University of Würzburg
Dept. of Distributed Systems
Prof. Dr. P. Tran-Gia

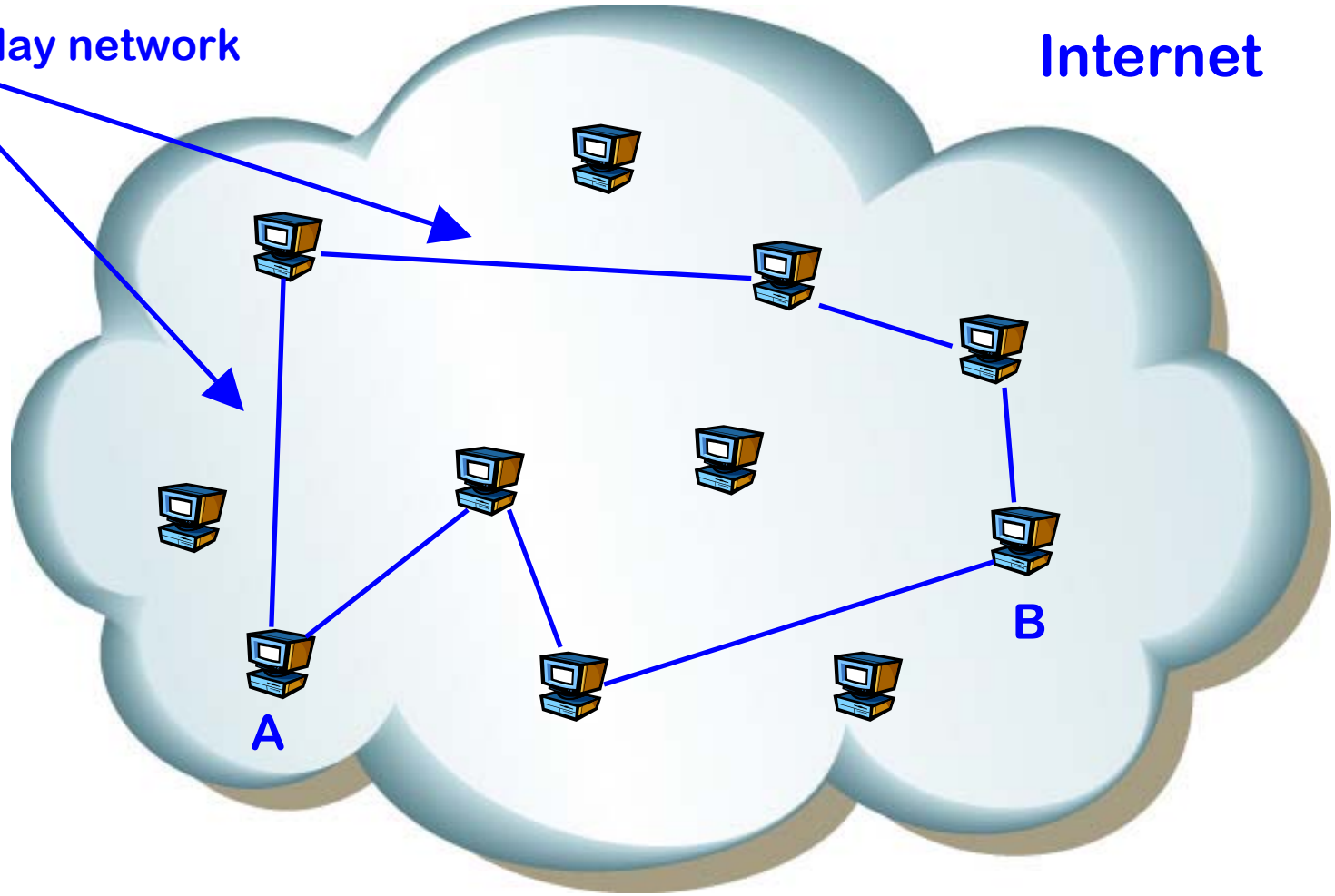
Delay Analysis of DHT-based Peer-to-Peer Networks

Andreas Binzenhöfer, University of Würzburg

Phuoc Tran-Gia, University of Würzburg

[binzenhoefer|trangia]@informatik.uni-wuerzburg.de

How to calculate the duration of a search?



real time applications:

- distributed phone book

- instant messaging

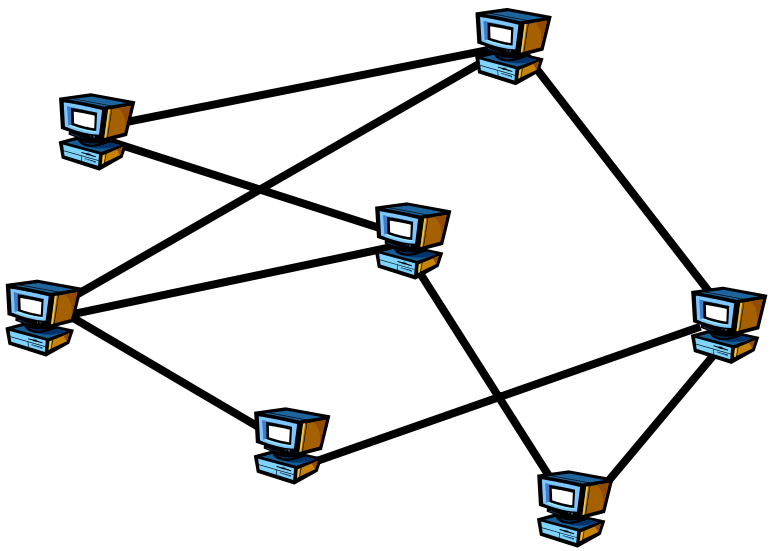
- chatting

Outline

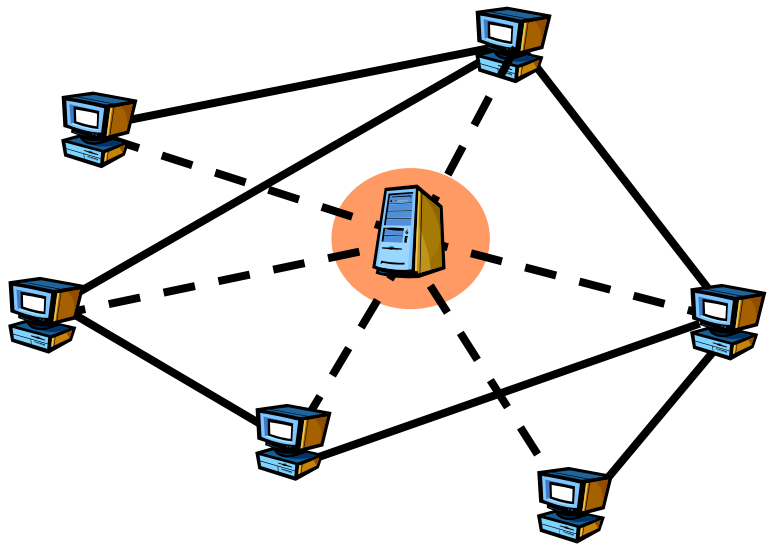
- Introduction to P2P networks
- Chord basics
  - How is a Chord ring organized?
  - How does a search in a Chord ring work?
- Analysis of the duration of a search in a Chord-based P2P-system
  - Required assumptions
  - Model of the search
- Results
  - Effects of the network variation on the search duration
  - Study of the scalability
- Conclusion

