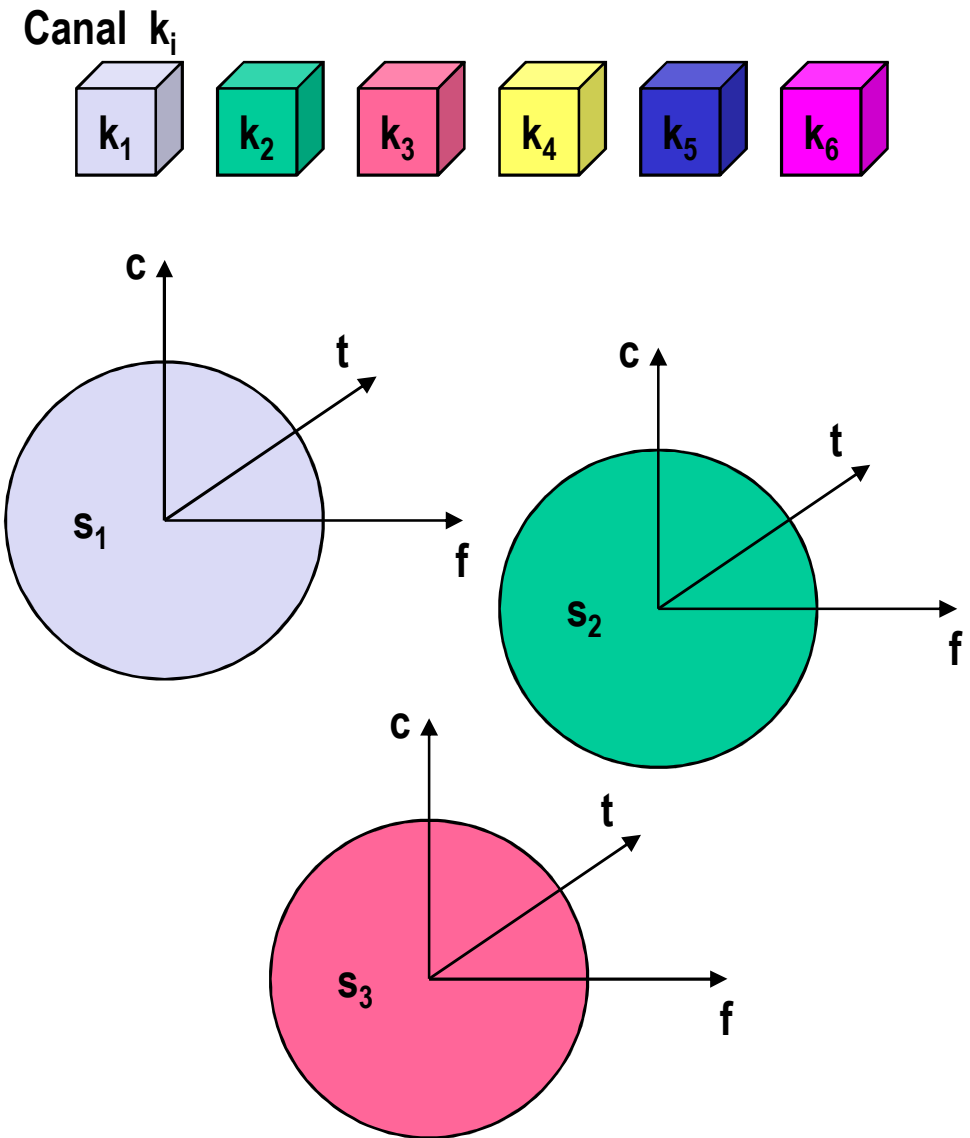


# Accesul la mediu

- **Fixe**
  - SDMA, FDMA, TDMA, CDMA
- **Aleator**
  - Aloha
  - Collision detection
  - CSMA
  - Terminale ascunse, expuse, near-far
- **De citit:**
  - Schiller 2.5
  - Tanenbaum 2.6.2
  - Schiller 3 – 3.4.4

- **Multiplexare în 4 dimensiuni**
  - spațiu ( $s_i$ )
  - timp ( $t$ )
  - frecvență ( $f$ )
  - cod ( $c$ )
- **Scop: utilizarea partajată a spectrului**
- **Canal: o resursă logică**
- **Important: sunt necesare spații de gardă!**

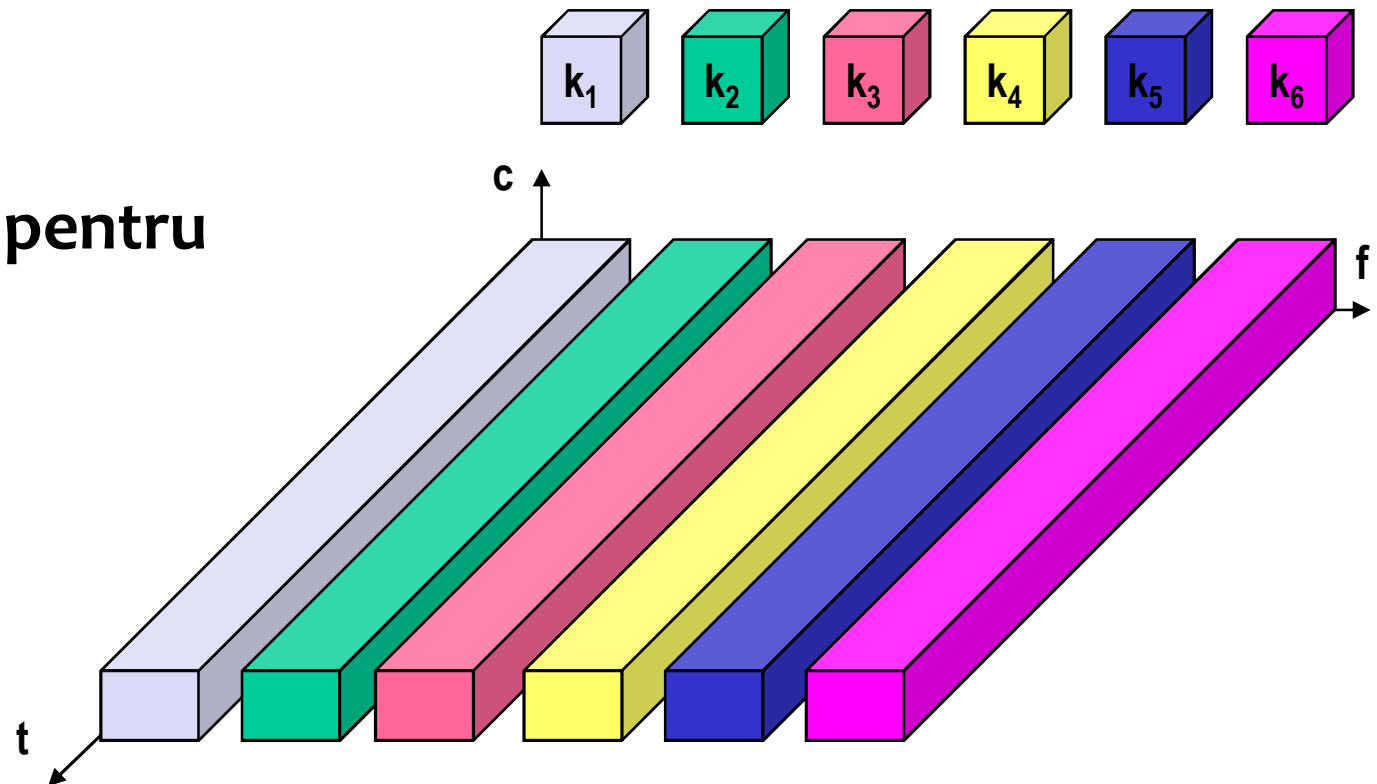


# Multiplexare în frecvență



- Separarea spectrului în benzi de frecvență mai mici
- Un canal obține o frecvență tot timpul
- Avantaje
  - Fără coordonare dinamică
  - Funcționează și pentru semnale analogice

- Dezavantaje
  - Risipă de bandă pentru traficul inegal
  - inflexibil



# Multiplexare în timp



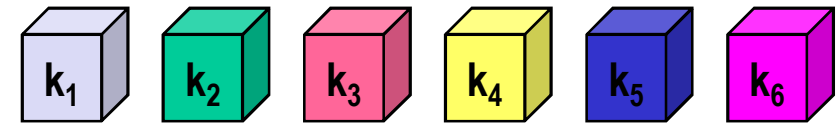
- Un canal primește mediul pentru o perioadă de timp

