Fault-Tolerant Broadcast of Routing Information

Radia Perlman
Digital Equipment Corp.

Computer Networks (1983)

Presented by: Yusuf Sarwar

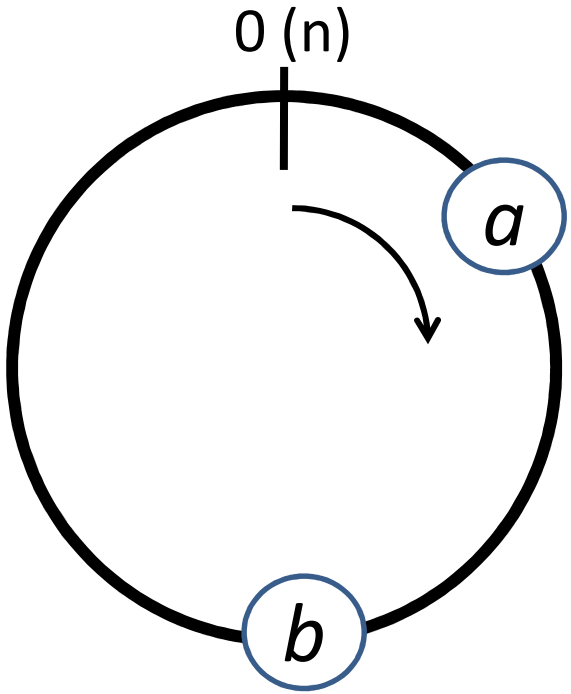
ARPANET routing

- Uses Link State Routing protocol
  - link information (link state packets) to neighbors are broadcasted to *all* nodes
    - By “Flooding”
    - Issues link updates now and then
- How to recognize the most recent info?
  - Global clock
    - Not quite possible in practice
  - Local clock
    - Can have arbitrary skew, and have no global meaning

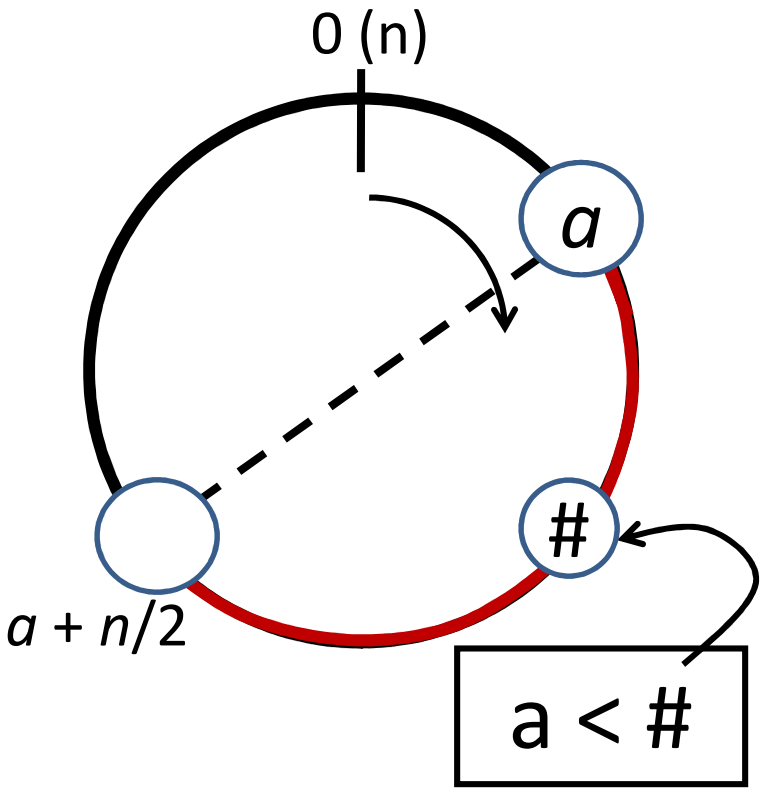
Sequence number

- A counter
  - Possibly large (64 bits)
  - Incremented each time a link packet is generated
- Counter may reach to the maximum value
  - Can be reset
    - *How to tell all nodes to reset, …*
  - Wrap around
    - *Which is newer, which is older, …*
- The problem: using seq field, how to manage link updates in the presence of failures

