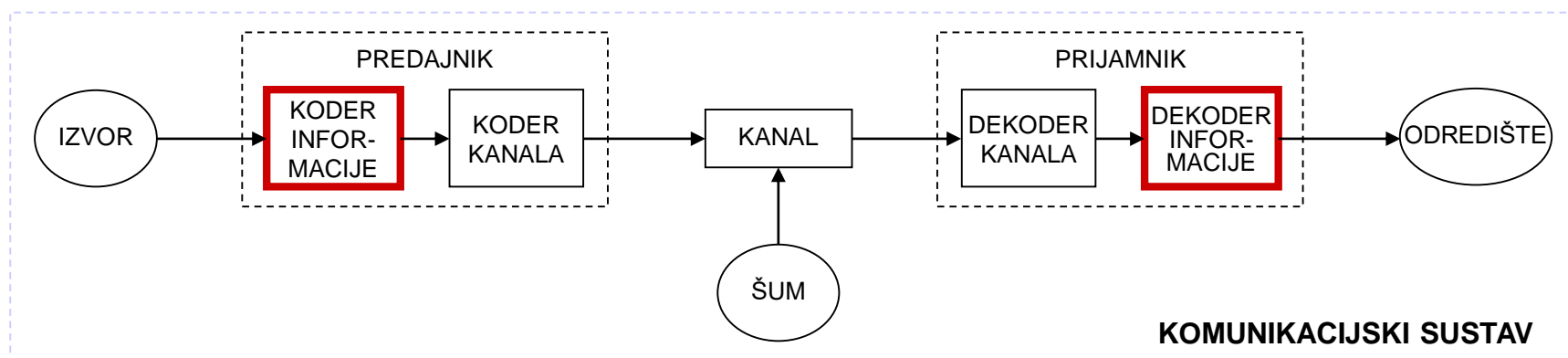


# Teorija informacije

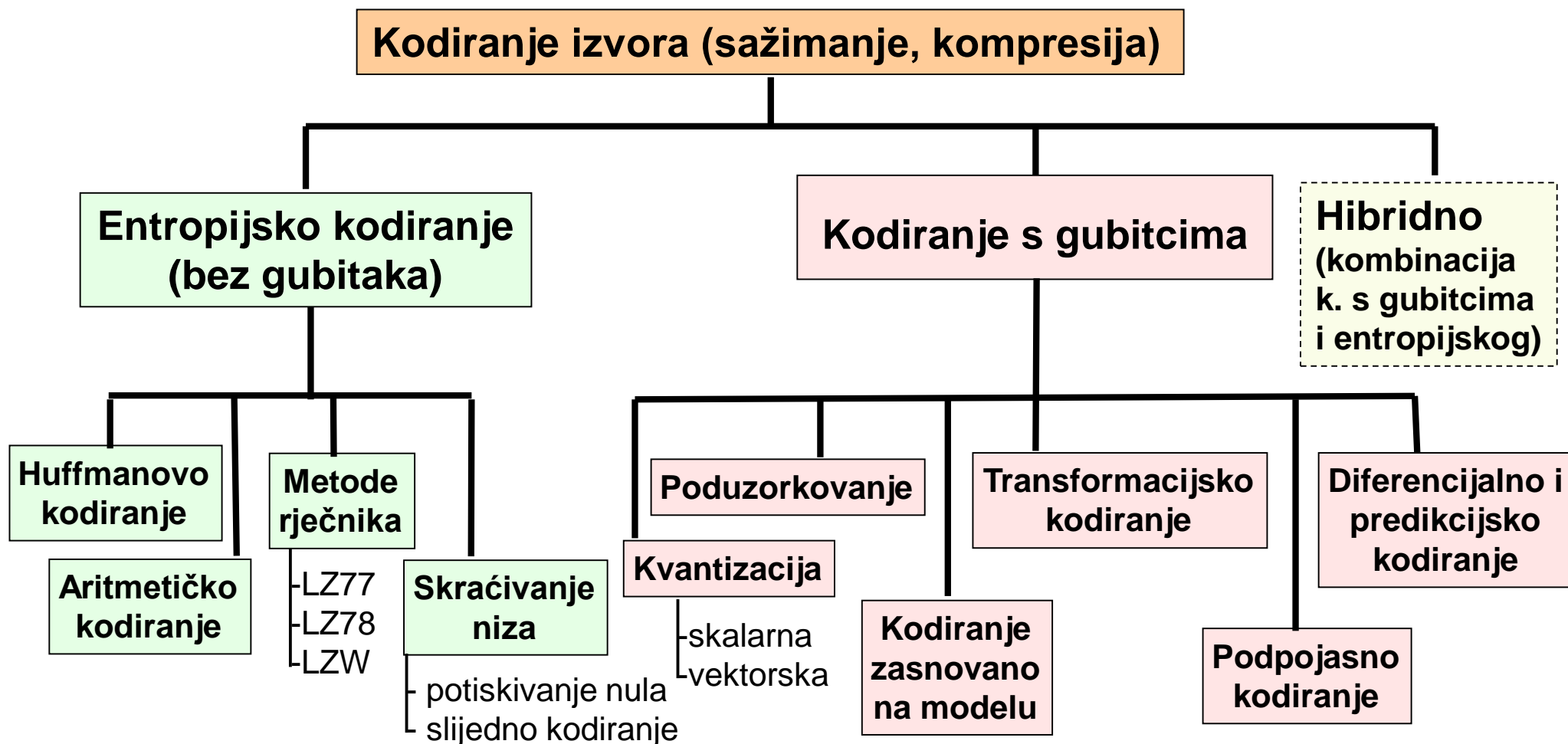
Entropijsko kodiranje

- ◆ Kodiranje: dodjela kodnih riječi simbolima poruke
- ◆ Kompresija: kodiranje koje smanjuje broj bitova potreban za izražavanje poruke
- ◆ U jasnom kontekstu, koristimo ove pojmove kao sinonime
- ◆ Kompresija se vrši u koderu informacije



- ◆ Uvod u kodiranje i kompresiju
  - Definicije, podjela metoda kompresije
  - Uvod u entropijsko kodiranje
- ◆ Karakteristike izvora informacije
  - Stacionarni izvor, ergodički izvor, izvori s memorijom (Markovljevi)
- ◆ Vrste kodova i njihova svojstva
  - Singularni, nesingularni, jednoznačno dekodabilni, prefiksni kodovi
- ◆ Optimalno kodiranje
- ◆ Metode entropijskog kodiranja
  - Shannon-Fanoovo kodiranje
  - Huffmanovo kodiranje
  - Aritmetičko kodiranje
  - Metode rječnika (LZ77, LZ78, LZW)
  - Metode skraćivanja niza (potiskivanje nula, slijedno kodiranje)

- ◆ Kompresija **bez gubitaka**
  - Komprimirani podaci mogu se dekomprimiranjem rekonstruirati bez gubitka informacije (*reverzibilno*)
  - Primjene: npr. tekst, medicinske slike, satelitske snimke
- ◆ Kompresija **s gubicima**
  - Cilj je ili dobiti najbolju vjernost rekonstruiranih podataka za zadanu brzinu (bit/s) ili postići najmanju brzinu za zadanu granicu vjernosti
  - Primjene: npr. govor, slika, video
- ◆ Važan parametar je **omjer kompresije**
  - Omjer veličine komprimiranih i originalnih podataka, npr. 1:10



- ◆ Osnovna ideja: skraćeno zapisati višestruko ili često ponavljane simbole ili nizove simbola
- ◆ Zajedničko svim metodama entropijskog kodiranja:
  - temelje se direktno na teoriji informacije
  - kodiranje bez gubitaka
  - omjer kompresije ovisi samo o statističkim svojstvima izvora informacije
  - poruka se promatra isključivo kao niz niz slučajnih vrijednosti, ne uzimaju se u obzir svojstva medija (za razliku od izvornog kodiranja)

# Karakteristike izvora informacije

# Matematički model izvora informacije

Zavod za telekomunikacije

- ◆ Izvor informacije promatramo kao stohastički proces, tj. niz slučajnih varijabli:  $X_1, X_2, \dots, X_t, \dots, X_n$ 
  - $A = \{\alpha_1, \dots, \alpha_m\}$  – abeceda izvora
  - Poruku od  $n$  znakova moguće je zapisati kao niz od  $n$  članova  $x = (a_1, a_2, \dots, a_n); a_i \in A, i = 1, \dots, n$ 
    - $x \in A^n, A^n = \{(a_1, a_2, \dots, a_n); a_i \in A, i = 1, \dots, n\}$
    - skup  $A^n$  (Kartezijev produkt skupa  $A$ ) je skup svih poruka od  $n$  članova abecede  $A$ ; njihov broj je  $m^n$
- ◆ Diskretni izvor  $(A, P)$  definiramo pomoću abecede  $A$  i razdiobe vjerojatnosti na skupu  $A^n$ :

$$P(x) = P(X_1 = a_1, X_2 = a_2, \dots, X_n = a_n), (a_1, a_2, \dots, a_n) \in A^n$$

$$\sum_{x \in A^n} P(x) = 1$$



- ◆ svaki simbol kojeg izvor generira ima određen stupanj neodređenosti (nepredvidivosti)
  - veća nepredvidivost slijeda znači i prijenos veće količine informacije po simbolu
  - ako izvor nametne određenu strukturu slijedu simbola, nepredvidivost se smanjuje, a s njom i količina informacije po simbolu
- ◆ izvor bez memorije
  - promatrano u različitim vremenskim trenucima nema korelacije između simbola na izlazu izvora
- ◆ izvori s memorijom
  - u bilo kojem trenutku simbol na izlazu izvora ovisi o jednom ili više simbola koji su generirani prije njega
  - primjer: markovljevi izvori

# Stacionarni izvor

- ◆ Statistička svojstva se ne mijenjaju s vremenom
- ◆ Matematički, za svaki par prirodnih brojeva  $n$  i  $k$  vrijedi:
$$P(X_{k+1} = a_1, X_{k+2} = a_2, \dots, X_{k+n} = a_n) = \\ = P(X_1 = a_1, X_2 = a_2, \dots, X_n = a_n) = P(a_1, a_2, \dots, a_n)$$
- ◆ Trivijalan primjer stacionarnog izvora:
  - generira naizmjenični slijed simbola A i E koji može započeti sa slovom A ili sa slovom E:
  - mogući ishodi procesa su:
  - AEAEAEAEAEAEAE...
  - EAEAEAEAEAEAEAE...
  - promatrano po vremenu, u svakom trenutku podjednaka je vjerojatnost nastupa simbola A i E

# Ergodični izvor

- ◆ Izvor kao skup svih mogućih proizvedenih nizova
  - Prosjek po skupu: prosjek pojavljivanja simbola na nekom mjestu u nizu, gledano među svim nizovima
  - Prosjek po vremenu: učestalost pojavljivanja simbola unutar pojedinog niza
- ◆ Ergodičnost: prosjek po skupu = prosjek po vremenu
- ◆ Svaki proizvedeni niz ima ista svojstva i ona se ne mijenjaju u vremenu
- ◆ Za entropijsko kodiranje promatramo ergodičke izvore (aproksimacija stvarnih izvora)

- ◆ Izvor počinje 1/3 sa A, 1/3 B i 1/3 E
  - Ako počne sa A ili B ponavlja ih izmjenično
  - Ako počne sa E, ponavlja samo E
  - Skup mogućih nizova:
    - Niz 1: ABABABABABABAB...
    - Niz 2: BABABABABABABA...
    - Niz 3: EEEEEEEEEEEEEEEE...

