PEER DISCOVERY IN TREE-STRUCTURED P2P OVERLAY NETWORKS BY MEANS OF CONNECTED DOMINATING SETS

Fraunhofer IML

Peter Detzner | Jana Gödeke

RUB

Steffen Bondorf

29. September 2022

Previously on LCN 2021

- Height-balanced [1]:
  - *"… at any node in the tree, the height of any two subtrees of its children differ by at most 1 …"*
- Null-balanced [4]:
  - *Definition: "An m-ary tree is null-balanced if any two leaf nodes differ in level by 1"*

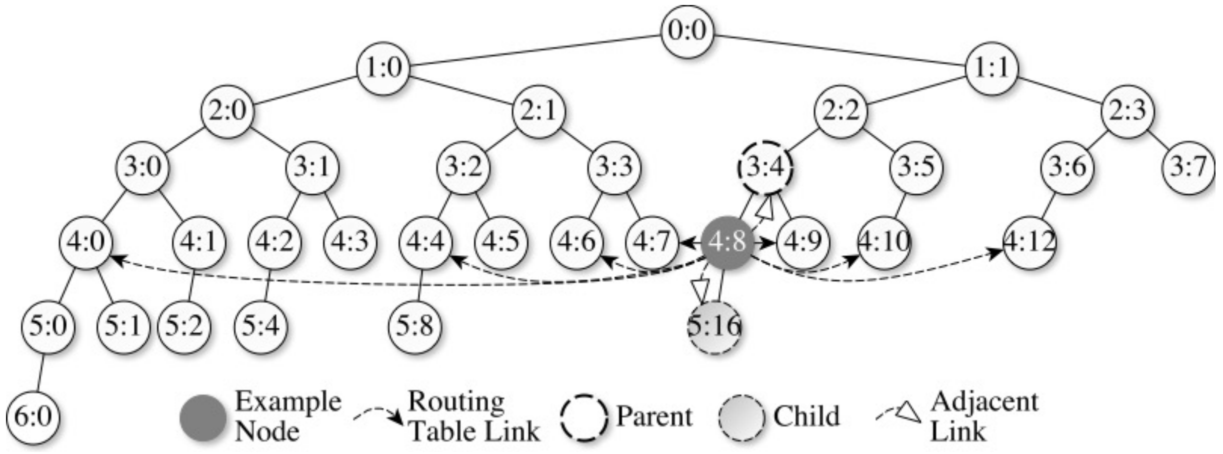


Figure 4: The BATON* tree has a height of 7 and 34 nodes in total

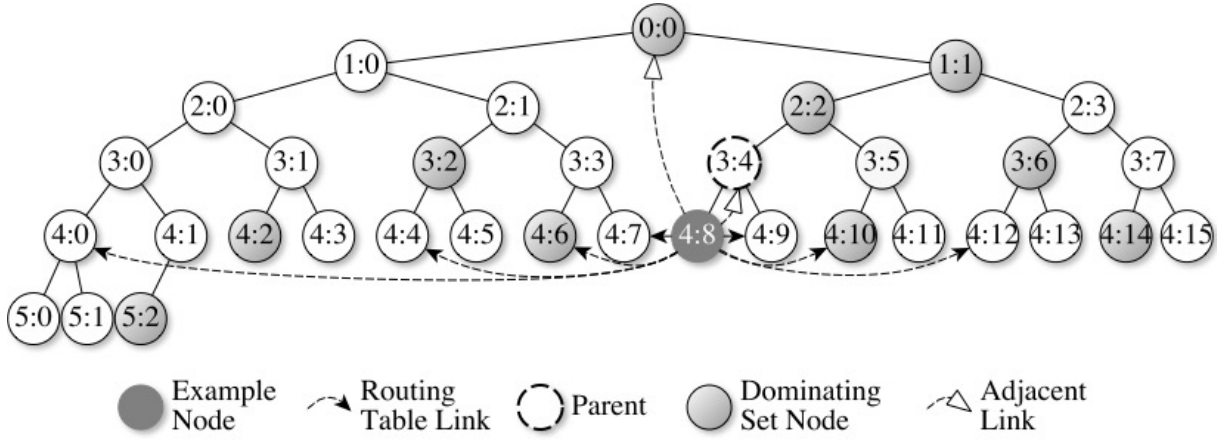


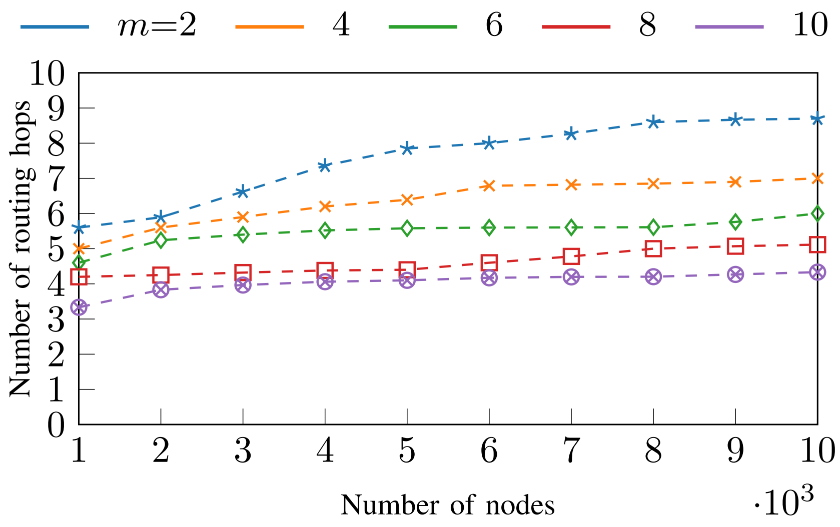
Figure 5: The nBATON* tree has a height of 6 and 34 nodes in total

