Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Institute of Computer Science
Department of Distributed Systems
Prof. Dr.-Ing. P. Tran-Gia

On the Trade-Off between Efficiency and Congestion in Location-aware Overlay Networks - Example of a Vertical Handover Support System

Thomas Zinner, Simon Oechsner, Tobias Hoßfeld,
Phuoc Tran-Gia

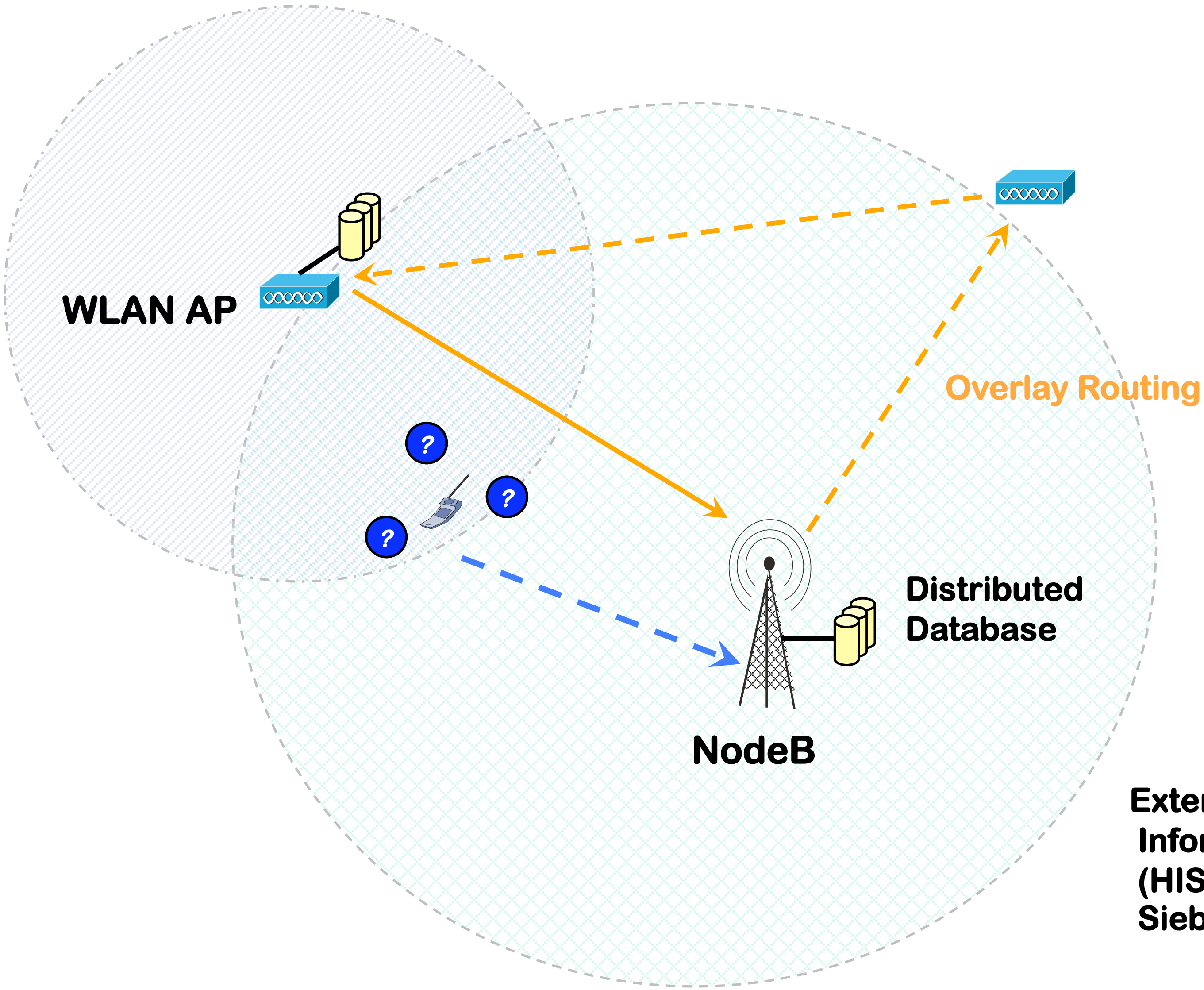
www3.informatik.uni-wuerzburg.de

8th International Conference on Peer-to-Peer Computing P2P'08,
Aachen, Germany, September 11th 2008

Agenda

- System Description
  - Vertical Handover (VHO)
  - A Distributed System for Supporting a Seamless VHO
- Model Description
  - Simulation Scenario
  - Peer Model
- Congestion vs. Efficiency
  - Why Congestion?
  - Adaptive Mechanism to Solve Congestion
  - Results
- Conclusion

Our Approach: Distribution of Database



Extension of the Hybrid Information System (HIS) proposed by Siebert et al. 2004

Problem Formulation

- Locality-bound measurement information has to be stored and made available (e.g. signal strength)
- Large number of handover requests has to be served
(more than 82 Mio mobile phones in Germany, Bitkom 2006)
- Fast response times are needed for viable handover

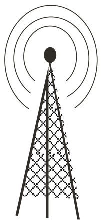
→ **VHO support system must be scalable and efficient**

How to Distribute the Database?

