An Algorithm for Distributing and Retrieving Information in Sensor Networks

OPODIS 2006

Hugo Miranda[1] Simone Leggio[2] Luís Rodrigues[1]
Kimmo Raatikainen[2]

[1] University of Lisbon and LaSIGE - Portugal

[2] University of Helsinki - Finland

December 13th, 2006

The Problem

- To make data available to all nodes in a sensor network
  - e.g. posts to a white board
- Taking into account that
  - Data should be replicated
    - Nodes may fail
  - It is not possible to replicate everything at all nodes
    - Storage space is limited
  - Using as few messages as possible
    - Nodes have limited batteries
  - Replicas should be scattered over the network
    - Reduces access latency
    - Saves messages
- Assumption
  - Sensors are not aware of their location
    - i.e. they do not have a GPS device
  - Omni-directional antennas

Related Work

Hara:{01,04} Requires neighborhood awareness. Number of replicas depends of an estimated access pattern.

Datta:04 (Autonomous gossiping) Nodes advertise the profile of the data they are interested. Items migrate.

Yin:06 Assumes a single source of data.

Data centric Requires location awareness. Some (Li:00,Ghose:03,Liu:06) geographically distribute the replicas.

The Algorithm

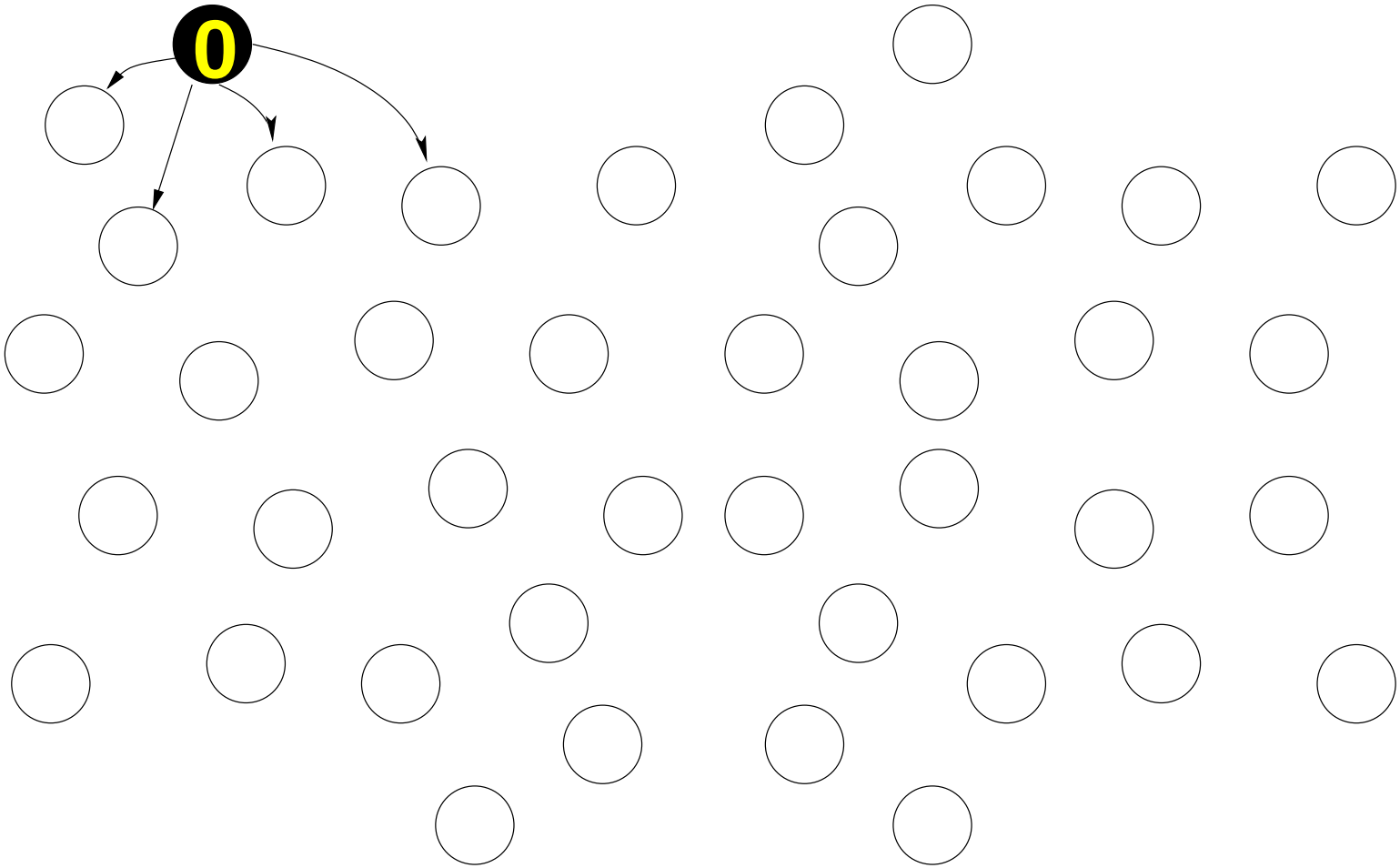
Idea

- Store a copy of each item at every n (DbC) hops
  - So that each node has a copy of each item at most $\left\lceil \frac{n+1}{2} \right\rceil$ hops away
- Use queries to fix the distribution

Requires

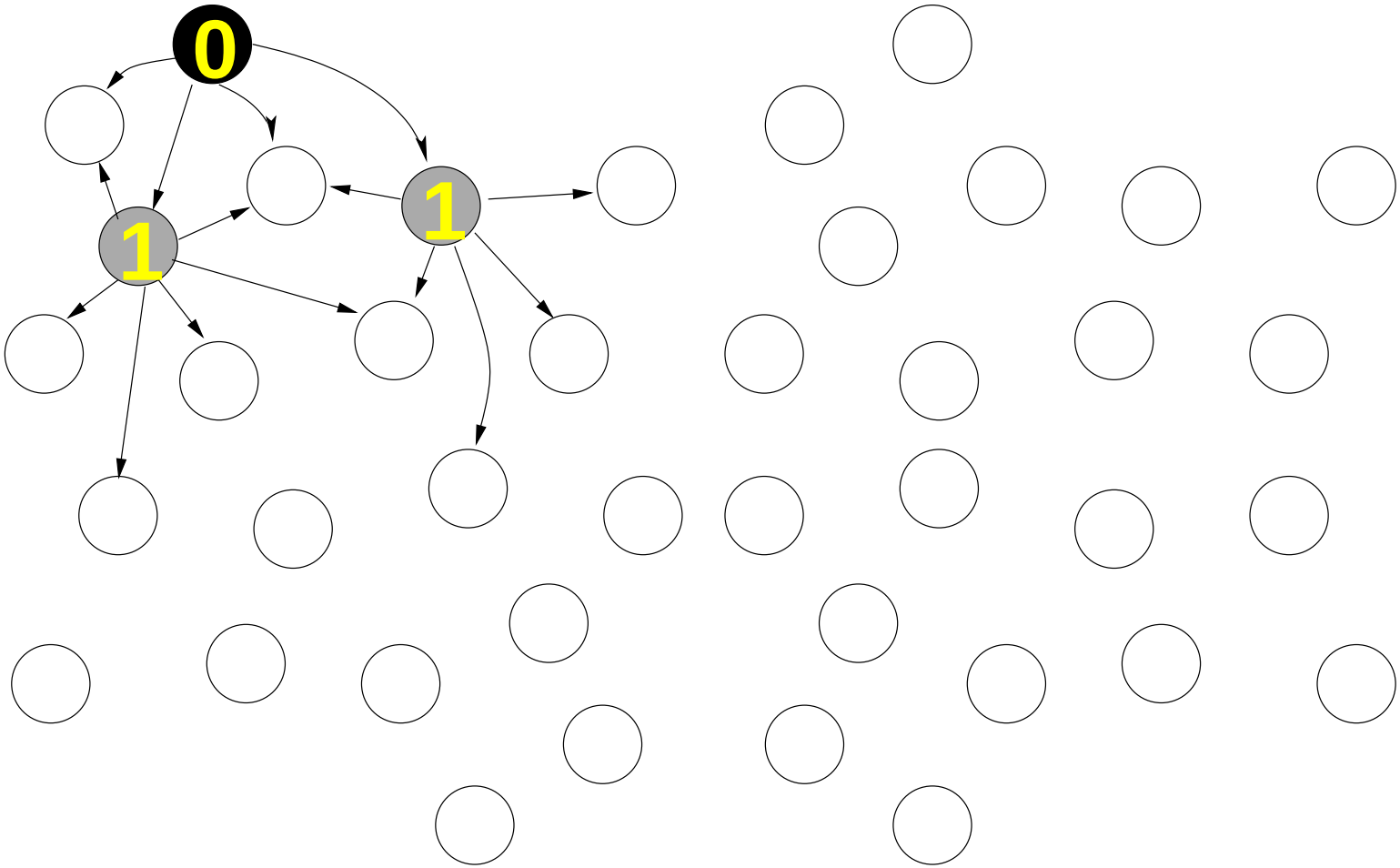
- A counter on the dissemination message (TFS)
  - Incremented by the nodes that forward the message
  - When the counter reaches DbC, the node stores the message and resets TFS
- An algorithm to prevent too many nodes from forwarding the message (PAMPA)
  - Sorts nodes according to the reception power of the messages
  - Only the nodes more distant to the source retransmit

