



Szenzorhálózatok és alkalmazásaik

S-MAC. Hálózati réteg (topológia + útvonalválasztás)

Tartalom

- Közeghozzáférés vezérlés
 - S-MAC

- Hálózati réteg
 - WSN topológia
 - Útvonalválasztás

S-MAC

- S-MAC = Sensor Medium Access Control
- Cél: Szenzorhálózatokban alkalmazható MAC protokoll kifejlesztése, amely
 - energiatakarékos,
 - önszerveződő,
 - skálázható és adaptív
 - pl: A hálózat mérete, topológiája, node-sűrűsége változik.
- Megjegyzés: A node „fairness” és az alacsony késleltetés nem elsődleges követelmény.

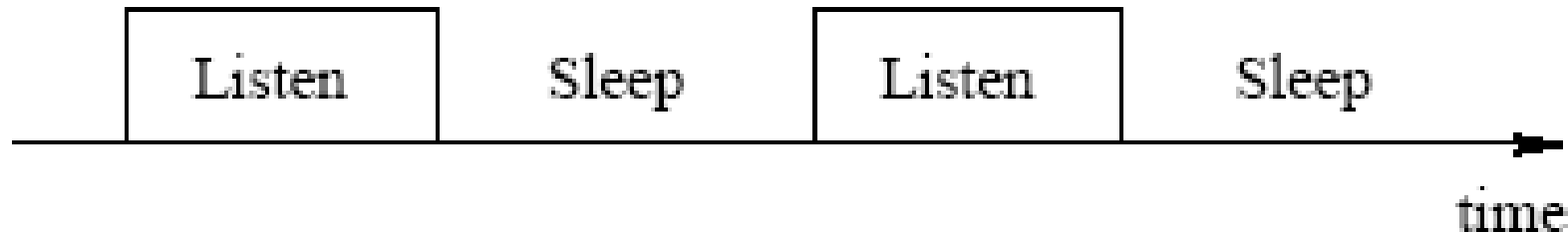
S-MAC építőkövek

A 4 fő építőelem:

- I. Periodikus figyelés és alvás.**
- II. Ütközés elkerülése.**
- III. Áthallás elkerülése.**
- IV. Üzenet továbbadás.**
 - Hosszú üzenetek hatékony továbbítására.

I. Periódikus figyelés és alvás

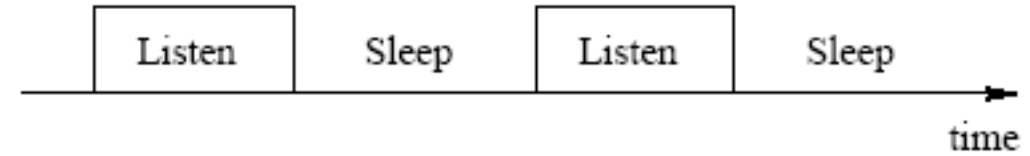
- Ha nem érzékelnek eseményt, a node-ok tétlenek. (?)
- Mivel a nyugalmas időszakokban a forgalom kicsi, nem szükséges az összes szenzornak aktívnak maradnia.
- Az S-MAC megengedi, hogy a node-ok periódikusan alvó módba kapcsoljanak.



- Alvó (sleep) módban az eszköz kikapcsolja a rádióját, és egy időzítőt (wake-up timer) állít be a felébredéséhez.

I. Periódikus figyelés és alvás (2)

- Egy teljes aktív-inaktív ciklus a keret.
- Az aktív (figyelő) időtartam fix
 - A fizikai réteg és MAC réteg paramétereit határozzák meg (pl. rádiós sávszélesség, ablakméret)
- Az alvó periódus hosszával állítható be a keret hossza, valamint az ún. „duty-cycle” érték (=aktív/inaktív arány).
 - Az aktív és alvó időtartamok a továbbiakban azonosak minden csomópont esetében (az egyszerűség kedvéért).
- Minden node megválaszthatja az aktív/inaktív ütemezését.
- → *Kérdés: De akkor hogyan szinkronizálódnak egymáshoz?*



Szinkronizáció

- Célszerű, ha szomszédos node-ok szinkronizáltak, azaz ugyanakkor alszanak ill. figyelnek. (= virtuális klaszter)
 - *Ez persze nem mindig lehetséges multi-hop esetben!*

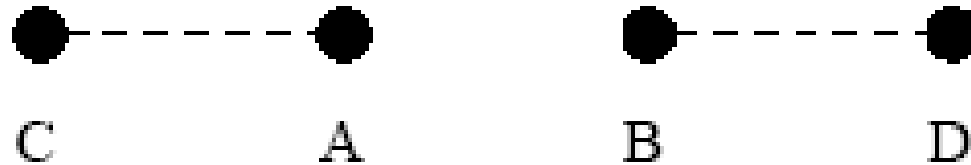


Fig. 2. Neighboring nodes A and B have different schedules. They synchronize with nodes C and D respectively.

