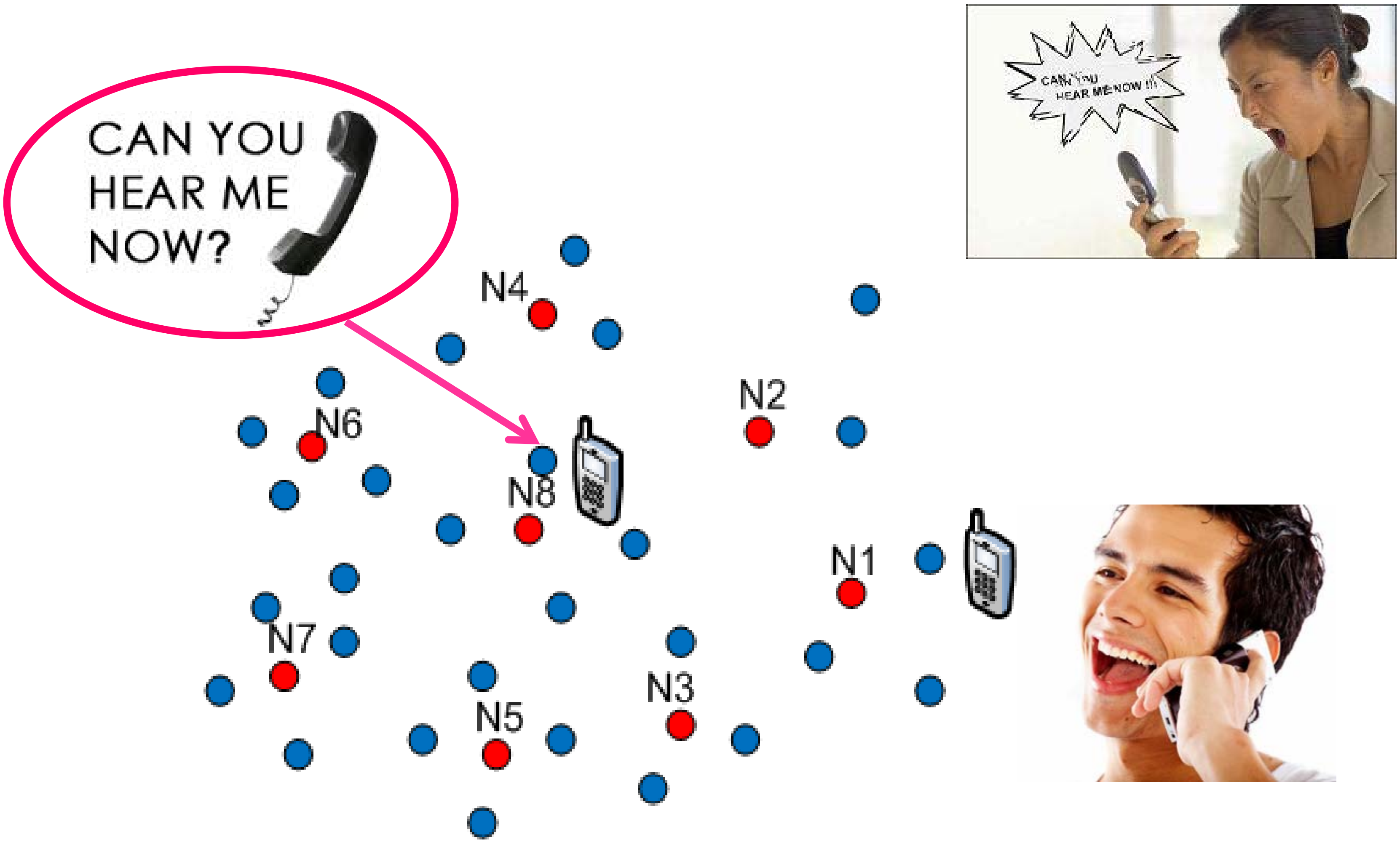
Distance-aware Overlay Routing with AODV in Large Scale Ad Hoc Networks

