Mobile P2P Networks for Highly Dynamic Environments

Kei Takeshita, Masahiro Sasabe, and Hirotaka Nakano

Osaka University, Japan

Outline

- Introduction
  - Classification of P2P networks
- Background
  - Problem of the structured P2P networks over mobile ad hoc networks
  - Link disconnections
- Related work
  - MADPastry
- Proposed method
- Conclusion & Future Work

P2P network

- P2P networks are classified into unstructured and structured
  - Unstructured P2P networks
    - Find a destination node by flooding
    - Induce a high amount of traffic into the network
  - Structured P2P networks
    - Find a destination node by unicast
      - DHT enables such a unicast-based routing
    - Keep low search costs with an increase in network size

Structured P2P networks are more scalable to the network size

Problem of the structured P2P networks over mobile ad hoc networks

- Success ratio of object search deteriorates as node velocities become high
  - Object: file, service etc.
- What causes the problem?
  - Link disconnections at the network layer
    - The next-hop node does not exist in its wireless transmission range due to its movement
    - They frequently occur under highly dynamic environments

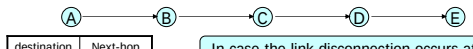
Unicast packet transfer is not reliable in MANETs

Objective

- Improving success ratio of object search in highly-dynamic mobile P2P networks
  - Understand how the link disconnections happen in the network-layer routing protocols
  - Propose an application-layer approach to tackle the link disconnection problem

Behavior of routing protocols in the network layer when link disconnections occur

- Proactive Routing Protocols (e.g. OLSR)
  - Maintain lists of destinations and their routes by periodically distributing routing tables
  - If link disconnections occur, the node abandons sending the packet
- Reactive Routing Protocols (e.g. AODV)



destination	Next-hop
E, 4 hop	B

In case the link disconnection occurs at D→E: Node D re-finds the route to node E by flooding

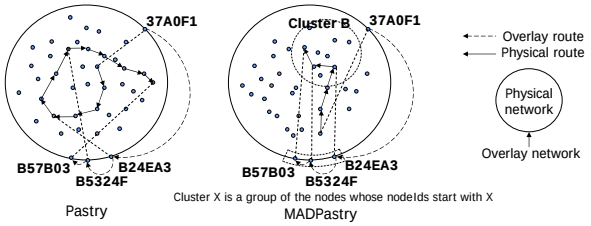
In case the link disconnection occurs at A→B, B→C, or C→D : The node rejects the packet

How to reduce the link disconnections

- Increase the link reliability at the network layer
  - AODV-BR, AOMDV etc.
  - × Limited reliability improvement
- Send multiple queries
  - Send multiple queries for each search
  - × Induce additional traffic that reduces the advantage of structured P2P networks
- Reduce the hop count to the destination node
  - Clustering ⇒ MADPastry
  - Replication ⇒ Proposed method

MADPastry (Pastry + AODV) [*1]

Routing from nodeId: 37A0F1 to searching objectId: B57BD0



- Form clusters consisting of physically-close nodes
  - By equalizing the first digits of the nodeIds among all nodes in a cluster

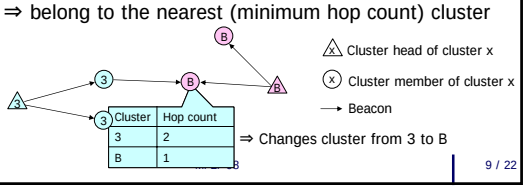
⇒ Reduce the physical hop count in routing

[*1] T. Zahn and J. Schiller: "MADPastry: A DHT substrate for practicably sized MANETs", Proceedings of the Fifth Workshop on Applications and Services in Wireless Networks, Paris, France (2005).

Clustering method

- Clustering using Landmark keys
  - Landmark keys are preassigned
  - A node whose nodeId is the closest to the landmark key becomes a cluster head
    - Sends beacon to members of the cluster
  - Non cluster-head nodes store physical hop count to each cluster head based on received beacons

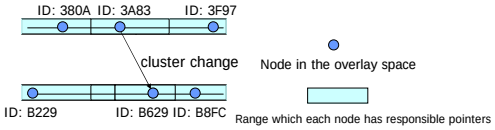
    ⇒ belong to the nearest (minimum hop count) cluster



⇒ Changes cluster from 3 to B

Problem of clustering method

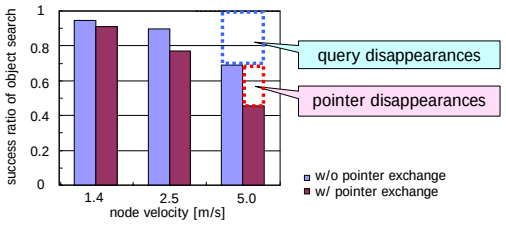
- Changing cluster results in a temporal churn
  - The corresponding node leaves and rejoins the network
    - Since it has responsible pointers (the pair of objectId and IP address) whose objectIds are the closest to its nodeId, it must update pointers



