

Szenzorhálózatok

LEACH - esettanulmány (2011.11.04)

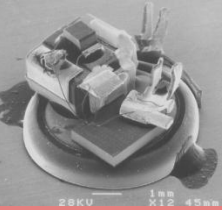
Vidács Attila

Távközlési és Médiainformatikai Tanszék

I.B.325, T:19-25, vidacs@tmit.bme.hu

LEACH cikkek

- W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, „**Energy-efficient Communication Protocol for Wireless Microsensor Networks**”, *Proc. of 33rd Hawaii International Conference on System Sciences*, 2000.
- W. B. Heinzelman, „**Application-Specific Protocol Architectures for Wireless Networks**”, PhD Dissertation, Massachusetts Institute of technology, June 2000.

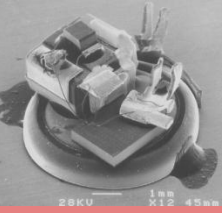


LEACH főbb jellemzői

LEACH = Low-Energy Adaptive Clustering Hierarchy

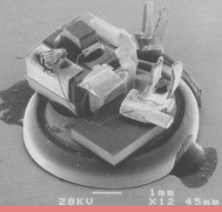
A LEACH protokoll főbb jellemzői:

- Energiatakarékos
- Önszerveződő
- Adaptív
- Klaszter alapú, elosztott koordinációval
- Klasztervezérlők véletlenszerű cserélgetése
- Skálázható, robosztus
- Adattömörítés a globális kommunikáció csökkentésére.

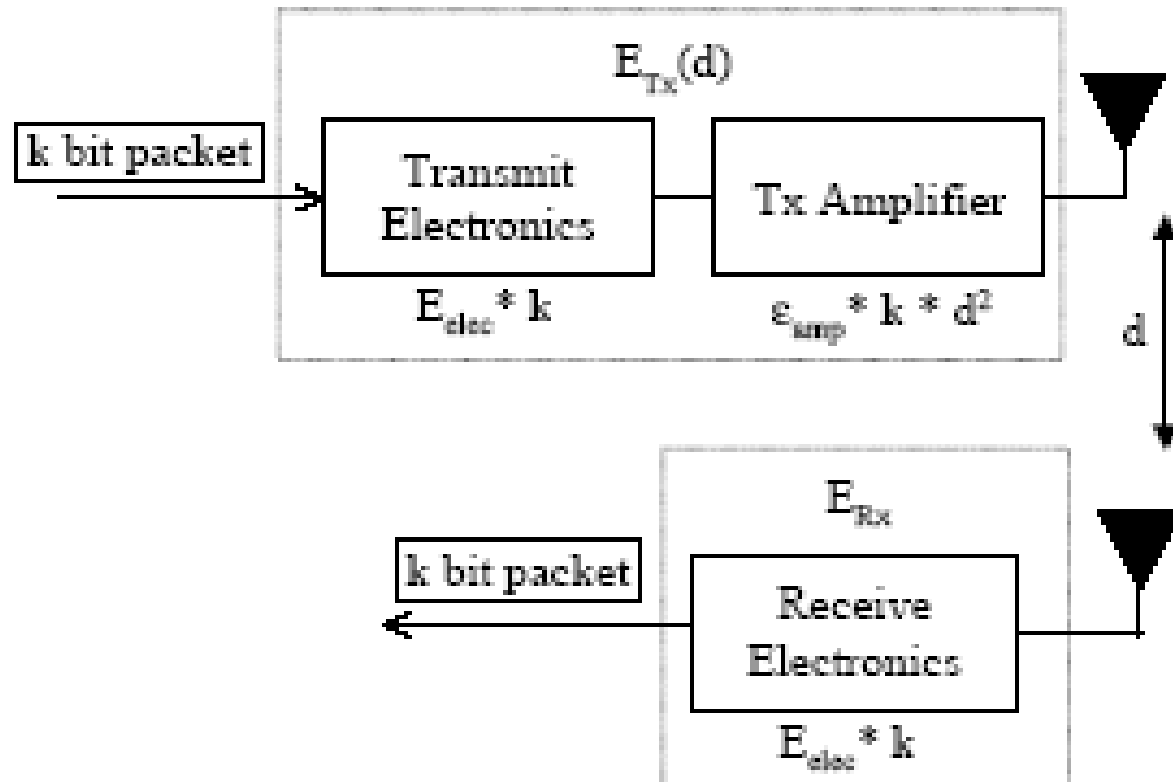


Feltevés

- A bázisállomás (BS) fix.
- A node-ok azonos felépítésűek, korlátozott energiával.
- Minden node képes a BS-sel kommunikálni ha szükséges.
- Idővezérelt működés: A szenzorok mindig mérnek, így mindig van továbbítandó adat.
- A node-ok képesek adóteljesítmény-szabályozásra.



Rádiómodell



| Operation | Energy Dissipated |
|---|---------------------------|
| Transmitter Electronics ($E_{Tx-elec}$) | 50 nJ/bit |
| Receiver Electronics ($E_{Rx-elec}$) | |
| $(E_{Tx-elec} = E_{Rx-elec} = E_{elec})$ | |
| Transmit Amplifier (ϵ_{amp}) | 100 pJ/bit/m ² |

Rádiómodell

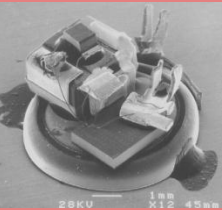
- k bit küldéséhez szükséges energia d távolságra:

$$E_{Tx}(k, d) = E_{elec} \cdot k + \varepsilon_{amp} \cdot k \cdot d^2$$

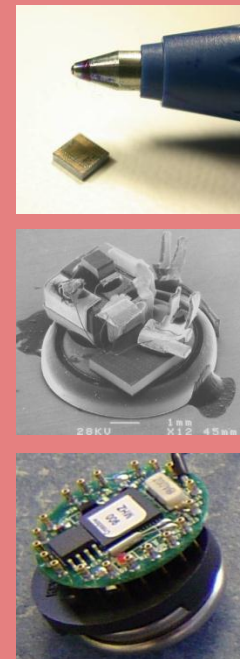
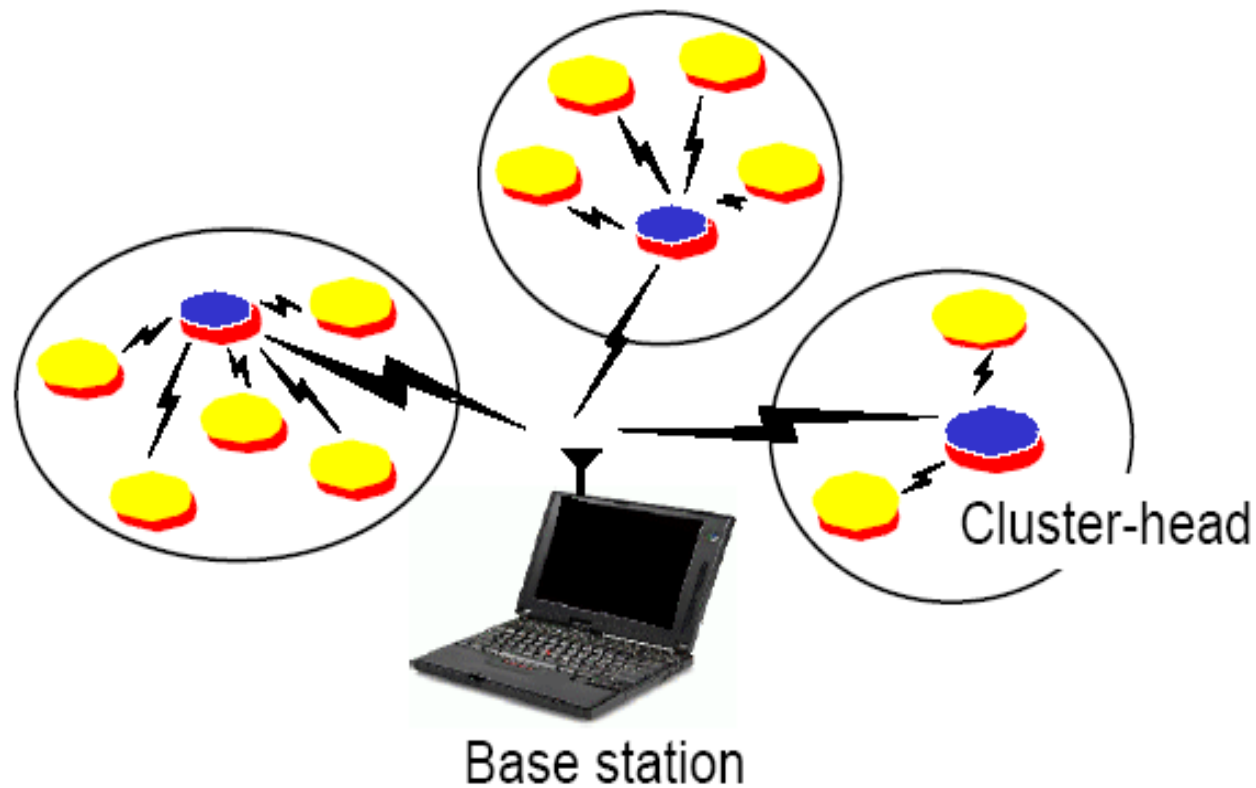
- k bit vételéhez szükséges energia:

$$E_{Rx}(k, d) = E_{elec} \cdot k$$

- A rádiócsatornák szimmetrikusak, azaz A-ból B-be ugyanannyi energiába kerül az adás, mint B-ből A-ba.