- Scalability
  - Basic idea is to split up the database among the peers
  - Use each AP, i.e. NodeB and WLAN Access Point, in the access networks as a peer
- Efficiency
  - Data should be stored close to the location where it's needed
  - Geographically close peers should be also close in the overlay (geographic layout)
- Our Solution
  - Reliable distributed search implemented by a DHT
  - DHT-Overlay to perform the interconnection of the distributed database
  - **Symmetric location-aware overlay metric required**

Pastry and its Modifications

- Used DHT: Pastry algorithm with modifications:
  - Location-aware ID: $x_1 y_1 x_2 y_2 \ldots$ encoded to base b
  - Technology-aware ID: $t_1 t_2 x_1 y_1 x_2 y_2 \ldots$
  - Distance d between two peers p,q: $$d(p,q) = \left| \sum_{i=0}^{m} (p_i - q_i) \cdot b^i \right|$$
  - Introduction of a Shortcut link: next slide



x = 243 m,
y = 53 m

$base_4(243)$ = **3 3 0 3**
$base_4(53)$ = **0 3 1 1**

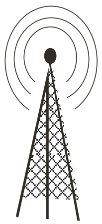
ID : 3 0 3 3 0 1 3 1

Location-aware

ID: 0 3 0 3 3 0 1 3 1

length of m digits

Technology & Location-aware



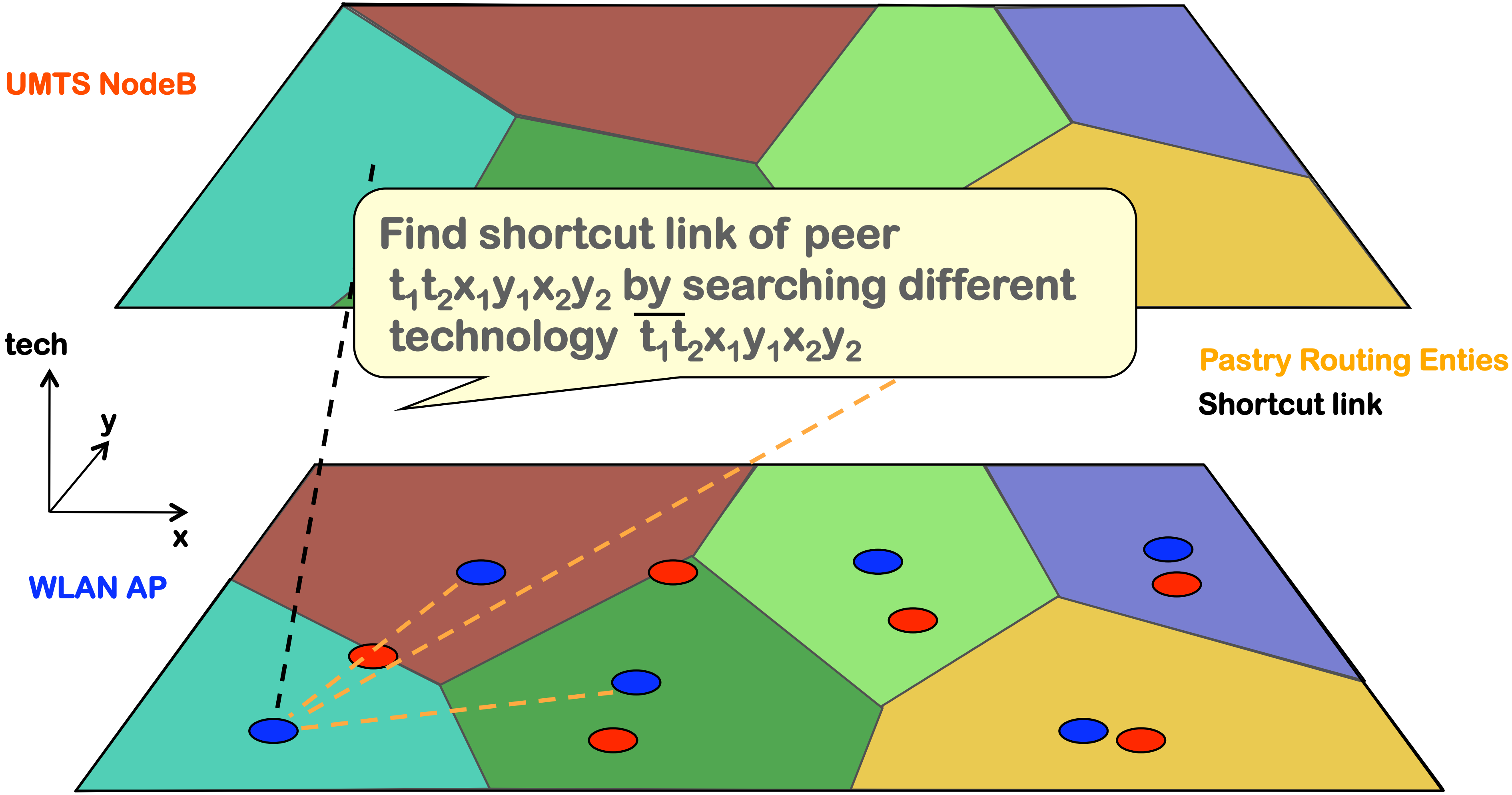
x = 213 m,
y = 73 m

ID_1: 0 3 0 3 3 0 1 3 1

ID_2: 0 3 1 1 0 1 2 1 1

d (ID_1, ID_2) = d (0 0 1 2 3 1 1 2 0)
= 7000

