CLOUD COMPUTING CONCEPTS

with Indranil Gupta (Indy)

MEMBERSHIP

Lecture D

WHICH IS THE BEST FAILURE DETECTOR?

FAILURE DETECTOR PROPERTIES …

- Completeness
- Accuracy
- Speed
  - Time to first detection of a failure
- Scale
  - Equal Load on each member
  - Network Message Load

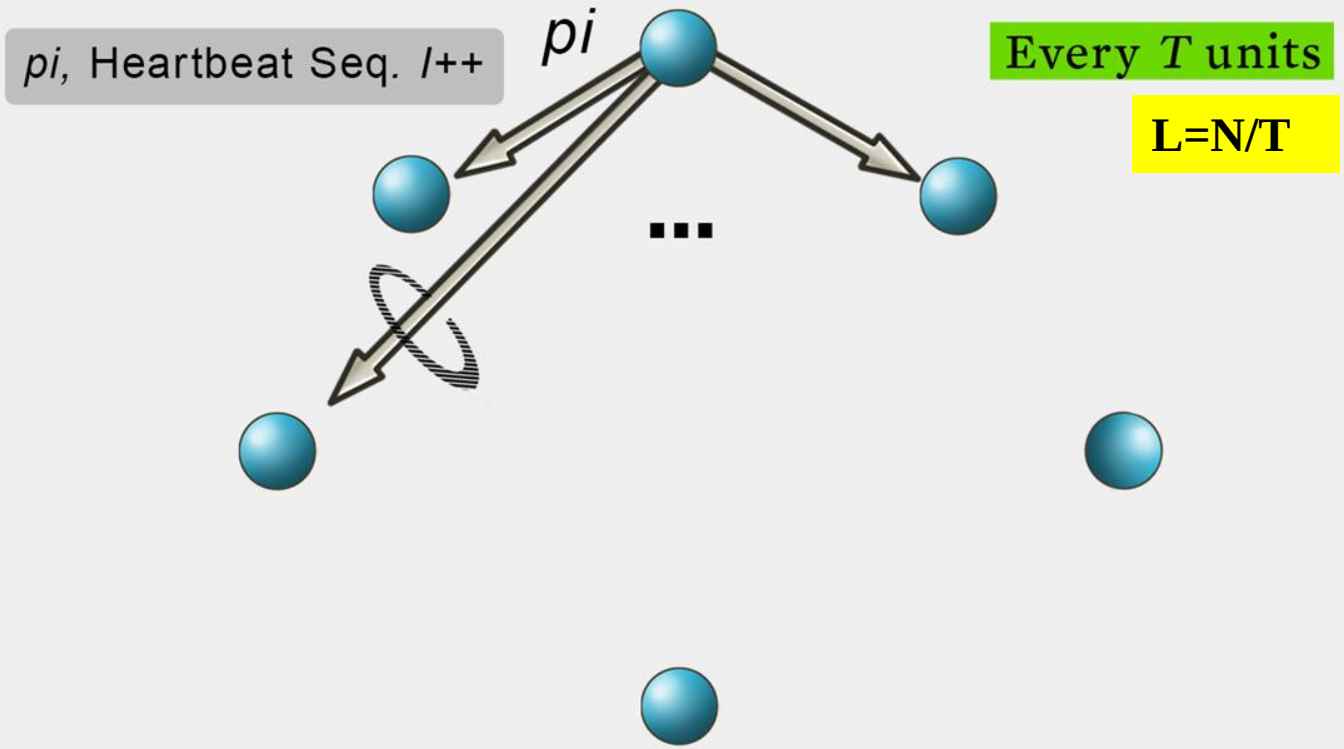
FAILURE DETECTOR PROPERTIES …

- Completeness — Guarantee always
- Accuracy — Probabillity PM(T)
- Speed — *T* time units
  - Time to first detection of a failure
- Scale
  - Equal Load on each member
  - Network Message Load

FAILURE DETECTOR PROPERTIES …

- Completeness — Guarantee always
- Accuracy — Probabillity PM(T)
- Speed — *T* time units
  - Time to first detection of a failure
- Scale — N*L Compare this across protocols
  - Equal Load on each member
  - Network Message Load

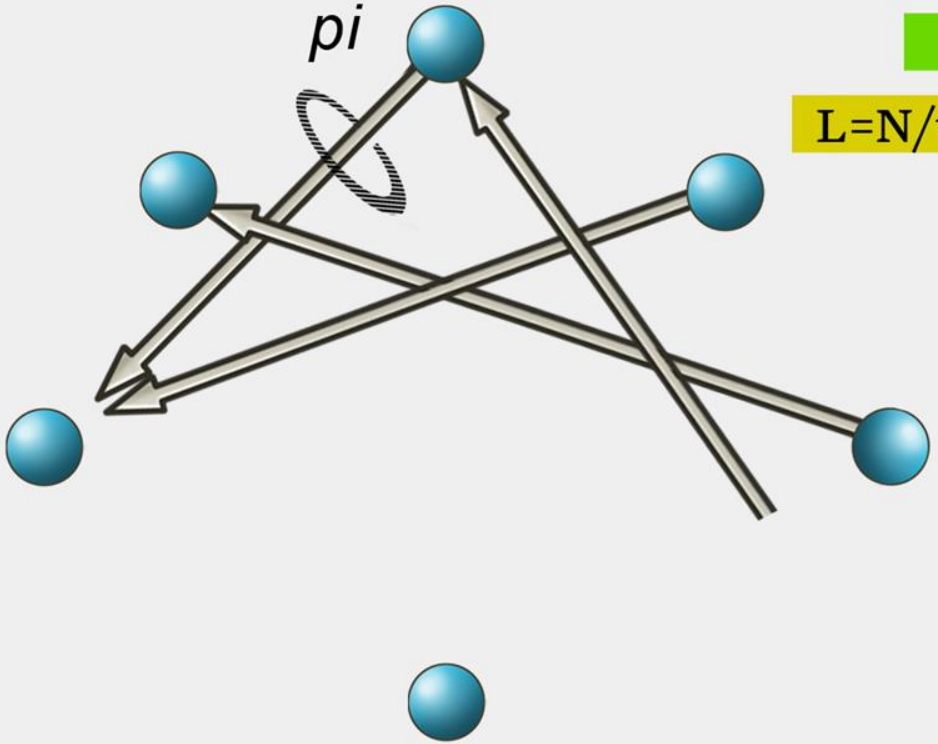
ALL-TO-ALL HEARTBEATING



ALL-TO-ALL HEARTBEATING

Array of Heartbeat seq. *l* for member subset

Every tg units =gossip period, send O(N) gossip message



T=LogN*tg

L=N/tg=N*logN/T

WHAT'S THE BEST/OPTIMAL WE CAN DO?

- Worst case load L*

  as a function of *T*, *PM(T)*, *N*

  Independent Message Loss probability p_{ml}

$$L^* = \frac{\log(PM(T))}{\log(p_{ml})} \cdot \frac{1}{T}$$

(try to work out the proof)

WHAT'S THE BEST/OPTIMAL WE CAN DO?

- Optimal L is independent of N (!)
- All-to-all and gossip-based: sub-optimal
  - L=O(N/T)
  - try to achieve simultaneous detection at ***all*** processes
  - fail to distinguish *Failure Detection* and *Dissemination* components

Key:

- Separate the two components
- Use a non heartbeat-based Failure Detection Component

NEXT

- Is there a better failure detector?