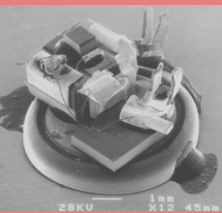


Klaszter alapú működés



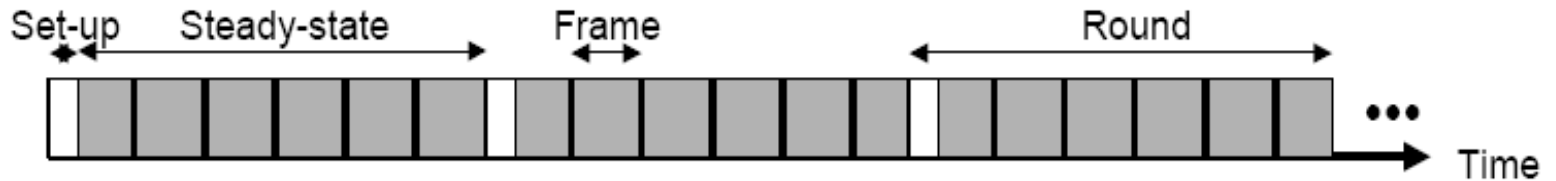
Klaszter alapú működés

- A node-ok önszerveződően klaszterekbe rendeződnek.
- Minden klaszteren belül választanak egy klasztervezérlőt (CH – Cluster Head)
- Minden node a klasztervezérlőnek küldi az adatait.
- A klasztervezérlő a begyűjtött adatokat előfeldolgozva (tömörítve) továbbítja a BS-nek.
- A klasztervezérlő terhelése sokkal magasabb, mint a többi node terhelése -> időnként cserélni kell a vezérlő szerepet



Protokoll működése

- A protokoll működése „körökre” osztható:



- A protokoll két fázisa:

- **Inicializálás** (setup phase)

- Klaszterek szervezése
- Klasztervezérlők kijelölése

- **Állandósult állapot** (steady phase)

- Az adatforgalom vezérlése

- Körönként periódikusan a hálózat visszatér az inicializáló állapotba.

Inicializálás

1. lépés: A klasztervezérlők kiválasztása.

□ **Cél:**

- Minden node kb. azonos ideig legyen klasztervezérlő.
- A vezérlők egyenletesen legyenek elosztva a hálózatban.

□ **Megoldás:** Az i . node a t időpontban kezdődő $(r+1)$ -edik kör elején $P_i(t)$ valószínűséggel lesz klasztervezérlő.

□ Egy körben a vezérlők átlagos száma k , azaz:

$$E[\#CH] = \sum_{i=1}^N P_i(t) = k$$

ahol N a hálózatban lévő node-ok száma



Inicializálás (folyt)

- Azt szeretnénk, ha N/k kör alatt minden node egyszer lenne klasztervezérlő. Ehhez legyen:

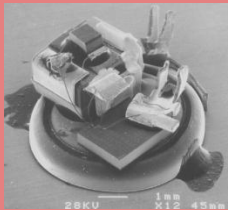
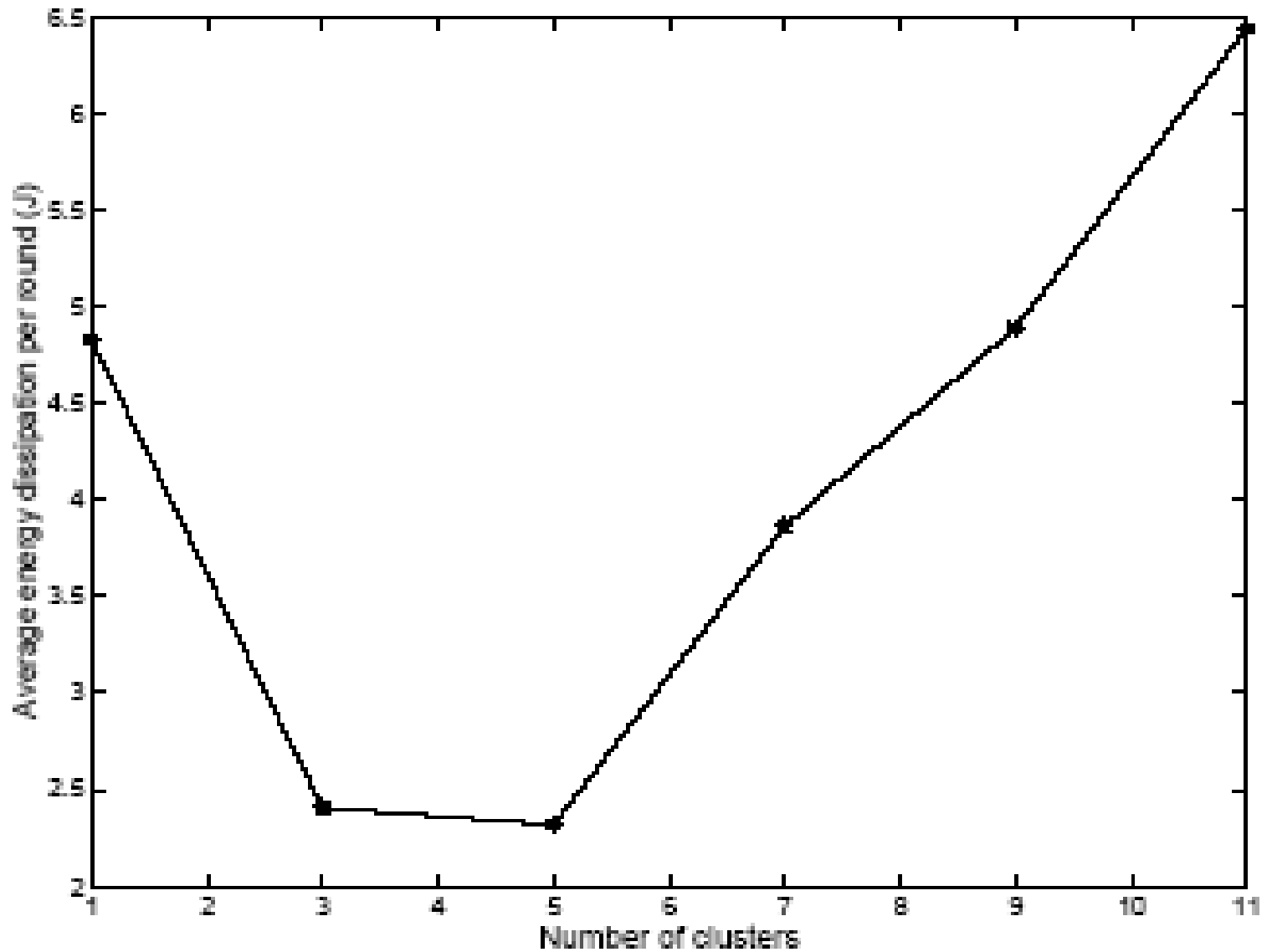
$$P_i(t) = \frac{k}{N - k \cdot (r \bmod (N/k))}$$

ha az i . node nem volt vezérlő az elmúlt $(r \bmod (N/k))$ körben.

- Az optimális k érték analitikusan meghatározható.
 - Túl kis k esetén messze lesznek a klasztervezérlők.
 - Túl nagy k esetén sok lesz a direkt kommunikáció a BS-sel.



Inicializálás (folyt)



Inicializálás (folyt)

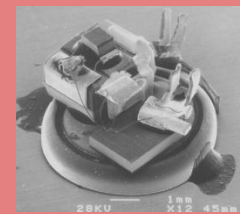
- A P_i értékek számításának hátránya, hogy nem veszi figyelembe a node-ok (esetlegesen) eltérő ütemű lemerülését.

- Egy lehetséges adaptív alternatíva:

$$P_i(t) = \frac{E_i(t)}{\sum_{j=1}^N E_j(t)} k$$

ahol $E_i(t)$ az i . node aktuális energiaszintje.

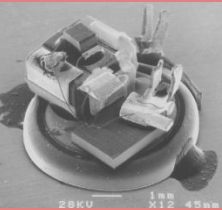
- Előny: A node-ok élettartama kiegyenlítettebb lesz.
- Hátrány: Minden node-nak ismernie kell az egész hálózat aktuális energiatartalékát.
 - Ötlet: a hálózat összenergiája az adott klaszter energiaszintjéből becsülhető.



Inicializálás (folyt)

2. lépés: Klasztervezérlők (CH) hirdetése.

- Nem-perzisztens CSMA MAC protokoll használata.
- Minden CH egy ADV (advertisement) csomagot küld üzenetszórással az összes node-nak, mert:
 1. így - a CSMA miatt - nincs rejtett terminál probléma;
 2. biztosítja, hogy senki ne maradjon vezérlő nélkül véletlenül.
- Az ADV csomag tartalmazza a node ID-jét, valamint a „bejelentés” üzenetet.



