D2HT: the best of both worlds, Integrating RPS and DHT

Marin Bertier  François Bonnet  Anne-Marie Kermarrec  
**Vincent Leroy**  Sathya Peri  Michel Raynal

IRISA/INRIA/INSA Rennes, France
Introduction

- Peer-to-peer overlays
- Different types for different uses
  - Structured (Distributed Hash Table)
  - Weakly structured (clustering)
  - Unstructured (Random Peer Sampling)
  - ...
- Application/Platform with several overlays at the same time
  e.g. search applications (DHT for exact match, RPS for flood search)
Introduction – DHT

Distributed Hash Table

- Map keys to peers
- usually ring structure
- short and long links
- $O(\log(N))$ routing
Introduction – DHT

- Well known subject
- Many existing DHTs
- Very structured (Chord, Pastry)
- More freedom (HRing, Simphony)
- On the fly deployment: T-Chord
Random Peer Sampling

- Provide each node with a random sample of the network
- Dynamic view, renewed
- Generate random topology, high resilience

Gossip: generic framework for P2P protocols

- Select destination from view
- Send some information
- Compute a new view
Introduction – Goal

- DHT with RPS for free
- SIMPLE protocol (gossip based)
- On the fly deployment
Protocol – Overview

▶ Short-range links: Neighbors Peer Sampling
▶ Long-range links: Kleinberg Peer Sampling
▶ NPS uses KPS for convergence (similar to clustering + RPS)
▶ Gossip-based protocols
▶ No join/leave/repair/massive failure repair . . . just gossip
Protocol – KPS

Derived from Cyclon (RPS protocol)

- Gossip destination selection: increment timestamps and select peer with highest timestamp
- Swapped peers selection: select peers using harmonic probabilities (1/d)
- New view selection: merge not selected peers with received peers

⇒ close to standard RPS, dynamic and random view, but biased on DHT ID
Protocol – NPS

Derived from Vicinity (clustering protocol)

▶ Gossip destination selection: increment timestamps and select peer with highest timestamp

▶ Swapped peers selection: select peers closest to destination from NPS and KPS (DHT ID distance)

▶ New view selection: keep closest peers from NPS, received peers and KPS

Clustering protocol based on DHT ID
Half of the view is selected clockwise, other half counterclockwise
Protocol – NPS

Improvement on gossip destination selection
- Peers maintain a history of peers they have gossiped with
- If peers in NPS are not in history, they have priority (based on ID distance)
- New gossip destinations have more new information
- Once NPS has converged, all the view is in history, use of timestamps only
Evaluation

Evaluation through P2P simulator

- D2HT as a RPS
- D2HT as a DHT
Evaluation – D2HT vs RPS

Compare KPS with Cyclon

- Graph properties very close (average path length, clustering)
  ⇒ flooding performance equivalent
- Peers discovery speed
- Clustering protocol
Evaluation – D2HT vs RPS

Peers discovery speed

Proportion of Nodes Seen

Number of Cycles (x100)

KPS(D2HT)
Cyclon

0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0 1 2 3 4 5 6 7 8 9 10

Number of Cycles (x100)
Evaluation – D2HT vs RPS

Clustering protocol (T-Man, 2D coordinates)

Proportion of Nodes Converged vs Number of Cycles for KPS and Cyclon.
Evaluation – D2HT vs RPS

- Although biased, KPS as good as Cyclon for resilience and flooding
- Slightly faster peers discovery
- Any clustering protocol benefits from a biased RPS for convergence speed
- ⇒ KPS outperforms standard RPS for weakly structured overlays
Evaluation – D2HT vs DHT

Compare with HRing, Symphony and Pastry
► On the fly creation / massive failure recovery
► Routing performance
► Load balancing
Evaluation – D2HT vs DHT

On the fly creation / massive failure recovery

Proportion of Nodes Converged
Number of Cycles
massive crash with history
massive crash no history
on the fly with history
on the fly no history
Evaluation – D2HT vs DHT

Routing performance (more details in the paper)

- D2HT
- HRing
- Symphony
- Pastry

Number of Nodes (x10000)

Hop Count
Evaluation – D2HT vs DHT

Routing performance: reason

Cumulative Links Distribution

ID Distance (normalized)

KPS(D2HT)  HRing  Symphony  Pastry  Reference Kleinberg  Cyclon
Evaluation – D2HT vs DHT

Load balancing: in-degree with uniform ID

Number of Nodes

Node Indegree

D2HT
HRing
Symphony
Pastry
Evaluation – D2HT vs DHT

Load balancing: in-degree with Gaussian ID

Number of Nodes
Node Indegree
D2HT
HRing
Pastry
Evaluation – D2HT vs DHT

Load balancing: hot-source problem

![Graph showing load balancing comparison between D2HT, Symphony, HRing, and Pastry.](image)

- **Node Load**
- **Number of Nodes**

- D2HT
- Symphony
- HRing
- Pastry

Number of Nodes:

- 0
- 2
- 4
- 8
- 16
- 32
- 64
- 128
- 256
- 512
- 1024

Node Load:

- 0
- 2
- 4
- 8
- 16
- 32
- 64
- 128
- 256
- 512
- 1024
Evaluation – D2HT vs DHT

- Deployed on the fly very quickly
- Great churn resilience (massive crash + pro-active recovery)
- Routing performance lower than other DHTs
- Efficient load balancing
Conclusion

- D2HT is a P2P platform oriented DHT
- RPS for free
- KPS improves clustering speed
- Deployed on demand
- Good load balancing
- Lower routing performance