

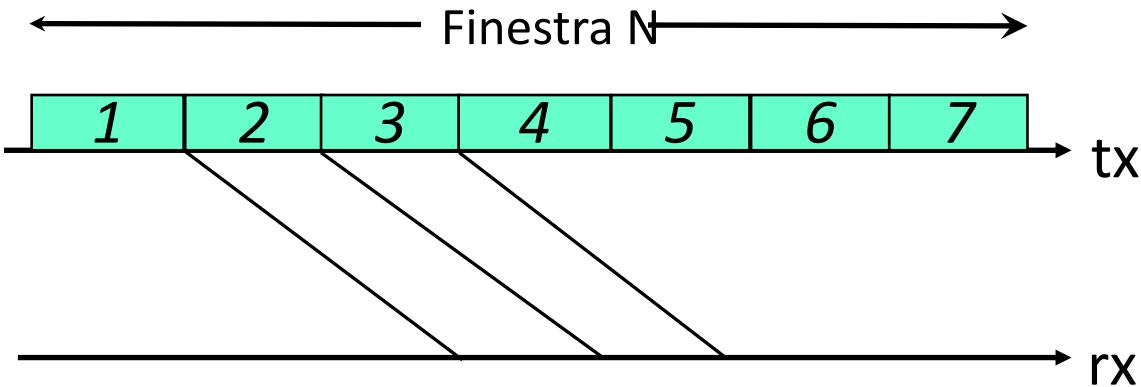


7 – Il Livello di Trasporto

Parte II

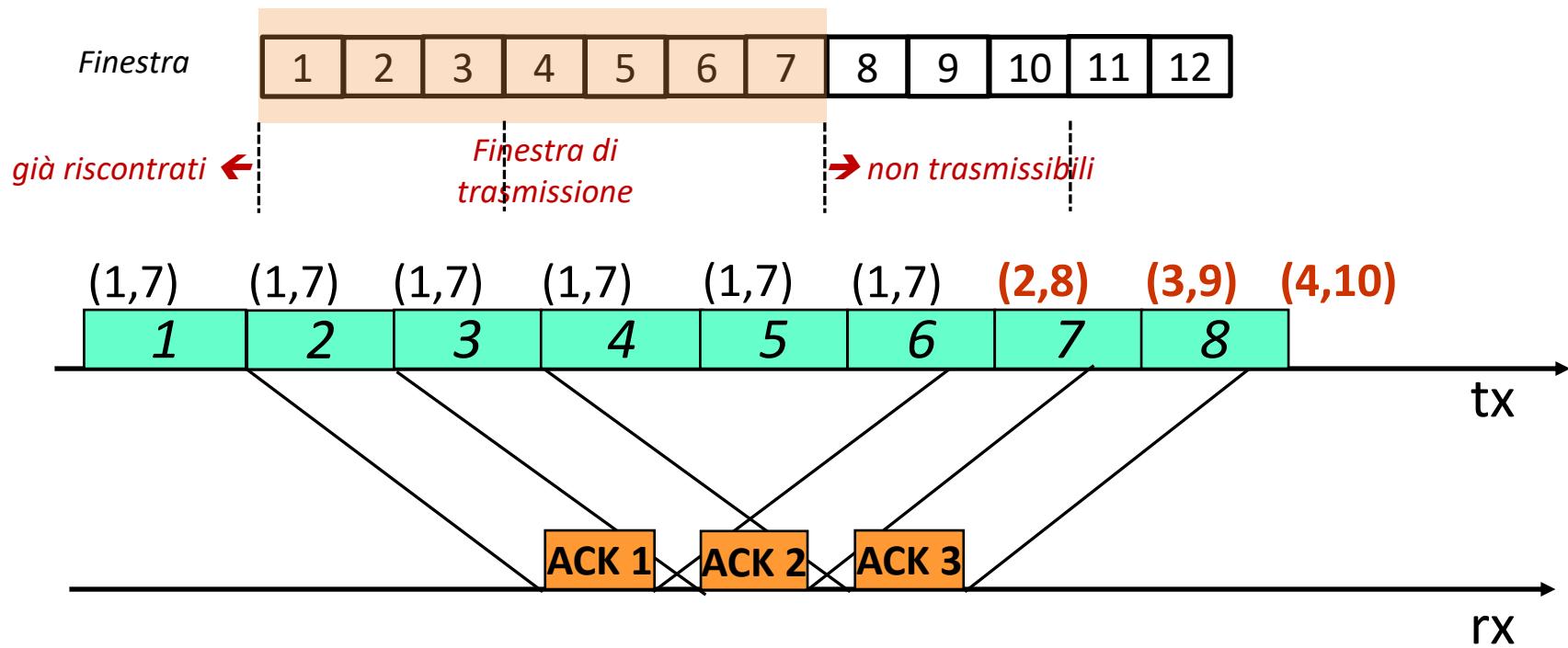
Protocollo Go-back-N

- Variante rispetto allo *Stop and Wait*:
 - Si possono trasmettere fino a N pacchetti (finestra) senza aver avuto il riscontro



Finestra Go-back-N

- Se il riscontro del primo pacchetto arriva prima della fine della finestra, la finestra viene fatta scorrere di una posizione (**sliding window**)

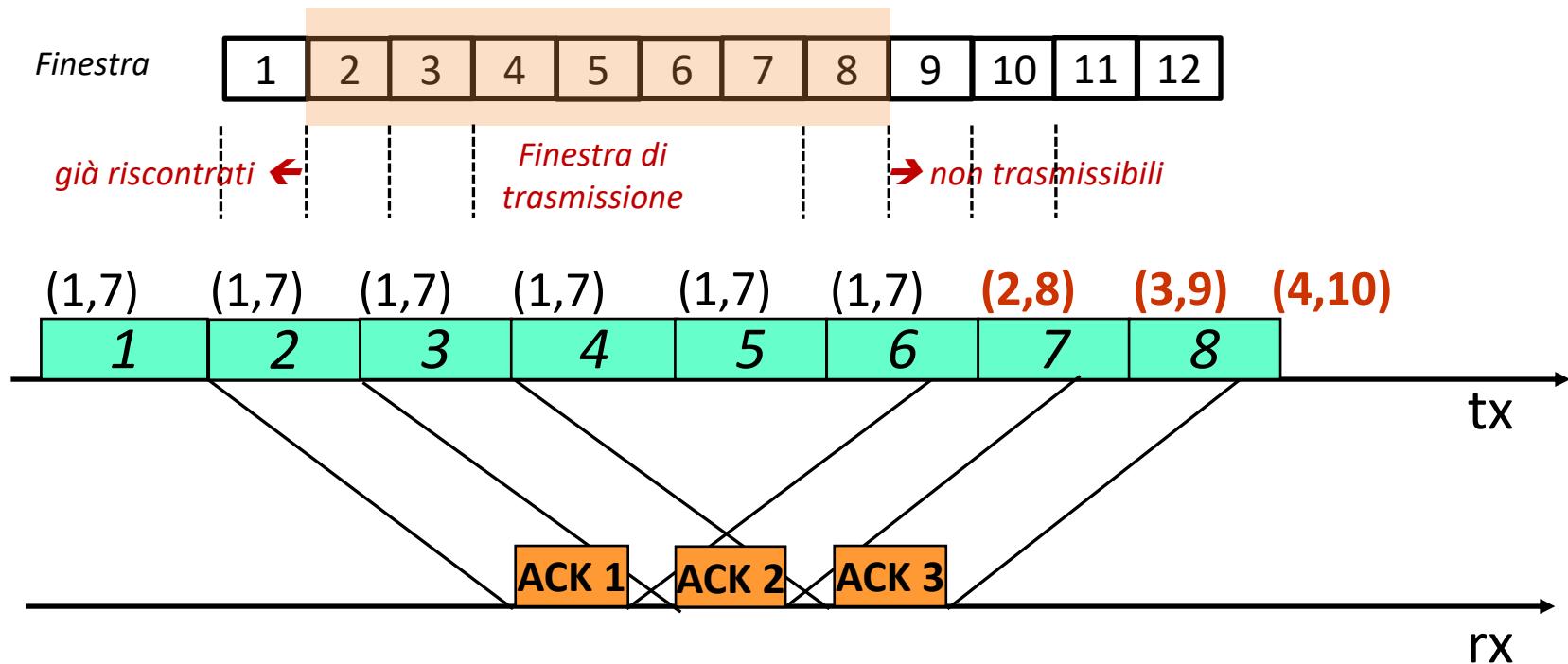


Se non ci sono errori la trasmissione non si ferma mai (efficienza 100%)



Finestra Go-back-N

- Se il riscontro del primo pacchetto arriva prima della fine della finestra, la finestra viene fatta scorrere di una posizione (**sliding window**)