Inicializálás (folyt)

3. lépés: Klasztervezérlő választás

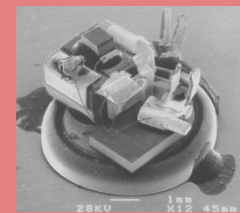
- Minden nem CH node a hirdetések alapján kiválasztja, hogy melyik CH-hoz akar tartozni (pl. vett jel erőssége alapján).
- Egy csatlakozás-kérés (Join-REQ) üzenetet küld a választott vezérlőnek, nem-perzisztens CSMA-t használva.
- Join-REQ tartalma: node ID; CH ID, üzenet típusa
- A Join-REQ üzenethez a node-ok nagy energiát használnak!
 - Így elkerülik a rejtett terminál problémáját.
 - Nincs RTS-CTS jelzésre szükség.



Inicializálás (folyt)

4. lépés: TDMA ütemezés kiosztása.

- Minden CH a hozzá tartozó klaszterben TDMA időszeleteket rendel a node-okhoz.
 - Előny: A node-ok a saját időrésükön kívül „sleep” módba kapcsolva energiát spórolnak meg.
- Az időzítés információkat tudatja mindenkivel.
- A hálózat átlép *állandósult állapotba* egy előre meghatározott fix ideig.



Inicializálás - folyamatábra

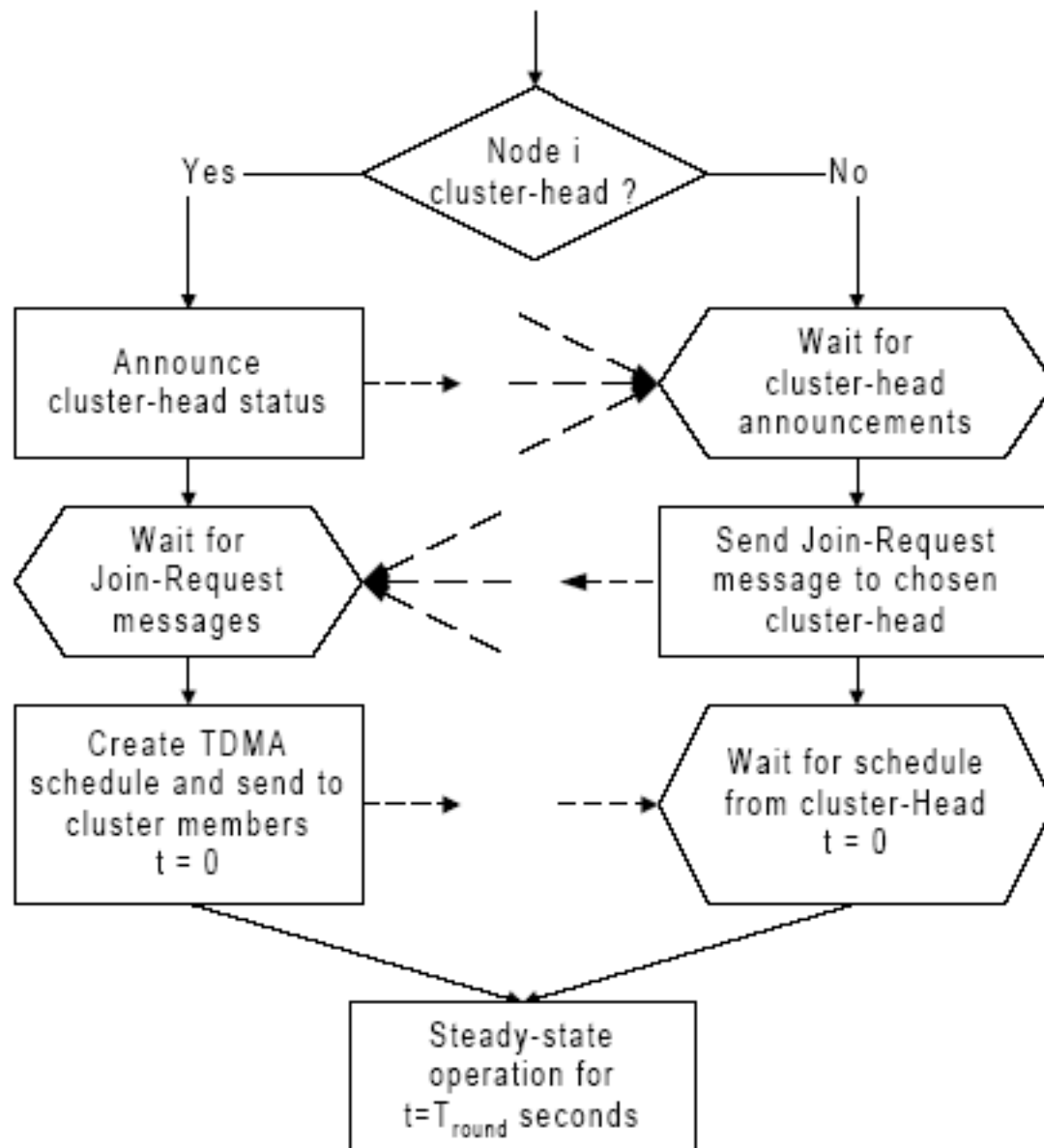
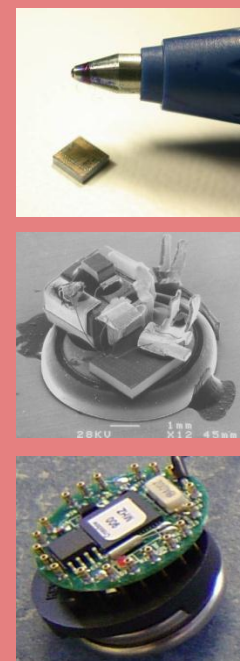
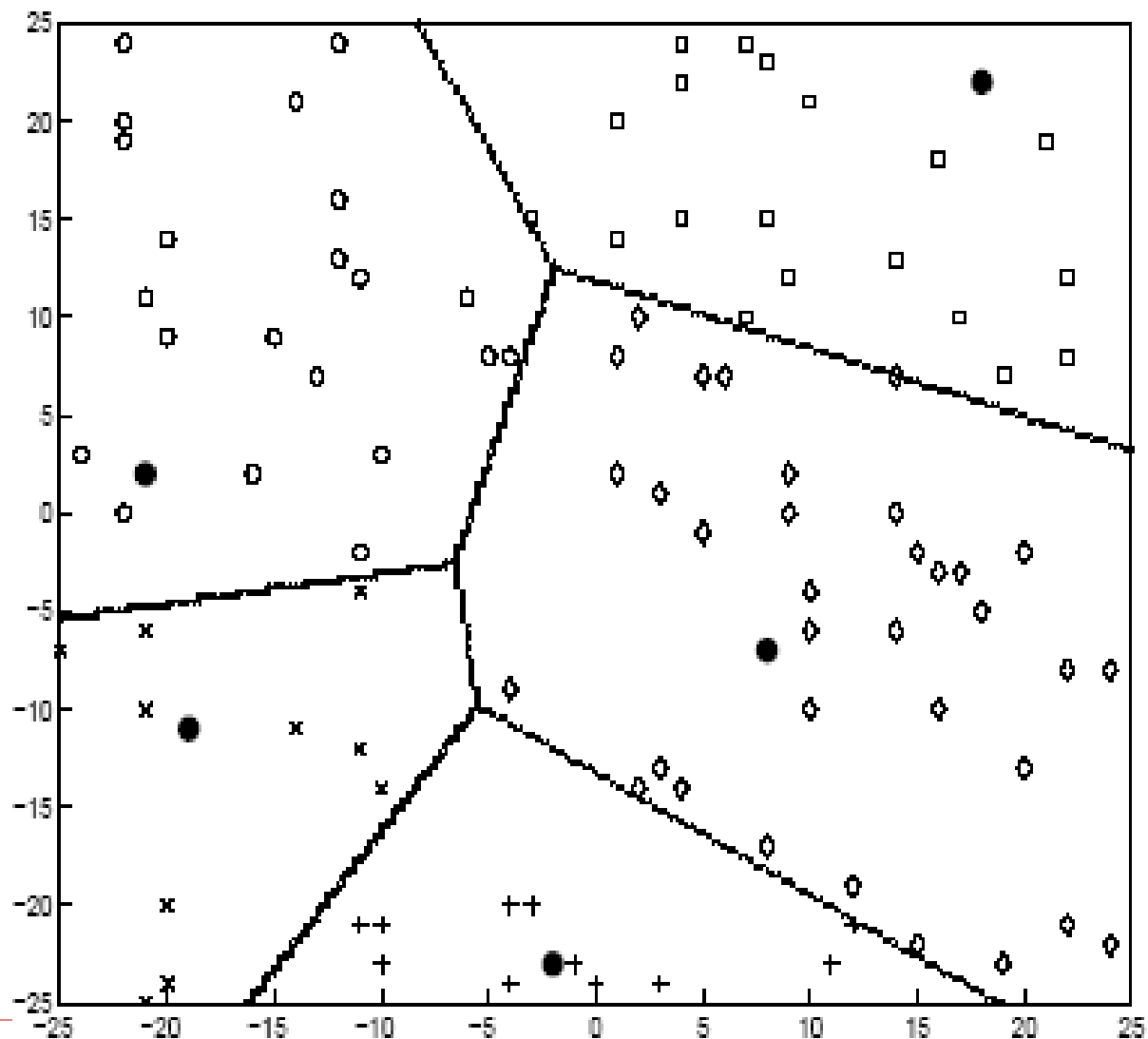


