

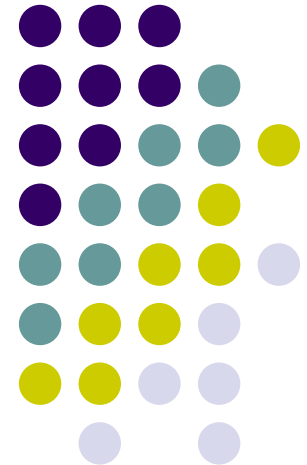
Beágyazott információs rendszerek

Szenzorhálózatok III.

Kommunikáció a szenzorhálózatokban

2005. április 6.

Simon Gyula



Budapesti Műszaki és Gazdaságtudományi Egyetem

© 2004 Méréstechnika és Információs Rendszerek Tanszék

Tartalom



Kommunikáció a szenzorhálózatokban

- Middleware fogalma, kapcsolata a kommunikációval
- Az alsóbb rétegek:
 - A fizikai réteg
 - Az adatkapcsolati réteg
- Routing protokollok
- Példák
 - Flooding
 - Gradiens alapú
 - LEACH
 - GEAR

Mi a *middleware*?



- Szoftver réteg az alkalmazás és a helyi operációs rendszer között
- Hálózati szintű operációs rendszerként szolgál:
 - hálózati szintű erőforrás-kezelést és
 - hálózati szintű függvényeket biztosít, amelyeket az alkalmazás írója felhasználhat.
- Erőforrás-hiányos környezetben!

Middleware szolgáltatások

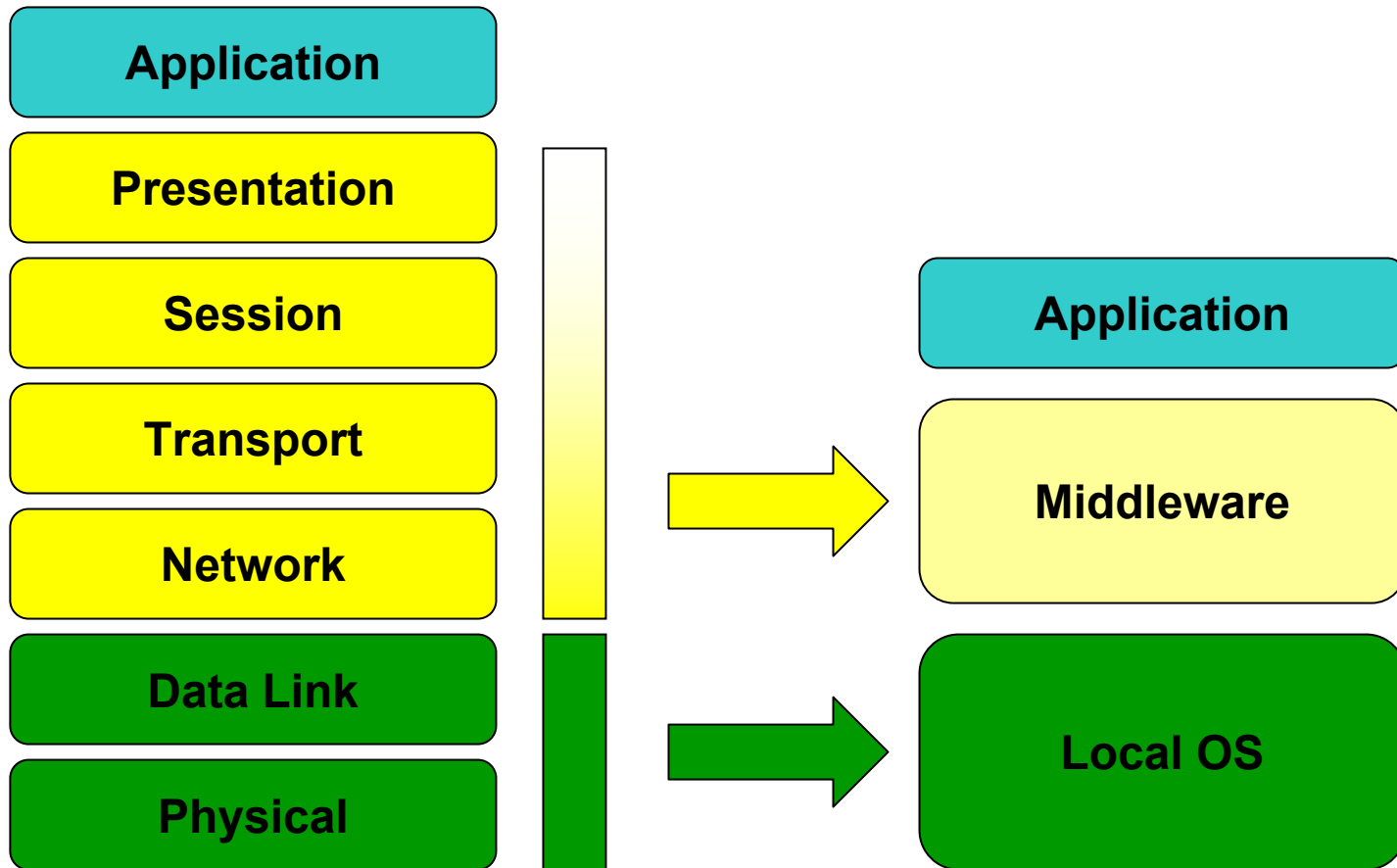


Tipikus szolgáltatások:

- Routing
- Lokalizáció
 - Ön
 - Célpont
- Idő-szinkronizáció



Kommunikáció és a middleware



Általános hálózati modell (OSI)

Szenzorhálózat egyszerűsített modellje

Kommunikáció



- Tipikus alkalmazás: Adatgyűjtés
 - Szélessávú, rövid élettartam
 - „Hagyományos” megoldások
 - Kis sávszélesség, **hosszú élettartam**
 - Speciális megoldások
 - Üzemelés kis kitöltési tényezővel
- Szenzorhálózatok általános követelménye:
 - Olcsó legyen

Kommunikáció szenzorhálózatokban



Lehetséges alternatívák:

- Rádiófrekvenciás
- Akusztikus
- Infravörös fény

Tipikus megoldás: **rádió**

- Szabványos megoldások:
 - Bluetooth
 - IEEE 802.11b
 - ZigBee / IEEE 802.15.4
- Speciális WSN megoldások:
 - PicoRadio
 - WINS
 - μ AMPS

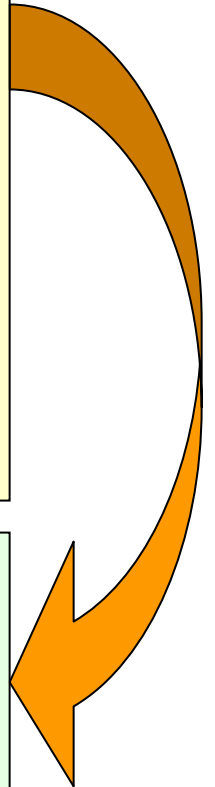
Fizikai réteg

Tervezési szempontok:

- Méret
 - szűrő
 - antenna
- Időalap pontossága
 - ne legyen kritikus
- Szabványos frekvenciatartomány
 - Békés együttélés más eszközökkel
- Energiatakarékos működés támogatása

