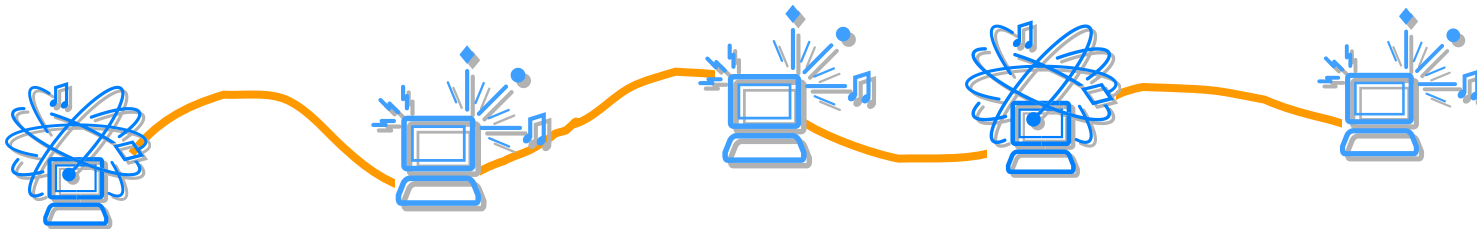




# Protocolos de retransmissão

---



Instituto Superior de Engenharia de Lisboa  
Departamento de Engenharia de Electrónica e Telecomunicações e de  
Computadores

*Redes de Computadores*

---



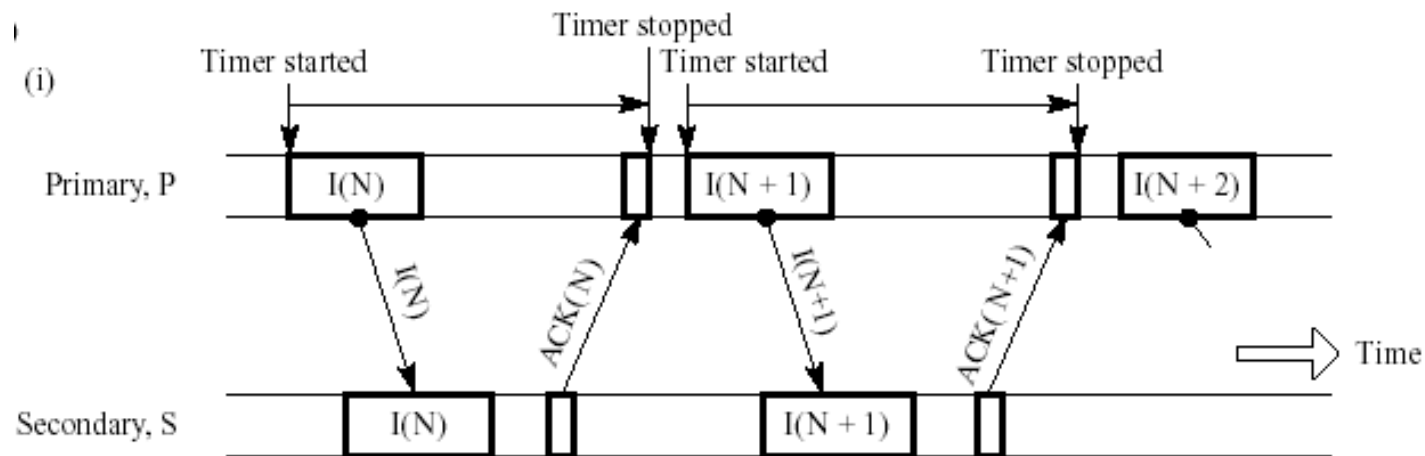
- Problemas
  - Controlo de fluxo
  - Controlo de erros, duplicações e sequência
- Comunicação humano – computador central
  - *Echo Checking*
    - Controlo manual
- Comunicação computador - computador
  - ARQ
    - Controlo automático



- ARQ - **A**utomatic **R**epeat re**Q**uest
  - Idle RQ
    - Send and Wait (Stop and Wait)
  - Continuous RQ
    - Go Back N
    - Selective Repeat
- Métodos de ARQ
  - Pedido de retransmissão implícito
  - Pedido de retransmissão explícito



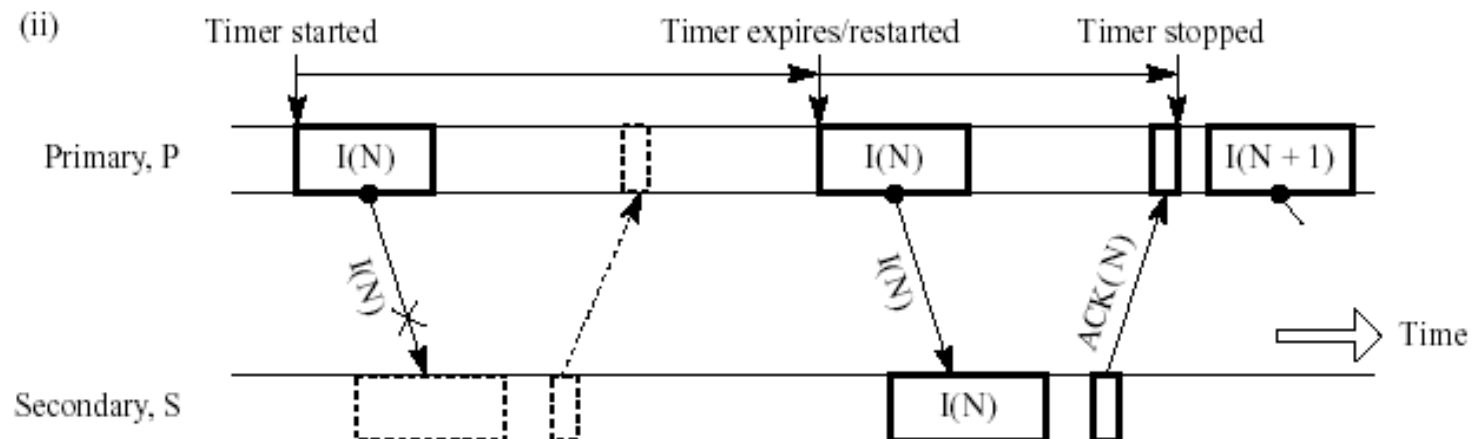
- Pedido de retransmissão implícito ou pedido de retransmissão explícito
  - Funcionamento normal



# Idle RQ: Send and Wait



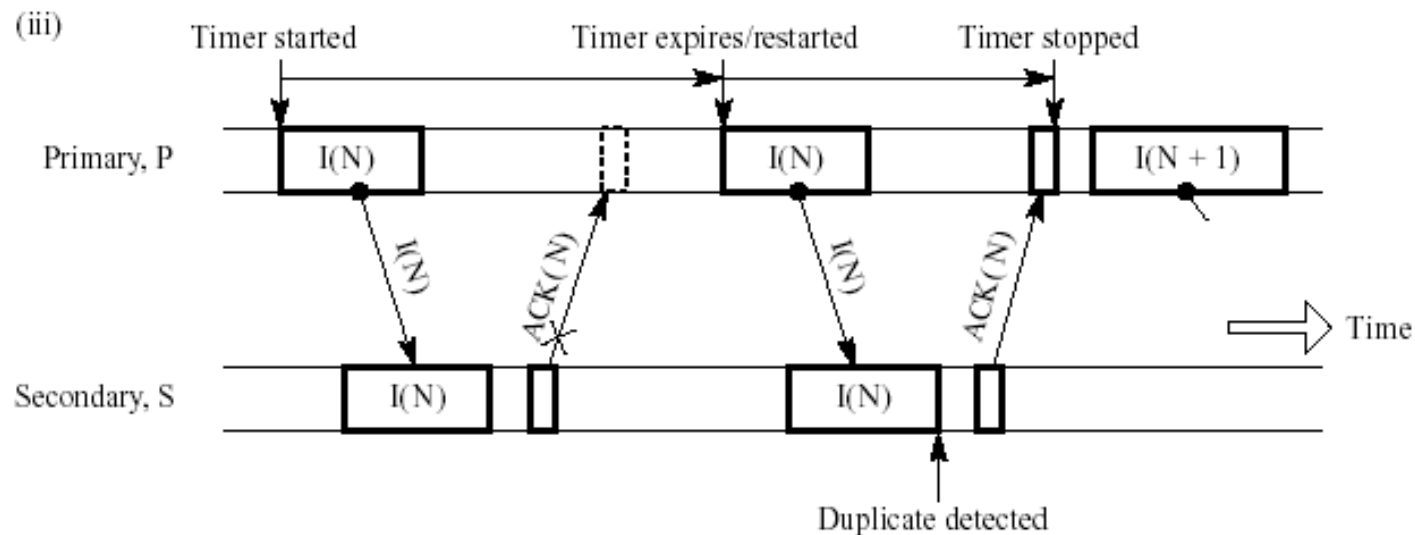
- Pedido de retransmissão implícito
  - Trama de dados corrompida