Simbol	Prosjek po vremenu za niz 1	Prosjek po vremenu za niz 2	Prosjek po vremenu za niz 3	Prosjek po skupu
A	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{3}$
B	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{3}$
E	0	0	1	$\frac{1}{3}$

- ♦ za  $n \geq 1$  moguće je definirati veličinu

$$H_n = \frac{1}{n} H(X_1, X_2, \dots, X_n) = -\frac{1}{n} \sum_{x \in A^n} P(a_1, \dots, a_n) \log P(a_1, \dots, a_n)$$

- ♦ a za  $n \geq 2$

$$h_n = H(X_n | (X_1, X_2, \dots, X_{n-1})) = - \sum_{x \in A^n} P(a_1, \dots, a_n) \log P(a_n | (a_1, \dots, a_{n-1}))$$

- u oba izraza vrijedi  $x = (a_1, a_2, \dots, a_n) \in A^n$

- ♦ moguće interpretacije:

- veličina  $H_n$  je srednja vlastita informacija simbola u  $n$ -članoj poruci
  - $h_n$  je uvjetna entropija  $n$ -tog simbola ako je poznato prethodnih  $n-1$  simbola

## ♦ Teorem:

- ako je niz  $h_n$  ( $n = 2, 3, \dots$ ) konvergentan,
- ako je niz  $H_n$  ( $n = 1, 2, \dots$ ) konvergentan,

■ tada vrijedi: 
$$\lim_{n \rightarrow \infty} h_n = \lim_{n \rightarrow \infty} H_n = H < \infty$$

- veličinu  $H_n \geq 0$  nazivamo **entropija** zadanog diskretnog stacionarnog **izvora informacije**
  - prosječna informacija koju "nosi" svaki pojedini simbol poslan iz zadanog izvora informacije

- ♦ ako diskretni stacionarni izvor emitira simbole iz  $A$  s jednakom vjerojatnošću, tj.  $P(\alpha_i) = p_i \geq 0$  u bilo kojem trenutku  $t_k$  tada vrijedi

$$\sum_{i=1}^m p_i = 1$$

$$P(a_1, a_2, \dots, a_n) = P(a_1) \cdot P(a_2) \cdot \dots \cdot P(a_n)$$

- ♦ nadalje vrijedi:  $P(a_n | (a_1, a_2, \dots, a_{n-1})) = \frac{P(a_1, a_2, \dots, a_n)}{P(a_1, a_2, \dots, a_{n-1})} = P(a_n)$

- ♦ ovakav se izvor naziva izvor bez memorije

- slanje simbola u sadašnjem trenutku stohastički je neovisno o prethodno poslanim simbolima

- ♦ entropija izvora bez memorije

$$H_n = \frac{1}{n} [H(X_1) + \dots + H(X_n)] = H(X_1) = - \sum_{i=1}^m p_i \log(p_i)$$

- također vrijedi:  $h_n = H_n$

# Markovljevi izvori - model izvora s memorijom



- ◆ Markovljevi lanci – podloga za Markovljeve izvore
  - Definicija
  - Matrica prijelaznih vjerojatnost
  - Dijagram stanja – grafički prikaz Markovljevog lanca
  - Ergodičnost i stacionarne vjerojatnosti
- ◆ Markovljevi izvori
  - Definicija
  - Vrste Markovljevih izvora
    - Unifilarni Markovljev izvor
    - Markovljev izvor  $m$ -tog reda
    - Jednostavni Markovljev izvor (Markovljev izvor prvog reda)
- ◆ Entropija Markovljevog izvora

# Markovljevi lanci

## ♦ Stohastički lanac:

- niz diskretnih slučajnih varijabli  $X_0, X_1, X_2, \dots$
- koje opisuju stanje nekog sustava u trenucima  $t_0, t_1, t_2, \dots$
- Skup svih stanja u kojima se lanac može nalaziti:  $S = \{\sigma_1, \sigma_2, \sigma_3, \dots\}$ 
  - Skup  $S$  može biti konačan ili beskonačan
  - Za modeliranje većine izvora pretpostavka je da je skup  $S$  konačan

## ♦ Definicija Markovljevog lanca:

- Lanac  $X_0, X_1, X_2, \dots$  je **Markovljev** ako za sve izbore stanja  $s_1, \dots, s_n \in S$  vrijedi  $P(X_{n+1} = s_{n+1} | X_n = s_n, \dots, X_0 = s_0) = P(X_{n+1} = s_{n+1} | X_n = s_n)$
- $t_n$  predstavlja sadašnjost
- **stanje u budućnosti ovisi samo o sadašnjem stanju**, ali ne i o načinu na koji je slučajni proces došao u to stanje

# Matrica prijelaznih vjerojatnosti

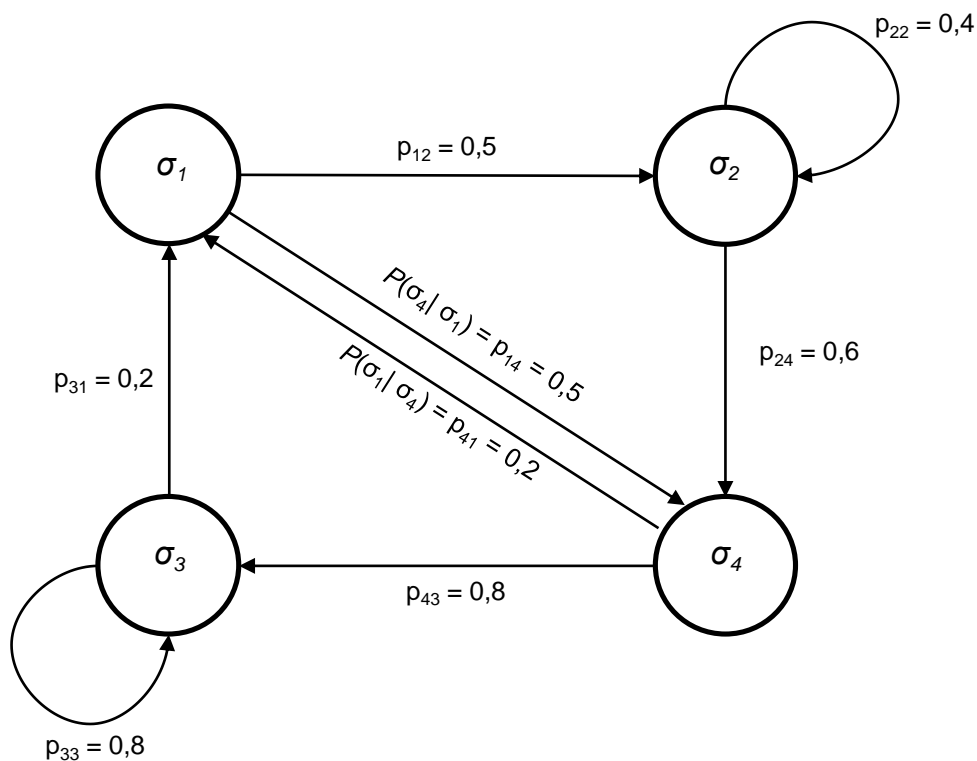
- ♦ Ako vrijedi:  $p_{ij} = P(X_{n+1} = \sigma_j | X_n = \sigma_i) = P(X_1 = \sigma_j | X_0 = \sigma_i)$ 
  - lanac je **homogen**, tj. **prijelazne vjerojatnosti se ne mijenjaju** - ovise samo o stanjima  $\sigma_i$  i  $\sigma_j$ , a ne o trenutku prijelaza
- ♦ Matrica prijelaznih vjerojatnosti daje vjerojatnosti prijelaza iz jednog stanja u drugo, u jednom koraku Markovljevog lanca
- ♦ Ako je skup stanja  $S = \{\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_N\}$ , tada je matrica prijelaznih vjerojatnosti matrica dimenzije  $N \times N$

$$\Pi = P(\sigma_j | \sigma_i) = [p_{ij}]$$
$$= \begin{bmatrix} P(\sigma_1 | \sigma_1) & P(\sigma_2 | \sigma_1) & \dots & P(\sigma_N | \sigma_1) \\ P(\sigma_1 | \sigma_2) & P(\sigma_2 | \sigma_2) & \dots & P(\sigma_N | \sigma_2) \\ \vdots & \vdots & \ddots & \vdots \\ P(\sigma_1 | \sigma_N) & P(\sigma_2 | \sigma_N) & \dots & P(\sigma_N | \sigma_N) \end{bmatrix}$$

stohastička matrica

$$P(\sigma_j | \sigma_i) \geq 0$$
$$\sum_{j=1}^N P(\sigma_j | \sigma_i) = 1, \forall i = 1, \dots, N$$

## ◆ Prikaz Markovljevog lanca grafom



$$\Pi = P(\sigma_j | \sigma_i) = [p_{ij}]$$

$$\Pi = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix}$$

$$\Pi = \begin{bmatrix} 0 & 0,5 & 0 & 0,5 \\ 0 & 0,4 & 0 & 0,6 \\ 0,2 & 0 & 0,8 & 0 \\ 0,2 & 0 & 0,8 & 0 \end{bmatrix}$$

