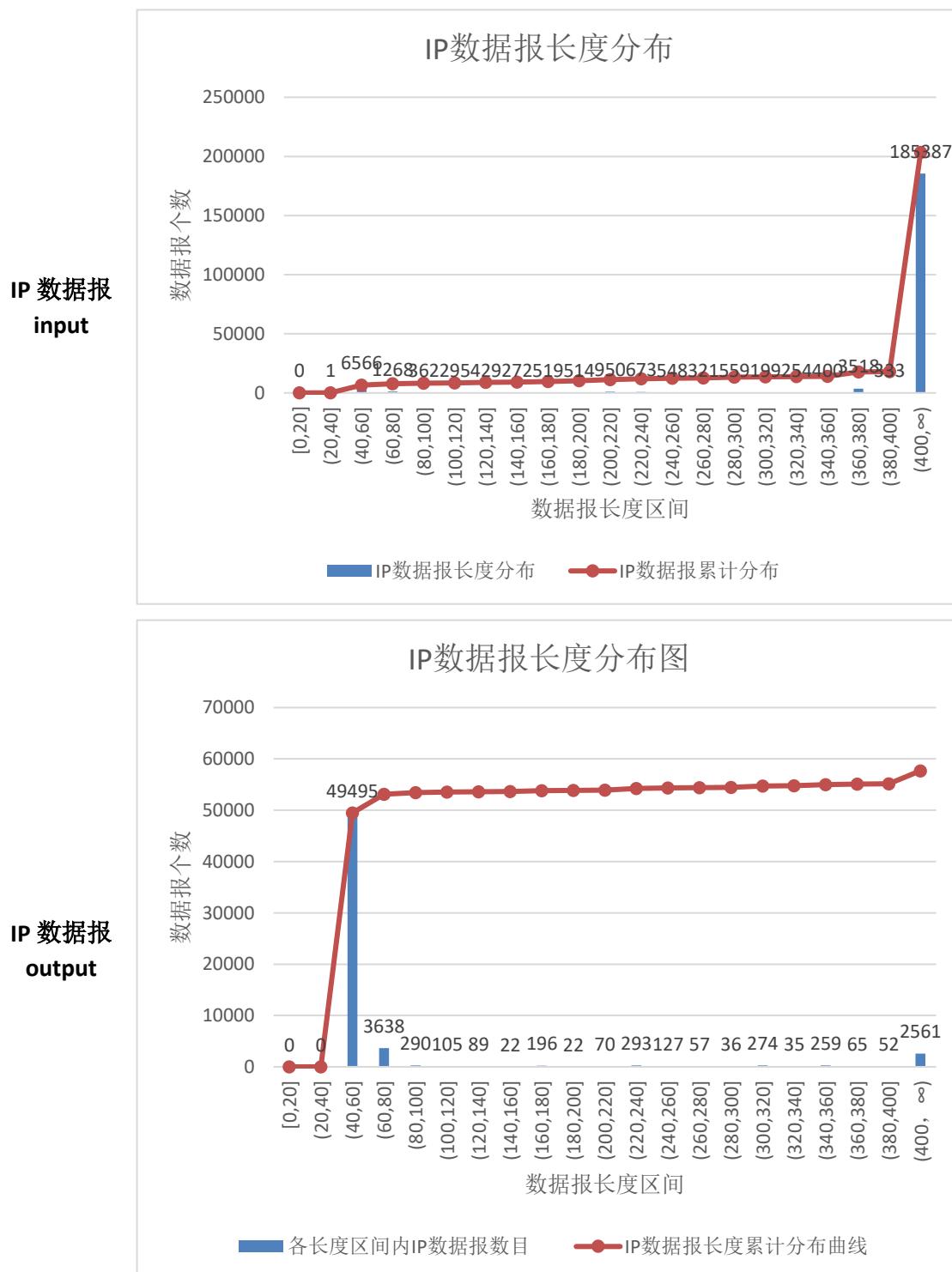
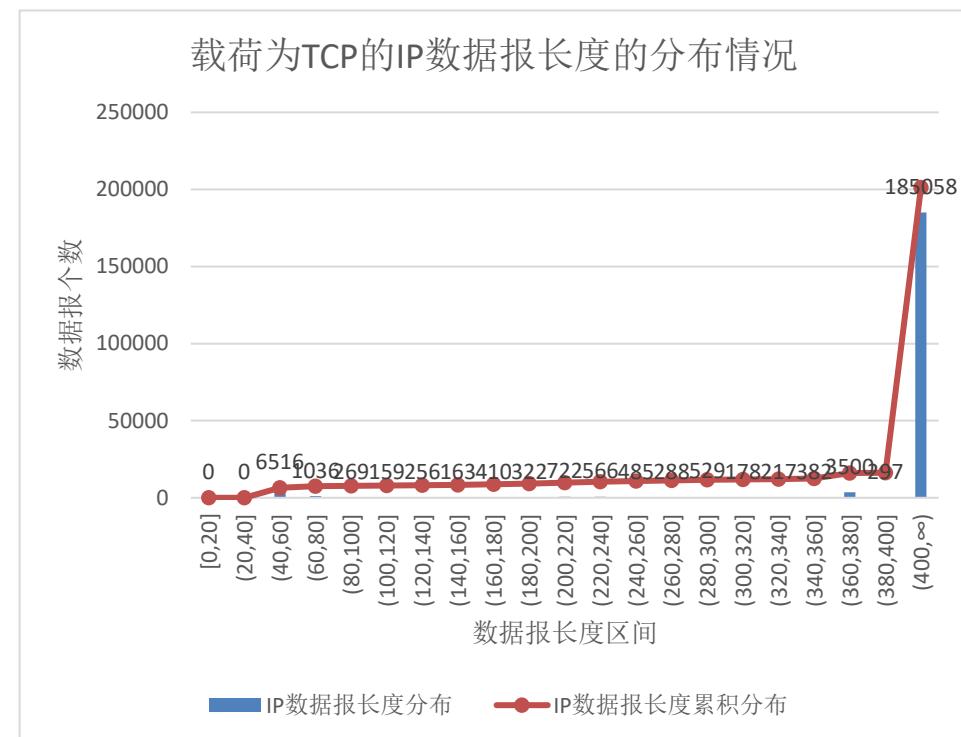


图 3-1 IP 数据报长度累积分布曲线

荷载为  
TCP 的 IP  
数据报  
input



荷载为  
TCP 的 IP  
数据报  
output

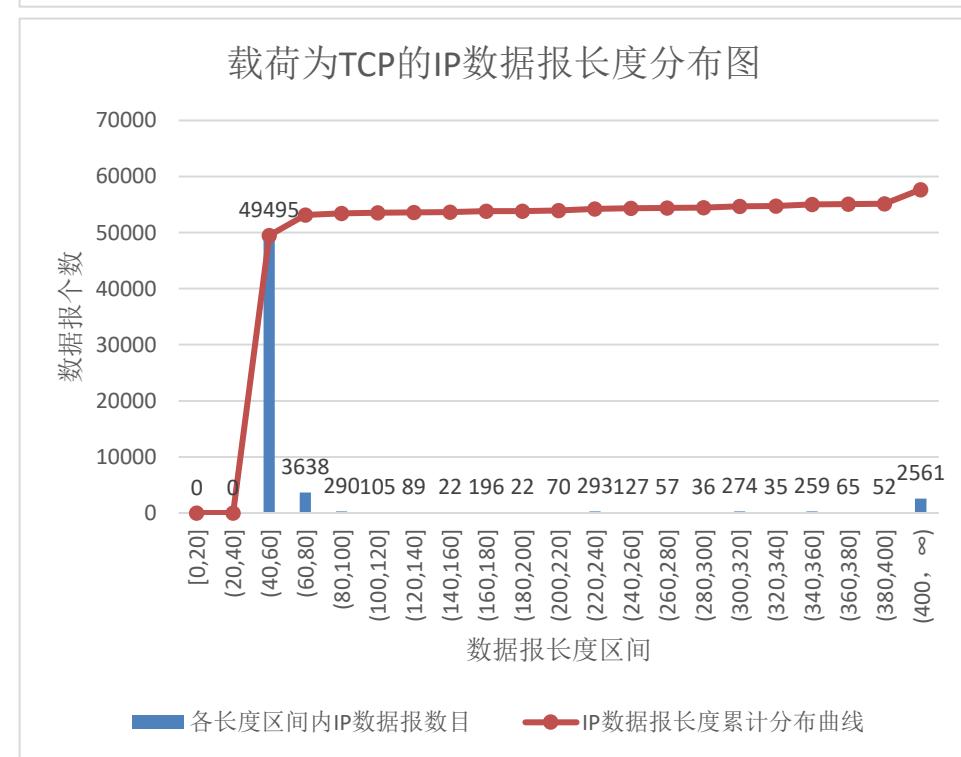


图 3-2input 和 output TCP 协议数据报长度累积分布统计

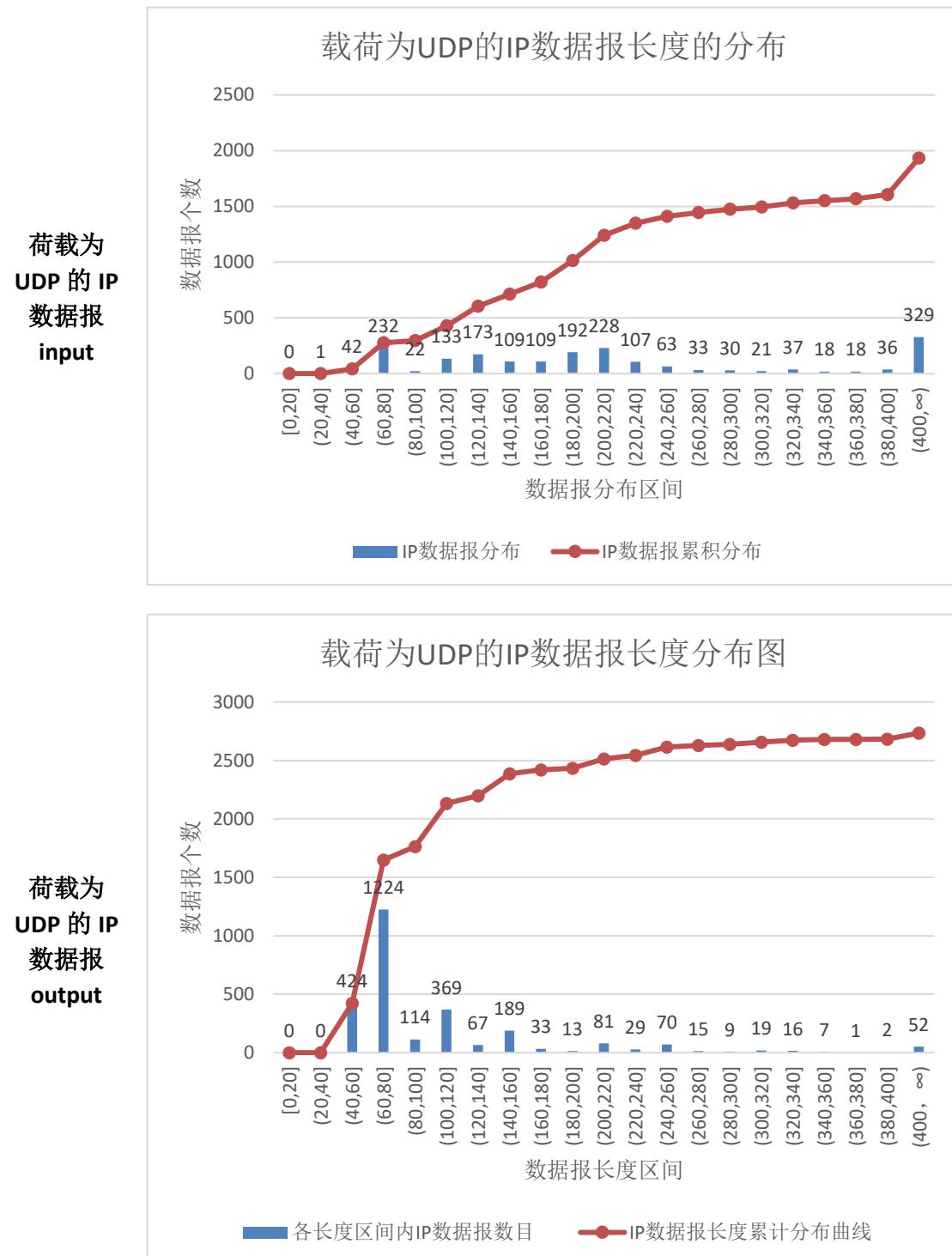
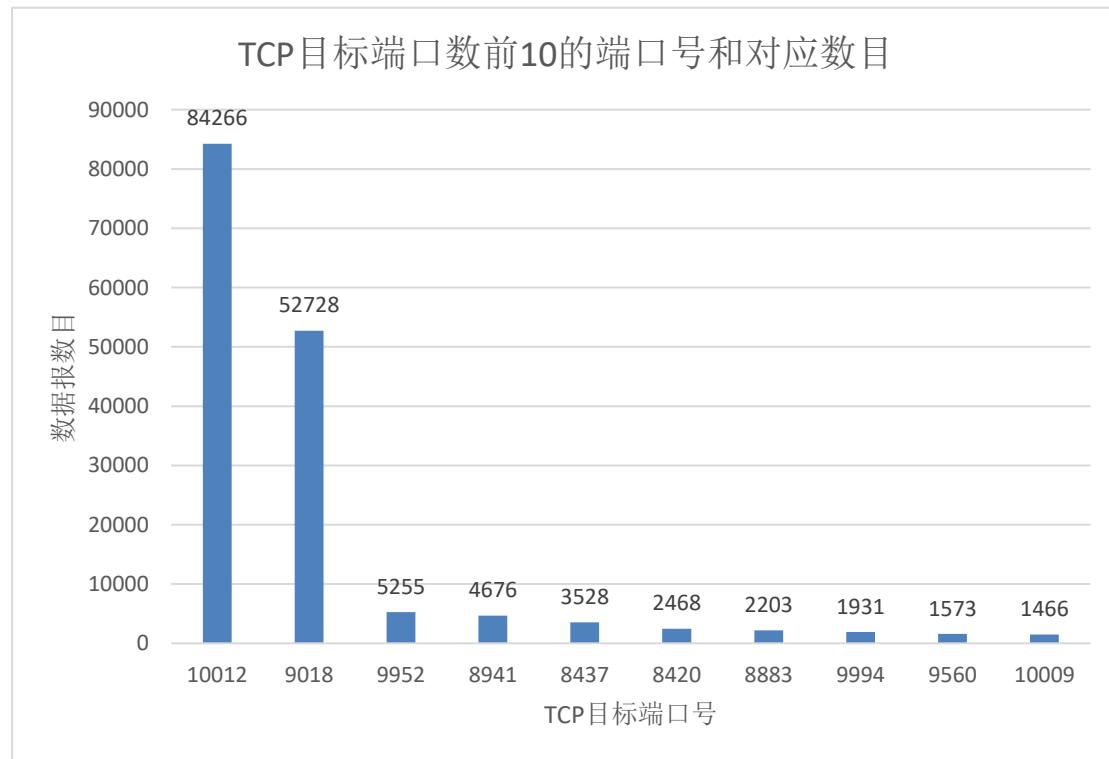
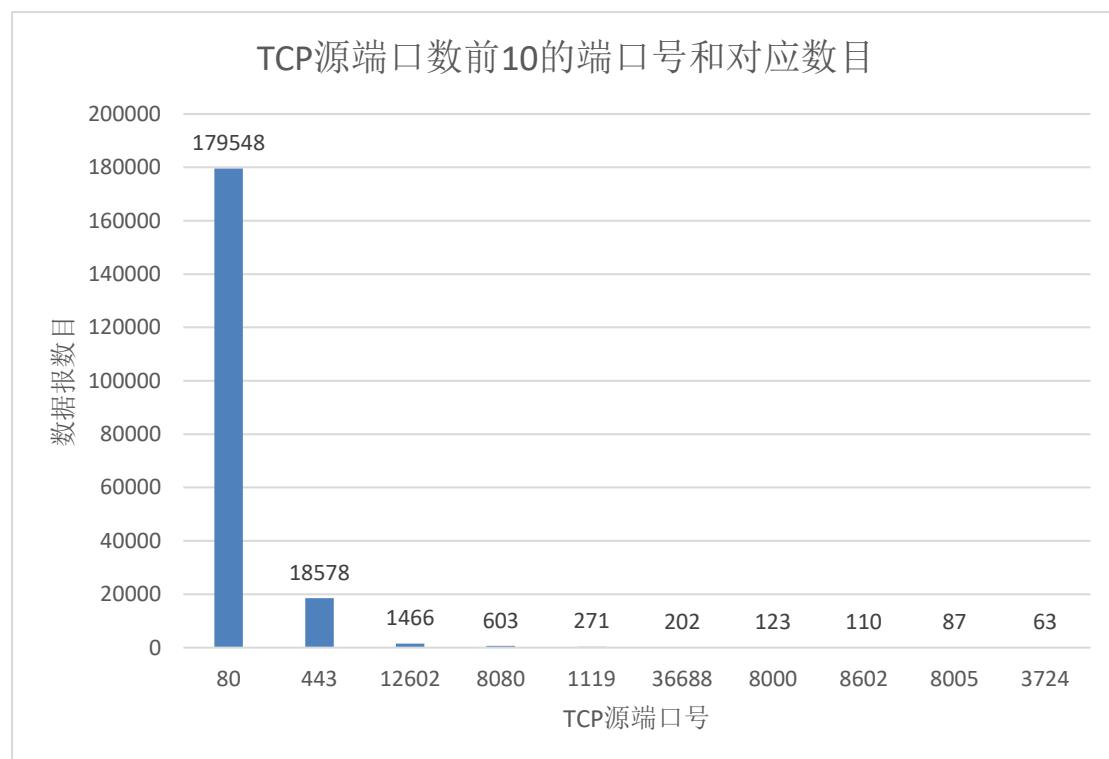


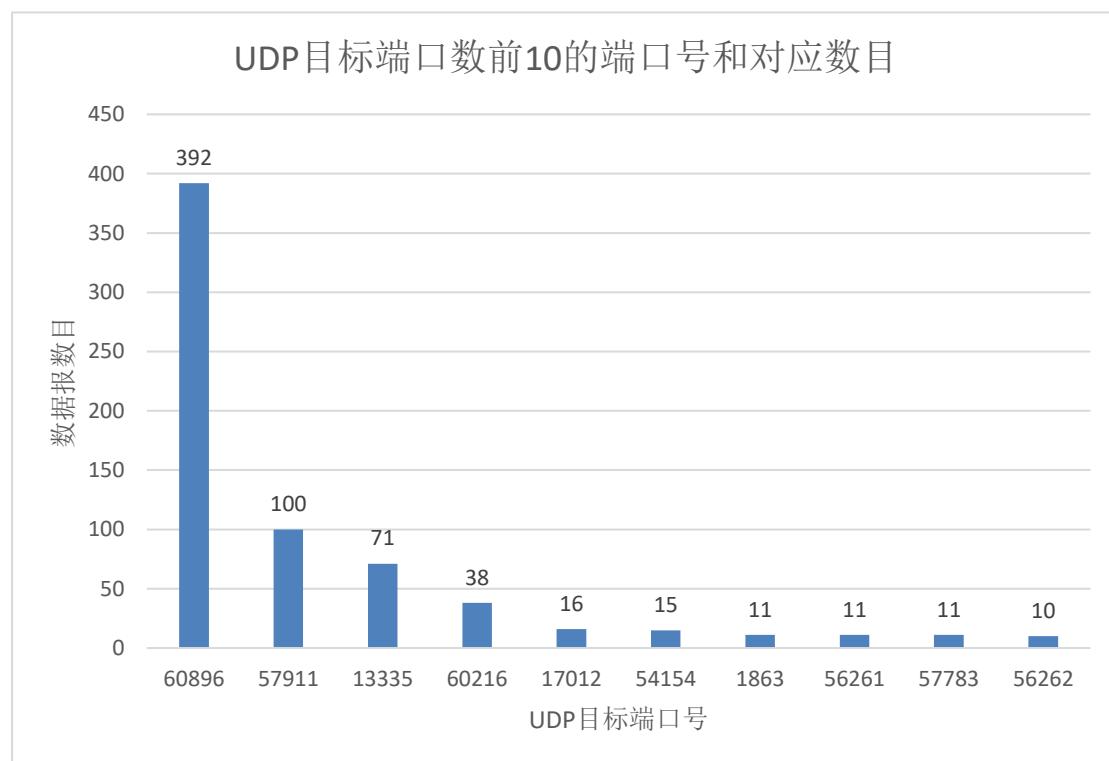
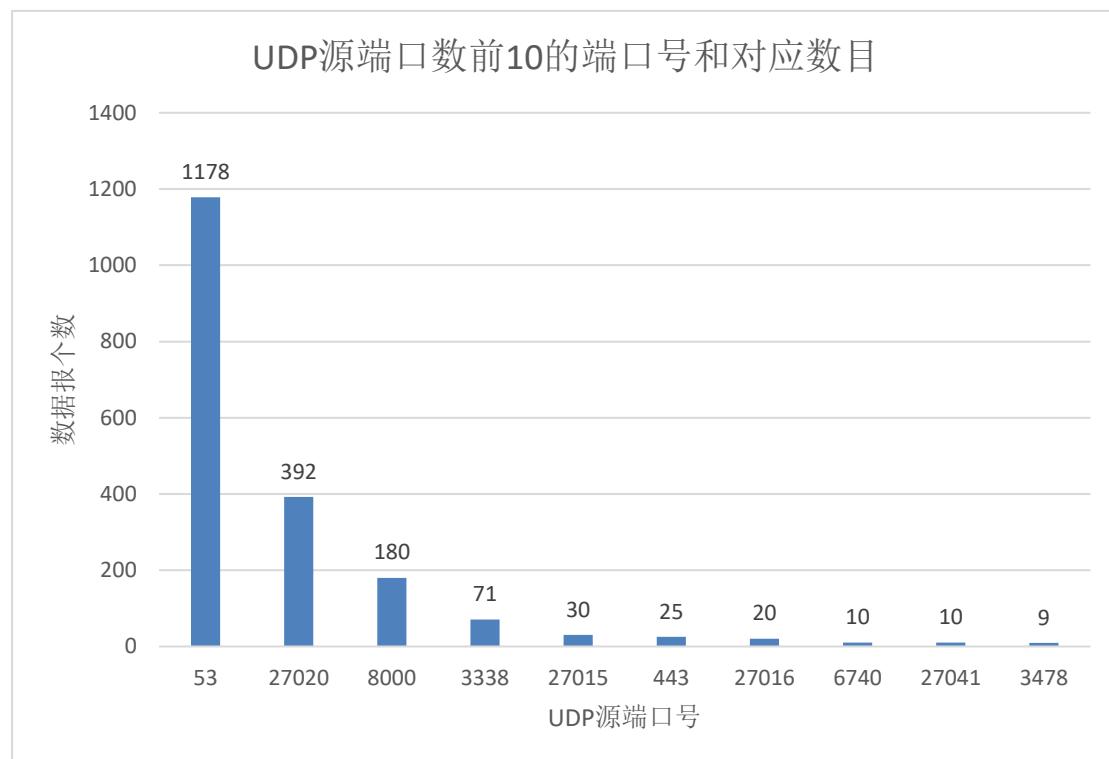
图 3-3 input 和 output UDP 协议数据报长度累积分布统计

## 问题 4

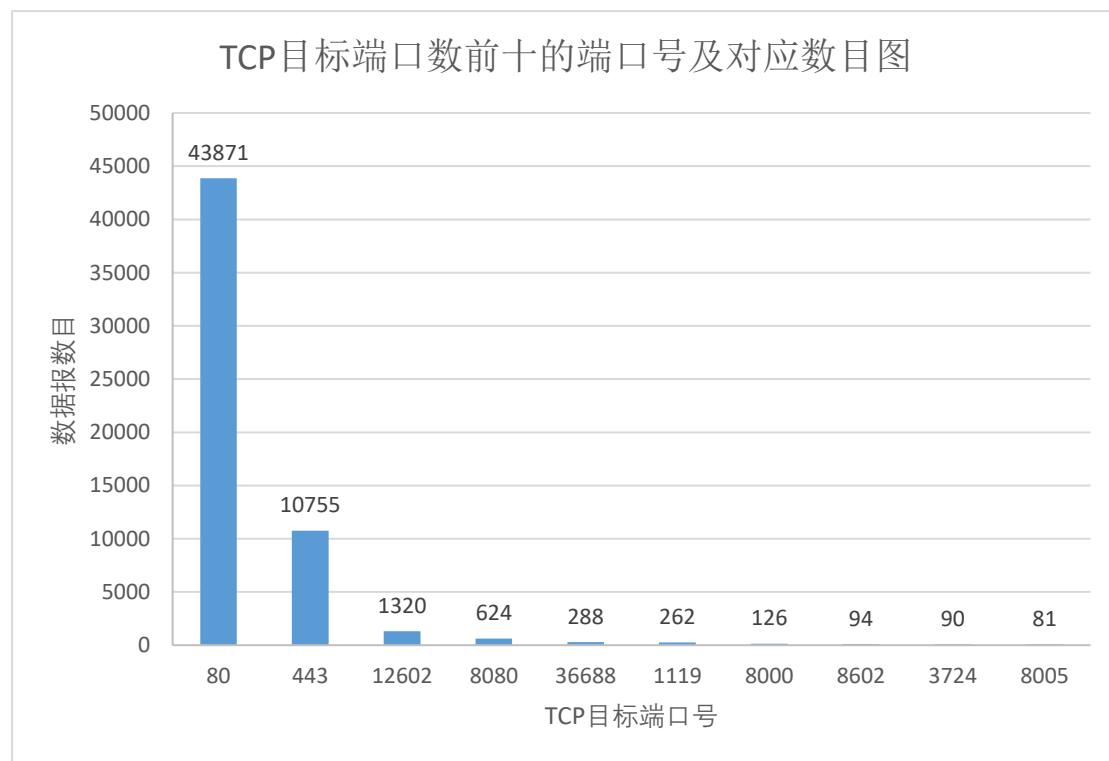
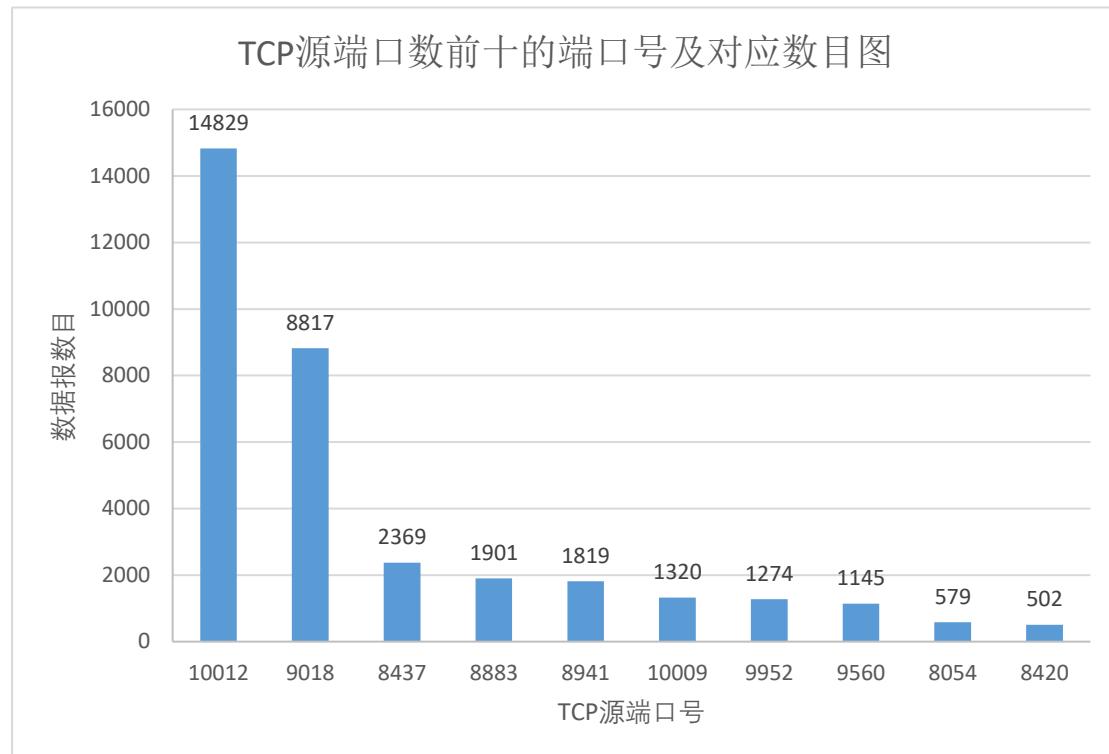
分别对 TCP 和 UDP 的 traffic 给出端口分布的直方图，比较前 10 名端口上数据报长度的累计分布曲线

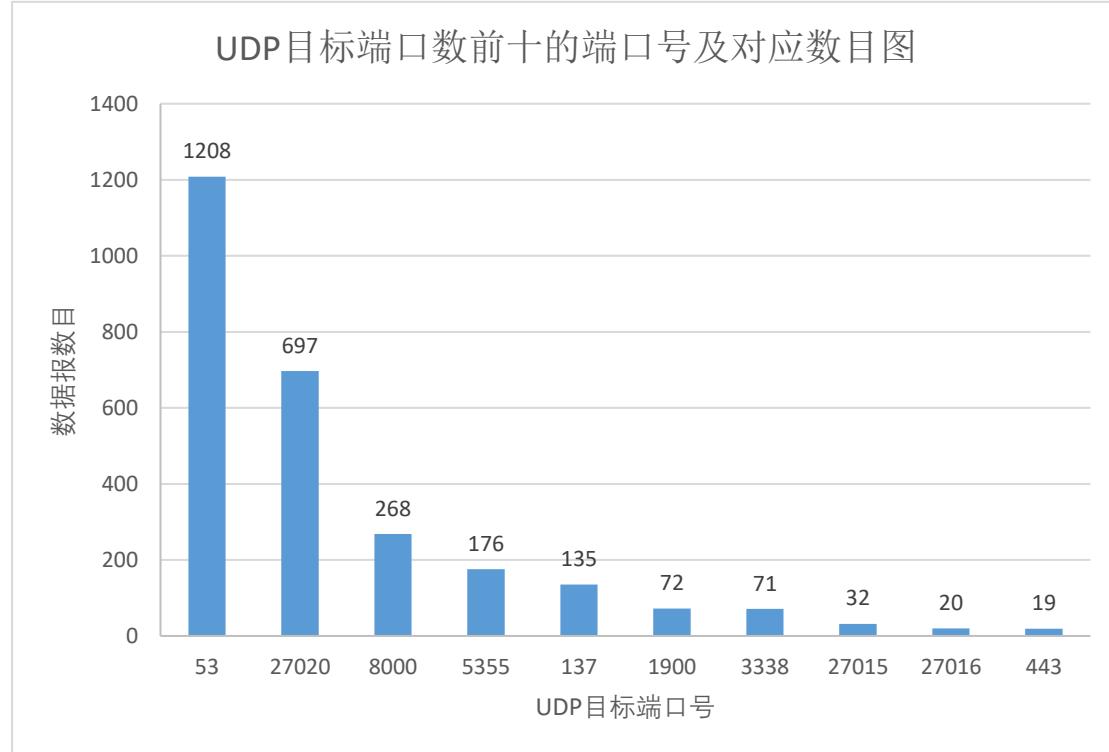
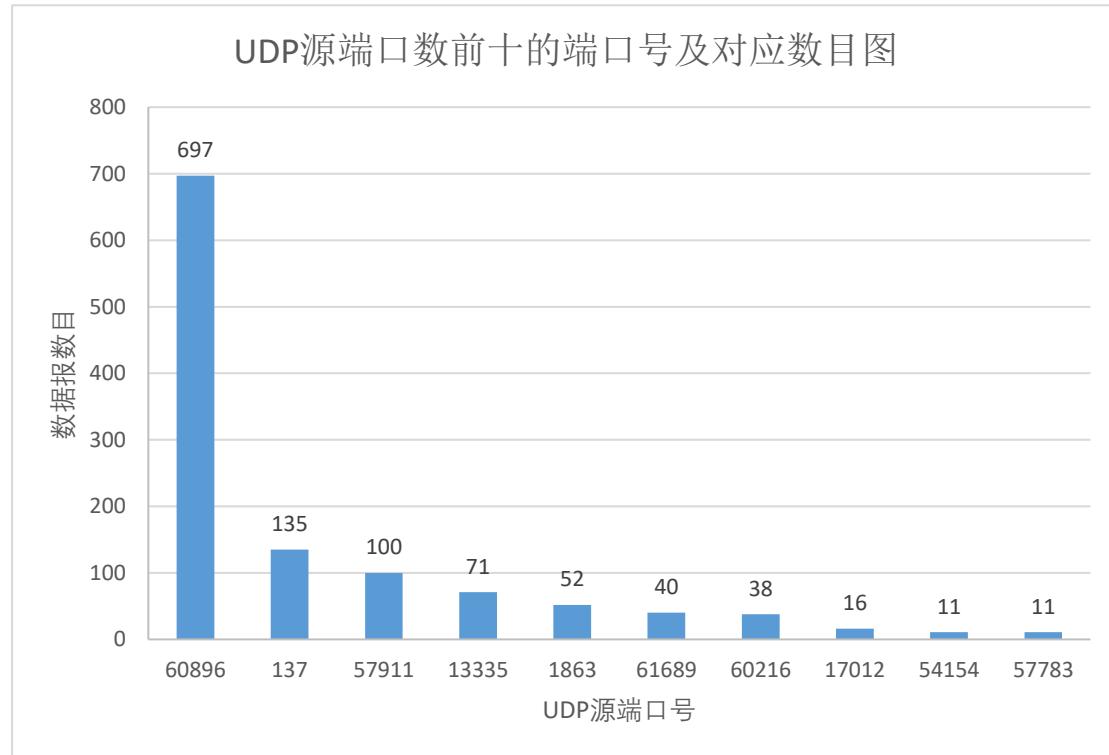
答：通过 excel 做出图像，所得到的 input 的直方图如下：

