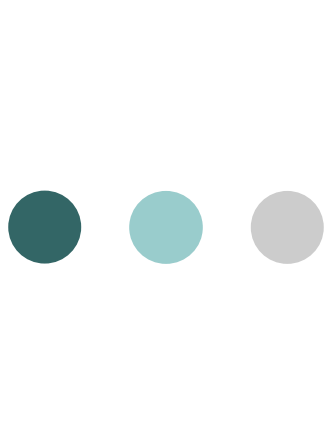


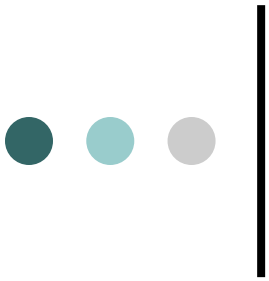
Multimedijalno inženjerstvo – master strukovne studije



Digitalni komunikacioni sistemi: **Lekcija 6: Rutiranje II**

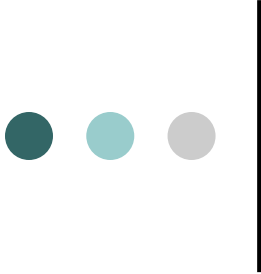
zima 2018/2019

Branimir M. Trenkić



Rutiranje

- rutiranje otporno na otkaze –



Otkazi u mreži

- **Problemi:** Linkovi i komutatori moгу otkazati
 - **Oglašavanje** se može *izgubiti*
 - Petlje prosleđivanja (**routing loop**) se mogu desiti
 - Niz uzastopnih tačaka na putanji prosleđivanja **koji ima u sebi ciklus** (tako da paket nikada neće doći do svog odredišta)
 - Nedostupno odredište (**dead-end**): ruta zapravo ne omogućuje dostupnost odreduštu
 - Petlje i dead-end-ovi dovode **do ne-validnosti ruta** (putanja)

Otkazi u mreži

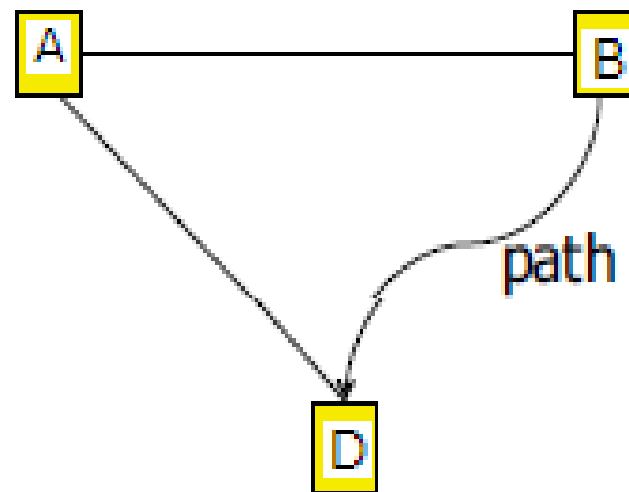
○ **Rešenje**

- **Ograničenje broja skokova** (*hop-limit*)
podržano u transportnom protokolu
- **HELLO protokol** kako bi se detektovala
operativnost suseda
- **Periodična oglašavanja** sa rutera
- **Periodična integracija** na ruterima

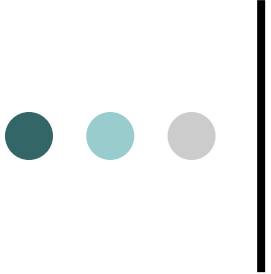
Dovode do **konvergencije u korektno stanje**

Petlja rutiranja u LS protokolu

- Optimalna *putanja od B do D*: *preko A*
- Otkaz linka A-D*
- A-ov *LSA paket* (oglašavanje) ka B – *izgubljen!*
- A* sada *koristi B* da dođe do D
- Ali *B*, nastavlja da *koristi A*
- Petlja rutiranja!*



