Decentralized Systems Engineering

CS-438 – Fall 2025

DEDIS

Pierluca Borsò-Tan and Bryan Ford

EPFL

Credits: P. Tennage, C. Basescu, et al.

So far...

- Decentralized communication & search
- Focusing on (mostly) unstructured networks

Characteristics:

- (Nearly) stateless
- Simple to engineer
- Expressive search
- Optimizations require (true) random sampling (hard)
- ☹ inefficient, $O(\sqrt{n})$ search at best

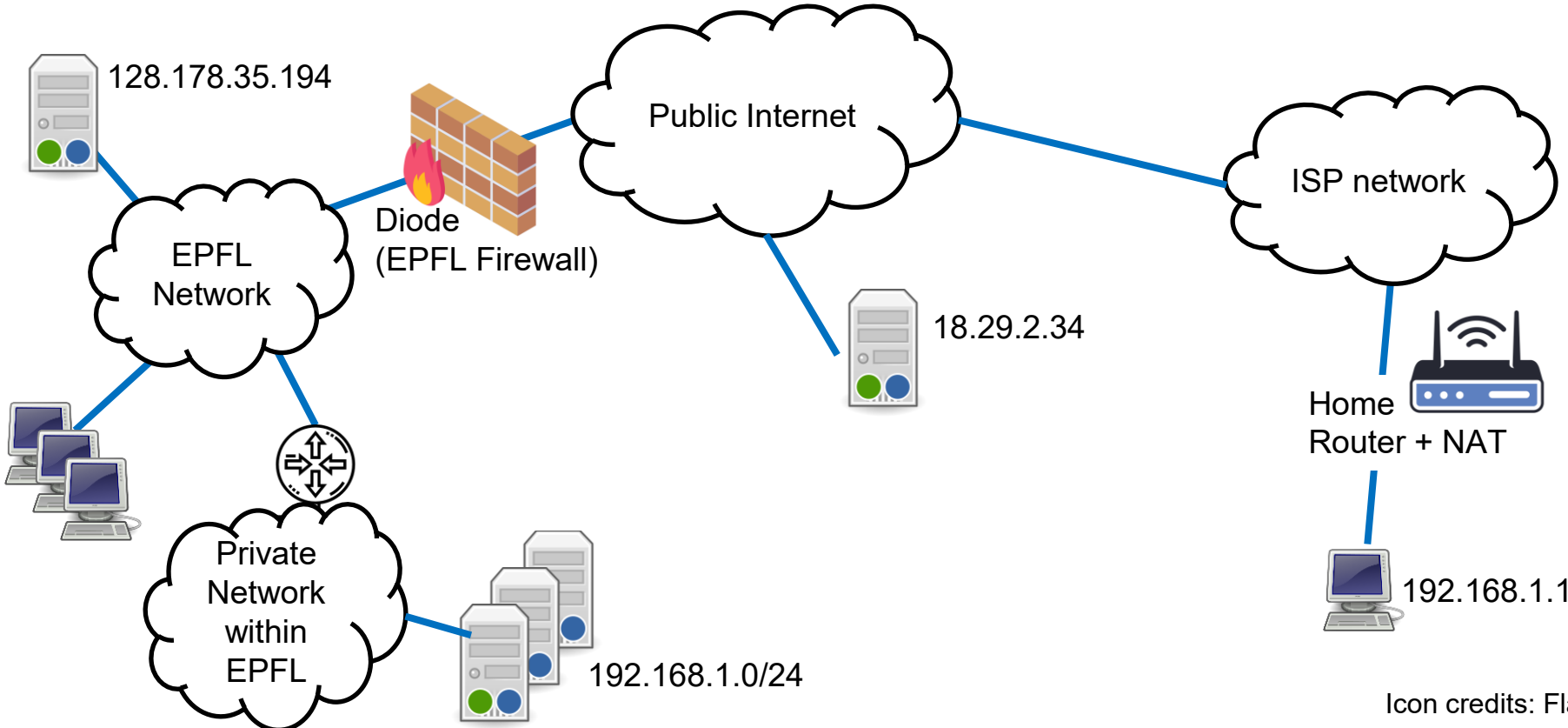
Can we aim for $O(\log n)$ efficiency ?
→ later today