- Szomszédos node-ok között időnként szinkronizációra van szükség az órajelek elcsúszása miatt.
- A node-ok saját ütemezésüket üzenetszórással (SYNC) juttatják el közvetlen szomszédaiknak.

Szinkronizáció (folyt.)

- Bármely két node kommunikálhat egymással akkor is, ha ütemezésük különböző:
 - Pl. ha „A” kommunikálni akar „B”-vel, megvárja, míg „B” aktív állapotba kerül.
 - Ha egyszerre többen is kommunikálni akarnak „B”-vel, versenyezniük kell a csatornáért.
- Versengés a csatornáért:
 - Az eljárás azonos az IEEE 802.11-ben használt RTS-CTS megoldással.
 - Az a node kapja a CTS csomagot „B”-től, aki elsőként küldte az RTS csomagját.
- Az adatkapcsolat létrejötte után, annak végéig a node-ok nem követik az addigi ütemezésüket. (Azaz nem mennek aludni.)

Szinkronizáció (folyt.)

Előny:

- Peer-to-peer topológia.
- Nincs szükség klaszterképzésre és koordinációra a klasztervezérlők között.

Hátrány:

- A node-ok periódikus elalvása miatt a késleltetés nagy is lehet.
- Multi-hop kommunikációnál a késleltetés arányos a node-ok számával.

Ütemezés választása és karbantartása

- Minden node karban tart egy ütemezési táblázatot a szomszédairól.
- Az ütemezés és az ütemezési tábla felállításához minden node a következő algoritmust használja:
 1. Egy bizonyos ideig figyeli a csatornát. Ha nem hall ütemezési információt egyetlen állomástól sem, véletlenszerűen választ egy időt az alváshoz, és közli ezt egy SYNC üzenetben a szomszédjaival: SYNC üzenet: „ t másodperc múlva alszom”
(Ebben az esetben az állomás lesz a „szinkronizáló”.)
 2. Ha a csatorna figyelése közben meghallja egy szomszédja ütemezését, akkor:
 - ugyanazt az értéket állítja be magának is. („követő” állomás)
 - **Véletlen t_d ideig vár**, majd elküldi az ütemezését a szomszédainak.
SYNC üzenet: „ $t-t_d$ másodperc múlva alszom”

Ütemezés választása és karbantartása

3. Ha egy állomás vesz egy sajátjától eltérő ütemezésű SYNC csomagot, akkor...
 1. ha az az egyetlen szomszédja, akkor átveszi azt és törli a sajátját.
 2. ha több szomszédja is van, akkor átveszi azt is, azaz felébreszti magát akkor is.



Fig. 2. Neighboring nodes A and B have different schedules. They synchronize with nodes C and D respectively.

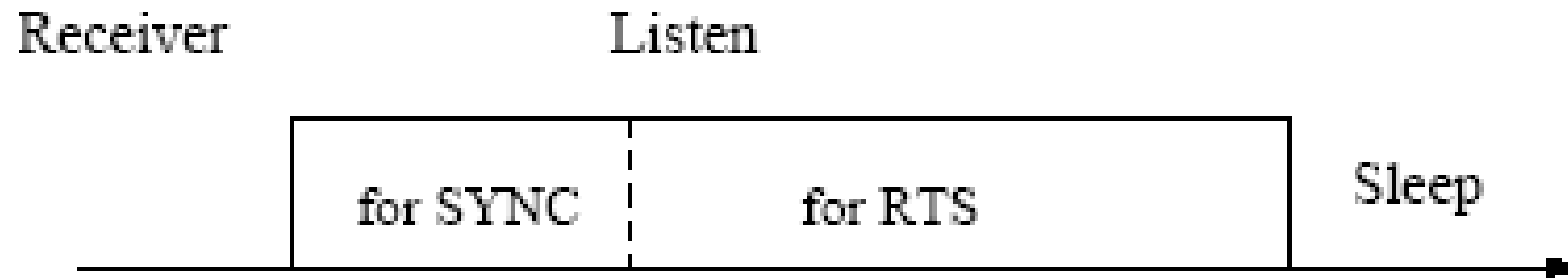
- Hátrány: A virtuális klaszter határ csomópontjai (A és B) kevesebbet alszanak.