# Svojstva matrice prijelaznih vjerojatnosti



Zavod za komunikacije

- ♦ Vjerojatnost da se izvor nalazi u nekom određenom stanju varira tijekom vremena
- ♦ Vjerojatnost da sustav pređe iz stanja  $\sigma_i$  u stanje  $\sigma_j$  u  $m$  koraka:

$$p_{ij}(m) = P(X_{n+m} = \sigma_j | X_n = \sigma_i)$$

- ♦ pri čemu vrijedi:  $p_{ij}(1) = p_{ij} = P(\sigma_j | \sigma_i)$  i  $\Pi(1) = \Pi$
- ♦ Vjerojatnosti  $p_{ij}(m)$  zadovoljavaju Chapman-Kolmogorovljeve jednačbe

$$p_{ij}(m) = \sum_k p_{ik}(r) p_{kj}(m-r), \forall r=1, 2, \dots, m-1$$

- ♦ odnosno u matričnom obliku  $\Pi(m) = \Pi(r) \Pi(m-r)$

$$\Pi(m) = \Pi^m$$

- ◆ Označimo vjerojatnost da se izvor u trenutku  $t_n$  nalazi u stanju  $\sigma_i$  kao  $p_i(n) := P(X_n = \sigma_i)$
- ◆ Tada je razdiobu tih vjerojatnosti moguće opisati vektorom

$$\mathbf{p}(n) = [p_1(n), p_2(n), \dots, p_N(n)] \quad \sum_{i=1}^N p_i(n) = 1, \forall n \in \mathbf{N}$$

- ◆ Ako je  $\mathbf{p}(0)$  **vektor početnih vjerojatnosti**, tada se stanje izvora u trenutku  $t_n$  može opisati kao

$$\mathbf{p}(n) = \mathbf{p}(0) \mathbf{\Pi}^n$$

- ◆ a vezu između dva uzastopna vremenska trenutka opisuje izraz  $\mathbf{p}(n) = \mathbf{p}(n-1) \mathbf{\Pi}$

# Stacionarne vjerojatnosti

Zavod za telekomunikacije

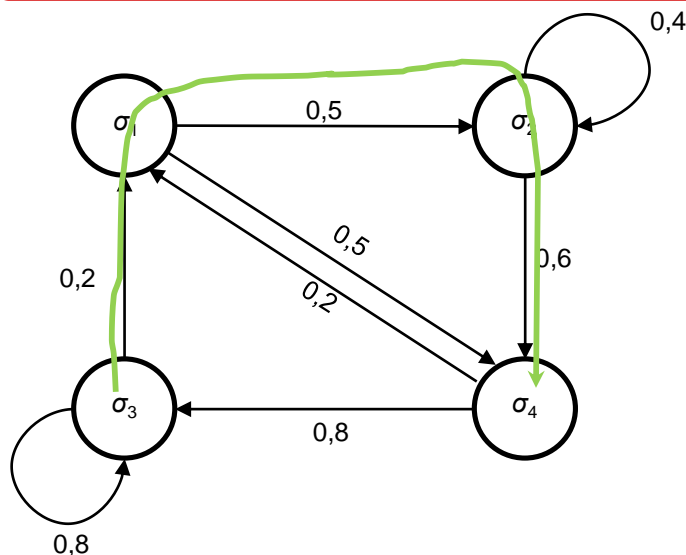
- ◆ Ponekad je važno poznavati ponašanje izvora nakon duljeg vremenskog razdoblja

- ◆ Ako postoji broj  $n$  takav da su svi elementi matrice  $\Pi^n$  strogo pozitivni (to znači da se u  $n$  koraka **može iz svakog stanja preći u bilo koje drugo stanje**), tada za svaki  $j$  postoji (i ne ovisi o  $i$ ):

$$\pi_j = \lim_{n \rightarrow \infty} p_{ij}(n)$$

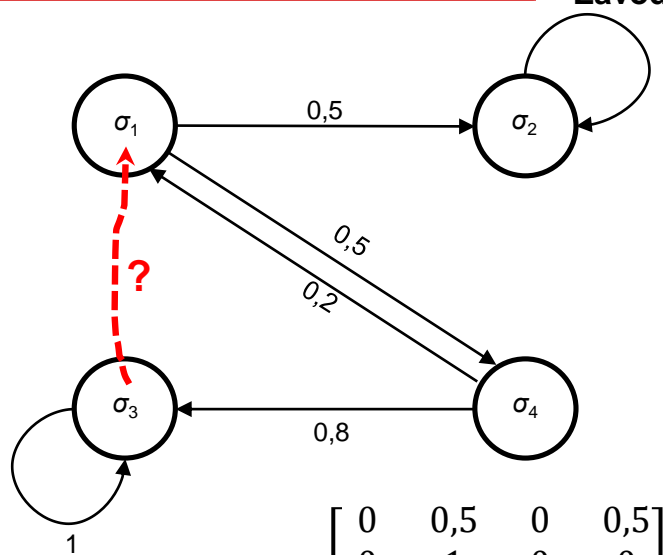
- ◆  $\pi_j$  su **stacionarne vjerojatnosti** - predstavljaju vjerojatnosti da će izvor u nekom dalekom trenutku (kad nestane utjecaj početnog stanja) generirati simbol  $j$
- ◆ Markovljev lanac za kojeg postoje stacionarne vjerojatnosti naziva se **ergodični ili regularan**

# Ergodičnost - primjeri



$$\Pi = \begin{bmatrix} 0 & 0,5 & 0 & 0,5 \\ 0 & 0,4 & 0 & 0,6 \\ 0,2 & 0 & 0,8 & 0 \\ 0,2 & 0 & 0,8 & 0 \end{bmatrix}$$

$$\Pi^2 = \begin{bmatrix} 0,1 & 0,2 & 0,4 & 0,3 \\ 0,12 & 0,16 & 0,48 & 0,24 \\ 0,16 & 0,1 & 0,64 & 0,1 \\ 0,16 & 0,1 & 0,64 & 0,1 \end{bmatrix}$$



$$\Pi = \begin{bmatrix} 0 & 0,5 & 0 & 0,5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0,2 & 0 & 0,8 & 0 \end{bmatrix}$$

$$\Pi^2 = \begin{bmatrix} 0,1 & 0,5 & 0,4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0,1 & 0,8 & 0,1 \end{bmatrix}$$

$$\Pi^3 = \begin{bmatrix} 0 & 0,55 & 0,4 & 0,05 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0,02 & 0,1 & 0,88 & 0 \end{bmatrix} \dots$$





- ♦ Početne jednadžbe:

$$\mathbf{p}(n) = \mathbf{p}(n-1)\mathbf{\Pi}$$

$$p_j(n) = \sum_k p_k(n-1) p_{kj}$$

- ♦ Ako postoje stacionarne vjerojatnosti, onda vrijedi:

$$\pi_j = \lim_{n \rightarrow \infty} p_j, \quad \pi_j = \sum_k \pi_k p_{kj}, \quad \forall j, \quad \sum_k \pi_k = 1$$

- ♦ ili u matričnom obliku:

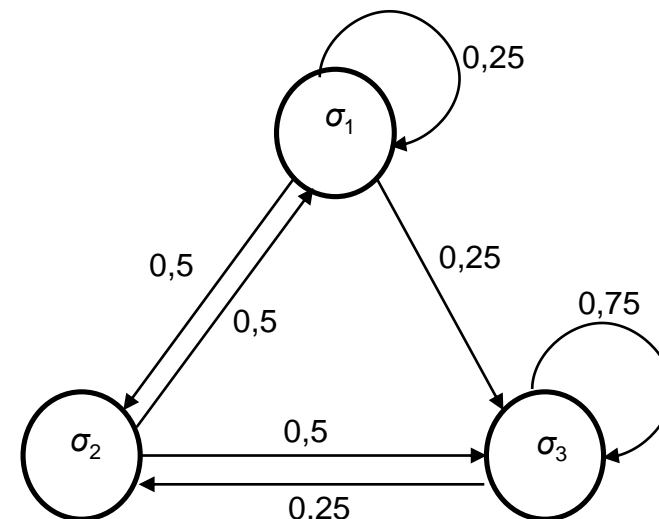
$$\mathbf{\Pi}^T \boldsymbol{\pi} = \boldsymbol{\pi}$$

# Primjer: Markovljev lanac sa tri stanja

- ◆ Zadana je matrica prijelaznih vjerojatnosti

$$\Pi = \begin{bmatrix} 0,25 & 0,5 & 0,25 \\ 0,5 & 0,0 & 0,5 \\ 0,0 & 0,25 & 0,75 \end{bmatrix}$$

- ◆ Dijagram stanja:

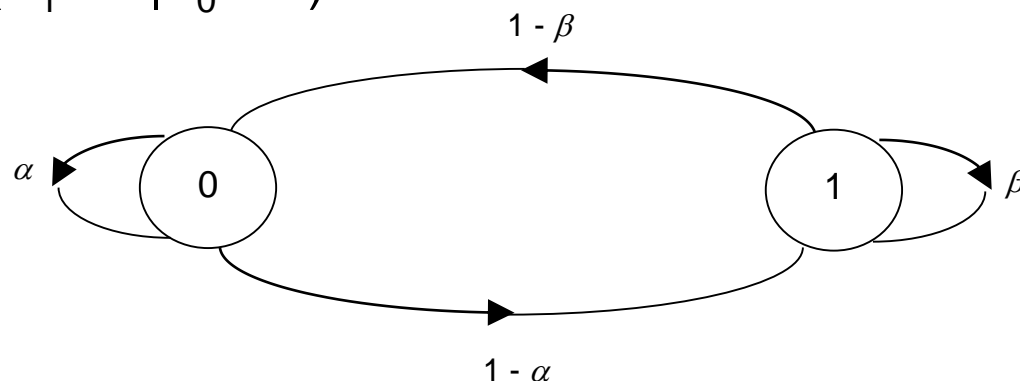


- ◆ Određivanje stacionarnih vjerojatnosti:

$$\Pi^T \boldsymbol{\pi} = \boldsymbol{\pi} \quad \Pi^T \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} = \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} \quad \begin{bmatrix} 0,25 & 0,5 & 0,0 \\ 0,5 & 0,0 & 0,25 \\ 0,25 & 0,5 & 0,75 \end{bmatrix} \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} = \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} \quad \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} = \begin{bmatrix} \frac{2}{13} \\ \frac{3}{13} \\ \frac{8}{13} \end{bmatrix}$$