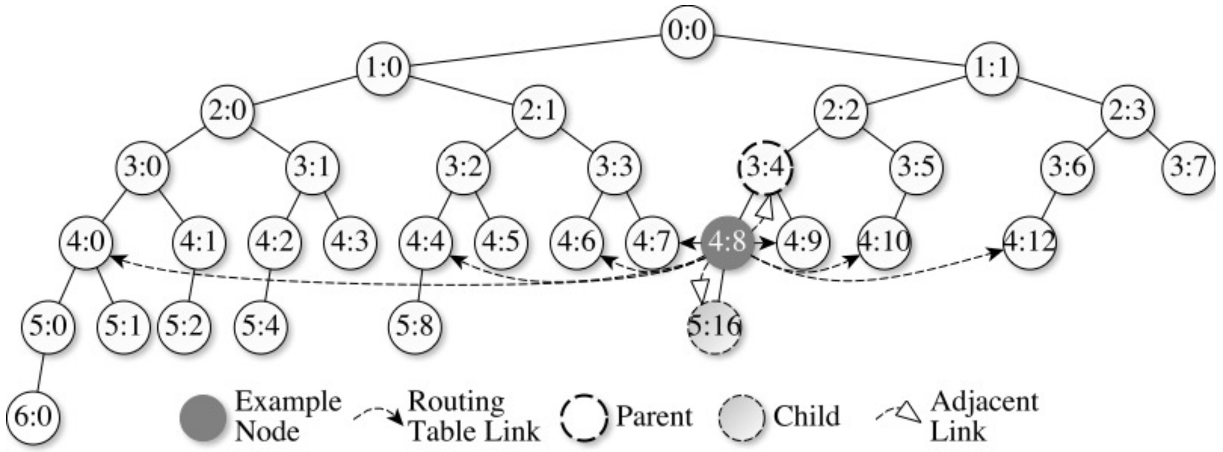
[1] H. V. Jagadish et al., "Speeding up Search in Peer-to-Peer Networks with a Multi-Way Tree Structure," in *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data - SIGMOD '06*

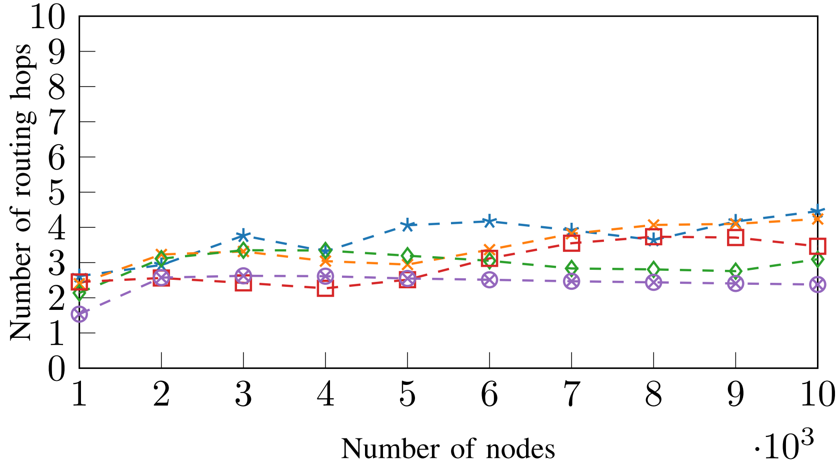
[4] Cha, S.-H., 2012. On integer sequences derived from balanced k-ary trees, in: Proceedings of the 6th WSEAS International Conference on Computer Engineering and Applications