Responsibility Areas, Links and Shortcuts



Simulation Scenario

- Simuative performance analysis of this system was conducted in "*Evaluation of a Pastry-based P2P Overlay for Supporting Vertical Handover*" (Hoßfeld et al., 2006) ➔proof of search efficiency

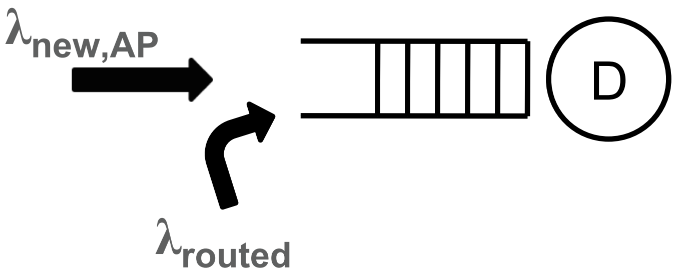
- But: No queuing model was considered for the peers
➔ No Statements about load distribution, robustness,...

- This work refines this simulation model:
  - t_p :Processing time at a peer (Routing, DB access)
  - t_n : Network transmission delay

- City-sized setup:
  - 1783 NodeBs (real antenna locations)
  - 879 WLAN APs (distributed randomly)

Queueing Model at a Peer

- Peer is modeled as a GI/GI/1 - n waiting system with finite queue and FIFO processing strategy
- Total external query load λ_{new}: 100.000 msg/s on average for all peers, generated by mobile subscribers
- Internal routing load λ_{routed} generated by DHT routing (previous studies: less than 2 hops per request on average)



Simple Calculation

- For the investigated scenario the mean load for the APs can be estimated simply as
  - $\rho_{WLan,AP} = \rho_{new,AP} + \rho_{routed,NodeB}$
    $= \lambda_{new,AP} \cdot t_s + (E[h]-1) \cdot t_s \cdot \lambda_{new,AP} \cdot 2 + t_p \cdot \lambda_{new,AP} \cdot 2$

