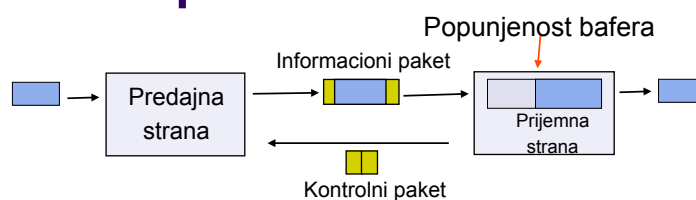


Kontrola protoka

Kontrola zagušenja



Kontrola protoka



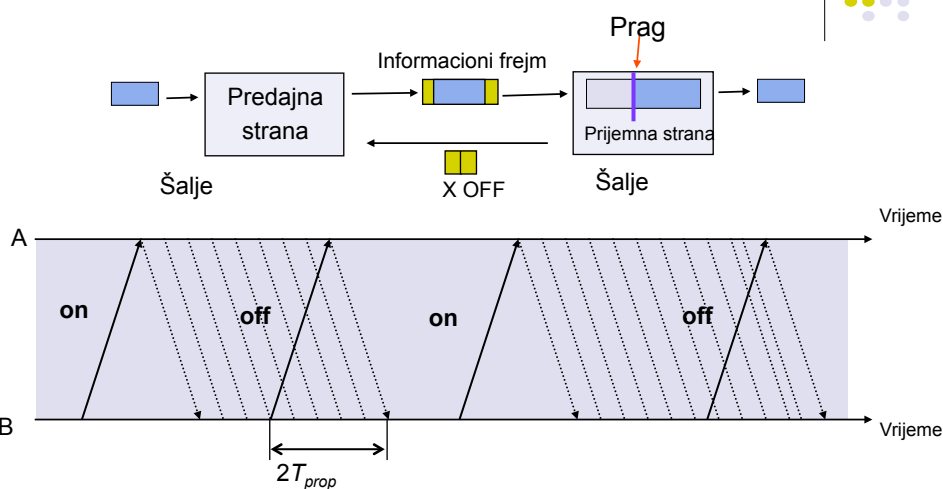
- Prijemna strana ima ograničeni prostor u baferima za prijem paketa
- Nekoliko situacija mogu dovesti do prekoračenja popunjenosti bafera
 - Nepodešenost brzine slanja i brzine obrade podataka
 - Veliki skok dolaznog saobraćaja
- *Kontrola protoka* sprečava prekoračenje popunjenosti bafera regulisanjem brzine kojom izvor podataka generiše informacije koje se prenose

Kontrola protoka

- Predajna kontrola protoka
 - Može biti realizovana između dva krajnja sistema ili između jednog krajnjeg sistema i mreže
 - Stop&wait ili pomjerački prozor
 - Software ili hardware realizacije
- OPEN LOOP kontrola protoka
 - Nema povratne informacije od prijemne strane
 - Resursi se rezervišu unaprijed prilikom uspostavljanja toka (ATM – Call Admission Control)
 - Koristi se u kontroli zagušenja
- CLOSED LOOP kontrola protoka
 - Mreža obavještava predajnu stranu o zagušenju
 - Koristi se u kontroli zagušenja



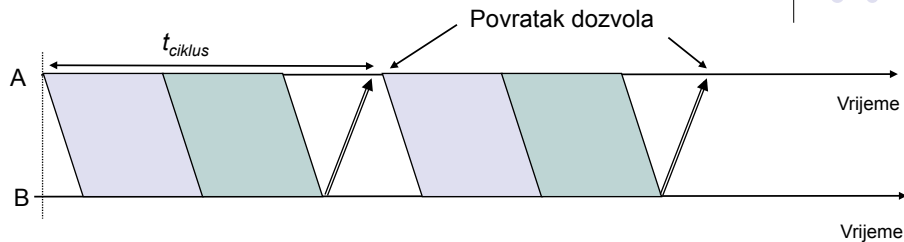
X ON / X OFF



Prilikom postavljanja praga mora se voditi računa da još $2T_{prop}R$ bita mora biti slobodno u baferu



Kontrola protoka metodom prozora



- ARQ metod baziran na pomjeračkim prozorima sa W_s
 - Predajna strana nikada ne može poslati više od W_s frejmova
- Potvrde ACK koje pomjeraju prozor unaprijed mogu biti tretirane kao dozvole za slanje
- Moguće je korišćenje potvrda i na krajevima ciklusa čime bi se regulisala brzina slanja
- Problemi sa korišćenjem pomjeračkih prozora za kontrolu protoka i greške
 - Izbor veličine prozora
 - Uzajamni uticaj brzine slanja i retransmisija
 - TCP ima odvojenu kontrolu greške i kontrolu protoka

