3.3.1 Go-Back-N ARQ

Suppose the transmitter currently has frames 3, 4 and 5 outstanding, and the receiver has successsfully received frames numbered up to and including 2. Suppose now frame 3 reaches the receiver. If the frame is received intact, the receiver will send an ACK(4), meaning that it has correctly received frames through number 3 and now is expecting frame 4. But if frame 3 is received in error, the receiver will send a NAK(3) frame, meaning that frame 3 is negatively acknowledged.[5] Under the Go-Back-N protocol, the sender would have to retransmit all previous frames, even though most were received correctly. (This way the receiver does not have to buffer so many frames.)

We will derive u for the case $N > 2\alpha + 1$. Here the transmitter sends continuously, but $u < 1$ because of retransmittals. We will also assume that α is an integer, and break time into slots of length 1. For concreteness, we will again take as our example N = 16, $\alpha = 5$.

This system is a **Markov chain**. (See `http://heather.cs.ucdavis.edu/~matloff/LaTeX/Math/Markov.pdf` for a quick introduction to Markov chains.) Let us say that this system is in state i if there are i frames sent but not yet ACKed, i = 0,1,...,2α. There is no state $2\alpha + 1$, because the $2\alpha + 1$-th frame will be sent out just as one is ACKed, leaving only 2α still-outstanding frames.

It is helpful here if you think of us observing the state a little bit after the integer times, say at times 0.0001, 1.0001, 2.0001, and so on. This way we don't have to worry whether the $2\alpha + 1$-th frame was sent out before or after the ACK arrives; by the time we take our observation, both have occurred, and thus we will never be in state $2\alpha + 1$ at the times we observe the system, which is all that counts.

How do we move from state to state? Well, consider our example N = 16, $\alpha = 5$. At time 0.0001 S has sent nothing, so the state i = 0. At time 1.0001, S has just finished (at time 1.0000) sending out its first frame, so now we are in state 1. At time 2.0001, we reach state 2, and so on, through time 10.0001 and state 10.

At time 11.0001, though, things get more complex. Just as S sends out its 11th frame, it will receive an ACK or NAK in response to the first of the frames it has sent. If it is an ACK, then there now will be only 10 unacknowledged frames on the line—there momentarily had been 11, but one of them was just ACKed, so now it is only 10. In other words, if R received the first frame correctly, then we will now again be in state 10, just as we were at time 10.0001.

On the other hand, if R found the first frame to be in error, then by the rules of Go-Back-N we must now send *all* of our unacknowledged frames again! In other words, if S receives a NAK at time 11, the new state at time 11.0001 will be 0.

Let π_i denote the long-run proportion of time we are in state i, i.e.

$$\pi_i = \lim_{t \to \infty} \frac{N_{it}}{t} \tag{6}$$

where N_{it} is the number of times we are in state i among times 1.0001, 2.0001,..., t.0001. Then from Markov chain theory,

[5] In some versions of this protocol, the receiver might not send an ACK after every frame. For instance, if frame 3 is received correctly, the receiver may wait until it receives frame 4, and then send ACK(5) or NAK(4), each of which would implicitly be an ACK for frame 3. However, here we will assume a response to every frame. For simplicity, we are also ignoring issues such as corrupted ACKs and so on.

$$\pi_i = \sum_k \pi_k p_{ki} \tag{7}$$

where p_{ki} is the probability of going from state k to state i (in one step).

Let us derive the equation first for $\pi_{2\alpha}$. For concreteness, let's again assume that N = 16 and $\alpha = 5$, so we want to find the equation for π_{10}. Thus we need to know $p_{k,10}$ for the k = 0,1,2,...,10. So, the question is, How can we get to state 10 (in one step)? There are only two possibilities:

- we could be in state 10 at one time point n.0001, and then get an ACK at time n+1, in which case we would be in state 10 at time (n+1).0001, or
- we could be in state 9 at time point n.0001, in which case we would automatically be in state 10 at time point (n+1).0001.

In other words,

$$p_{10,10} = 1 - p \tag{8}$$

and

$$p_{9,10} = 1 \tag{9}$$

So, from Equation (7), we have that

$$\pi_{2\alpha} = \pi_{2\alpha}(1-p) + \pi_{2\alpha-1} \times 1 \tag{10}$$

so that

$$\pi_{2\alpha} = \frac{1}{p}\pi_{2\alpha-1} \tag{11}$$

Also, by the same reasoning which led to Equation (9), $p_{j,j+1} = 1$ for j = 0,1,...,$2\alpha - 1$, so

$$\pi_i = \pi_{i-1} \tag{12}$$

for i = 1,2,...,$2\alpha - 1$. Among other things, this means we can rewrite (11) as

$$\pi_{2\alpha} = \frac{1}{p}\pi_0 \tag{13}$$

Note that

$$\pi_0 + \pi_1 + ... + \pi_{2\alpha} = 1 \tag{14}$$