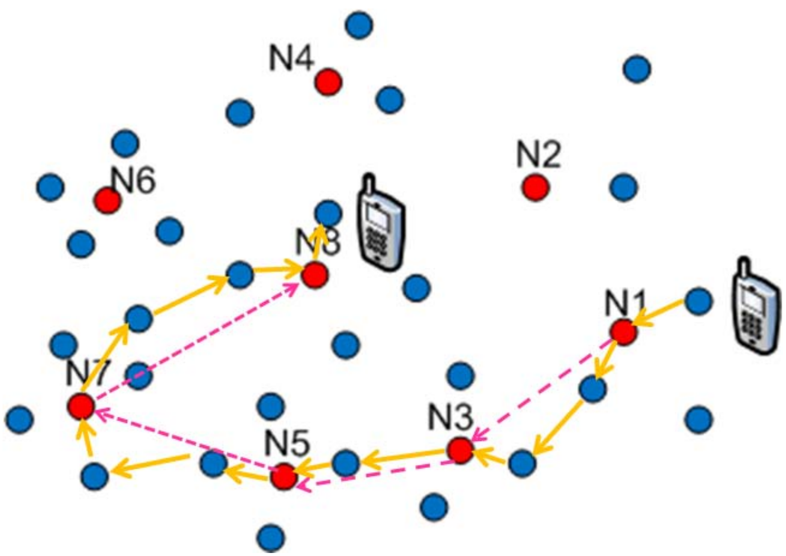
Ying Liu, Xiruo Liu, Wade Trappe, Radhika Roy

Scenario



Background

- In large-scale Ad Hoc networks, nodes are grouped by clusters with a cluster head (red node) in each group.
- Cluster heads build the logical route among each clusters from source to the destination. (dotted line)
- Logical routing orients to the application data and functions among cluster heads.
- Physical routing underneath builds a true path to the destination (solid yellow line).



Problems and Shortcomings

- Back tracking problem
- Bad outcomes are:
  - ❑ Huge latency
  - ❑ High loads of unnecessary traffic
  - ▪ "Global mobile data traffic will increase nearly 11-fold between 2013 and 2018", by CISCO[1] in Global Mobile Data Traffic Forecast Update
  - ❑ Larger packet lost rate during transmission

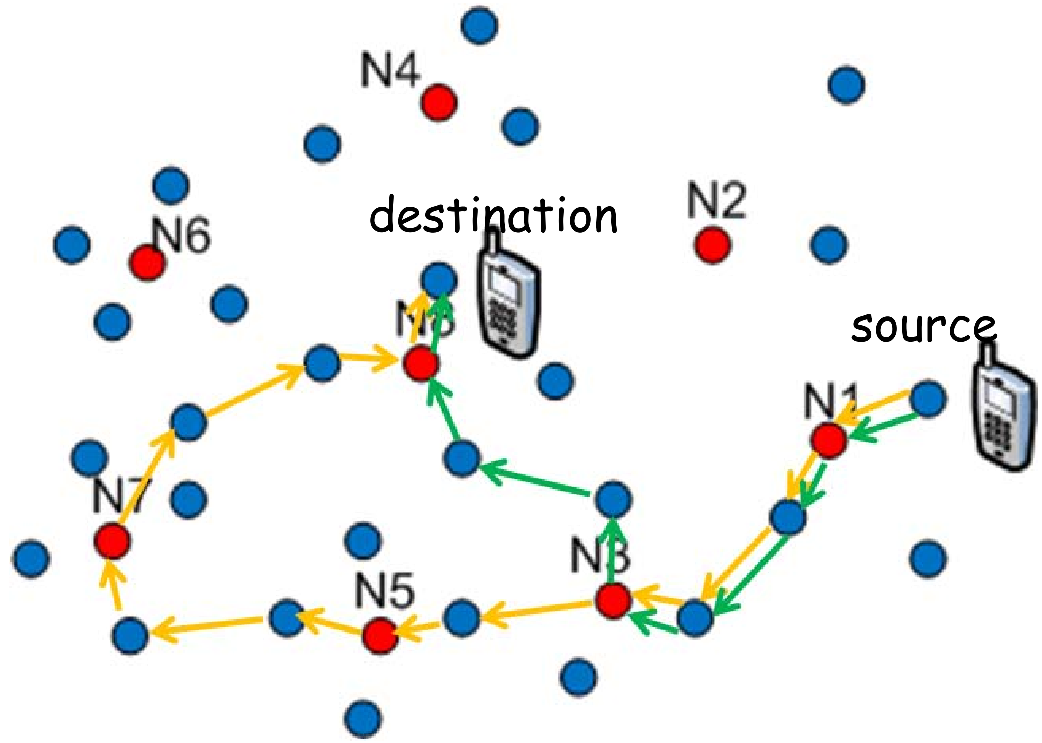
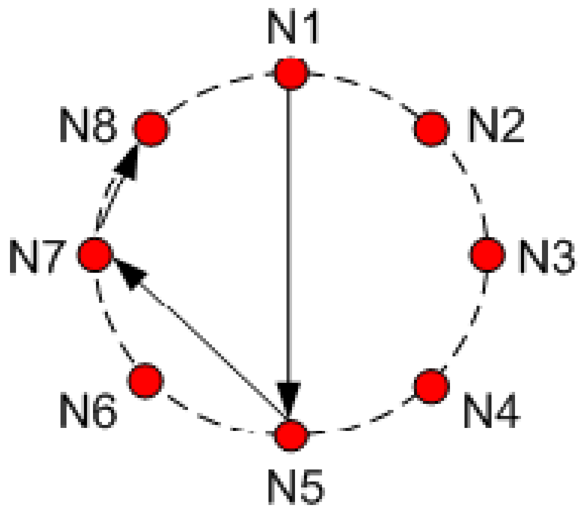


