



Szenzorhálózatok és alkalmazásaik

Hálózati réteg (topológia + útvonalválasztás)

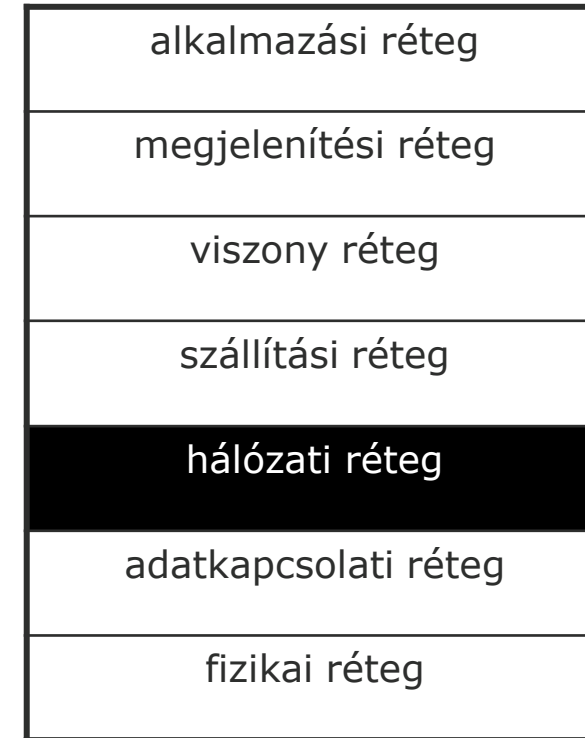
Hálózati réteg

WSN topológia. Útvonalválasztás.

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 logikai 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...)