Dissemination Example



Numbers show TFS of the message. DbC=2

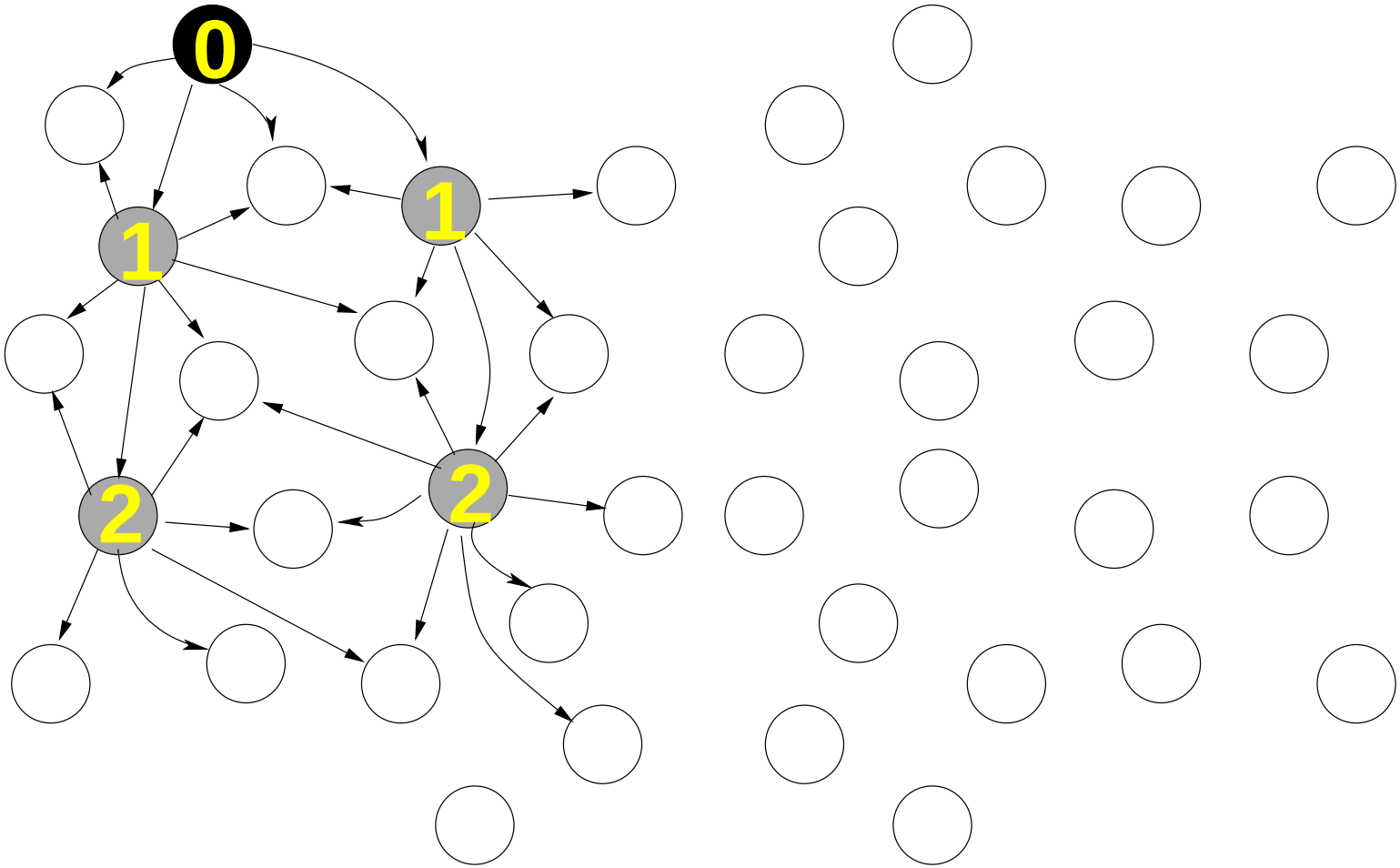
Dissemination Example



Numbers show TFS of the message. DbC=2

Hugo Miranda[1], Simone Leggio[2], Luís Rodrigues[1], Kimmo Raatikainen[2]

