

PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE
ACADEMIC YEAR: 2022-23
DEPARTMENT of COMPUTER ENGINEERING DEPARTMENT
CLASS: T.E. SEMESTER: I
SUBJECT: CNSL

ASSINGMENT NO.	A4
TITLE	Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in Peer-to-Peer mode.
PROBLEM STATEMENT /DEFINITION	Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in peer to peer mode.
OBJECTIVE	To demonstrate Go back N and Selective Repeat Modes of Sliding Window Protocol in peer to peer mode .
OUTCOME	Demonstrate Go back N and Selective Repeat Modes and also captured packets using Wireshark Packet Analyzer Tool for peer to peer mode.
S/W PACKAGES AND HARDWARE APPARATUS USED	PC with the configuration as Pentium IV 1.7 GHz. 128M.B RAM, 40 G.B HDD, 15”Color Monitor, Keyboard, Mouse. Operating Systems (64-Bit)64-BIT Fedora 20 or latest 64-BIT Update of Equivalent Open source OS Programming Tools (64-Bit) GCC/G++, packet tracer tool
REFERENCES	Mansfield K., Antonakos J., "An Introduction to Computer Networking", Perason Education, 2002, ISBN 81 - 7808 - 828 – 2 Fourauzan B., "Data Communications and Networking", 5 th Edition, Tata McGraw- Hill, Publications, ISBN: 0 – 07 – 058408 – 7
STEPS	Refer to student activity flowchart
INSTRUCTIONS FOR WRITING JOURNAL	<ol style="list-style-type: none"> 1. Date 2. Assignment no. 3. Problem definition 4. Learning objective 5. Learning Outcome 6. Concepts related Theory 7. Algorithm 8. Test cases 10. Conclusion/Analysis

Prerequisites: frames, data link layer, flow control, client server architecture

Concepts related theory:

The basic idea of sliding window protocol is that both sender and receiver keep a

“window” of acknowledgment. The sender keeps the value of expected acknowledgment; while the receiver keeps the value of expected receiving frame. When it receives an acknowledgment from the receiver, the sender advances the window. When it receives the expected frame, the receiver advances the window.

In transmit flow control, sliding window is a variable-duration window that allows a sender to transmit a specified number of data units before an acknowledgement is received or before a specified event occurs.

Flow Control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver. The flow of data should not be allowed to overwhelm the receiver. Receiver should also be able to inform the transmitter before its limits (this limit may be amount of memory used to store the incoming data or the processing power at the receiver end) are reached and the sender must send fewer frames. Hence, Flow control refers to the set of procedures used to restrict the amount of data the transmitter can send before waiting for acknowledgment.

There are two methods developed for flow control namely Stop-and-wait and Sliding-window Sliding window algorithms, used by TCP, permit multiple data packets to be in simultaneous transit, making more efficient use of network bandwidth.

Sliding Window Protocol:

With the use of multiple frames for a single message, the stop-and-wait protocol does not perform well. Only one frame at a time can be in transit. Efficiency can be greatly improved by allowing multiple frames to be in transit at the same time. Efficiency can also be improved by making use of the full-duplex line. To keep track of the frames, sender station sends sequentially numbered frames. Since the sequence number to be used occupies a field in the frame, it should be of limited size. If the header of the frame allows k bits, the sequence numbers range from 0 to $2^k - 1$. Sender maintains a list of sequence numbers that it is allowed to send (sender window).

The size of the sender's window is at most $2^k - 1$. The sender is provided with a buffer equal to the window size. Receiver also maintains a window of size $2^k - 1$. The receiver acknowledges a frame by sending an ACK frame that includes the sequence number of the next frame expected. This also explicitly announces that it is prepared to receive the next N frames, beginning with the number specified. This scheme can be used to acknowledge multiple frames. It could receive frames 2, 3, 4 but withhold ACK until frame 4 has arrived. By returning an ACK with sequence number 5, it

acknowledges frames 2, 3, 4 in one go. The receiver needs a buffer of size 1.

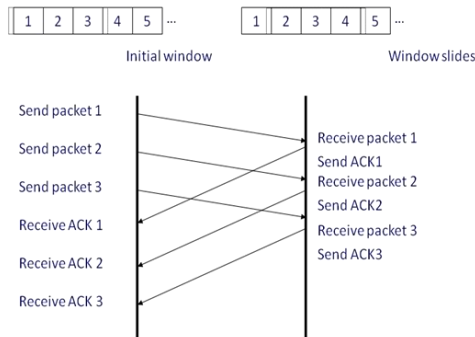
Sliding window algorithm is a method of flow control for network data transfers. TCP, the Internet's stream transfer protocol, uses a sliding window algorithm.