- **Avantaje**

- un singur vorbitor pe mediu

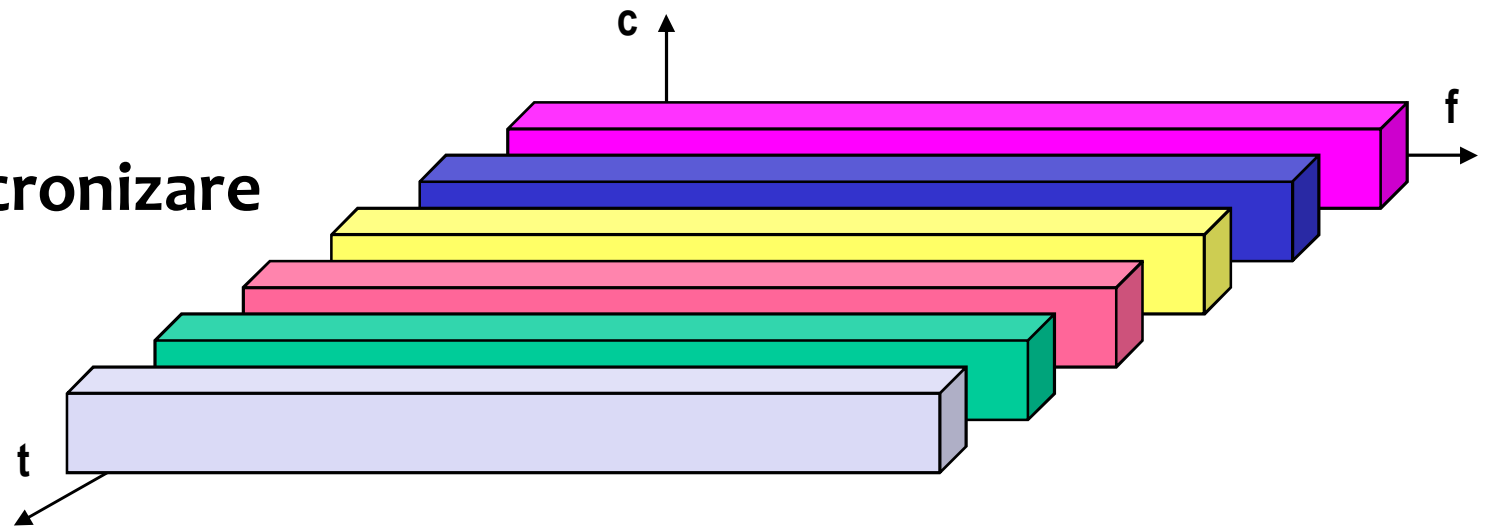
- la un moment dat

- utilizare mare la încărcare mare



- **Dezavantaje**

- Necesită sincronizare precisă



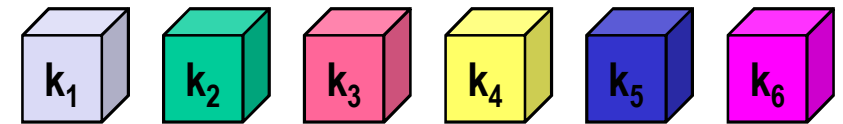
# Multiplexare în timp și frecvență



- Combinație a metodelor precedente
- Un canal primește o frecvență pentru o perioadă de timp
- Exemplu: GSM

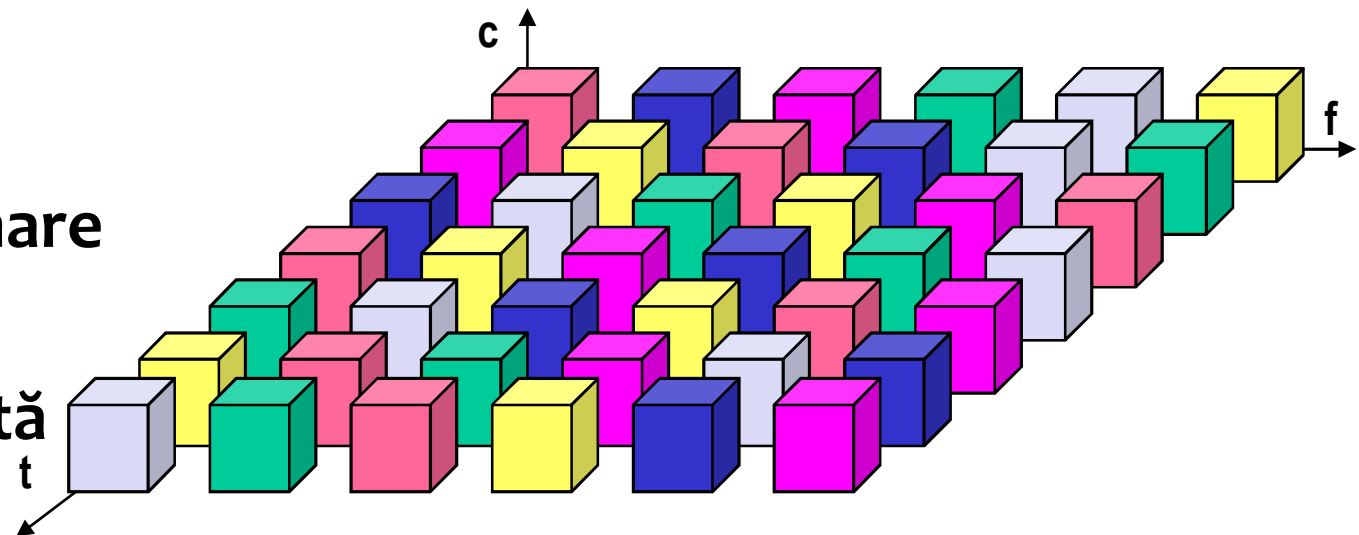
- **Avantaje**

- Protecție împotriva interferenței selective



- **Dezavantaje**

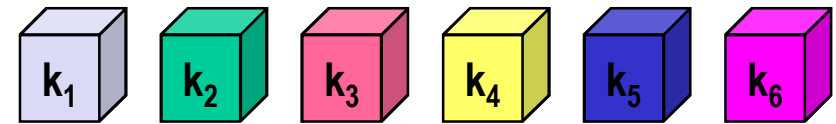
- necesită coordonare precisă
- viteza luminii finită



# Multiplexare în cod



- Fiecare canal primește un cod unic



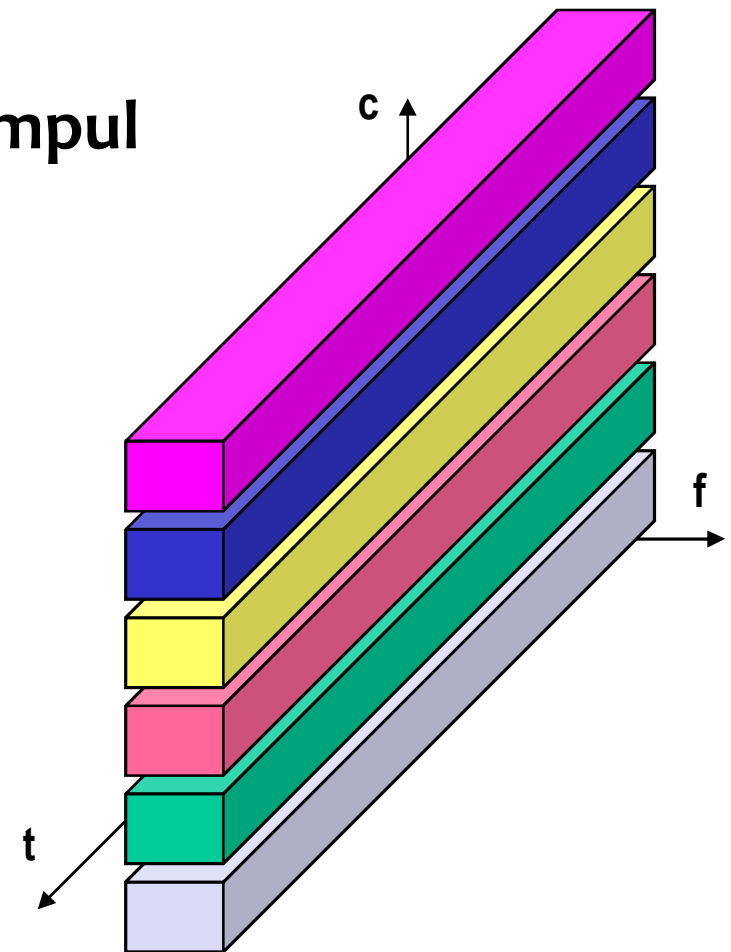
- Toate canalele folosesc tot spectrul tot timpul

- **Avantaje**

- Eficient în utilizarea capacității
- Nu necesita coordonare
- Protecție bună împotriva interferenței

- **Dezavantaje**

- Performanță variabilă la utilizator
- Procesare de semnal mai complexă
- Necesită control *fin* al puterii
- ... to be continued



- **SDMA (Space Division Multiple Access)**
  - Împarte spațiul în segmente, celule
  - Poate folosi antene direcționale
- **FDMA (Frequency Division Multiple Access)**
  - asignează o frecvență unui canal sender <>receiver
  - permanent (e.g., radio broadcast), slow hopping (e.g., GSM), fast hopping (FHSS, Frequency Hopping Spread Spectrum)
- **TDMA (Time Division Multiple Access)**
  - Asignează pe o frecvență dată un canal de transmisie pentru un timp  $t$
- **CDMA (Code Division Multiple Access)**
  - Un cod per utilizator

- Uplink: canalul de comunicare **de la** mobil **către** 3G basestation, WiFi access point
- Downlink: canalul de comunicare **de la** 3G basestation, WiFi access point **către** mobil
- Upstream: trafic de la mobil către Internet
- Downstream: trafic dinspre Internet către mobil
- TX: transmisie
- RX: recepție



# Definiții: Multiplexing, Duplex



- **Multiplexing** = partajare resursă între mai mulți utilizatori
  - Modul de acces: FDMA bazat pe FDM, etc
- **Comunicația între doi parteneri**
  - Simplex = o direcție
  - Full duplex = ambele direcții, simultan
  - Half duplex = ambele direcții, alternativ
- **Duplexing** = multiplexarea unui uplink sau downlink
  - full or half (multe carduri execută fie TX, fie RX)
  - TDD = time division duplexing
    - Problemă: Timp comutare între TX și RX
  - FDD = frequency division duplexing
    - Problemă: Timp de a comuta frecvența

# CDMA - simplificat



- **Sender A**
  - Secvența binară A = 00011011 (8 chip-uri)
  - Secvența bipolară (assign: “0”= -1, “1”= +1)
    - Pentru a trimite bitul “1”  $A = (-1, -1, -1, +1, +1, -1, +1, +1)$
    - Pentru a trimite bitul “0”  $-A = (+1, +1, +1, -1, -1, +1, -1, -1)$
- **Sender B**
  - Secvența binară B = 00101110
  - Secvența bipolară (assign: “0”= -1, “1”= +1)
    - Pentru a trimite bitul “1”  $B = (-1, -1, +1, -1, +1, +1, +1, -1)$
    - Pentru a trimite bitul “0”  $-B = (+1, +1, -1, +1, -1, -1, -1, +1)$
- **Exemplu  $A_s=0$   $B_s=1$  semnalele bipolare se “adună” în mediu**
  - Neglijăm interferența, zgomotul
  - Recepție  $X = -A + B = (0, 0, 2, -2, 0, 2, 0, -2)$
- **Receiverul decodează ce au transmis A și B**
  - Se aplică produsul scalar cu secvența bipolară a lui A
    - $A_r = (0, 0, 2, -2, 0, 2, 0, -2) * A = 0 + 0 - 2 - 2 + 0 - 2 + 0 - 2 = -8$  (negativ = “0” =  $A_s$ )
  - $B_r = B * X$ 
    - $B_r = (0, 0, 2, -2, 0, 2, 0, -2) * B = 0 + 0 + 2 + 2 + 0 + 2 + 0 + 2 = 8$  (positiv = “1” =  $B_s$ )