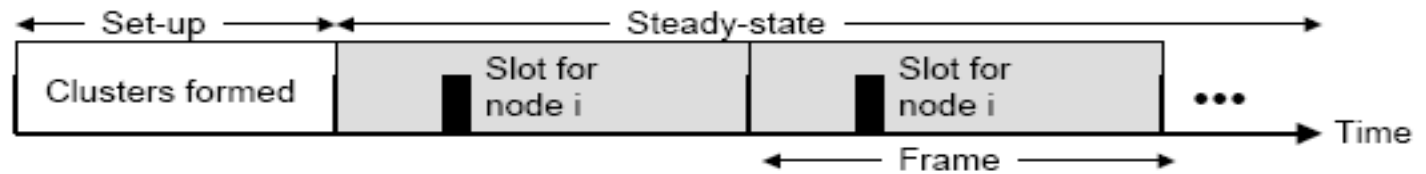
Figure 3-3: Flow-graph of the distributed cluster formation algorithm for LEACH.

Inicializálás – klaszterformálás példa



Állandósult állapot

- A kommunikáció keretekre bontva működik:



- A node-ok keretenként max egyszer küldhetnek adatot a CH-nak a számukra allokált fix időrésben.
- Egy keret hossza a klaszterben levő node-ok számával egyenesen arányos.
 - Átlagosan k node van egy klaszterben, de az aktuális érték ettől nagyon is eltérhet!

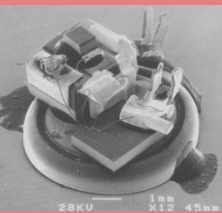
Állandósult állapot (folyt)

□ A nem-CH szenzorok

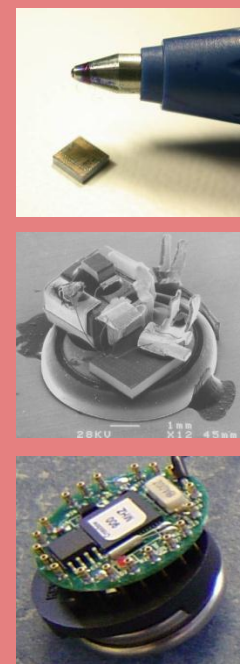
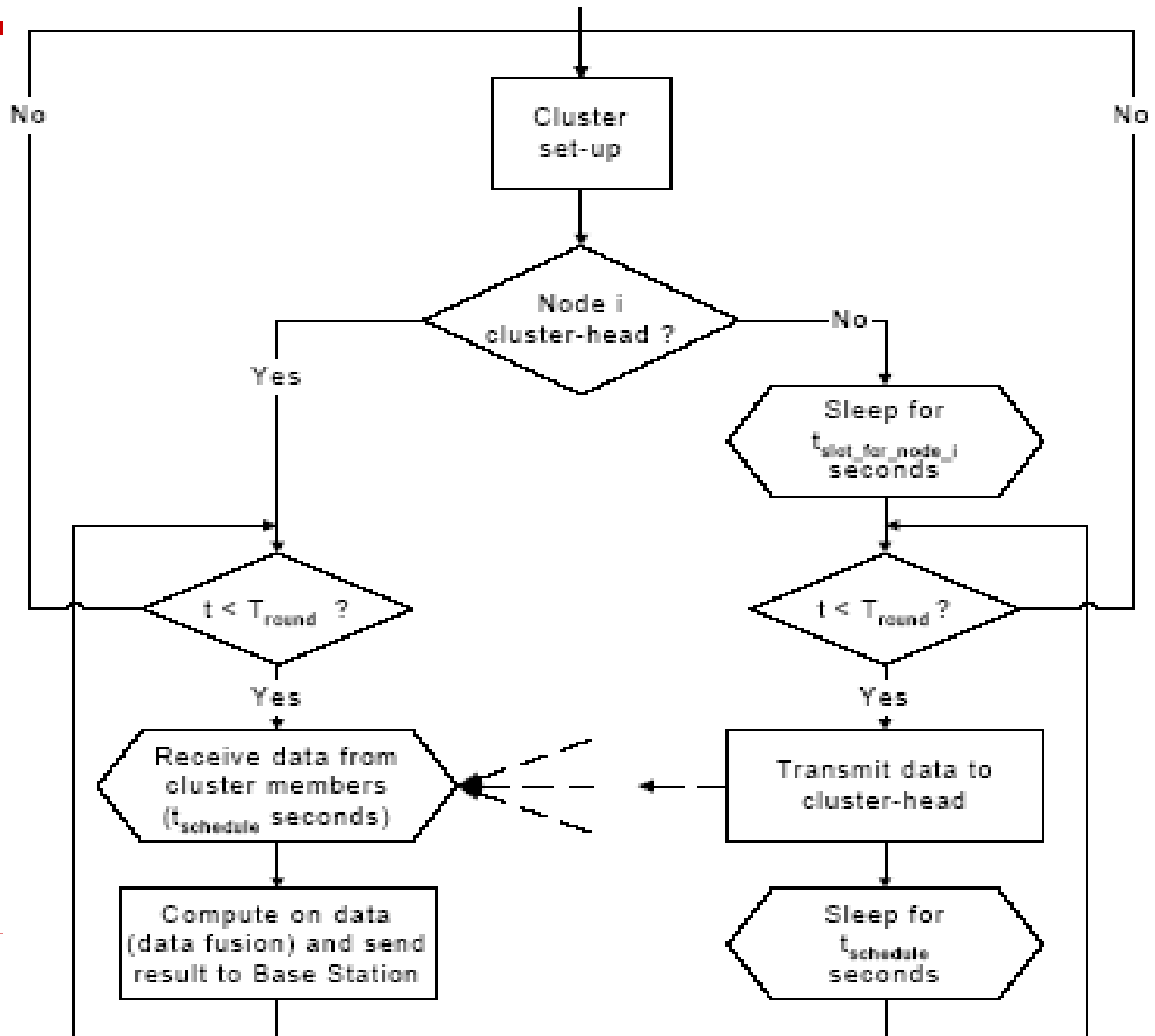
- teljesítményszabályozást használnak a CH-tól vett jel erőssége alapján;
- két allokált időrés között kikapcsolják a rádiójukat.

□ A CH node

- folyamatosan bekapcsolva tartja a rádióját;
- keretenként összegzi és előfeldolgozza a begyűjtött adatokat;
- továbbítja az információt a BS-nek. (nagy energia szükséges!)

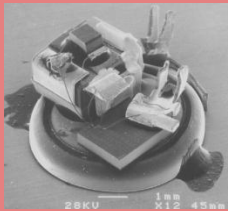
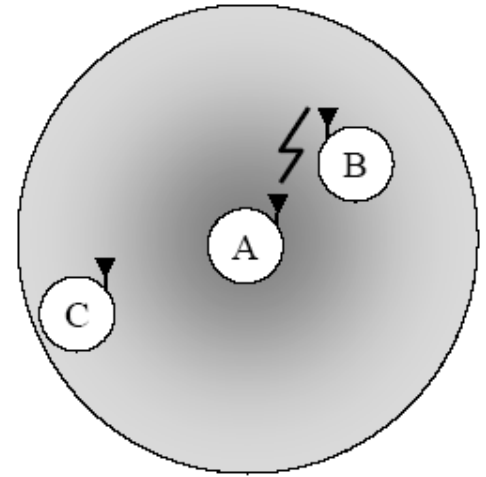


Állandósult állapot



Állandósult állapot

- ❑ Probléma: Egy klaszteren belüli kommunikáció hatással van a környező klaszterek rádiós adásaira is.
- ❑ Megoldás: DS-SS
(direkt-szekvenciális szórt spektrum)
 - Klaszterenként különböző kódok.
 - Klaszteren belül azonos kód.
- ❑ CDMA \neq DS-SS + TDMA
 - A CH-ban csak egyetlen kódra kell figyelni!
- ❑ Az összes CH azonos kódot használva, CSMA hozzáféréssel küldi adatait a BS-nek.



Szenzor adat-aggregáció

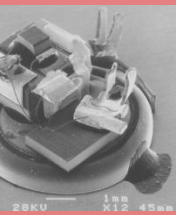
- Az adatok feldolgozása elvégezhető:
 - A BS-ben minden adat begyűjtése után;
 - A CH-kban előfeldolgozva a klaszteren belüli adatokat.

- Az adatfeldolgozás a CH-ban megéri, ha a feldolgozás költsége kisebb, mint a megtakarított átviteli költség.

