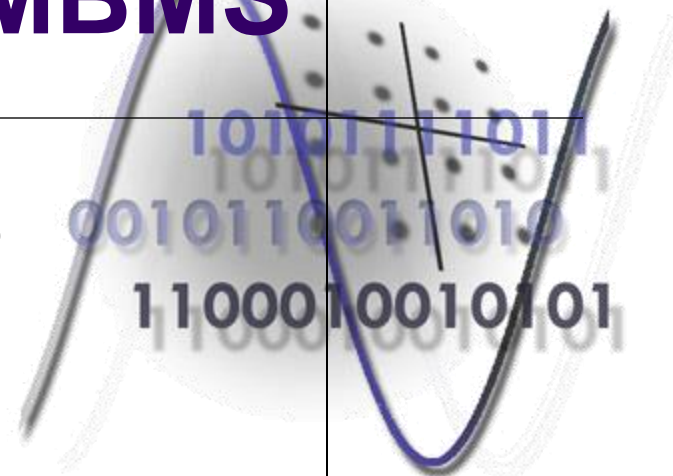
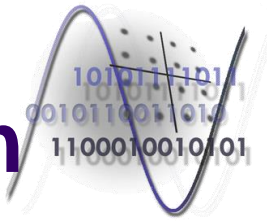


# Algoritmi practici de codare și decodare Raptor Utilizarea codurilor Raptor în MBMS

---

**TACCFDRT Curs 5**

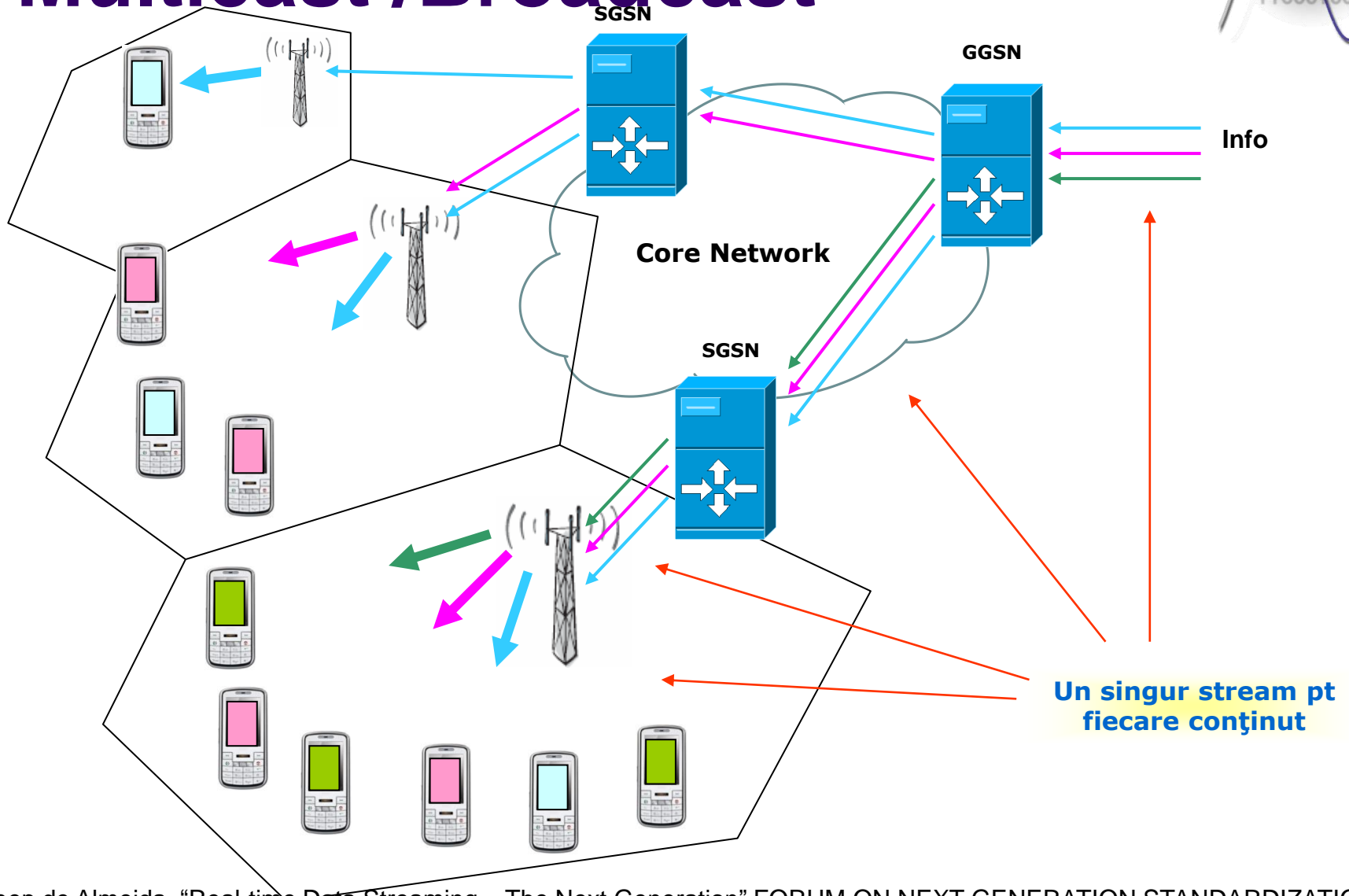
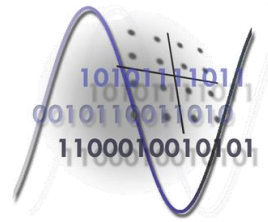