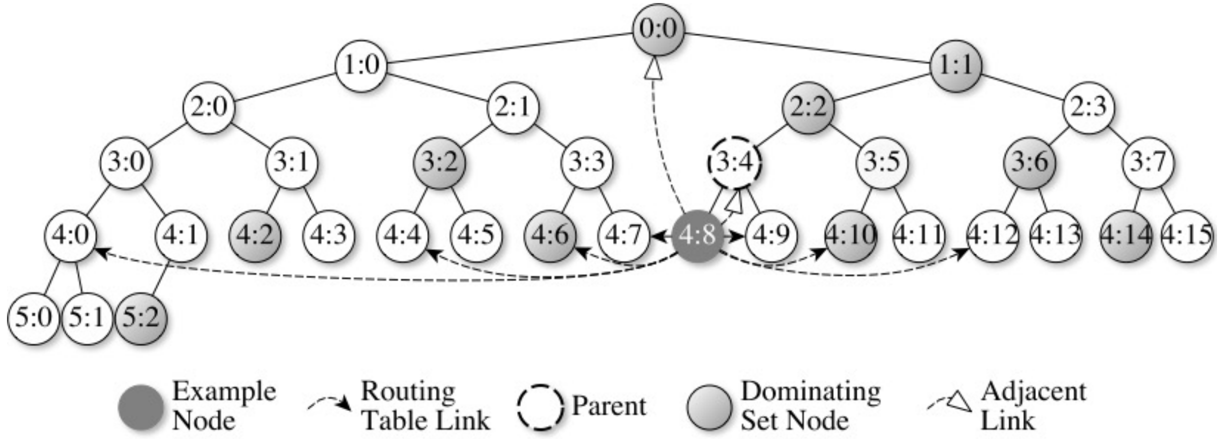
Fraunhofer IML

Previously on LCN 2021 - SearchExact





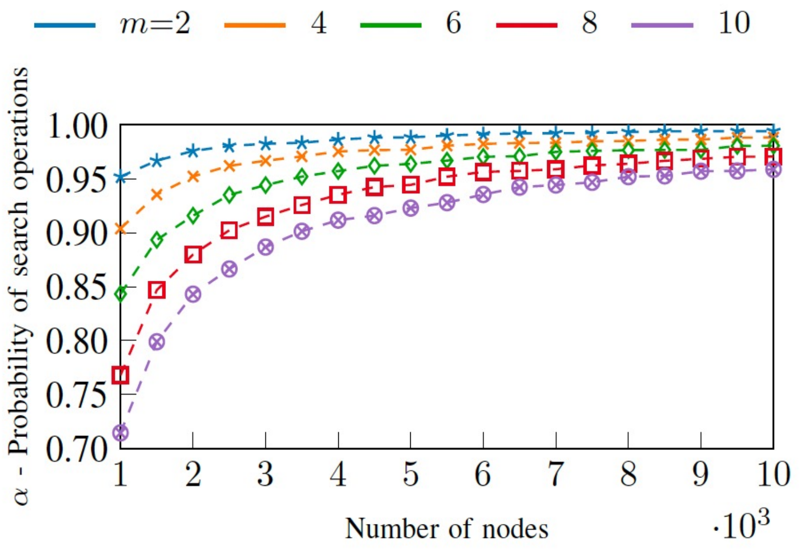




© Fraunhofer IML · open · slide 3

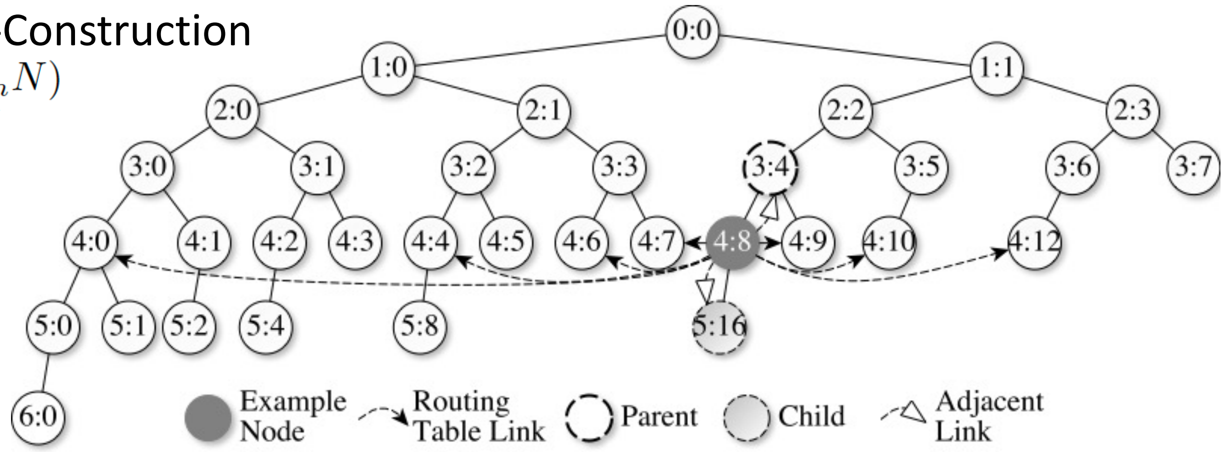
Fraunhofer IML

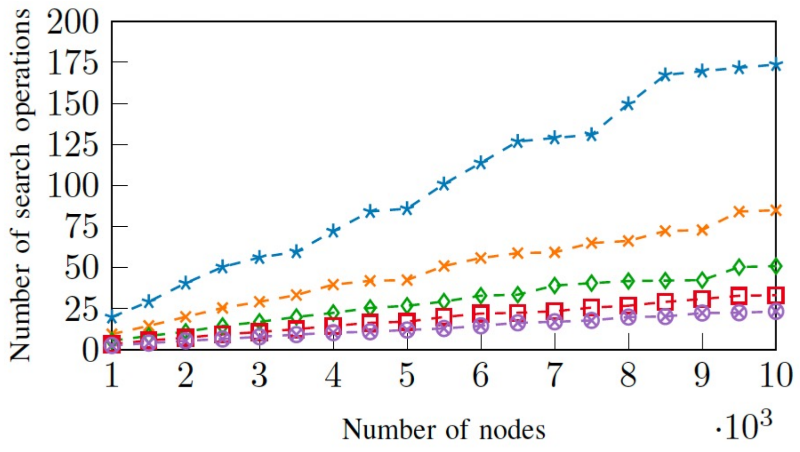
Previously on LCN 2021 – SearchExact Amortization



