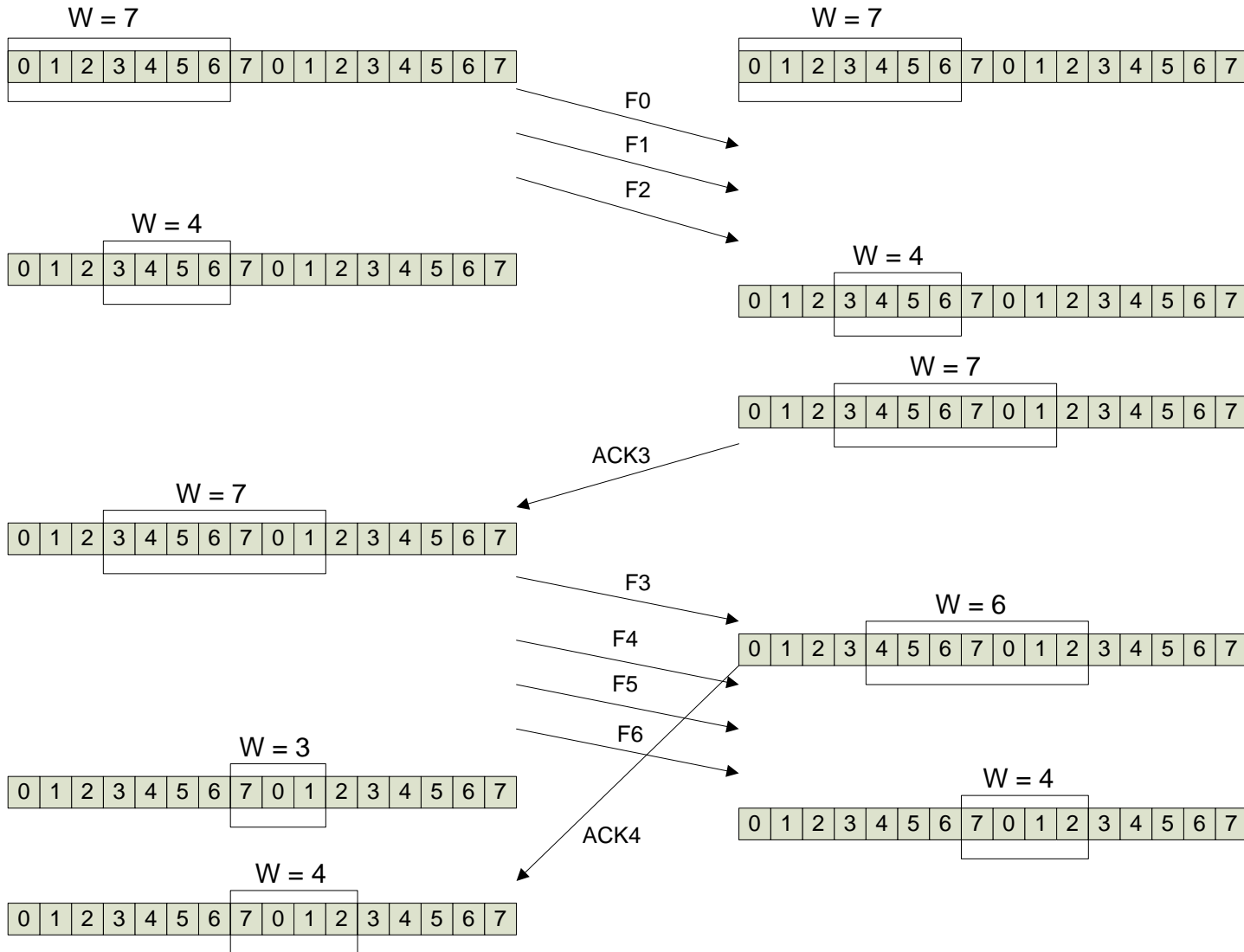
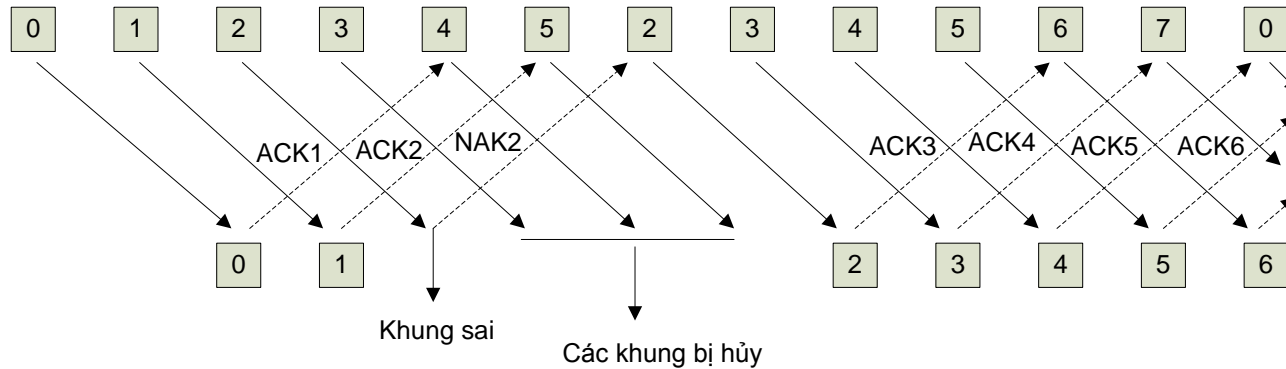