Circular sequence number

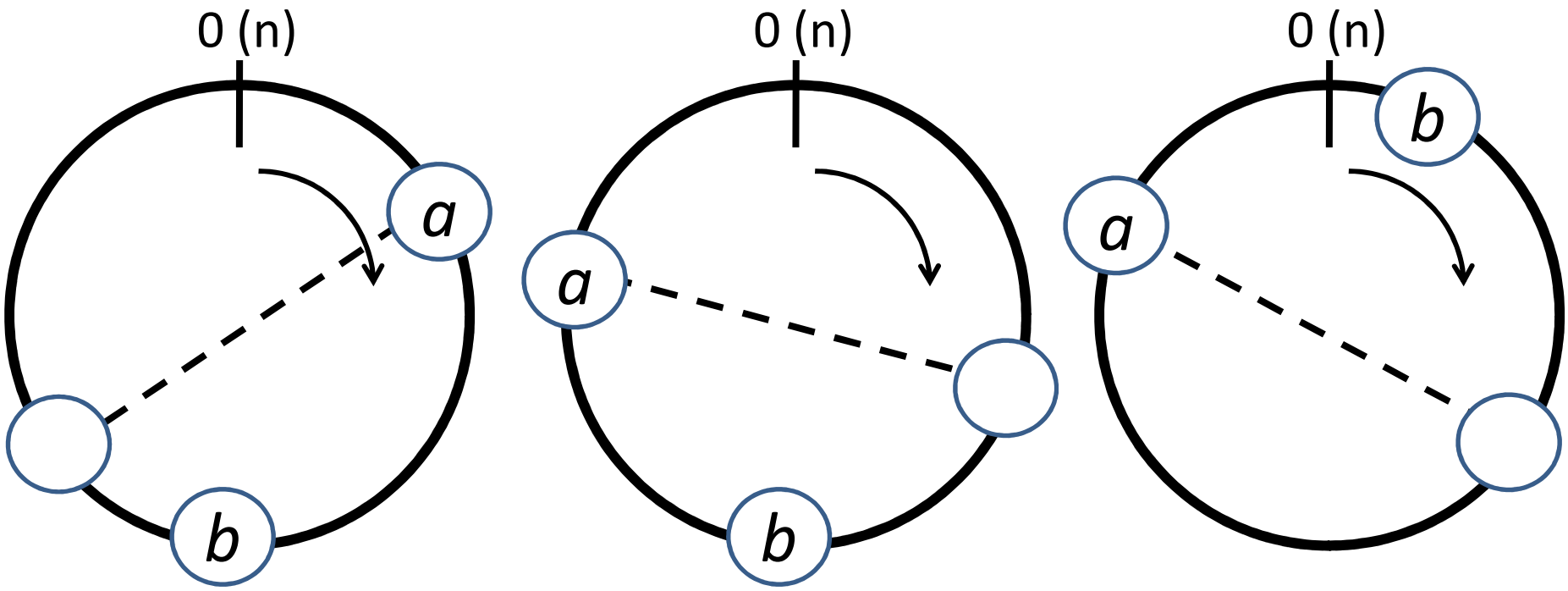


$a > b$ or $b < a$?



