Peer-to-Peer Networks

14 Security

Christian Schindelhauer
Technical Faculty
Computer-Networks and Telematics
University of Freiburg

Cuckoo Hashing for Security

- Awerbuch, Scheideler, Towards Scalable and Robust Overlay Networks
- Problem:
  - Rejoin attacks
- Solution:
  - Chord network combined with
  - Cuckoo Hashing
  - Majority condition:
    - honest peers in the neighborhood are in the majority
  - Data is stored with O(log n) copies

Cuckoo Hashing

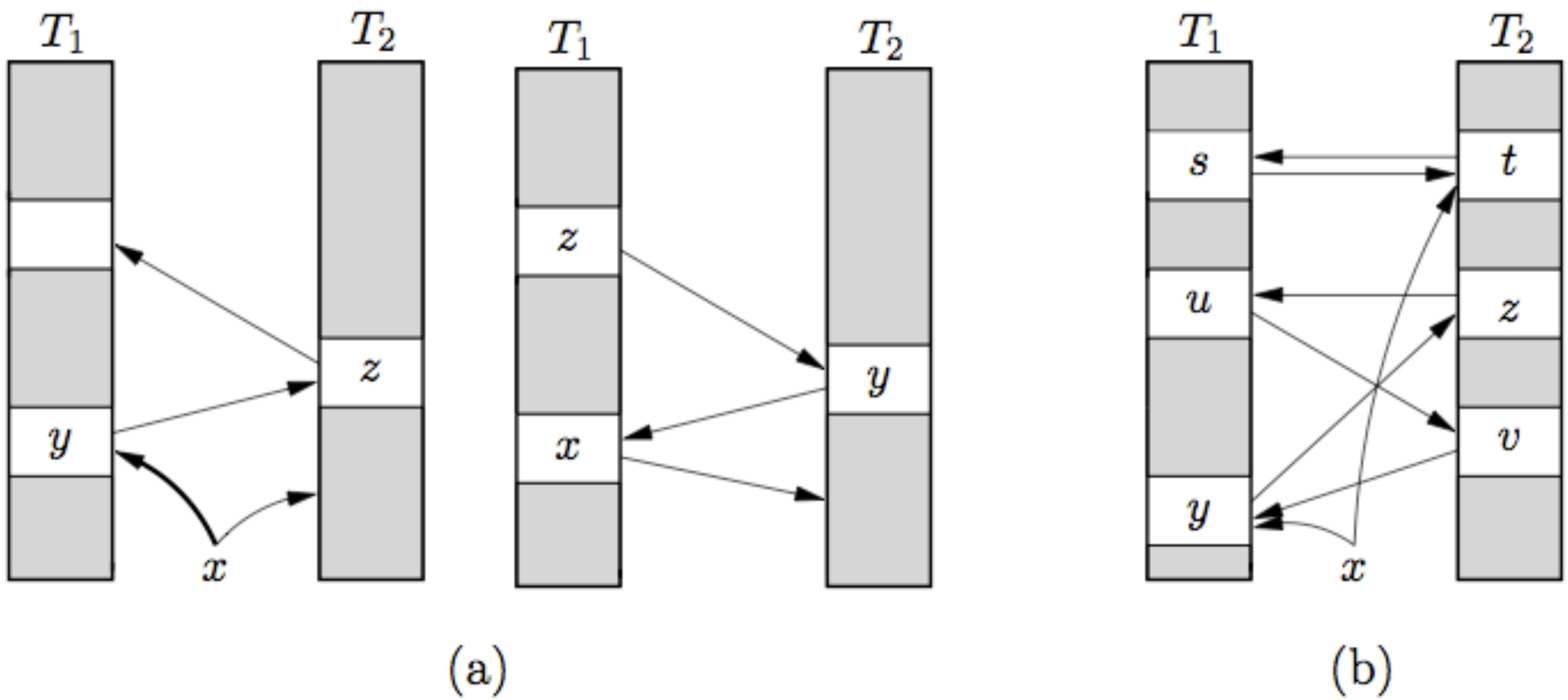


Fig. 1. Examples of CUCKOO HASHING insertion. Arrows show possibilities for moving keys. (a) Key x is successfully inserted by moving keys y and z from one table to the other. (b) Key x cannot be accommodated and a rehash is necessary.