Cost Tree-Construction

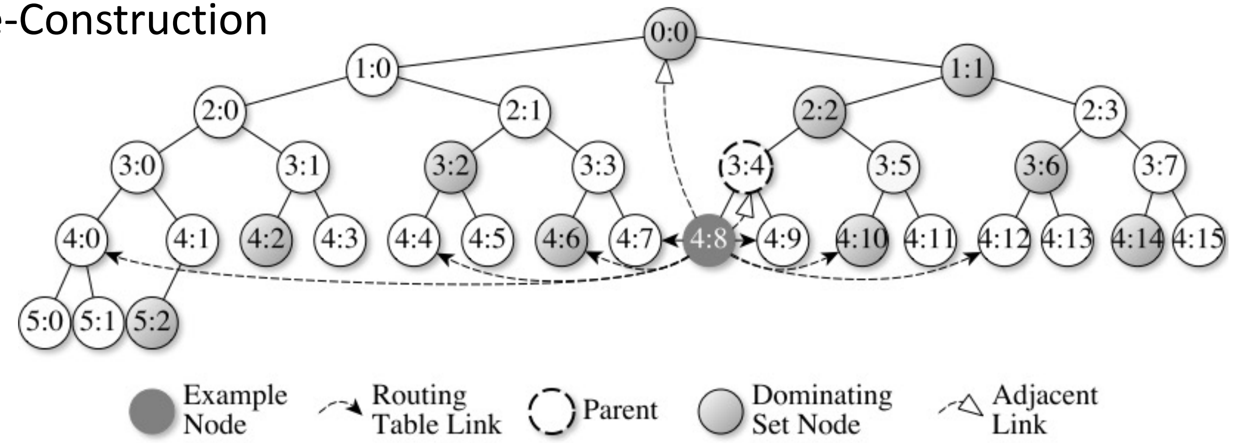
$O(m \cdot log_m N)$





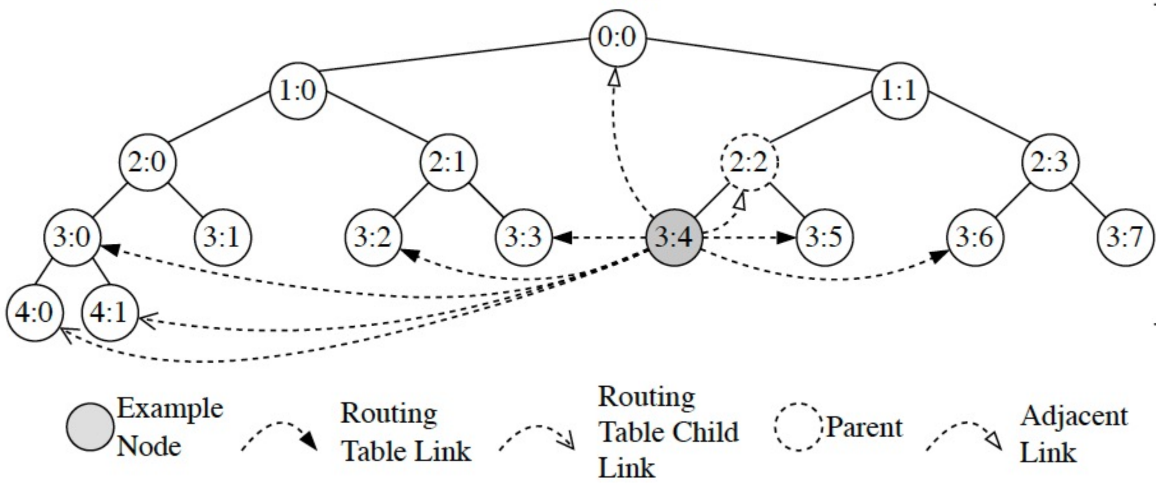
Cost Tree-Construction

$O(\frac{N}{2m^2})$

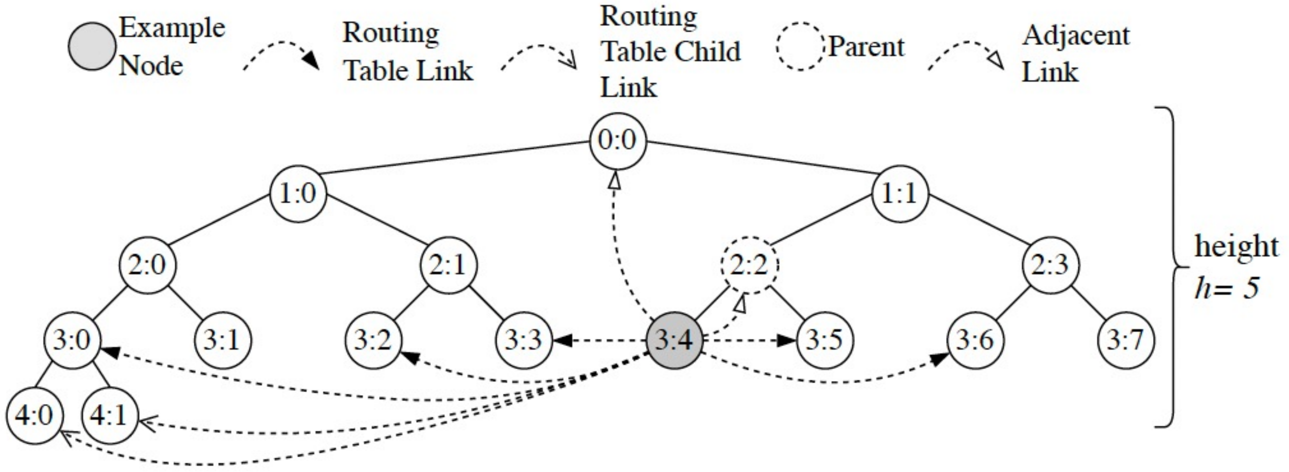


Challenges in Peer Discovery

- Peers are autonomous and have heterogenous in terms of capabilities and properties
- Modelled as Key-Value-Pairs
  - Heterogenous
  - Inserting/ Removing at any time
  - Various Update Intervals
  - Flexible Value Types
  - Complex Discovery Select
    - SQL-like: “select Peers where Conference < 2022 and Title == “IEEE LCN”
- Each peer has only a local view
- Data is scattered all over the network