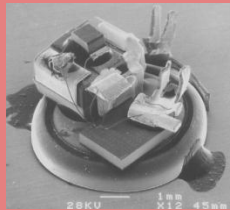
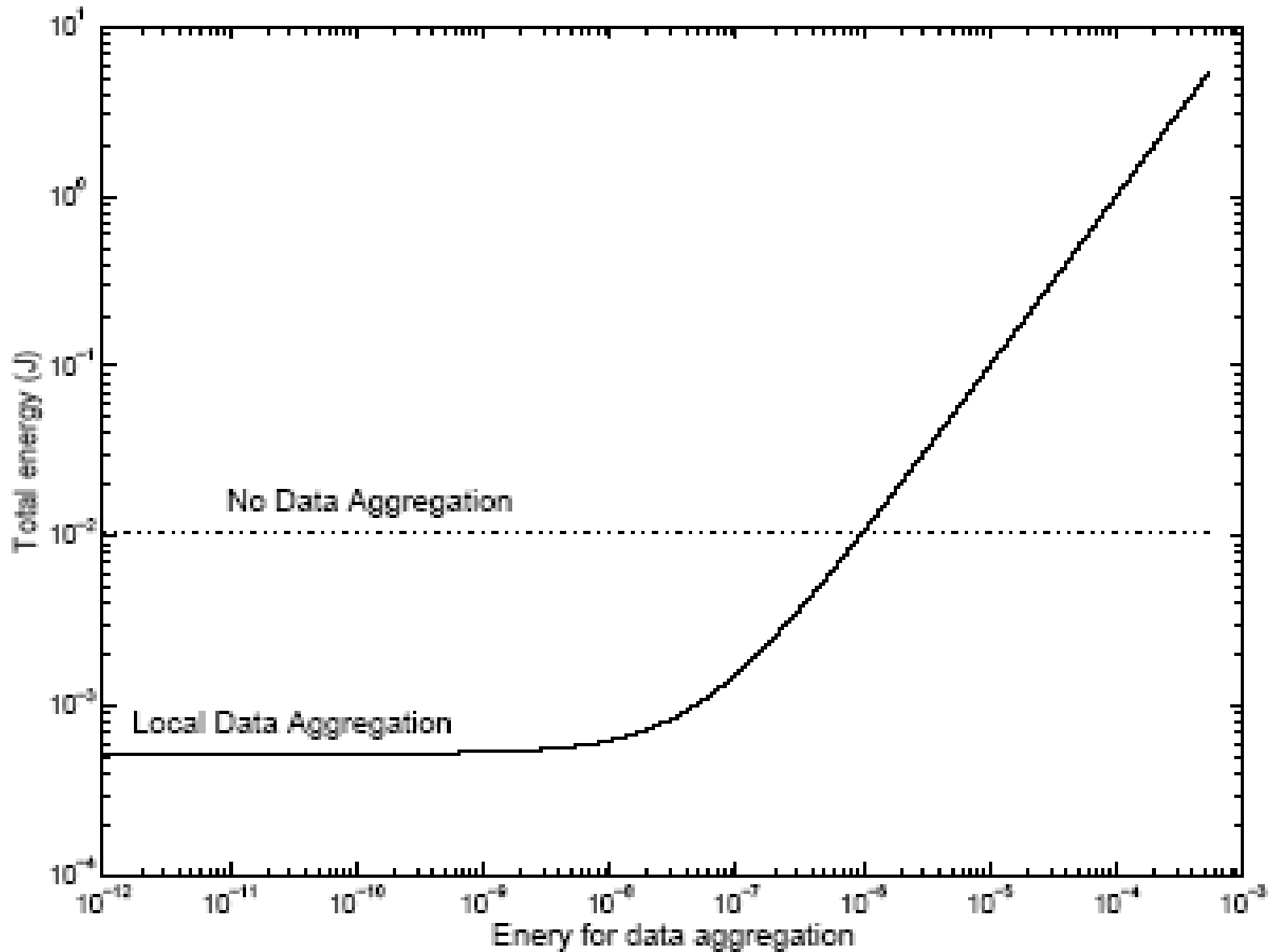
- Paraméterek:
 - 1 bit feldolgozása: E_{DA}
 - 1 bit továbbítása: E_{TX}
 - tömörítési arány: $L:1$

- Megéri az előfeldolgozás, ha: $L * E_{DA} + E_{TX} < L * E_{TX}$


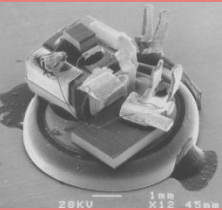

- A CH csak akkor tud hatékonyan tömöríteni, ha a szenzorok által mért adatok erősen korreláltak.



Szenzor adat-aggregáció

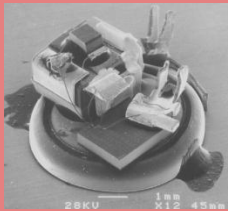
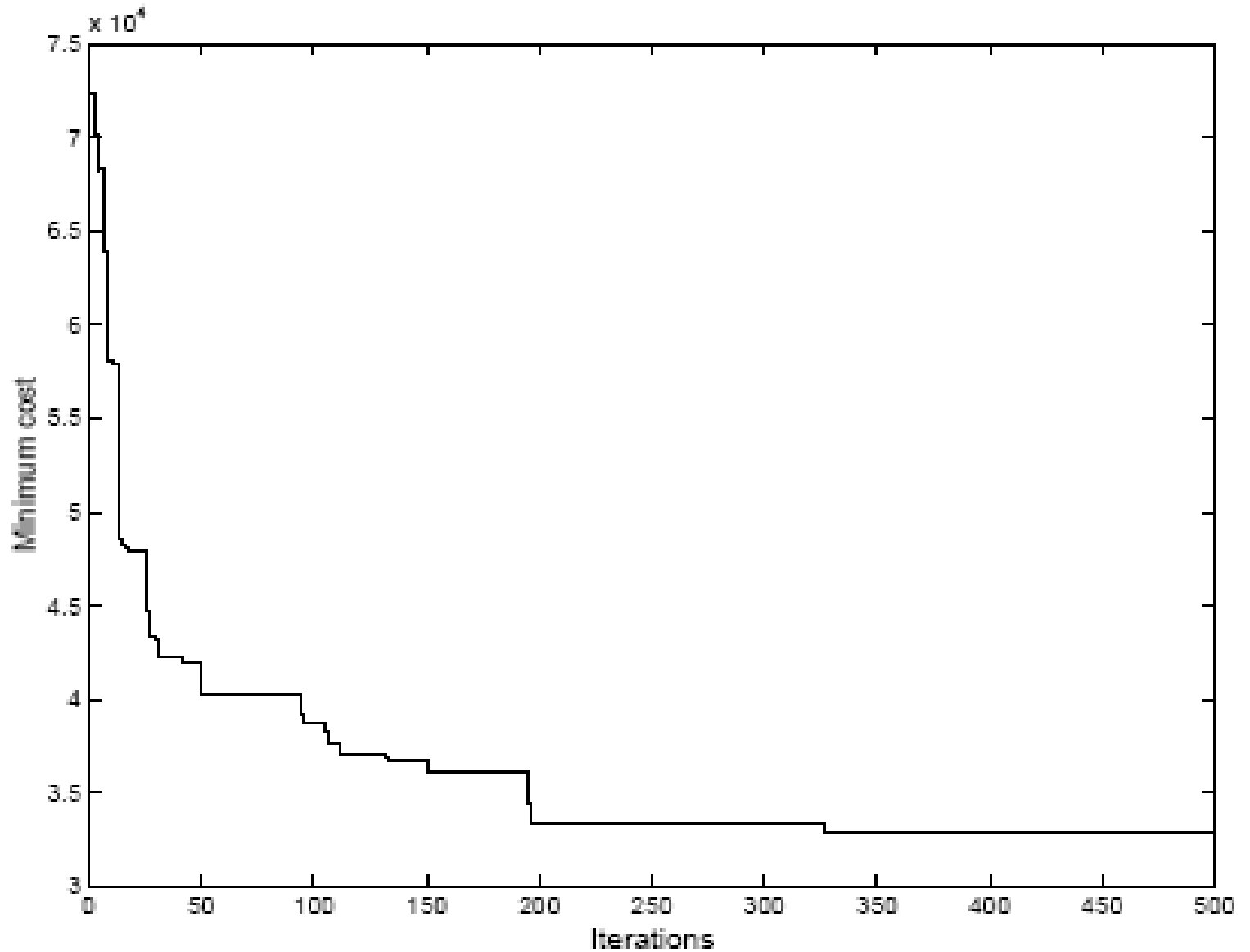


LEACH-C

- 
- 
- 
- A LEACH protokoll problémája: Nincs semmi garancia a klaszterek számára és elhelyezkedésére.
 - „Statisztikailag” több körön keresztül azért „kiátlagolódik”.
 - Megoldás: LEACH-Centralized
 - Az inicializáló fázist egy **központi vezérlő** végzi.
 - Az állandósult állapot nem változik.
 - LEACH-C setup fázis:
 - Minden node elküldi helyzetét(!) és energiáját.
 - A BS optimalizációs algoritmussal meghatározza a CH-kat. („NP-nehéz” probléma!)

megoldás: pl. szimulált lehűtéssel

LEACH-C klaszterformálás-optimalizálás



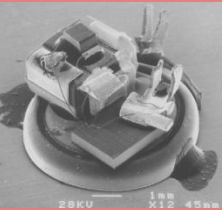
LEACH-C

Előny:

- ❑ Optimális klaszterkiosztás (energia, BW, ...)

