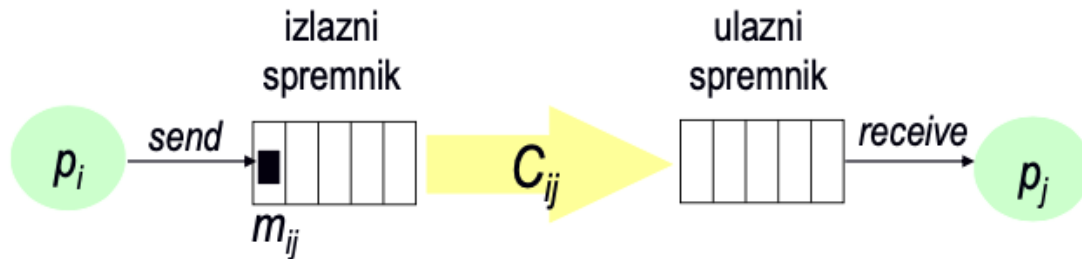


#### 4. preza - Formalni model raspodijeljenog sustava i primjeri raspodijeljenih algoritama

##### Model raspodijeljenog sustava

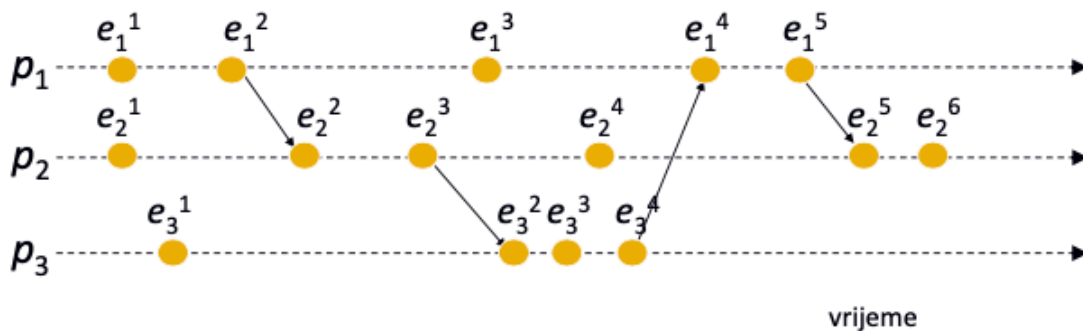
- skup autonomnih procesa  $p_1, p_2, \dots, p_n$
- $C$  - kanal koji povezuje procese
- $m$  - poruka između procesa

-Primjer slanja jedne poruke



Model raspodijeljenog izvođenja -  $e$  predstavlja događaj na nekom procesu, donji index predstavlja na kojem se procesu događa, a gornji redni broj događaja na tom procesu

## Primjer raspodijeljenog izvođenja



**Ovisnost** događaja označujemo s oznakom  $\rightarrow$   
Npr.  $e_{12} \rightarrow e_{22}$

Događaji su ovisni u 3. slučaja:

- dese se jedan za drugim na istom procesu
- slanje i primanje poruke
- tranzitivnost ( $e_{12} \rightarrow e_{22}$  i  $e_{22} \rightarrow e_{33}$ , tada bi vrijedilo  $e_{12} \rightarrow e_{33}$ )

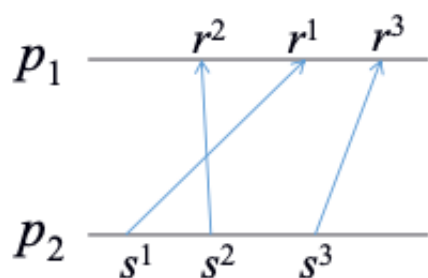
**Neovisnost** događaja označujemo s prekrštenom oznakom  $\rightarrow$