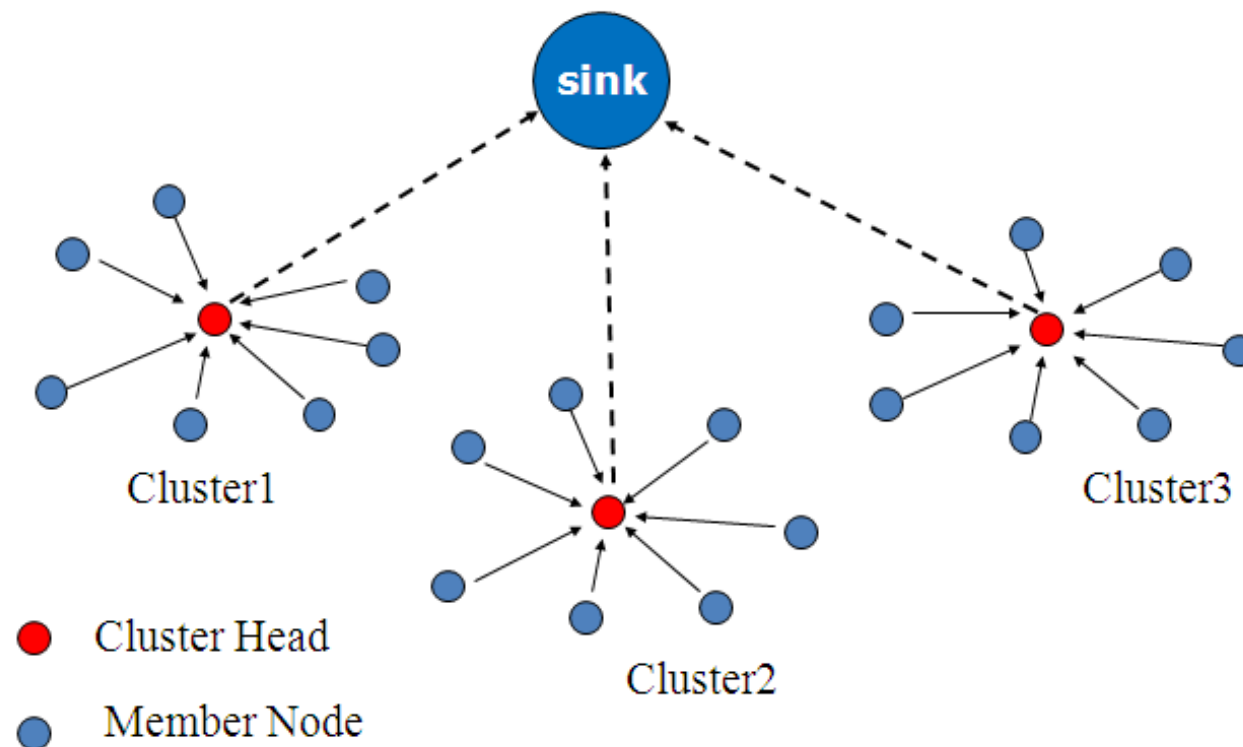
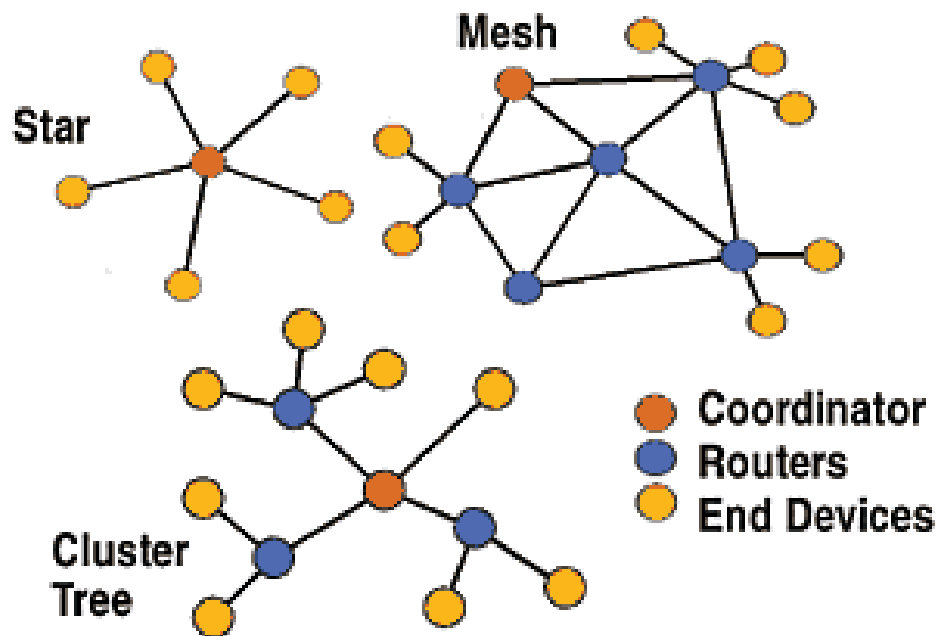
Klaszterezés

- 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.

WSN topológia – klaszterképzés



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?)

Útvonalválasztás

Tartalom

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



Routing feladata, definíciója

- Egy igény felmerülésekor a hálózatnak 3 kérdést kell megválaszolnia:
 1. Létezik-e útvonal a két csomópont között? (routing)
 2. Megengedett-e a használata? (forgalomszabályozás)
 3. Ha 1-re nemleges a válasz, mit kell tenni?
- Routing = szabályok + adatok
 - Szabályok: Döntés, hogy melyik útvonalon továbbítódik az üzenet.
 - Adatok: A döntés meghozatalához szükséges info.
- Döntés helye szerint: **centralizált** vagy **elosztott**
- **Adaptivitás**: Képes-e alkalmazkodni a hálózat állapotához.

Routing protokollok csoportosítása

- Hálózati struktúra szerint:
 - elosztott (flat)
 - hierarchikus
 - elhelyezkedés alapú
- Protokoll működése alapján:
 - többutas (multipath-based)
 - lekérdezésen alapuló (query-based)
 - megállapodáson alapuló (negotiation-based)
 - szolgálatminőségi (QoS-based)
 - koherens (coherent-based)



Routing tervezési kérdések WSN-ben

- **Energiatakarékosság** a pontosság és megbízhatóság elvesztése nélkül.
 - A **node-ok kettős szerepe**: adatküldő és útvonalirányító.
 - Egy node kiesése hibás működés vagy energiahiány következtében jelentős topológia-módosulást és új útválasztás szükségességét eredményezheti.