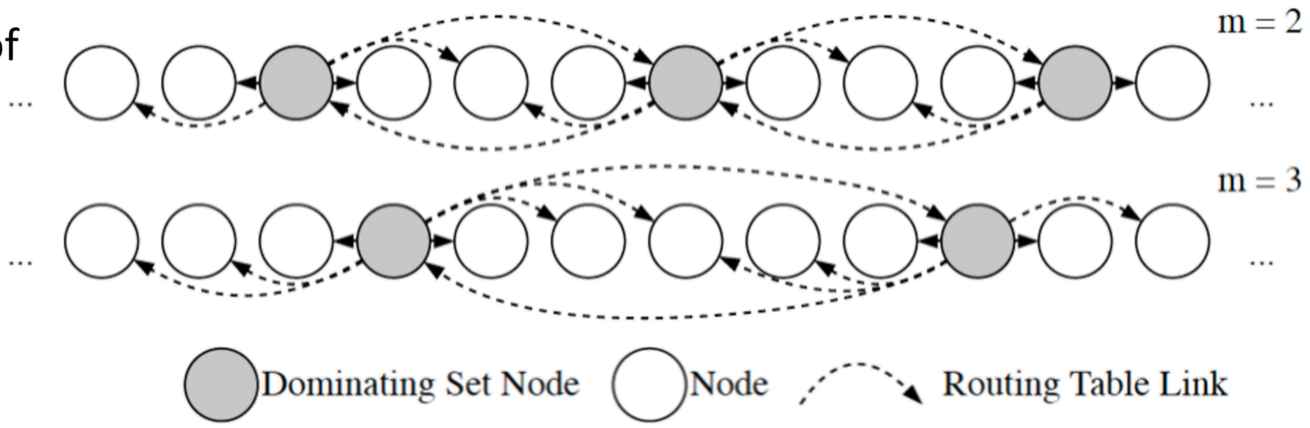


System Model



- **Each Peer maintains links to**
  - Parent, up to m-children,
  - RoutingTable (RT) = LRT + RRT
  - Left- and right adjacent

- **Connected Dominating Set (CDS) [2, 3]:** A subset of peers P in a network N is a dominating set if every peer not in P has a neighbor in P. If the peers in P are connected, then P is called CDS



- **A CDS consists of Dominating Set Nodes**

$$DSN_m(l) = \{i \cdot m \mid i = 2k+1, k \in \mathbb{N}_0, i \cdot m < m^l\}$$

$$DSN_m(0) = \{0\} \qquad DSN_m(1) = \{\lceil \frac{m}{2} \rceil\}$$

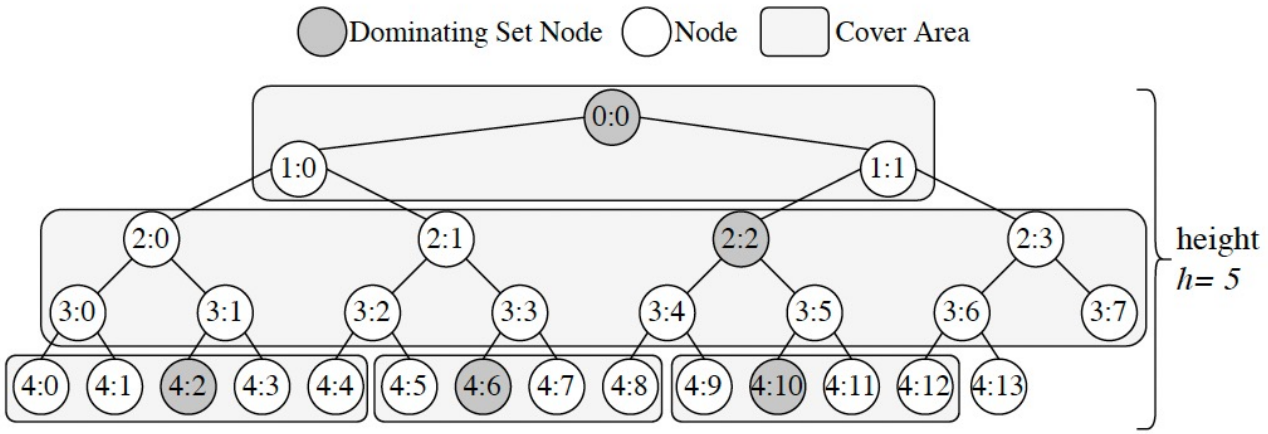
[2] Chunlin Yang and Xiuqi Li, "'Dominating-Set-Based' Searching in Peer-to-Peer Networks," *International Journal of High Performance Computing and Networking* 3, no. 4 (December 1, 2005)

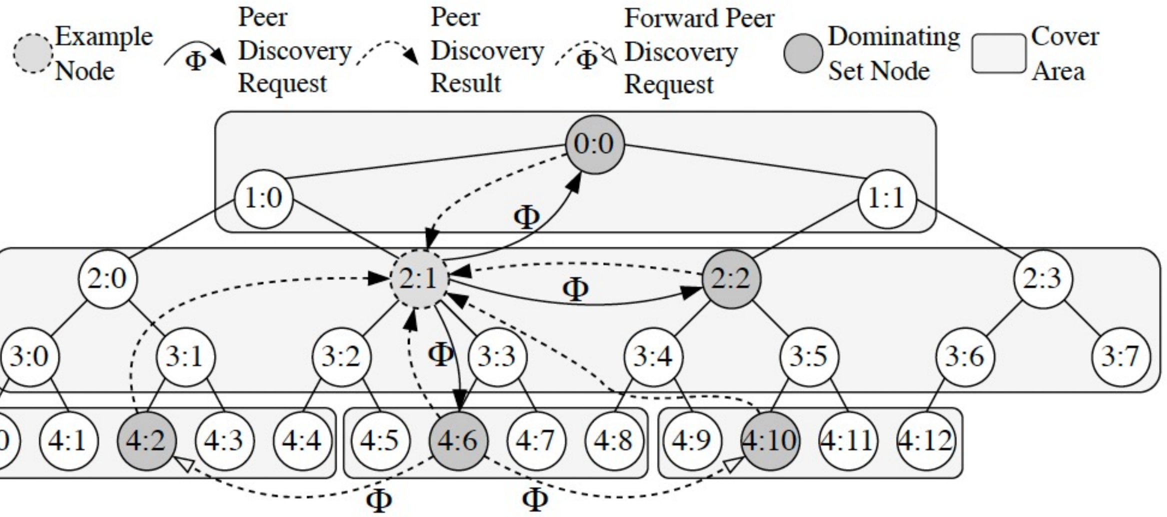
[3] Xiuqi Li and Jie Wu, "Searching Techniques in Peer-to-Peer Networks," Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks (2005): 31.