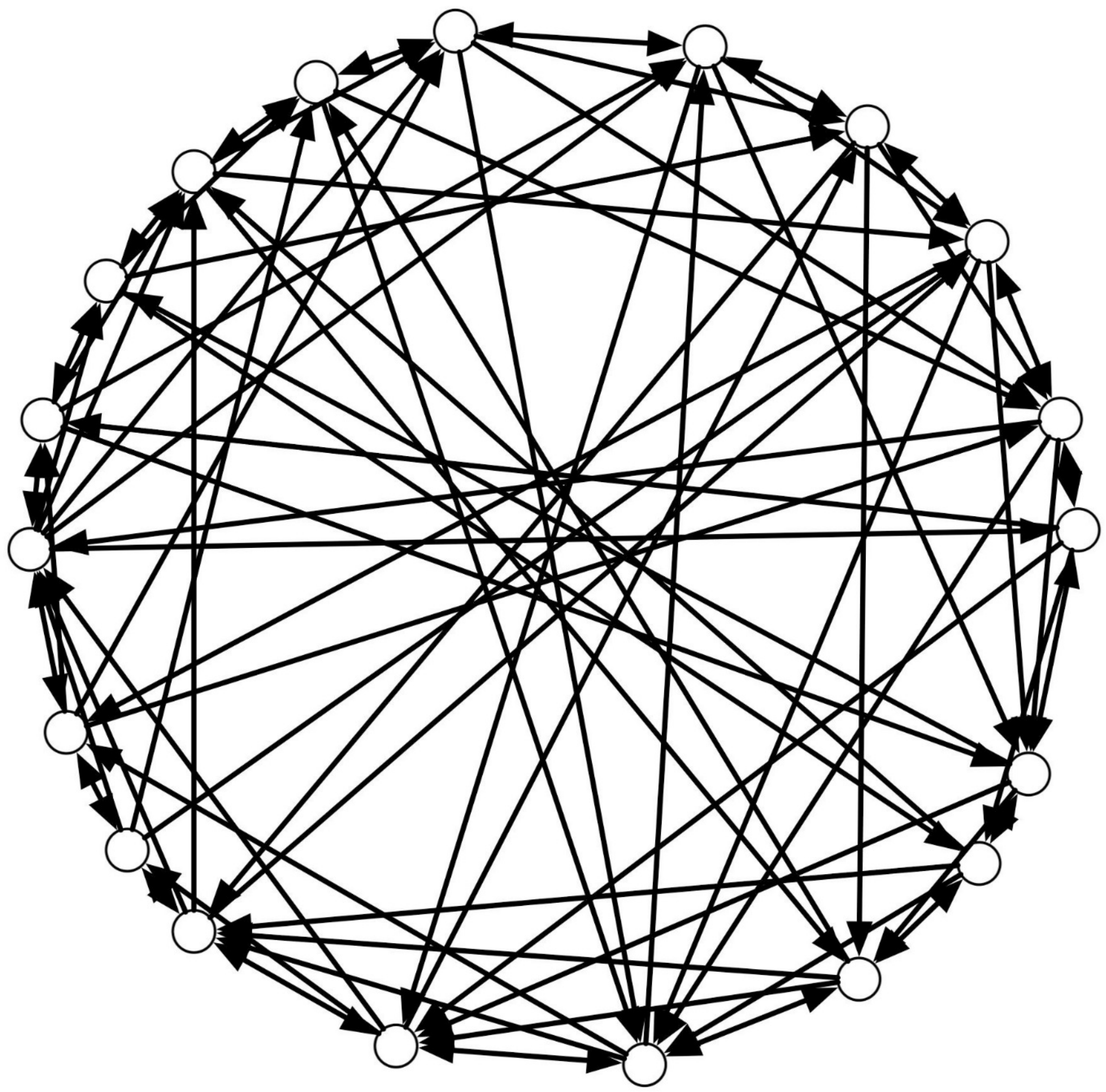
From Cuckoo Hashing
Rasmus Pagh, Flemming Friche Rodler
2004

Efficiency of Cuckoo Hashing

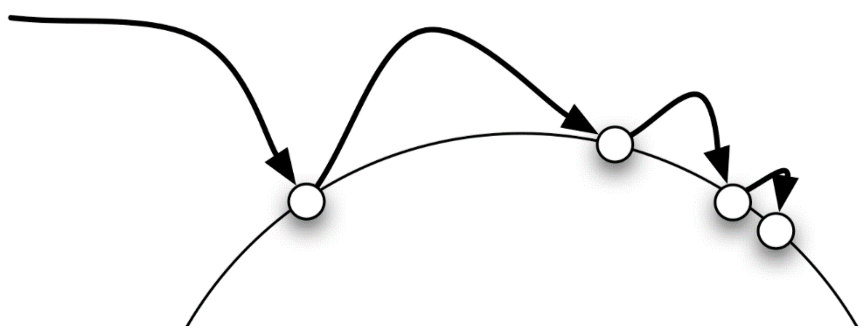
- Theorem
  - Let $\epsilon>0$ then if at most n elements are stored, then Cuckoo Hashing needs a hash space of $2n+\epsilon$.
- Three hash functions increase the load factor from 1/2 to 91%
- Insert
  - needs O(1) steps in the expectation
  - O(log n) with high probability
- Lookup
  - needs two steps