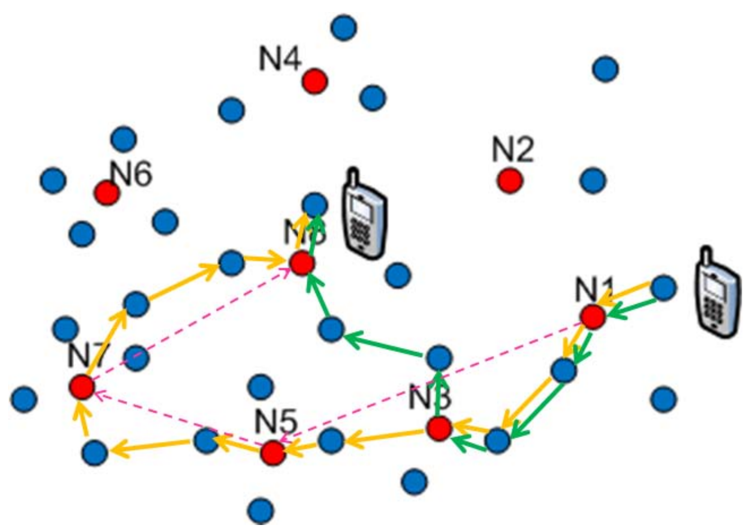
Fig 1, back tracking problem in overlay networks

Reasons for Back Tracking

- Logical routing (Chord) works **independently** from underlayer physical routing.
- Chord does not have information of geographical distance among each nodes, so it does not arrange node in a geographical order.
- For example,