Ütemezés választása és karbantartása

- Az órajelek elcsúszása miatt az állomásoknak időről időre frissíteniük kell az időzítési információikat. (pl. percenként)
 - SYNC: node ID + hátralévő idő a következő alvásig
- Egy állomás akkor is küld időnként SYNC csomagot, ha nincsenek követői. (?)
 - Ez lehetővé teszi az esetleg újonnan érkező állomások szinkronizálódását is.

Adatküldés és szinkronizáció

- Hogy egy állomás SYNC csomagok mellett adatcsomagokat is fogadni tudjon, az aktív intervalluma két részre van osztva:



- Minden csomag adása előtt az állomás egy véletlen ideig figyel a csatornát, hogy az szabad-e. (**vivőérzékelés – carrier sense**)

Adatküldés és szinkronizáció

- Egy állomás a vevő aktív idejében küldhet egy SYNC és/vagy egy adatcsomagot:

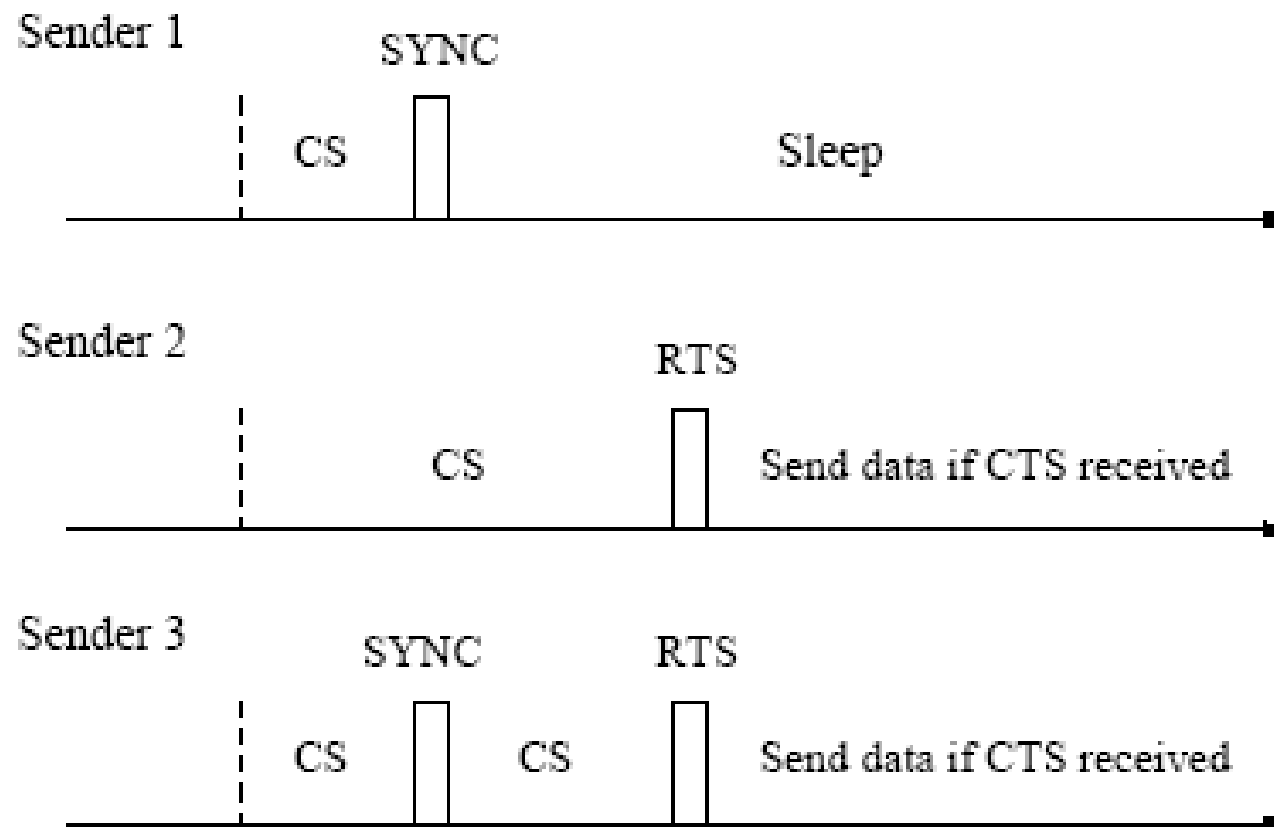


Fig. 3. Timing relationship between a receiver and different senders. CS stands for carrier sense.

