

# Telekomunikacijski protokoli IV

*Drago Hercog*

*Univerza v Ljubljani,  
Fakulteta za elektrotehniko*

*Aplikacijska elektrotehnika, 3. letnik  
2011/2012*

# **Telekomunikacijski protokoli**

**Zagotavljanje zanesljivega  
prenosa**

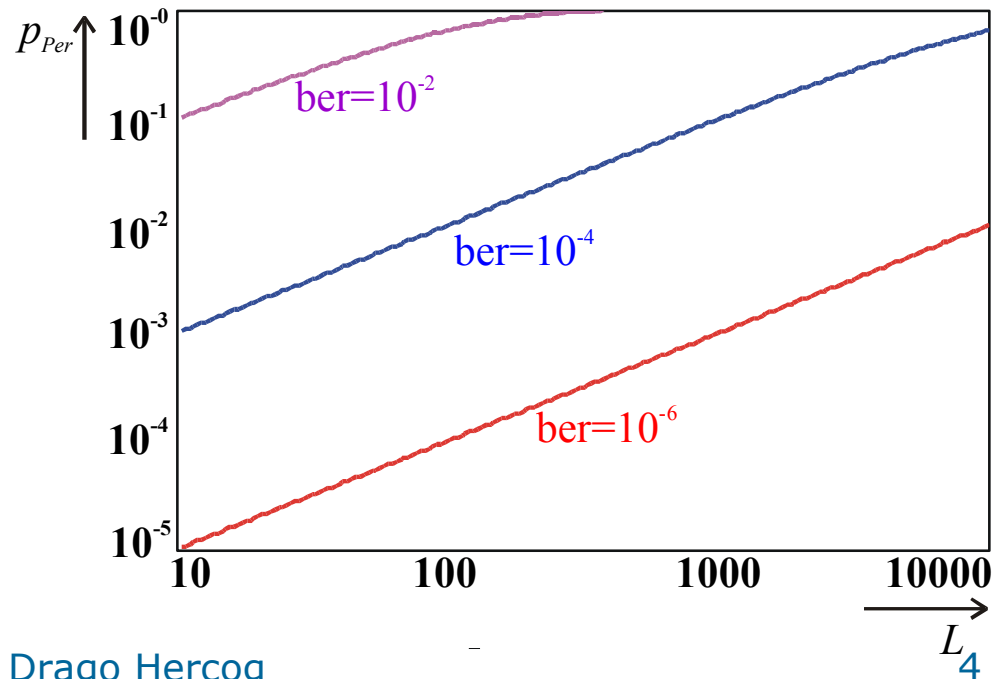
# Slabosti kanala

- kvarjenje sporočil
- izguba sporočil
- podvajanje sporočil
- spreminjanje vrstnega reda sporočil
- zamašitve prometa
  - pri sprejemnem osebku
  - v omrežju

# Razlogi za napake

- šum
- pogostnost bitnih napak
- pogostnost napačno prenešenih paketov

$$P_{pe} = 1 - (1 - P_{be})^L$$





# Razlogi za izgube

- napake v sporočilih
- pomanjkanje virov

# Razlogi za podvajanje sporočil

- protokolne napake
- mrtve duše

# Razlogi za napačen vrstni red

- nepovezavni način komuniciranja
- protokolne napake



# Razlogi za zamašitve

- pomanjkanje virov



# Kaj pomeni zanesljiv prenos

- prenos brez napak
- prenos brez izgub
- vsako oddano sporočilo je sprejeto točno enkrat
- sporočila so sprejeta v istem vrstnem redu kot oddana
- protokol zagotavlja zanesljiv prenos med uporabniki

# Detekcija in popravljanje napak

- detekcija in popravljanje napak na sprejemni strani (FEC)
- detekcija napak na sprejemni strani in popravljanje s ponovnim pošiljanjem (BEC , ARQ)
- zagotavljanje zanesljivega prenosa (ARQ)
- prednosti in slabosti FEC in ARQ

# Prednosti in slabosti FEC in ARQ

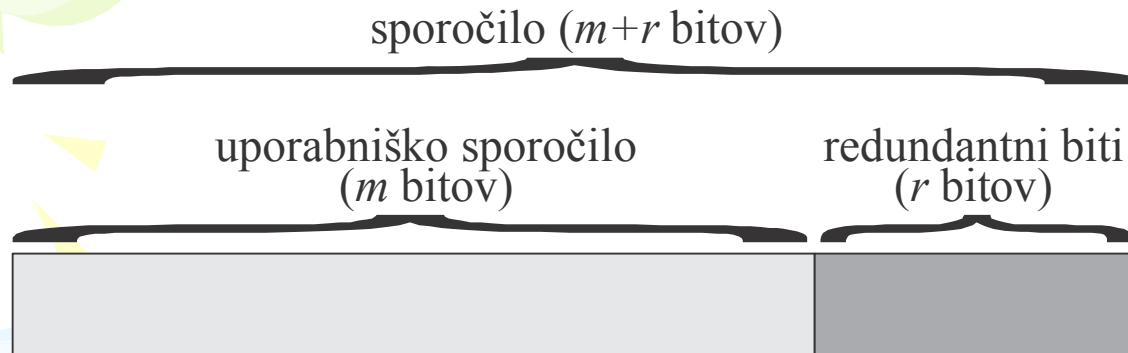
- FEC

- enakomerne zakasnitve
- večja režija
- ne moremo detektirati izgub

- ARQ

- manjša režija
- pri majhnih BER učinkovito
- spremenljive zakasnitve
- pri dolgih kanalih zelo spremenljive zakasnitve in neučinkovitost
- pri velikih BER neučinkovito

# Princip kanalskega kodiranja



**kanalsko kodiranje: uporabniško sporočilo  $\times$  metoda  $\Rightarrow$  sporočilo**

število možnih uporabniških sporočil:  $2^m$

število vseh možnih sporočil:  $2^{m+r}$

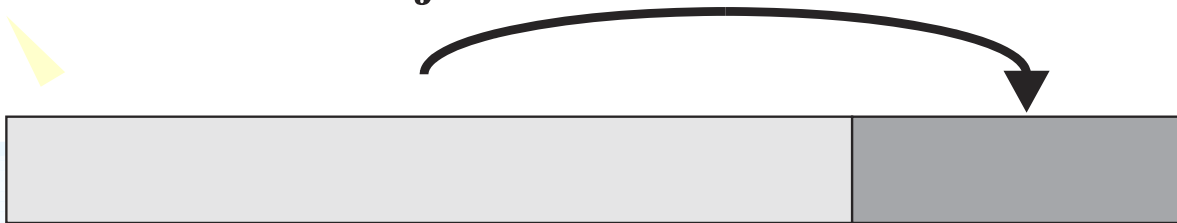
število veljavnih sporočil:  $2^m$

**detekcija in/ali popravljanje napak!**

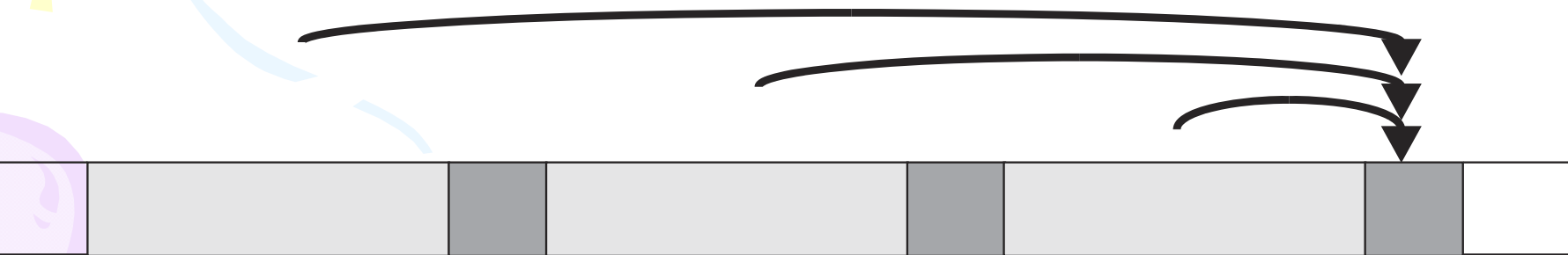
**kvaliteta kodiranja: verjetnost, da napake ne odkrijemo/popravimo!**

# Bločno in konvolucijsko kodiranje

**bločno kodiranje:**



**konvolucijsko kodiranje:**



# Posplošena obravnava bločnega kodiranja

- minimalna Hammingova razdalja
- Hammingove kode
- kode za detekcijo
- kode za popravljanje
- popravljanje na sprejemni strani
- popravljanje s ponovnim pošiljanjem

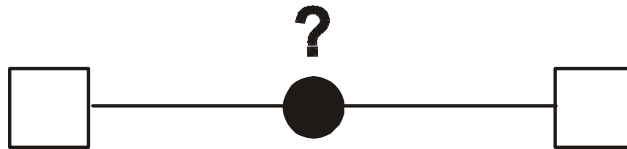
$$d = d_{H_{\min}} - 1 \qquad c = \frac{d_{H_{\min}} - 1}{2}$$

# Minimalna Hammingova razdalja

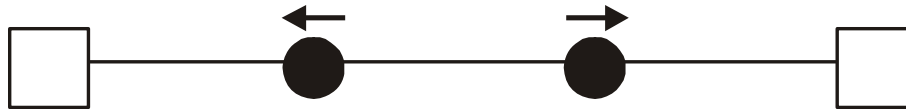
$d_h = 1$



$d_h = 2$



$d_h = 3$



$d_h = 4$



# Metoda paritete (Parity)

- soda pariteta
- liha pariteta
- metoda je preprosta
- delež odkritih napak je majhen
- seštevanje po modulu 2
- zaščita znakov



# Metoda seštevanja (checksum)

- sporočilo razdelimo na besede dolžine  $n$  bitov
- besede seštejemo po modulu  $2^n$
- vsoto dodamo kot zaščitno besedo

# Metoda krožnega redundantnega preizkusa - CRC

- $M$  : uporabniško sporočilo
- $G$  : generacijski polinom
- $R$  : ostanek pri deljenju  $M$  z  $G$
- $CRC : M \times G \rightarrow R$
- $T$  : poslano sporočilo
- $T$  je deljivo z  $G$ !
- postopek dolgega deljenja temelji na seštevanju po modulu 2

$T$  :



# CRC – matematično ozadje

- Predstavitev binarnega zaporedja kot polinoma:

$$M = \mu_{m-1} \cdot 2^{m-1} + \mu_{m-2} \cdot 2^{m-2} + \dots + \mu_1 \cdot 2^1 + \mu_0 \cdot 2^0$$

- Oddano zaporedje mora biti deljivo z  $G$ :

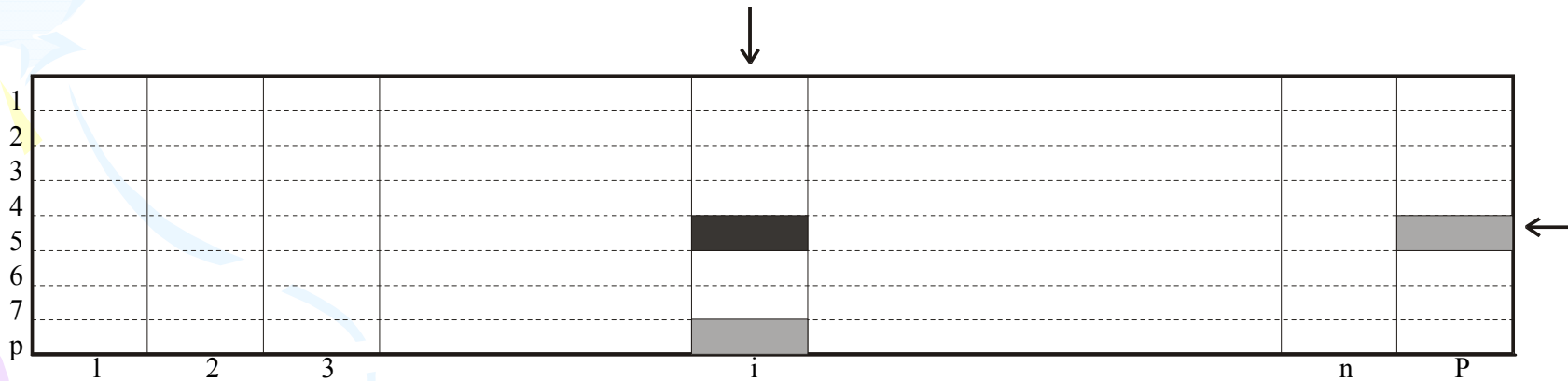
$$T = M \cdot 2^r \oplus R = Q \cdot G$$

- Torej:

$$M \cdot 2^r = Q \cdot G \oplus R$$

# Metoda dvodimenzionalne paritete

- sprejemnik popravi enobitne napake



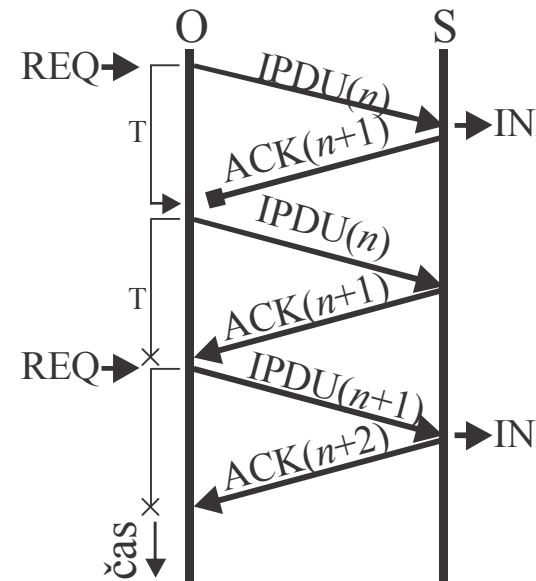
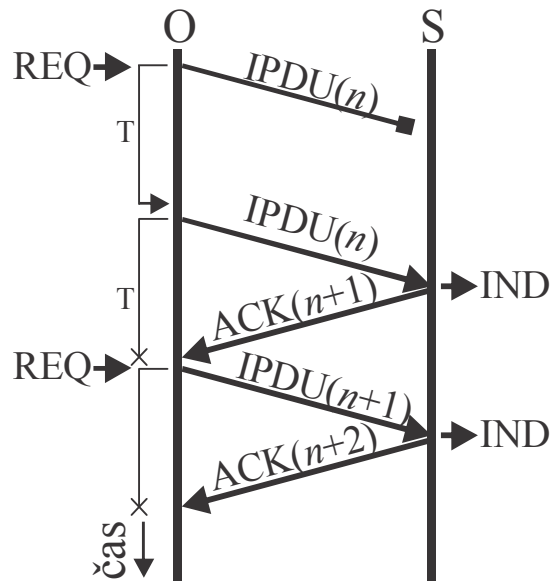
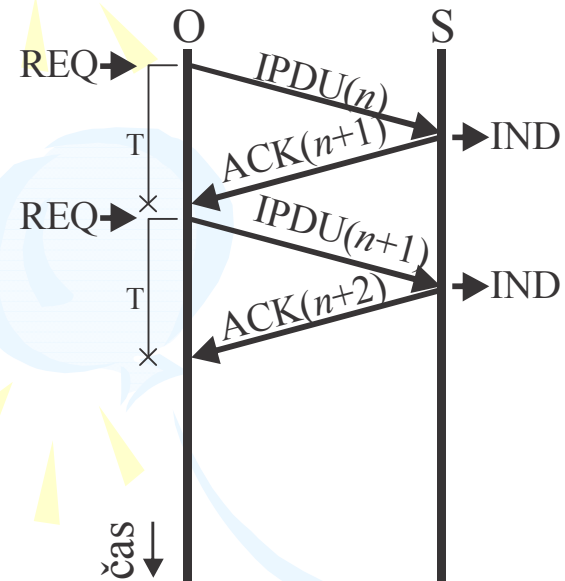
# Protokoli s ponavljanjem

- oddajnik pokvarjeno/izgubljeno sporočilo ponovno pošlje
- pogoji za logično pravilno delovanje
  - kanalsko kodiranje in detekcija napak
  - potrjevanje pravilnega sprejema
  - uporaba časovnika
  - zaporedno številčenje protokolnih sporočil
- dodatni mehanizmi za izboljšanje učinkovitosti
- povezavno orientirani prenos
- poldupleksni ali dupleksni prenos
- protokoli z drsečim oknom
  - protokol s čakanjem
  - protokol s tekočim pošiljanjem
    - protokol s ponavljanjem zaporedja
    - protokol s selektivnim ponavljanjem

# Protokol s čakanjem (Stop-and-Wait Protocol)

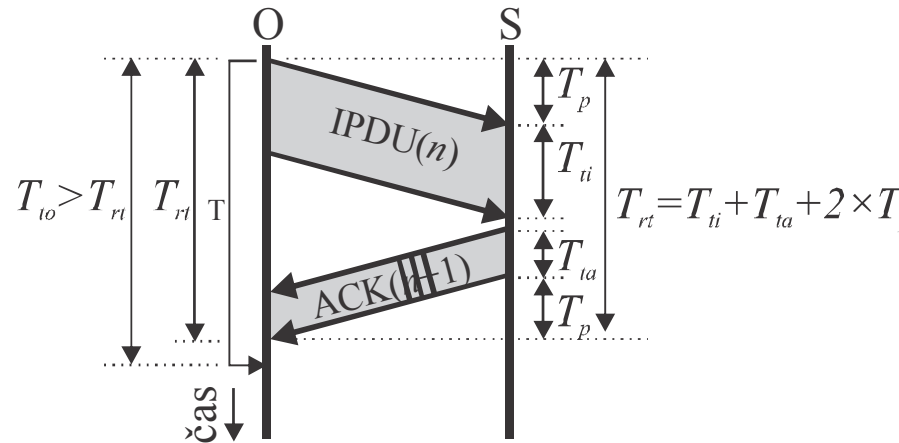
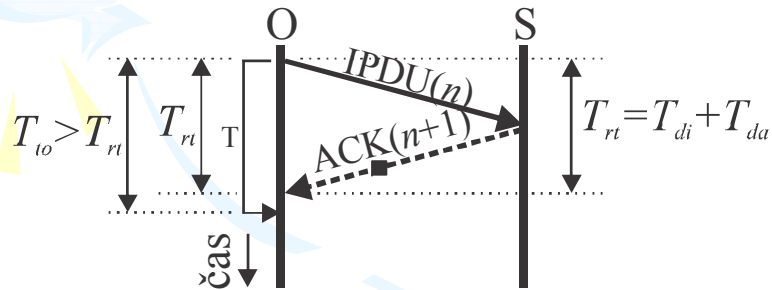
- oddajnik pošlje IPDU
- sproži časovnik
- čaka na potrditev
- pošlje naslednji IPDU ali ponovi prejšnjega
- protokol je preprost, a neučinkovit

# Protokol s čakanjem – tipični scenariji



# Uporaba časovnika

- časovnik mora teči, če ima oddajnik kakšen nepotrjen paket
- proženje, zaustavitev, iztek časovnika
- čas izteka časovnika

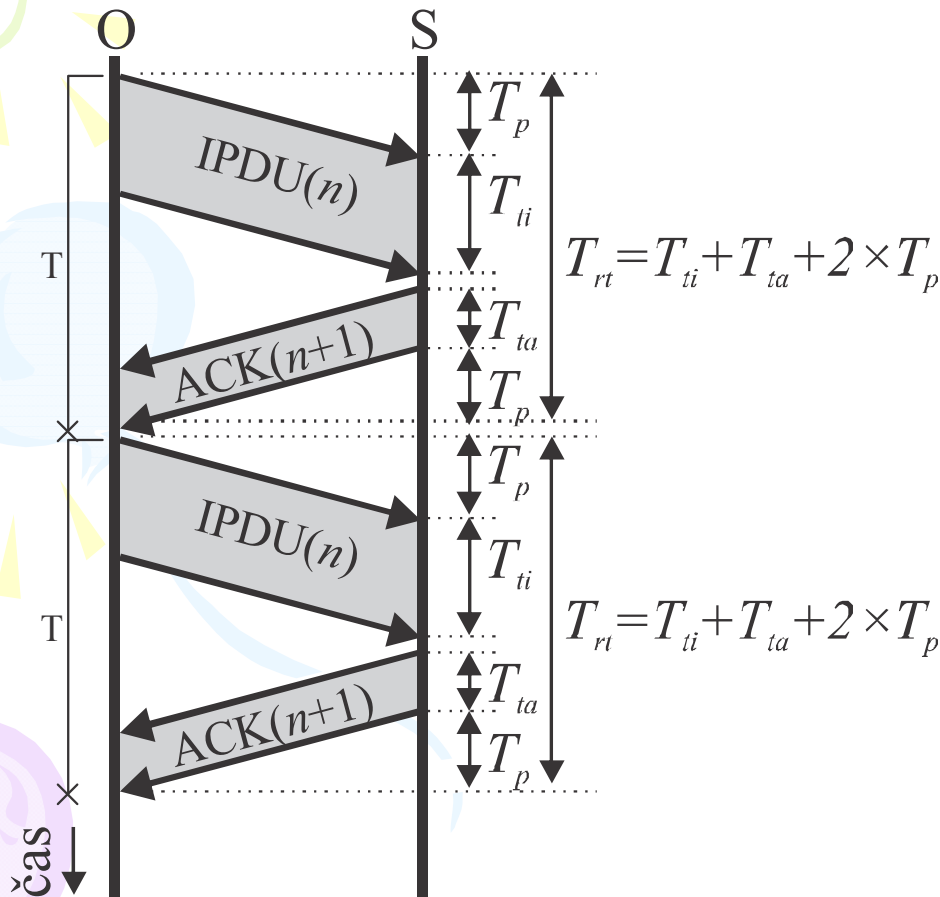




# Številčenje sporočil

- omogoča detekcijo izgubljenih sporočil
- omogoča detekcijo podvojenih sporočil

# Učinkovitost protokola s čakanjem



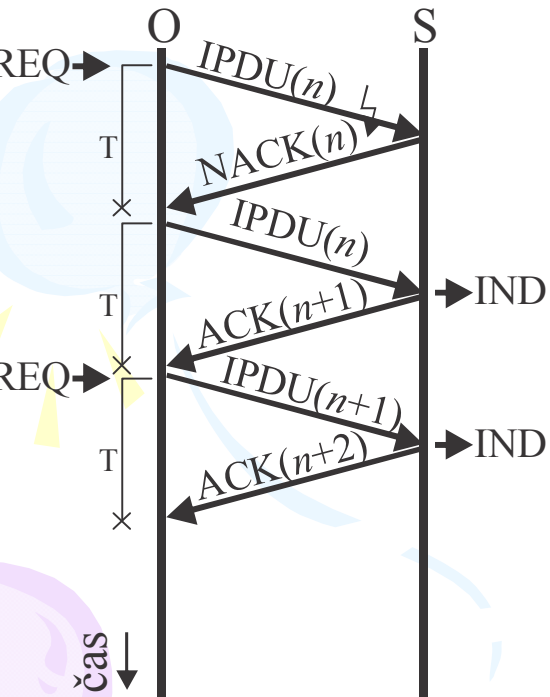
$$\eta = \frac{t_{ti}}{t_{rt}} = \frac{t_{ti}}{t_{ti} + t_{ta} + 2 \cdot t_p}$$

# Učinkovitost protokola s čakanjem - primer

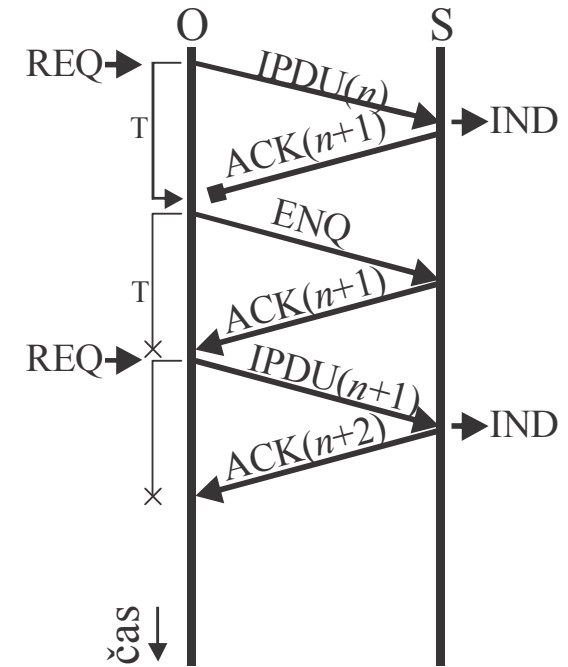
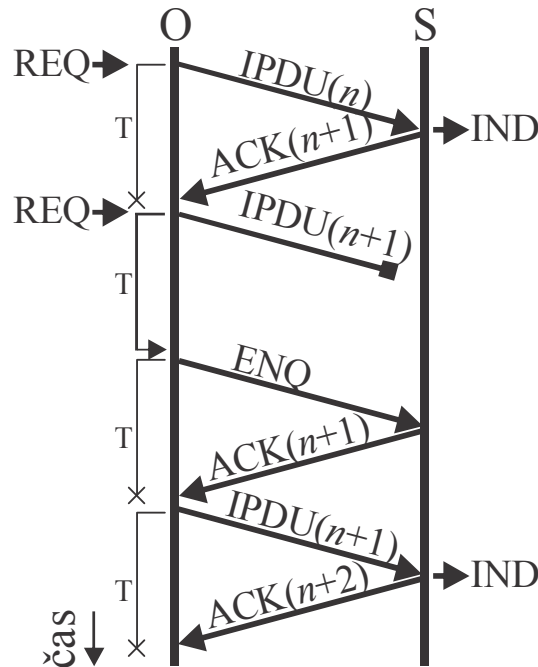
$R$	$L_i$	$L_a = L_r$	$D$	$v$	$\eta$	$\eta_u$	$R_u$
64 kb/s	1000 o	3 o	10 km	$c$	99,6 %	99,3 %	63,6 kb/s
64 kb/s	50 o	3 o	1000 km	$c$	47,0 %	44,2 %	28.3 kb/s