- **Simplificări**
  - Stațiile sunt perfect sincronizate
  - Chipurile se adună liniar <= sunt receptate la aceeași putere
- **Stațiile vecine folosesc secvențe **ortogonale****
  - $S^*T = (\sum S_i T_i)/m = (S_1 T_1 + S_2 T_2 + \dots + S_m T_m)/m = 0$
  - $S^*(-T) = 0$
  - $S^*S = (S_1 S_1 + \dots + S_m S_m)/m = (1 + 1 + \dots + 1)/m = 1$
  - $S^*(-S) = -1$
- **Exemplu: A trimite 1, C trimite 1, B trimite 0**
  - La recepție  $S = A - B + C$
  - Pentru a recupera bitul lui C, receptorul “aplică” codul lui C pe S
  - $S^*A = (A - B + C)^* A = A^*A - B^*A + C^*A = 1 - 0 + 0$
  - $S^*B = (A - B + C)^* B = A^*B - B^*B + C^*B = 0 - 1 + 0$
  - $S^*C = (A - B + C)^* C = A^*C - B^*C + C^*C = 0 - 0 + 1$

# CDMA exemple



a) Secvențe binare pentru A,B,C,D

A: 0 0 0 1 1 0 1 1  
 B: 0 0 1 0 1 1 1 0  
 C: 0 1 0 1 1 1 0 0  
 D: 0 1 0 0 0 0 1 0

(a)

A: (-1 -1 -1 +1 +1 -1 +1 +1)  
 B: (-1 -1 +1 -1 +1 +1 +1 -1)  
 C: (-1 +1 -1 +1 +1 +1 -1 -1)  
 D: (-1 +1 -1 -1 -1 -1 +1 -1)

(b)

b) Secvențe bipolare

Six examples:

c) 6 exemple de transmisiuni

-- 1 -	<b>C</b>	$S_1 = (-1 +1 -1 +1 +1 +1 -1 -1)$
- 1 1 -	<b>B + <math>\overline{C}</math></b>	$S_2 = (-2 0 0 0 +2 +2 0 -2)$
1 0 - -	<b>A + <math>\overline{B}</math></b>	$S_3 = (0 0 -2 +2 0 -2 0 +2)$
1 0 1 -	<b>A + <math>\overline{B}</math> + C</b>	$S_4 = (-1 +1 -3 +3 +1 -1 -1 +1)$
1 1 1 1	<b>A + B + C + D</b>	$S_5 = (-4 0 -2 0 +2 0 +2 -2)$
1 1 0 1	<b>A + B + <math>\overline{C}</math> + D</b>	$S_6 = (-2 -2 0 -2 0 -2 +4 0)$

(c)

d) Cum se recuperează bitul emis de C

$S_1 \cdot C = (1 +1 +1 +1 +1 +1 +1 +1)/8 = 1$   
 $S_2 \cdot C = (2 +0 +0 +0 +2 +2 +0 +2)/8 = 1$   
 $S_3 \cdot C = (0 +0 +2 +2 +0 -2 +0 -2)/8 = 0$  C is silent  
 $S_4 \cdot C = (1 +1 +3 +3 +1 -1 +1 -1)/8 = 1$   
 $S_5 \cdot C = (4 +0 +2 +0 +2 +0 -2 +2)/8 = 1$   
 $S_6 \cdot C = (2 -2 +0 -2 +0 -2 -4 +0)/8 = -1$

(d)

# CDMA – este robust?



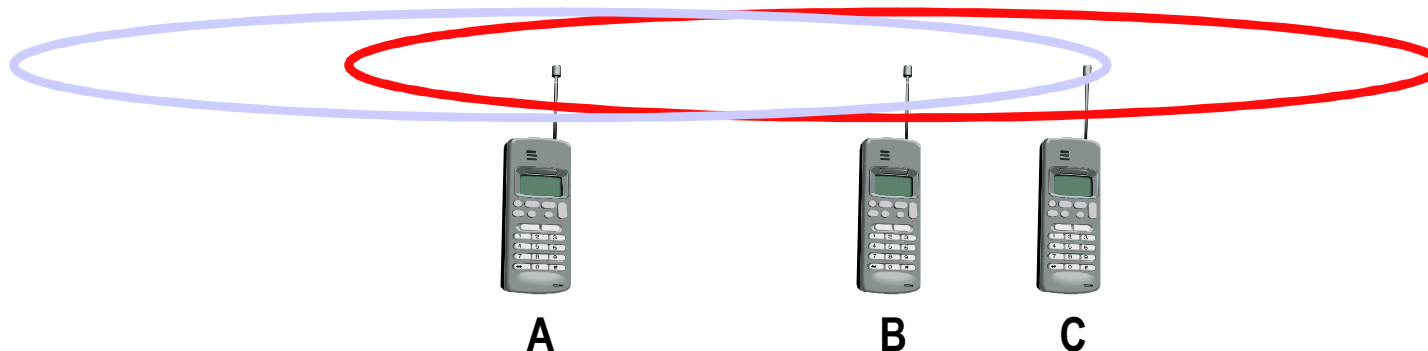
- Ce se întâmplă dacă codurile nu sunt ortogonale?
  - $A = (-1, -1, -1, +1, +1, -1, +1, +1)$
  - $B' = (-1, -1, +1, -1, +1, +1, +1, +1)$
  - $A * B' = (+1+1-1-1+1-1+1+1) = 2$  (trebuia să fie 0)
  - $S = A - B' = (0 \ 0 \ -2 \ 2 \ 0 \ -2 \ 0 \ 0)$
  - $S * A = 6$ , adică “1” OK
  - $S * B' = -6$ , adică “0” OK
  
- Ce se întâmplă dacă se pierde un chip la recepție?
  - $S = A - B = (0, 0, -2, 2, 0, -2, 0, 2)$ 
    - $X_1 = (1, 0, -2, 2, 0, -2, 0, 2)$ ;  $X_1 * A = 7$ ;  $X_1 * B = -9$  OK
    - $X_2 = (0, 0, 0, 2, 0, -2, 0, 2)$ ;  $X_2 * A = 6$ ;  $X_2 * B = -6$  OK
    - $X_3 = (0, 0, 2, -1, 0, -2, 0, 2)$ ;  $X_3 * A = 1$ ;  $X_3 * B = -1$ 
      - **A&B nu tac? Acest bit se pierde.**

- Dar dacă nu se scalează corect puterea la recepție?
  - $A - B = X = (0, 0, -2, 2, 0, -2, 0, 2)$
  - $A - 3B = X_1 = (2, 2, -4, 4, -2, -4, -2, 4)$
  - $X_1 * A = 8$  OK,  $X_1 * B = -24$  OK
  
  - $A/4 - 4B = X_2 = (3.75, 3.75, -4.25, 4.25, -3.75, -4.25, -3.75, 4.25)$
  - $X_2 * B = -32$  OK,  $X_2 * A = 2$ , **A pare tăcut!**

# CDMA: problema terminal near-far



- Terminalele A și B trimit, C primește
  - Puterea semnalului descrește  $\sim 1/d^2$
  - Semnalul lui B domină semnalul lui A
  - C nu primește semnalul de la A (exemplul precedent)



- Problemă severă pentru CDMA
  - Necesită control precis al puterii!
  - Afectează bateria terminalului. De ce?

- Separare folosind coduri ortogonale
- Un receptor poate vedea toți emițătorii
- Secvențe cu multe chipuri => protecție mai bună
- Puterea primita depinde de terminal/distanță/condiții
  - Stacia de bază **trebuie să** controleze puterea terminalelor
- Folosit în 3G (WCDMA)
- **CDMA != CSMA**



# Comparație SDMA/TDMA/FDMA/CDMA



Approach	SDMA	TDMA	FDMA	CDMA
<b>Idea</b>	segment space into cells/sectors	segment sending time into disjoint time-slots, demand driven or fixed patterns	segment the frequency band into disjoint sub-bands	spread the spectrum using orthogonal codes
<b>Terminals</b>	only one terminal can be active in one cell/one sector	all terminals are active for short periods of time on the same frequency	every terminal has its own frequency, uninterrupted	all terminals can be active at the same place at the same moment, uninterrupted
<b>Signal separation</b>	cell structure, directed antennas	synchronization in the time domain	filtering in the frequency domain	code plus special receivers
<b>Advantages</b>	very simple, increases capacity per km <sup>2</sup>	established, fully digital, flexible	simple, established, robust	flexible, less frequency planning needed, soft handover
<b>Dis-advantages</b>	inflexible, antennas typically fixed	guard space needed (multipath propagation), synchronization difficult	inflexible, frequencies are a scarce resource	complex receivers, needs more complicated power control for senders
<b>Comment</b>	only in combination with TDMA, FDMA or CDMA useful	standard in fixed networks, together with FDMA/SDMA used in many mobile networks	typically combined with TDMA (frequency hopping patterns) and SDMA (frequency reuse)	still faces some problems, higher complexity, lowered expectations; will be integrated with TDMA/FDMA

## Condiții, presupuneri:

- **Canal unic:** toate stațiile pot primi/trimitte, folosesc același hardware, unic
- **Coliziuni:** cadrele suprapuse (chiar parțial) se pierd
- **Timp continuu:** cadrele pot apărea pe mediu oricând  
sau
- **Timp discret:** cadrele pot apărea pe mediu la momente fixe
  - 0 cadre în aer = slot neutilizat
  - 1 cadru în are = succes
  - >1 cadre în aer = coliziune
- **Carrier sense** - stațiile pot detecta dacă mediul este liber