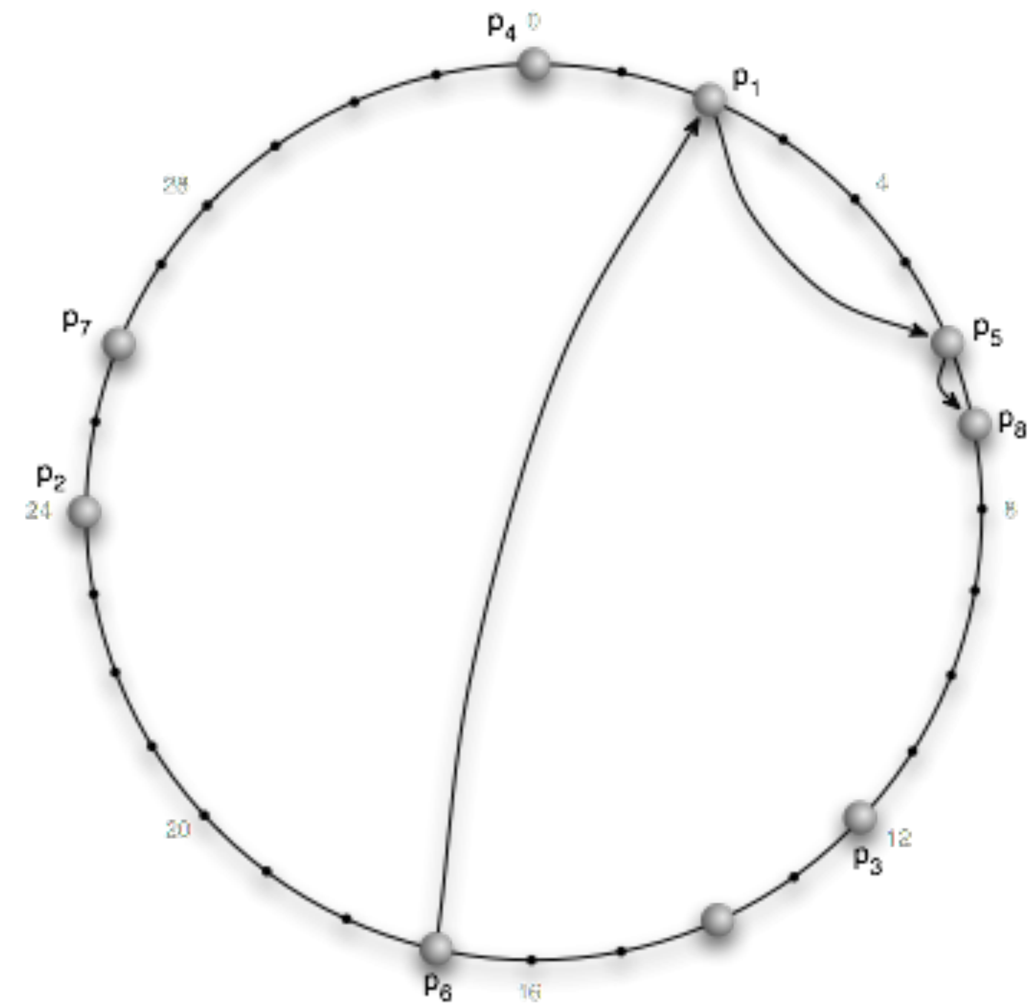
Chord

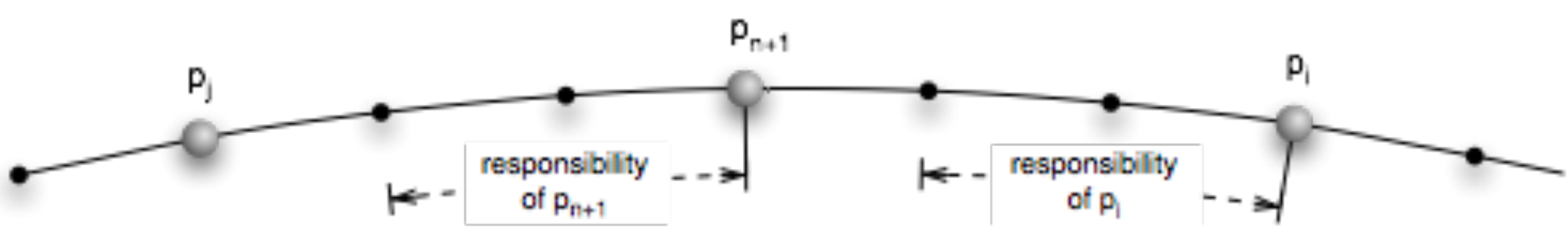
- Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek and Hari Balakrishnan (2001)
- Distributed Hash Table
  - range {0,..,2^m-1}
  - for sufficient large m
- for this work the range is seen as [0,1)
- Network
  - ring-wise connections
  - shortcuts with exponential increasing distance