Introduction

- P2P-systems are mainly used to store and retrieve data



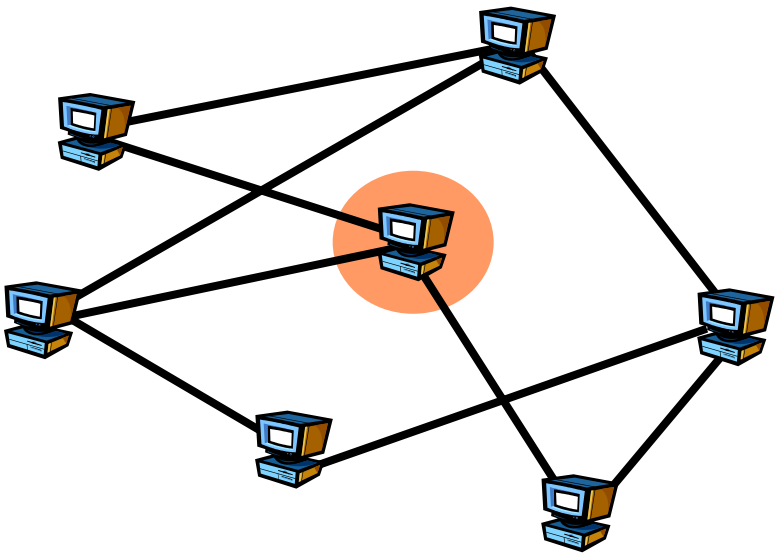
broadcasting mechanisms
(Gnutella)



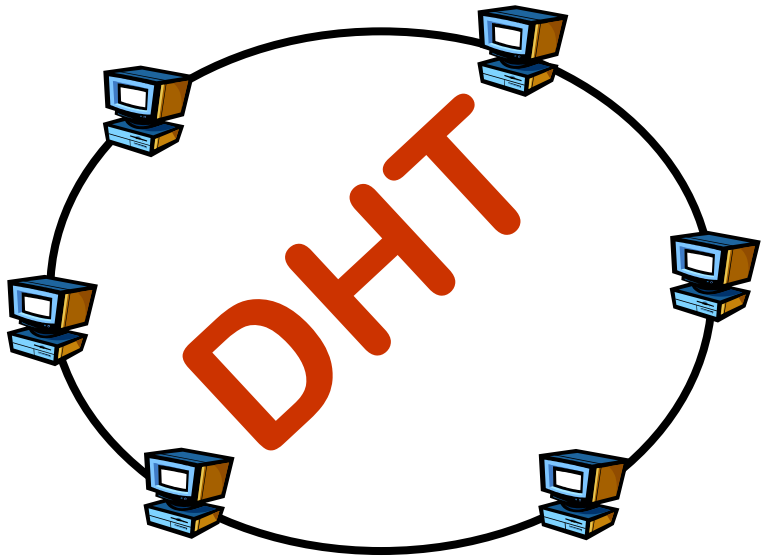
index server
(edonkey, emule)

Introduction

- P2P-Systems are mainly used to store and retrieve data



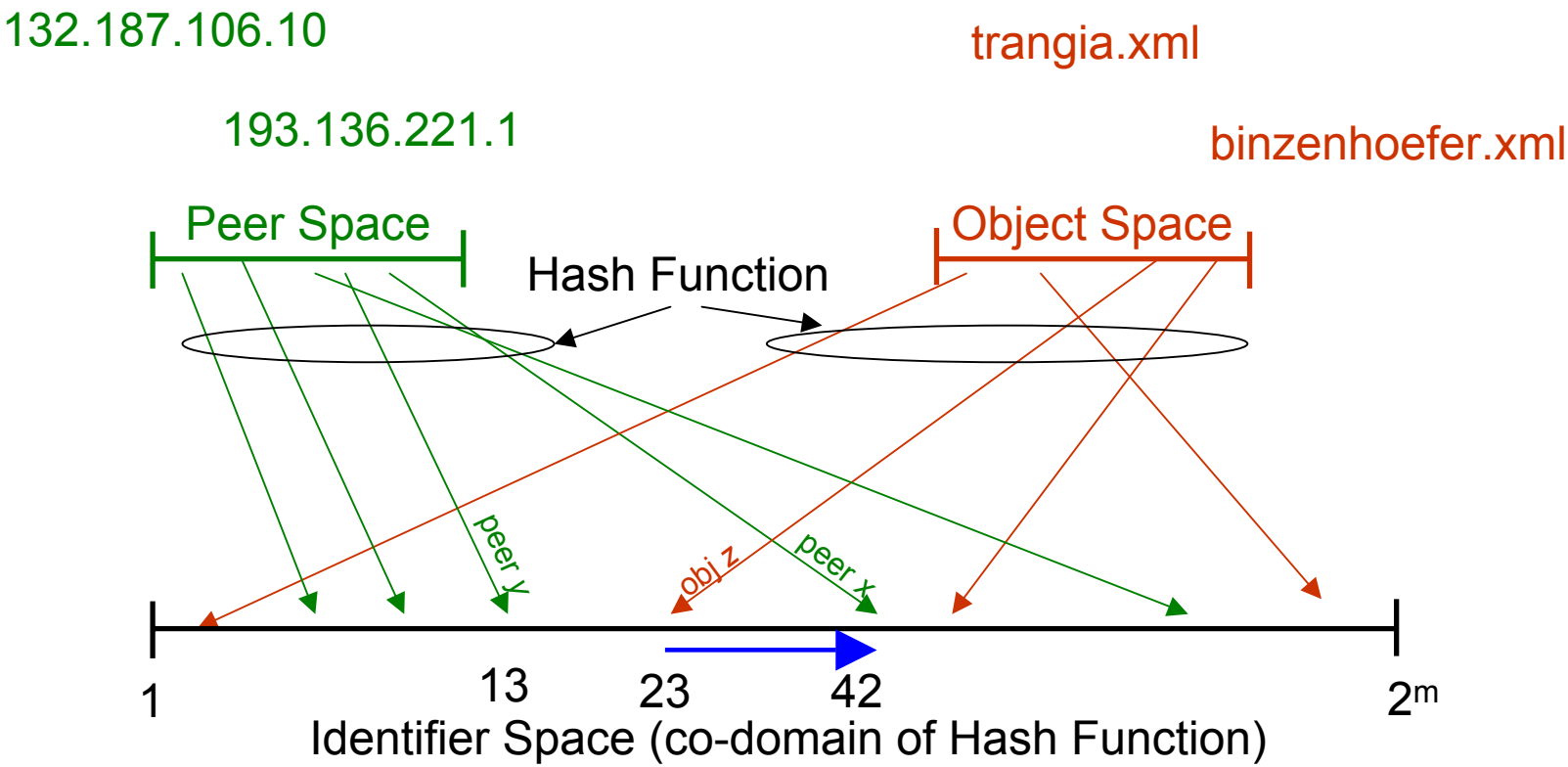
super peers
(Kazaa)



structured overlay
(Chord, Kademlia)

- Little is known about the influence of the variation of the network delay on the time needed to complete a search