# Idle RQ: Send and Wait



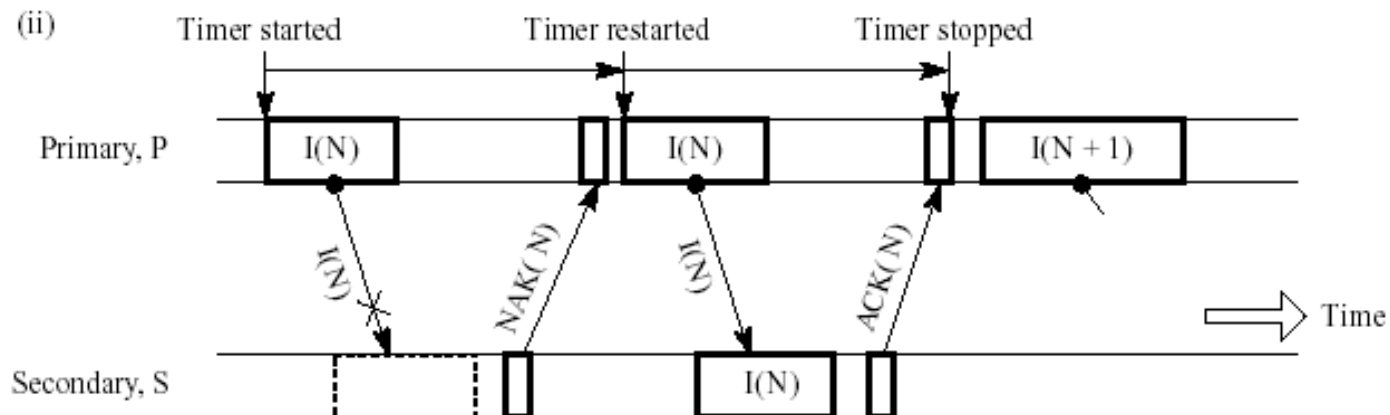
- Pedido de retransmissão implícito
  - Trama de ACK ou NAK corrompida



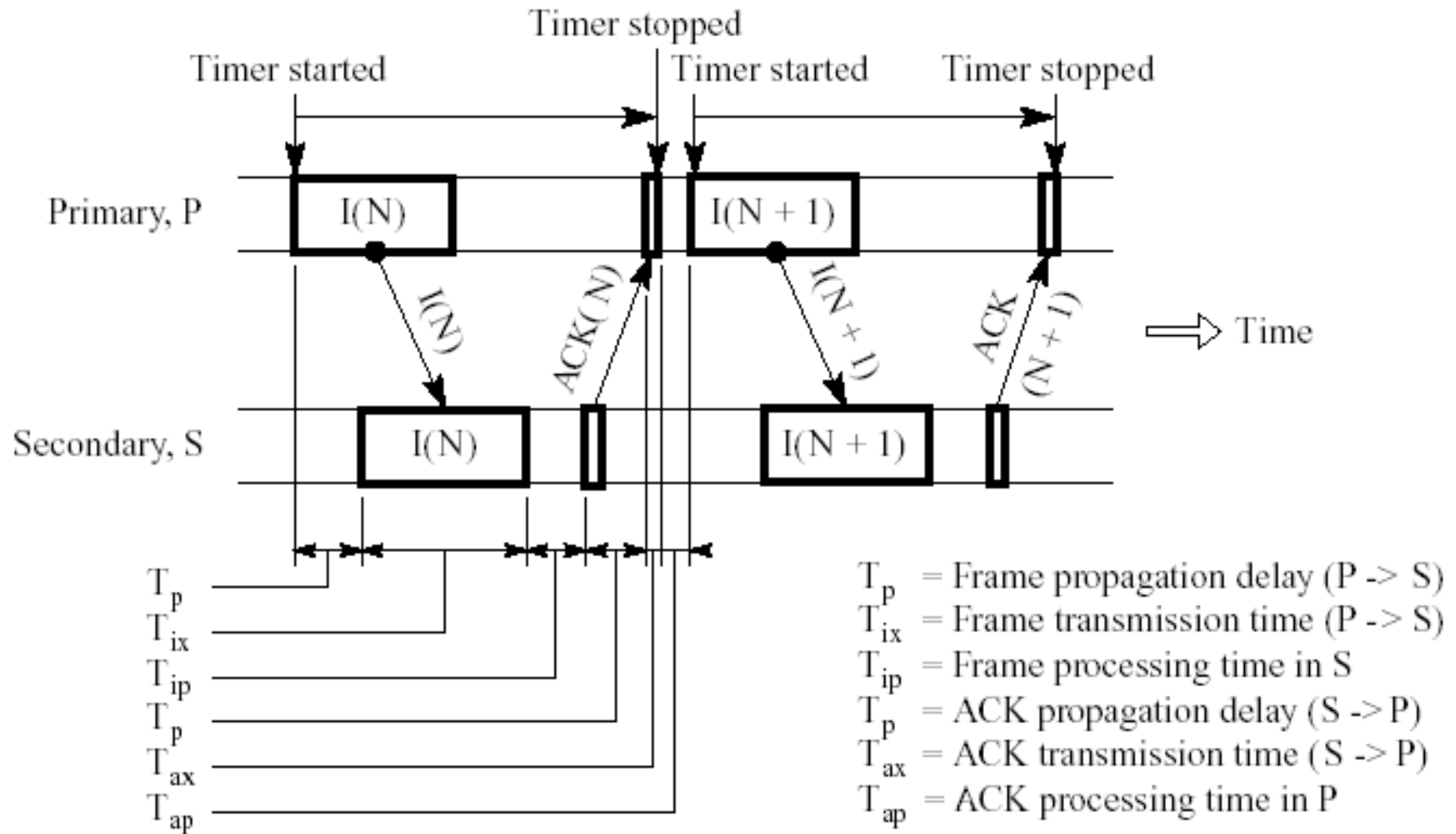
# Idle RQ: Send and Wait



- Pedido de retransmissão explícito
  - Trama de dados corrompida



# Idle RQ: taxa de utilização

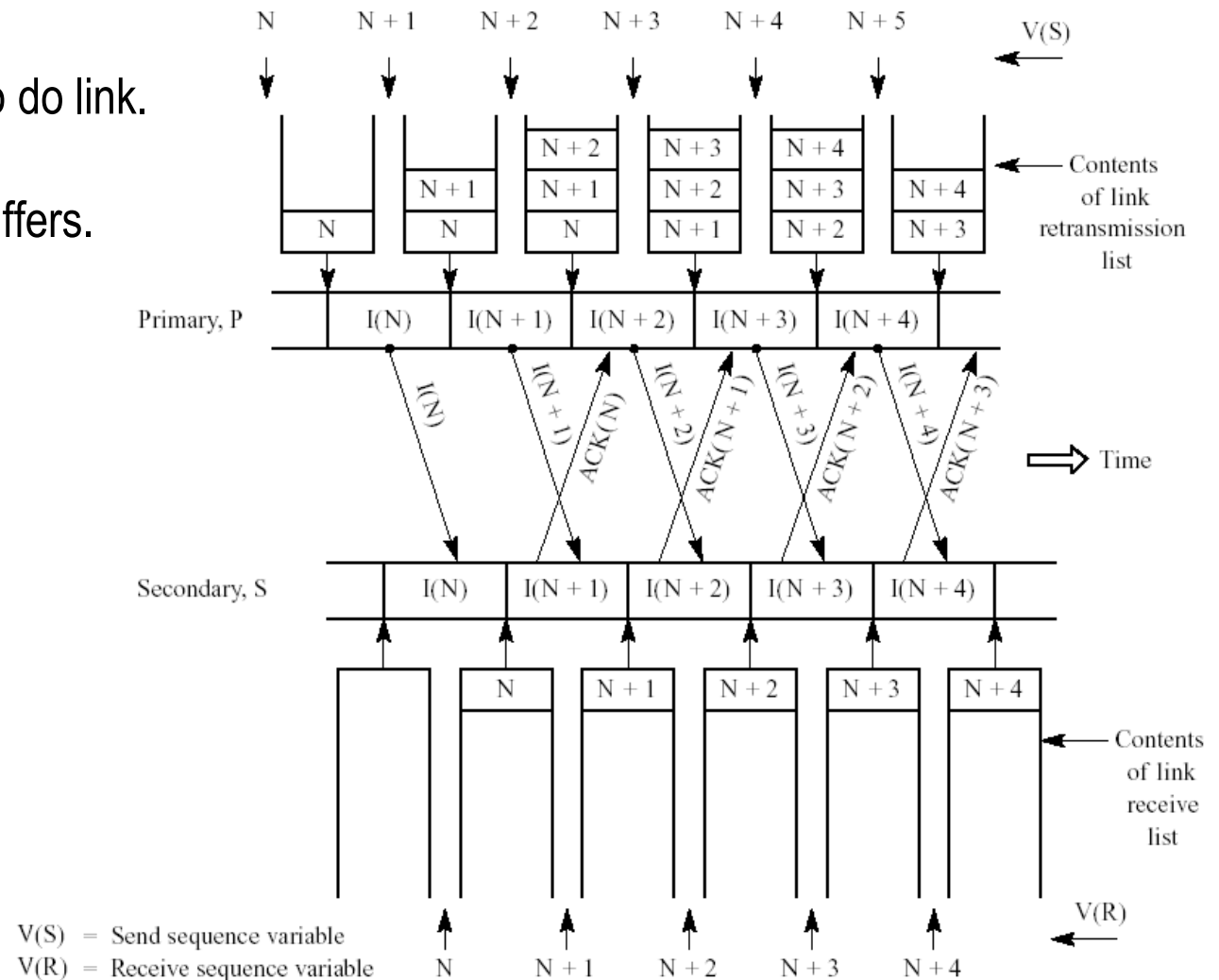




# Continuous RQ



- Aumento da taxa de utilização do link.
- Aumento da dimensão dos buffers.
- Métodos de retransmissão:
  - Selective Repeat
    - Por pedido implícito
    - Por pedido explícito
  - Go-back-N