S-MAC építőkövek

A 4 fő építőelem:

- I. Periodikus figyelés és alvás.
- II. Ütközés elkerülése.**
- III. Áthallás elkerülése.
- IV. Üzenet továbbadás.
 - Hosszú üzenetek hatékony továbbítására.



II. Ütközés elkerülés

- Az ütközések elkerülése a MAC protokoll egyik alapvető feladata.
- S-MAC eljárásai az ütközés elkerülésére:
 - Az IEEE 802.11 vivőérzékeléses eljárását alkalmazza a fizikai rétegben.
 - Az RTS-CTS mechanizmust alkalmazza a rejtett terminál problémájának elkerülésére.
 - Alkalmaz egy virtuális vivőérzékeléses eljárást is.
- **Virtuális vivőérzékelés:**
 - Ha egy állomás veszi egy nem neki címzett csomag fejrészét, akkor abból ki tudja olvasni a csomag hosszát.
 - Az adott időtartamig a csatornát foglaltnak tekinti anélkül, hogy közben megvizsgálná azt.
- Üzenetszórás (SYNC) esetén nincs RTS-CTS (?)
- Adatkommunikáció: RTS/CTS/DATA/ACK

S-MAC építőkövek

A 4 fő építőelem:

- I. Periodikus figyelés és alvás.
- II. Ütközés elkerülése.
- III. Áthallás elkerülése.**
- IV. Üzenet továbbadás.
 - Hosszú üzenetek hatékony továbbítására.

III. Áthallás elkerülése

- Az IEEE 802.11 szerint minden állomás figyeli a szomszédai adásait, hogy hatékony virtuális vivőérzékelést végezhesen.
 - A sok „felesleges” csomag vétele energiapazarló, különösen nagy node-sűrűség és nagy forgalom esetében.
- S-MAC az áthallások csökkentésére egy állomás aludni megy, ha hall egy RTS vagy CTS csomagot.
 - Így nem kell feleslegesen vennie egy (esetlegesen) hosszú adatcsomagot és az ACK csomagot.
- Pl. „A” ad „B”-nek. Kérdés: *Kinek kell aludni?*

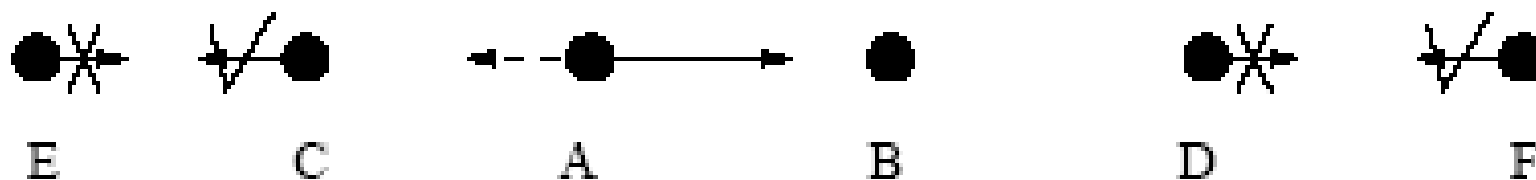


Fig. 4. Who should sleep when node A is transmitting to B?

- Válasz: Mind az adó, mind a vevő közvetlen szomszédjának. (Interferencia a vevőnél lép fel!)

S-MAC építőkövek

A 4 fő építőelem:

- I. Periodikus figyelés és alvás.
- II. Ütközés elkerülése.
- III. Áthallás elkerülése.
- IV. Üzenet továbbadás.**
 - Hosszú üzenetek hatékony továbbítására.