Fraunhofer IML

An example

- **Each DSN is responsible for a set of nodes**
  - Upper bound: (2m+1)*m
- Scales automatically with the number of nodes
- Nodes can add/update/remove KVPs in O(1)
- No additional communication overhead need
- Knowledge is scattered across the network
- Node (2:1) sends a peer discovery request φ to one DSN
- DSN forwards the request vertically/horizontally to others DSNs





© Fraunhofer IML · open · slide 7

Fraunhofer IML

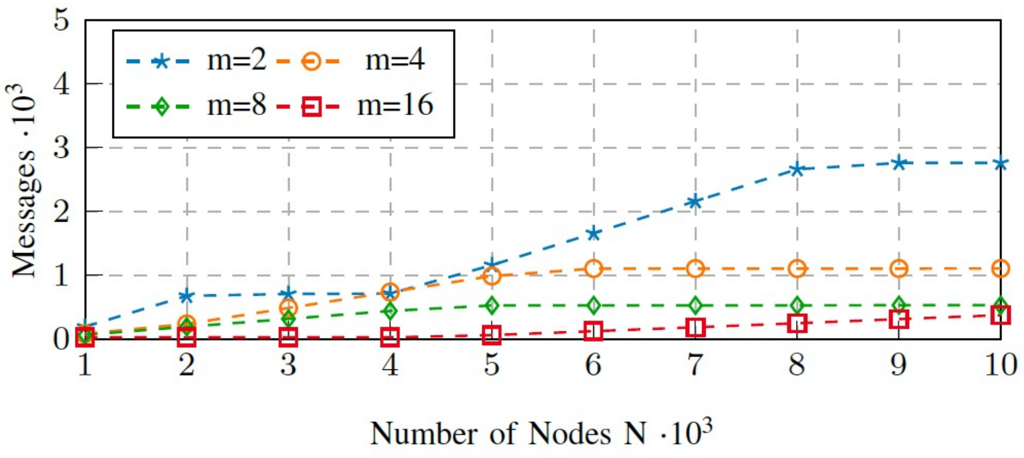
PEER DISCOVERY IN TREE-STRUCTURED P2P OVERLAY NETWORKS BY MEANS OF CONNECTED DOMINATING SETS

Experimental Setup with ns-3

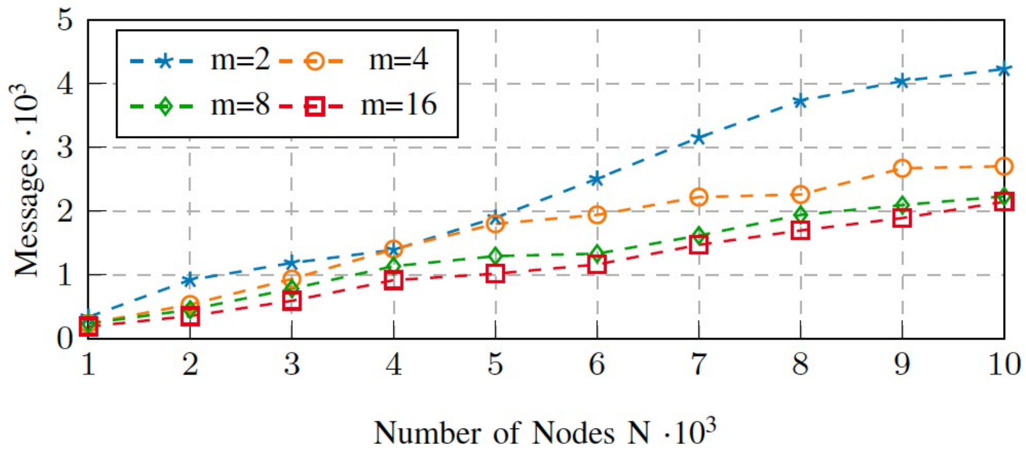
Parameter	ES1	ES2	ES3	ES4
Fanouts $m \in \mathbb{N}$	$m = \{2,\ 4,\ 8,\ 16\}$		$m = \{8\}$	
Network Size $N \in \mathbb{N}$	$N = \{1k,\ 2k,\ 3k,\ \ldots,\ 10k\}$		$N = \{2k,\ 4k,\ 6k,\ 8k,\ 10k\}$	
Requesting Nodes	Randomized distribution; 10% of N			
Request Frequency	Normal distribution; $\mu = 20\ s, \sigma = 4\ s$			
Query Form	$(\psi_i \wedge \psi_j)$ or $(\psi_i \vee \psi_j)$			
Literal Form	$\psi_k := (Key_k \leq x)$			
Maximum Number of KVPs each Node has	10 static	10 dynamic		
KVP Range $\in \mathbb{R}$	Uniform distribution; Interval $[0, 10]$			
Update Frequency for each KVP	-	$\{\mu_1 = 0.625\ s, \sigma_1 = 0.125\ s\}$, $\{\mu_2 = 1.25\ s, \sigma_2 = 0.25\ s\}, \ldots,$ $\{\mu_{10} = 320\ s, \sigma_{10} = 64\ s\}$		
Probability for each KVPs Appearance in a Node	0.95		$\{0.5, 0.6, \ldots, 1.0\}$	0.95
Freshness Threshold for KVPs in a DSN	$5\ s$			$\{1, 3, 5, \ldots, 21\}\ s$