Điều khiển luồng kết hợp ARQ Go-back-N

- Phát nhiều khung thông tin liên tiếp trước khi nhận được xác nhận, nhiều nhất là W , được gọi là kích thước cửa sổ. Go-back-N và phương pháp *selective repeat* (trình bày ở phần sau) được gọi là cơ chế cửa sổ trượt (sliding window)
- Mỗi khi phát xong một khung thông tin, phía phát giảm kích thước cửa sổ đi 1. Khi kích thước cửa sổ bằng 0, phía phát sẽ ngừng. Điều này đảm bảo phía thu kịp xử lý
- Khi nhận được một ACK, phía phát sẽ tăng kích thước cửa sổ W lên 1. Tổng số khung mà phía thu phải xử lý tại một thời điểm vẫn không vượt quá W
- Để phân biệt các khung, cần đánh số thứ tự. Nếu dùng k bit để đánh số thì tổng số khung được đánh số sẽ là 2^k (từ 0 đến $2^k - 1$) và do đó, kích thước cửa sổ tối đa $W_{max} = 2^k$
- ACK có thể được đính vào gói phát theo chiều ngược (piggy back)

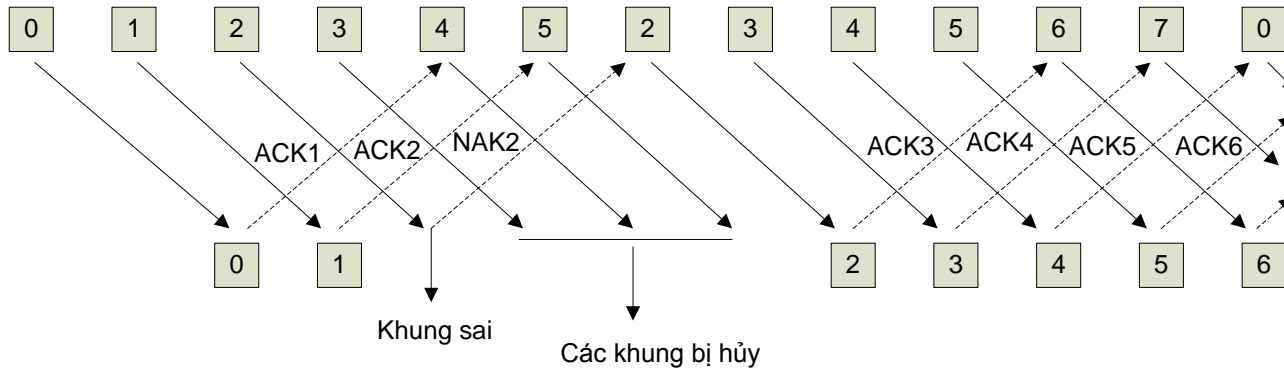


- Trong trường hợp lý tưởng (không có lỗi xảy ra) thì cơ chế cửa sổ trượt đảm bảo số khung thông tin từ phía phát đến phía thu không vượt quá kích thước cửa sổ. Trong trường hợp này, không có sự phân biệt giữa Go-back-N và selective repeat
- Khi có lỗi xảy ra, việc truyền lại các khung lỗi của cơ chế cửa sổ trượt được thực hiện theo hai cách khác nhau:
 - Go-back-N: phía phát sẽ thực hiện phát lại khung thông tin bị sai và tất cả các khung thông tin khác đã được truyền, tính từ khung bị sai.
 - Selective repeat: phía phát sẽ chỉ phát lại các khung thông tin bị sai



- **Khung thông tin bị lỗi – có thể xảy ra một trong ba trường hợp:**
 - Phía phát đã phát khung i , phía thu đã thu đúng các khung từ $i - 1$ trở về trước. Lúc này phía thu sẽ gửi $NAK\ i$ ($RN = i$). Khi phía phát nhận được $NAK\ i$, nó sẽ thực hiện phát lại khung i và tất cả các khung sau i (trường hợp khung 2)
 - Khung thông tin i bị mất trên đường truyền, giả sử phía thu nhận được khung $i+1$, lúc này phía thu thấy các khung đến không theo thứ tự (nhận được $i+1$ trước khi nhận được i) và hiểu rằng khung i đã mất. Phía thu sẽ gửi lại $NAK\ i$ cho phía phát

Khung thông tin bị lỗi



- Khung thông tin i bị mất trên đường truyền và phía phát không gửi thêm khung thông tin nào nữa. Lúc này phía thu không nhận được gì và không gửi lại ACK hay NAK. Phía phát chờ đến timeout của khung thông tin i và thực hiện truyền lại khung này