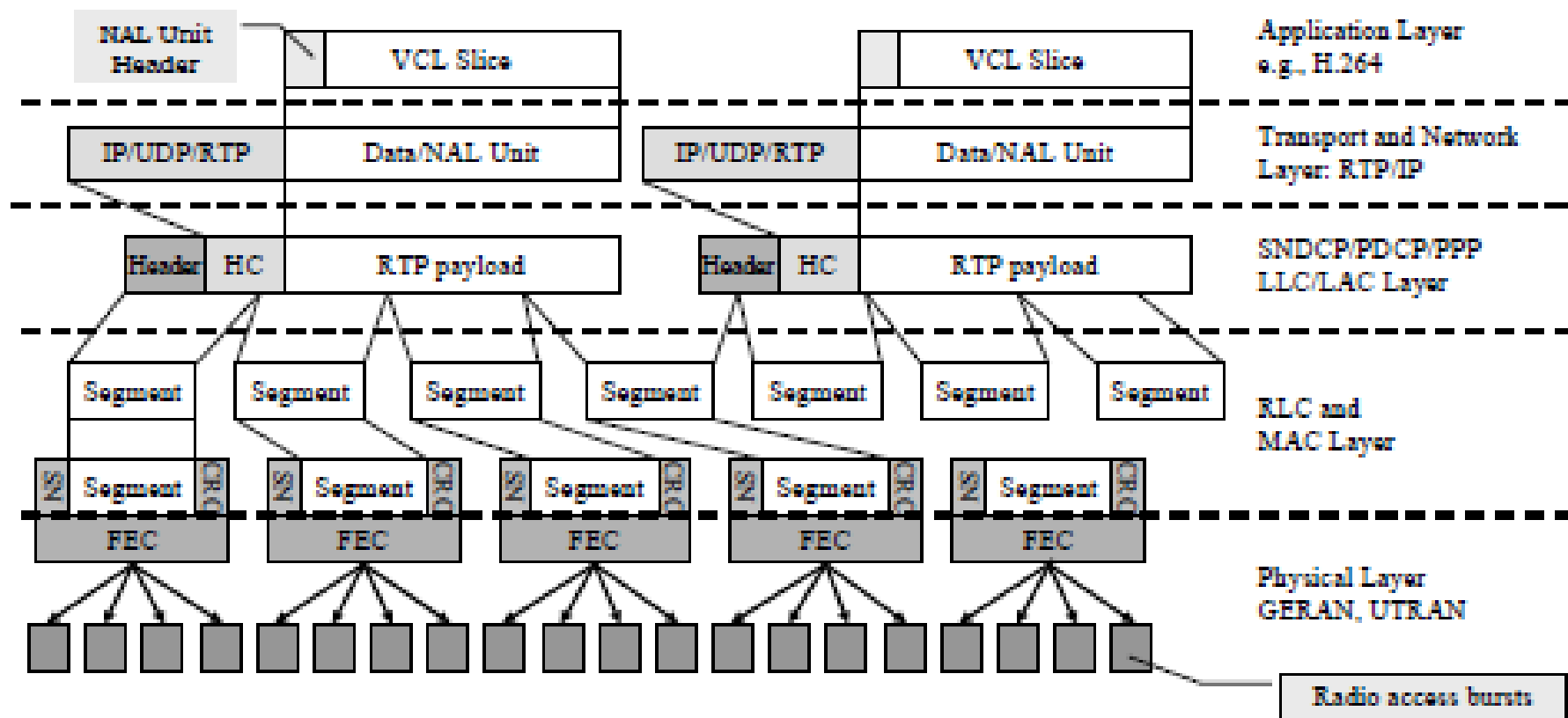
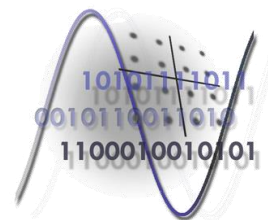
# Tehnici de codare de tip Digital Fountain

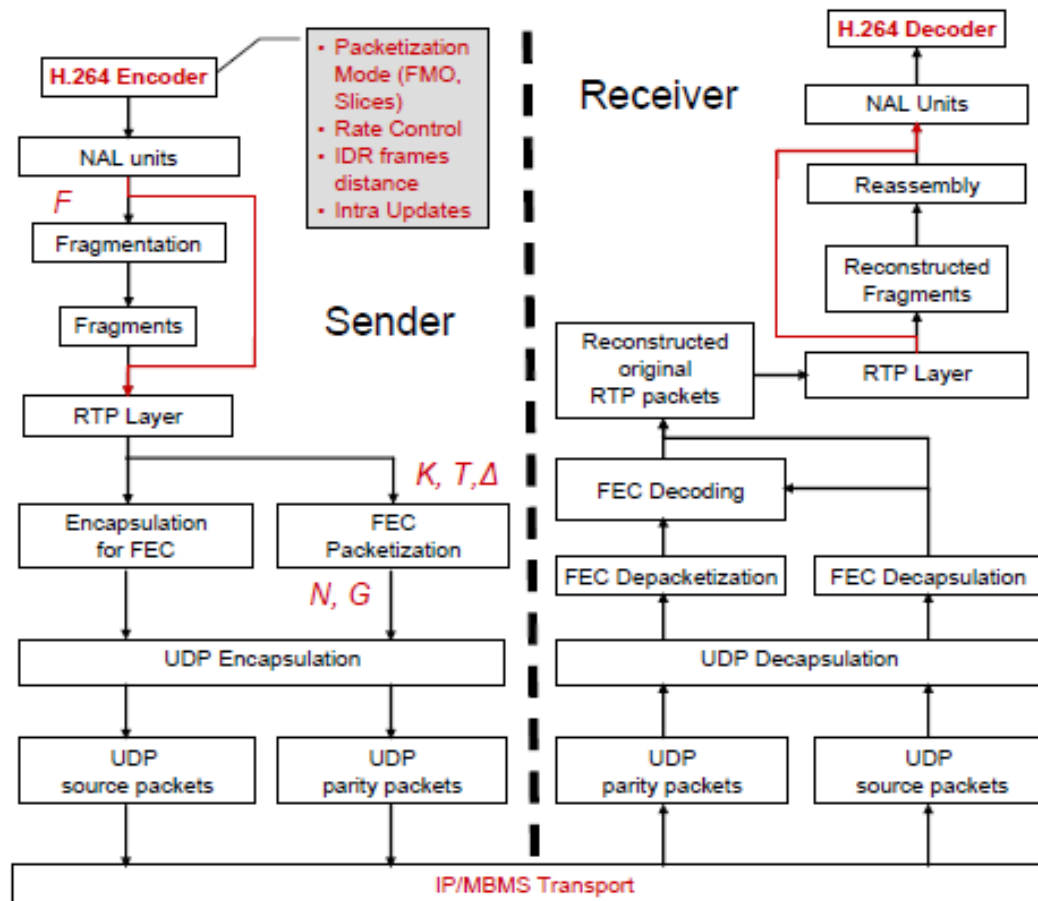
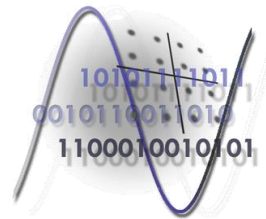
- Utilizarea codurilor raptor în MBMS
- DF în layered coding
- NC