Jelenlegi trend: ISM 2.4GHz, szórt spektrum

- ZigBee / IEEE 802.15.4
- IEEE 802.11b (Wi-Fi) WLAN
- Bluetooth WPAN



Fizikai réteg szabványok



Bluetooth

- Eredeti cél: Wireless Personal Area Network
- 1Mbit/s, 2.4GHz, szórt spektrum (frekvenciaugratásos)
- Problémák:
 - Hosszas hálózat-felderítés (aszinkron működés)
 - Költséges szűrő (keskeny sávú moduláció)
 - Hosszú „bemelegedési idő”, rossz hatékonyság

IEEE 802.11b wireless LAN szabvány

- 1-2Mbit/s, 2.4GHz, szórt spektrumú (direkt szekvenciális)

IEEE 802.15.4 (Zigbee)

- Kifejezetten kis sebességű WSN alkalmazásokhoz
- Max. 250 kbit/s, 0.8-0.9-2.4GHz, szórt spektrum

Adatkapcsolati réteg



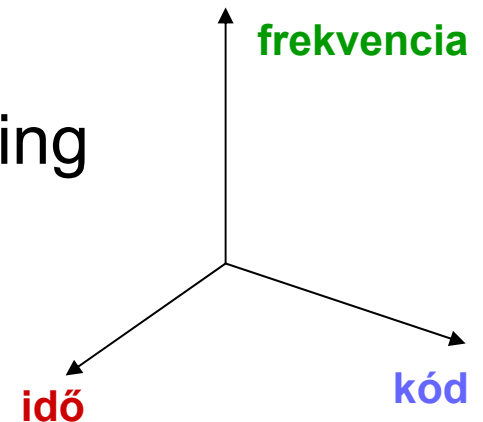
- Adatkapcsolati réteg fő feladatai:
 - keretképzés
 - hibadetektálás és –javítás
 - pl. Hamming kód, CRC
 - forgalomszabályozás (flow control)
 - pl: ACK
 - közeghozzáférés vezérlése
 - MAC – Media Access Control
 - Ütközés elkerülése

MAC – Media Access Control



- Statikus csatornamegosztási módszerek:

- **TDM** – Time Division Multiplexing
- **FDM** – Frequency Division Multiplexing
- **CDM** – Code Division Multiplexing



- Dinamikus csatornamegosztás

- Csatornahozzáférési jog kiosztása dinamikusan igények szerint
- pl. CSMA, RTS/CTS

CSMA



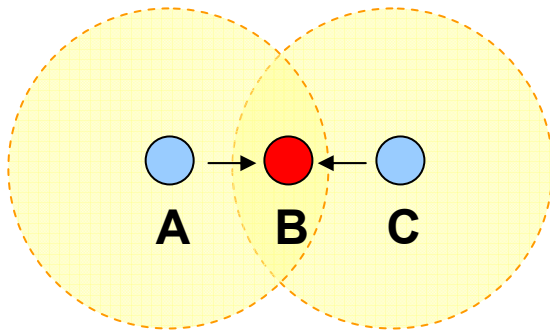
Tipikus megoldás szenzorhálózatokban: CSMA

- CSMA: *Carrier Sense Multiple Access*
- Ütközés elkerülés:
 - Adás előtt belehallgat a csatornába
 - Ha nem érzékel adást, adni kezd
 - Ha adást érzékel, várakozik

CSMA problémák

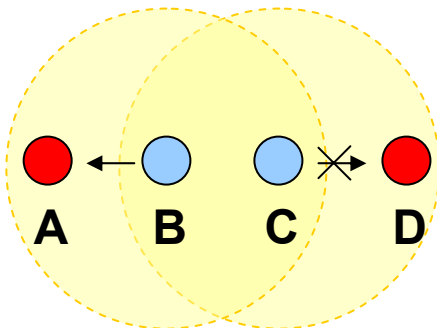


Rejtett terminál problémája



- A ad B-nek
- C nem hallja A-t !
- C is ad B-nek
- B egyik adást sem tudja venni

Látható terminál problémája



- B ad A-nak
- C is szeretne adni D-nek
- C hallja B-t
- C nem ad, bár nem okozna ütközést

CSMA módosítások



CSMA foglalt jelzéssel

- Két csatorna:
 - Adat
 - Foglaltság
 - Vevő a foglaltság csatornán folyamatosan jelez
 - Adás előtt ellenőrzik a
 - Adatcsatornát
 - Foglaltság-csatornát
- ☹ Egyszerre adás/vétel (ár!)
- ☹ Két csatorna, nagyobb sávzélesség

RTS/CTS

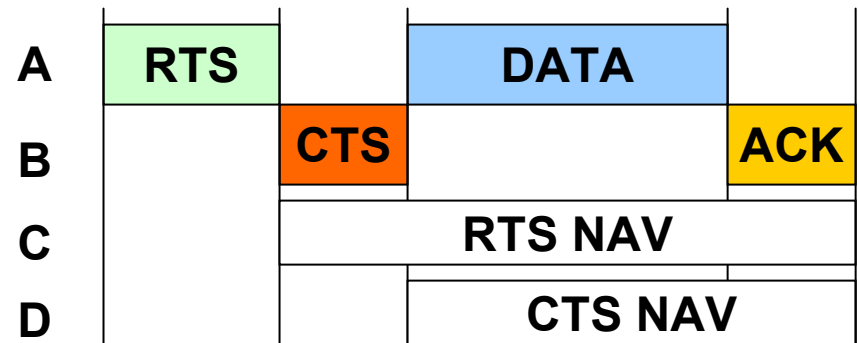
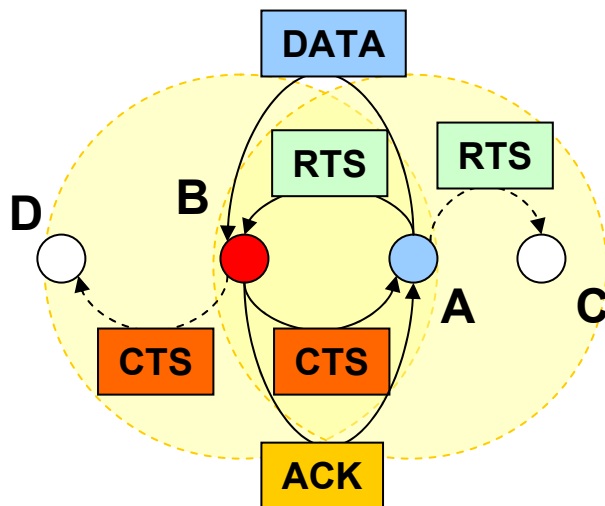


- Request To Send/Clear To Send
- Wireless LAN is használja (IEEE 802.11)
- Két fázis:
 - Handshake
 - Adattovábbítás
- Alapgondolat: ütközés a vevőnél történik
- Kizárja a rejtett terminál problémát
- Hátrány: overhead
- Hosszabb üzeneteknél előnyös

