

# 计算机网络 课程实验报告

实验名称	可靠数据传输协议-GBN 协议的设计与实现					
姓名	王艺丹		院系	计算机		
班级	1903601		学号	1190201303		
任课教师	李全龙		指导教师	李全龙		
实验地点	G207		实验时间	2021.11.06		
实验课表现	出勤、表现得分(10)		实验报告		实验总分	
	操作结果得分(50)		得分(40)		入掘心力	
教师评语						
l						

### 实验目的:

本次实验的主要目的:

- 1. 理解可靠数据传输的基本原理;
- 2. 掌握停等协议的工作原理;
- 3. 掌握基 于 UDP 设计并实现一个停等协议的过程与技术;
- 4. 理解滑动窗口协议的基本原理;
- 5. 掌握 GBN 的工作原理;
- 6. 掌握基于 UDP 设计并实现一个 GBN 协议的过程与技术。

### 实验内容:

本次实验的主要内容:

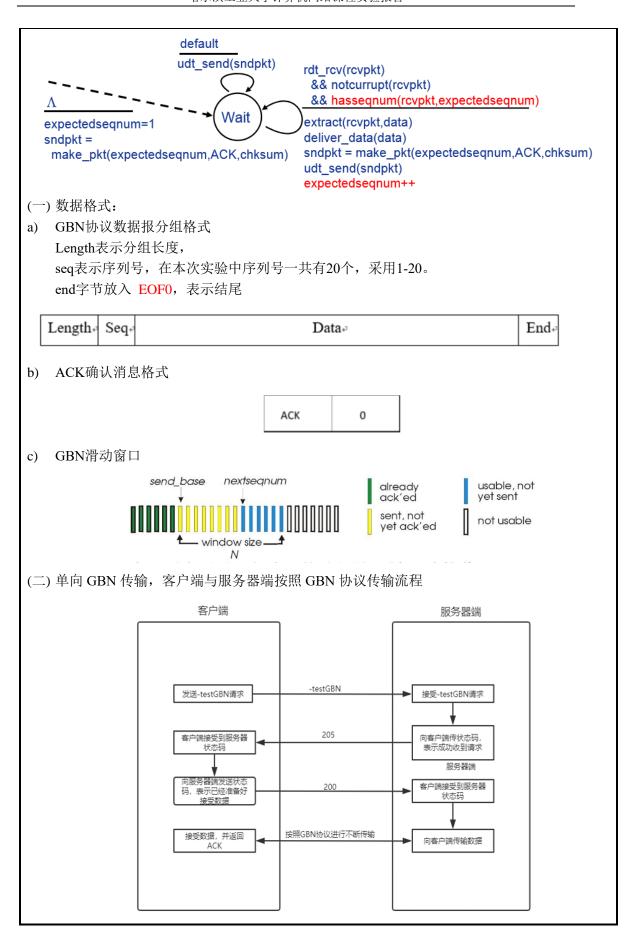
- 1. 基于UDP设计一个简单的GBN协议,实现单向可靠数据传输(服务器到客户的数据传输);
- 2. 模拟引入数据包的丢失,验证所设计协议的有效性;
- 3. 改进所设计的 GBN 协议,支持双向数据传输;
- 4. 将所设计的 GBN 协议改进为 SR 协议。

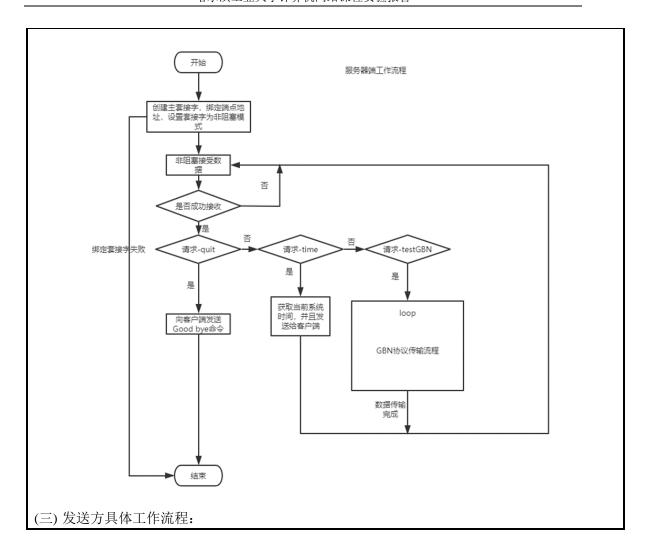
### 实验过程:

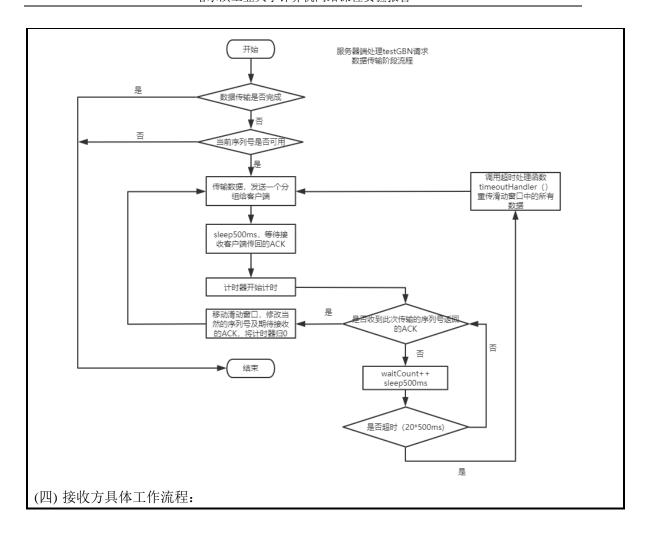
一、实现GBN协议,进行单向可靠数据传输 GBN协议的CS架构:

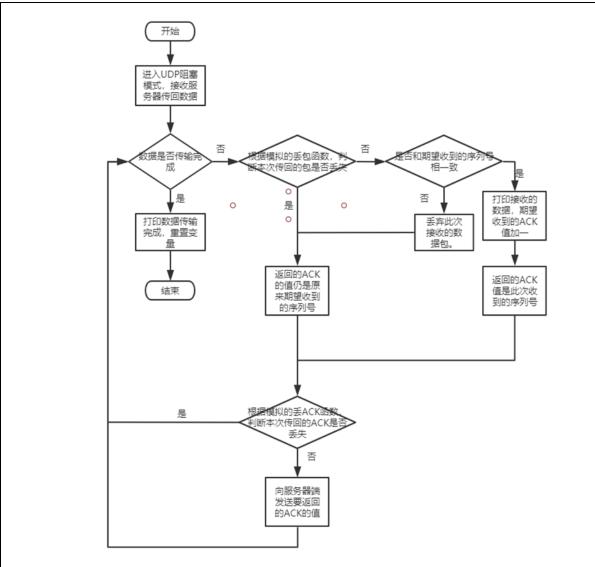
服务器端(发送方):

```
rdt_send(data)
                              if (nextseqnum < base+N) {
                                sndpkt[nextseqnum] = make_pkt(nextseqnum,data,chksum)
                                udt_send(sndpkt[nextseqnum])
                                if (base == nextseqnum)
                                  start_timer
                                nextseqnum++
                              else
                               refuse data(data)
          base=1
         nextseqnum=1
                                                 timeout
                                                start timer
                                   Wait
                                                udt_send(sndpkt[base])
                                                udt_send(sndpkt[base+1])
       rdt_rcv(rcvpkt)
        && corrupt(rcvpkt)
                                                udt_send(sndpkt[nextseqnum-1])
                                rdt_rcv(rcvpkt) &&
                                 notcorrupt(rcvpkt)
                                base = getacknum(rcvpkt)+1
                                If (base == nextseqnum)
                                  stop_timer
                                 else
                                  start timer
客户端(接收方):
```





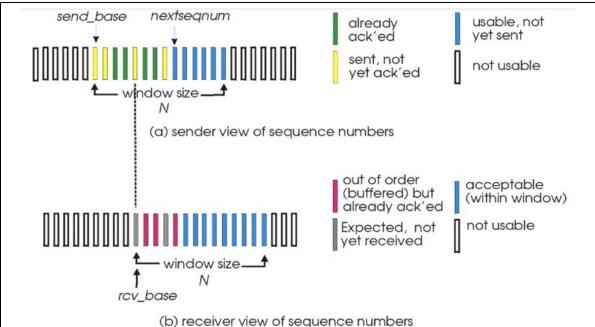




### 二、GBN双向数据传输

对于数据的双向传输,只需要在服务器端增加接受方的功能,在客户端增加发送方的功能即可。对于命令行参数,增设了一个命令参数"-testgbn\_Send",表示从客户端往服务器端发送数据。具体流程和单向传输相似,这里就不做赘述。

- 三、实现 SR 单向传输协议
- (一) 发送方接收方窗口:



### (二) 发送方具体工作流程

对于发送方,也就是服务器端,如果收到 ACK,加入该帧序号在窗口内,则 SR 发送方将此 个被确认的帧标记为已接收。 若该帧序号是窗口的下界,则窗口向前移动到未确认帧中序号最小 处;若窗口移动了,且有仍在窗口内的未发送帧,则发送这些帧。

修改服务器端代码:为每个分组设置计时器,只重传那些未到达的分组。修改处理超时重传 的函数 timeoutHandler(),不需要再重传所有已发送未被接受的分组,在接收到 ACK 时,只确定 当前分组。

### (三)接收方具体工作流程

SR协议在GBN的基础上增设了缓冲区,可以处理乱序到达的数据包,如果序列号范围在可 接受范围内,发送此次接收到的数据包的ACK,之后将该数据包缓存,重复循环,直至它之前连 续所有数据包都已成功被接受,就将缓存的数据全部交付给上层,而后窗口向前滑动。

修改客户端代码:增加了数据缓存区dataBuffer[SEQ SIZE][BUFFER LENGTH],接受窗口 大小设置为RECEIVE WIND SIZE, (需要满足Ns+Nr<=2^k)

### 补充说明模拟数据包/ACK随机丢失的方法:

使用lossInLossRatio函数:

实现原理如下:设lossRatio表示丢包率,随机产生一个数N,求取N%101,得到余数Y(0=< Y <= 100), 判断Y的取值范围。若Y <= int(lossRatio\*100), 返回True (表示丢包); 否则返回False (表示未丢包)。默认设置数据包与ACK的丢包率均为0.2。

实验结果:

## 哈尔滨工业大学计算机网络课程实验报告 一、GBN的单向数据传输 在客户端输入-testGBN进行测试 G:\consnet\wyd-lab2\GBN\GBN 1\GBN Client\Debug\G The Winsock 2.2 dll was found okay \*\*\*\*\*\*\*\*\*\*\*\*\*\* time to get current time -quit to exit client | -testGBN to test the gbn \*\*\*\*\*\*\*\*\*\*\*\*\*\* testGBN 本次数据传输未发生丢包现象: 客户端: Ready for file transmission recv a packet with a seq of 1 send a ack of 1 recv a packet with a seq of 2 send a ack of 2 recv a packet with a seq of 3 send a ack of 3 recv a packet with a seq of 4

send a ack of 4 数据传输全部完成!!!

### 服务器端:

再次测试: 客户端:

```
testGBN
Begin to test GBN protocol, please don't abort the process
The loss ratio of packet is 0.20,the loss ratio of ack is 0.20
Ready for file transmission
recv a packet with a seq of 1
The ack of 1 loss
The packet with a seq of 2 loss
recv a packet with a seq of 3 send a ack of 1
The packet with a seq of 4 loss
recv a packet with a seq of 2
send a ack of 2
recv a packet with a seq of 3
send a ack of 3
The packet with a seq of 4 loss
send a ack of 3
The packet with a seq of 4 loss
The packet with a seq of 4 loss
recv a packet with a seq of 4
send a ack of 4
数据传输全部完成!!!
服务器端:
```

```
ecv from client: -testGBN
Begain to test GBN protocol, please don't abort the process
Shake hands stage
Begin a file transfer
File size is 4096B, each packet is 1024B and packet total num is 4 send a packet with a seq of : 1 totalSeq now is : 1 send a packet with a seq of : 2 totalSeq now is : 2
                                                      totalSeq now is : 3
send a packet with a seq of : 3
Recv a ack of seq 1
                     curAck <= index , totalAck += 1
----totalAck Now is : 1
send a packet with a seq of 2 totalSeq Now is : 1
Recv a ack of seq 2
send a packet with a seq of : 4
                                                      totalSeq now is: 4
                                                      totalSeq now is : 2
                     curAck <= index , totalAck += 1 ----totalAck Now is : 2
send a packet with a seq of : 3
                                                       totalSeq now is: 3
Recv a ack of seq 3
                   curAck <= index , totalAck += 1
                      ----totalAck Now is : 3
send a packet with a seq of : 4 totalSe
limer out error. ---totalSeq Now is : 3
send a packet with a seq of : 4 totalSe
limer out error. ----totalSeq Now is : 3
                                                       totalSeq now is: 4
                                                      totalSeq now is: 4
send a packet with a seq of : 4
                                                      totalSeq now is : 4
Recv a ack of seq 4
                    curAck <= index , totalAck += 1
                          totalAck Now is : 4
数据传输全部完成!!!
```