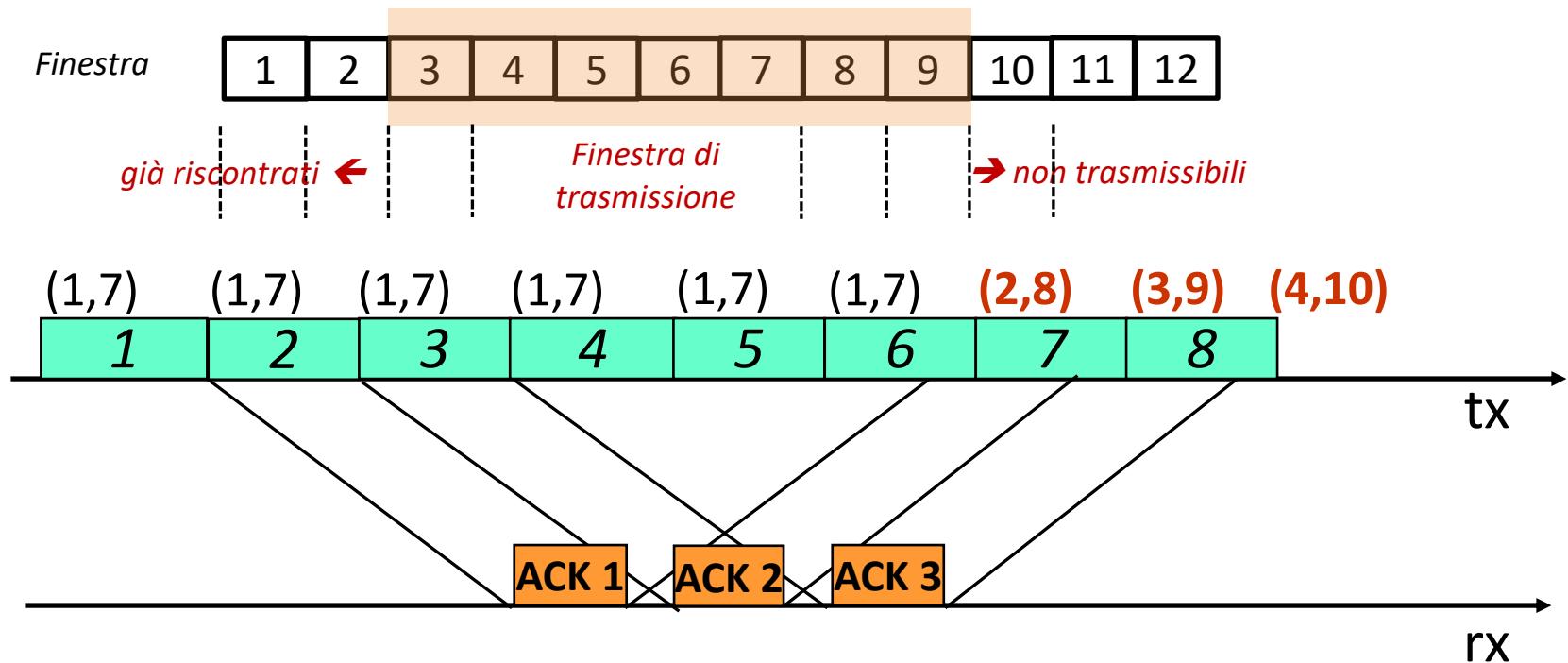


Se non ci sono errori la trasmissione non si ferma mai (efficienza 100%)



Finestra Go-back-N

- Se il riscontro del primo pacchetto arriva prima della fine della finestra, la finestra viene fatta scorrere di una posizione (**sliding window**)

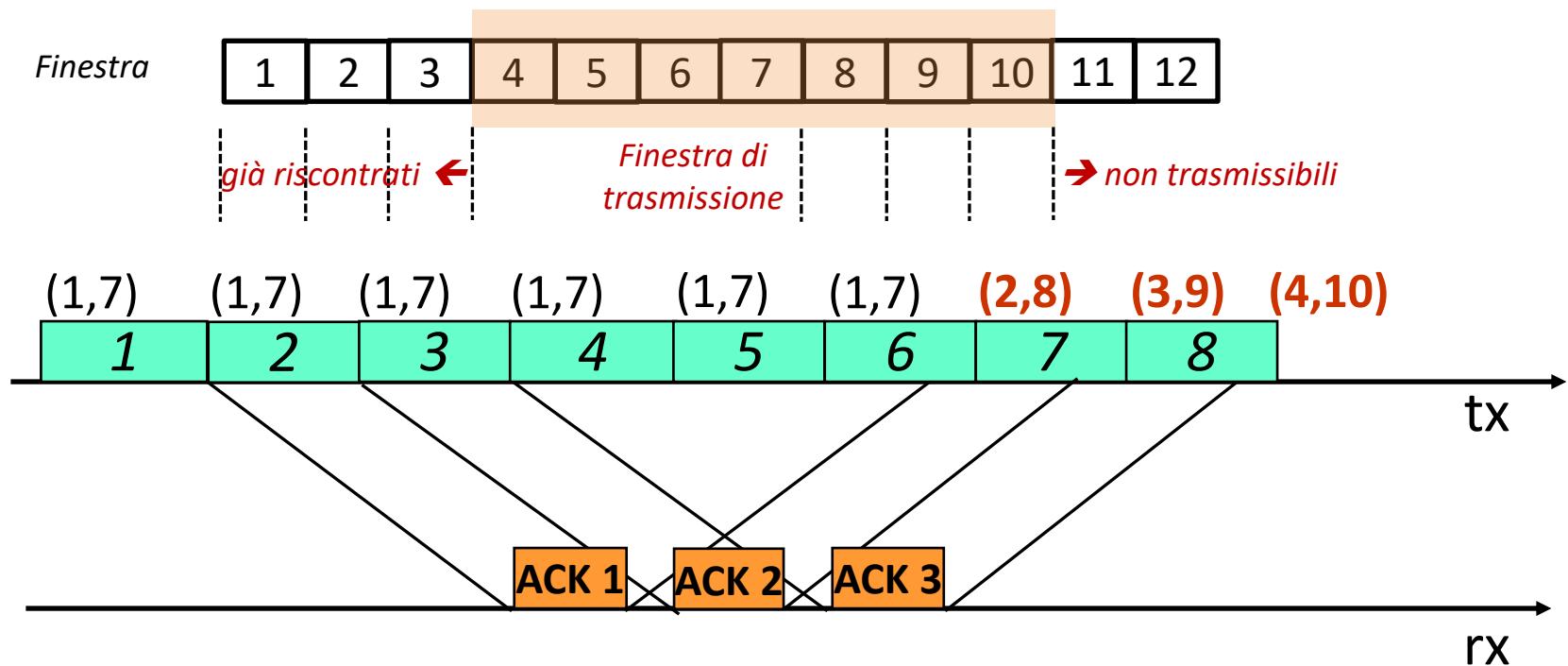


Se non ci sono errori la trasmissione non si ferma mai (efficienza 100%)



Finestra Go-back-N

- Se il riscontro del primo pacchetto arriva prima della fine della finestra, la finestra viene fatta scorrere di una posizione (**sliding window**)

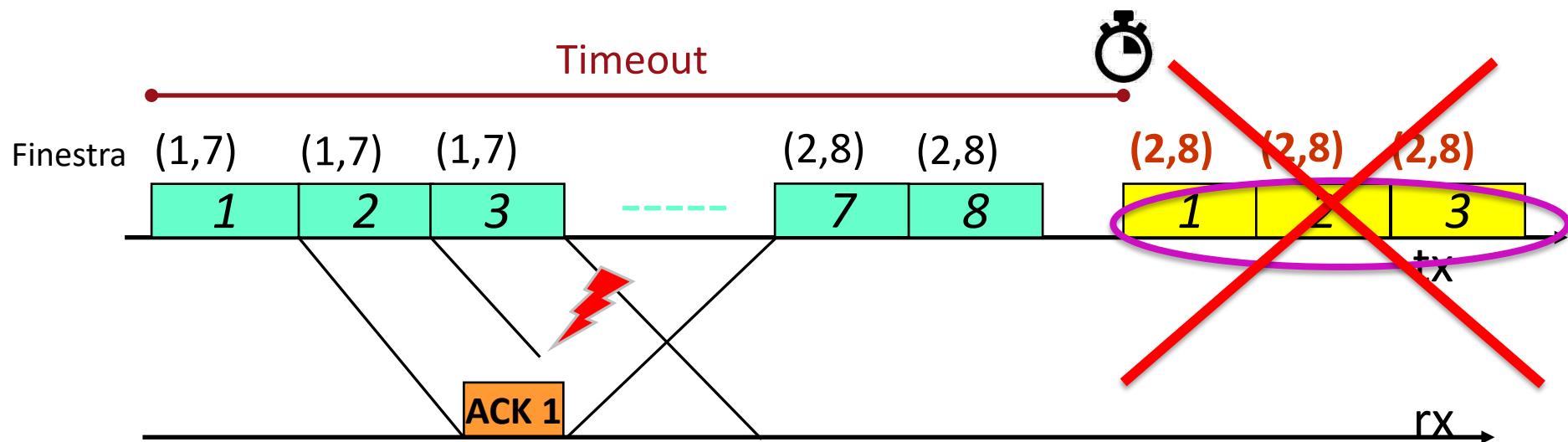


Se non ci sono errori la trasmissione non si ferma mai (efficienza 100%)



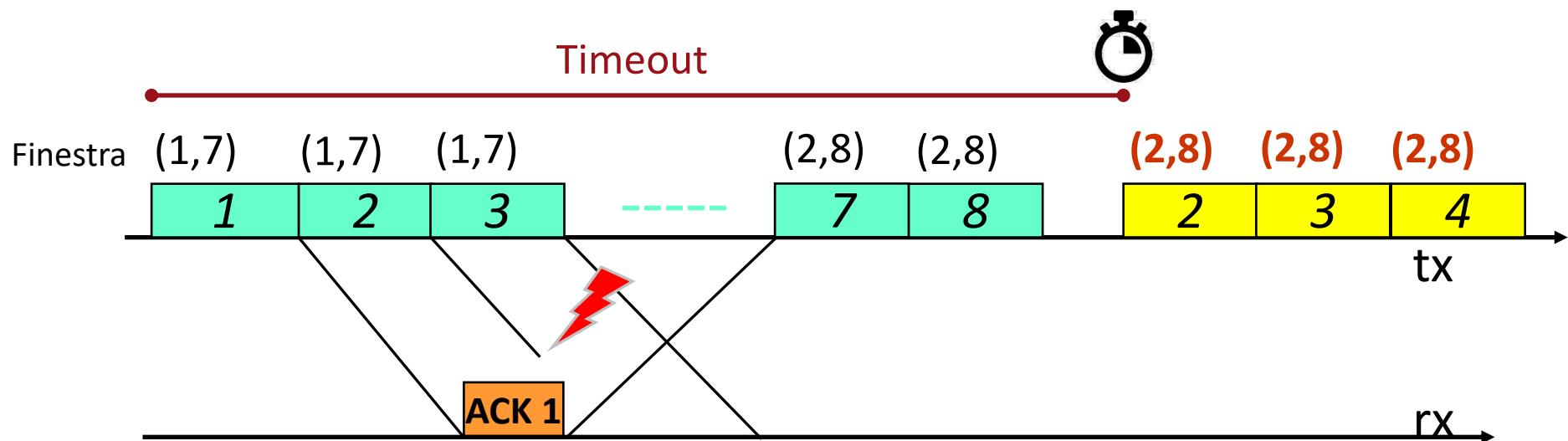
Finestra Go-back-N

- ... altrimenti (quando si verifica un errore): **si ricomincia a trasmettere la finestra dal primo pacchetto non riscontrato**
 - “Torna indietro di N pacchetti”
- il *time out* ha lo stesso significato dello Stop&Wait
 - Raggiunto l’ultimo pacchetto della finestra, la trasmissione si blocca in attesa di un nuovo ACK o della scadenza del timeout
 - La ritrasmissione del primo pacchetto non riscontrato inizia allo scadere del timeout