# Protocol pentru acces aleator



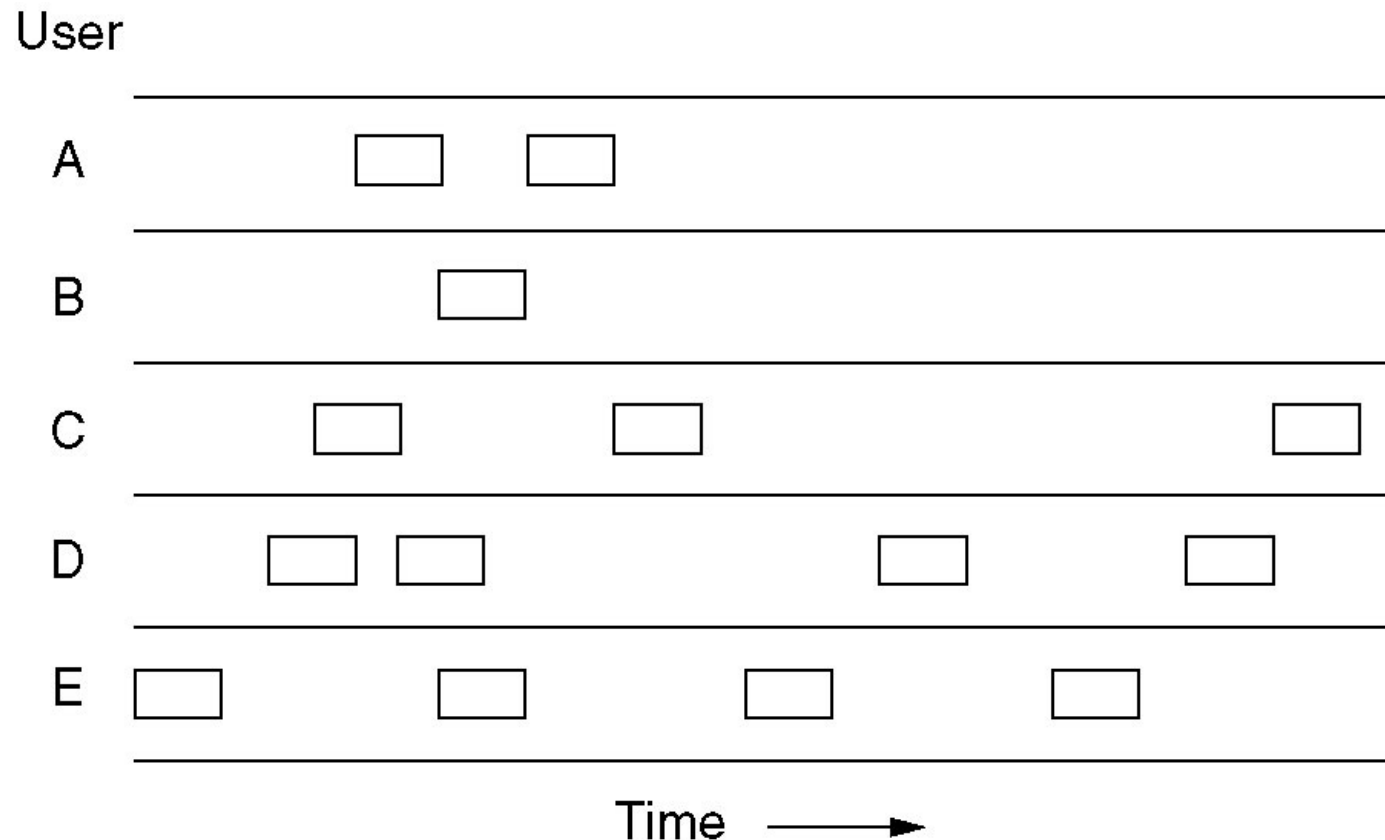
- **ALOHA**
- **CS – carrier sense**
- **CSMA/CD – cu detecția coliziunilor**
- **CSMA/CA – fără CD**

CS = collision avoidance, va fi discutat la WiFi

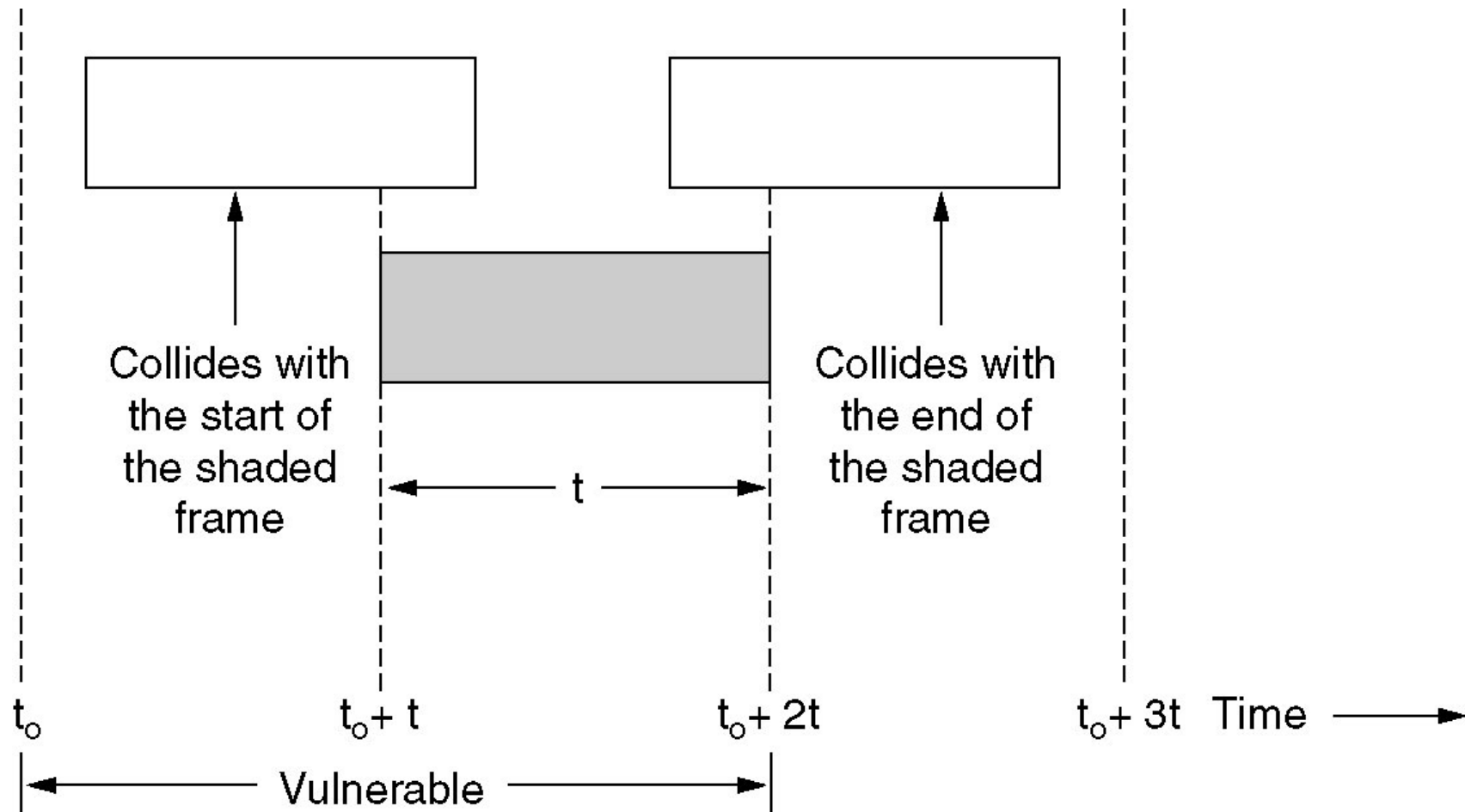
Algoritm simplu: trimite oricând are un cadru de trimis

- La coliziune
  - random wait (de ce?)
  - retransmite
- Cum detectează coliziunile?
  - Topologie stea, 2 canale (*inbound*, *outbound*)
  - Doar stația de bază BS folosește *outbound*
  - Stațiile trimit pe *inbound*, citesc de la *outbound*
  - BS repetă pe *outbound* cadrele primite pe *inbound*

- Cadre de aceeași lungime --> ar crește eficiența
- Probabilitatea de coliziune **crește cu încărcarea**
- throughput maxim ~ 18% - nu putem încărca oricât!



# Efficiența lui ALOHA pur



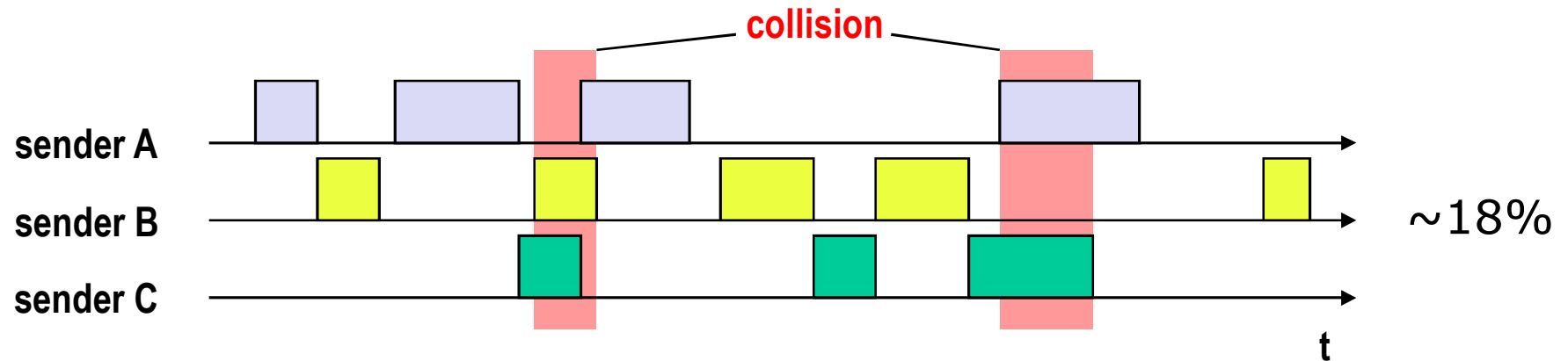
Zona vulnerabilă = durata a două sloturi