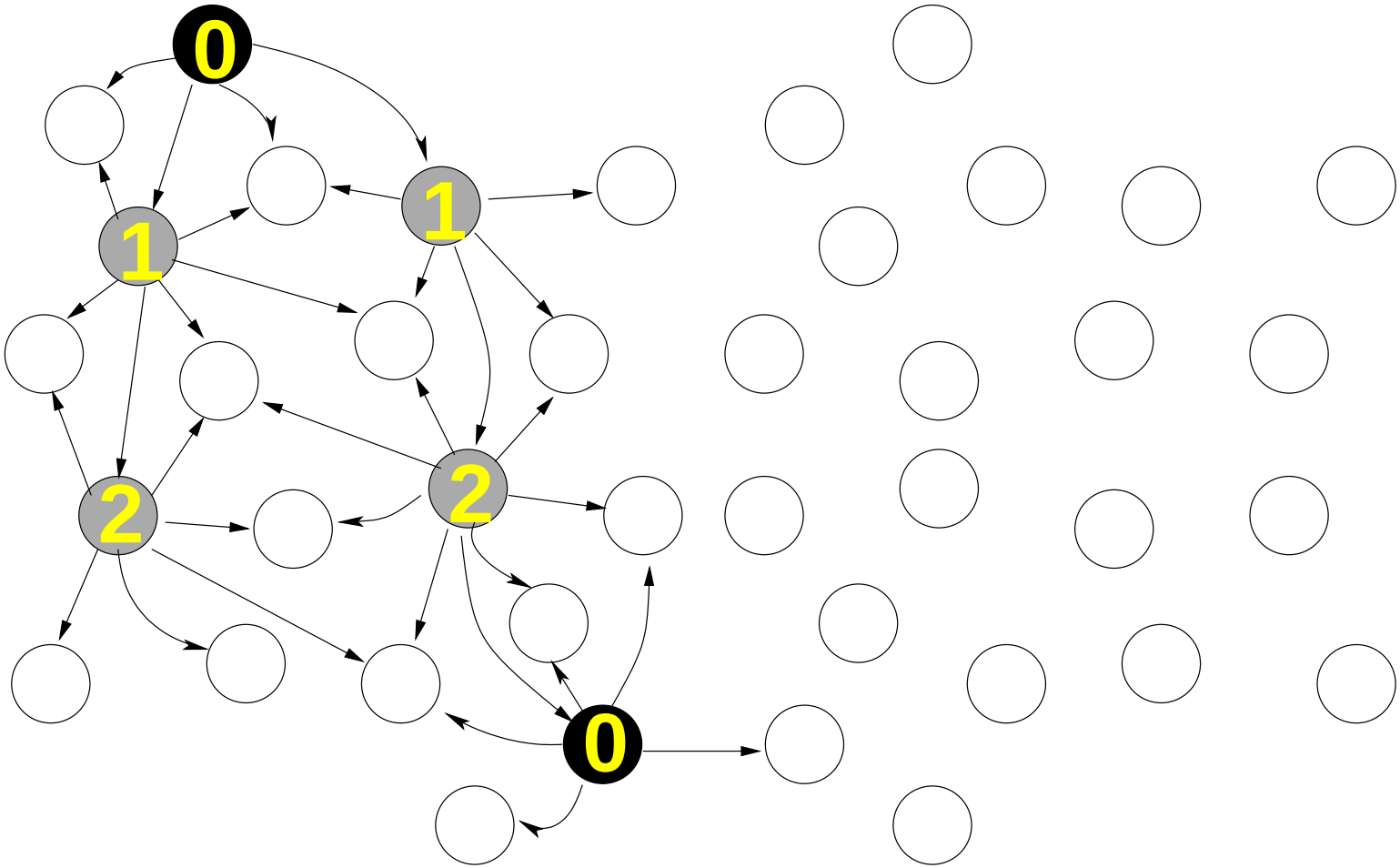
Dissemination Example



Numbers show TFS of the message. DbC=2