Ad-hoc Routing Protocols

Finding your way through ad-hoc networks

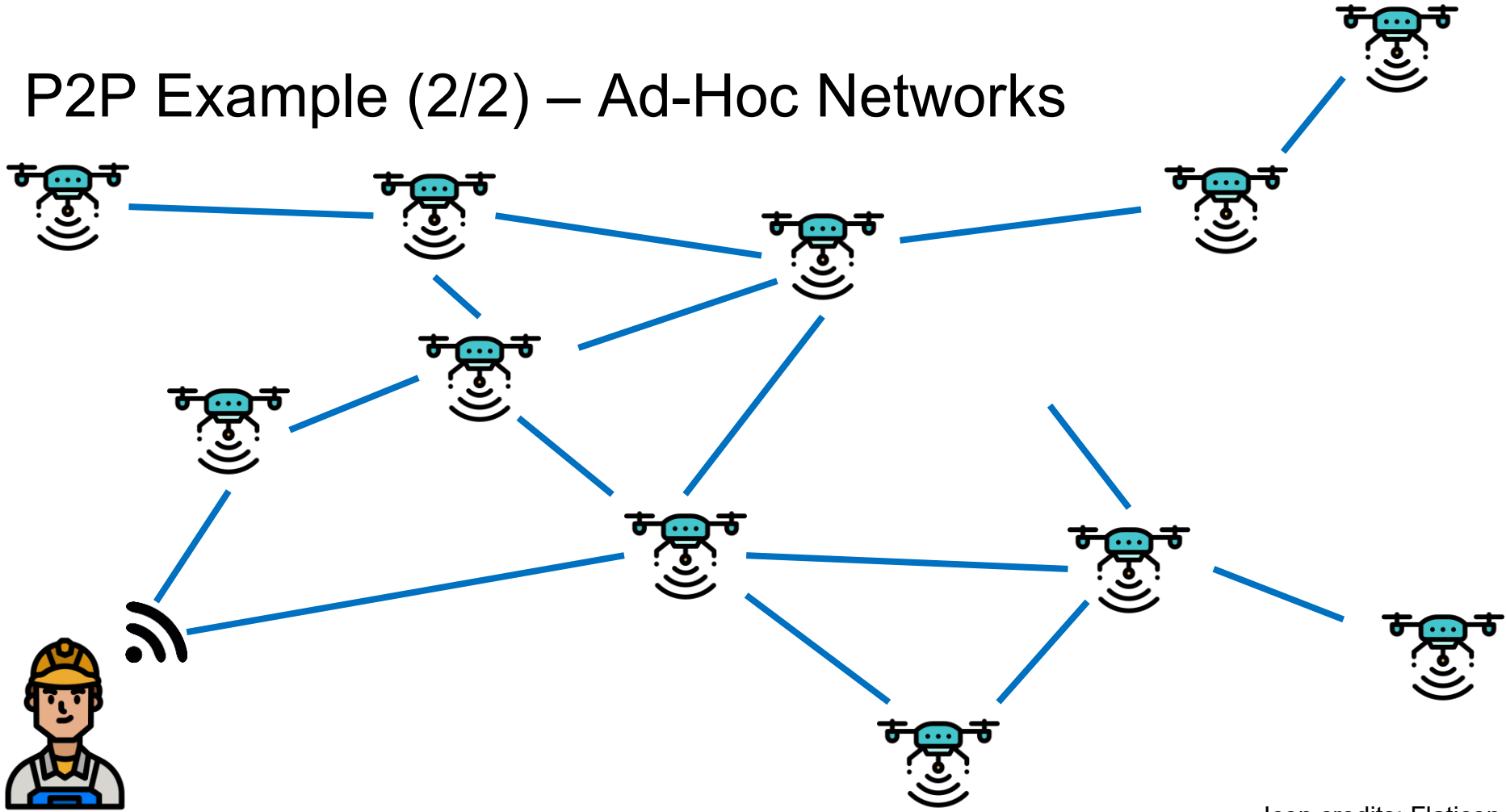
(Homework 1)

P2P Example (1/2) – Your Home & EPFL Networks



Icon credits: Flaticon

P2P Example (2/2) – Ad-Hoc Networks



Icon credits: Flaticon

P2P Examples – Quick analysis

- Peers may not be directly accessible
- Peers may join or leave the network at arbitrary times
- We need to route packets through the system

Some differences:

-
-
-
-

Naive routing – don’t do this at home

Nodes advertise a distance to other nodes

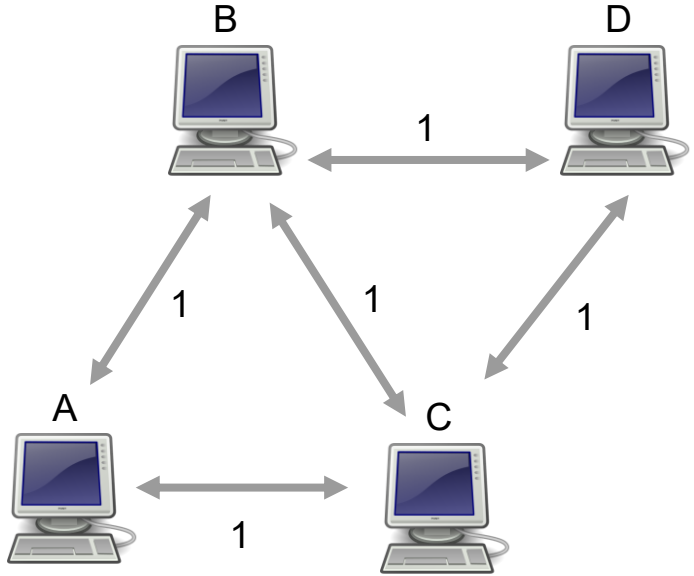
B → D = 1

A → D = 2 (through B or C)

