



Marco Listanti

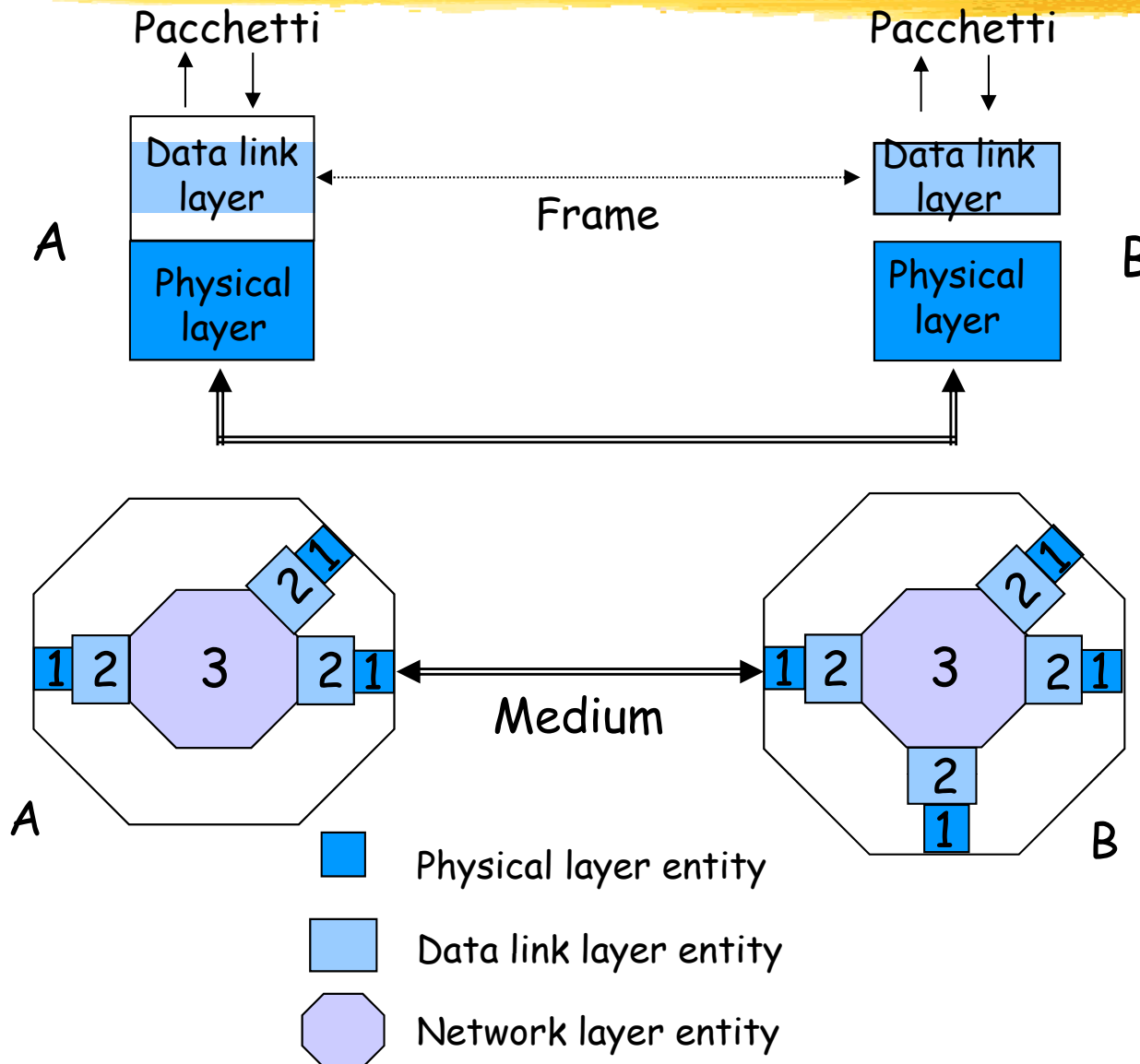
Lo strato di collegamento Parte 3

"Error Recovery"

End-to-End vs. Hop-by-Hop

- Una funzione può essere eseguita
 - da estremo a estremo (**end-to-end**) (Strato di trasporto)
 - tratta per tratta (**hop-by-hop**) (Strato di Data link)
- Esempi
 - Controllo d'errore eseguito in ogni hop del percorso di rete oppure solamente tra sorgente e destinazione
 - Controllo di flusso eseguito in ogni hop del percorso di rete oppure solamente tra sorgente e destinazione
- Quali sono i vantaggi e gli svantaggi dei due approcci ?

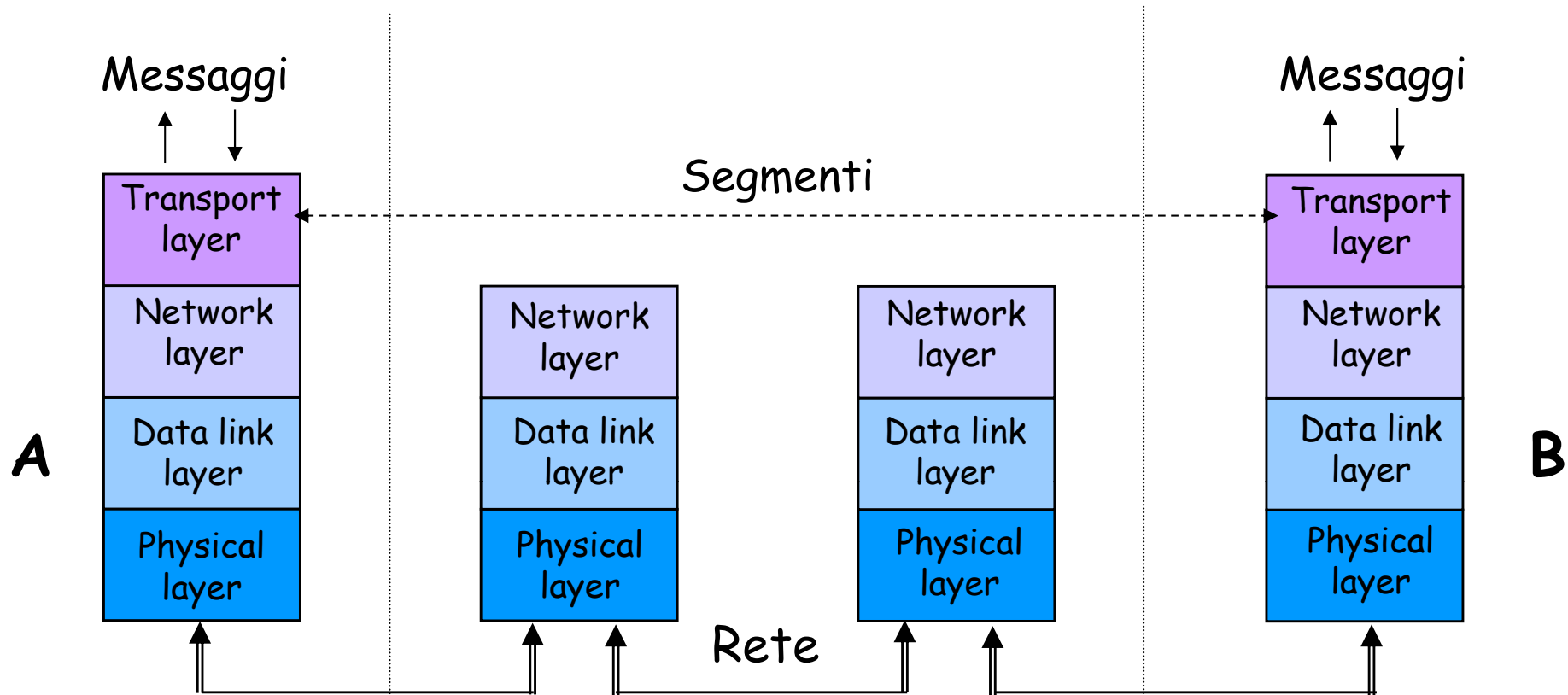
Controllo d'errore nello strato di Data Link



- Lo strato di data link opera punto-punto tra due elementi di rete direttamente connessi
- Le frame possono subire errori, ma è preservata la sequenza
- Lo strato di Data link esegue la rivelazione degli errori e la ritrasmissione delle frame errate
- Viene assicurato il trasferimento affidabile dei pacchetti

Controllo d'errore nello strato di Trasporto

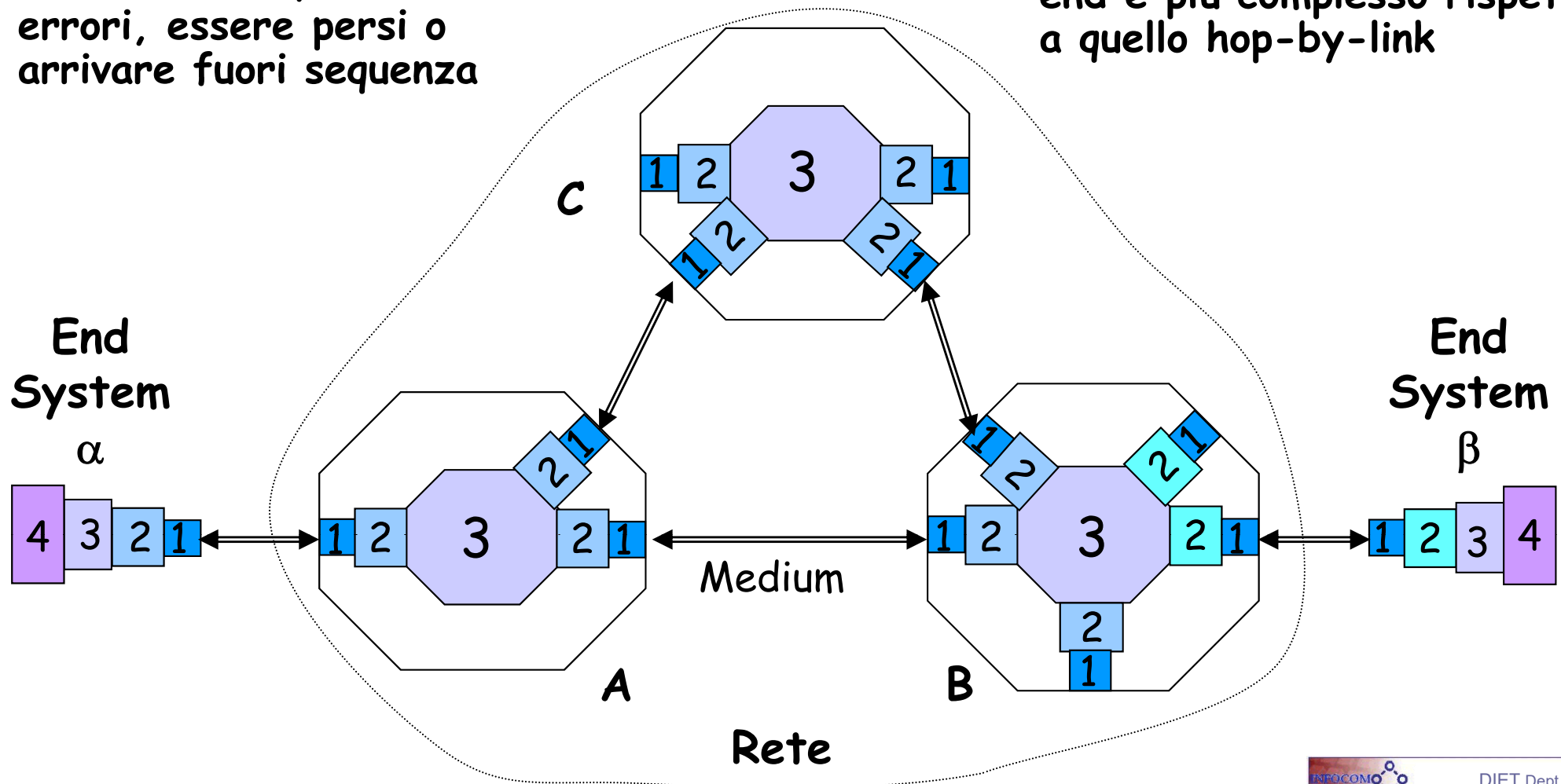
- Il protocollo di strato di trasporto (es. TCP) emette i segmenti ed esegue end-to-end sia la rivelazione d'errore che la ritrasmissione
- La rete è considerata inaffidabile