  Path: From N1 to N8, it increases latency & loss of packet & network loads



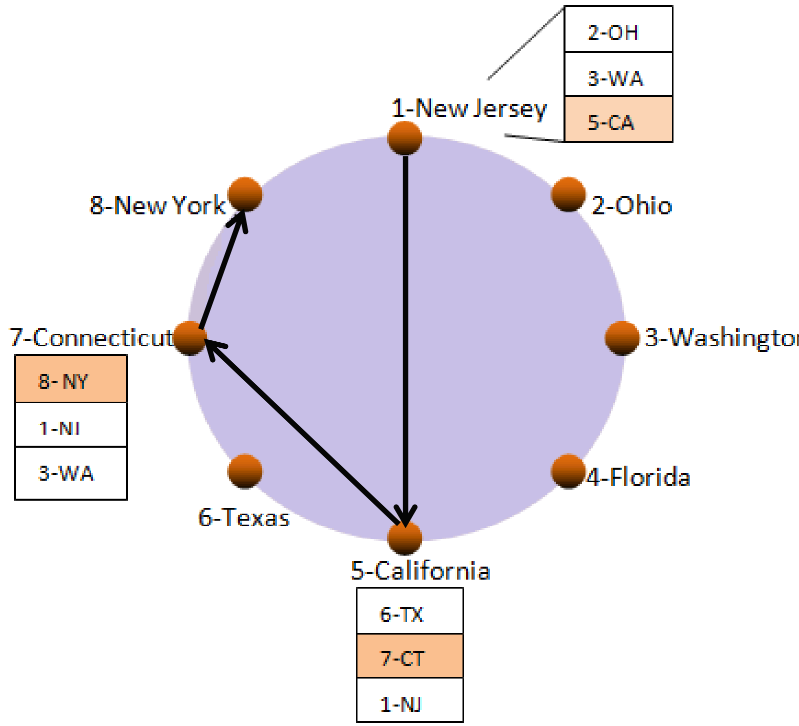


Chord -Overlay Routing

- Nodes are randomly positioned on the Chord of logical ring
- Each node maintains a Distributed Hash Table (DHT) and is mapped onto DHT by collision-free hash function:

$$n + 2^{k-1} \bmod 2^m$$

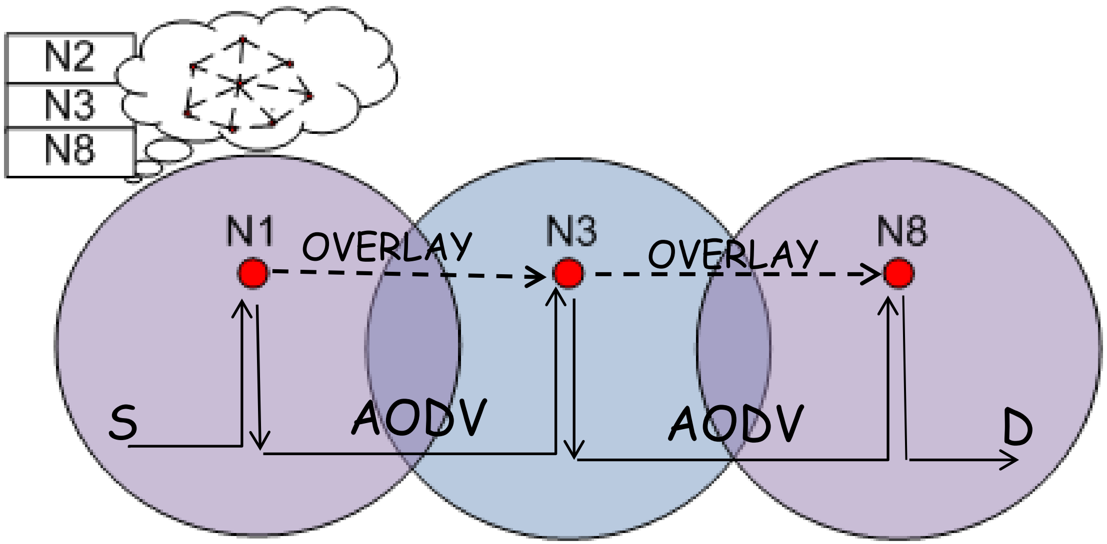
- During the routing process,
  - First, the source node finds whether the destination node is in its DHT
  - If not, it will continue to find the closest predecessor until reaching the destination node.




```
Data: Source, Destination, DHT
Result: Find the closest preceding node
for id = m downto 1 in DHT do
    if id ϵ (Source, Destination) then
        return id
    end
end
```