# Možne izboljšave učinkovitosti

negativne  
potrditve



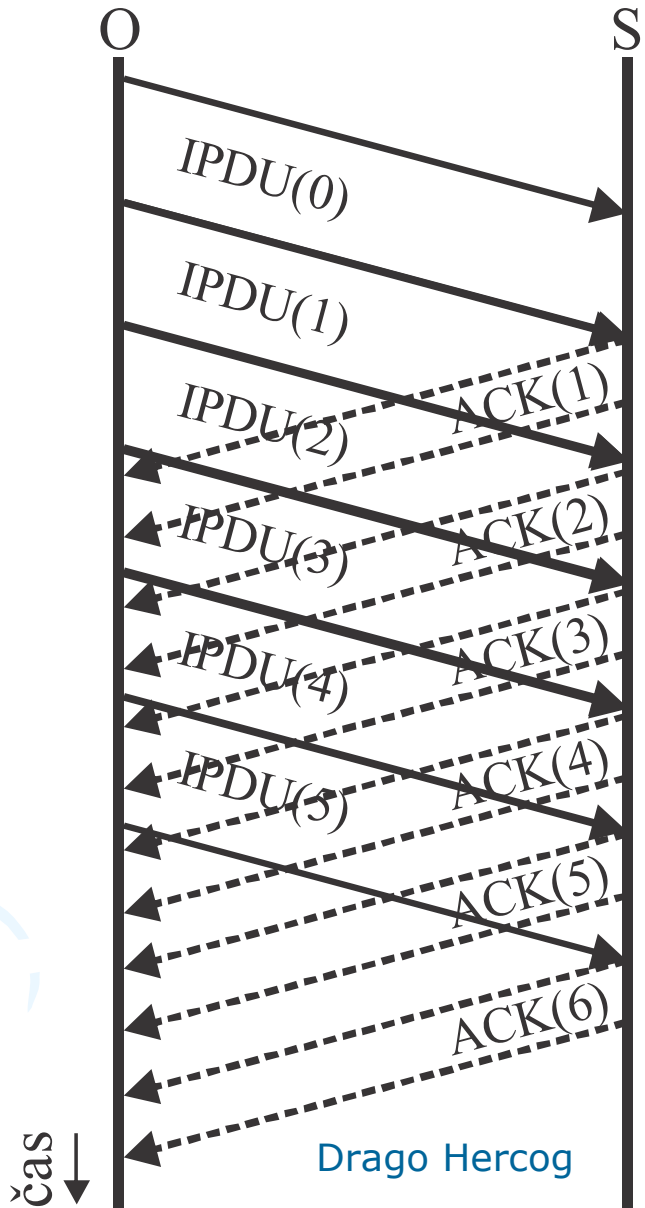
poizvedbe



# Protokoli s tekočim pošiljanjem

- boljša učinkovitost
- potreben je duplexni kanal
- oddajnik po oddaji enega sporočila nadaljuje z oddajanjem naslednjih
- pomnjenje nepotrjenih sporočil
- omejitev oddajanja – oddajno okno
- širina oddajnega okna  $Ws$
- drseče okno

# Teško pošiljanje



# Oddajno okno:

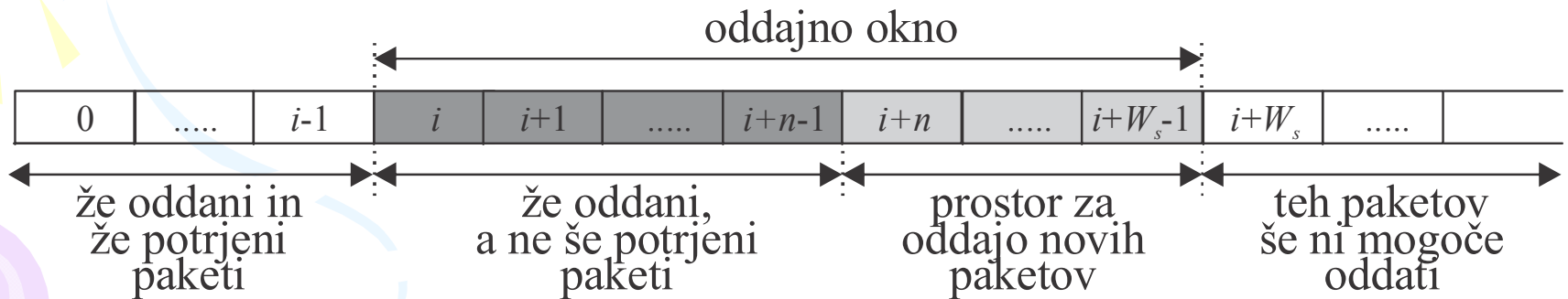
- $W_s$
- $i$
- $n$

širina oddajnega okna

število že oddanih in potrjenih paketov

število že oddanih, a ne potrjenih paketov

spomin za hranjenje oddanih, a ne potrjenih paketov  
(oddajni vmesnik)



# Oddajno okno: posebni dogodki in stanja

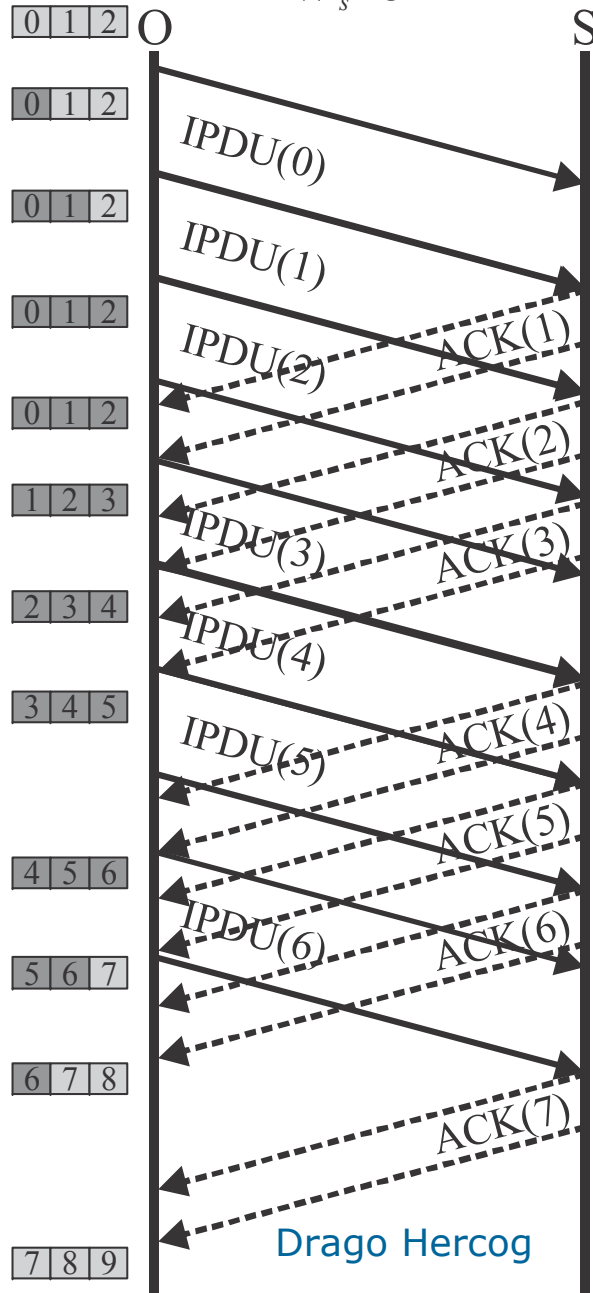
- prispe potrditev paketa  $i+1$ : okno se premakne naprej
- $n < W_s$ : oddaja novih paketov je mogoča
- $n \geq W_s$ : okno je polno, oddaja novih paketov ni mogoča



# Zgled

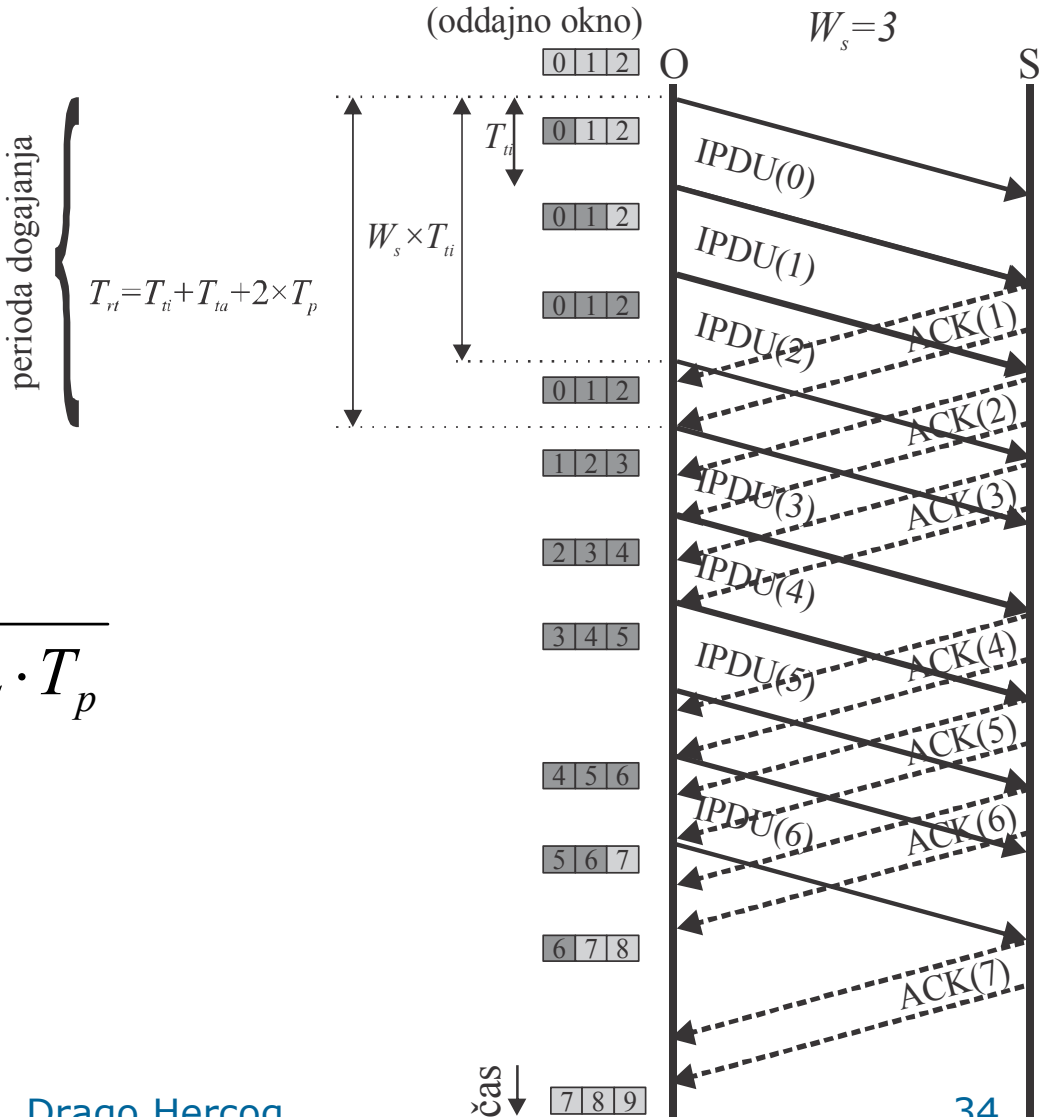
(oddajno okno)

$W_s = 3$



# Učinkovitost protokola s tekočim pošiljanjem

- predpostavka: kanal brez izgub



$$\eta = \frac{W_s \cdot T_{ti}}{T_{rt}} = \frac{W_s \cdot T_{ti}}{T_{ti} + T_{ta} + 2 \cdot T_p}$$

VENDAR:  $\eta \leq 1$  !!!

# Načrtovanje velikosti oddajnega okna

- učinkovitost 100 % pri kanalu brez izgub:

$$\eta = \frac{W_s \cdot T_{ti}}{T_{rt}} = 1 \Rightarrow W_s \geq \frac{T_{rt}}{T_{ti}} \Rightarrow W_s \geq \frac{T_{ti} + T_{ta} + 2 \cdot T_p}{T_{ti}}$$

- kompromis: učinkovitost - spomin

# Sprejemno okno

- spomin za začasno shranjevanje sprejetih paketov
- spomin za urejanje sprejetih paketov v pravem vrstnem redu
- $W_r$  – širina sprejemnega okna
- $i$  – število že sprejetih, po vrsti urejenih in uporabniku predanih paketov

spomin za hranjenje sprejetih,  
a ne urejenih in uporabniku predanih paketov  
(sprejemnik vmesnik)



sprejemno okno

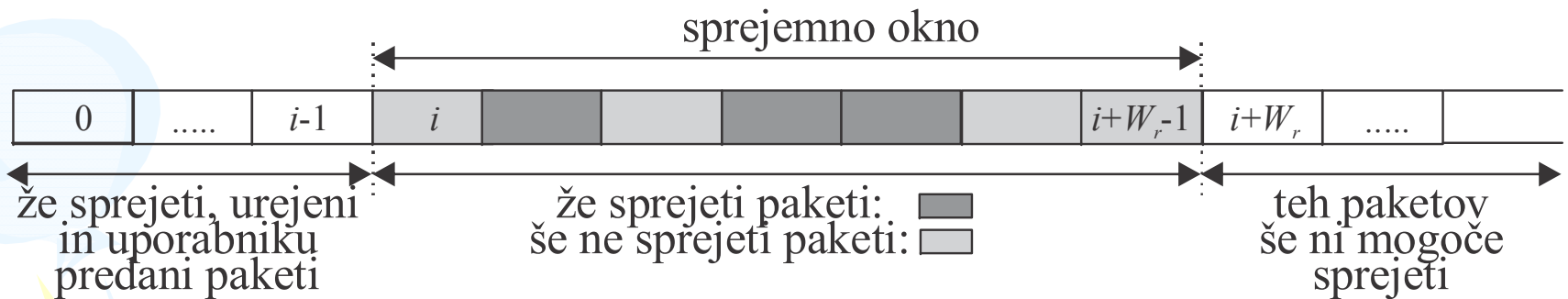


že sprejeti, urejeni  
in uporabniku  
predani paketi

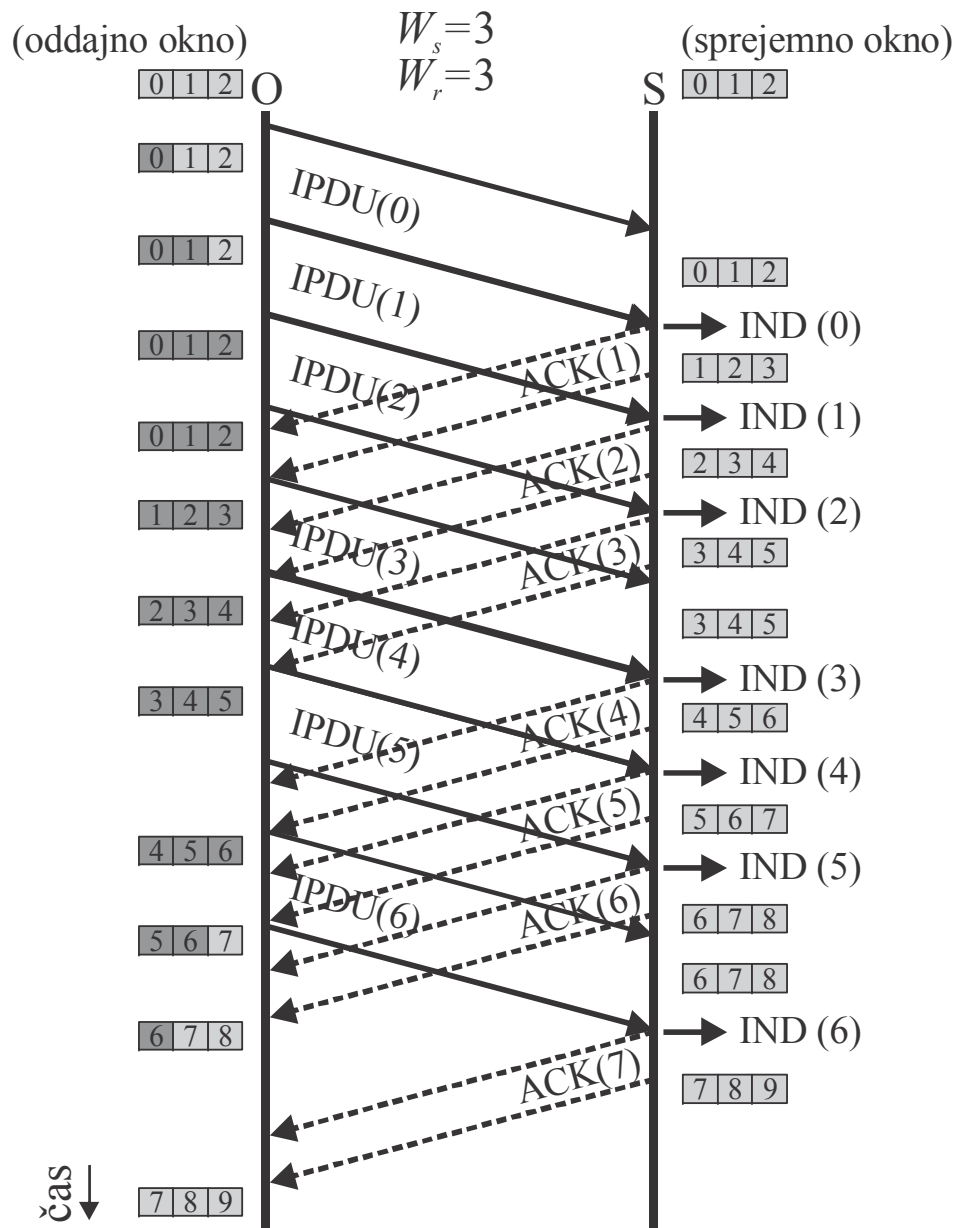
prostor za sprejem paketov;  
nekateri so že sprejeti, a še ne urejeni,  
nobeden še ni predan uporabniku

teh paketov  
še ni mogoče  
sprejeti

# Urejanje paketov v sprejemnem vmesniku



# Drsenje sprejemnega okna, kanal brez izgub



# Potrditve

- vrste potrditev
  - individualne potrditve
  - kumulativne potrditve
  - bločne potrditve
- običajno: kumulativne potrditve

# Časovnik

- eden ali več časovnikov
- običajno: en časovnik
  - mora teči, če ima oddajnik v oddajnem vmesniku vsaj en nepotrjen paket
  - oddajnik požene časovnik, ko odda paket, če časovnik že ne teče
  - oddajnik ustavi časovnik, ko sprejme novo potrditev, in ga takoj spet požene, če ima kakšen nepotrjen paket
- $T_{to} > T_{rt}$
- komentar časa izteka časovnika

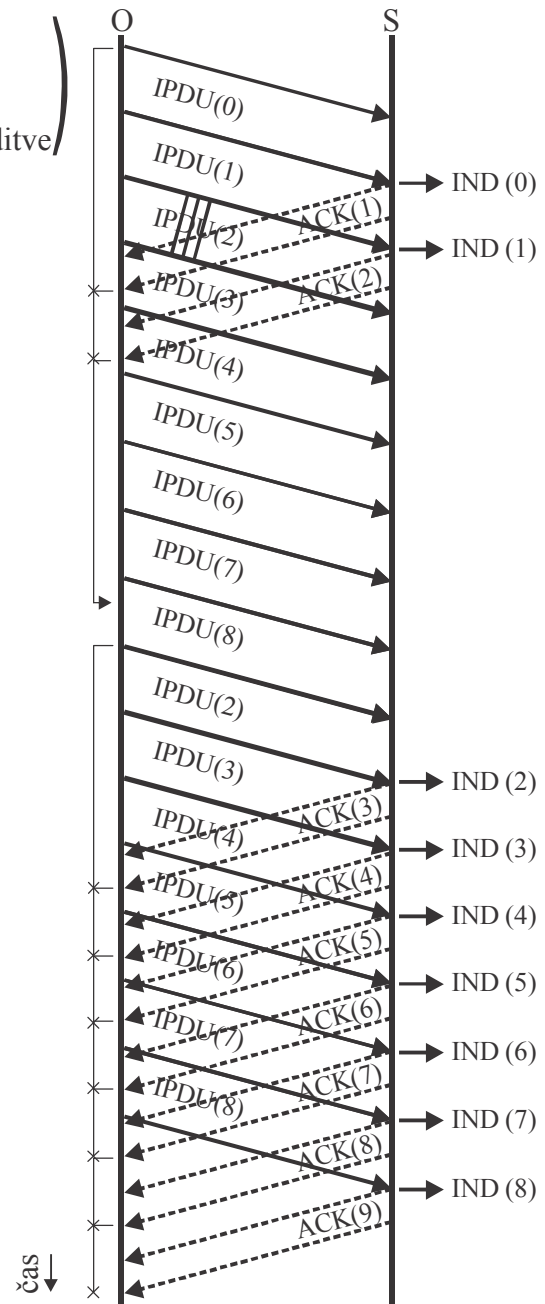


# Prisotnost izgub: protokol s ponavljanjem zaporedja

- ang. Go-Back-N Protocol
- oddajnik ponovno pošlje vse pakete od prvega pogrešanega dalje
- sprejemnik zavrže pakete, ki jih ne sprejme v pravem vrstne redu
- sprejemnik potrди le pakete, ki jih sprejme v pravem vrstnem redu
- sprejemnik sprejema pakete v pravem vrstnem redu
- sprejemnik ne potrebuje spomina za urejanje paketov po vrsti,  $W_r=1$

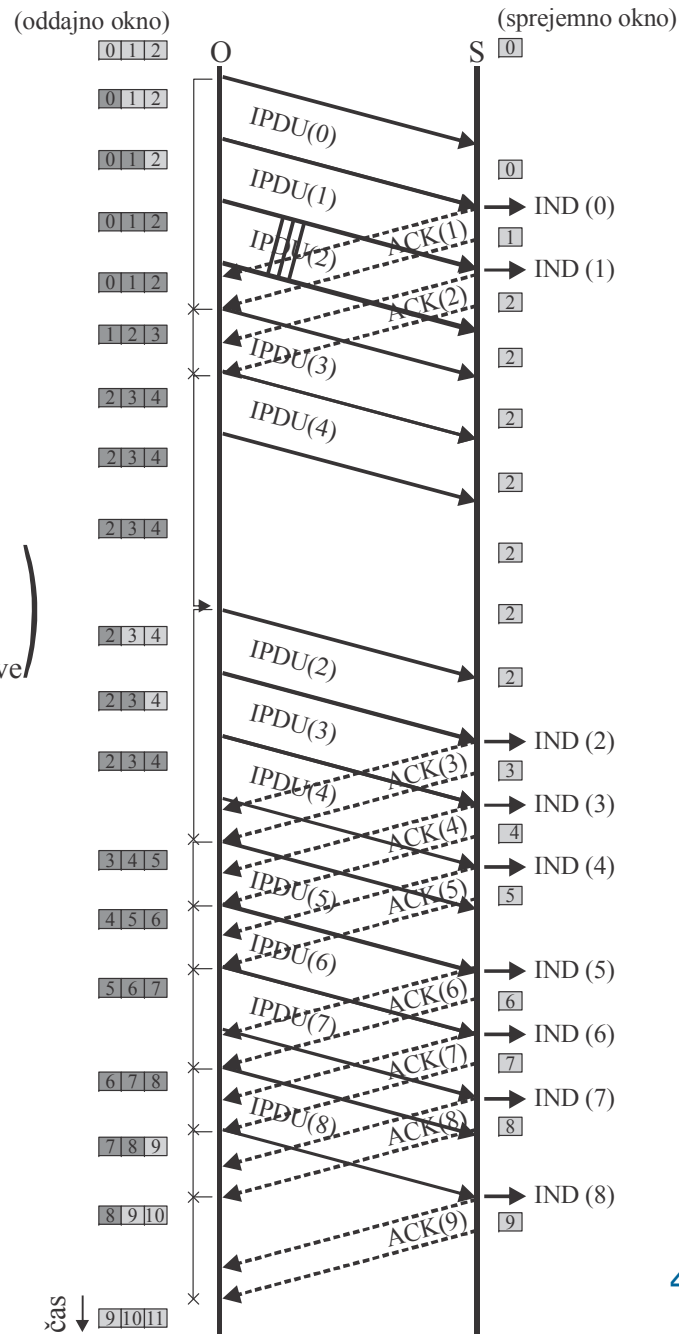
# Zgled: protokol s ponavljanjem zaporedja