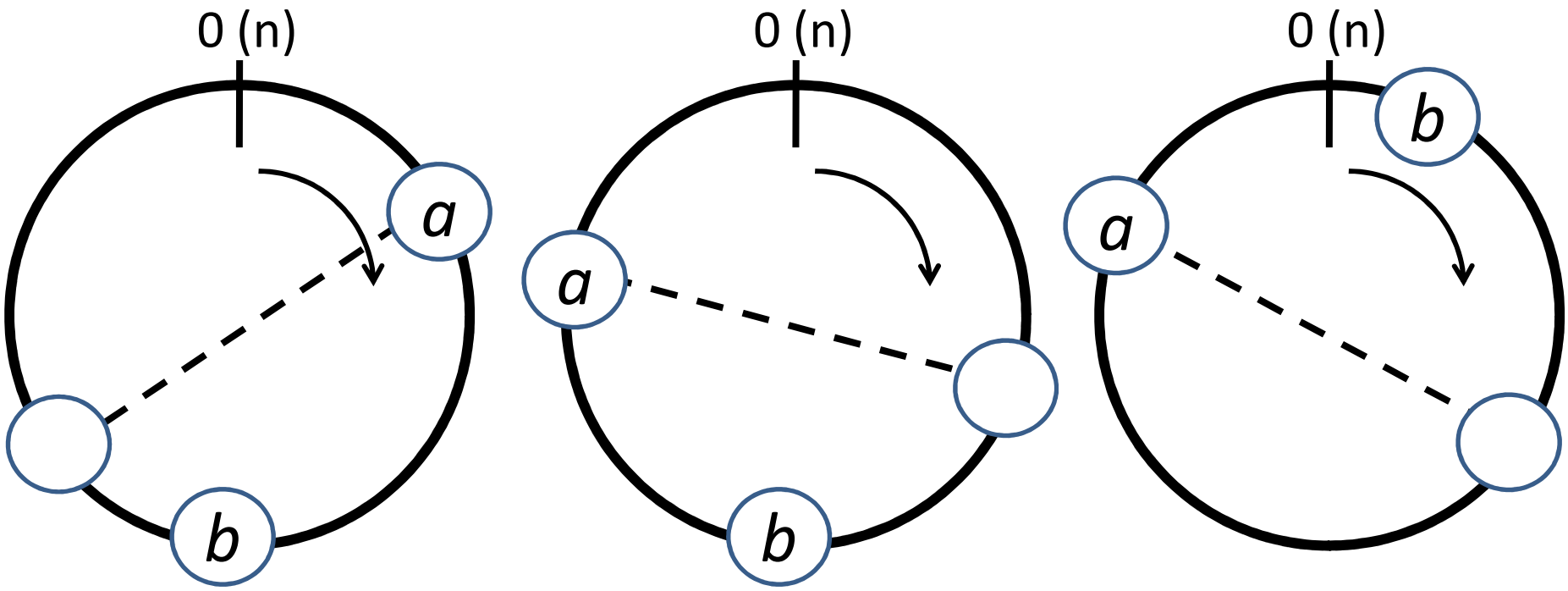
A quick test

Is $a < b$?