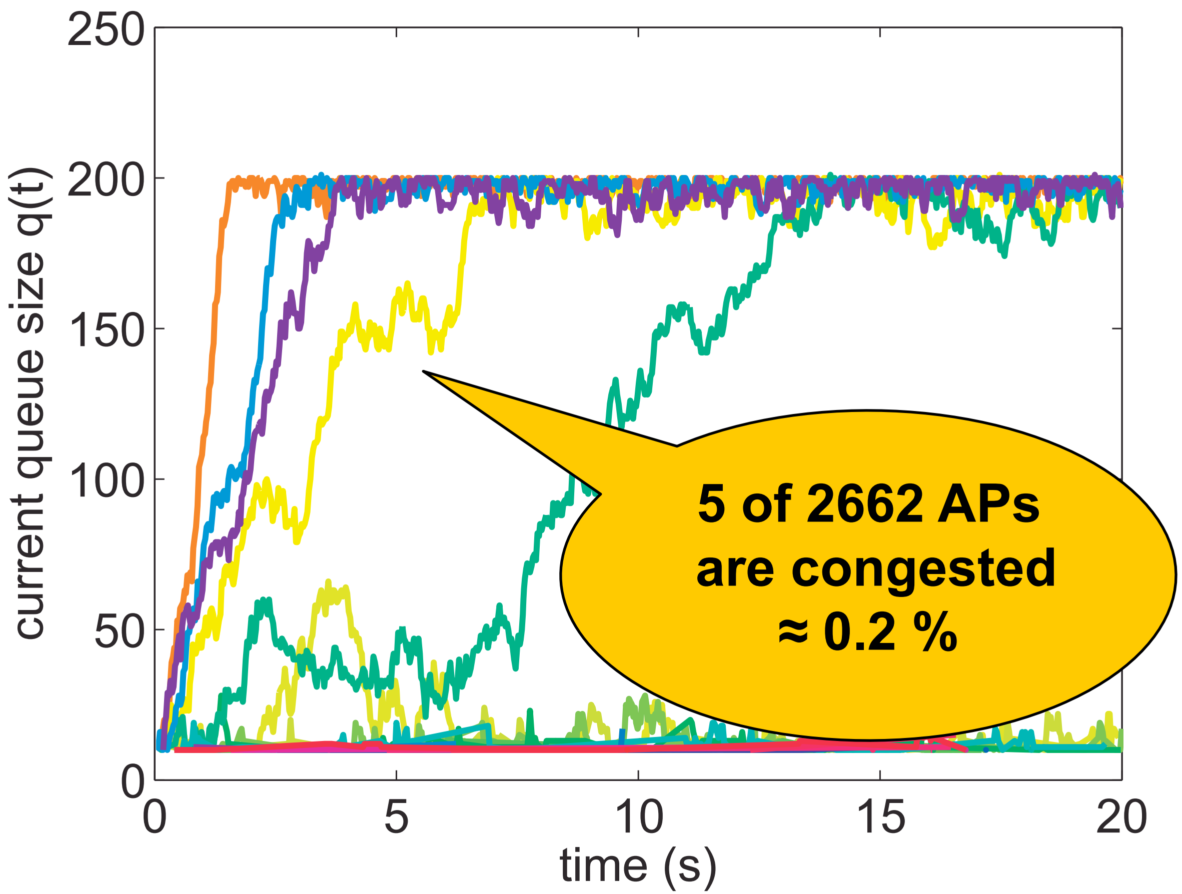
    $=$ **0.398**
  - $\rho_{NodeB,AP} = \rho_{new,AP} + \rho_{routed,WLan}$

    $=$**0.161**

➔ **From this point of view, the system is well dimensioned, under the assumption of perfect load distribution**

Investigation of the buffer occupancy

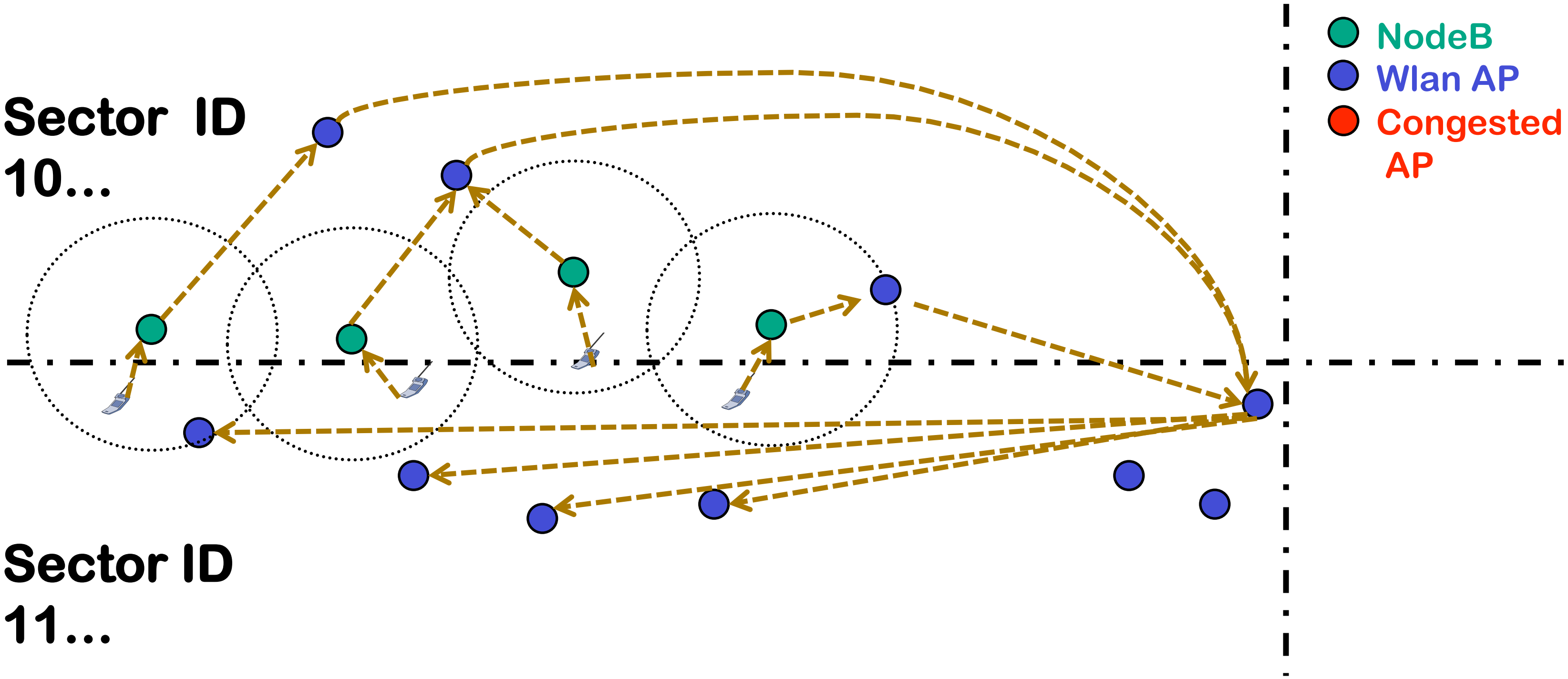
- Investigation of a 1 Mbit/s scenario with t_s = 2 ms
- Maximum queue size q_s = 200