RTS/CTS működése



1. Adó RTS üzenetet küld
 2. Vevő CTS üzenettel válaszol
 3. CTS vétele után az adó továbbítja az adatcsomagot
- Többi node RTS, CTS vétele után nem adhat!
- RTS, CTS tartalmazza az adás hosszát

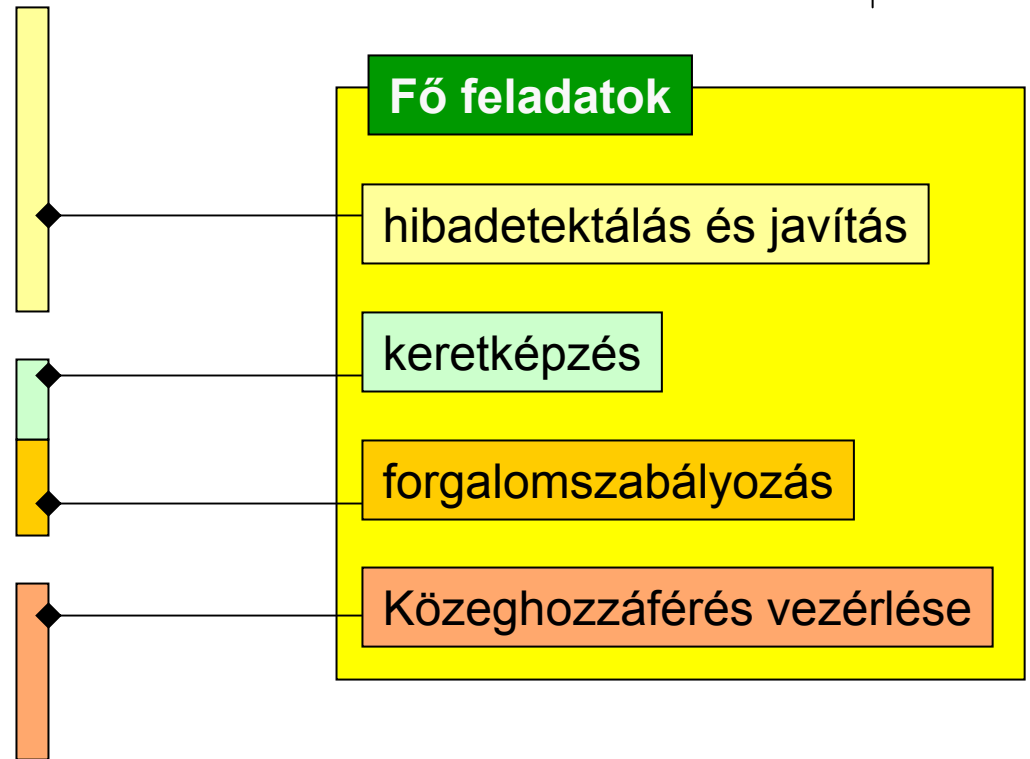


* NAV = Network Allocation Vector

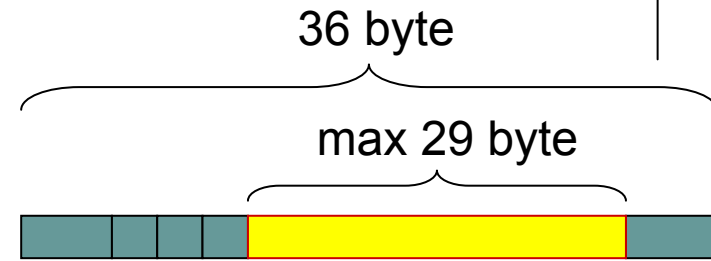
Esettanulmány: Mica mote



- Üzenetstruktúra
- Csatornakódolás
- Adás és vétel
- MAC-layer



Mica: Üzenet-struktúra



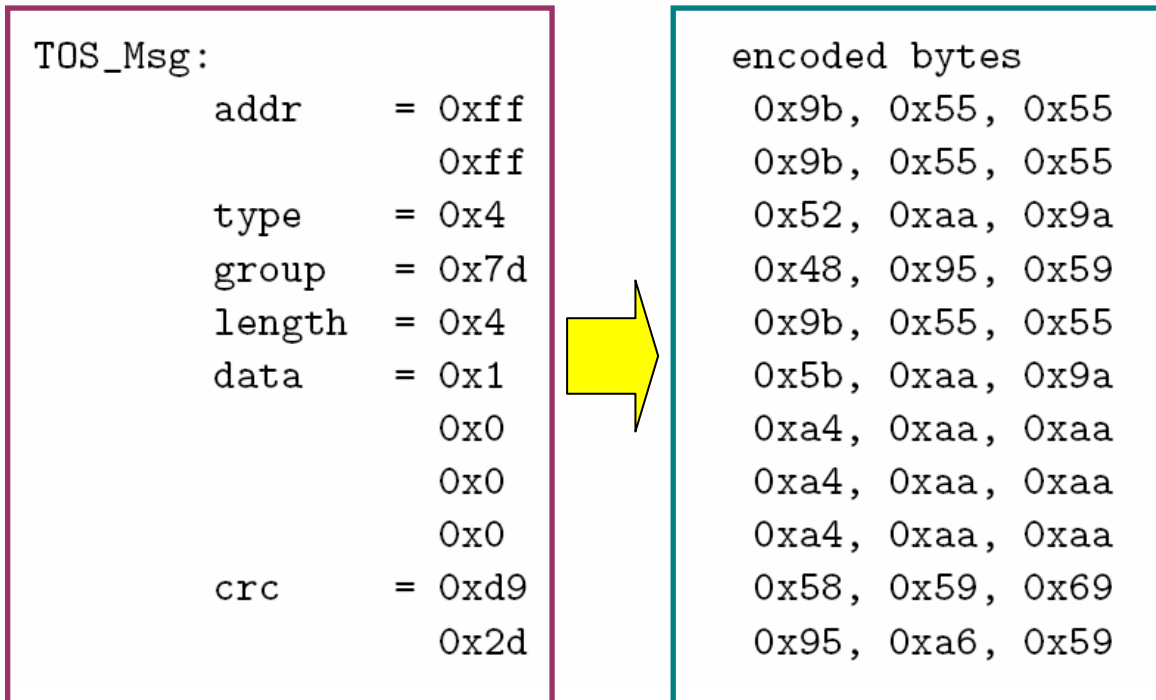
```
typedef struct TOS_Msg
{
    uint16_t addr;
    uint8_t type;
    uint8_t group;
    uint8_t length;
    int8_t data[TOSH_DATA_LENGTH];
    uint16_t crc;
    uint16_t strength;
    uint8_t ack;
    uint16_t time;
} TOS_Msg;
```