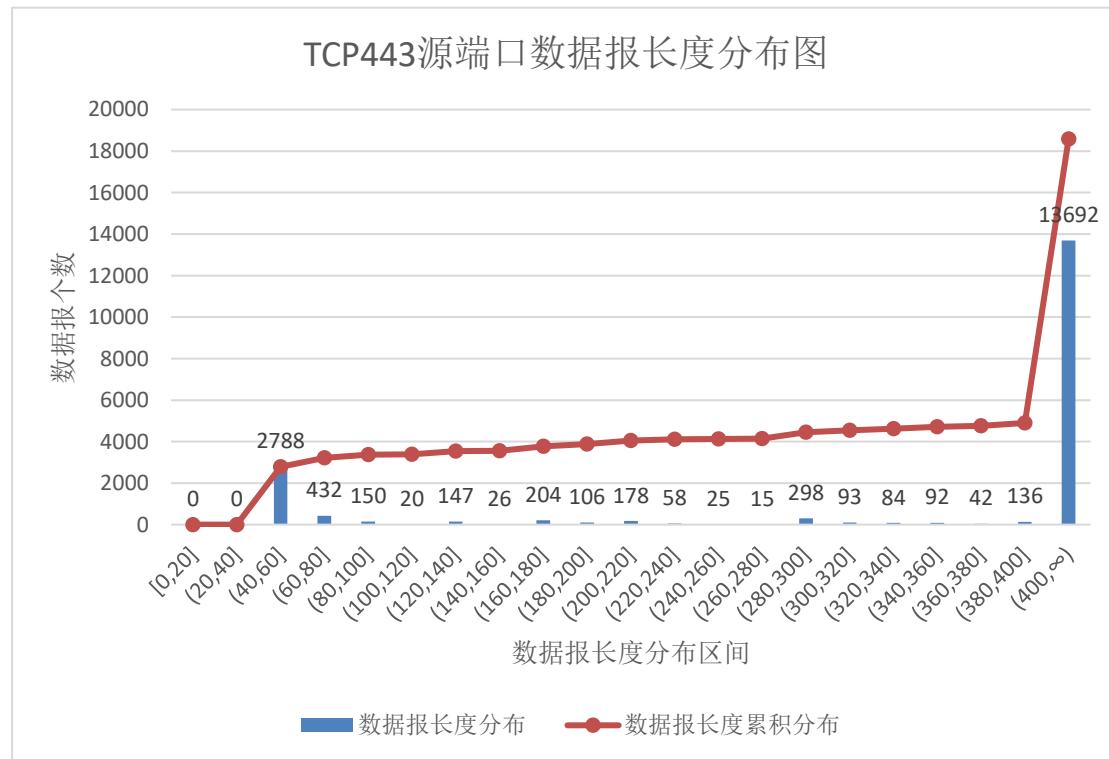
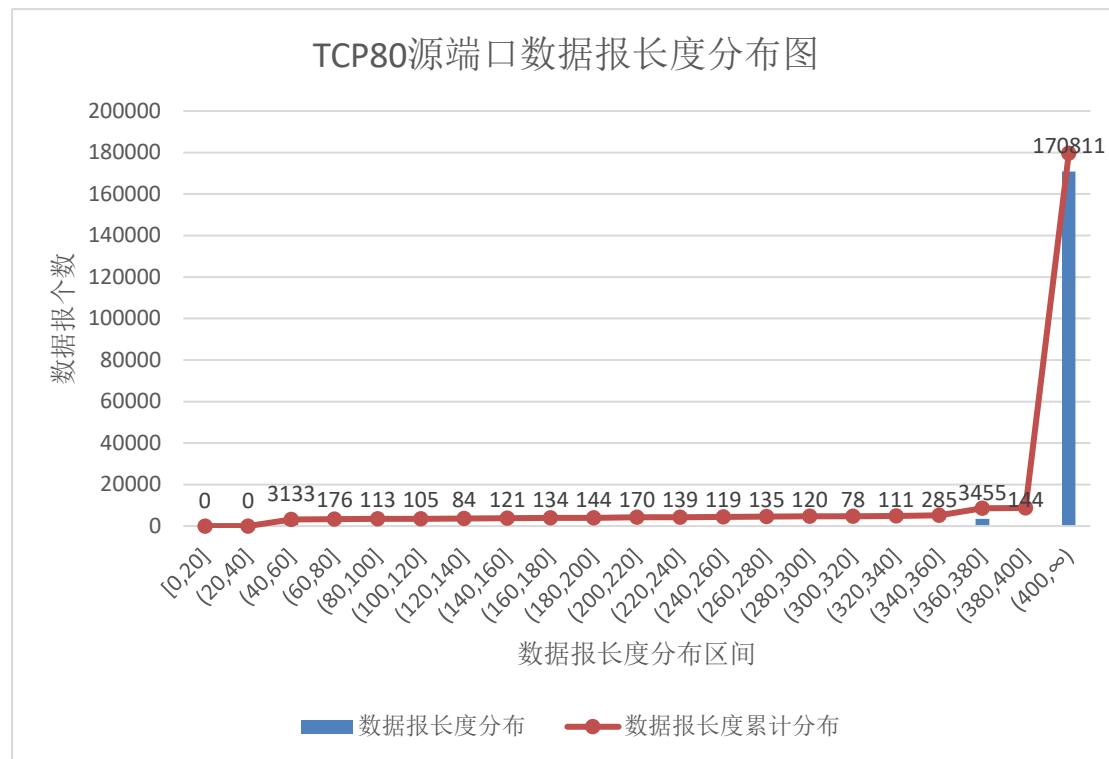


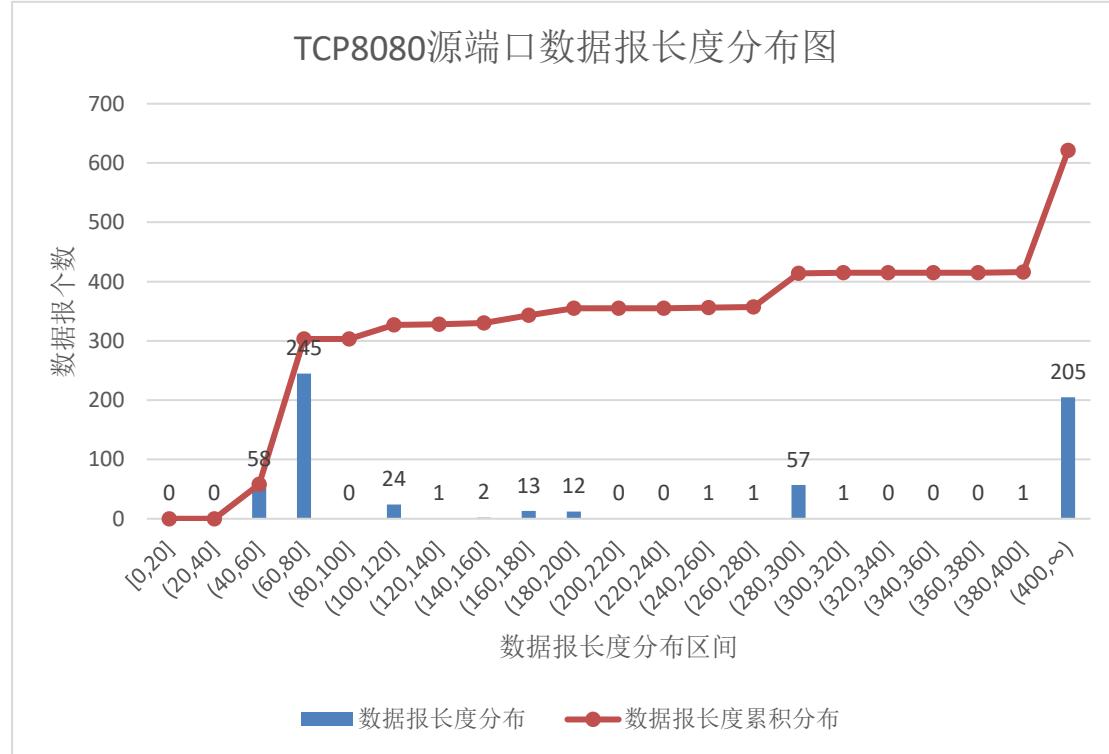
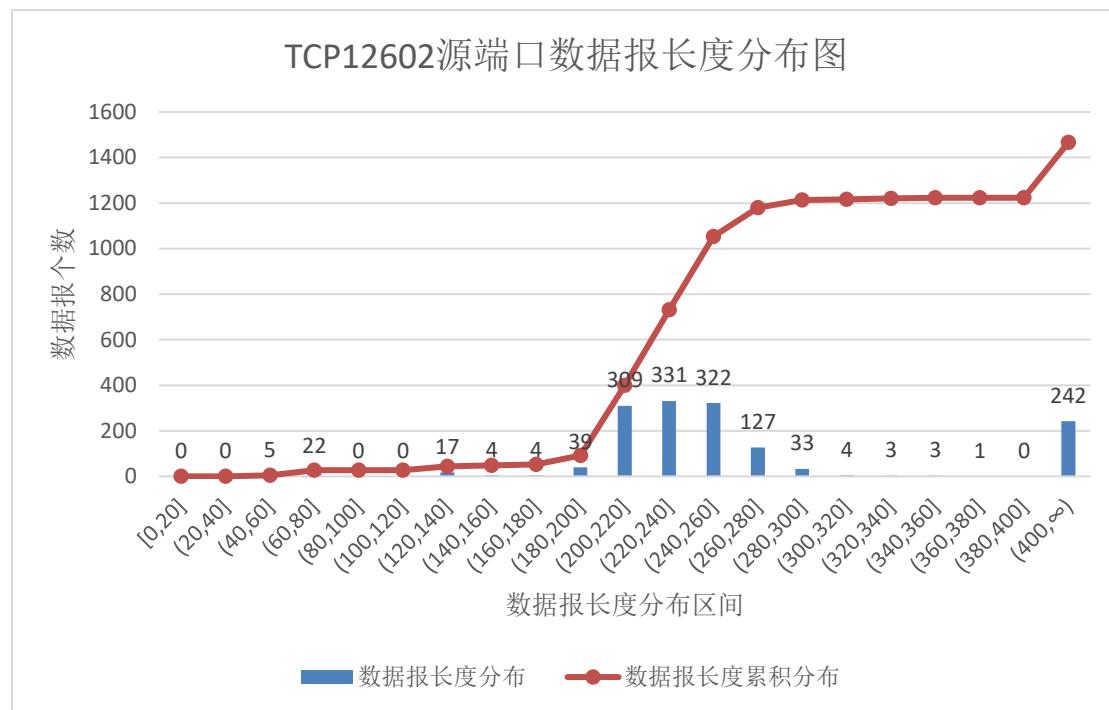


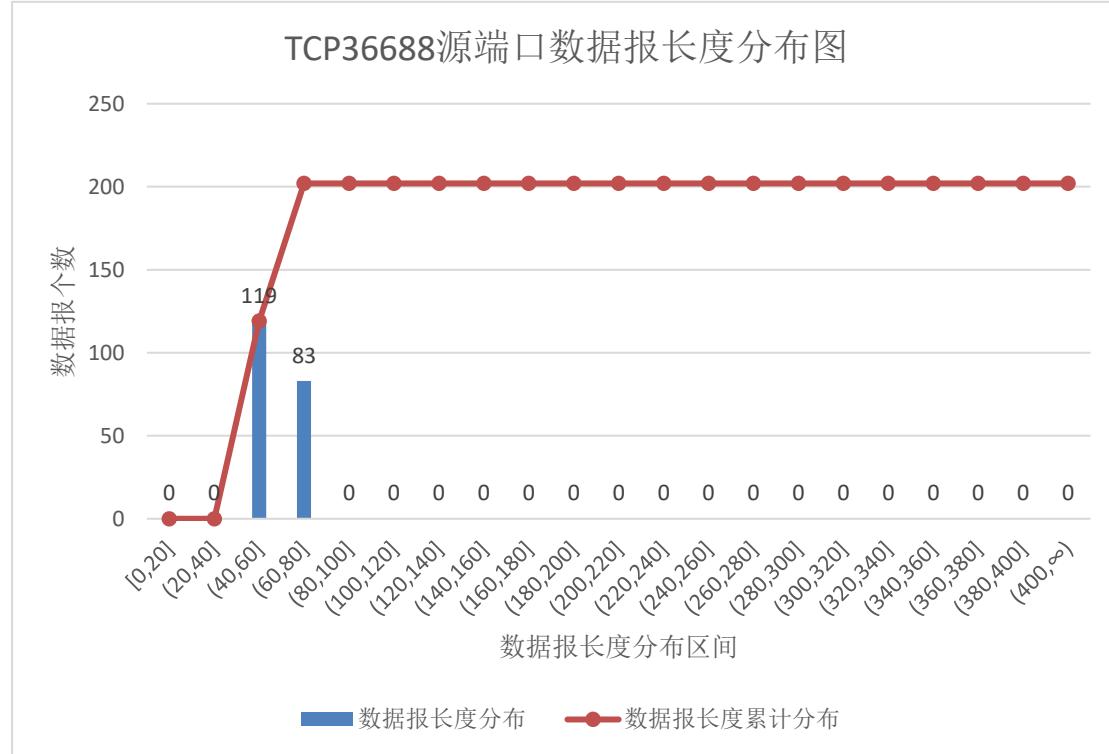
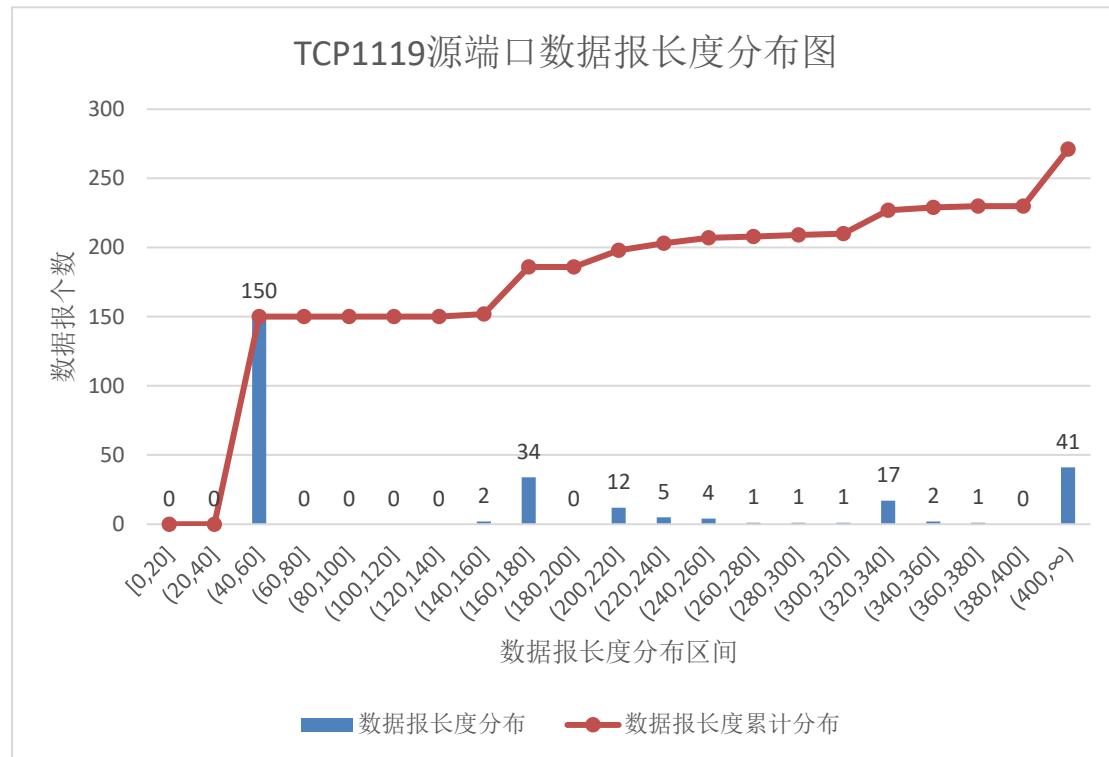
通过 excel 做出图像，所得到的 output 的直方图如下：

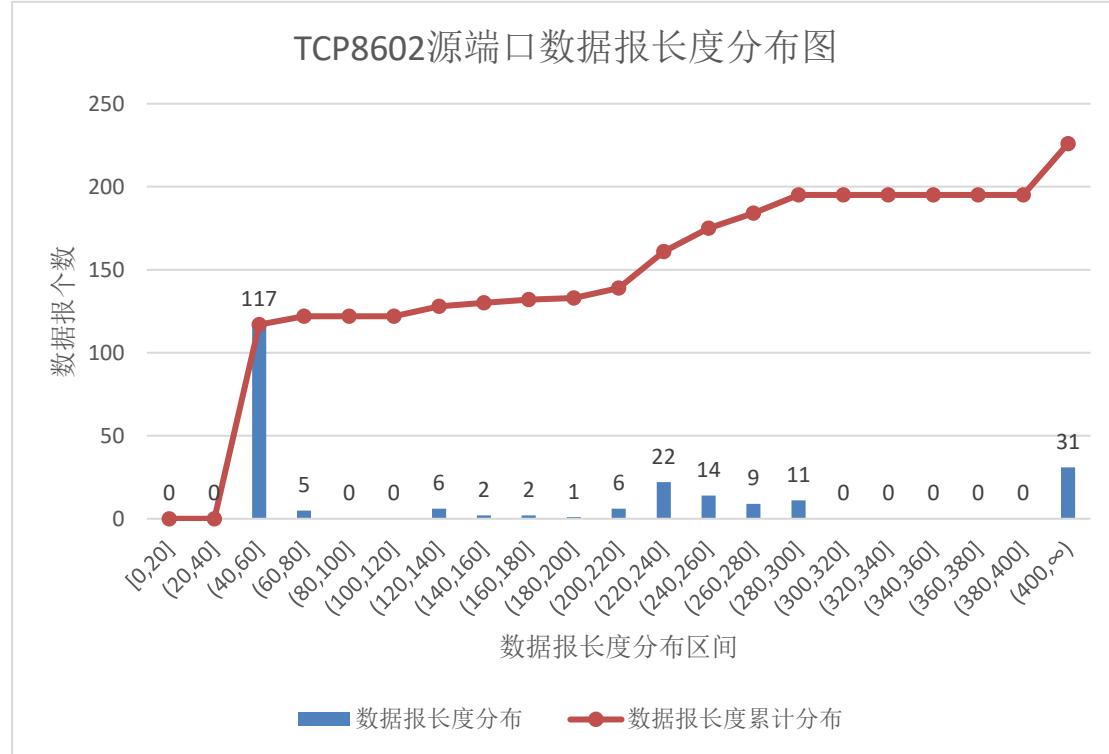
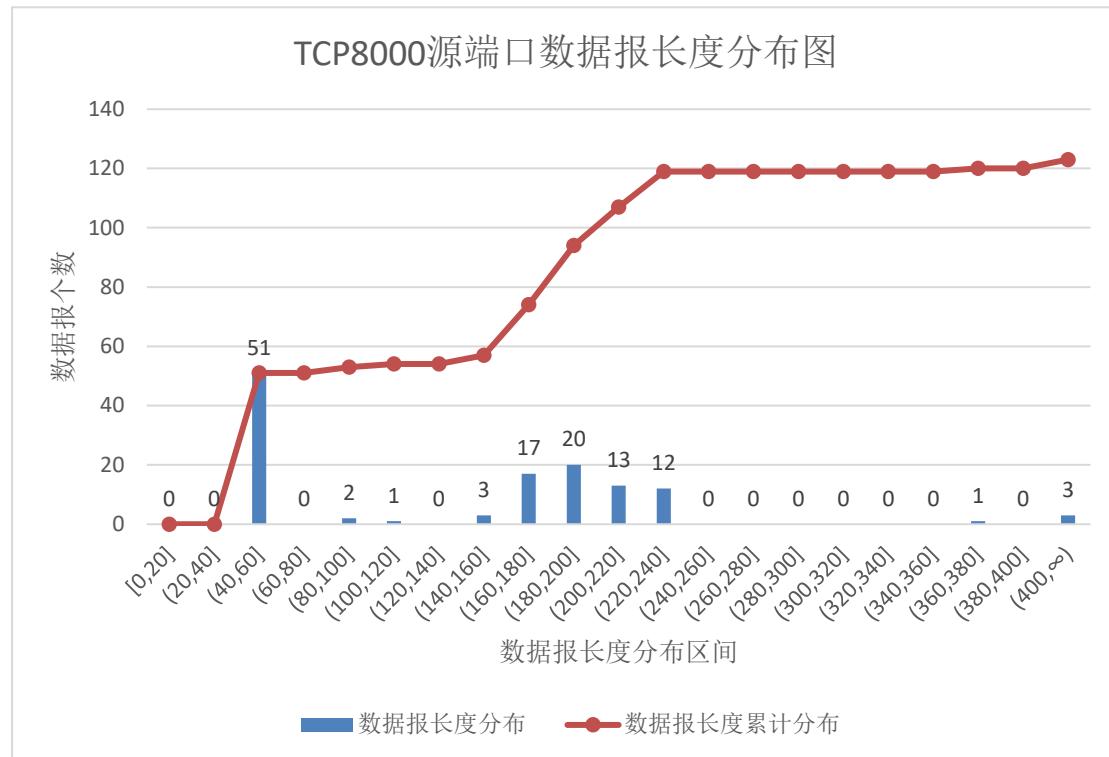


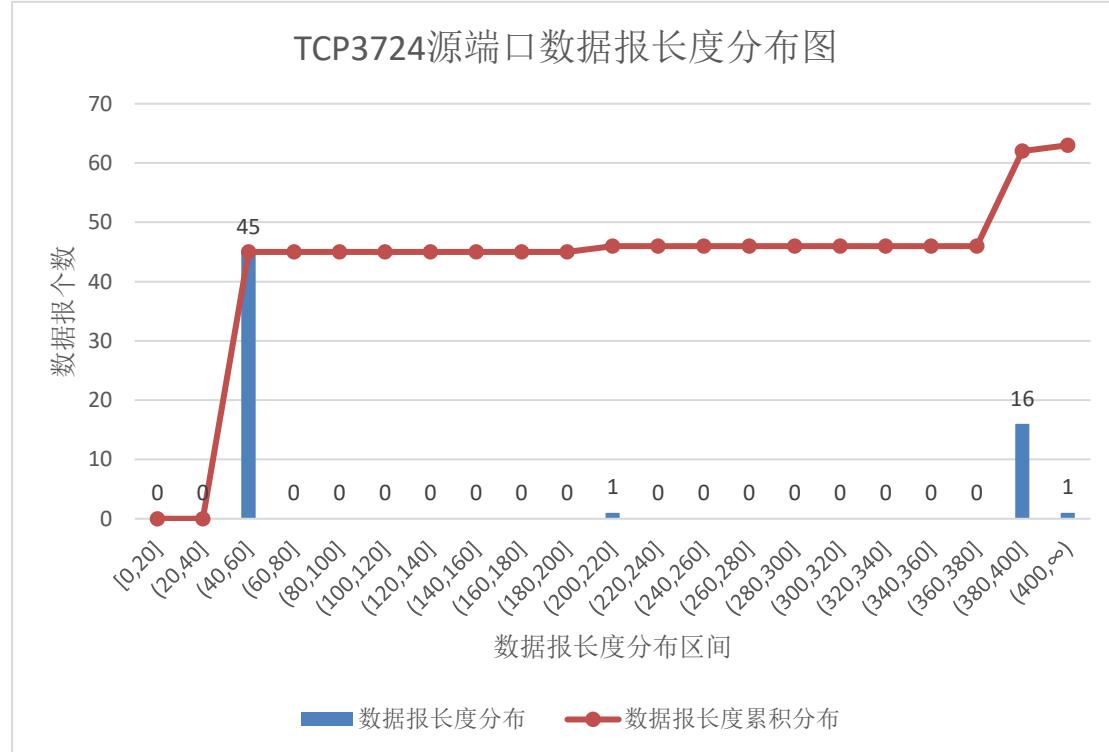
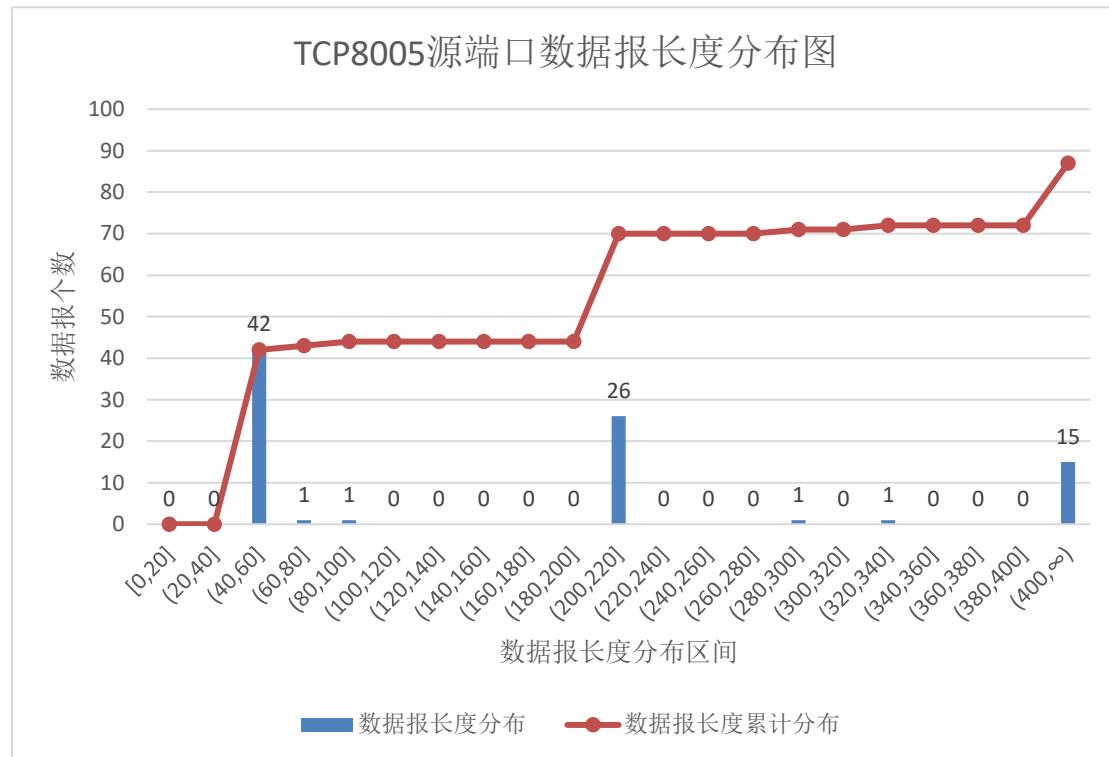


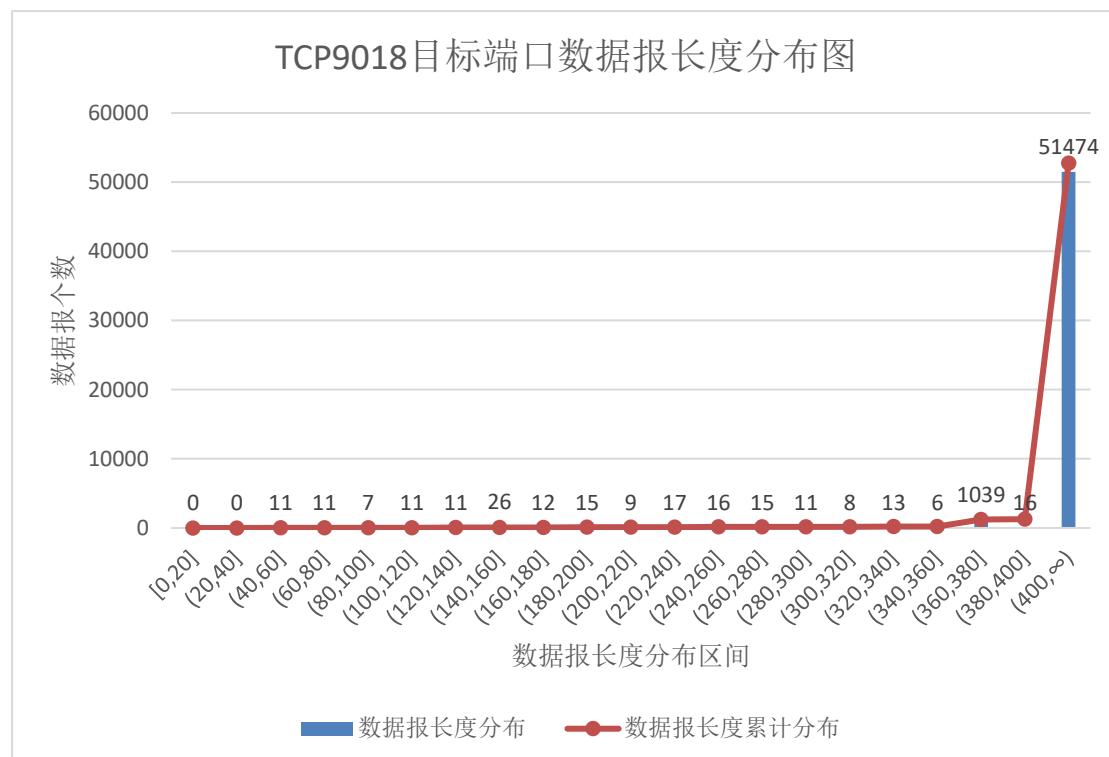
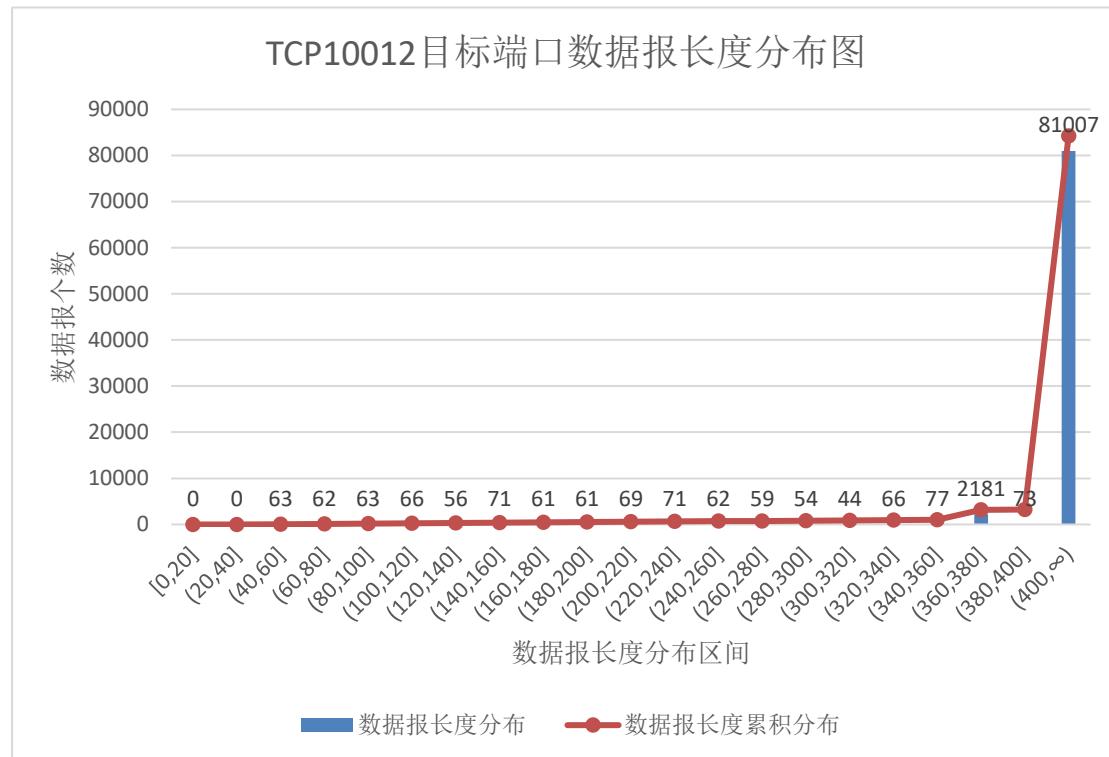
**Input:****[1]TCP 源端口前十的端口数据报长度分布情况:**