A quick test

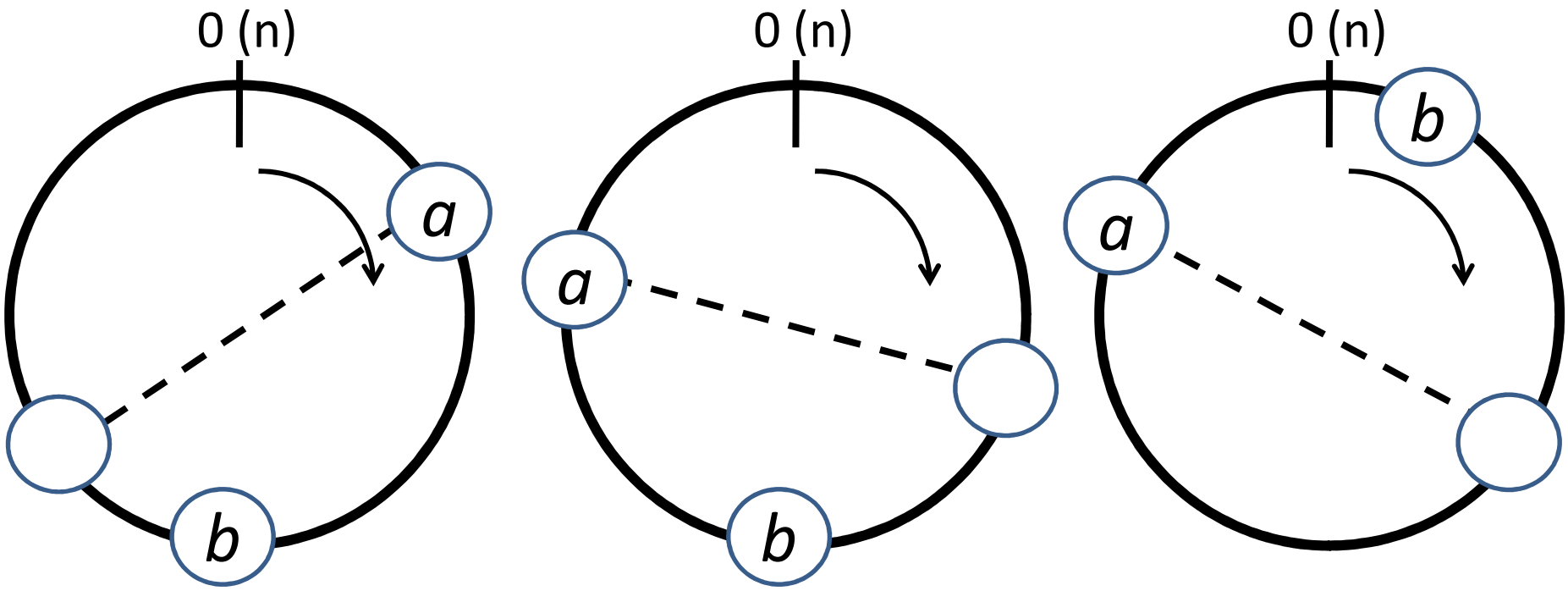
Is $a < b$?



YES

A quick test

Is $a < b$?