(velika širina  
oddajnega okna,  
en časovnik,  
kumulativne potrditve)



# Zgled: protokol s ponavljanjem zaporedja

( $W_s=3$ ,  
 $W_r=1$ ,  
en časovnik,  
kumulativne potrditve)

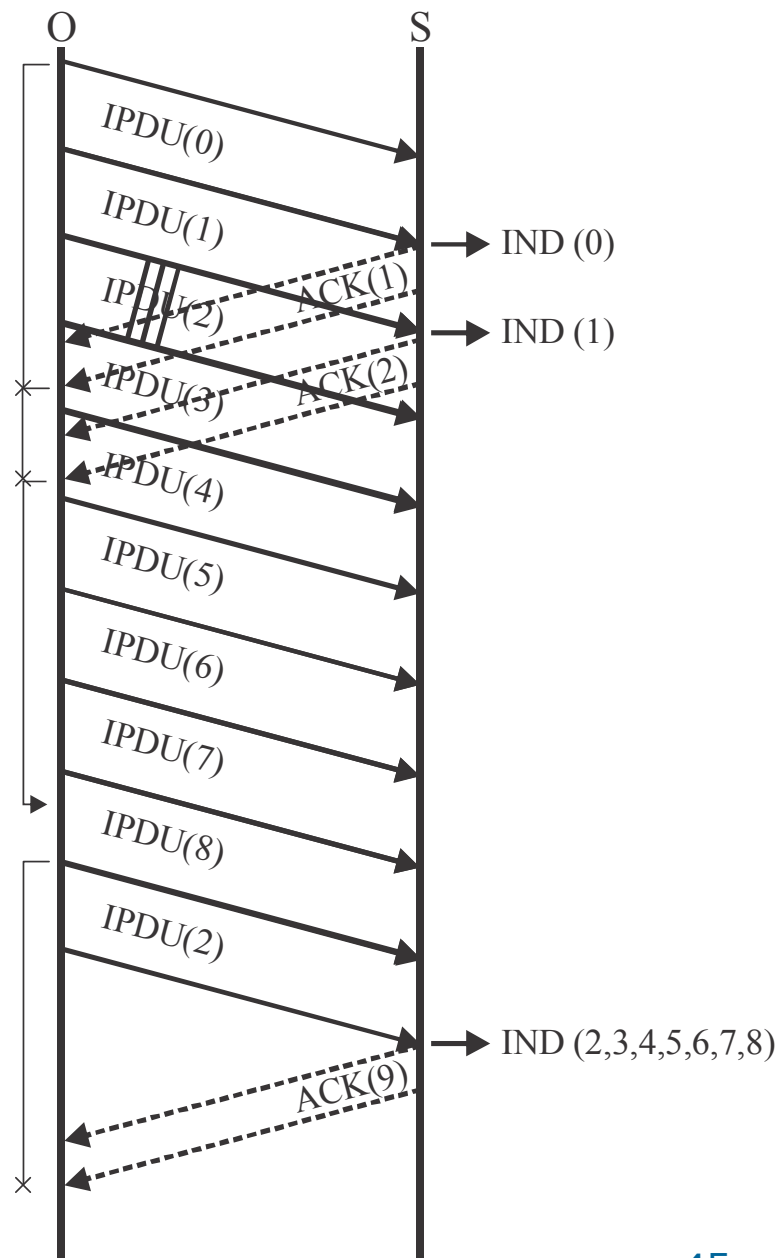


# Prisotnost izgub: protokol s selektivnim ponavljanjem

- ang. Selective-Repeat Protocol
- oddajnik ponovno pošlje le pogrešani paket
- sprejemnik shrani pakete, ki jih ne sprejme v pravem vrstnem redu, v sprejemnem vmesniku
- sprejemnik ne sprejema paketov v pravem vrsnem redu
- sprejemnik potrebuje spomin za urejanje paketov po vrsti,  $W_r > 1$

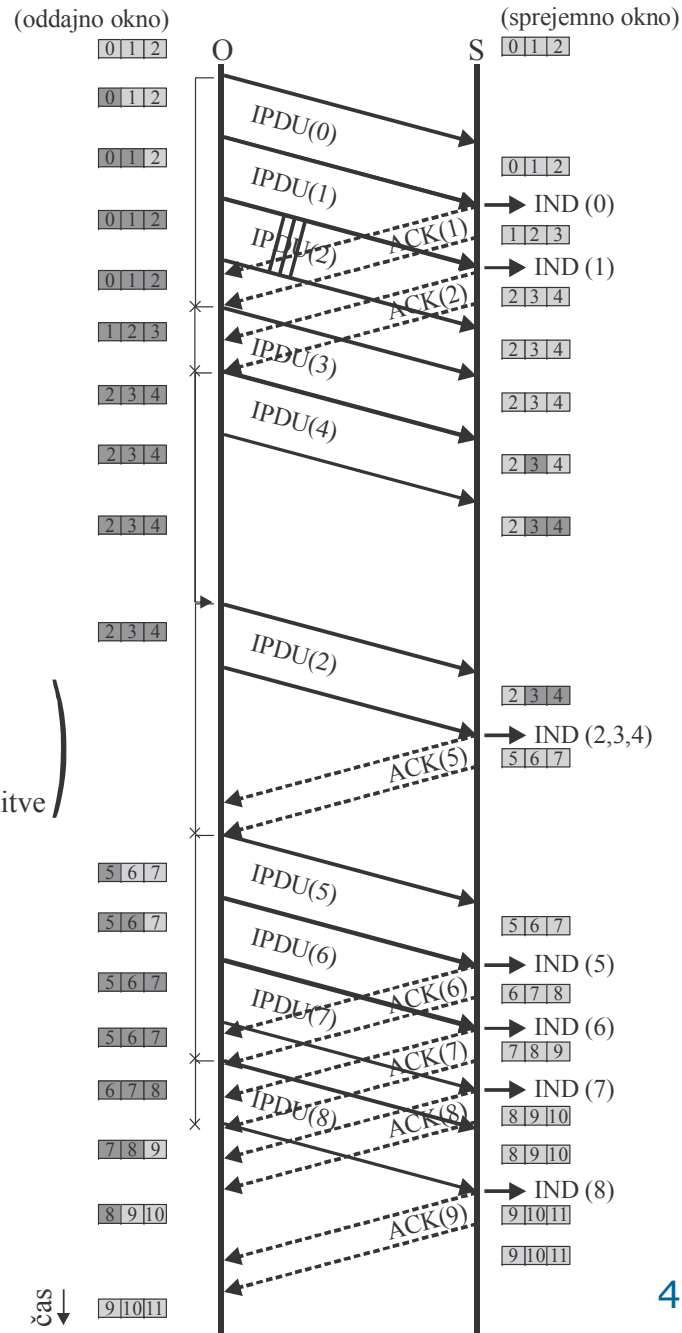
# Zgled: protokol s selektivnim ponavljanjem

( velika širina  
oddajnega okna, en časovnik,  
kumulativne potrditve )



# Zgled: protokol s selektivnim ponavljanjem

( $W_s=3,$   
 $W_r=3,$   
en časovnik,  
kumulativne potrditve)

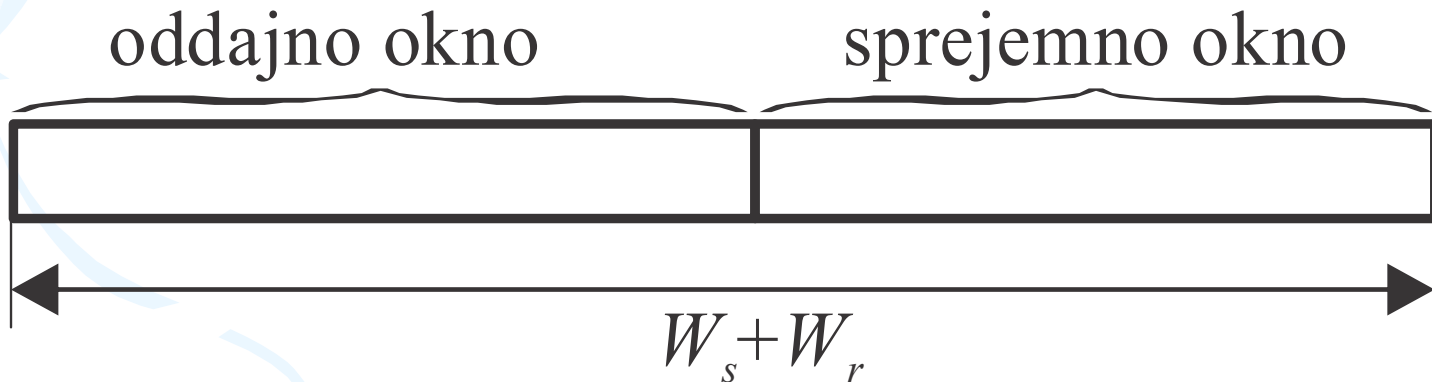


# Protokoli z drsečim oknom

protokol	$W_s$	$W_r$
protokol s čakanjem	1	1
protokol s ponavljanjem zaporedja	$W_s > 1$	1
protokol s selektivnim ponavljanjem	$W_s > 1$	$W_r = W_s$
posplošeni protokol	$W_s$	$W_r \leq W_s$

# Štetje po modulu

- Sekvenčne številke v spominu in v PDU
- Štetje po modulu  $M$ :  $0, 1, \dots, M-1, 0, \dots$
- Sekvenčna številka mora kadarkoli enoumno označevati katerikoli paket



$$M \geq W_s + W_r$$



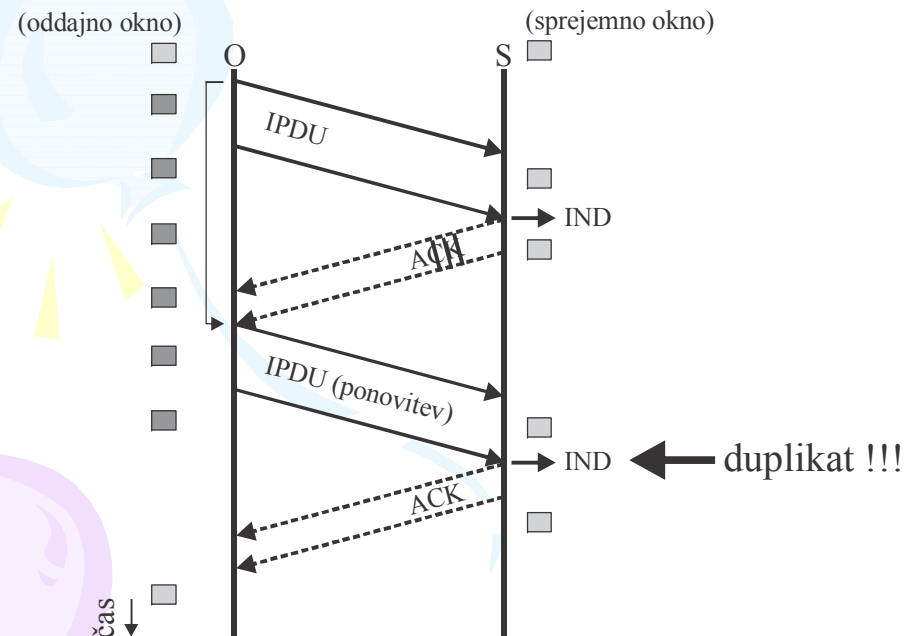
# Protokoli z drsečim oknom: modul štetja

protokol	$M \geq$
protokol s čakanjem (protokol z alternirajočim bitom)	2
protokol s ponavljanjem zaporedja	$W_s + 1$
protokol s selektivnim ponavljanjem	$2 \times W_s$
posplošeni protokol	$W_s + W_r$

# Protokol s čakanjem: napačno / pravilno delovanje

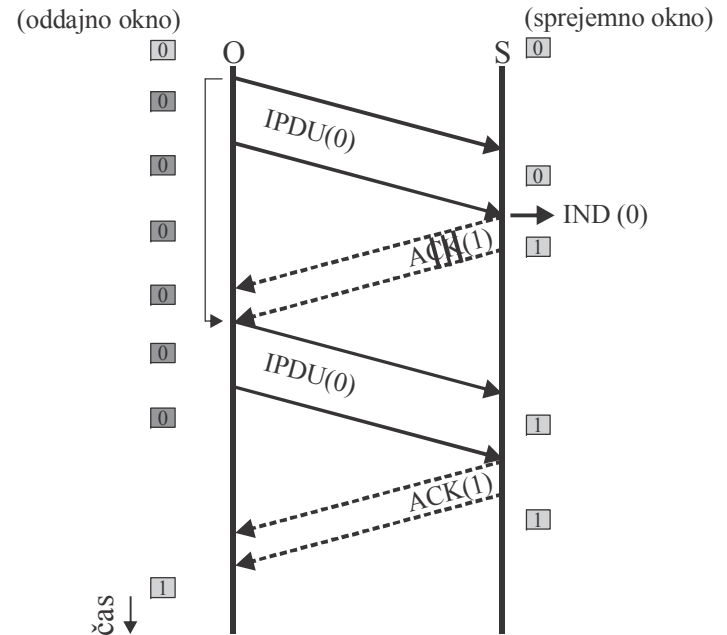
napačno:

$(W_s=1,$   
 $W_r=1,$   
 $M=1 \rightarrow$  štetja ni!  
en časovnik,  
kumulativne potrditve)



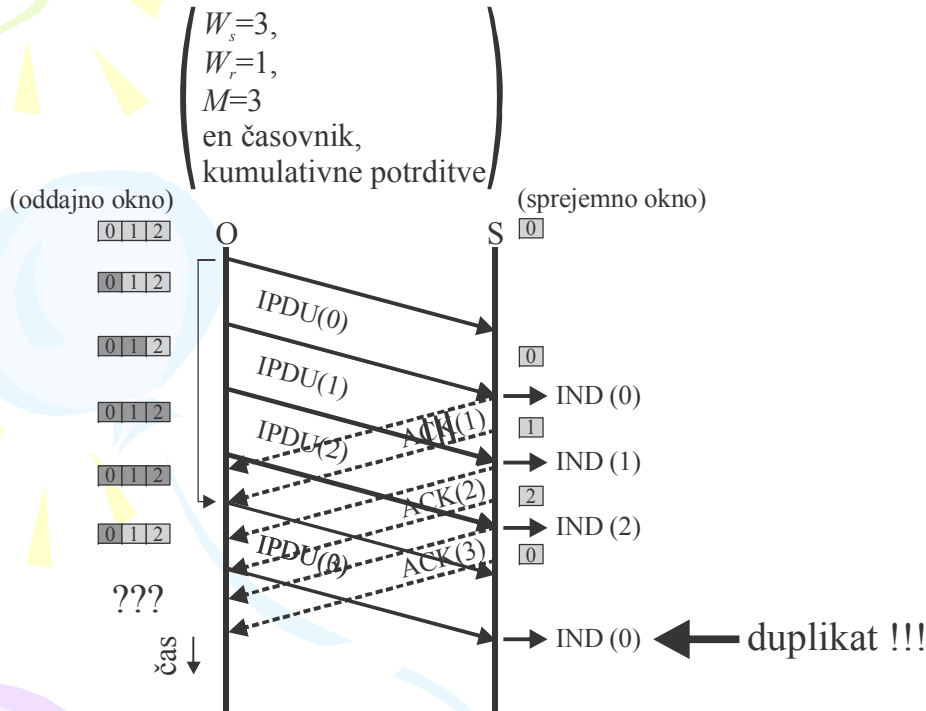
pravilno:

$(W_s=1,$   
 $W_r=1,$   
 $M=2$   
en časovnik,  
kumulativne potrditve)

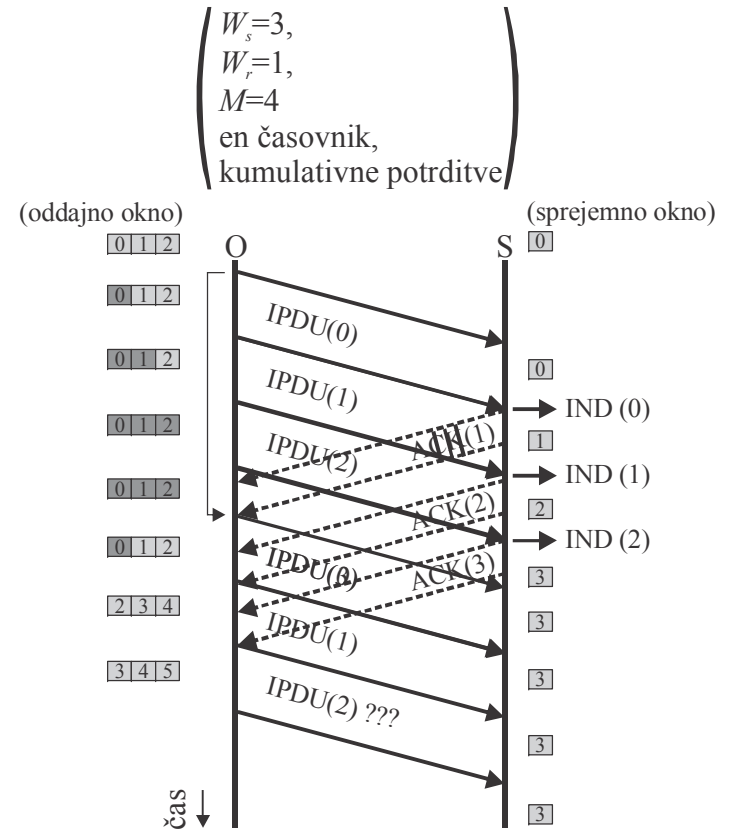


# Protokol s ponavljanjem zaporedja: napačno / pravilno delovanje

napačno:

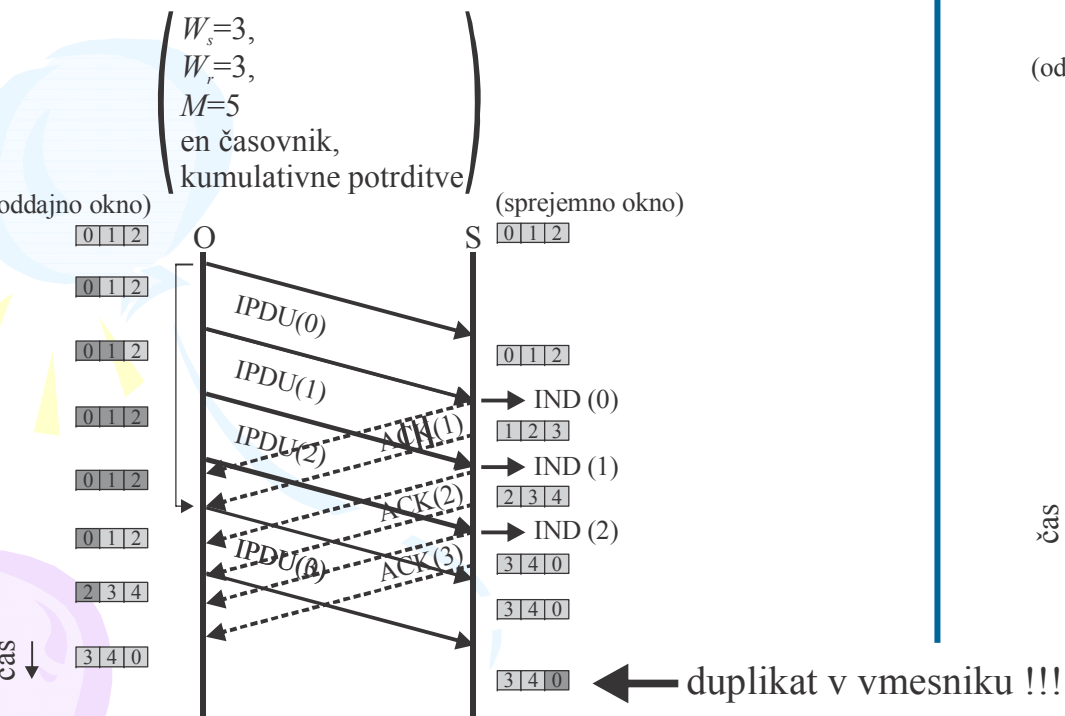


pravilno:

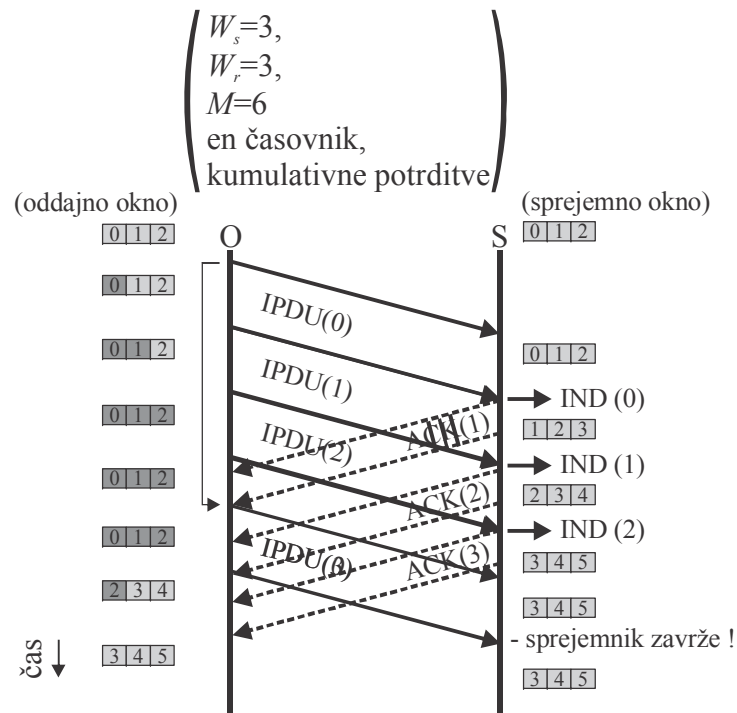


# Protokol s selektivnim ponavljanjem: napačno / pravilno delovanje

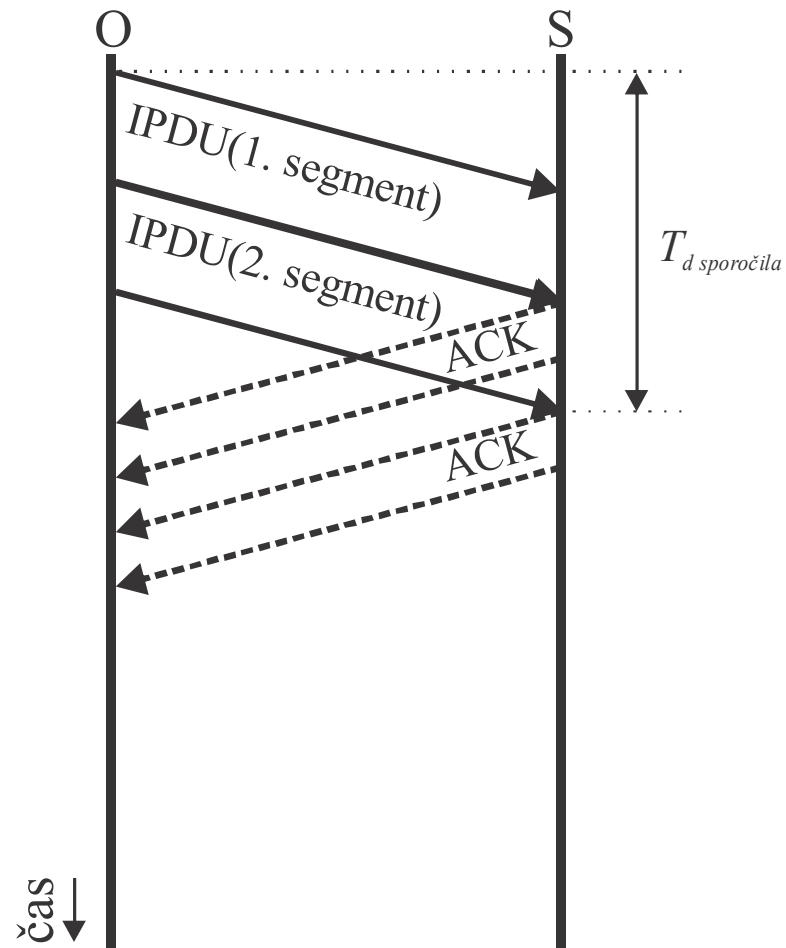
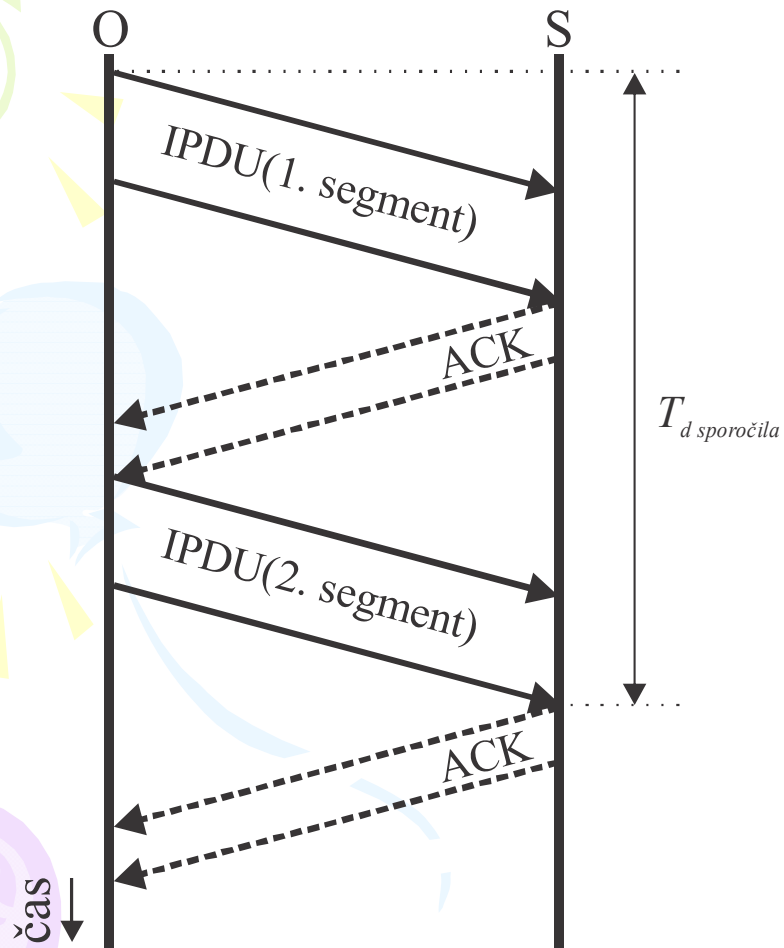
napačno:



pravilno:



# Protokol s čakanjem – protokol s tekočim pošiljanjem: zakasnitev sporočila, sestavljenega iz dveh segmentov



# Učinkovitost

- učinkovitost protokolov z drsečim oknom:
  - protokol s čakanjem
  - protokol s ponavljanjem zaporedja
  - protokol s selektivnim ponavljanjem
- učinkovitost protokola pada s pogostnostjo napak
- učinkovitost protokola pada z dolžino sporočil
- učinkovitost, ki jo vidita uporabnika, narašča s faktorjem režije
- faktor režije narašča z dolžino uporabniškega sporočila (pri konstantni režiji)
- obstaja optimalna dolžina sporočila (ko uporabnika vidita največjo učinkovitost)

# Učinkovitost protokolov z drsečim oknom

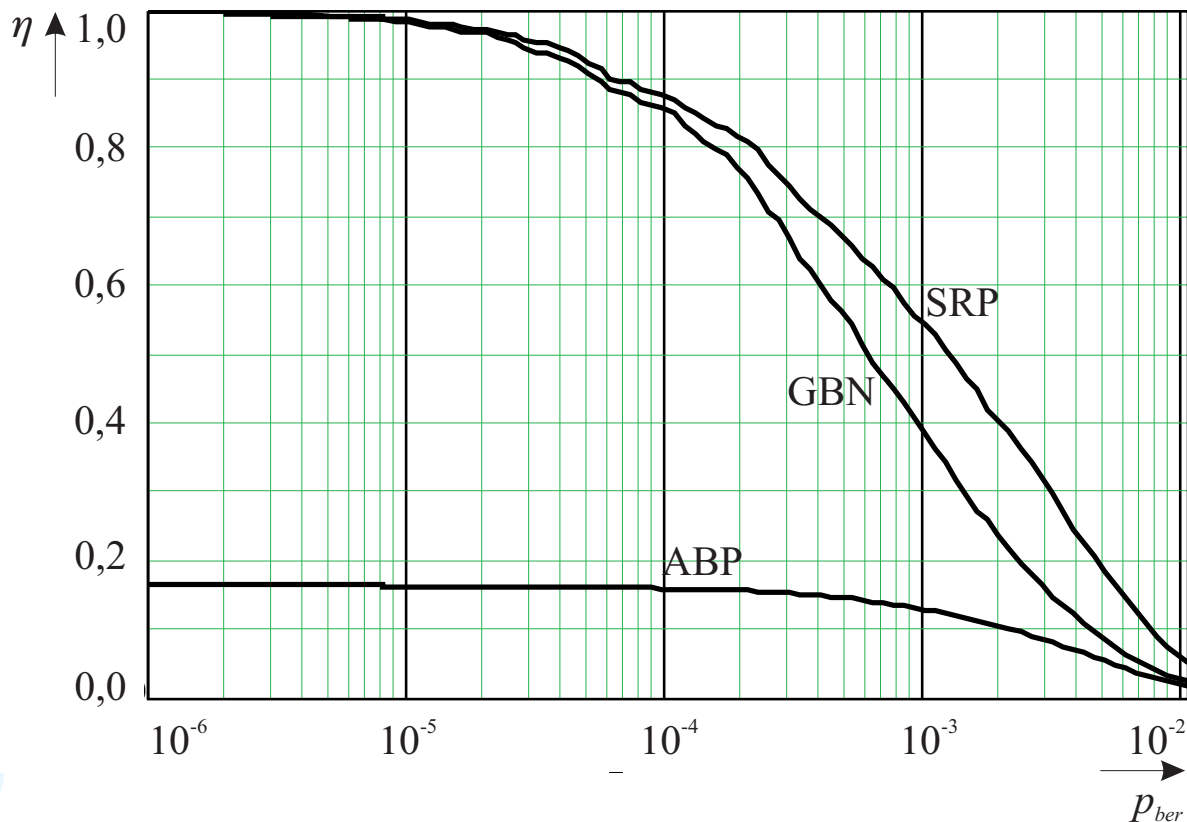
$$t_p = 500 \mu\text{s}$$

$$R = 1 \text{ Mb/s}$$

$$L_i = 200 \text{ b}$$

$$L_r = L_r = 40 \text{ b}$$

$$W = 7$$



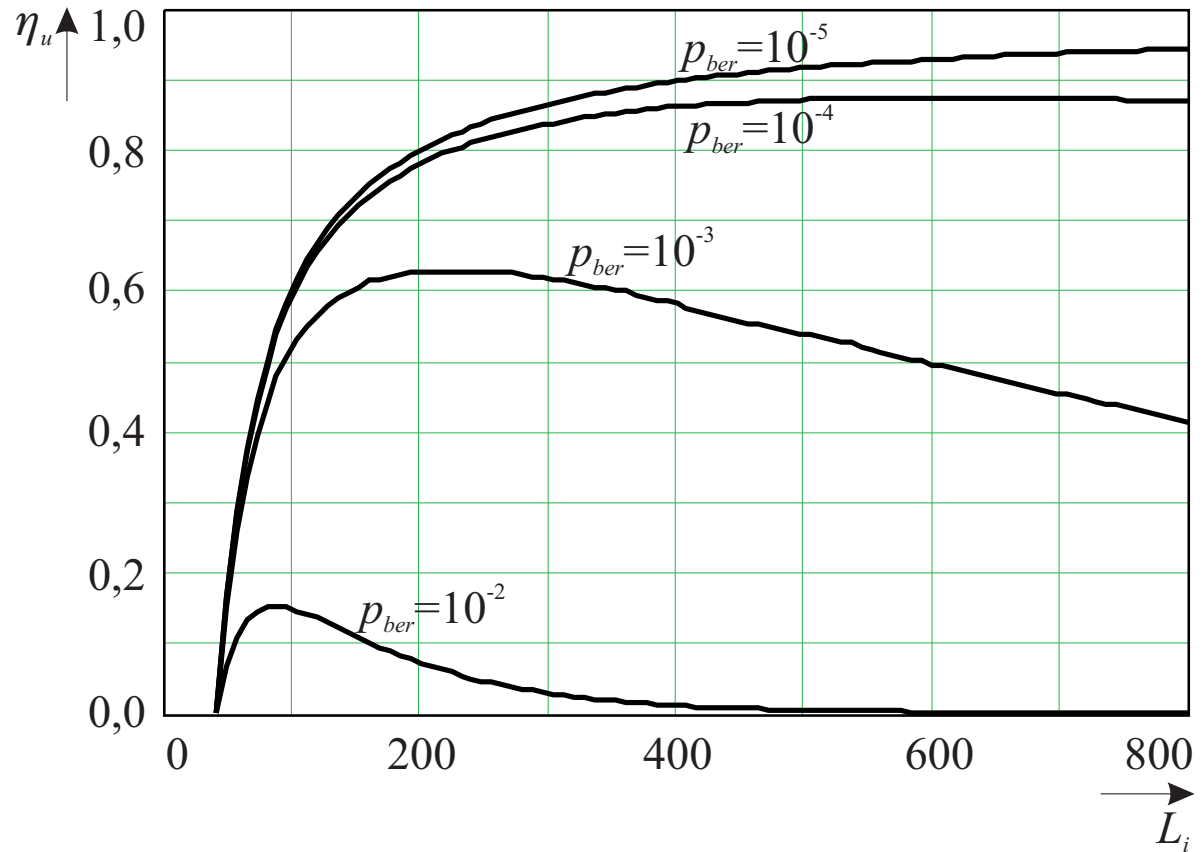
# Učinkovitost, ki jo vidita uporabnika (protokol s selektivnim ponavljanjem)

$$t_p = 500 \mu\text{s}$$

$$R = 1 \text{ Mb/s}$$

$$L_r = L_a = 40 \text{ b}$$

$$W = 100$$





# Dodatki za povečanje učinkovitosti

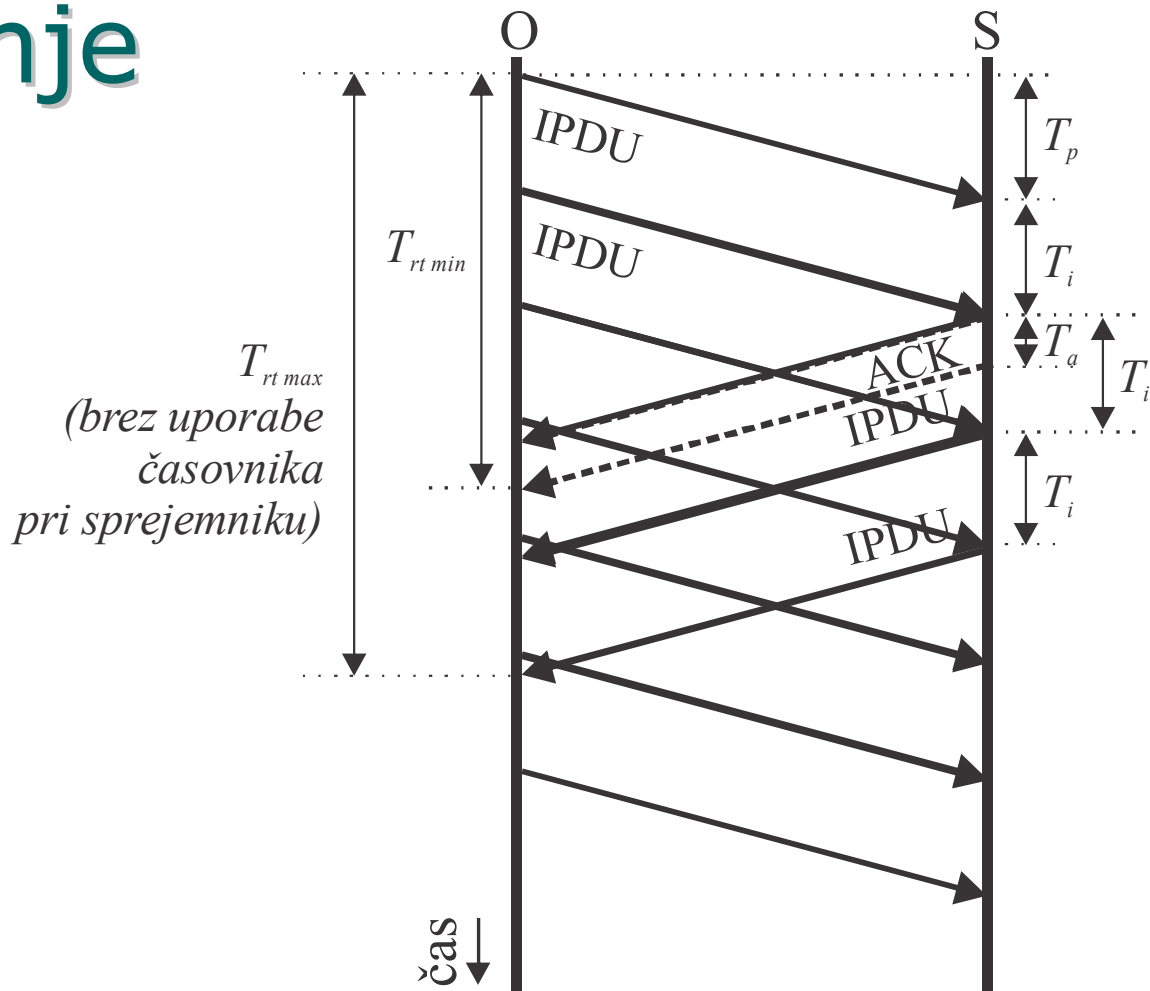
- negativne potrditve (NAK)
  - sprejemnik pošlje NAK
    - ko sprejme pokvarjeno sporočilo (če kanal ni multipleksen)
    - ko sprejme nepričakovano sporočilo
- poizvedbe (ENQ)
  - oddajnik pošlje ENQ
    - po izteku časovnika
    - pri krmiljenju pretoka

# Dvosmerni prenos uporabniške informacije

- protokol s čakanjem preko poldupleksnega kanala, izmenjava vloge oddajnik/sprejemnik
  - podobnost s protokoli MAC
  - s pozivanjem
  - z žetonom
- protokol s tekočim pošiljanjem preko dupleksnega kanala
  - natovarjanje (ang. piggybacking)

# Natovarjanje

- časovnik oddajnika
- časovnik sprejemnika
- čas izteka časovnikov



$$t_i + t_a + 2 \cdot t_p \leq t_{rt} \leq 2 \cdot t_i + 2 \cdot t_p + t_{tor} \quad , \quad t_{tor} \geq t_i$$

$$(t_{tor} = t_i \rightarrow t_{rt} \leq 3 \cdot t_i + 2 \cdot t_p)$$

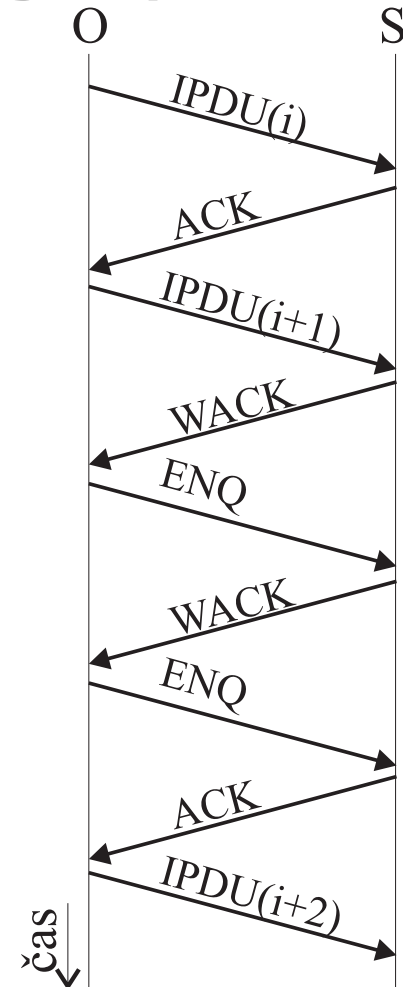
# Krmiljenje pretoka in zamašitev

- probleme zaradi neenakomernega prometnega pretoka preprečujemo s
  - čakalnimi vrstami (kratkotrajne preobremenitve)
  - krmiljenjem pretoka in zamašitev (dolgotrajne preobremenitve)
- krmiljenje pretoka: med terminali
- krmiljenje zamašitev: v omrežju
- krmiljenje
  - s povratnim vplivom (sprejemnika na oddajnik)
    - eksplicitno
    - implicitno
  - brez povratnega vpliva

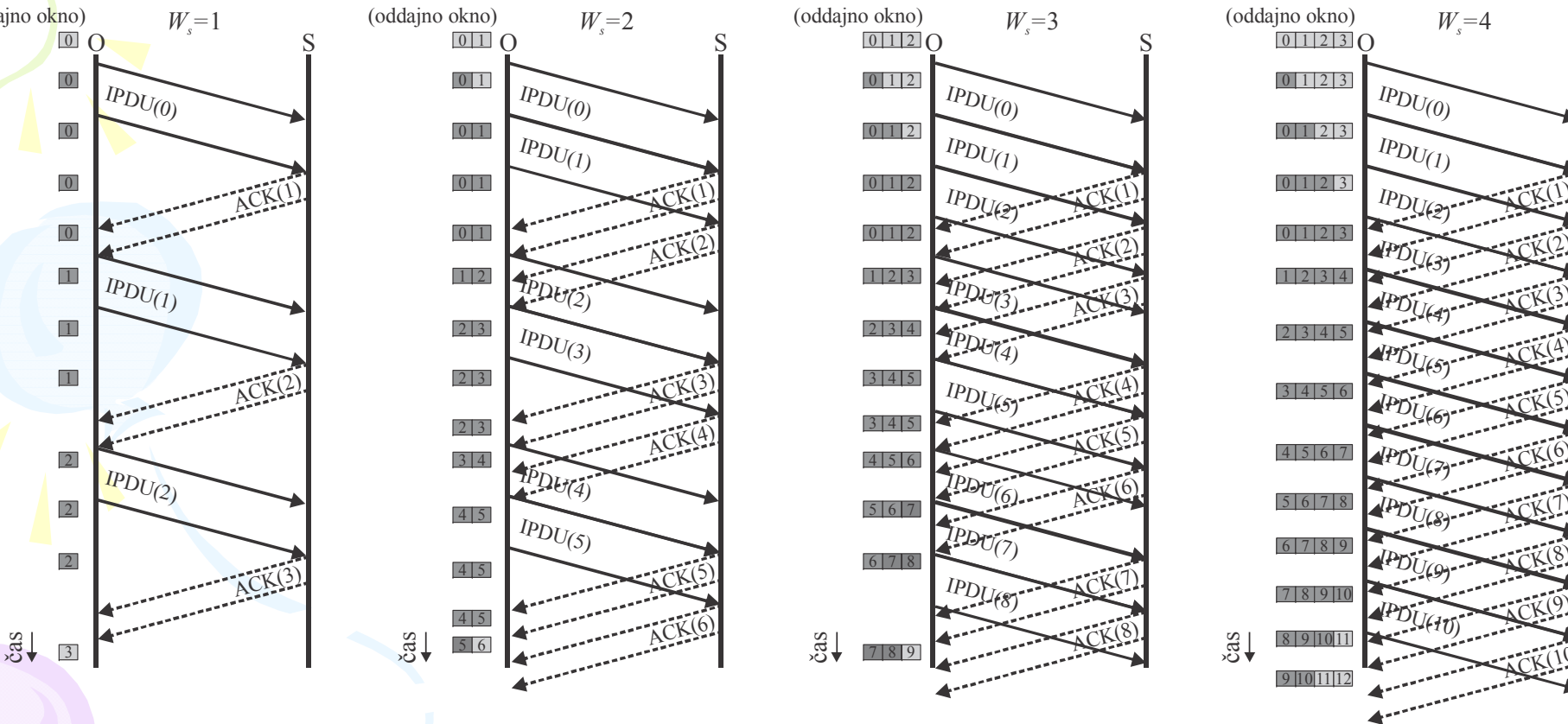
# Protokoli s ponavljanjem in metode krmiljenja

- XON/XOFF (zgodovina)
  - eksplicitno krmiljenje
- zahteva po zaustavitvi oddajanja
  - eksplicitno krmiljenje
- krmiljenje s širino okna
  - odvisnost hitrosti prenosa od širine okna
  - eksplicitno krmiljenje
  - implicitno krmiljenje

# Primer: protokol s čakanjem in eksplicitno krmiljenje pretoka



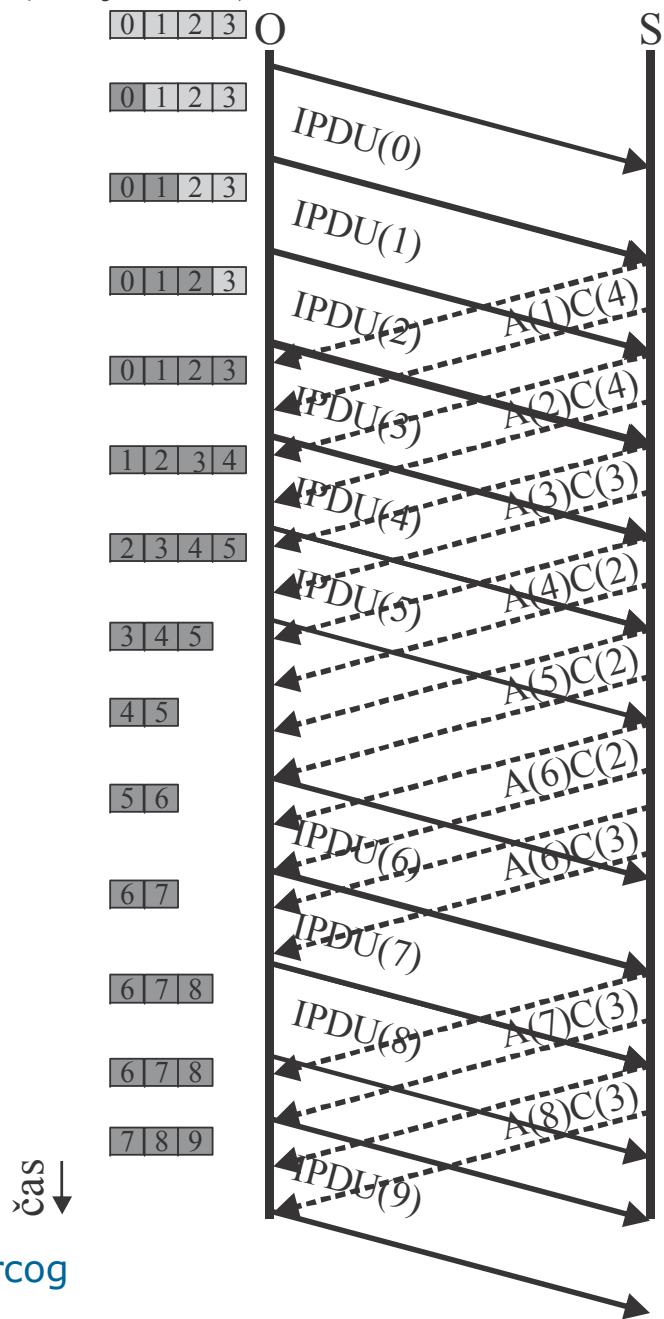
# Odvisnost hitrosti prenosa od širine oddajnega okna



# Protokol z eksplicitnim krmiljenjem in spremenljivo širino okna

- potrditev
- kredit

(oddajno okno)





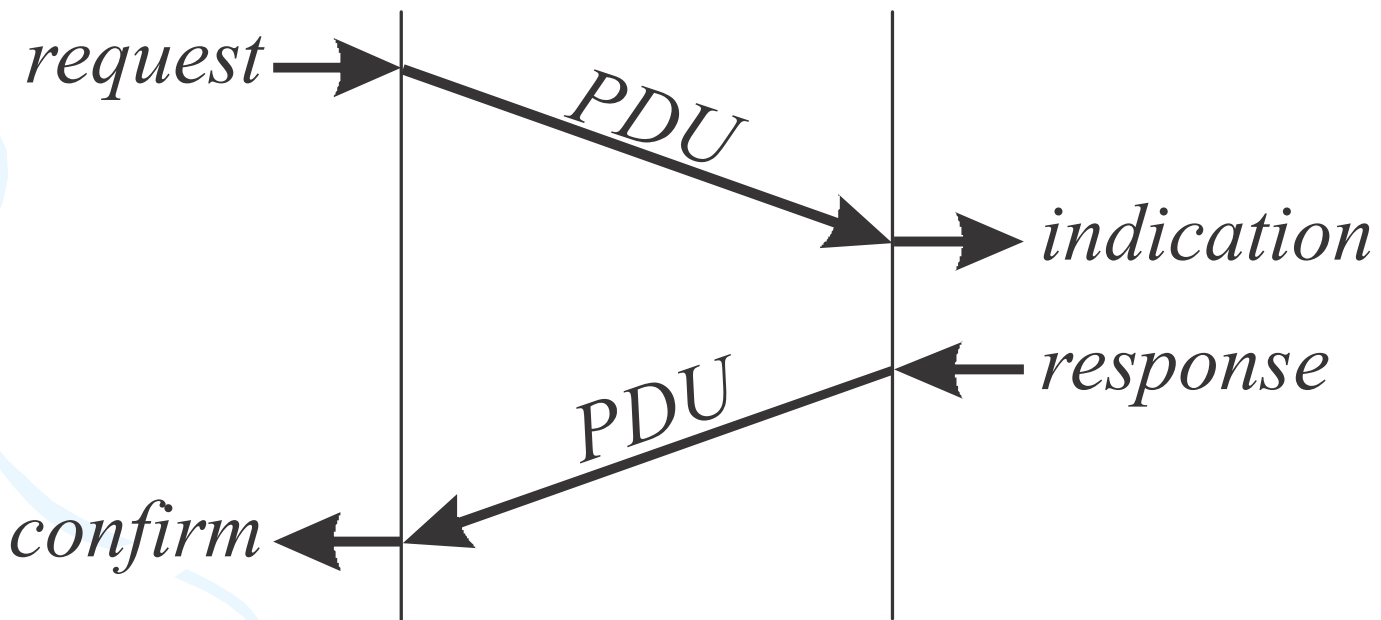
# Protokol z implicitnim krmiljenjem in spremenljivo širino okna

- oddajnik prilagaja širino oddajnega okna razmeram v omrežju
- pokazatelji razmer v omrežju
  - pozitivne in negativne potrditve
  - iztek časovnikov
  - izmerjeni čas do potrditve
  - ...
  - (ne)zanesljivost pokazateljev ???