C → D = 1

On link failure, B updates:

Then C updates:

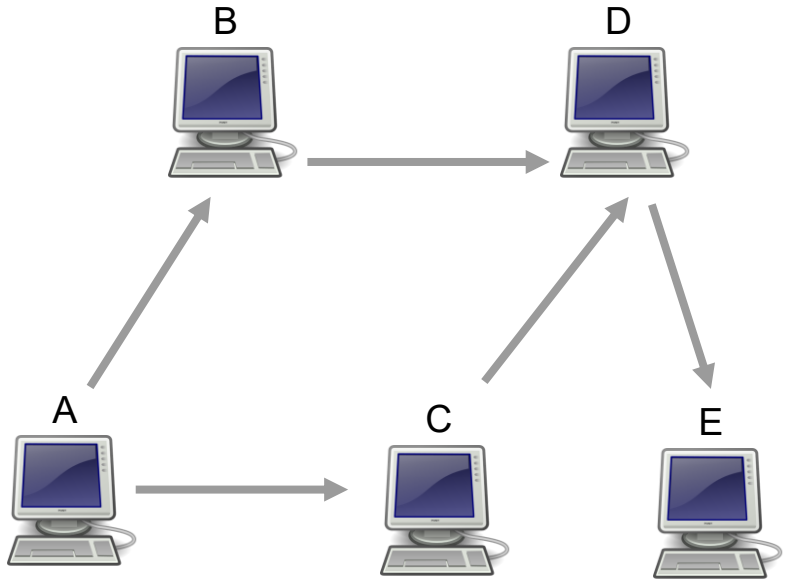


Reaching arbitrary peers in a network : AODV

Ad-hoc On-demand Distance Vector

Key idea:

-



Reactive (on-demand) routing, cached

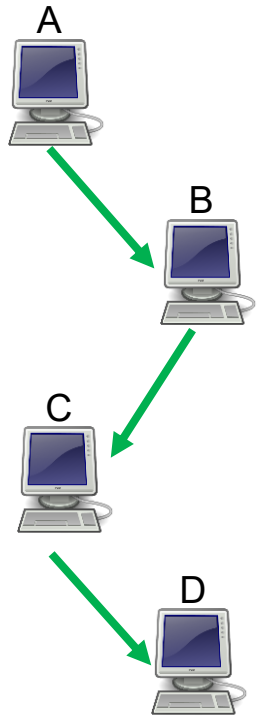
Used in the ZigBee wireless protocol

Reaching arbitrary peers in a network : DSDV

Destination-Sequenced Distance Vector

Key idea:

-
-



Quality factors in ad-hoc routing

-
-
-
-
-

Compact Routing & Structured Search

$O(\log n)$, here we come !

General Approach

- Build a structured *overlay network*
- Enables significant efficiency gains

We'll pay a price:

Distributed Hash Table

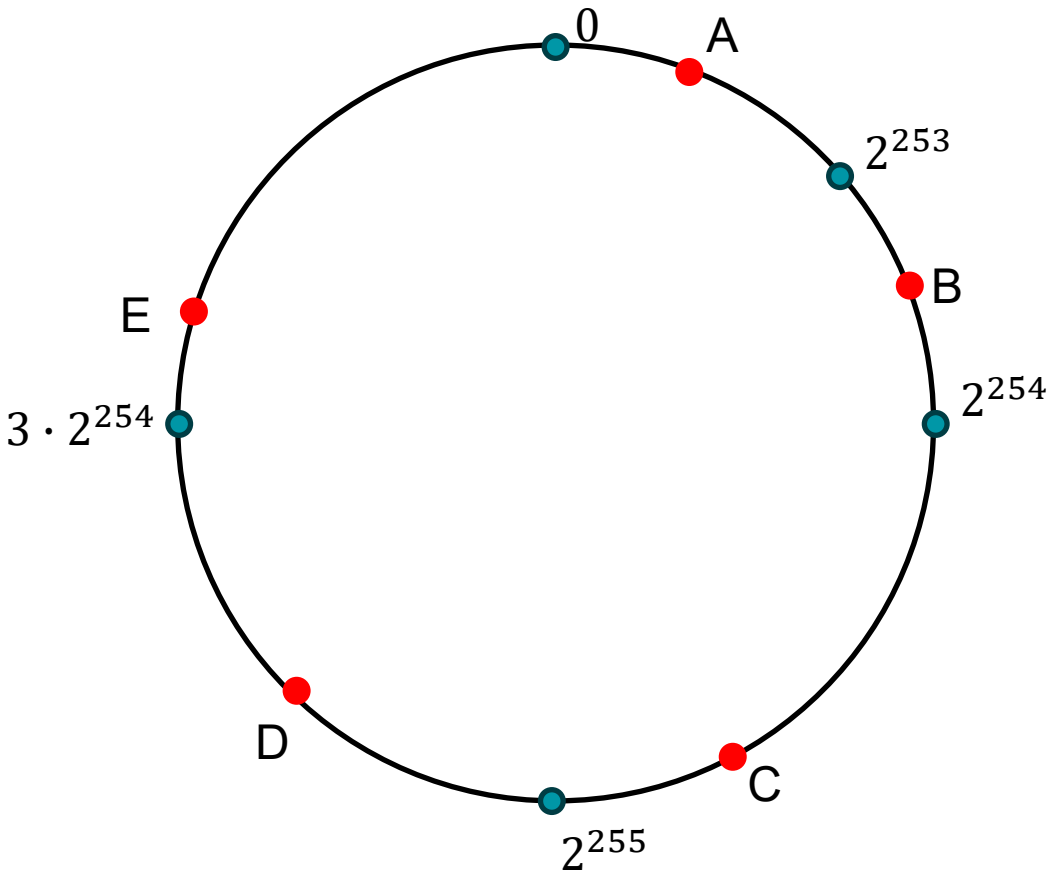
Local hash tables need:

-
-
-

Distributed hash tables considerations:

Chord DHT

- Hash into a collection of RAMs



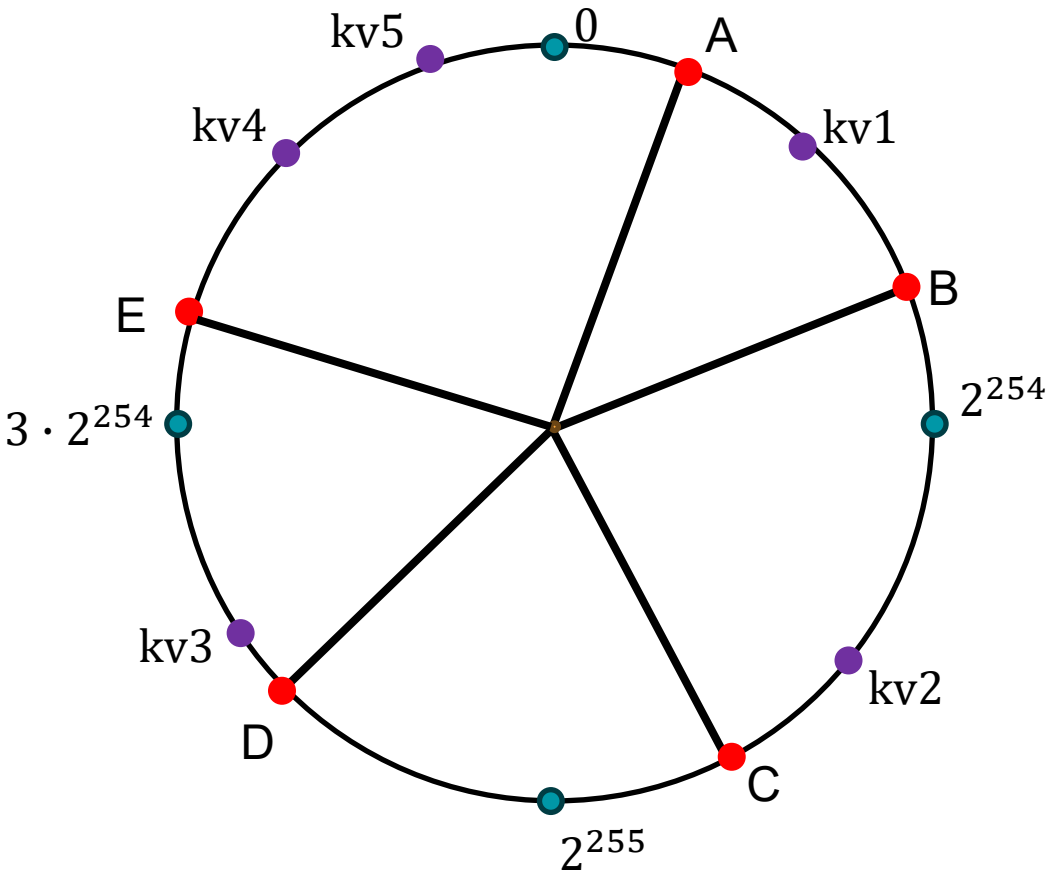
Chord DHT

How do we approximate RAM ?

