Time Synchronization

Iowa ◆ Ohio State ◆ UT Austin ◆ Michigan State ◆ Kent State

Towards energy saving in ad hoc sensor networks, time synchronization is of fundamental importance. However, it must be efficient, reliable, and accurate.

Time Sync Objectives

- ❖ Synchronize neighboring nodes within a few jiffies (1 jiffy = 30.51 microseconds)
- ❖ Be monotonically increasing
- ❖ Be simple and power efficient
- ❖ Be fault-tolerant and self-stabilizing
- ❖ Does not require anchor nodes

Time Synchronization Demonstration

- ❖ Demonstration 1
    - ❖ Initially, nodes beep randomly when not synchronized
    - ❖ After the nodes exchange time synchronization messages, they beep synchronously

- ❖ Demonstration 2
    - ❖ Network is initially partitioned into two networks
    - ❖ Each network synchronizes within its own partition and beeps synchronously
    - ❖ The nodes in different network partitions beep at different intervals
    - ❖ When a node leaves one network and joins the other network, it synchronizes with the new network

Time Synchronization Protocol

- ❖ Every node maintains two time values: local time and global time. Local time is never adjusted. Only global time is adjusted (monotonically) that reflects the time of the network
- ❖ Every node in the network periodically (once every 15 seconds) broadcasts its global time value to its neighbors
- ❖ When a node receives a time value that is higher than its own time value, it updates its time value to the new higher time value

Conclusions and Future Plans

- ❖ Neighboring nodes synchronize within 7 jiffies
- ❖ Successful demo at OSU for classification and tracking of objects using time synchronization
- ❖ Design and implement a scalable and fault-local stabilizing time synchronization service

Applications

- ❖ Snapshots based on time sync support target classification as a soldier, tank, etc.
- ❖ Tracking the trajectory of an object
- ❖ Determining the velocity of an object
- ❖ TDMA media access control
- ❖ Localization, beam-formation
- ❖ Time-based task scheduling

t_2

t_1

t_3

t_5

t_4

Group 1: Global Time (GT_1) = max (t_1, t_2, t_3)

Group 2: Global Time (GT_2) = max (t_4, t_5)

One sensor from Group 2 joins Group 1

t_2

t_1

t_3

t_5

t_4

Group 1: Global Time (GT_1) = max (t_1, t_2, t_3, t_5)

Group 2: Global Time (GT_2) = t_5