IV. Üzenet továbbadás

- Kérdés: Hogyan lehet egy hosszú adatcsomagot továbbítani energiatakarékosan és lehetőleg kis késleltetéssel? (?)
- Két lehetséges megoldás:
 1. Egyben továbbítjuk.
Hátrány: Ha néhány bit meghibásodik, az egész csomagot újra kell küldeni.
 2. Szegmensekre bontva továbbítjuk.
Hátrány: Nagy „overhead” a kontrollcsomagok miatt. (Minden szegmenshez szükség van egy RTS/CTS/ACK csomagra.)
- Ötlet: S-MAC esetén az üzenetet kis szegmensekre bontjuk, majd a szegmenseket börsztben küldjük el.
 - Az összes szegmenshez egyetlen RTS-CTS pár tartozik.
 - Az RTS csomag az egész börszt küldéséhez szükséges időt jelzi.

S-MAC: üzenet továbbítás

- Minden szegmens továbbítása után az adó vár egy ACK csomagot a vevőtől. Ha nem kapja meg, újraküldi a szegmenst.
- Minden szegmens és ACK tartalmazza a teljes börszt továbbításához szükséges időt.
- A gyakori nyugták kiküszöbölik a rejtett terminál problémáját.
 - Ha egy állomás egy börszt továbbítása közben ébred fel, és nincs az adó rádiósugarán belül, akkor üresnek találja a csatornát és adásával interferenciát okoz.

S-MAC és 802.11 szegmentálása

- Megjegyzés: A 802.11 szintén támogatja a szegmentálást.
 - Az RTS és CTS csomagok csak az első szegmens idejére foglalja le a csatornát. A további foglalást a következő csomagok és nyugták tartalmazzák.
 - Ez nem teszi lehetővé az „alvást”.
- A 802.11 szegmentálása a fairness biztosítására hivatott.
 - Ha egy állomás nem kap nyugtát, abba kell hagynia az adást és újra versenyeznie kell a csatornáért.
- S-MAC esetében a fairness másodlagos kérdés.
- Határt kell azonban szabni az újraküldéseknek és a foglalási ablak állandó kitolásának.
 - Pl. A fogadó állomás működésképtelenné vált.

Hálózati réteg

WSN topológia. Útvonalválasztás.

Tartalom

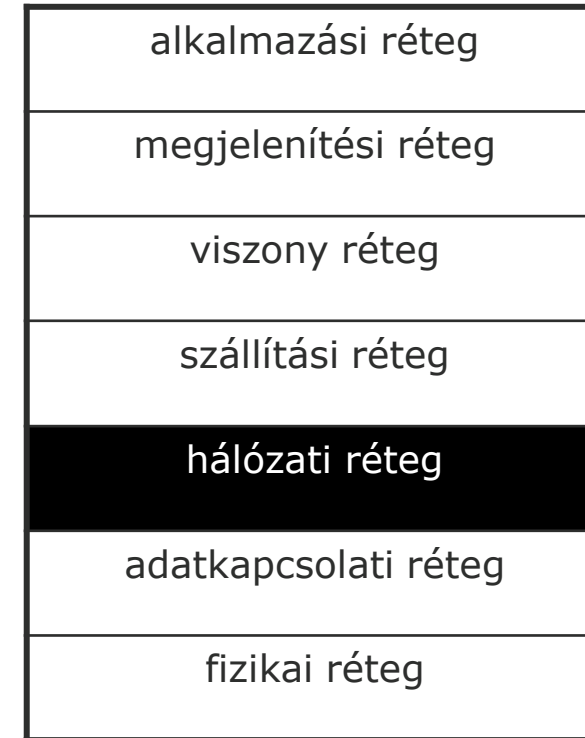
- Hálózati réteg
 - WSN topológia
 - Útvonalválasztás (jövő hét)



Hálózati réteg

- A hálózati réteg feladatai:
 - Vég-vég hálózati kapcsolat létrehozása és karbantartása bármely két hálózati csomópont között.
 - Útvonalválasztás.
 - Forgalomszabályozás, QoS biztosítás.
 - ...

ISO OSI



Hálózati réteg tervezési kérdései

- A szenzorhálózatok hálózati rétegének két, egymással szorosan összefüggő fontos területe:
 - A hálózat struktúrája (**topológia**)
 - A hálózaton belül az üzenetek továbbítására használt algoritmus (**routing**)

Topológia

WSN topológiája

- Mivel tipikusan a szenzorhálózatok önszerveződő ad-hoc hálózatok, a hálózati struktúra „nem tervezhető”.
- A fizikai összeköttetések (linkek) véletlenszerűen jönnek létre, véletlen (fizikai) topológiát alkotva.
- A logikai topológia kialakítása azonban fontos!
 - Különös tekintettel a skálázhatóságra.