- Mora se čekati eventualni prijem korektnog LSA kako bi se prekinula petlja

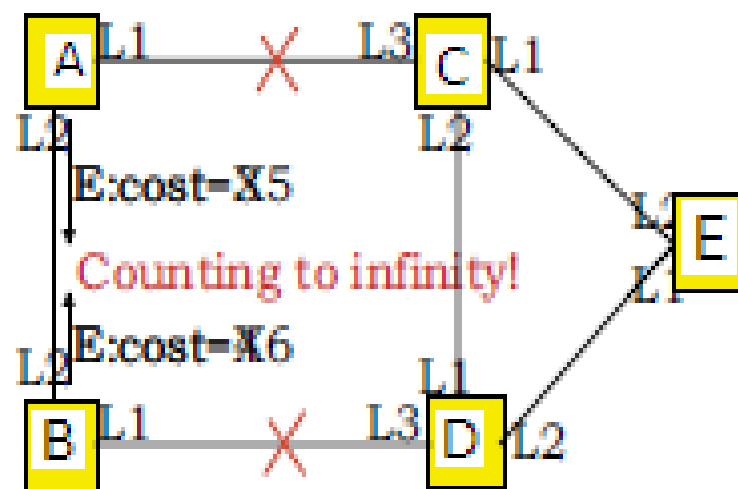


Vektor udaljenosti: za i protiv

- + Jednostavan protokol
- + Radi dobro u malim mrežama
- - Radi samo u malim mrežama

DV: Primer ruting petlje

- Predpostavimo **otkaz A-C linka**
- Kada **A** otkrije to stanje (otkaz),
šalje ka B E: cost = ∞
- Nakon toga, **B oglašava (ka A)**
E: cost = 2
- **A ažurira** svoju **tabelu rutiranja**
E: cost = 3
- Sada, predpostavimo da **dolazi**
do otkaza B-D linka



DV: Primer ruting petlje

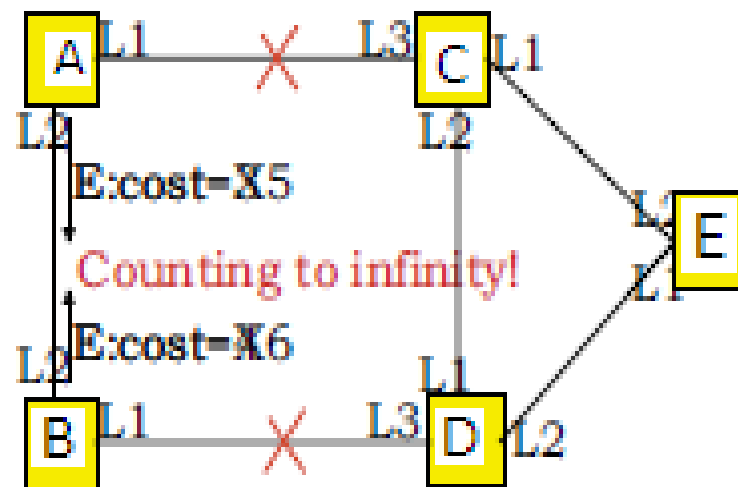
- Sada, predpostavimo da **dolazi do otkaza B-D linka**