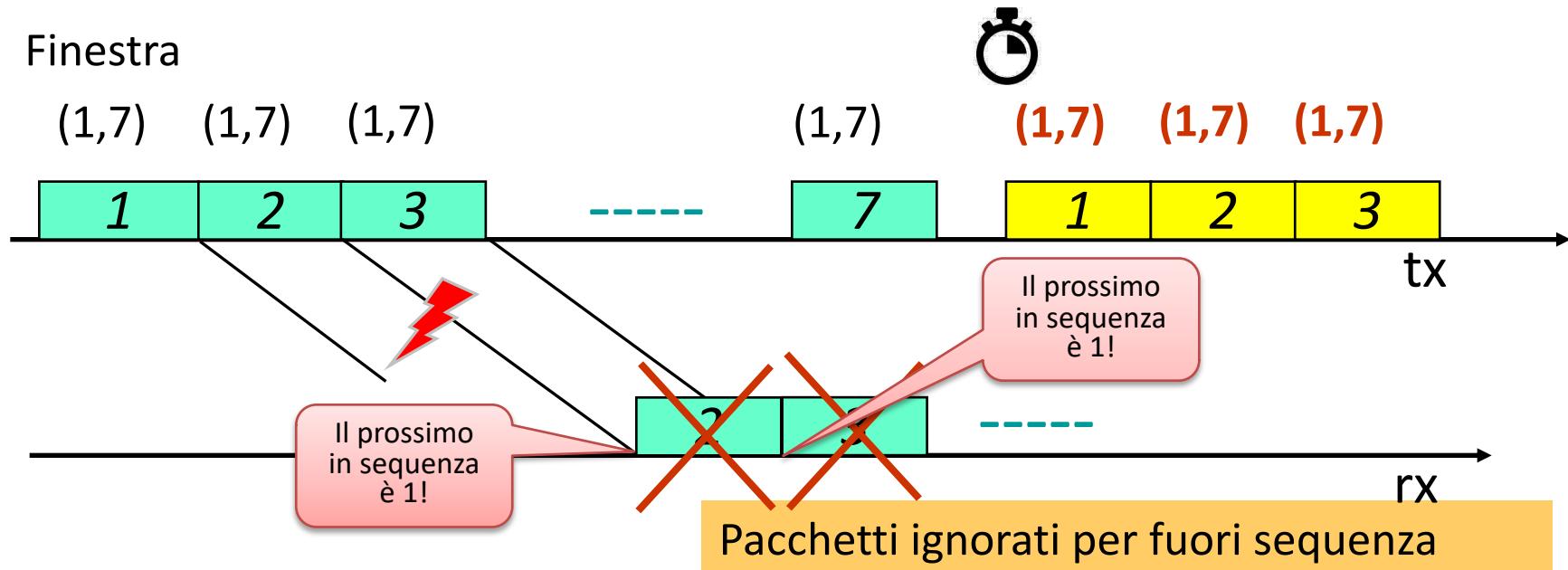
Finestra Go-back-N

- ... altrimenti (quando si verifica un errore): **si ricomincia a trasmettere la finestra dal primo pacchetto non riscontrato**
 - “Torna indietro di N pacchetti”
- il *time out* ha lo stesso significato dello Stop&Wait
 - Raggiunto l’ultimo pacchetto della finestra, la trasmissione si blocca in attesa di un nuovo ACK o della scadenza del timeout
 - La ritrasmissione del primo pacchetto non riscontrato inizia allo scadere del timeout



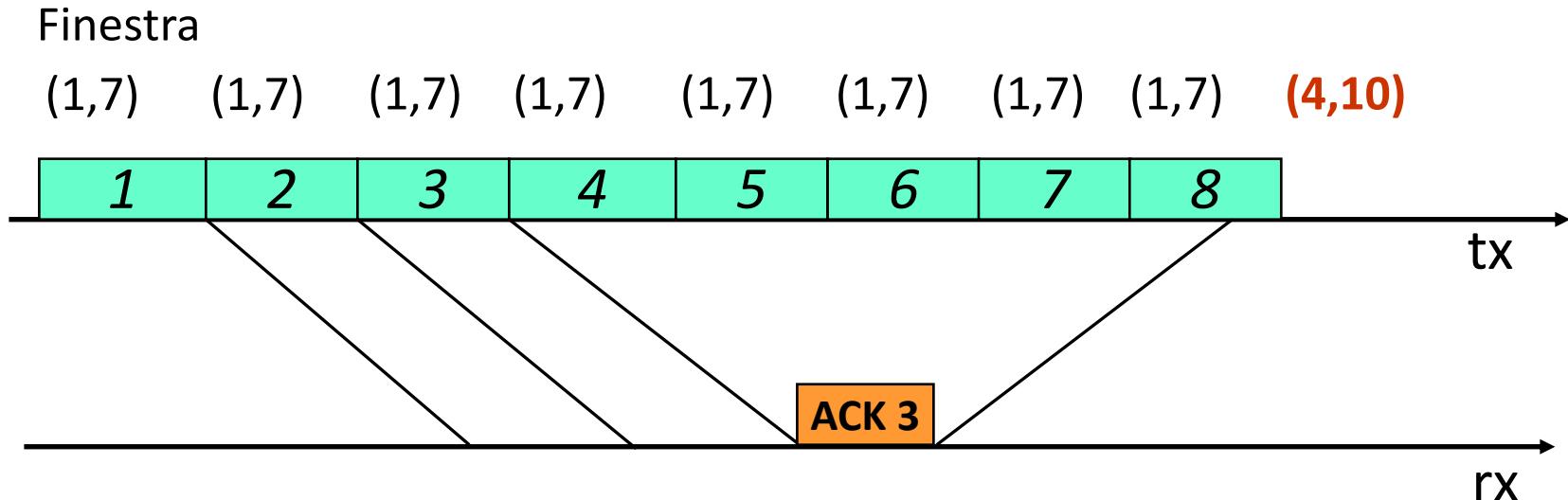
Finestra Go-back-N

- Ciò può causare la ritrasmissione di pacchetti corretti, ma semplifica il funzionamento perché
- .. permette al ricevitore di ignorare le ricezioni fuori sequenza (l'ordine è mantenuto automaticamente)



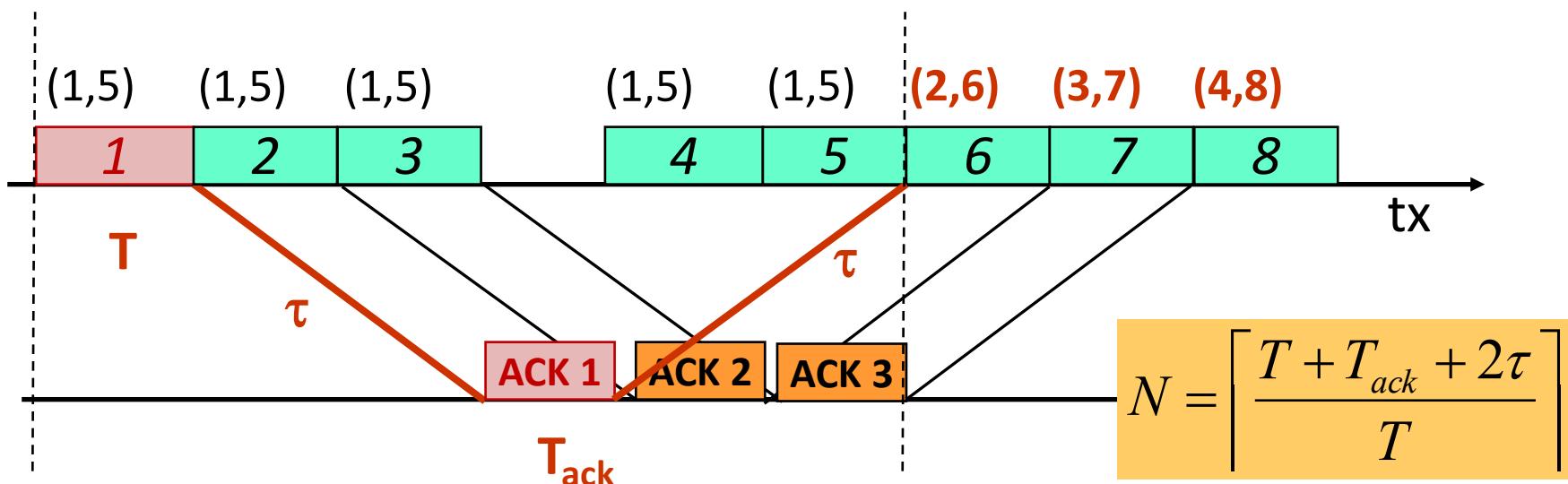
Go-back-N: ACK collettivo

- Escludendo il fuori sequenza, il riscontro (ACK) può essere collettivo
- Questo, entro certi limiti, rimedia alla perdita di ACK



Dimensionamento finestra

- La finestra ottima coincide con il **Round Trip Time**
 - Trasmissione pacchetto, propagazione tx-rx,
Trasmissione ACK e propagazione rx-tx
 - La finestra aumenta di un pacchetto alla volta e la trasmisione non si interrompe mai