Mica: Csatorna kódolás



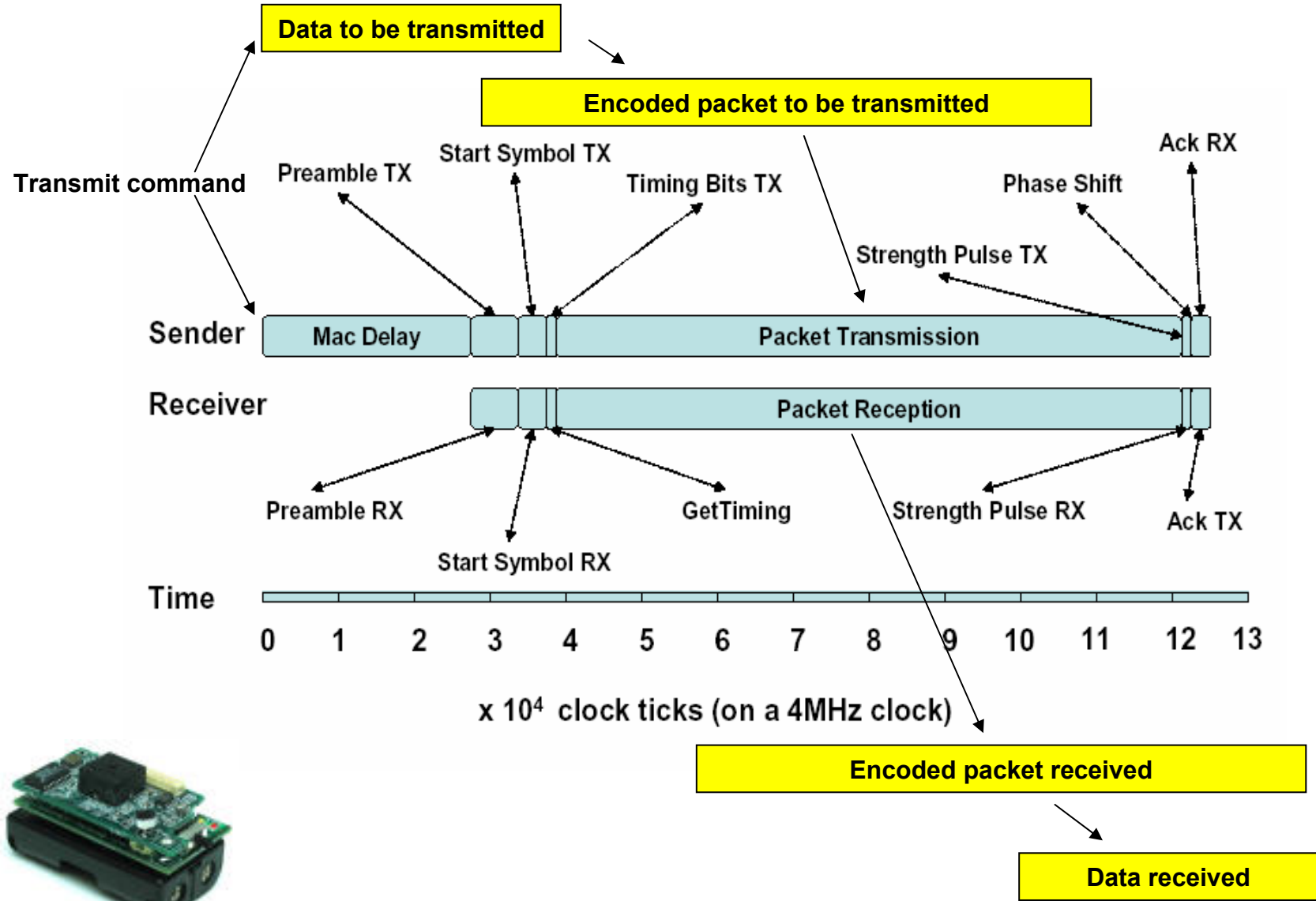
- **SecDed** (Single Error Correction, Double Error detection) **kódolás**
- **Példa:**



Adatátviteli hibadetektálás/javítás: CRC, SecDed



Mica: Adás és vétel

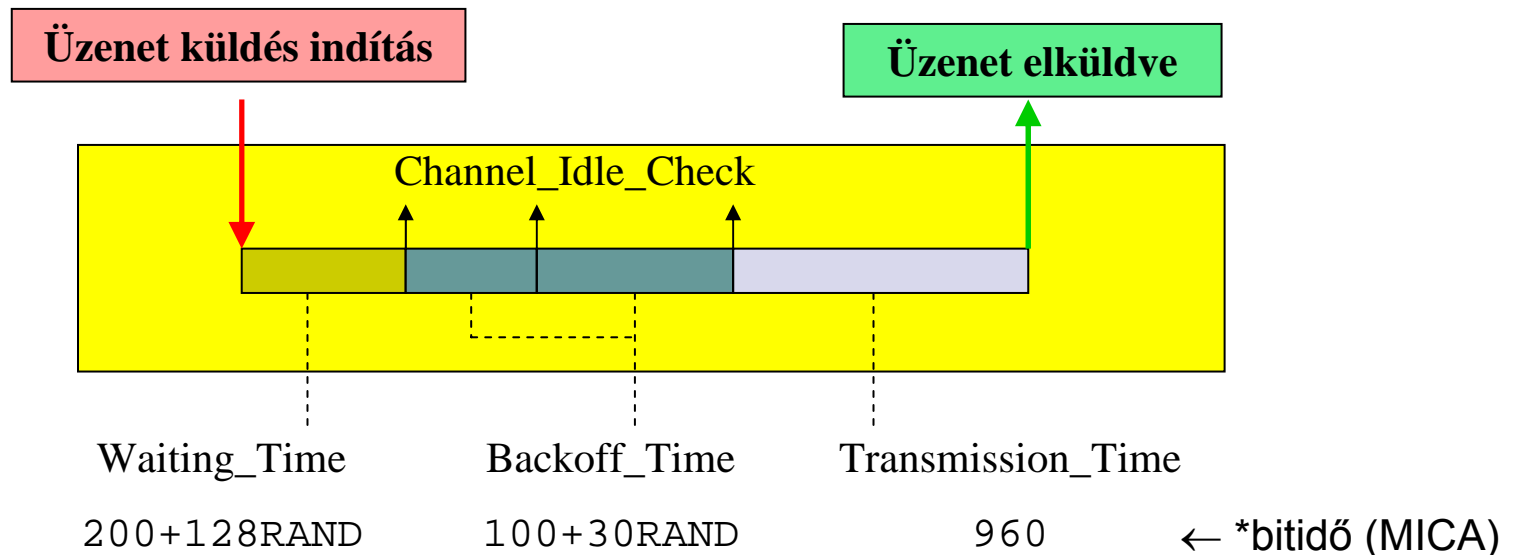


Mica: MAC layer



Media Access Control: CSMA

- Filozófia (miért nem kell valami hatékonyabb [pl. RTS/CTS]):
 - Rövid üzeneteket használnak (36 byte)
 - Üzenet hossza \cong control üzenet hossza (pl. RTS/CTS)
 - Akkor pedig minek a control üzenet?