TCP kontrola protoka



- TCP prijemna strana kontrolira brzinu kojom predajna strana šalje podatke kako bi spriječio prekoračenje bafera
- TCP prijemna strana obavještava predajnu stranu koliko bajta može da primi

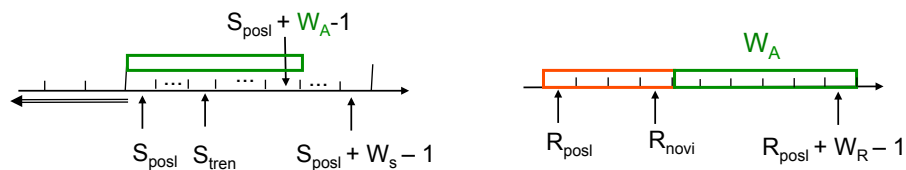
$$W_A = W_R - (R_{novi} - R_{posl})$$

- TCP predajna strana je zadužena da nivo poslatih podataka drži ispod W_A

$$(S_{tren} - S_{posl}) \leq W_A$$

Prozor slanja

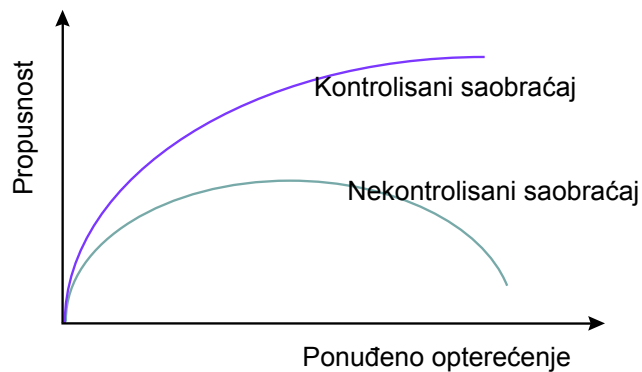
Prozor prijema





Idealni uticaj kontrole zagušenja:

Resursi se efikasno koriste dok je nivo saobraćaja niži od dostupnog kapaciteta



Open-Loop kontrola

- Mrežne performanse se garantuju svim saobraćajnim tokovima koji pristupaju mreži
- Razvijen je za konektivne mreže
- Ključni mehanizmi
 - Kontrola prijema (Admission Control)
 - Administrativno podešavanje (Policing)
 - Oblikovanje saobraćaja (Traffic Shaping)
 - Raspoređivanje saobraćaja (Traffic Scheduling)

Kontrola prijema



Tipična brzina prenosa koju zahtijeva izvor informacija je varijabilna veličina

- Prilikom uspostavljanja toka pregovaraju se njegovi parametri sa mrežom
- Specificiraju se zahtjevi:
 - Maksimalna, Srednja i Minimalna brzina prenosa
 - Maksimalna veličina grupe (burst)
 - Kašnjenje, nivo gubitaka
- Mreža utvrđuje koji su resursi potrebni
 - “Efektivni” kapacitet
- Ako je tok prihvaćen, mreža alokira resurse tako da obezbijedi kvalitet servisa (QoS) dok god izvor saobraćaja poštuje dogovor

Administrativno podešavanje



- Mreža neprekidno nadzire saobraćajne tokove da bi obezbijedila poštovanje dogovora
- Kada neki paket nije poslat u skladu sa dogovorom, mreža ga može odbaciti ili mu dodijeliti niži prioritet što može dovesti do naknadnog odbacivanja ako se za to javi potreba
- Ako se zagušenje pojavi, paketi označeni sa nižim prioritetom se prvi odbacuju
- *Leaky Bucket Algoritam (Algoritam probušenog bureta)* je najčešće korišćeni mehanizam ove namjene
 - Bucket (Bure) ima brzinu oticanja koja odgovara srednjoj ugovorenoj brzini
 - Bucket ima određenu dubinu tako da može podržati varijacije dolazne brzine
 - Dolazni paketi su prihvatljivi ako ne izazivaju prekoračenje

Leaky Bucket algoritam se koristi za administrativni nadzor dolazne brzine paketa