- B** otkriva to stanje i **ažurira svoju ruting tabelu**

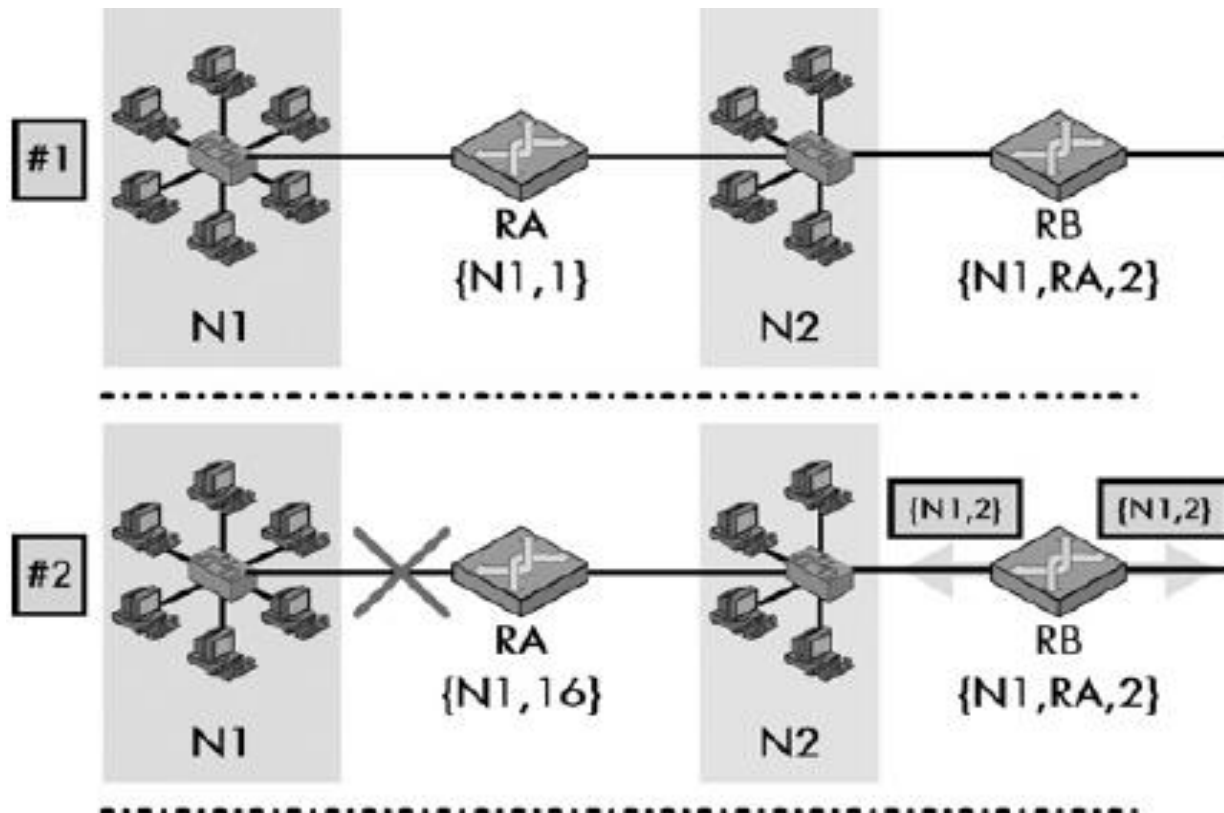
E: cost = ∞

- Šalje tu informaciju do A, **A ažurira ruting tabelu**

E: cost = ∞



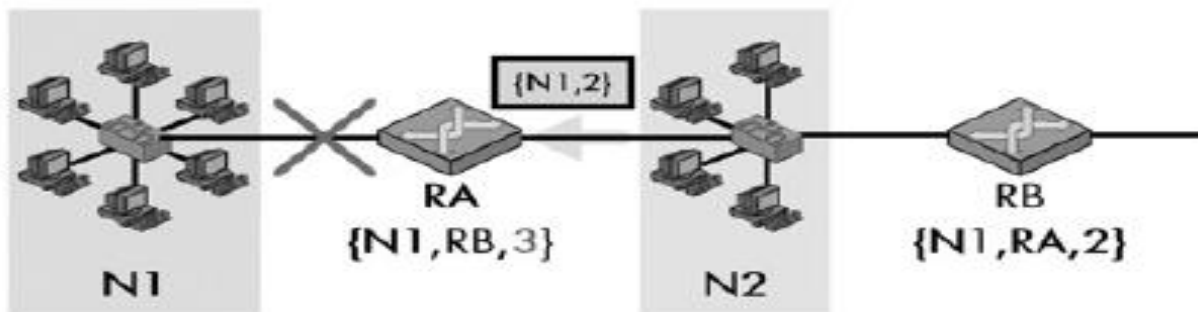
- Ali šta ako se desi oglašavanje od strane A rutera (ka B) **pre** oglašavanja rutera B (ka A)?