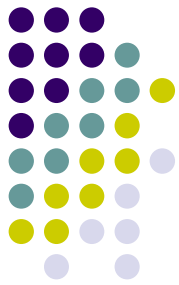
Routing



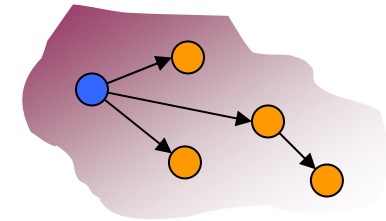
- Fontos tervezési „paraméter”: topológia
- A hálózat ad-hoc:
 - Node-ok véletlenszerű eloszlásúak
 - Linkek véletlenszerűen jönnek létre
 - Linkek nem megbízhatóak (fading!)
 - Lehetnek mobil node-ok
 - Nagy számú node
- Gráfelméleti megközelítés
 - ☺ Kezelhető modell, precíz matematikai tárgyalás
 - ☹ Statikus, determinisztikus modellek nem igazak

Routing

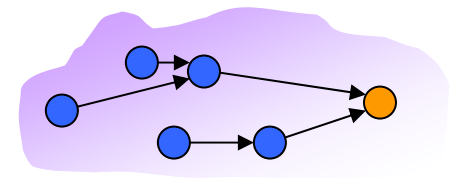
Tipikus feladatok



- Egy forrás → sok (akár minden node) cél
 - Pl: központ utasításokat terjeszt a hálózatban

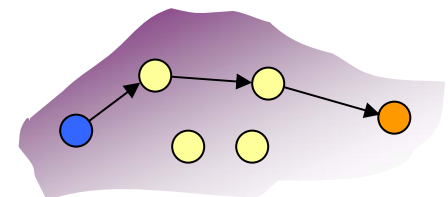


- Sok forrás → egy cél
 - Pl: adatgyűjtés és továbbítás a központba



Egy forrás → egy cél

- Pl: adatcsere node-ok között



Routing



Adatküldési modellek

- Idővezérelt
 - Szenzorok aktivitása és az adatküldés idővezérelt
 - PI. Adatgyűjtés
 - Energiatakarékos működéshez előnyös
 - alvás – szinkronizált ébredés
- Eseményvezérelt
 - Időkritikus alkalmazás
 - Szenzoraktivitást környezeti események váltják ki
 - Energiatakarékos üzem nehezebben megvalósítható
- Lekérdezéses
 - Szenzorok a központ lekérdező parancsára aktiválódnak

Routing

Tipikus hálózati struktúrák



- Flat
 - Egyenrangú node-ok
 - Nehezen skálázható
- Hierarchikus
 - Csoportok (cluster-ek) alakulnak
 - Csoporton belüli
 - Csoportok közötti kommunikáció
 - Csoportok között „vezérlő” node-ok tartanak kapcsolatot
 - Heterogén: vezetők kitüntetett képességűek
 - Homogén / dinamikus: vezető szerep változik

Tipikus routing algoritmusok

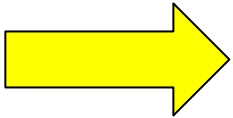


- Számos megoldás létezik
- Néhány tipikus példa:
 - Flat
 - Flood routing
 - Gradiens alapú
 - Hierarchikus
 - LEACH (Low Energy Adaptive Clustering Hierarchy)
 - Egyéb:
 - GEAR (Geographic and Energy Aware Routing)

Routing algoritmusok



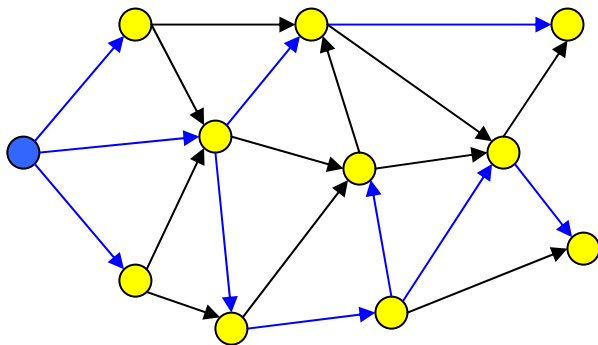
- Flat
 - **Flood routing**
 - Gradiens alapú
- Hierarchikus
 - LEACH
- Egyéb:
 - GEAR



Flood routing



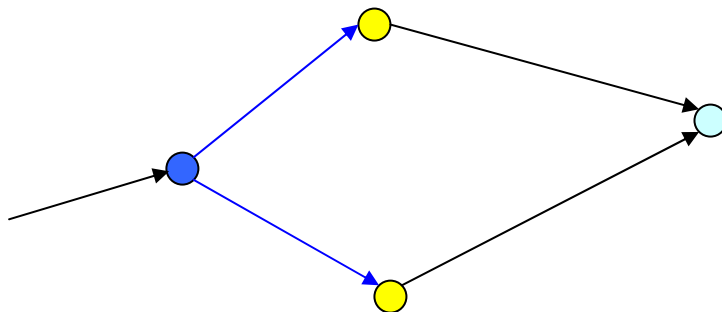
- Flooding - elárasztás
- Broadcast üzenetek (nem pont-pont!)
- Minden üzenet első vételénél a vevő
 - megjegyzi az üzenetet (pl. azonosítóját)
 - szétsugározza az üzenetet
- Tipikus alkalmazás: egy → sok (command)



Flood routing: tulajdonságok



- Előnyök:
 - Egyszerű
 - Hibatűrő (nagy redundancia)
- Hátrányok:
 - Nagyszámú (felesleges?) üzenet
 - Energiafogyasztás
 - Ütközések (rejtett terminál)



Flood routing: módosítások

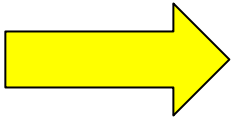


- Túl sok üzenet:
 - Nem kell mindet továbbítani
 - Vevő csak p valószínűséggel terjeszt tovább
 - $p=?$ Topológiafüggő (szomszédság)
- Ütközés
 - Vétel után nem azonnal történik a továbbítás
 - Véletlen várakozási idő

Routing algoritmusok

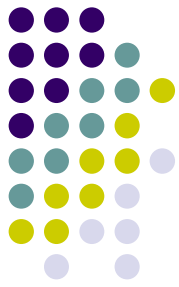


- Flat
 - Flood routing
 - **Gradiens alapú**
- Hierarchikus
 - LEACH
- Egyéb:
 - GEAR



Gradiens-alapú routing

Gradient Based Routing (GBR)



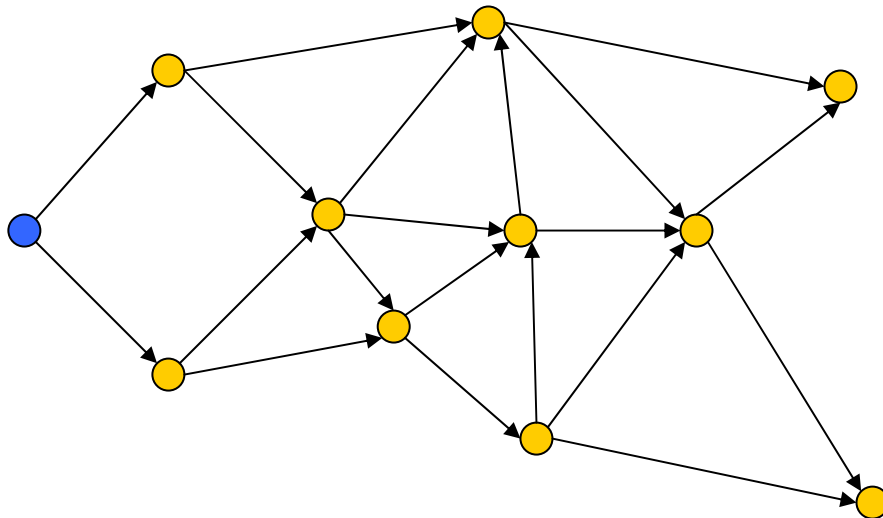
- Három fázis:
 1. Kérés
 2. Gradiens számítás
 3. Adatszolgáltatás
- Tipikus alkalmazás: sok \rightarrow egy (adatgyűjtés)