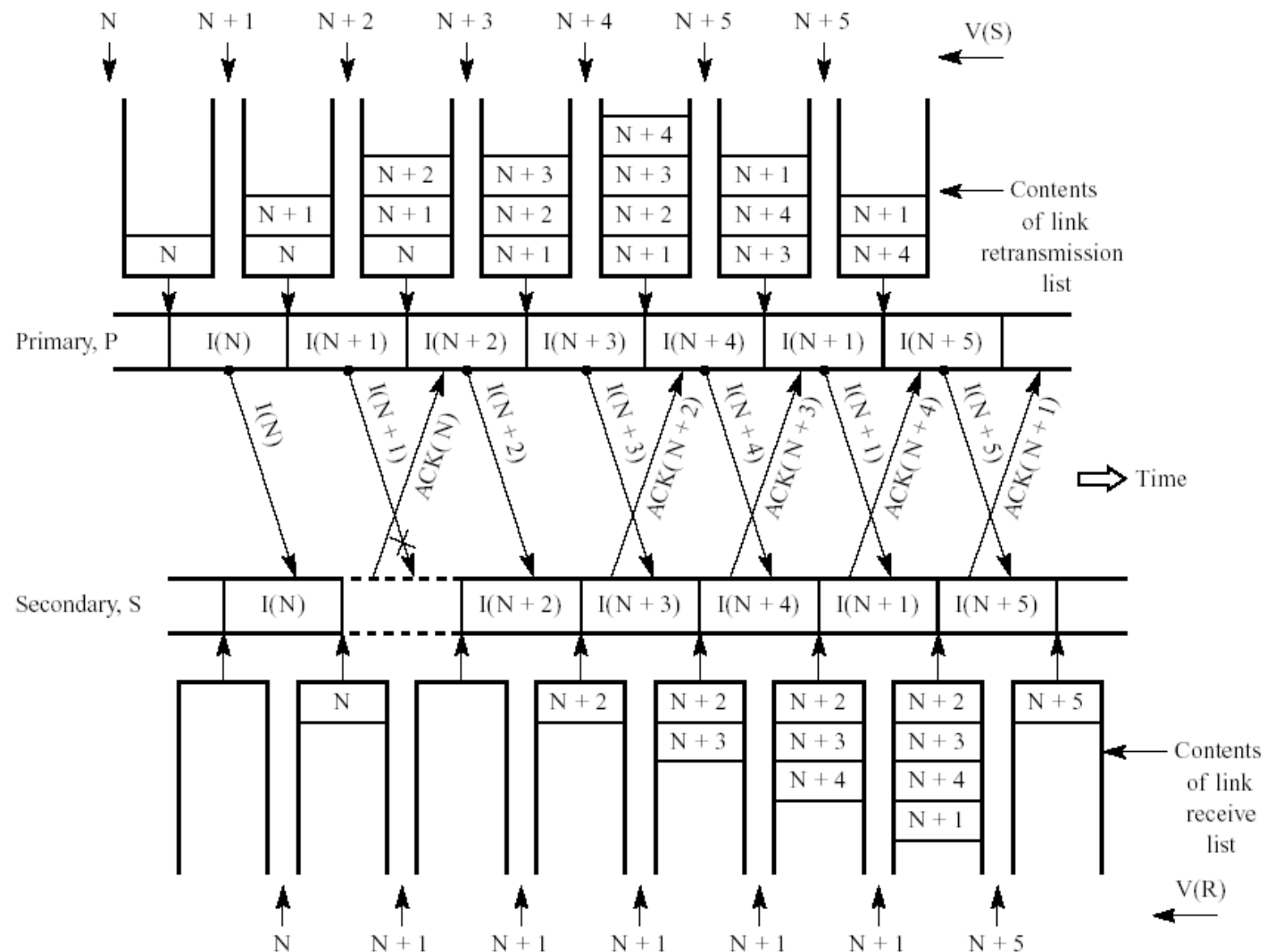


# Continuous RQ - Selective Repeat

Pedido de retransmissão implícito (trama de dados corrompida)



Um ACK por cada  
trama de dados

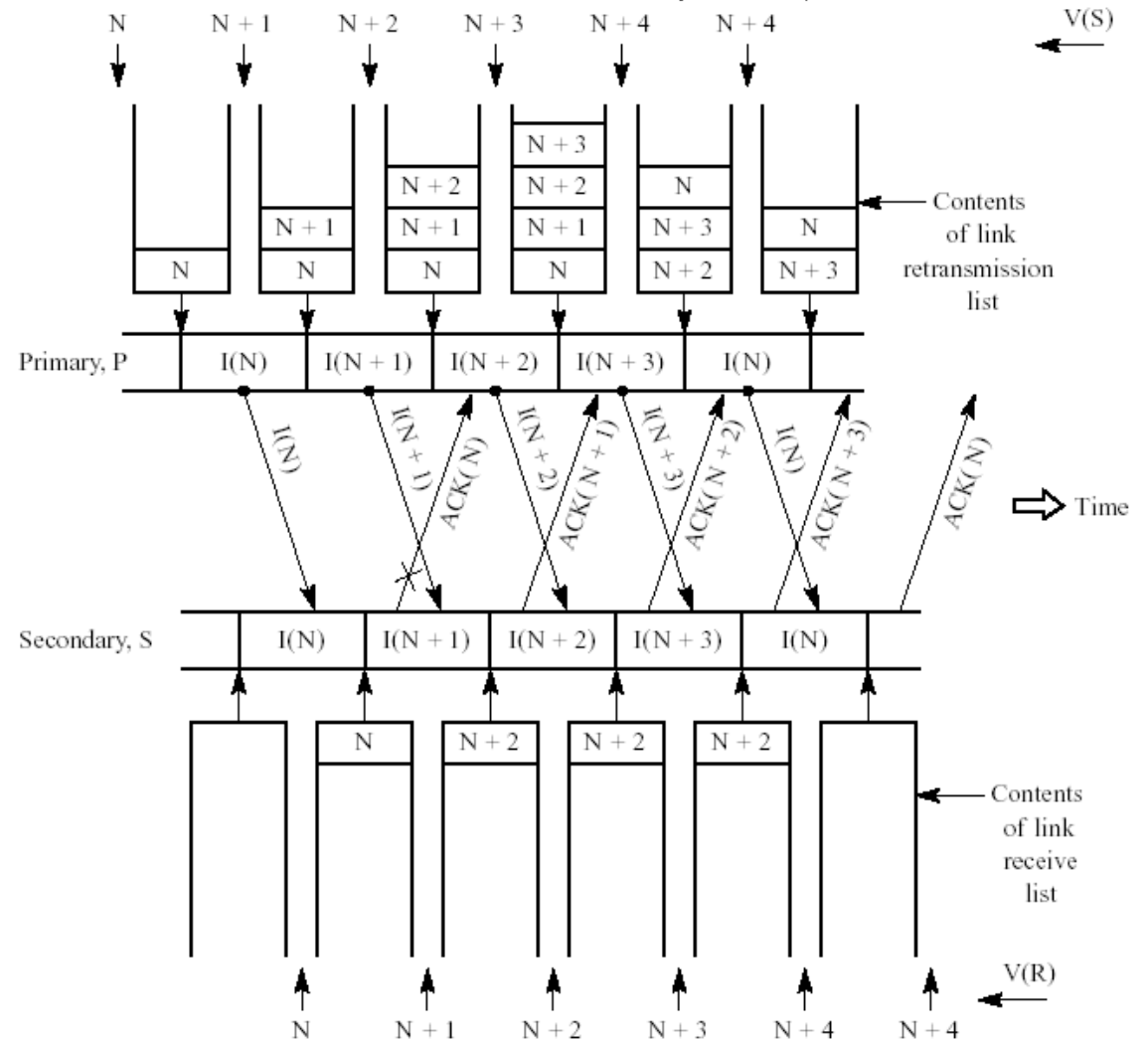


# Continuous RQ - Selective Repeat

Pedido de retransmissão implícito (trama de ACK corrompida)