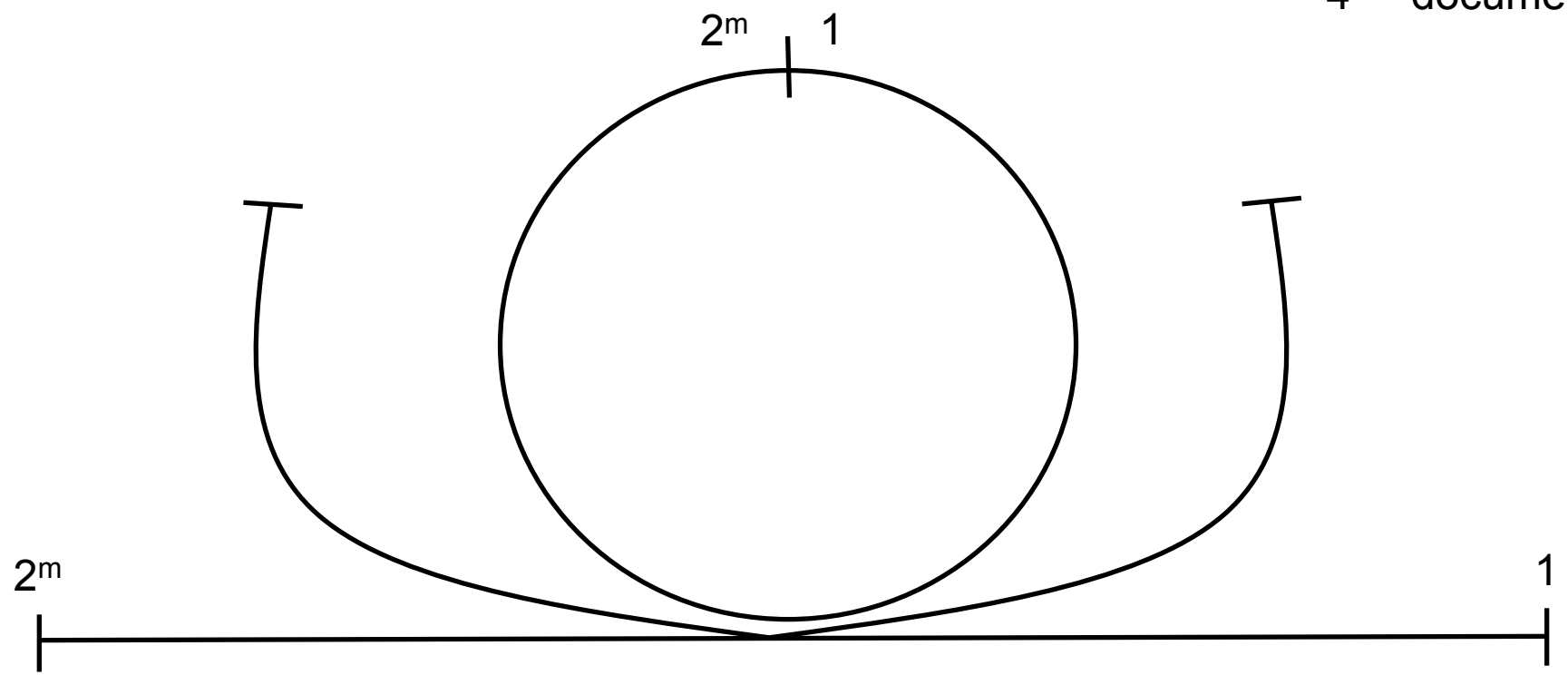
How does a DHT work?



Each object is stored at the first peer succeeding the objects hash value

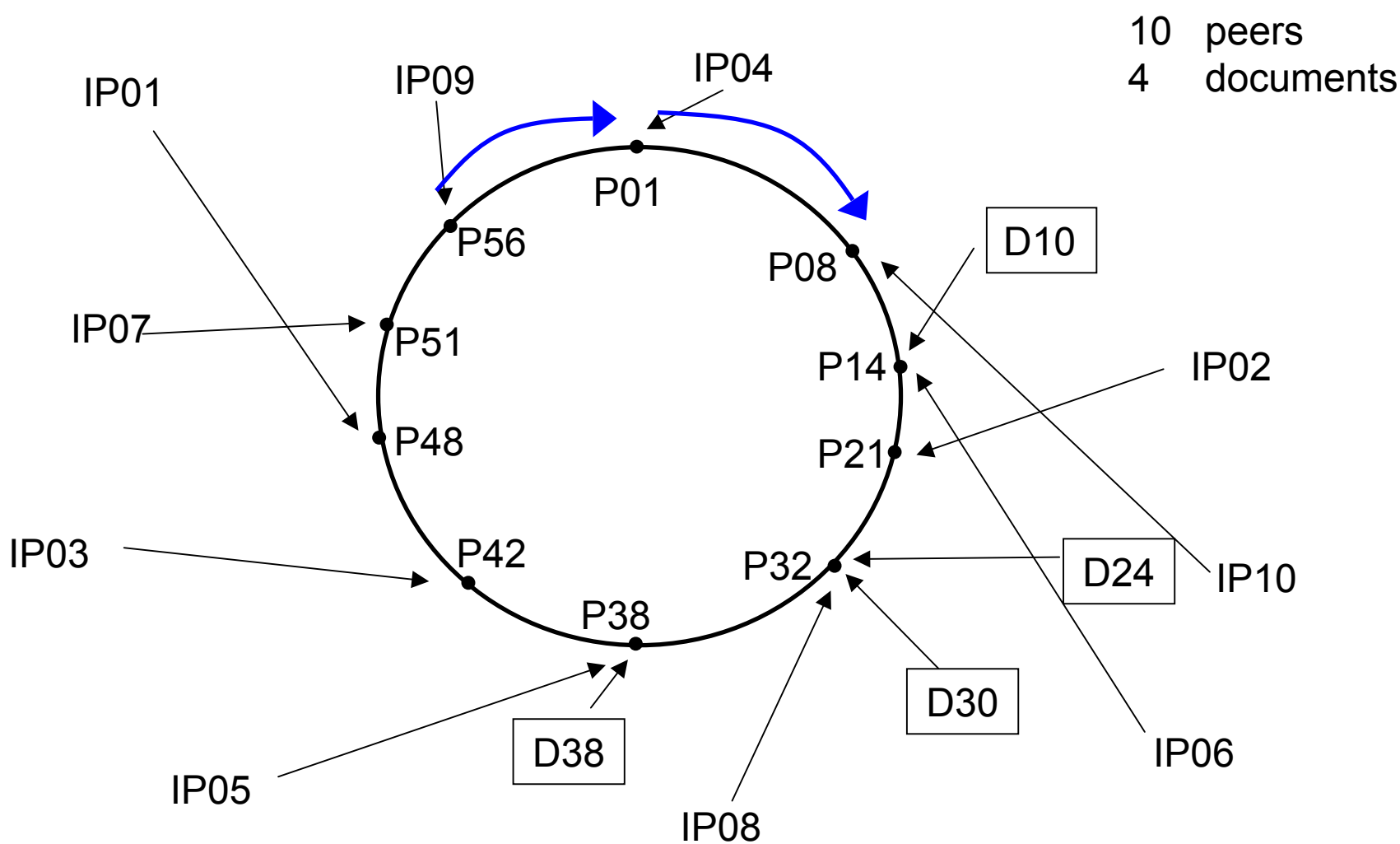
Chord Example

10 peers
4 documents



- The identifier space is transformed into a circle to cope with its edges

Chord Example



Assumptions

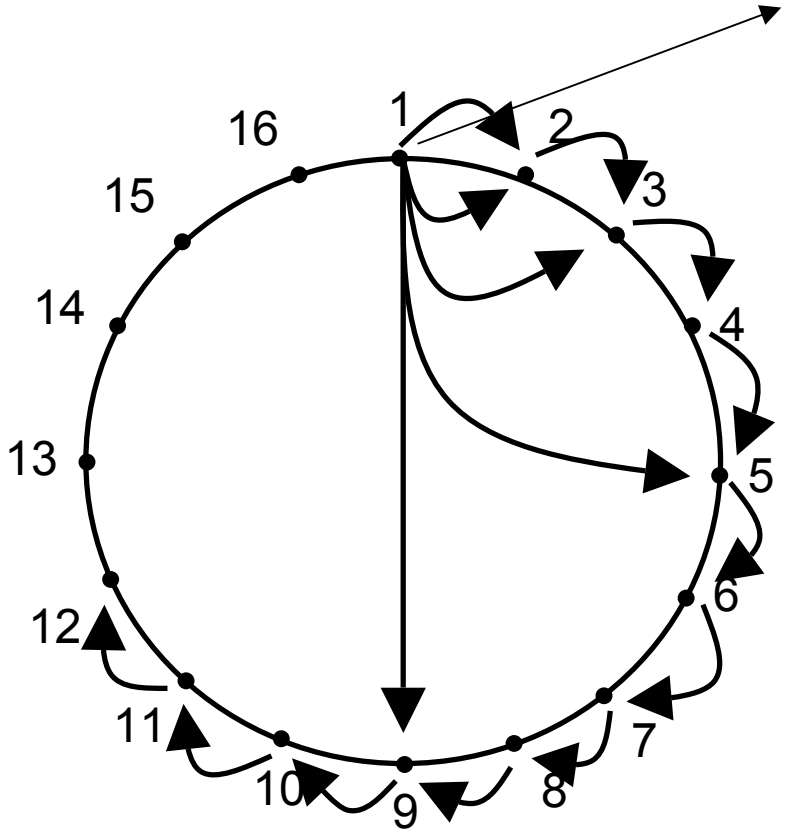
- Stable Chord ring (no changes in the overlay structure)
- All peers and documents uniformly allocated in the identifier space
- Each document looked up with the same probability

⇨ Each peer will be responsible for the same number of documents and will therefore answer the same number of queries.