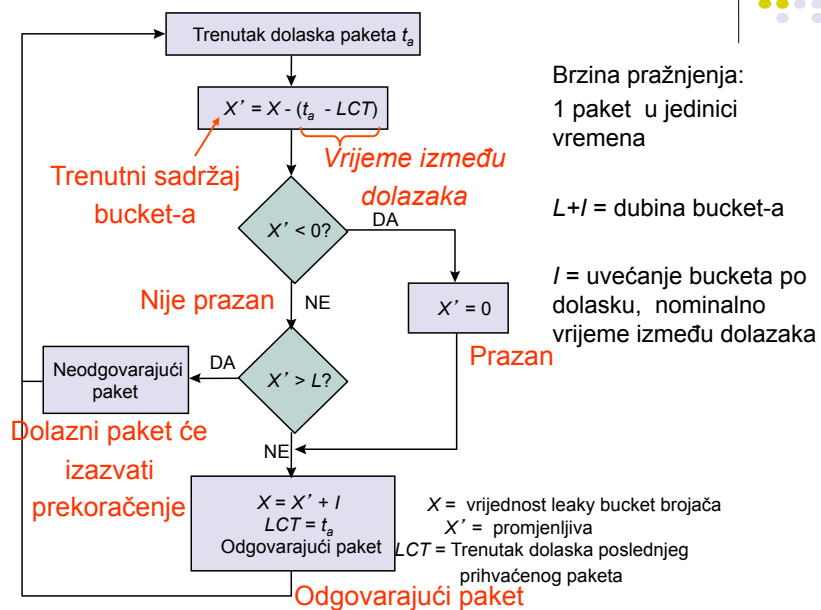
Brzina oticanja odgovara nekoj brzini u ravnotežnom stanju

Dubina Bucket-a odgovara maksimalnom dozvoljenom dolaznom nivou bursta

1 paket u jedinici vremena
Neka su paketi konstantne veličine kao kod ATM

Neka je X = sadržaj bucket-a u momentu kada je stigao poslednji prihvatljivi paket
Neka je t_a – trenutak dolaska paketa

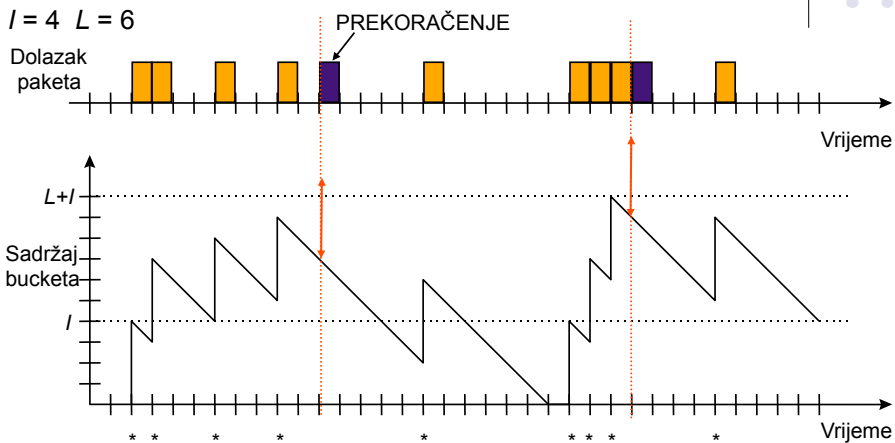
Leaky Bucket Algoritm



Leaky Bucket primjer

$I = 4$ $L = 6$

Dolazak paketa



Paketi koji izazovu prekoračenje ne ulaze u bucket jer nijesu uključeni u inicijalne kalkulacije

