CS 455  HW 4  Solutions
(2018)
Question 4

- If you look at step-and-wait carefully, you will see that:

\[
\text{Step-and-wait} = \text{Go-back-}N \quad \text{using:}
\]

- Send window size = 1
- FPCV window size = 1

We have proved that for Go-Back-\(N\) that:

\[
W_{\text{send}} + W_{\text{recv}} \leq N \quad N \leq H \text{ Seq. numbers}
\]

The protocol will guarantee reliable communication.

Thus here:

\[
W_{\text{send}} + W_{\text{recv}} = 1 + 1 = 2.
\]

If we choose 2 Seq. Numbers in Step-and-wait, Step-and-wait will guarantee reliable communication.
Question 2:

Go-back-N: Recv. Window size = 1.

Given: Send-Window size = 4.

Initial state:

\( Sw = \{3, 0, 1, 2, 3\} \)

Sender

Receiver

\( Rw = \{3, 3\} \)

Exchange:

\( 3, 0, 1, 2, 3 \)

1. Sender 13

Receiver sends

ACK

New state:

\( Sw = \{0, 1, 2, 3\} \)

\( Rw = \{0, 3\} \)
Frames received in error are ignored
(= exactly like lost!!!)

\( \Rightarrow \) Receiver does nothing!
Question 3

8000 bits = $\frac{8000}{1M}$ sec = 0.008 sec

Earth

100 frames = 100 x 0.008 sec = 0.8 sec

Moon

1 sec.

When there are no errors, sender and receiver in sliding window will exchange frames/ACKs as follows:

Send

Earth: (window size = 100)

100 frames

Another 100 frame

1 sec

ACK

ACK

ACK

0.008 sec

2.0008 sec
If you look at the sender:

\[ \text{transmit data frame for 0.8 sec} \]
\[ \text{Then idle (wait for ACKs) until time 2.00 sec} \]
\[ \text{Then repeat the same cycle.} \]

\[ \text{Utilization (fraction of time used in transmission)} \]

\[ \text{Utilization} = \frac{0.8}{2.00 \text{ sec}} = 39.04\% \]
(B) In order for the sender to transmit continuously, the ACK for the first frame must return in time:

(Note: the ACK for the 2nd frame will be on time (assuming no error) if the first ACK is on time!!!)

The sender must transmit this number of frames:

\[ \frac{2.008 \text{ sec}}{0.008 \text{ sec}} = 251 \]

\# frames = 251

\# bits = 8
Given: \( W_{send} = 5 \)  
\( W_{recv} = 5 \)

Initially: \( SW = 0, 1, 2, 3, 4, 3 \)
\( RW = 0, 1, 2, 3, 4, 3 \).

(1) \( SW = 0, 1, 2, 3, 4, 3 \)
\[ \begin{array}{cccc}
10 & 0 & 1 & 2 \\
& & & 3
\end{array} \]

\( RW = 0, 1, 2, 3, 4, 3 \)
\[ \begin{array}{c}
\uparrow \\
3
\end{array} \]
\( \text{receives this frame} \)

\( \text{expects 0.} \)

\( \text{Receive will } \Rightarrow \) (1) buffer [3] (because it's in the RW range)

(2) send ACK 7 (0 - 1 = 7) in 3 bits

Answer:
\( \text{sends \ ACK 7} \)
(2) Fact: because receiver did not send an ACK that allow the sender to shift its send window forward, we know:

$$Sw = \{0, 1, 2, 3, 4\} \text{ (Unchanged)}$$

Sw:

$$Sw = \{0, 1, 2, 3, 4\} \quad \text{[\varnothing \ 1 \ 2 \ \ldots]}$$

RW:

$$RW = \{0, 1, 2, 3, 4\} \quad \text{[\varnothing \ 1 \ 2]}$$

3 is buffered.

Recent will:

1. Buffer [\varnothing \ 1 \ 2]
2. Deliver [\varnothing], shift RW

$$RW = \{1, 2, 3, 4, 5\}$$

3. Send ACK \varnothing

Answer: sends ACK \varnothing

New state is used to answer the next question!!!
(3) Fact: Because receiver sent: ACK 0

Sender's window can be:

\[ SW = \{0, 1, 2, 3, 4, 5\} \Rightarrow \text{ACK 0 lost} \]

or \[ SW = \{1, 2, 3, 4, 5\} \Rightarrow \text{ACK 0 received} \]

In both cases: \#1 denotes an "old" frame.