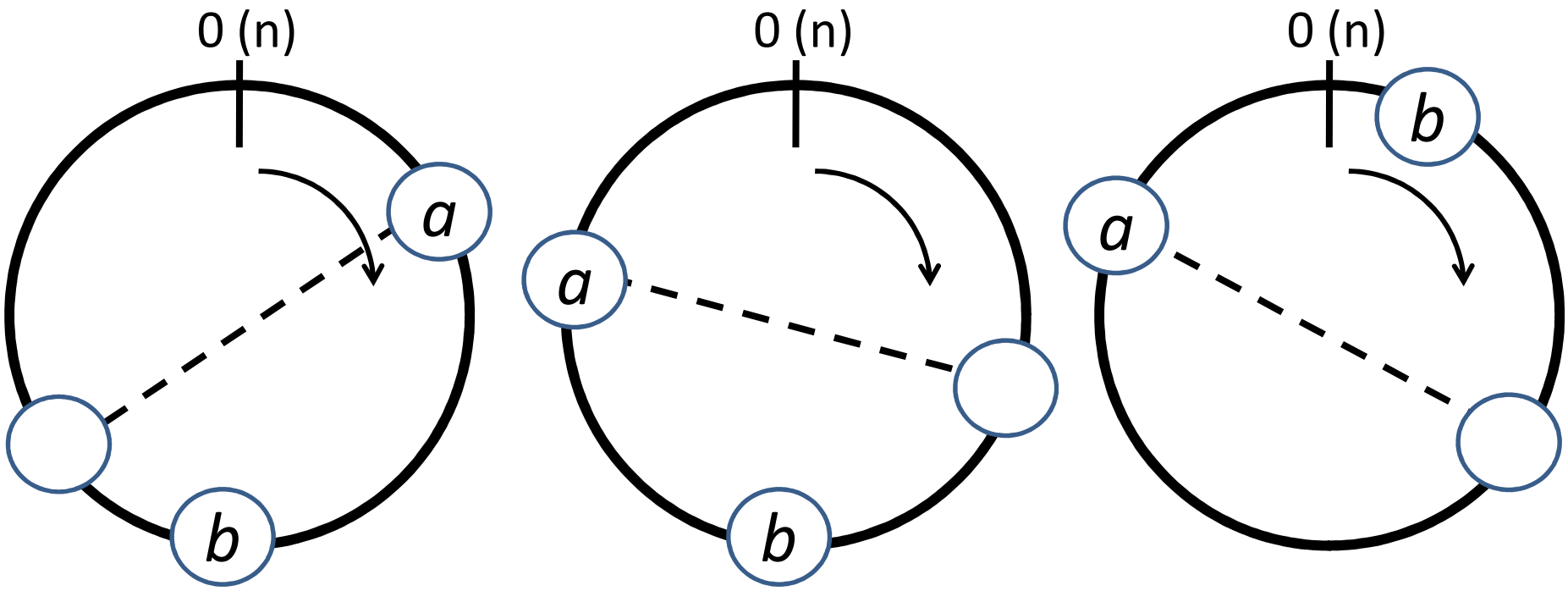


YES

NO

A quick test

Is $a < b$?



YES NO YES

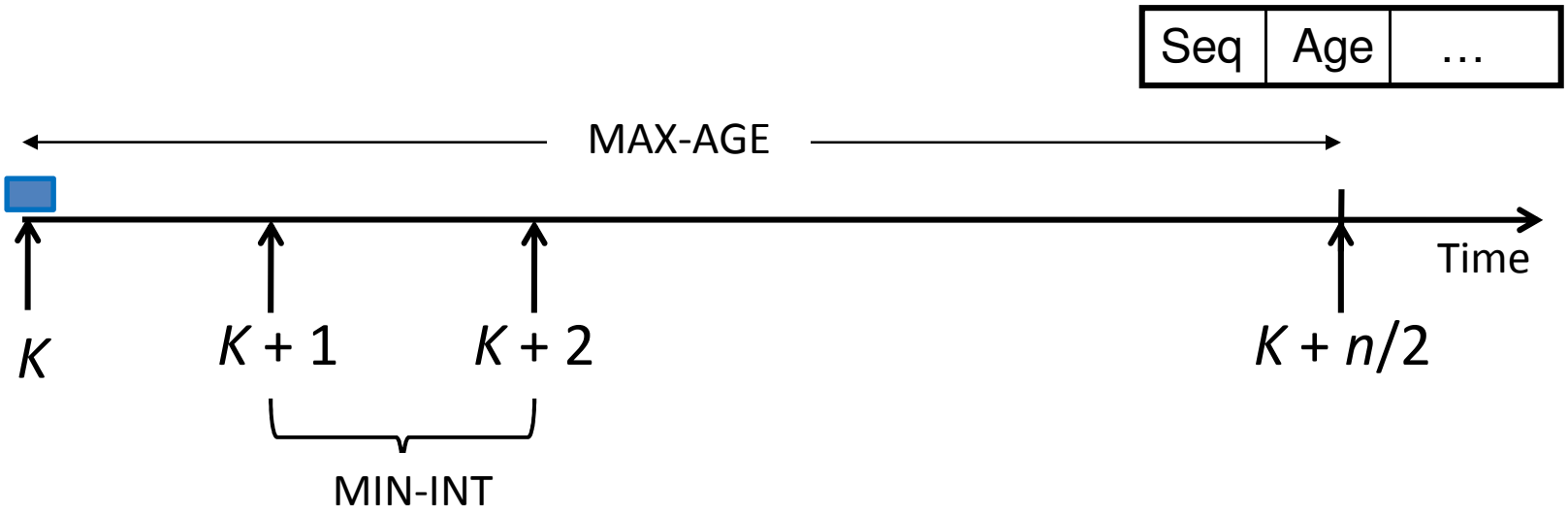
Seq. number + Aging

- Only sequence number is not enough
- There are some problems:
  - Node goes down and comes back
  - Network partition
  - Corrupted sequence number

- Aging is required
  - A life-time after which link packet expires
    - Possibly before seq number wraps around

ARPANET routing

- MIN-INT: interval between two link packets
- MAX-AGE: maximum age, set at source and decremented at others (at every 8 seconds)