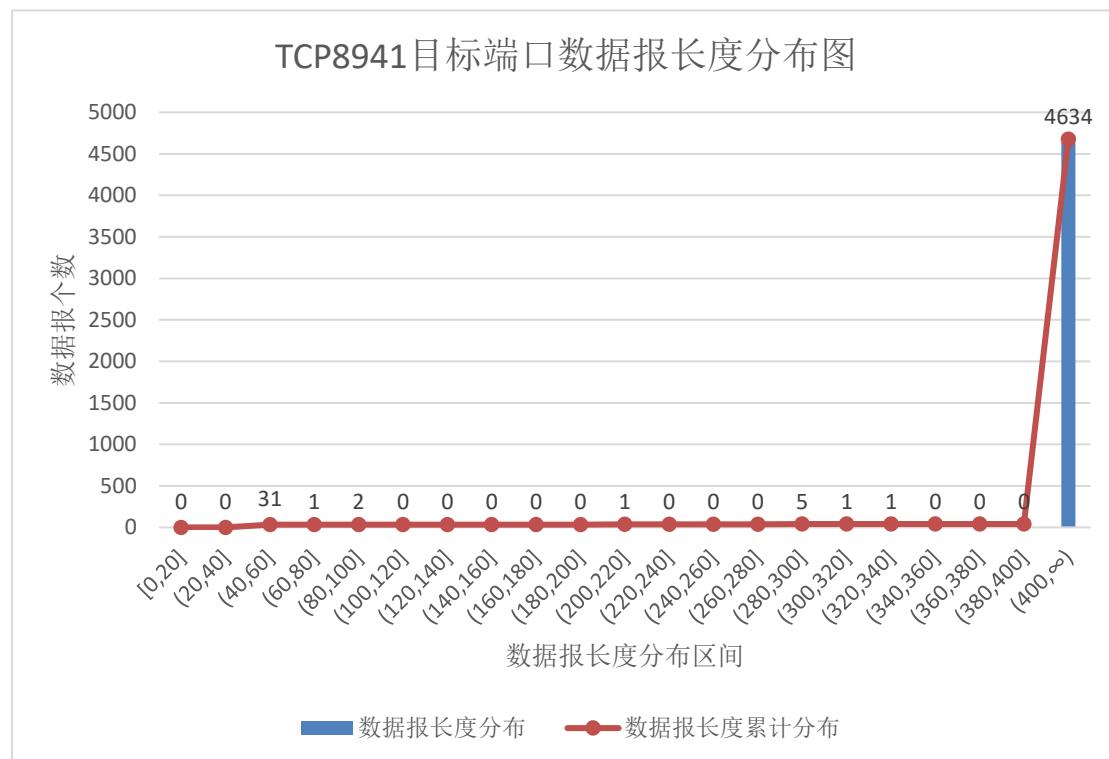
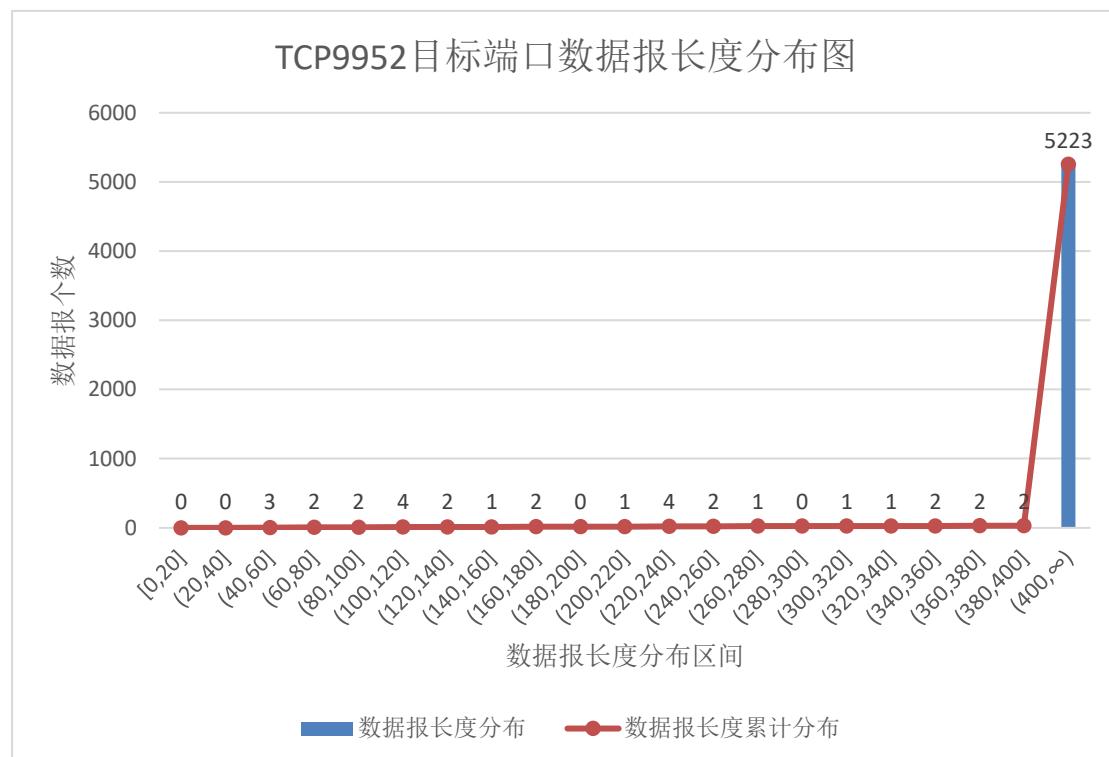


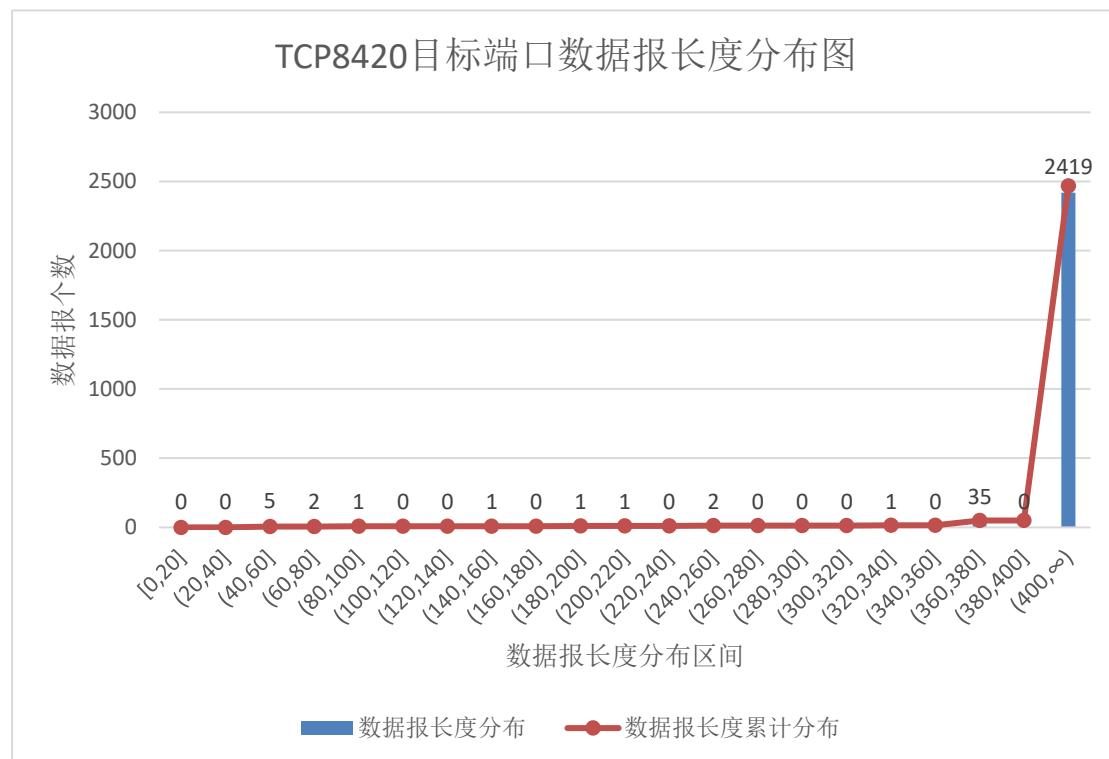
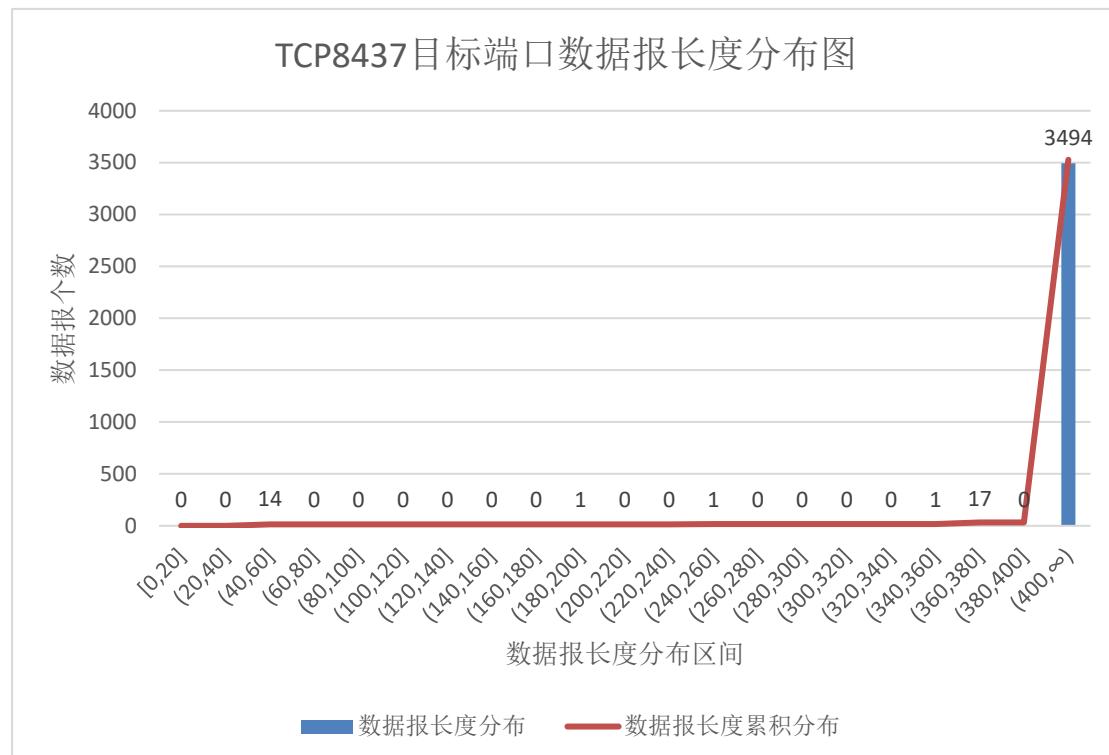


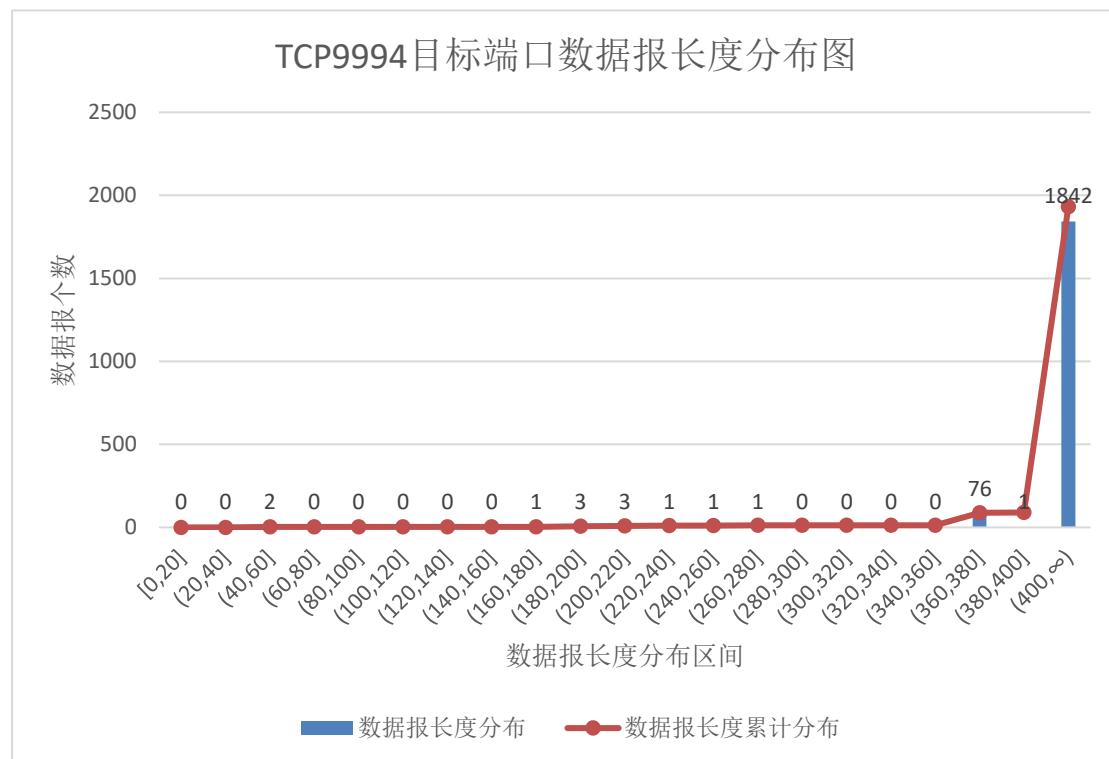
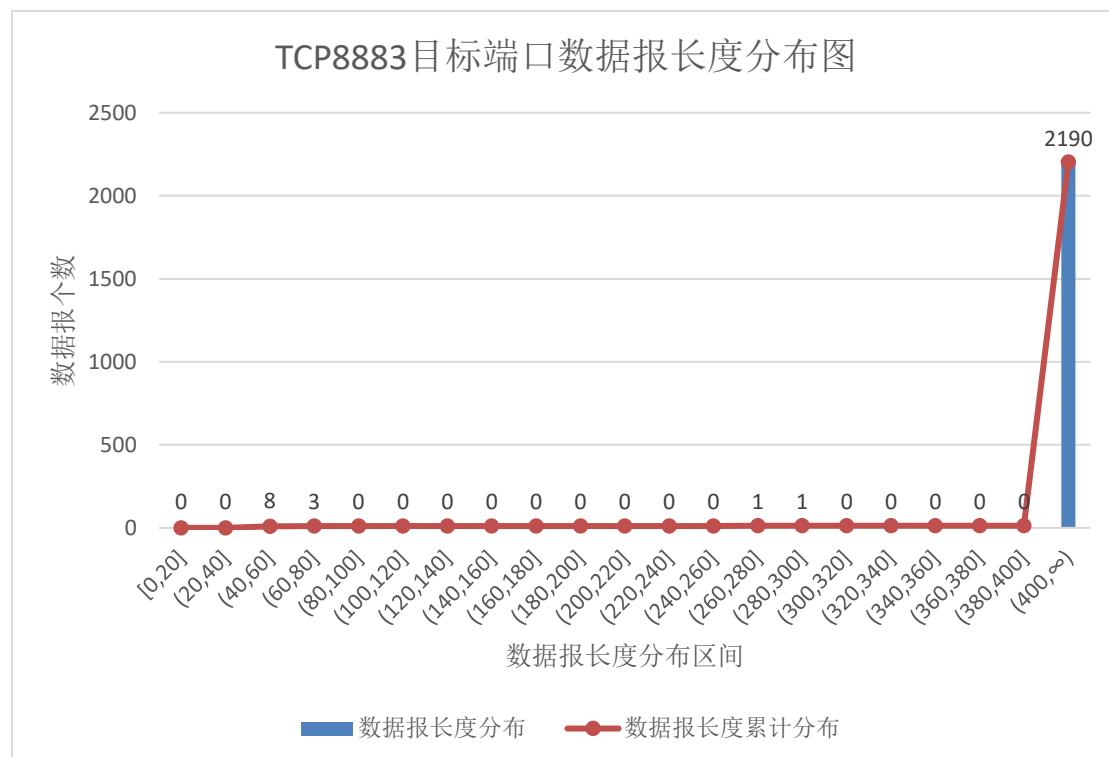


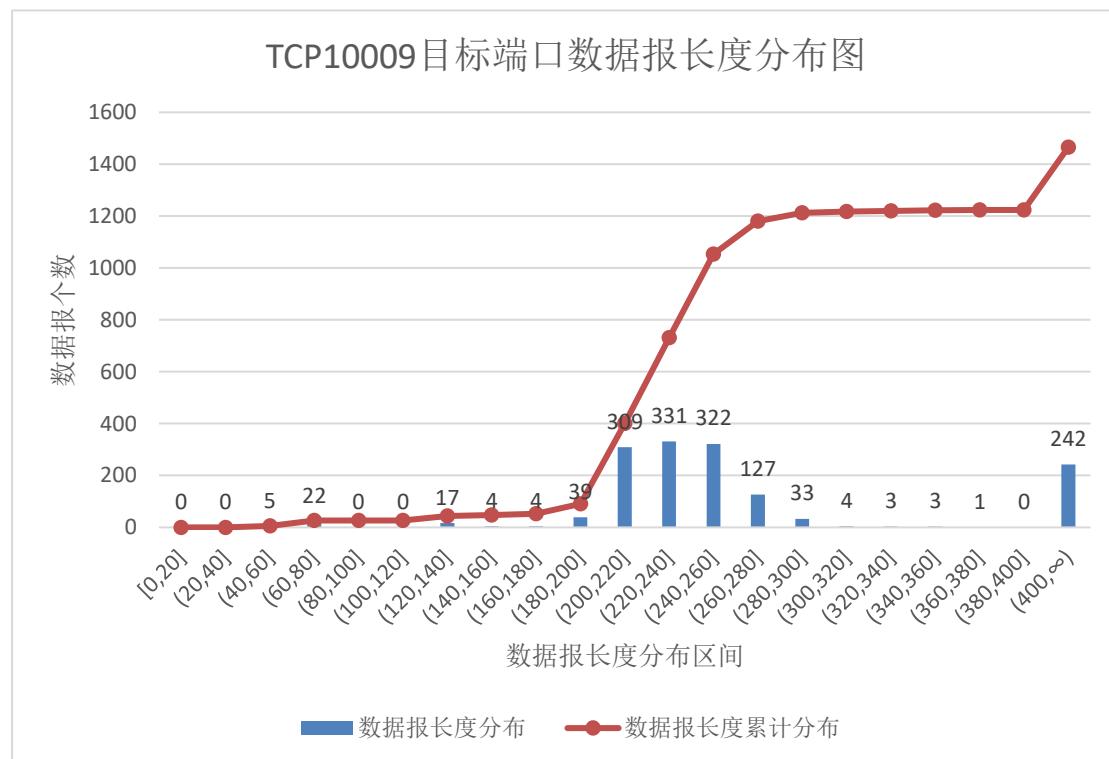
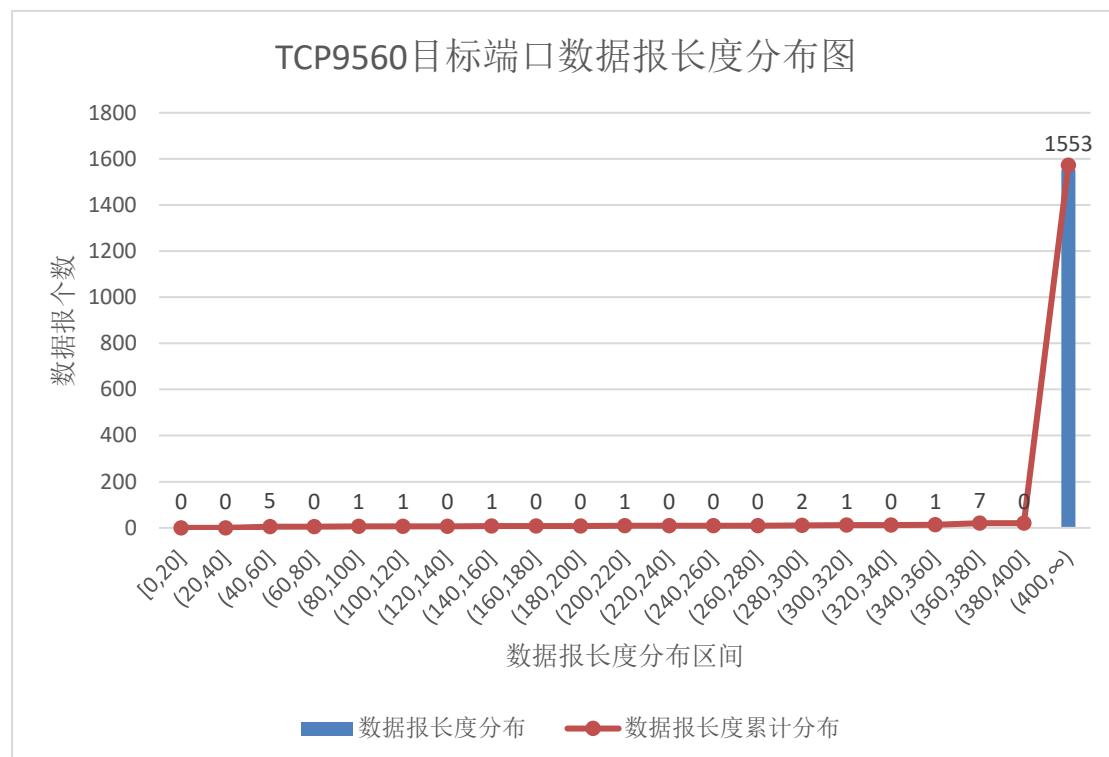


**[2]TCP 目标端口数前十的数据报长度分布情况**

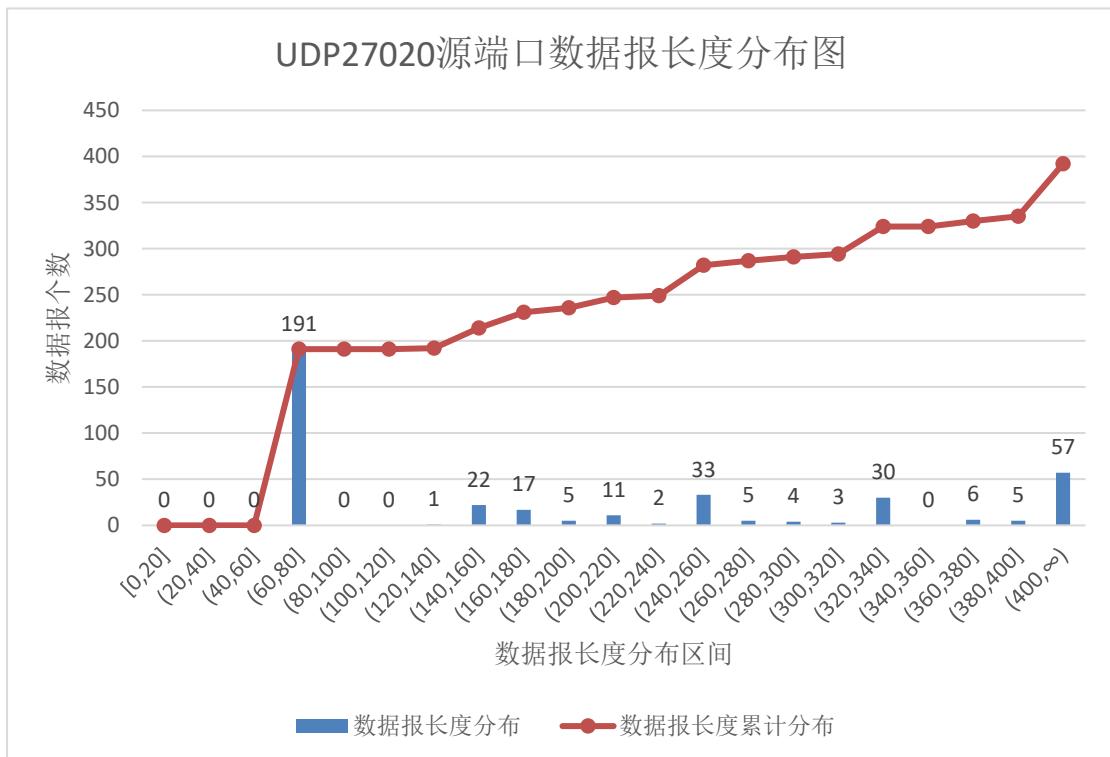
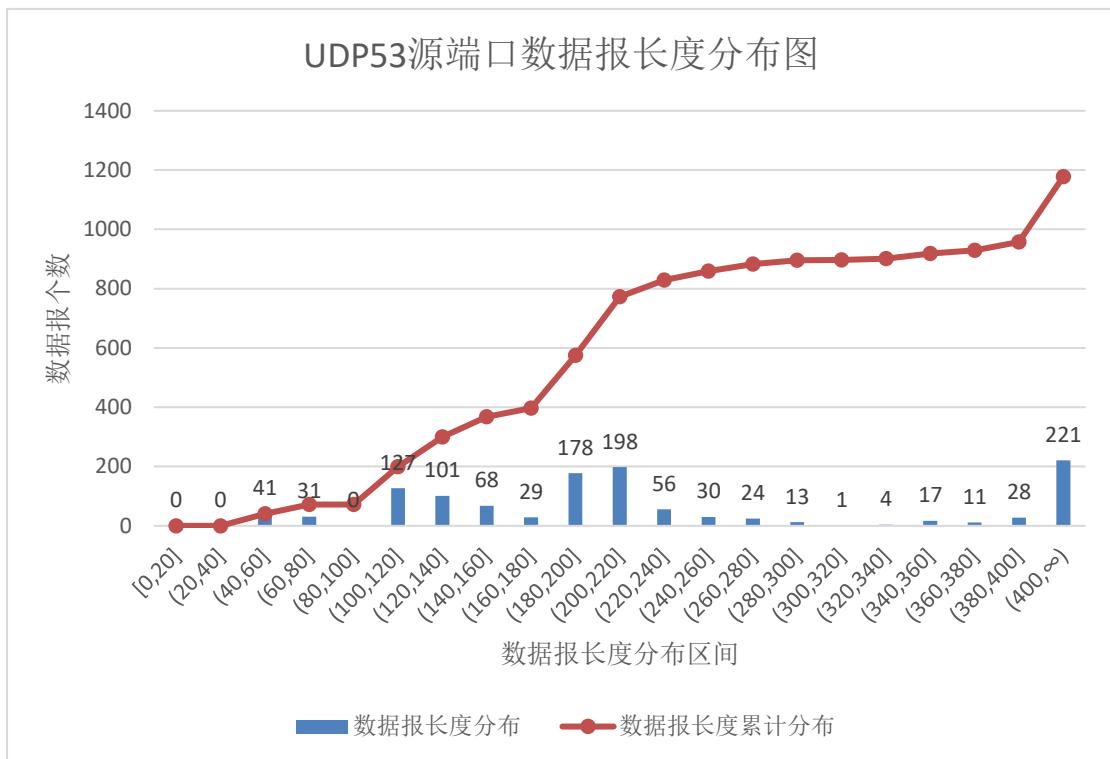


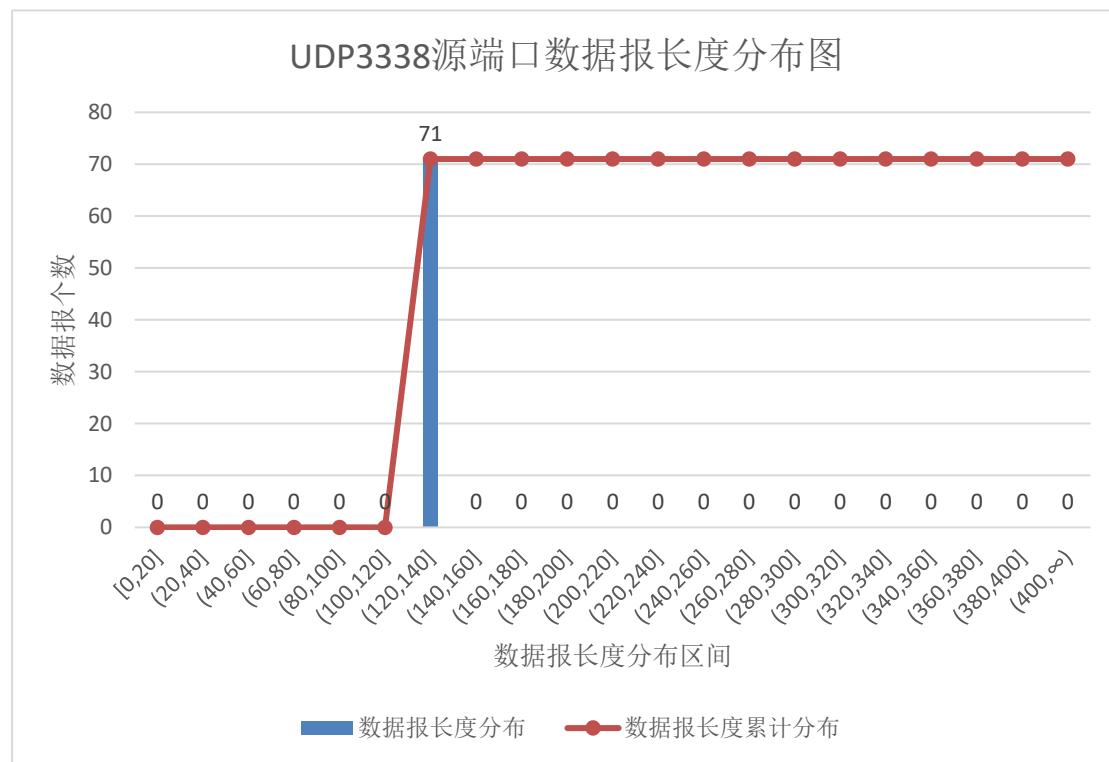
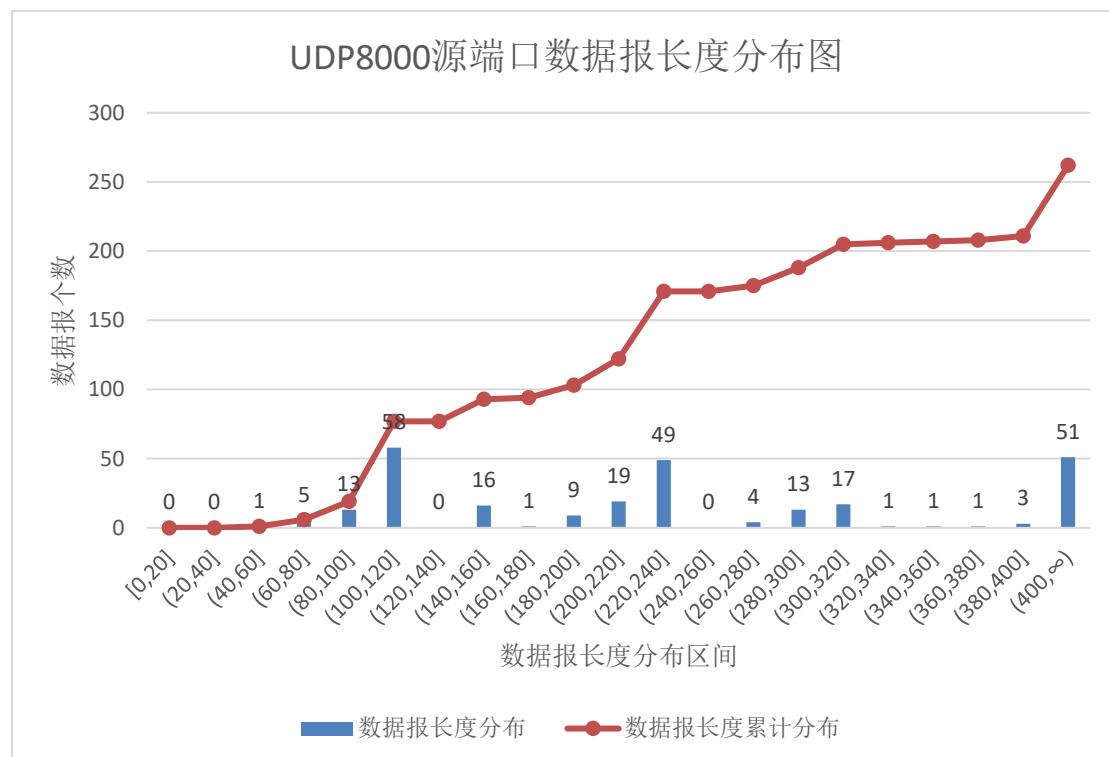