Routing tervezési kérdések WSN-ben

▪ Adatküldési modell

- Az információ érzékelése és az adatok továbbítása alkalmazásfüggő.

Az érzékelés és küldés lehet

▪ idővezérelt (folyamatos)

- pl. periódikus monitorozást igénylő alkalmazásoknál
- A szenzorok időről időre aktiválódnak, mérnek, majd elküldik a mért adatot.

▪ eseményvezérelt

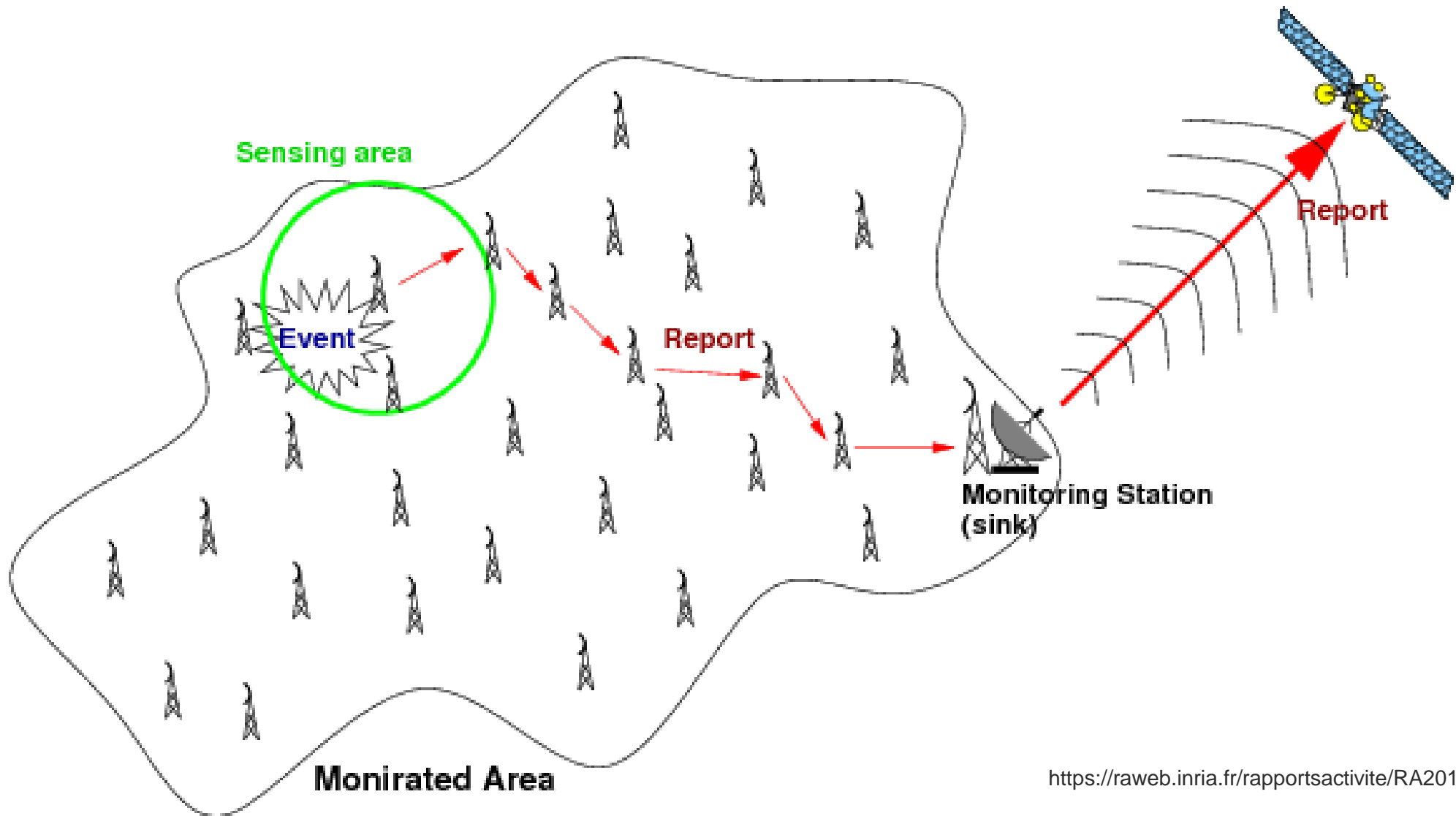
- A szenzorok reagálnak a környezet hirtelen vagy jelentős változásaira.
- Időkritikus alkalmazásoknál fontos.