Chord: Fingertable

A peer with ID id_p has its *i*-th finger at:

$id_p + 2^{(i-1)}$

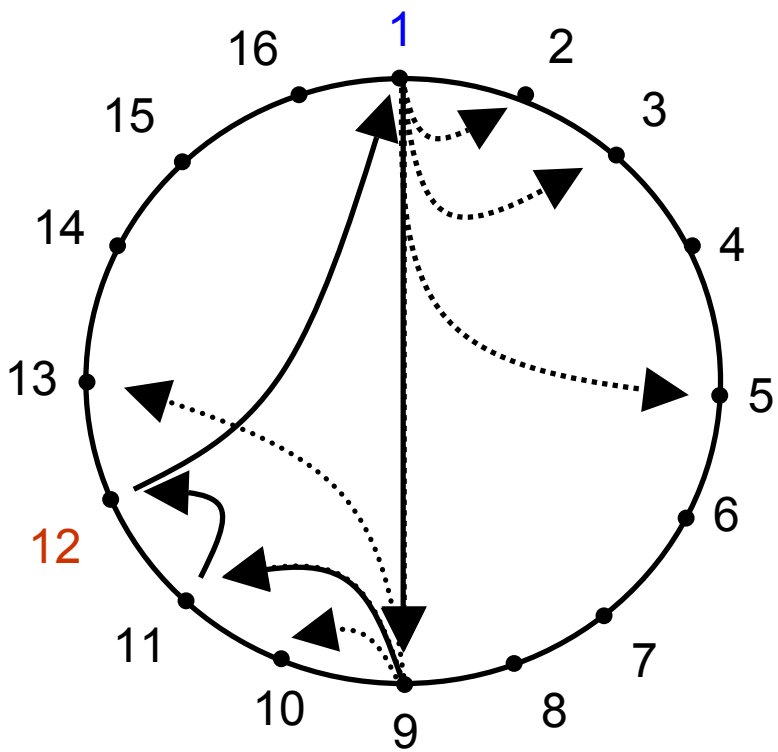


The peer with id_p=1 has its fingers at:

2 = 1 + 1
3 = 1 + 2
5 = 1 + 4 and
9 = 1 + 8

Chord: Search

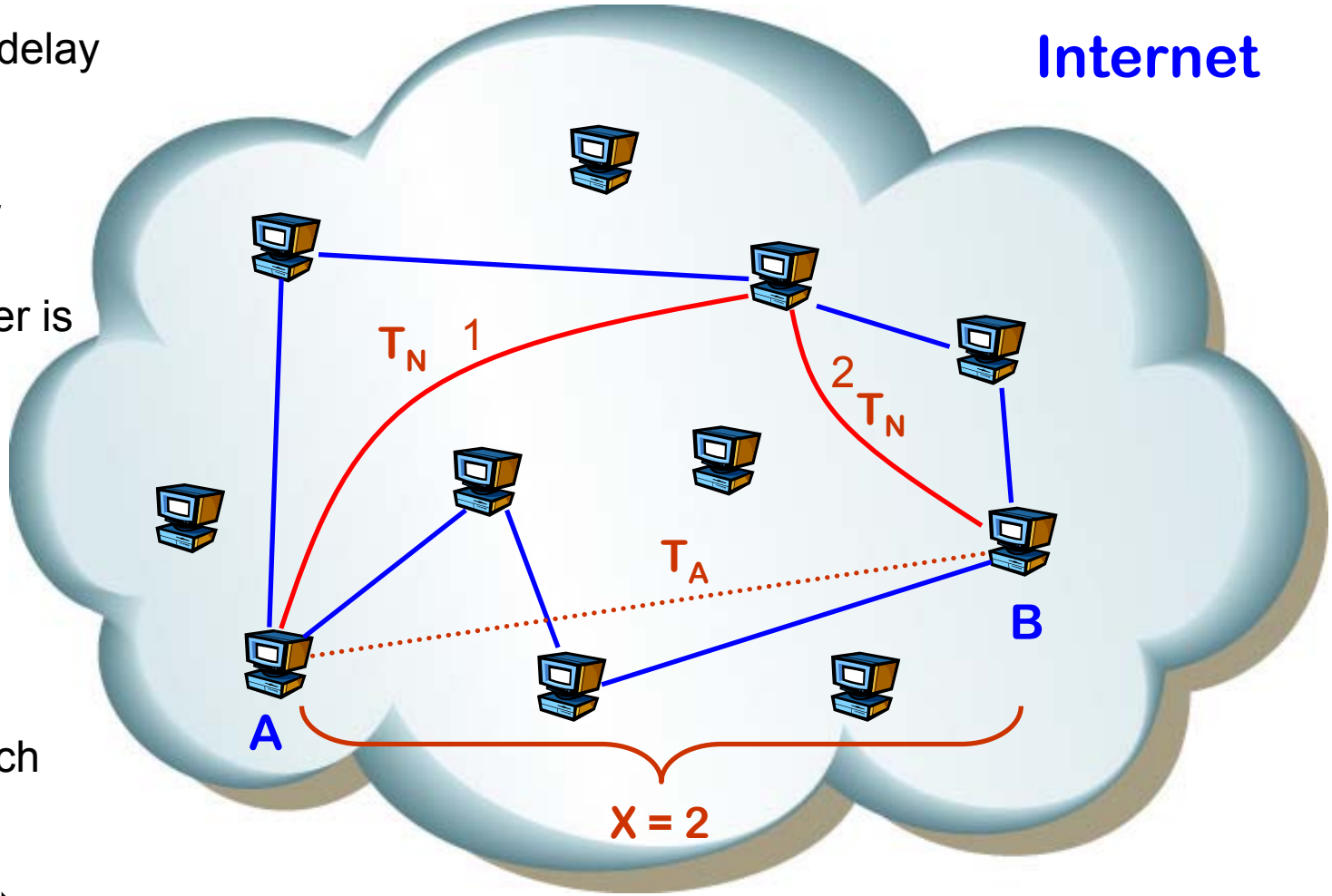
peer 1 is looking up peer 12



A search is always forwarded to the closest peer

Definitions

- T_N: one hop delay
- X: number of hops until searched peer is found
- T_A: delay of the answer
- T: total search duration