GBR: kérés



1. Kérés

- Központ kérést küld hálózatba
- Terjesztés teljes elárasztással



1. Kérés

2. Gradiens számítás

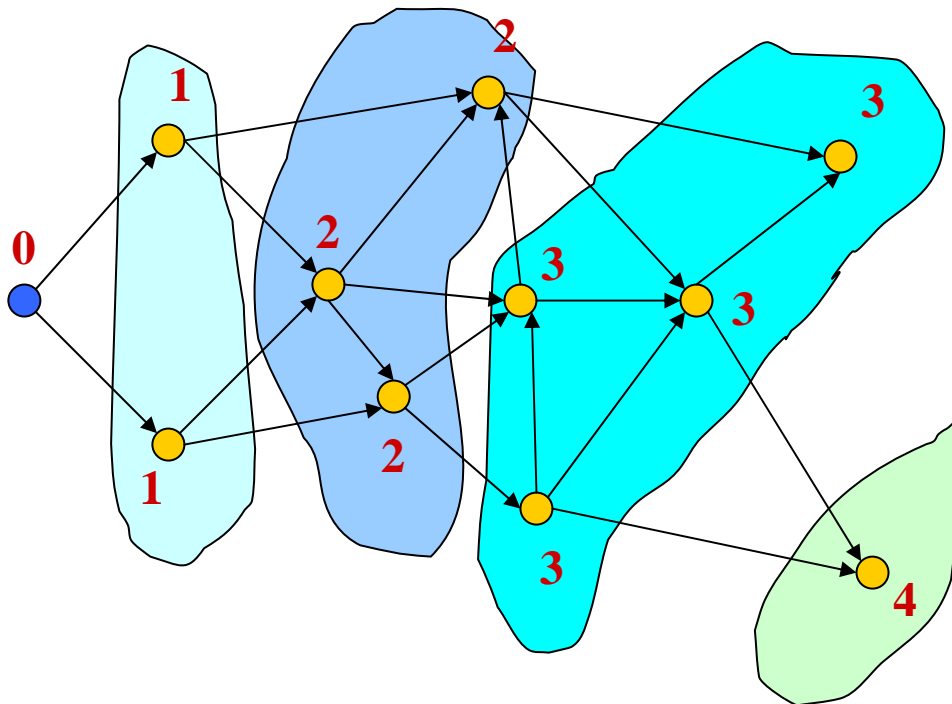
3. Adatszolgáltatás

GBR: gradiens számítás



2. Gradiens számítás

- Kérés terjesztése közben gradiens „mérés”
- Gradiens a legrövidebb távolság a központtól



1. Kérés

2. Gradiens számítás

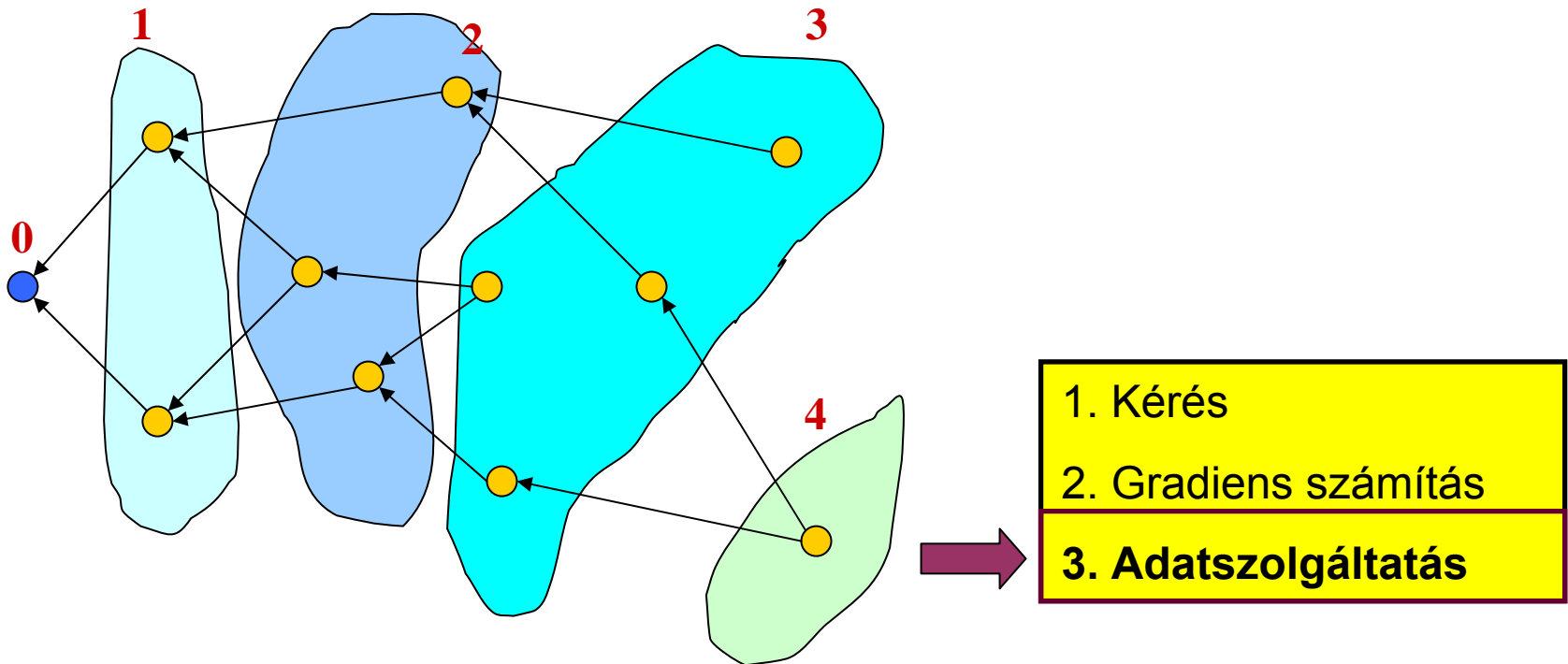
3. Adatszolgáltatás

GBR: adatszolgáltatás



3. Adatszolgáltatás

- Adat továbbítás a legkisebb gradiens irányába
- Aggregáció lehetséges adattovábbítás közben