Lookup in Chord

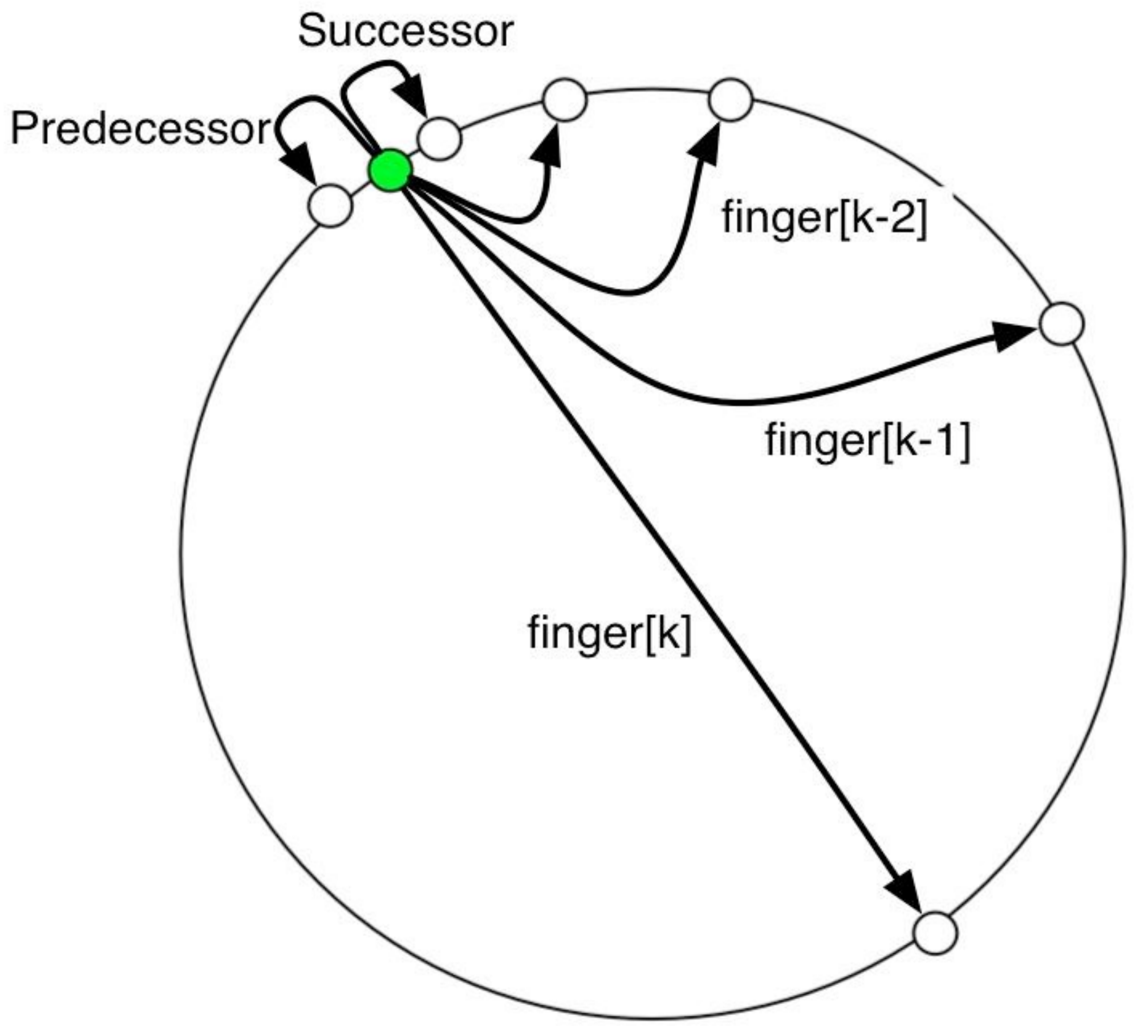






Data Structure of Chord

- For each peer
  - successor link on the ring
  - predecessor link on the ring
  - for all $i \in \{0,..,m-1\}$
    - Finger[i] := the peer following the value $r_V(b+2^i)$s
- For small i the finger entries are the same
  - store only different entries
- Chord
  - needs O(log n) hops for lookup
  - needs $O(\log^2 n)$ messages for inserting and erasing of peers



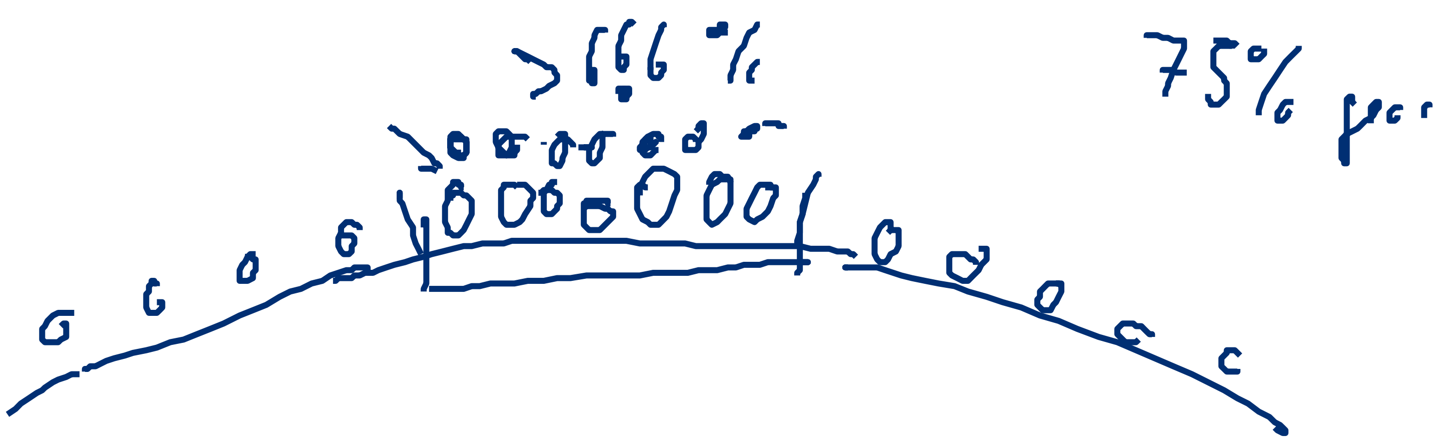
Cuckoo Hashing for Security

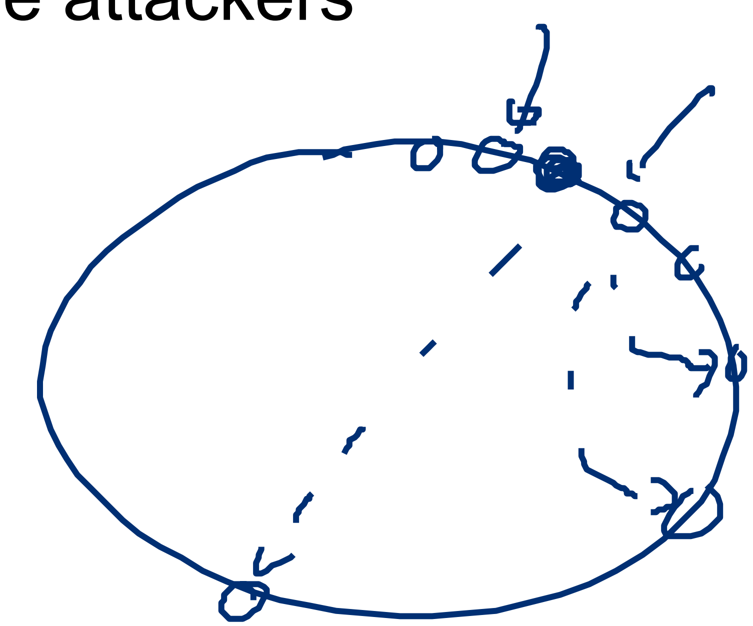
- Given n honest peers and ϵ n dishonest peers
- Goal
  - For any adversarial attack the following properties for every interval $I \subseteq [0, 1)$ of size at least $(c \log n)/n$ we have
  - Balancing condition
    - I contains $\Theta(|I| \cdot n)$ nodes
  - Majority condition
    - the honest nodes in I are in the majority
- Then all majority decisions of $O(\log n)$ nodes give a correct result