⇨ needed later to calculate quantiles

Distance Distribution: special case

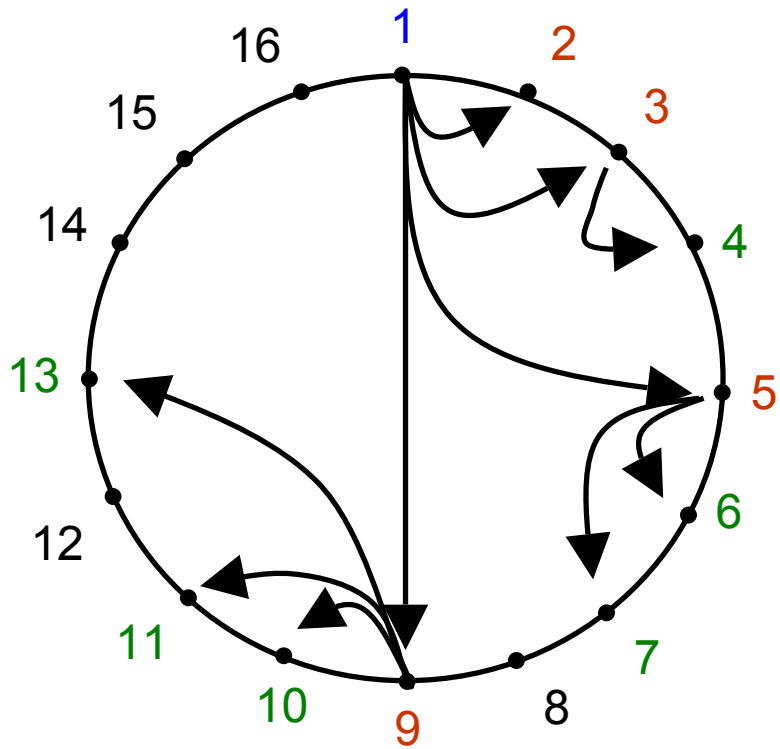
Assume Chord Size: $n = 2^k$

Number of peers that are *i* hops away:

0 hops: 1 peer (only peer 1 itself)

1 hop : 4 peers (all Fingers)

2 hops: 6 peers (4, 6, 7, 10, 11, 13)