Simulation – MADPastry with considering pointer exchange

- Assess the overheads and risks of cluster changes that were not evaluated in Ref. [*1]
- Simulation environments
  - Number of nodes: 250
  - Number of objects: 1000
  - Query interval: 10 s/query in each node
  - Mobility model: Random Waypoint Model
  - Simulation time: 3600 s, use the latter 2000 s
  - Number of cluster: 16
  - Transmission range: 250 m
  - Node density: 100 node/km^2

Simulation result – MADPastry considering pointer exchange



- Metric: Success ratio of object search
  - w/o pointer exchange : ratio of the number of queries reaching nodes whose nodeIds are the closest to searching objectId
  - w/ pointer exchange: ratio of the number of queries reaching nodes that have the corresponding pointers

・Success ratio of object search deteriorates as the node velocity increases
・Pointer exchanges fail more and more in highly dynamic environments

How to reduce the link disconnections

- Increase the link reliability at the network layer
  - AODV-BR, AOMDV etc.
  - × They can slightly improve the reliability
- Send multiple queries
  - Send multiple queries for each search
  - × They induces additional traffic that reduces the advantage of structured P2P networks
- Reduce the hop count to the destination node
  - Clustering ⇒ MADPastry
  - Replication ⇒ Proposed method

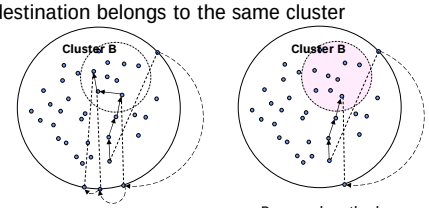
Proposed method

- Pointer replication
  - Where should a node deploy replications?
    - Intermediate nodes on the route for the pointer
  - How the replication can be achieved with low traffic costs?
    - A node can overhear packets transmitted among surrounding nodes

• Sharing pointers among nodes in a cluster
  • In a cluster, overlay-close nodes are also physically close

Sharing pointers among nodes in a cluster (1/2)

- Each node sends its responsible pointers to other nodes in the same cluster
  - Using the periodic beacon messages of MADPastry
  - Each node stores the pointers received or overheard
  - ⇒ All nodes in a cluster can reply to a query whose destination belongs to the same cluster

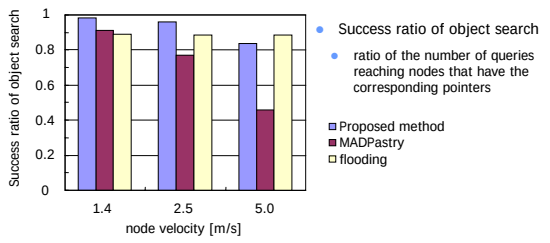


MADPastry Proposed method

Sharing pointers among nodes in a cluster (2/2)

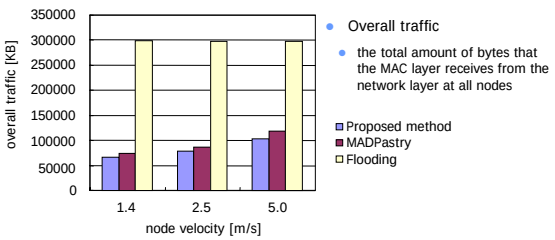
- Effects of pointer sharing
  - Success ratio of object search is
    - Increased by reducing the overlay hop count
  - Overall Traffic is
    - Increased by flooding in clusters to replicate pointers
    - Decreased by reducing the overlay hop count

Simulation Results (success ratio of object search)



- The effect of pointer sharing increases as the node velocity becomes high
- The success ratio of object search of the proposed method is about 6 % lower than that of flooding when the node velocity is 5.0 m/s
  - Since flooding does not rely on physical topology

Simulation Results (overall traffic)



- Flooding requires at most third times as much overall traffic as the proposed method
- Overall traffic of the proposed method becomes lower than that of MADPastry as the node velocity increases
  - (reduction of forwarded queries inside clusters > increase of the beacon size)

Discussion - negative aspects of pointer replications

- Duplicate responses caused by overhearing
  - ⇒ They can be avoided by caching received or overheard responses to the query at each node
- Pointer inconsistency
  - When an object holder changes or leaves the network, a node with the corresponding pointer may be unaware of the event
  - ⇒ It can be alleviated by maintaining replicated pointers in a soft-state manner

Discussion - What is "highly dynamic environment"?

- Network dynamics depends on not only node velocities but also transmission range, parameters of the network routing protocol, etc.

⇒ Link disconnections may frequently occur even when the nodes move slowly

	Outdoor & Running speed	Indoor & Walking speed
Transmission range [m]	250	50
Node velocity [m/s]	5.0	1.0
Node density [node/Km^2]	100	2500

These two environments have the same degree of network dynamics

Conclusion

- Improving success ratio of object search in highly-dynamic mobile P2P networks
  - Propose an application-layer approach to tackle the link disconnections
    - deploy pointer replications
  - Proposed method could improve the success ratio of object search up to 40% compared with MADPastry

Future Work

- Further reductions of overall traffic without deteriorating the success ratio of object search
  - Each node does not send pointer information by flooding
  - Each node stores all pointer information in overheard or received packets
- Formulate a new metric to model the network dynamics