Dimensionamento finestra

- La finestra può anche essere dimensionata in tempo, in byte,
- Il dimensionamento si complica se i tempi di propagazione non sono noti
 - es. con Go-back-N a livello di trasporto i tempi di attraversamento della rete (τ) possono variare col tempo..
 - e/o i pacchetti sono di lunghezza variabile
- Rimedi
 - Fare la finestra grande
 - Non pregiudica il funzionamento in assenza d'errore
 - Ma in caso d'errore, aumenta il ritardo con cui si scopre ed il numero di ritrasmissioni inutili
 - Uso del NACK
 - Stimare il tempo di RTT e adattare la finestra o il timeout

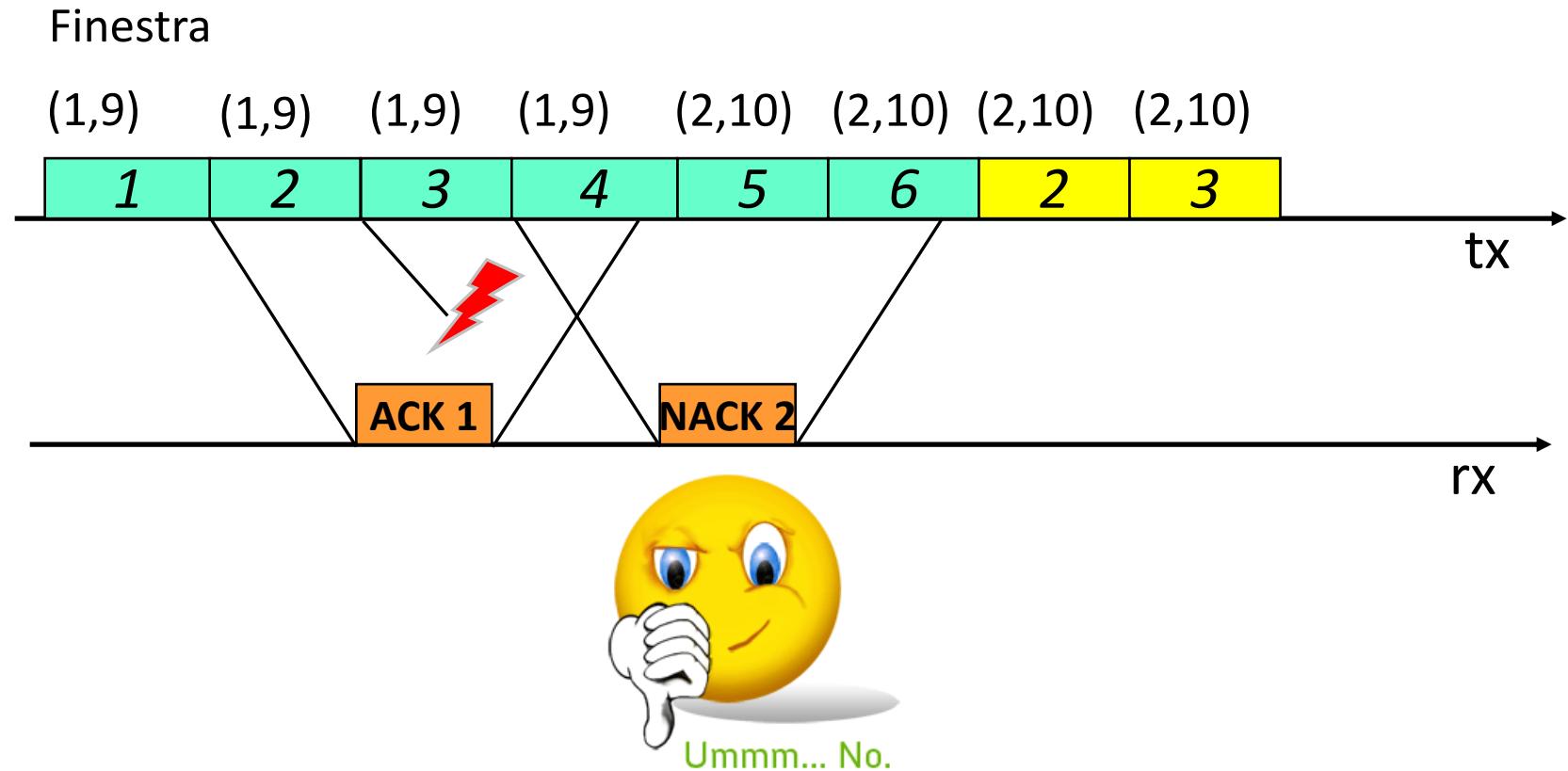


Uso del NACK

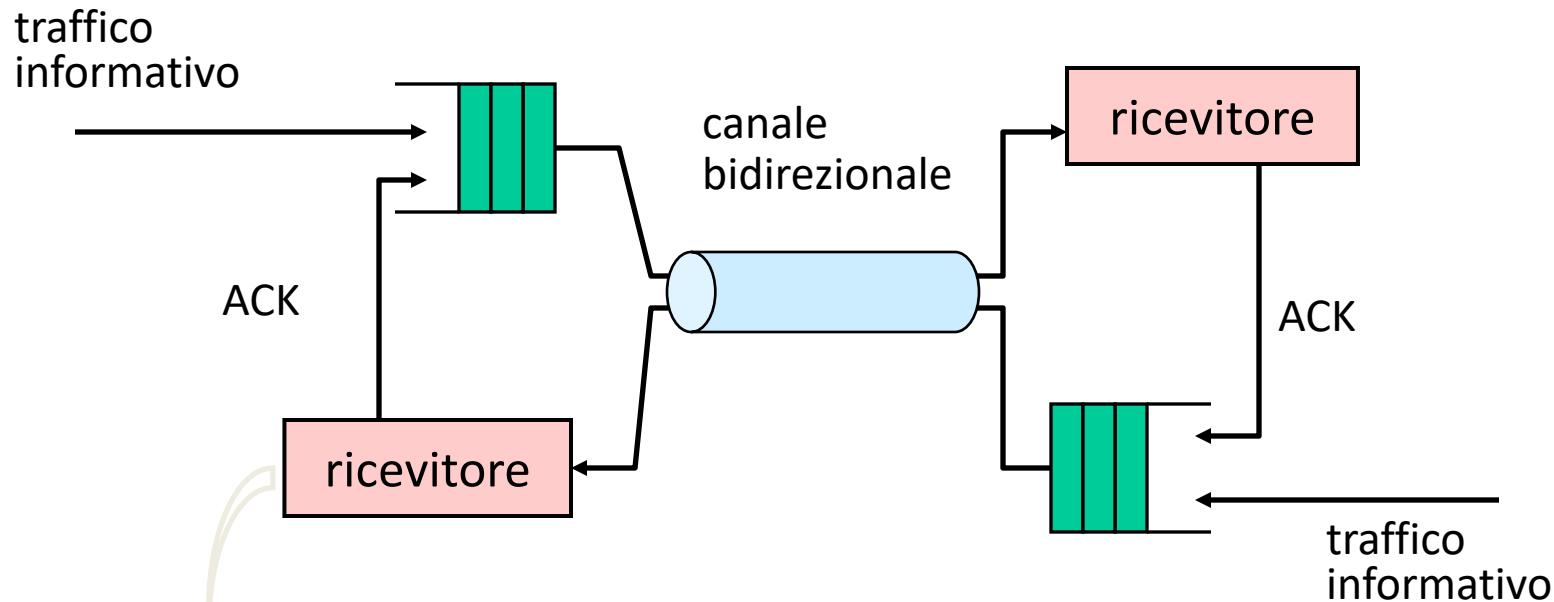
- L'uso del NACK può abbreviare i tempi di ritrasmissione in caso d'errore, evitando di aspettare la fine della finestra
- Non è possibile inviare immediatamente il NACK perché il ricevitore dovrebbe conoscere il SN del pacchetto errato, ma se e' errata ...
- Se arriva un pacchetto fuori sequenza, posso ipotizzare che sia andata perso il pacchetto precedente
- Non è possibile applicarlo nei meccanismi in cui non è garantita la consegna in ordine (livelli superiori al livello di link)



Uso del NACK

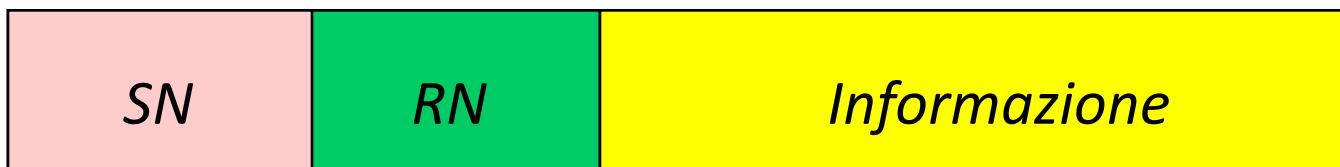


Go-back-N full-duplex



Go-back-N e piggy backing

- Gli ACK possono anche essere inseriti negli *header* dei pacchetti che viaggiano in direzione opposta (*Piggy-backing*)



- *SN*: numero di sequenza del pacchetto trasmesso (canale diretto)
- *RN*: numero di sequenza delle pacchette atteso in direzione opposta, vale come riscontro cumulativo dei pacchetti fino a *RN-1*



Regole Go-back-N

- Trasmettitore:
 - N : dimensione finestra
 - N_{last} : ultimo riscontro ricevuto
 - N_C : numero corrente disponibile per pacchetto in trasmissione
 - **Regole:**
 - Ogni nuovo pacchetto viene accettato per la trasmissione solo se $N_C < N_{last} + N$, altrimenti viene messo in attesa; se $N_C < N_{last} + N$ il pacchetto viene trasmesso con SN pari a N_C , viene inizializzato il timer di timeout e il valore di N_C viene incrementato di uno ($N_C := N_C + 1$);
 - Ad ogni riscontro RN ricevuto, si pone $N_{last} = RN$;
 - I pacchetti nella finestra vengono trasmessi senza vincoli di temporizzazione;
 - In caso di scadenza di *timeout* la ritrasmissione deve ripartire dal pacchetto N_{last}
 - Intervallo N_{last} e $(N_{last} + N - 1)$: finestra di trasmissione

