

Κινητός και Διάχυτος Υπολογισμός (Mobile & Pervasive Computing)

Δημήτριος Κατσάρος, Ph.D.

Χειμώνας 2005

Διάλεξη 10η

Ιστοσελίδα του μαθήματος

- http://skyblue.csd.auth.gr/~dimitris/courses/mpc_fall05.htm
- http://skyblue.csd.auth.gr/~dimitris/courses/mpc_fall05/
 - [books/](#)
 - [lectures/](#)
 - [papers/](#)
 - [proj_papers/](#)
 - [present_papers/](#)
- Τοποθετούνται οι διαφάνειες του επόμενου μαθήματος
- Τοποθετούνται τα research papers που αντιστοιχούν σε κάθε διάλεξη. Τα σημαντικά με πρόθεμα **MUST_BE_READ**

Περιεχόμενα

- Κινητά Ad Hoc δίκτυα
 - Clustering σε ad hoc δίκτυα
 - **Αλγόριθμος με Maximal Independent Sets**
 - Αλγόριθμος με Weakly Connected Dominating Sets
 - Σχηματισμός max-min d-cluster
- Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks)
 - Διάχυση πληροφορίας σε ασύρματα δίκτυα αισθητήρων

Κάτω όριο

Θεώρημα:

Σε asynchronous ασύρματα ad hoc δίκτυα των οποίων ο unit-disk-graph είναι δακτύλιος, ο αριθμός των μηνυμάτων που θα στείλει οποιοσδήποτε κατανεμημένος αλγόριθμος για nontrivial CDS, είναι τουλάχιστον $\Omega(n \log n)$

Αλγόριθμος

Αποτελείται από δυο φάσεις:

1. Κατασκευή του Maximal Independent Set (MIS)
2. Dominating Tree

Κατασκευή του MIS

- Δεδομένο ένα rooted spanning tree T
- Rank ενός κόμβου είναι το ταξινομημένο ζεύγος του επιπέδου του και του ID του
- Ranking διαδικασία
- Color Marking

Διαδικασία Ranking

Δομή:

1. Κάθε κόμβος διατηρεί δυο τοπικές μεταβλητές x_1 , x_2
 - x_1 = # των γειτόνων των οποίων τα επίπεδα δεν έχουν αναγνωριστεί ακόμα (αρχικά είναι ο # των γειτόνων)
 - x_2 = # των παιδιών που δεν έχουν αναφέρει ακόμα την περάτωση (αρχικά είναι ο # των παιδιών)
2. Κάθε κόμβος διατηρεί μια levelList
 - Αποθηκεύει τα επίπεδα των γειτόνων του (αρχικά είναι κενή)
3. Κάθε κόμβος διατηρεί την τοπική μεταβλητή y
 - Αποθηκεύει τον αριθμό των lower-ranked γειτόνων

Διαδικασία Ranking

- Ρίζα ανακοινώνει το επίπεδό της 0 με broadcasting ενός μηνύματος LEVEL
- Όταν ληφθεί ένα μήνυμα LEVEL:
 - levelList += (sender's level, sender's rank)
 - x_1 - -
- Εάν sender = its parent στο T τότε
 - its level = sender's level + 1
 - Broadcast its LEVEL message
- Εάν $x_1 = 0$
 - $y = \#$ lower ranked γείτονες

Διαδικασία Ranking

- Εάν $x_2=0$ και its level is set
 - Μετάδοση ενός μηνύματος LEVEL-COMPLETE στον πατέρα
 - Εάν είναι non-leaf:
 - $x_2 = \#$ of children
 - Εάν είναι root:
 - $x_2 = \#$ of children
 - Τερματισμός
- Όταν ληφθεί μήνυμα LEVEL-COMPLETE:
 - $x_2 - -$

Διαδικασία Color Marking

- Αρχικά marked με white χρώμα
- Κάθε κόμβος διατηρεί blackList (ids of its black neighbors)
- Root marks black, broadcasts BLACK message
- Όταν ληφθεί μήνυμα ένα μήνυμα BLACK:
 - blackList += sender's ID
 - Εάν node = white
 - Mark itself gray
 - Broadcast GRAY message
- Όταν ληφθεί μήνυμα ένα μήνυμα GRAY:
 - Εάν sender's rank < its rank
 - A white node decrements y by 1
 - If y = 1 after update, marks itself black., broadcast BLACK message

Διαδικασία Color Marking

- Όταν ένα φύλλο γίνεται marked, στέλνει μήνυμα MARK-COMPLETE στον πατέρα του
- Όταν ληφθεί ένα μήνυμα MARK-COMPLETE:
 - x_2 - -
 - Εάν $x_2 = 0$ και not root:
 - Μεταδίδει μήνυμα MARK-COMPLETE στον πατέρα του
 - Εάν $x_2 = 0$ και root:
 - $x_1 = \#$ of its neighbors
 - Τερματισμός

Κατασκευή του Dominating Tree T^*

Δομή κάθε κόμβου:

- Διατηρεί μια τοπική boolean z
 - Αρχικά είναι 0, τίθεται στο 1 εάν join T^*
- Local variable parent
 - Αποθηκεύει ID of its parent in T^* , αρχικά κενή
- childrenList
 - ID of its children in T^* , αρχικά κενή

Ειδικά, η ρίζα στο T διατηρεί τη μεταβλητή $root$, και $degree$, αρχικοποιημένο στο 0

Κατασκευή του Dominating Tree T^*

- Επιλογή μια ρίζας για το T^* από το T
- Προσκάλεσε κόμβους για να ενωθούν στο T^*
- Οι internal κόμβοι του T^* σχηματίζουν ένα CDS

Επιλογή ρίζας για το T^*

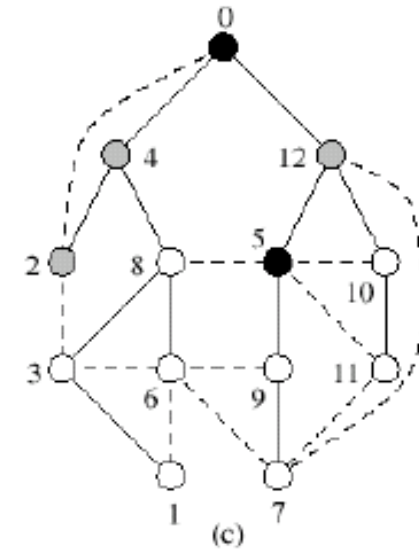
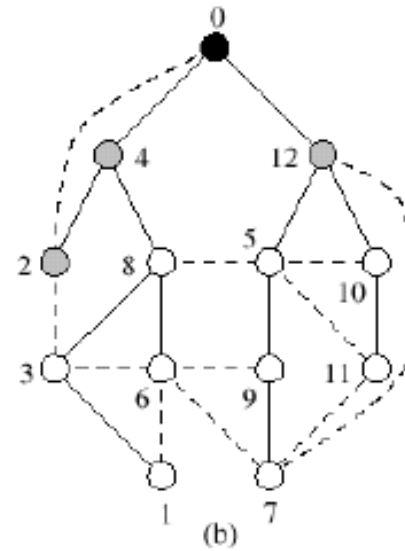
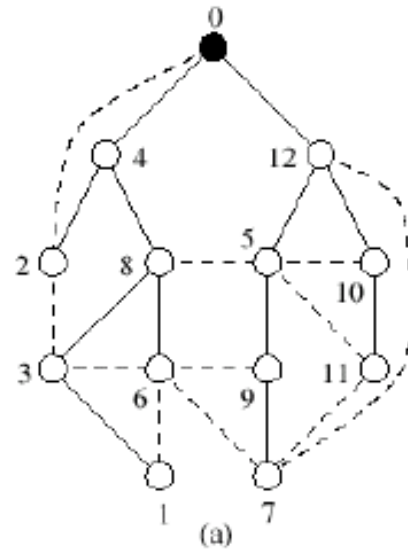
Βασική ιδέα: μια ρίζα του T^* είναι γείτονας της ρίζας του T που έχει τον μεγαλύτερο αριθμό από black γείτονες

- Ρίζα του T κάνει broadcast ένα μήνυμα QUERY
- Όταν ληφθεί το μήνυμα QUERY:
 - Ο κόμβος μεταδίδει στον αποστολέα ένα μήνυμα REPORT που περιέχει τον # των black γειτόνων
 - Ρίζα ελαττώνει x_1 κατά 1; Θέτει degree ίσο με #black γειτόνων εάν είναι μεγαλύτερη; Θέτει root variable ίσο με αυτό το id.
- Εάν $x_1 = 0$, στέλνει μήνυμα ROOT στον κόμβο που το id του είναι στην root variable

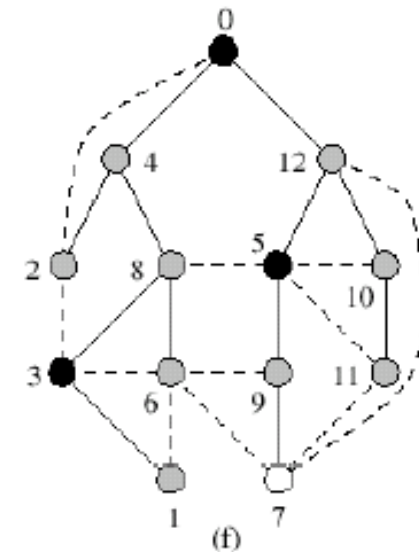
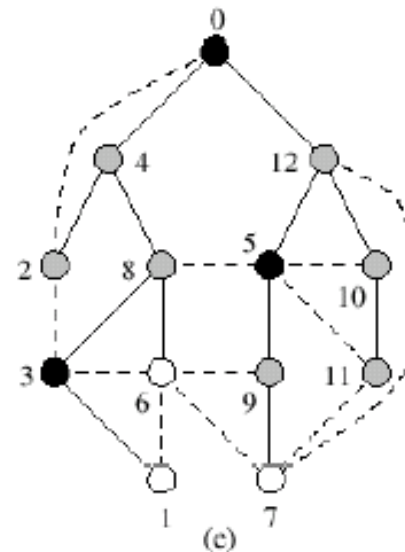
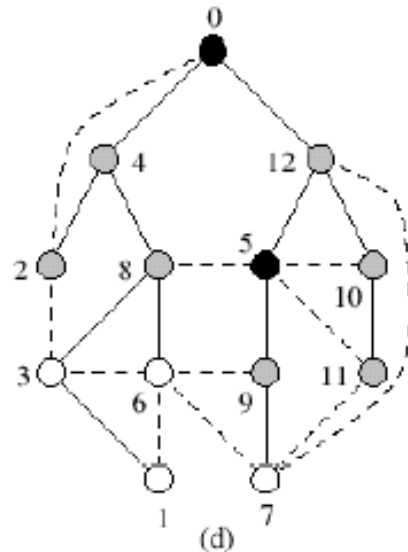
Πρόσκληση κόμβων

- Η ρίζα του T^* broadcasts μήνυμα INVITE2
- Όταν ληφθεί το μήνυμα INVITE2
 - Εάν color=black, $z=0$ τότε
 - Θέσε Set $z = 1$
 - Parent = sender's ID
 - Μετάδοση JOIN μηνύματος στον αποστολέα (sender)
 - Broadcast μήνυμα INVITE1
- Όταν ληφθεί μήνυμα INVITE1
 - Εάν color=gray, $z=0$ τότε
 - Θέσε $z = 1$
 - Parent = sender's ID
 - Μετάδοση JOIN μηνύματος στον αποστολέα
 - Broadcast μήνυμα INVITE2
- Όταν ληφθεί μήνυμα JOIN
 - Προσθήκη του sender's ID στην childrenList

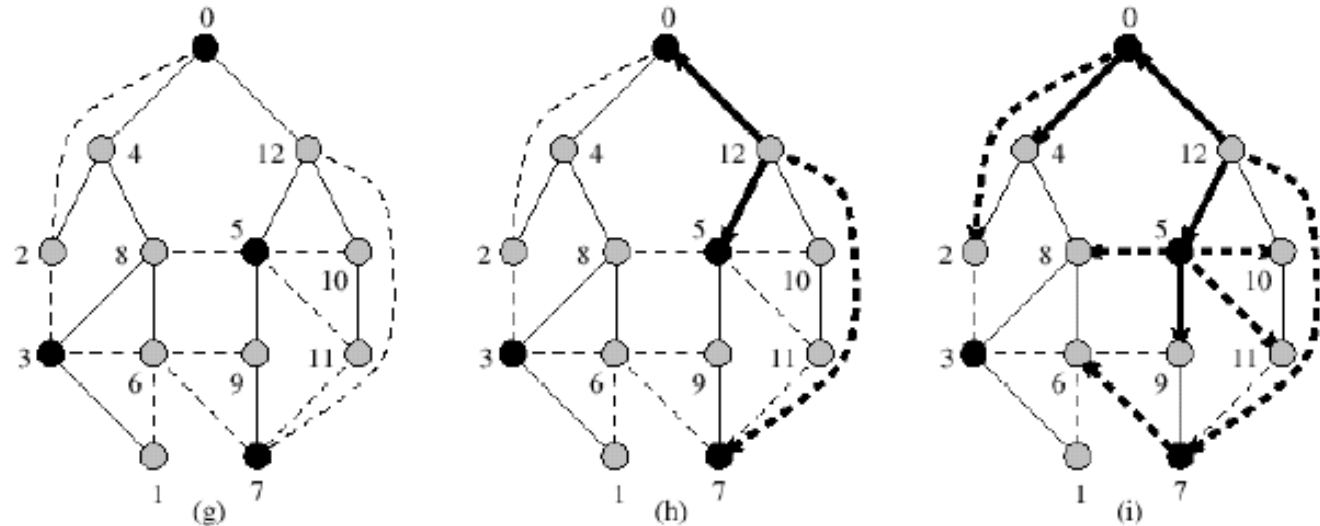
Κατασκευή του MIS και Dom. Tree T^*



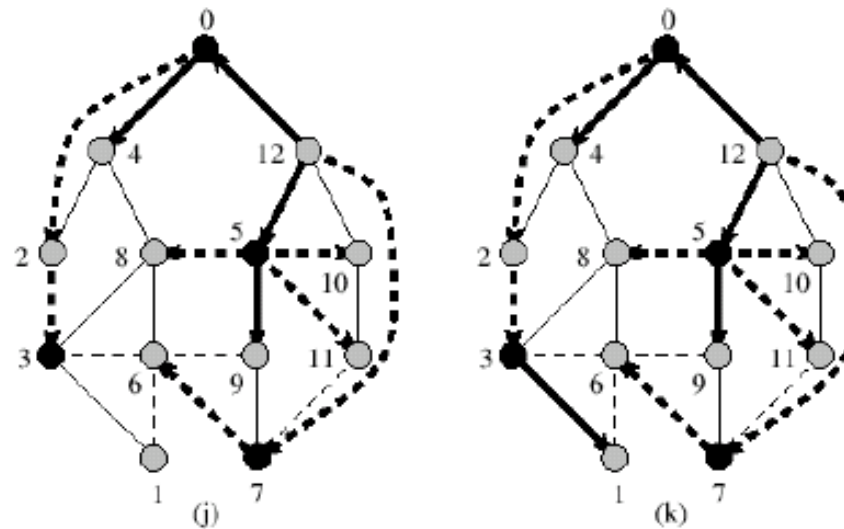
Κατασκευή MIS
(a) - (g)



Κατασκευή του MIS και Dom. Tree T^*



**Κατασκευή DomTree
(h) - (k)**



$$\text{CDS} = \{12, 0, 5, 7, 2, 3\}$$

Ανάλυση επίδοσης

- Approximation factor ≤ 8
- Πολυπλοκότητα χρόνου: $O(n)$
- Πολυπλοκότητα μηνυμάτων: $O(n \cdot \log n)$
- Nontrivial CDS

Περιεχόμενα

- Κινητά Ad Hoc δίκτυα
 - Clustering σε ad hoc δίκτυα
 - Αλγόριθμος με Maximal Independent Sets
 - **Αλγόριθμος με Weakly Connected Dominating Sets**
 - Σχηματισμός max-min d-cluster
- Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks)
 - Διάχυση πληροφορίας σε ασύρματα δίκτυα αισθητήρων