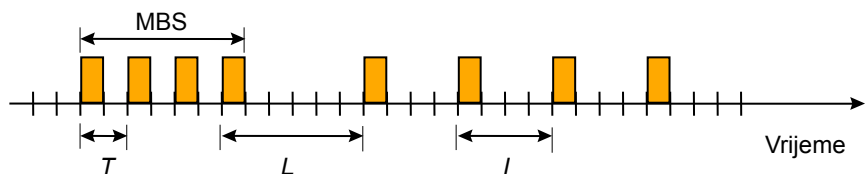
Parametri algoritma

$T = 1 /$ maksimalna brzina

MBS = maksimalna veličina burst-a

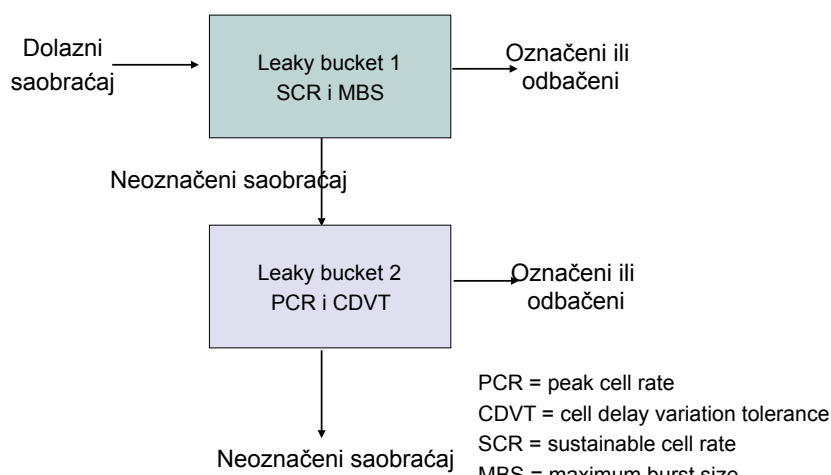
$I =$ nominalno vrijeme između odlazaka = $1 /$ prihvatljiva brzina

$$MBS = 1 + \left[\frac{L}{I - T} \right]$$

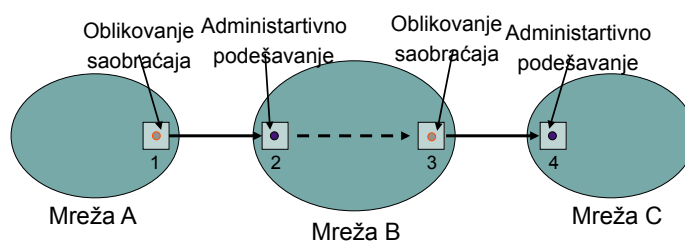


Dvojni Leaky Bucket

Dvojni leaky bucket za podešavanje PCR, SCR i MBS:

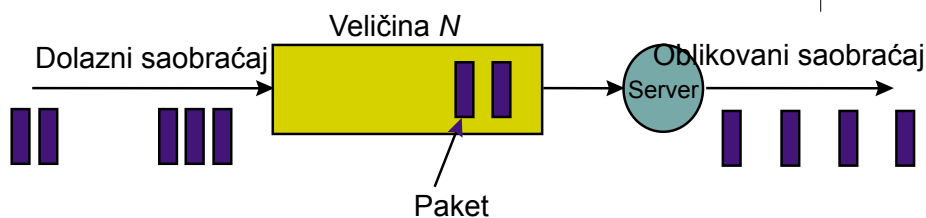


Oblikovanje saobraćaja



- Mreže administrativno podešavaju dolazne saobraćajne tokove
- *Oblikovanje saobraćaja* se koristi kada se želi postići da određeni saobraćajni tok odgovara određenim parametrima
- Mreže mogu oblikovati svoj saobraćaj prije predaje narednoj mreži

Leaky Bucket oblikovanje saobraćaja

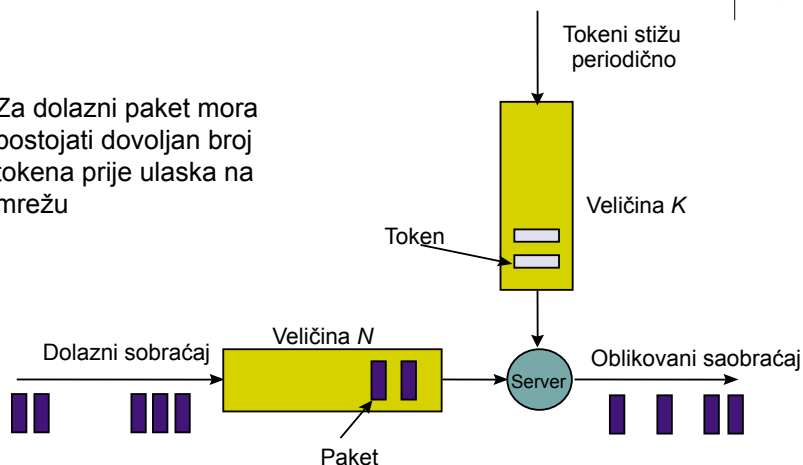


- Baferovanje dolaznih paketa
- Periodično se poslužuju saglasno odgovarajućim parametrima
- Skokovi saobraćaja se baferuju i poravnavaju
- Moguć je gubitak paketa zbog prekoračenja bafera
- Previše restriktivno podešavanja, pošto željeni saobraćaj ne treba da bude u potpunosti ravnomyeran