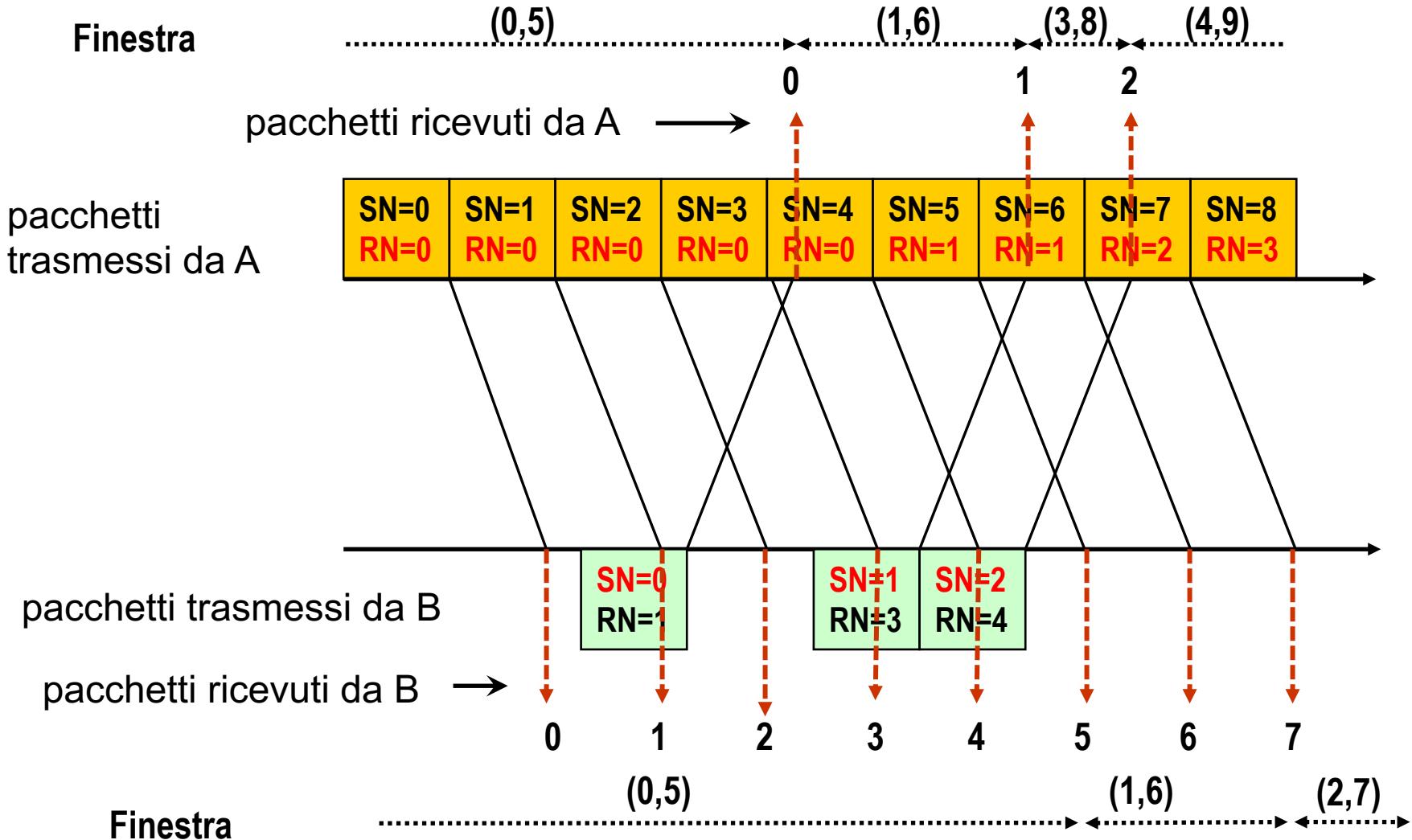


Regole Go-back-N

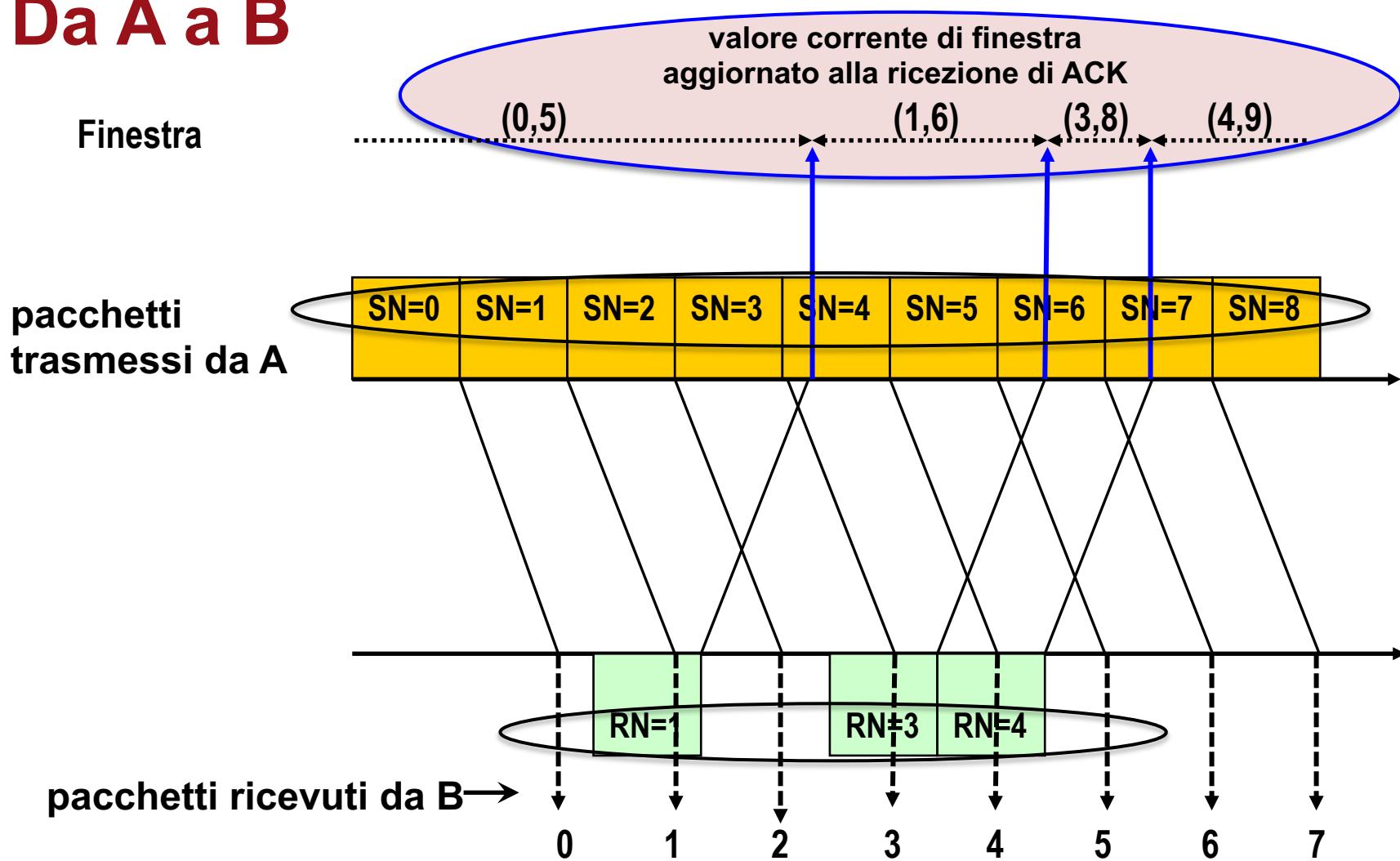
- Ricevitore:
 - *RN: stato dei riscontri corrente (SN che il ricevitore si aspetta di ricevere)*
 - **Regole:**
 - Se viene ricevuto correttamente un pacchetto con $SN=RN$, questo viene inoltrato ai livelli superiori e si pone $RN:=RN+1$;
 - Ad istanti arbitrari ma con ritardo finito, RN viene trasmesso al mittente utilizzando i pacchetti in direzione opposta.