# Multicast / Broadcast

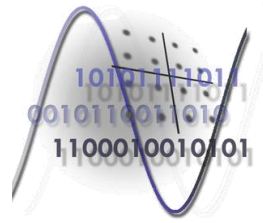


# Utilizarea codurilor raptor în MBMS





# Multimedia streaming

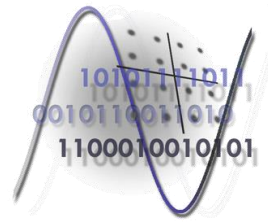


Retea eterogenă

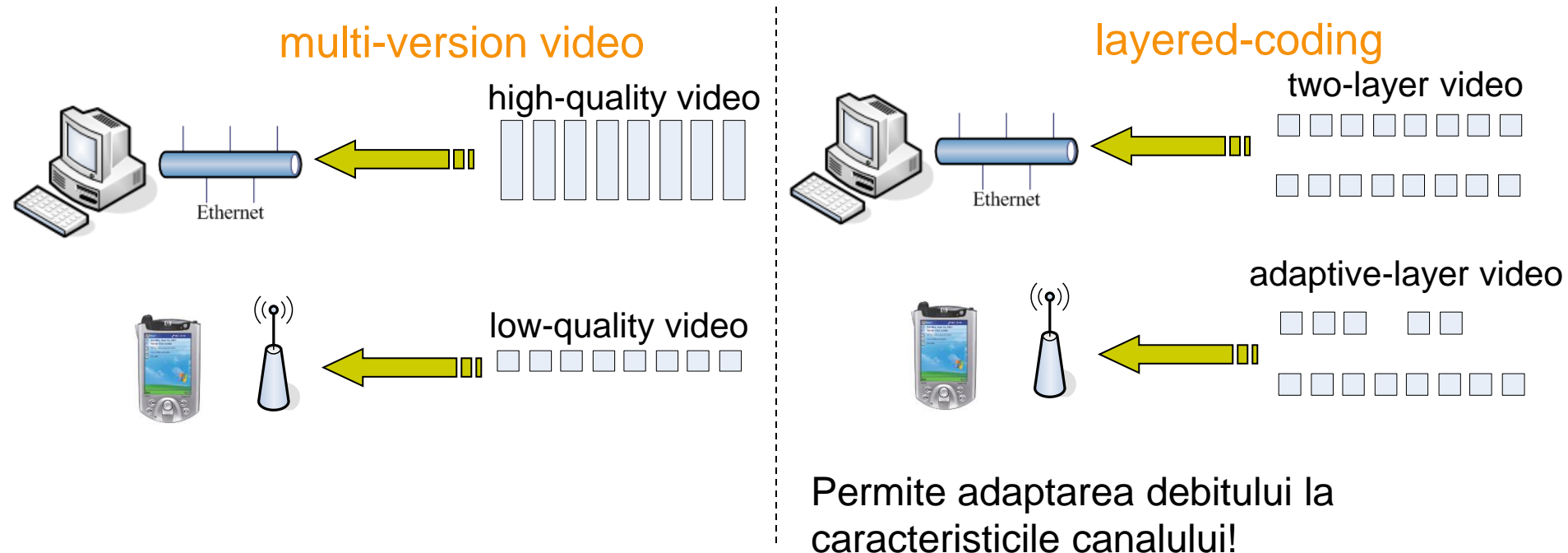


Cum se transmite un video stream?

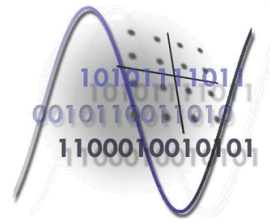
# Layered (scalable) coding



- multi-version v.s. layered coding



# layered coding

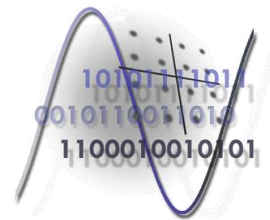


- Mecanismul de funcționare

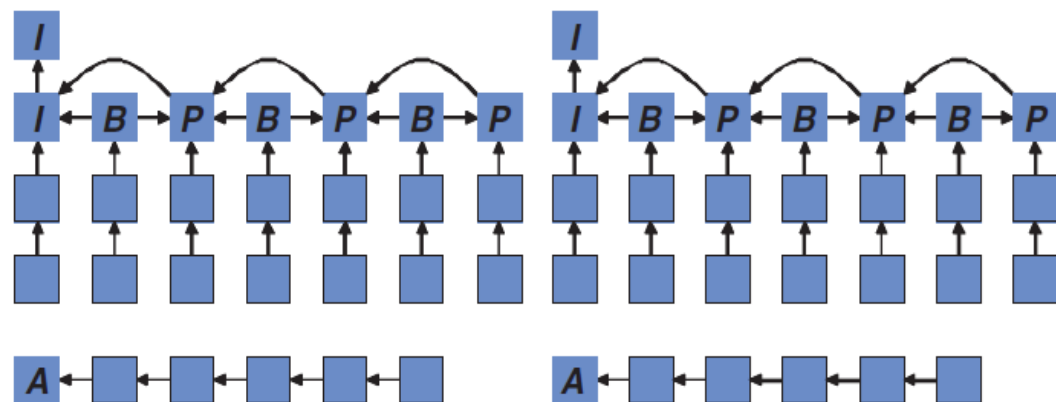




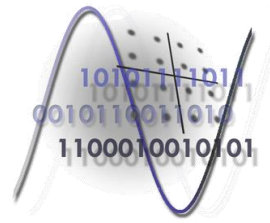
# layered coding



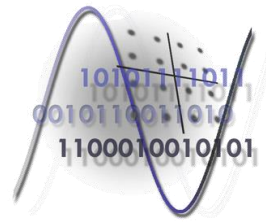
- Datele codate sunt împărțite în unități elementare
- Interdependențele între unitățile elementare pot fi reprezentate pe un graf direcționat aciclic
  - Fiecare nod al grafului reprezintă o unitate elementară
  - O legătură direcționată de la nodul I' către nodul I înseamnă că nodul I poate fi decodat numai după decodarea nodului I'



# layered coding



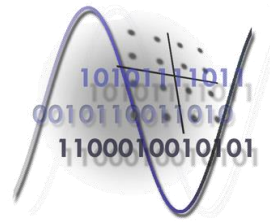
- La fiecare unitate elementară  $l$  este caracterizat de:
  - Dimensiune  $B_l$
  - Timpul de decodare  $t_D$ 
    - Reprezintă intervalul maxim de timp în care pachetul trebuie să fie decodat
  - Factor de importanță



# layered coding

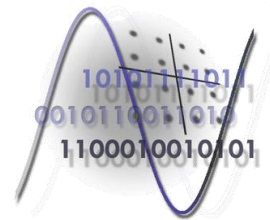
- Se utilizează protecție neuniformă
  - Nivelul de bază trebuie codat cu un cod puternic
  - Nivelele adiționale pot fi codate cu coduri mai slabe sau să fie transmise necodate
- Protocol de transport trebuie să asigure ca pachetele din nivelul de bază să ajungă înaintea pachetelor din nivele adiționale

# layered coding

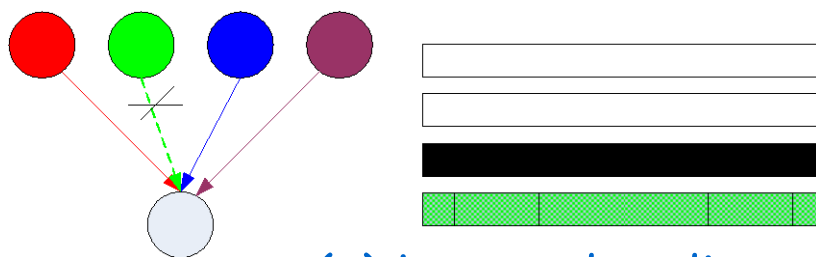


- Dezavantajul
  - dacă pachetul de pe nivelul de bază nu a sosit în timp util pachetele recepționate aparținând nivelelor adiționale devin redundante

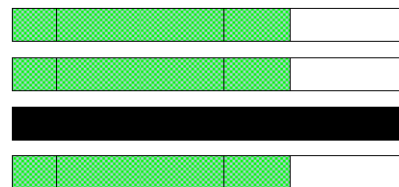
# Multiple description coding



- A fost propus ca o alternativă pentru LC in video streaming
- Fiecare "descriere" conține informația necesară pentru o redare cu calitatea de bază, și fiecare "descriere" în plus îmbunătățește calitatea redării