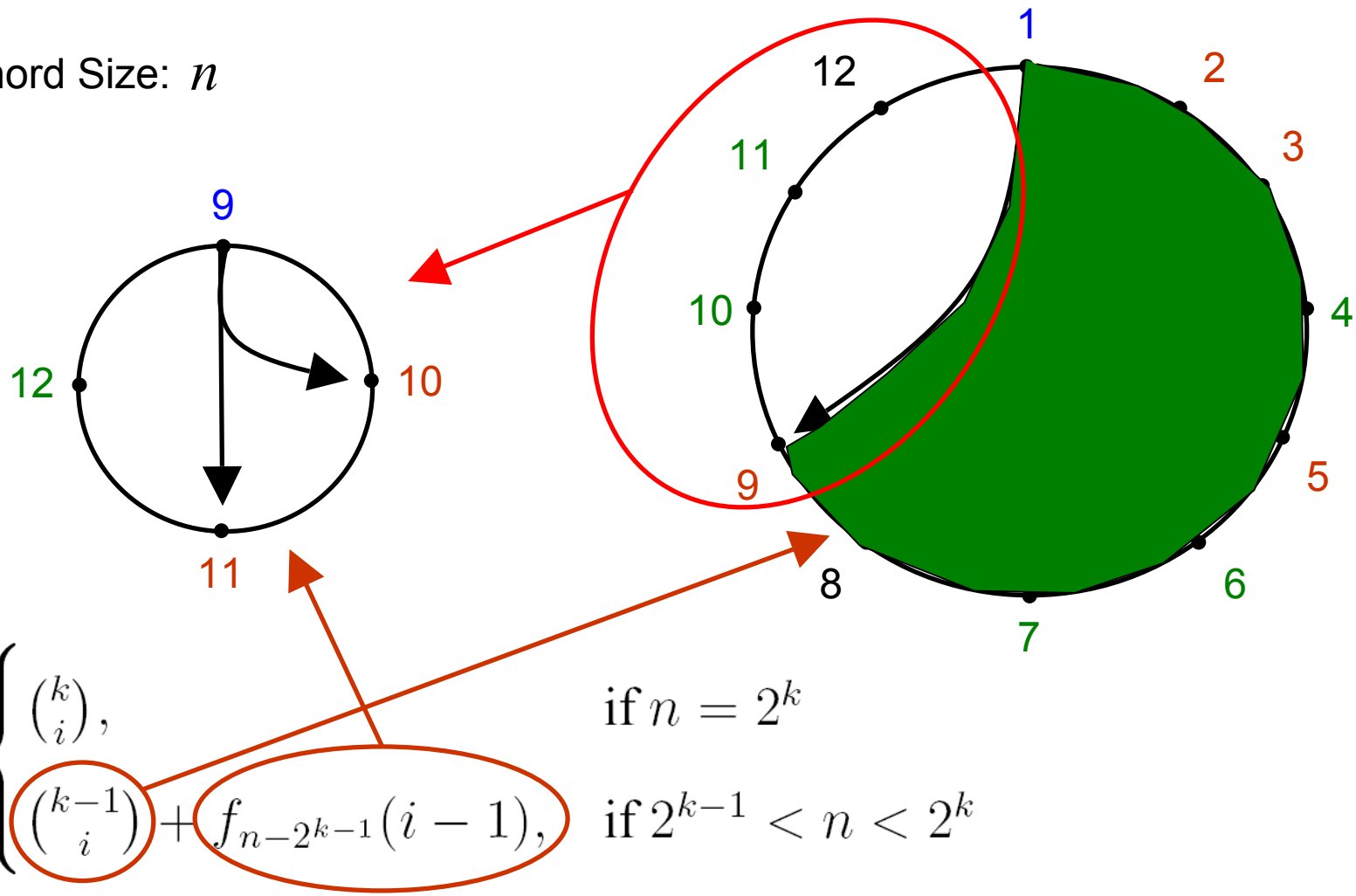


Probability, that the searched peer is *i* hops away

$$p_i = P(X = i) = \frac{\binom{k}{i}}{2^k}$$

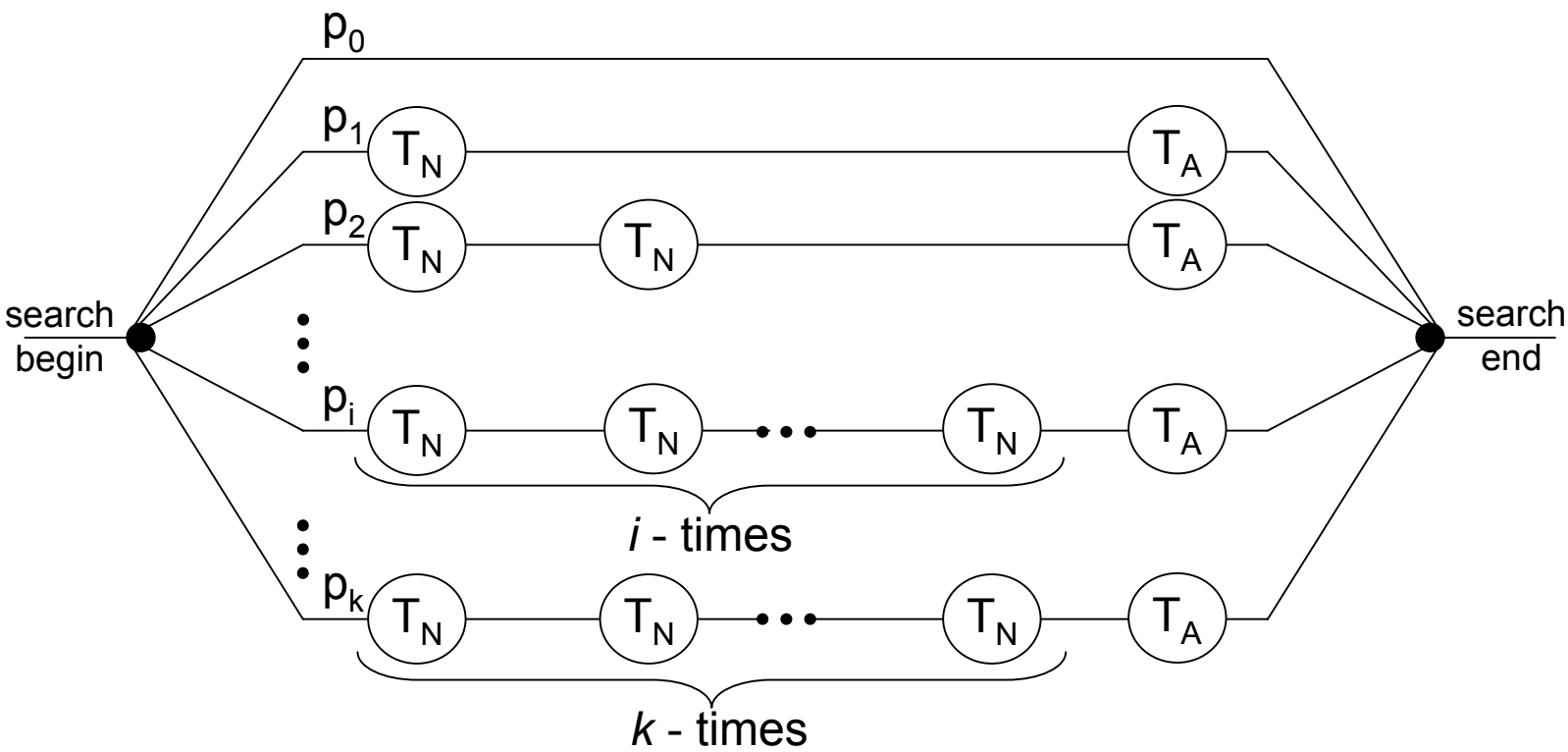
Distance Distribution: arbitrary case

Assume Chord Size: n



$$f_n(i) = \begin{cases} \binom{k}{i}, & \text{if } n = 2^k \\ \binom{k-1}{i} + f_{n-2^{k-1}}(i-1), & \text{if } 2^{k-1} < n < 2^k \end{cases}$$

Phase Diagram of a Chord Search



Results: Mean search delay

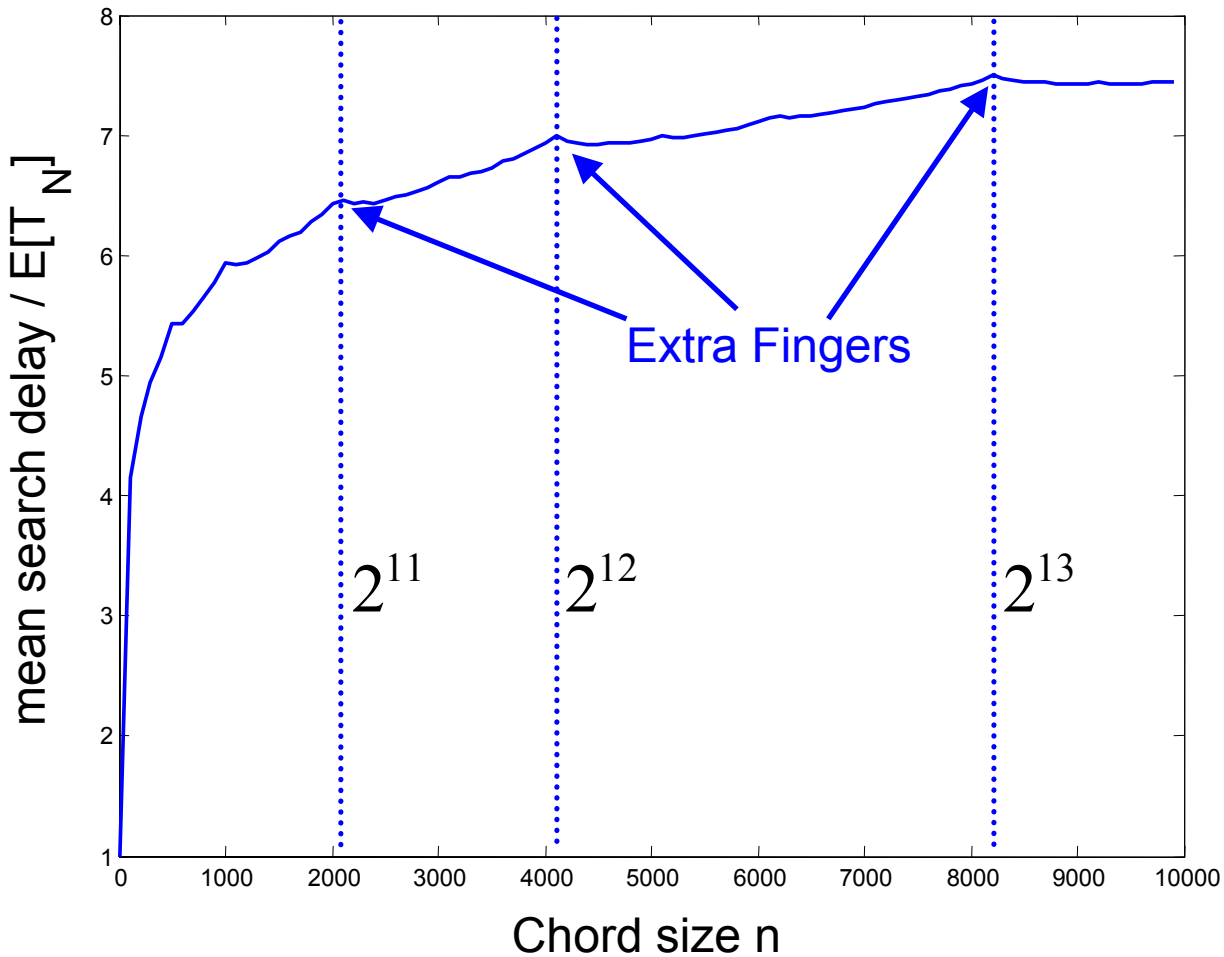
$T_N = T_A$

T_N: negative-binomially distributed

Two moments of the network delay:

$c_{T_N} = 1$

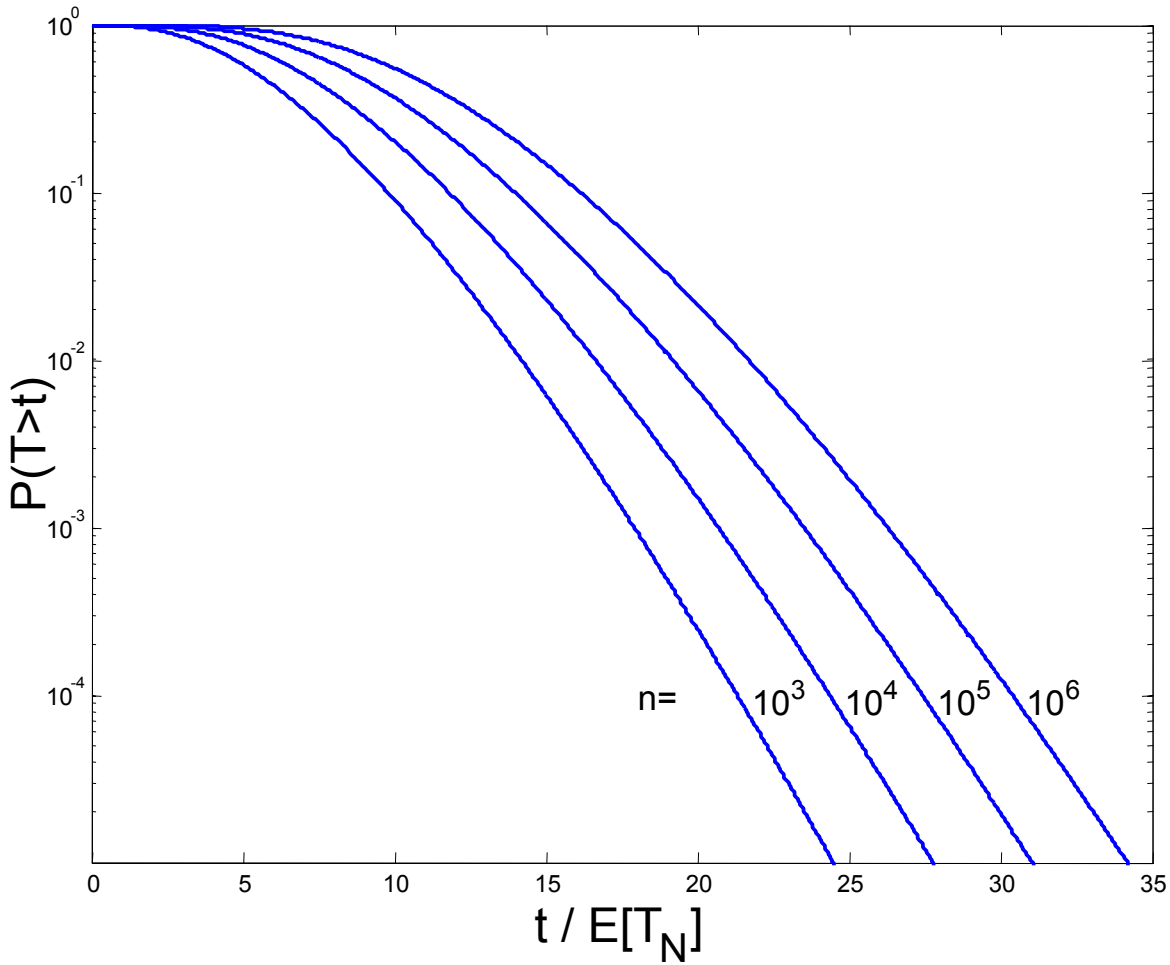
$E[T_N] = 50$ ms



Results: Proof of scalability

$E[T_N]$ = 50 ms

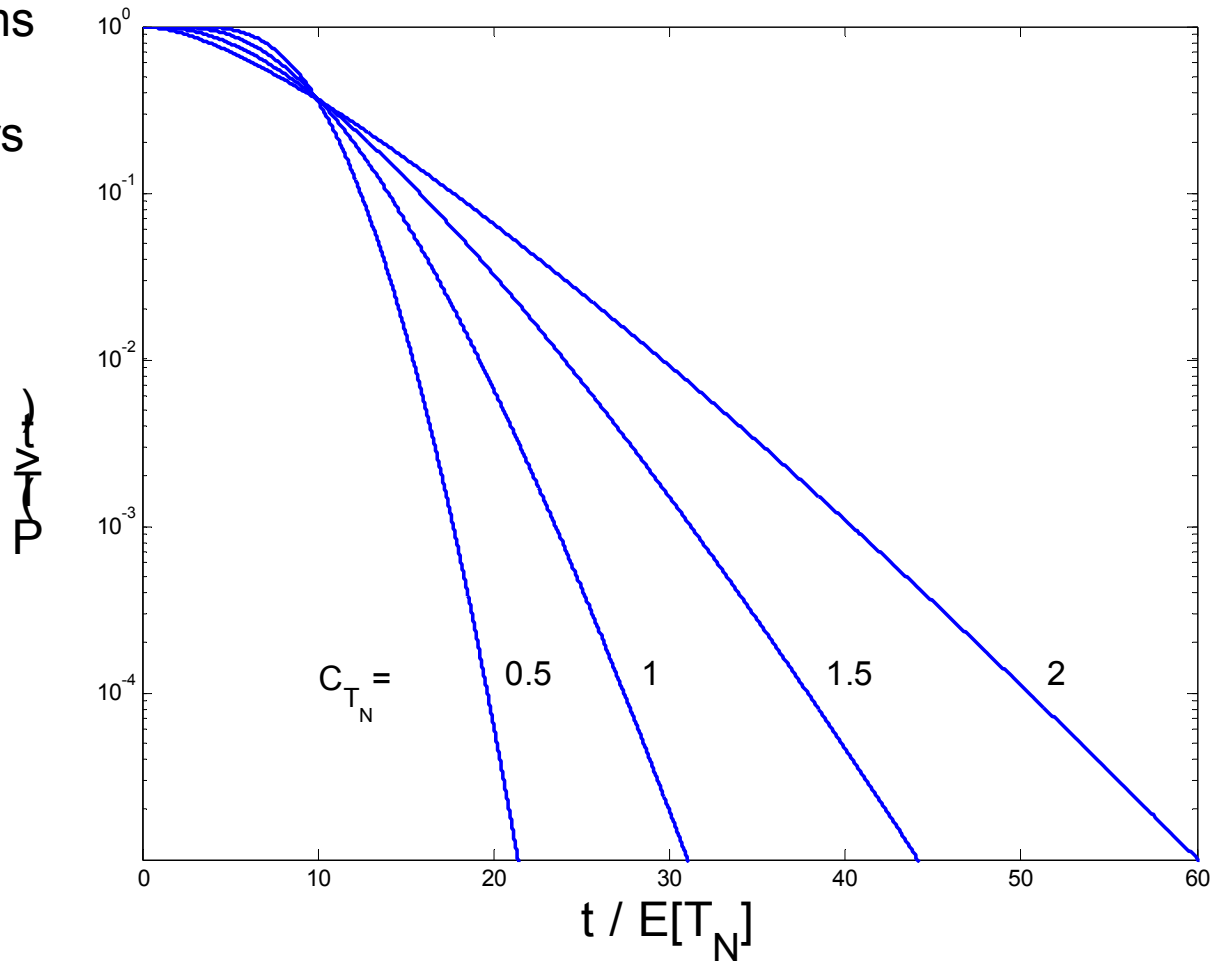
c_{T_N} = 1



Results: Distribution Function

$E[T_N]$ = 50 ms

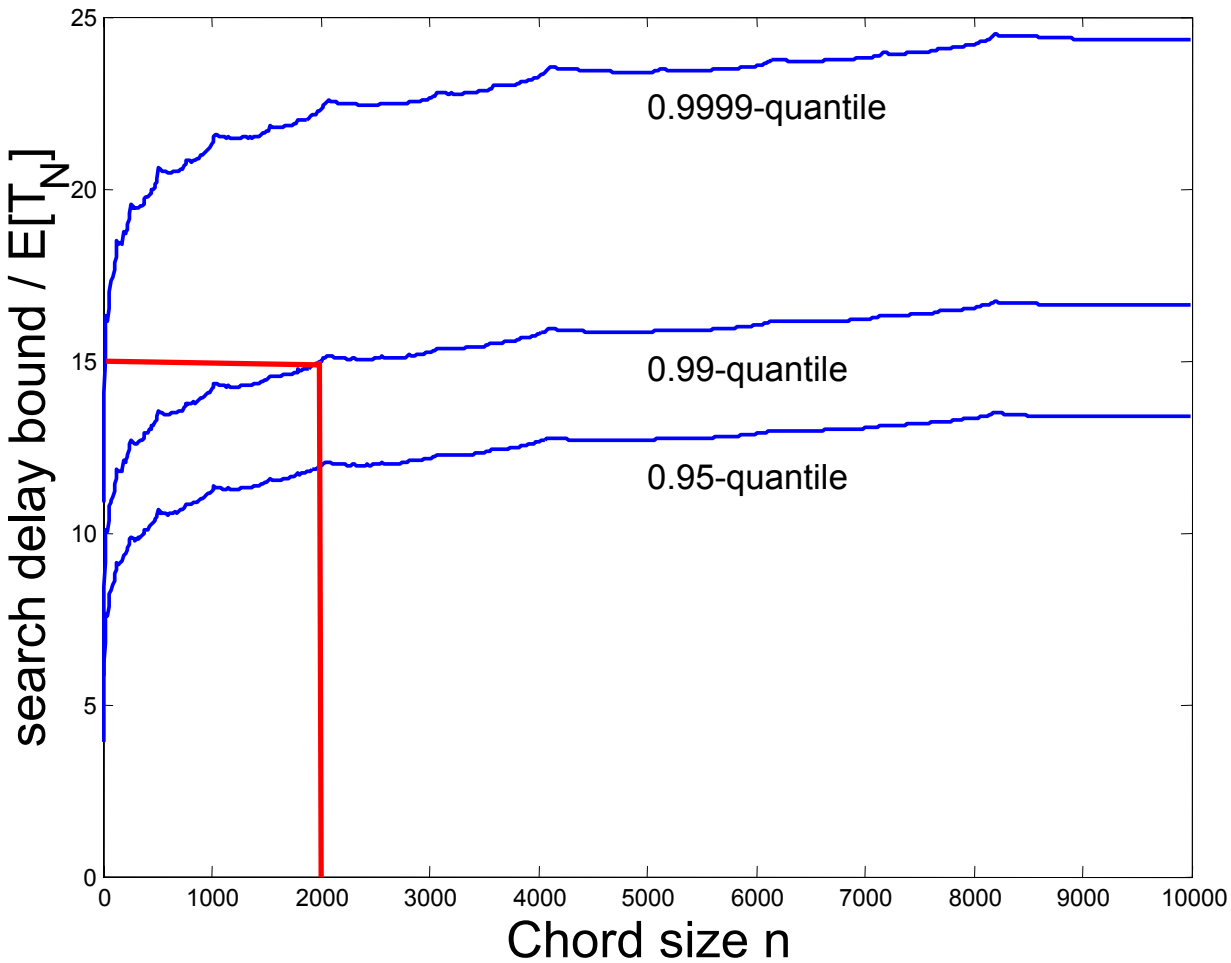
n = 10^5 peers



Results: Quantiles of the search delay

$E[T_N]$ = 50 ms

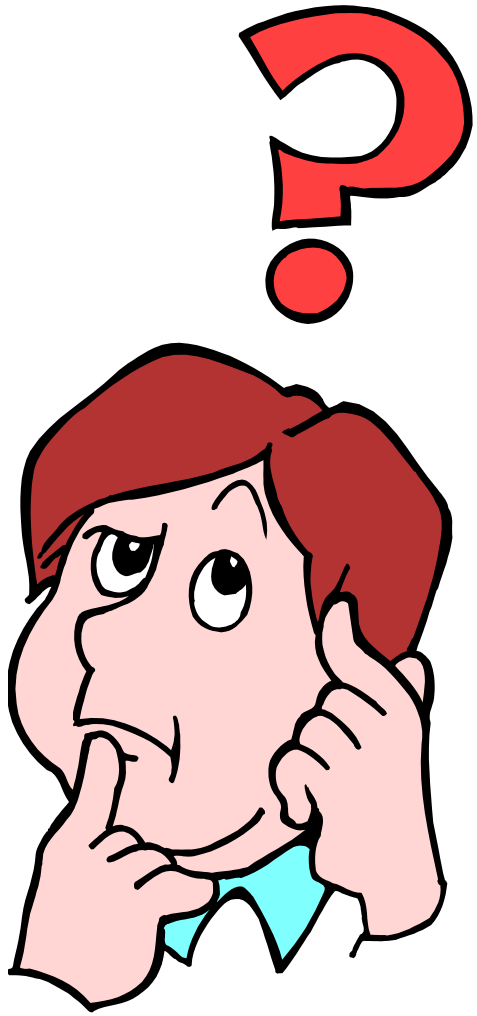
$c_{T_N} = 1$



Conclusion

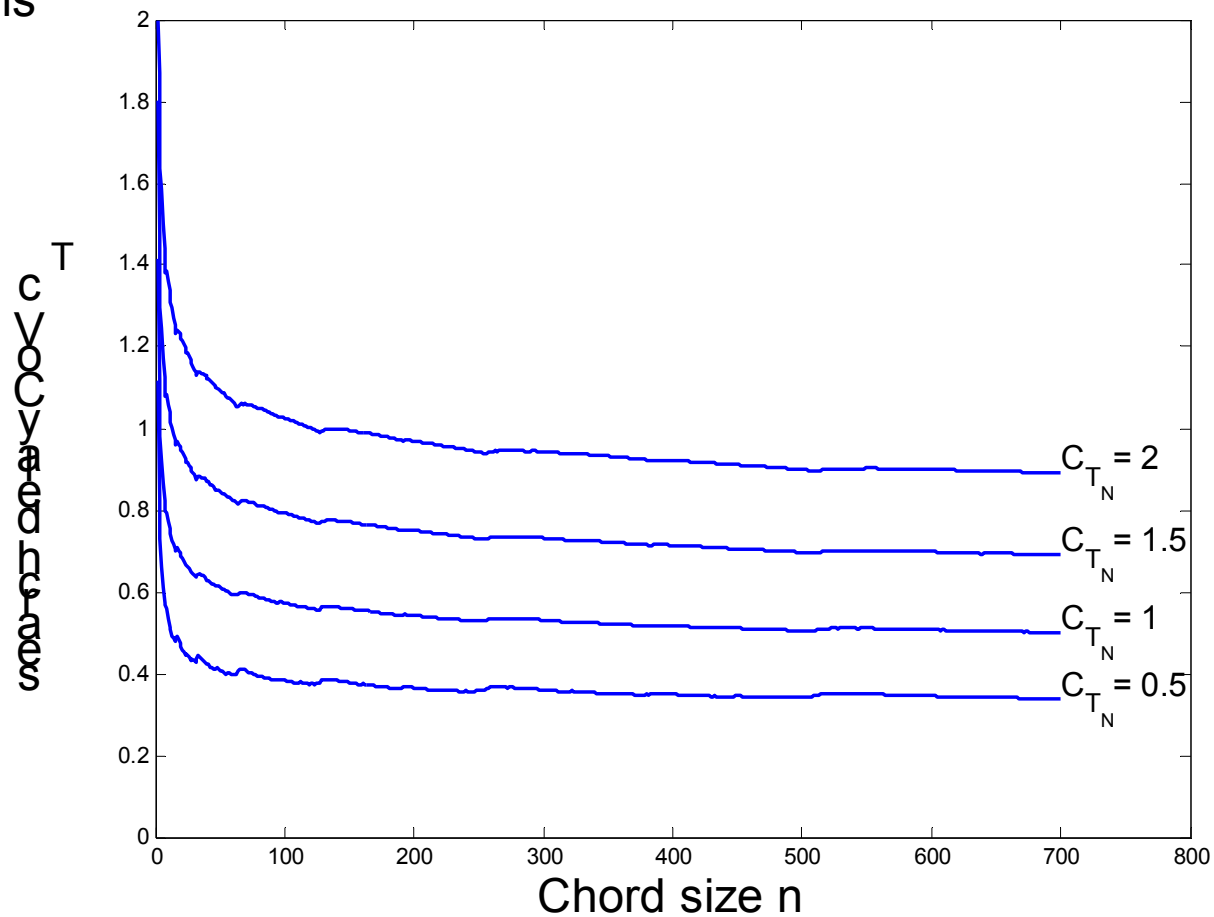
- Search in stable Chord rings does indeed scale
- Entire distribution of the search duration can be calculated
- Quantiles of the search delay can be used for system dimensioning purposes
- Results used in a business case project regarding a distributed directory
- Extension to Chord enhancements like proximity neighbor selection possible

Q&A



Results: CoV of the search delay

$E[T_N]$ = 50 ms



Results: CoV of the search delay

$E[T_N]$ = 50 ms