▪ lekérdezés-vezérelt

- Egy „vezérlő” lekérdezésére válaszolnak a node-ok

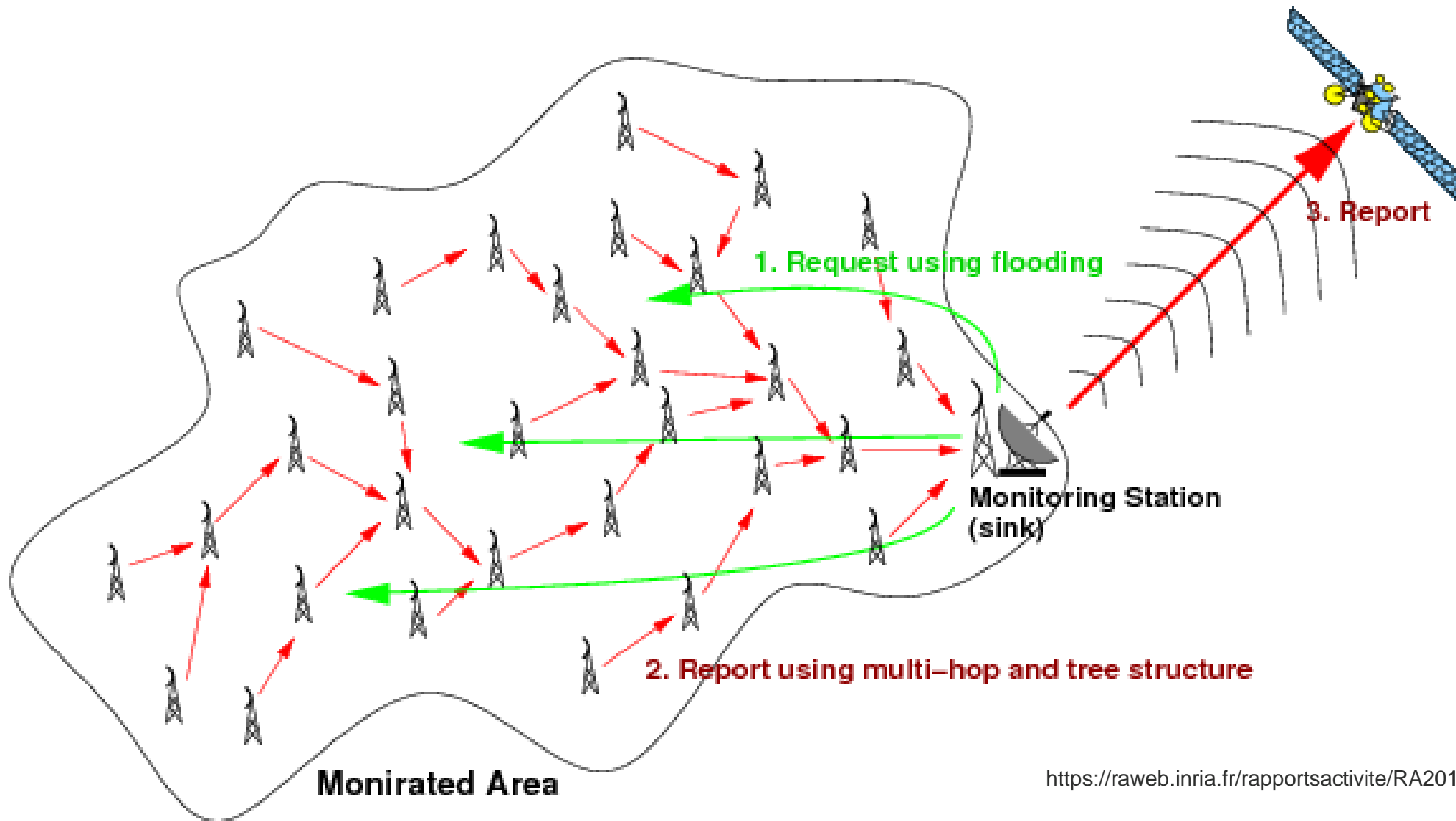
- ***Az útvonalválasztás nagyban függ az adatküldési modelltől!***