➔ Only a few Attachment Points (in this case WLAN APs) are congested, the rest of the system performs well

Why Congestion?

- By the ID generation algorithm for the geographic layout, the ID space is essentially divided into zones
- Due to the modified Pastry routing algorithm, all 'border' traffic from one zone into another is routed to one node in the target zone

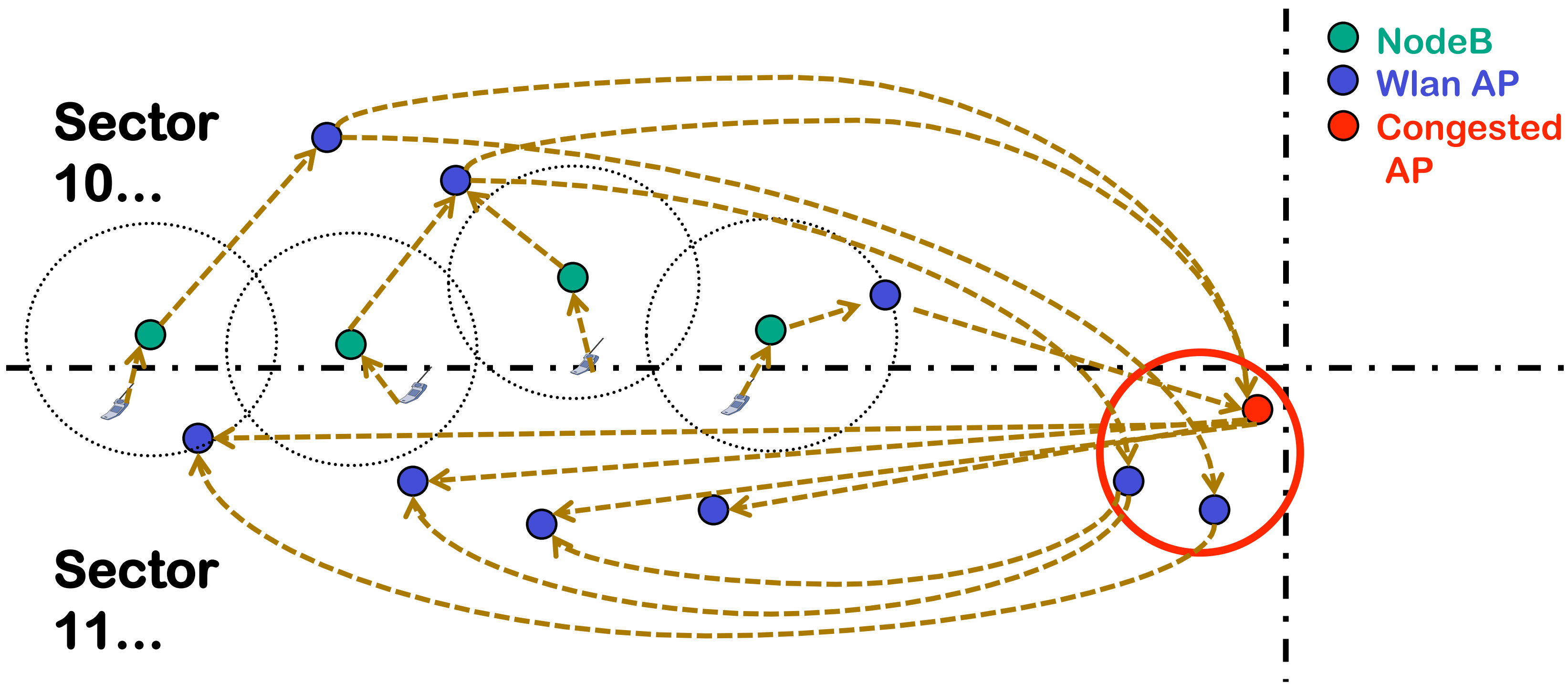


Congestion

- The optimization that improves the performance of the system in this case also leads to a congestion problem
- While the system as a whole is well dimensioned for the offered load, the load is not distributed evenly on all nodes
- Due to the layout and routing, a few nodes have become bottlenecks
- Possible solutions:
  - Other distance function ➔ may be a solution for the specific system
  - Faster processors, better connections ➔ more expensive
  - Distribute the ‘border’ traffic on other APs which can handle this traffic as good as the congested AP ➔ next slides