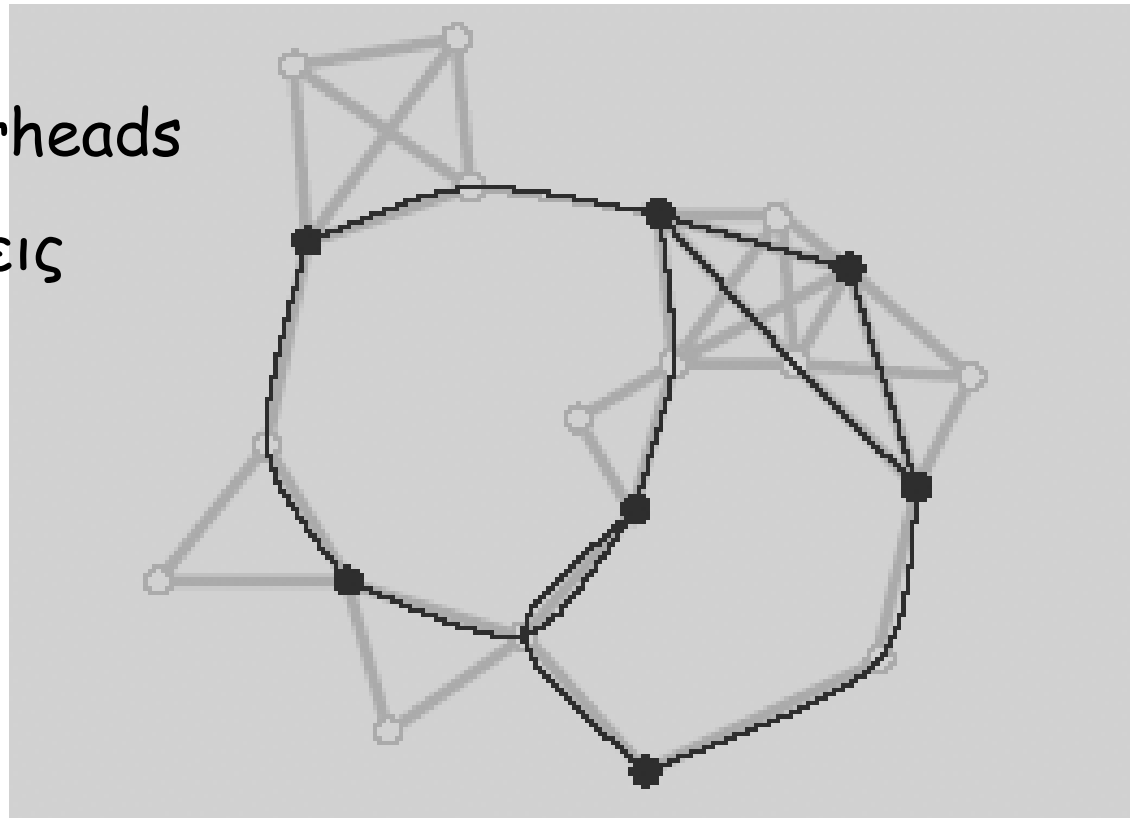
Clustering ad hoc δίκτυα

- Επιθυμητό να δημιουργήσουμε μια αφηρημένη δομή πάνω από το δίκτυο, έτσι ώστε τοπικές αλλαγές να μην χρειάζεται να γίνουν γνωστές σε όλο το δίκτυο
- Με χρήση υποδομών που λέγονται **clusters**
- **Clustering**: η διαδικασία ορισμού αυτών των υποδομών μέσα σε όλη την τοπολογία του δικτύου
- Οι κόμβοι διακρίνονται σε:
 - **Clusterheads**
 - **Gateways**
 - **Ordinal nodes**

Παράδειγμα ad hoc δικτύου

Μαύροι κόμβοι: clusterheads

Μαύρες ακμές: συνδέσεις
μεταξύ των clusters

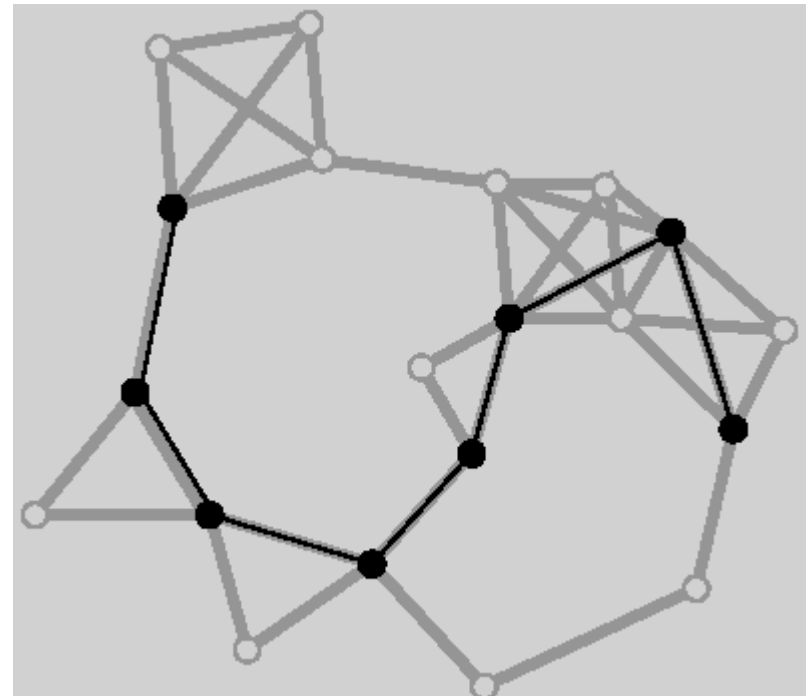


Μικρό dominating set

- Χρήση των κόμβων ενός dominating set ως **clusterheads**
- Αναθέτουμε κάθε κόμβο σ' εκείνο το cluster που αντιστοιχεί στην κορυφή που κυριαρχεί πάνω της
- Μικρός αριθμός από clusterheads για να απλοποιήσουμε τη δομή του δικτύου
- Ως γνωστό, η εύρεση του ελάχιστου dominating set είναι NP-complete

Connected dominating set

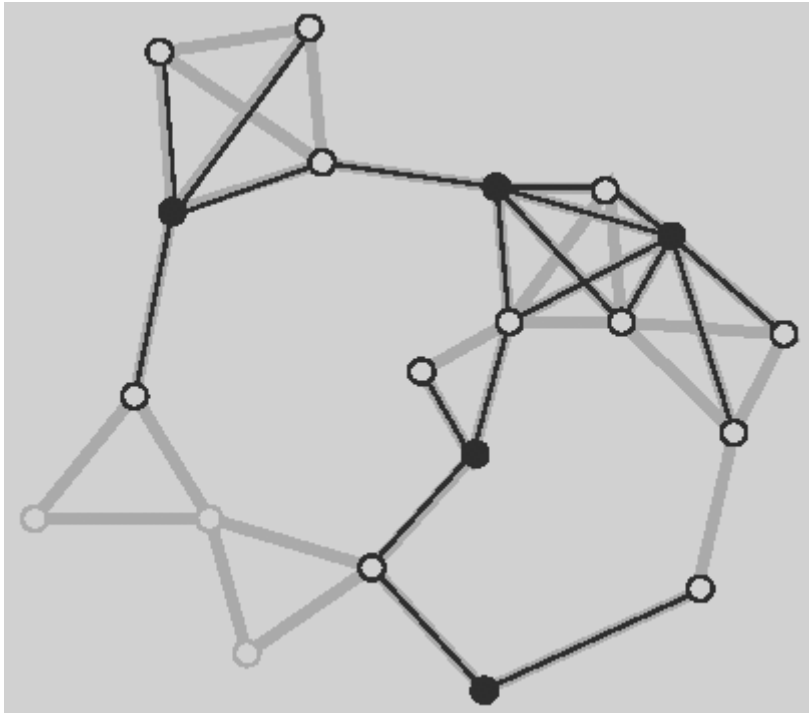
- Ένα connected dominating set (CDS) ενός γραφήματος G είναι ένα dominating set του οποίου το **induced γράφημα** είναι συνδεδεμένο
- Χρησιμοποιείται εύκολα για routing μηνυμάτων μεταξύ των clusters
- Η εύρεση ενός ελάχιστου CDS is NP-complete
- Ο αριθμός των clusters μεγάλος



Weakly induced subgraph

- Ελάττωση του αριθμού των clusters με την “χαλάρωση” της απαίτησης για συνδεσμικότητα
- Το υπογράφημα που είναι **weakly induced** από το $S (S \subseteq V)$ είναι το γράφημα $\langle S \rangle_w = (N[S], E \cap (N[S] \times S))$.
- $\langle S \rangle_w$ περιλαμβάνει τους κόμβους του S και όλους τους κόμβους που είναι γείτονες αυτών ως σύνολο κόμβων του $\langle S \rangle_w$
- Οι ακμές του $\langle S \rangle_w$ είναι όλες οι ακμές του G , που έχουν τουλάχιστον το ένα άκρο τους στο S

Παράδ. weakly induced subgraph

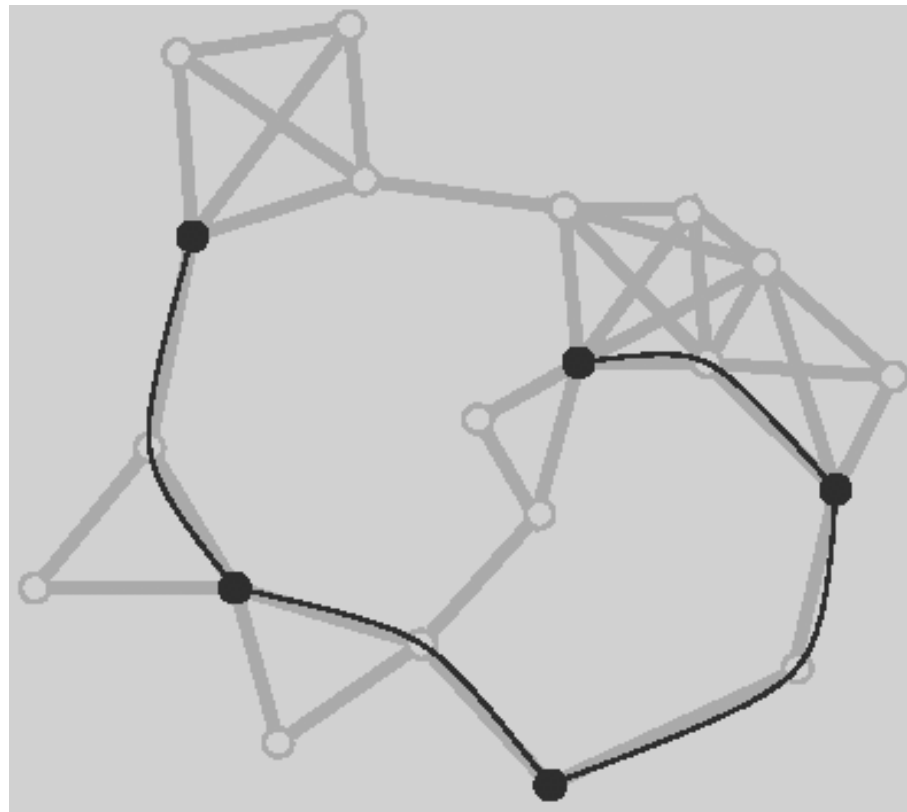


Σύνολο κόμβων: μαύροι κόμβοι

Σύνολο ακμών: μαύρες γραμμές

Weakly-connected dominating set

- Ένα υποσύνολο κόμβων S είναι ένα **weakly-connected dominating set (WCDS)**, εάν το S είναι dominating set και το $\langle S \rangle_w$ είναι συνδεδεμένο



Αλγόριθμ. για εύρεση μικρού WCDS

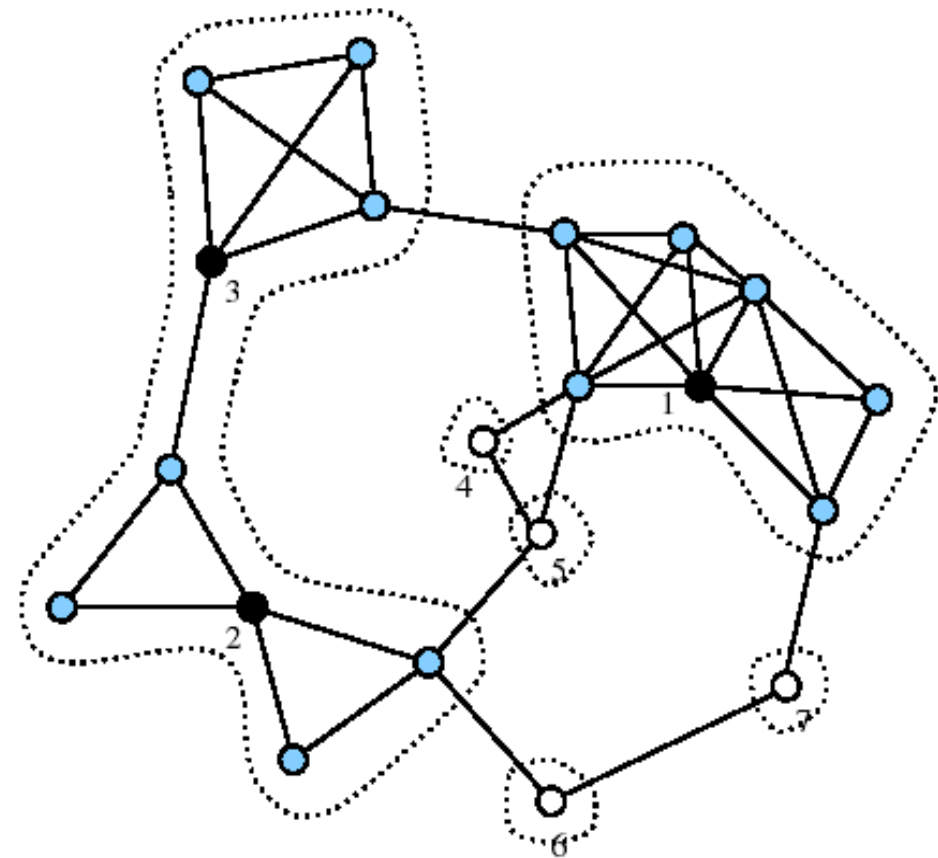
- **Αλγόριθμοι I και II**: Δυο centralized αλγόριθμοι
- **Αλγόριθμοι III και IV**: Distributed υλοποιήσεις των αλγορίθμων I και II
- **Αλγόριθμος V**: Distributed Asynchronous προσέγγιση

Αλγόριθμος I (περίληψη)

- Δεδομένου ενός γραφήματος $G=(V,E)$, κάθε κόμβος συσχετίζεται με ένα χρώμα (*white*, *gray*, ή *black*)
- Όλοι οι κόμβοι αρχικά είναι *white*
- Σε κάθε επανάληψη, ο αλγόριθμος χρωματίζει *black* έναν *white* ή *gray* κόμβο καθώς επίσης χρωματίζει *gray* όλους τους γειτονικούς του κόμβους
- Στο τέλος, οι *black* κόμβοι συνιστούν ένα *weakly-connected dominating set*

Ορολογία: η έννοια “piece”

- Piece αναφέρεται σε μια επιμέρους υποδομή του γραφήματος
- Ένα white piece είναι απλά ένας white κόμβος
- Ένα black piece περιέχει ένα maximal σύνολο από black κόμβους των οποίων το weakly induced υπογράφημα είναι συνδεδεμένο συν όποιοι προσκείμενοι gray κόμβοι