KVP = Key-Value Pair; DSN = Dominating Set Node

ES1



ES2

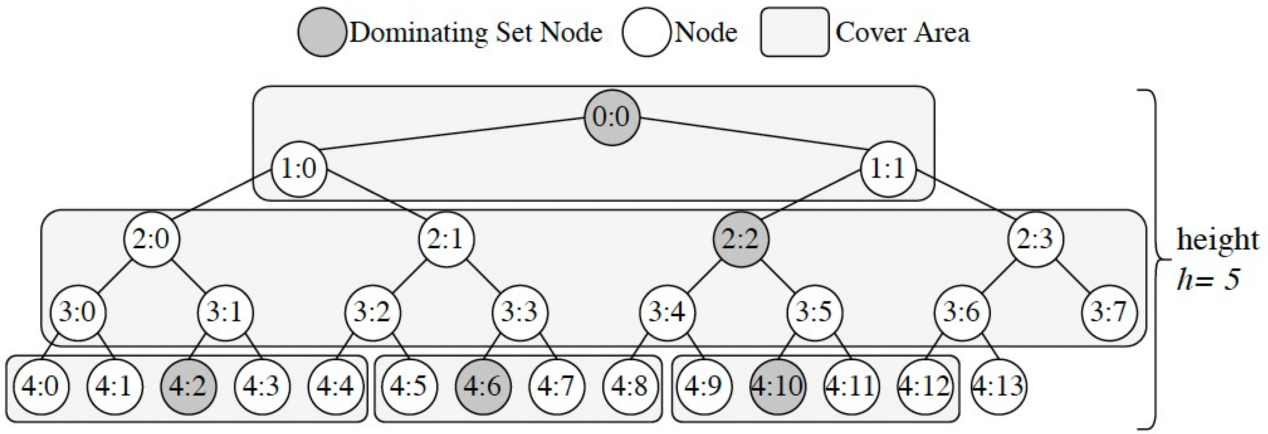


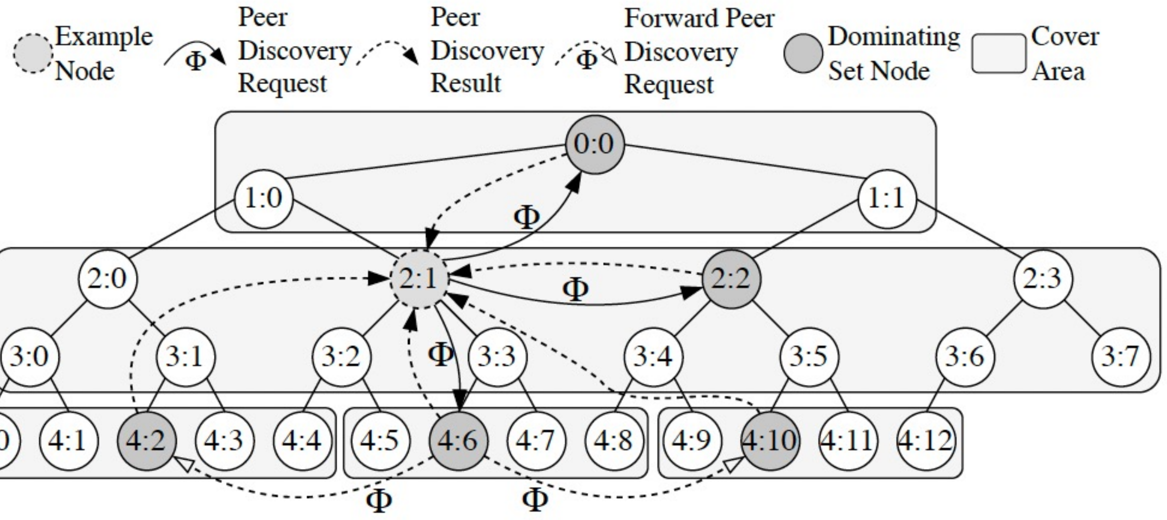
Fraunhofer IML

Conclusion and Future Work

- Creation of a Null-balanced tree
- Finding peers according to a peer discovery query
- Approach is scalable, requires no additional communication overhead
- Cost O(1) for adding/updating/removing data

- Future Work:
  - Minimizing the cost for retrieving data
  - Minimizing the cost for creating the tree
  - Investigation of efficient broadcast algorithms