Um ACK por cada  
trama de dados



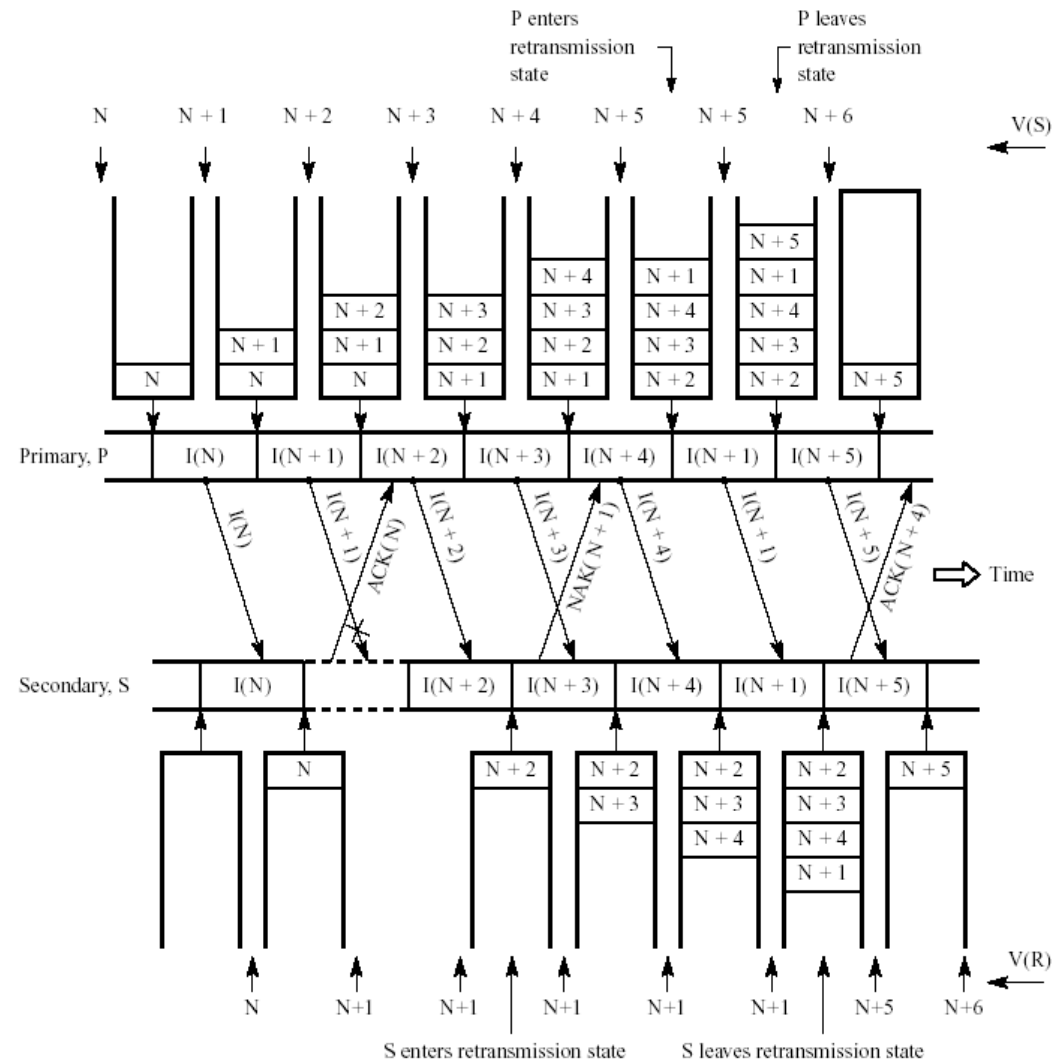
# Continuous RQ - Selective Repeat

## Pedido de retransmissão explícito (operação correcta)



Um ACK indica que recebeu bem a trama e todas as tramas anteriores.

No estado de retransmissão, S, suspende o envio de tramas de ACK.



# Continuous RQ - Selective Repeat

## Pedido de retransmissão explícito (trama de dados e de NACK corrompida)

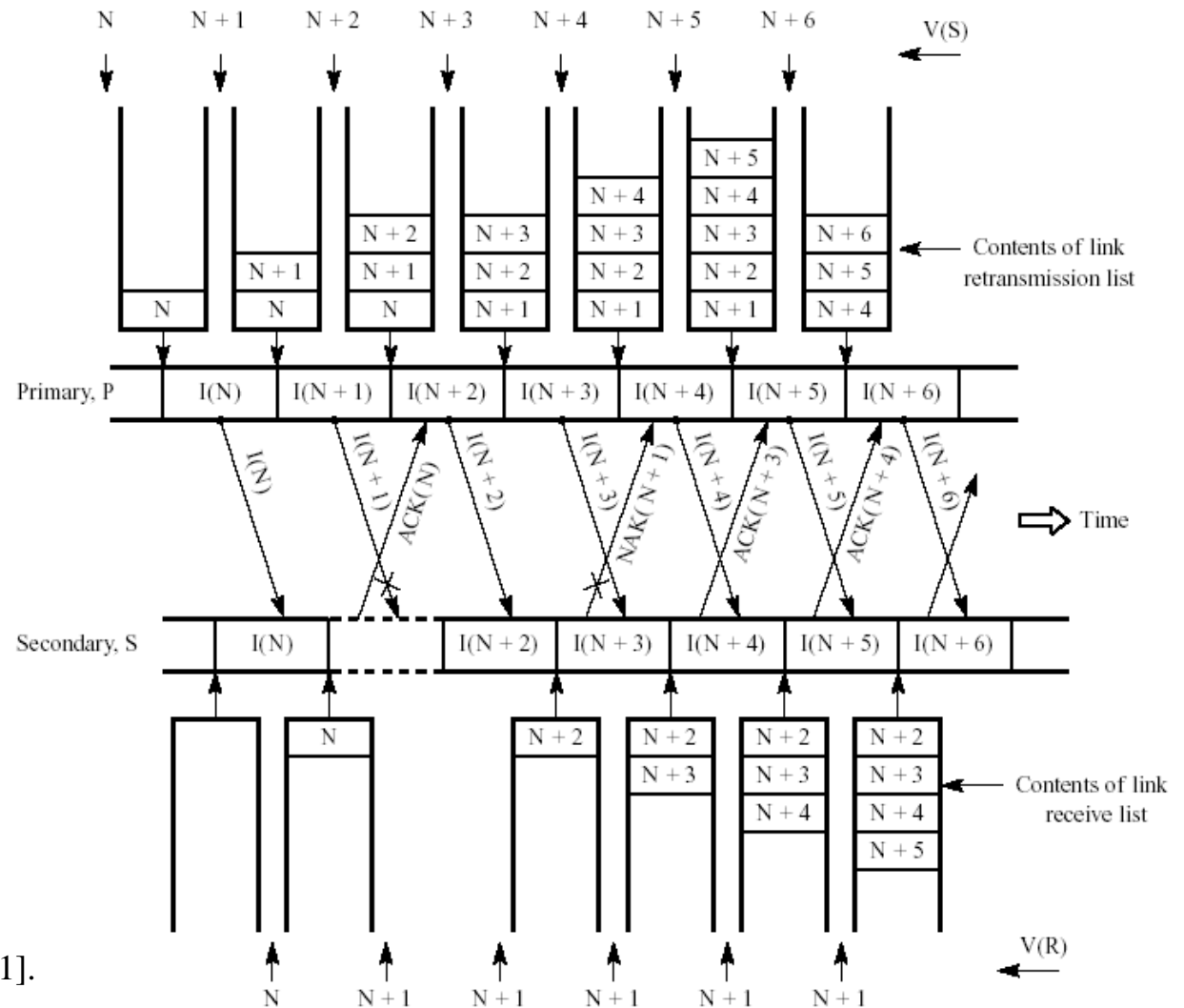


Se no estado de retransmissão não fosse suspenso o envio de ACK haveria falta de tramas visto que...



um ACK indica que recebeu bem a trama e todas as tramas anteriores

Na figura: S nunca recebe a trama I[N+1].