Τα pieces σηματοδοτούνται με dotted περιοχές

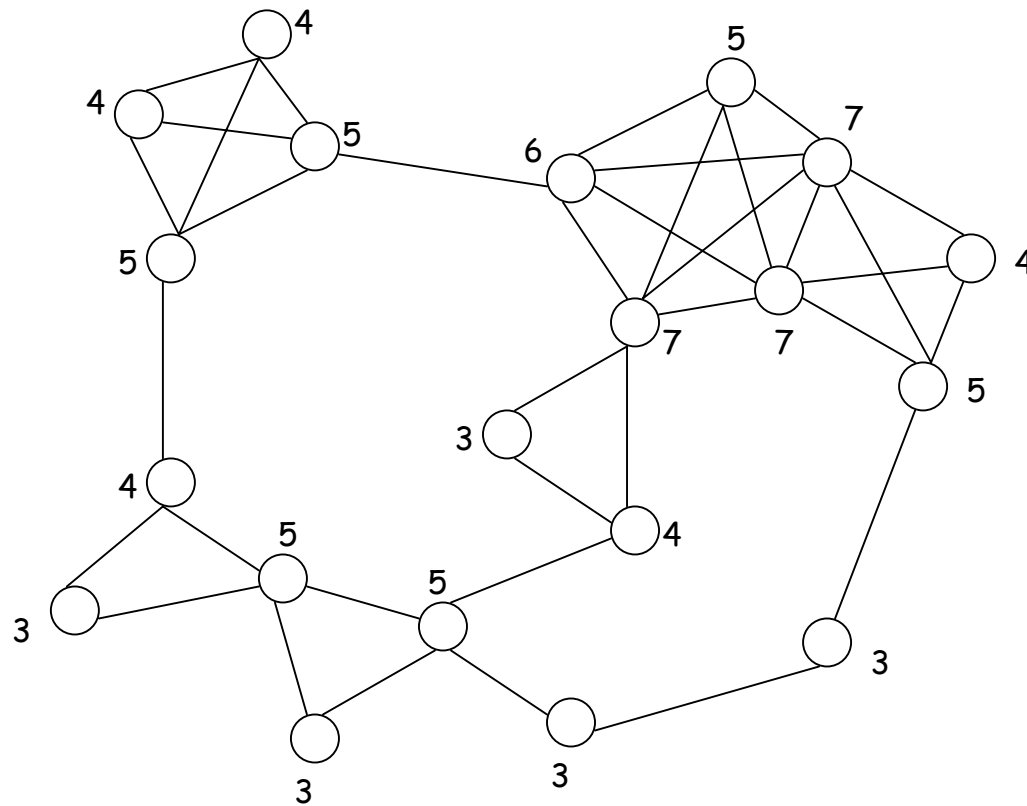
Ορολογία: η έννοια “improvement”

- Το improvement ενός (non-black) κόμβου u είναι ο αριθμός των pieces που θα συνενώνονταν σε ένα μόνο black piece εάν ο u χρωματιζόταν black
- Στο τελευταίο παράδειγμα, χρωματίζοντας τον κόμβο 5 black θα συνένωνε 4 pieces, ενώ χρωματίζοντας black τον κόμβο 4 θα συνένωνε 3 pieces

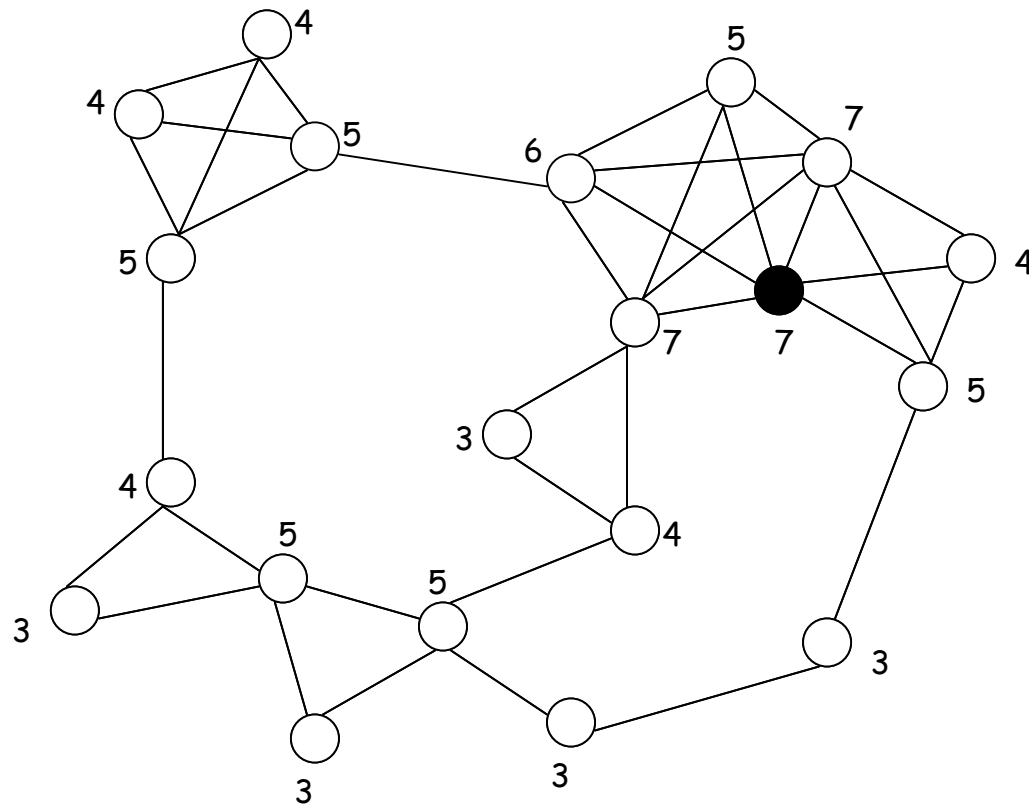
Αλγόριθμος I (λεπτομέρειες)

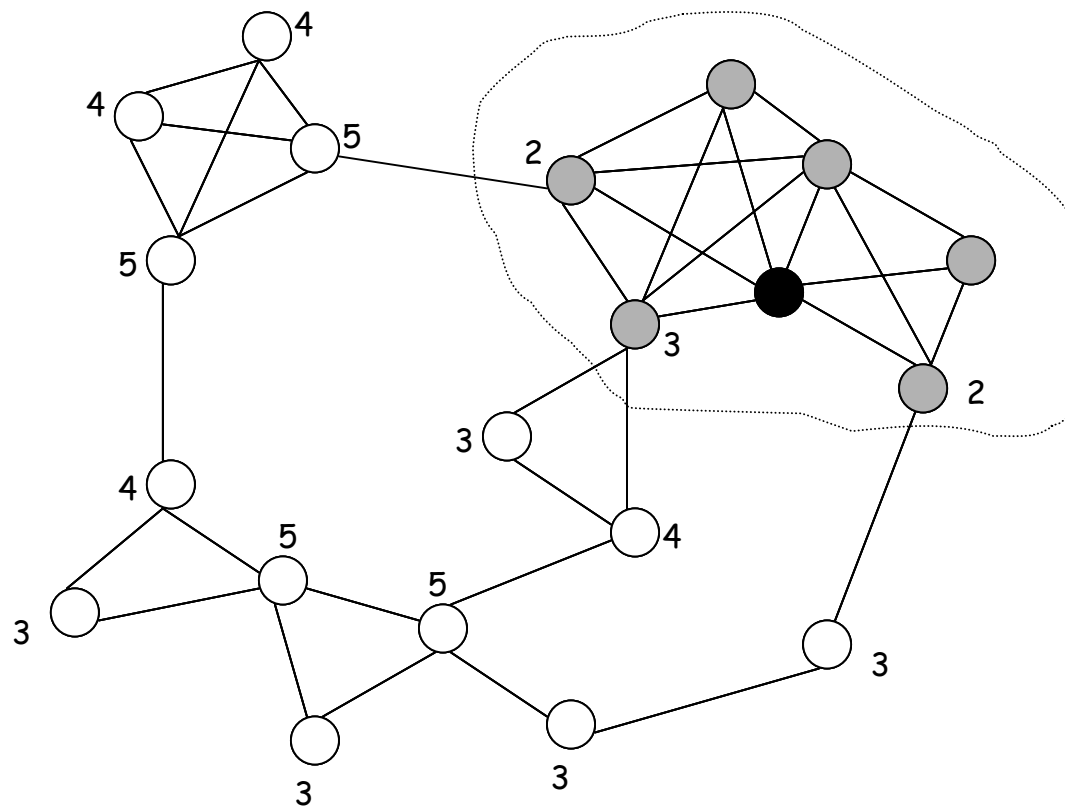
- Σε κάθε επανάληψη, ο αλγόριθμος επιλέγει ένα μόνο white ή gray κόμβο για να χρωματίσει black
- Η επιλογή γίνεται με άπληστο τρόπο: a επιλέγεται ο κόμβος με το μεγαλύτερο improvement
- Μέχρι να απομείνει μόνο ένα piece