Hátrányok:

- ❑ Minden node körönként a BS-sel kommunikál.
- ❑ Mindenkinek ismernie kell a koordinátáit.



Teljesítményanalízis

□ Vizsgált routing megoldások:

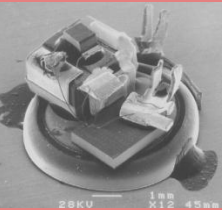
- Közvetlen kommunikáció („Direct”)
- Minimális energiájú multi-hop kommunikáció („MTE”)
- LEACH

Közvetlen kommunikáció:

- Minden node közvetlenül a BS-nek ad.

MTE

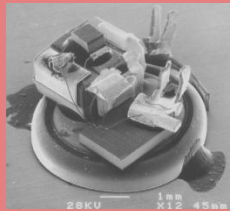
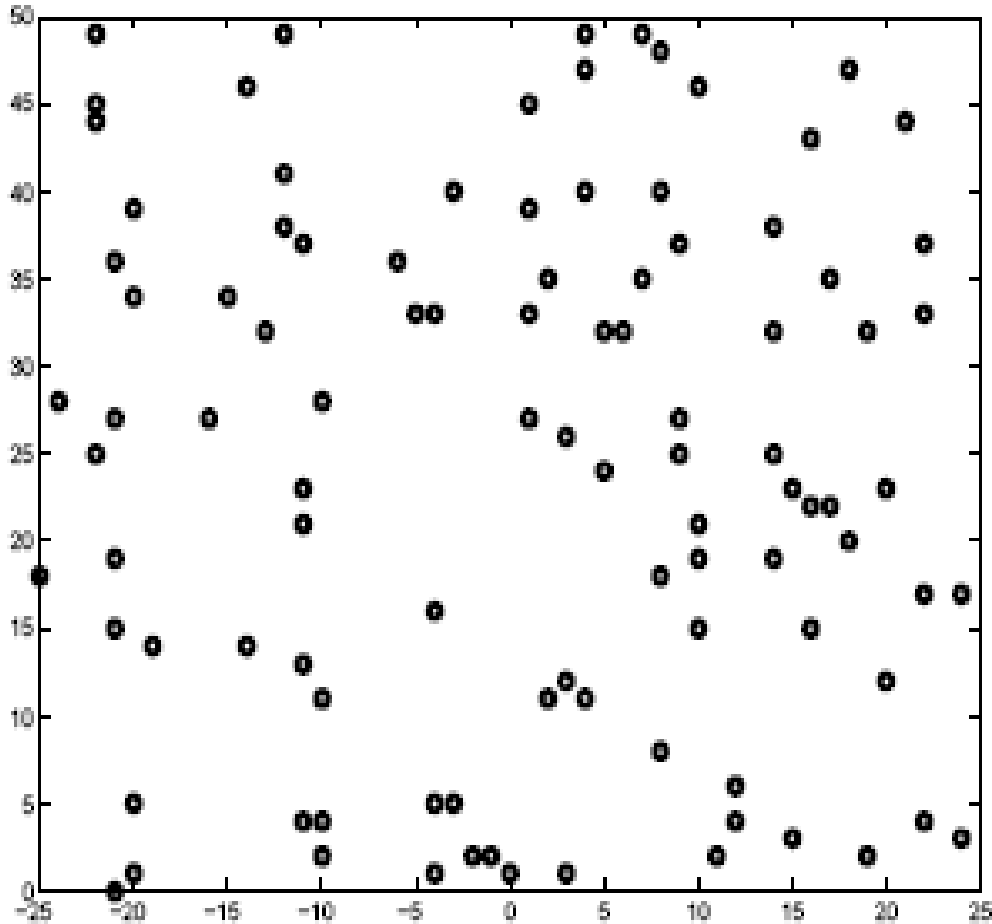
- Minden node a legközelebbi szomszédjának továbbítja a csomagot a BS-felé vezető út mentén.



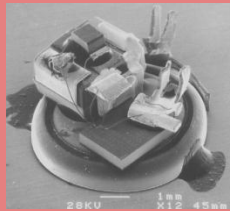
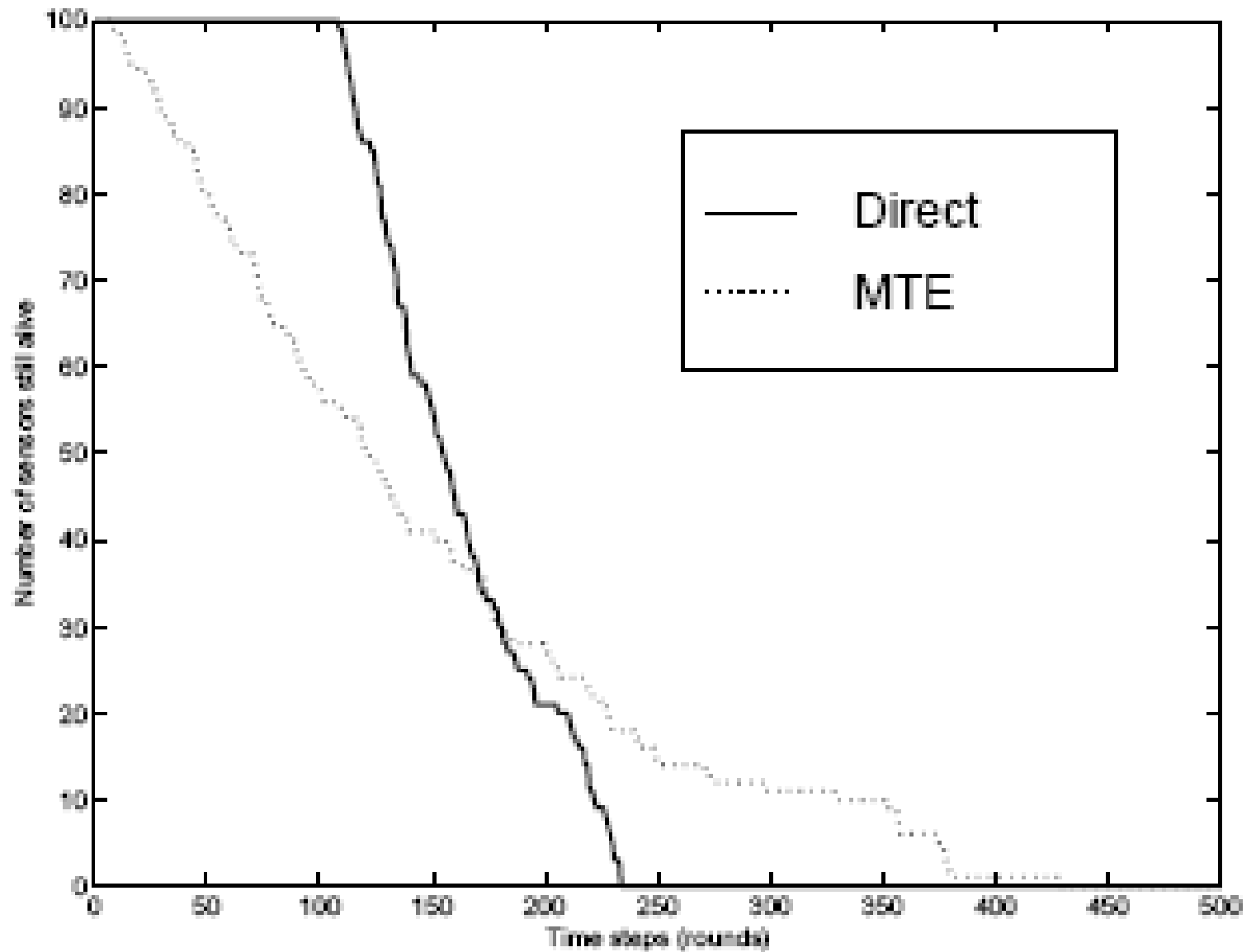
Teljesítményanalízis

vizsgált topológia

BS koordinátái:
($x=0$, $y=-100$)

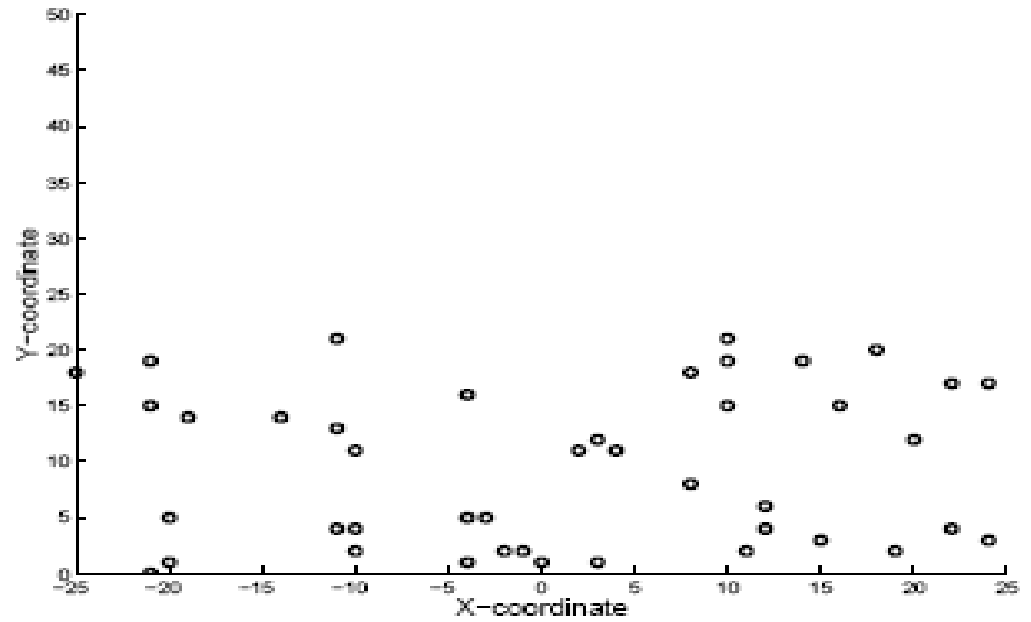


Hálózat élettartama (Direct, MTE)