Zona vulnerabilă poate fi redusă prin transmiterea **NUMAI** la început de slot

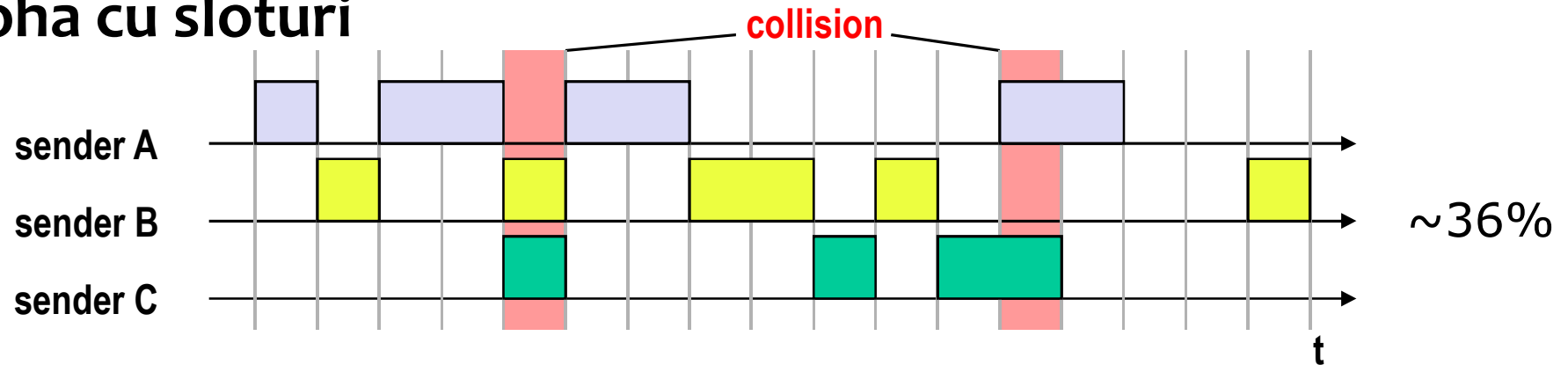
# ALOHA pur/cu sloturi



- Aloha pur



- Aloha cu sloturi



- Aloha pur
  - random, distribuit (fără arbitru), multiplexare în timp
  - ~ 18% capacitate
- Aloha cu sloturi
  - Emisia NUMAI la început de slot
  - ~ 36% capacitate
- dezavantaje
  - Bune doar pentru încărcare redusă
  - Coliziuni care de fapt pot fi evitate

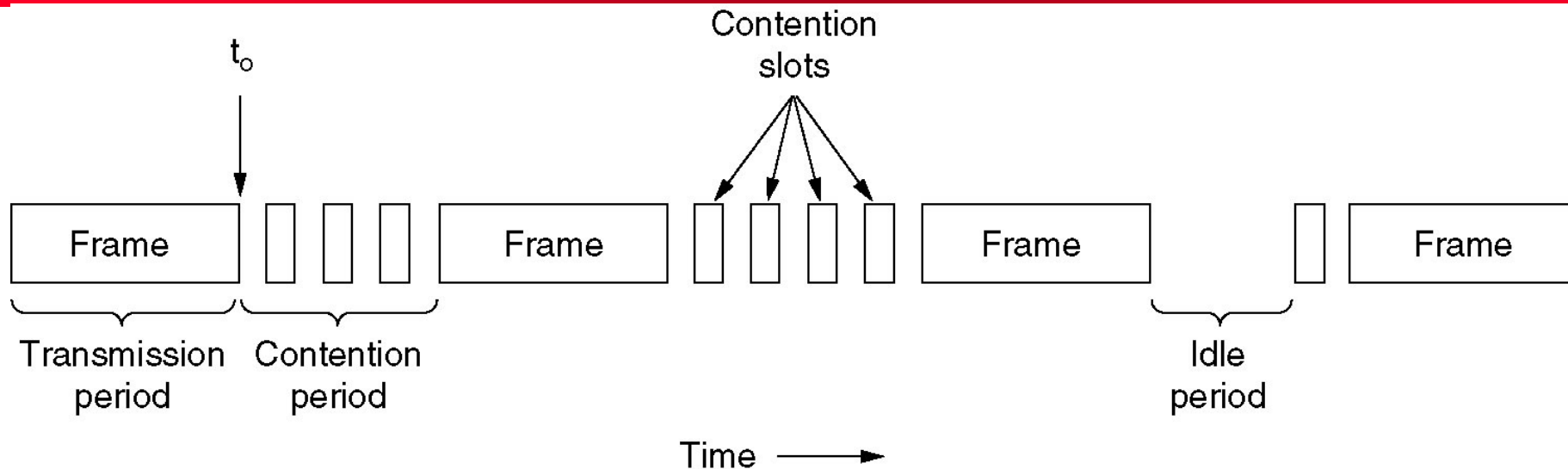


## Carrier Sense CS = posibilitatea de a detecta un vorbitor

- Bazat pe un prag de putere
- Fără demodulare/decodare
- De exemplu 802.11g 6Mbps
  - MCS de 1Mbps necesită o putere de  $-90\text{dBm}$ , specific cardului/MCS-ului
  - **Pragul CS =  $-100\text{dBm}$** , reglabil
- Speră să reducă coliziunile

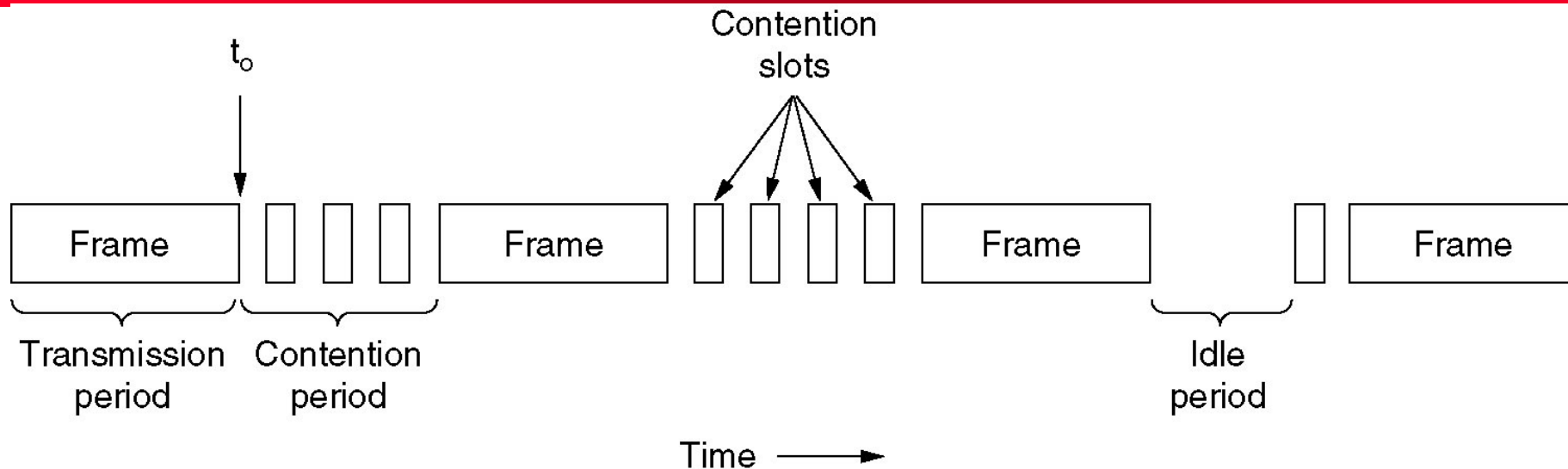
- Aloha și CSMA fără CD:
  - La coliziune se continuă transmisia întregului cadru
  - Se risipește timpul pentru un întreg cadru
- CSMA/CD = carrier sense multiple access with collision detection
  - a) Mediul liber? => trimite după IFG
    - Coliziune?
      - Stop transmisie, dar trimite totuși un cadru minim
      - Random wait; goto a)
  - b) Ocupat? => Așteaptă mediu liber; goto a)

- Abandonează transmisiya după ce este detectată coliziunea
- Nu mai apar coliziuni după ce canalul devine efectiv ocupat
- Folosit de Ethernetul clasic (azi perimat)
- Cum se detectează coliziunile?
  - Se măsoară puterea,
    - Detectie – proces *analogic*
    - Stațiile monitorizează permanent mediul => half duplex
- La coliziune, emițătorul generează un semnal de jamming (48 biți)
  - Receptonat de toate statiile
  - Se trimite o lungime minimă... de ce?