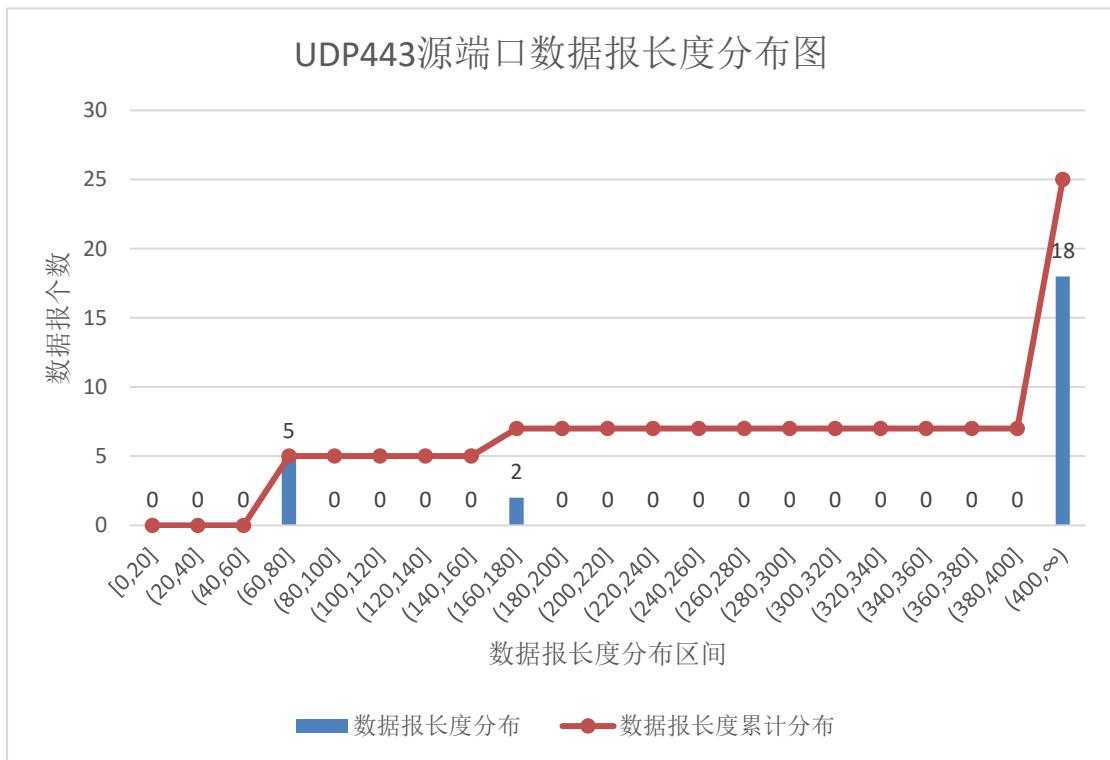
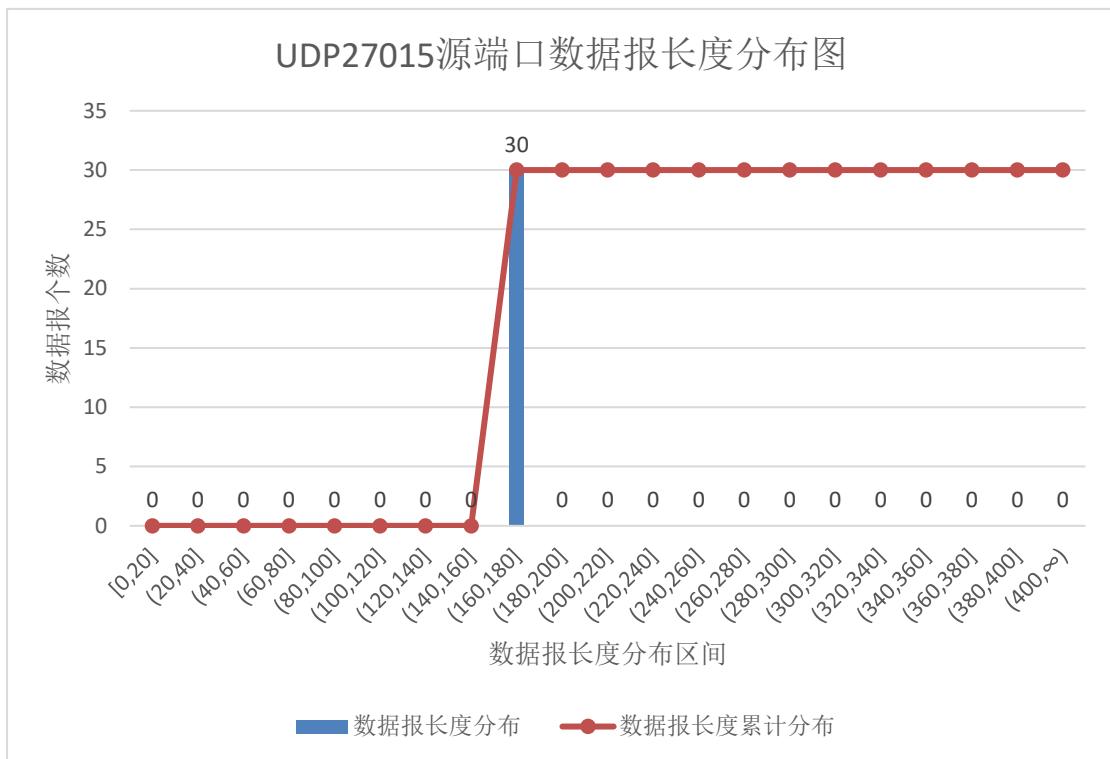


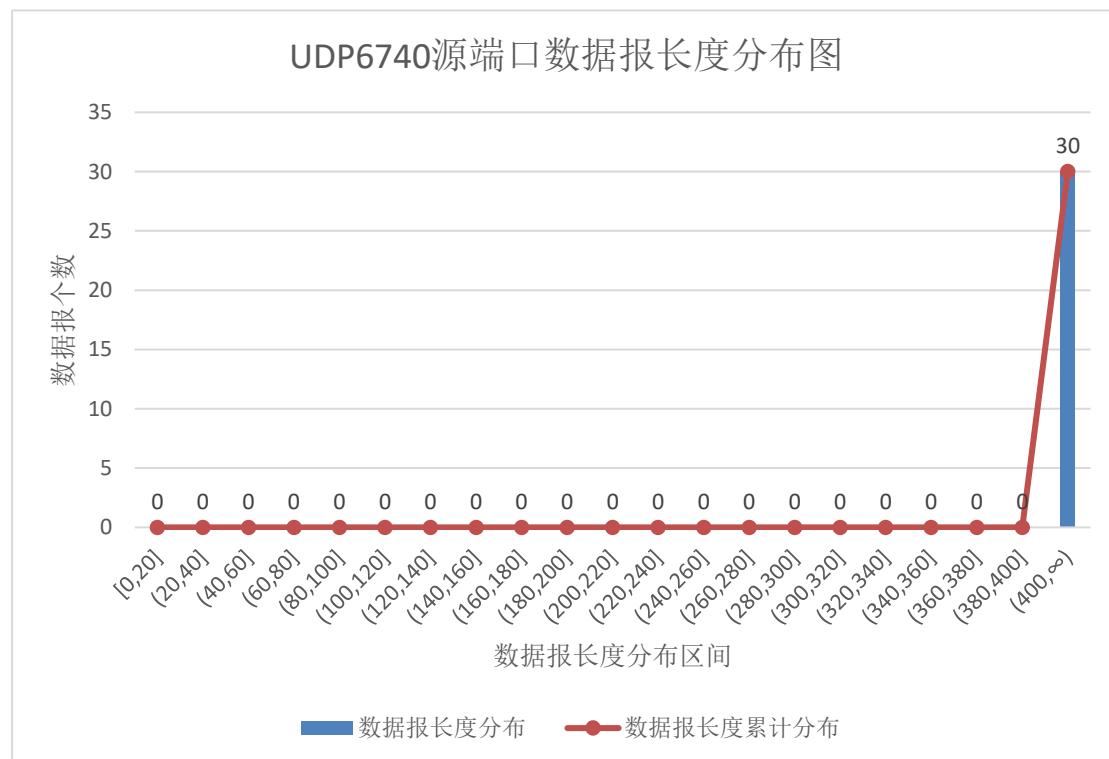
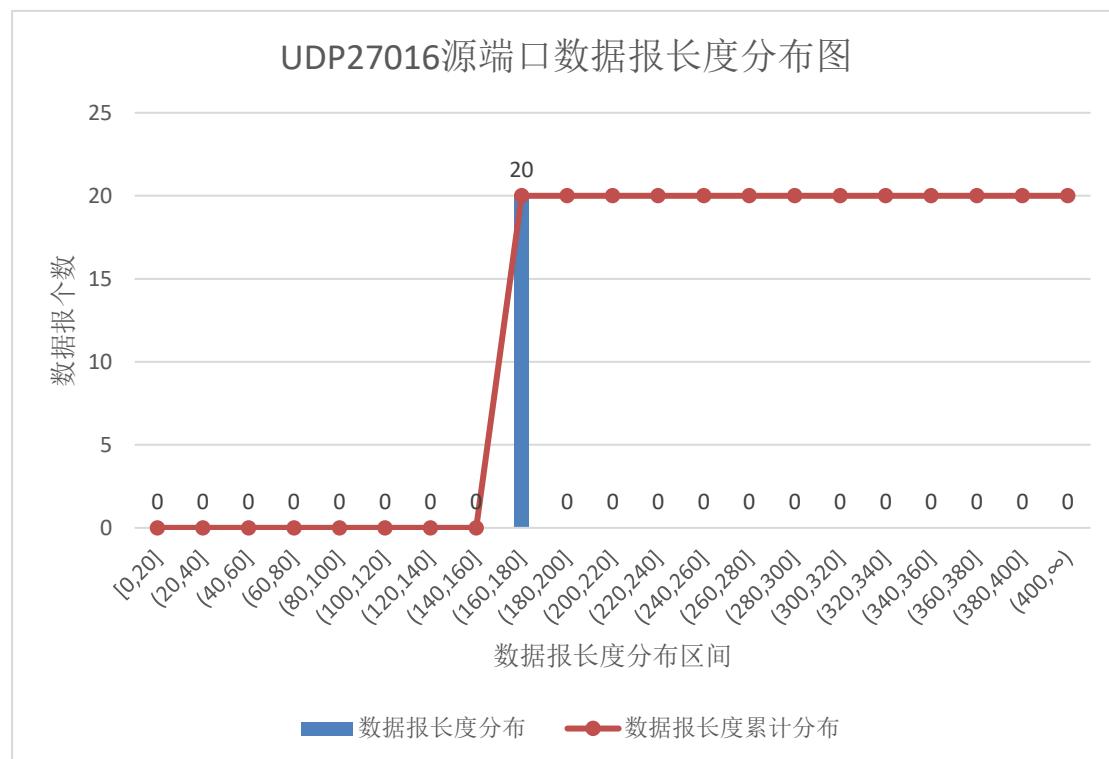


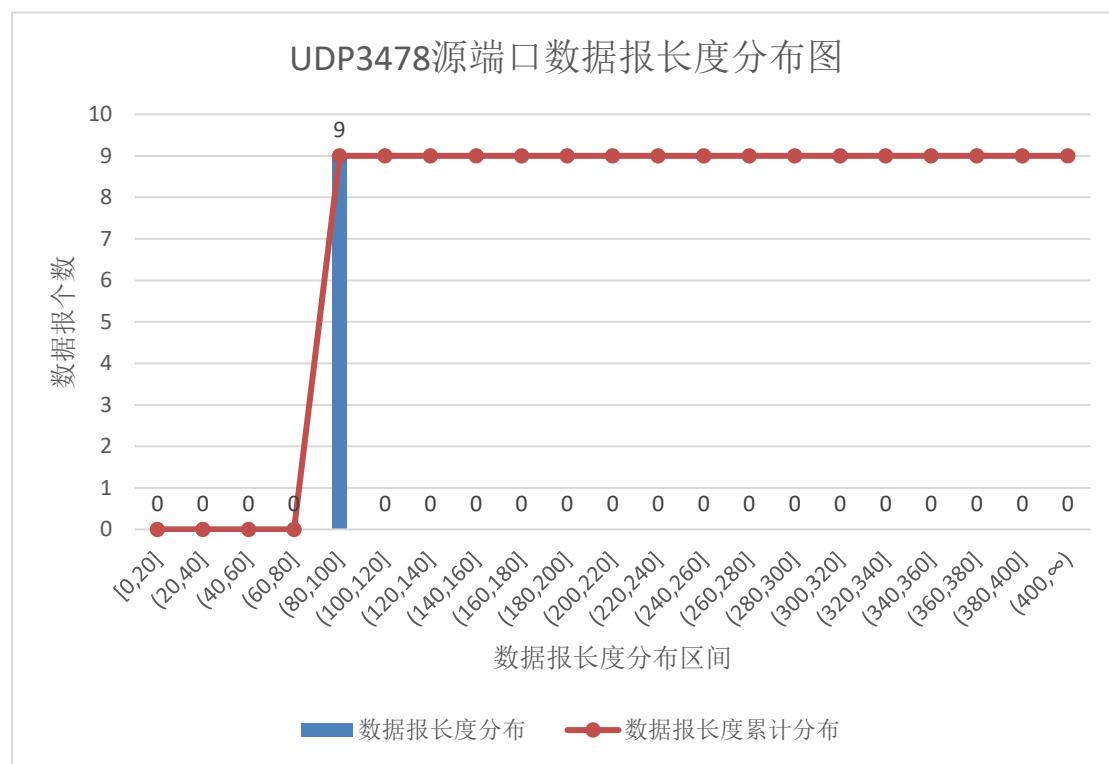
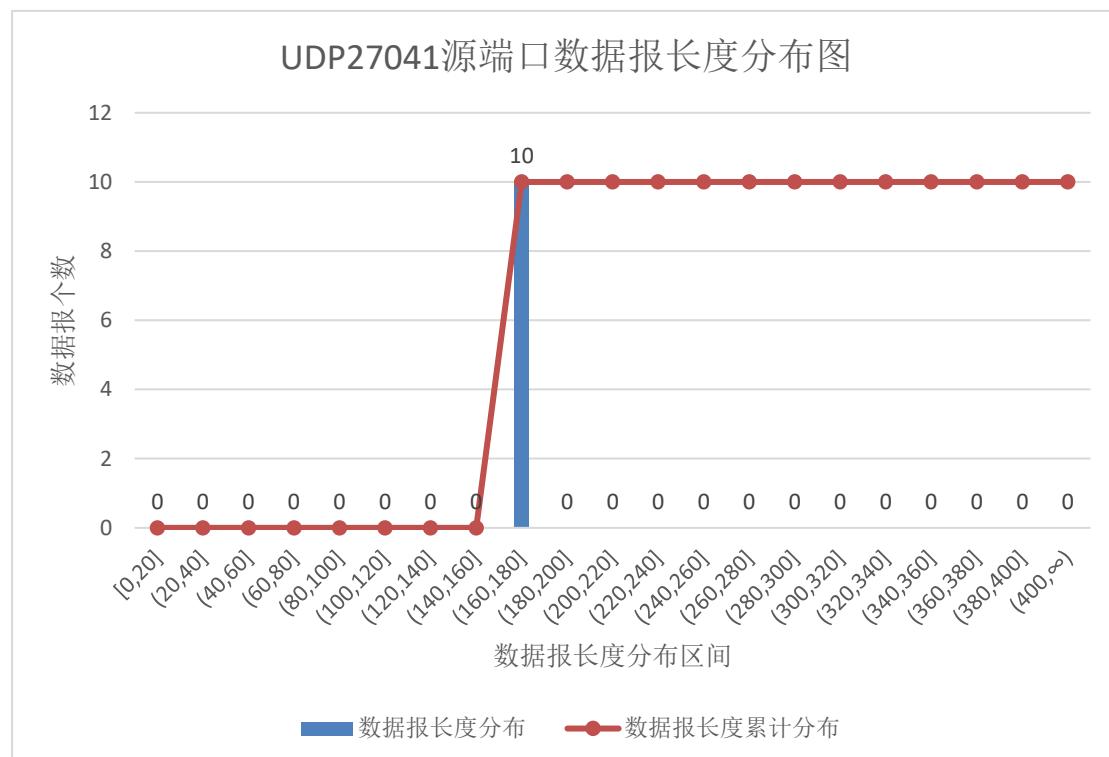
[3] UDP 源端口数前 10 的数据报长度分布情况:



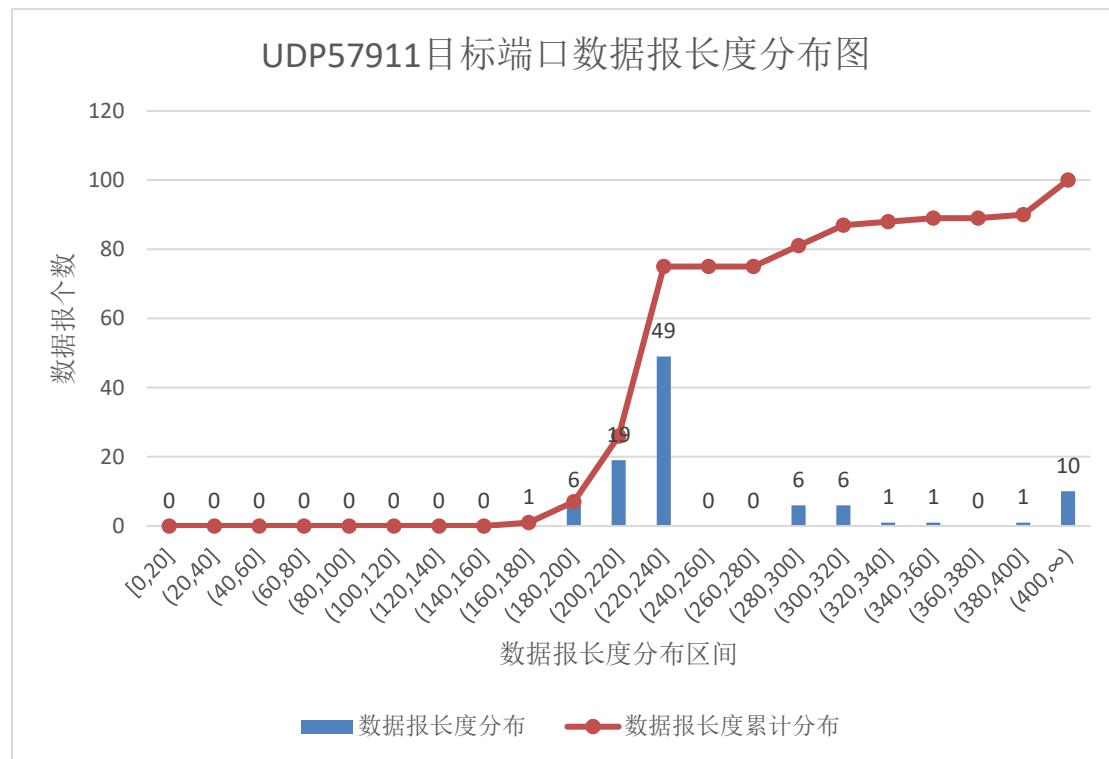
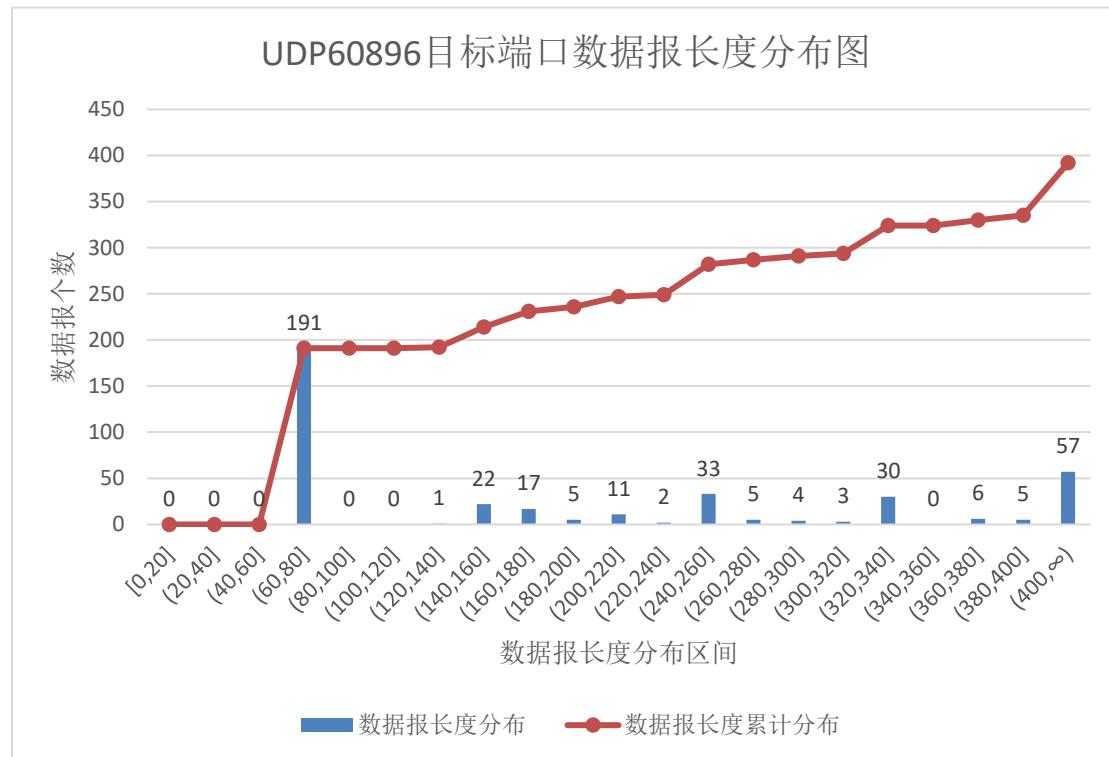


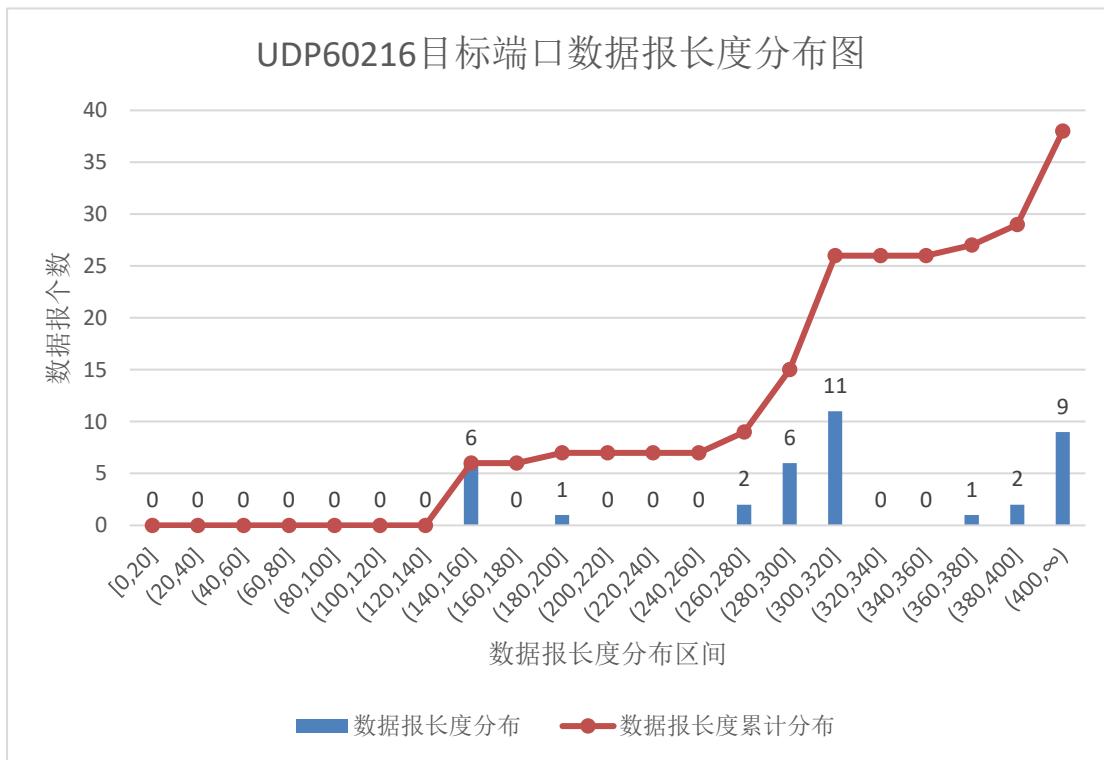
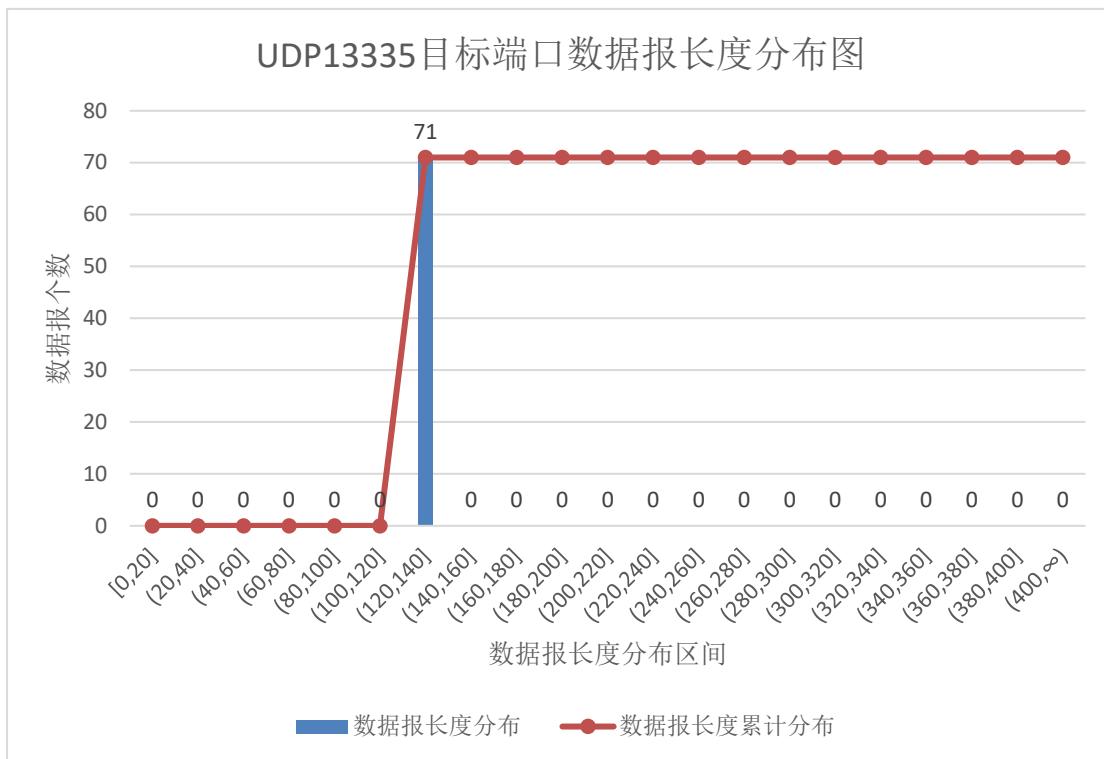


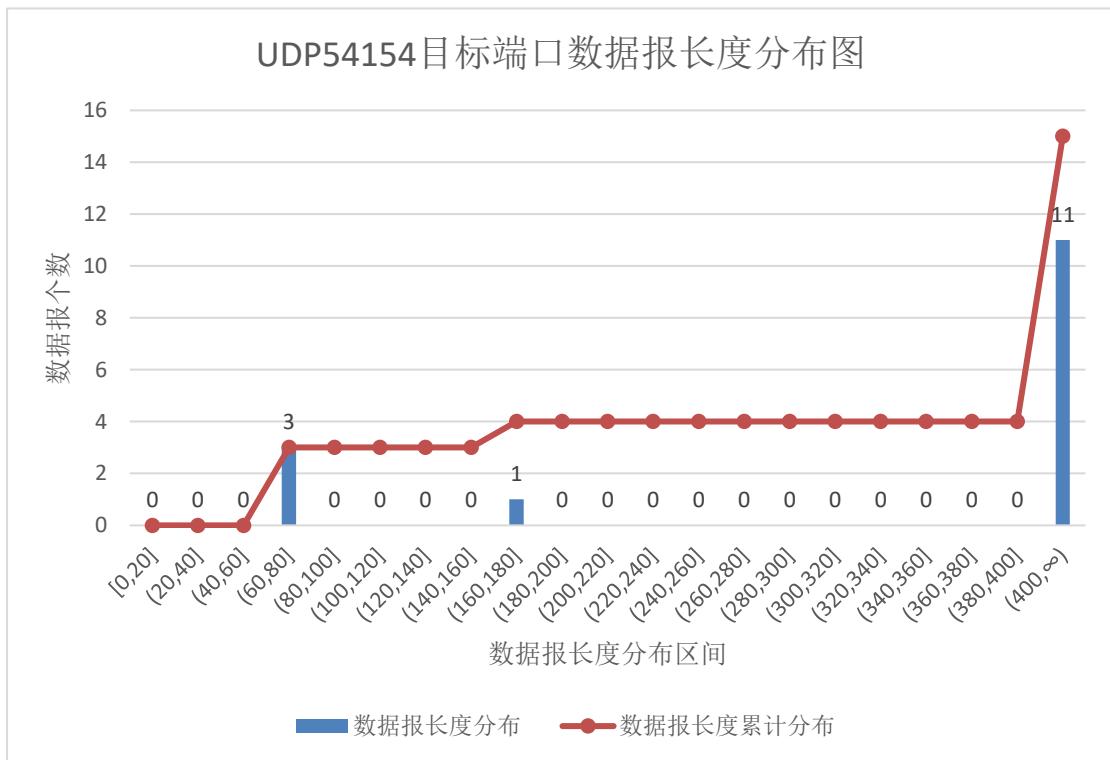
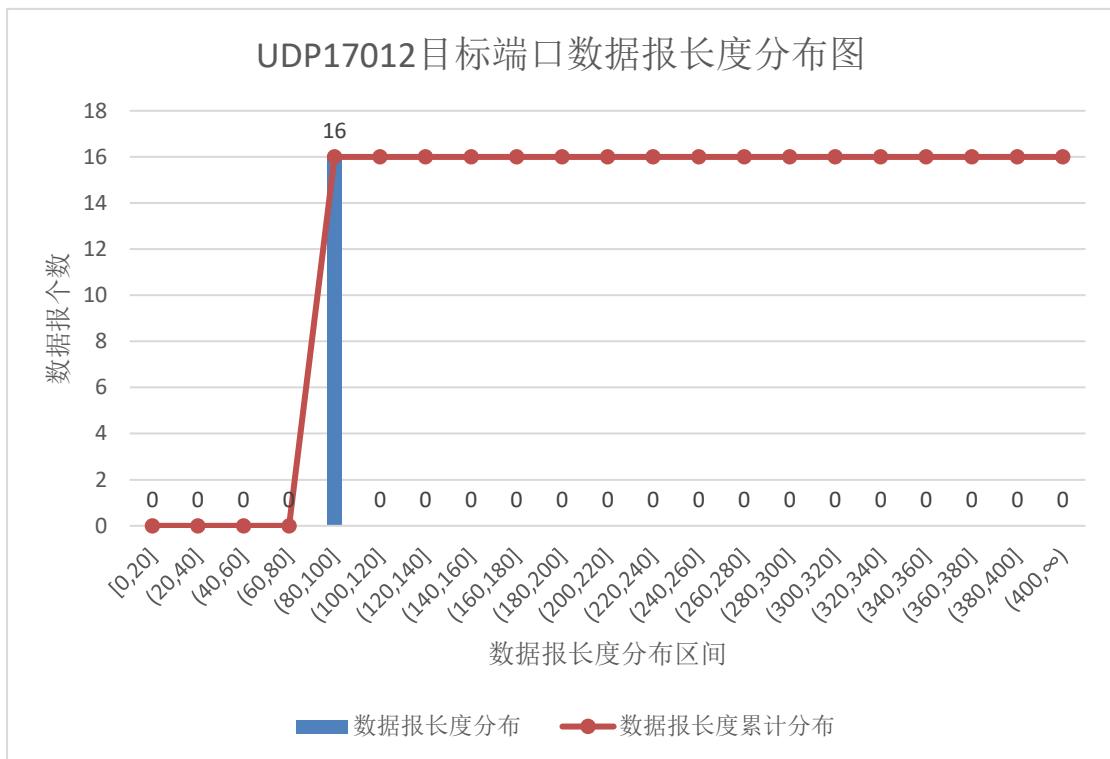