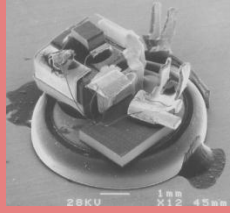
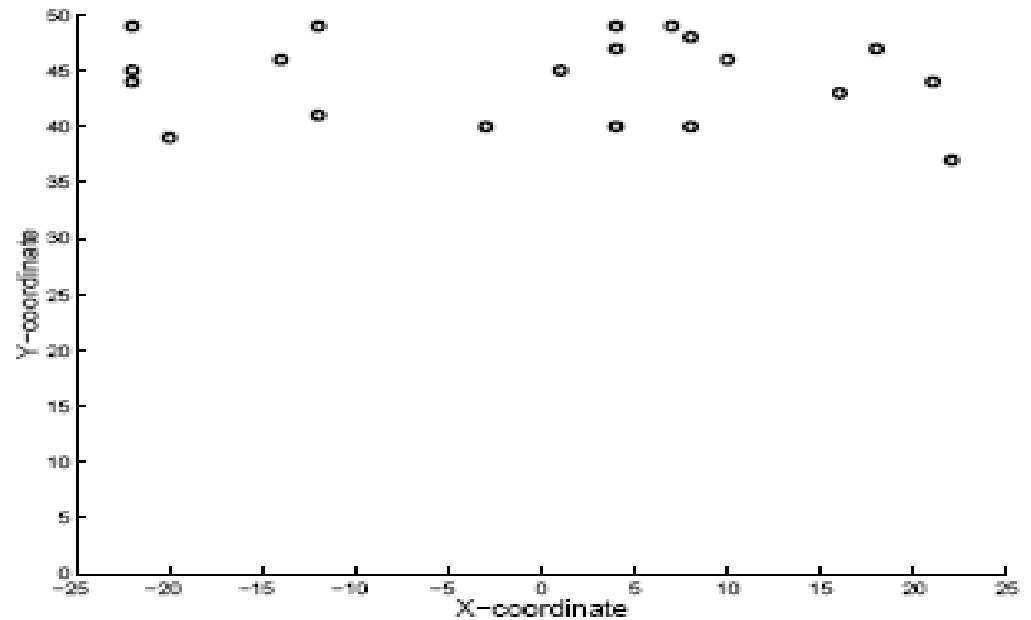


Hálózat élettartama (Direct, MTE)

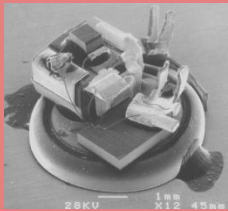
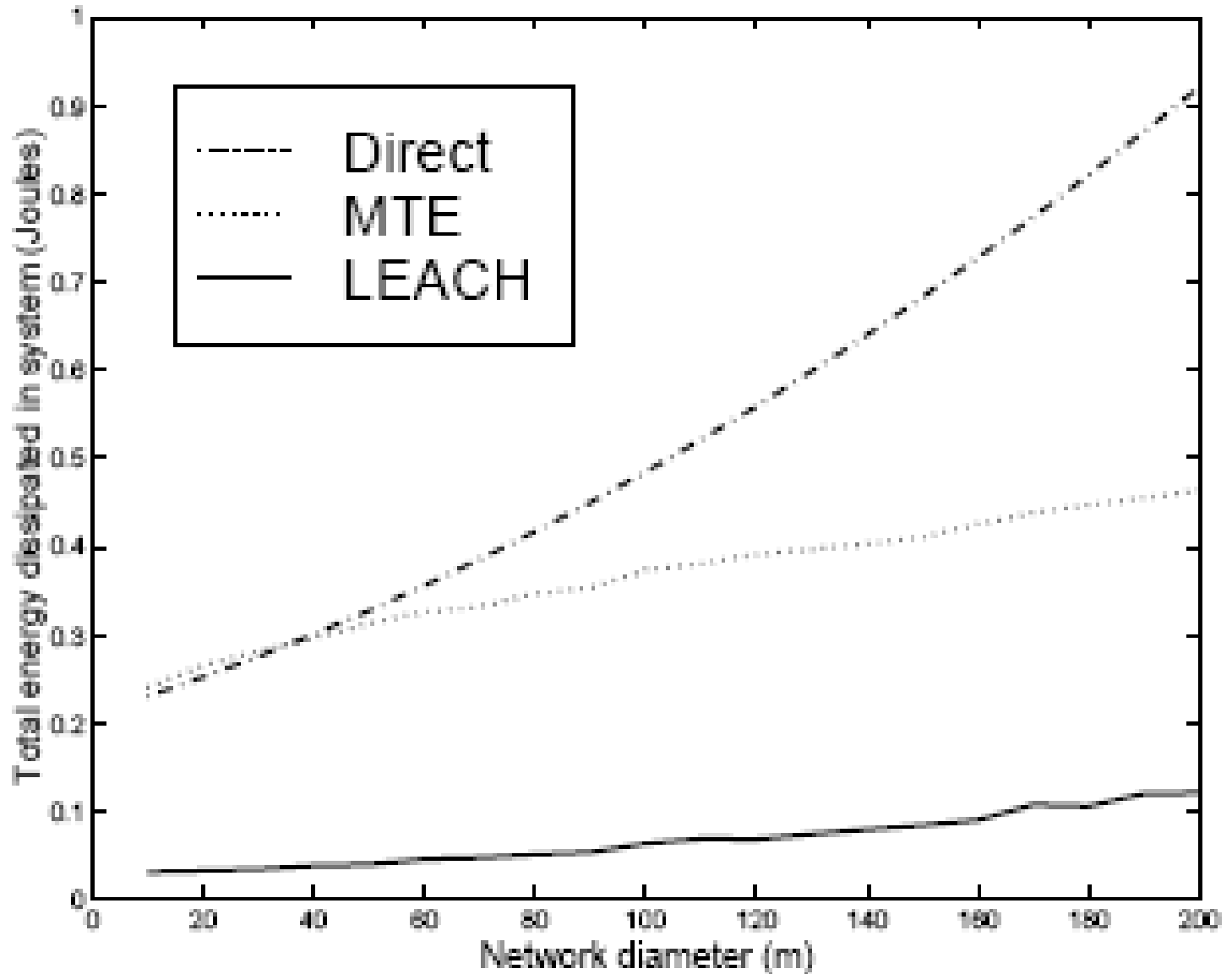
Direct:



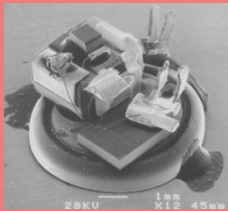
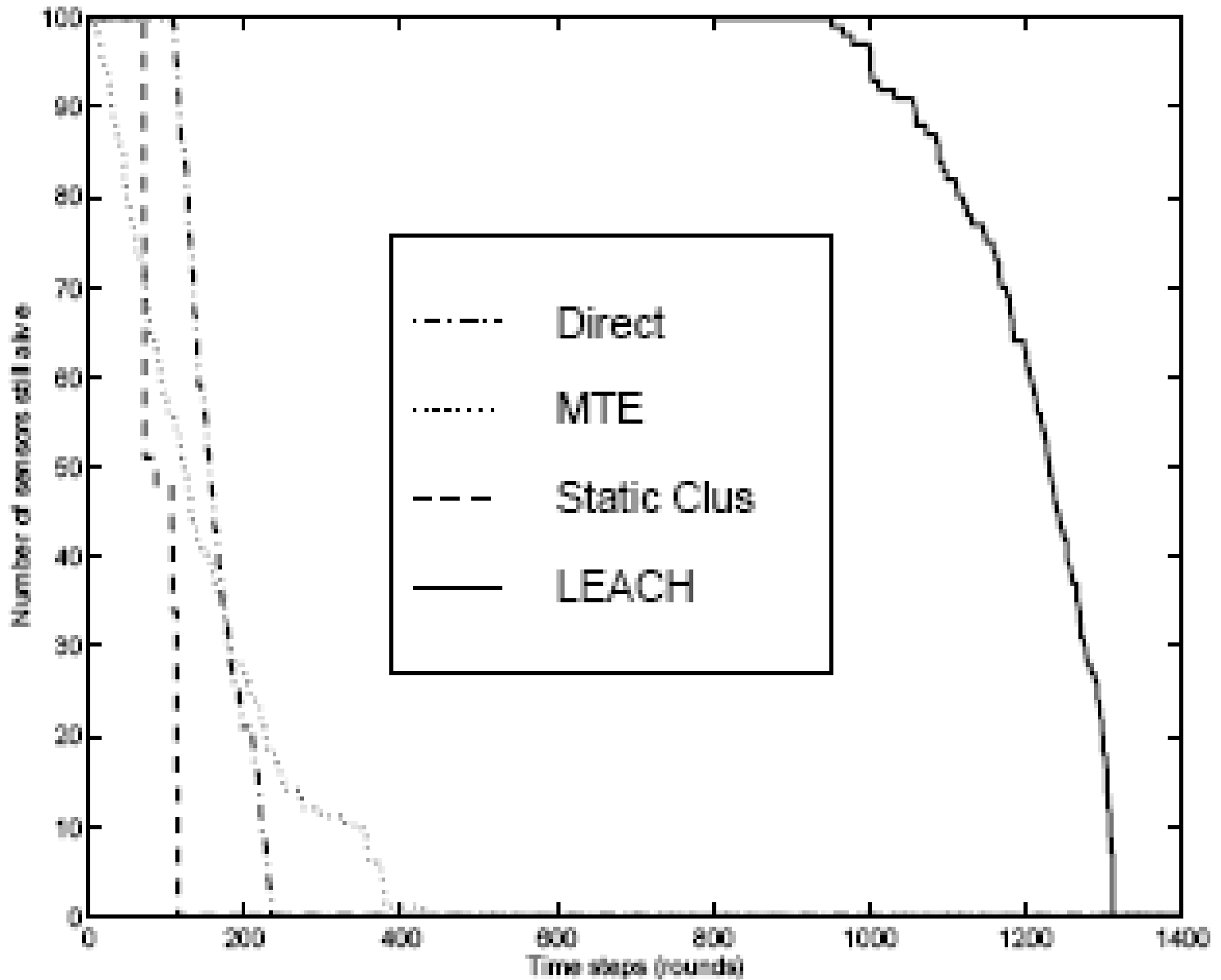
MTE:



Energiafelhasználás



Hálózat élettartama



Hálózat élettartama (2)

Table 2. Lifetimes using different amounts of initial energy for the sensors.

| Energy (J/node) | Protocol | Round first node dies | Round last node dies |
|-----------------|-------------------|-----------------------|----------------------|
| 0.25 | Direct | 55 | 117 |
| | MTE | 5 | 221 |
| | Static Clustering | 41 | 67 |
| | LEACH | 394 | 665 |
| 0.5 | Direct | 109 | 234 |
| | MTE | 8 | 429 |
| | Static Clustering | 80 | 110 |
| | LEACH | 932 | 1312 |
| 1 | Direct | 217 | 468 |
| | MTE | 15 | 843 |
| | Static Clustering | 106 | 240 |
| | LEACH | 1848 | 2608 |

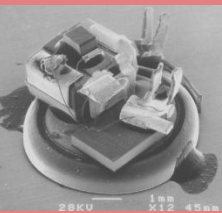


LEACH összefoglalás

- LEACH = Low-Energy Adaptive Clustering Hierarchy

A LEACH protokoll főbb jellemzői:

- Önszerveződő
- Adaptív
- Klaszter alapú, elosztott koordinációval
- Klasztervezérlők véletlenszerű cserélgetése
- Skálázható, robosztus
- Adattömörítés a globális kommunikáció csökkentésére.



ANCAEE: A Novel Clustering Algorithm for Energy Efficiency in Wireless Sensor Networks

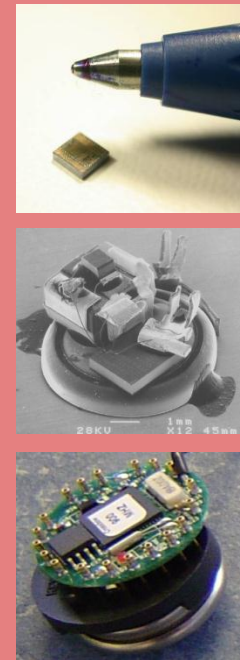
Ademola P. Abidoye^{1,3*}, Nureni A. Azeez^{2,3}, Ademola O. Adesina^{1,3}, Kehinde K. Agbele^{1,3}

¹Wireless Sensor Networks & Natural Language Research Group, Cape Town, South Africa

- „... However, Low Energy Adaptive Clustering Hierarchy (LEACH) Protocol was proposed by [15]. This protocol is one of the most famous hierarchical routing algorithms for energy efficiency in WSNs. Other algorithms developed thereafter were based on this algorithm.”

- TL-LEACH
 - Two CHs per cluster: primary and secondary

- ANCAEE
 - Multi-hop inter-cluster transmission towards the BS



ANCAEE performance

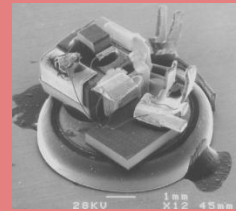
From the radio model, the energy expended by the radio to transmit k bits of data to distance d is given by

$$\begin{aligned}
 E_{tx}(k, d) &= E_{elec} * k + E_{amp} * k \\
 &= E_{elec} * k + \varepsilon_{fs} * d^2 * k \text{ where } d < d_0 \quad (3) \\
 &= E_{elec} * k + \varepsilon_{mp} * d^4 * k \text{ } d \geq d_0
 \end{aligned}$$

To receive k bits of data is given by

$$E_{rx}(k) = E_{elec} * k \quad (4)$$

| Parameters | Values |
|---|------------------------------|
| Number of Sensor Nodes | 100 |
| Network Dimension | 100 m × 100 m |
| Nodes initial energy (K) | 0.5 J |
| Transmitter circuitry dissipation (E_{ele}) | 50 nJ/bit |
| Amplifier dissipation multipath (ε_{mp}) | 0.0015 pJ/bit/m ⁴ |
| Data packet size | 100 bytes |
| Amplifier dissipation free state (ε_{fs}) | 10 pJ/bit/m ² |
| Broadcast size | 25 bytes |
| Distance between sensor field and Base-station | $x = 50, y = 350$ |



ANCAEE results

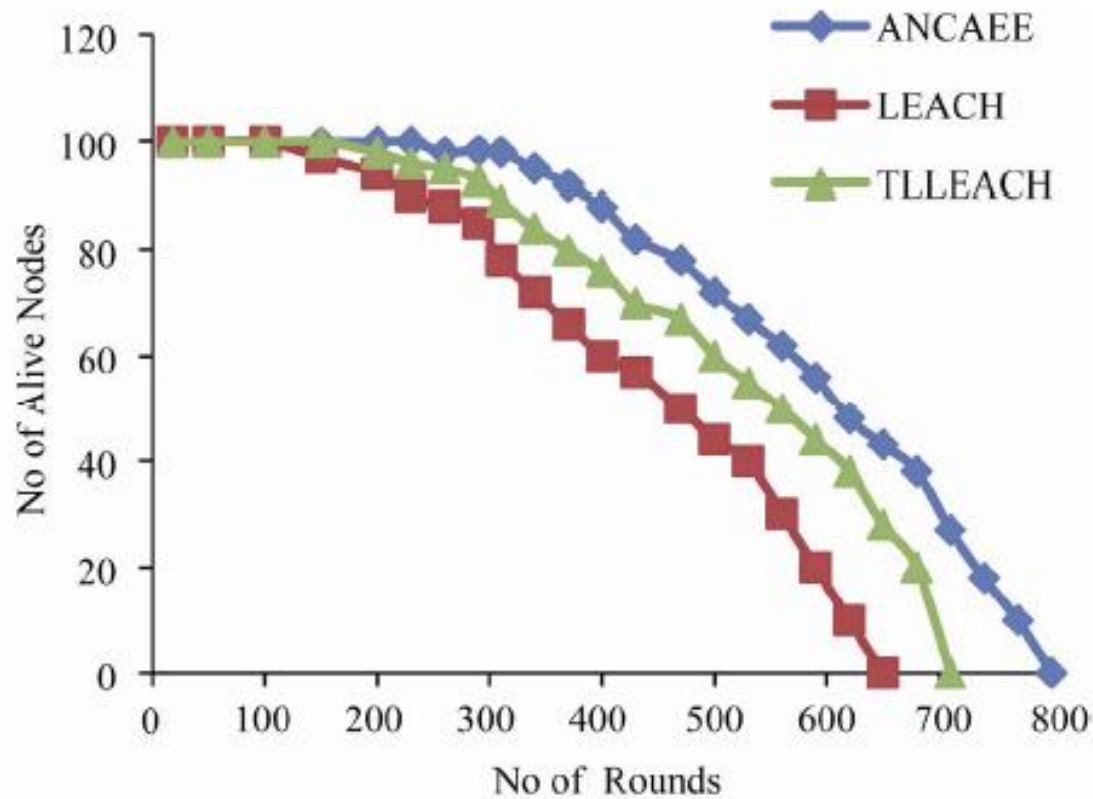


Figure 3. Number of nodes still alive over no of rounds.