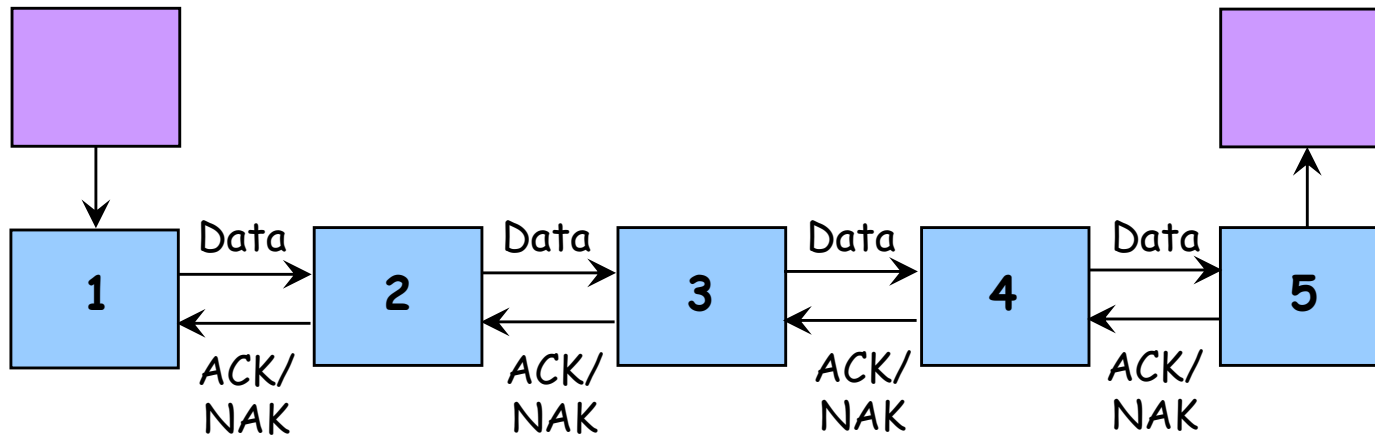
Controllo d'errore nello strato di Trasporto

- I segmenti possono subire ritardi elevati, subire errori, essere persi o arrivare fuori sequenza

- Il controllo d'errore end-to-end è più complesso rispetto a quello hop-by-link

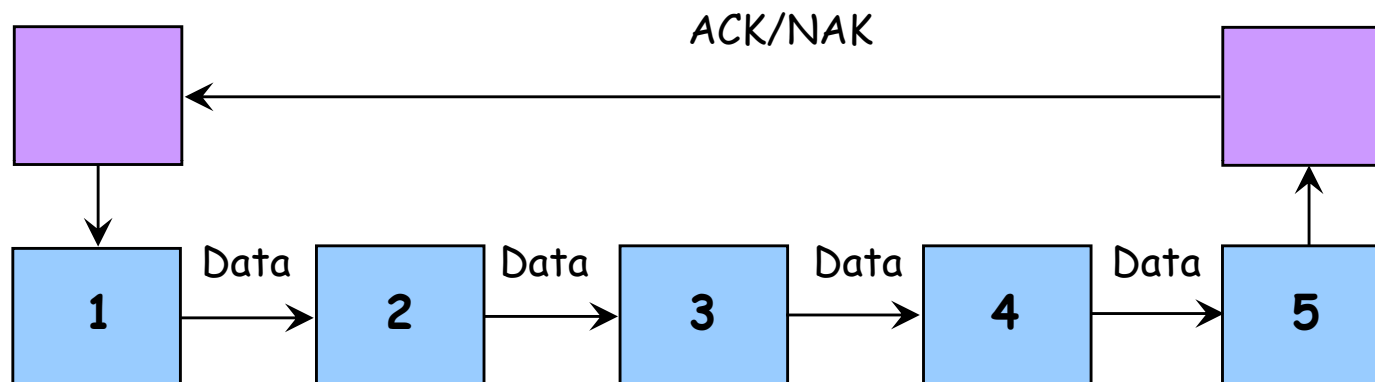


End-to-End vs. Hop-by-Hop



■ Hop-by-hop

- non può assicurare la correttezza end-to-end
- Recupero più veloce



■ End-to-end

- Semplicità delle procedure di rete
- Maggiore scalabilità



Automatic Repeat Request (ARQ)

Automatic Repeat Request (ARQ)

■ Obiettivo

- assicurare che una sequenza di PDU sia consegnata in ordine e senza errori o duplicazioni in presenza di un servizio offerto dagli strati sottostanti che introduce errori e/o perdite

■ Possibili procedure alternative

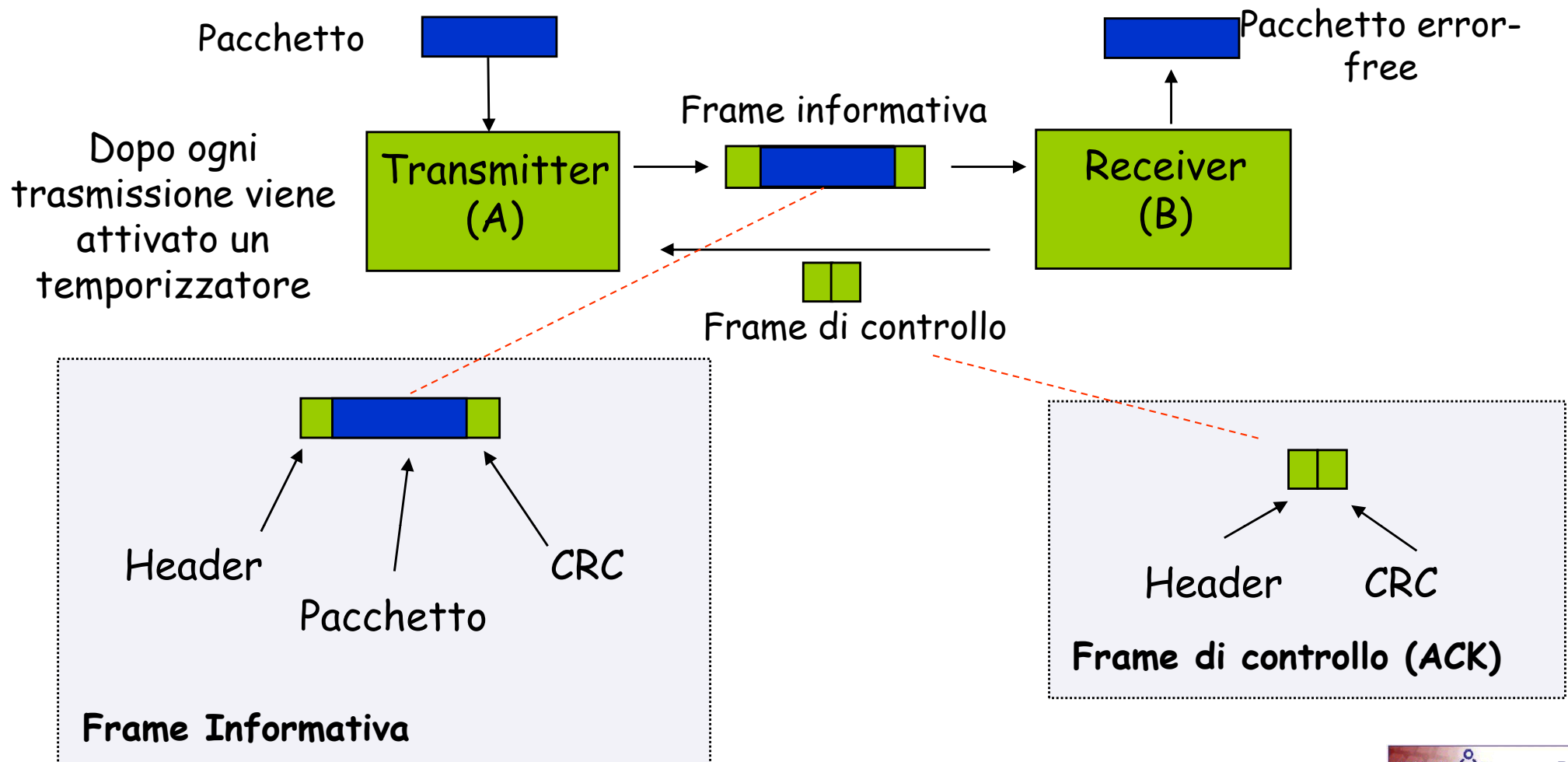
- Stop-and-Wait ARQ
- Go-Back N ARQ
- Selective Repeat ARQ

■ Elementi chiave delle procedure ARQ

- Codici di rivelazione d'errore
- Riscontri positivi (ACK)
- Riscontri negativi (NACK)
- Timeout