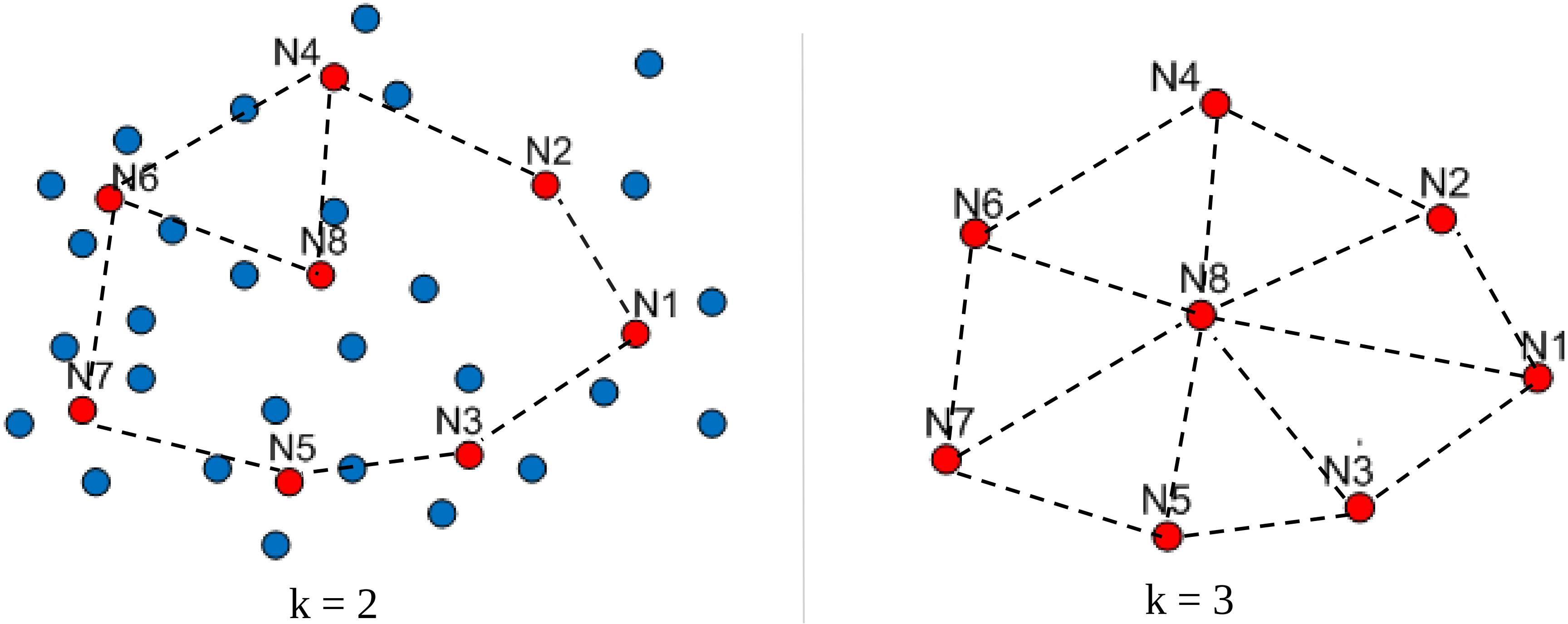
Solution-Distance Aware Overlay Routing

- Each cluster head stores the k-nearest cluster heads.
- Build an adjacency link list among these cluster heads
- An adjacency link list corresponds to one topological graph among the cluster heads.
- The overlay routing finds the shortest path in this logical graph.



Length of distributed distance table a free parameter

- Make a tradeoff between the number of hops of the logical path, which affects the packet delivery time, and the complexity of finding a shortest path
- The larger the value of k is, the more connected the logical topology

