Utilization for Stop and Wait Algorithm

$$U = \frac{1}{1 + 2\frac{T_{Prop}}{T_{Frame}}}$$

If you have a really small frame compared to the number of the bits the link can hold, it can result in a very low utilization.

Sliding Window Protocol

Objective: To increase utilization
Allows multiple frames to be sent without being acknowledged.

The transmitter embeds a sequence number into each frame of size n (bits).

The window size would be $w = 2^n - 1$

Satellite Communication Example

$T_{Prop} = 240ms$

$T_{Frame} = 8ms$

How big should the window be?

$240ms / 8ms = 30\, frames$

When the receiver sends an acknowledge frame, it sends back the frame number that expects to get back. That tells the transmitter that the frames up to that number were received successfully.

Remember: transmission is full-duplex. The transmitter and receiver can be accessing the line at the same time.

For the sliding window algorithm, $U = \frac{w}{1 + 2\frac{T_{Prop}}{T_{Frame}}}$

Find the window size that yields the maximum utilization (100%)

$$w = 1 + 2\frac{240ms}{8ms}$$

$$w = 61$$

Problem with large window sizes: In the event of an error in the very first frame, all the frames after it will need to be retransmitted.

Error Control

If a frame has any error in it, the receiver just discards it.

If the transmitter sends f0, f1, f2, f3, f4, f5 and f2 gets broken in transmission, the transmitter does not know this error occurred.

When a receiver gets a frame with the wrong sequence number on it, it sends a special frame back to the transmitter. This frame is called a **reject** frame … a sort of "not-acknowledge" frame. This reject frame will contain the number of the frame that it expected to get.

This is called a **Go-Back-N** algorithm.