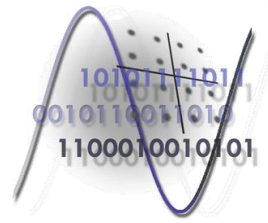


(a) Layered coding

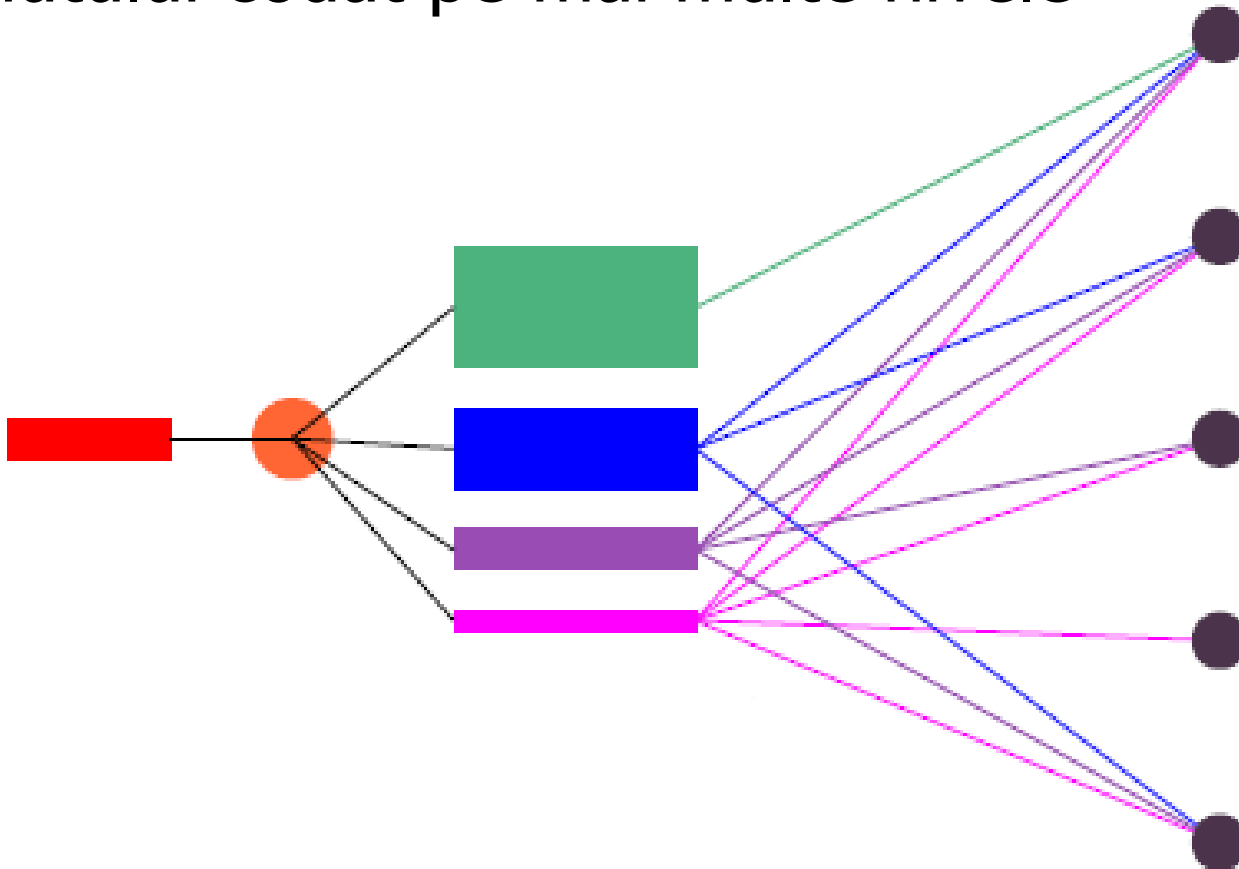


(b) MDC

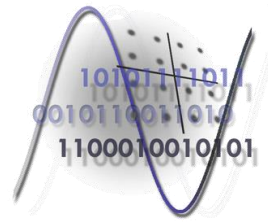
# layered coding & MDC



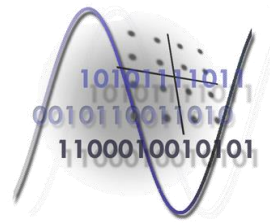
- Prin utilizarea codurilor DF se simplifica multicastul conținutului codat pe mai multe nivele



# layered coding & MDC

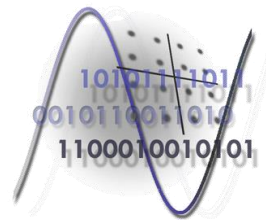


- pentru fiecare nivel se inițializează o legătură între serverul multimedia și aplicația de redare a fluxului multimedia
- Legătura pe care se transmite nivelul de bază are prioritatea cea mai mare
- Nivelele adiționale se transmit pe legături cu prioritate mai redusă



# Network coding

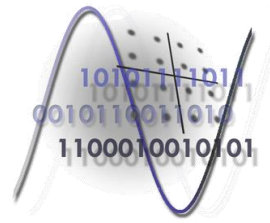




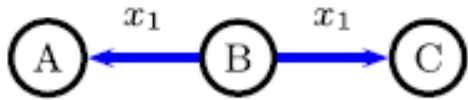
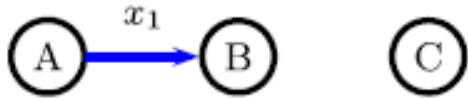
# Introducere in NC

- În rețelele de calculatoare fiecare flux de date se propagă independent chiar dacă împarte același resurse cu alte fluxuri de date
- NC este o tehnică de combinare a fluxurilor de date care utilizează aceleași resurse

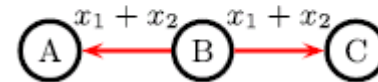
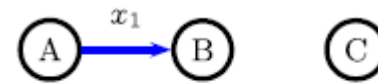
# Wireless network