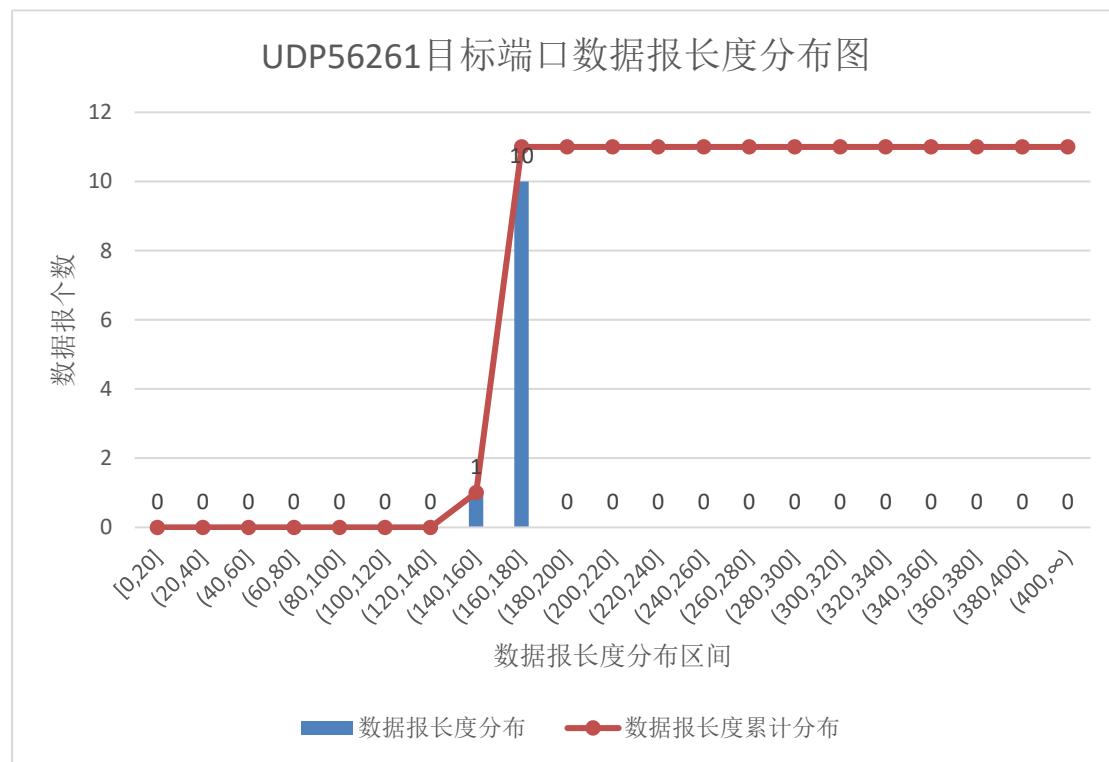
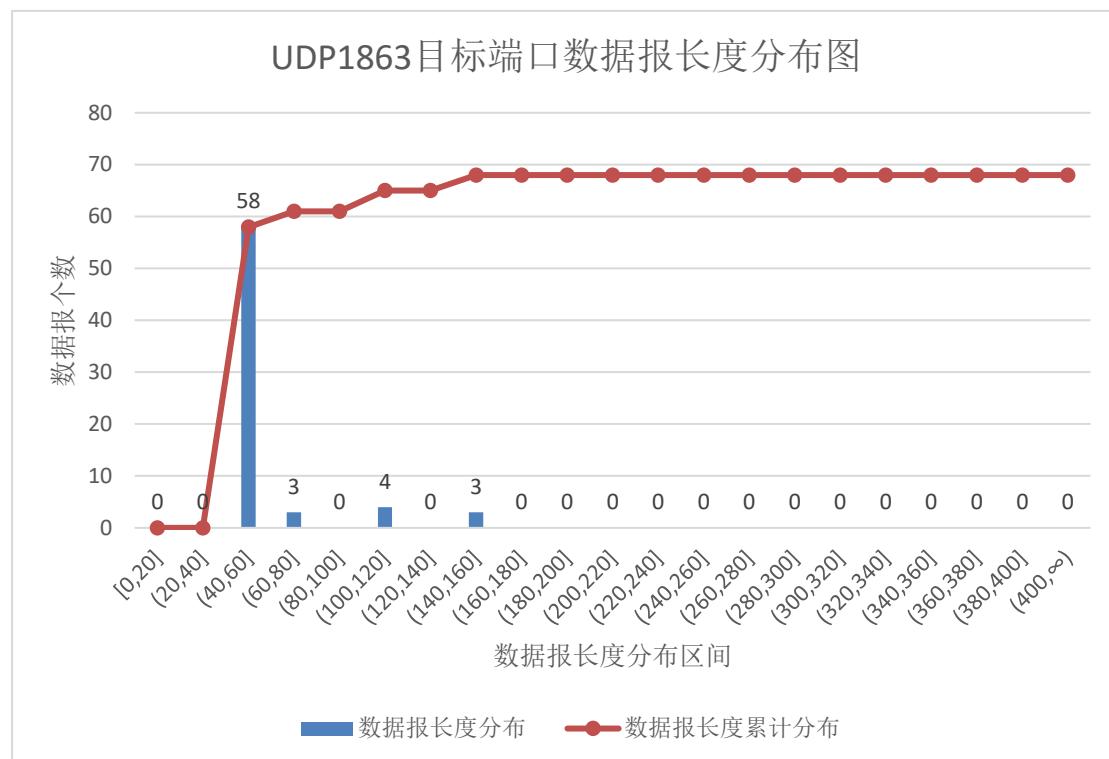


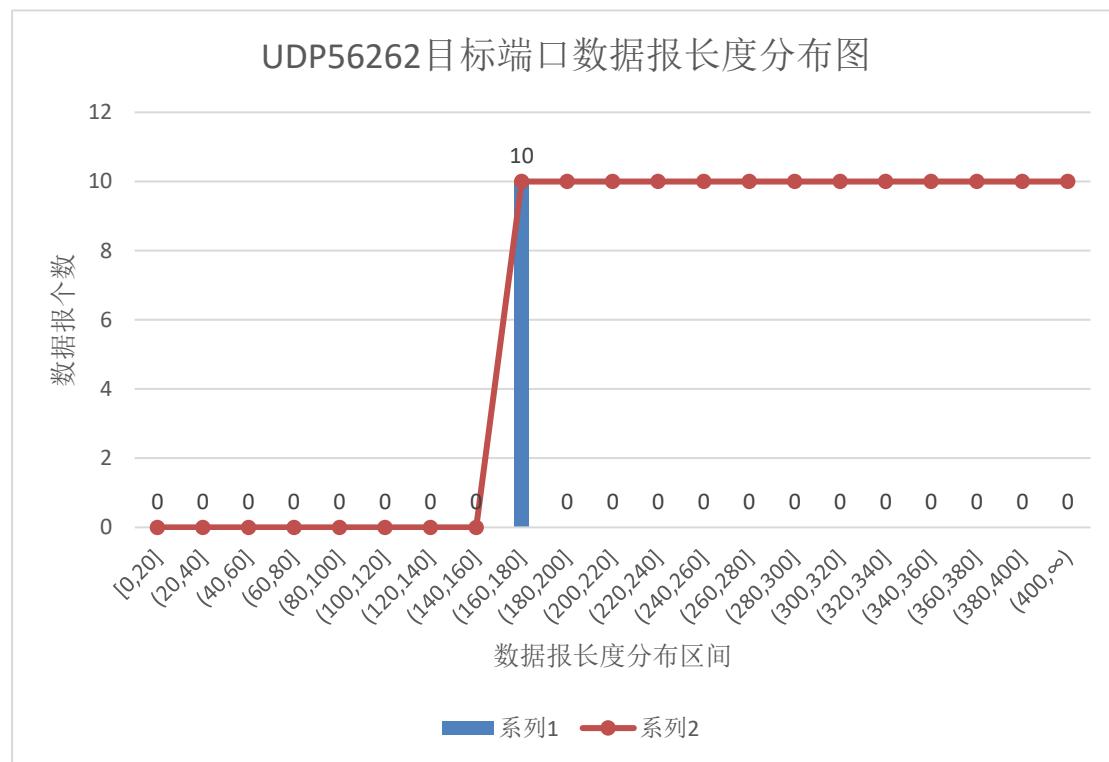
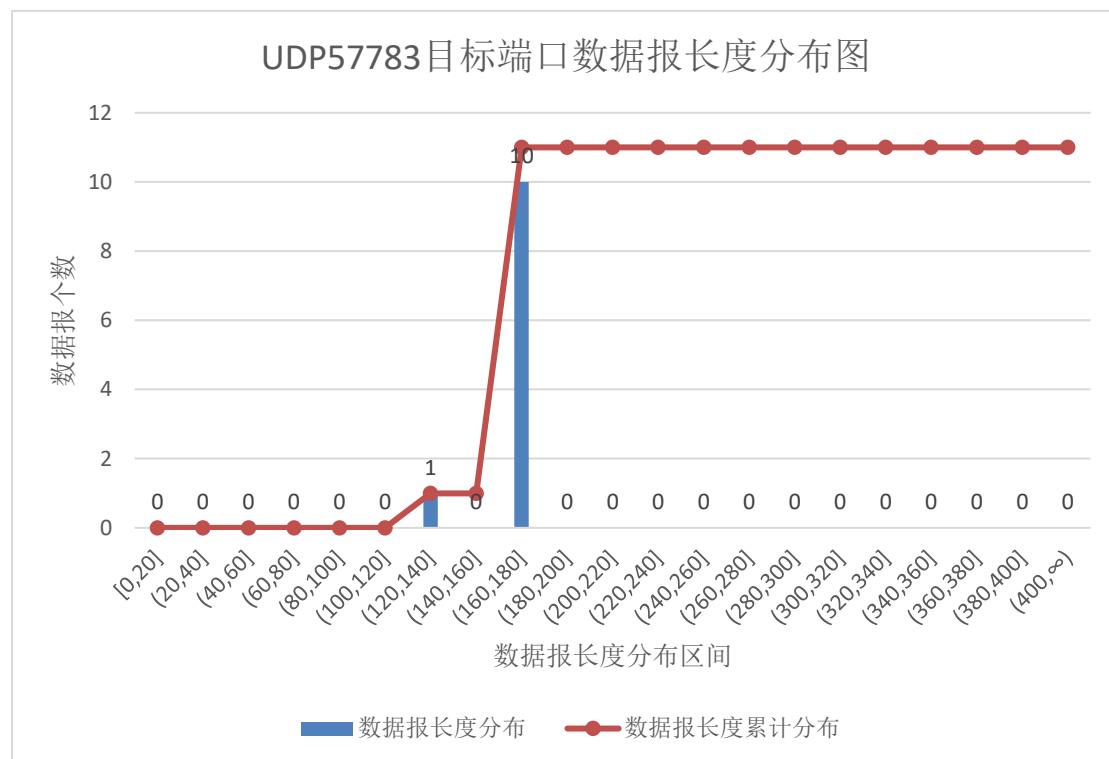
**[4] UDP 目标端口数前 10 的数据报长度分布情况:**





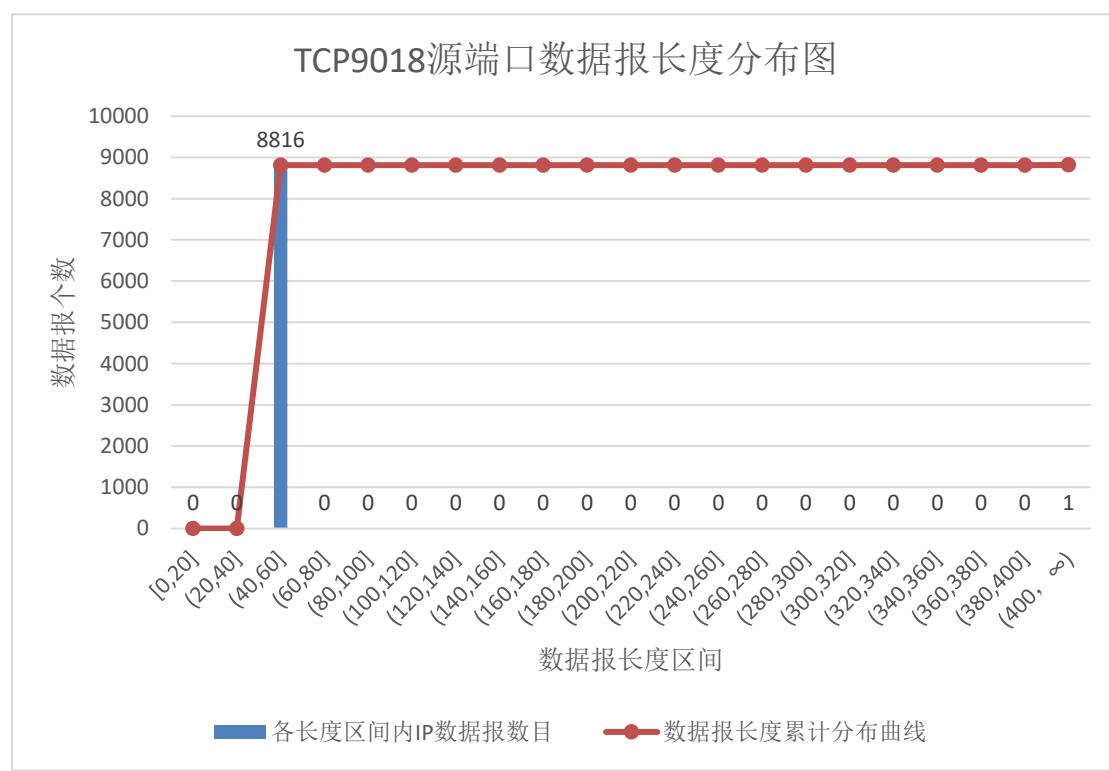
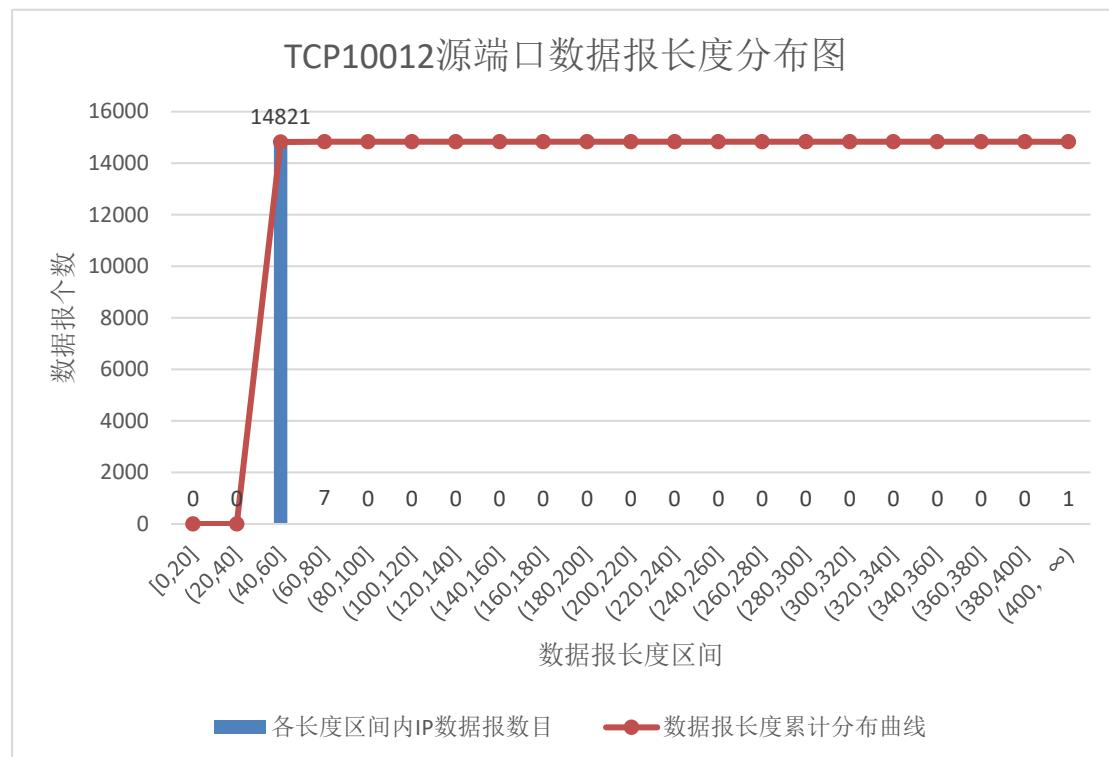


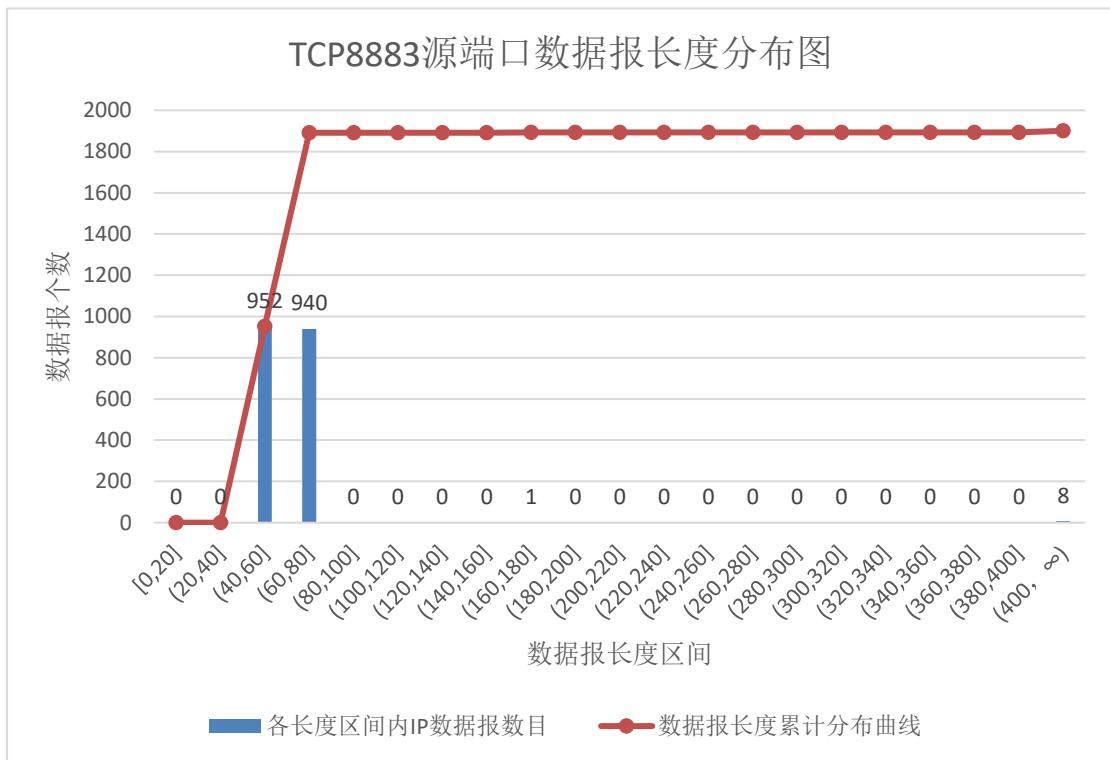
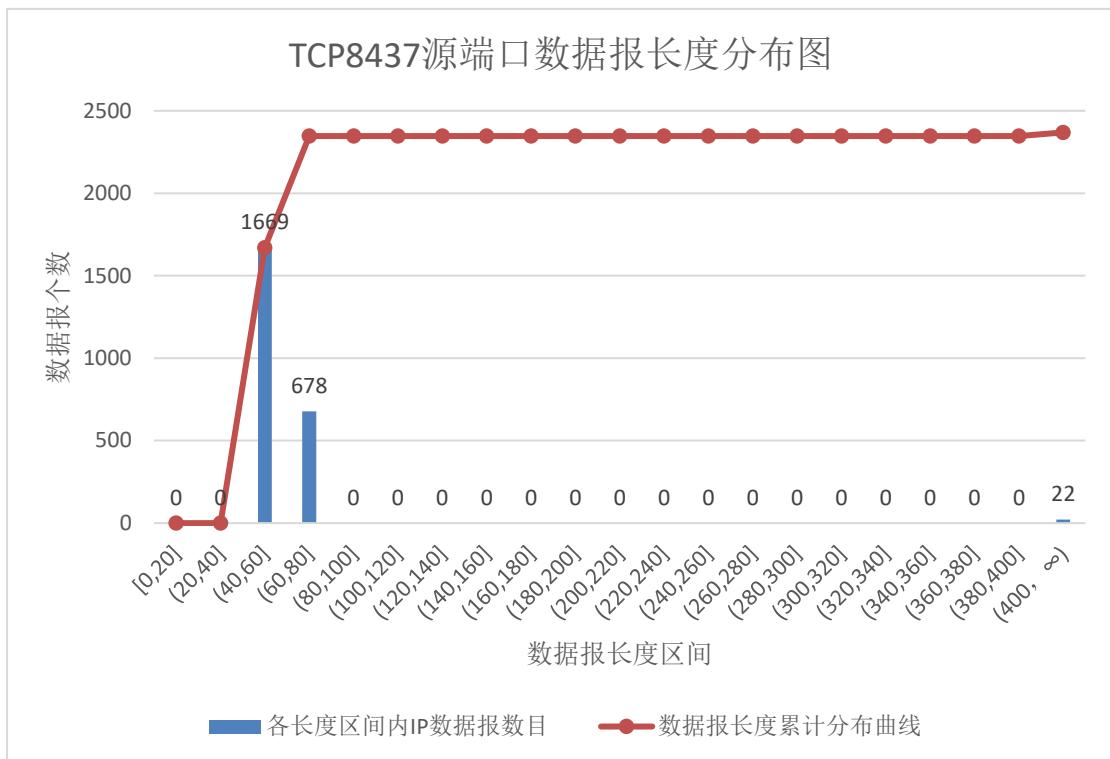


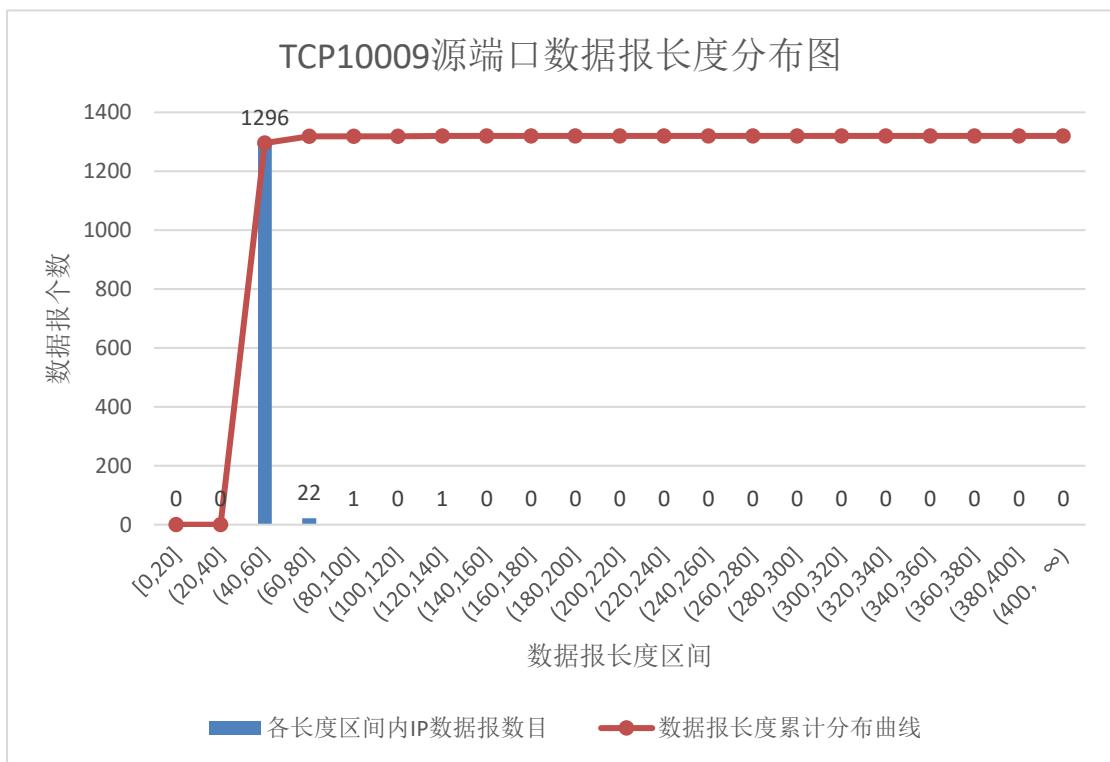
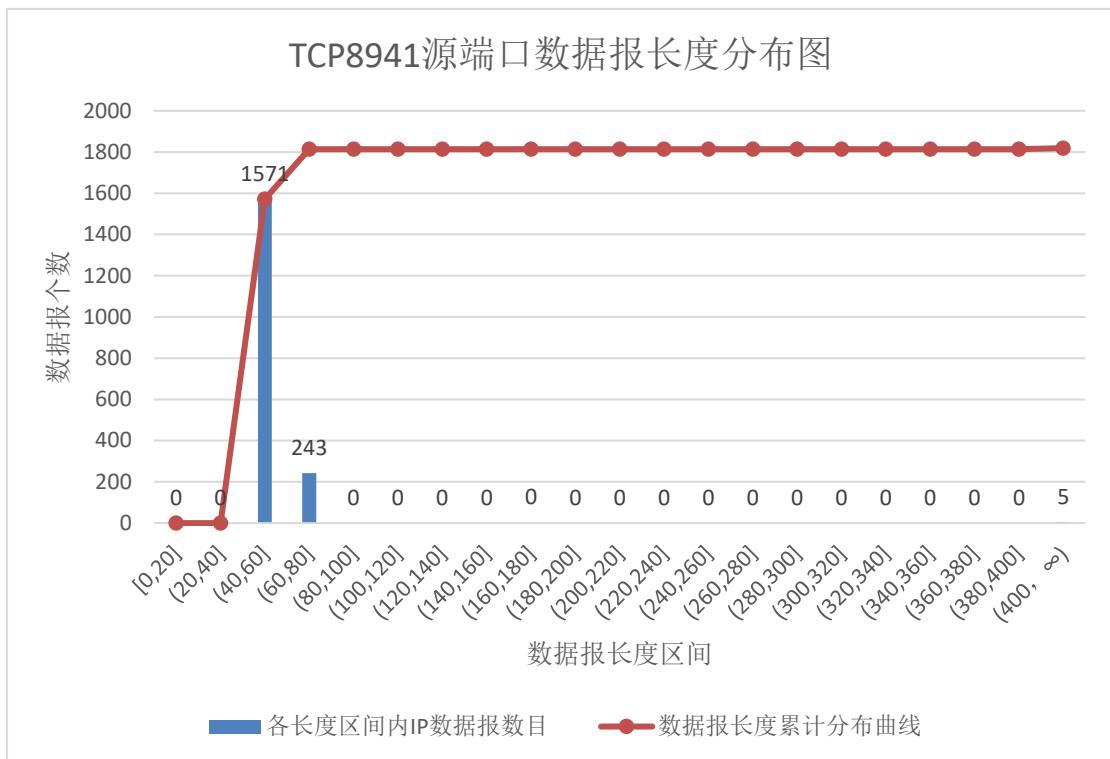