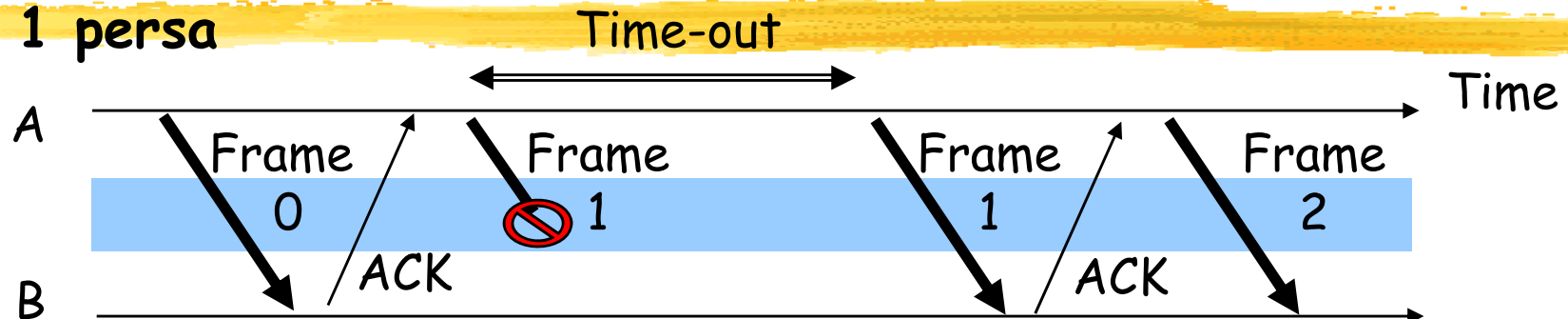
Stop-and-Wait ARQ

- L'entità A trasmette una frame ed aspetta l'ACK

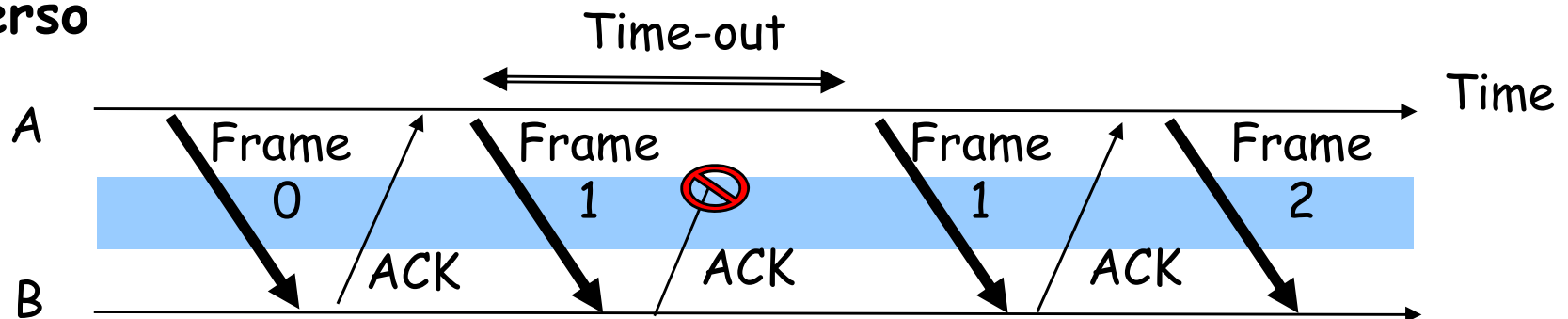


Numeri di Sequenza

Frame 1 persa



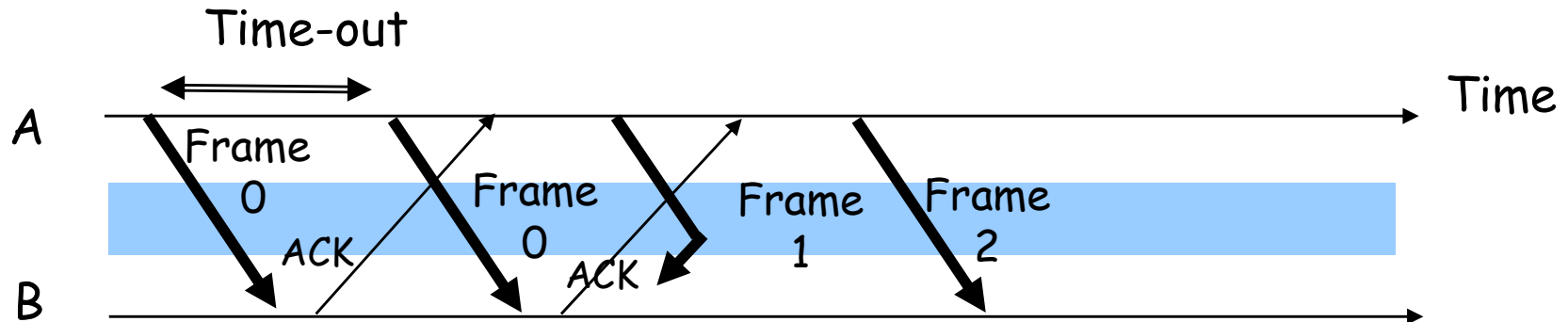
ACK perso



- L'entità emittente (A) si comporta sempre nello stesso modo
- Nel secondo caso, l'entità ricevente (B) riceve la frame 1 due volte (duplicazione)
- B rivela la duplicazione mediante il numero di sequenza (S_{last}) contenuto nell'header di ciascuna frame

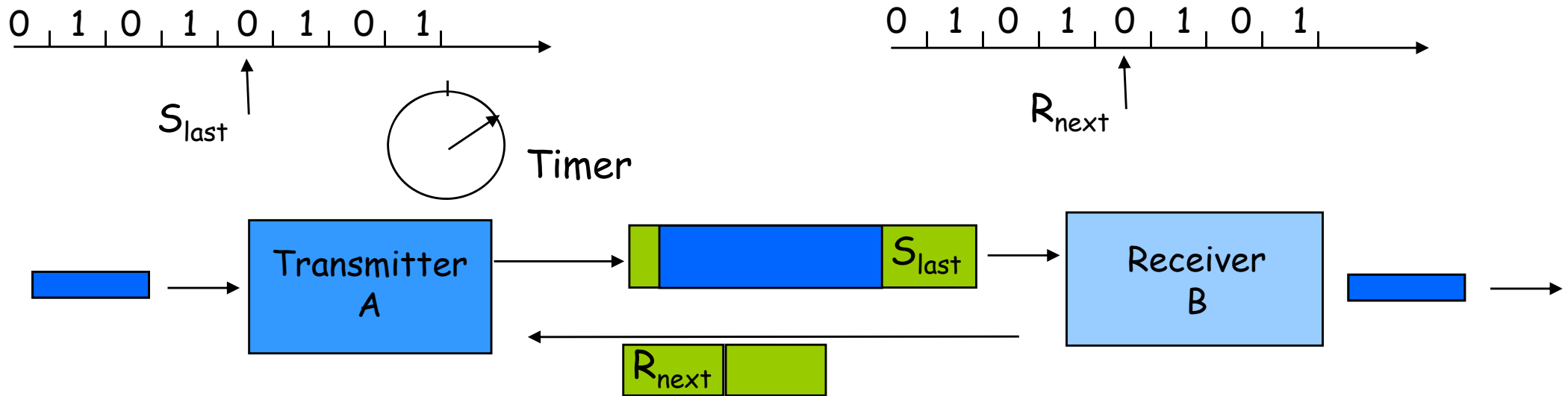
Numeri di Sequenza

Esaurimento prematuro del time-out



- La stazione emittente interpreta in modo scorretto gli ACK
 - Per a il secondo ACK riscontra la frame 1 che invece è persa
- Occorre inserire il numero di sequenza anche negli ACK (R_{next})
 - indica il numero di sequenza della prossima frame che il ricevitore si aspetta di ricevere
 - implicitamente riscontra tutte le frame con numero di sequenza $R' < R$

Numero di sequenza 1-Bit



Stato:
(S_{last} , R_{next})



Stop-and-Wait ARQ (Trasmitter)

■ Stato Ready

- Attesa di una richiesta di invio di un pacchetto dallo strato superiore
- Quando arriva una richiesta, si trasmette la frame con numero di sequenza S_{last} e completa di CRC
- Transizione nello stato Wait

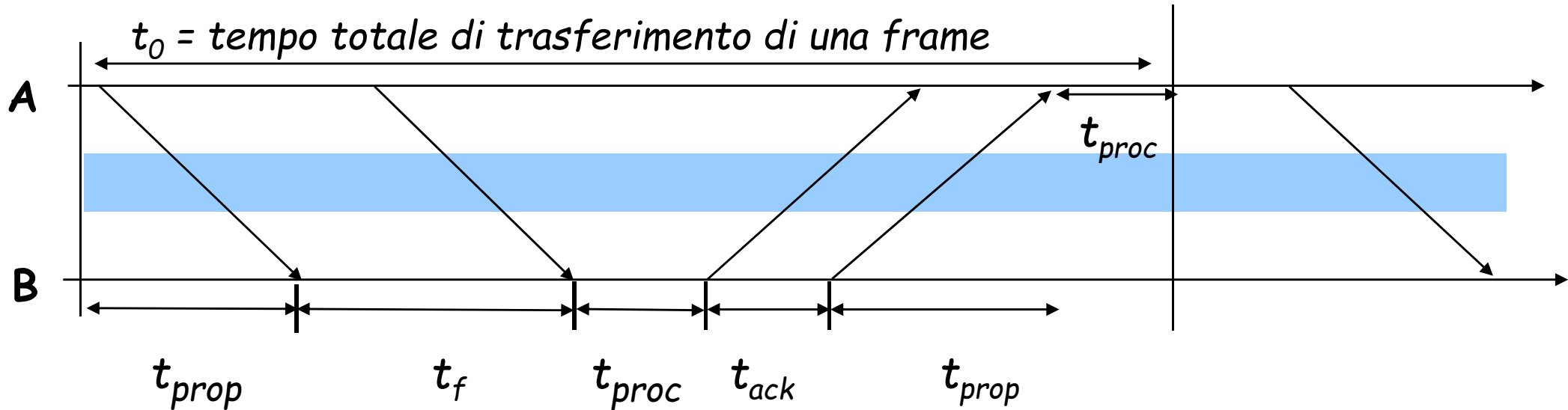
■ Stato Wait

- Attesa del riscontro della frame emessa o dell'esaurimento del timeout (la ricezione delle richieste dallo strato superiore sono bloccate)
- Se il timeout scade viene ritrasmessa la frame e viene riavviato il timer
- Se viene ricevuto un ACK
 - Se il numero di sequenza non è corretto l'ACK è ignorato
 - Se il numero di sequenza è corretto ($R_{next} = S_{last} + 1$), la frame è accettata e si torna nello stato Ready

Stop-and-Wait ARQ (Receiver)

- **Sempre nello stato Ready**
 - Attesa dell'arrivo di una nuova frame
 - Quando arriva una frame viene eseguito il controllo d'errore (CRC)
 - Se non sono rivelati errori e il numero di sequenza è corretto ($S_{last} = R_{next}$)
 - la frame è accettata
 - viene aggiornato il valore di R_{next}
 - viene emesso l'ACK con valore R_{next}
 - il pacchetto è consegnato allo strato superiore
 - Se non sono rivelati errori e il numero di sequenza non è corretto
 - la frame è scartata
 - viene emesso un ACK with R_{next} (ACK duplicato)
 - Se sono rivelati errori
 - la frame è scartata

Modello Stop-and-Wait ARQ



$$\begin{aligned}
 t_0 &= 2t_{prop} + 2t_{proc} + t_f + t_{ack} \\
 &= 2t_{prop} + 2t_{proc} + \frac{n_f}{R} + \frac{n_a}{R}
 \end{aligned}$$

Lunghezza di una frame
 Lunghezza di un ACK
 Bit rate canale

Efficienza su un canale senza-errori

Rate di trasmissione efficace

bit di overhead

$$R_{eff}^0 = \frac{\text{numero di bit informativi consegnati a destinazione}}{\text{tempo totale necessario per la consegna dei bit informativi}} = \frac{n_f - n_o}{t_0},$$

Efficienza di trasmissione

$$\eta_0 = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_o}{t_0}}{R} = \frac{1 - \frac{n_o}{n_f}}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}}$$