Should not issue more than n/2 packets within MAX-AGE

i.e., MAX-AGE / MIN-INT < *n*/2

Problems and fixes

- Immortal packets:
  - Packets that become neither older nor aged
  - Due to non-aging (holding packets less than 8 seconds)
  - **Solution:** Decrement age with some probability
- Premature aging and old packets:
  - Packets are aged out far before they reach all nodes, or already expired
  - **Solution:** Floods the packets with zero age

RESTART-TIME

- If node restarts after a failure, they cannot send link packets immediately
  - Don’t know what sequence to use
  - Have to wait until all previous packets die out
- Usually, RESTART-TIME > MAX-AGE

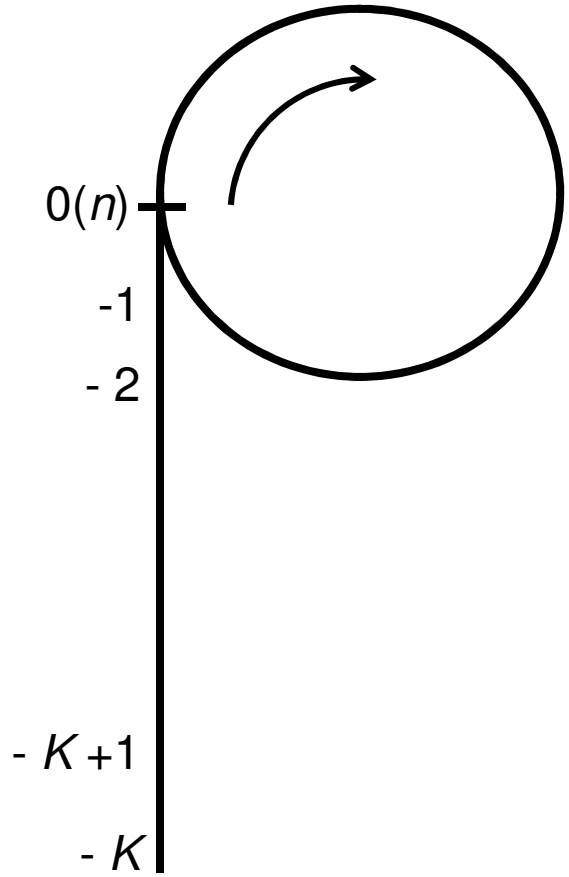
- This introduces delay in update propagation

RESTART-TIME elimination

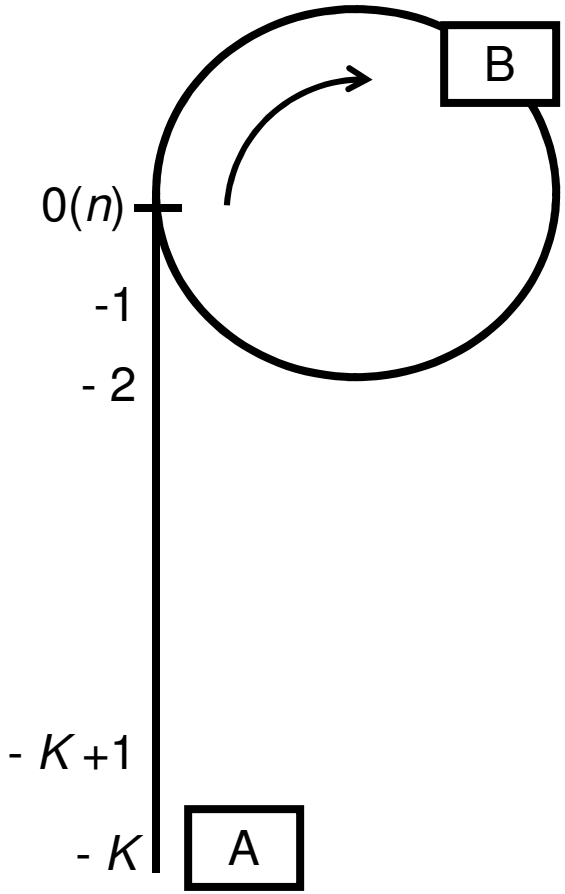
- Three modifications eliminate this waiting
  - ‘lollipop-shaped’ seq number instead of circular
    - Allows nodes to start with some seq right away
  - Send a stored packet to neighbor in response to receipt of an older packet
    - catches the latest sequence number immediately without waiting for long time
  - Quick obtaining of all information upon restarting