# Continuous RQ - Go Back N

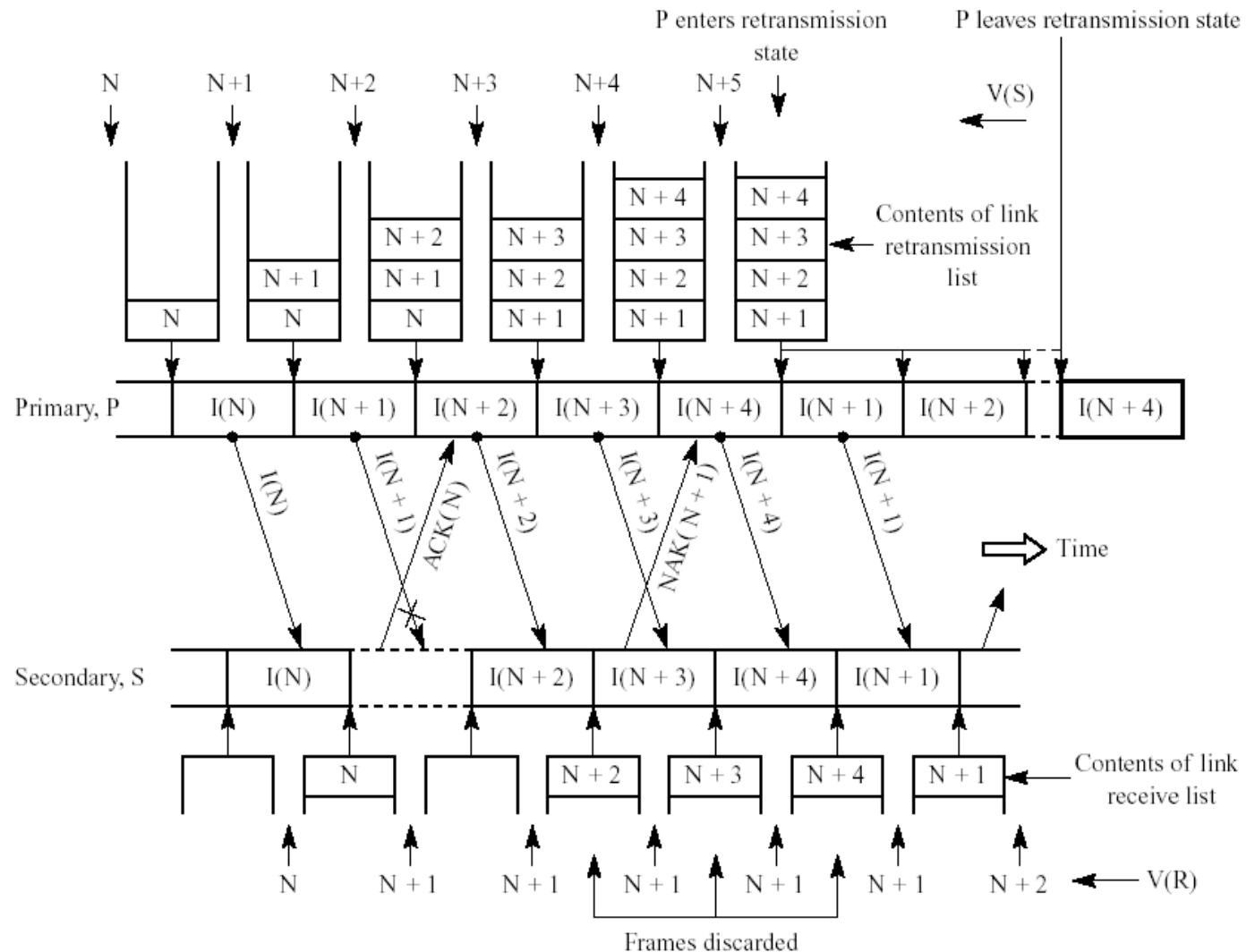
## Pedido de retransmissão explícito (trama de dados corrompida)



Um ACK indica que recebeu bem a trama e todas as tramas anteriores

Um NAK pede retransmissão de todas as tramas a partir da trama indicada.

Após uma falha, S, volta a enviar tramas de ACK apenas quando receber a trama pedida no NACK.