Αρχικά όλοι οι κόμβοι είναι white

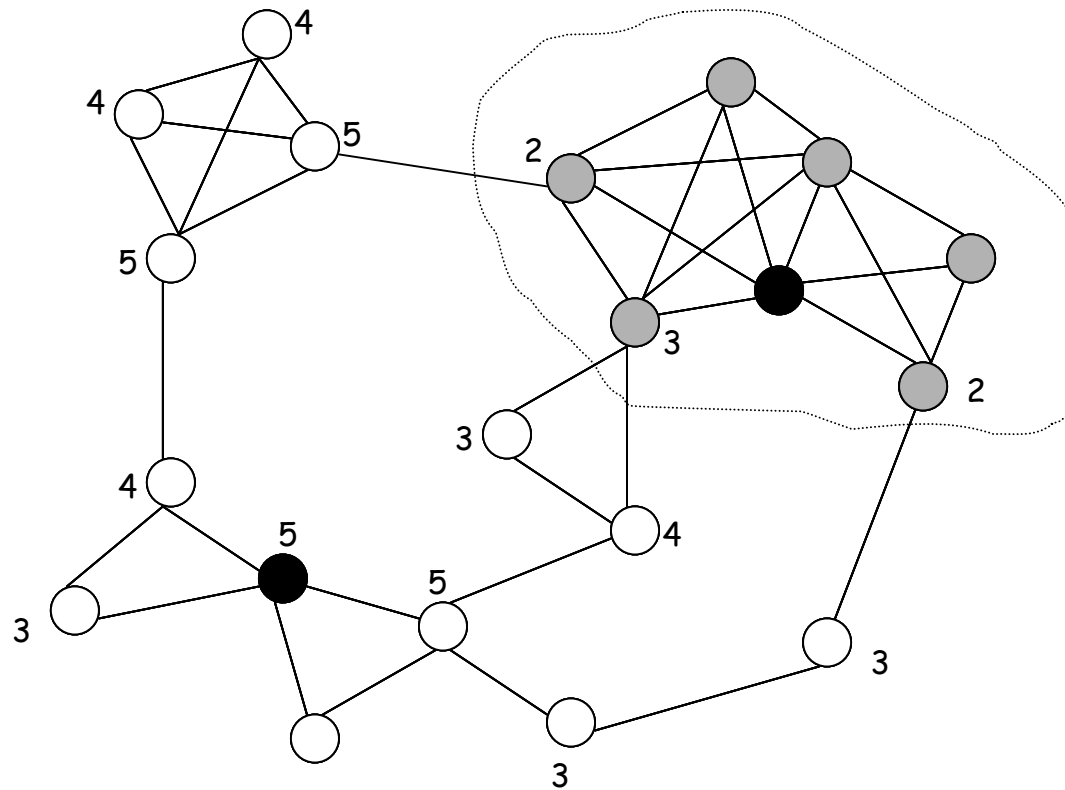


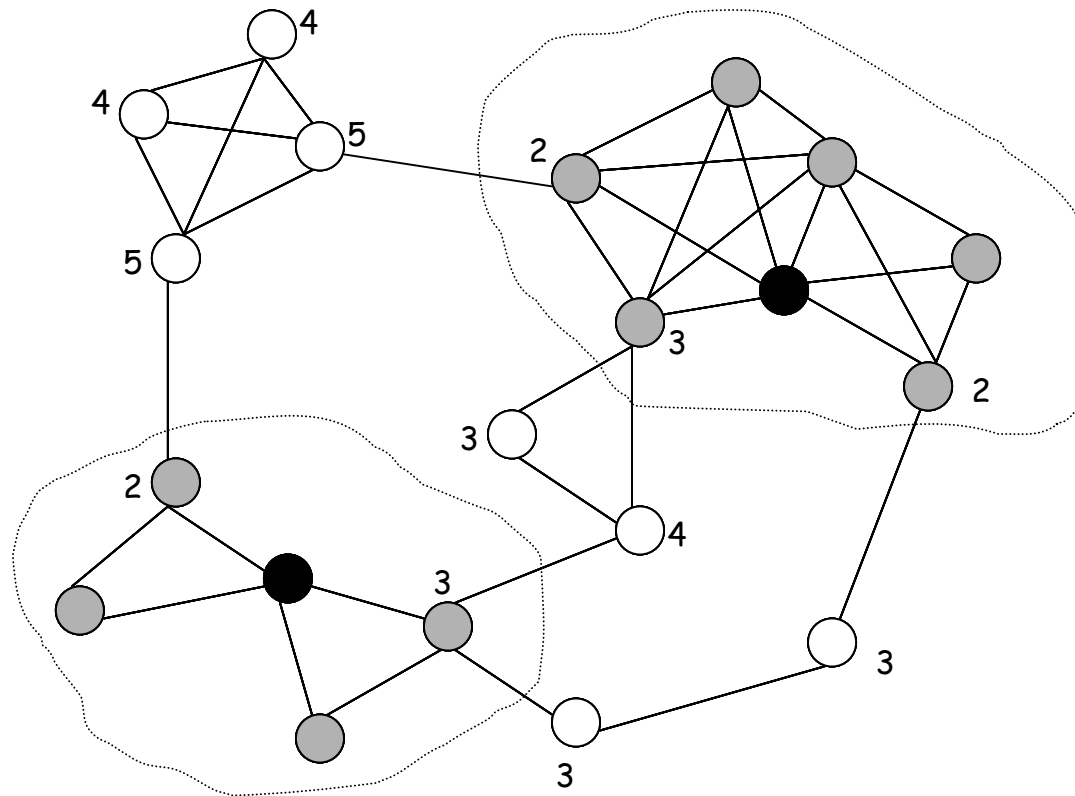
Πρώτη επανάληψη



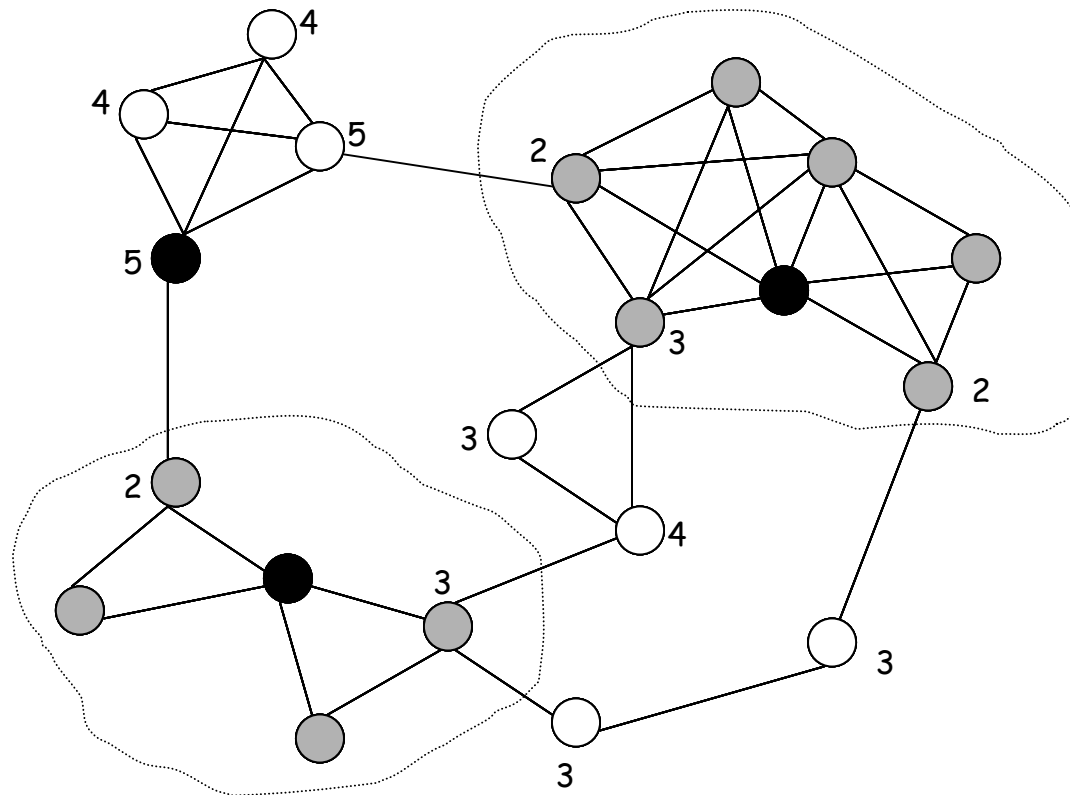


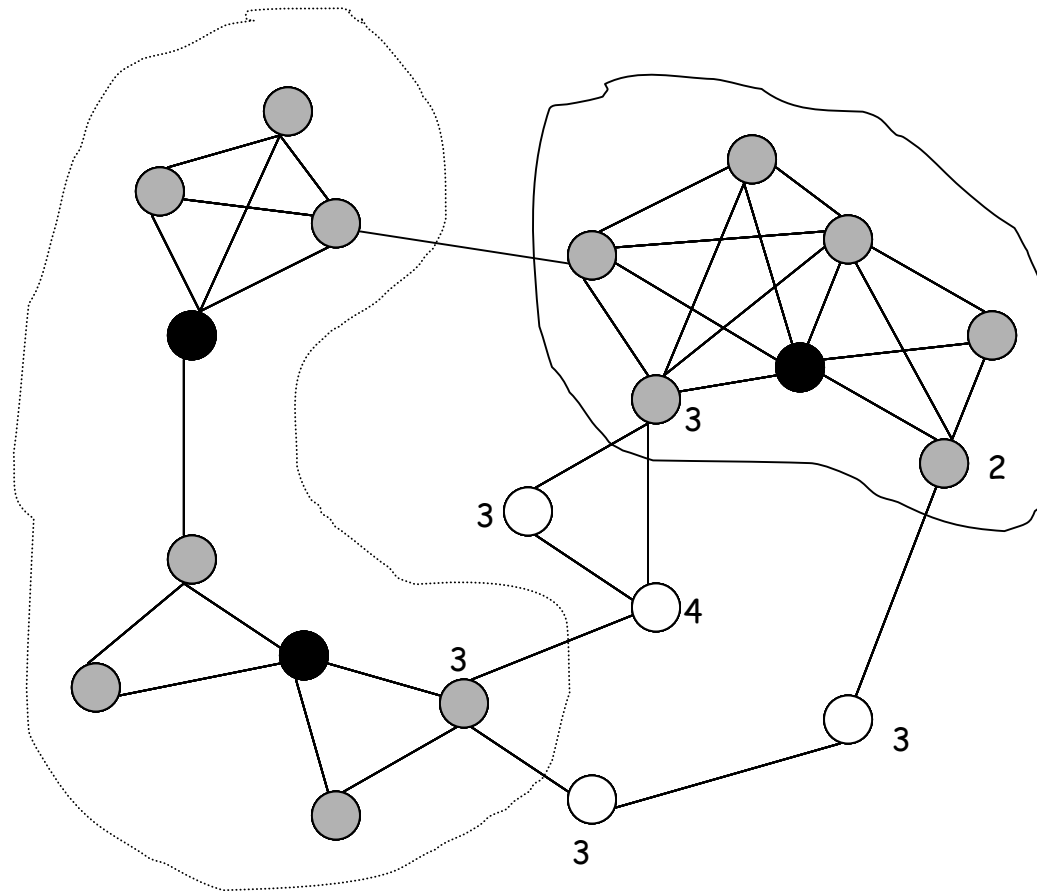
Δεύτερη επανάληψη



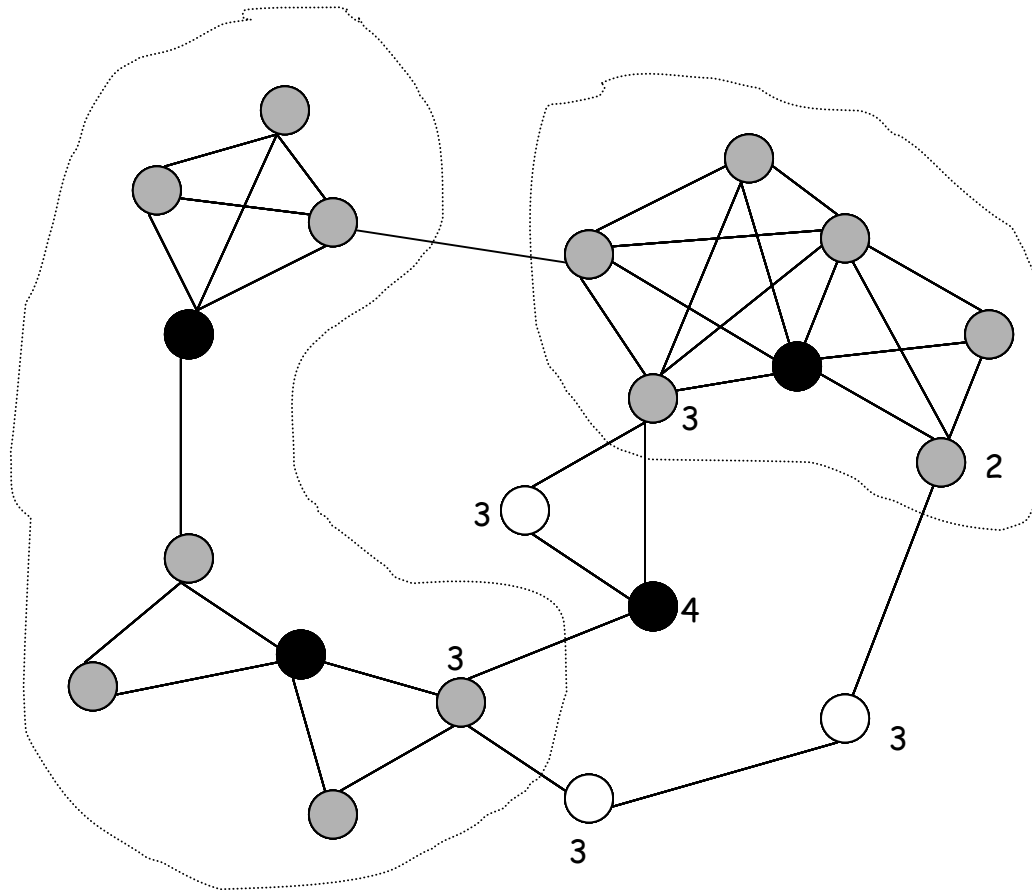


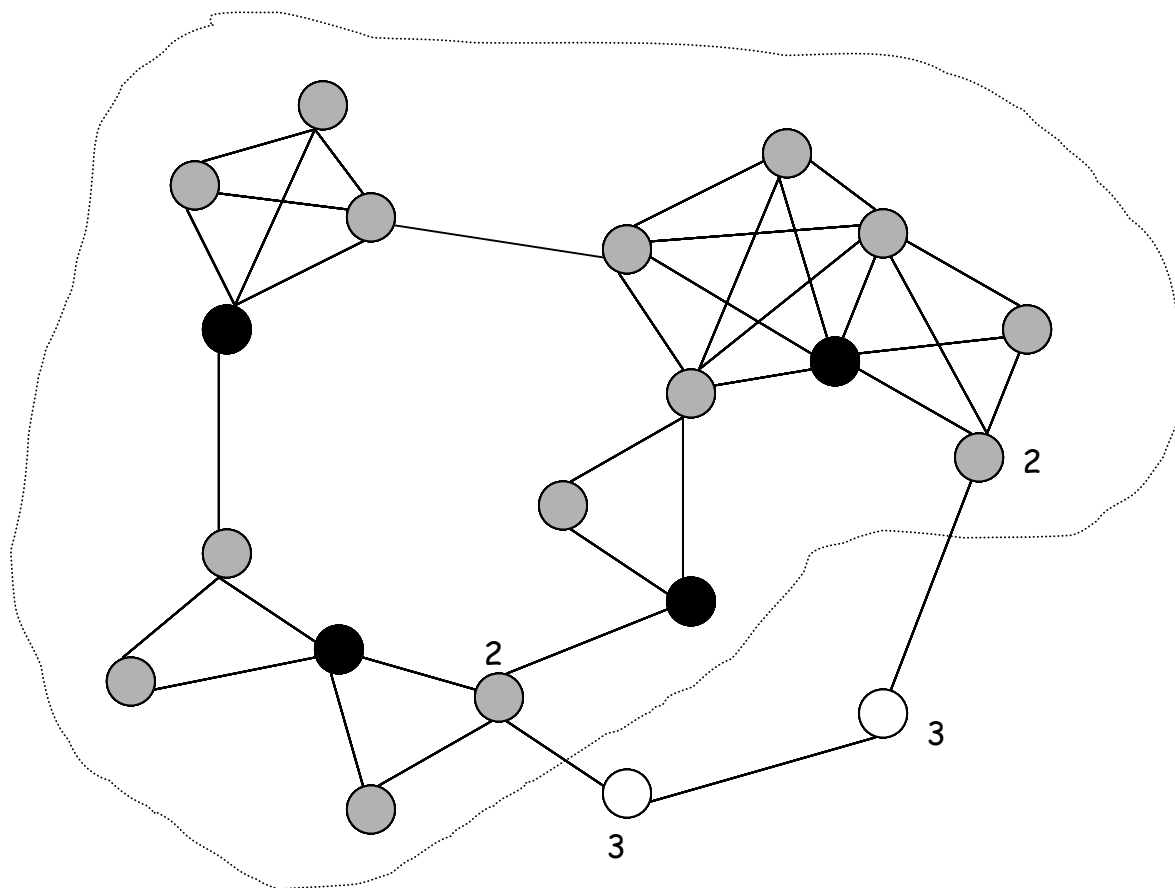
Τρίτη επανάληψη



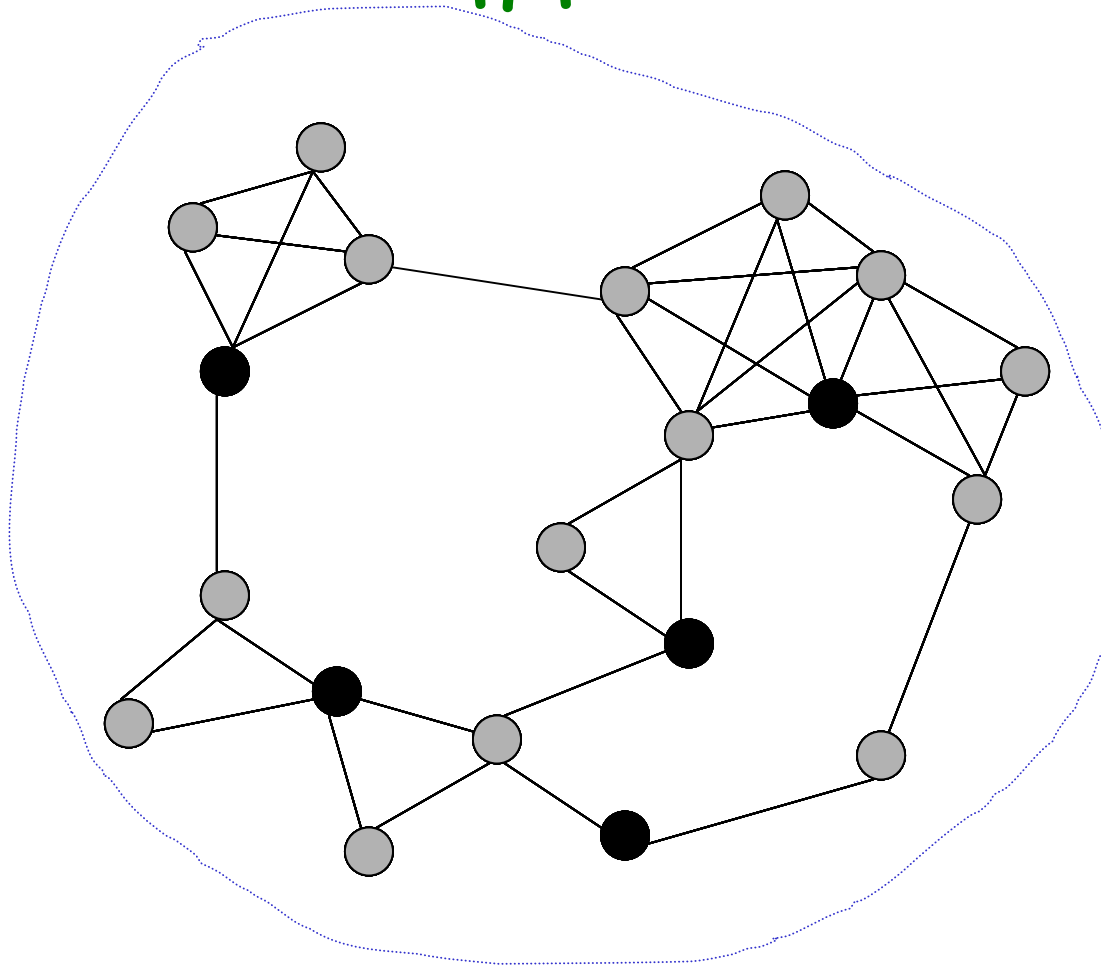


Τέταρτη επανάληψη





Τελευταία επανάληψη



Επίδοση του WCDS

- Το μέγεθος του weakly-connected dominating set που σχηματίζεται από τον Αλγόριθμο I είναι το πολύ $(\ln \Delta + 1) |\text{OPT}|$
- OPT συμβολίζει το ελάχιστο μέγεθος του weakly-connected dominating set για τον G
- Δ συμβολίζει το μέγιστο βαθμό του G

Αλγόριθμος II

- Ο αλγόριθμος ξεκινά με την επιλογή ενός τυχαίου κόμβου του G για να τον χρωματίσει black
- Στις επόμενες επαναλήψεις, επιλέγεται να χρωματιστεί black ο υποψήφιος κόμβος με τον μεγαλύτερο αριθμό white κόμβων στην γειτονιά του
- Το μέγεθος του weakly-connected dominating set είναι το πολύ $(\lg \Delta + 2) |OPT|$

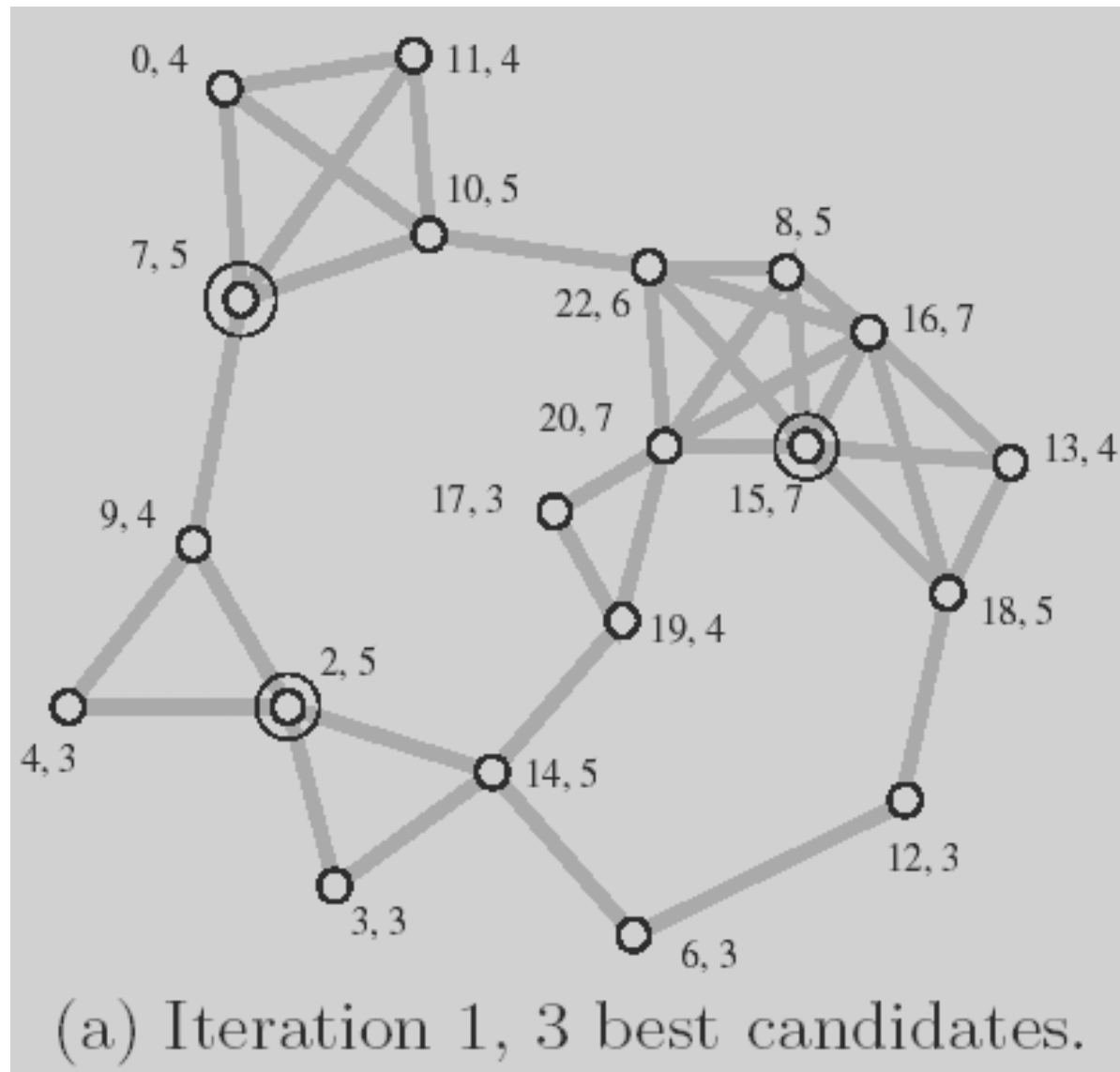
Αλγόριθμοι III και IV

- Σε ad hoc δίκτυα, ο mobile host δεν γνωρίζει τη δομή του δικτύου πέρα από τη γειτονιά της
- Distributed εκδόσεις των Αλγορίθμων I και II
- Εξακολουθούν να είναι κληρονομικά ακολουθιακοί, αφού μόνο ένας κόμβος μπορεί να χρωματιστεί black σε κάθε επανάληψη