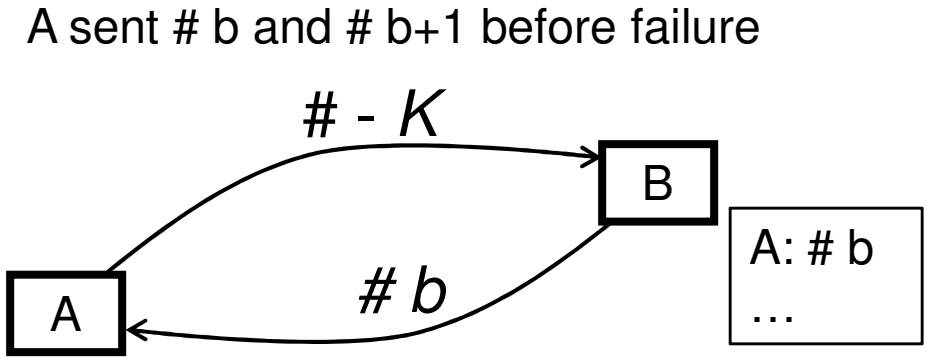
Lollipop-shaped seq space

- Circular seq space does not have the smallest seq number
- Instead, nodes start from – *K*, proceeds to 0-n cycle
  - Once in the cycle, remain there
  - # – *K* must be smaller than any on-going update
- K > MAX-AGE / MIN-INT
  - Node should not be in cycle until MAX-AGE elapses

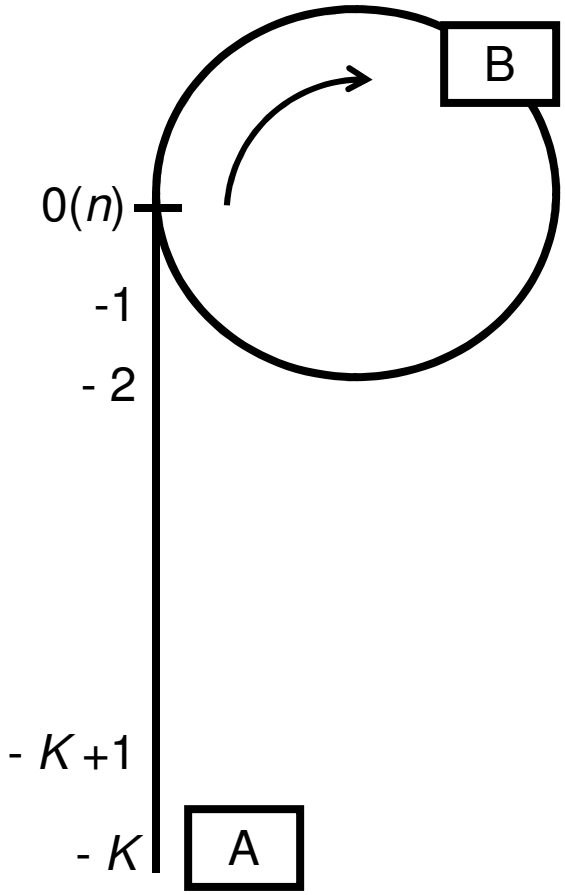


How does “Lollipop”help?