Effetto dell'overhead di una frame

Effetto di un ACK

Effetto del prodotto Banda-Ritardo

Esempio: Impatto del prodotto banda-ritardo

- $n_f = 1250$ byte = 10000 bits, $n_a = n_o = 25$ byte = 200 bit

2xDelayxBW Efficiency	1 ms 200 km	10 ms 2000 km	100 ms 20000 km	1 sec 200000 km
1 Mbit/s	10^3 88%	10^4 49%	10^5 9%	10^6 1%
1 Gbit/s	10^6 1%	10^7 0.1%	10^8 0.01%	10^9 0.001%

- La tecnica Stop-and-Wait non è efficiente in link ad alta velocità o con elevati ritardi di propagazione

Efficienza su un canale con errori

- Sia $1-P_f$ = probabilità che una frame arrivi senza errori
- $1/(1-P_f)$ = numero medio di trasmissioni necessarie per avere una trasmissione corretta di una frame
- $t_o/(1 - P_f)$ = tempo medio di trasferimento di una frame

$$\eta_{SW} = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_o}{t_o}}{R} = \frac{1 - P_f}{R} = \frac{1 - \frac{n_o}{n_f}}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f} (1 - P_f)}$$

Effetto della probabilità di perdita delle frame

Esempio: Impatto del Bit Error Rate

- $n_f = 1250$ byte = 10000 bit, $n_a = n_o = 25$ byte = 200 bit
- Calcolo dell'efficienza per un BER $p = 0, 10^{-6}, 10^{-5}, 10^{-4}$

$1 - P_f = (1 - p)^{n_f} \approx e^{-n_f p}$ per grandi valori di n_f e per piccoli valori di p

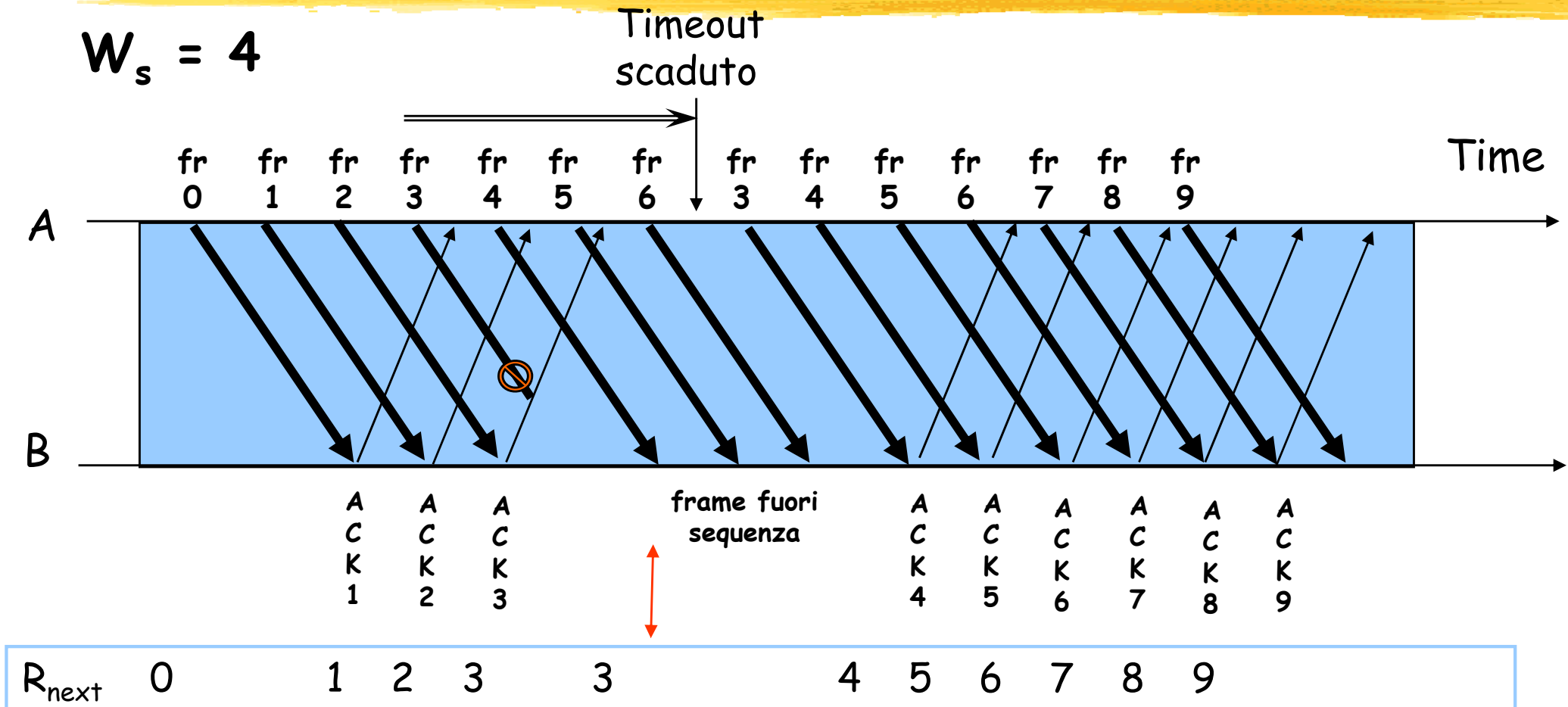
$1 - P_f$ Efficiency	0	10^{-6}	10^{-5}	10^{-4}
R=1 Mbps	1	0.99	0.905	0.368
$T_{prop} = 1$ ms	88%	86.6%	79.2%	32.2%

- Gli errori introducono un effetto significativo quando il prodotto $n_f p$ si avvicina ad 1

Go-back N ARQ

- **Miglioramento del protocollo Stop-and-Wait**
- **Elimina le attese dei riscontri**
 - Il canale è mantenuto occupato inviando altre frame
 - Utilizza una **finestra in trasmissione** di ampiezza W_s frame
 - Usa m bit per la numerazione delle frame
- **Se vengono ricevuti gli ACK delle frame emesse prima di esaurire la finestra, la finestra è aggiornata e la trasmissione delle frame può continuare**
- **Se la finestra si esaurisce, la trasmissione viene interrotta in attesa degli ACK**
- **Se non sono ricevuti ACK, allo scadere di un timeout le frame della finestra vengono ritrasmesse**

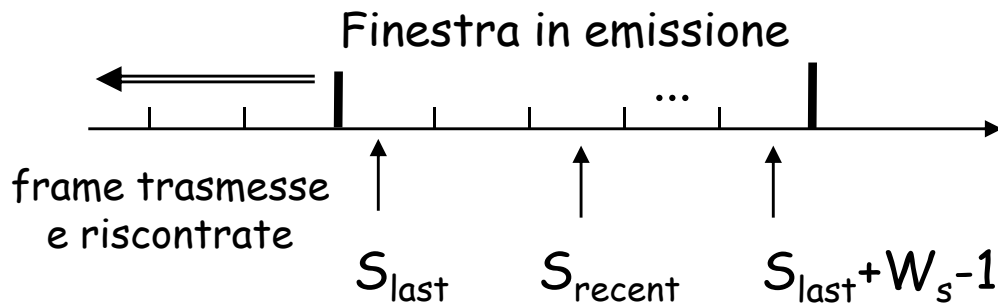
Go-back-N ARQ



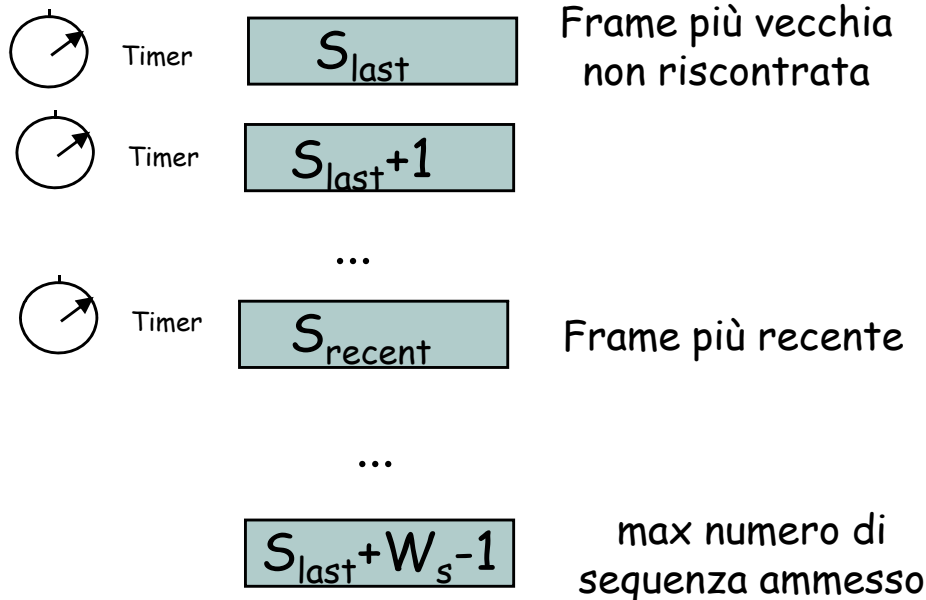
- Le frame con errori e fuori sequenza sono scartate

Go-Back-N Transmitter & Receiver

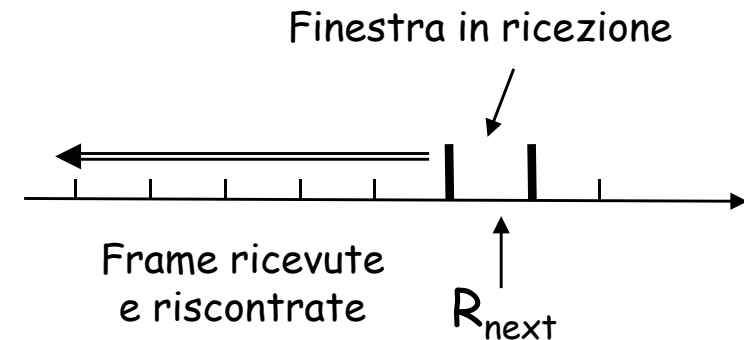
Transmitter



Buffer



Receiver

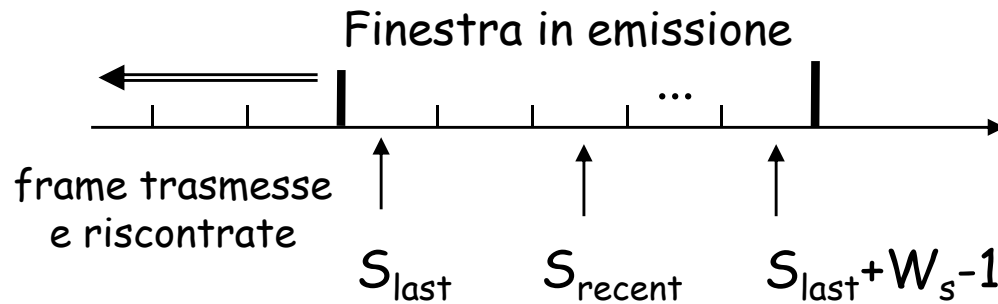


Il Receiver accetta solo frame corrette e in sequenza (con numero di sequenza = R_{next})

Quando arriva una nuova frame in sequenza, viene incrementato di uno R_{next} , quindi la finestra in ricezione slitta di una unità