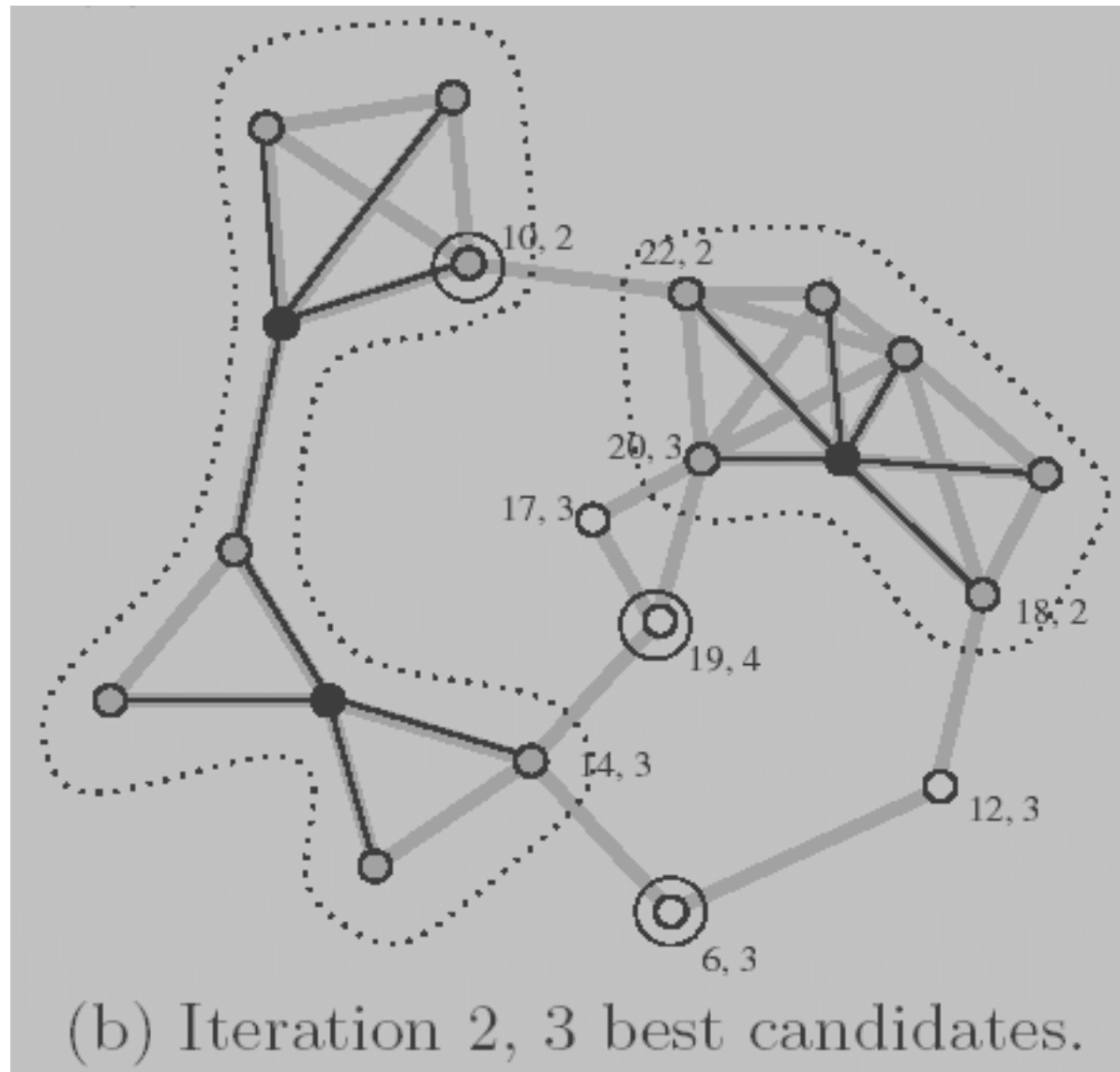
Αλγόριθμος V

- Πλήρως distributed προσέγγιση
- Επέκταση πολλαπλών black pieces παράλληλα
- Σε κάθε επανάληψη, κάθε piece υπολογίζει τους δικούς της υποψήφιους
- Ένας υποψήφιος κόμβος ήταν είτε gray κόμβος ή white κόμβος προσκείμενος σε κάποιο gray κόμβο
- Κάθε piece επιλέγει από τους δικούς της υποψήφιους κόμβους τον υποψήφιο με τη μεγαλύτερη improvement και τον χρωματίζει black καθώς και τους γείτονές τους gray

Πρώτη επανάληψη



Δεύτερη επανάληψη



Περιεχόμενα

- Κινητά Ad Hoc δίκτυα
 - Clustering σε ad hoc δίκτυα
 - Αλγόριθμος με Maximal Independent Sets
 - Αλγόριθμος με Weakly Connected Dominating Sets
 - **Σχηματισμός max-min d-cluster**
- Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks)
 - Διάχυση πληροφορίας σε ασύρματα δίκτυα αισθητήρων

Εισαγωγικά (1/2)

- An ad hoc network may be logically represented as a set of clusters. The clusterheads form a d-hop dominating set. Clusterheads form a virtual backbone and may be used to route packets for nodes in their cluster.
- In this paper, the author shows that the minimum d-hop dominating set problem is NP-complete and then presents a heuristic to form d-clusters in a wireless ad hoc network.

Εισαγωγικά (2/2)

- Besides of the clusterheads, it also propose an efficient algorithm to construct gateway nodes which are at the fringe of a cluster and typically communicate with gateway nodes of other clusters.
- Furthermore, this heuristic has time complexity of $O(d)$ rounds which compares favorably to $O(n)$ for earlier heuristics for large mobile networks. This reduction in time complexity is obtained by increasing the concurrency in communication.

Σχεδιαστικές επιλογές

- All nodes maintain knowledge of the overall network and manage themselves. (high communication overhead)
- Identify a subset of nodes within the network and vest them with the extra responsibility of being a leader (clusterhead) of certain node set in their proximity. (LCA, LCA2, Degree)

Χαρακτηριστικά νέου αλγορίθμου

- No need for synchronized clocks
- Limit the No. of messages sent between nodes to $O(d)$
- Minimize the size of the data structures
- Minimize the number of clusterheads as a function of d
- Formation of backbone using gateways
- Re-elect clusterheads when possible: *stability*
- Control the number of the clusterheads and cluster density by the parameter d
- Distribute responsibility of managing clusters is equally distributed among all nodes: *fairness*

Πολυπλοκότητα των d-clusters

- Reduce from 1-hop cluster problem which is also called dominating problem and has been proven as NP-complete.
- Auxiliary approach: **Construction of the unit disk graph** G . Define $\delta = 1/(2d+1)$ unit as the radius of the unit disk graph G_0 . For each unit length in G we add $(2d+1)$ new intermediate vertices in equal distance δ . Thus, for each *original edge* (u, v) in G of length $l_{u,v}$, we add $(2d+1) \times l_{u,v}$ *intermediate vertices*. Moreover we add $(2d+1)$ auxiliary vertices $u_1, u_2, \dots, u_{d-1}$ sequentially from origin vertex u at each distance d .

Ευρεστική επίλυση (1/6)

- The heuristic runs for $2d$ rounds of information exchange. Each node maintains two arrays, WINNER and SENDER, each of size $2d$ node ids: one id per round of information exchange.
- Step1: Initially, each node sets its WINNER to be equal to its own node id.
- Step2: (Floodmax) - Each node locally broadcasts its WINNER value to all of its 1-hop neighbors. For a single round, the node chooses the largest value among its own WINNER value and the values received in the round as its new WINNER. This process continues for d rounds.

Ευρεστική επίλυση (2/6)

- Step3: This follows Floodmax and also lasts d rounds. It is the same as Floodmax except a node chooses the smallest rather than the largest value as its new WINNER.
- Step4 (overtake): At the end of each flooding round a node decides to maintain its current WINNER value or change to a value that was received in the previous flood round.
- Step5 (node pair): A node pair is any node id that occurs at least once as a WINNER in both the 1st (Floodmax) and 2nd (Floodmin) d rounds of flooding for an individual node.

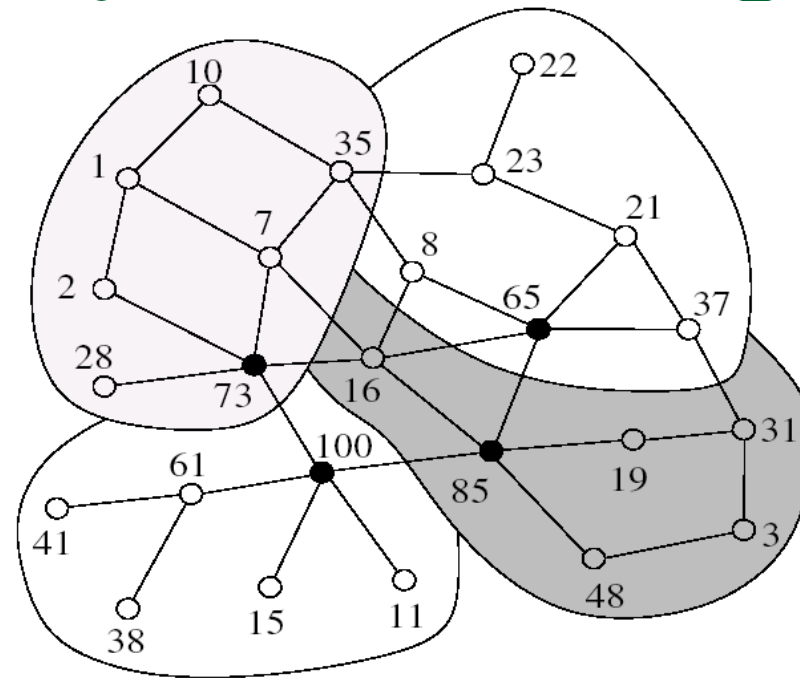
Ευρεστική επίλυση (3/6): Κριτήρια επιλογής των clusterheads

- After completion of the 2nd d rounds each node looks at its logged entries for the 2d rounds of flooding. The following rules explain the logical steps of the heuristic that each node runs on the logged entries.
- *Rule 1:* First, each node checks to see if it has received its own original node id in the 2nd d rounds of flooding. If it has then it can declare itself a clusterhead and skip the rest of this phase of the heuristic.
- *Rule 2:* Each node looks for node pairs. Once a node has identified all node pairs, it selects the minimum node pair to be the clusterhead. If a node pair does not exist for a node then proceed to Rule 3.
- *Rule 3:* Elect the maximum node id in the 1st d rounds of flooding as the clusterhead for this node.

Ευρεστική επίλυση (4/6): Επιλογή των gateways και convergecast

- To reduce overhead, the communication starts from the fringes of the cluster, gateway nodes, inward to the clusterhead.
- If some nodes of a node's neighbors have chosen different clusterhead, then the node is a gateway node. (1-hop local broadcast)
- The SENDER data structure is used to determine who next to send the convergecast message.
- The heuristic maximizes the number of gateways resulting in a backbone with multiple paths between neighboring clusterheads. (for reliability)

Ευρεστική επίλυση (5/6): Επιλογή των gateways και convergecast



Node	10	1	2	7	35	8	23	22	21	65	37	31	19	85	16	100	73	28	41	61	11	48	3	15	38
Max 1	35	10	73	73	35	65	35	23	65	85	65	37	85	100	85	100	100	73	61	100	100	85	48	100	61
Max 2	35	73	100	100	73	85	65	35	85	100	85	85	100	100	100	100	100	100	100	100	100	100	85	100	100
Max 3	73	100	100	100	100	100	85	65	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
Min 1	73	73	100	100	73	100	65	65	85	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
Min 2	73	73	73	73	65	73	65	65	65	85	85	100	100	100	100	100	100	100	100	100	100	100	100	100	100
Min 3	65	73	73	65	65	65	65	65	65	65	65	85	100	85	73	100	73	100	100	100	100	100	100	100	100
Result	73	73	73	73	73	65	65	65	65	65	65	85	100	85	100	100	73	100	100	100	100	100	100	100	100

WINNER

3-cluster formation in a network of 25 nodes.

Ευρεστική επίλυση (6/6): Ορθότητα

- **Assumption 1:** During the floodmin and floodmax algorithms no node's id will propagate farther than d-hops from the originating node itself (definition of flooding).
- **Assumption 2:** All nodes that survive the floodmax elect themselves clusterheads.
- **Lemma 1:** If node A elects node B as its clusterhead, then node B becomes a clusterhead.