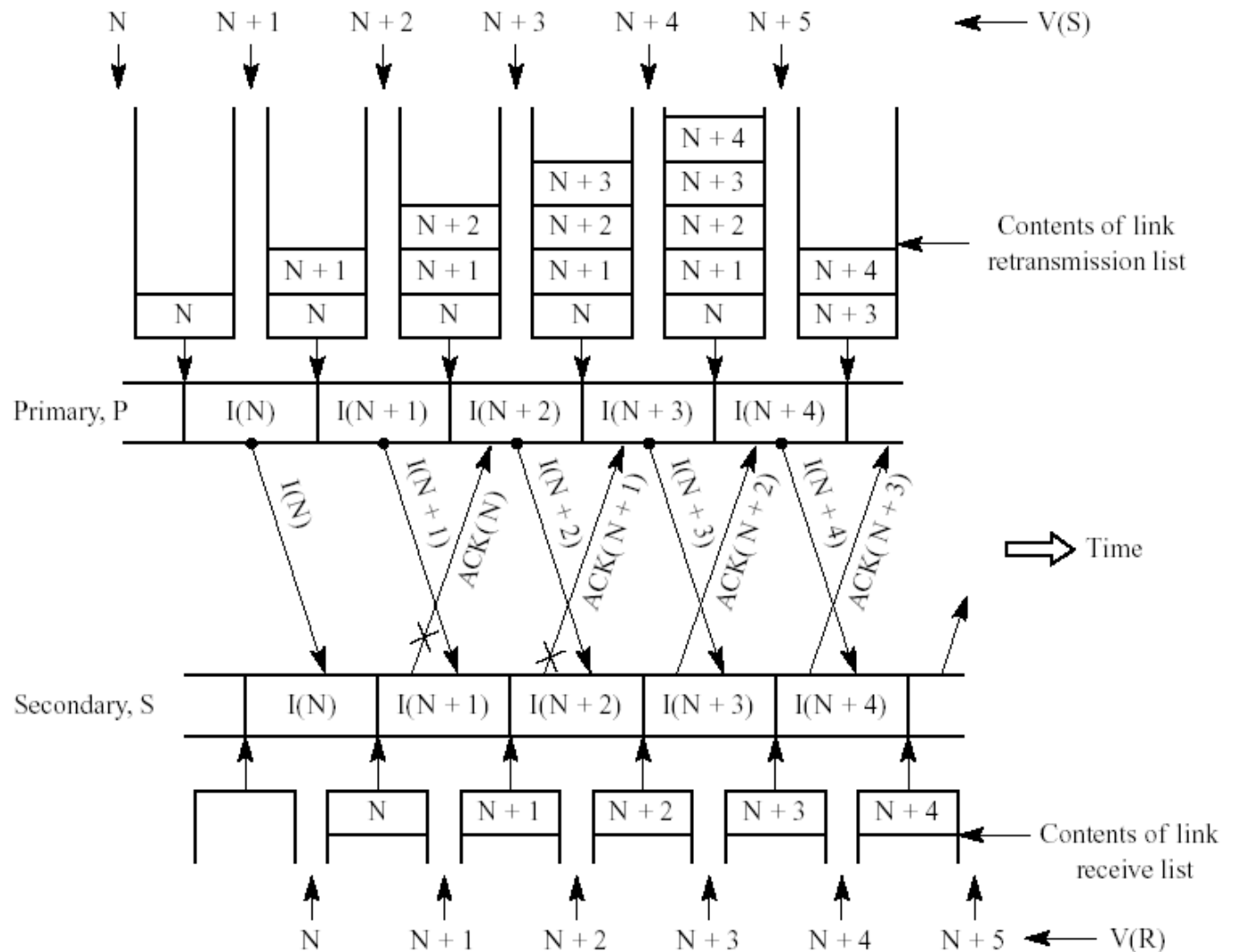


# Continuous RQ - Go Back N

## Trama de ACK corrompida



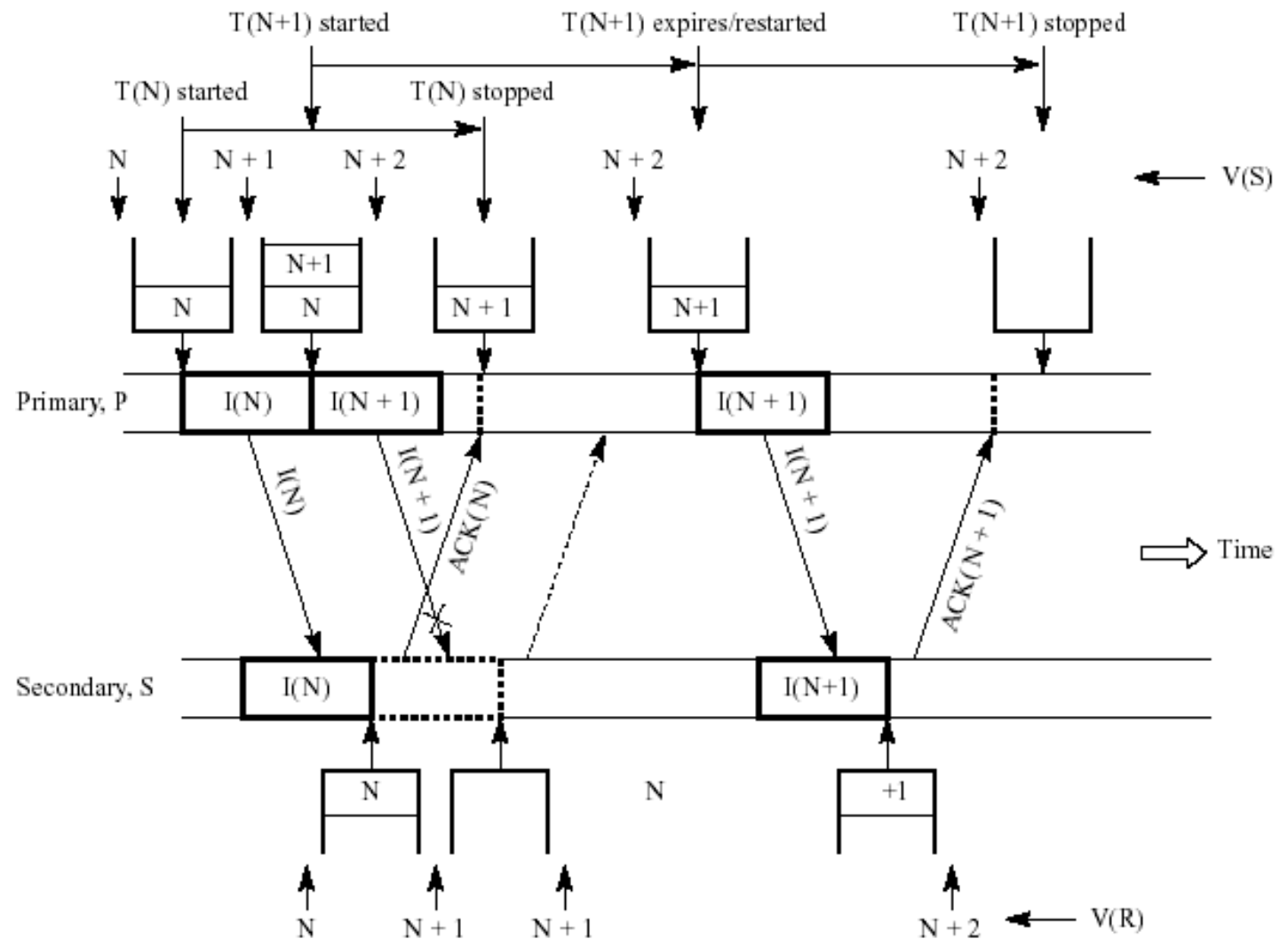
Um ACK indica que recebeu bem a trama e todas as tramas anteriores



# Continuous RQ com temporização



Para quando há uma interrupção no envio de tramas





# Piggyback acknowledgment



- Consideremos um canal full-duplex (não só do ponto de vista físico, mas também lógico, ou seja, que haja lugar a troca de tramas de informação em ambos os sentidos)
- Para aumentar a eficiência de utilização do canal é embebida a informação de *ack* ( $N_R$ ) no cabeçalho das tramas que circulam em sentido oposto, no qual além do número da sequência da trama, existe um campo onde são transportados os bits que informam sobre o estado da ligação no sentido oposto.
- Contudo continua a existirem as tramas de controlo ACK (NACK) para o caso de instantes em que não haja tráfego num dos sentidos.

# Sliding Window



- Mecanismo de controlo do fluxo de transmissão de tramas.
- Limita o número de tramas que podem ser enviadas sem receber *acknowledge*.
- Se houver congestionamento do lado do receptor este não envia *acknowledges* o que faz com que o emissor atinja o limite e pare de enviar tramas.
- Parâmetros:
  - Janela de envio
  - Janela de recepção
  - Números de sequência - identificadores de trama

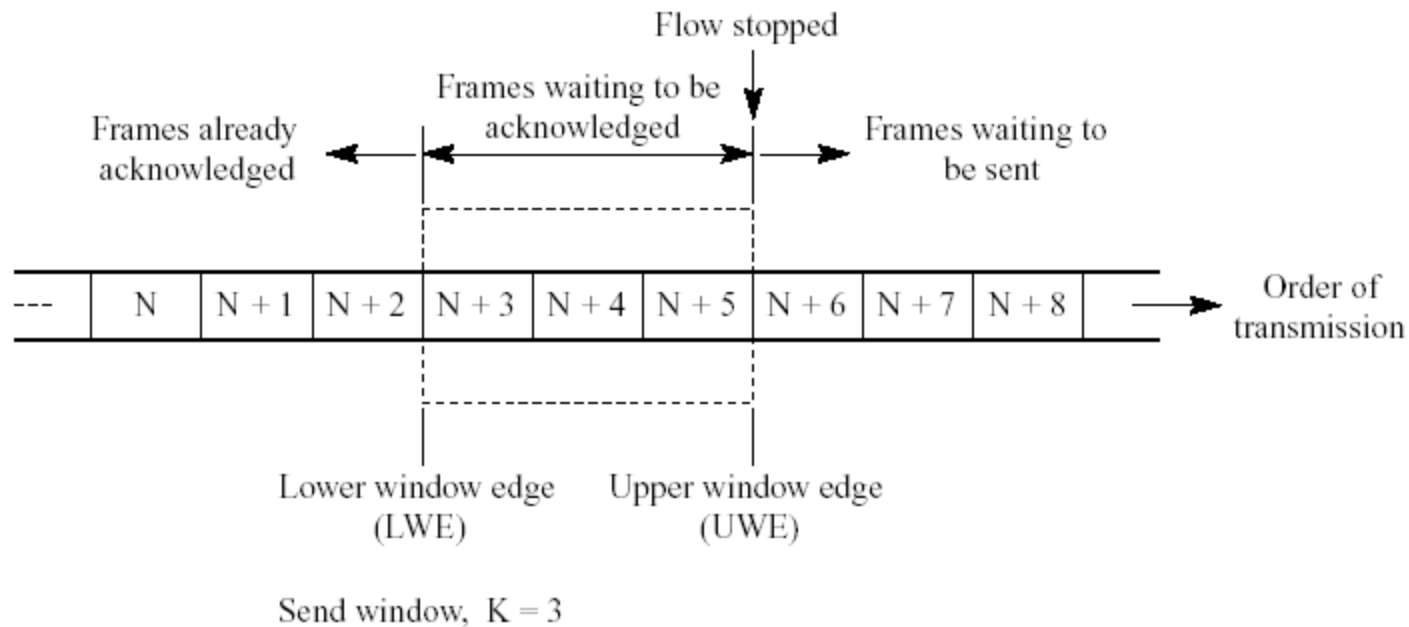
# Sliding Window – Parâmetros

---



- Janela de envio (*Send Window*).
  - Identificadores das tramas que foram enviadas e estão à espera de *acknowledge*.
- Janela de recepção (*Receive Window*).
  - Identificadores das últimas N tramas bem recebidas de modo a que seja possível distinguir tramas duplicadas ou fora de sequência.
- Números de sequência.
  - Número mínimo de Identificadores de trama que são necessários para que os protocolos funcionem sem problemas.

# Controlo de fluxo: *Sliding window*



	Janela de envio	Janela de recepção	Mínimo de Id. da trama
IdleRQ	1	1	2
Selective Repeat	K	K	2K
Go Back N	K	1	K+1

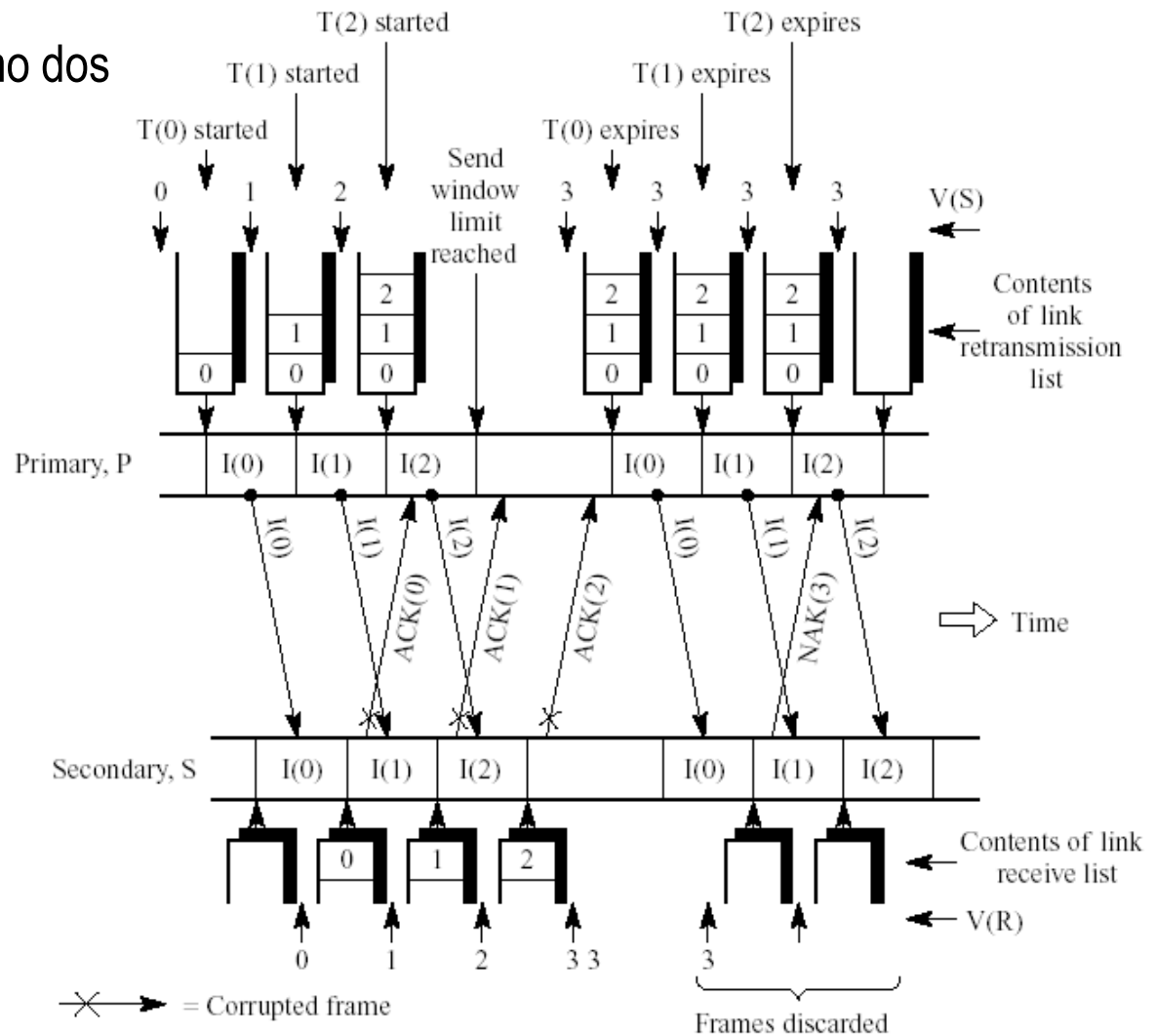
Parametros do mecanismo “Sliding Window” para os vários protocolos:

# Sliding Window



Situação que define o limite mínimo dos números de sequência.

*Send Window* = 3



# Sliding Window

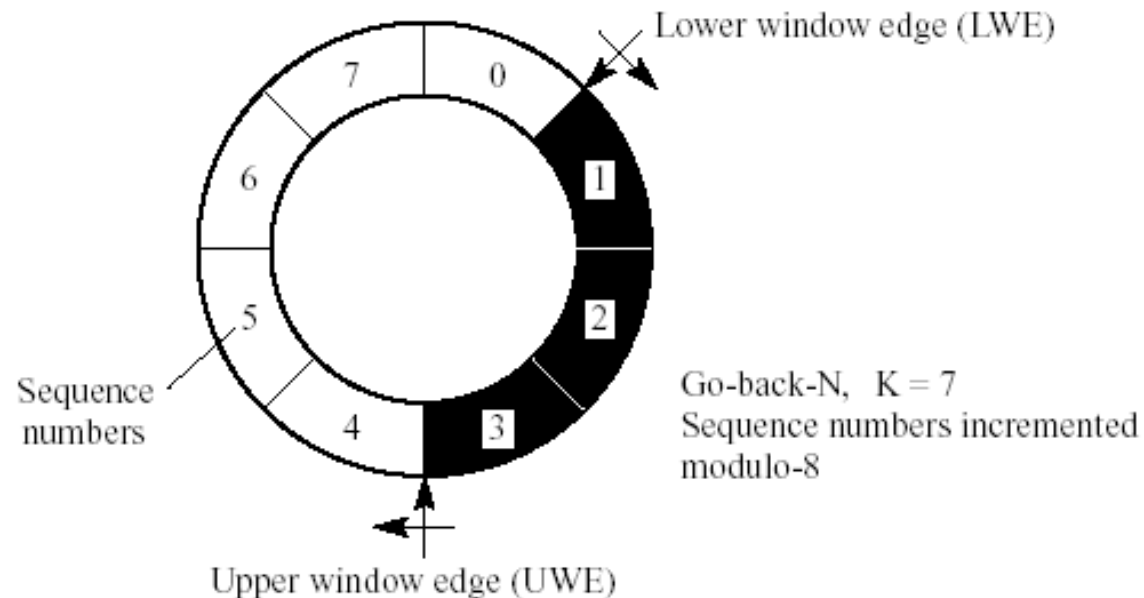


- Consideremos as seguintes hipóteses com base na figura anterior:
  - Go Back N – *Send Window* = 3    *Receive Window* = 1
    - Nº identificadores = 3 - Receptor espera a trama 0
      - A retransmissão da trama 0 pode ser tomada como nova (**errado**)
    - Nº identificadores = 4 - Receptor espera a trama 3
      - Nenhuma das tramas retransmitidas pode ser tomada como nova (**correcto**)
  - Selective Repeat – *Send Window* = 3    *Receive Window* = 3
    - Nº identificadores = 3 - Receptor espera as tramas 0,1,2
      - As tramas 0,1,2 retransmitidas podem ser tomadas como novas (**errado**)
    - Nº identificadores = 4 - Receptor espera as tramas 3,0,1
      - As tramas 0,1 retransmitidas podem ser tomadas como novas (**errado**)
    - Nº identificadores = 6 - Receptor espera as tramas 3,4,5
      - Nenhuma das tramas retransmitidas pode ser tomada como nova (**correcto**)

# Sequencialização: tamanho de janela



Protocol	Maximum number of frame identifiers
Idle RQ	2
Selective repeat	$2K$
Go-back-N	$K + 1$



# Eficiência de utilização do protocolo (*Idle RQ*)

## Sem erros de transmissão



- Nos protocolos **idle RQ**, o tempo de processamento de uma trama de dados ( $T_{ip}$ ) e trama de ACK associada ( $T_{ap}$ ) são bastante mais curtos que os tempos de transmissão ( $T_{ix}$  e  $T_{ax}$ ). Também a trama ACK é muito mais curta quando comparada com a de dados,  $T_{ax}$  é negligenciável quando comparada com  $T_{ix}$ . Assim o tempo mínimo total até que uma próxima trama possa ser transmitida é  $T_{ix} + 2T_p$ . A expressão aproximada para U é:
- $U = T_{ix} / T_t$
- $U = T_{ix} / (T_{ix} + 2T_p)$  ou  $U = (1 + 2T_p / T_{ix})^{-1}$
- $a = T_p / T_{ix}$ ,  $U = (1 + 2a)^{-1}$
- $T_{ix} = N / R$ ,  $T_p = S / V$ 
  - N - Número de bits da trama
  - R - Bit rate em bit/segundo (bps)
  - S - Distância da ligação (m)
  - V - Velocidade de propagação



# Eficiência de utilização do protocolo (*Idle RQ*)

## Com os erros de transmissão



- Reconhecendo a existência de erros nas ligações, o BER será superior a zero. Assim para transmitir uma trama com sucesso, uma media de  $N_r$  tentativas de transmissão são necessárias. A expressão final fica:
- $U = T_{ix} / (N_r T_{ix} + 2N_r T_p)$  ou  $U = (N_r (1 + 2T_p / T_{ix}))^{-1}$
- $P_f = 1 - (1 - P)^{N_i}$ ,  $N_r = 1 / (1 - P_f)$ 
  - $P_f$  - Probabilidade de uma trama ser recebida com erros
  - $N_i$  - Número de bits da trama
  - $P$  - BER - Probabilidade de um bit com erro.
- $a = T_p / T_{ix}$ ,  $U = (1 - P_f) / (1 + 2a)$

# Eficiência de utilização do protocolo (*Continuous RQ*), Sem erros de transmissão



- $T_{ix} = N/R$ ,  $T_p = S/V$ .
  - N - Número de bits da trama.
  - R - Bit rate em bit/segundo (bps).
  - S - Distância da ligação (m).
  - V - Velocidade de propagação.
  - K - Número de tramas da janela de envio.
- $a = T_p / T_{ix}$ .
- $U = 1$ , se  $K \geq 1 + 2a$ .
- $U = (K \cdot T_{ix}) / (T_{ix} + 2T_p) = K / (1 + 2T_p / T_{ix}) = K / (1 + 2a)$  se  $K < 1 + 2a$ .
- Tal pode verificar-se considerando  $T_p = T_{ix}$ . Neste caso o ultimo bit da trama transmitida não é recebido antes de  $2T_p$  ( $2T_{ix}$  também). A trama ACK associada leva  $T_p$  (e  $T_{ix}$ ) a ser recebida. Se  $K=1$  (Idle RQ), então  $U=1/3$ . De modo a elevar U a 100% ( $K > 1+2a$ ), K tem de estar em excesso de 3, isto é três ou mais tramas têm de ser enviadas antes que algum ACK seja recebido.

