Time-Free Leader Election in Mobile Ad Hoc Networks

Luis Rodrigues Petr Kouznetsov Rachid Guerraoui

Distributed Programming Laboratory
EPFL

Problem

Static network:

- **Eventually there is a leader**
- **There is never more than one leader at a time**

Problem

Mobile ad hoc network:

- ***For every component whose topology does not change sufficiently long*, there is a time after which the component has exactly one leader**

Requirements:

- **“Local” computation only (no flooding)**
- **No source of global information**

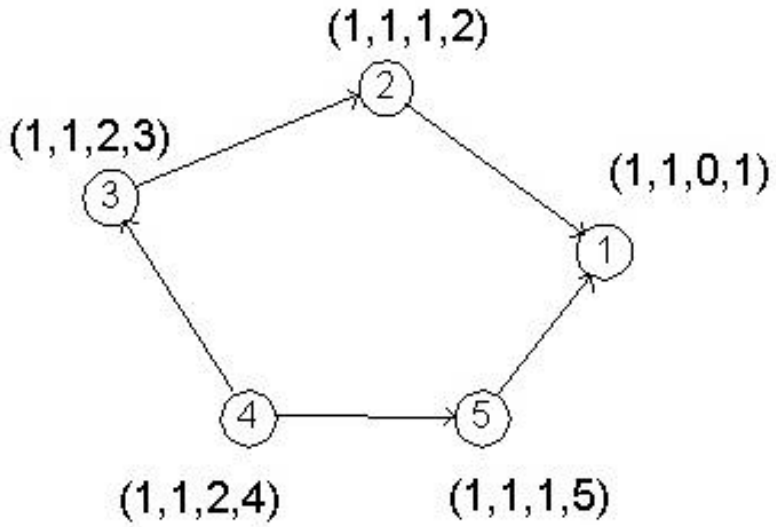
Related work

- **Loop-free routing for unstable topologies [Gafni and Bertsekas, 1981]**
  - ✓ destination-oriented DAG based on totally ordered heights
  - ✓ no flooding, only local computations
  - ✓ no partitioning: destination is always a member
  - ✓ proofs
- **Mobile ad hoc routing scheme (TORA) [Park and Corson, 1997]**
  - ✓ handles partitioning and merging
  - ✓ use of global time source
- **Leader election in a mobile network [Malpani, Welch, Vaidya, 2000]**
  - ✓ *Leader-oriented* DAG in every component
  - ✓ use of global time source
  - ✓ correctness proof for a single topology change only

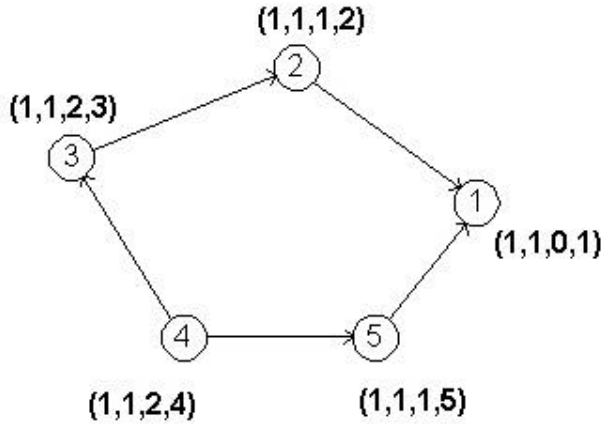
Proposed *time-free* solution

- Leader-oriented DAG in every component
  - ✓ **When a partition occurs, a new leader is elected**
  - ✓ **When two components merge, a contest between two leaders takes place**

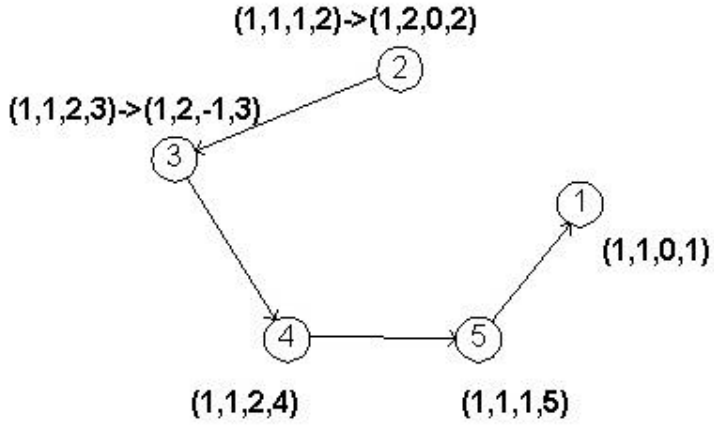
- A height of a node *i* : *hi=(li,ri,di,i)*
  - ✓ ***li*** **- the current leader (the smallest id in a component)**
  - ✓ ***ri*** **- the current leader candidate**
  - ✓ ***di*** **- the distance to the current leader (candidate)**



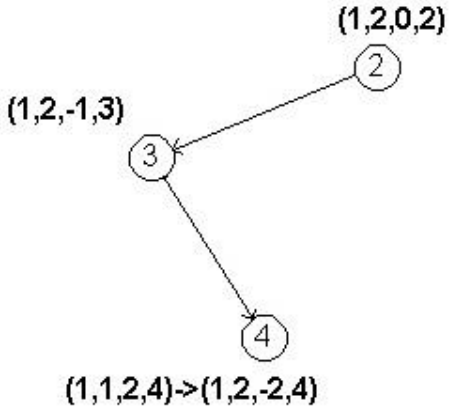
Example of operation



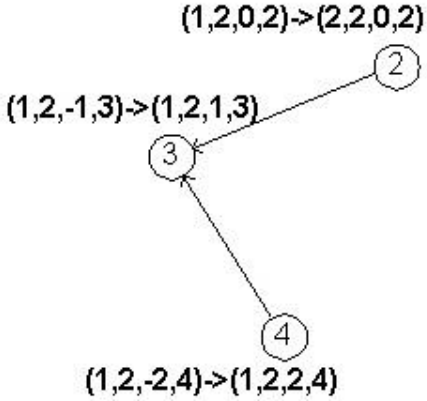
(a)



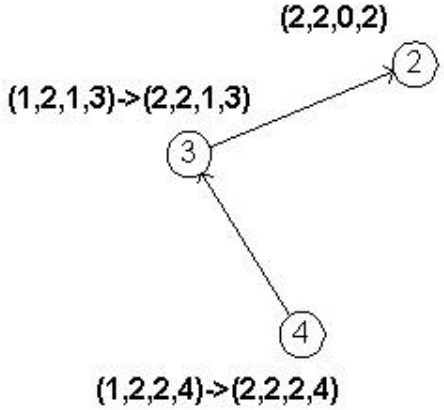
(b)



(c)



(d)



(e)

Contributions & open questions

- Time-free: suitable for asynchronous systems
- Almost no performance degradation comparing to [MWV00]
- Correctness proof for the case of a single topology change

- Concurrent topology changes: is a solution possible?
  - ✓ If it is: can we prove the correctness?
  - ✓ If it is not: minimal amount of synchrony sufficient to solve the problem?