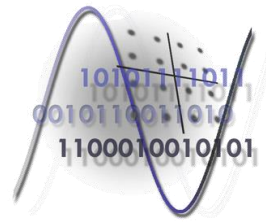


- Fără NC



## Cu NC

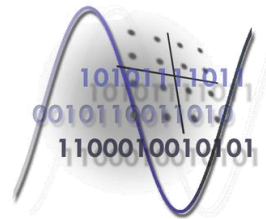




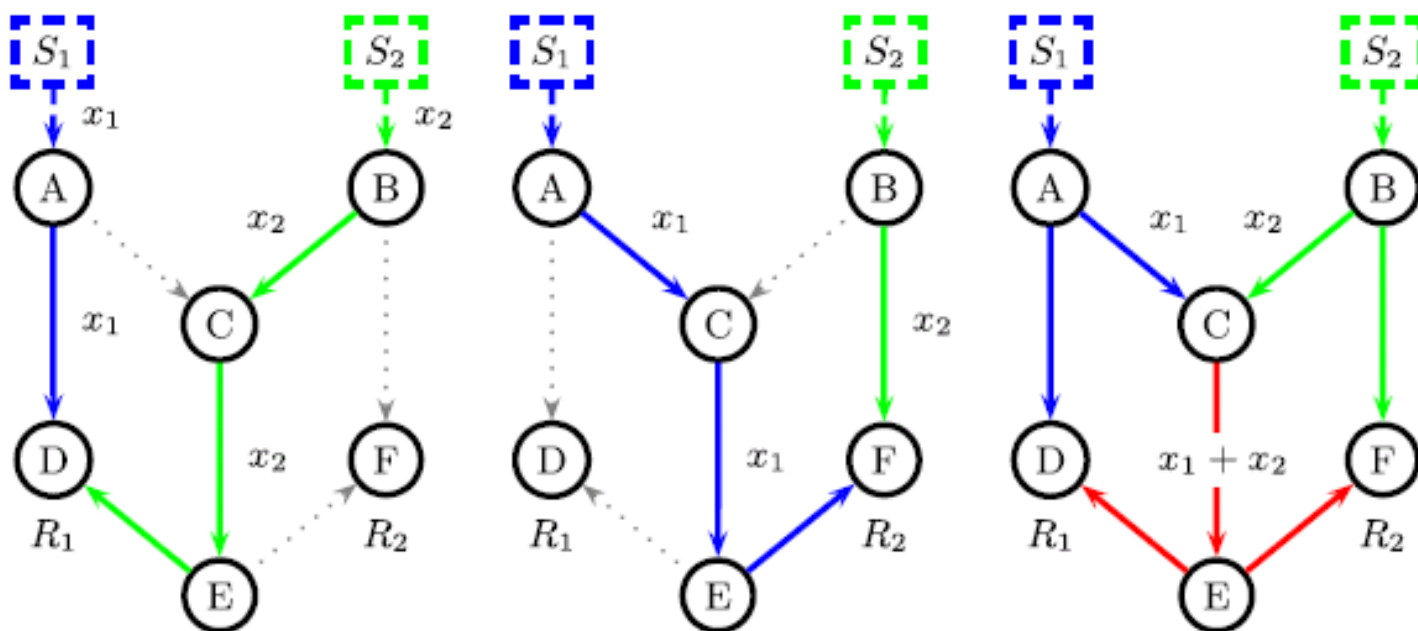
# Modelare Matematică

- O rețea de comunicații poate fi reprezentată printr-un graf orientat  $G=(V,E)$ 
  - $V$  reprezintă mulțimea nodurilor,
  - $E$  reprezintă ramurile grafului, fiecare ramură din graf reprezintă un canal de comunicație având capacitate de o unitate de date pe o unitate de timp
  - pot exista mai multe ramuri între două noduri
- Un nod care nu are ramuri de intrare se numește sursă
- Un nod care nu are legături de ieșire se numește destinație

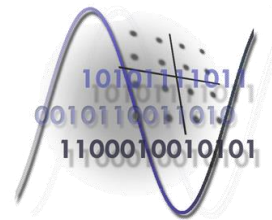
# Example



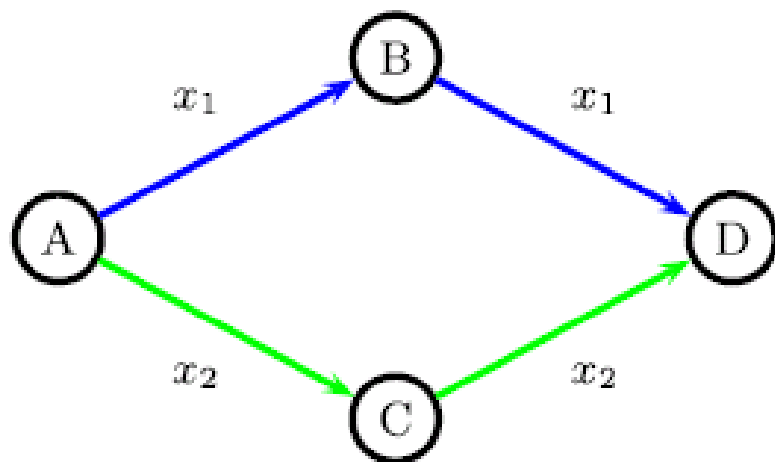
- Butterfly network



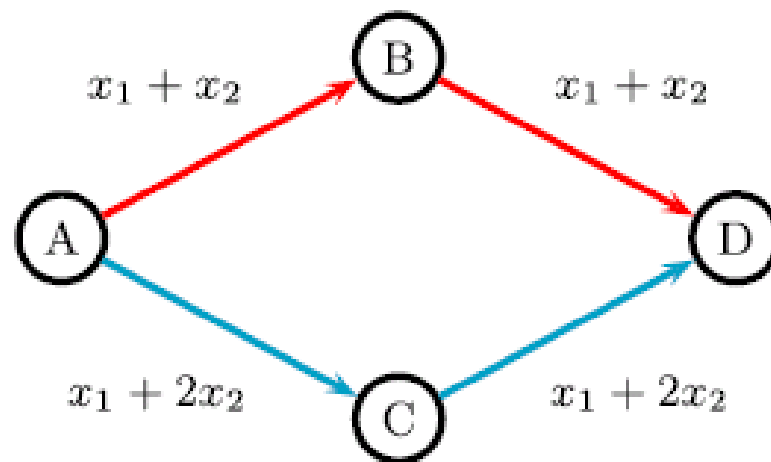
# Criptare cu NC

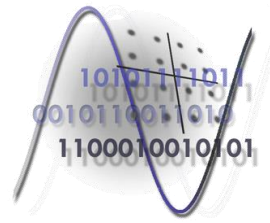


- Fără NC



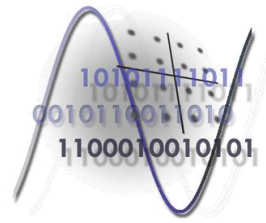
## Cu NC



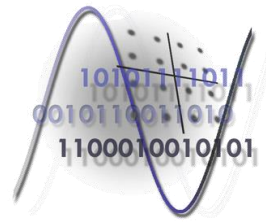


# Cuprins

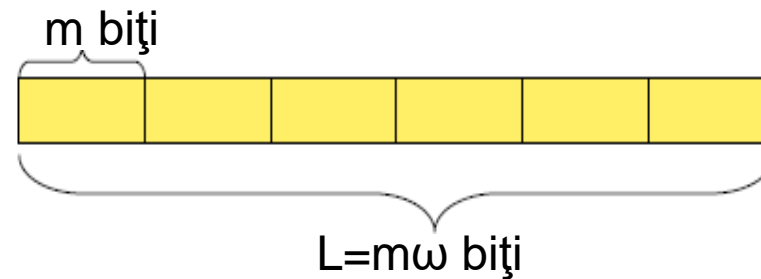
- Introducere în NC
- Descriere matematică
- Random NC
- NC în rețele cu erori
- Optimizarea rețelei



- Pentru fiecare nod  $T$  din rețea  $In(T)$  este setul de canale de intrare în  $T$ , iar  $Out(T)$  este setul de canale de ieșire din  $T$
- $In(S)$  este setul de canale imaginare, care se termină la nodul sursă  $S$ , dar nu au nod de origine. Numărul acestor canale imaginare se notează cu  $\omega$
- Ramurile grafului sunt notate cu  $(v_1, v_2, i) \in E$
- Începutul și sfârșitul unei muchii  $e = (v_1, v_2, i)$  vor fi notate cu  $v_1 = head(e)$  și  $v_2 = tail(e)$ .