GBR: változatok

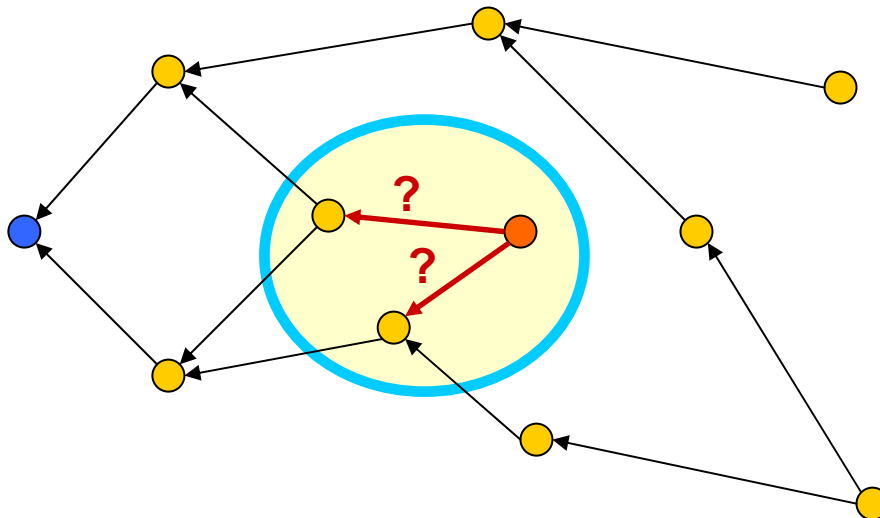


Több lehetséges útvonalból melyiket válasszuk?

- Sztochasztikus
 - Véletlen választás
- Stream-alapú
 - új stream igyekszik elkerülni már létező stream-eket

Energia-alapú kiegészítés:

- kevés energiájú node megnöveli a saját gradiensét
- így másfelé tereli a forgalmat

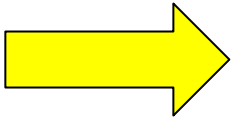


1. Kérés
2. Gradiens számítás
3. Adatszolgáltatás

Routing algoritmusok



- Flat
 - Flood routing
 - Gradiens alapú
- Hierarchikus
 - **LEACH**
- Egyéb:
 - GEAR

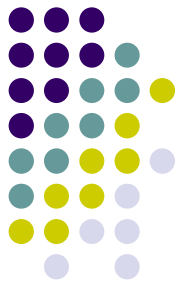


LEACH

Low Energy Adaptive Clustering Hierarchy

- Hierarchikus protokoll
- Klaszter alapú
- Dinamikus klaszter formáció

- Feltételezések:
 - Periodikus **adatgyűjtés**
 - Szomszédok adatai korreláltak
 - Node-onként változtatható adás térerősség
 - Minden node képes adni a központnak [!]
 - Minden node tud TDMA és CSMA alapon is kommunikálni [!]



LEACH



- Protokoll fázisai:
 - Setup
 - Klaszter formáció
 - Klaszter vezérlő választás
 - Steady state
 - Adattovábbítás a központ felé
- Rövid *setup*, hosszú *steady state*
- Periodikusan ismétlődik
- Mindig más vezérlő → egyenletes energiafelhasználás

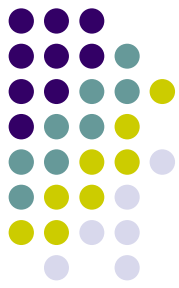
Setup

Steady State

Setup

Steady State

LEACH – setup



1. Klasztervezérlők választása:

- Node-ok önmagukat jelölik ki vezérlőnek
 - Egy node p valószínűséggel lesz vezérlő
 - Minden vezérlő egy új klasztert alkot

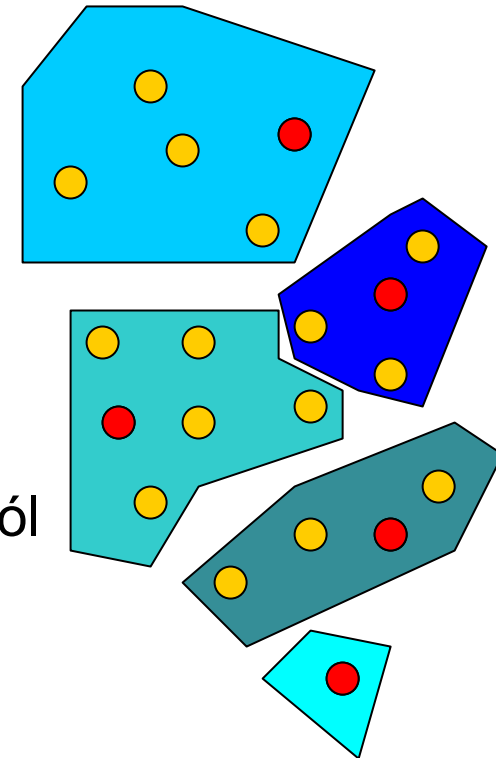
2. A vezérlők broadcast üzenetben hirdetik magukat (ill. klaszterüket)

3. A nem vezérlők csatlakoznak egy klaszterhez

- pl. jelerősség alapon
- Üzenetben értesítik a vezérlőt csatlakozásukról
- Klaszterek megalakultak

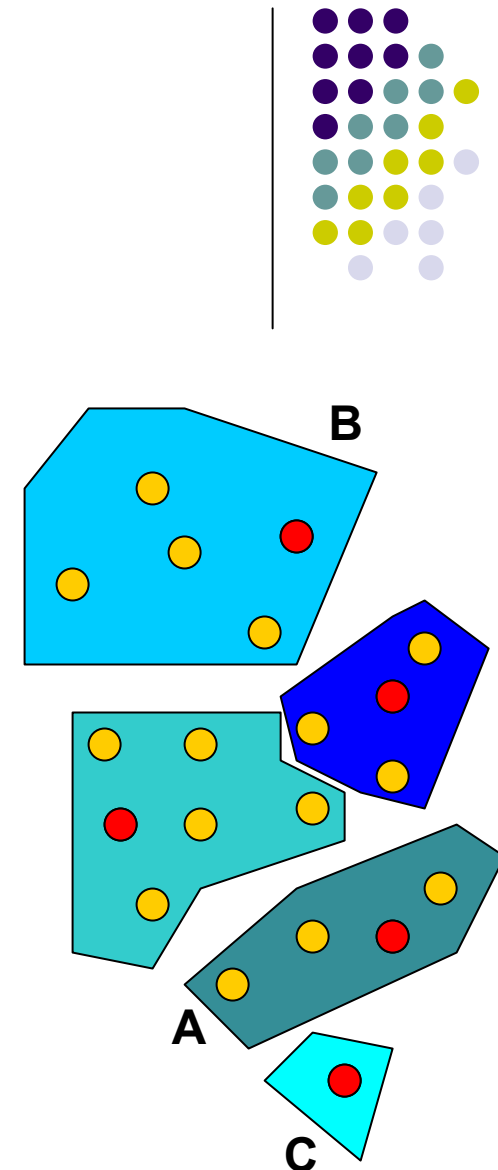
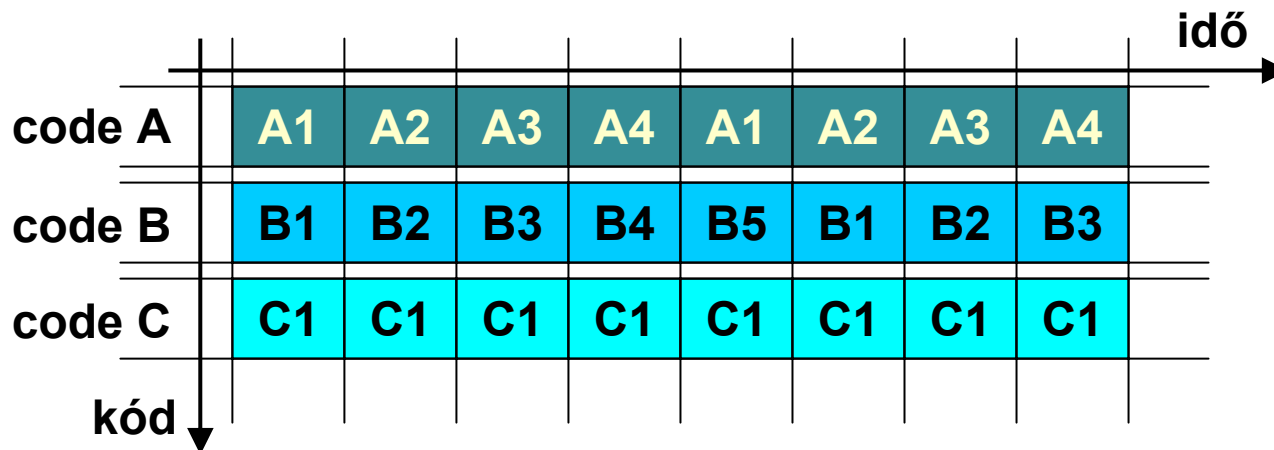
4. Vezérlő TDMA beosztást készít