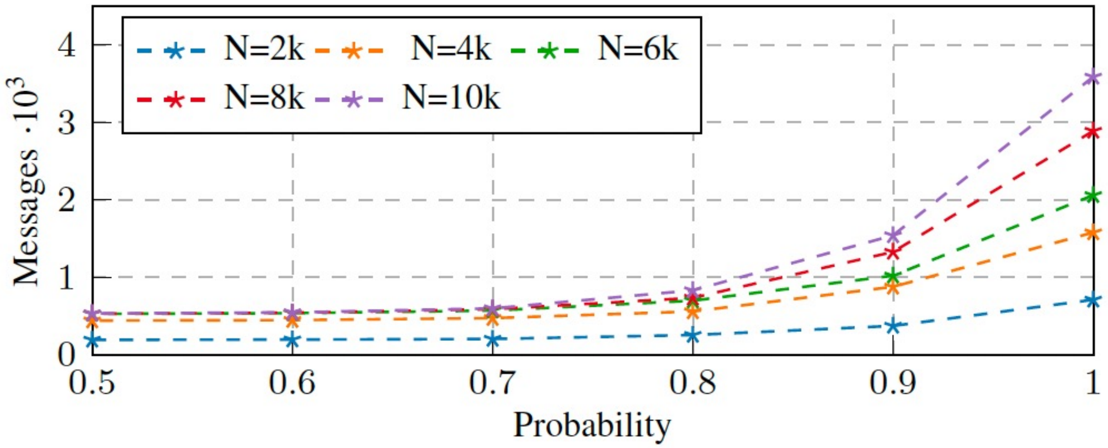
Fraunhofer IML

Experimental Setup with ns-3

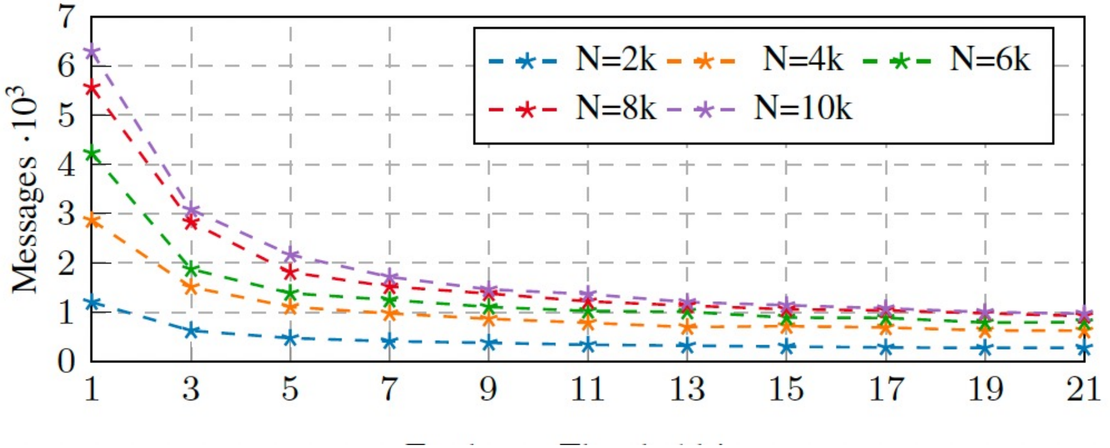
Parameter	ES1	ES2	ES3	ES4
Fanouts $m \in \mathbb{N}$	$m = \{2,\ 4,\ 8,\ 16\}$		$m = \{8\}$	
Network Size $N \in \mathbb{N}$	$N = \{1k,\ 2k,\ 3k,\ \ldots,\ 10k\}$		$N = \{2k,\ 4k,\ 6k,\ 8k,\ 10k\}$	
Requesting Nodes	Randomized distribution; 10% of N			
Request Frequency	Normal distribution; $\mu = 20\ s, \sigma = 4\ s$			
Query Form	$(\psi_i \wedge \psi_j)$ or $(\psi_i \vee \psi_j)$			
Literal Form	$\psi_k := (Key_k \leq x)$			
Maximum Number of KVPs each Node has	10 static	10 dynamic		
KVP Range $\in \mathbb{R}$	Uniform distribution; Interval $[0, 10]$			
Update Frequency for each KVP	-	$\{\mu_1 = 0.625\ s, \sigma_1 = 0.125\ s\}$, $\{\mu_2 = 1.25\ s, \sigma_2 = 0.25\ s\}, \ldots,$ $\{\mu_{10} = 320\ s, \sigma_{10} = 64\ s\}$		
Probability for each KVPs Appearance in a Node	0.95		$\{0.5, 0.6, \ldots, 1.0\}$	0.95
Freshness Threshold for KVPs in a DSN	5 s			$\{1, 3, 5, \ldots, 21\}$ s

KVP = Key-Value Pair; DSN = Dominating Set Node

ES3



ES4



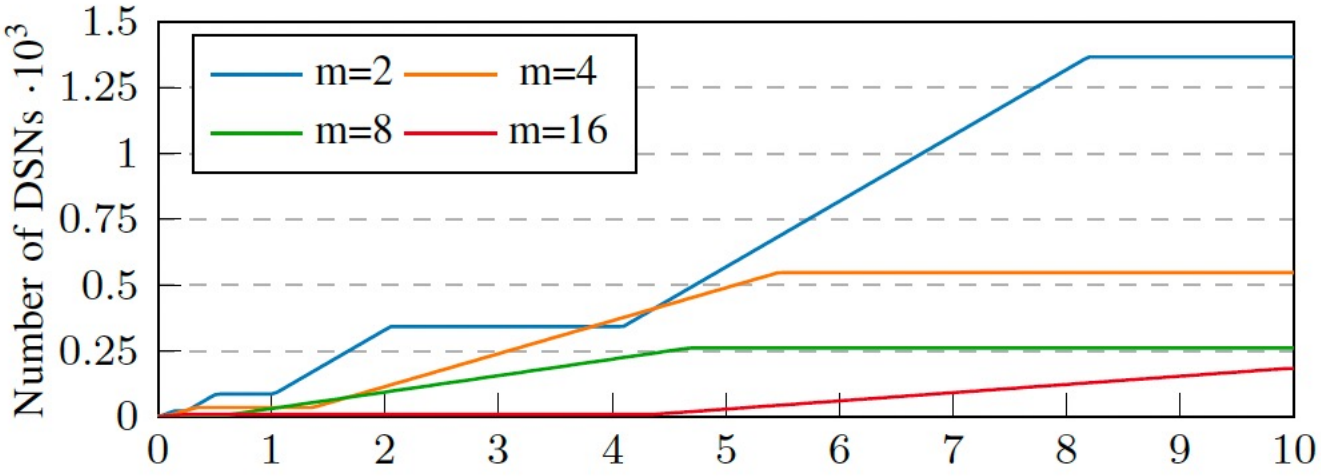
© Fraunhofer IML · open · slide 11

Experimental Setup with ns-3

- Number of DSN scales with the network

- m -> infinity results n a star-topology

- Query evaluation based on local view



DSN	CA	K_1	freshness	ψ_1	K_2	freshness	ψ_2	ϕ
(0:0)	(0:0)	4	✓	t	1	✓	t	t
	(1:0)	10	✓	f	1	✗	u	f
	(1:1)	6	✗	u	-	-	u	u
(2:2)	(2:0)	3	✗	u	1	✓	t	u
	(2:1)	1	✗	u	2	✗	u	u
	(2:2)	2	✓	t	-	-	f	f
	(2:3)	10	✗	u	1	✓	t	u

CA = cover area; t = true; f = false; u = undecided

Fraunhofer IML