Token Bucket oblikovanje saobraćaja



Za dolazni paket mora postojati dovoljan broj tokena prije ulaska na mrežu

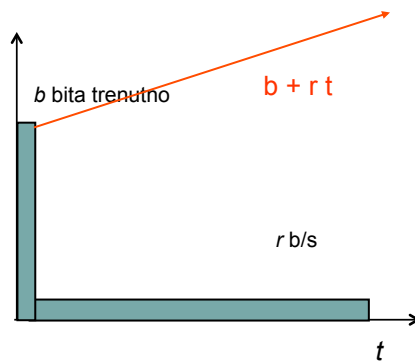


- Brzina tokena reguliše prenos paketa
- Ako je dovoljan broj tokena na raspolaganju, paketi ulaze na mrežu bez kašnjenja
- K određuje kolika je grupisanost saobraćaja dozvoljena na mreži

Token Bucket efekat oblikovanja



Token bucket ograničava saobraćaj koji dolazi od izvora tako da je manji od $b + r t$ bita u intervalu dužine t



Raspoređivanje za garantovane servise



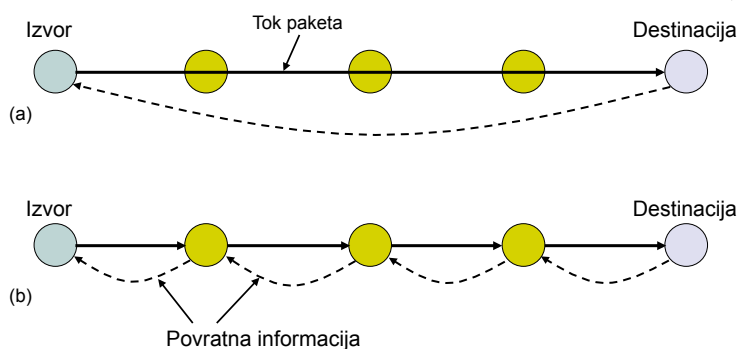
- Pretpostavimo da su garantovane granice kašnjenja od kraja do kraja
- CAC procedura je potrebna za alokaciju resursa i postavljanje schedulera
- Saobraćajni tokovi moraju biti oblikovani/ regulisani tako ne prelaze dodijeljene resurse
- Striktna ograničenja kašnjenja moraju biti postignuta

Closed-Loop kontrola zagušenja



- Kontrola
 - Za regulisnje brzine kojom izvor šalje podatke na mrežu koristi se povratna informacija
 - Bazirana je na sadržaju bafera, iskorišćenju linkova,....
 - Primjeri: TCP; ATM
- End-to-end ili Hop-by-hop?
- Implicitna ili Eksplicitna povratna informacija?
 - Izvorišta na bazi kašnjenja i gubitaka u prenosu zaključuju u kakvom je stanju mreža.
 - Mrežna čvorišta generišu poruke koje upozoravaju na zagušenje

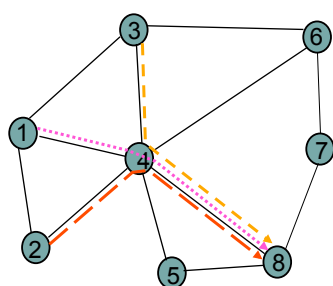
End-to-End ili Hop-by-Hop kontrola zagušenja?



Upravljanje saobraćajem

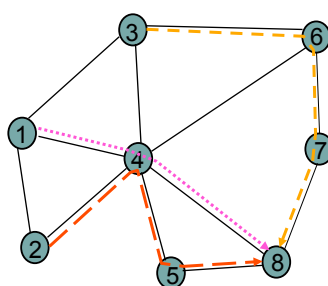


- Upravljanje se obavlja na nivou agregacije tokova
- Tokovi se distribuiraju tako da se postigne efikasno iskorišćenje resursa
- Algoritam najkraćeg puta više nije dovoljan
 - Ne uzima u obzir pojedinačke zahtjeve prema brzini prenosa
 - Ne uzima u obzir međusoban uticaj tokova
- Moraju se posmatrati agregirani zahtjevi tokova



(a)

Putanja sa najkraćim brojem skokova zagušuje link od 4 do 8



(b)

Bolja alokacija tokova distribuirala tokove uniformnije