New Algorithm to Avoid Congestion

- Distribute inter-zone traffic by redirecting links in the Pastry routing table



Self – Organizing Routing Algorithm

Algorithm 2 Self-Organizing Routing Algorithm

input : A = id of local node

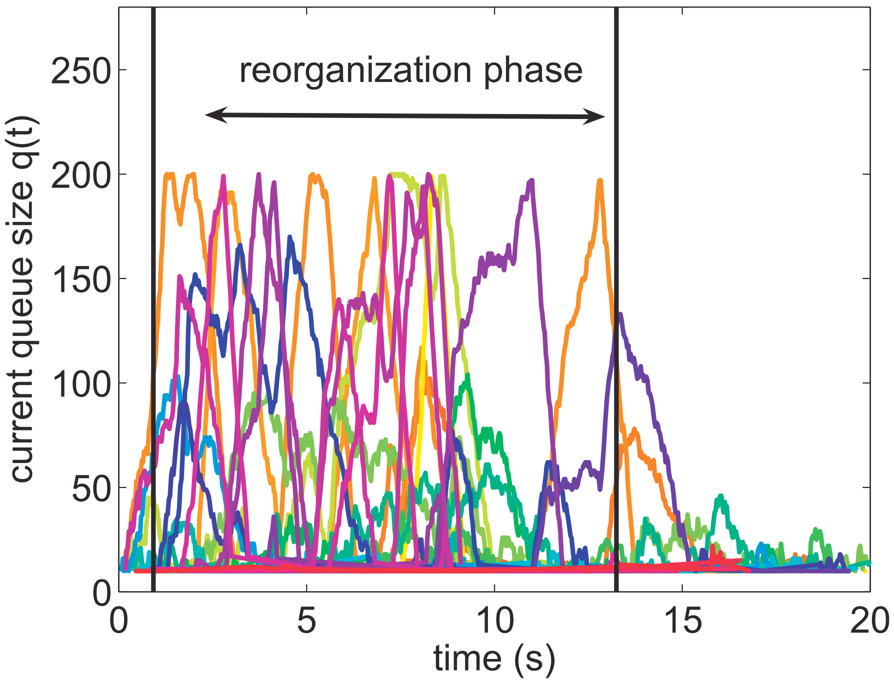
1: **if** $(f(k_1, k_2, \ldots k_l) < \tau)$ **then**
2: normal routing procedure
3: **else if** $(\tau < f(k_1, k_2, \ldots k_l) < \upsilon)$ **then**
4: Congestion Avoidance Algorithm with (a_r, a_s) {example algorithm used in this paper}
5: **else if** $(f(k_1, k_2, \ldots k_l) \geq \upsilon)$ **then**
6: Congestion Avoidance Algorithm with (a'_r, a'_s) {Redefine thresholds in such a way that $a'_r \leq a_r, a'_s \leq a_s$ }
7: **end if**

- Congestion Avoidance: Redirecting Boarder Traffic if overload occurs

➔ Straight forward solution which is not problem specific any more

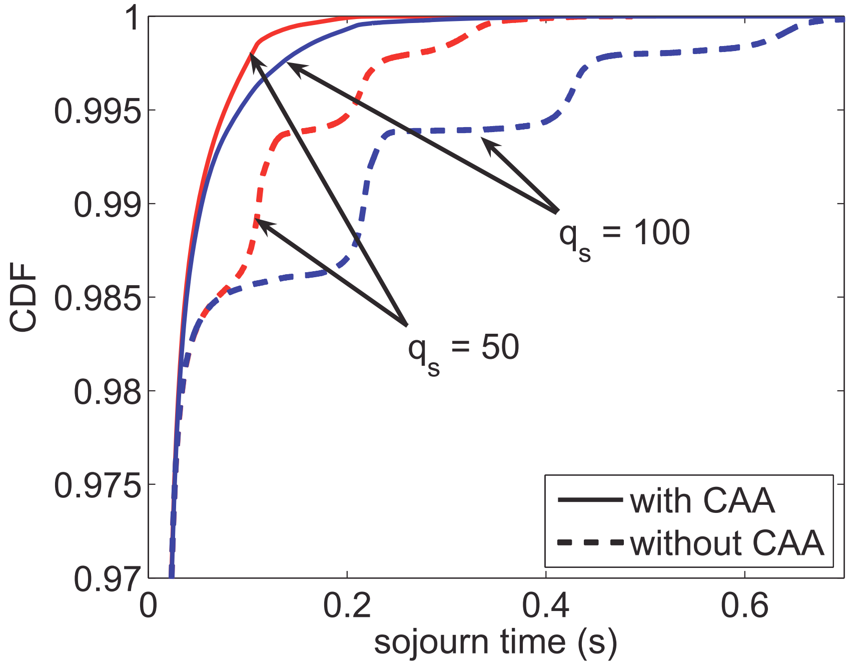
Results for the Congestion Avoidance Algorithm (CAA)