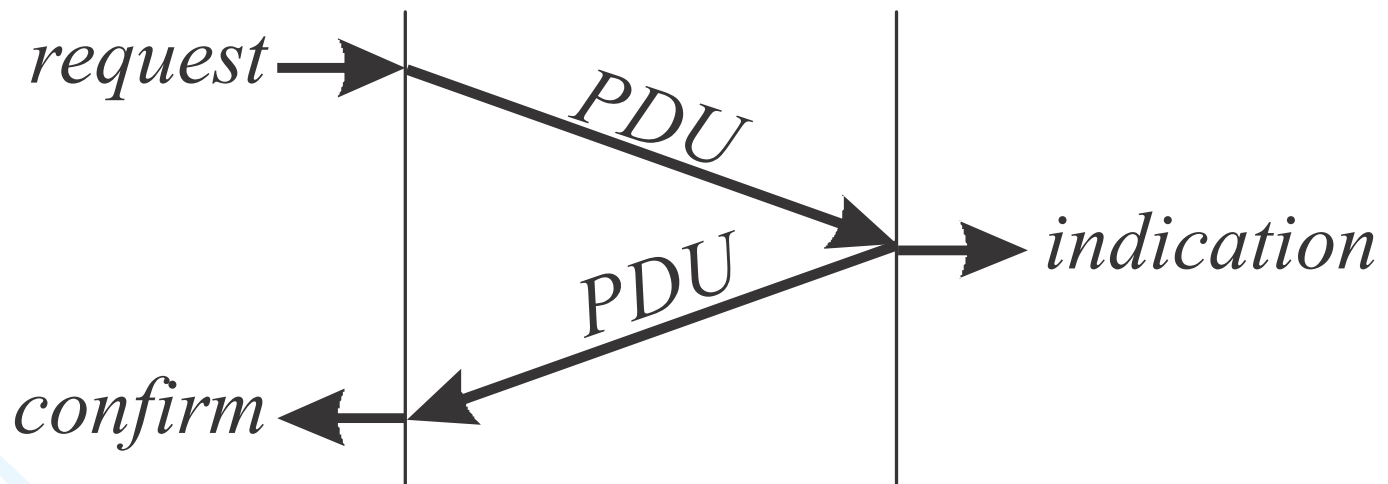
# Usklajevanje stanja

- vzpostavljanje zveze
- vrževanje zveze
- sproščanje zveze
- pogajanja o parametrih stanja
- določitev sinhronizacijskih točk
- ...

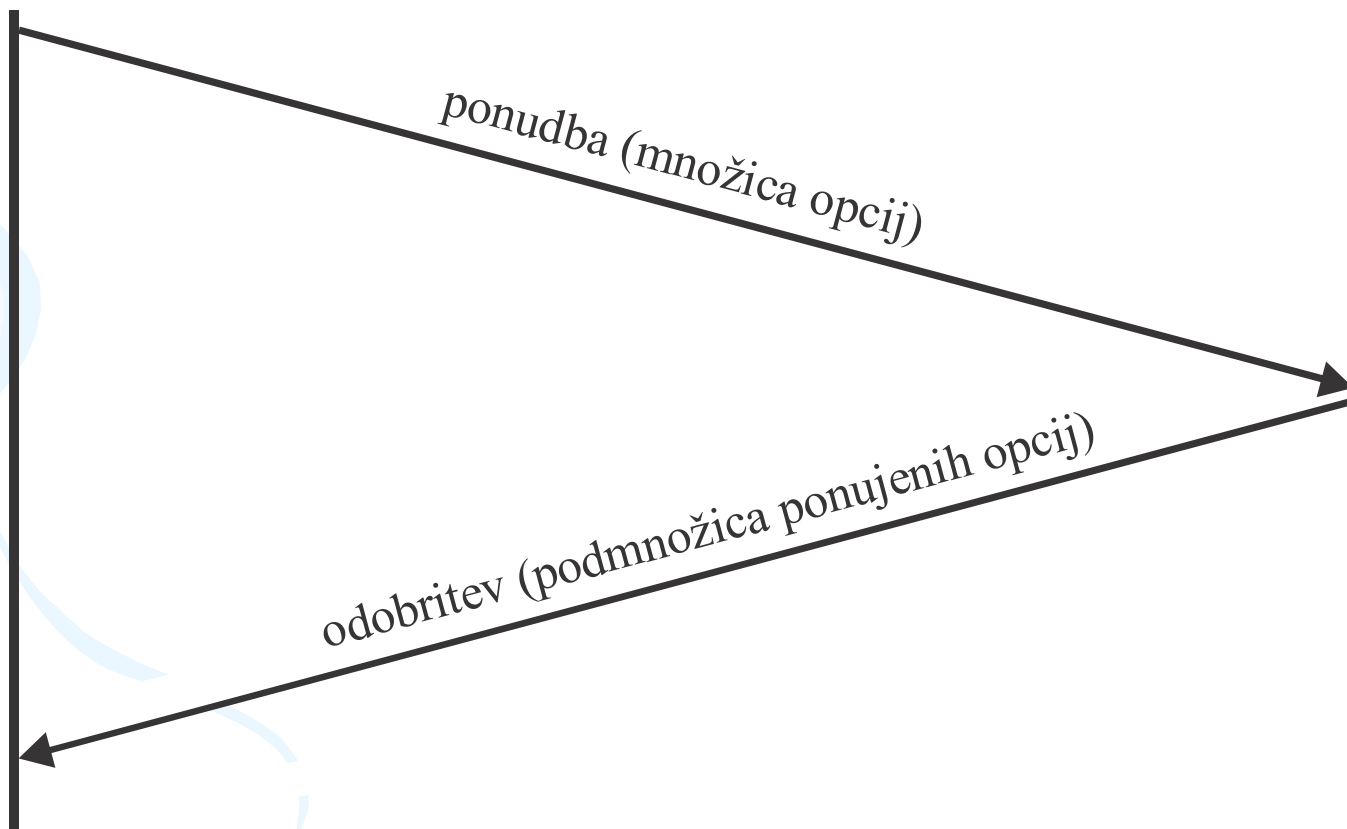
# Dvojni dogovor (štirje primitivi)



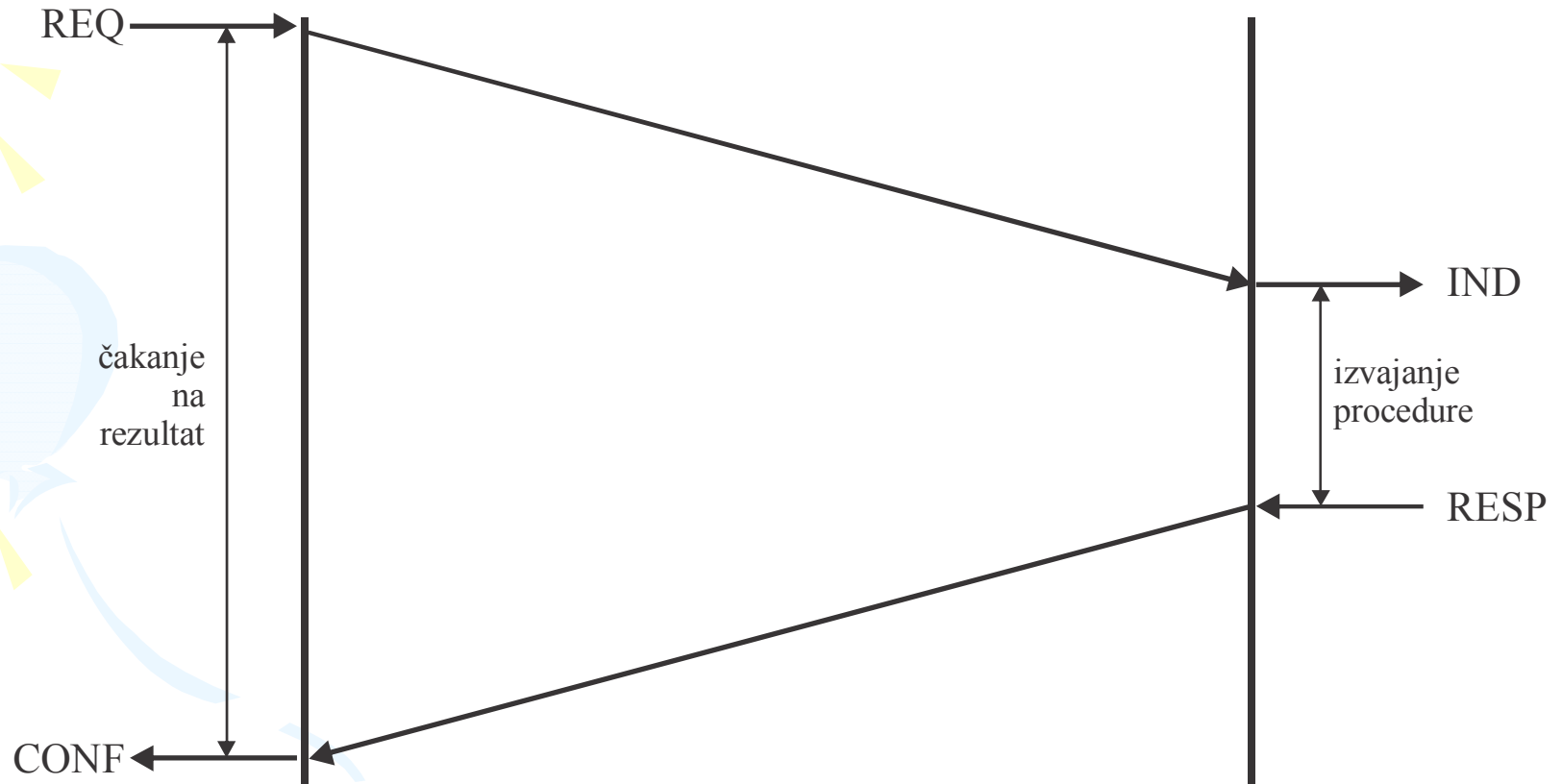
# Dvojni dogovor (trije primitivi)



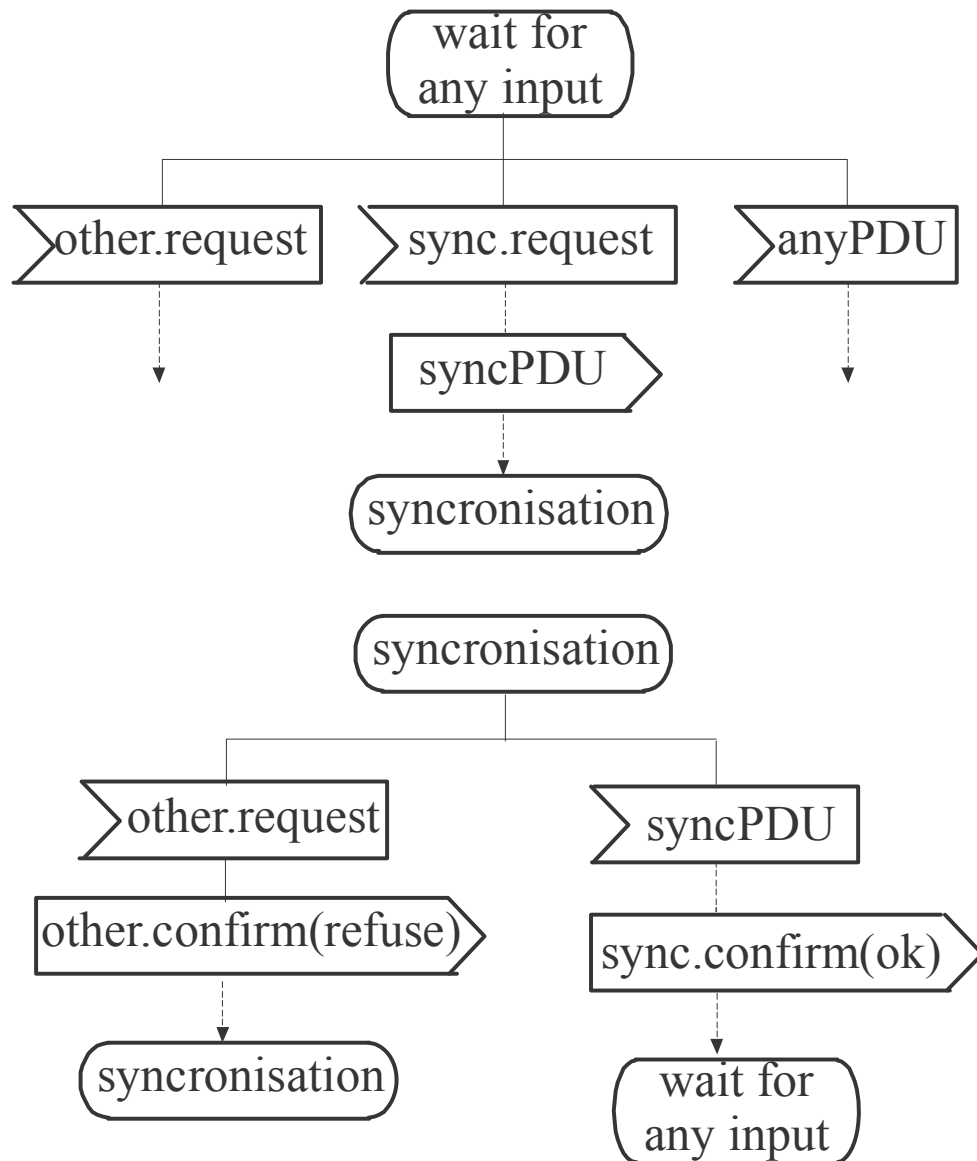
# Pogajanja o parametrih



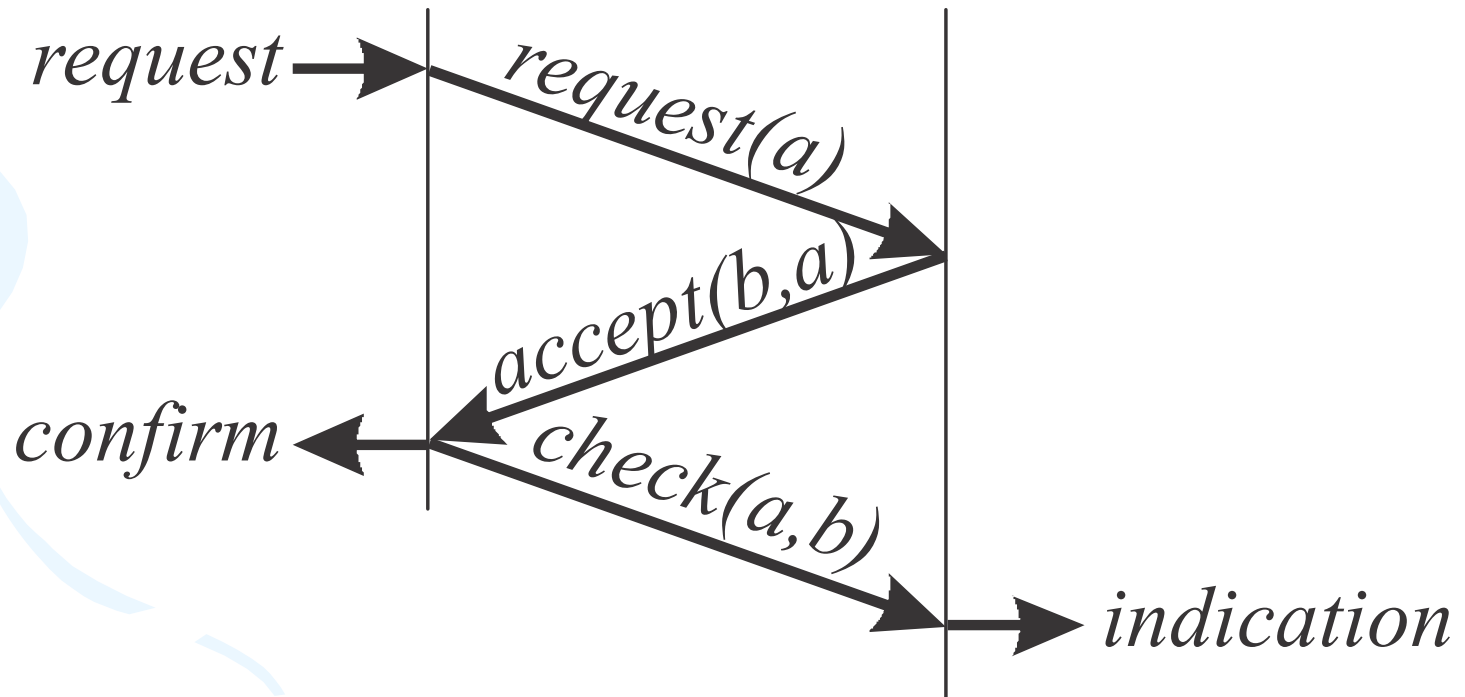
# Elementarna uskladitev



# Elementarna uskladitev – primer sinhronizacije stanja



# Upravljanje zveze – trojni dogovor



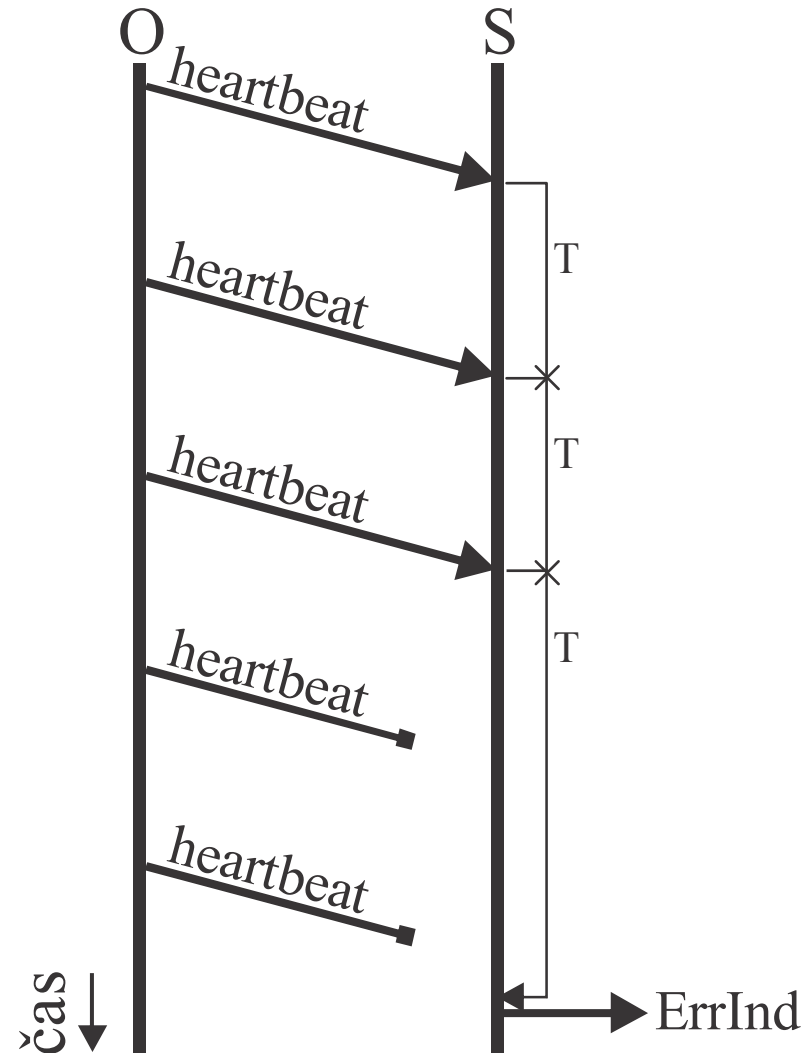


# Problem mrtvih duš

- kaj so mrtve duše (ang. floating corpses)
- problem pri nepovezavno orientiranem prenosu
- mrtve duše so posledica
  - nestabilnega delovanja usmerjevalnikov
  - nestabilnega usmerjanja
- problemi, ki jih mrtve duše povzročajo
- reševanje problemov
  - življenjska doba
  - trojni dogovor

# Preverjanje, ali je zveza še uporabna

- robustnost protokola
- v času aktivnosti
  - izteki časovnikov
- v času neaktivnosti
  - sporočila "heartbeat"
  - časovnik "keepalive"



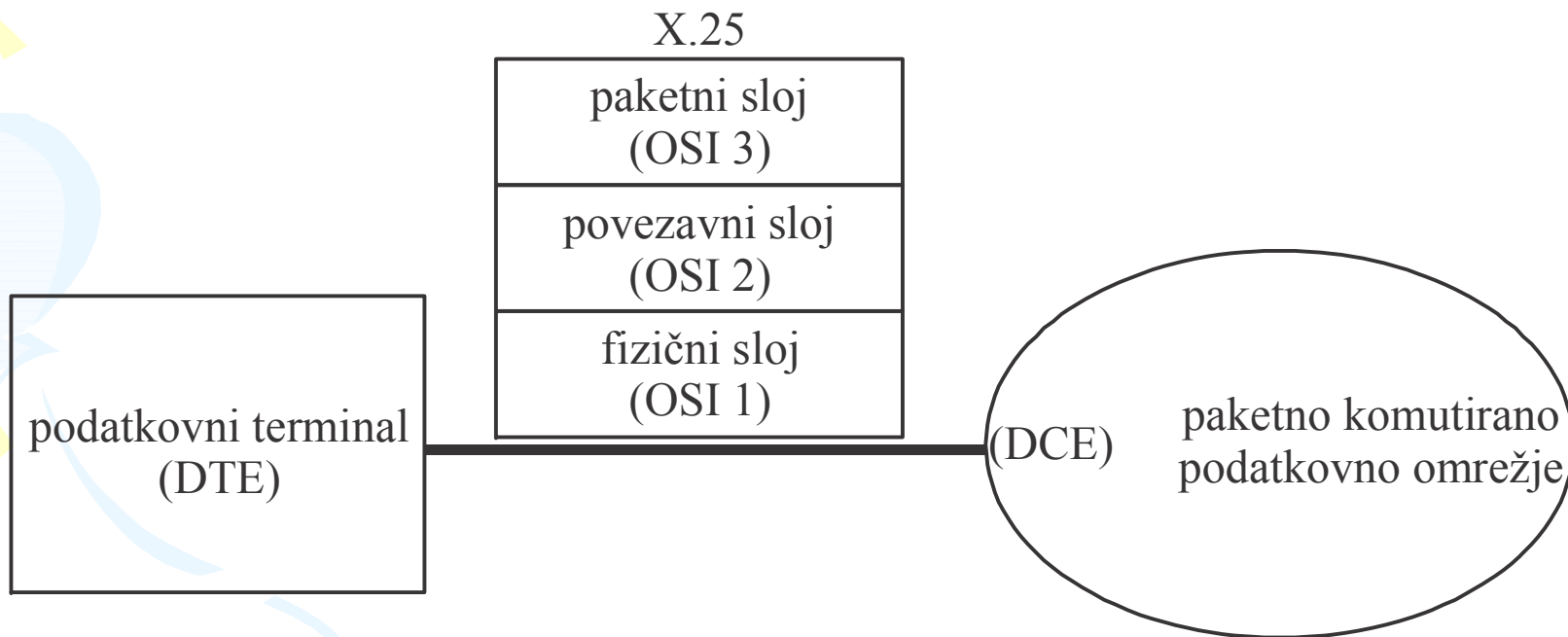
# Protokoli družine HDLC

- HDLC (High-level Data Link Control)
  - ISO
  - povezavni sloj
  - uravnotežen, neuravnotežen način
  - povezavno/nepovezavno orientiran prenos
  - protokol s ponavljanjem zaporedja / selektivnim ponavljanjem
- LAPB (Link Access Procedure – Balanced)
  - ITU-T
  - povezavni sloj, dostop do omrežja X.25
  - uravnotežen način
  - povezavno orientiran prenos
  - protokol s ponavljanjem zaporedja
- LAPD (Link Access Procedure on D channel)
  - ITU-T
  - povezavni sloj, dostop ISDN terminala do omrežja preko kanala D
  - uravnotežen način
  - povezavno orientiran prenos
  - protokol s ponavljanjem zaporedja
- LAPF (Link Access Procedure on Frame relay)
  - ITU-T
  - povezavni sloj, omrežje z blokvnim posredovanjem (Frame Relay)
  - uravnotežen način
  - ne zagotavlja zanesljivega prenosa
- LAPDm (protocol LAPD on Mobile channel)
  - ETSI, 3GPP
  - povezavni sloj, dostop GSM terminala do omrežja (signalizacija, podatki)
  - uravnotežen način (?)
  - povezavno orientiran prenos, lahko tudi nepovezavno
  - protokol s ponavljanjem zaporedja