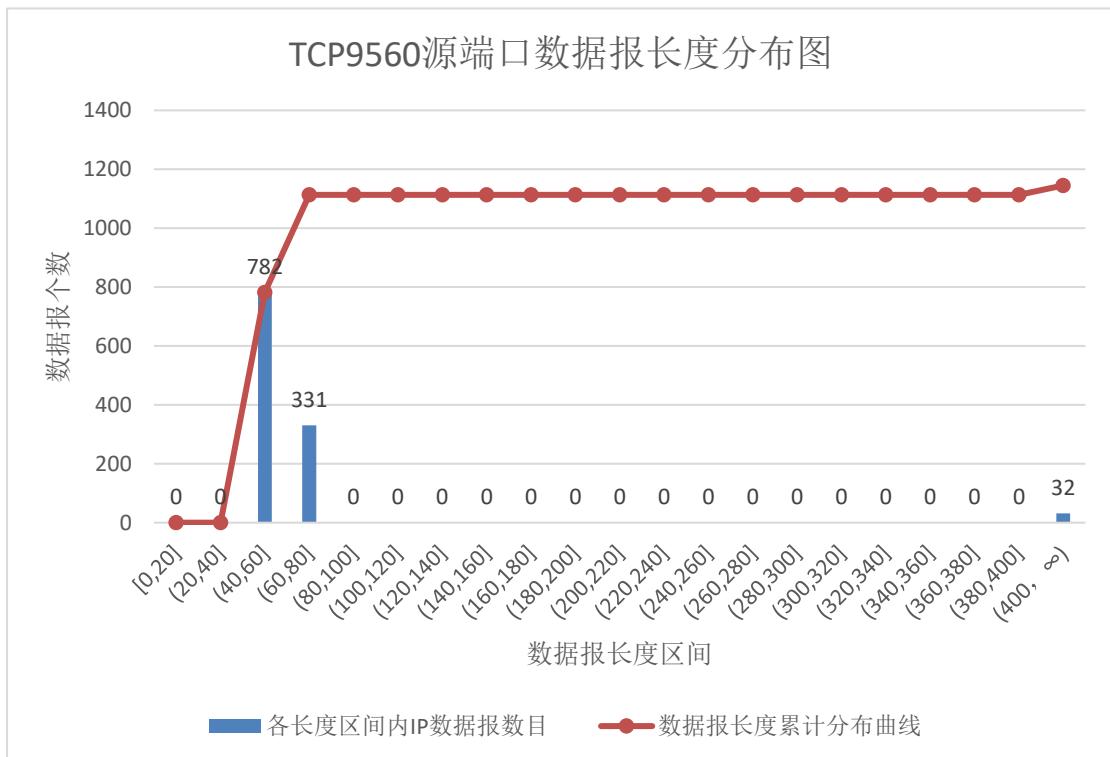
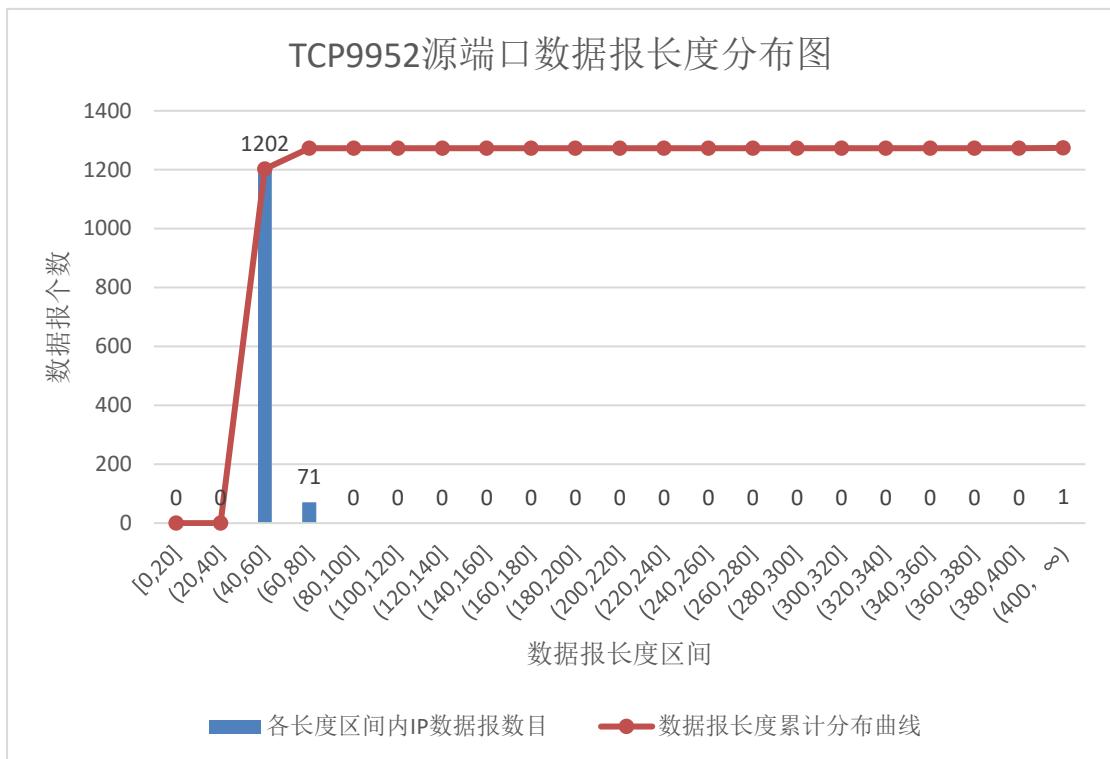
## Output:

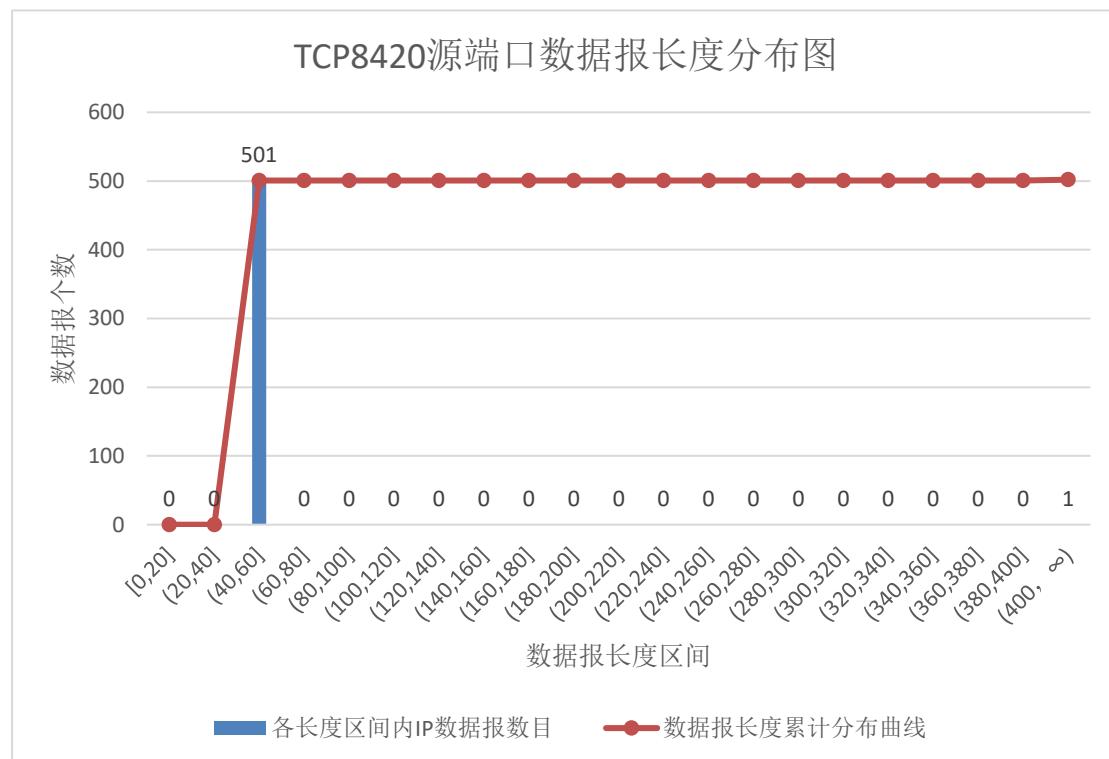
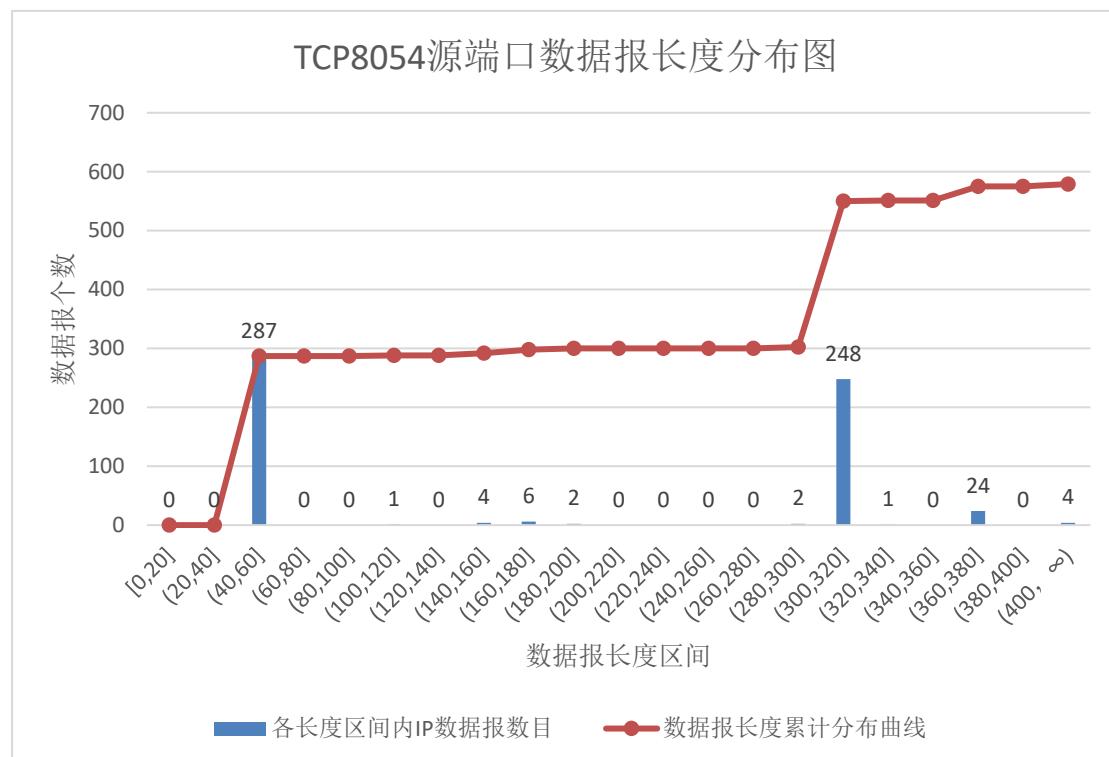
[1]TCP 源端口前十的端口数据报长度分布情况:



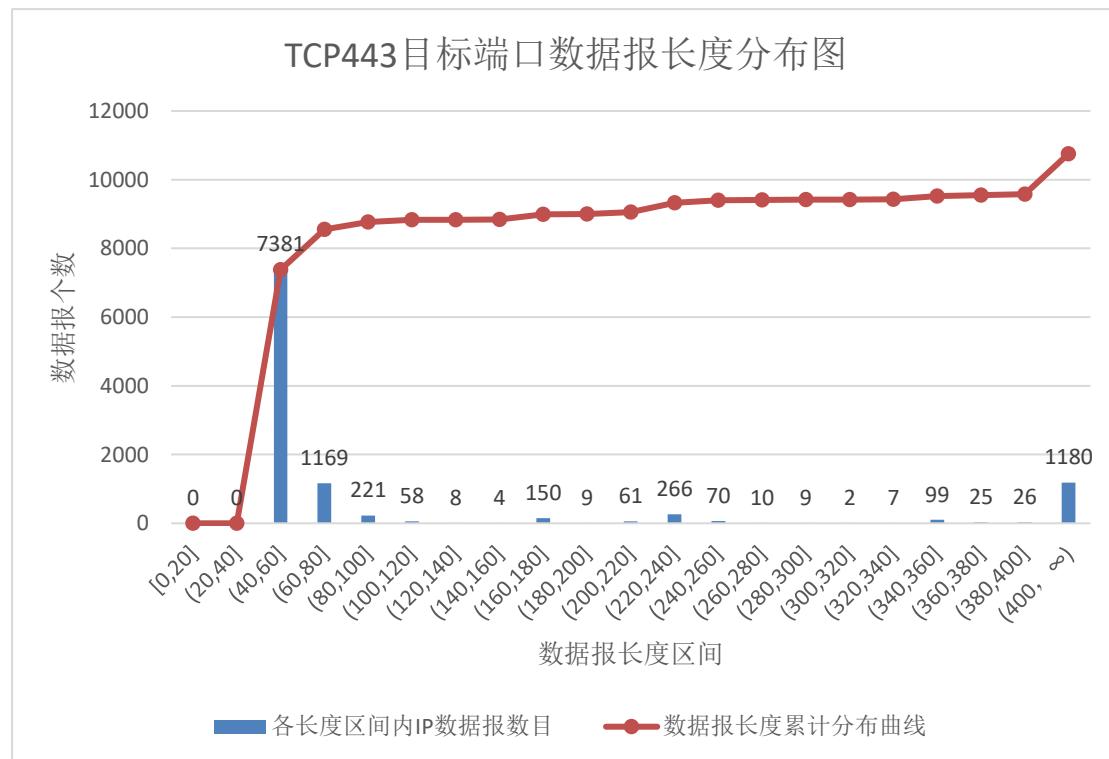
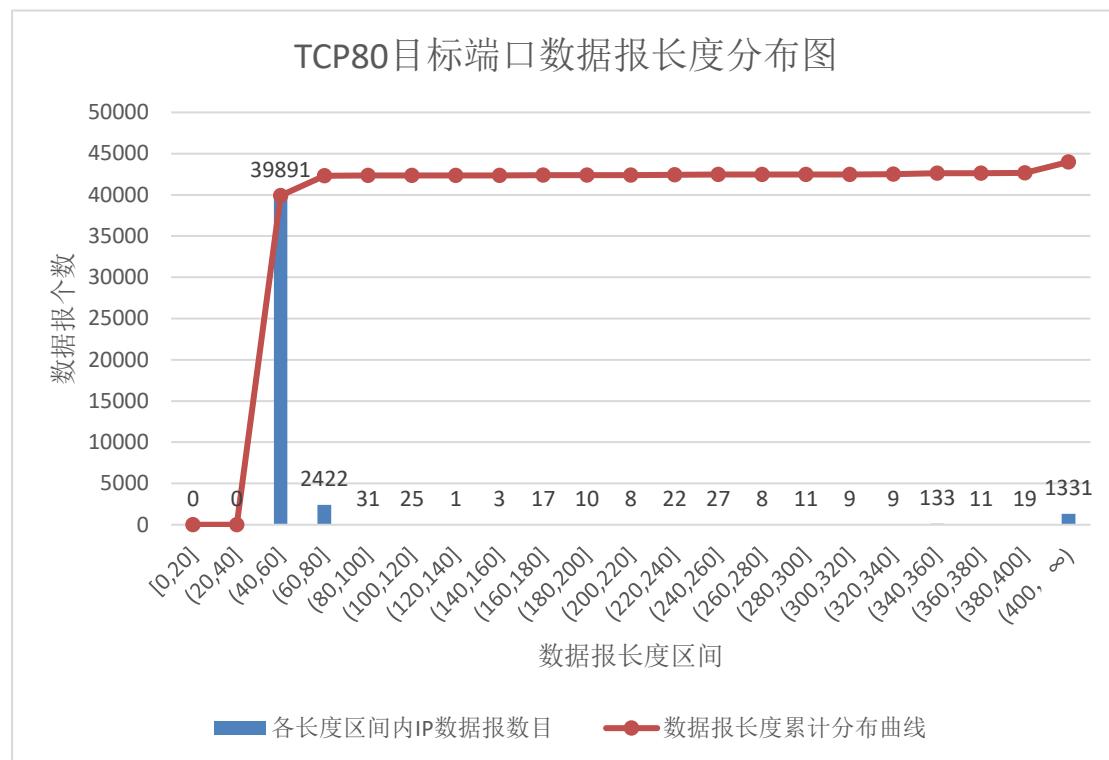


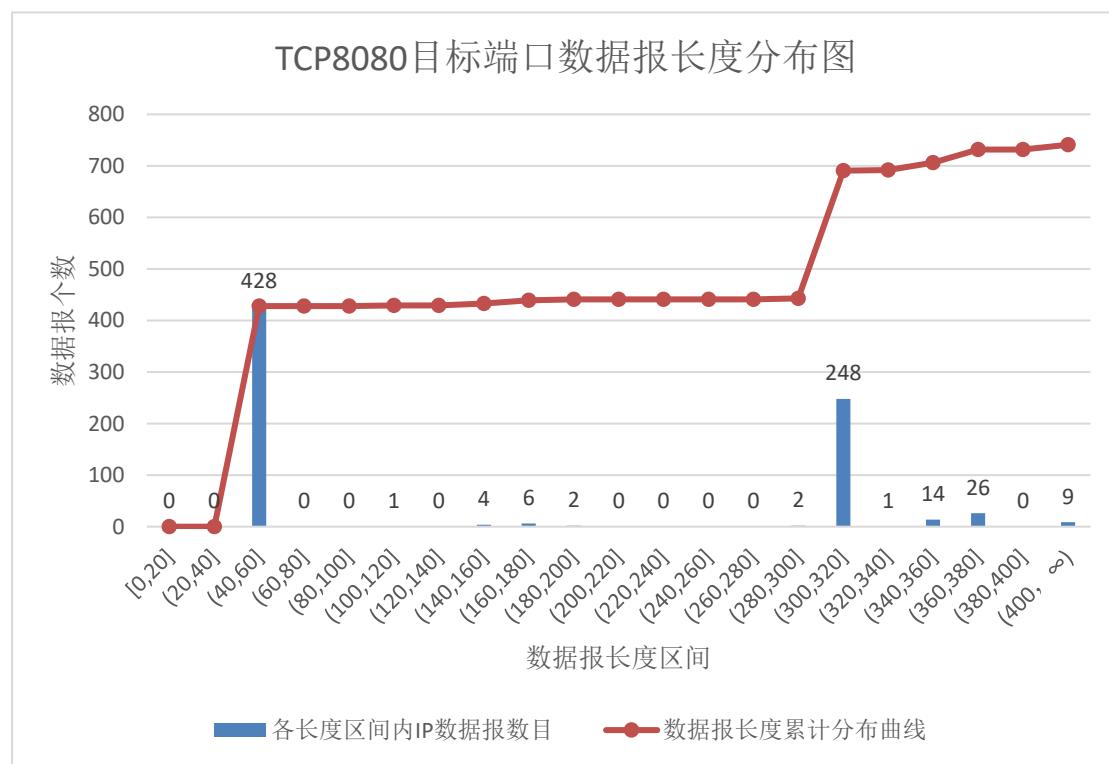
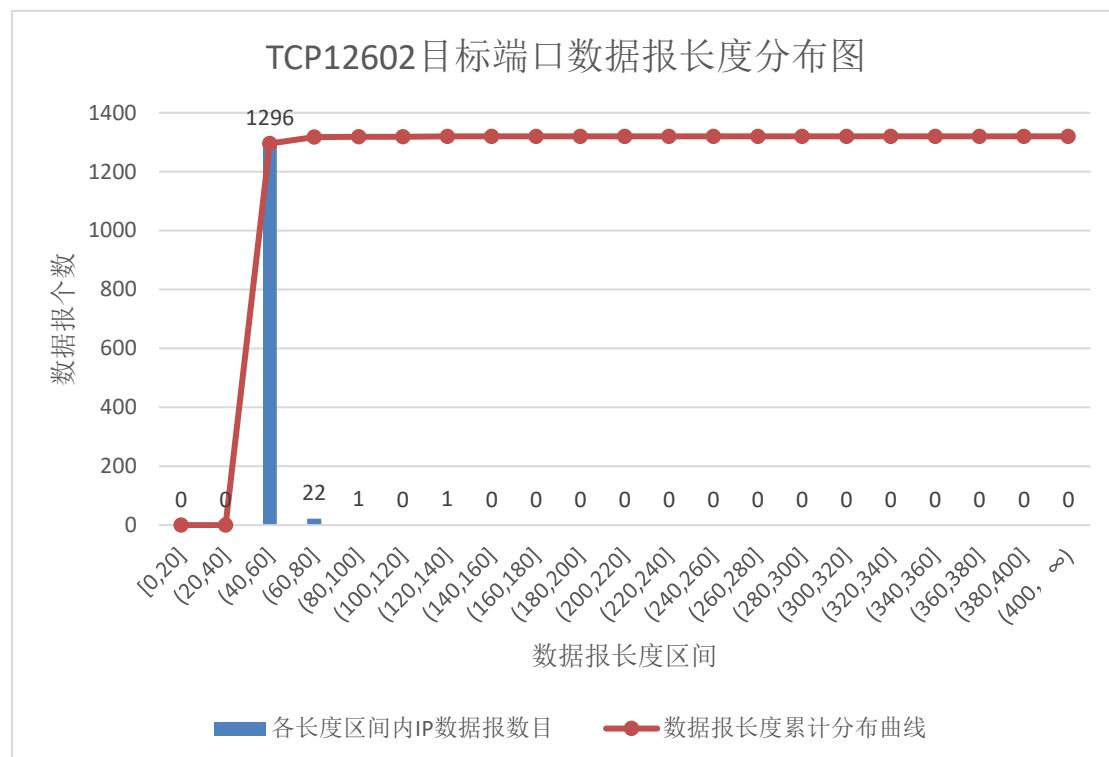


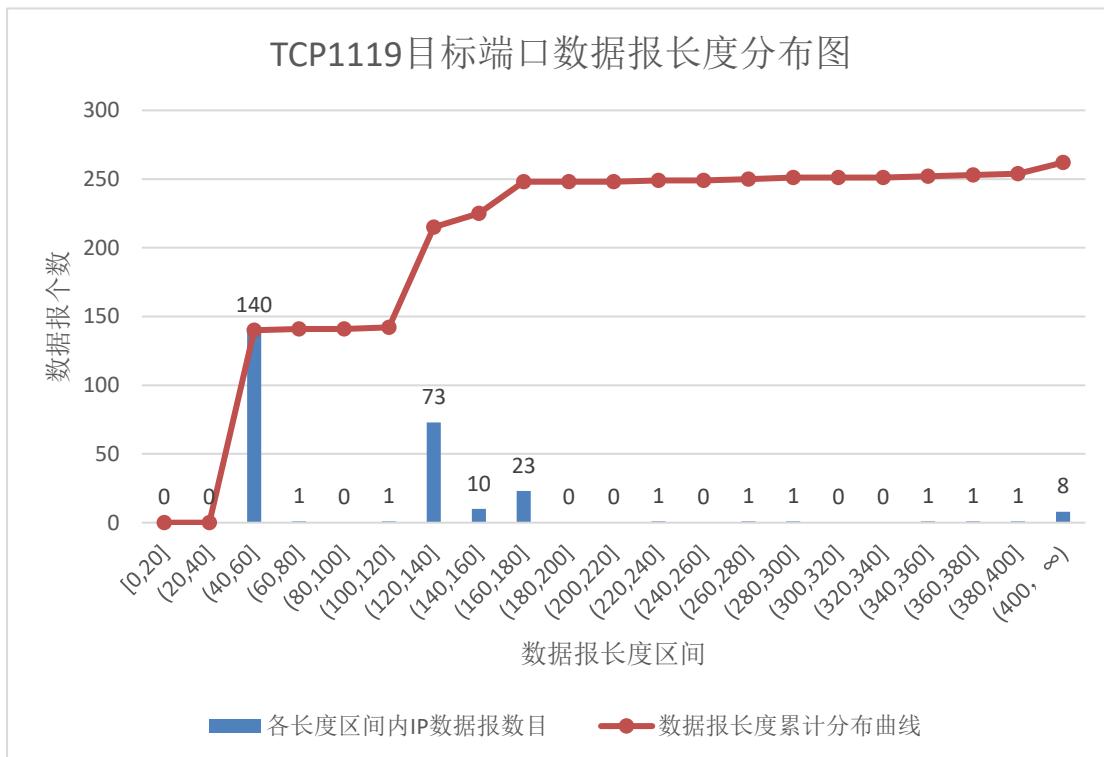
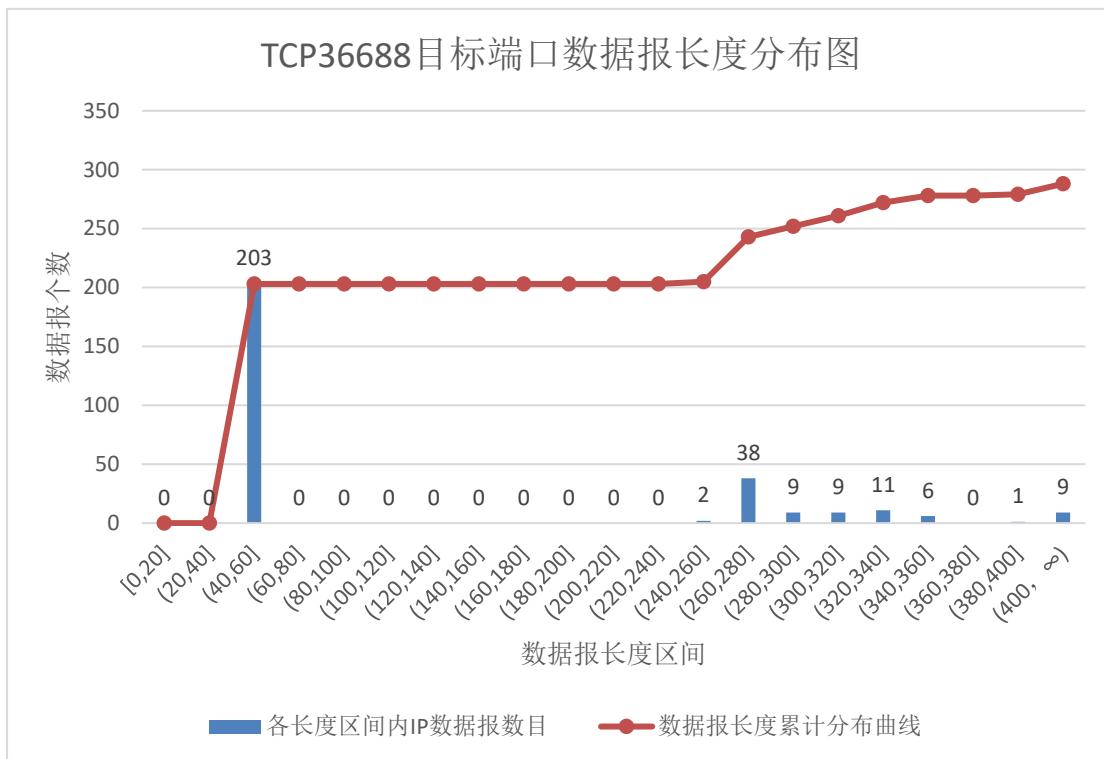


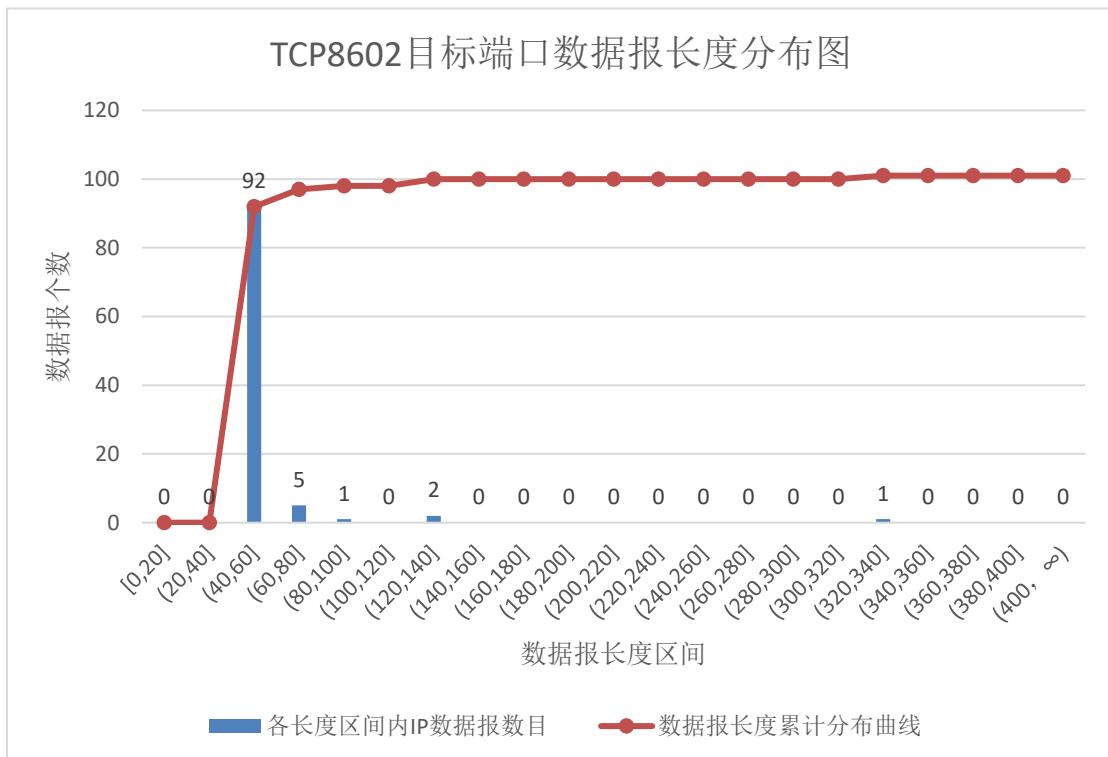
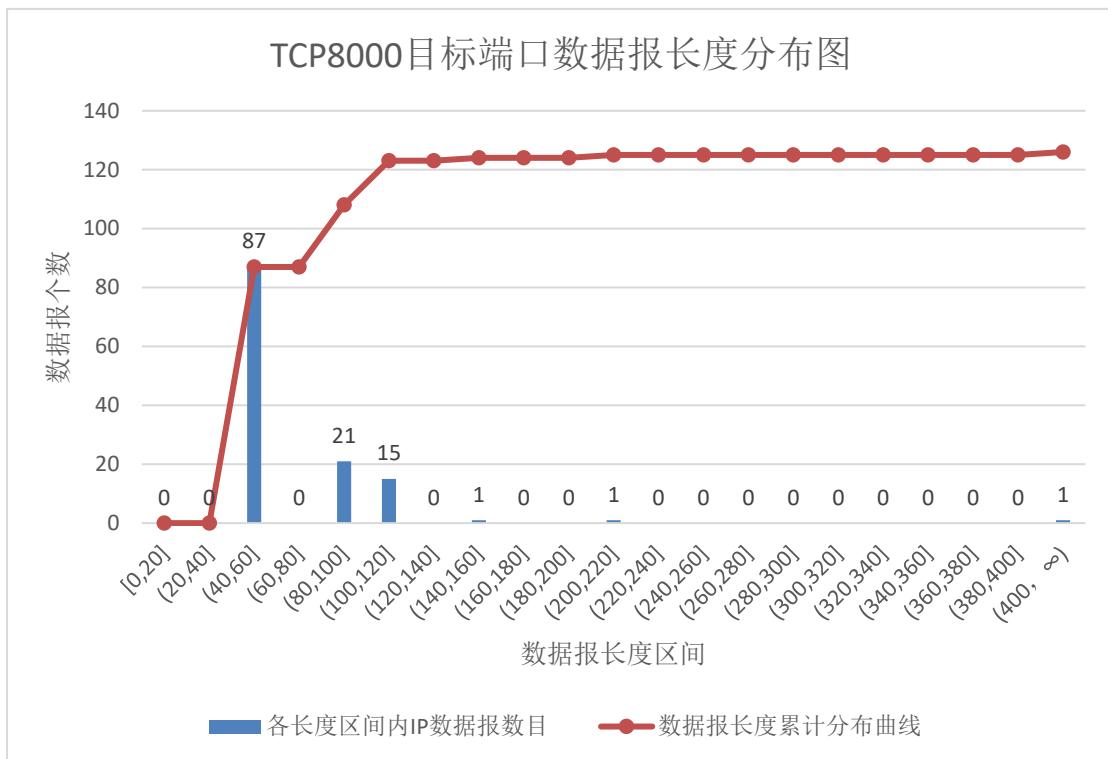