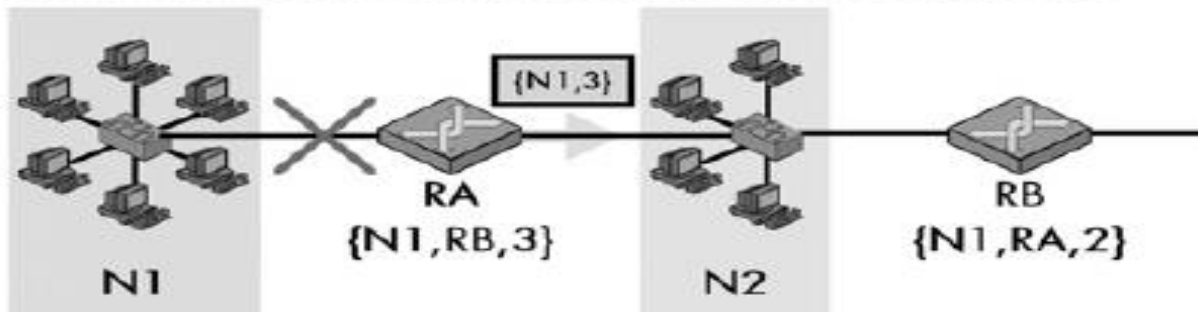
Problem: „Brojanje do beskonačnosti“

RB tajmer 30-sek.
pre RA sledeće difuzije

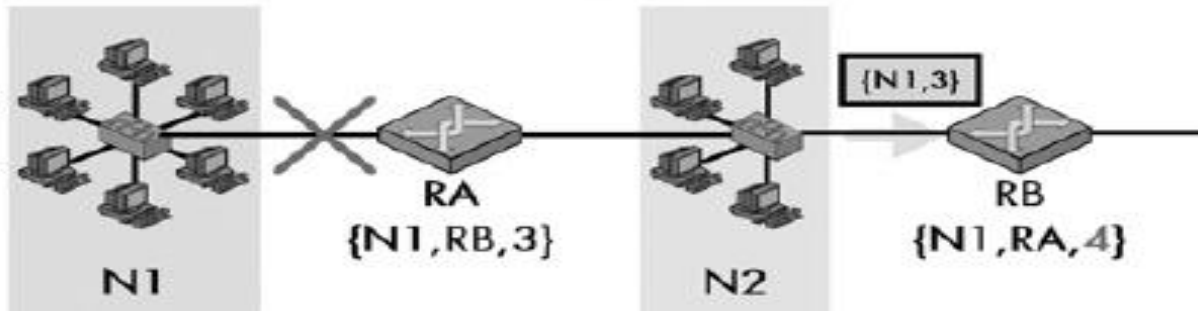
#3



#4

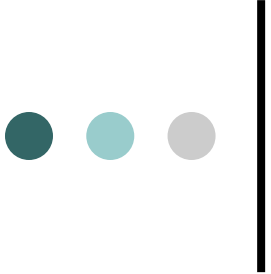


#5



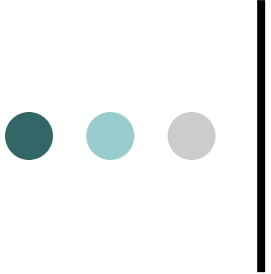
Problem:
spora konvergencija

Rešenje:
mala vrednost za
beskonačnost
(15 rutera)
Nedostatak!



Rešenje problema „brojanje do beskonačnosti“

- Podela horizonta
- Forsirano ažuriranje (triggered updates)
- Rutiranje na osnovu vektora putanje