5. Vezérlő a beosztást elküldi a klaszter tagjainak



LEACH – steady state

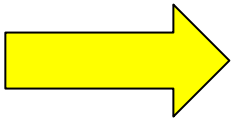
1. Node-ok periodikusan mintavételeznek
 2. Adatküldés a klaszter vezérlőnek
 3. Vezérlő továbbküldés előtt vár/aggregál
- Klaszteren belüli ütközés elkerülés:
 - TDMA
 - Klaszterek közötti ütközés elkerülés:
 - CDMA



Routing algoritmusok

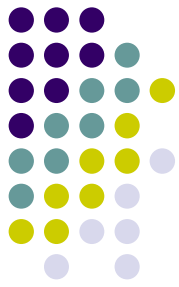


- Flat
 - Flood routing
 - Gradiens alapú
- Hierarchikus
 - LEACH
- Egyéb:
 - **GEAR**

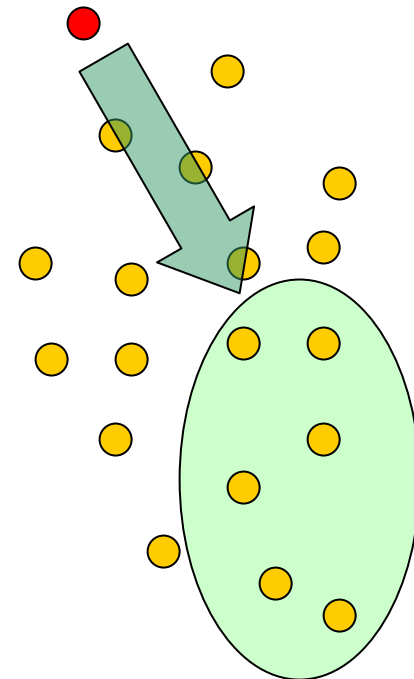


GEAR

Geographic and Energy Aware Routing



- Elhelyezkedés alapú
 - Node-ok ismerik földrajzi helyzetüket
 - Üzenet céljának helye is ismert
- Energiamegtakarítás:
 - Üzenet csak a célzott régió elé halad
- Két fázis:
 - Régió megközelítés
 - Régióon belüli terjesztés



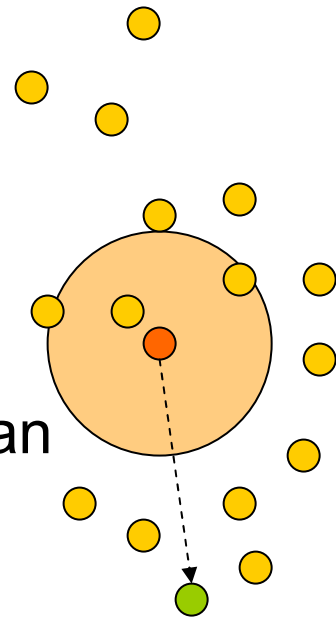
GEAR



Régió megközelítés: mértékek

Mértékek:

- Becsült költség: kombináció
 - Cél távolsága
 - Energiatartalék
- Mért költség
 - Becsült költség pontosítása lyukas hálózatban



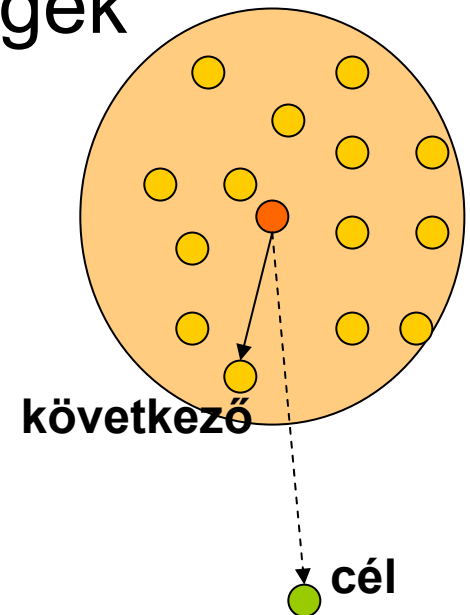
* Lyuk: a szomszédok között nincs közelebbi node a célhoz

GEAR



Régió megközelítés: üzenettovábbítás

- Üzenet annak a szomszédnak továbbítódik, akinek becsült költsége a legkisebb
- Ha lyuk van, akkor a mért költségek alapján választ következőt
- Mért költséget az üzenet célba érése után visszaterjeszti

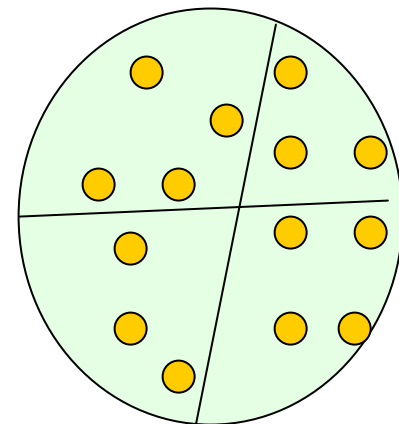


GEAR



Régióon belüli terjesztés

- A régió elérése után az üzenet szétterjed a régióban
 - Korlátozott elárasztás
 - Nem túl sűrű hálózatnál előnyös
 - Rekurzív GEAR
 - Sűrű hálózat esetén előnyösebb
 - Kisebb régiókra osztás
 - minden kis régióba üzenet küldés



Speciális megoldások



PicoRadio

- sleep mód
- külön kisteljesítményű wake-up rádió ($1\mu\text{W}$)
- kommunikáció ébredés után a normál rádión
- ébresztés célzott, csak a címzett ébred fel
- nem kell szigorú szinkronizáció az eszközök között