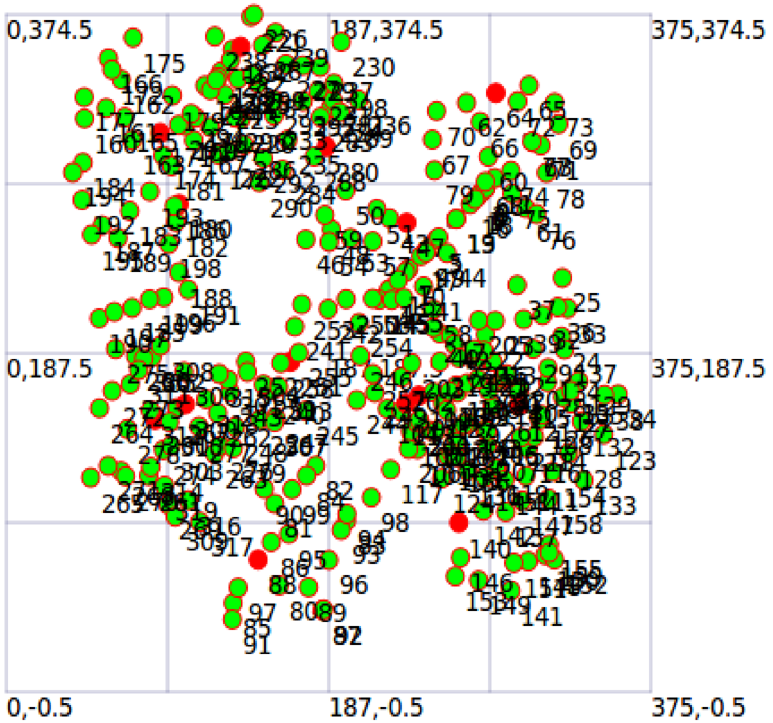


Cluster Heads

- Aim: Design a cluster head to compare the number of hops of each protocol in terms of good and bad cluster head
- Ideal cluster heads are assigned according to the requirement of application (VoIP) which demands not too much delay

$$CH = argmin_i \sum_{j \epsilon S \backslash i} hop(i, j)$$

- For comparison, a bad cluster head is also randomly chosen within a cluster

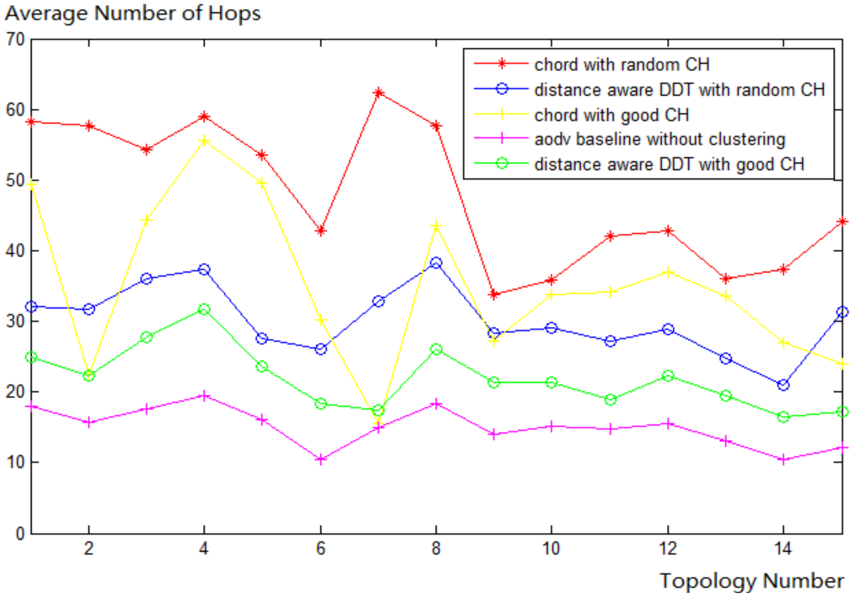


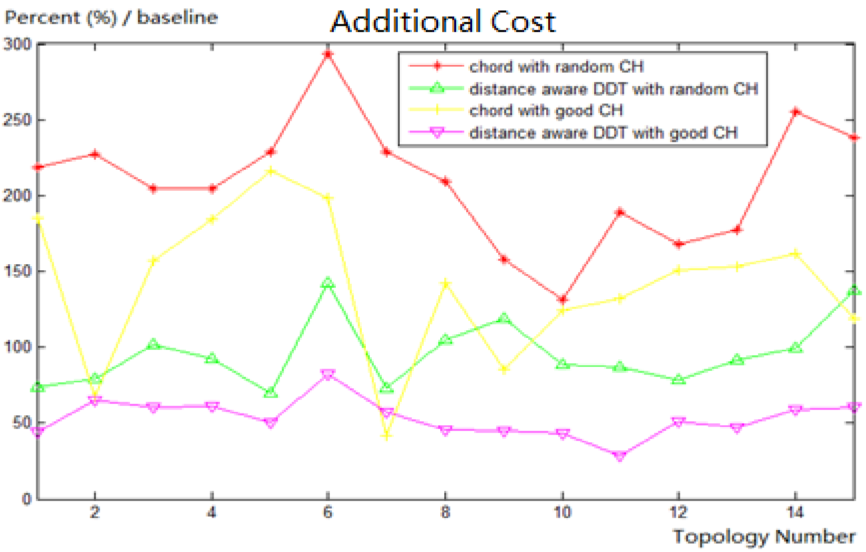
Simulation

- Networks consisting of 320 nodes composed of 16 clusters and each cluster contained 20 nodes.
- We produced 15 topologies.
- Simulation runs 500 time units
- Average hops are collected by averaging the results obtained from running the simulation 100 times.
- Observe:
  I. The average number of hops a packet went through for each protocols
  II. The additional cost of Distributed Distance Table-AODV (DDT-AODV) and standard Chord-AODV compared to the baseline AODV
  III. The path failure probability for the above three different routings