Παράδειγμα

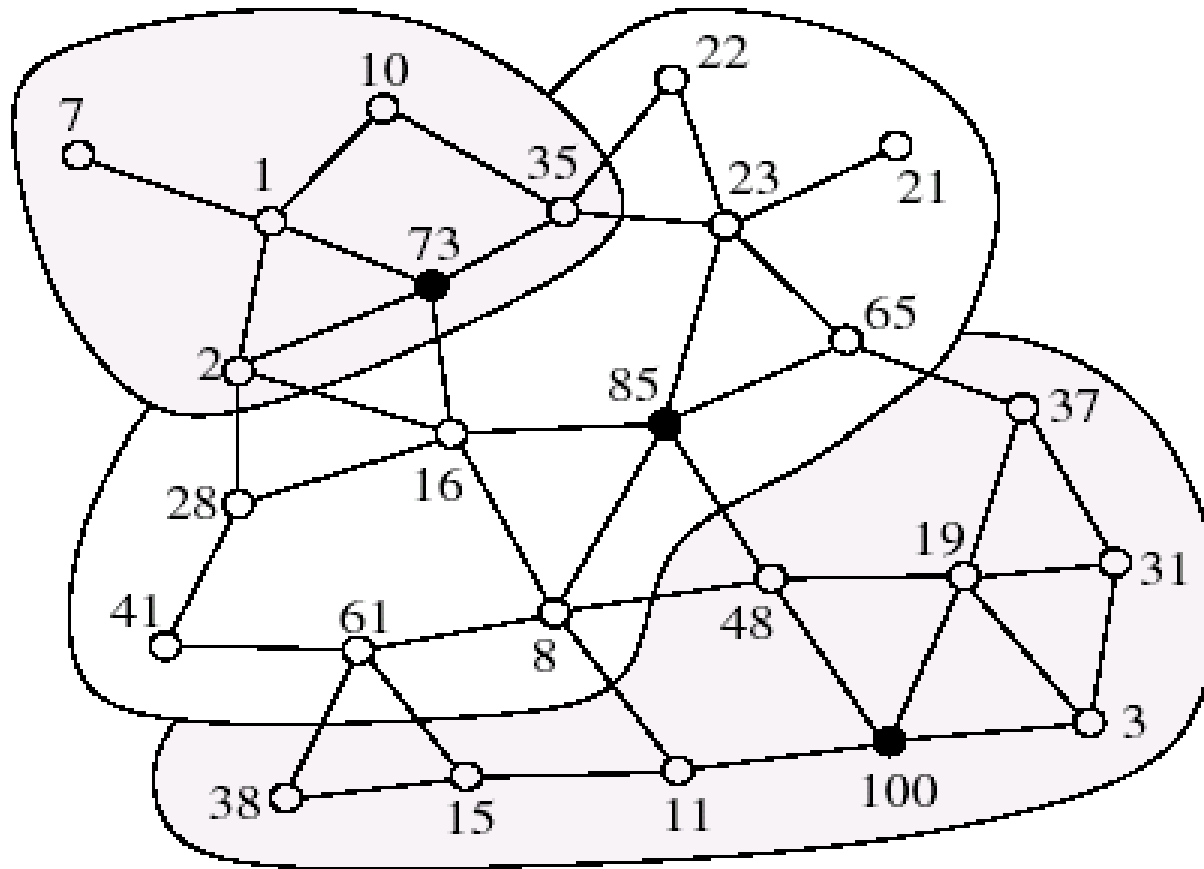
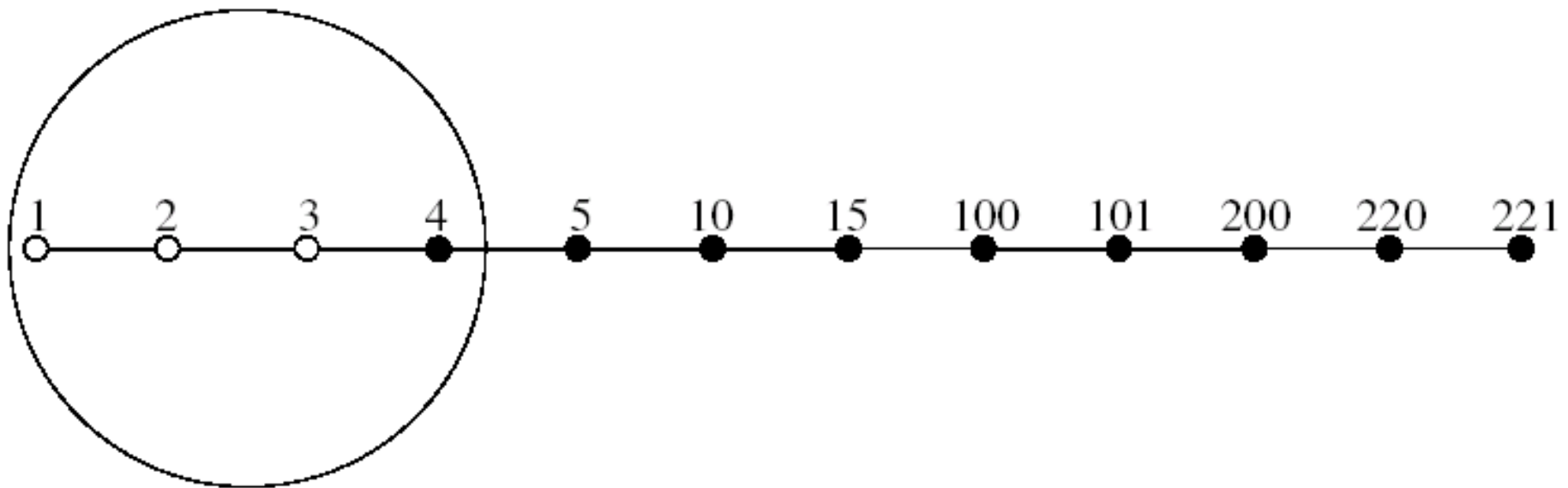


Figure 5 shows an example of the network topology generated by the heuristic with 25 nodes. Here we see four clusterheads elected in close proximity with one another, namely nodes 65, 73, 85, and 100.

3-cluster formation after topology change.

Παράδειγμα (χειρότερη επίδοση)



Worst case performance scenario for the proposed heuristic, $d=3$.

Παράδειγμα (πολυπλοκότητα)

- Since no node is more than d hops from its clusterhead the convergecast will be $O(d)$ rounds of messages. Therefore, the time complexity of the heuristic is $O(2d + d)$ rounds = $O(d)$ rounds.
- Each node has to maintain $2d$ node ids in its WINNER data structure, and the same number of node ids in its SENDER data structure. Thus, the storage complexity is $O(d)$.

Περιεχόμενα

- Κινητά Ad Hoc δίκτυα
 - Clustering σε ad hoc δίκτυα
 - Αλγόριθμος με Maximal Independent Sets
 - Αλγόριθμος με Weakly Connected Dominating Sets
 - Σχηματισμός max-min d-cluster
- **Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks)**
 - Διάχυση πληροφορίας σε ασύρματα δίκτυα αισθητήρων

Κατηγορίες δικτύων αισθητήρων

- Εφαρμογές
- Κινητά vs. στατικά
 - Υβριδικά
- Ομογενή vs. ετερογενή
 - Sensors vs. relays
- Επίπεδα vs. ιεραρχικά
- Αριθμός και θέση των sinks
- Τύπος μέσου επικοινωνίας
 - Ενσύρματα?
 - Ασύρματα RF, οπτικά, ακουστικά
 - Υβριδικά

Εφαρμογές

□ Ecological Habitat Monitoring

- UCB/Intel Berkeley: Great Duck Island
- UCLA-CENS: James Reserve
- Princeton: ZebraNet in Kenya



□ Structural Monitoring

- UCLA-CENS: Factor Building
- USC: Networked SHM
- UCB/Intel Berkeley: SF Golden Gate Bridge



○ Biomedical Applications

- Artificial retina
- "Bio-monitors"



□ Industrial and Commercial Apps

- Ember Corp: Thermal Process Control, Shipment Tracking

ember
embedded RF



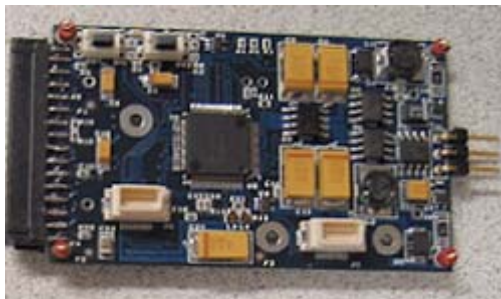
Κόμβοι δικτύων αισθητήρων



WINS (Rockwell)



MICA 2 Mote (Berkeley)

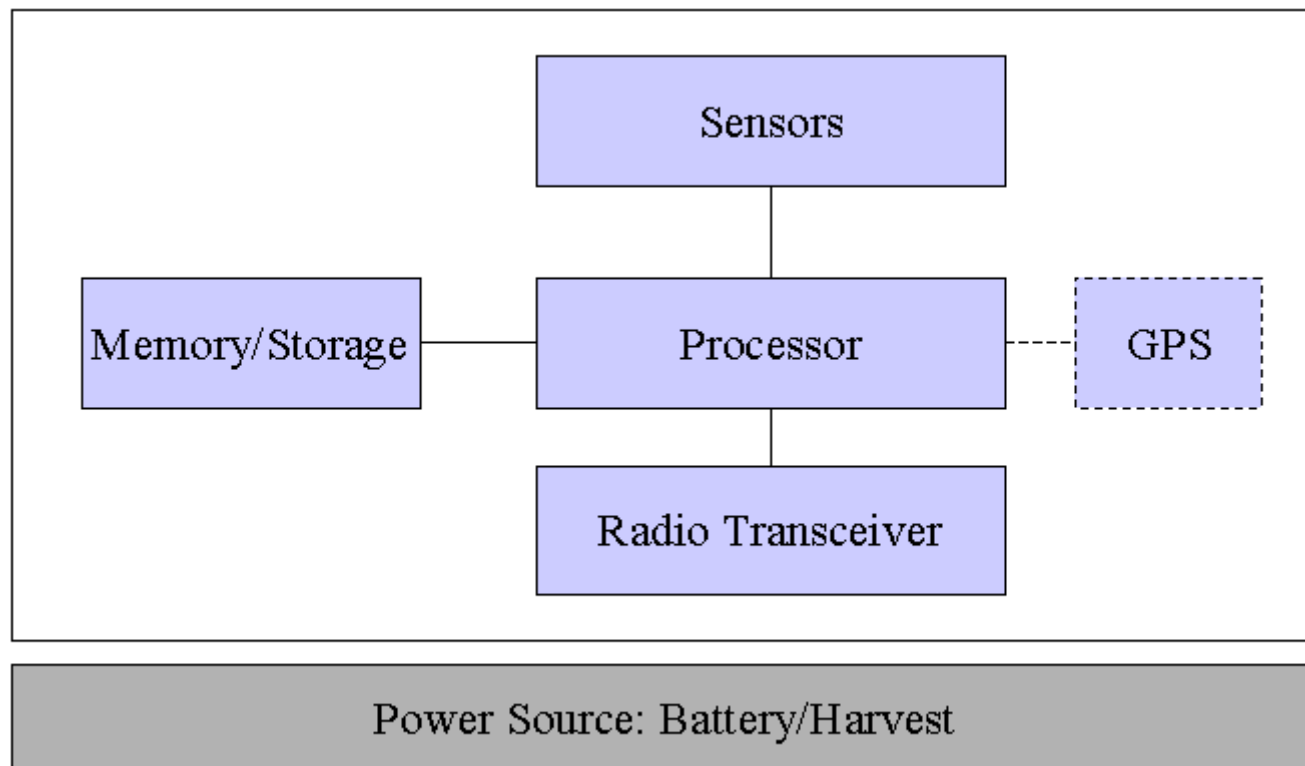


GNOMES (Rice)



MANTIS Nymph (Colorado)

Βασικό υλικό (hardware)



Berkeley Motes



- ❑ MICA 2.
- ❑ Runs TinyOS, programmed using NesC
- ❑ ATMEL Atmega 128 Processor ATmega 128L 8-bit, 8MHz, 4KB EEPROM, 4KB RAM, 128KB flash
- ❑ Chipcon CC100 multichannel radio 38.4 Kbaud, (Manchester encoding, FSK), 500-1000ft max range.
- ❑ Standalone sensor boards
- ❑ TOSSIM software to simulate a mote-based sensor network*

* Συνιστάται ανεπιφύλακτα για λεπτομερή έργα προσομοίωσης!

Εφαρμογή στο Great Duck Island

- ❑ Petrel habitat on Great Duck Island in Maine.
- ❑ Ερωτήσεις προς απάντηση:
 - pattern χρήσης των nesting burrows για τον κύκλο 24-72 ωρών
 - Αλλαγές στην burrow και επιφανειακών περιβαλλοντικών παραμέτρων
 - Διαφορές στα micro-περιβάλλοντα με και χωρίς μεγάλους αριθμούς από nesting petrels
 - Ρυθμοί δειγματοληψίας

Πλεονεκτήματα δικτύων αισθητήρων

- Απόλυτη σύνδεση με το περιβάλλον
- Όχι-επιθετικά
- Ασφαλή
- Οικονομικά
 - Μια εγκατάσταση, πολλαπλές χρήσεις

Κόμβοι

- Αισθητήρες:
 - Mica 1 mote.
 - Sensor board με θερμοκρασία, φως, πίεση, υγρασία, θερμότητα.
- Gateways:
 - Mica 1 mote συνδεδεμένη σε ένα mote και στον σταθμό βάσης
 - Εφοδιασμένα με directional antennae

Ενέργεια

- Περιορισμένος χρόνος ζωής της μπαταρίας
- Ομοιόμορφη λειτουργία για 9 μήνες
- Οι κόμβοι εκτελούν διαφορετικές λειτουργίες
- Συγκεκριμένο (tabulated) κόστος για τις βασικές λειτουργίες
- Κατάσταση ύπνου (sleep mode)

Ζητήματα εγκατάστασης

- 32 motes, 9 in burrows
- Recorded- και live data
- Duty-cycle based broadcast
- Multi-hop
- Network management
 - Re-tasking
 - Fault detection/diagnosis

Έξυπνη σκόνη (Smart Dust)

- Σχεδιαστικοί στόχοι
 - Cubic millimeter.
 - Πολύ χαμηλή κατανάλωση ενέργειας

Αποτέλεσμα: το πακέτο αισθητήρα περιέχει

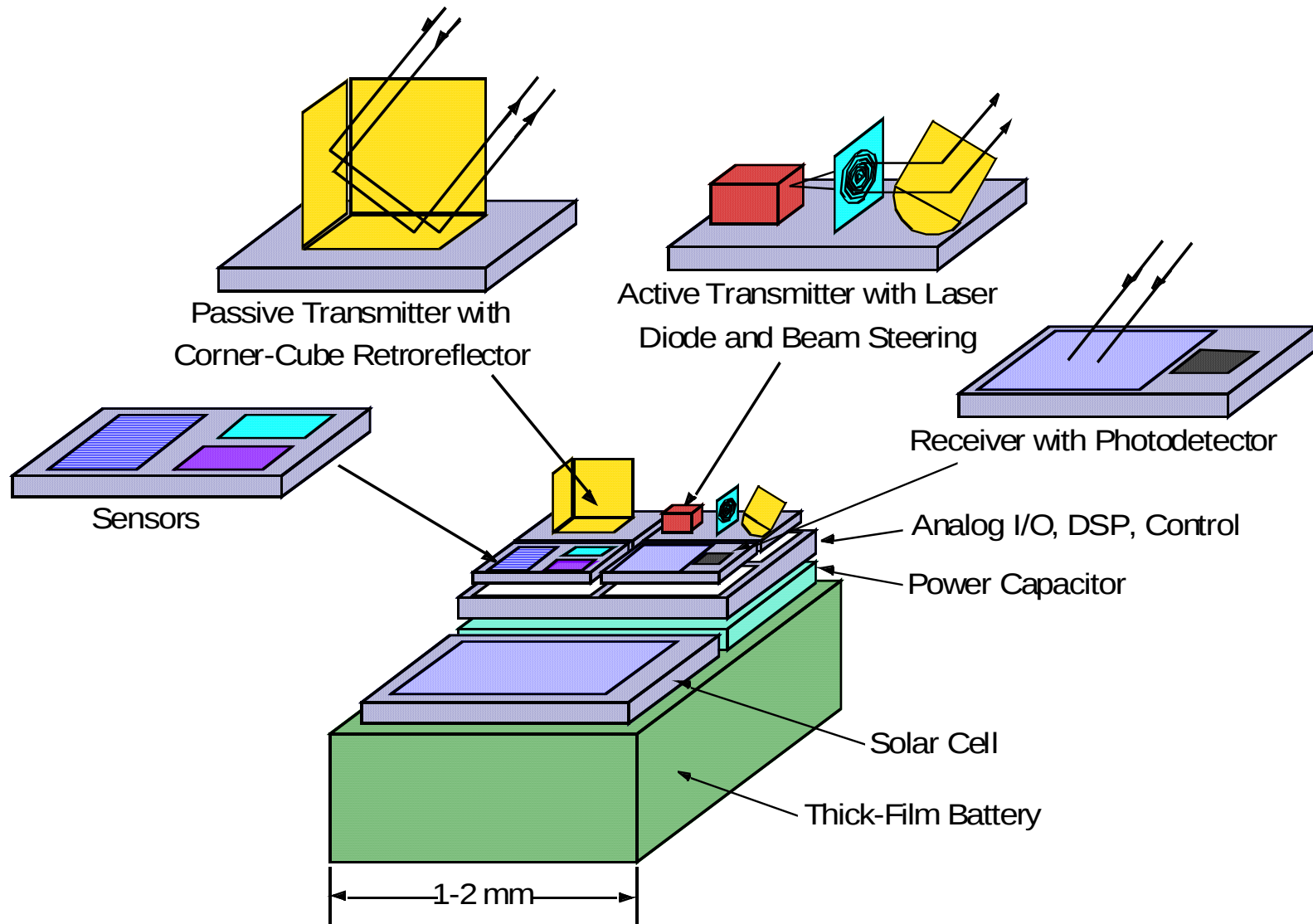
- Αισθητήρες
- Οπτικό αναμεταδότη (passive και active) και λήπτη (receiver)
- Επεξεργασία σήματος
- Ηλιακή πηγή ενέργειας

