PERFORMANCE ISSUES

Here, we examine some of the performance issues related to the use of sliding-window flow-control.
Stop-and-Wait Flow Control
Let us determine the maximum potential efficiency of a half-duplex point-to-point line using stop-and-wait scheme . Suppose that a long message is to be sent as sequence of frames fl, f2, . . . fn, in the following fashion:
Station S1 sends f1.
Station S2 sends an acknowledgment.
Station S1 sends f2.
Station S2 sends an acknowledgment.
.
.
.
.

Station S1 sends fn.
Station S2 sends an acknowledgment.

The total time to send the data. T, can be expressed as T = nTF, where TF is the time send one frame and receive an acknowledgment. We can express TF as follows:

TF = tprop + tframe + tproc + tprop + tack + tproc

where

tprop, = propagation time from S1 to S2
tframe = time to transmit a frame (time for the transmitter to send out all of the bits of the frame)
tproc = processing time at each station to react to an incoming event
tack = time to transmit an acknowledgment

Let us assume that the processing time is relatively negligible, and that the acknowledgment frame is very small compared to a data frame, both of which are reasonable assumptions. Then we can express the total time to send the data as

T= n(2*tprop + tframe)

Of that time, only n X tframe is actually spent transmitting data and the rest is overhead. The utilization, or efficiency, of the line is

U = n * tframe / n(2*tprop + tframe)

= tframe / (2*tprop + tframe)

It is useful to define the parameter a = tprop / tfrarne. Then,

U = 1 / (1+2a) 6.2

This is the maximum possible utilization of the link. Because the frame containsbits, actual utilization is lower. The parameter a is constant if both tprop and tframe are constants, which is typically the case. Fixed length frames are often used for all except frame in a sequence, and the propagation delay is constant for point-to-point links. To get some insight into Equation (6.2), let us derive a different expression

a = Propagation Time / Transmission Time

The propagation time is equal to the distance d of the link divided by the velocity propagation V. For unguided transmission through air or space, V is the speed 3 X 10' dsec. For guided transmission, V is approximately the speed of light for optical and about 0.67 times the speed of light for copper media. The transmission time the length of the frame in bits, L, divided by the data rate R. Therefore,

a = (d / V) / (L / V)
= Rd / VL

Thus, for fixed-length frames and a fixed distance between stations, a is proportional data rate times the length of the medium. A useful way of looking at a is that it the length of the medium in bits (R d / V) compared to the frame length (L).

Sliding-Window Control
For sliding-window flow control, the efficiency of the line depends on both the window size,N, and the value of a. For convenience, let us again normalize frame transmission time to a value of 1; thus, the propagation time is a. Consider two stations A and B . Station A begins to emit a sequence of frames at time to. The leading edge of the first frame reaches station B at to + a. The first frame is entirely absorbed by to + a + 1. Assuming negligible processing time, B can immediately acknowledge the first frame (ACK1). Let us also assume that the acknowledgment frame is so small that transmission time is negligible. Then the ACKl reaches A at to + 2a + 1. To evaluate performance, we need to consider two cases:

Case 1: N ¿= 2a + 1. The acknowledgment for frame 1 reaches A before A has exhausted its window. Thus, A can transmit continuously with no pause, and utilization is 1.0.

Case 2: N ¡ 2a + 1. A exhausts its window at to + N and cannot send additional frames until to + 2a + 1. Thus, line utilization is N time units out of a period of (2a + 1) time units.

Therefore. we can state that

U = 1 if N ¿=2a+1 .... 6.4

= N / (2a+1) if N ¡ 2a+1

Typically, the sequence number is provided for in an n-bit field, and the maximum window size is N = 2" - 1 .

ARQ

We have seen that sliding-window flow control is more efficient than stop-and-wait flow control. We would expect that when error-control functions are added, this would still be true that is, that go-back-N and selective-reject ARQ are more efficient than stop-and-wait ARQ. Let us develop some approximations to determine the degree of improvement to be expected. First, consider stop-and-wait ARQ. With no errors, the maximum utilization is 1/(1 +2a) as shown . We want to account for the possibility that some frames are repeated because of bit errors. To start, note that the utilization U can be defined as

U = Tf / Tt 6.5

Where
Tf = time for transmitter to emit a single frame
Tt = total time that line is engaged in the transmission of a single frame
For error-free operation using stop-and-wait ARQ,

U = Tf / (Tf + 2 Tp)

where Tp is the propagation time. Dividing by Tf and remembering that a = TpITf, we again have Equation (6.2). If errors occur, we must modify Equation (6.5) to

U = Tf / Nr Tt

where Nr is the expected number of transmissions of a frame. Thus, for stop-and-wait ARQ,we have

U = 1 / Nr (1+ 2a)

A simple expression for Nr can be derived by considering the probability P that a single frame is in error. If we assume that ACKs and NAKs are never in error, the probability that it will take exactly k attempts to transmit a frame successfully is p$^(k-1)*(1-p)$.
That is, we have (k - 1) unsuccessful attempts followed by one successful attempt; the probability of this occurring is just the product of the probability of the individual events occurring. Then,

Nr = E[transmissions] = ? (i * Pr[i transmissions])

? (i p$^(i-1)*(1-p)) = 1/(1-p)$
So we have, Stop and wait : U = (1 - p) / (1 + 2a)

For the sliding-window protocol, Equation (6.4) applies for error-free operation. For Selective-reject ARQ, we can use the same reasoning as applied to stop-and-wait ARQ. That is, the error-free equations must be divided by Nr. Again, Nr = 1/(1 - P). So, Selective reject:

U = 1 - p N ¿= 2a+1
= N(1 - p) /(2a+1) N ¡ 2a+1

The same reasoning applies for go-back-N ARQ, but we must be more careful in approximating Nr,. Each error generates a requirement to retransmit K frames rather than just one frame. Thus,

Nr = E [number of transmitted frames to successfully transmit one frame]
= ? f(i) P$^(i-1) * (1-p)$
where f(i) is the total number of frames transmitted if the original frame must be transmitted 'i' times. This can be expressed as

f (i) = 1 + (i- l)K = (I - K) + K i

Substituting yields2

Nr = (1 - k) ? (1 - p) p$^(i-1) + K?i(1-p)p^(i-1) = 1 - K + K/(1-p)$

But K is approximately equal to (2a + 1) for N ¿ (2a + I), and K = N for N ¡ (2a + 1). Thus,

Go Back-N:

U = (1- p) / (1 + 2ap) N ¿= 2a+1

= N(1 - p) / (2a+1)(1 - p + Np) N¡2a+1