Simulation Results – Average Hops

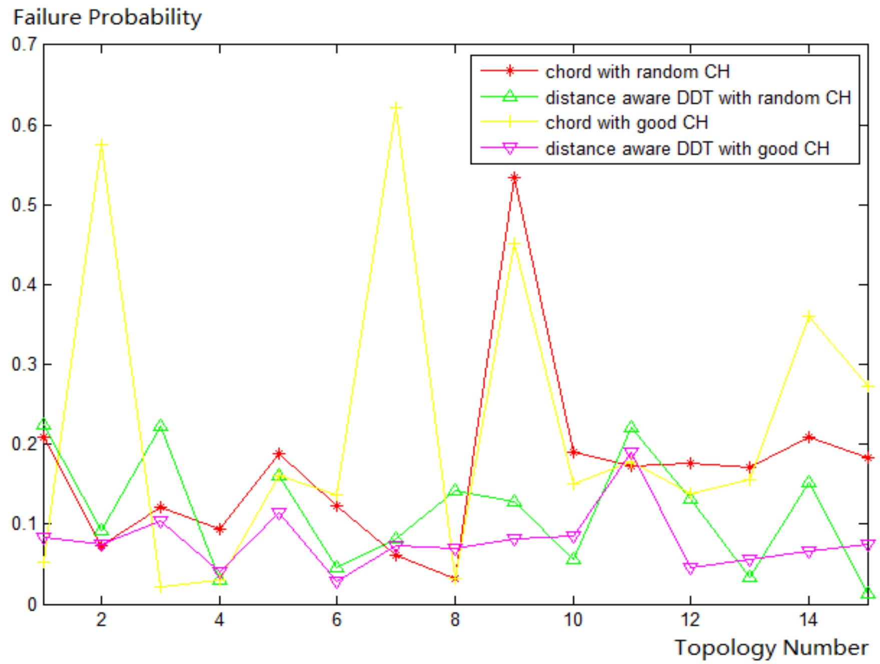
- DDT-AODV gave much fewer average hops than Chord
- Even with a bad cluster head, DDT-AODV performed almost same as Chord with good cluster head.
- The performance of Chord with minimum cluster head had a larger variance compared to DDT-AODV with good cluster head
- The Large variance reveals the general behavior associated with “back tracking” in Chord-selected paths.





Simulation Results – Failure Probability

- DDT-AODV with minimum hop cluster head had a low failure probability compared to Chord.
- Its failure probability was always below 0.2
- Reasons for higher and inconsistent variance on failure probability of Chord:
  I. "back tracking" caused by nodes randomly positioned on Chord ring without considering the underlying broken links
  II. The longer path increases the failure probability



- Choosing a good cluster heads was more important for DDT-AODV than Chord

Conclusions

- We have analyzed the weakness of Chord-based overlay routing in peer-to-peer networks, that is “back tracking” problem. Chord may produce some unnecessary twisted path resulting in large latency in time-sensitive application;
- Solve the “back tracking” problem by building a distance-aware overlay routing protocol;
- Use the physical connectivity to guide the logical routing
- Implement a distributed distance table work along with underlying AODV routing protocol.

References

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018", http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html

[2] I. Stocia, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications" in ACM SIGCOMM Computer Communication Review, vol. 31, no. 4, ACM, 2001, pp. 149-160.

[3] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing" in Mobile Computing Systems and Applications. 1999. Proceedings. WMCSA'99. Second IEEE Workshop on. IEEE, 1999, pp. 90-100

The End

Any questions?

Thank you!

PLEASE EMAIL TO ME:

yingliu@winlab.rutgers.edu