- Queue size over time



- Maximum queue size $q_s = 200$
- No congestion occurs

- Sojourn time with and without CAA for different queue sizes



- Sojourns times decrease by using the CAA
- Less requests are discarded

Conclusion

- Implementation of a DHT – Mechanism to support a seamless Vertical Handover between different Access Technologies

- But: Identification of a weak point:
  - Efficiency leads to congestion
  - Congestion leads to a decrease of efficiency for certain areas

- Our Suggestion: Congestion Avoidance Algorithm (CAA): Adaptive adjustment of the routing behavior of congested peers

- Further work:
  - Mechanism is very simple (based on fixed thresholds): Optimization of the presented mechanism

Q & A

BACKUP

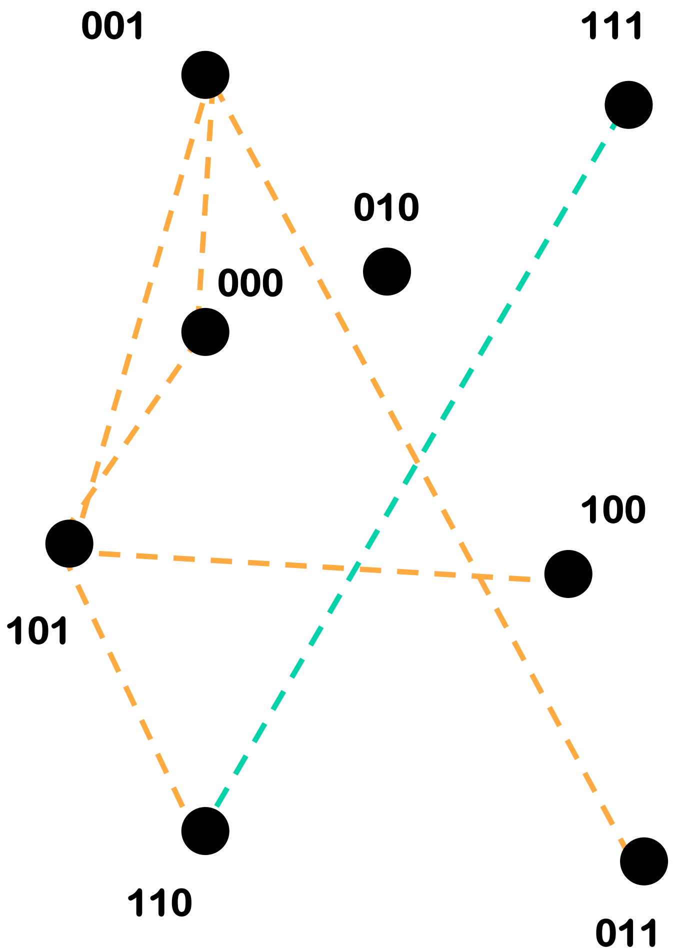
Simple Calculation without Answer Messages