# Eficiência de utilização do protocolo (*Continuous RQ*)

## Com erros de transmissão (*Selective Repeat*)



- Dada a existência numa situação real de erros que provocarão a retransmissão de tramas, teremos eficiências distintas para os dois esquemas de transmissão *Selective Repeat* e para o *Go Back N*. Por exemplo, com um esquema de *selective repeat*,  $U$  é reduzido pelo número de tentativas para transmitir cada trama  $N_r$  dado que unicamente a trama corrompida é retransmitida. Se  $P_f$  é a taxa de erros de tramas, então, assumindo erros aleatórios:
- $N_r = 1 / (1 - P_f)$ ,  $P_f = 1 - (1 - P)^{N_i}$
- $a = T_p / T_{ix}$
- $U = K / N_r (1 + 2a) = K (1 - P_f) / (1 + 2a)$  se  $K < 1 + 2a$
- Se  $K \geq 1 + 2a$ ,  $U = (1 + 2a) (1 - P_f) / (1 + 2a) = 1 - P_f$

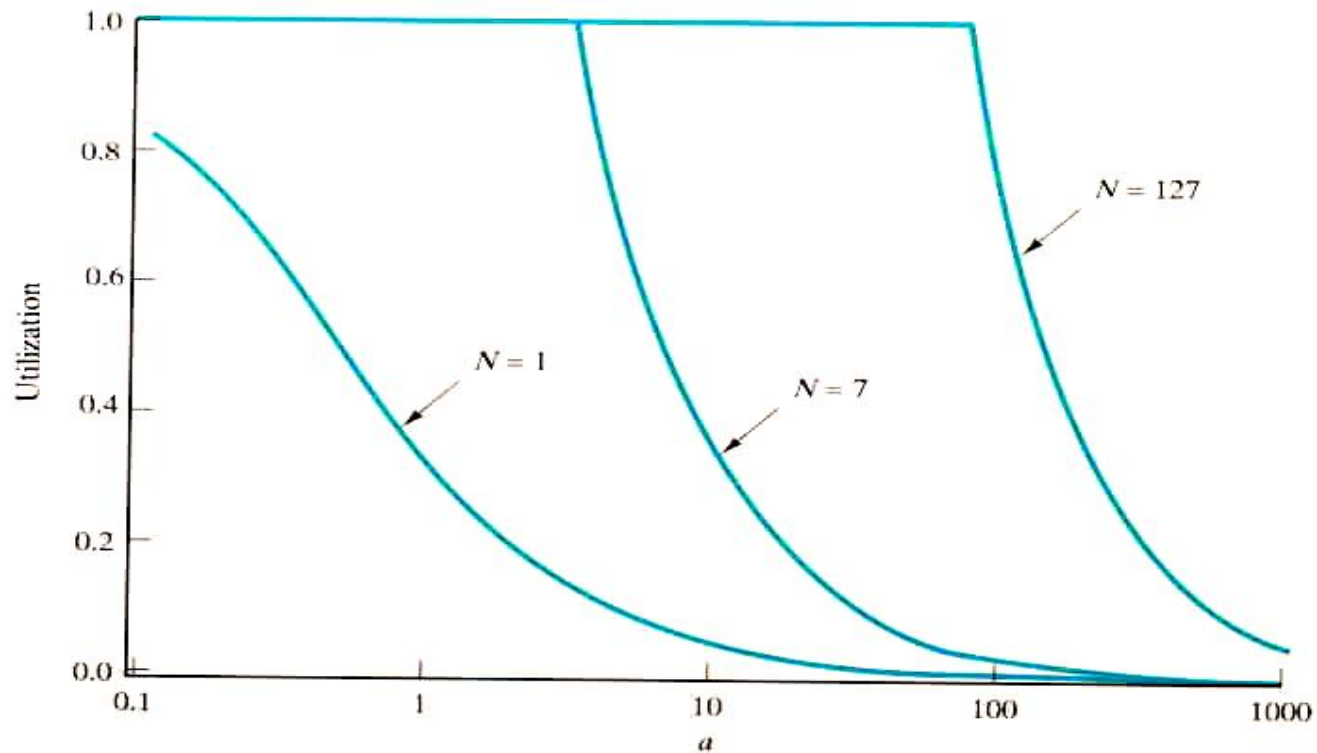
# Eficiência de utilização do protocolo (*Continuous RQ*)

## Com erros de transmissão (*Go Back N*)



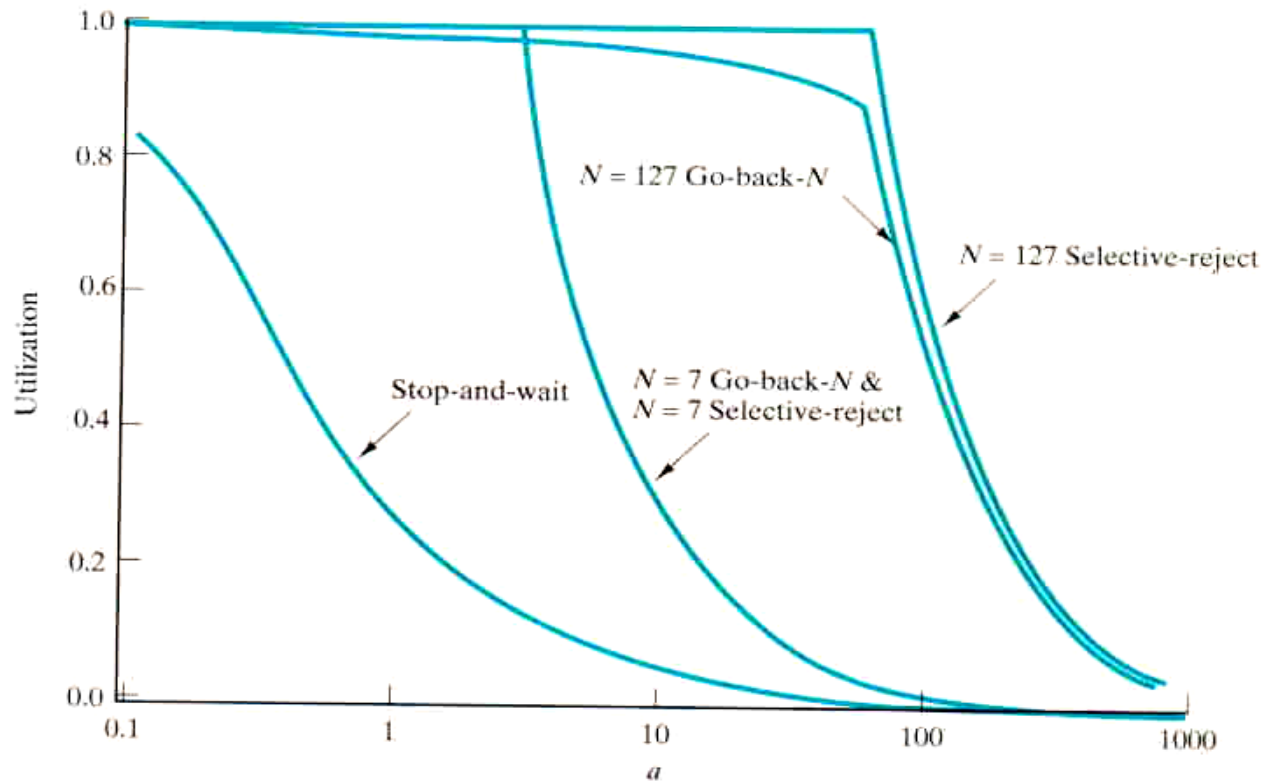
- Para o caso do *Go Back N*, a utilização da linha é ainda mais reduzida pois por cada trama corrompida, mais que uma trama tem de ser retransmitida. Novamente o número de tramas a retransmitir é determinado pela magnitude de  $K$  relativamente a  $1+2a$ . Para  $K$  inferior a  $1+2a$ , o número de vezes que as tramas têm de ser retransmitidas é  $P_f(K-1)$ . Para cada ocorrência, mais um atraso de  $1+2a$  se verificará.
- $N_r = 1 / (1 - P_f)$ ,  $P_f = 1 - (1 - P)^{N_i}$
- $a = T_p / T_{ix}$
- Se  $K < 1 + 2a$ 
  - $U = K(1 - P_f) / ((1 + 2a) + (1 + 2a)P_f(K - 1)) =$
  - $U = K(1 - P_f) / ((1 + 2a)(1 + P_f(K - 1)))$  se  $K < 1 + 2a$
- Se  $K \geq 1 + 2a$ 
  - $U = ((1 + 2a)(1 - P_f)) / ((1 + 2a)(1 + P_f(K - 1))) =$
  - $U = (1 - P_f) / (1 + P_f(K - 1))$

# Comparação de eficiência – sem erros



**FIGURE 6.16** Line utilization as a function of window size.

# Comparação de eficiência – com erros



**FIGURE 6.17** Line utilization for various error-control techniques ( $P = 10^{-3}$ ).



- Sumário:
  - Protocolos de ARQ:
    - Idle RQ (*Send and Wait* ou *Stop and Wait*)
    - Continuous RQ
      - Go Back N
      - Selective Repeat
  - Controlo de fluxo com algoritmo de *Sliding Window*
  - Calculo da eficiência dos protocolos
  - Especificação de protocolos
- Bibliografia:
  - Jim Kurose, Keith Ross, “Computer Networking: A Top Down Approach,” Addison-Wesley, July 2007.