CoNe Freiburg

Rejoin Attacks

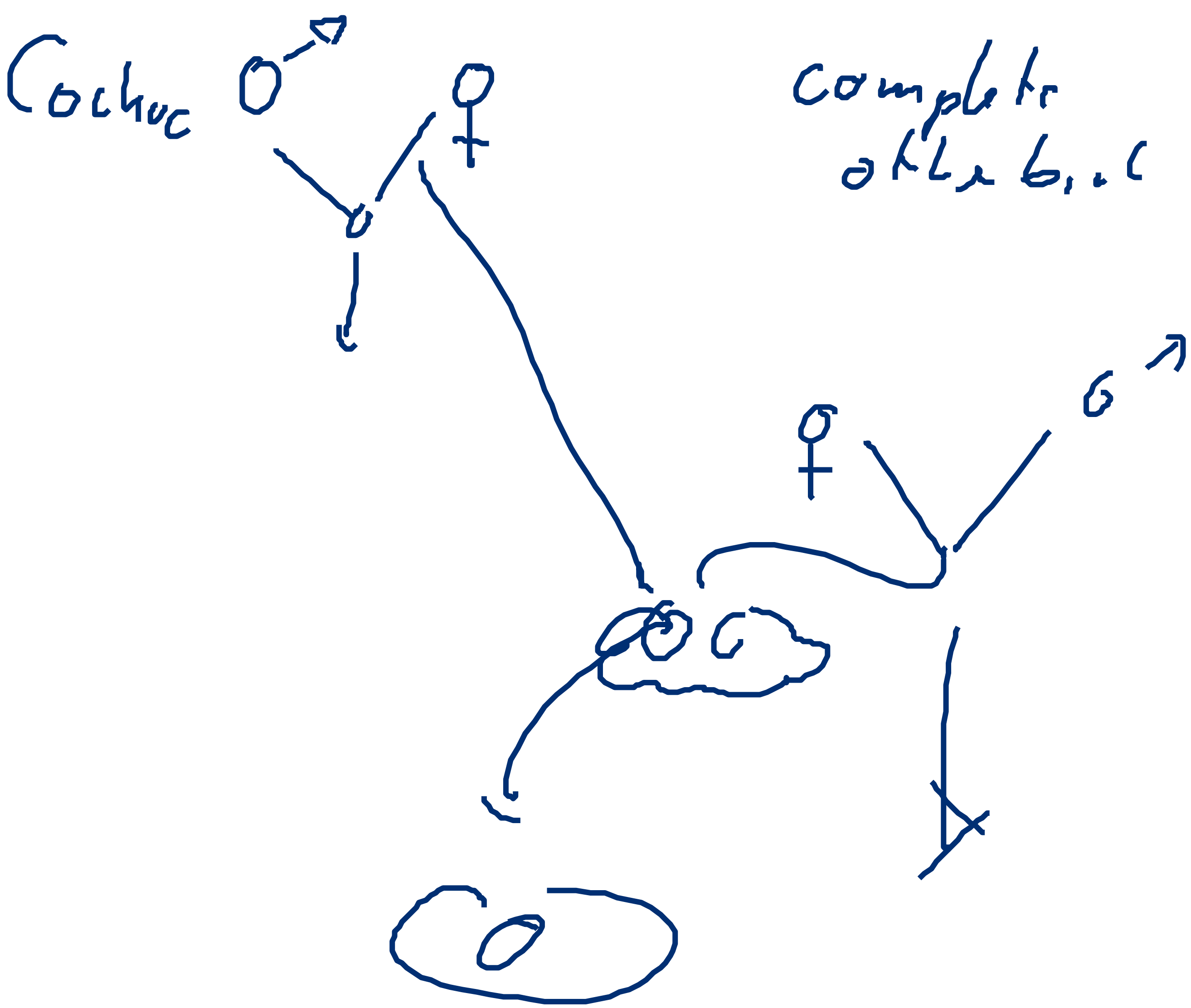
- Secure hash functions for positions in the Chord
  - if one position is used
  - then in an O(log n) neighborhood more than half is honest
  - if more than half of al peers are honest
- Rejoin attacks
  - use a small number of attackers
  - check out new addresses until attackers fall in one interval
  - then this neighborhood can be ruled by the attackers