-
-

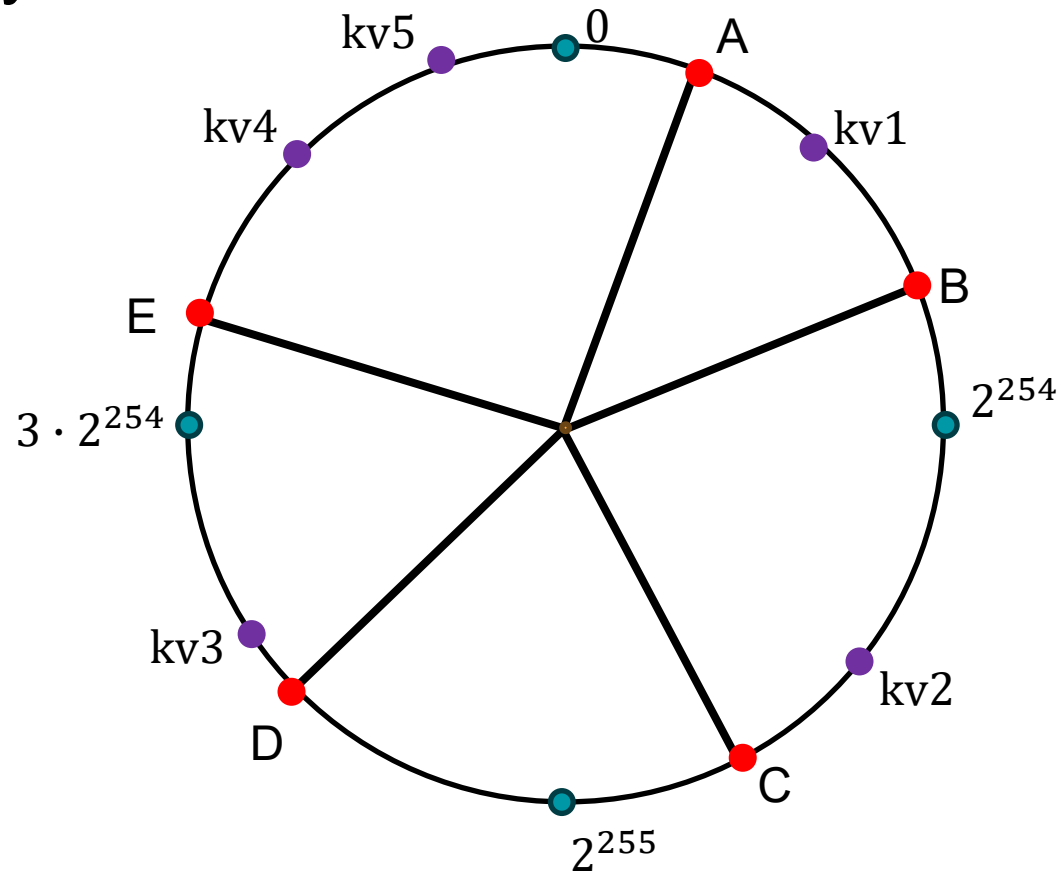
API:

-
-



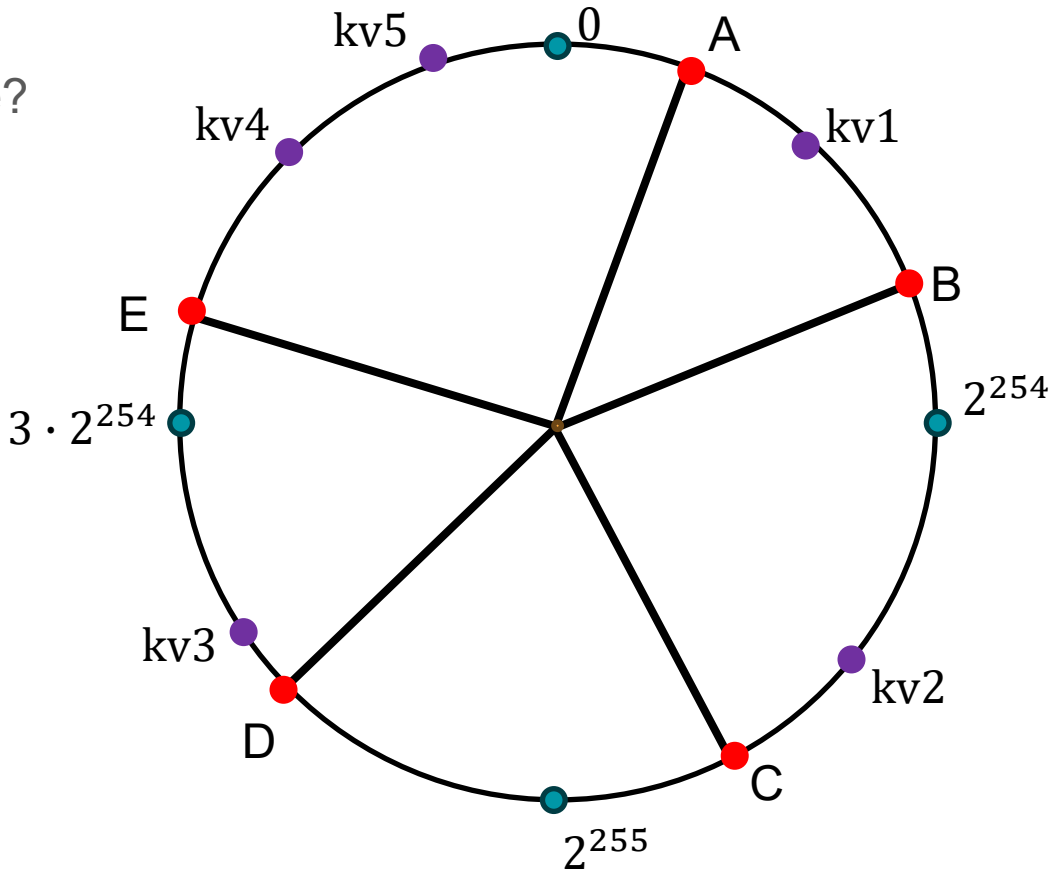
Chord DHT – Reliability

How do we prevent data loss ?



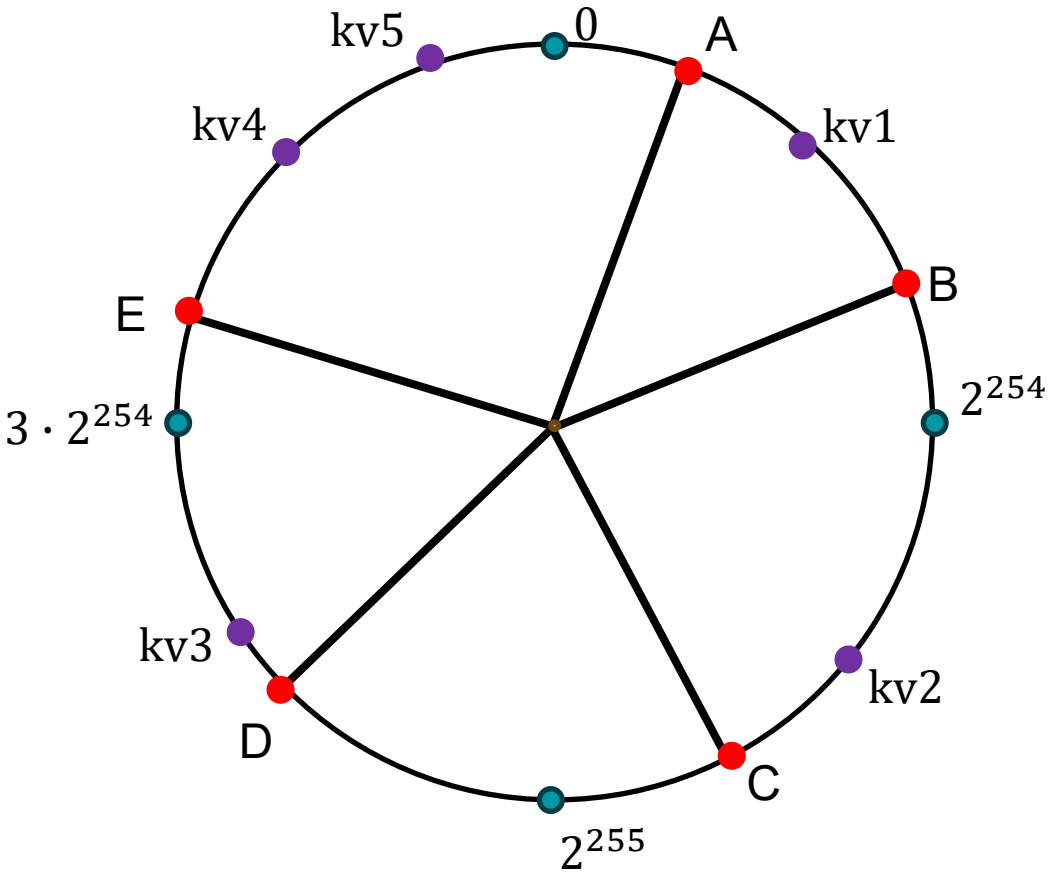
Chord DHT – Load

What is the expected load per node?