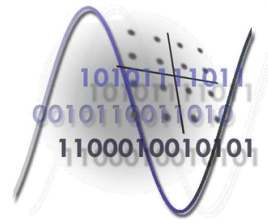
- O unitate de date (simbol) este un element dintr-un câmp finit  $F$
- Un mesaj constă din  $\omega$  unități de date și va fi reprezentat printr-un vector  $x \in F^\omega$



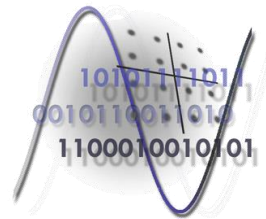
- Adică fiecare pachet poate fi privit ca un vector de  $\omega$  simboluri din câmpul  $GF(2^m)$



# NC lineare



- Fiecare simbol din rețea aparține câmpului  $GF(2^m)$
- Nodurile pot să proceseze simbolurile recepționate, efectuând combinații liniare ale acestora
  - Simbolul obținut după o combinație liniară aparține tot câmpului  $GF(2^m)$
  - Lungimea pachetului după o combinație liniară nu se modifică
- Notăm cu
  - $X(v, l)$ ,  $l=1, \dots, \omega(v)$ - simbolurile de intrare a unui nod sursă
  - $Y(e)$ , simbolul transmis pe o ramură
  - $Z(v, l)$  simbolurile de ieșire a unui nod destinație



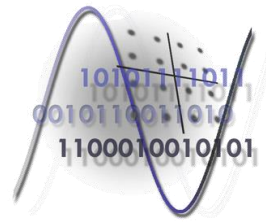
- **Definiție** Fie  $G=(V,E)$  o rețea fără întârzieri.  $G$  va fi liniară peste câmpul  $GF(2^m)$ , dacă pentru toate legăturile, “procesele”  $Y(e)$  pe o legătură  $e=(v,u,i)$  satisfac

$$Y(e) = \sum_{j=1}^{\omega(v)} \alpha_{e,j} X(v,j) + \sum_{e': \text{head}(e')=\text{tail}(e)} \beta_{e',e} Y(e')$$

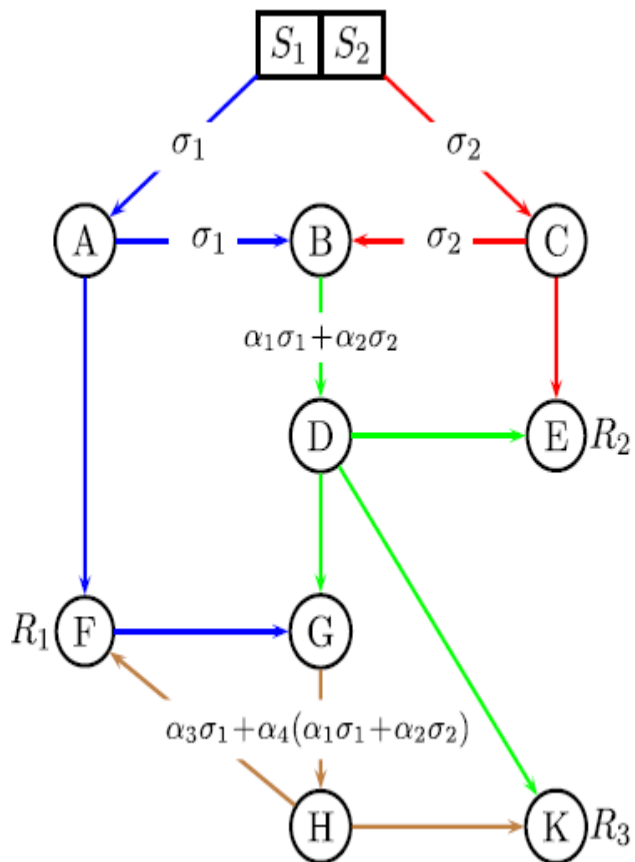
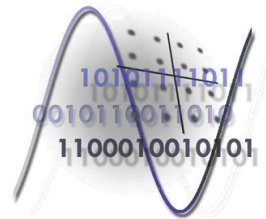
Unde coeficienții  $\alpha_{e,j}$  și  $\beta_{e',e}$  sunt elemente din  $GF(2^m)$

Ieșirile  $Z(v,j)$  a unui nod  $v$  vor fi

$$Z(v,j) = \sum_{e': \text{head}(e')=v} \varepsilon_{e',j} Y(e')$$

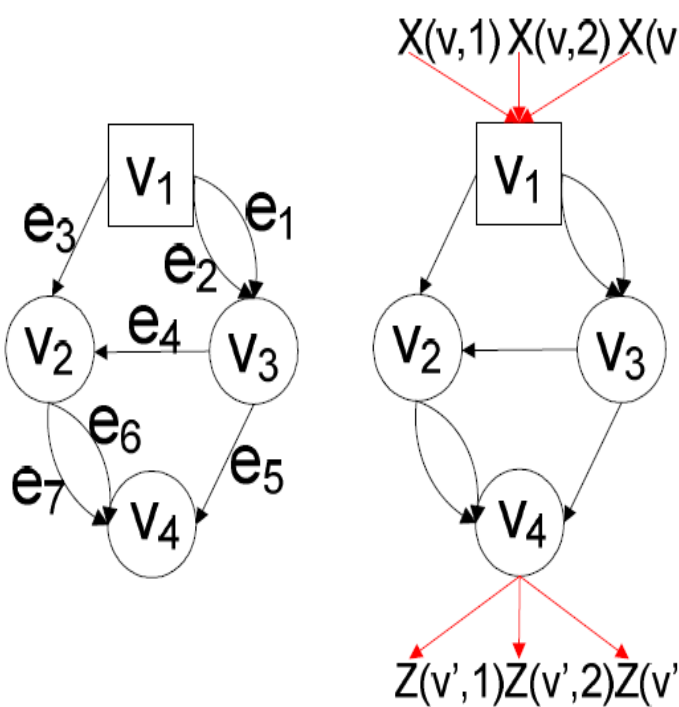


- Vectorul de codare locală pentru o ramură  $e$  este definită ca un vector de dimensiune  $1 \times |\text{In}(v)|$  care conține coeficienții din câmpul  $\text{GF}(2^m)$  cu care sunt înmulțiți simbolurile de pe fiecare intrare a nodului  $v$  de unde pornește ramura  $e$
- Vectorul de codare globală pentru ramura  $e$  este un vector de lungime  $\omega$ , elementele acestui vector arată cum sunt mapate în mesajul transmis pe ramura  $e$  cele  $\omega$  simboluri de intrare a rețelei



Vectorul de codare locală  $f(\text{BD})$   
pentru ramura BD este  $[\alpha_1 \ \alpha_2]$

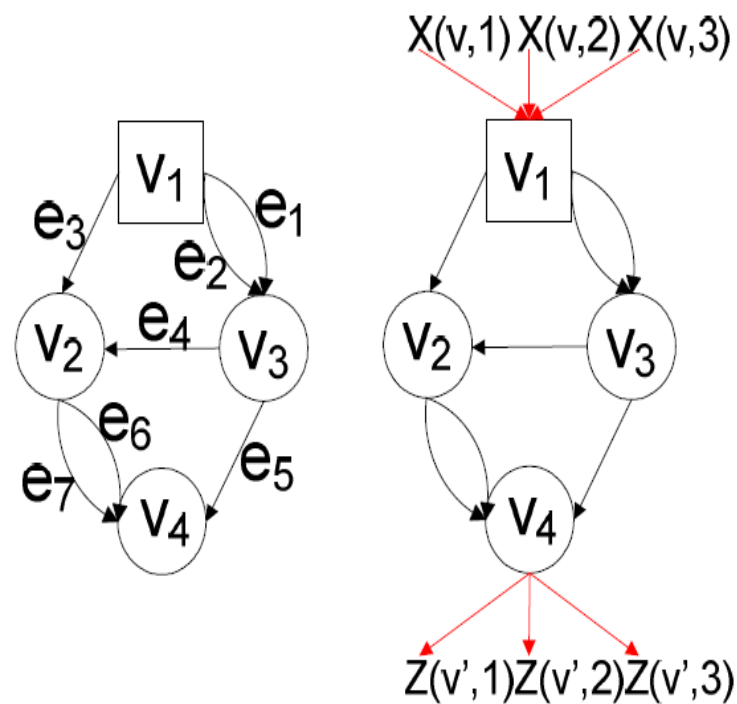
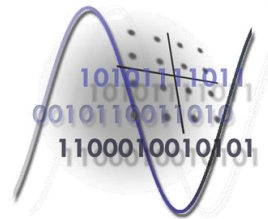
Vectorul de codare globală  
pentru ramura GH este  
 $[\alpha_3 + \alpha_1 \ \alpha_4, \alpha_2 \ \alpha_4]$