- **Khung ACK bị lỗi – ACK bị lỗi có thể xảy ra một trong hai trường hợp:**
 - Phía thu nhận được khung i và gửi $ACK(i+1)$ về phía phát và bị mất. Giả sử trước khi time-out của khung i xảy ra, phía phát nhận được $ACK(i+2)$ (hoặc $ACK(i+n)$ với $n > 1$) thì phía phát hiểu rằng khung i đã được nhận đúng. Kết luận này được giải thích như sau: khi phía thu gửi $ACK(i+2)$ nghĩa là phía thu đã nhận đúng (và chấp nhận) khung $i+1$, điều đó cũng đồng nghĩa với việc phía thu đã nhận đúng khung i . Người ta nói cơ chế của Go-back-N sử dụng *cummulative ACK* (nghĩa là các ACK sau cũng đồng thời báo nhận cho các khung trước đó)
 - Nếu trong khoảng thời gian time-out của khung i , phía phát không nhận được $ACK(i+n)$ nào cả thì sau time-out, phía phát sẽ phải phát lại khung i (và tất cả các khung sau đó)

- **Khung NAK bị lỗi – trong trường hợp NAK bị lỗi, nghĩa là khung i bị lỗi, lúc này phía thu sẽ không nhận thêm một khung nào sau khung i (và cũng sẽ không gửi báo nhận). Với trường hợp này phía phát bắt buộc phải chờ đến time-out và thực hiện phát lại khung thông tin i .**
- **Để đơn giản vấn đề, chúng ta không xem xét trường hợp ACK và NAK bị sai đồng thời**

- Trong trường hợp phía thu có khả năng xử lý W khung thông tin thì không cần bộ đệm. Phía thu chỉ nhận và xử lý thông tin theo đúng thứ tự (dựa trên số thứ tự đánh trên các khung)
- Phía phát phải lưu tối đa là W khung thông tin trong bộ đệm để chờ ACK

- Ta biết với kỹ thuật stop-and-wait, nếu không có lỗi đường truyền thì:

$$\eta_{stop-and-wait} = \frac{1}{1+2a} \quad \text{với} \quad a = \frac{T_F}{T_D}$$

- Có thể hiểu rằng $(1+2a)$ chính là số gói có thể phát đi trong khoảng thời gian tính từ lúc phát đi gói đầu tiên cho đến khi nhận xong gói ACK đầu tiên
- Với go-back-N có $W < 1+2a$ thì $\eta_{ideal} = \frac{W}{1+2a}$
- Với go-back-N có $W \geq 1+2a$ thì $\eta_{ideal} = 1$

- Xác suất lỗi là p , khi đó:

$$\eta_{\text{reality}} = \frac{\eta_{\text{ideal}}}{N_R}$$

với N_R là số lần trung bình phát lại cho đến khi nhận đúng

- Xác suất để phải phát một gói i lần là:

$$p(i) = p^{i-1}(1-p)$$

- Gọi số gói phải phát lại trong mỗi lần là K , suy ra tổng số gói phải phát trong i lần là:

$$f(i) = 1 + (i-1)K$$

- Có:

$$N_R = \sum_{i=1}^{\infty} f(i).p(i) = \sum_{i=1}^{\infty} [1 + (i-1)K].p^{i-1}(1-p)$$

- **Chú ý rằng:**
$$\sum_{i=0}^{\infty} r^i = \sum_{i=1}^{\infty} r^{i-1} = \frac{1}{1-r}$$

và

$$\sum_{i=1}^{\infty} i \cdot r^{i-1} = \frac{1}{(1-r)^2}$$

- **Ta có kết quả:**
$$N_R = 1 - K + \frac{K}{1-p} = \frac{1-p+Kp}{1-p}$$

- **Dễ thấy:**

- Nếu $W \geq 2a + 1$ thì $K \approx 2a + 1$ – do khi NACK của khung i về thì phía phát đã phát thêm được $\approx 2a + 1$ khung
- Nếu $W < 2a + 1$ thì $K = W$ – do khi NACK của khung i về thì phía phát đã phát xong kích thước cửa sổ (W khung) và đang chờ báo nhận cho khung i để phát tiếp.

- Cuối cùng, hiệu suất được tính:

$$\eta_{Go-back-N} = \frac{1-p}{1+2ap}$$

nếu $W \geq 2a + 1$, và:

$$\eta_{Go-back-N} = \frac{W(1-p)}{(2a+1)(1-p+Wp)}$$

nếu $W < 2a + 1$

- Như vậy:
 - Ưu điểm của phương pháp ARQ Go-back-N là hiệu suất cao hơn so với phương pháp ARQ dừng và đợi.
 - Cơ chế xử lý thông tin ở phía thu khá đơn giản và không cần bộ đệm
 - Tuy nhiên, có thể phải truyền lại quá nhiều khung thông tin trong trường hợp bị lỗi. Để khắc phục nhược điểm này, người ta đề xuất sử dụng cơ chế ARQ phát lại theo yêu cầu (Selective repeat ARQ)