# Protokol LAPB

- Link Acces Procedure – Balanced
- CCITT / ITU-T
- dostop do omrežja X.25
- povezavni sloj
- uravnotežen način
- povezavno orientiran prenos
- protokol s ponavljanjem zaporedja

# Dostop do omrežja X.25



# Protokol LAPB – tipi okvirjev

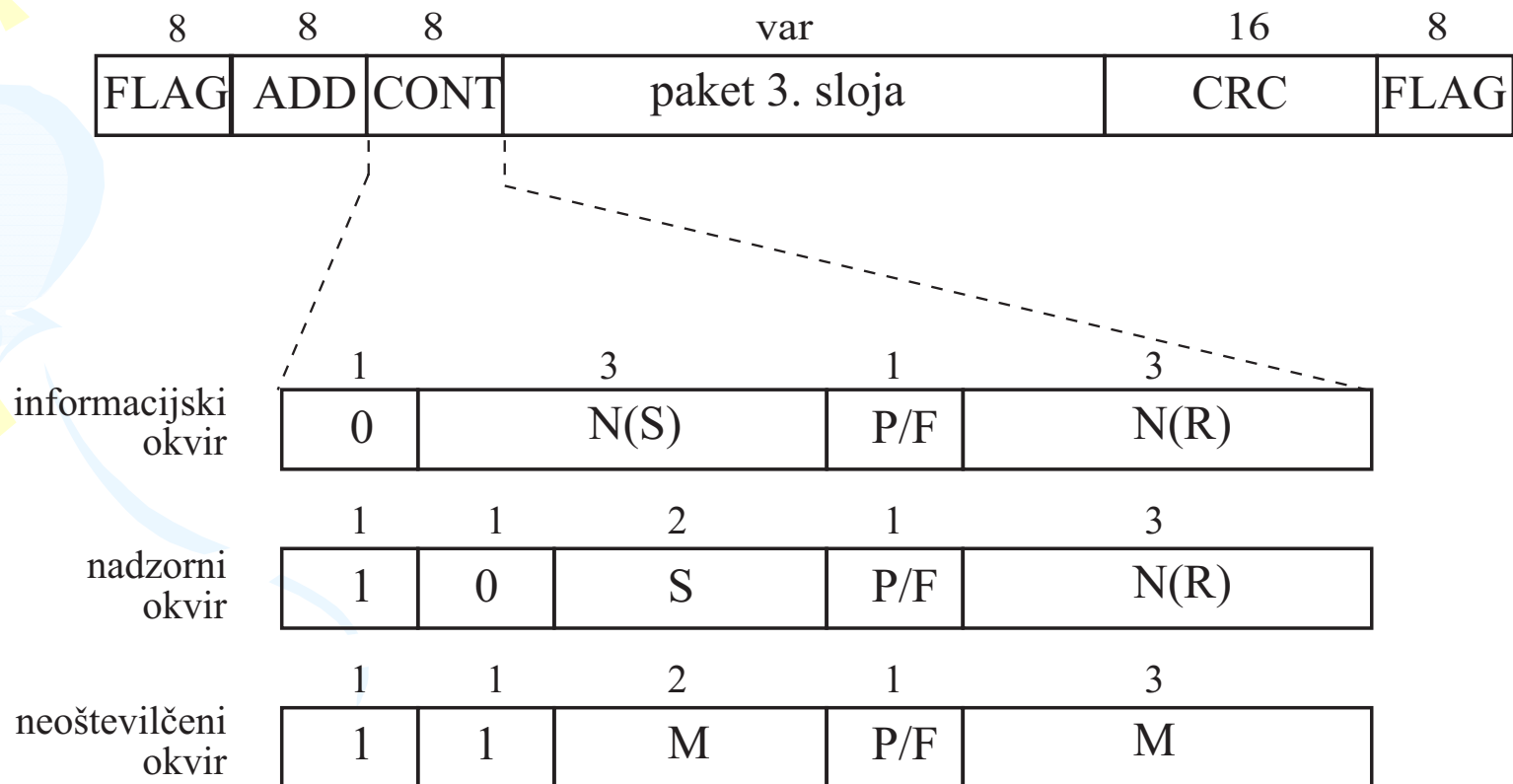
- neoštevilčeni (U-frame)
  - SABM (Set Asynchronous Balanced Mode)
  - DISC (DISConnect)
  - UA (Unnumbered Acknowledge)
  - DM (Disconnected Mode)
  - FRMR (FRaMe Reject)
- informacijski (I-frame)
- nadzorni (S-frame)
  - RR (Receive Ready)
  - REJ (REJect)
  - RNR (Receive Not Ready)

krmiljenje zveze

prenos uporabniške  
informacije

# LAPB - format okvirja

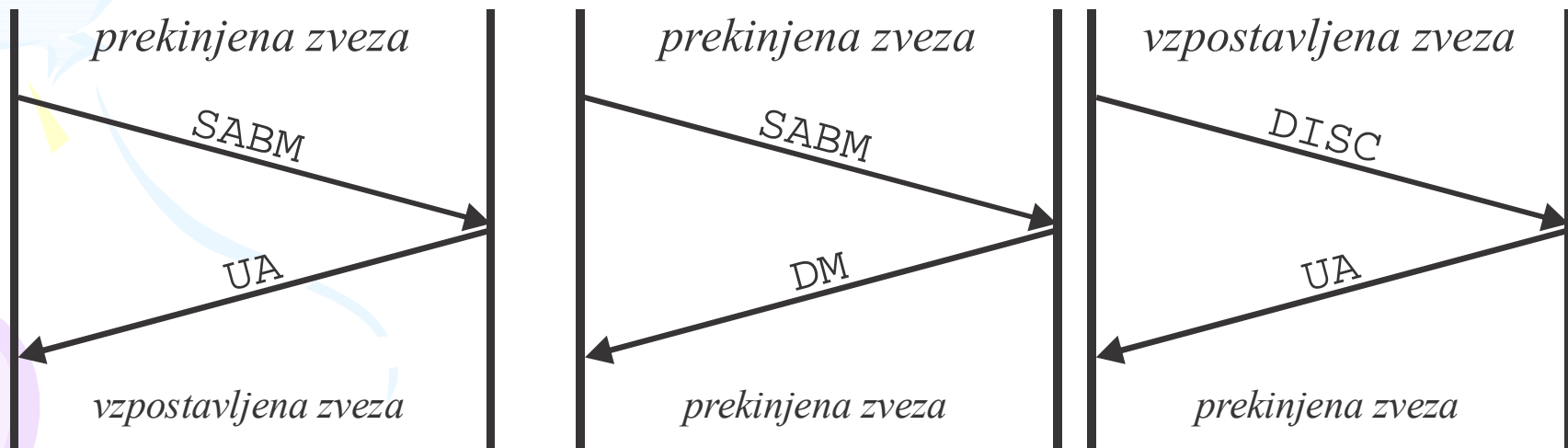
- Mirovno stanje na liniji: same zastavice
- Transparentni prenos: vrivanje ničel



# LAPB - upravljanje zveze

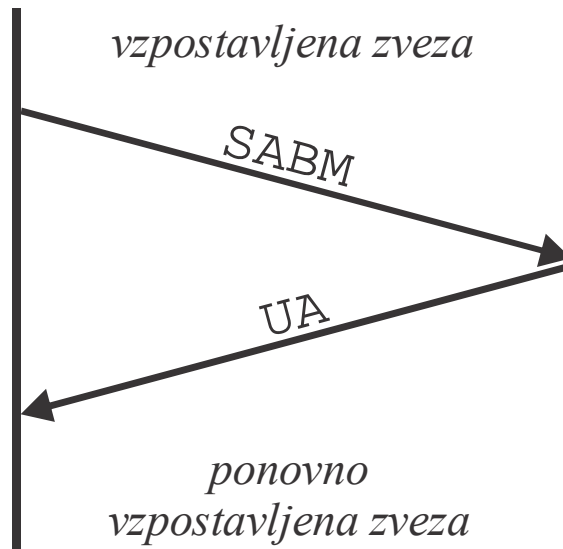
- Stanje brez vzpostavljene zveze: osebka oddajata same zastavice (stanje pripravljenosti)
- Stanje vzpostavljene zveze: kadar osebka ne oddajata okvirjev, oddajata same zastavice

## Vzpostavljanje in sproščanje zveze:



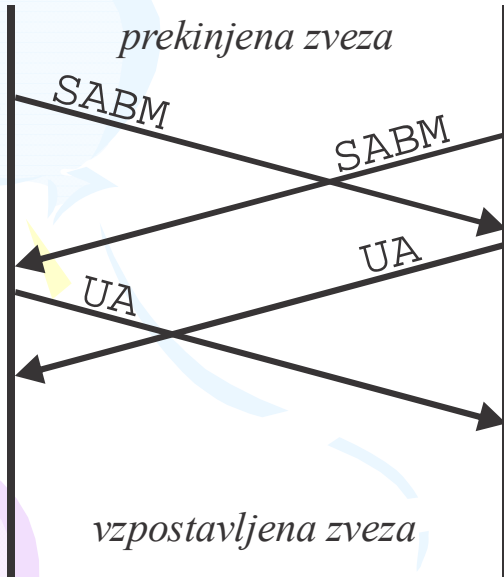


# LAPB – ponovna vzpostavitev zveze (reset)

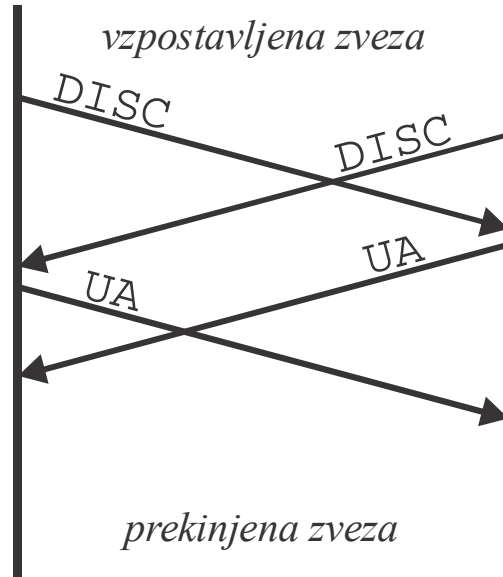


# LAPB - upravljanje zveze (reševanje kolizij)

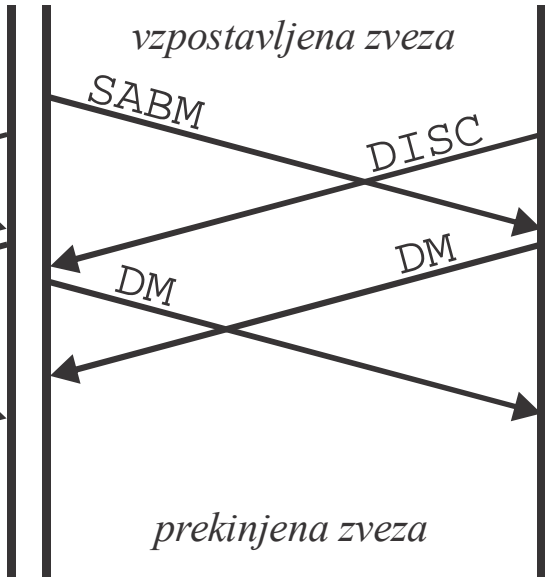
kolizija enakih sporočil



kolizija enakih sporočil



kolizija različnih sporočil



# LAPB - primer delovanja v fazi prenosa

- $W_s \geq 5$

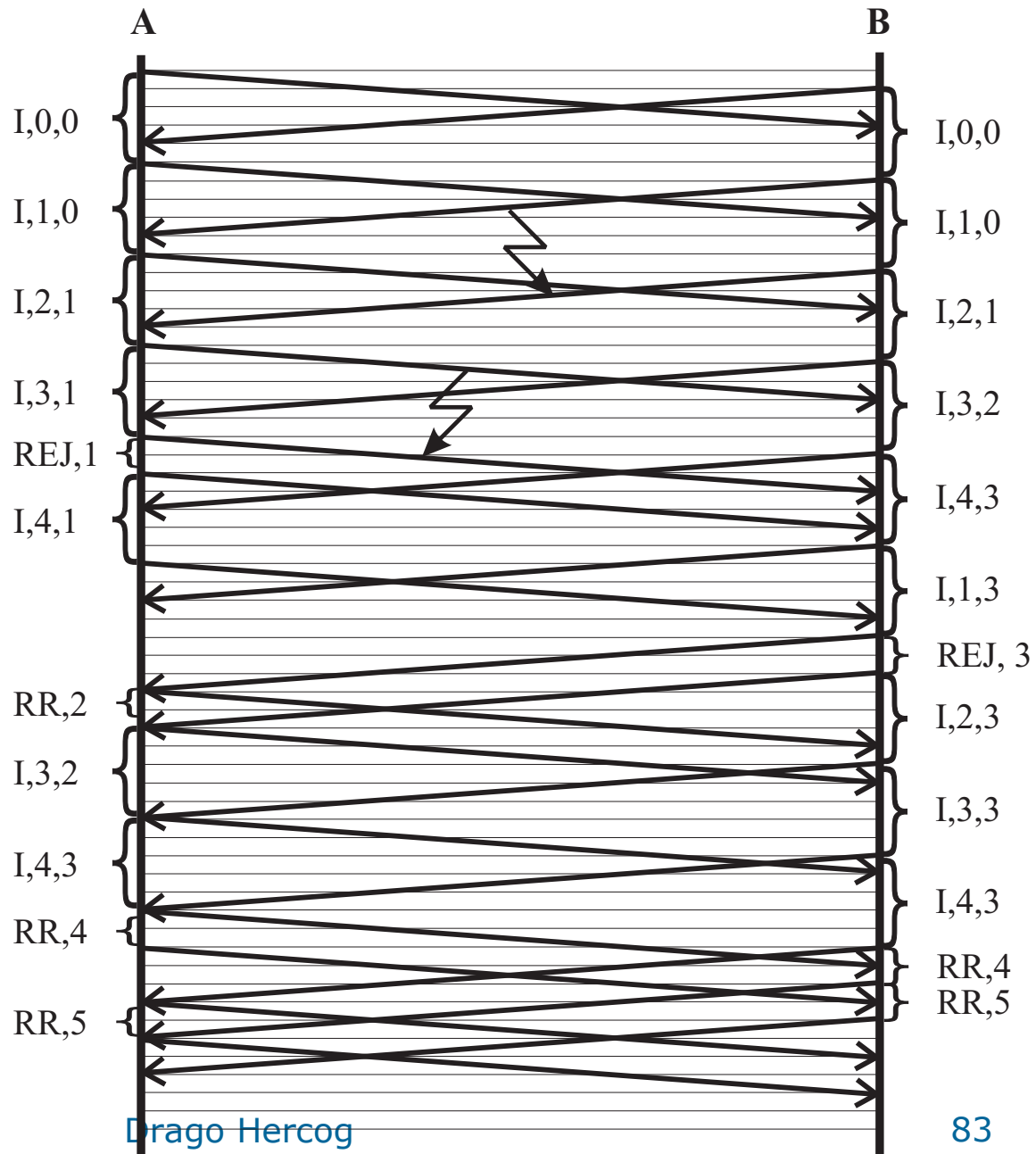
- v vsako stran se prenese po pet I okvirjev

Format okvirjev:

- $I, N(S), N(R)$

- $REJ, N(R)$

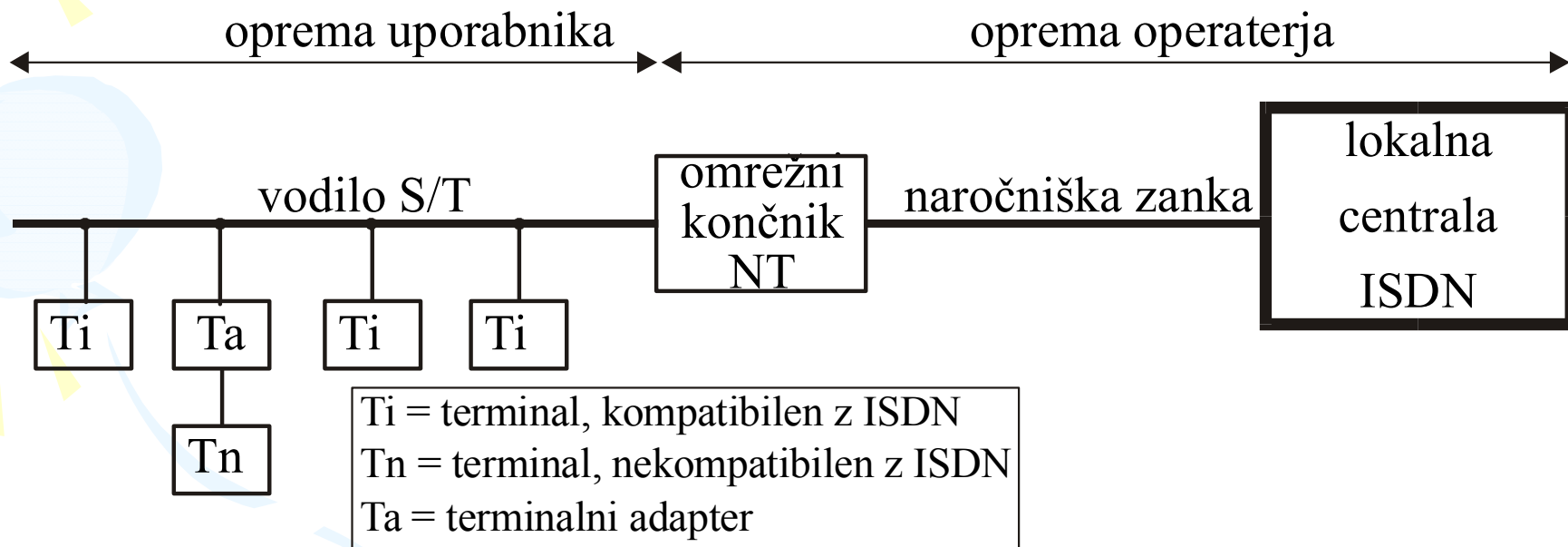
- $RR, N(R)$



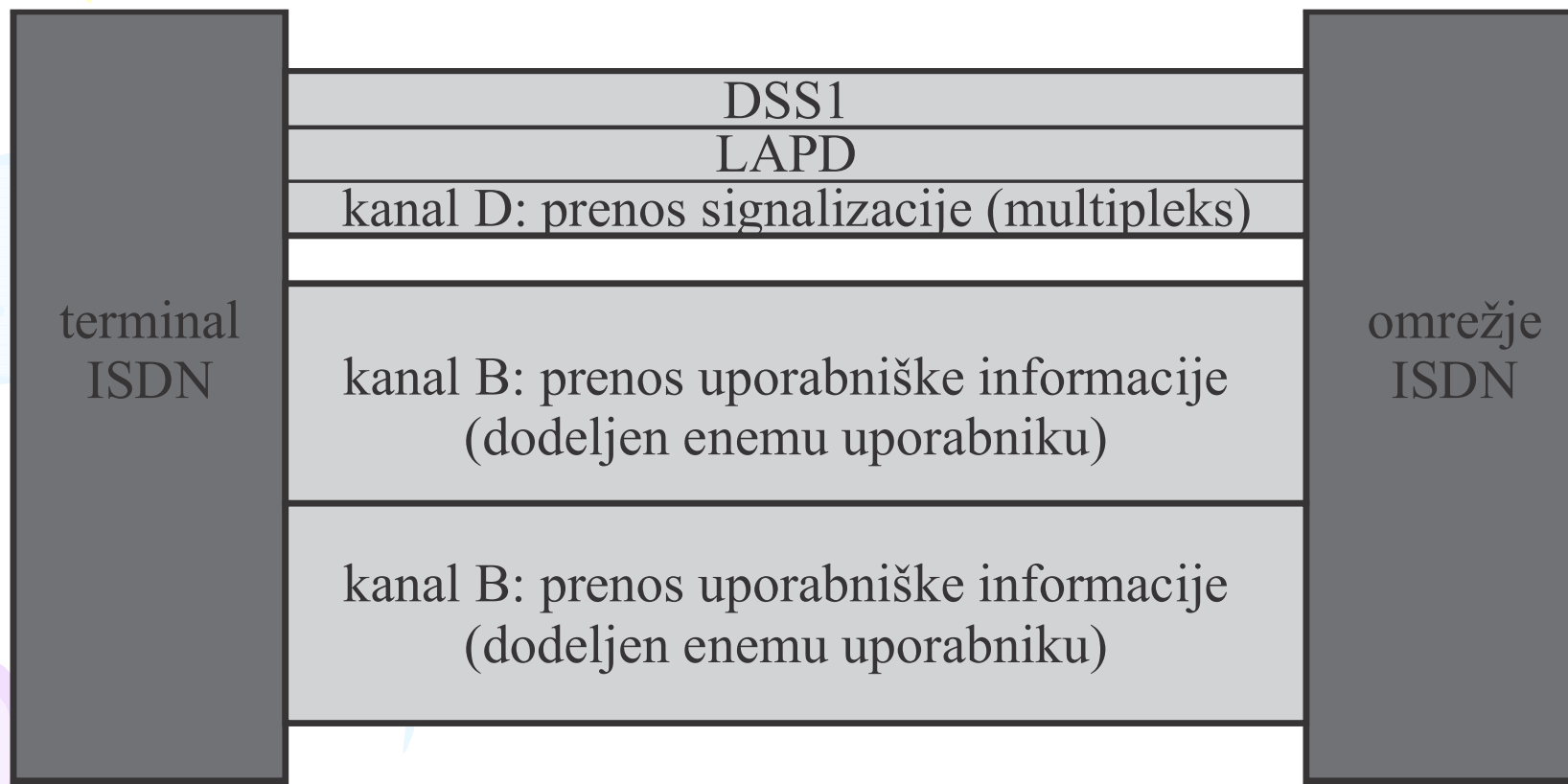
# Protokol LAPD

- Link Access Procedure on D Channel
- ITU-T
- povezavni sloj
- prenos signalizacije med ISDN terminalom in omrežjem preko kanala D
- uravnotežen način
- povezavno orientiran prenos
- protokol s ponavljanjem zaporedja