# Primjer: Markovljev lanac s dva stanja

- ◆ Skup stanja  $S = \{0, 1\}$
- ◆ Zadane prijelazne vjerojatnosti:
  - $\alpha = P(X_1 = 0|X_0 = 0)$ ,  $\beta = P(X_1 = 1|X_0 = 1)$
  - $0 < \alpha, \beta < 1$
- ◆ Dijagram stanja:



- ◆ Vrijedi:
  - $1 - \alpha = P(X_1 = 1|X_0 = 0)$ ,  $1 - \beta = P(X_1 = 0|X_0 = 1)$
  - $0 < \alpha, \beta < 1$
- ◆ Stacionarne vjerojatnosti:  $\pi_1 = \frac{1 - \beta}{2 - \alpha - \beta}$ ,  $\pi_2 = \frac{1 - \alpha}{2 - \alpha - \beta}$

$$\Pi = \begin{bmatrix} \alpha & 1 - \alpha \\ 1 - \beta & \beta \end{bmatrix}$$

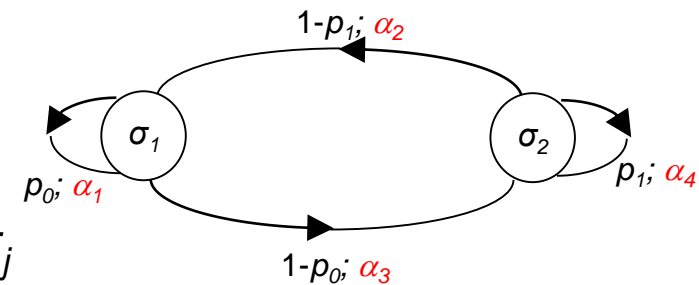
# Markovljevi izvori

Zavod za telekomunikacije

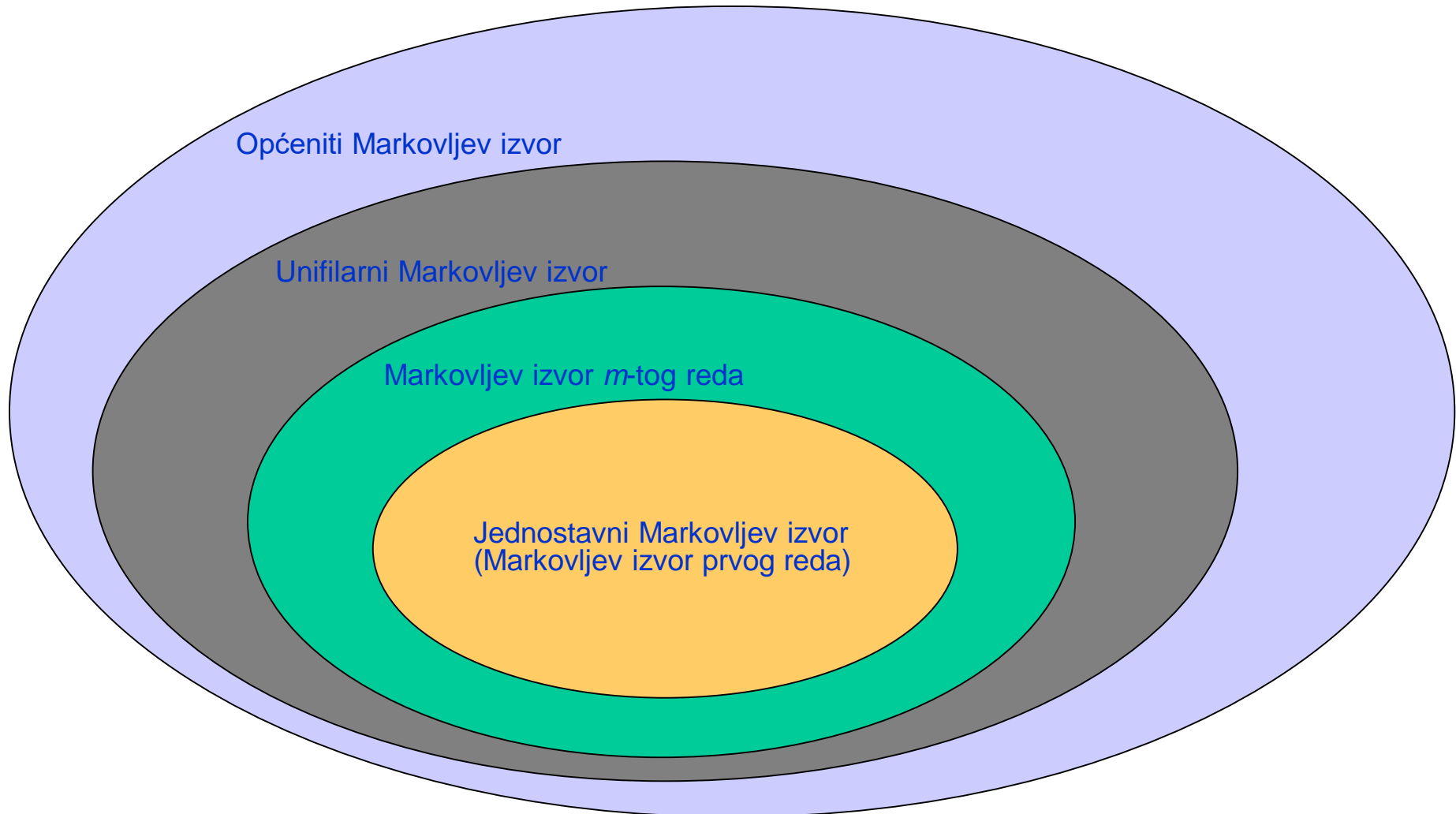
- ◆ Podsjetnik - izvor informacije:
  - Niz slučajnih varijabli  $X_1, X_2, \dots, X_n$  opisuje izlaze izvora u vremenu
  - $A = \{\alpha_1, \dots, \alpha_m\}$  – abeceda izvora
  - Poruka od  $n$  znakova:  $x \in A^n$  tj.  $x = (a_1, a_2, \dots, a_n)$ ;  $a_i \in A$ ;  $\sum_{x \in A^n} P(x) = 1$
- ◆ Podsjetnik - Markovljev lanac:
  - Niz slučajnih varijabli  $X_0, X_1, X_2, \dots$  opisuje stanja lanca u vremenu
  - $S = \{\sigma_1, \sigma_2, \sigma_3, \dots\}$  – skup stanja u kojima se lanac može nalaziti

## ◆ Markovljev izvor:

- izvor informacije čiju dinamiku određuje Markovljev lanac mapiranjem  $S \times S \rightarrow A$
- Drugim riječima, svaki prijelaz stanja iz  $\sigma_i$  u  $\sigma_j$  generira jedan simbol  $\alpha_k$  iz abecede izvora



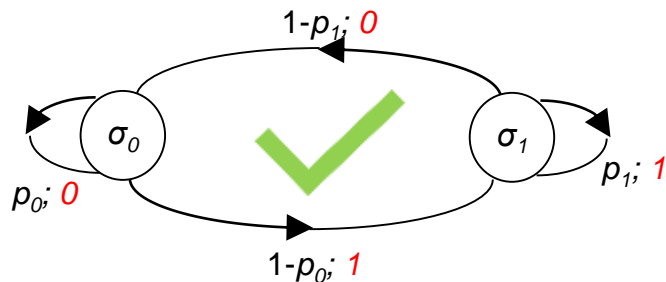
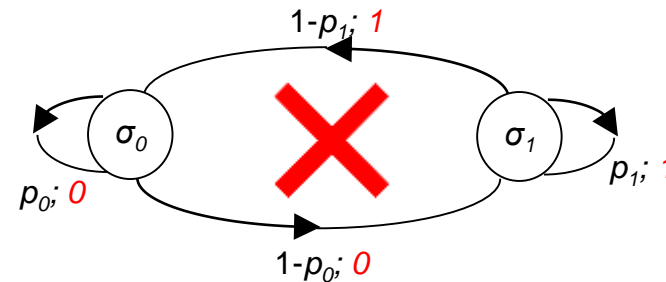
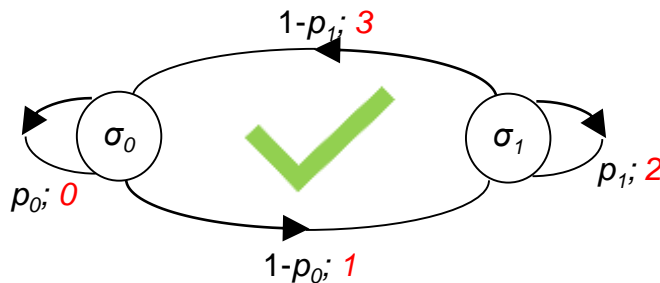
# Vrste Markovljevih izvora



# Unifilarni Markovljev izvor

Zavod za telekomunikacije

- Unifilarni MI je MI sa svojstvom da sve tranzicije iz pojedinog stanja generiraju različite simbole.