Sliding window

Transmitter

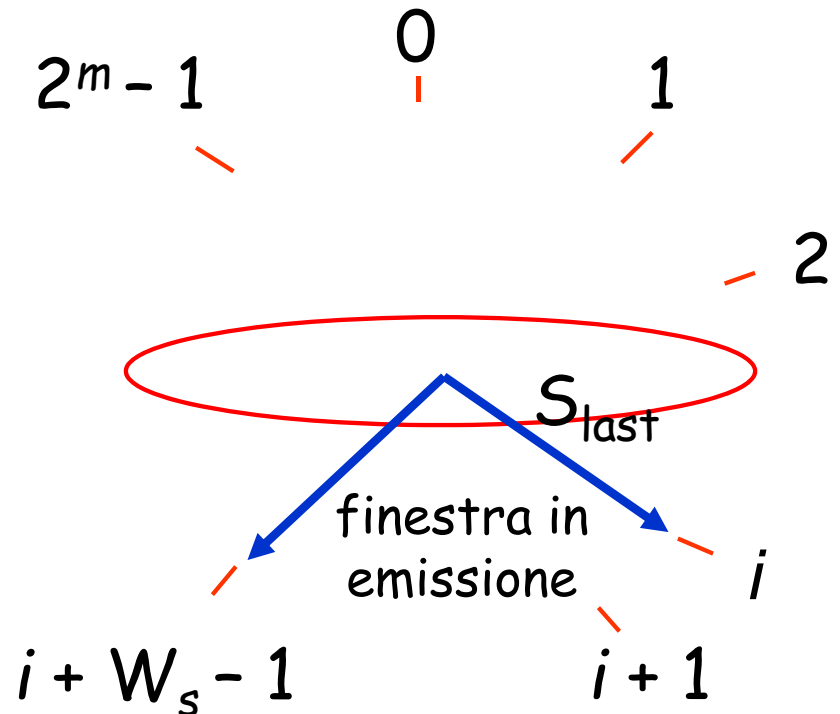


Il Transmitter attende gli ACK (con numero di sequenza $S \geq S_{last}$)

Quando arriva un ACK, con numero di sequenza S , viene posto $S_{last} = S$

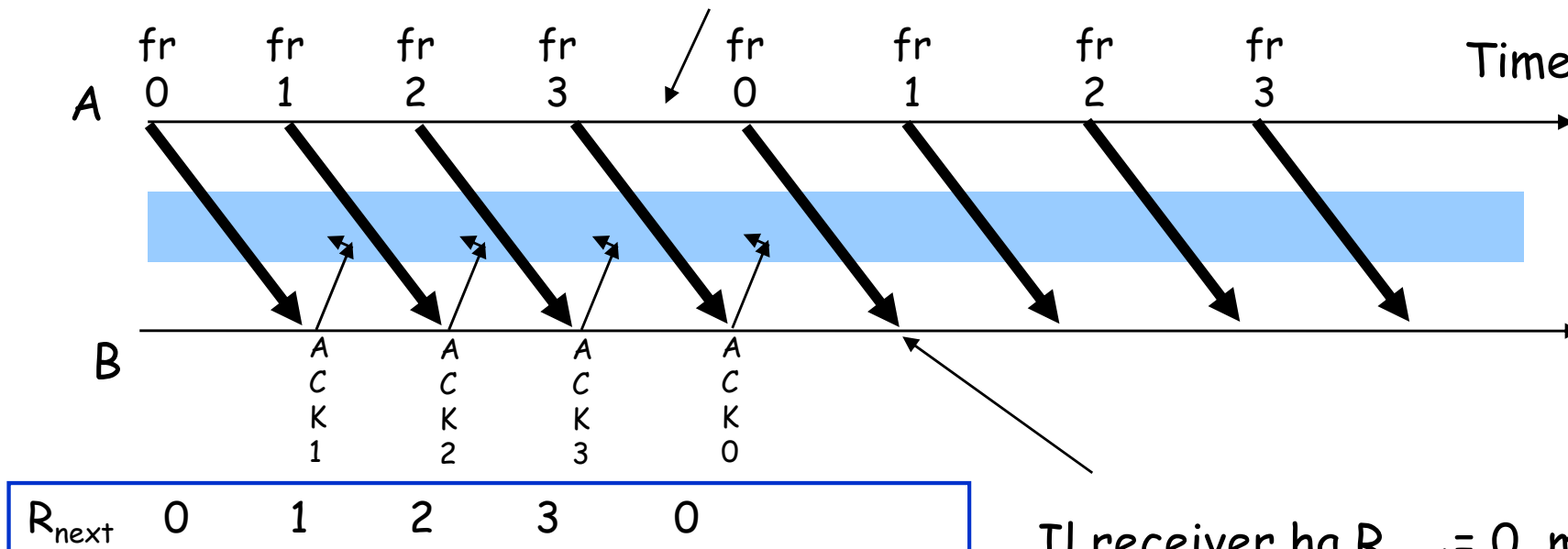
L'estremo superiore della finestra sarà quindi $S_{last} + W_s - 1$

Numeri di sequenza a m bit
modulo $M = 2^m$



Dimensione massima della finestra

$$W_s = M = 2^m = 4$$

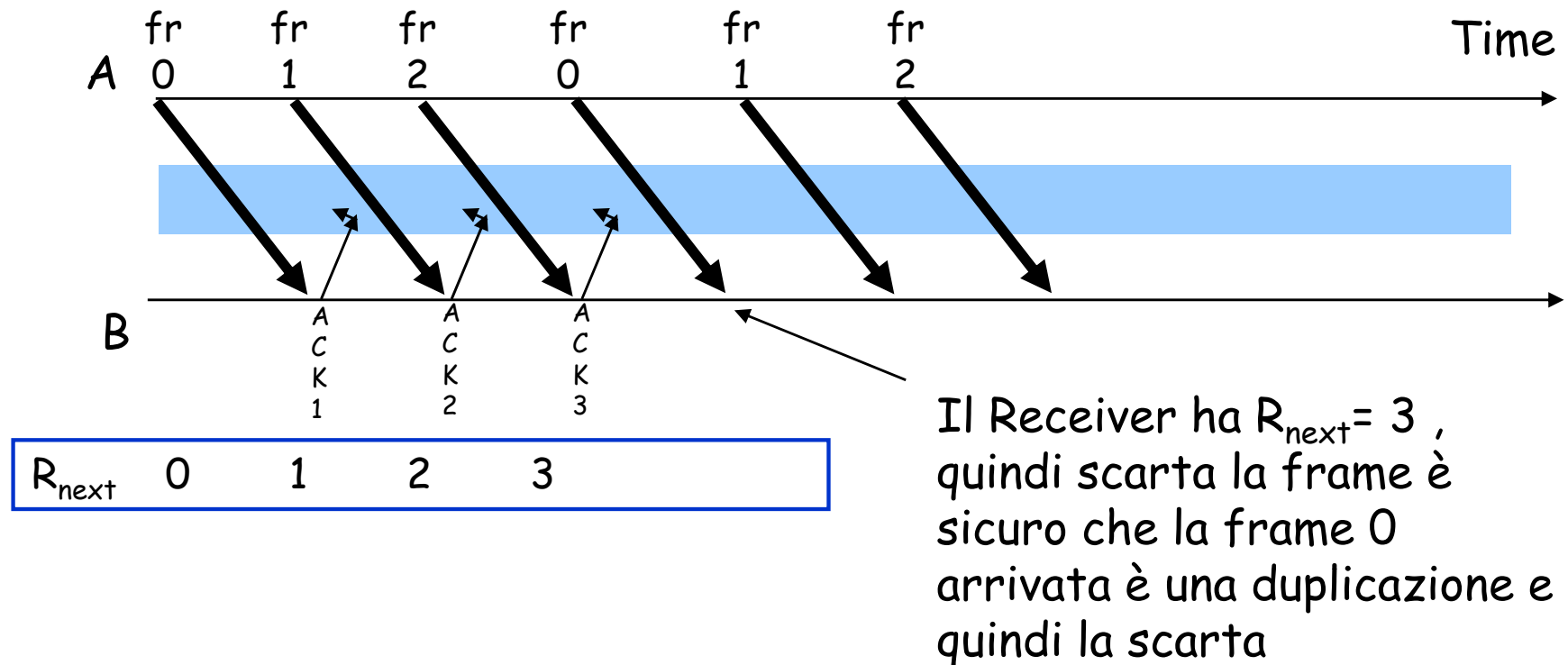


Il massimo valore della finestra è uguale a $W_s = M = 2^m$

Il receiver ha $R_{next} = 0$, ma non è in grado di distinguere se il suo ACK per la frame 0 è stato ricevuto e quindi la frame arrivata è nuova oppure si tratta della ritrasmissione della vecchia frame 0

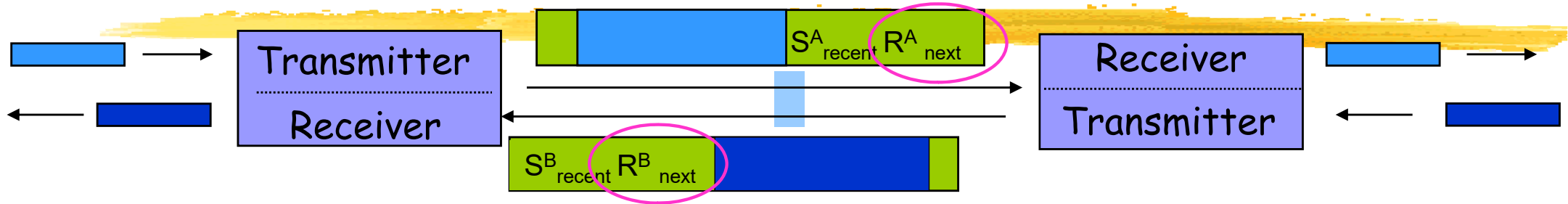
Dimensione massima della finestra

$$W_s = M = 2^m - 1 = 3$$

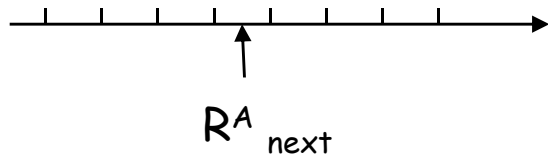


Il massimo valore della finestra è uguale a $W_s = M - 1 = 2^m - 1$

Piggybacking



"A" finestra in ricezione



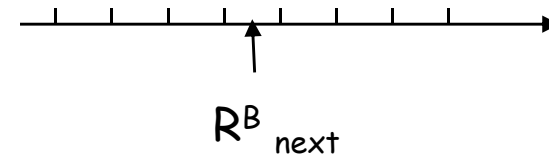
"A" Finestra in trasmissione



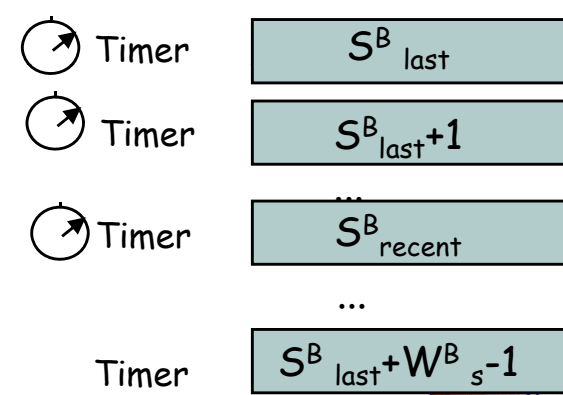
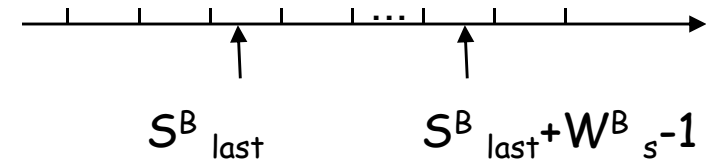
Buffer