$$\begin{cases} Y(e_1) = \alpha_{e_1,1}X(v,1) + \alpha_{e_1,2}X(v,2) + \alpha_{e_1,3}X(v,3) \\ Y(e_2) = \alpha_{e_2,1}X(v,1) + \alpha_{e_2,2}X(v,2) + \alpha_{e_2,3}X(v,3) \\ Y(e_3) = \alpha_{e_3,1}X(v,1) + \alpha_{e_3,2}X(v,2) + \alpha_{e_3,3}X(v,3) \end{cases}$$

$$\begin{cases} Y(e_4) = \beta_{e_1,e_4}Y(e_1) + \beta_{e_2,e_4}Y(e_2) \\ Y(e_5) = \beta_{e_1,e_5}Y(e_1) + \beta_{e_2,e_5}Y(e_2) \\ Y(e_6) = \beta_{e_3,e_6}Y(e_3) + \beta_{e_4,e_6}Y(e_4) \\ Y(e_7) = \beta_{e_3,e_7}Y(e_3) + \beta_{e_4,e_7}Y(e_4) \end{cases}$$

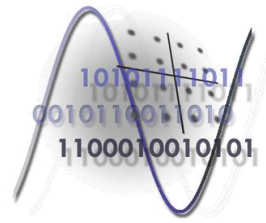
$$\begin{cases} Z(v',1) = \varepsilon_{e_5,1}Y(e_5) + \varepsilon_{e_6,1}Y(e_6) + \varepsilon_{e_7,1}Y(e_7) \\ Z(v',2) = \varepsilon_{e_5,2}Y(e_5) + \varepsilon_{e_6,2}Y(e_6) + \varepsilon_{e_7,2}Y(e_7) \\ Z(v',3) = \varepsilon_{e_5,3}Y(e_5) + \varepsilon_{e_6,3}Y(e_6) + \varepsilon_{e_7,3}Y(e_7) \end{cases}$$



$$A = \begin{pmatrix} \alpha_{e_1,1} & \alpha_{e_2,1} & \alpha_{e_3,1} \\ \alpha_{e_1,2} & \alpha_{e_2,2} & \alpha_{e_3,2} \\ \alpha_{e_1,3} & \alpha_{e_2,3} & \alpha_{e_3,3} \end{pmatrix} \quad B = \begin{pmatrix} \varepsilon_{e_5,1} & \varepsilon_{e_5,2} & \varepsilon_{e_5,3} \\ \varepsilon_{e_6,1} & \varepsilon_{e_6,2} & \varepsilon_{e_6,3} \\ \varepsilon_{e_7,1} & \varepsilon_{e_7,2} & \varepsilon_{e_7,3} \end{pmatrix}$$

Matricea de transfer  $M$  a sistemului este egală cu:

$$M = A \begin{pmatrix} \beta_{e_1,e_5} & \beta_{e_1,e_4}\beta_{e_4,e_6} & \beta_{e_1,e_4}\beta_{e_4,e_7} \\ \beta_{e_2,e_5} & \beta_{e_2,e_4}\beta_{e_4,e_6} & \beta_{e_2,e_4}\beta_{e_4,e_7} \\ 0 & \beta_{e_3,e_6} & \beta_{e_3,e_7} \end{pmatrix} B^T$$



- Se definește matricea de adiacență  $F$  cu elementele  $F_{i,j}$  date prin

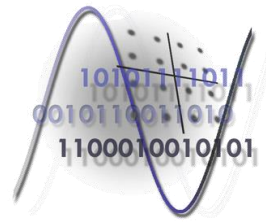
$$F_{i,j} = \begin{cases} \beta_{e_i, e_j} & \text{head}(e_i) = \text{tail}(e_j) \\ 0 & \text{in rest} \end{cases}$$

- Elementele matricii  $A$  de dimensiune  $\omega \times |E|$  definește ca

$$A_{i,j} = \begin{cases} \alpha_{e_j, 1} & x_i = X(\text{tail}(e_j), 1) \\ 0 & \text{in rest} \end{cases}$$

- Elementele matricii  $B$  de dimensiune  $\omega \times |E|$  definește ca

$$B_{i,j} = \begin{cases} \varepsilon_{e_j, 1} & z_i = Z(\text{head}(e_j), 1) \\ 0 & \text{in rest} \end{cases}$$



- Matricea de transfer a rețelei este dată de

$$M=A(I-F)^{-1}B^T$$

Coeficienții se aleg astfel încât toate matricele de transfer între sursă și oricare nod terminal să pot fi inversate

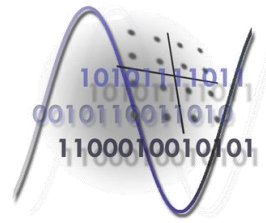
Putem scrie ca

$$z=Mx^T$$

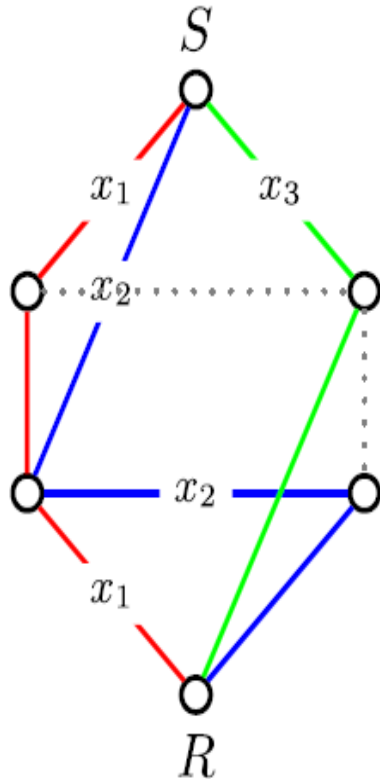
Adică NC transformă "rețeaua" într-un "canal"



# Teorema min cut max flow

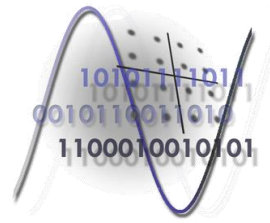


- *Min cut* numărul de ramuri minime care trebuie eliminate dintr-un graf pentru ca un nod destinație să nu fie legată la un nod sursă



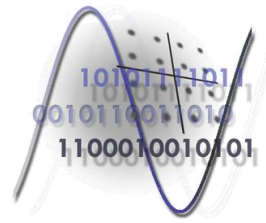
- Rata maximă cu care se poate transmite date de la sursa  $S$  la destinația  $R$  este egală cu *min cut*, adică se pot găsi exact *min cut* căi direcționate între  $S$  și  $R$