Podela horizonta

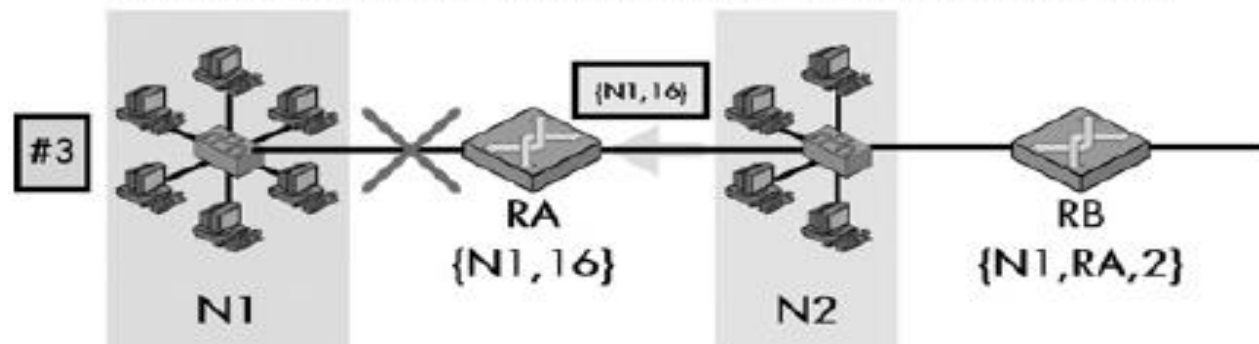
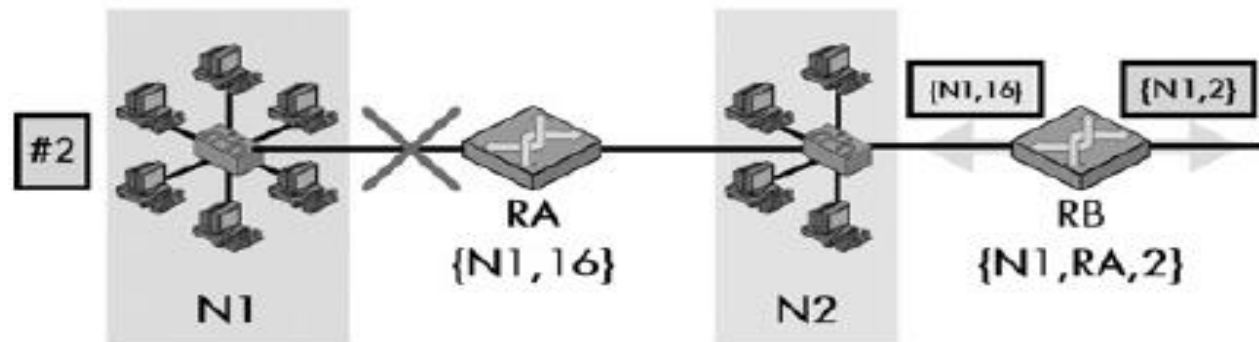
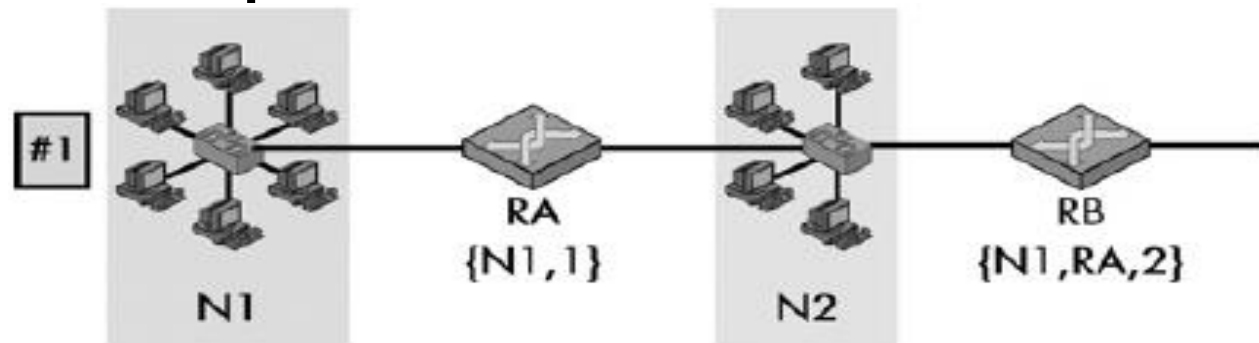
○ *Podela horizonta*

- ***Nema smisla slati oglašavanje*** ruteru (A) o nekom odredištu ako optimalna ruta do nje ide preko rutera (A)
- Što je upravo i ***uzrok problema***

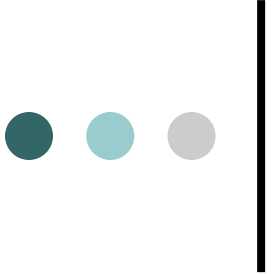
○ *Podela horizonta (poisoned reverse)*

- Ako želimo da naznačimo da se određena ruta ***ne sme nikako koristiti kao ruta*** ka određenoj destinaciji (zatrovati je!)