A sliding window algorithm places a buffer between the application program and the network data flow. For TCP, the buffer is typically in the operating system kernel, but this is more of an implementation detail than a hard-and-fast requirement.

Data received from the network is stored in the buffer, from where the application can read at its own pace. As the application reads data, buffer space is freed up to accept more input from the network. The window is the amount of data that can be "read ahead" - the size of the buffer, less the amount of valid data stored in it. Window announcements are used to inform the remote host of the current window size.

An example of a sliding window in packet transmission is one in which, after the sender fails to receive an acknowledgement for the first transmitted packet, the sender "slides" the window, i.e. resets the window, and sends a second packet. This process is repeated for the specified number of times before the sender interrupts transmission. Sliding window is sometimes (loosely) called *acknowledgement delay period*.

Idea Behind Sliding Window

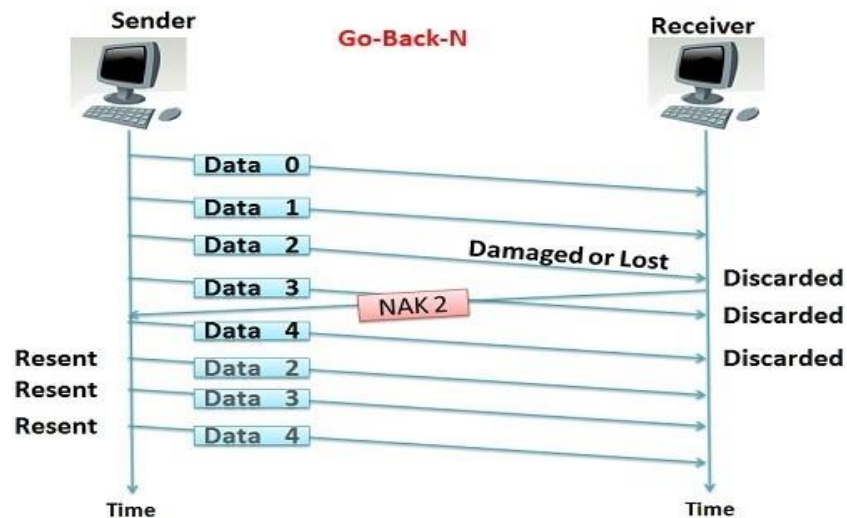


118

Go-Back-N Protocol and "Selective Repeat Protocol" are the sliding window protocols. The sliding window protocol is primarily an error control protocol, i.e. it is a method of error detection and error correction. The basic difference between go-back-n protocol and selective repeat protocol is that the "go-back-n protocol" retransmits all the frames that lie after the frame which is damaged or lost. The "selective repeat protocol" retransmits only that frame which is damaged or lost.

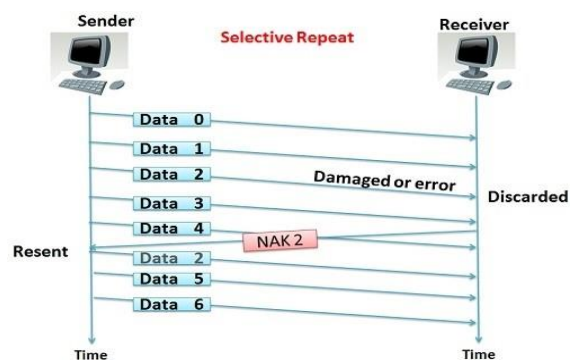
Go back NARQ

In the Go-Back-N Protocol, the sequence numbers are modulo 2^m , where m is the size of the sequence number field in bits.



Selective Repeat ARQ

Go-Back-N ARQ simplifies the process at the receiver site. The receiver keeps track of only one variable, and there is no need to buffer out-of-order frames; they are simply discarded. However, this protocol is very inefficient for a noisy link. In a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission. For noisy links, there is another mechanism that does not resend N frames when just one frame is damaged; only the damaged frame is resent. This mechanism is called Selective Repeat ARQ.