# Uporabniški priključek ISDN



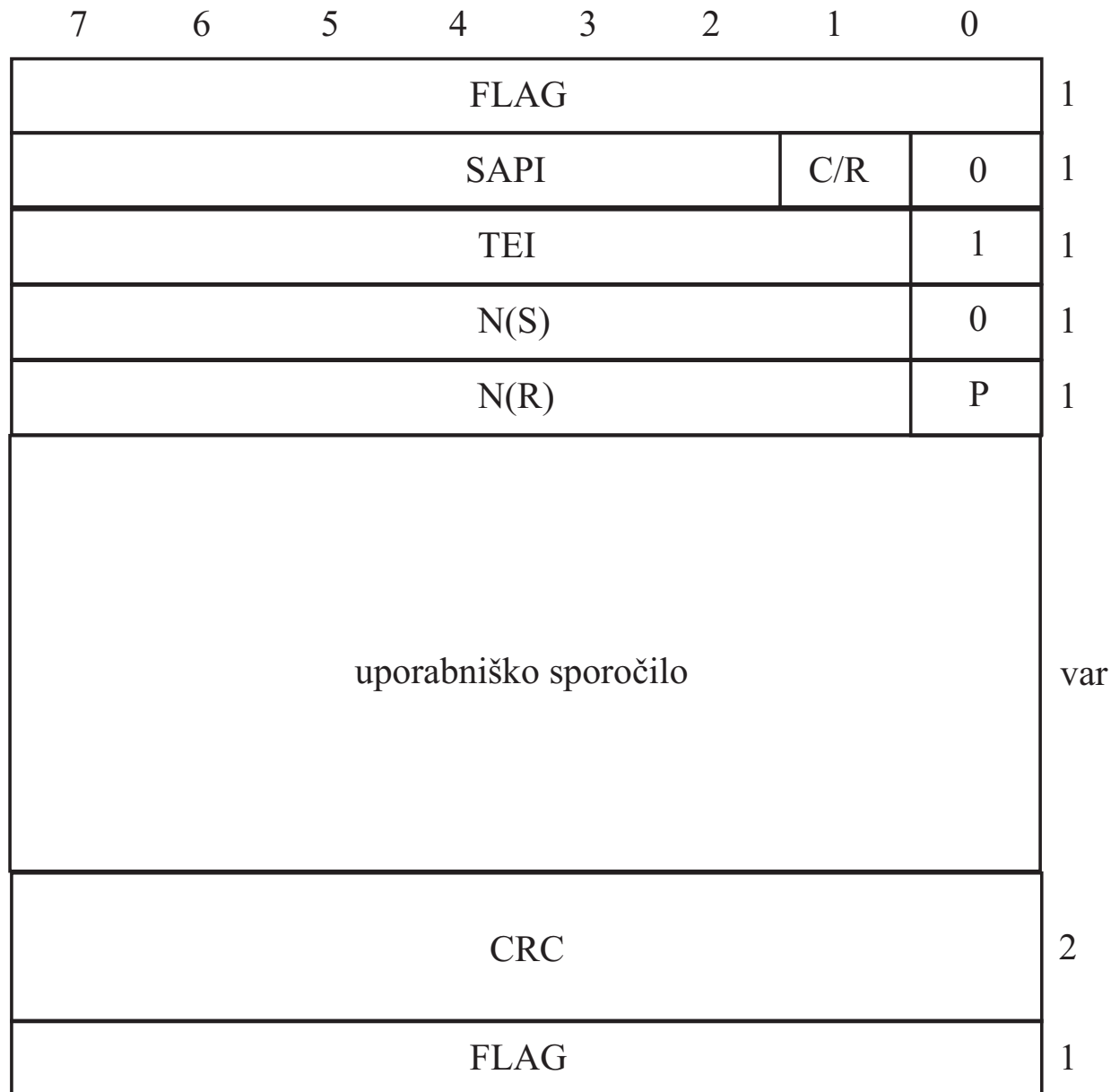
# Dostop terminala ISDN do omrežja ISDN



# Bistvene razlike med LAPB in LAPD

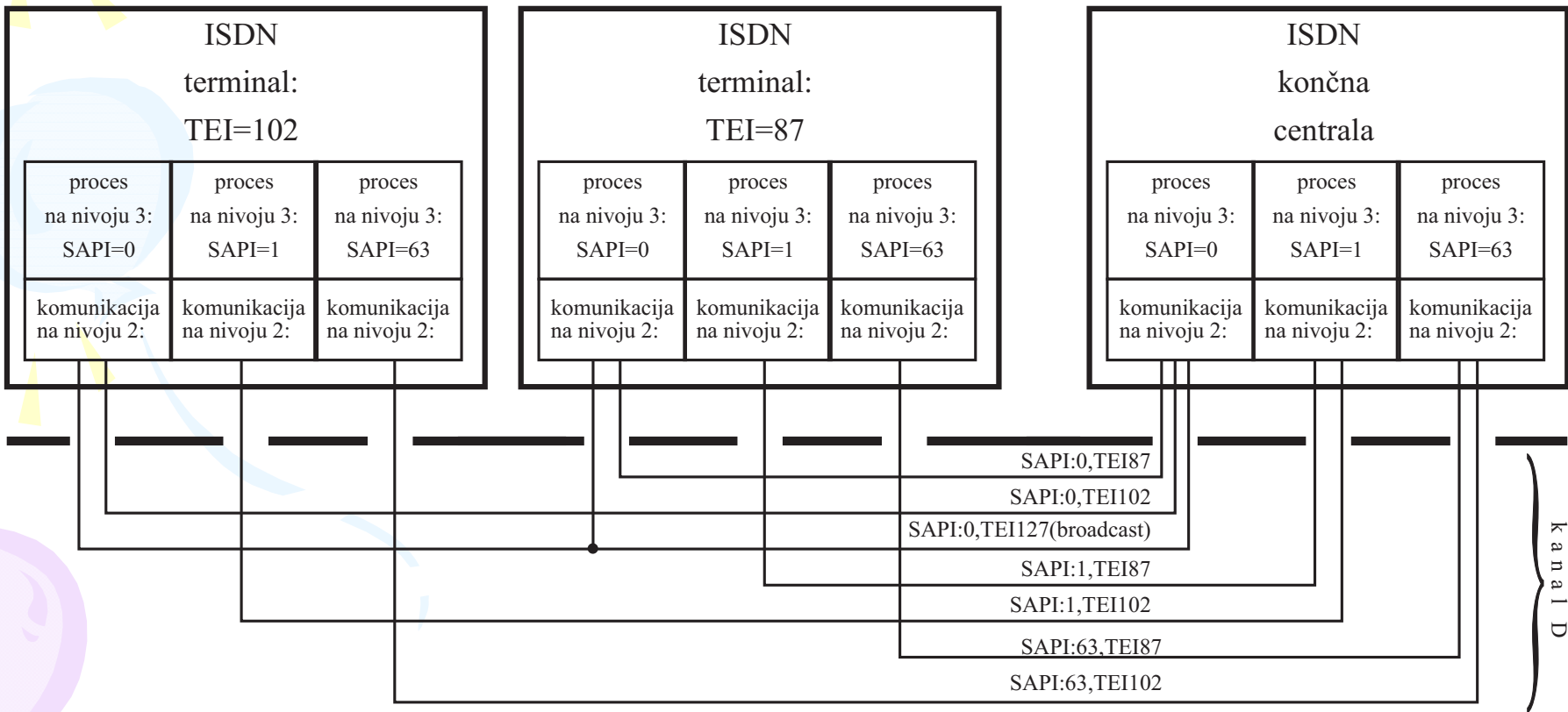
protokol	LAPB	LAPD
naslavljanje	ni pomembno	TEI (terminal) SAPI (proces)
modul štetja	8	128

# Okvir LAPD





# LAPD - postopek multipleksiranja



# Protokol LAPDm

- ETSI, 3GPP
- povezavni sloj, dostop GSM terminala do omrežja (signalizacija, podatki)
- uravnotežen način (?)
- povezavno orientiran prenos, lahko tudi nepovezavno
- protokol s ponavljanjem zaporedja

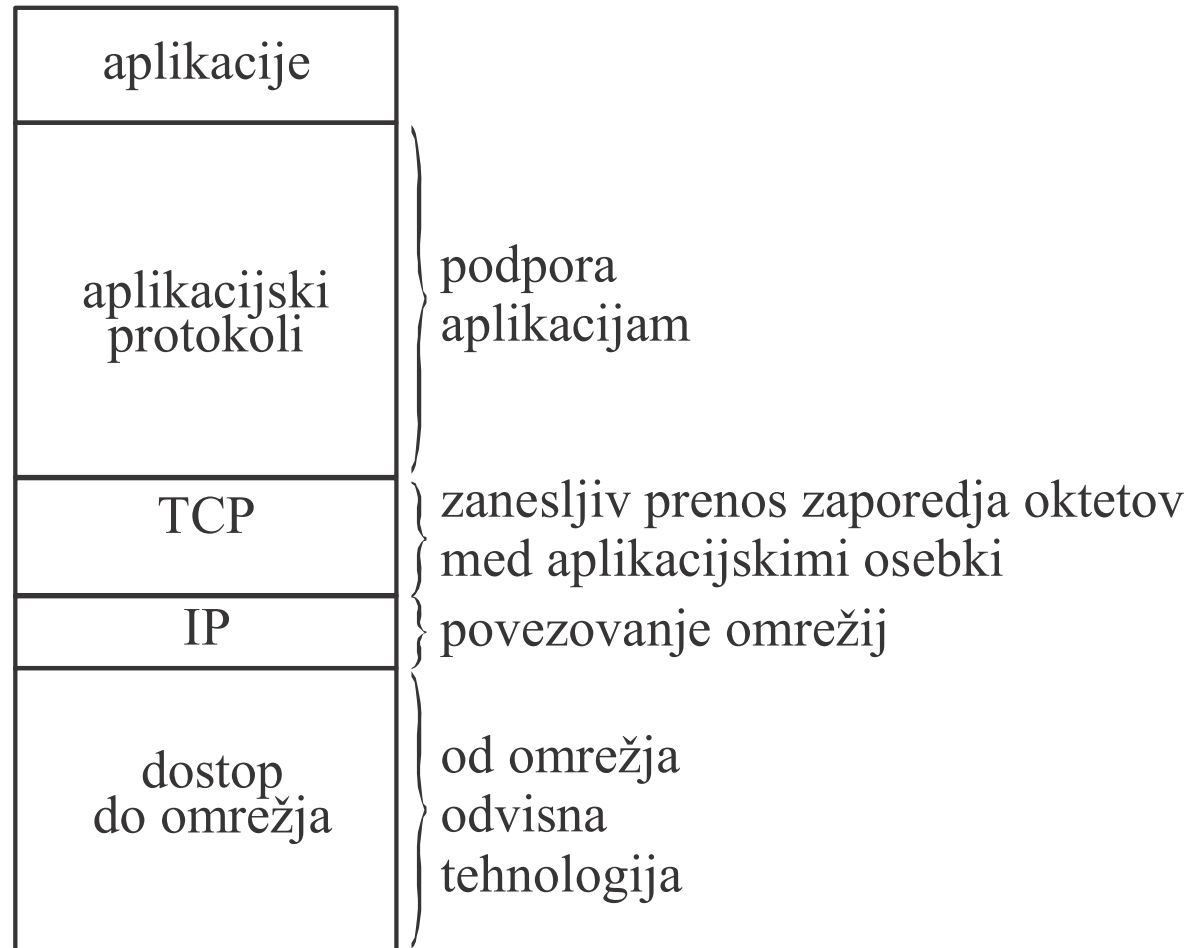
# Bistvene razlike med LAPD in LAPDm

protokol	LAPD	LAPDm
dolžina sporočila	spremenljiva	stalna (segmentiranje)
sinhronizacija	zastavica	v fizičnem sloju
detekcija napak	CRC	v fizičnem sloju
naslavljanje	TEI+SAPI	SAPI
modul štetja	128	8
okvirja SABM, UA	ne nosita uporabniške informacije	lahko nosita uporabniško informacijo

# Protokol TCP

- Transmission Control Protocol
- protokol transportnega sloja protokolnega sklada TCP/IP
- komunikacija med terminali omrežja
- povezavno orientiran
- običajno začne zvezo vzpostavljati odjemalec
- zagotavlja zanesljiv prenos zaporedja oktetov
- TCP uporablja nezanesljiv kanal (omrežje IP)
- protokol s selektivnim ponavljanjem
- krmiljenje pretoka
  - z eksplicitno povratno informacijo
- krmiljenje zamašitev
  - z implicitno povratno informacijo

# Umestitev protokola TCP v protokolni sklad TCP/IP



# Posebnosti protokola TCP

- kanal je nepovezavno omrežje
- štetje oktetov
- štetje po modulu in mrtve duše
- čas izteka časovnika
- vzpostavljanje zveze in mrtve duše
- segmentiranje v internetnem sloju in maksimalna dolžina segmenta
- krmiljenje pretoka
- krmiljenje zamašitev

# Naslavljanje v omrežjih IP

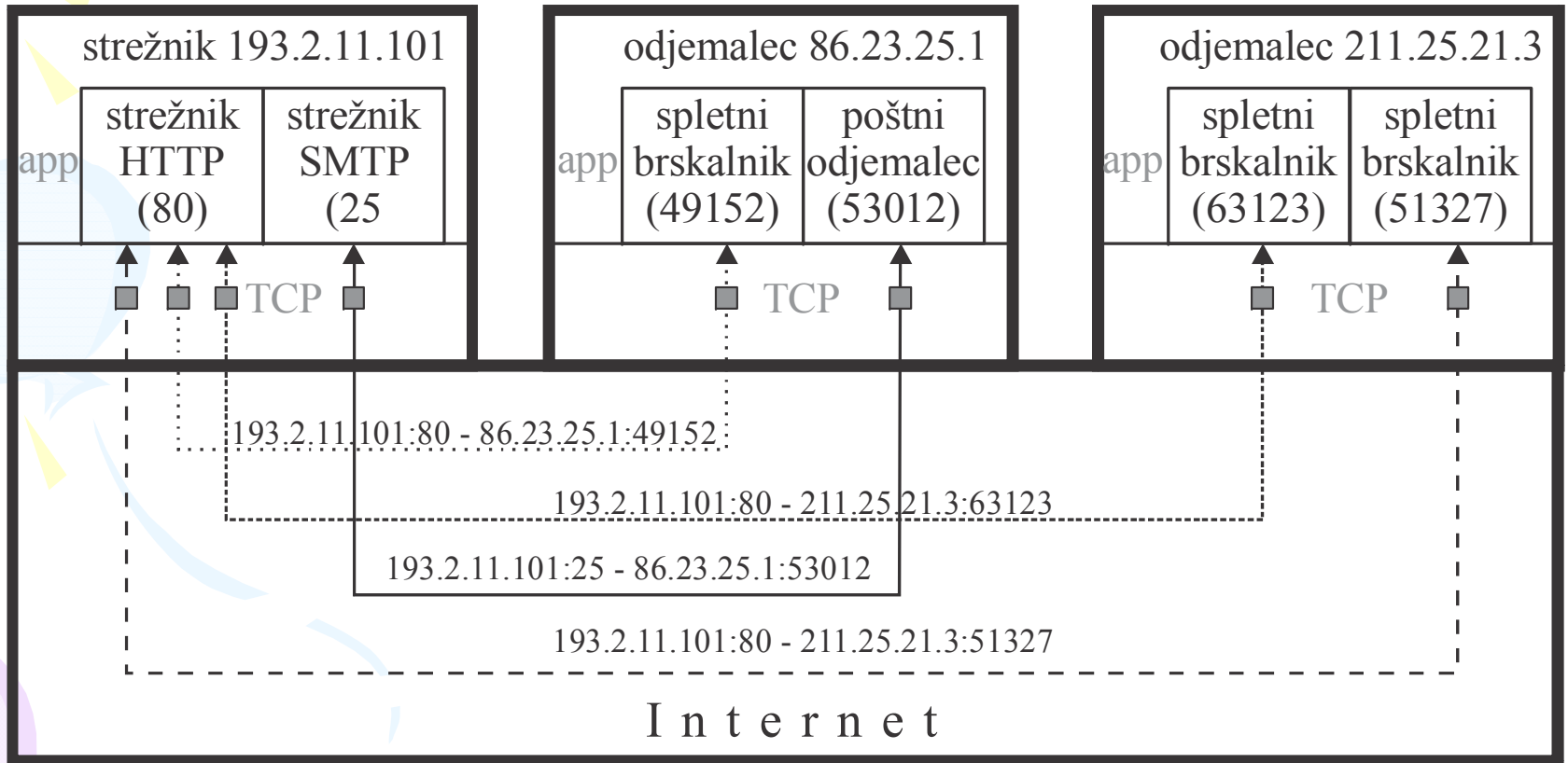
- vtičnica
  - naslov IP
  - številka vrat
    - 16 b
    - standardne številke vrat (0÷1023)  
(well-known port numbers)
    - začasne številke vrat (49152÷65535)  
(ephemeral port numbers)
- določljivost TCP zveze v Internetu


# Nekatere standardne številke vrat

številka vrat	aplikacijski protokol	transportni protokol
20	FTP (podatki)	TCP
21	FTP (seja)	TCP
22	SSH	TCP
23	TELNET	TCP
25	SMTP	TCP
69	TFTP	UDP
80	HTTP	TCP
110	POP3	TCP



# Multipleksiranje TCP zvez v Internetu





# Format segmenta TCP

(en sam tip  
segmenta)

# Maksimalna dolžina uporabniškega sporočila

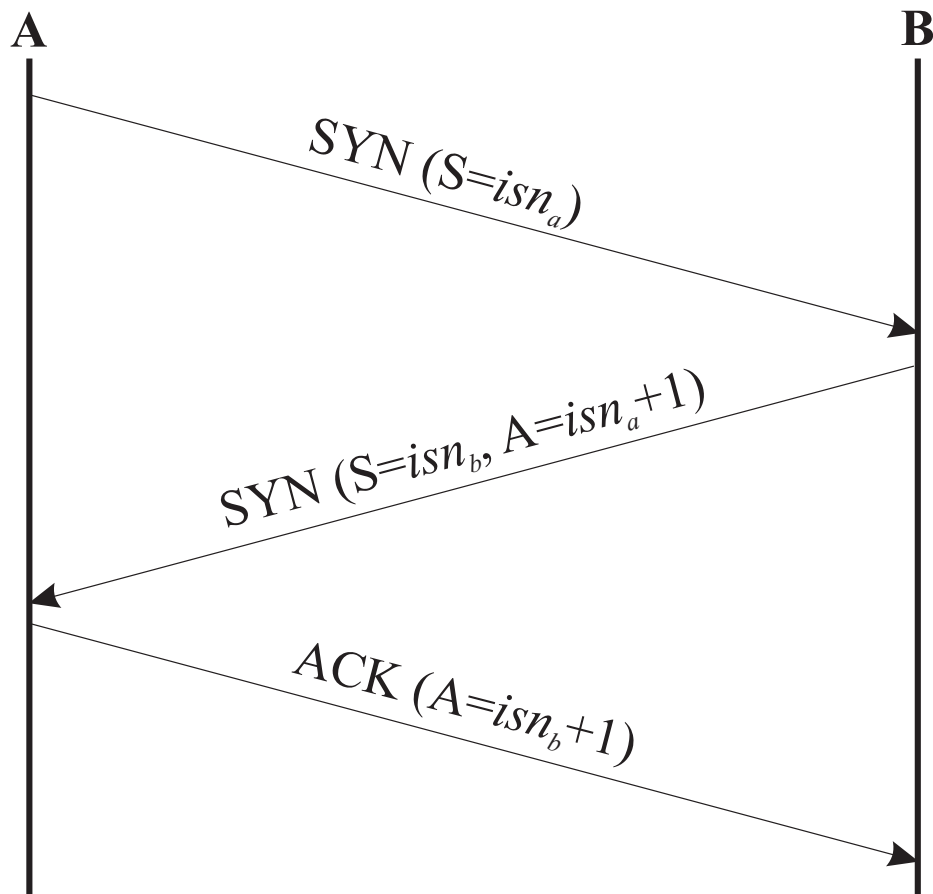
- možnost drobljenja datagramov IP
- MSS - Maximum Segment Size
- maksimalna dolžina podatkov v oktetih
- maksimalna dolžina segmenta =  $MSS+20$
- maksimalna dolžina IP datagrama =  $MSS+20+20$
- privzeta vrednost za MSS = 536

# Osnovni mehanizem prenosa

- protokol s selektivnim ponavljanjem
- štetje uporabniških oktetov
- sekvenčna številka  $s$  v oktetih
- potrditev  $a$  v oktetih
- širina okna  $W$  v oktetih

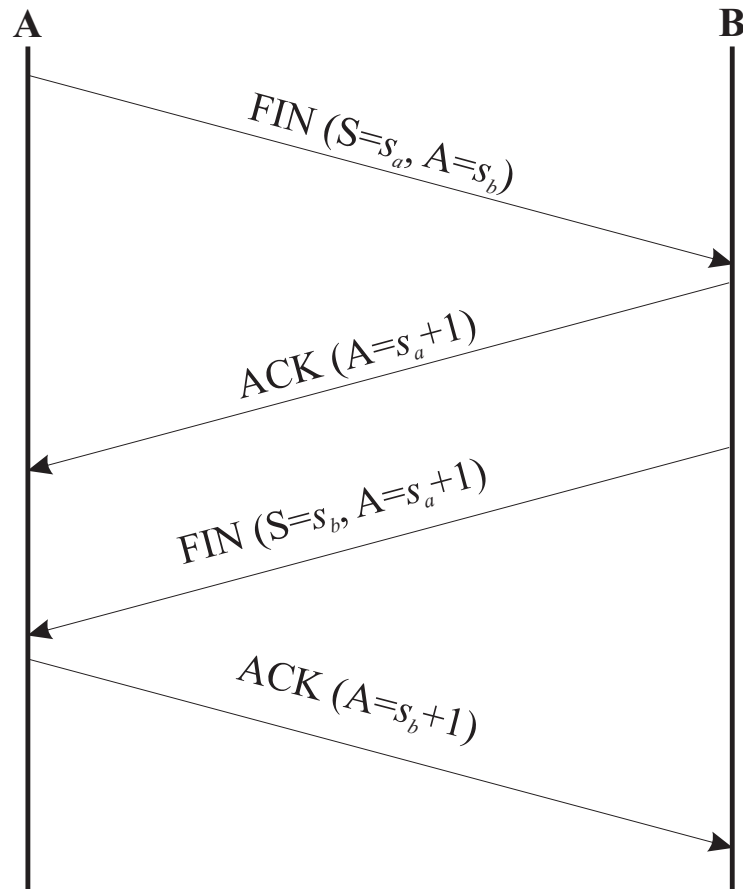
# Vzpostavitev TCP zveze: trojni dogovor

- zakaj trojni dogovor
- inicializacija sekvenčnih števil
- preprečevanje mrtvih duš
- inicializacija širine okna
- dogovor o MSS
- dogovor o drugih parametrih zveze

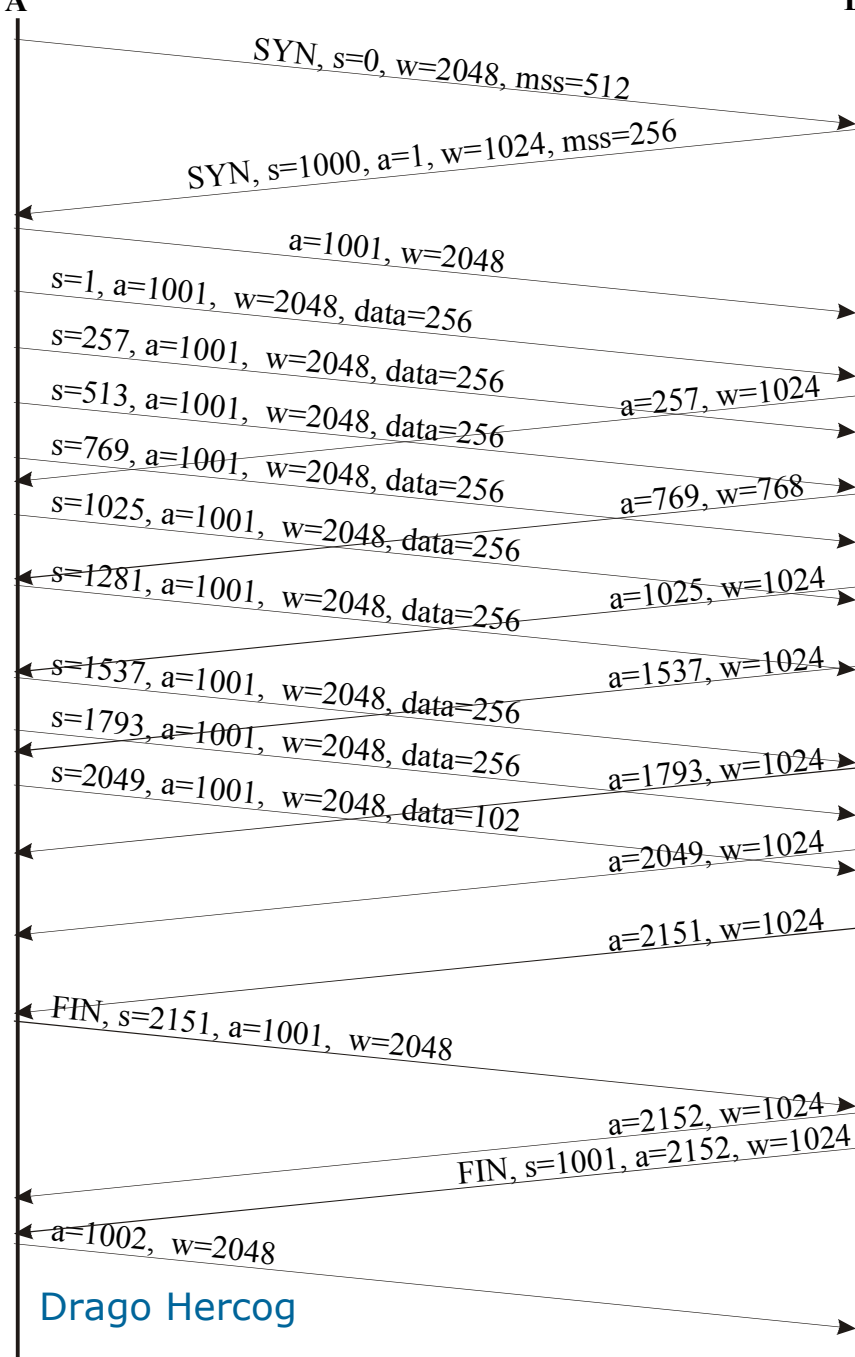


# Sprostitev TCP zveze: dvakratni dvojni dogovor

- sprostitve zveze v vsako smer posebej
- preprečevanje izgub uporabniških podatkov ob sproščanju zveze