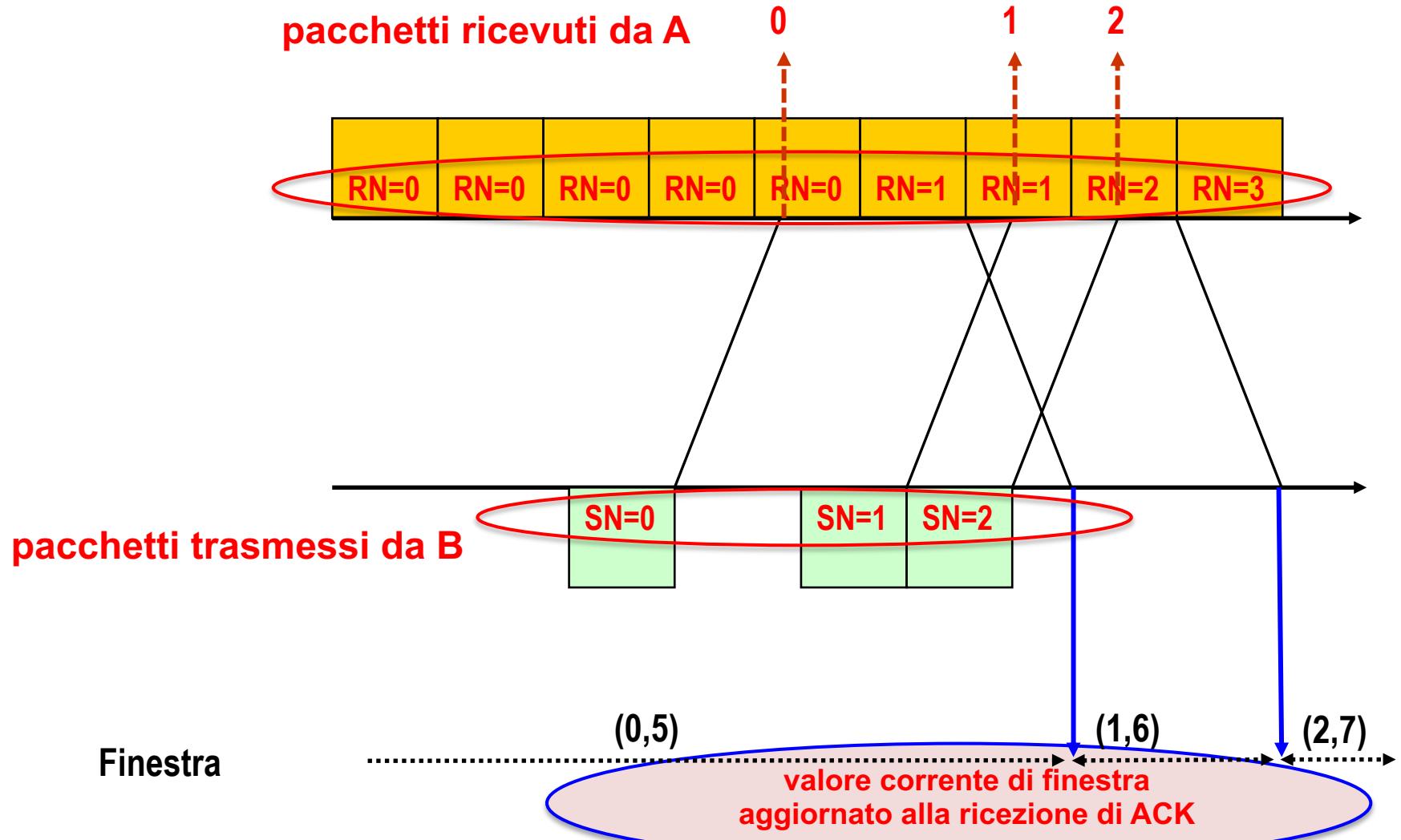
Go-back-N full duplex senza errori



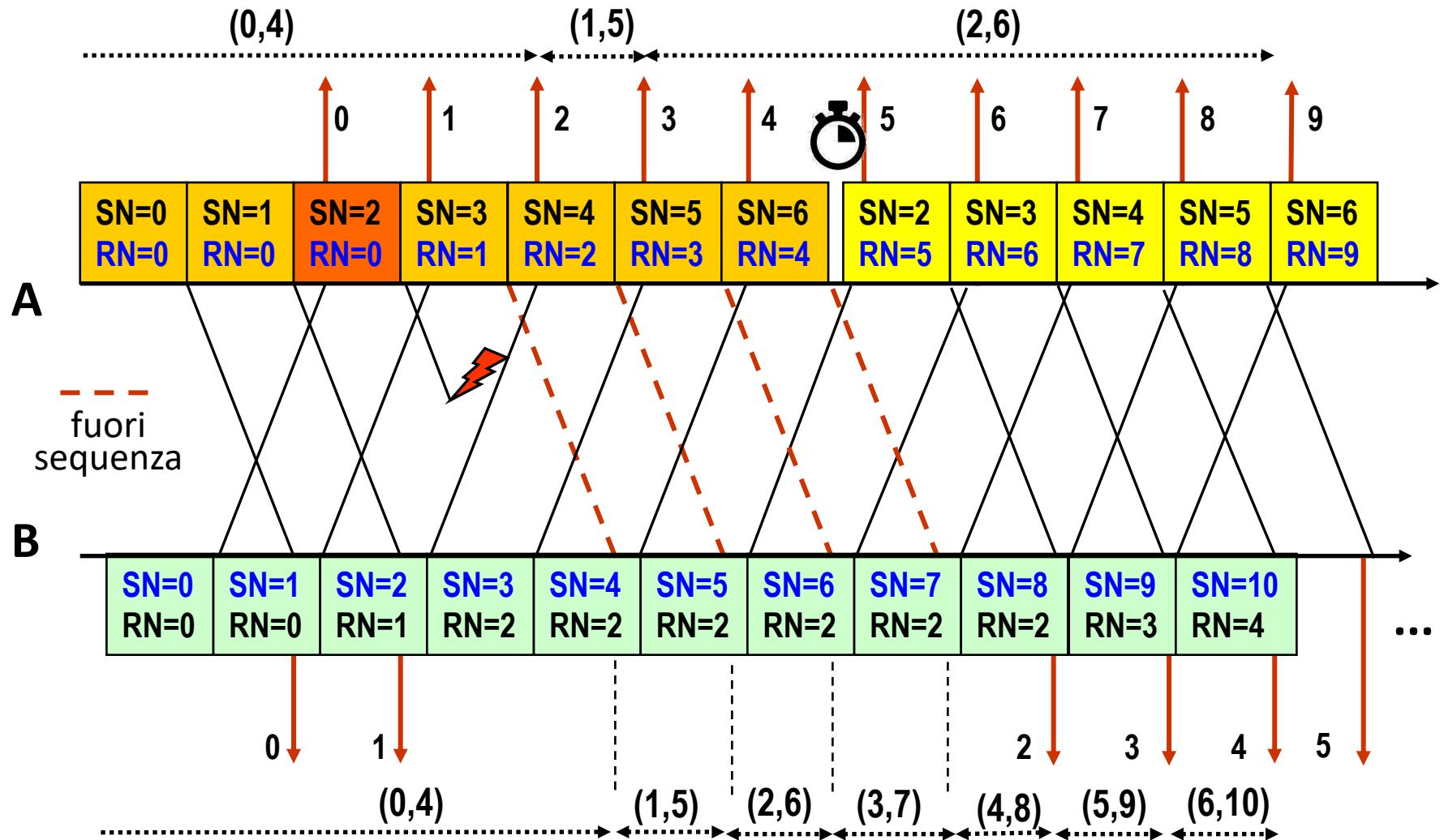
Da A a B



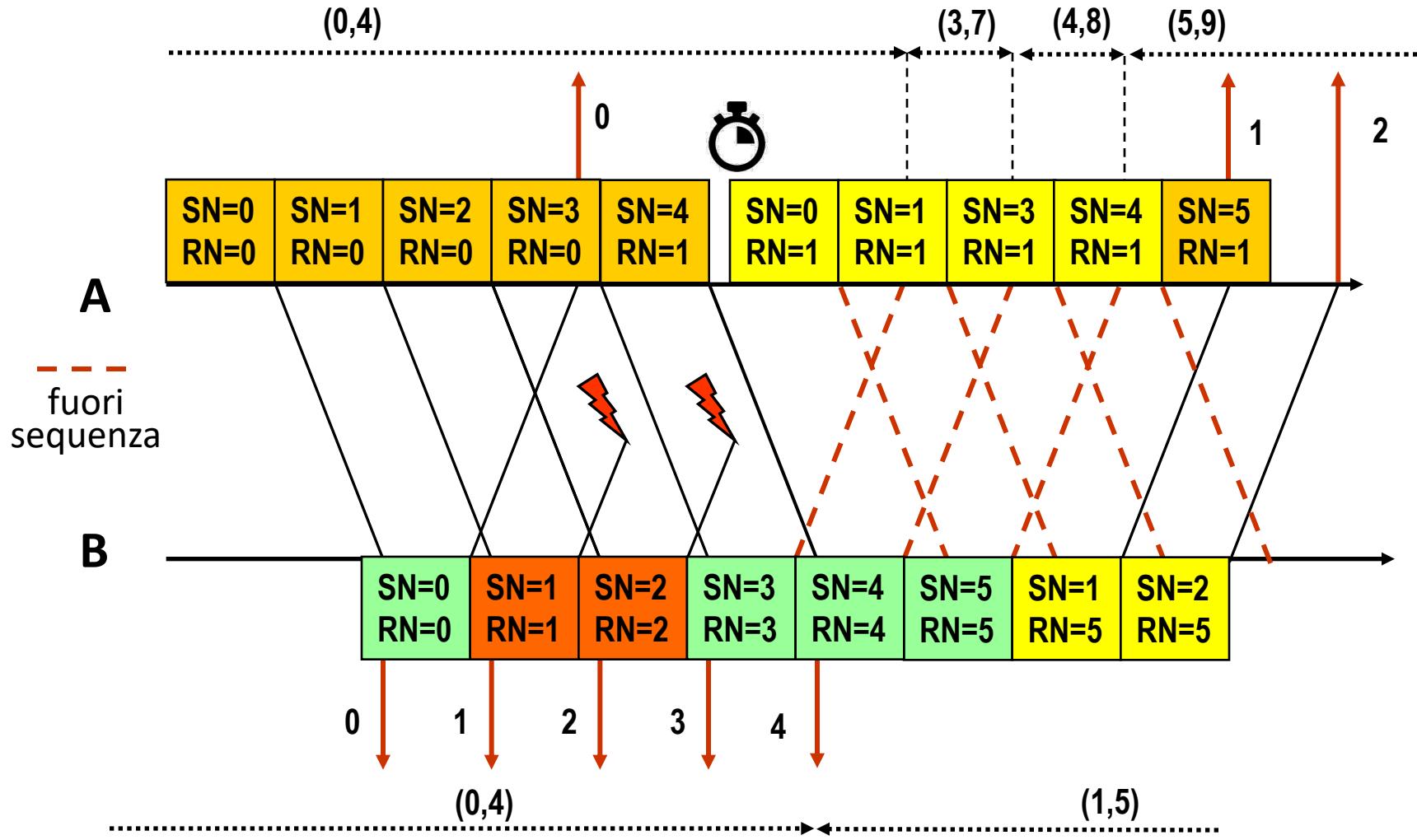
Da B a A



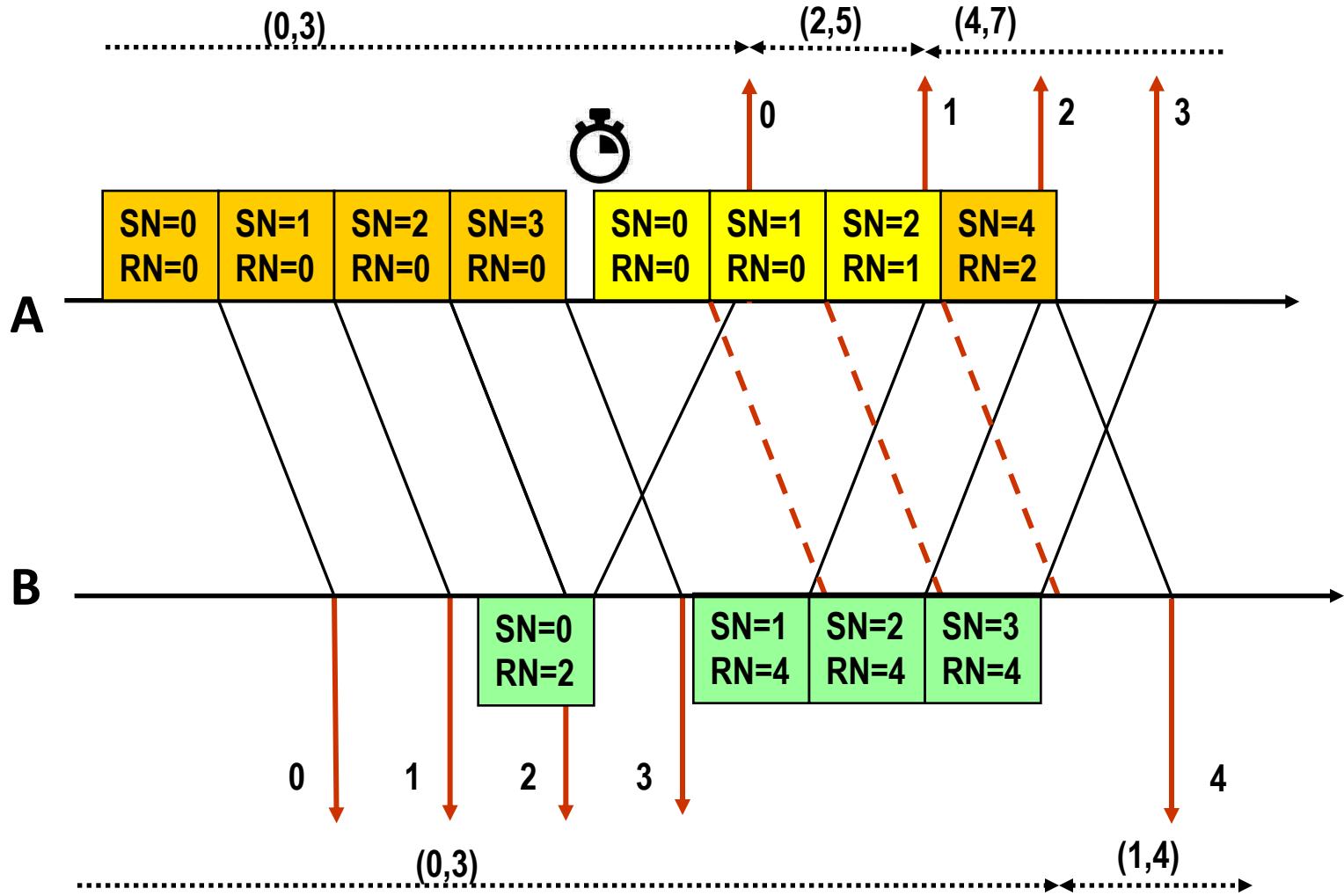
Go-back-N full duplex con errore



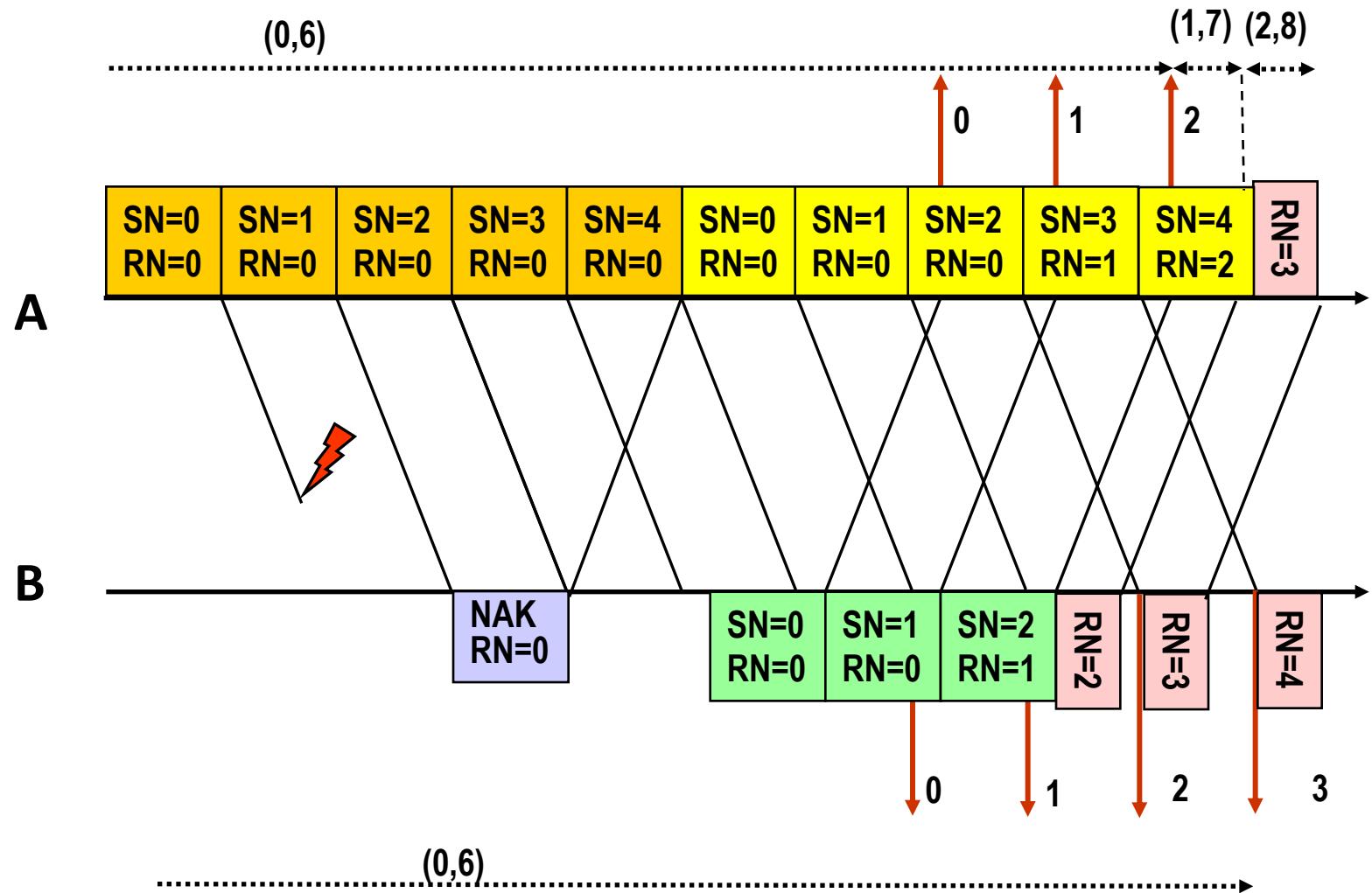
Go-back-N full duplex con errore



Go-back-N full con ACK ritardati



Go-back-N con NAK



Osservazioni sul controllo d'errore

- Necessità di inizializzare il protocollo