Key Differences Between Go-Back-N and Selective Repeat

1. Go-Back-N protocol is design to retransmit all the frames that are arrived after the damaged or a lost frame. On the other hand, Selective Repeat protocol retransmits only that frame that is damaged or lost.
2. If the error rate is high i.e. more frames are being damaged and then retransmitting all the frames that arrived after a damaged frame waste the lots of bandwidth. On the other hand, selective repeat protocol re-transmits only damaged frame hence, minimum bandwidth is wasted.
3. All the frames after the damaged frame are discarded and the retransmitted frames arrive in a sequence from a damaged frame onwards, so, there is less headache of sorting the frames hence it is less complex. On the other hand only damaged or suspected frame is retransmitted so, extra logic has to be applied for sorting hence, it is more complicated.
4. Go-Back-N has a window size of $N-1$ and selective repeat have a window size $\leq (N+1)/2$.
5. Neither sender nor receiver need the sorting algorithm in Go-Back-N whereas, receiver must be able to sort the as it has to maintain the sequence.
6. In Go-Back-N receiver discards all the frames after the damaged frame hence, it don't need to store any frames. Selective repeat protocol does not discard the frames arrived after the damaged frame instead it stores those frames till the damaged frame arrives successfully and is sorted in a proper sequence.
7. In selective repeat NAK frame refers to the damaged frame number and in Go-Back-N, NAK frame refers to the next frame expected.
8. Generally the Go-Back-N is more is use due to its less complex nature instead of Selective Repeat protocol.

Code Implementation

-----AT CLIENT SIDE-----

```
import java.lang.System;
```

```
import java.net.*;
```

```
import java.io.*;
```

```
public class Client {
```

static Socket connection;

```
public static void main(String a[]) throws SocketException {  
    try {  
        int v[] = new int[9];  
        //int g[] = new int[8];  
        int n = 0;  
        InetAddress addr = InetAddress.getByName("Localhost");  
        System.out.println(addr);  
        connection = new Socket(addr, 8011);  
        DataOutputStream out = new DataOutputStream(  
            connection.getOutputStream());  
        DataInputStream in = new DataInputStream(  
            connection.getInputStream());  
        int p = in.read();  
        System.out.println("No of frame is:" + p);  
  
        for (int i = 0; i < p; i++) {  
            v[i] = in.read();  
            System.out.println(v[i]);  
            //g[i] = v[i];  
        }  
        v[5] = -1;  
        for (int i = 0; i < p; i++)  
        {  
            System.out.println("Received frame is: " + v[i]);
```

```

    }

    for (int i = 0; i < p; i++)

        if (v[i] == -1) {

            System.out.println("Request to retransmit packet no "

                               + (i+1) + " again!!");

            n = i;

            out.write(n);

            out.flush();

        }


        System.out.println();


        v[n] = in.read();

        System.out.println("Received frame is: " + v[n]);


        System.out.println("quiting");

    } catch (Exception e) {

        System.out.println(e);

    }

}

}

```

*** OUTPUT**

iotlab@iotlab-Veriton-M200-B360:~/sliding window\$ javac Client.java

iotlab@iotlab-Veriton-M200-B360:~/sliding window\$ java Client

Localhost/127.0.0.1

No of frame is:9

30

40

50

60

70

80

90

100

110

Received frame is: 30

Received frame is: 40

Received frame is: 50

Received frame is: 60

Received frame is: 70

Received frame is: -1

Received frame is: 90

Received frame is: 100

Received frame is: 110

Request to retransmit packet no 6 again!!

Received frame is: 80

quiting

P:F-LTL-UG/03/R1

/-----AT SERVER SIDE-----

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;
import java.net.SocketException;

public class Server {
    static ServerSocket Serversocket;
    static DataInputStream dis;
    static DataOutputStream dos;

    public static void main(String[] args) throws SocketException {

        try {
            int a[] = { 30, 40, 50, 60, 70, 80, 90, 100, 110 };

            Serversocket = new ServerSocket(8011);

            System.out.println("waiting for connection");

            Socket client = Serversocket.accept();

            dis = new DataInputStream(client.getInputStream());
            dos = new DataOutputStream(client.getOutputStream());

            System.out.println("The number of packets sent is:" + a.length);

            int y = a.length;

            dos.write(y);
```

```

dos.flush();

for (int i = 0; i < a.length; i++) {
    dos.write(a[i]);
    dos.flush();
}

int k = dis.read();

dos.write(a[k]);
dos.flush();

} catch (IOException e) {
    System.out.println(e);
} finally {
    try {
        dis.close();
        dos.close();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

}

}

}

```

/*OUTPUT

waiting for connection

The number of packets sent is:9

Conclusion: Hence we have implemented of sliding window protocol(Go back N and Selective Repeat).

Review questions

1. What is flow control?
2. Which type of frame sliding window protocol used?
3. What is the sender and receiver window size of Go Back N and selective repeat protocol