# Primer preprostega prenosa podatkov brez izgub (brez počasnega zagona)



# Prenos kratkih sporočil (telnet, rlogin)

- Nagel-ov algoritem:
  - en sam kratek segment v vmesniku



# Merjenje časa do potrditve in čas izteka časovnika

- $M_i$  : meritev  $Trt$

- $A_i$  : ocena povprečne vrednosti  $A_i = A_{i-1} \cdot (1 - g) + M_i \cdot g$

- $D_i$  : ocena standardnega odklona  $D_i = D_{i-1} \cdot (1 - h) + |M_i - A_{i-1}| \cdot h$

- $Tto$  : čas izteka časovnika  $Tto_i = A_i + 4 \cdot D_i$

- Začetni vrednosti:

---

$$A_0 = M_0 + 0,5$$

$$D_0 = \frac{A_0}{2}$$

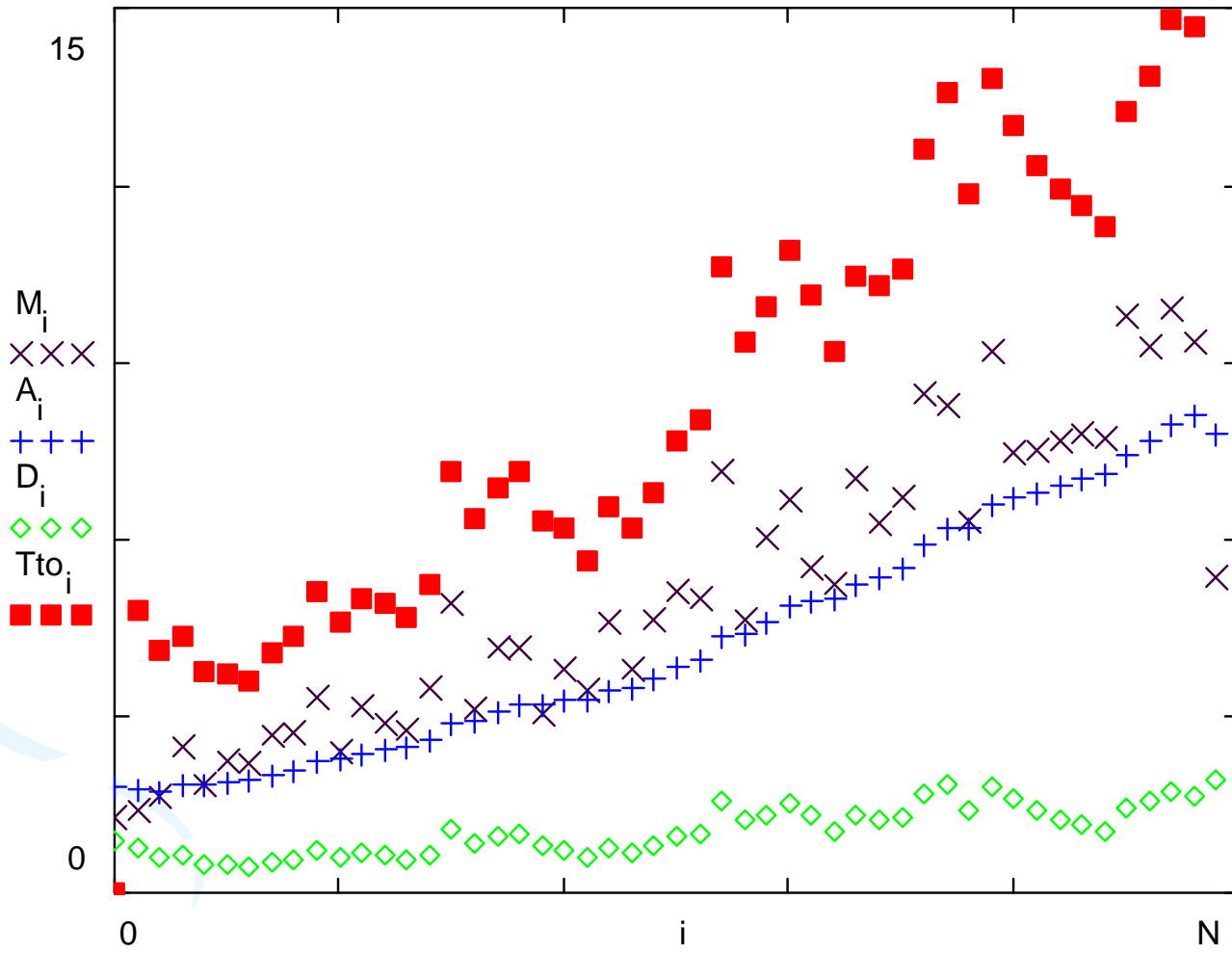
- Konstanti  $g$  in  $h$ :

---

$$g = 0,125 = \frac{1}{8}$$

$$h = 0,25 = \frac{1}{4}$$

# Čas izteka časovnika: primer

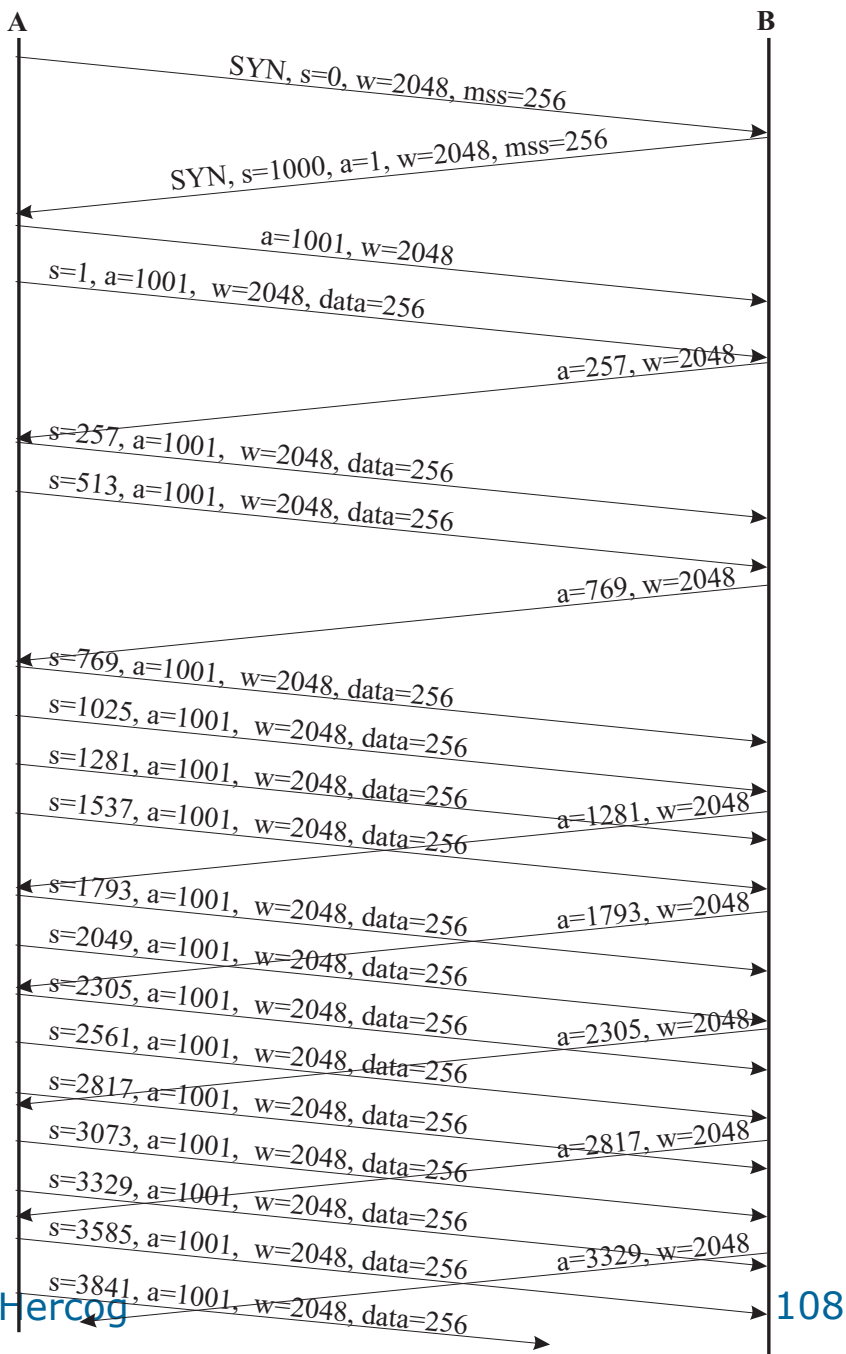


# EkspONENTNO povečevanje časa izteka časovnika

- predpostavka: časovnik se izteče zaradi (grozeče) zamašitve
- pri vsaki ponovitvi pošiljanja segmenta je čas izteka časovnika večji
  - $Tto_{n+1} = q \cdot Tto_n$
  - npr:  $q=2$

# Vzpostavitev TCP zveze in počasni zagon

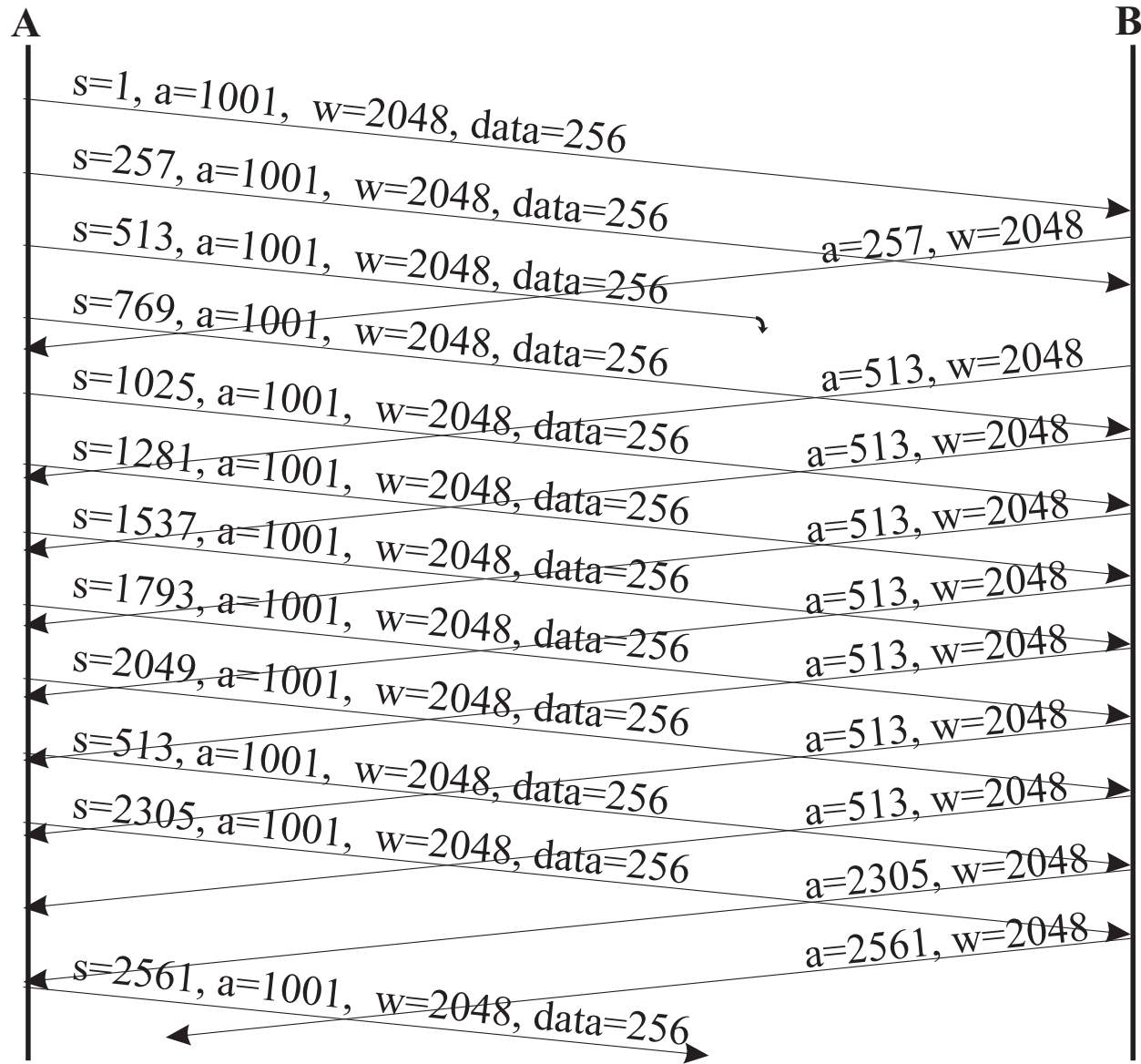
- v Internetu ni krmiljenja dostopa
- če oddajnik začne s polno hitrostjo oddajati v (skoraj) zamašeno omrežje, bo še poslabšal stanje
- počasni zagon (slow start)
- začetna širina oddajnega okna: en segment
- po prejemu vsake potrditve se širina oddajnega okna poveča za en segment
- eksponentno povečevanje širine oddajnega okna
- "otipavanje" razmer v omrežju



# Prenos podatkov v prisotnosti majhnih izgub

- če sprejemnik sprejme nepričakovan segment, ponovi potrditev
- tri duplikatne potrditve (štiri enake potrditve) naznanjajo lahke izgube
- algoritem hitre ponovitve
  - po prejemu treh duplikatnih potrditev oddajnik ponovi oddajo domnevno izgubljenega segmenta
- algoritem hitrega zdravljenja
  - po prejemu treh duplikatnih potrditev oddajnik zmanjša širino oddajnega okna na polovico in ga nato linearno povečuje (za en segment v  $T_{rt}$ )

# Primer prenosa podatkov pri majhnih izgubah



# Krmiljenje zamašitev v primeru izteka časovnika

- iztek časovnika pomeni velike izgube (zamašitev)
- oddajnik ponovno pošlje domnevno izgubljeni segment
- oddajnik izvaja algoritem počasnega zagona, dokler ne doseže polovične širine okna pred zamašitvijo
- nato linearno povečuje širino oddajnega okna

# Širina oddajnega okna

- oglaševano okno (advertised window):  
širina oddajnega okna, ki jo dovoljuje sprejemnik
- zamašitveno okno (congestion window):  
širina oddajnega okna, ki jo dovoljuje krmiljenje pretoka in zamašitev
- zamašitveno okno  $\leq$  oglaševano okno



# Krmiljenje pretoka med terminaloma

- metoda krmiljenja s spremenljivo širino okna (krediti)

# Krmiljenje zamašitev skozi omrežje

- metode preprečevanja zamašitve
  - Nagelov algoritem
  - počasni zagon po vzpostavitvi zveze
  - pri duplikatnih potrditvah oddajnik zoža širino okna na polovico in linearno povečuje širino okna (1 segment v  $T_{rt}$ )
- metoda zdravljenja zamašitve
  - eksponentno povečevanje časa izteka časovnika
  - po izteku časovnika zoža širino okna na 1 segment in eksponentno povečuje širino okna (za 1 segment ob vsaki potrditvi)

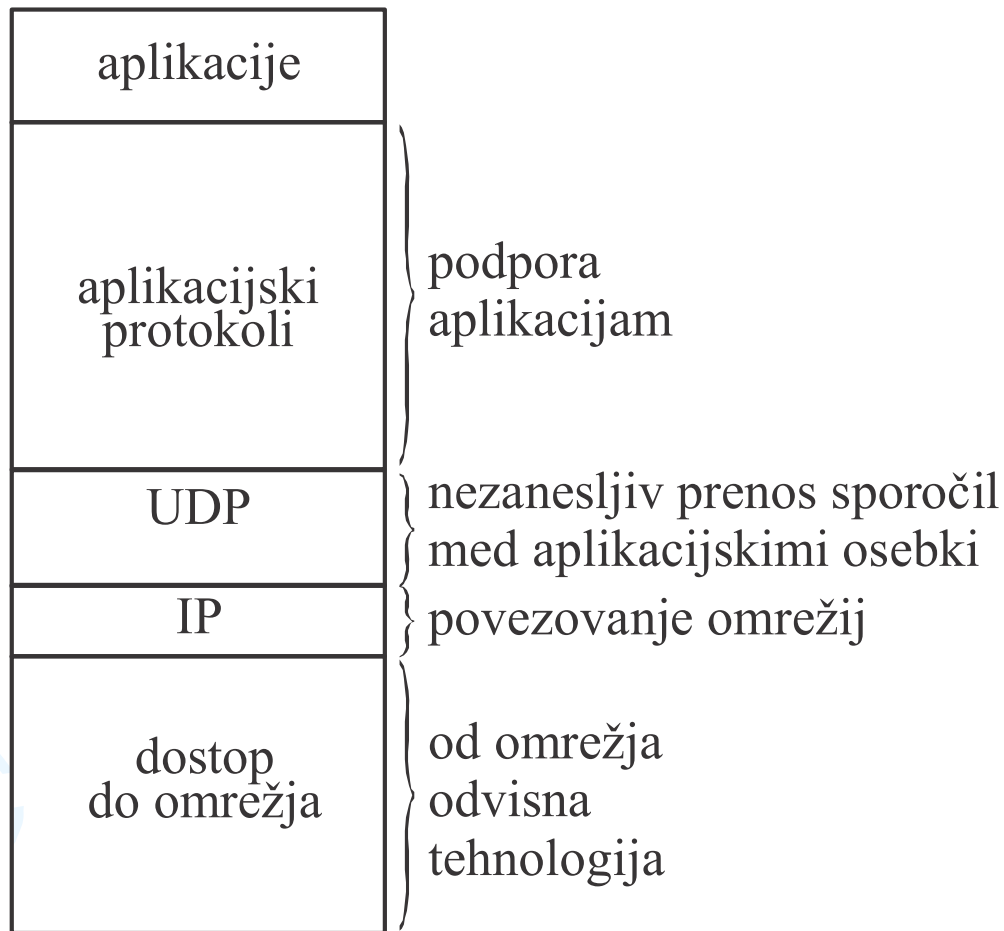
# Protokol TCP

- kompleksen protokol
- povezanost protokola TCP s protokolom IP
- vitalnost protokola TCP
- aktualnost protokola TCP
- problemi v radijskih omrežjih

# Protokol UDP

- User Datagram Protocol
- protokol transportnega sloja protokolnega sklada TCP/IP
- komunikacija med terminali omrežja
- nepovezavno orientiran
- ne zagotavlja zanesljivega prenosa
- UDP uporablja nezanesljiv kanal (omrežje IP)
- Aplikacijski protokol nad UDP mora sam poskrbeti za zanesljiv prenos, če ga potrebuje (npr. TFTP)
- ne izvaja krmiljenja pretoka in zamašitev
- skrbi za naslavljanje aplikacijskih procesov (s številkami vrat)

# Umestitev protokola UDP v protokolni sklad TCP/IP



# Prednosti protokola UDP pred TCP

- je preprostejši
- zahteva preprostejšo implementacijo
- ne uvaja spremenljivih zakasnitev
- uporabljamo ga pri
  - aplikacijah, ki ne zahtevajo zanesljivega prenosa
  - aplikacijah, ki imajo na voljo malo virov
  - prenosu v realnem času (npr. VoIP)

# Format segmenta UDP

št. vrat pošiljatelja (16 bit)	št. vrat naslovnika (16 bit)
dolžina datagrama (16 bit)	zaščitni biti (16 bit)
// podatki //	

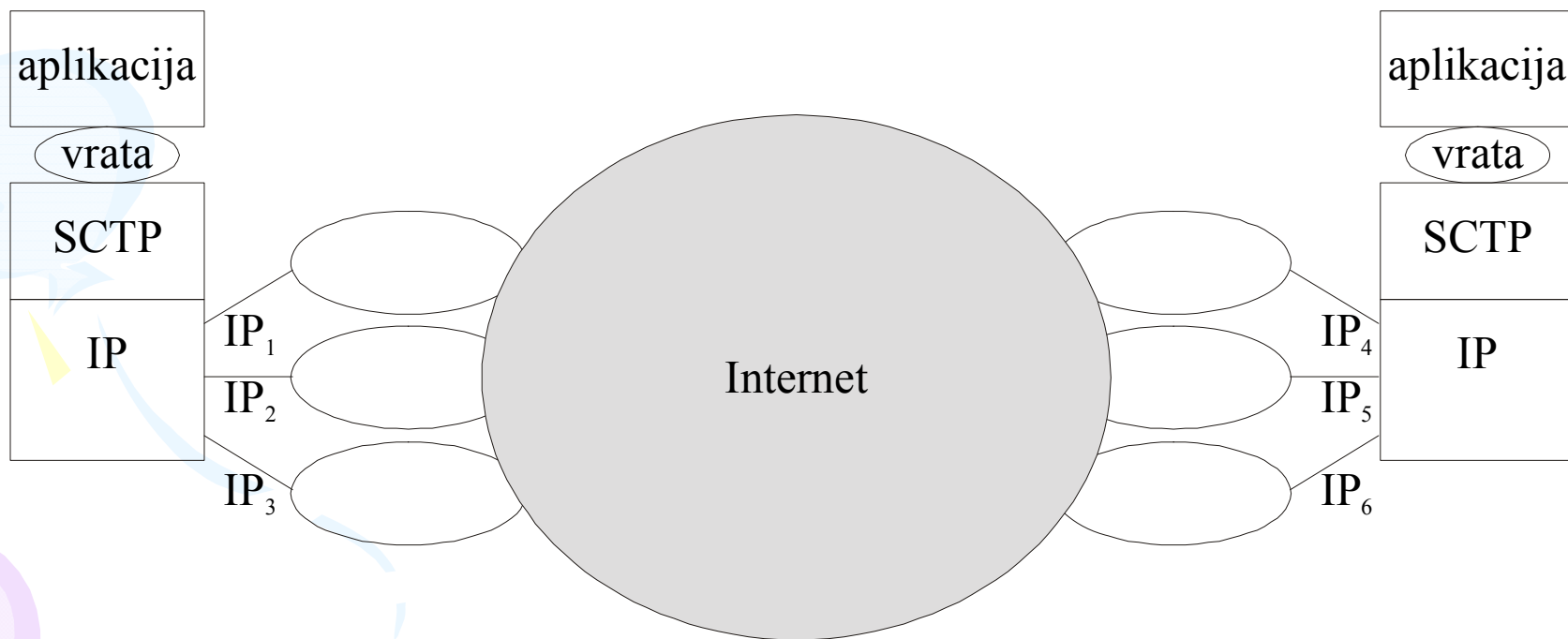
# Protokol SCTP

- Stream Control Transmission Protocol
- prvotno namenjen prenosu signalizacije preko omrežij IP
- protokol transportnega sloja protokolnega sklada TCP/IP
- komunikacija med terminali omrežja
- povezavno orientiran (asociacije)
- podpira večdomnost
- vtičnico definira skupina naslovov IP in ena številka vrat
- podpira prenos več tokov v okviru ene asociacije
- zagotavlja zanesljiv prenos sporočil



# Večdomna asociacija SCTP

- ena vrata, več naslovov IP na vsaki strani



# Struktura sporočila SCTP



- sporočilni kos
  - nadzorni
  - informacijski
- različni sporočilni kosi lahko pripadajo različnim tokom
- vsak kos ima svojo glavo