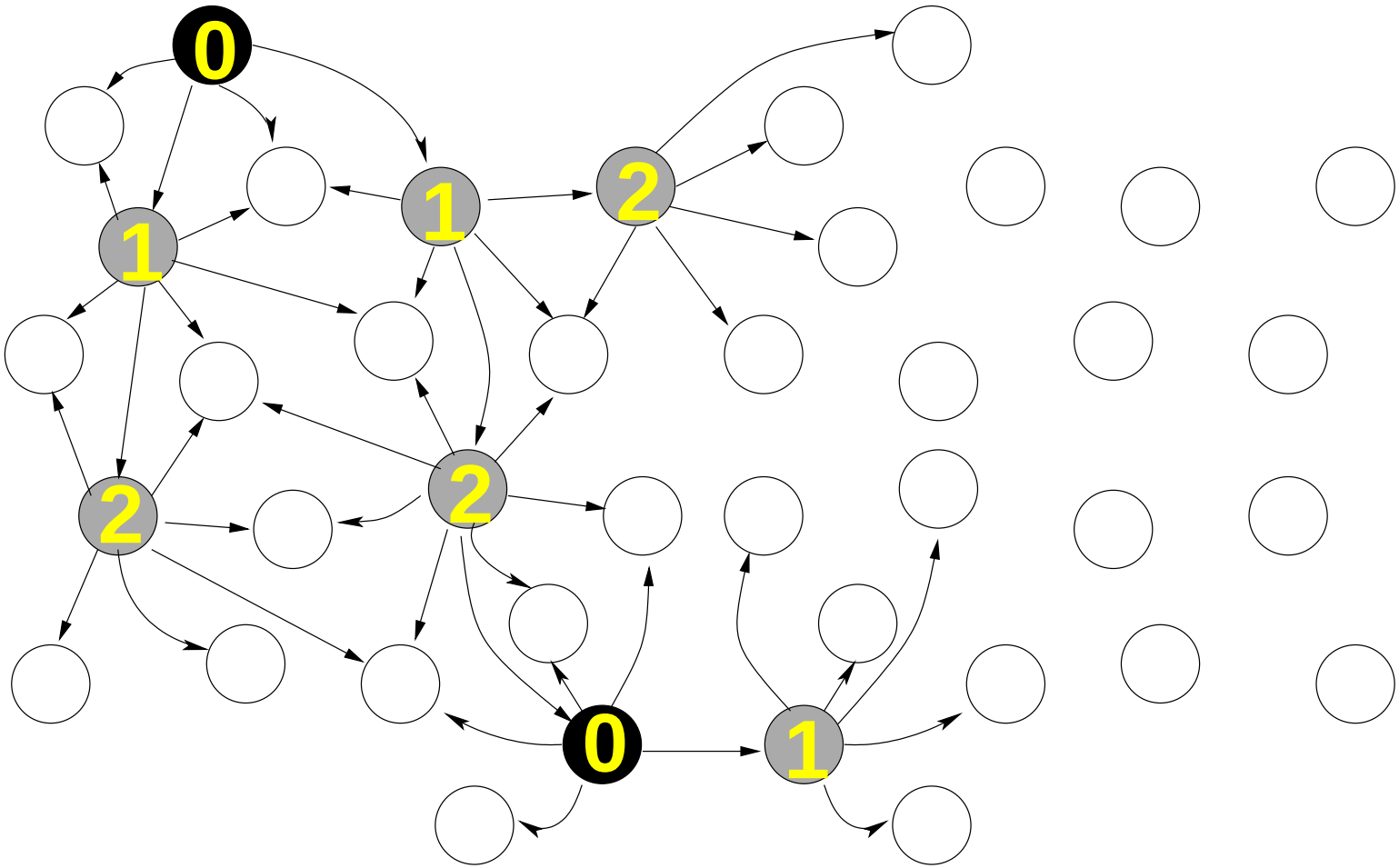
Hugo Miranda[1], Simone Leggio[2], Luís Rodrigues[1], Kimmo Raatikainen[2]