Le frame fuori
sequenza sono
scartate

"B" finestra in ricezione

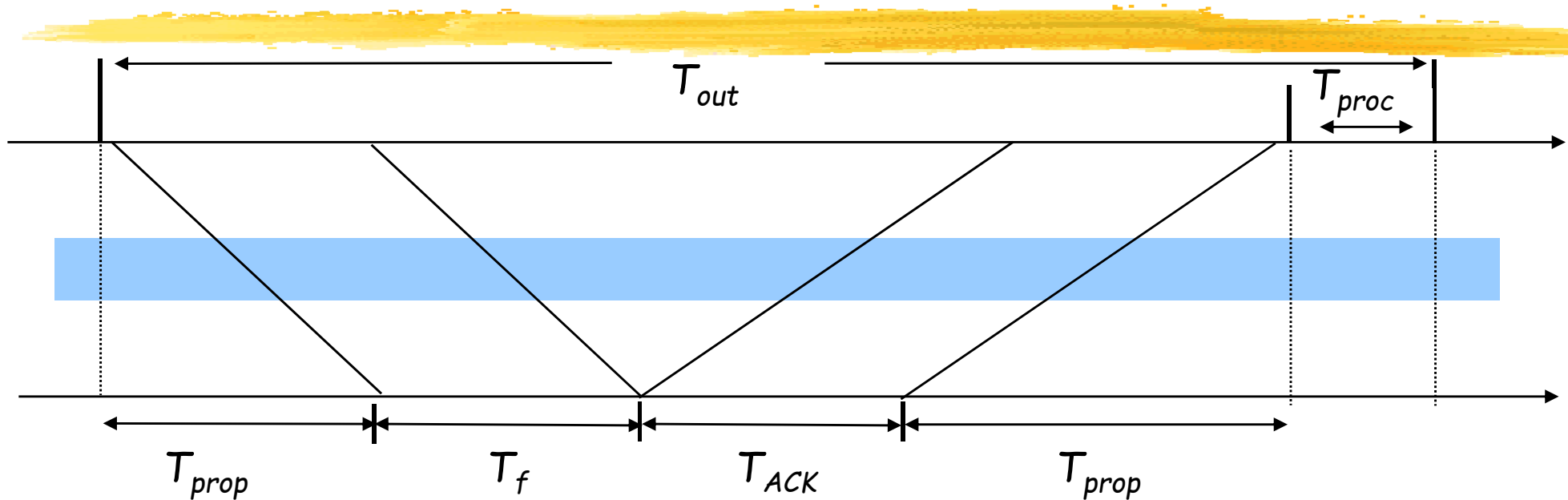


"B" Finestra in trasmissione



Buffer

Dimensionamento della finestra e del timeout



- Il valore del Timeout (T_{out}) deve essere la somma delle seguenti componenti
 - due tempi di propagazione + un tempo di processing = $2 T_{prop} + T_{proc}$
 - Un tempo di trasmissione di una frame informativa T_f
 - Un tempo di trasmissione della frame ACK, T_{ACK}
- W_s deve essere grande abbastanza da poter mantenere il canale occupato per tutto il periodo T_{out}



Dimensione della finestra vs. prodotto banda-ritardo

Frame = 1250 bytes = 10,000 bits, $R = 1$ Mbps

$2(t_{\text{prop}} + t_{\text{proc}})$	$2 \times \text{Delay} \times \text{BW}$	Window
1 ms	1000 bits	1
10 ms	10,000 bits	2
100 ms	100,000 bits	11
1 second	1,000,000 bits	101

Efficienza del Go-Back-N

Tempo di trasferimento di una frame

$$t_{GBN} = t_f (1 - P_f) + P_f \left\{ t_f + \frac{W_s t_f}{1 - P_f} \right\} = t_f + P_f \frac{W_s t_f}{1 - P_f}$$

Efficienza

$$\eta_{GBN} = \frac{n_f - n_o}{R} = \frac{1 - \frac{n_o}{n_f}}{1 + (W_s - 1)P_f} (1 - P_f)$$

Impatto del BER su GBN

- $n_f=1250$ bytes = 10000 bits, $n_a=n_o=25$ bytes = 200 bits
- Random bit errors with $p=0, 10^{-6}, 10^{-5}, 10^{-4}$
- $R = 1$ Mbps, Delay = 100 ms
- $1 \text{ Mbps} \times 100 \text{ ms} = 100000 \text{ bits} = 10 \text{ frames} \rightarrow W_s = 11$

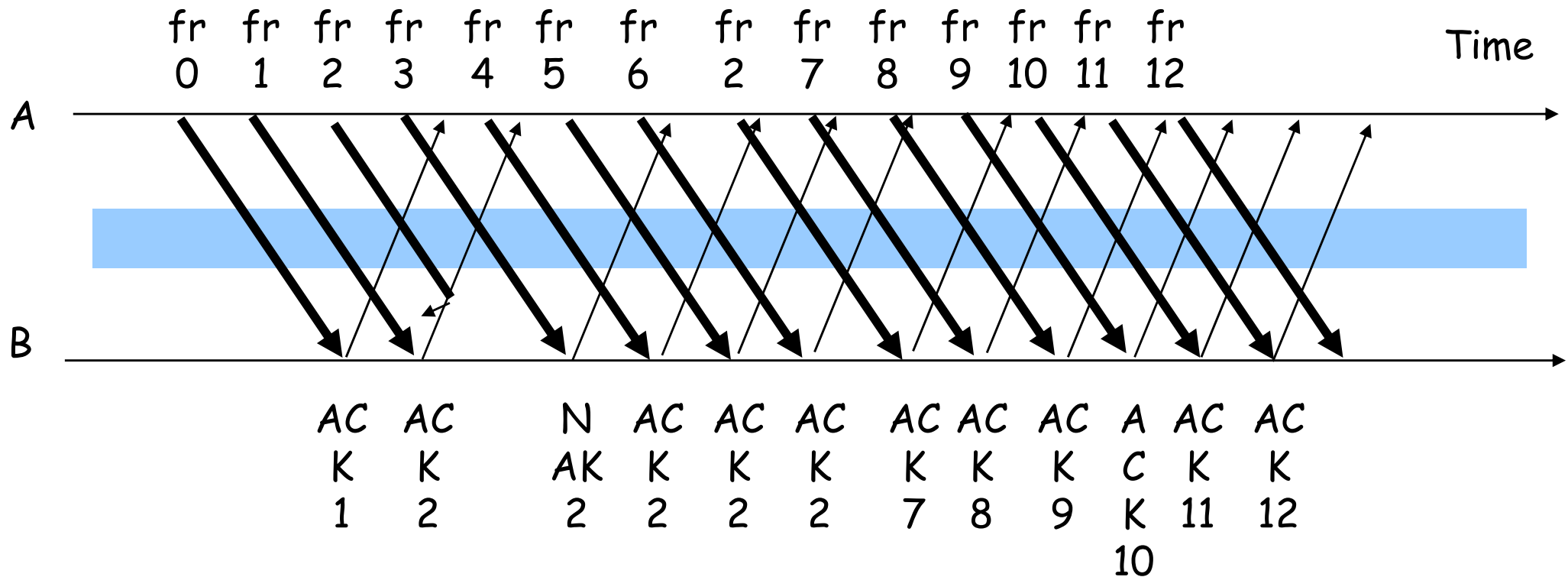
Efficiency	0	10^{-6}	10^{-5}	10^{-4}
S&W	8.9%	8.8%	8.0%	3.3%
GBN	98%	88.2%	45.4%	4.9%

- Go-Back-N è migliore di S&W nei casi di elevato valore del prodotto banda ritardo
- Go-Back-N diviene inefficiente se il BER cresce

Selective Repeat ARQ

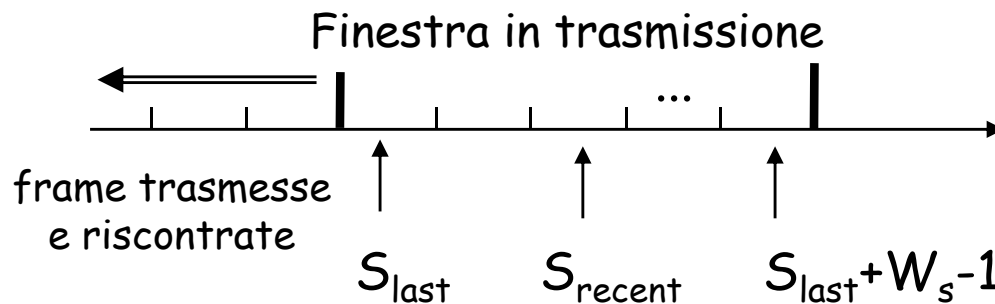
- **Go-Back-N ARQ è inefficiente poichè, in caso di ritrasmissione, è riemesso un numero elevato di frame, anche se ricevute correttamente dal receiver**
- **Selective Repeat ritrasmette solo le frame che sono state perse**
 - l'esaurimento del Timeout determina la ritrasmissione solo del frame corrispondente
 - La ricezione di un NAK causa la ritrasmissione della trama non riscontrata più vecchia
- **Il Receiver gestisce una finestra in ricezione che indica i numeri di sequenza che possono essere accettati**
 - Frame corrette, ma fuori sequenza con numero di sequenza compreso nella finestra in ricezione non sono scartate, ma sono bufferizzate
 - Un arrivo di una frame con R_{next} determina lo scorrimento della finestra in trasmissione