可以看到,首先服务器收到客户端的请求,三次握手建立连接,开始发送数据:序列号为1的数据包的ACK丢失了,数据包2和4丢失了,接收到了数据包3,由于GBN协议不缓存乱序到达的数据包,直接丢弃,超时,接收方重发ACK 1,此时收到了数据包2,进行缓存,发送ACK2,……,发送数据时数据包4丢失了,超时,重发数据包4,丢失再次重发,直到收到ACK 4,由于GBN采用累计确认机制,此时ACK=TotalSeqNum,意味着完成全部数据传输,解除连接。

二、GBN双向数据传输

通过-testgbn Send测试

### C:\WINDOWS\system32\cmd.exe

客户端:

```
send a packet with a seq of: 1 totalSeq now is: 1 send a packet with a seq of: 2 totalSeq now is: 2 send a packet with a seq of: 3 totalSeq now is: 3 send a packet with a seq of: 4 totalSeq now is: 3 send a packet with a seq of: 4 totalSeq now is: 4 gev a ack of seq 1 curAck <= index , totalAck += 1 ----totalAck Now is: 1 Recv a ack of seq 2 curAck <= index , totalAck += 1 ----totalAck Now is: 2 Timer out error. ----totalSeq Now is: 2 send a packet with a seq of: 3 totalSeq now is: 3 Recv a ack of seq 3 curAck <= index , totalAck += 1 ----totalAck Now is: 3 send a packet with a seq of: 4 totalSeq now is: 4 Timer out error. ----totalAck Now is: 3 send a packet with a seq of: 4 totalSeq now is: 4 Recv a ack of seq 4 curAck <= index , totalAck += 1 -----totalAck Now is: 3 send a packet with a seq of: 4 totalSeq now is: 4 Recv a ack of seq 4 curAck <= index , totalAck += 1 -----totalAck Now is: 4 w据传输全部完成!!
          -time to get current time |
-quit to exit client |
-testgbn [X] [Y] to test the gbn (Receive message from the Server) |
-testgbn Send [X] [Y] to test the gbn Send (Send message to the Server) |
-testgbn Send [X] [Y] to test the gbn Send (Send message to the Server) |
服务器端:
          G:\consnet\wyd-lab2\GBN\GBN 2\GBN Client\Debug\GBN Server.exe
      send a ack of 1
      recv a packet with a seq of 2
      send a ack of 2
       recv a packet with a seq of 3
       The ack of 3 loss
       The packet with a seq of 4 loss
      recv a packet with a seq of 3 send a ack of 3
       recv a packet with a seq of 4
      The ack of 4 loss
recv a packet with a seq of 4
      send a ack of 4
数据传输全部完成!!!
不做赘述。
三、SR
```

### C:\WINDOWS\system32\cmd.exe

### 客户端:

### 服务器端:

G:\consnet\wvd-lab2\GBN\SR\GBN Client\Debug\GBN Server.exe

```
The Winsock 2.2 dll was found okay
totalPacket is : 4
recv from client: -testSR
Begain to test SR protocol, please don't abort the process
Shake hands stage
Begin a file transfer
File size is 4096	ext{B}, each packet is 1024	ext{B} and packet total num is 4
                                                 totalSeq now is: 1
send a packet with a seq of : 1
Recv a ack of seq 1
                  curAck <= index , totalAck += 1
----totalAck Now is : 1
                                                 totalSeq now is : 2 totalSeq now is : 3
send a packet with a seq of : 2 •
send a packet with a seq of : 3
Recv a ack of seq 3
                  curAck <= index , totalAck += 1
----totalAck Now is : 2
send a packet with a seq of : 4 totals
                                                 totalSeq now is: 4
send a packet with a seq of : 2
                                                 totalSeq now is : 2
Recv a ack of seq 2
                  curAck <= index , totalAck += 1

----totalAck Now is : 3

----totalZSeq Now is : -8
Timer out error.
                                                 totalSeq now is : -7
send a packet with a seq of : 4
Recv a ack of seq 4
                  curAck <= index , totalAck += 1
----totalAck Now is : 4
数据传输全部完成!!!
```

可以看到,ACK 2与ACK 4 丢失了,但由于SR对乱序到达的数据包进行缓存并逐个确认,客户端处超时仅重传未收到ACK的数据包2和数据包4,最后完成全部数据传输。

### 问题讨论:

- ▶ 最初写实验时,当发送数据到达窗口边界时,有时会出现一些问题
- ✓ 利用数据包数量与ACK序列号,增加判断数据传输是否完成的语句。

### 心得体会:

- ◆ 更加深入理解了滑动窗口协议,对GBN协议和SR协议有了更深的理解;
- ♦ 对win Socket编程进一步熟悉了原理。