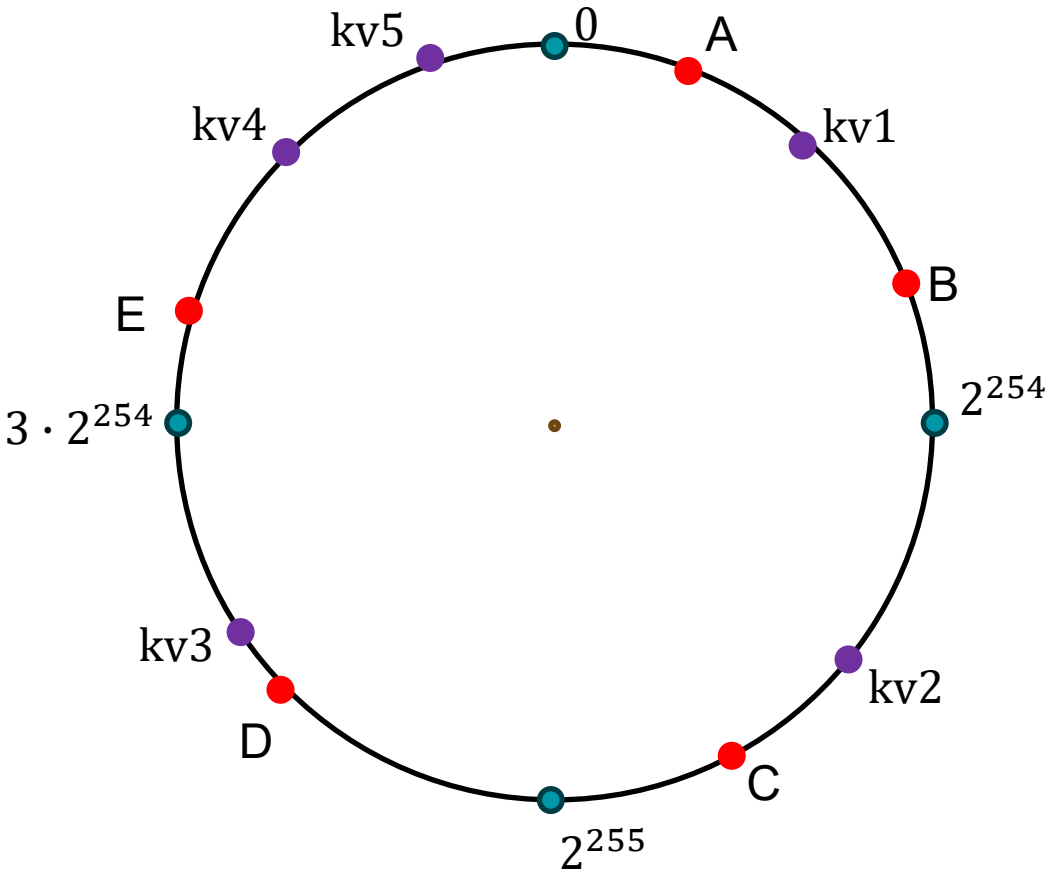
Chord DHT – Performance

How do we make this $O(\log n)$?
(in storage, network, etc.)



Chord DHT – Finger tables

Distance	Bucket
1	(successor) = B
½ circle	
¼ circle	
⅛ circle	
...	...