Podela horizonta



RA odbacuje
oglašavanje od strane
RB



Vektor putanje

Rešenje problema „brojanje do beskonačnosti“ sa rutiranjem na bazi vektora putanje

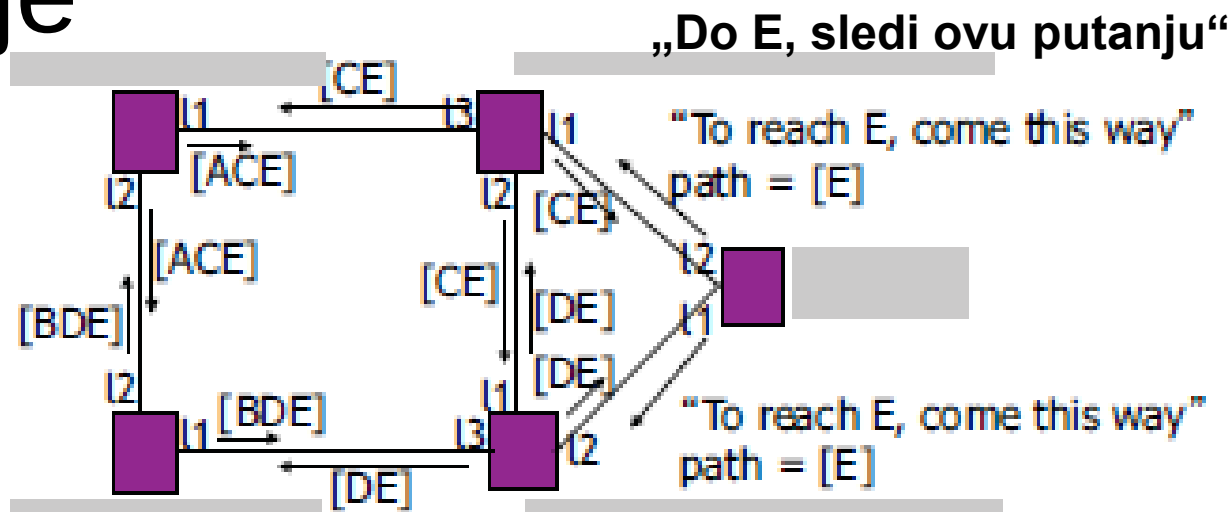
- Pored (ili umesto) izveštavanja o cenama, ***oglašava se putanja (dobijena inkrementalno Bellman-Ford-ovim pravilom)***

- Naziva se „vektor putanje“ (***path-vector***)

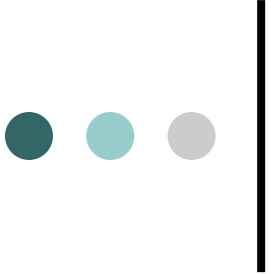
- Modifikuje se Bellman-Ford-ovo pravilo ažuriranja sa novim pravilom:

“tačka će ignorisati bilo koju oglašenu rutu koja sadrži adresu te tačke”

Rutiranje na bazi vektora putanje

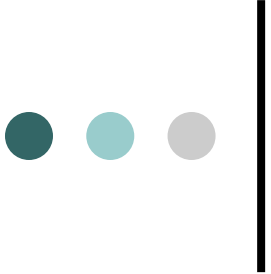


- **Nakon svakog oglašavanja**, izvršava se **„integracioni korak“**
 - Npr. Izaberi najkraću, najjeftiniju, najbržu,....
- **Ignorirati oglašavanja** sa vektorom putanje koja **sadrži sopstvenu adresu** u vektoru putanje
 - Radi izbegavanja petlji rutiranja koje dovode do probleme „brojanja od beskonačnosti“



Pregled

- ***Mrežni sloj*** implementira „lepak“ koji obezbeđuje ***povezanost tačaka u mreži***
- Čine ga: (I) ***adresiranje***, (II) ***prosleđivanje*** i (III) ***rutiranje***
- ***Prosleđivanje*** podrazumeva ***pretraživanje rutinng tabele***
 - Tabela rutiranja se formira u okviru protokola za rutiranje



Pregled

- **DV protokol**: **distribuiraju se** dobijene (izračunate) **rute**; svaka **tačka oglašava** do svojih suseda - njene **najbolje rute**
 - **Vektor- putanje**: **uključuje putanju**, ne samo cenu u oglašavanju kako bi se izbegao problem „brojanja do beskonačnosti“
- **LS protokol**: **distribuiraju** (plavi) **informacije o susedima** (neposredno okruženje)
 - **Centralizovana obrada** rutiranja informacija koristeći SP (shortest-path) algoritam