Elosztott (flat) struktúra

- Nincs kialakított logiai struktúra, minden node részt kell vegyen a hálózat vezérlésében.
- Mivel a node-ok csak a szomszédaikról rendelkeznek közvetlen információval, időről időre **terjeszteniük kell ezt az információt a hálózatban.**
 - Pl. periodikusan minden állomás szétküldi az általa tárolt routing táblát.
- Hátrány: skálázhatatlan! (pl. több ezer node esetén...)

Klaszterezés

- A hálózatot ún. **klaszterekre** (cluster) bontjuk úgy, hogy minden node legalább egy klaszterben szerepeljen.
- Minden klaszternek vagy egy **vezérlője** (cluster head), amely állomás vezérli a klaszteren belüli node-okat.
- **Átjáró** (gateway) állomások biztosítják a klaszterek közötti kommunikációt.
- Jobban skálázható.
- Probléma: Minden klaszter ismeri a szomszéd klasztereket, de honnan értesül a távolabbi klaszterekről?
 - Ugyanazon probléma eggyel magasabb hierarchiaszintre került!
- Megoldás: A klaszter vezérlők hierarchikus fába szervezése.

Klaszterezés (folyt.)

- A klaszterek optimális kialakítása nem egyszerű
 - „NP-nehéz”, de $O(n^2)$ heurisztikus klaszterképző eljárások is léteznek
- További probléma, hogy a klasztervezérlők (sokkal!) nagyobb terhelésnek vannak kitéve.
 - Megoldás: A vezérlő szerepét időnként cserélik.

Szenzorhálózatok sajátosságai

- A hálózati csomópontok (szenzorok) nagy száma:
 - **Nem lehetséges a globális címezés**, mert nem menedzselhető önálló ID minden node számára.
 - Következmény: A „hagyományos” IP alapú protokollok nem(?) alkalmazhatóak.
- A szenzorok ad-hoc módon telepítettek:
 - A hálózatnak **önszerveződőnek** kell lennie.
 - Meg kell birkóznia az esetleges **véletlenszerű** node-eloszlással.
 - Biztosítani kell a **felügyelet nélküli** működést.

Szenzorhálózatok sajátosságai

- Tipikusan a szenzorok stacionáriusak a telepítés után.
 - Ellentétben a mobil ad-hoc hálózatokkal, ahol az állomások szabadon mozoghatnak.
 - Alkalmazástól függően lehet néhány mobil állomás is (tipikusan alacsony mobilitással).
- Tipikusan több forrástól (szenzortól) áramlik az információ egy nyelő (bázisállomás) felé.
 - De lehet akár multicast, vagy peer-to-peer forgalom is!

Szenzorhálózatok sajátosságai

- A szenzorok erősen energia-, számítási- és tárolási-kapacitás korlátozottak.
 - Hatékony **erőforrás-menedzsment** szükséges.
- A szenzorhálózatok alkalmazás specifikusak.
 - A tervezési követelmények alkalmazásról alkalmazásra változnak.
(Pl. precíziós felügyeleti rendszerek kontra periodikus időjárás-monitorozás.)

Szenzorhálózatok sajátosságai

- A **helytudatos működés** fontos, tipikusan az adatgyűjtés elhelyezkedéshez kötötten történik.
 - Pl. a GPS hardver alkalmazása nem lehetséges, mert túl költséges.
 - A helymeghatározás tipikusan háromszögelésen és jelszint mérésen alapul, referenciapontok segítségével.
- A forgalmazott adatok tipikusan **redundánsak**.
 - Pl. Több szenzor érzékel és küld adatokat ugyanarról a jelenségről.
 - A redundancia kihasználható útvonalválasztáskor.
- A legtöbb szenzorhálózat **adatcentrikus**.
 - Az adatokra bizonyos attribútumok alapján vagyunk kíváncsiak. (Pl. Hol magasabb a hőmérséklet, mint 40 fok?)

Következik: Útvonalválasztás

- Routing tervezési kérdések
- Hálózat és routing modellezése
- Hálózati struktúra alapú protokollok
 - Elosztott (flat)
 - Hierarchikus
 - Elhelyezkedés alapú (location based)