SW:

\[ RW = \{1, 2, 3, 4, 5\} \Rightarrow \text{This must be a retransmission.} \]

\[ \uparrow \uparrow \]

\[ 2, 3 \text{ are buffered.} \]

Receiver will:

1. Buffer \[ \text{[1]} \]
2. Deliver: \[ \text{[1]} [2] [3] \]
3. Shift: \[ RW = \{4, 5, 6, 7, 0\} \]
4. Send ACK 3

Answer: send ACK 3
(4) Fact: Because receiver sent:

\[ \text{ACK } 0 \quad \text{(part 2)} \]

and \( \text{ACK } 3 \quad \text{(part 3)} \)

The sender window can be one of:

\[ \text{SW} = \{0, 1, 2, 3, 4, 3\} \quad \text{or } \text{ACK } 0, \text{ACK } 3 \text{ lost} \]

or:

\[ \text{SW} = \{1, 2, 3, 4, 5, 3\} \quad \text{or } \text{ACK } 0, \text{recv. ACK } 3 \text{ lost} \]

or:

\[ \text{SW} = \{4, 5, 6, 7, 0\} \quad \text{or } \text{ACK } 3 \text{ recv'd.} \]

Therefore:

sender can send an old frame \( \varnothing \)

\( \text{SW} = \{0, 1, 2, 3, 4, 3\} \)  \( \Rightarrow \) old frame \( \varnothing \)

or a new frame \( \varnothing \)

\( \text{SW} = \{4, 5, 6, 7, 0\} \)  \( \Rightarrow \) new frame.

Reciever action when \( \varnothing \) is received:

\[ \text{RW} = \{4, 5, 6, 7, 0\} \]

- \( \varnothing \) in \( \text{RW} \) window.
- It is a new frame.
- Send \( \text{ACK } 3 \)
- Send \( \text{ACK } 3 \)
(5) The sender window can be one of:

- SW: \(50, 51, 52, 53, 54\) & ACK3 lost
- W: SW: \(51, 52, 53, 54\) & ACK3 rec'd, ACK3 lost
- W: SW: \(54, 55, 6, 7, 0\) & ACK3 rec'd.

The seq. #1 has NOT wrapped around!!!

\[\rightarrow\] Su frame? must be an old frame!!!

Receiver action when \(\square\) is received:

- RW: \(\{4, 5, 6, 7, 0\}\).

\(\square\) is NOT in RW.

\[\rightarrow\] discard \(\square\) (drop & buffer)

\[\rightarrow\] send ACK3
**Question 5**

**Given:**
- $W_{send} = 4$
- $W_{recv} = 4$
- 3 bit seq #: 0, 1, 2, ..., 7

**Initially:**
- $SW = [0, 1, 2, 3]$
- $RW = [0, 1, 2, 3]$

(1) $SW = [0, 1, 2, 3]$

Receiver will:
1. buffer $\boxed{3}$
2. send Ack 7

**Answer:** Sends Ack 7
(2) Next:

\[ SW = \{0, 1, 2, 3\} \]

\[ RW = \{0, 1, 2, 3\} \]

Receiver will:

1. buffer \( \emptyset \) and \[ 2 \]
2. deliver \( \emptyset \) !!!
3. Send ACK \( \emptyset \)
   (to acknowledge \( \emptyset \)).

Answer:

Sends ACK \( \emptyset \)
3) Pre-analysis:

ACK 0 can be received or lost

Sender window can be:

\[ SW = \{0, 1, 2, 3\} \]

or \[ SW = \{1, 2, 3, 4\} \]

Seq. no 4 is used to label an OLD frame in BOTH SW window!!

Therefore:

\[ RW = \{1, 2, 3, 4\} \]

Receive will: 

1) buffer [ ]

2) deliver: [1 2 3]

\[ RW = \{4, 5, 6, 7\} \]

3) send ACK 3

Answer: Send ACK 3
(4) Pre-analysis:

Sender window can be one of these:

- $SW = \{0, 1, 2, 3\}$ & $\phi$, $ACK\phi$, $ACK\ 3$ both lost.
- $SW = \{1, 2, 3, 4\}$ & $ACK\ 3$ lost.
- $SW = \{4, 5, 6, 7\}$ & $ACK\ 3$ received.

$\phi$ will ALWAYS be used for a re-transmission of an old frame.

Receiver action:

- $RW = \{4, 5, 6, 7\}$

Receiver will:
1. Discard $\phi$
2. Send $ACK\ 3$

Answer: send $ACK\ 3$

This frame $\phi$ is ALWAYS a retransmission, cannot be a new frame.
(5) Same pre-analysis:
Sender window can be one of these:

- \( SW = (0, 1, 2, 3, 4) \) & ACK1, ACK3, ACK4 lost
- \( SW = (1, 2, 3, 4, 5) \) & ACK3, ACK4 lost
- \( SW = (4, 5, 6, 7, 3) \) & one of ACK3 received

1 is used when ACK3's were lost

\( \implies \) 1 must be a retransmission

(frame 1 was not acknowledged yet!!!)

Receiver action:

\[ \text{Receiver will:} \]
- Discard 0
- Send ACK3

Answer: send ACK3

This frame 0 is ALWAYS a retransmission
(because sender did not receive any ACK3
(which would ack frame 0).}