## Perioadă de contenție

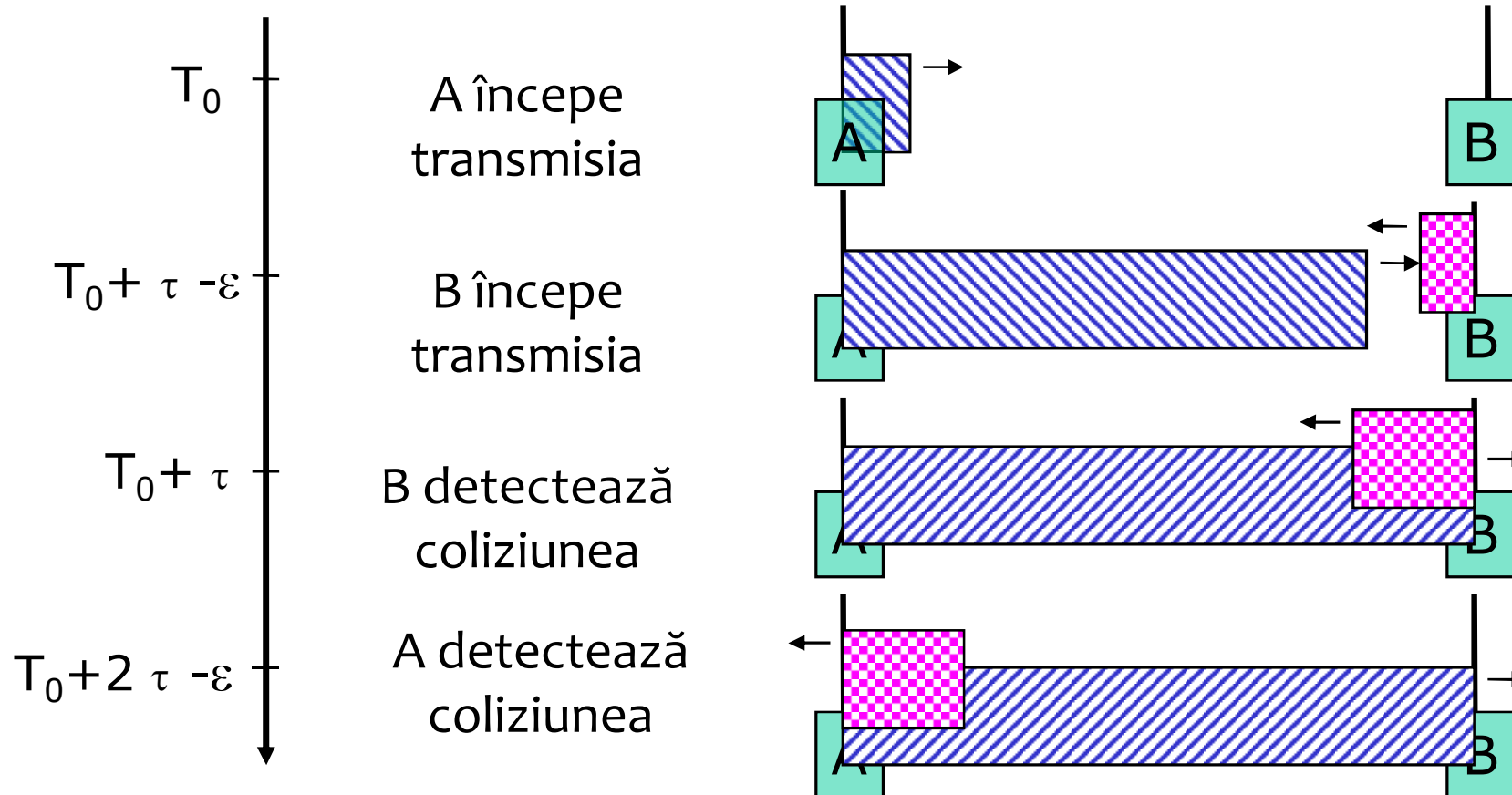
- Așteaptă aleator după coliziune
- Perioadele de contenție alternează cu cele free și busy
- Cât durează detecția coliziunii?
  - Depinde de timpul de priopagare  $\tau$
- Este adevărat că după  $\tau$ , toata lumea poate considera canalul ocupat?



## Perioadă de contenție

- Așteaptă aleator după coliziune
- Perioadele de contenție alternează cu cele free și busy
- Cât durează detecția coliziunii?
  - Depinde de timpul de priopagare  $\tau$
- Este adevărat că după  $\tau$ , toata lumea poate considera canalul ocupat?
  - Nu, este de fapt nevoie de  $2\tau$

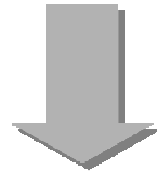
# CSMA/CD detection example



# De ce funcționează CSMA/CD?



1. Emițătorul poate trimite și asculta în același timp  
if (send == listen) then success
2. Semnalul este aproape același la Tx și Rx



**EMIȚĂTORUL** poate detecta când și unde  
se produce coliziunea

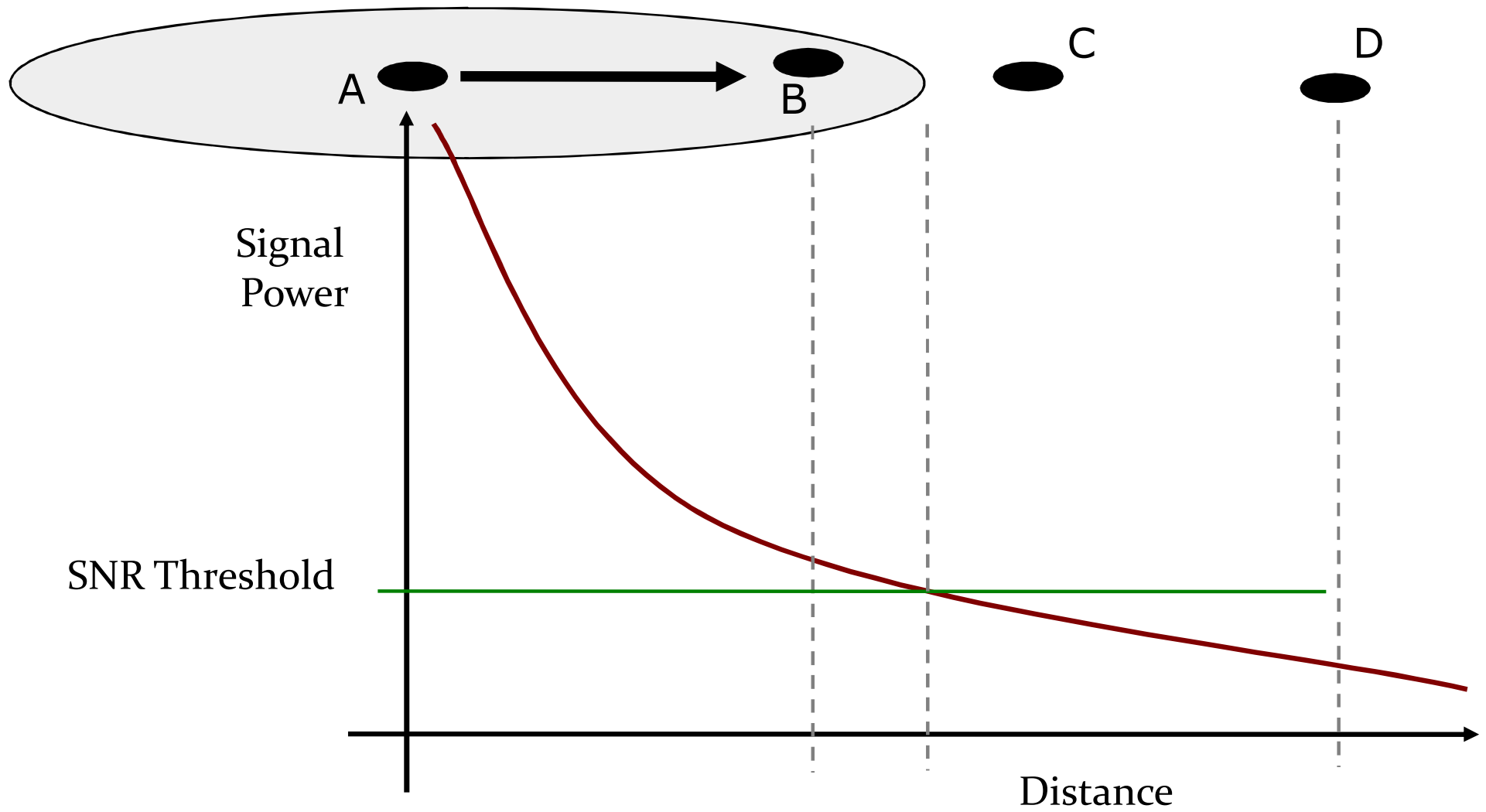
**Costul detecției coliziunii =  $\sim 2\tau$**

**(de aici minimum frame 64 octeți în Ethernet)**

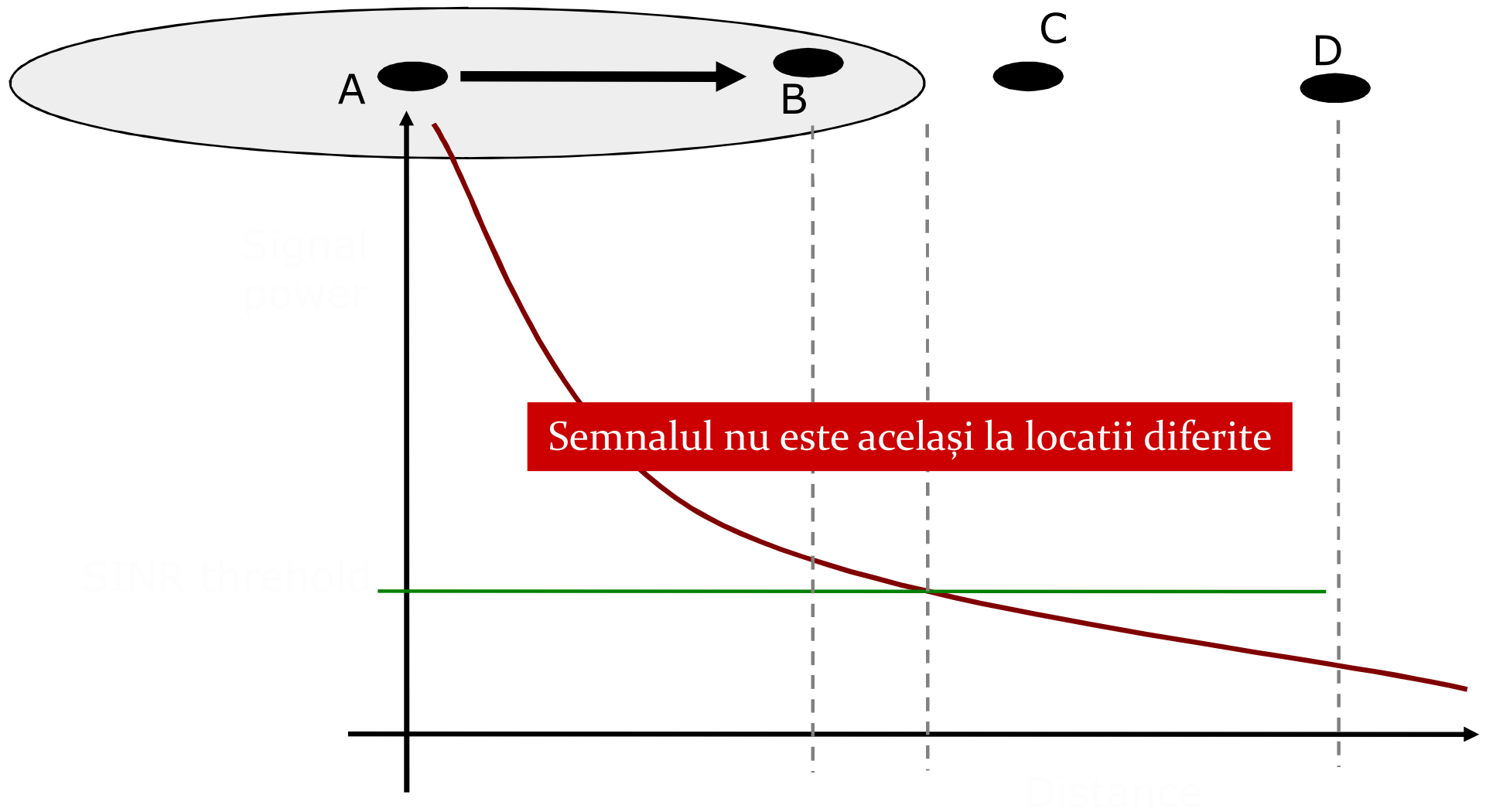
**Nici una dintre cele două presupuneri  
nu ține în wireless :-)**