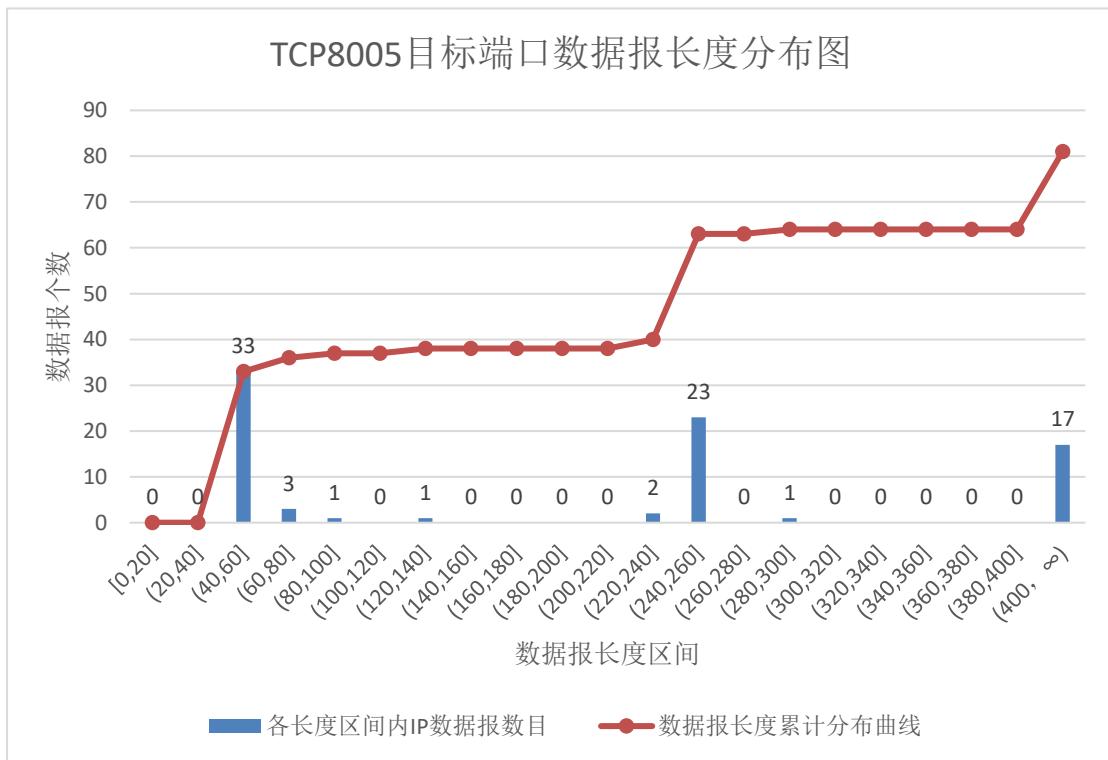
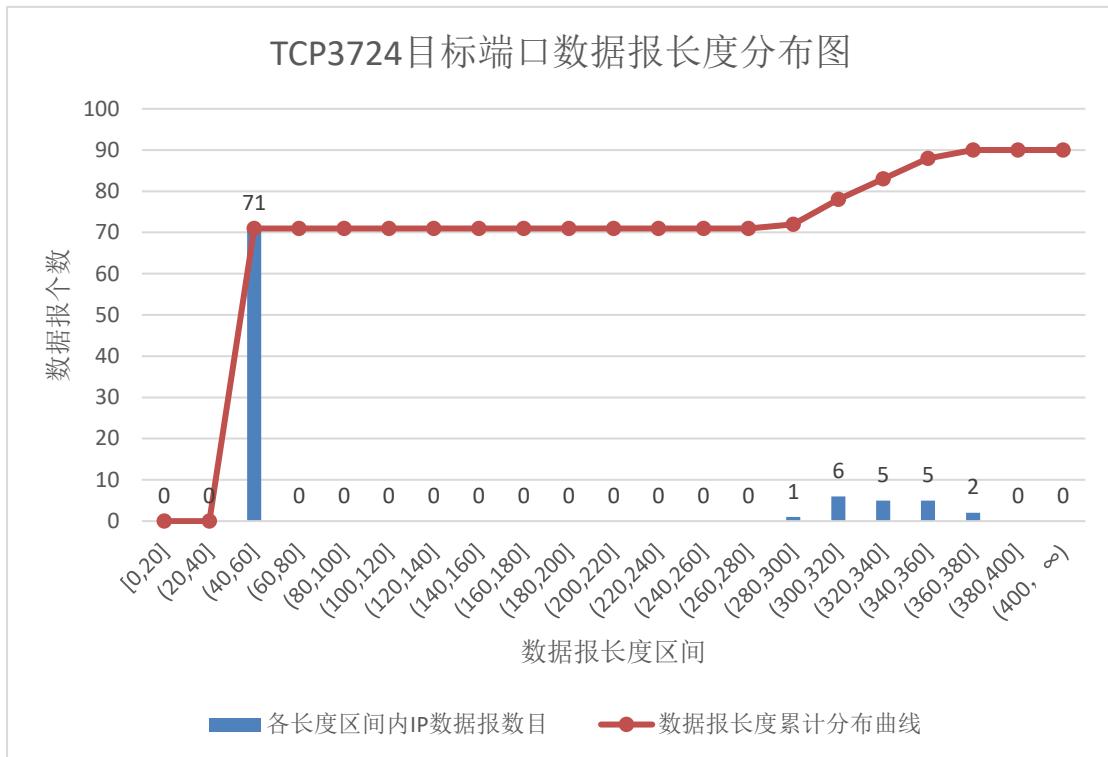
[2]TCP 目标端口数前十的数据报长度分布情况:



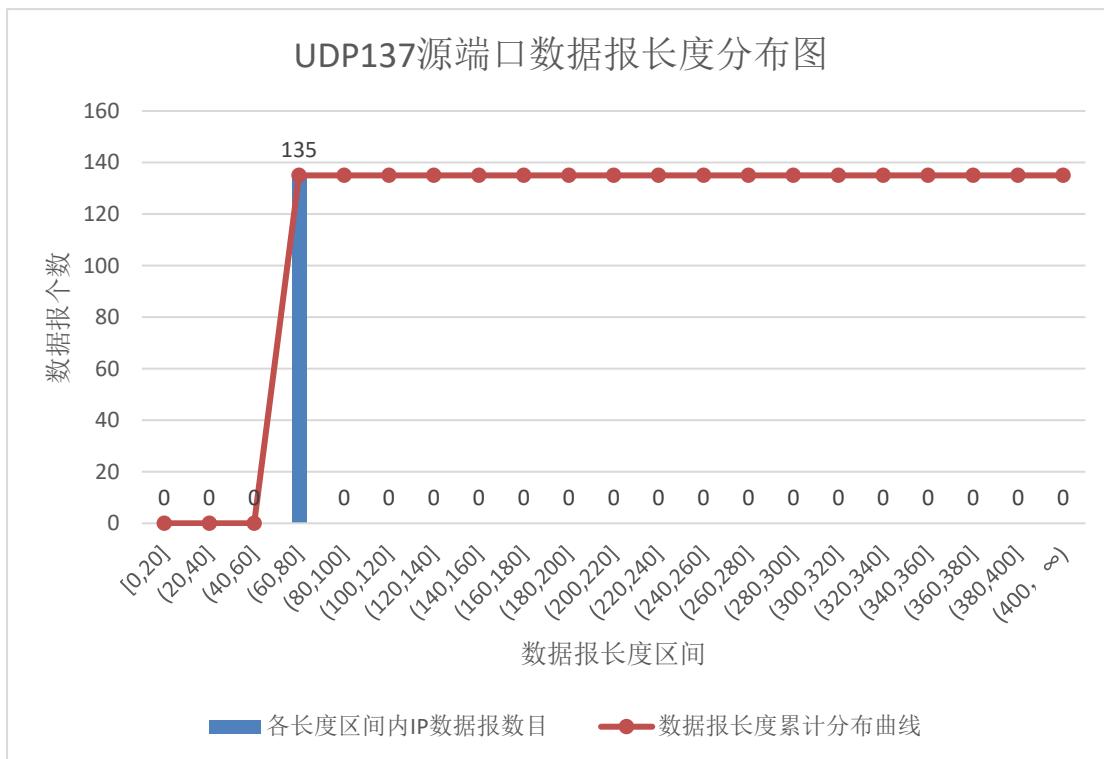
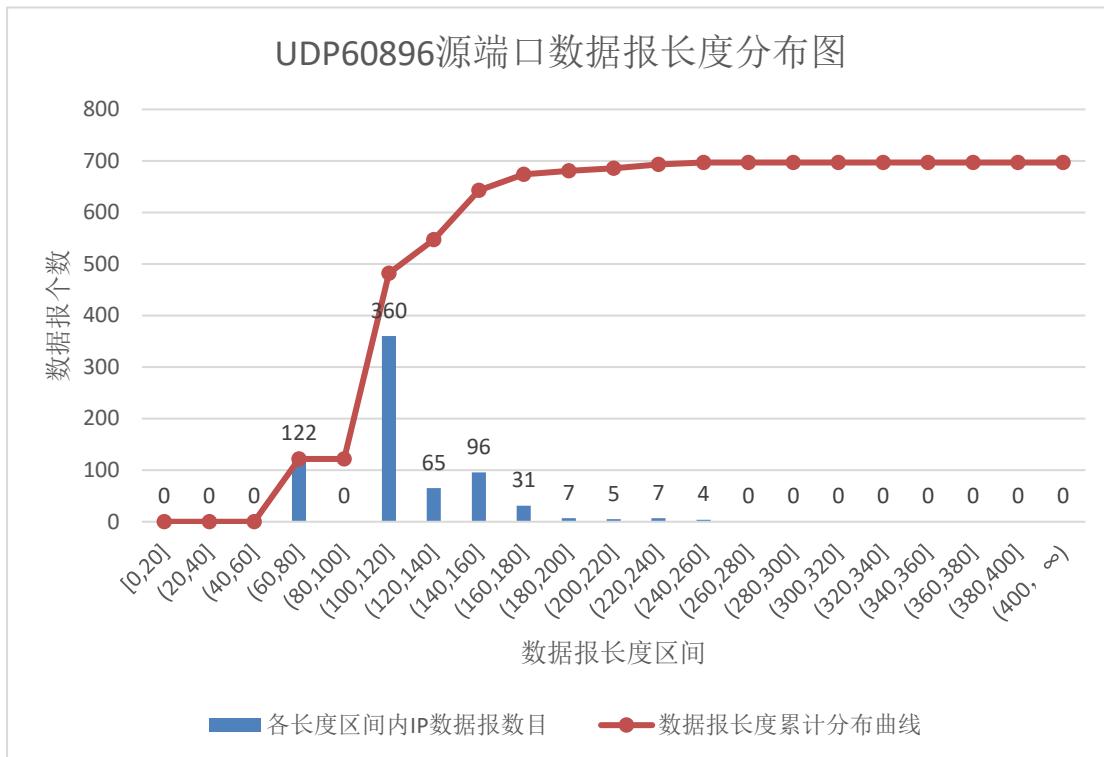


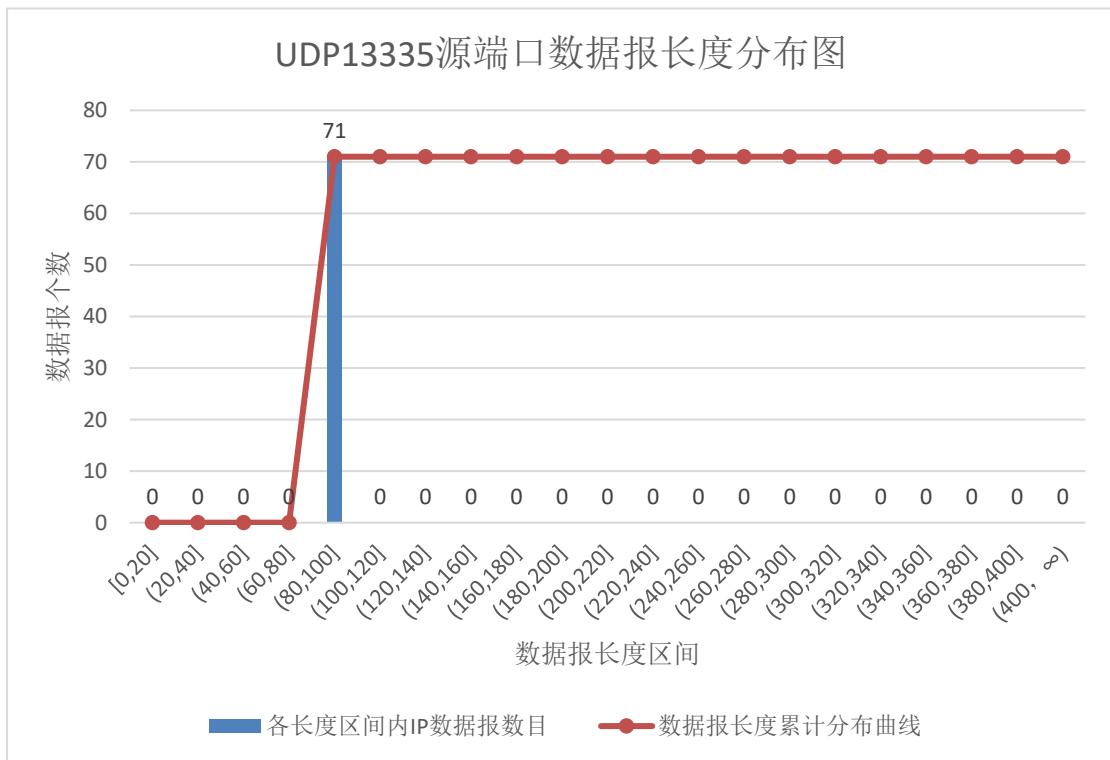
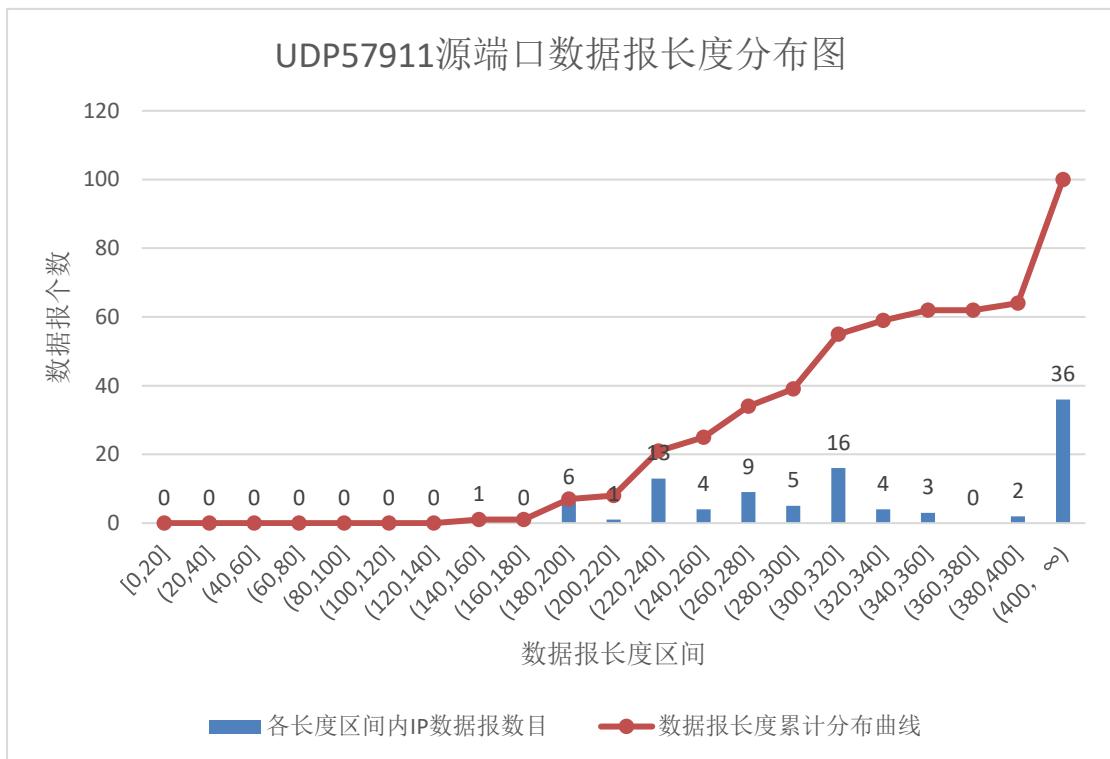


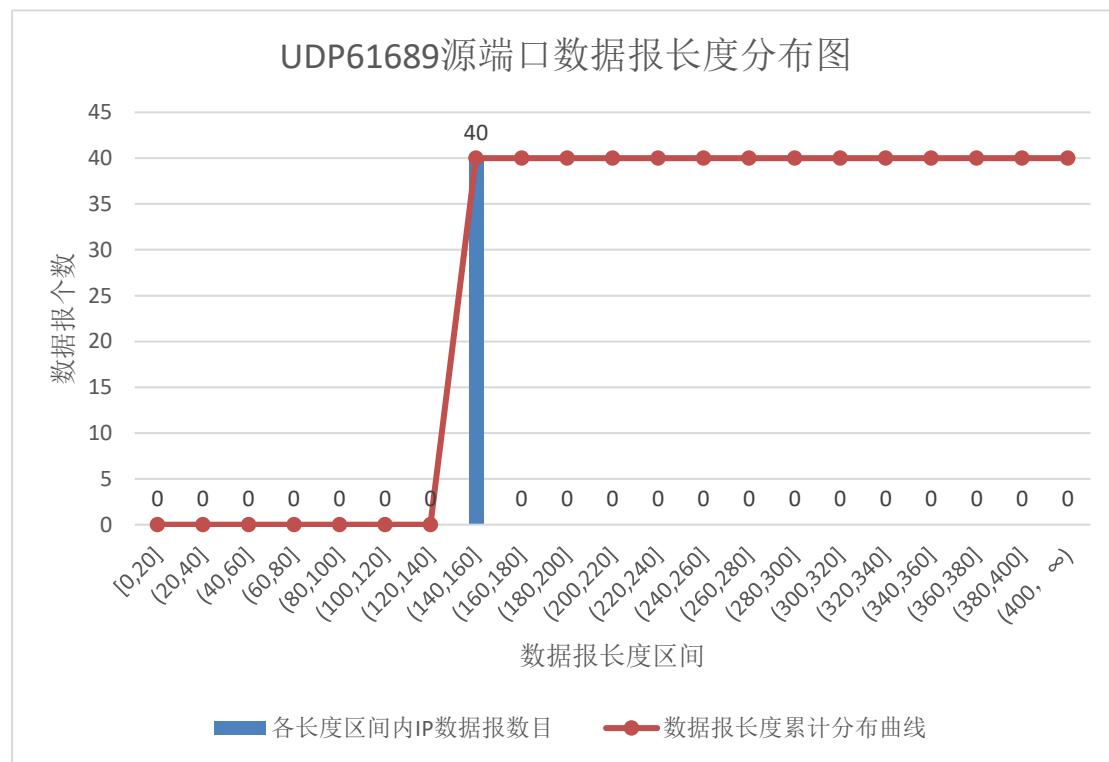
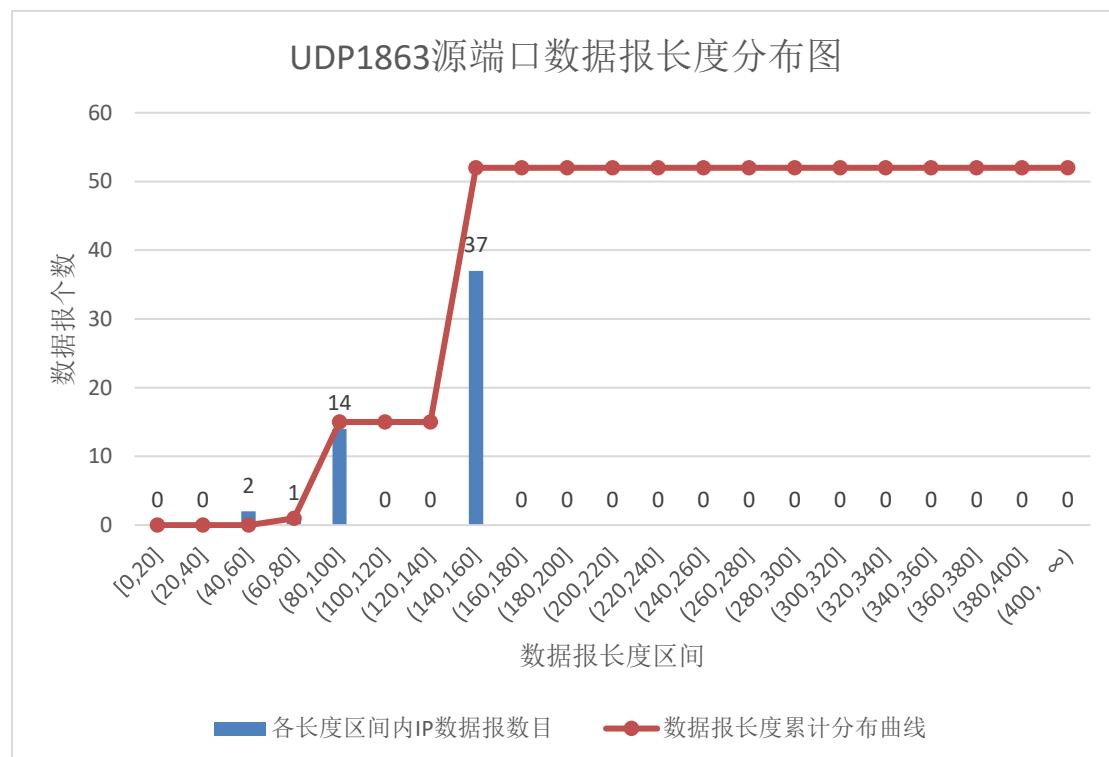


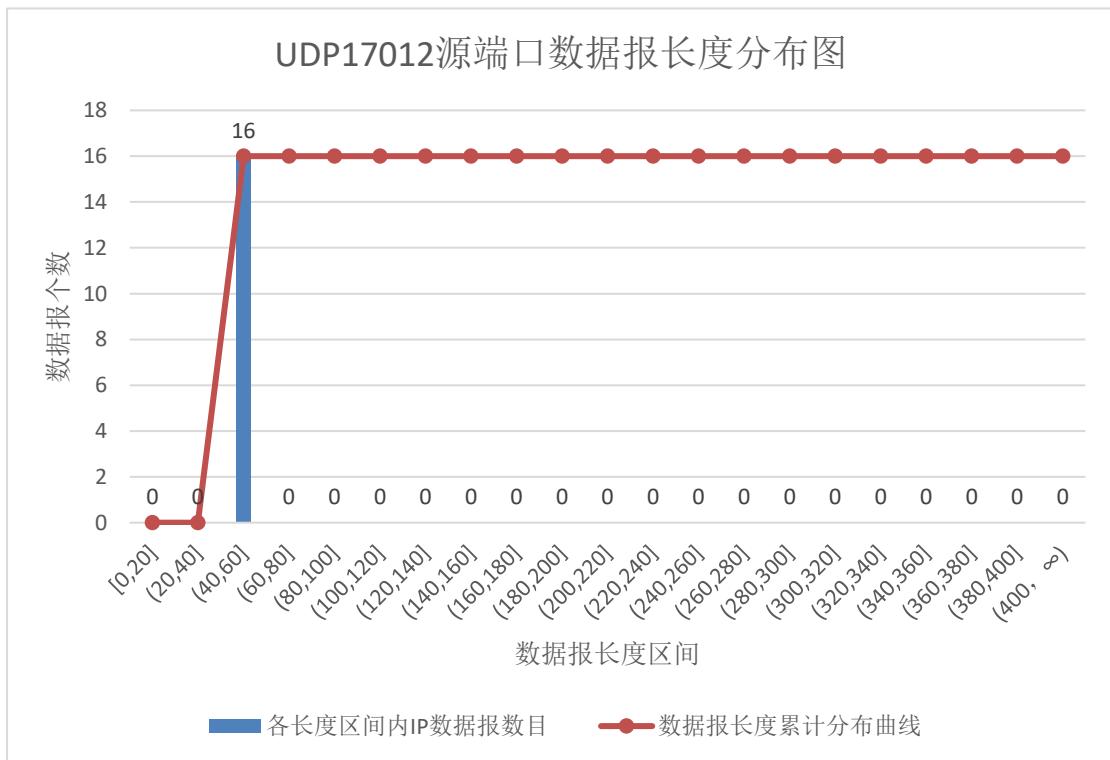
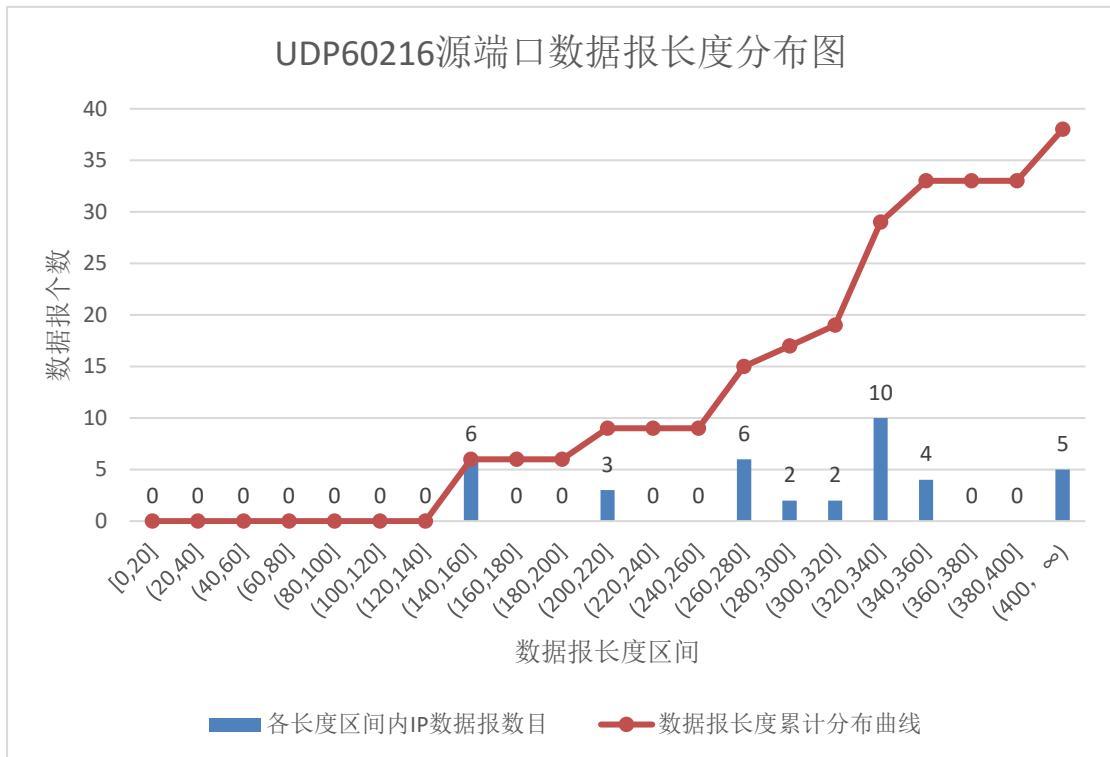


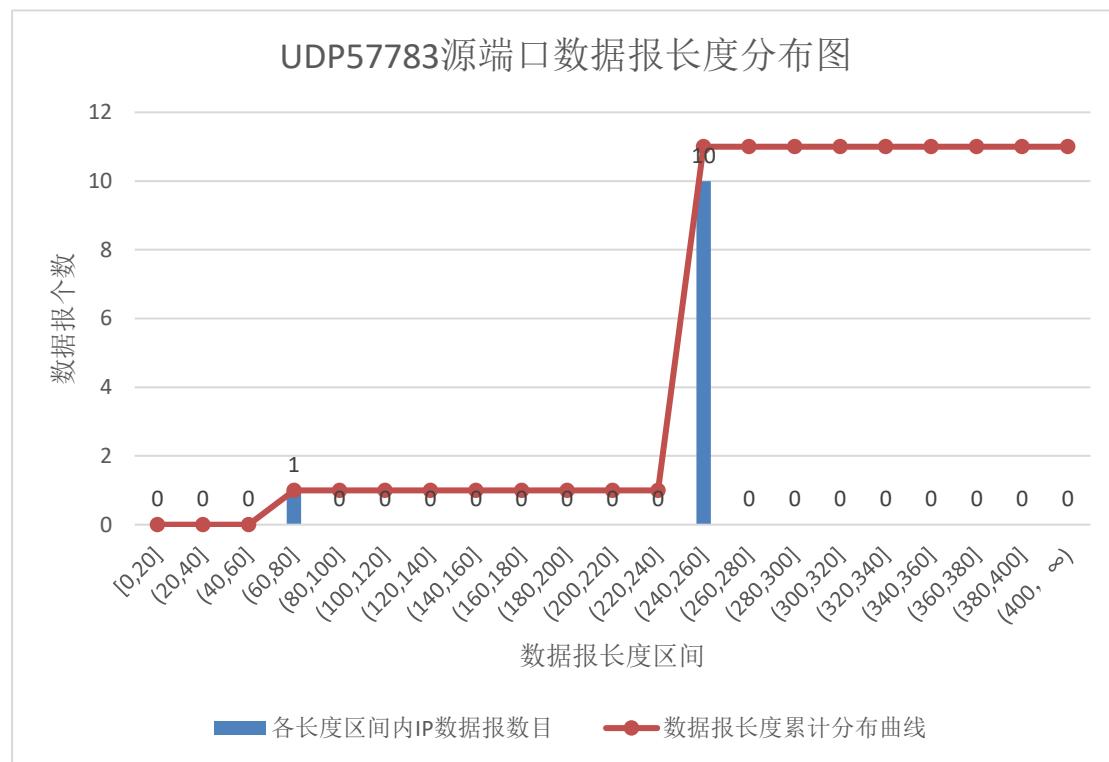
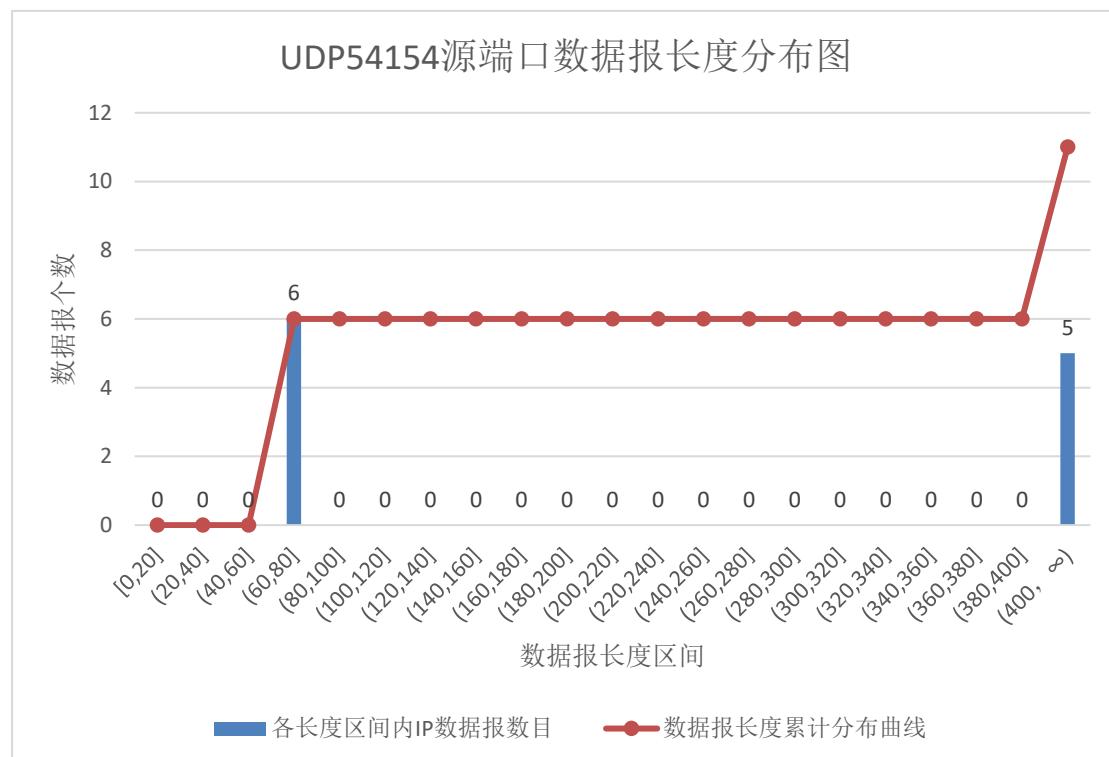
[3] UDP 源端口数前 10 的数据报长度分布情况:



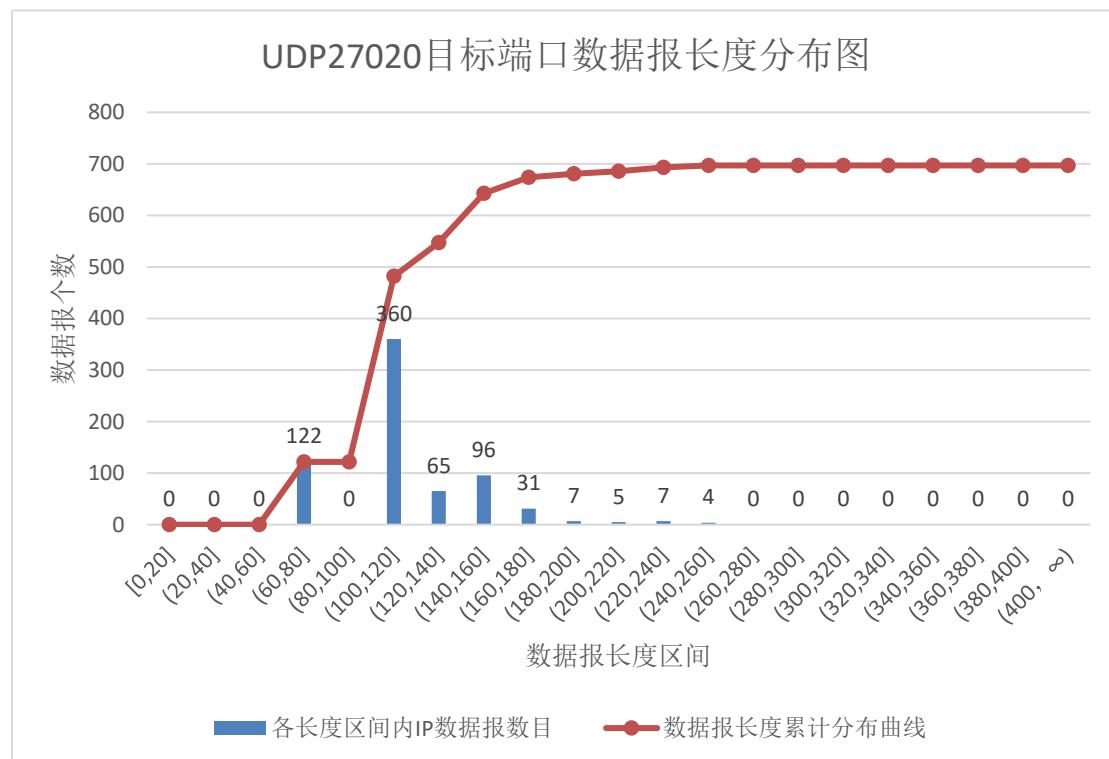
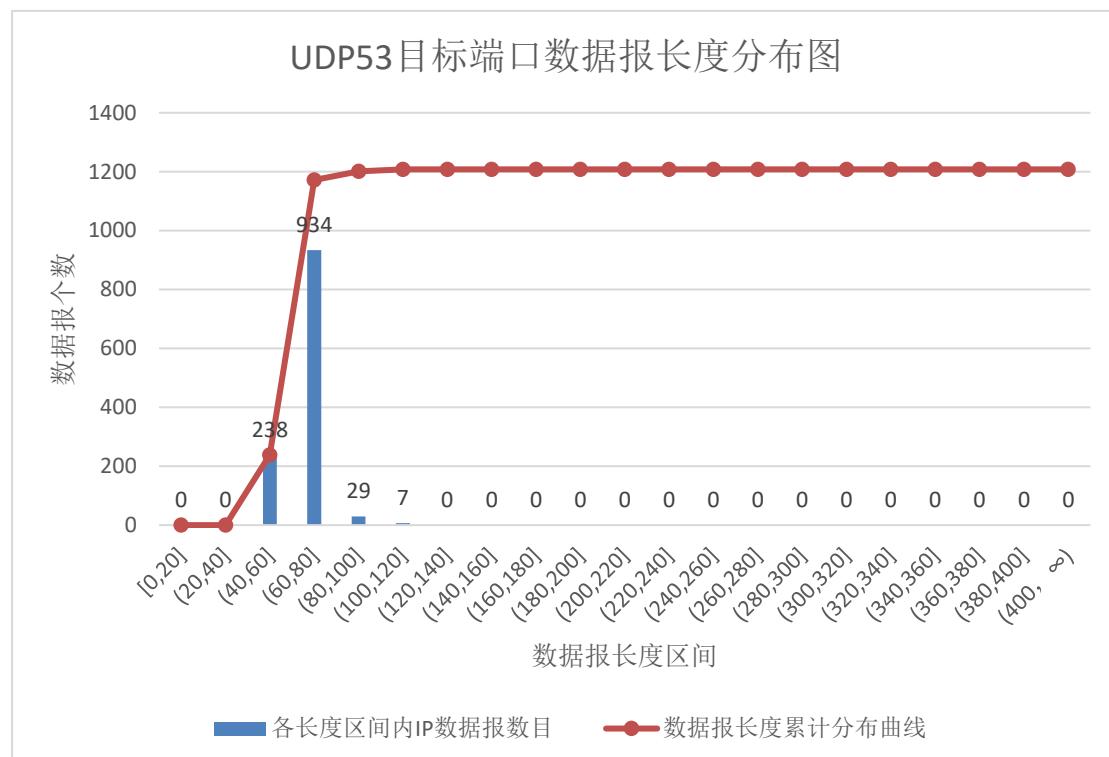


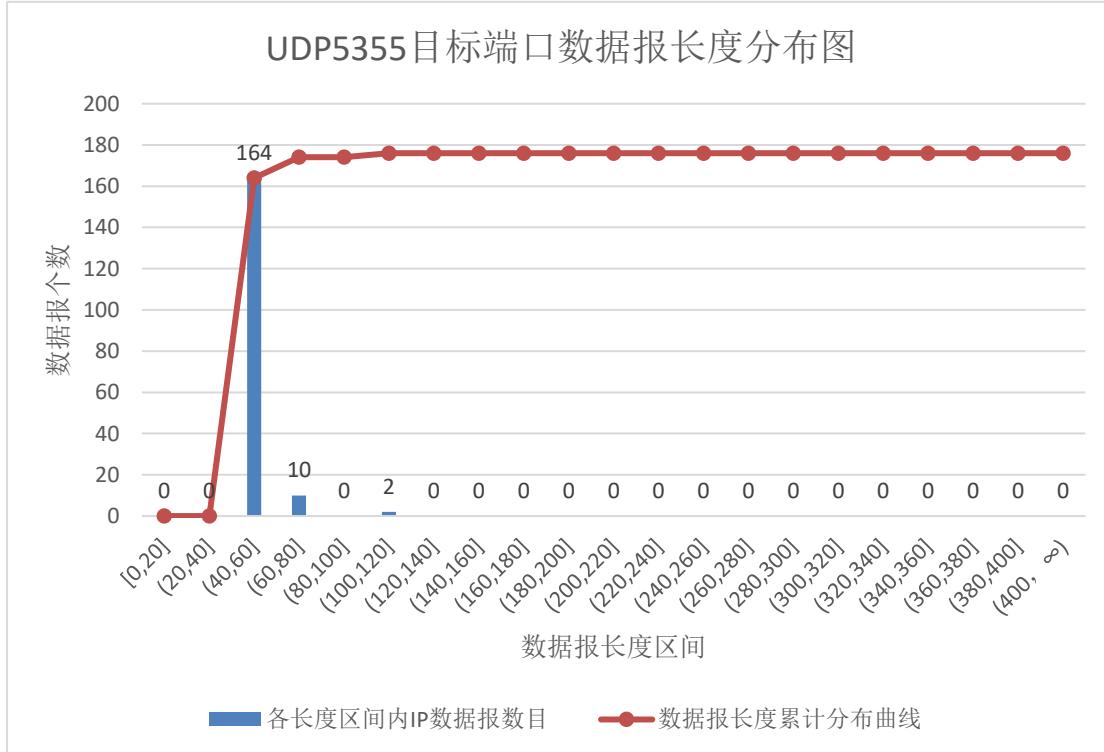
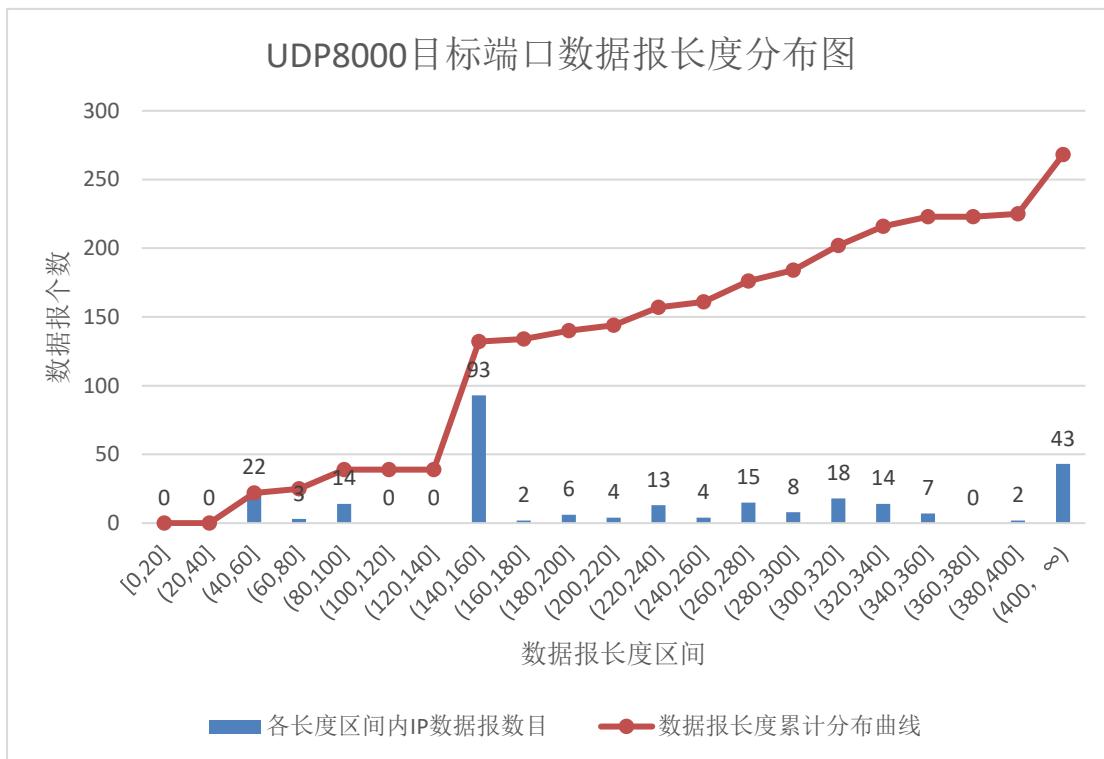


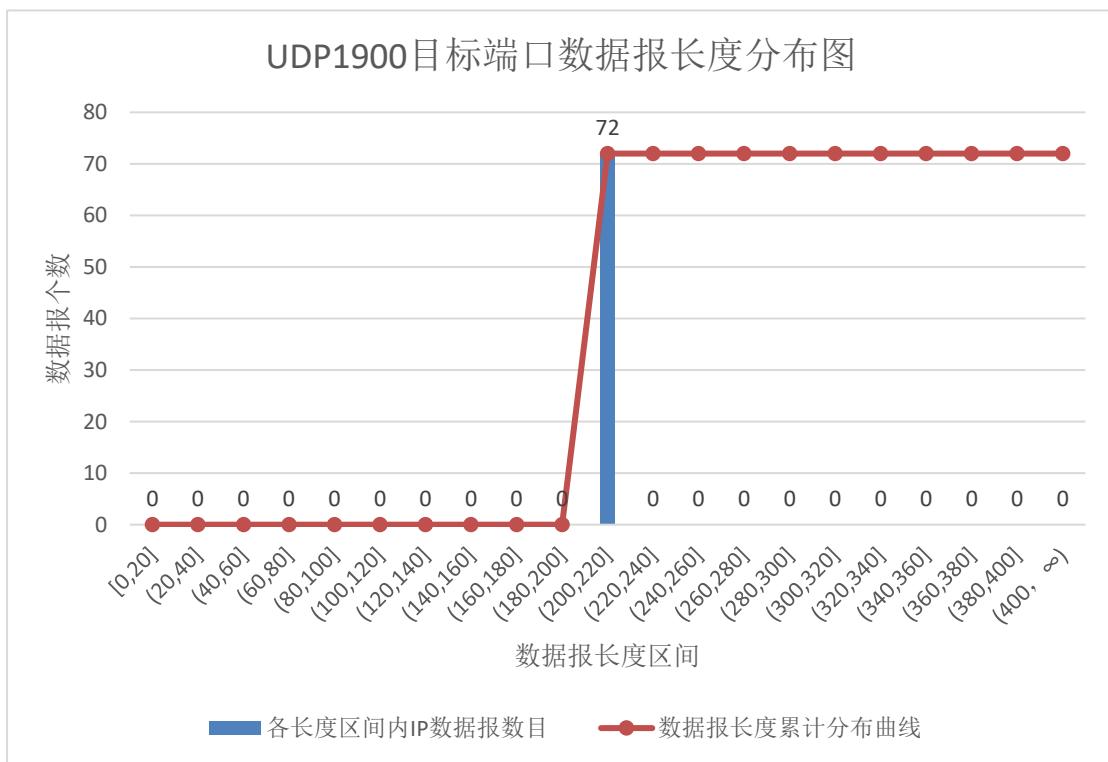
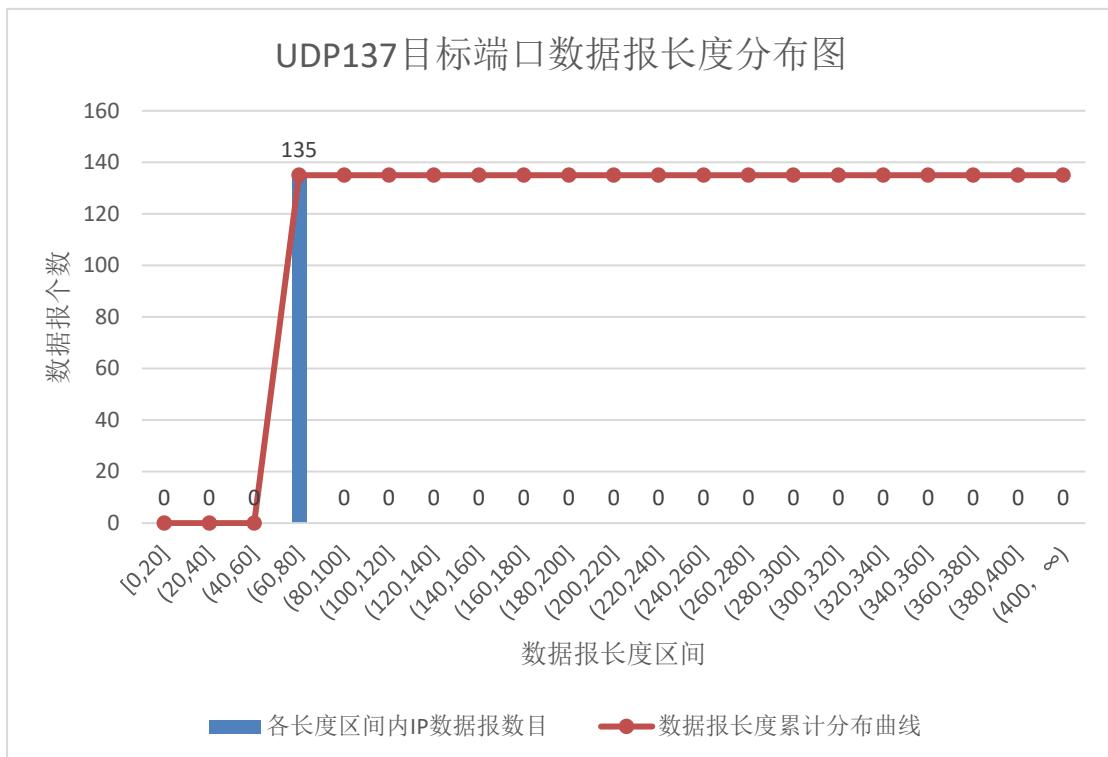


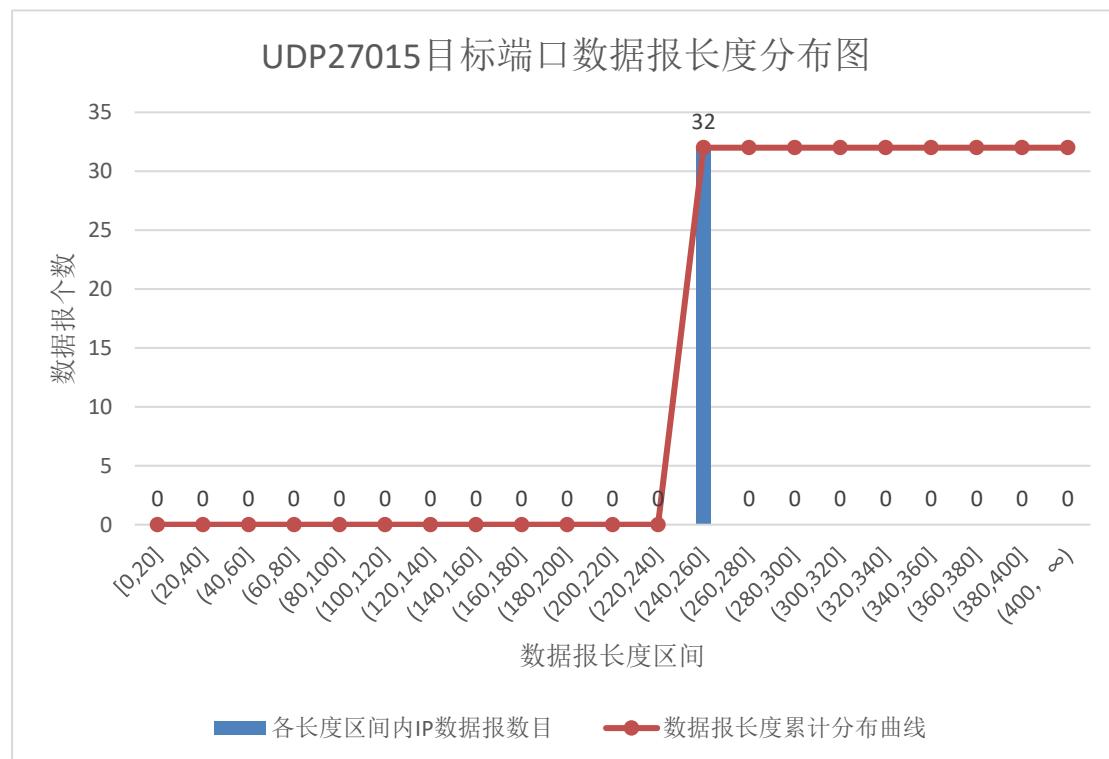
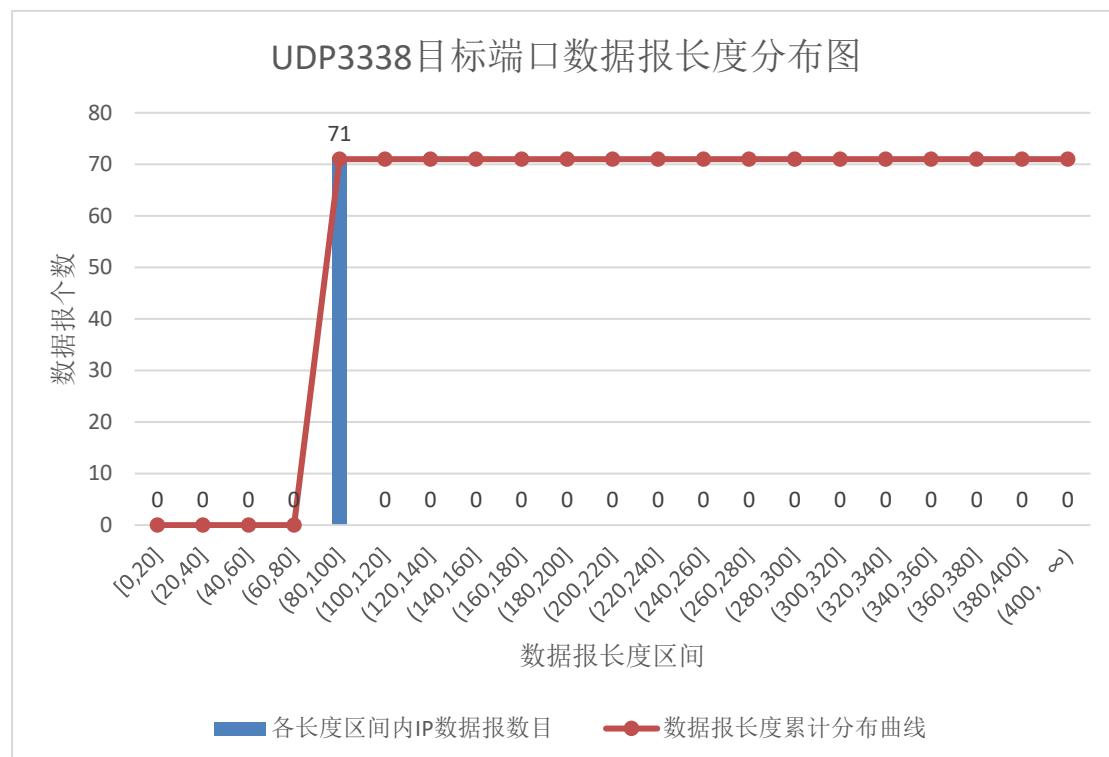


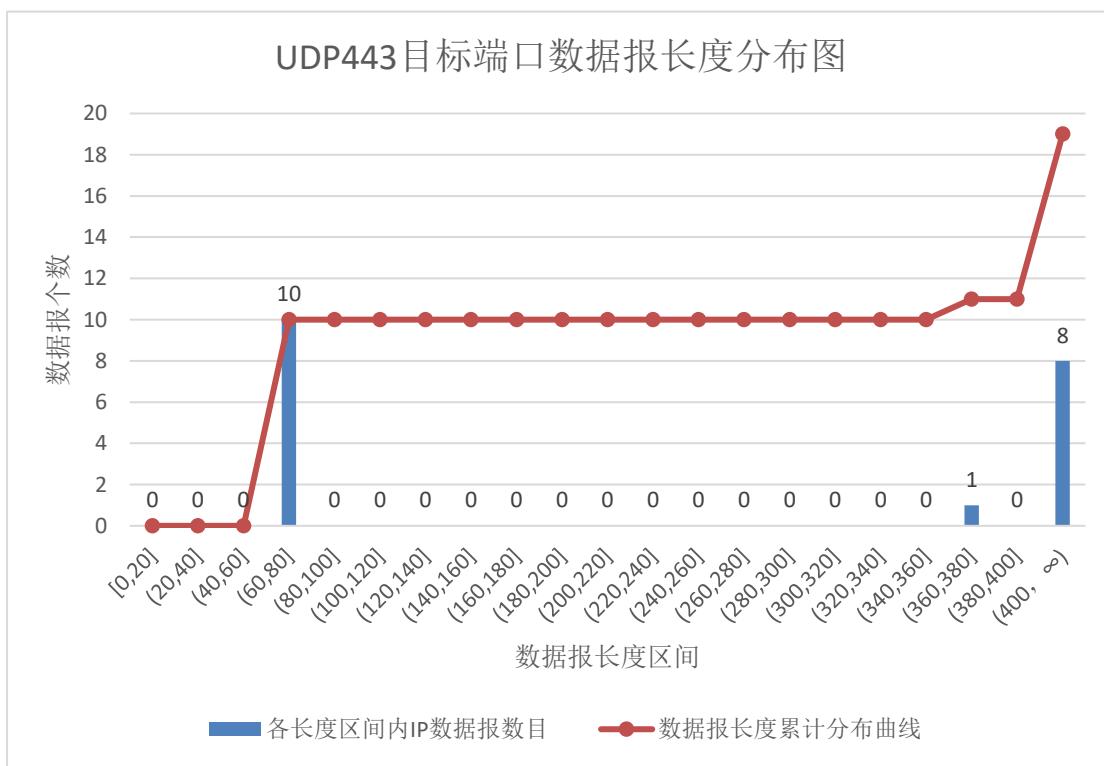
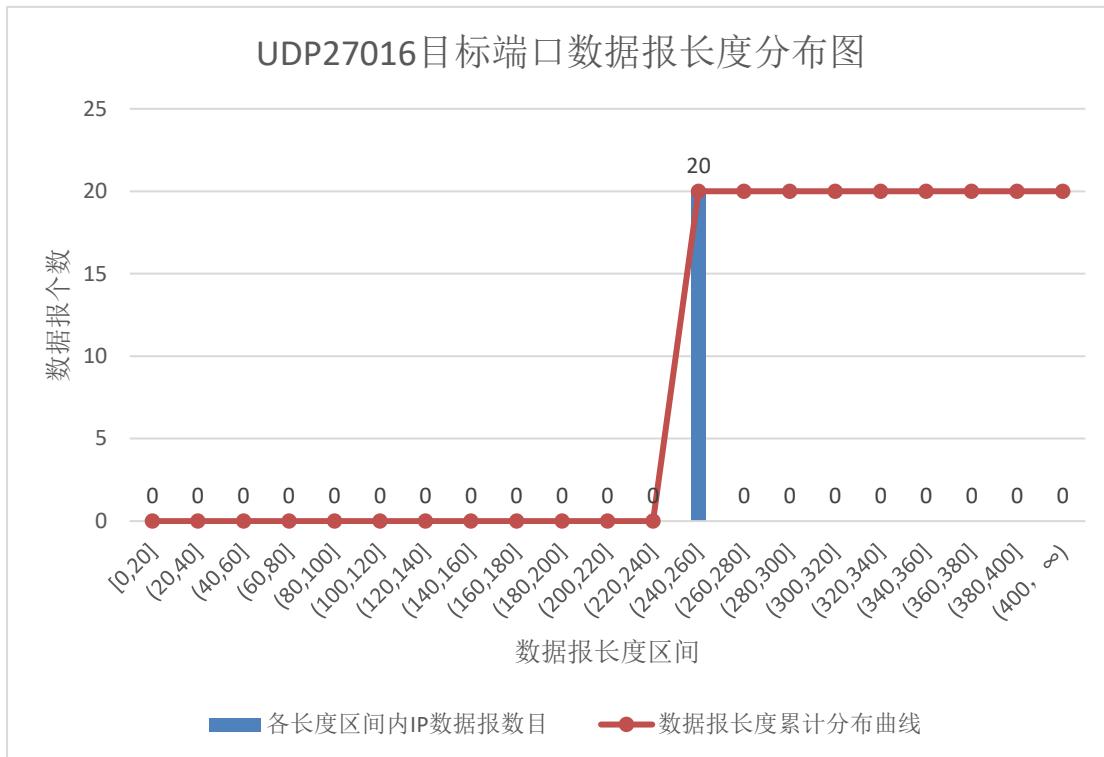
**[4] UDP 目标端口数前 10 的数据报长度分布情况:**











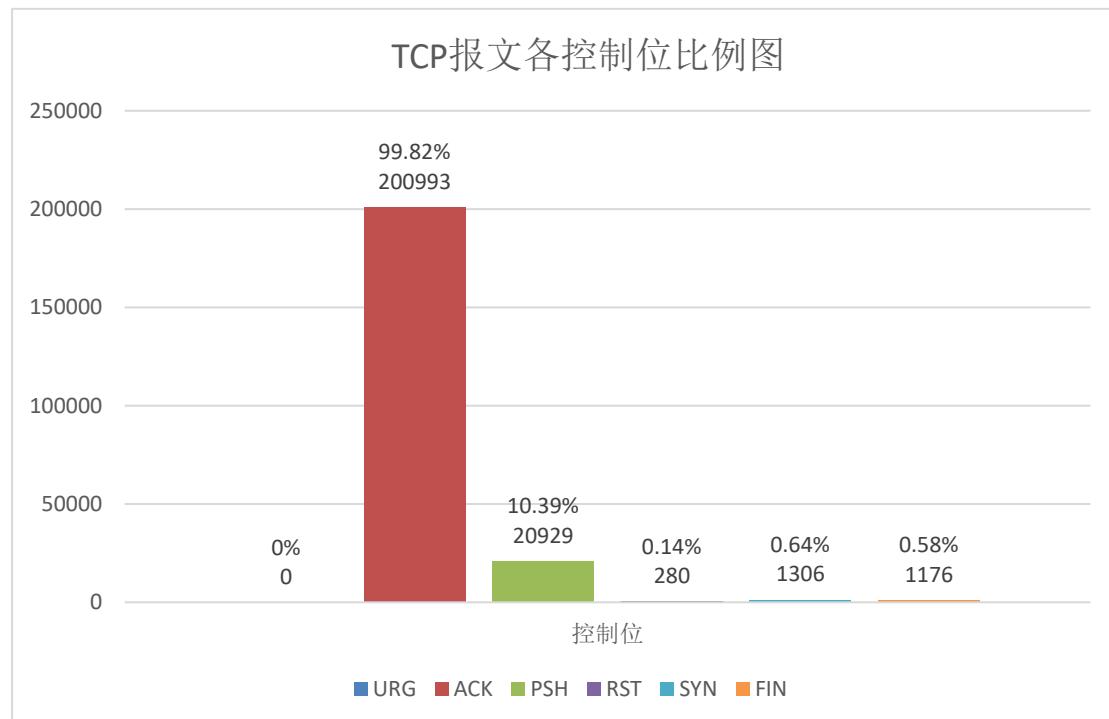
从曲线可以看出，大部分情况特定端口只有某一长度范围的区间有数据报，其他长度区间为 0，而 TCP 的 80 端口、443 端口等除外，因为这些端口访问频繁，80 端口是我们联网必须的开放端口，而且在 IP 数据报长度超过 400 的区间仍然有大量的数据和分组。

## 问题 5

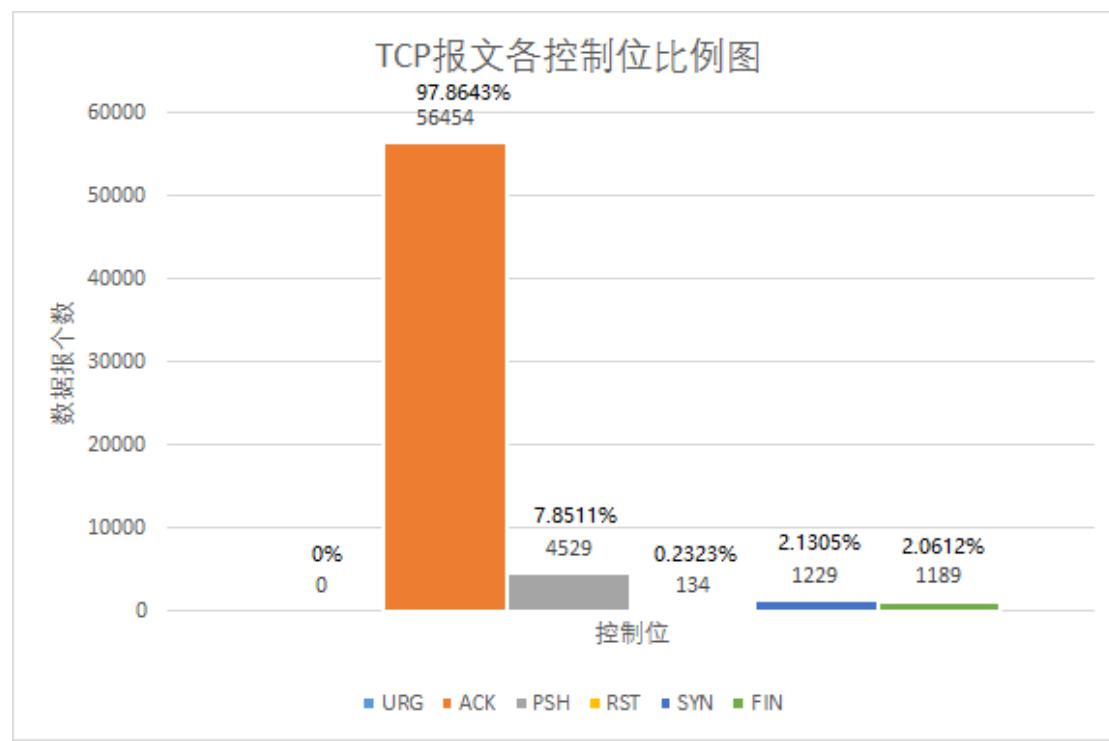
对于载荷为 TCP 的报文，给出其中各个控制位出现的百分比

答：

Input：



Output：



从图中可以看出，`input` 与 `output` 的 TCP 控制位有一定的相似性。

附：TCP 控制位作用如下：

### 6位的标识位：置1表示有效

- URG：和紧急指针配合使用，发送紧急数据；
- ACK：确认号是否有效；
- PSH：指示发送方和接收方将数据不做缓存，立刻发送或接收；
- RST：由于不可恢复的错误重置连接；
- SYN：用于连接建立指示；
- FIN：用于连接释放指示

URG 一般不用。ACK 位大多处于置位状态。第三位在具有交互功能的位中使用。第四位较少，因为 Internet 连接发生错误较少。第五和第六位为 1 的包的数量比为 1:1，因为一次连接一般对应着一次释放。

## 实验总结

通过本实验我们逐渐熟悉了 `windump` 的使用方法，对网络 `traffic` 中 `ip` 分组分析有了更加深刻的理解认识。这次实验中，大家集思广益，合作完成了数据的收集、分析、处理、绘图等一系列环节，既培养了我们动手实践的能力，也提高了我们分析数据、处理绘图的技巧，加强了合作学习的意识，让我们对网络协议分布有了更好的了解。最后，非常感谢老师和助教的耐心指导。

## 文件清单

原始抓包数据	<code>result.txt</code>
<code>Input</code> 端解析数据	<code>input_result.txt</code>
<code>Output</code> 端解析数据	<code>output_result.txt</code>
数据报解析代码	<code>./code</code>
实验报告	计网大作业书面报告.pdf