Selective Repeat ARQ

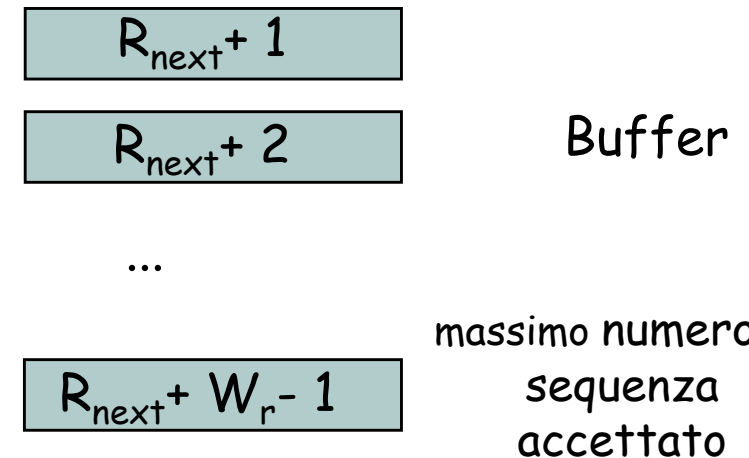
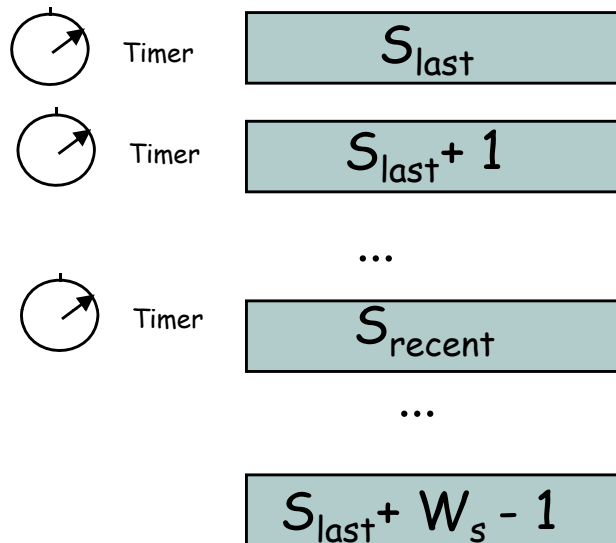
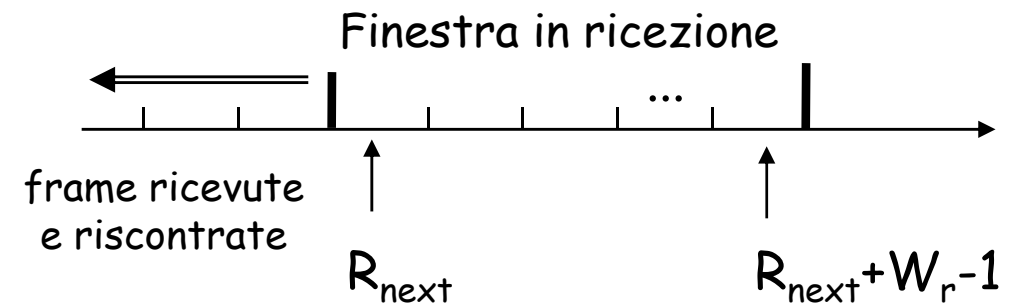


Selective Repeat ARQ

Transmitter

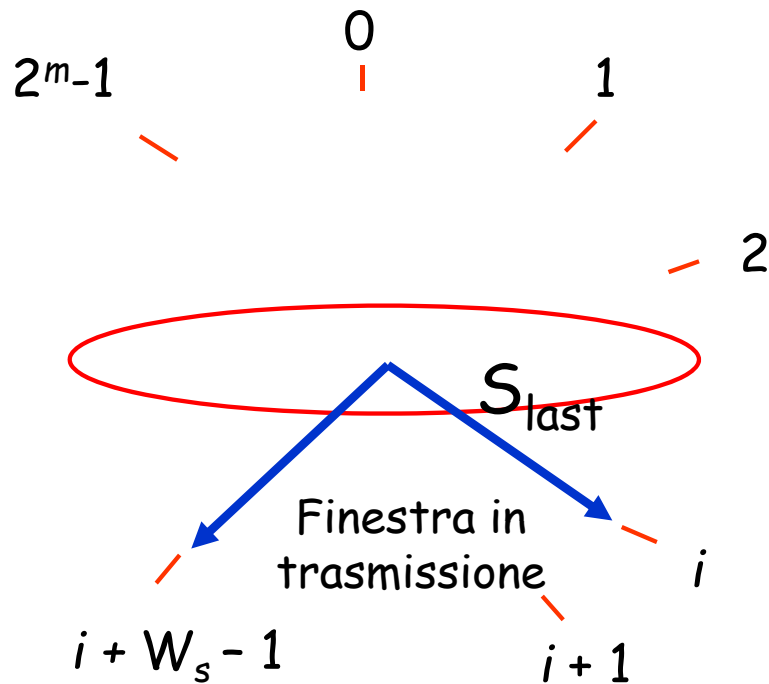


Receiver



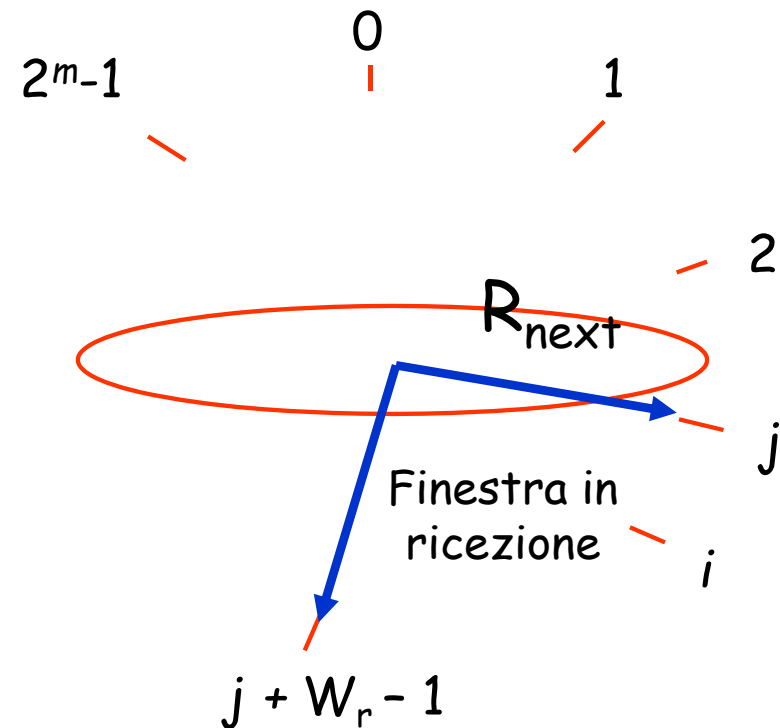
Finestre in trasmissione e ricezione

Transmitter



I limiti della finestra si spostano in avanti di k se arriva un ACK con $R_{next} = S_{last} + k$
 $k = 1, \dots, W_s - 1$

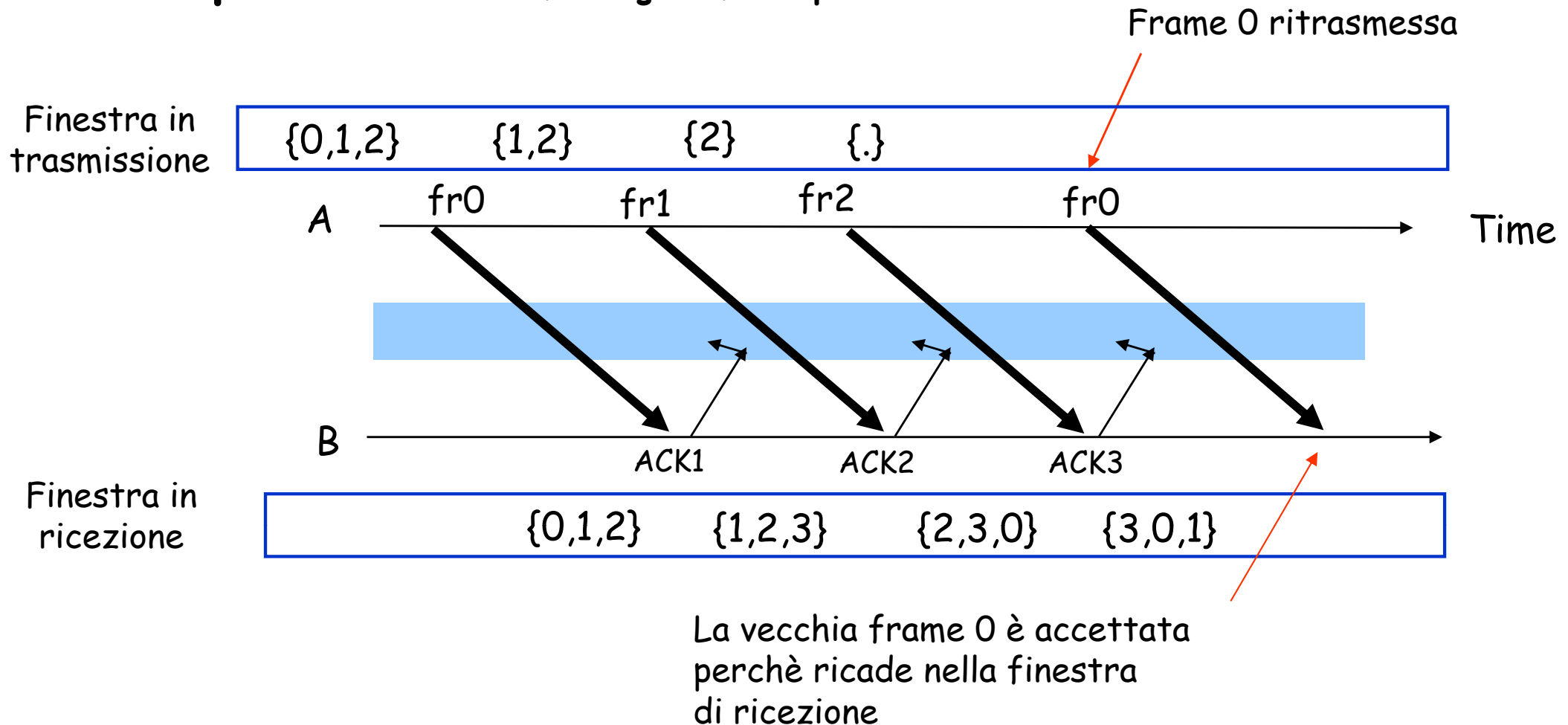
Receiver



I limiti della finestra si spostano in avanti di 1 se arriva una frame con numero di sequenza = R_{next}