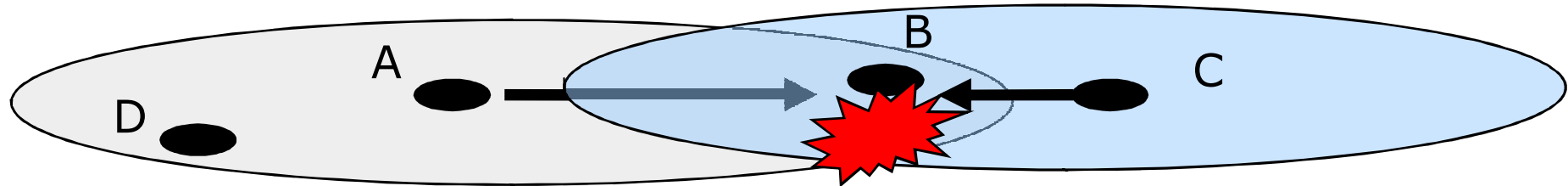


- **Similarități cu Ethernet:**
  - » wireless este un mediu partajat
  - » Interferență între emițatori
  - » CSMA (carrier sense multiple access)
    - Emițatorii și pot detecta purtătoarea unuia altuia
    - “listen before talking”
    - De dorit: maximum o stație să vorbească la un moment dat
- **Diferențe:**
  - » CD ~~este dificil~~ **imposibil**
    - Canal unic, comunicație de tip simplex
  - » Canale slabe: BER mare, variabilitate mare în spațiu/timp
  - » Terminale ascunse și expuse
  - » Costul coliziunii: întregul cadru + retransmisii



A nu poate trimite si primi în același timp

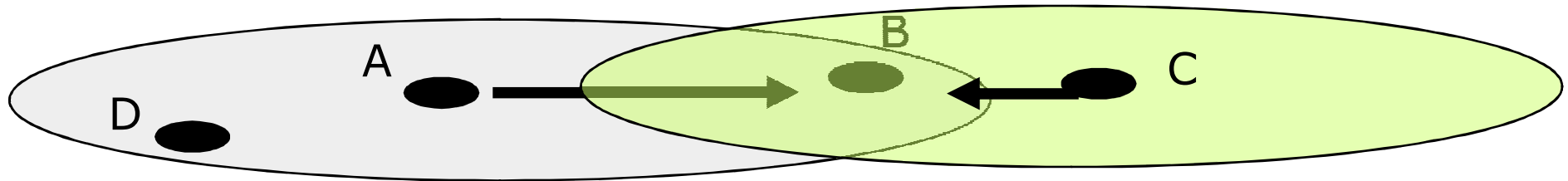




## Recepția semnalului bazată pe SINR

- Emițătorul se aude doar pe sine
- Nu poate estima calitatea semnalului la receptor

# SINR computation



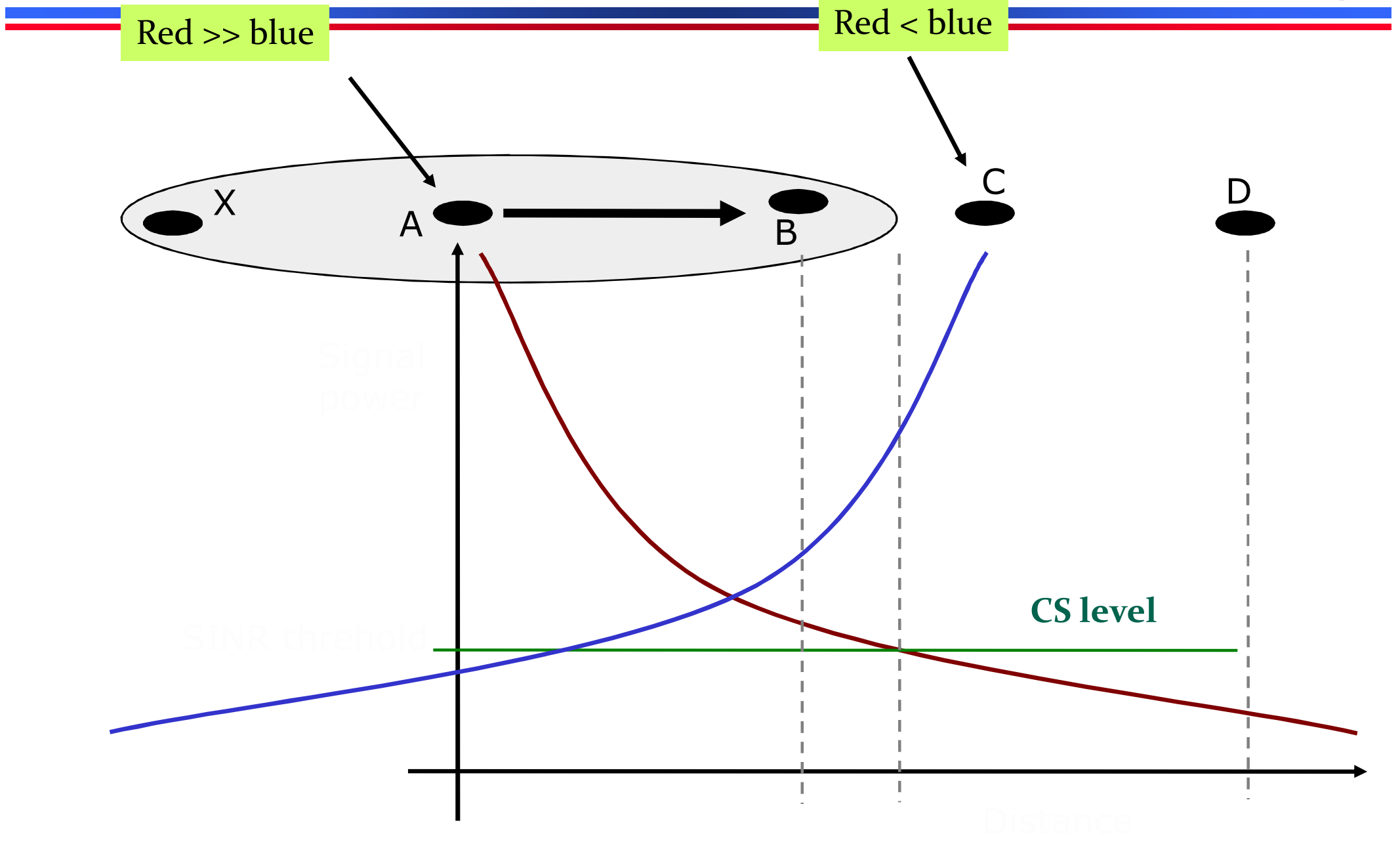
$$SINR = \frac{Semnal(S)}{Interferenta(I) + Zgomot(N)}$$

$$S_B^A = \frac{P_{transmit}^A}{d_{AB}^\alpha}$$

$$I_B^C = \frac{P_{transmit}^C}{d_{CB}^\alpha}$$

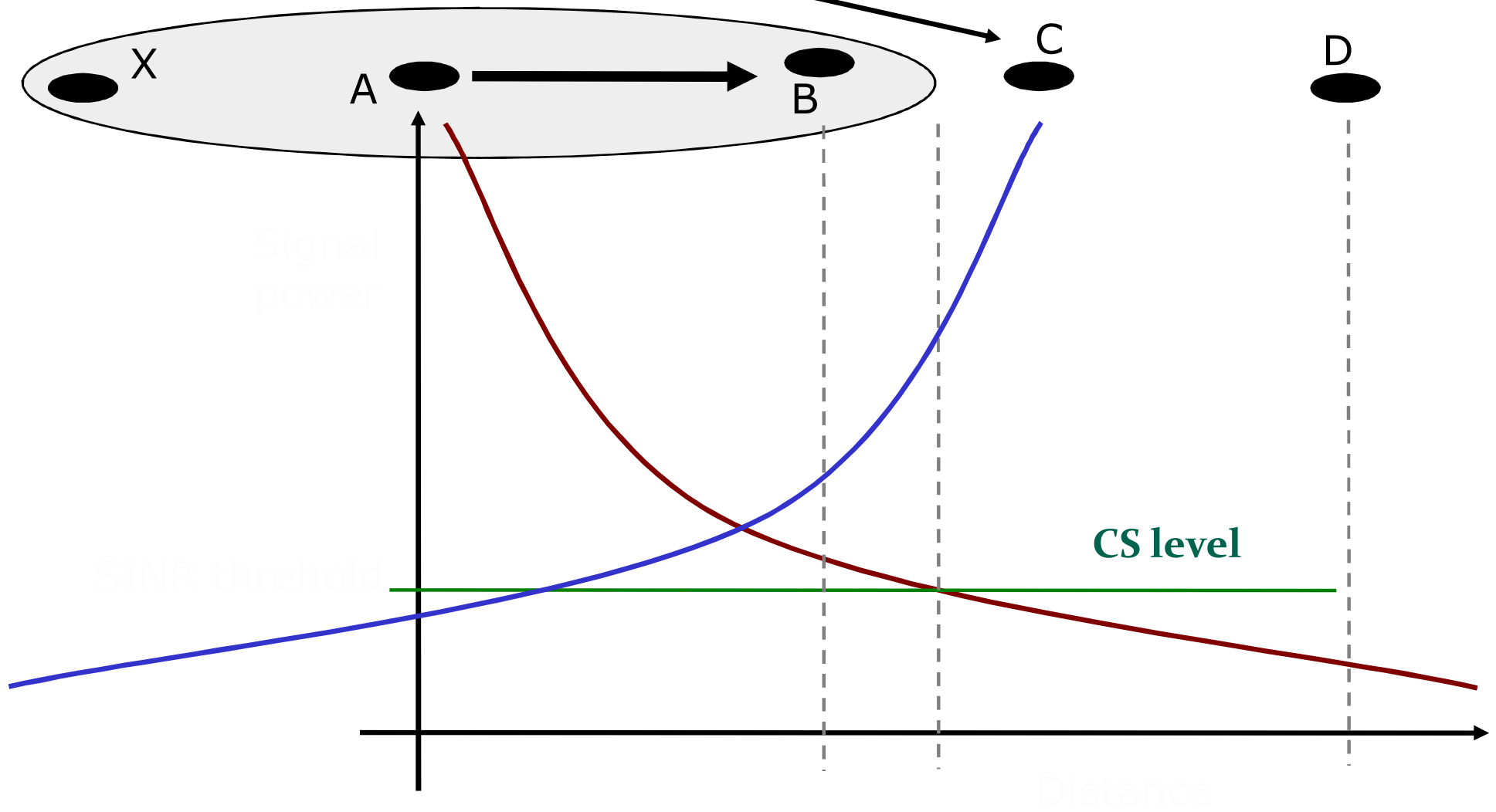


$$SINR_B^A = \frac{\frac{P_{transmit}^A}{d_{AB}^\alpha}}{N + \frac{P_{transmit}^C}{d_{CB}^\alpha}}$$