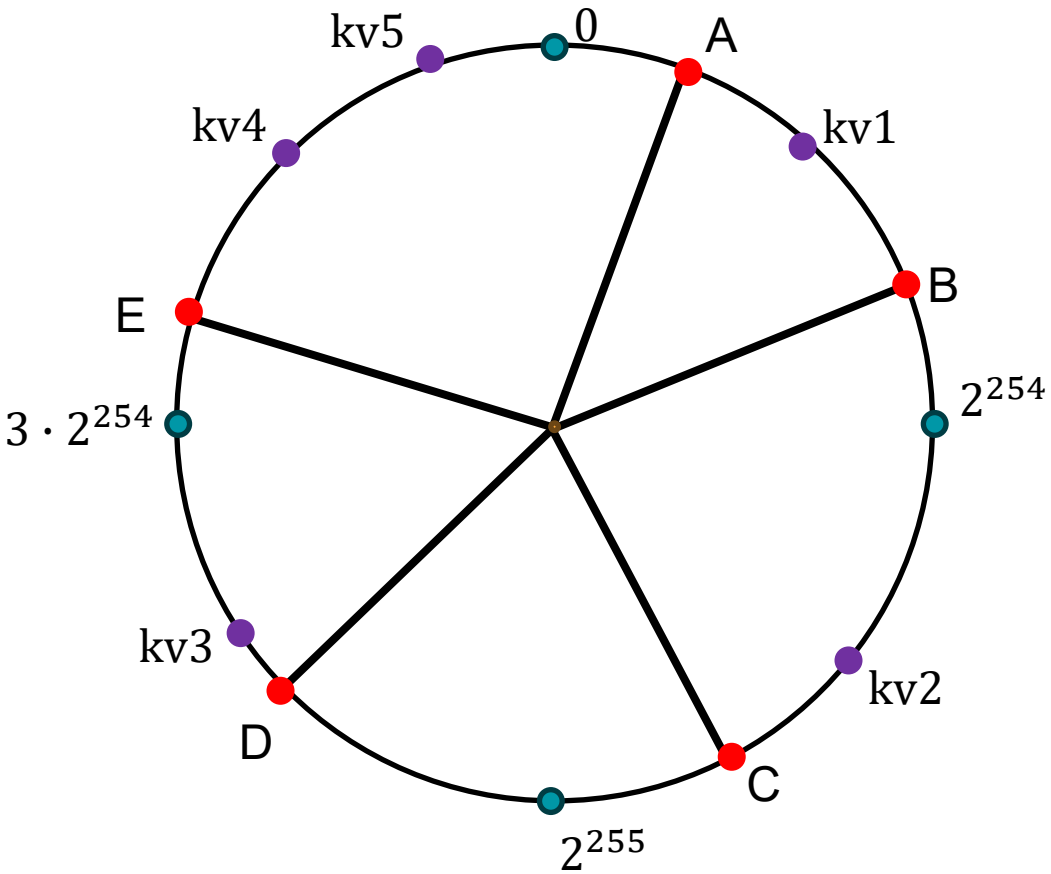
Chord DHT – Churn

Need to handle:

-
-
-

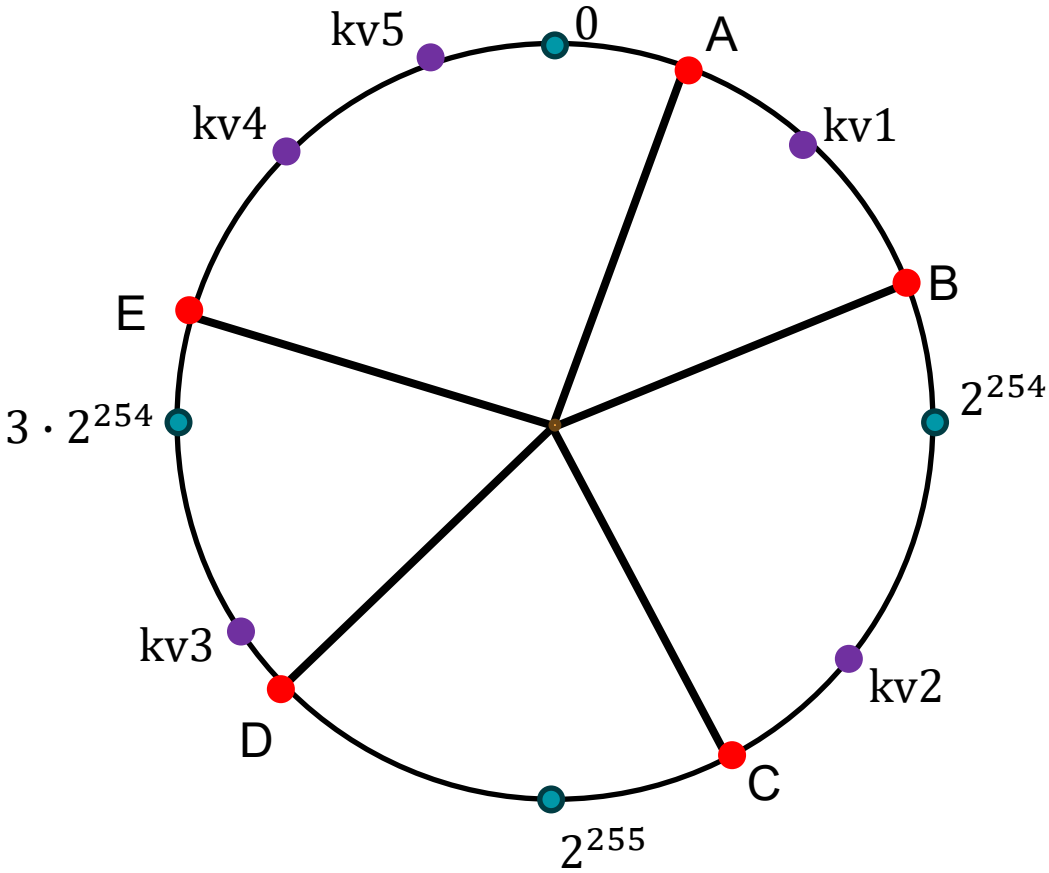
Approach:

-
-
-



Chord DHT – Degenerate cases

Network partitions can lead to some (transient) degenerate cases.



Chord DHT – Possible attacks

-
-
-
-
-

Next steps - Readings

Mandatory:

- DSDV: Routing over a Multihop Wireless Network of Mobile Computers
- Ad-hoc On-Demand Distance Vector Routing (AODV)
- Chord: A Scalable P2P Lookup Service for Internet Applications

Recommended (Engineering):

- The Babel Routing Protocol (RFC 8966)
- Kademlia: A Peer-to-Peer Information System Based on the XOR Metric

... and a few others for the curious among you ...

→ Use Friday's session to ask questions