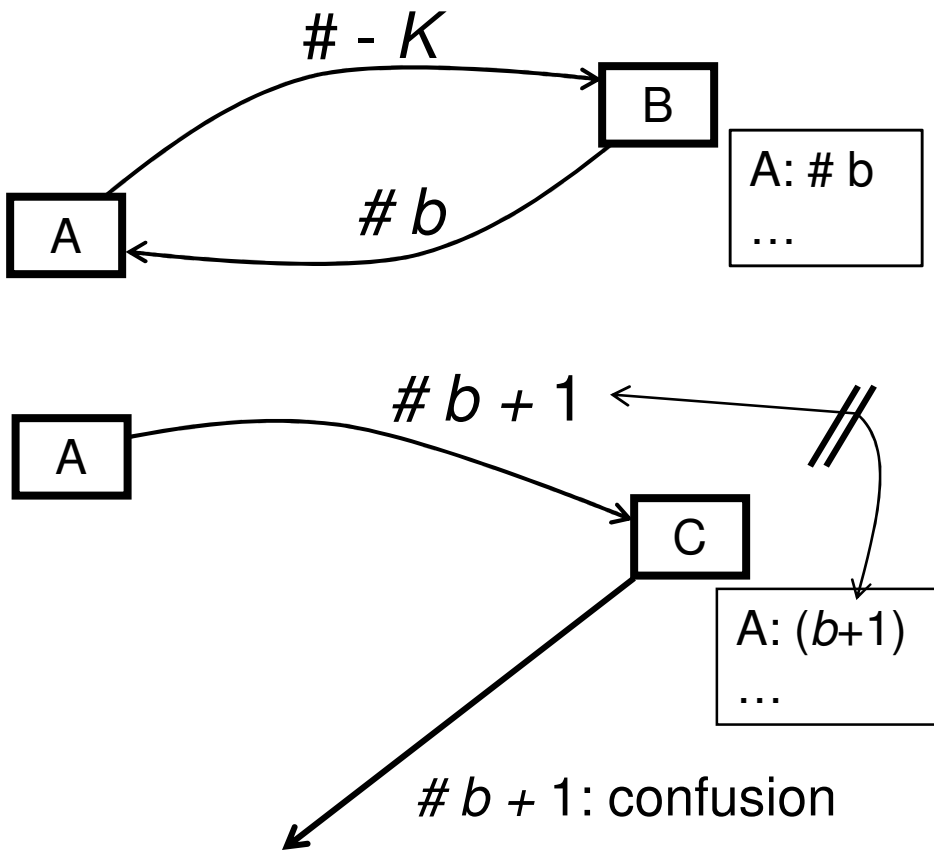




How does “Lollipop”help?



A sent # b and # b+1 before failure



b + 1 < # b+1: confusion < # b + 2

Obtaining info after restart

- Two alternatives:
  - The restarted node sends all neighbors “I haven’t ACKed any link state packet, so send them all”
  - Node A marks B, as line to B goes down and A does not receive ACKs to B
    - As soon as, B arrives, A sends all current link states to B

Summary

- Proposes a sequencing technique to handle updated flooding of link state packets
  - No artificial delay on restart
  - Less reliance of timers, timers are less prone to fire, only on rare events

Discussions

- What could be the other alternatives to “Lollipop” to eliminate RESTART-TIME?
- How to choose parameter values, MAX-AGE and MIN-INT?
- What are the security implications?

Thanks

ARPANET approach

- Source node
  - issues link packets at MIN-INT intervals
  - increments sequence number, $seq_{new} = seq_{old} + 1$
  - sets the age field to MAX-AGE (64 seconds)
- Receiving node
  - accepts, stores and floods, if the packet is newer
    - If the same or older, discard!
  - decrements ‘age’ in every 8 seconds (clock resolution)
    - If age becomes zero, holds in database, but does not flood

Problems and fixes (cont.)

- Old packets
  - Nodes hold old packets, since they don't get a new one from the source
    - But, update may not reach due to routing failure
  - Node holds packets upto MAX-AGE, then?
  - **Solution:** Synchronized expiration
    - Expired packets are flooded with age = 0
    - If the seq number is matched with an unexpired one, accept the zero-aged one and flood
      - Otherwise, discard the packet

Other issues

- Packets checksum
  - To detect duplicate of packets
- Accidental change to confusion bits
- Selecting parameter values
  - Conflicting goals in MAX-AGE, MAX-INT, MIN-INT