Eseményvezérelt működés (példa)



<https://raweb.inria.fr/rapportsactivite/RA2010/pops/IMG/fig1.png>

Lekérdezés vezérelt (példa)



<https://raweb.inria.fr/rapportsactivite/RA2010/pops/IMG/fig1b.png>

Routing tervezési kérdések WSN-ben

▪ Hibatűrés

- Bármelyik szenzor bizonyos okok miatt (pl. interferencia, fizikai sérülés, energiahiány) kieshet a hálózathoz.
- Egy node kiesése nem szabad, hogy az egész alkalmazás működését veszélyeztesse.
- A MAC és routing protokolloknak képesnek kell lenniük több node kiesését is megoldani.
- **Többszintű redundancia** beépítése lehet szükséges robusztus szenzorhálózatok tervezésénél.

▪ Skálázhatóság

- A szenzorok száma egy hálózatban több száz, esetleg több ezer(!) is lehet.
- Pl. Egy állomásban implementált routing tábla messze nem tartalmazhatja az összes útvonalat minden node-hoz.

Routing tervezési kérdések WSN-ben

- Hálózati **dinamika**
 - Legtöbb esetben a node-ok **statikusak**, nem mozognak.
 - **Mozgó állomások** (pl. mobil BS) estén a mobilitás-vezérlés az útvonalválasztásnál nagy kihívás.
 - Pl. a hálózat topológiája az időben változik, ...
 - Az érzékelt jelenség maga is lehet statikus vagy dinamikus. (pl. hőmérséklet kontra célkövetés)

Routing tervezési kérdések WSN-ben

▪ Lefedettség

- Minden szenzor csak a „közeli” környezetét tudja kellő pontossággal figyelni.
- A teljes terület lefedettsége (ill. a lefedettség megtartása az idő előrehaladtával) fontos szempont a tervezésnél.

▪ Adatösszegzés (aggregation)

- A szenzorok nagyfokú redundanciával generálnak és küldenek adatokat. (Pl. hőmérsékletmérés közeli pontokban)
- A lényeges információ kiemelésével az átvitt adatmennyiség nagyban csökkenthető.
 - Pl. minimum, maximum, átlag
- Adat összegzéssel energiatakarékosság és nagyobb hatékonyság érhető el.
- Jelfeldolgozási módszerek is bevethetők a tömörítésnél.

Routing tervezési kérdések WSN-ben

- **Szolgáltatásminőség (QoS – Quality of Service)**
 - Pl. Adott esetben a mért adatokat egy időkorláton belül el kell küldeni, különben érdektelenné válik.
 - Időkritikus alkalmazásoknál a késleltetést korlátok közé kell szorítani.
 - Legtöbb alkalmazás esetében az energiatakarékosság fontosabb, mint a rövid válaszidő.
 - Pl. Ötlet: az energia fogytával az adat minősége csökkenthető.
- **Energiatudatos útvonalválasztás a szenzorhálózatokban kulcskérdés!**

Tartalom

- Hálózati réteg
 - WSN topológia
 - Útvonalválasztás
 - Routing tervezési kérdések
 - **Hálózat és routing modellezése**
 - Útvonalválasztó protokollok
 - Elosztott (flat)
 - Hierarchikus
 - Elhelyezkedés alapú (location based)