- Za unifilarni izvor, iz početnog stanja i niza generiranih simbola možemo jednoznačno odrediti niz stanja kojim su generirani ti simboli.
- U primjerima lijevo, početno stanje  $\sigma_0$  i niz simbola 00 jednoznačno određuju niz stanja  $\sigma_0\sigma_0$  ali u primjeru desno može biti  $\sigma_0\sigma_0$  ili  $\sigma_0\sigma_1$

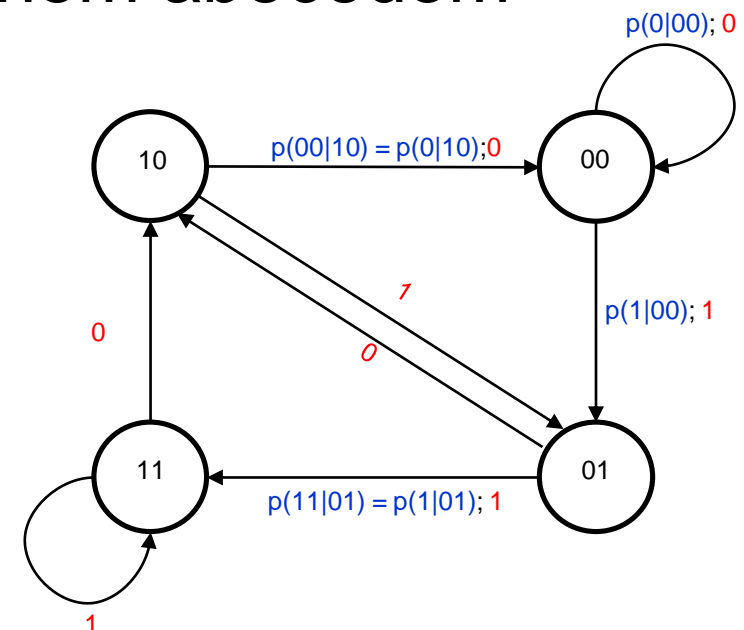
# Markovljev izvor $m$ -tog reda

Zavod za telekomunikacije

- ♦ MI  $m$ -tog reda je MI čija stanja su nizovi od posljednjih  $m$  generiranih simbola iz abecede izvora  $A$
- ♦ Primjer: MI drugog reda s binarnom abecedom
  - Abeceda izvora:  $A = \{0,1\}$
  - Skup stanja:  $S = \{00,01,10,11\}$

- ♦ Prijelazne vjerojatnosti Markovljevog lanca su ujedno i uvjetne vjerojatnosti izvora:

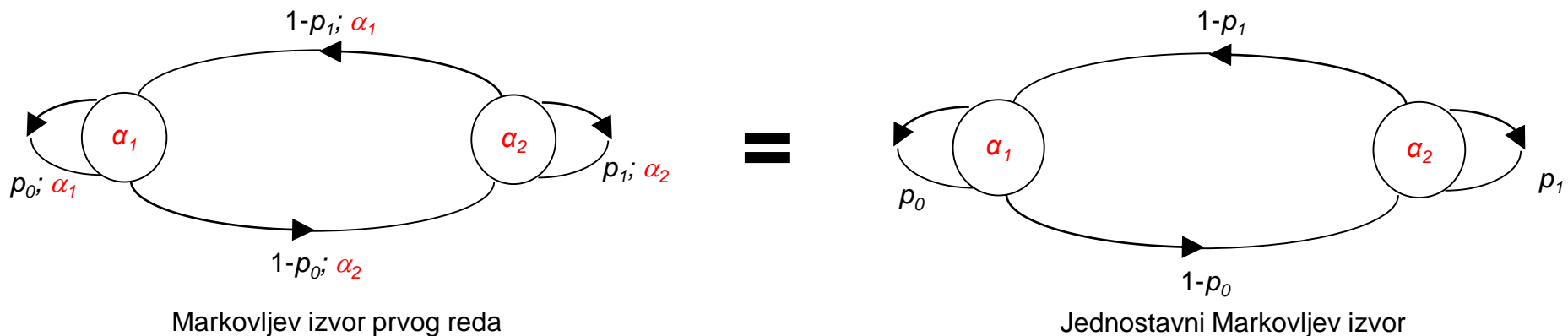
$$p(s_n | s_{n-1}) = p(a_n | (a_{n-m}, \dots, a_{n-1}))$$



# Jednostavni Markovljev izvor

Zavod za telekomunikacije

- ♦ Jednostavni MI je MI kod kojeg su stanja lanca ujedno i simboli izvora
- ♦ To je MI prvog reda, te ima memoriju prvog reda (pamti jedan simbol)





- ♦ Uvjetna entropija  $n$ -tog simbola za općeniti izvor:

$$h_n = H(X_n | (X_1, X_2, \dots, X_{n-1})) = - \sum_{x \in A^n} P(a_1, \dots, a_n) \log P(a_n | (a_1, \dots, a_{n-1}))$$

- ♦ Iz definicije Markovljevog lanca, za jednostavan Markovljev izvor vrijedi:

$$P(a_n | (a_1, \dots, a_{n-1})) = P(a_n | a_{n-1}), (a_1, \dots, a_n) \in A^n$$

- ♦ pa je entropija jednostavnog Markovljevog izvora:

$$H = \lim_{n \rightarrow \infty} (h_n) = - \sum_{i=1}^N p_i \sum_{j=1}^N p_{ij} \log(p_{ij})$$

# Kodiranje

# Kodiranje

Zavod za telekomunikacije

- ◆ Dodjela kodnih riječi simbolima poruke

$$X = \{x_1, x_2, \dots, x_i, \dots, x_n\}$$

$$x_i \in X \xrightarrow{\text{KODIRANJE}} C(x_i)$$

$$C(x_i) \in D^*, D = \{a_1, a_2, \dots, a_d\},$$

- ◆ Kodiranje sa svojstvom sažimanja: kompresija
- ◆ U praksi gotovo uvijek binarna abeceda
  - $d = 2, D = \{0, 1\}$
  - Izlaz koder: struja bitova (engl. *bitstream*)

# Prosječna duljina kodne riječi

Zavod za telekomunikacije

- ◆ Duljina pojedine kodne riječi:  $l(x_i)$ , skraćeno  $l_i$ 
  - broj simbola koji čine tu kodnu riječ
- ◆ Prosječna duljina kodne riječi (prosječna duljina koda):

$$L = \sum_{i=1}^n p(x_i) l(x_i) = \sum_{i=1}^n p_i l_i$$

- ◆ Za dugačku poruku od  $N$  simbola, očekivana duljina kodirane poruke je  $NL$
- ◆  $L$  [bit/simbol] je mjera efikasnosti koda

# Primjer kodiranja 1

SIMBOL ( $x_i$ )	VJEROJATNOST POJAVLJIVANJA $p(x_i) = p_i$	KODNA RIJEČ ( $C_i$ )	DULJINA KODNE RIJEČI ( $l_i$ )
1	1/2	0	1
2	1/4	10	2
3	1/8	110	3
4	1/8	111	3

♦ Prosječna duljina kodne riječi:

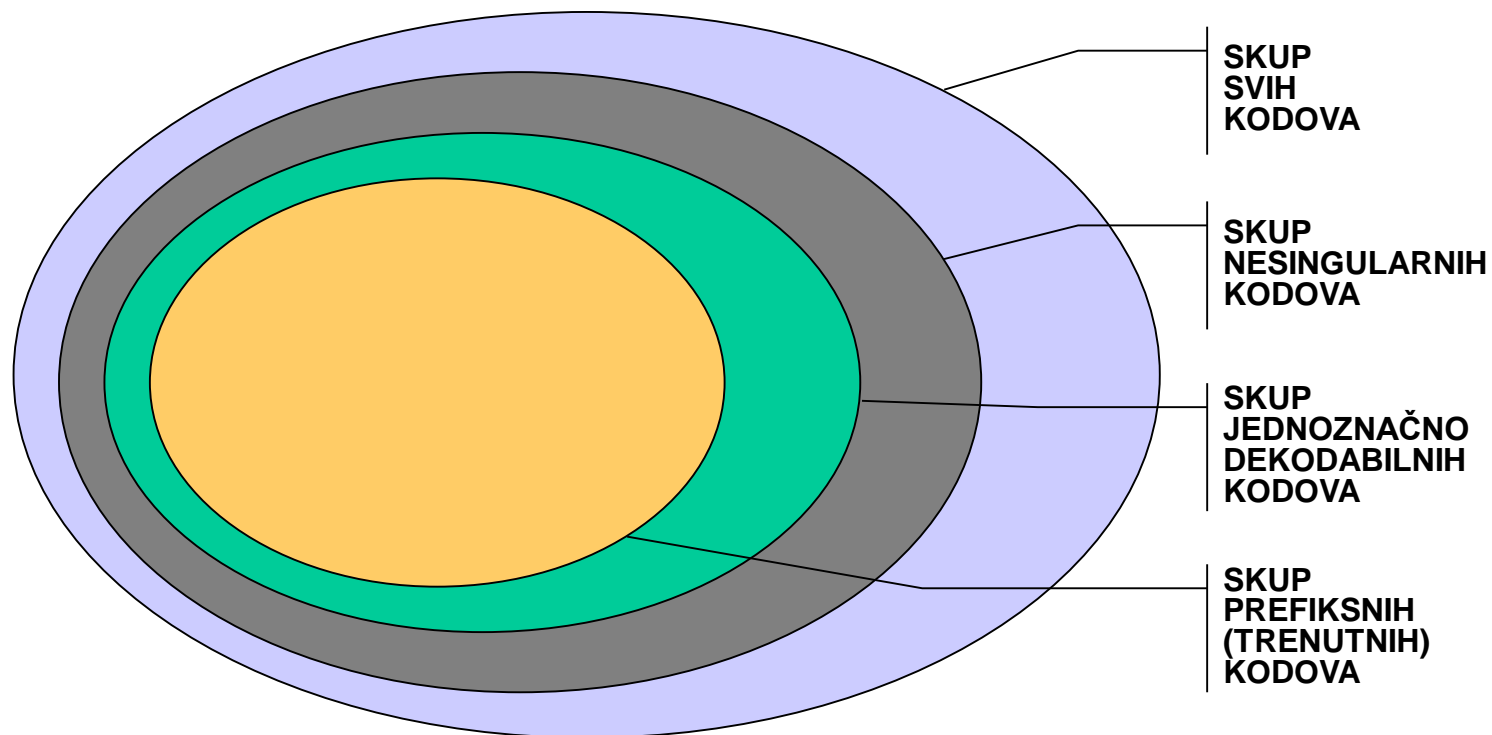
$$L = \sum_{i=1}^n p_i l_i = 0.5 \cdot 1 + 0.25 \cdot 2 + 0.125 \cdot 3 + 0.125 \cdot 3 = 1.75 [\text{bit} / \text{simbol}] = H(X)$$

# Primjer kodiranja 2

SIMBOL ( $x_i$ )	VJEROJATNOST POJAVLJIVANJA $p(x_i) = p_i$	KODNA RIJEČ ( $c_i$ )	DULJINA KODNE RIJEČI ( $l_i$ )
1	1/3	0	1
2	1/3	10	2
3	1/3	11	2

$$H(X) = -\sum_{i=1}^n p_i \log p_i = -\log \frac{1}{3} = 1.58 \text{ [bit/simbol]},$$

$$L = \sum_{i=1}^n p_i l_i = \frac{1}{3} \cdot 1 + \frac{1}{3} \cdot 2 + \frac{1}{3} \cdot 2 = 1.66 \text{ [bit/simbol]}.$$



# Nesingularni kodovi

- ◆ Kod je nesingularan ako svakom simbolu dodjeljuje drugačiju kodnu riječ

$$x_i \neq x_j \Rightarrow C(x_i) \neq C(x_j)$$

- ◆ To nije garancija jednoznačnosti
- ◆ Primjer:
  - Simboli A, B, C; kod:  $C(A) = 0$ ,  $C(B) = 01$  i  $C(C) = 1$
  - “ABC”  $\rightarrow$  “0011”
  - “0011”  $\rightarrow$  ?