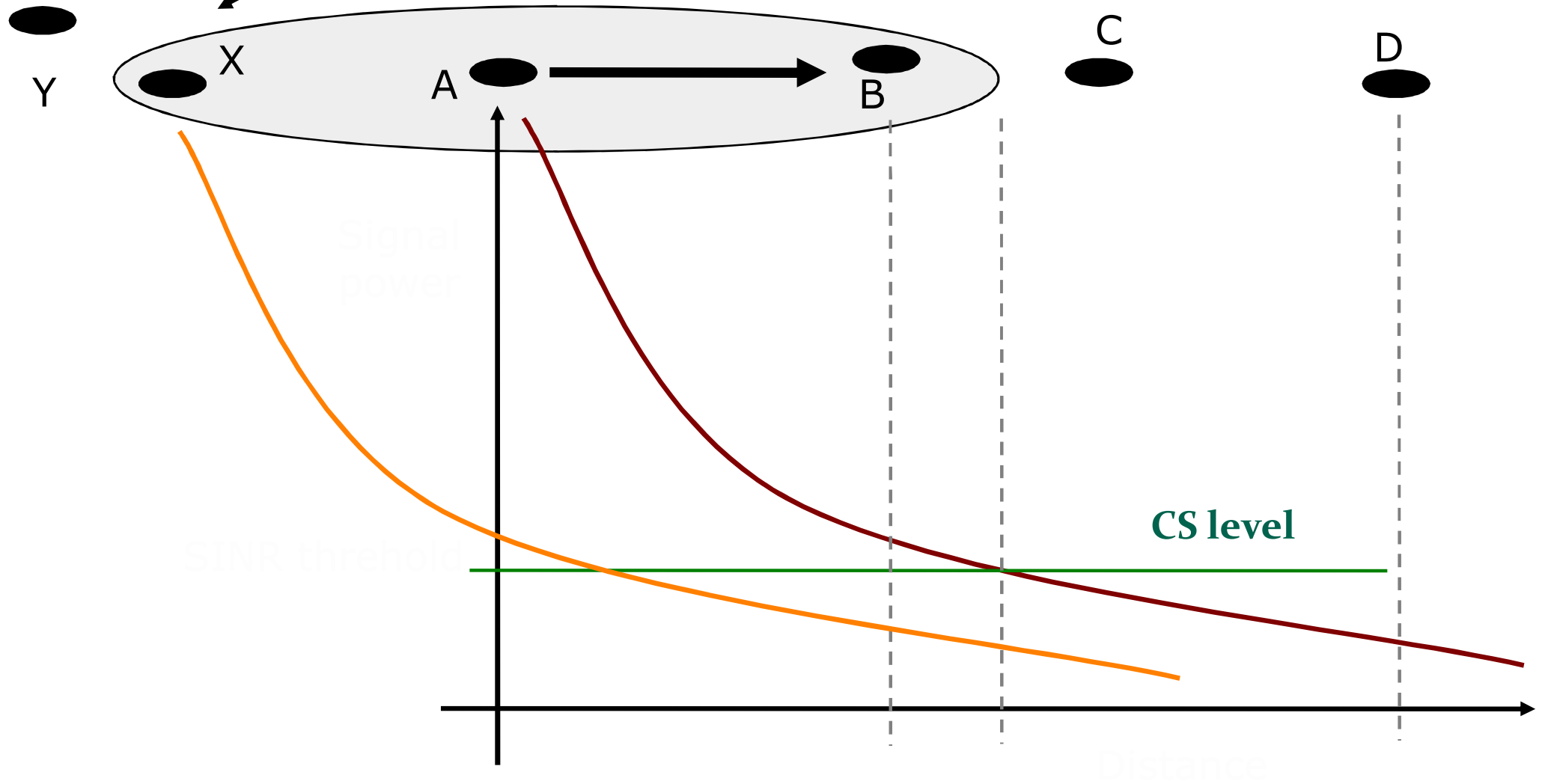
Important: C nu-l aude pe A, crează interferență la B

C este terminal ascuns pentru A



Important: X îl aude pe A, dar nu trebuie să cedeze accesul deoarece vorbește cu Y

X este un terminal expus pentru A

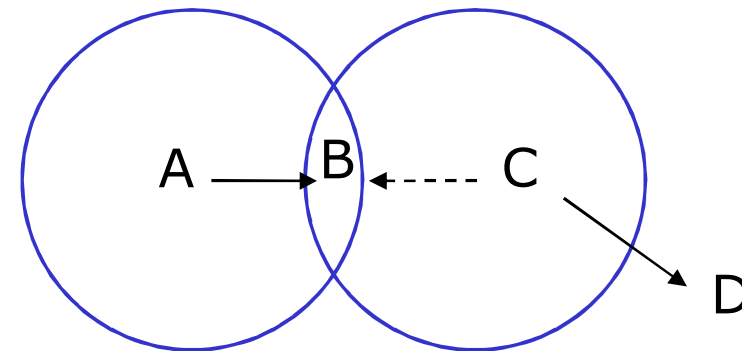




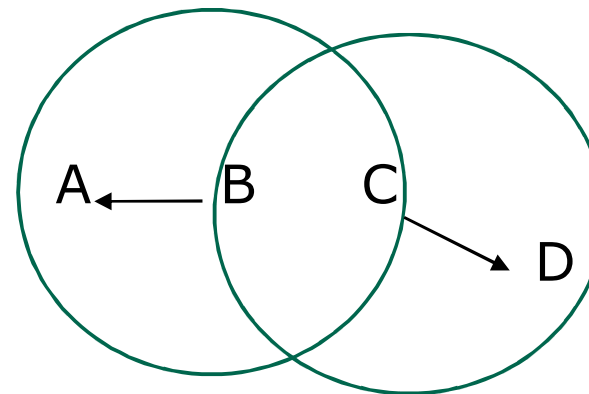
# Terminale ascunse & expuse



- ↙ terminal ascuns (HT)
  - » A și C pot trimite în același timp



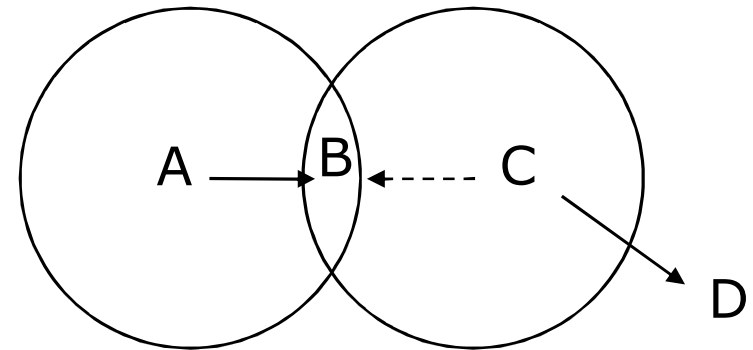
- ↙ Terminal expus (ET)
  - » B and C nu pot trimite în același timp



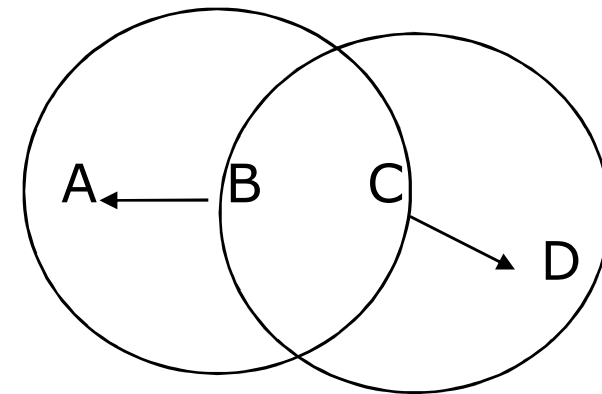
# Terminale ascunse & expuse



- Situațiile reale sunt rareori doar HT/ET
  - Canale asimetrice
  - Hardware diferit
  - Combinații de HT, ET



- ET Asimetric
  - doar B aude pe C, cedează mereu
  - Inechitate între fluxurile B->A, C->D



- Partajare fixă : TDMA, FDMA, CDMA
  - Scalabilitate și granularitate reduse
  - Zone de gardă – separare timp, frecvență
  - terminale near-far, controlul puterii
- Acces aleator
  - Aloha, CS, CSMA/CD
- Probleme în rețelele wireless
  - Semnalul se pierde drastic  $1/d^n$
  - Emițătorii folosesc CS, dar coliziunile se petrec la receptor
  - Emițătorii nu percep coliziunile => CD nu funcționează
  - terminal ascuns => când CS nu ajută
  - terminal expus => CS funcționează, dar nu ar trebui
  - terminal near-far => terminalele apropiate se ‘aud’ prea tare
  - controlul puterii