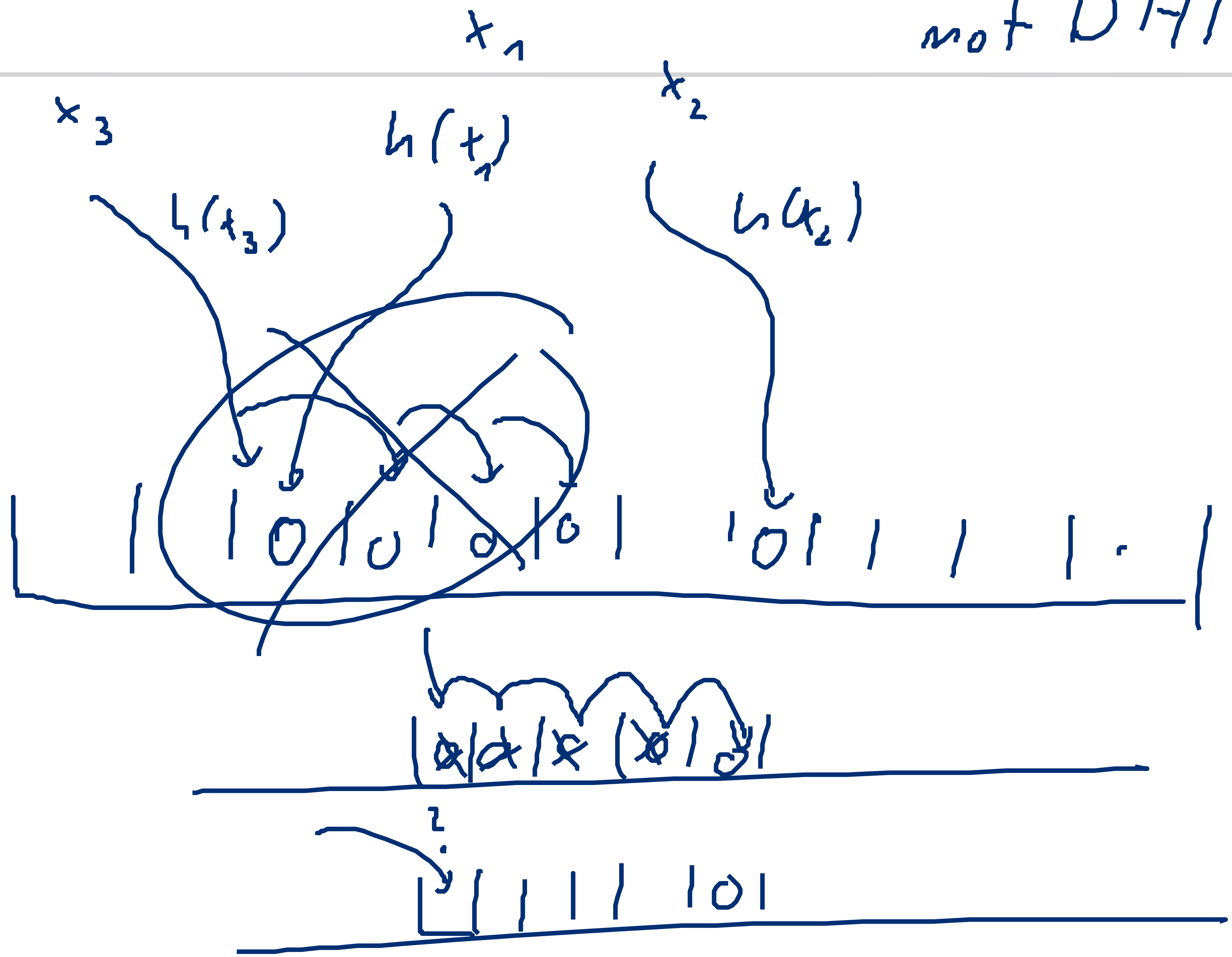


UNI FREIBURG

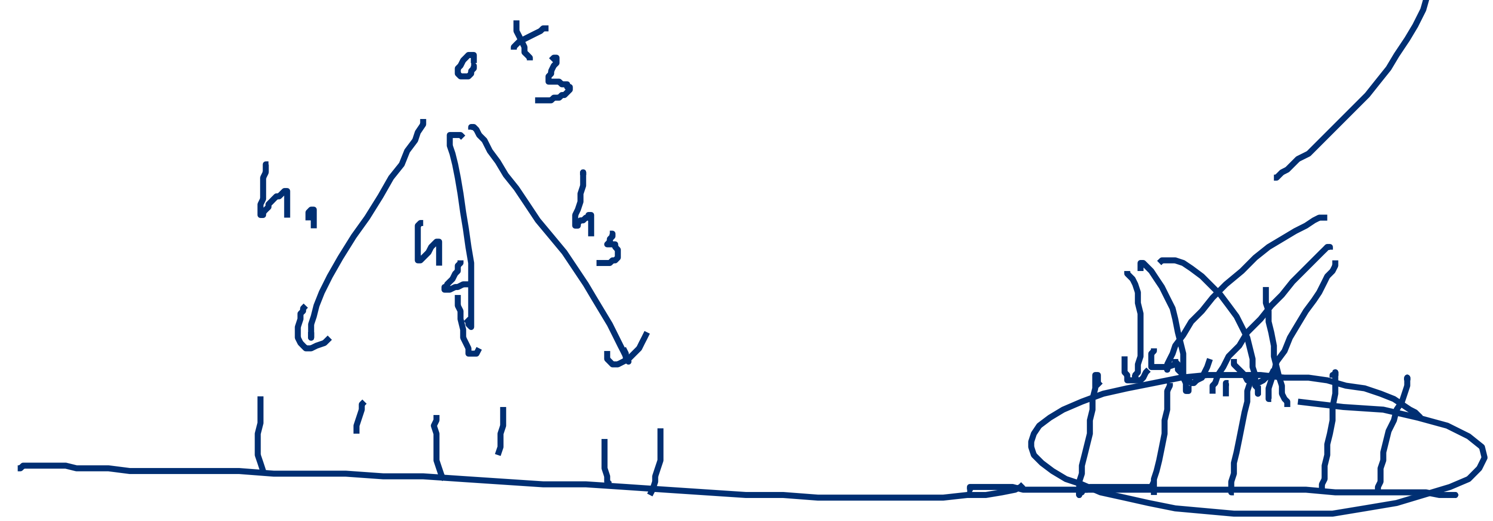
Cuckoo Kuckuck



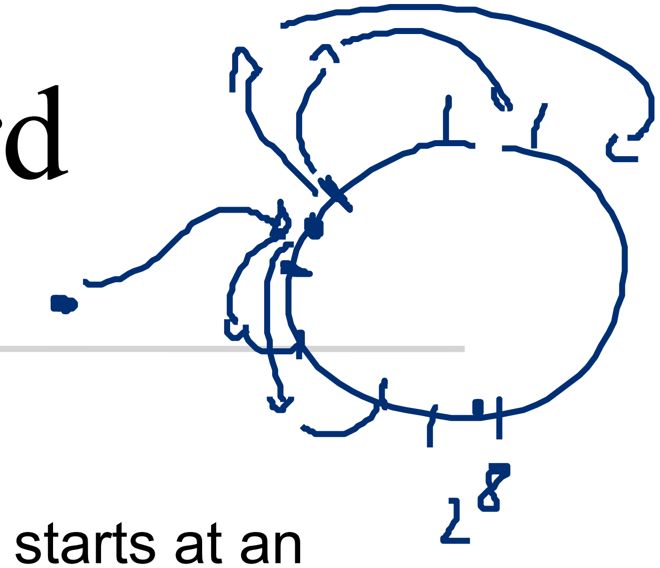
mot DHT



Cuckoo Hashing



The Cuckoo Rule for Chord

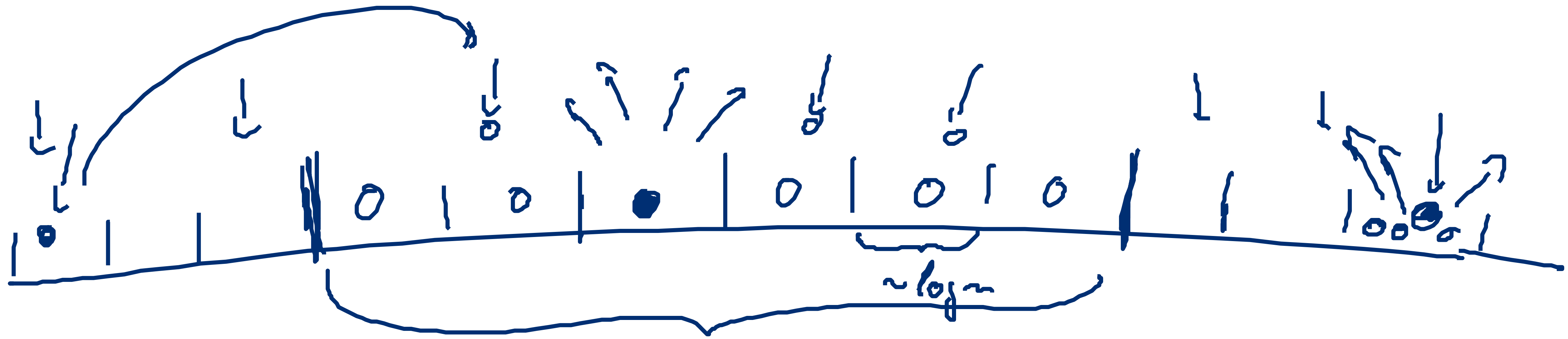


- Notation
  - a region is an interval of size $1/2^r$ in [0, 1) for some integer r that starts at an integer multiple of $1/2^r$
  - There are exactly 2^r regions
  - A k-region is a region of size (closest from above to) k/n, and for any point $x \in [0, 1)$
  - the k-region $R_k(x)$ is the unique k-region containing x.
- Cuckoo rule
  - If a new node v wants to join the system, pick a random $x \in [0, 1)$.
  - Place v into x and move all nodes in $R_k(x)$ to points in [0, 1) chosen uniformly at random
    - (without replacing any further nodes).
- Theorem
  - For any constants ϵ and k with $\epsilon < 1-1/k$, the cuckoo rule with parameter k satisfies the balancing and majority conditions for a polynomial number of rounds, with high probability, for any adversarial strategy within our model.
  - The inequality $\epsilon < 1 - 1/k$ is sharp

Operations

- Data storage
  - each data item is stored in the $O(\log^3 n)$ neighborhood as copies
- Primitives
  - robust hash functions
    - safe against attacks
  - majority decisions of each operation
  - use multiple routes for targeting location

$$\frac{1}{16}\cdot\frac{15}{16}\cdot\frac{1}{16} \qquad \sum_{i=1}^{m}\left(\frac{15}{16}\right)^{i-1}\cdot 16^{i}\cdot i \rightarrow 16$$