 - I numeri SN e RN devono essere inizializzati
 - Deve esistere un momento di inizio non equivocabile in cui scambiare l'informazione per l'inizializzazione

Occorre un meccanismo a connessione che stabilisca l'istante $t=0$



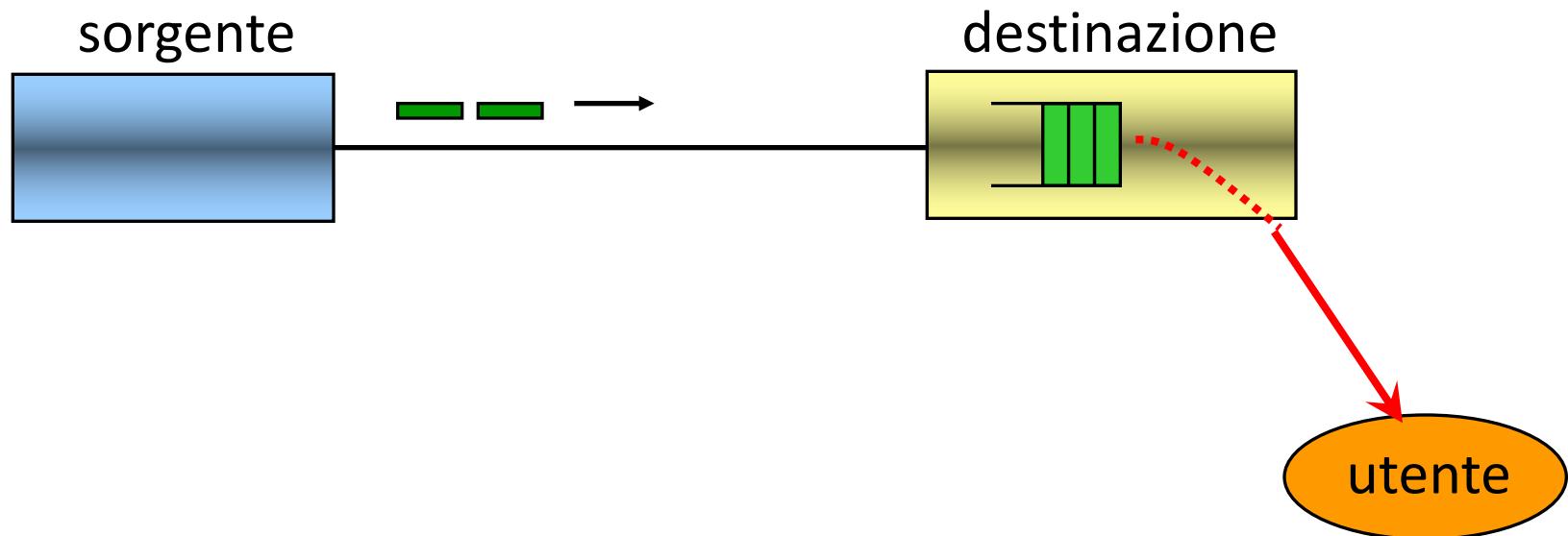
Livello di Trasporto

- Introduzione
- Protocollo UDP
- Trasporto affidabile
 - Protocolli di ritrasmissione
 - Controllo di flusso a finestra mobile
- Protocollo TCP
 - Generalità
 - Formato
 - Controllo d'errore
 - Controllo di congestione