# Jednoznačno dekodabilni kodovi

Zavod za telekomunikacije

$$x \xrightarrow{KOD} C(x)$$

$$x_1 x_2 \dots x_n \xrightarrow{PROŠIRENIKOD} C(x_1 x_2 \dots x_n) = C(x_1) C(x_2) \dots C(x_n)$$

- ◆ Kod jednoznačno dekodabilan ako je proširenje nesingularno

- Različite poruke  $\rightarrow$  različite kodirane poruke

- ◆ Primjer:

- Simboli A, B, C; kod:  $C(A) = 0$ ,  $C(B) = 01$  i  $C(C) = 011$

- “ABC”  $\rightarrow$  “001011”  $\rightarrow$  “ABC”

- “001...”  $\rightarrow$  ?

- ◆ Ne može se trenutno dekodirati

# Prefiksni (trenutni) kodovi

Zavod za telekomunikacije

- ◆ Prefiksni kod je kod u kojem niti jedna kodna riječ nije prefiks neke druge kodne riječi
- ◆ Svaka kodna riječ se može trenutno dekodirati, bez znanja iduće kodne riječi
- ◆ U prethodnom primjeru, problem je upravo u tome što su kodne riječi jedna drugoj prefiks

# Vrste kodova: primjer

SIMBOL ( $x_i$ )	VRSTA KODA			
	SINGULARNI	NESINGULARNI	JEDINSTVENO DEKODABILNI	PREFIKSNI
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111
“1234” →	0000	00100110	100011110	010110111
Dekodirano	?	?	1234	1234
Prvih 6 simbola	?	?	? (123 ili 124)	123

# Kraftova nejednakost

Zavod za telekomunikacije

- ♦ Za svaki prefiksni kod sa abecedom od  $d$  simbola i duljinama kodnih riječi  $l_1, l_2, \dots, l_n$  vrijedi:

$$\sum_{i=1}^n d^{-l_i} \leq 1$$

i obrnuto, za bilo koji skup duljina kodnih riječi  $l_i$  koje zadovoljavaju ovu nejednakost, postoji prefiksni kod s takvim duljinama kodnih riječi.

- ♦ Određuje minimalne duljine kodnih riječi potrebne za prefiksni kod

## 1. Prethodni primjer koda {0, 10, 110, 111}

- Binarna abeceda,  $d=2$

$$\sum_{i=1}^n 2^{-l_i} \leq 1$$

$$2^{-1} + 2^{-2} + 2^{-3} + 2^{-3} = 1$$

- Nema kraćeg koda

## 2. Tražimo kod za tri simbola

$$2^{-1} + 2^{-2} + 2^{-2} = 1 \Rightarrow \text{mora postojati pref. kod duljina } 1, 2, 2$$

- ♦ Općenito, više kodova zadovoljava K.N.; koji je optimalan?
  - npr: {0, 10, 110, 111}, {111, 0, 10, 110}...
- ♦ Optimalan kod: prefiksni kod sa najmanjom mogućom prosječnom duljinom kodne riječi

$$\min \left[ L = \sum_{i=1}^n p_i l_i \right] \text{ uz uvjet } \sum_{i=1}^n d^{l_i} \leq 1$$

- ♦ Minimum se dobiva za:

$$l_i^* = -\log_d p_i \Rightarrow L = -\sum_{i=1}^n p_i \log_d p_i = H(X)$$

- ♦ Ali  $l_i$  moraju biti cijeli brojevi, pa se ne može uvijek postići  $L=H$ :

$$L \geq H(X)$$

- ♦ Za optimalni kod, prosječna duljina kodne riječi je unutar jednog bita od entropije:  $H(X) \leq L < H(X) + 1$

- ♦ Efikasnost koda:  $\varepsilon = \frac{H(X)}{L}$

- ◆ Shannon-Fanoovo kodiranje
- ◆ Huffmanovo kodiranje
  - optimalno kodiranje
  - binarno stablo
  - kraći zapis čestih znakova
- ◆ Aritmetičko kodiranje
  - poopćenje Huffmanovog kodiranja
  - cijela poruka se pretvara u jednu kodnu riječ
- ◆ Metode rječnika
  - isti rječnik kodnih riječi na strani pošiljatelja i primatelja
  - dinamička konstrukcija rječnika
  - Lempel-Ziv (LZ77, LZ78), Lempel-Ziv-Welch (LZW)
- ◆ Metode skraćivanja niza
  - potiskivanje nula, slijedno kodiranje



- ◆ Jedna je od prvih metoda kodiranja utemeljenih na teoriji informacije
- ◆ Ne daje uvijek optimalan kod
  - Vrlo rijetko se koristi
- ◆ Zasniva se na željenim svojstvima kôda:
  - Niti jedna kodna riječ ne smije biti prefiks neke druge kodne riječi;
  - Želimo da se u kodiranim porukama simboli 0 i 1 pojavljuju s podjednakom vjerojatnošću.

- ◆ Posložiti simbole po padajućim vjerojatnostima
- ◆ Podjela simbola u grupe
- ◆ Dodjela znamenke 0 jednoj, a 1 drugoj grupi
- ◆ Postupak se ponavlja dok se grupe ne svedu na 1 simbol

# Shannon-Fanoovo kodiranje: primjer

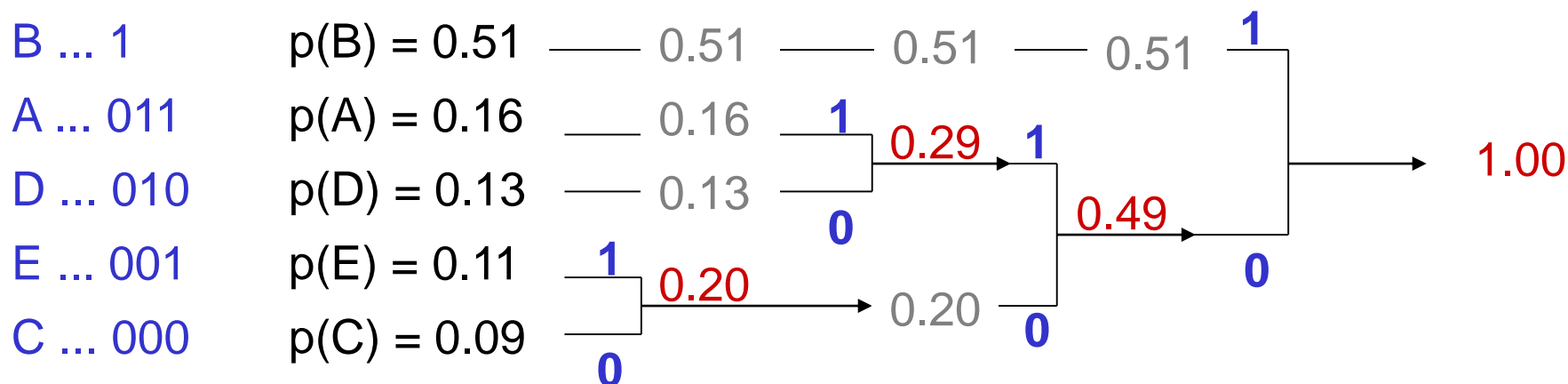
$x_i$	$p(x_i)$	KORAK 1	KORAK 2	KORAK 3	KORAK 4	KODNA RIJEČ	DULJINA KODNE RIJEČI
$x_1$	0.25	0	0			00	2
$x_2$	0.25	0	1			01	2
$x_3$	0.125	1	0	0		100	3
$x_4$	0.125	1	0	1		101	3
$x_5$	0.0625	1	1	0	0	1100	4
$x_6$	0.0625	1	1	0	1	1101	4
$x_7$	0.0625	1	1	1	0	1110	4
$x_8$	0.0625	1	1	1	1	1111	4
Prosječna duljina kodne riječi:							2.75

- ◆ D. A. Huffman, 1952. godine
- ◆ Kodira pojedinačne simbole kodnim riječima promjenjive duljine, ovisno o (poznatim!) vjerojatnostima njihova pojavljivanja
- ◆ Temelji se na dvije jednostavne činjenice:
  - (1) U optimalnom kodu, simboli s većom vjerojatnošću pojavljivanja imaju kraće kodne riječi od onih s manjom vjerojatnošću
  - (2) U optimalnom kodu, dva simbola s najmanjim vjerojatnostima imaju kodne riječi jednake duljine (vrijedi za prefiksni kod)
- ◆ Ishod: sažetiji zapis (npr. tipičan tekst se sažima za 45%)

- ◆ Algoritam stvaranja koda:
  1. Sortiraj simbole po padajućim vjerojatnostima
  2. Pronađi dva simbola s najmanjim vjerojatnostima
  3. Jednom od njih dodijeli simbol “0”, drugom “1”
  4. Kombiniraj ta dva simbola u jedan nadsimbol (nadsimbol je novi simbol čija je vjerojatnost pojavljivanja jednaka zbroju vjerojatnosti pojavljivanja dvaju simbola od kojih je nastao) i zapiši ih kao dvije grane binarnog stabla, a nadsimbol kao račvanje iznad njih
  5. Ponavljaj 1-4 dok ne dobiješ samo jedan nadsimbol
  6. Povratkom kroz stablo očitaj kodove
- ◆ Podatkovna struktura algoritma je binarno stablo
- ◆ Algoritam dekodiranja koristi isti postupak za gradnju stabla
  - Dekoder mora znati vjerojatnosti pojavljivanja simbola

# Huffmanovo kodiranje: primjer

- ♦ Skup simbola {A, B, C, D, E} s vjerojatnostima pojavljivanja  $p(A) = 0.16$ ,  $p(B) = 0.51$ ,  $p(C) = 0.09$ ,  $p(D) = 0.13$ ,  $p(E) = 0.11$
- ♦ Za uniformni kod, prosječna duljina koda je **3 bit/simbol** (jer je  $2^2 \leq 5 \leq 2^3$ ).
- ♦ Entropija: **1.96 bit/simbol**