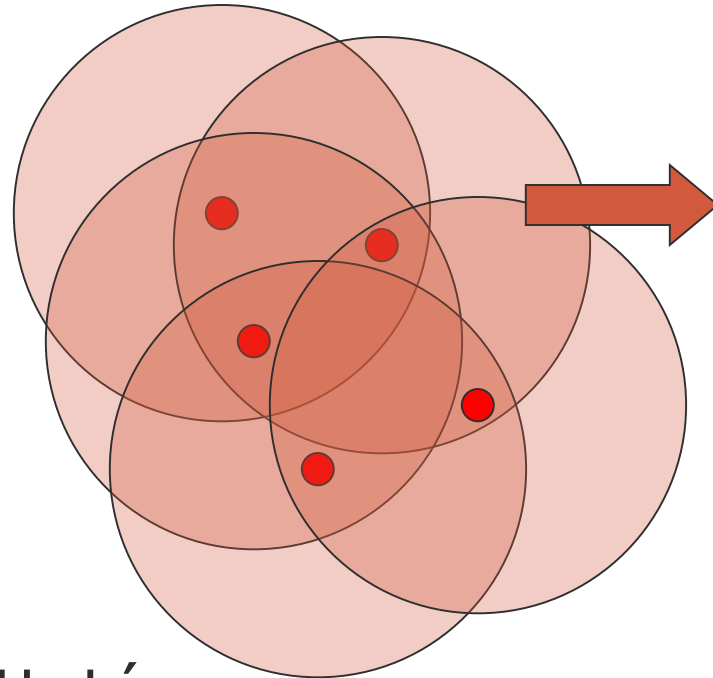


Hálózat és routing modellezése

Gráfok, algoritmusok

Hálózat és routing modellezése

- Matematikai modell:
 - hálózat -> gráf
 - routing algoritmusok -> gráfelméleti algoritmusok



hálózat
- szenzorok
- rádiós összeköttetés

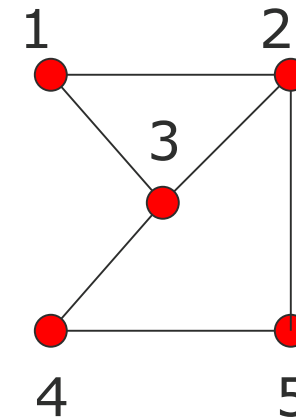
gráf
- csomópontok
- élek

Hálózat és routing modellezése

- Hálózat gráfja
 - Az élek irányítottak(?) vagy irányítatlanok lehetnek
 - pl. Egy állomás nem vesz részt üzenet továbbításában
 - A távolság(?) fogalma két csomópont között értelmezhető
 - pl. Meghatározható a link „költsége” vagy „minősége”
- Jelölések: $G(N, A)$, ahol a G gráfban N a csomópontok halmaza, A pedig az élek halmaza.

$$N = \{1, 2, 3, 4, 5\}$$

$$A = \{(1, 2), (1, 3), (2, 3), (3, 4), (2, 4), (2, 5)\}$$



Hálózat gráfja, fogalmak

- Út: $(n_1, n_2, \dots, n_m), n_i \in N, (n_i, n_{i+1}) \in A$
- Útvonal: Út, melyben egyetlen él sem ismétlődik.
- A gráf összefüggő, ha minden csomópontból létezik útvonal minden más csomópontba.
- Hurok: Út, melyre igaz, hogy $n_1 = n_m$
- Fa: Összefüggő gráf, amelyben nincs hurok.
- Feszítő fa (teljes fa): Fa, amely minden csomópontot tartalmaz.

Példa: Legrövidebb út keresése

- Feladat: Legrövidebb út keresése a $G(N, A)$ gráfban az s (forrás) és t (nyelő) között.
- Az élekhez súlyokat rendelünk:
$$d_{i,j} \in R \Leftrightarrow (i, j) \in A$$
$$d_{i,j} = \infty \Leftrightarrow (i, j) \notin A$$
$$d_{i,i} = 0$$
- Keresendő az az L útvonal s és t között, melynek súlyösszege minimális (shortest path).
 - Egy lehetséges megoldás: Dijkstra algoritmus

Példa: Dijkstra algoritmus

Jelölés:

- D_i = az i . csomópont iterált távolsága s -től ($s=„1”$)
- P = feldolgozott csomópontok halmaza

Algoritmus:

1. inicializálás: $P = \{1\}$; $D_j = d_{1j}$ minden j -re
2. iteráció: keressük i -t, melyre $D_i = \min_{\forall j \notin P} D_j$
3. legyen: $P = P \cup \{i\}$
$$D_j = \min \{D_j, d_{ij} + D_i\}$$
4. leállás: Ha $N = P$, STOP
else GOTO 2.