Controllo di flusso

- Buffer di ricezione limitato a **W** posizioni
- Ritmo di assorbimento dell'utente arbitrario
- Obiettivo: regolando il ritmo di invio, evitare che pacchetti vadano persi perché all'arrivo trovano il buffer pieno

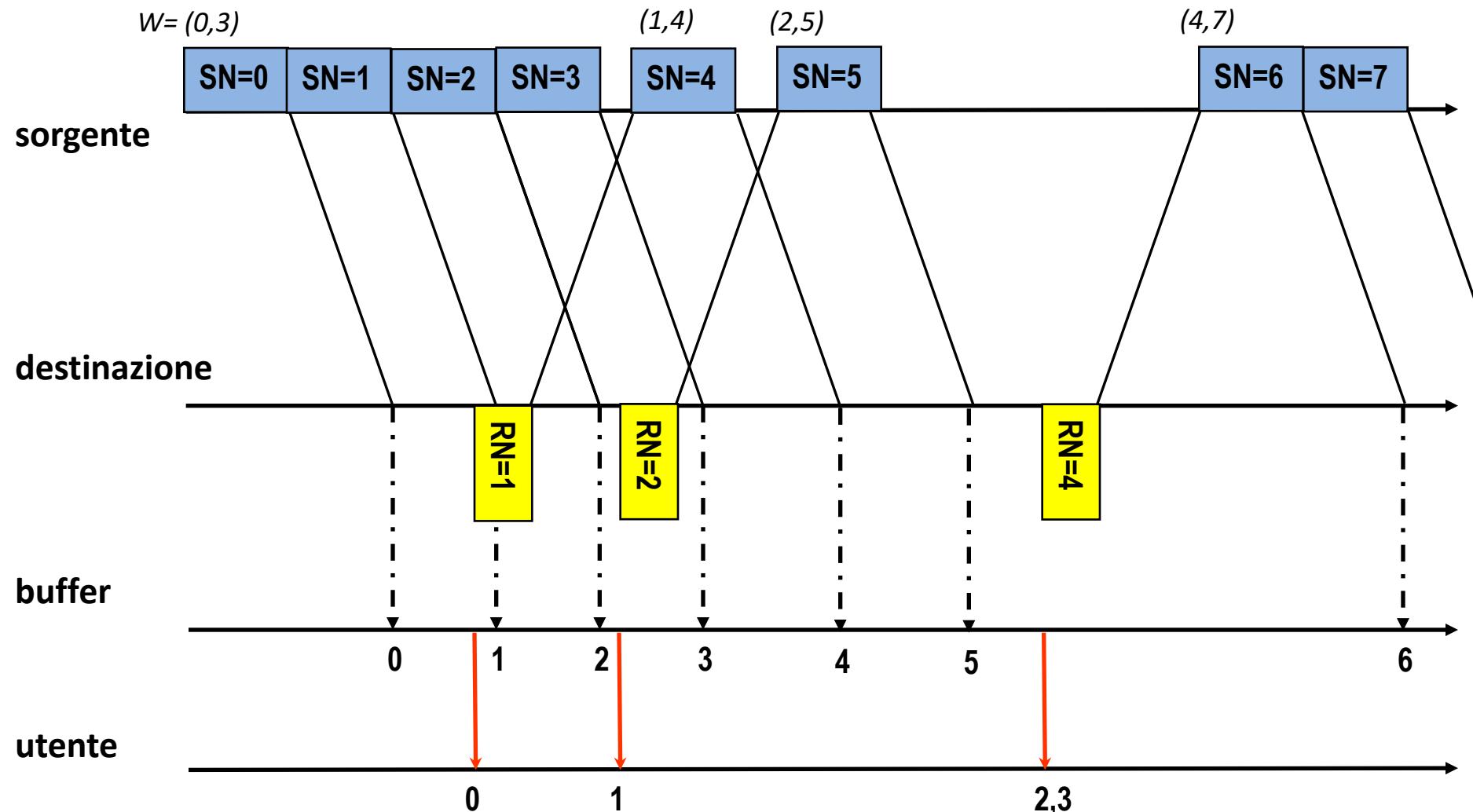


Sliding window flow control

- Controllo di flusso a finestra mobile
- E' possibile usare un meccanismo come quello del *Go-back-N*
- La sorgente non può inviare più di W trame (stessa funzione del parametro N) senza aver ricevuto il riscontro
- I riscontri vengono inviati dal ricevitore solo quando i pacchetti vengono letti (**tolti dal buffer**) dal livello superiore



Sliding-window flow control



Problema delle ritrasmissioni

- Il meccanismo di controllo di flusso descritto nelle slide precedenti è strettamente legato al meccanismo di controllo d'errore e questo può essere fonte di problemi
- Se il ricevitore ritarda molto l'invio dei riscontri a causa del livello superiore lento, il trasmettitore inizia la ritrasmissione perché scade il time-out
- Aumentare troppo il time-out non è ovviamente una soluzione in quanto l'aumento del time-out aumenta i ritardi in caso di errore

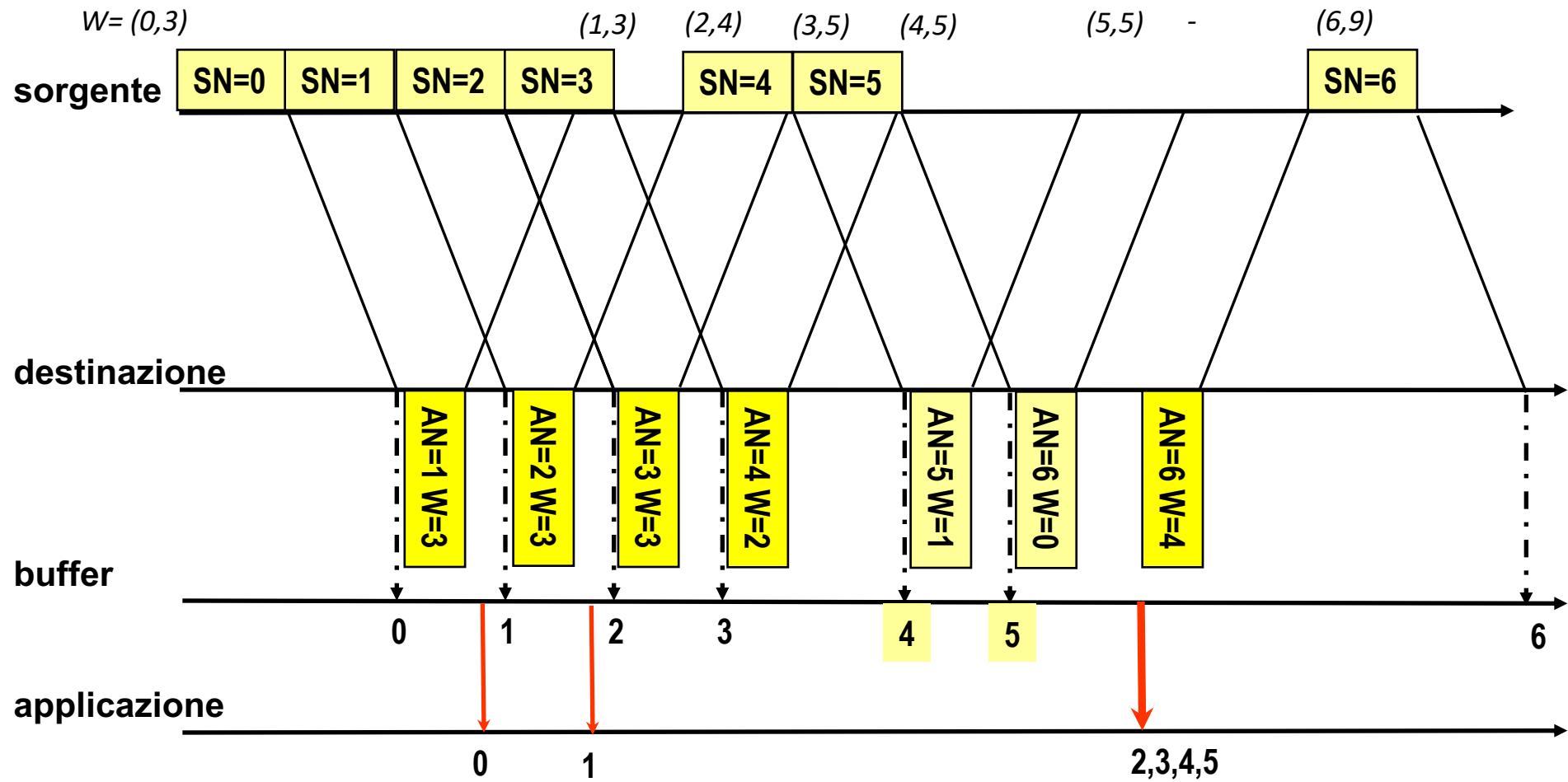


Uso del campo W

- Il problema può essere risolto in modo radicale separando i meccanismi di controllo d'errore e di controllo di flusso a finestra
- Si inserisce nei riscontri (o nell'header delle trame in direzione opposta) un campo finestra W (insieme a quella del GBN)
 - Il ricevitore invia i riscontri sulla base dell'arrivo dei pacchetti
 - E usa il campo W per indicare lo spazio rimanente nel buffer



Uso del campo W



Uso del campo W

- Gestione della finestra del controllo di flusso
 - Non è necessario che il ricevitore dica la “verità” sullo spazio restante R
 - Può tenersi un margine di sicurezza ($W=R-m$)
 - Può aspettare che il buffer si sia svuotato per una frazione (ad es. $W=0$ se $R < K/2$ ed $W=R$ altrimenti)
 - Può usare dei meccanismi adattativi