Dissemination Example



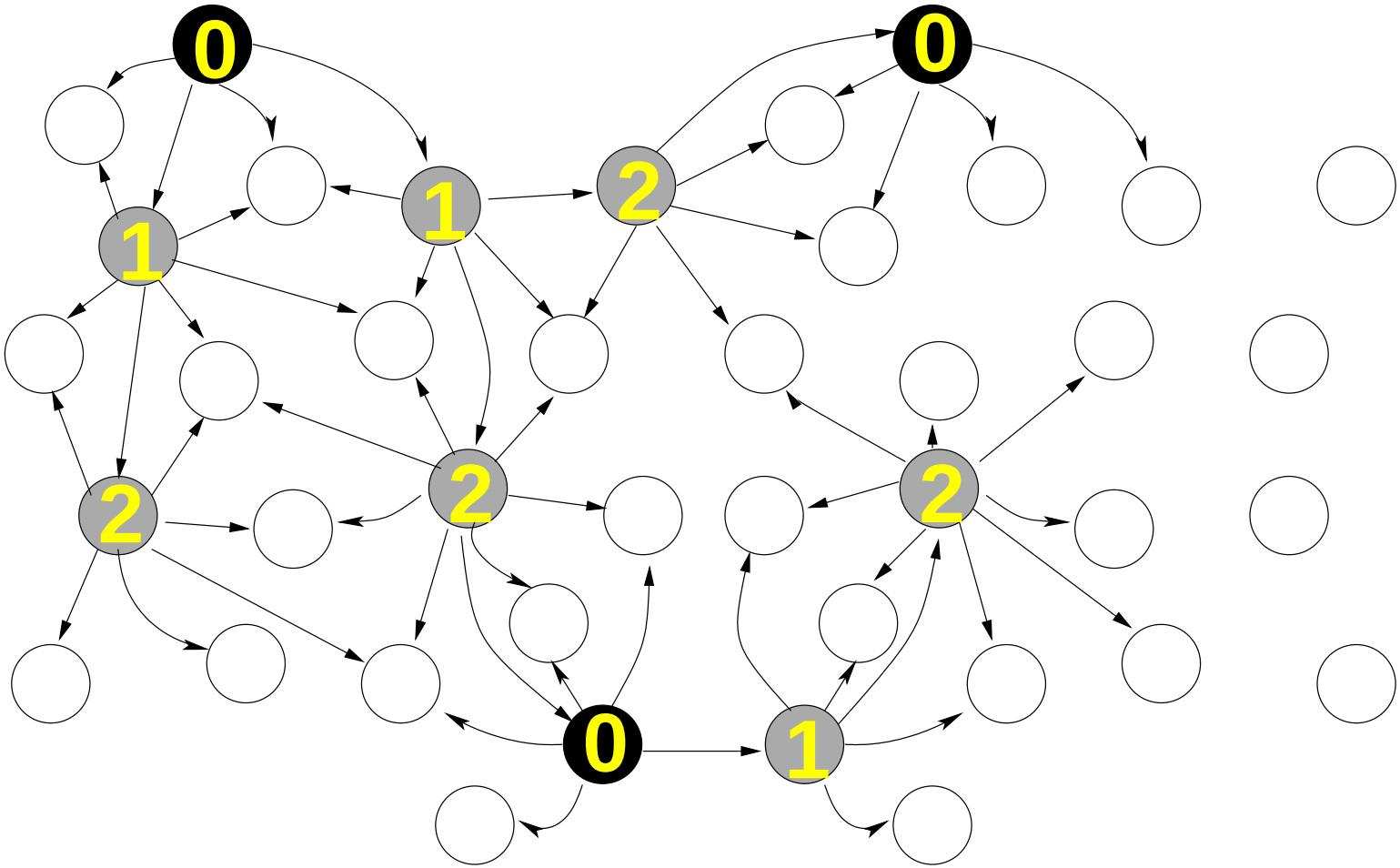
Numbers show TFS of the message. DbC=2

Dissemination Example



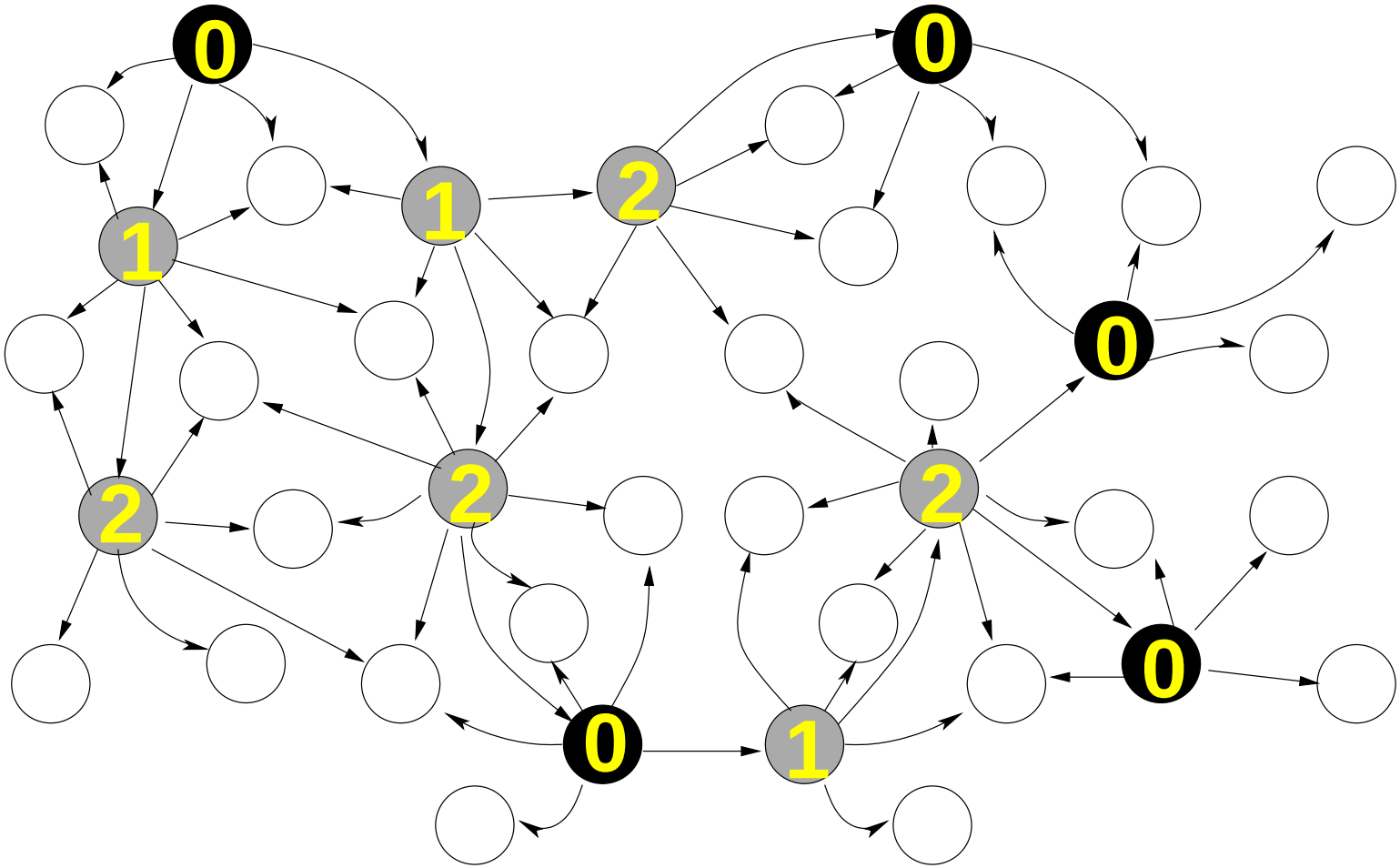
Numbers show TFS of the message. DbC=2

Dissemination Example



Numbers show TFS of the message. DbC=2

Dissemination Example



Numbers show TFS of the message. DbC=2

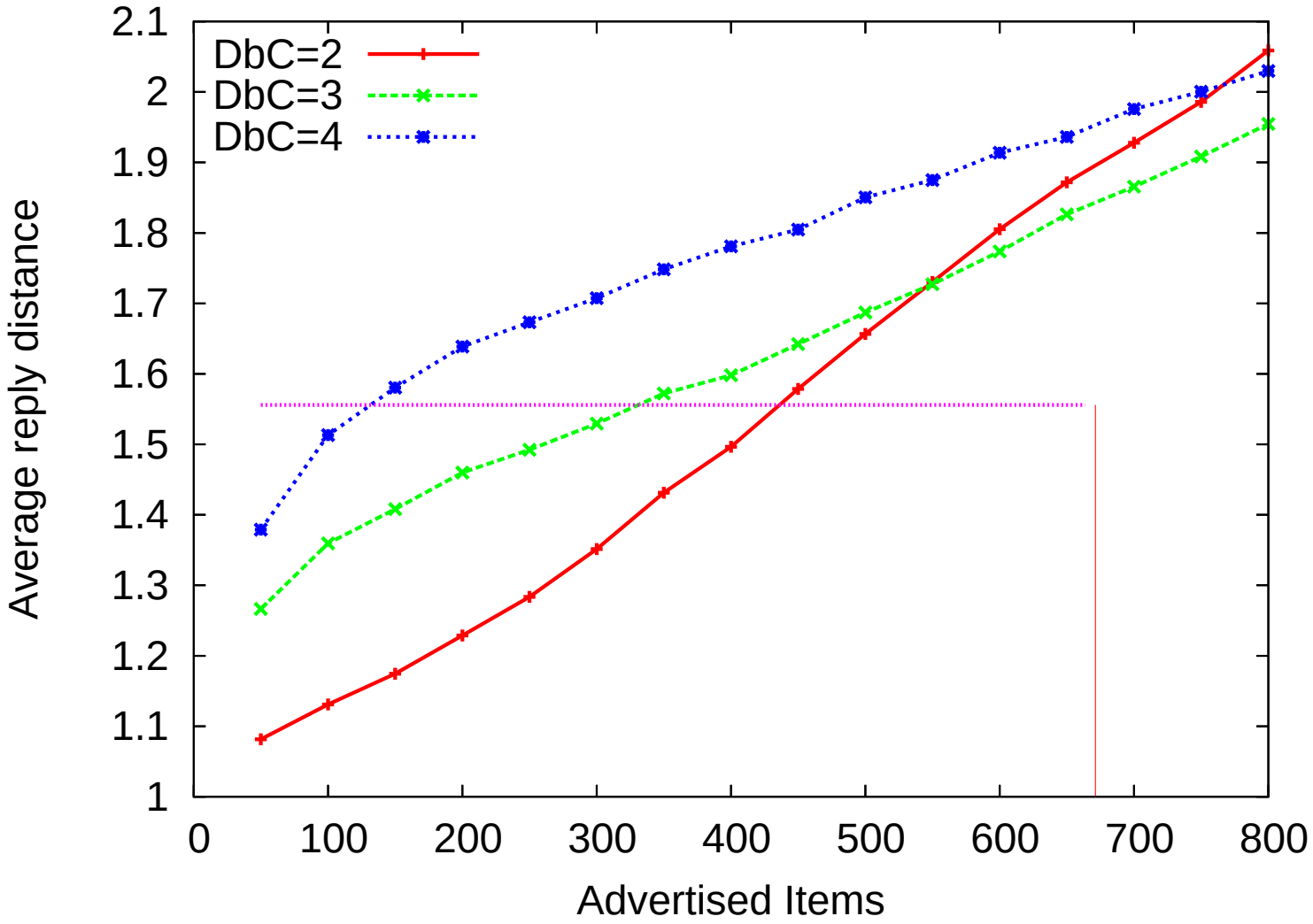
Hugo Miranda[1], Simone Leggio[2], Luís Rodrigues[1], Kimmo Raatikainen[2]

Query

- Nodes first broadcast the query with a small TTL
  - Adaptive value from past experiences
- If no reply is received, flood the network
- Locally store the item if the reply was received from far away
- Replies are sent point-to-point
  - Use the route constructed during query propagation (like DSR)

Evaluation

Distance of the replies



- Simulations in *ns-2*, 100 nodes
- Square defines the theoretical limit for DbC=2
- Degrades gracefully with the number of items

Hugo Miranda[1], Simone Leggio[2], Luís Rodrigues[1], Kimmo Raatikainen[2]

Summary

- We presented an algorithm for disseminating replicas of data items that
  - Geographically distribute the replicas
    - Even when nodes are not aware of their location
  - Creates a number of replicas depending of the size of the network
- Future Work
  - "Repair" the distribution when nodes move