Εφαρμογές της smart dust

- Παρακολούθηση περιβάλλοντος
 - Έντομα
 - Μετεωρολογικά φαινόμενα
- Ειδικές λειτουργίες



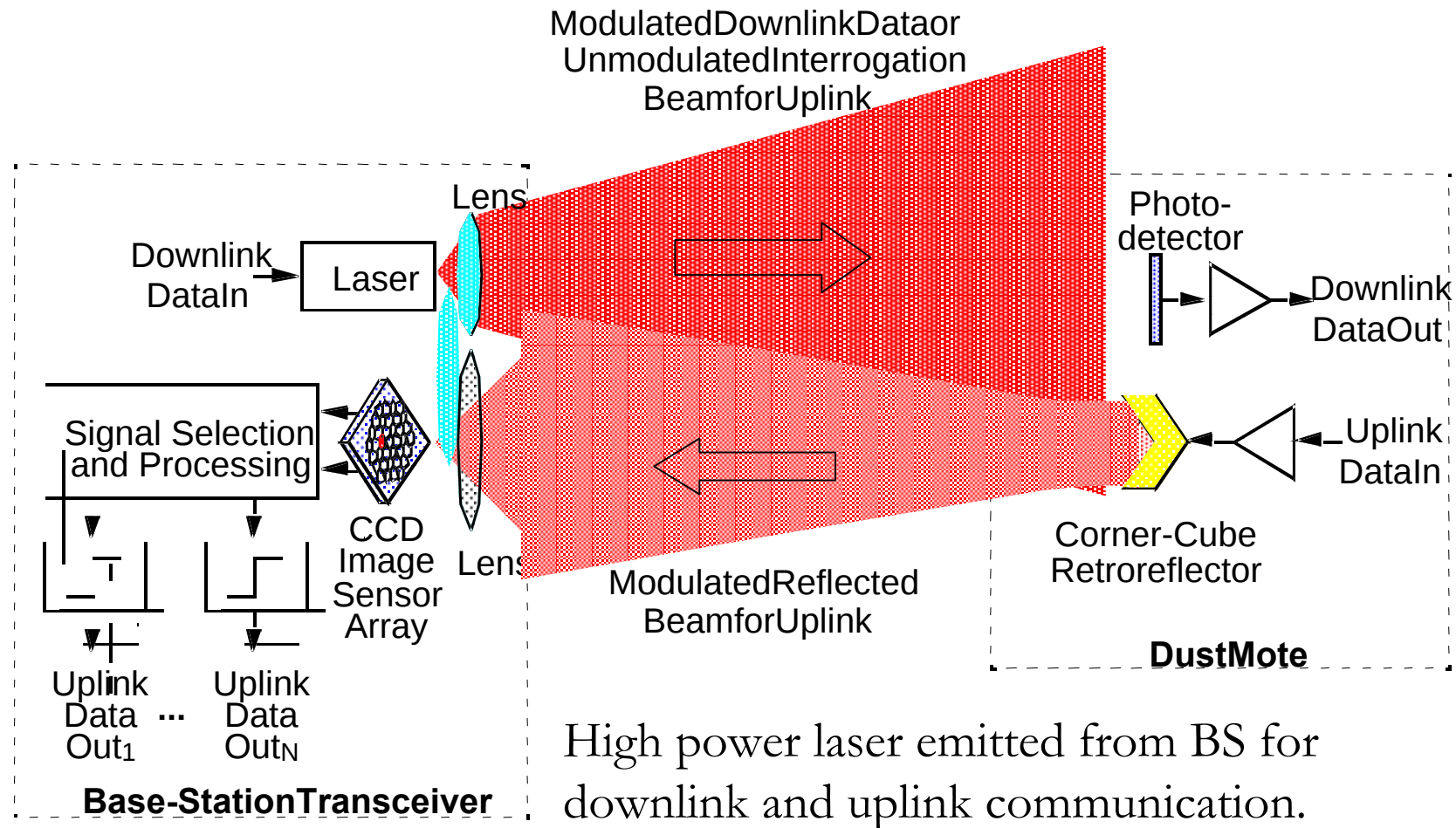
Συνιστώσες της smart dust



Passive μεταδόσεις

- Ανάκλαση φωτεινής ακτίνας (από το BS) πίσω στο σταθμό με κωδικοποιημένα δεδομένα
- Ο BS αποκωδικοποιεί τα δεδομένα με ανάγνωση των “on” και “off” ανακλάσεων
- Ρυθμοί μέχρι 1 kbps για 150m
- Αλλά, uninterrupted LoS με BS

Smart dust: passive transmission



Active transmissions

- Laser
- Multi-hop για μικρή ενέργεια
- Data rate vs. range vs. power trade-off

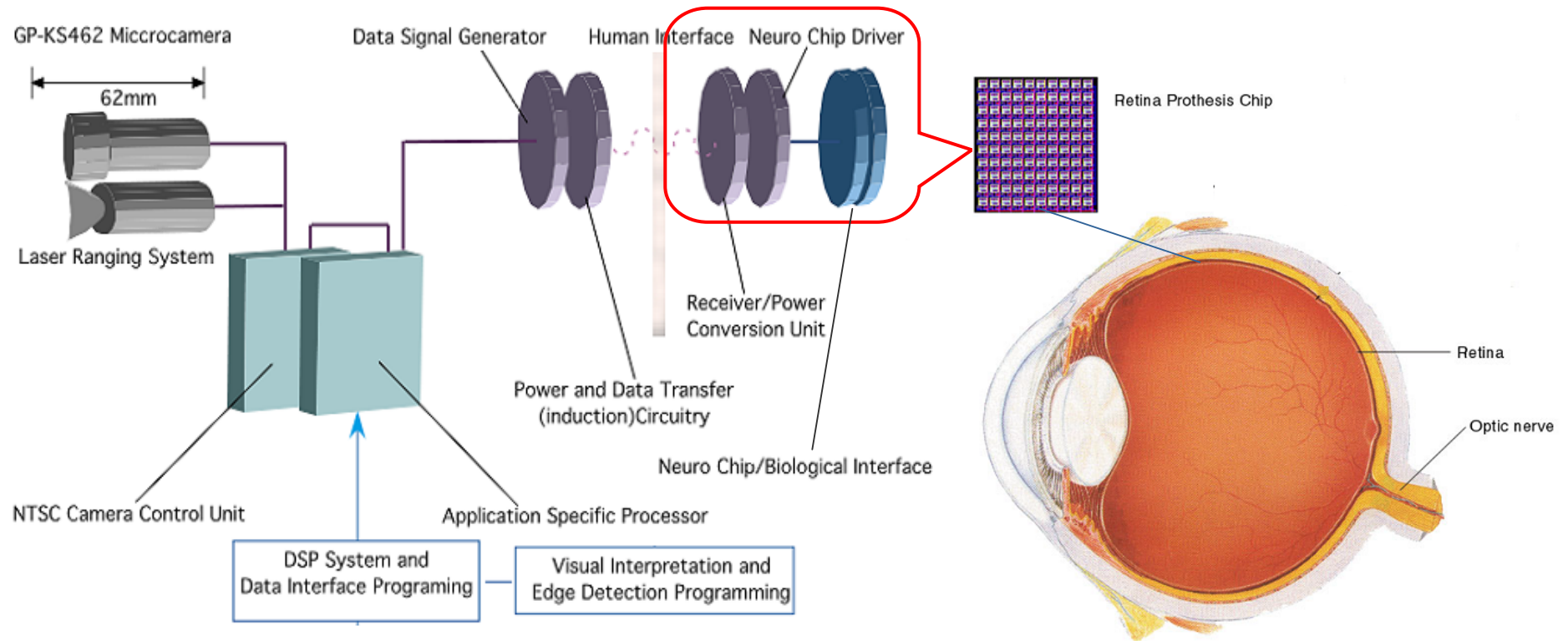
Smart dust: προκλήσεις

- Απαίτηση LoS vs. κινητικότητα
- Κατεύθυνση σύνδεσης
- Ενέργεια, ρυθμός δεδομένων, και απόσταση trade-off
- Link unidirectionality και ασυμμετρία
- Τοπικότητα

Βιοϊατρικές εφαρμογές

- Παρατήρηση υγείας
 - Επίπεδο γλυκόζης
 - Πεπτικό σύστημα
 - Μυϊκό σύστημα
 - Καρδιαγγειακό σύστημα, κ.τ.λ.
- Τεχνητός αμφιβληστροειδής

Αισθητήρες για την όραση



Συνοπτικά οι προκλήσεις

- Περιορισμένες δυνατότητες: ενέργεια, επεξεργασία, αποθήκευση, και επικοινωνία
- Συνεχής λειτουργία
- Ευρωστία και fault tolerance
- Κλιμάκωση
- Αυτό-διαμόρφωση, αυτό-διαχείριση, αυτό-επιδιόρθωση
- Ζητήματα σχετικά με τα δεδομένα

Ζητήματα ανά εφαρμογή

- Περιορισμοί υλικών
 - Βιο-συμβατότητα
 - Δυσδιάκριτα
 - Μιμούμενα το περιβάλλον
 - Μη-ανιχνεύσιμα: π.χ. stealth πτήση
- Ασφάλεια
 - Ιδιωτικότητα
 - Interference
- Ζητήματα νομοθεσίας
 - Όπως απαιτήσεις της FDA

Περιορισμένη υπολογιστική και αποθηκευτική ισχύς

- Συνεργασία μεταξύ κόμβων
- Συσσώρευση δεδομένων (data aggregation)

Περιορισμένη ενέργεια

- Χαμηλής ενέργειας λειτουργικές συνιστώσες
- Διαχείριση ενέργειας
 - Διάφορες λειτουργικές καταστάσεις
 - Χαμηλό κόστος μετάβασης από-σε κατάσταση
 - Deep-sleep, Sleep, On
 - Παροχή διαφορετικής QoS
- Διαχείριση ενέργειας
 - Μέτρηση ενέργειας
 - Κατανομή του προϋπολογισμού ενέργειας
 - Μεταβάσεις μεταξύ διαφορετικών καταστάσεων ενέργειας

Ασύρματη επικοινωνία

- Μέσο επικοινωνίας
 - Radio Frequency: παρακολούθηση “συνηθειών”, βιοϊατρικοί αισθητήρες, κ.τ.λ.
 - Φως (active και passive): Smart Dust.
 - **Acoustic**: υποθαλάσσια δίκτυα αισθητήρων
- Ad hoc vs. αρχιτεκτονική με υποδομή
- Έλεγχος τοπολογίας
- Δρομολόγηση

Ad hoc vs. infrastructure επικοινωνία

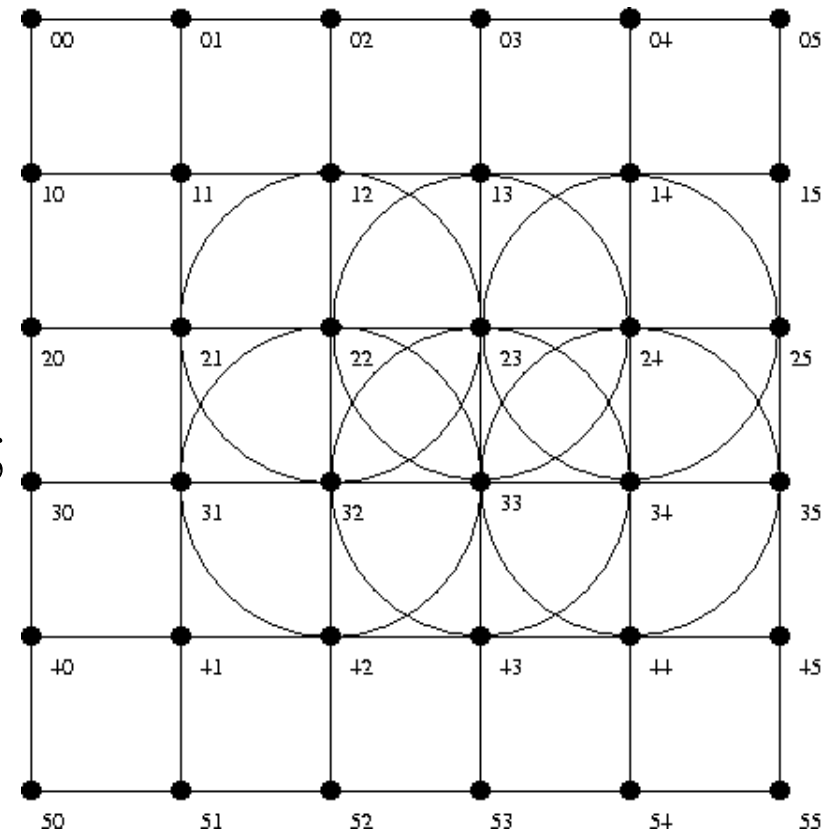
- Επικοινωνία από αισθητήρα σε αισθητήρα:
 - Μικρή απόσταση
 - Ad hoc
- Επικοινωνία από αισθητήρα με σταθμό βάσης:
 - Μακρινής απόστασης επικοινωνία από αισθητήρα προς το σταθμό βάσης
 - Υποδομή

Τοπολογία

- Σταθερή τοπολογία
 - Βασισμένη σε tree
 - Βασισμένη σε clusters
- Δυναμική τοπολογία - κινητικότητα
 - Ad hoc
 - Υποδομή
 - Μεικτή

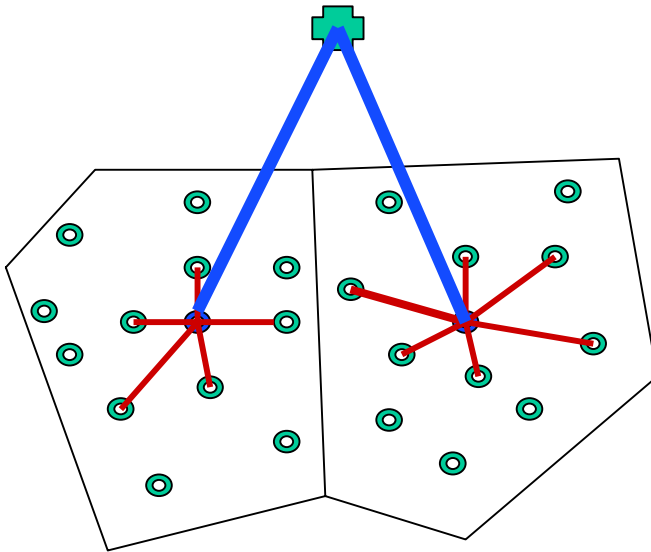
Σταθερές τοπολογίες (1/2)

- Μεταβολή # γειτόνων
- Trade-offs
 - Αριθμός των hops
 - Αριθμός των γειτόνων
- Εκτίμηση χρήσης ενέργειας
- Έλεγχος power-aware δρομολόγησης
- **Αποτελέσματα:**
 - Power-aware routing reduces power usage.
 - 3D είναι καλύτερα από 2D

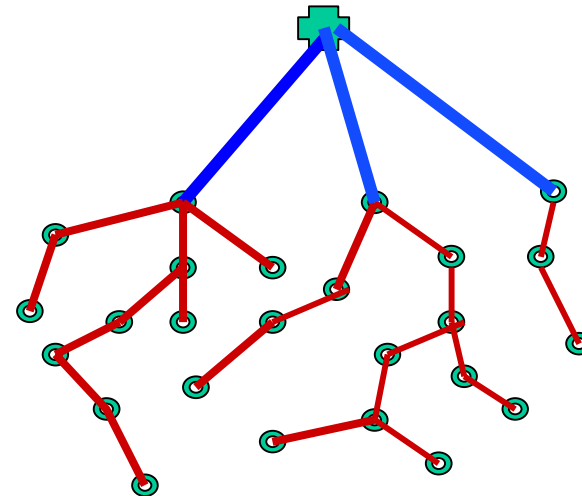


Σταθερές τοπολογίες (2/2)