• Vrijede sljedeća pravila

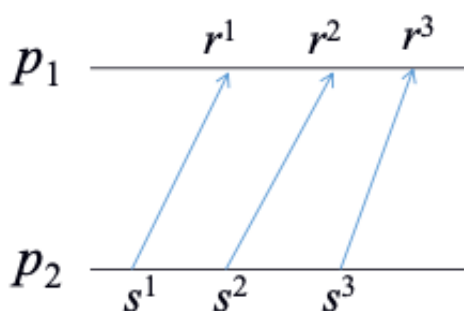
1. za 2 događaja  $e_i$  i  $e_j$ ,  $e_i \not\rightarrow e_j \not\Rightarrow e_j \not\rightarrow e_i$
2. za 2 događaja  $e_i$  i  $e_j$ ,  $e_i \rightarrow e_j \Rightarrow e_j \not\rightarrow e_i$
3. ako za 2 događaja  $e_i$  i  $e_j$ , vrijedi  $e_i \not\rightarrow e_j$  i  $e_j \not\rightarrow e_i$ , onda su  $e_i$  i  $e_j$  konkurenti događaji i to možemo napisati na sljedeći način  $e_i \parallel e_j$

Model komunikacijskog kanala

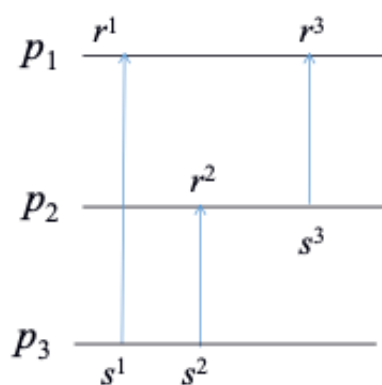
FIFO:



NON-FIFO:



Sinkrona slijednost:



S

Lokalna stanja - stanja svakog procesa koja se mijenjaju nekim događajem na tom procesu  
Globalno stanje - stanje cjelokupnog sustava određeno na temelju lokalnih stanja svih njegovih procesa u jednom točno definiranom trenutku

### Raspodijeljeni algoritam

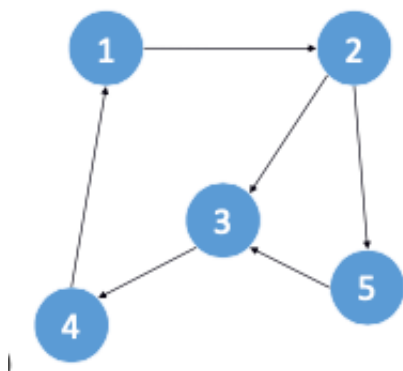
- alg. koji se izvodi u raspodijeljenoj okolini na 2 ili više procesa, samim time i na više računala zbog čega je vrlo složen i često težak za definirati

### Proširenje osnovnog modela rassusa:

#### 1. Sinkroni model

- pretpostavka-svi procesi rassusa izvode događaje istovremeno
- nije u teoriji izvedivo na kompleksnijim sustavima, ali se koristi za lakše razumijevanje
- procesu imaju sinkronizirana lokalna vremena

**Čvor predstavlja proces, a grana predstavlja kanal**



**Primjer 1:** Odabir vođe u sinkronom prstenu gdje je cvor proces

- svi procesi imaju svoj uuid i rade se algoritam pronalaska vođe -> procesa s najvećim uuid
- ako je primljeni uuid manji od mog, dalje šalji moj
- ako je primljeni uuid veci od mog, dalje šalje primljeni
- ako je primljeni uuid jednak mom, proglasi mene vođom i završi program

-složenost  $2n$

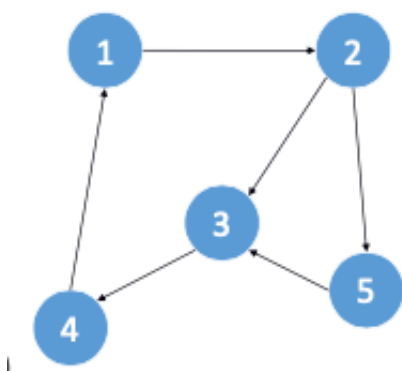
**Primjer 2:** Odabir vođe u usmjereoju mreži(kao na slici iznad) gdje je cvor proces

-složeniji primjer za koji treba uvesti i globalno stanje

#### 2. Asinkroni model

- pretpostavka-procesi izvode događaje u proizvoljnom slijedu
- realna situacija
- procesu nemaju sinkronizirana lokalna vremena

**Čvor predstavlja proces, a grana predstavlja kanal**



-ulazno/izlazni(I/O) automat

**Primjer 1:** Odabir vođe u asinkronoj mreži

-ovo je adaptacija sinkronog modela gdje sada ne primamo istovremeno jednu poruku već više, pa zbog toga svaki proces ima ulazni spremnik koji može primiti maksimalno  $n$  poruka