- ♦ Prosječna duljina dobivenog koda u našem slučaju je:

$$L = \sum_{x \in X} p_x l_x = 3 \times (0.09 + 0.11 + 0.13 + 0.16) + 0.51 = 1.98 \text{ bit/simbol}$$

- ◆ kodiranje je idealno ako su vjerojatnosti  $1/2, 1/4, \dots, 1/2^n$
- ◆ u stvarnim slučajevima to obično nije slučaj, te rezultat ovisi o vjerojatnostima pojavljivanja simbola
- ◆ prednosti:
  - jednostavan za izvedbu
  - vrlo dobro kodiranje za „dobre“ vjerojatnosti pojavljivanja simbola
- ◆ nedostaci:
  - vjerojatnosti pojavljivanja simbola moraju biti poznate; ovise o primjeni (tekst, slika)
  - za “loše raspoređene” vjerojatnosti pojavljivanja dobiju se izrazito loši kodovi

# Primjer lošeg koda i prošireni Huffmanov kod

Simbol	Vjerojatnost	Kodna riječ
$a_1$	0.95	0
$a_2$	0.02	10
$a_3$	0.03	11

PROSIRENI KOD		
Simbol	Vjerojatnost	Kodna riječ
$a_1a_1$	0.9025	0
$a_1a_2$	0.0190	111
$a_1a_3$	0.0285	100
$a_2a_1$	0.0190	1101
$a_2a_2$	0.0004	110011
$a_2a_3$	0.0006	110001
$a_3a_1$	0.0285	101
$a_3a_2$	0.0006	110010
$a_3a_3$	0.0009	110000

- ◆ Entropija: 0.335 bit/simbol
- ◆ Prosječna duljina: 1.05 bit/simbol: **213% više od entropije!!**
- ◆ Prošireni kod:  $1.222 / 2 = 0.611$  bit/simbol: 72% više od entropije.
- ◆ Bolje je kodirati duže sekvence, ali tada broj kodnih riječi raste eksponencijalno

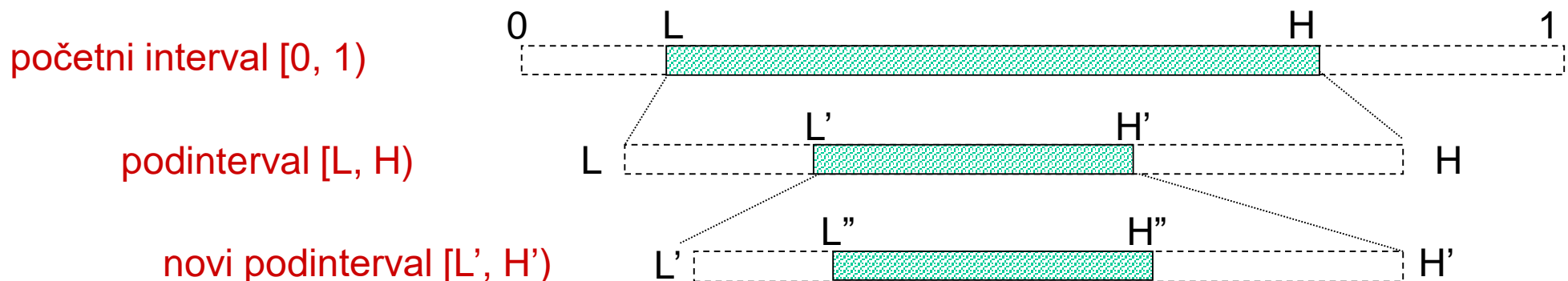


- ◆ Česta primjena unutar složenijih algoritama
- ◆ Primjeri:
  - standardi za telefaks (T.4, T.6)
  - standard za nepomičnu sliku JPEG

- ♦ Autori Pasco & Rissanen (nezavisno), 1976. godine
- ♦ Algoritam uzima kao ulaz cijele nizove simbola (“poruke”) i preslikava ih na realne brojeve, ovisno o (poznatim!) statističkim svojstvima

# Aritmetičko kodiranje: postupak

1. Podijeli interval  $[0, 1)$  u  $n$  podintervala koji odgovaraju simbolima iz abecede; duljina svakog podintervala proporcionalna vjerojatnosti odgovarajućeg simbola
2. Iz promatranog skupa podintervala, odaberi podinterval koji odgovara sljedećem simbolu u poruci
3. Podijeli taj podinterval u  $n$  novih podintervala, proporcionalno vjerojatnostima pojavljivanja simbola iz abecede; tako nastaje novi skup podintervala koji promatramo
4. Ponavljaj korake 2 i 3 dok cijela poruka nije kodirana
5. Konačni kod za čitavu poruku je jedan broj iz intervala u binarnom obliku



# Aritmetičko kodiranje: primjer (1)

- $M=2$
- simboli: X, Y  
 $p(X) = 2/3$   
 $p(Y) = 1/3$
- poruka duljine 2  
(moguće poruke  
XX, XY, YX, YY)  
kodira se onim  
brojem bita  
dovoljnim za  
jedinstveno  
određivanje  
intervala  
(binarni razlomak!)

Poruka duljine 2

Bilo koji broj iz intervala može poslužiti kao kodna riječ

Kodna riječ (binarno)

0		0		
			1/4	0.01
X 2/3	XX	4/9		
			2/4	0.1
	XY	6/9		
			3/4	0.11
Y 1/3	YX	8/9		
			15/16	0.1111
	YY	1		
1				

# Aritmetičko kodiranje: primjer (2)

- primjer za poruku duljine 3

- $M=2$

- simboli:

X, Y

$$p(X) = 2/3$$

$$p(Y) = 1/3$$

Kodna riječ  
(binarno)

X	XX	XXX	←	1/4	0.01
		XXY	←	3/8	0.11
	XY	XYX	←	4/8	0.100
		XYY	←	10/16	0.1010
Y	YX	YXX	←	6/8	0.110
		YXY	←	14/16	0.1110
	YY	YYX	←	15/16	0.1111
		YYY	←	31/31	0.11111

1. Podijeli početni interval  $[0, 1)$  u podintervale po vjerojatnostima pojavljivanja simbola
2. Uzmi primljeni kod kao realni broj
3. Pronađi podinterval u kojem se nalazi broj (kod)
4. Zapiši simbol koji odgovara tom podintervalu
5. Podijeli taj podinterval u  $n$  novih podintervala, proporcionalno vjerojatnostima pojavljivanja simbola iz abecede; tako nastaje novi skup podintervala koji promatramo
6. Ponavljaj korake 3-5 dok ne dođe kraj poruke

# Dekodiranje: primjer

- primjer za poruku duljine 3

- $M=2$

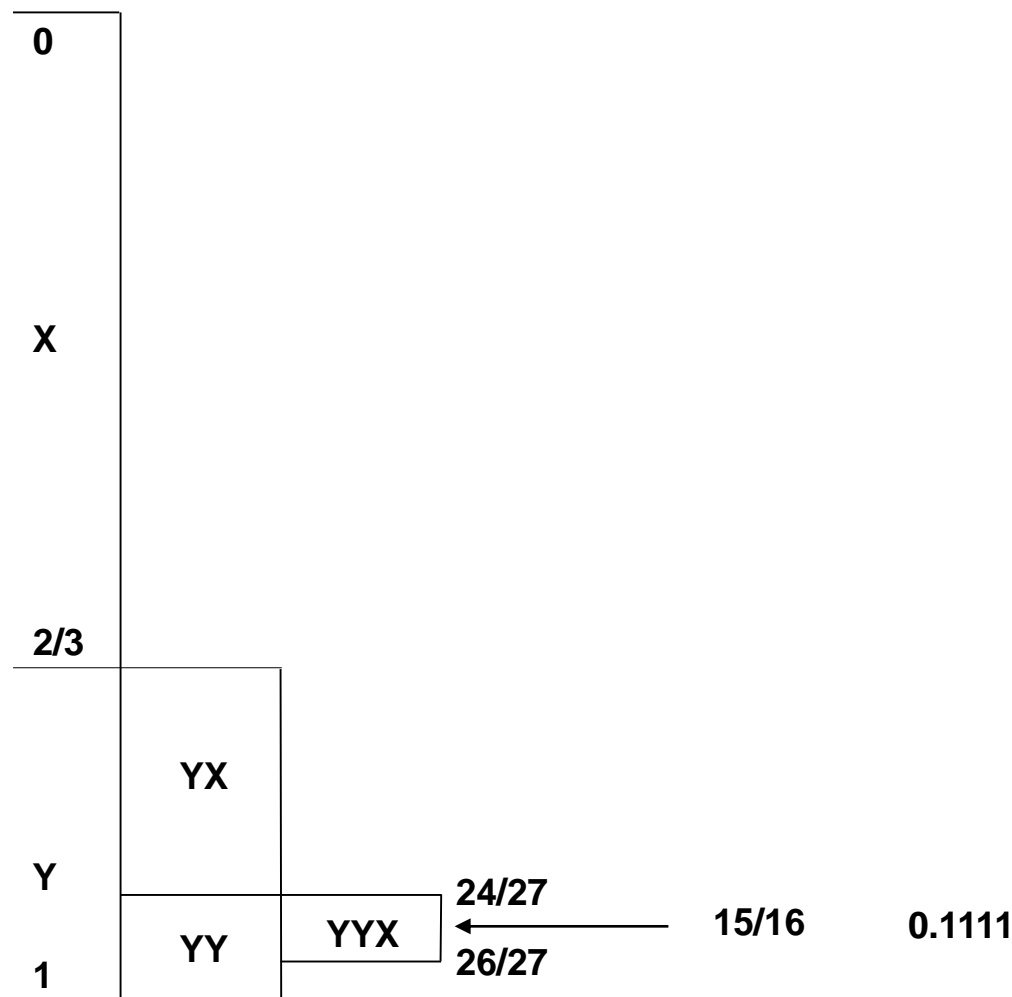
- simboli:

X, Y

$$p(X) = 2/3$$

$$p(Y) = 1/3$$

- Primljeni kod 1111  
tj. 15/16



- ♦ Kojim brojem iz podintervala kodirati poruku?
- ♦ Može se uzeti bilo koja vrijednost iz podintervala
- ♦ Dovoljan broj znamenki:

$$l(x) = \left\lceil \log \frac{1}{P(x)} \right\rceil + 1 \text{ [bit]}$$

- ♦ Na ovakav način dobiva se uvijek prefiksni kod



- ◆ Do sada opisani algoritam neupotrebljiv
  - Neprihvatljivo čekanje do kraja poruke
  - Algoritam podrazumijeva beskonačnu preciznost realnih brojeva – na računalu prikaz s pomičnim zarezom
  - Operacije s realnim brojevima su skupe
- ◆ Potreban je algoritam koji:
  - Koristi operacije sa cijelim brojevima
  - Koristi prikaz sa fiksnim brojem bitova
  - Proizvodi simbole koda tokom postupka kodiranja, a ne na kraju

- ◆ Osnovni postupak podjele na podintervale je isti
- ◆ Koristi se fiksni broj znamenki za prikaz intervala
- ◆ Kada je prva znamenka u prikazu gornje i donje granice ista, interval se *renormalizira*:
  - Prvih  $n$  znamenki se šalje na izlaz koderu
  - Znamenke se pomiču ulijevo za jedno mjesto
  - Desno se dodaje znamenka: 0 na donju, 1 na gornju granicu intervala (ako su znamenke binarne)

# Renormalizacija: primjer

x	p(x)
RAZMAK	1/10
A	1/10
B	1/10
E	1/10
G	1/10
I	1/10
L	2/10
S	1/10
T	1/10

	GORNJA GRANICA	DONJA GRANICA	DULJINA INTERVALA	KUMULATIVNI IZLAZ
Početno stanje	99999	00000	100000	
Kodiraj B (0.2-0.3)	29999	20000		
Renormalizacija, izlaz: 2	99999	00000	100000	.2
Kodiraj I (0.5-0.6)	59999	50000		.2
Renormalizacija, izlaz: 5	99999	00000	100000	.25
Kodiraj L (0.6-0.8)	79999	60000	20000	.25
Kodiraj L (0.6-0.8)	75999	72000		.25
Renormalizacija, izlaz: 7	59999	20000	40000	.257
Kodiraj RAZMAK (0.0-0.1)	23999	20000		.257
Renormalizacija, izlaz: 2	39999	00000	40000	.2572
Kodiraj G (0.4-0.5)	19999	16000		.2572
Renormalizacija, izlaz: 1	99999	60000	40000	.25721
Kodiraj A (0.1-0.2)	67999	64000		.25721
Renormalizacija, izlaz: 6	79999	40000	40000	.257216
Kodiraj T (0.9-1.0)	79999	76000		.257216
Renormalizacija, izlaz: 7	99999	60000	40000	.2572167
Kodiraj E (0.3-0.4)	75999	72000		.2572167
Renormalizacija, izlaz: 7	59999	20000	40000	.25721677
Kodiraj S (0.8-0.9)	55999	52000		.25721677
Renormalizacija, izlaz: 5	59999	20000		.257216775
Renormalizacija, izlaz: 2				.2572167752
Renormalizacija, izlaz: 0				.25721677520

# Usporedba aritmetičko - Huffman



Zavod za telekomunikacije

<b>Huffman</b>	<b>Aritmetičko kodiranje</b>
Kodira svaki simbol posebno	Kodira cijelu poruku jednim kodom: realni broj 0 - 1
Minimalno 1 bit/simbol	Moguće < 1 bit/simbol
Duljina poruke nije važna	Teoretski optimalno za dugačke poruke
Kodiranje niza simbola moguće samo proširenim Huffman kodom	Uvijek se kodira cijela poruka
Jednostavno za računanje	Zahtjevnije za računanje

- ◆ Primjena kao komponente u raznim standardima i za razne vrste medija
- ◆ Dokumenti
  - JBIG (Joint Bi-level Image Processing Group)
- ◆ Slika
  - JPEG
- ◆ Sintetički sadržaji/animacija
  - MPEG-4 FBA (Face and Body Animation)

# Metode rječnika

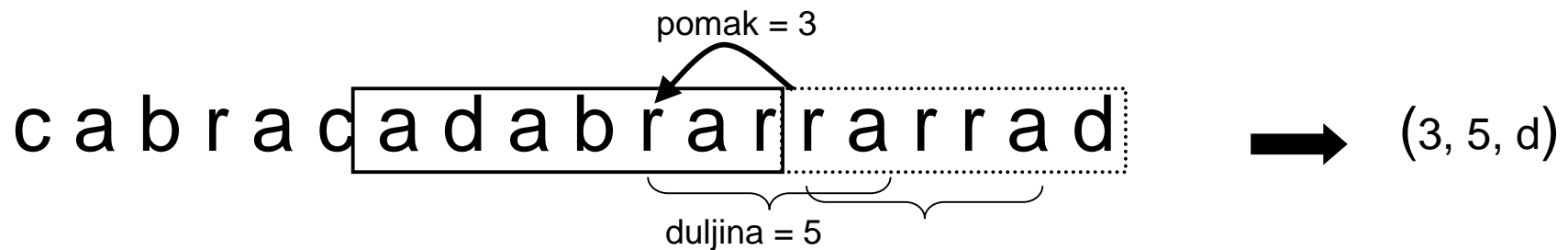
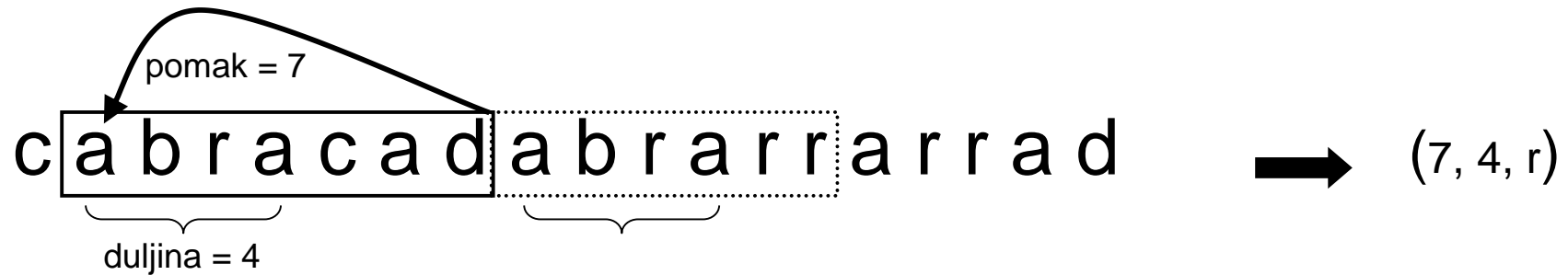
- ♦ Algoritmi kodiranja metodama rječnika uzimaju kao ulaz nizove simbola (“riječi”) promjenjive duljine i kodiraju ih kodnim riječima stalne duljine iz rječnika
- ♦ Ne trebaju znati vjerojatnosti pojavljivanja simbola, nazivaju se i *univerzalni koderi*
- ♦ Koder i dekoder moraju imati isti rječnik
- ♦ Rječnik može biti statičan, no najčešće je prilagodljiv

- ◆ Koder i dekoder dinamički grade rječnik
  - LZ77: Rječnik je posmični prozor
  - LZ78: riječi se grade dodavanjem slova na postojeće riječi (u početku rječnik je prazan)
  - Lempel-Ziv-Welch (LZW) algoritam
    - izvorni algoritam smislili Ziv i Lempel (1977 - LZ77, 1978 - LZ78), a Welch ga je doradio i poboljšao 1984 (zato **LZW**)
    - algoritam relativno jednostavan, iako složeniji od Huffmanovog
    - izvorni LZW algoritam koristi rječnik s 4K riječi, s tim da su prvih 256 riječi standardni ASCII kodovi

- ♦ Rječnik je posmični prozor od  $N$  zadnjih simbola
- ♦ U svakom koraku traži se u rječniku najduži niz simbola jednak nadolazećim simbolima, te se kodira kao uređena trojka (*pomak*, *duljina*, *sljedeći\_simbol*)
- ♦ Nedostatak: “kratka” memorija



# LZ77: primjer kodiranja



- ◆ Umjesto posmičnog prozora, zasebna memorija za rječnik
  - Rječnik je poredana lista riječi (nizova simbola)
  - Riječ se dovaća pomoću indeksa (rednog broja)
- ◆ LZ78
  - Rječnik u početku prazan
  - U svakom koraku šalje se (*indeks, idući simbol*)
    - Indeks pokazuje na najdulju riječ u rječniku jednaku nadolazećem nizu simbola
    - Rječnik se nadopunjava novim riječima tijekom kodiranja

- Algoritam kodiranja:

```
1. RadnaRiječ = slijedeći simbol sa ulaza
2. WHILE (ima još simbola na ulazu) DO
3.     NoviSimbol = slijedeći simbol sa ulaza
4.     IF RadnaRiječ+NoviSimbol postoji u rječniku THEN
5.         RadnaRiječ = RadnaRiječ+NoviSimbol
6.     ELSE
7.         IZLAZ: kod za RadnaRiječ
8.         dodaj RadnaRiječ+NoviSimbol u rječnik
9.         RadnaRiječ = NoviSimbol
10.    END IF
11. END WHILE
12. IZLAZ: kod za RadnaRiječ
```

# Kodiranje algoritmom LZW: primjer

Sadržaj rječnika na početku:

kodna riječ	znak
(1)	A
(2)	B
(3)	C

Niz znakova koje treba kodirati:

Mjesto	1	2	3	4	5	6	7	8	9
Simbol	A	B	B	A	B	A	B	A	C

LZW:

korak	mjesto	sadržaj rječnika	izlaz iz kodera
1.	1	(4) A B	(1)
2.	2	(5) B B	(2)
3.	3	(6) B A	(2)
4.	4	(7) A B A	(4)
5.	6	(8) A B A C	(7)
6.	9		(3)

# LZW kodiranje: primjer dekodiranja

KORAK	RADNA RIJEČ	ULAZ DEKODERA	DEKODIRANI SIMBOLI	SADRŽAJ RJEČNIKA
1		(1)	A	
2	A	(2)	B	(4) AB
3	B	(2)	B	(5) BB
4	B	(4)	AB	(6) BA
5	AB	(7)	ABA	(7) ABA
6	ABA	(3)	C	

- ◆ LZW
  - UNIX compress
  - GIF
  - Modem V.24 bis
- ◆ LZ77
  - ZIP

- 894**f32**
- ↑  
broj ponavljanja

- ← “isplati” se za 4+ znakova

- 81