Cluster-based



Tree-based



Οι προσεγγίσεις βασισμένες σε clusters παρέχουν καλύτερη αποδοτικότητα σε ενέργεια από ότι οι προσεγγίσεις βασισμένες σε tree

Ζητήματα σχετικά με τα δεδομένα

- Trade-off μεταξύ καθυστέρησης (latency) και ενέργειας
- Αναπαράσταση δεδομένων
 - Raw/συμπιεσμένα δεδομένα
 - Τιμή δειγματοληψίας: Απόλυτη/Σχετική
- Διακρίβωση σφάλματος
 - Όχι πρόσβαση στις πραγματικές τιμές
 - Προκύπτουν από άλλους αισθητήρες

Συνεχής λειτουργία

- Συλλογή δεδομένων για μεγάλα χρονικά διαστήματα
- Ανανεώσιμη ενέργεια
 - Ηλιακή ενέργεια
 - Μηχανικές ταλαντώσεις
 - Radio-Frequency επαγωγική αντίσταση
 - Infrared επαγωγική αντίσταση

Ευρωστία και fault tolerance

- Αυτό-προσαρμοζόμενοι αισθητήρες:
 - Προσαρμογή σε αλλαγές περιβάλλοντος
 - Προσαρμογή σε αλλαγές ενέργειας
- Κατανεμημένο δίκτυο:
 - Κάθε αισθητήρας λειτουργεί αυτόνομα από τους γείτονές του
 - Αλληλο-επικαλυπτόμενες περιοχές εξυπηρέτησης
 - Όχι μόνο σημείο αποτυχίας
- Παρακολούθηση κατάστασης “υγείας”
 - Π.χ., αναφορά ενέργειας κατά τη διάρκεια μετάδοσης δεδομένων

Επιφυλάξεις

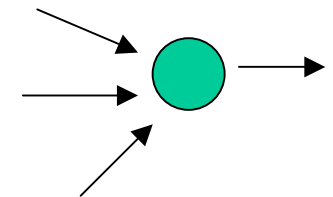
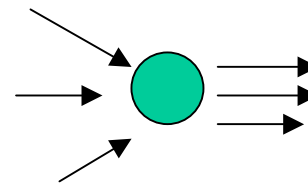
- Υπάρχει μοναδική λύση κατάλληλη για όλες τις εφαρμογές?
- Τα πιο σημαντικά ζητήματα στη σχεδίαση:
 - Κόστος?
 - Κατανομή πόρων?
 - Διαχειρισιμότητα?
 - Εγκαιρότητα?
 - Αναπροσαρμογή λειτουργίας?
 - ...
- Κλιμάκωση?
 - Εκατομμύρια κόμβων αισθητήρων?

Περιεχόμενα

- Κινητά Ad Hoc δίκτυα
 - Clustering σε ad hoc δίκτυα
 - Αλγόριθμος με Maximal Independent Sets
 - Αλγόριθμος με Weakly Connected Dominating Sets
 - Σχηματισμός max-min d-cluster
- Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks)
 - **Διάχυση πληροφορίας σε ασύρματα δίκτυα αισθητήρων: Το πρωτόκολλο Directed Diffusion**

Το υπολογιστικό παράδειγμα

- Δεδομενο-κεντρικό (Data Centric)
 - Το δίκτυο αισθητήρων επερωτάται για συγκεκριμένα δεδομένα
 - No sensor-specific query
 - Ταυτότητα της πηγής των δεδομένων είναι άσχετη
- Εξειδικευμένο για κάθε εφαρμογή
 - In-sensor επεξεργασία
 - In-sensor caching
- Τοπικοί (Localized) αλγόριθμοι
 - Επιτυγχάνουν τον καθολικό σκοπό μέσα από την τοπική συνεργασία και συντονισμό



Ονοματισμός δεδομένων

- Ονομασία με βάση το περιεχόμενο (Content based naming)
 - “Εργασίες” έχουν όνομα: ζεύγη από (Attribute – value)
 - Επιλογή του σχήματος ονοματισμού είναι σημαντική
 - Όχι καθολικά (μοναδικά) ID κόμβων: μόνο τοπικά μοναδικά

Request: Interest

type = four-legged animal
interval = 20 ms
duration = 10 seconds
rect = [-100,100,200,200]

Reply: Data

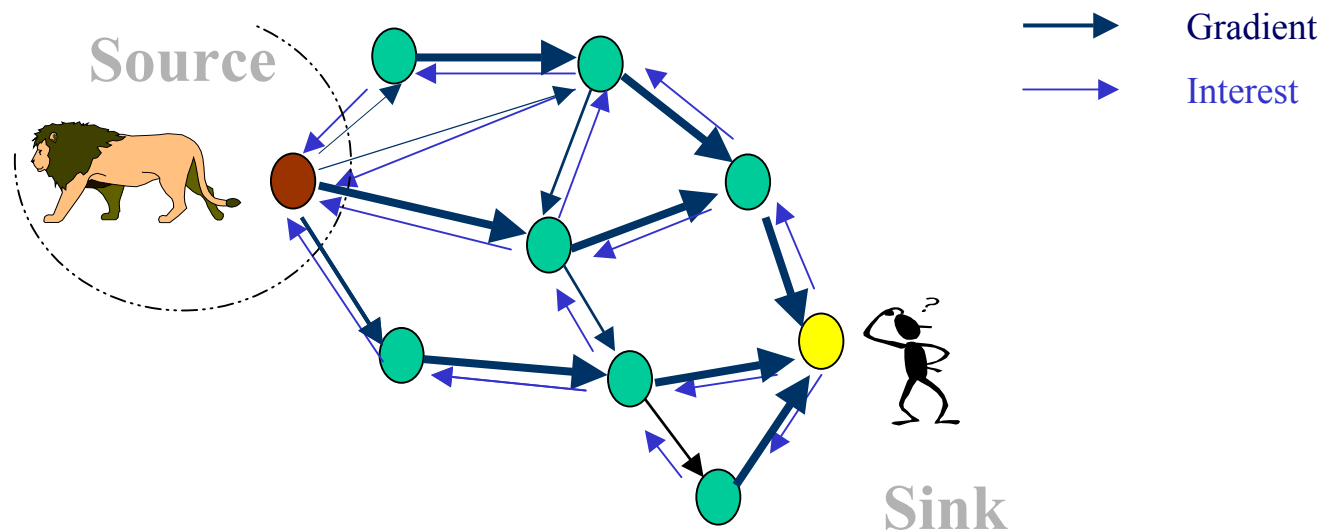
type = four-legged animal
interval = 1s
rect = [-100,100,200,200]
timestamp = 01:20:40
expiresAt = 01:30:40

Interest & Gradient

- **Interest** περιγράφει μια ενέργεια που πρέπει να έρθει σε πέρας από το δίκτυο αισθητήρων
 - Τα interests εγχέονται στο δίκτυο από το **sink**
 - Το sink εκπέμπει (broadcasts) ένα interest
 - Το **interval** καθορίζει ένα ρυθμό για το event data
 - Αρχικά, το αιτούμενο interval είναι πολύ μεγαλύτερο από ότι χρειάζεται
 - Κάθε κόμβος διατηρεί μια “**interest cache**”
- Κάθε interest entry διατηρεί **gradients**
 - Καθορίζει το ρυθμό δεδομένων και μια **direction** (neighbor)
 - Τα δεδομένα ρέουν από την πηγή προς το sink κατά μήκος της gradient

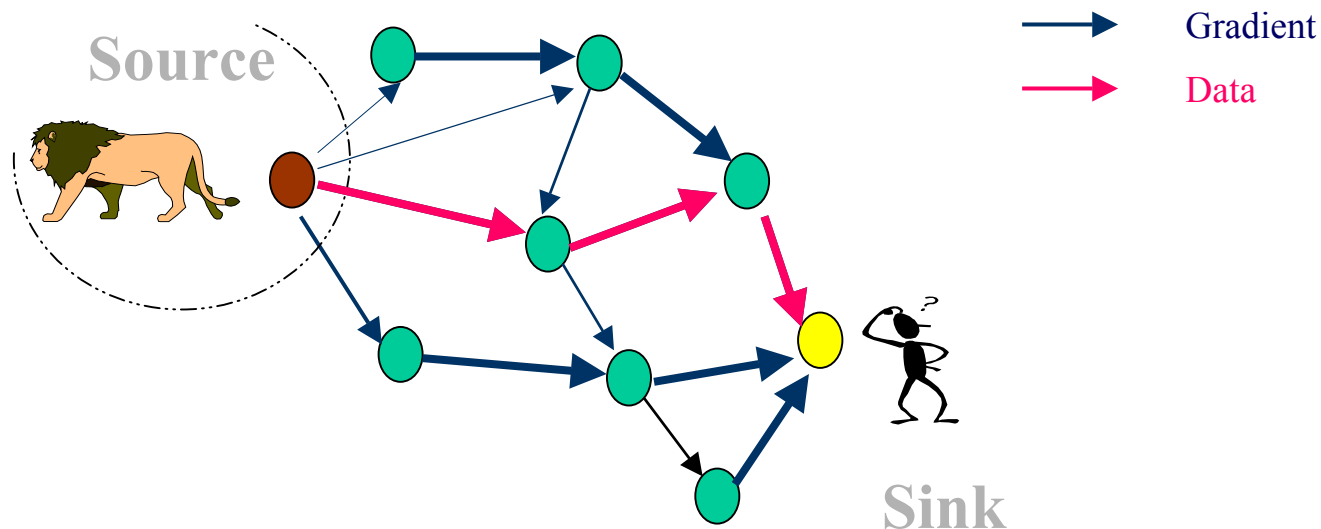
Διάδοση του interest

- Πλημμυρίδα (Flooding)
- Constrained ή Directional flooding ανάλογα με τη τοποθεσία
- Directional Propagation βασίζεται στα προηγούμενα cached data



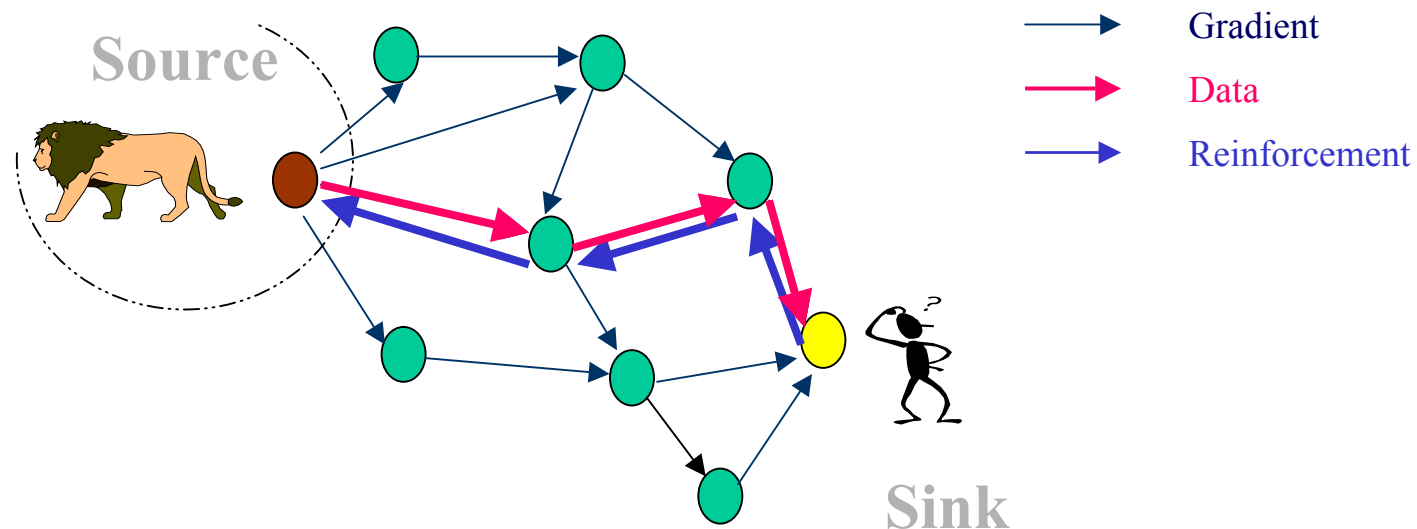
Διάδοση των δεδομένων

- Ενίσχυση, ώστε διανομή σε ένα μόνο μονοπάτι
- Διανομή από πολλαπλά μονοπάτια με πιθανοκρατική προώθηση
- Διανομή από πολλαπλά μονοπάτια με επιλεγμένη ποιότητα κατά μήκος διαφορετικών μονοπατιών



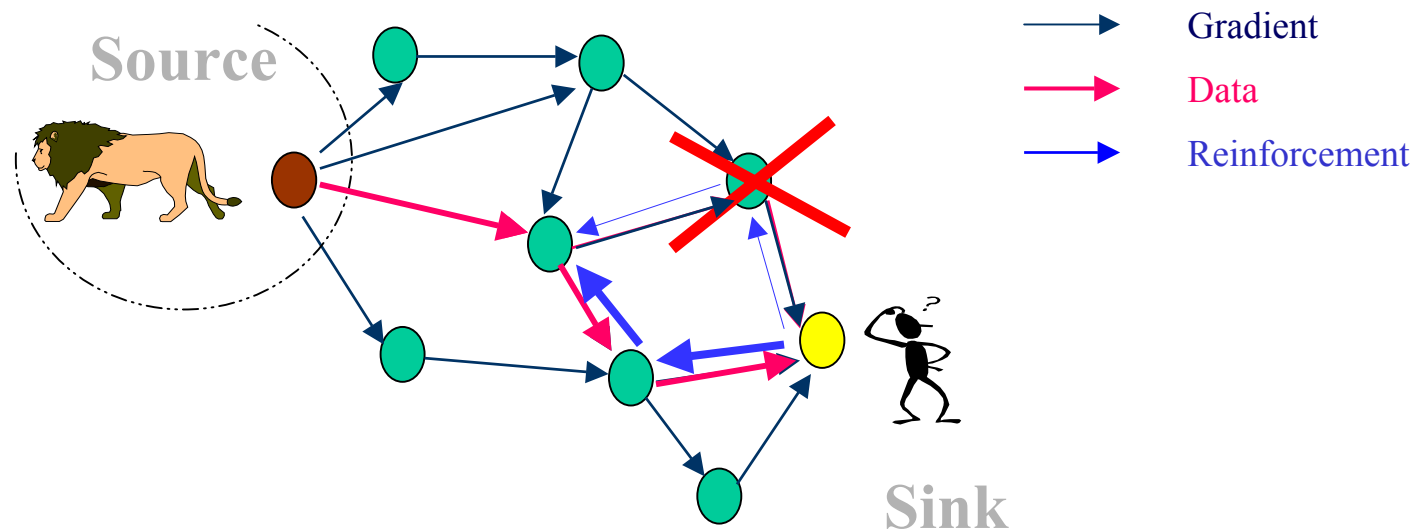
Ενίσχυση (Reinforcement)

- Ενίσχυση ενός από τους γείτονες μετά τη λήψη των αρχικών δεδομένων
 - Γείτονας/ες από τον/τους οποίο/ους ελήφθησαν νέα events
 - Γείτονας ο οποίος συστηματικά “αποδίδει” καλύτερα από τους άλλους
 - Γείτονας από τον οποίο ελήφθησαν τα περισσότερα events



Αρνητική ενίσχυση (Negative Reinforcement)

- Ρητή υποβάθμιση του μονοπατιού με αποστολή ξανά ενός *interest* με χαμηλότερο ρυθμό δεδομένων (data rate)
- Time out



Σύνοψη του Directed Diffusion

- Η μετάδοση δεδομένων σε επίπεδο εφαρμογής έχει τη δυνατότητα να ελαττώσει σημαντικά την κατανάλωση ενέργειας
 - Δεδομενο-κεντρική διάχυση πληροφορίας
 - Ενίσχυση (Reinforcement) βασισμένη σε προσαρμογές των μονοπατιών
 - Εξάλειψη και συσσώρευση (suppression, aggregation) των διπλοτύπων