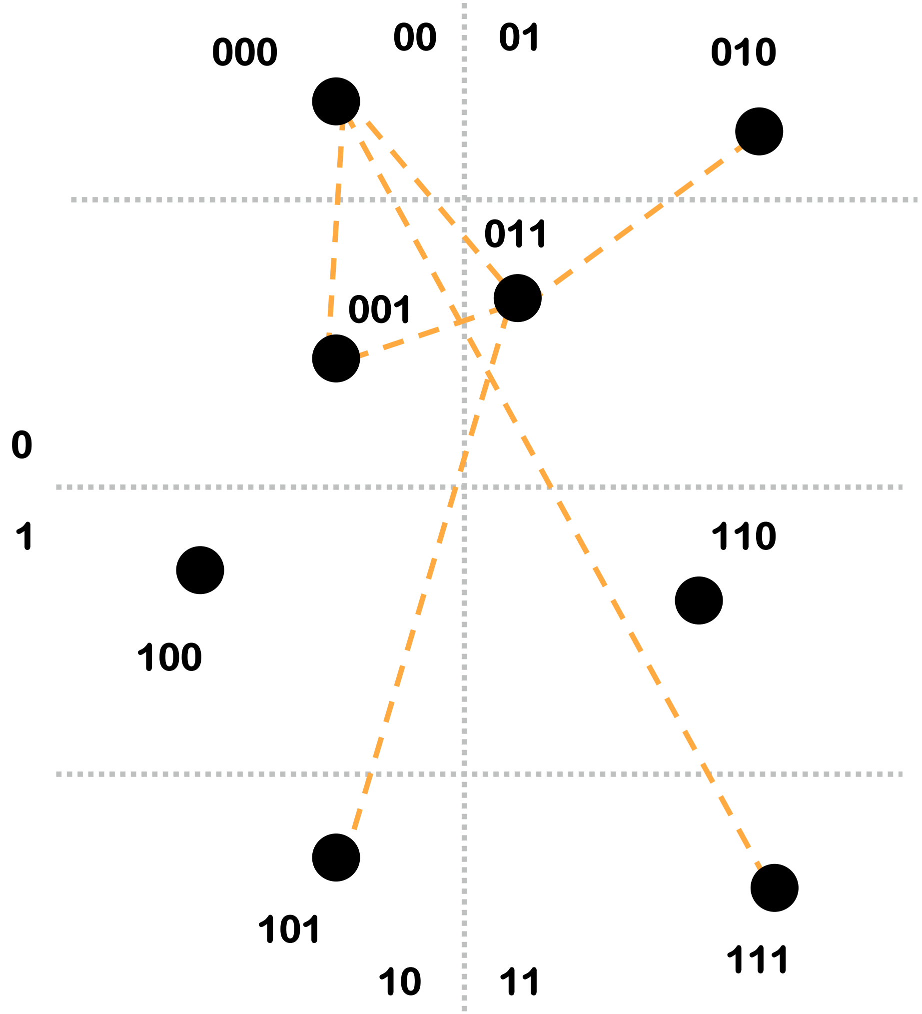
- External load λ_{new} = 100,000 msg/s ➔ 38 msg/s per AP
- t_p = 2 ms, t_n =0.204 ms (Capacity 1 Mbit/s) ➔ t_s = 2.204 ms
- $n_{WLan} : n_{NodeB} = 1 : 2$
- 2 hops per request on average: E[h]=2
- Mean Load per peer on average:
  - $\rho_{WLan,AP} = \rho_{new,AP} + \rho_{routed,NodeB}$
    $= \lambda_{new,AP} \cdot t_s + \underbrace{(E[h]-1) \cdot t_s \cdot \lambda_{new,AP}}_{\text{Forwarded requests}} \cdot 2 + \underbrace{t_p \cdot \lambda_{new,AP}}_{\text{Finished requests}} \cdot 2$
    $= 0.398$
  - $\rho_{NodeB,AP} = \rho_{new,AP} + \rho_{routed,WLan}$
    $=0.161$

Twice as many NodeBs as WLan Aps

➔ From this point of view, the system is well dimensioned

Pastry and its Modifications





- Random ID: $r_1r_2r_3r_4$…

- Location-aware ID: $x_1y_1x_2y_2$…
- Pastry routing not changed

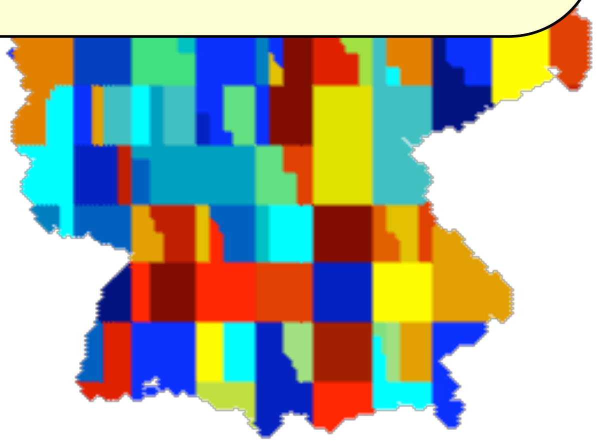
Responsibility Areas

- Measurements are a… closest peer accordi… prefix matching
- UMTS NodeBs only… measurements, WL… WLAN measuremen…
- Replacement of the standard Pastry ID (SHA-1 hash) by a location- and technology-aware ID: $t_1t_2x_1y_1x_2y_2$

→ overlay structure is adapted to physical network layout

Problem: Consider UMTS to WLAN handover

- **UMTS nodeB knows only some WLAN APs (with prefix 0)**
- **unnecessary routing (between WLAN APS) to find a WLAN AP close to the UMTS nodeB**



Why Peer-to-Peer?

A peer-to-peer system might offer several advantages:

- Scalability (ability to handle a large number of nodes, requests)
- Load distribution
- Fault tolerance (no central point of failure)
- Self-organization
- Network can be made up of inexpensive components

Hop distributions

- Connection to server modeled as one hop
- Normal Pastry takes log(N) hops to route messages
- **Shortcut improvement cuts off the most ineffective part of the search**
- **Our modifications improve the routing by shortening the search path due to local searches**