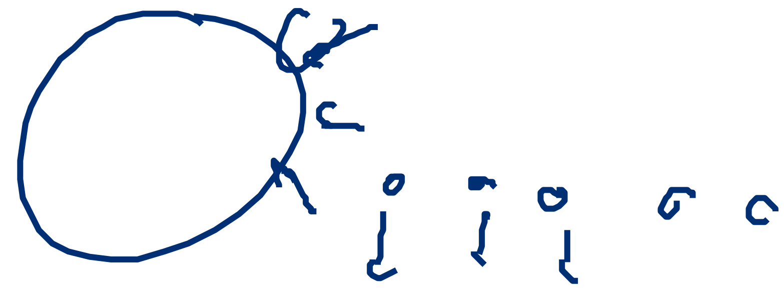


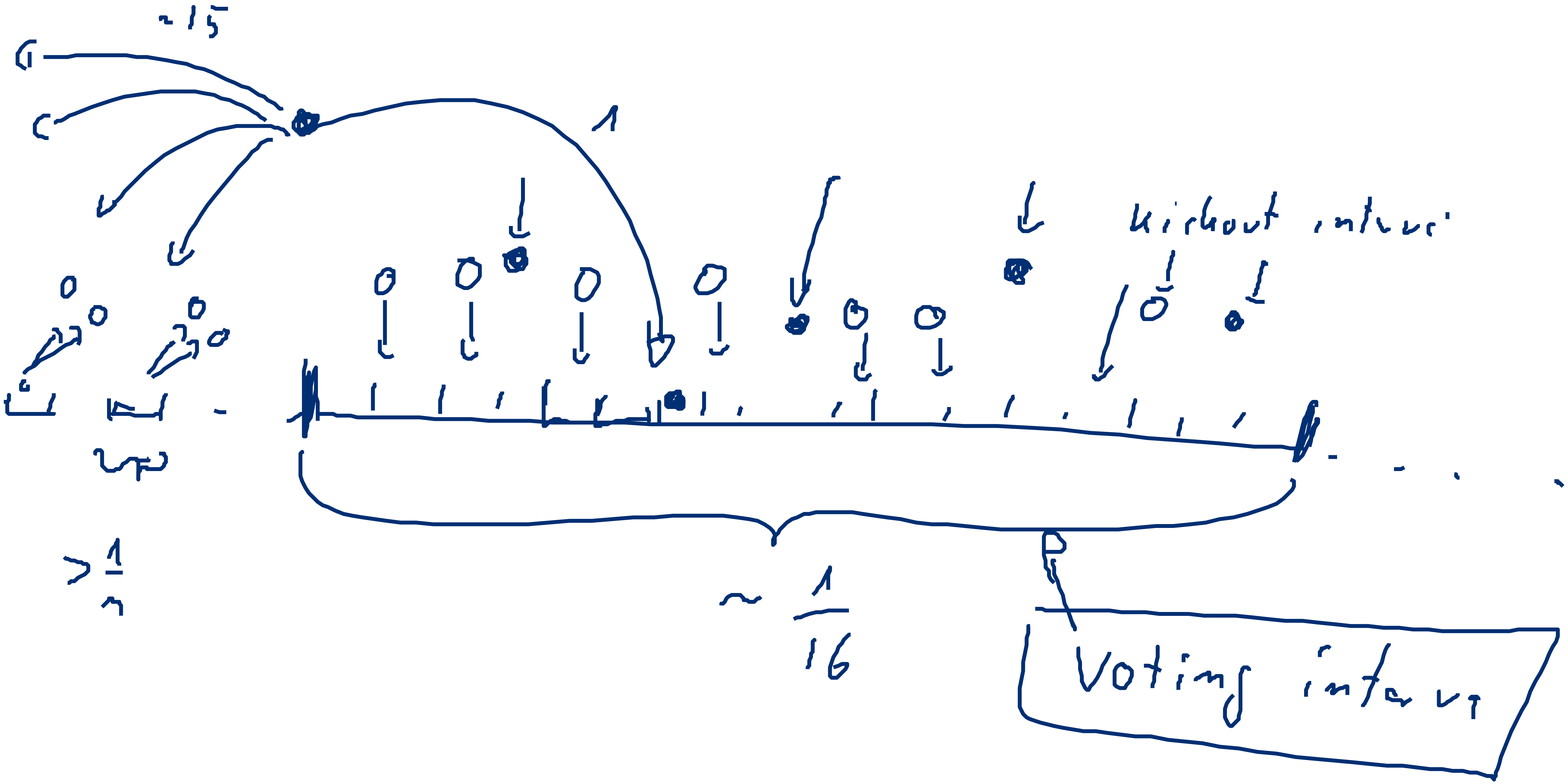
$m \sim \log^2 n$

$$\frac{2^k}{n} \approx \frac{1}{16} \rightarrow \text{Chernoff}$$

$$\#\text{attacks}\cdot\frac{n}{2^k}\cdot\text{misses}.$$

CoNe Freiburg





Efficiency

- Lookup
  - works correctly with high probability
  - can be performed with $O(\log^5 n)$ messages
- Inserting of data
  - works in "polylogarithmic time"
  - needs $O(\log^5 n)$ messages
- Copies stored of each data: $O(\log^3 n)$

$\log_2^5 1000 \approx 10^5 = 100{,}000$

Discussion

- Advantage
  - Cuckoo Chord is safe against adversarial attacks
  - Cuckoo rule is simple and effective
- Disadvantage
  - Computation of secure hash function is complex
  - Considerate overhead for communication
- Theoretical breakthrough
- Little impact to the practical world

Peer-to-Peer Networks

14 Security

Christian Schindelhauer
Technical Faculty
Computer-Networks and Telematics
University of Freiburg