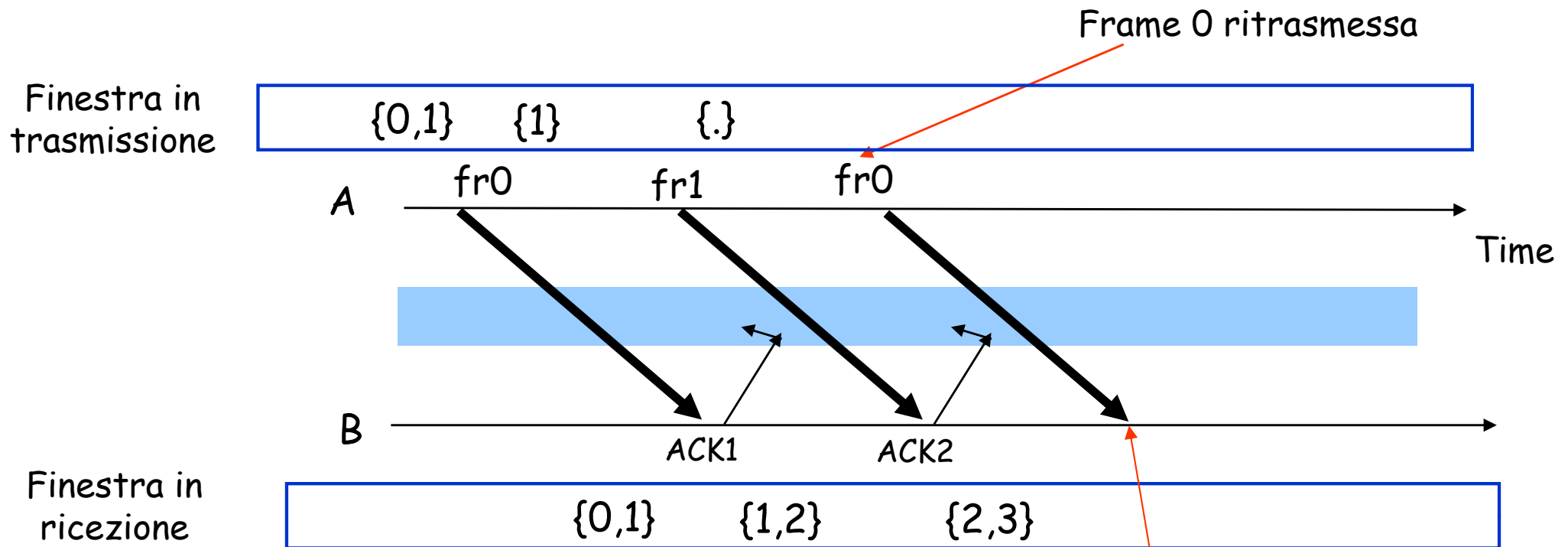
Valori massimi di W_s e W_r

■ Esempio: $M=2^2=4$, $W_s=3$, $W_r=3$



Valori massimi di W_s e W_r

■ Esempio: $M=2^2=4$, $W_s=2$, $W_r=2$

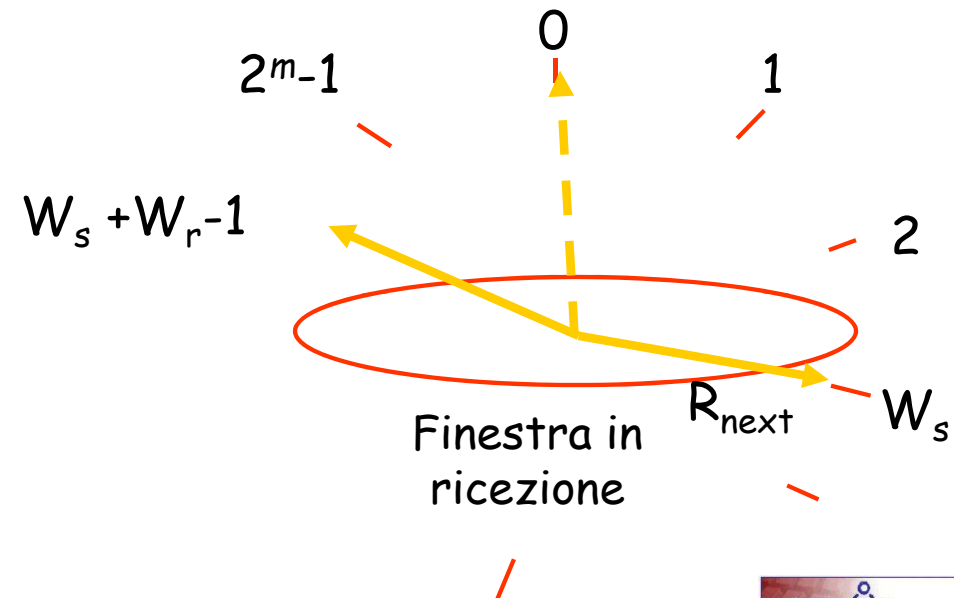
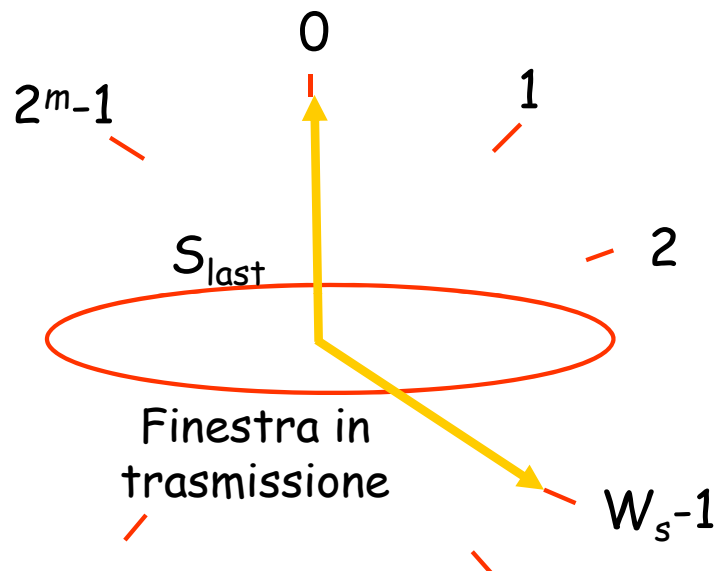


Il massimo valore permesso è
 $W_s + W_r = 2^m$

La vecchia frame 0 è rifiutata
 perchè ricade fuori dalla finestra
 di ricezione

Perchè $W_s + W_r = 2^m$

- Il Transmitter emette le frame da 0 a W_s-1 ; la finestra di trasmissione è vuota
- Tutte le frame arrivano al receiver
- Tutti gli ACKs sono persi
- Il Transmitter riemette la frame 0
- La finestra di ricezione inizia a $\{0, \dots, W_r\}$
- La finestra di ricezione slitta a $\{W_s, \dots, W_s+W_r-1\}$
- Il ricevitore rifiuta la frame 0 perchè è fuori dalla finestra di ricezione



Efficienza del Selective Repeat

- Assumiamo P_f = frame loss probability
- Il numero di trasmissioni richieste per trasferire una frame è

$$1/(1-P_f)$$

- Il tempo di trasferimento è quindi

$$t_f / (1-P_f)$$

- L'efficienza è data da

$$\eta_{SR} = \frac{\frac{n_f - n_o}{t_f / (1 - P_f)}}{R} = (1 - \frac{n_o}{n_f})(1 - P_f)$$

Esempio: Impatto del BER sul Selective Repeat

- $n_f = 1250$ bytes = 10000 bits, $n_a = n_o = 25$ bytes = 200 bits
- Random bit errors with $p = 0, 10^{-6}, 10^{-5}, 10^{-4}$
- $R = 1$ Mbps, Delay = 100 ms

Efficiency	0	10^{-6}	10^{-5}	10^{-4}
S&W	8.9%	8.8%	8.0%	3.3%
GBN	98%	88.2%	45.4%	4.9%
SR	98%	97%	89%	36%

- Il Selective Repeat ha prestazioni migliori rispetto a GBN e S&W, ma l'efficienza diminuisce al crescere del BER

Confronto tra i metodi ARQ

- Assumiamo n_a e n_o trascurabili rispetto a n_f , e

$$L = 2(t_{prop} + t_{proc})R/n_f = (W_s - 1)$$

Selective-Repeat

$$\eta_{SR} = (1 - P_f)(1 - \frac{n_o}{n_f}) \approx (1 - P_f)$$

Go-Back-N

$$\eta_{GBN} = \frac{1 - P_f}{1 + (W_s - 1)P_f} = \frac{1 - P_f}{1 + LP_f}$$

per $P_f \approx 0$,
SR & GBN uguali

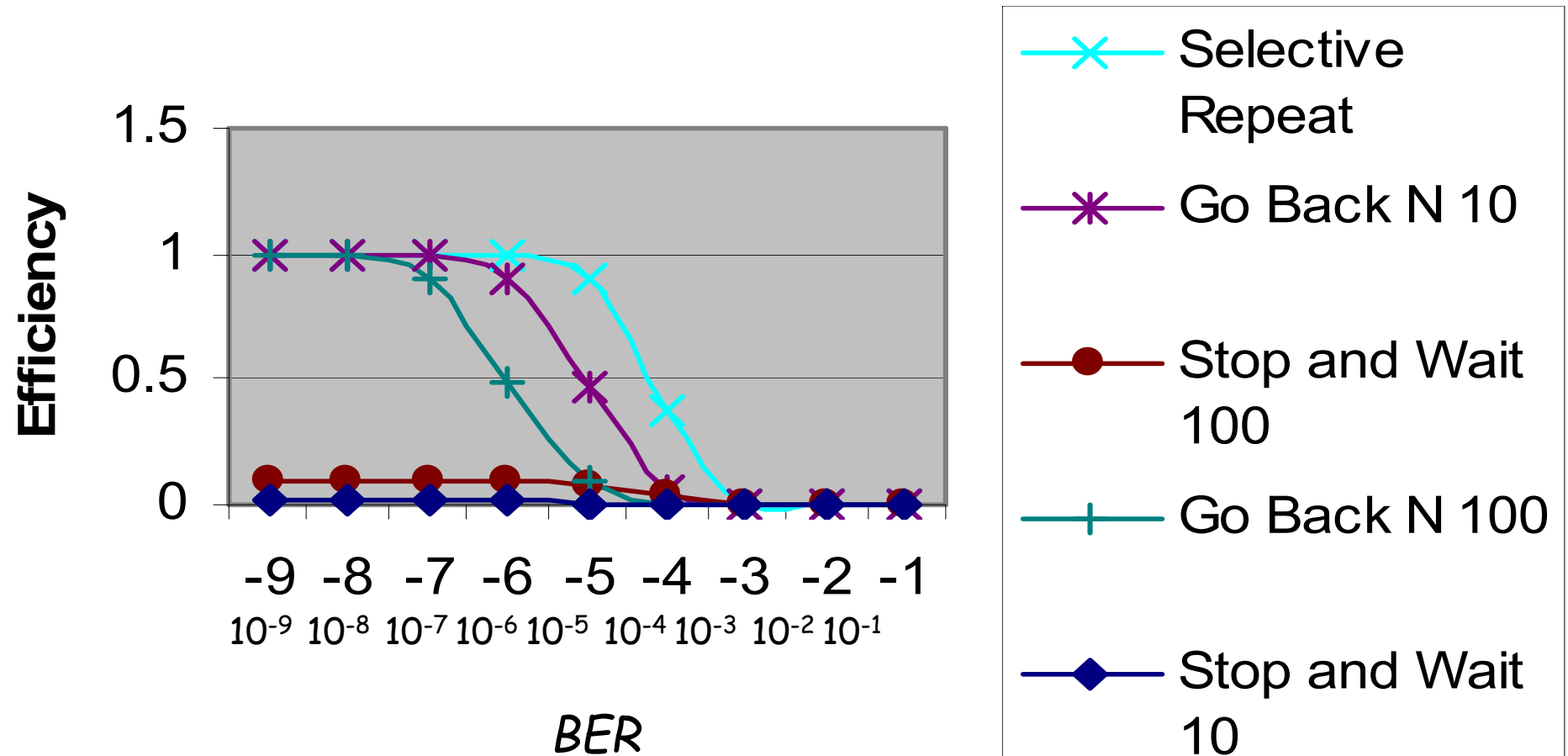
Stop-and-Wait

$$\eta_{SW} = \frac{(1 - P_f)}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}} \approx \frac{1 - P_f}{1 + L}$$

per $P_f \rightarrow 1$,
GBN & SW uguali

Efficienza ARQ

ARQ Efficiency Comparison



Prodotto banda ritardo = 10, 100