# Teorema min cut max flow

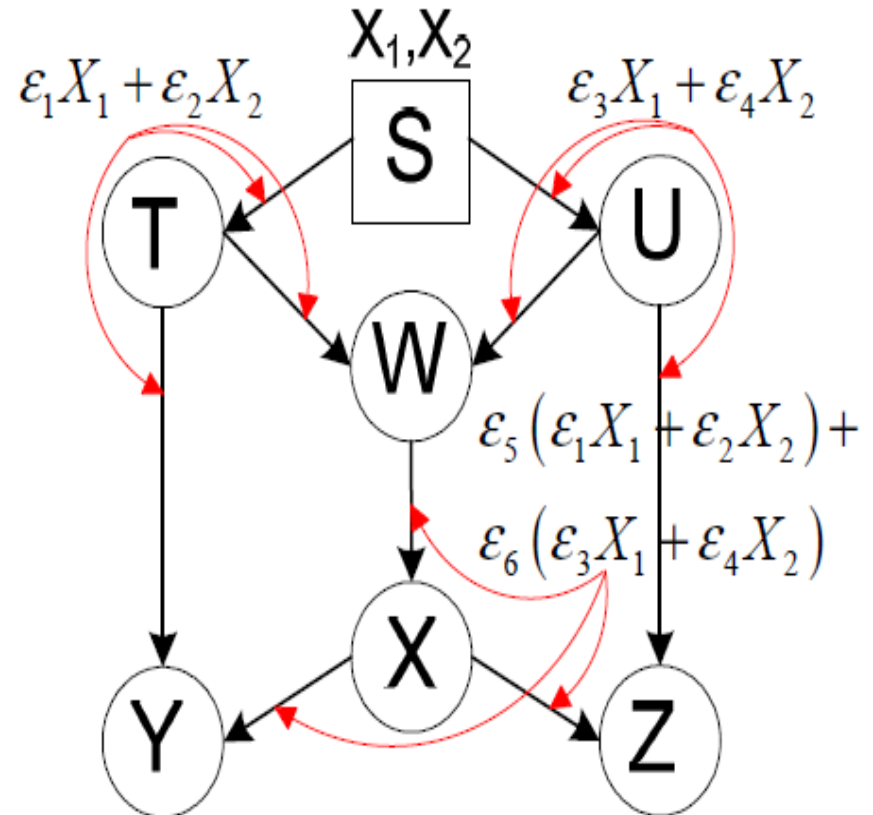


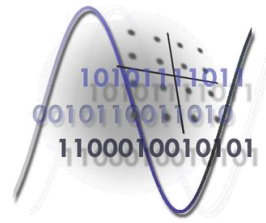
Prin utilizarea NC într-o rețea, o sursă poate să transmită simultan la mai multe destinații cu debitul maxim egal cu *min cut*-ul minim dintre destinații și surse.

# Random NC



- În acest caz fiecare nod va efectua o mapare liniară aleatoare a simbolurilor de intrare în simbolurile de ieșire
- Nodurile terminale trebuie să știe numai vectorul de codare globală a ramurilor de intrare

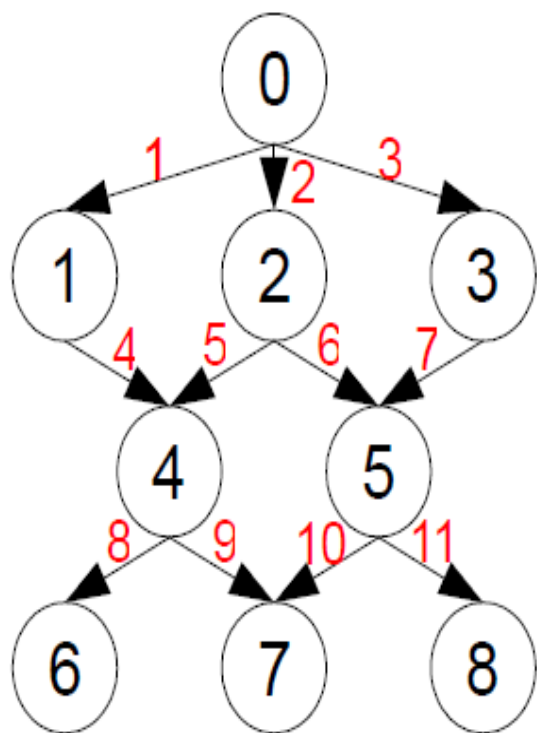
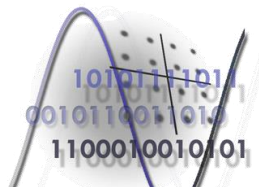




# RNC cu întârzieri

- Dacă rețeaua este aciclică operațiunea de decodare poate fi sincronizată, astfel întârzierile nu au efect asupra decodării
- Dacă rețeaua este ciclică, decodare nu tolerează întârzierile

# NC în rețele cu erori

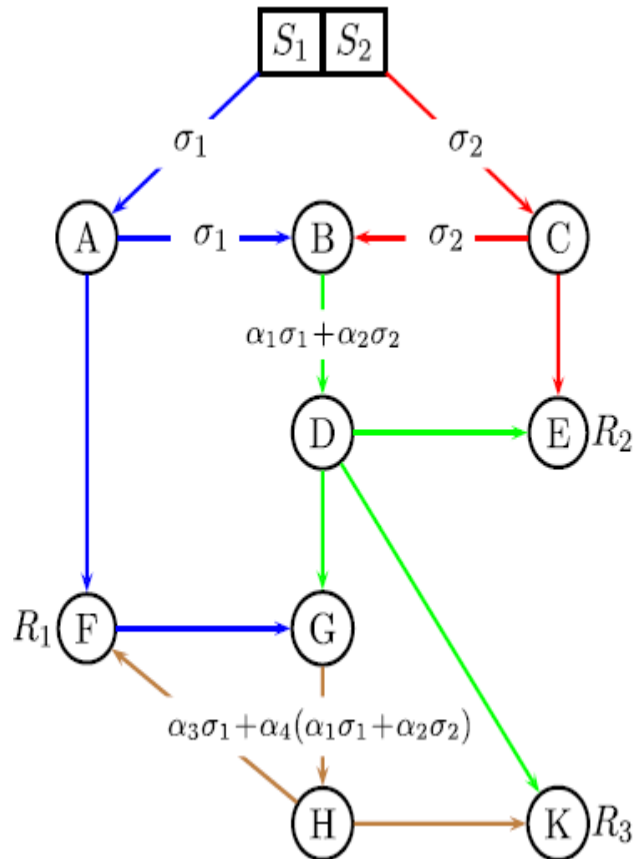
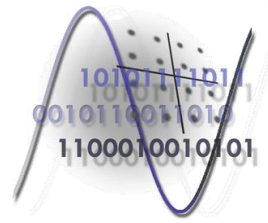


$$A = \begin{pmatrix} a_{0,0} & a_{0,1} & a_{0,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_{1,0} & a_{1,1} & a_{1,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$F = \begin{pmatrix} 0 & 0 & 0 & f_{0,3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & f_{1,4} & f_{1,5} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & f_{2,6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & f_{3,7} & f_{3,8} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & f_{4,9} & f_{4,10} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$B_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_{0,0,7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_{0,1,7} & 0 & 0 & 0 \end{pmatrix}$$

# Optimizarea arhitecturii rețelei



- Un algoritm genetic centralizat are următorii pași

```
[C1] initializare populatie;  
[C2] evaluare populatie;  
[C3] WHILE se ajunge la criteriul de terminare  
{  
[C4] selectare solutii pentru populatia urmatoare;  
[C5] crossover;  
[C6] mutatie;  
[C7] evaluarea populatiei;  
}
```