

**RNFD:**

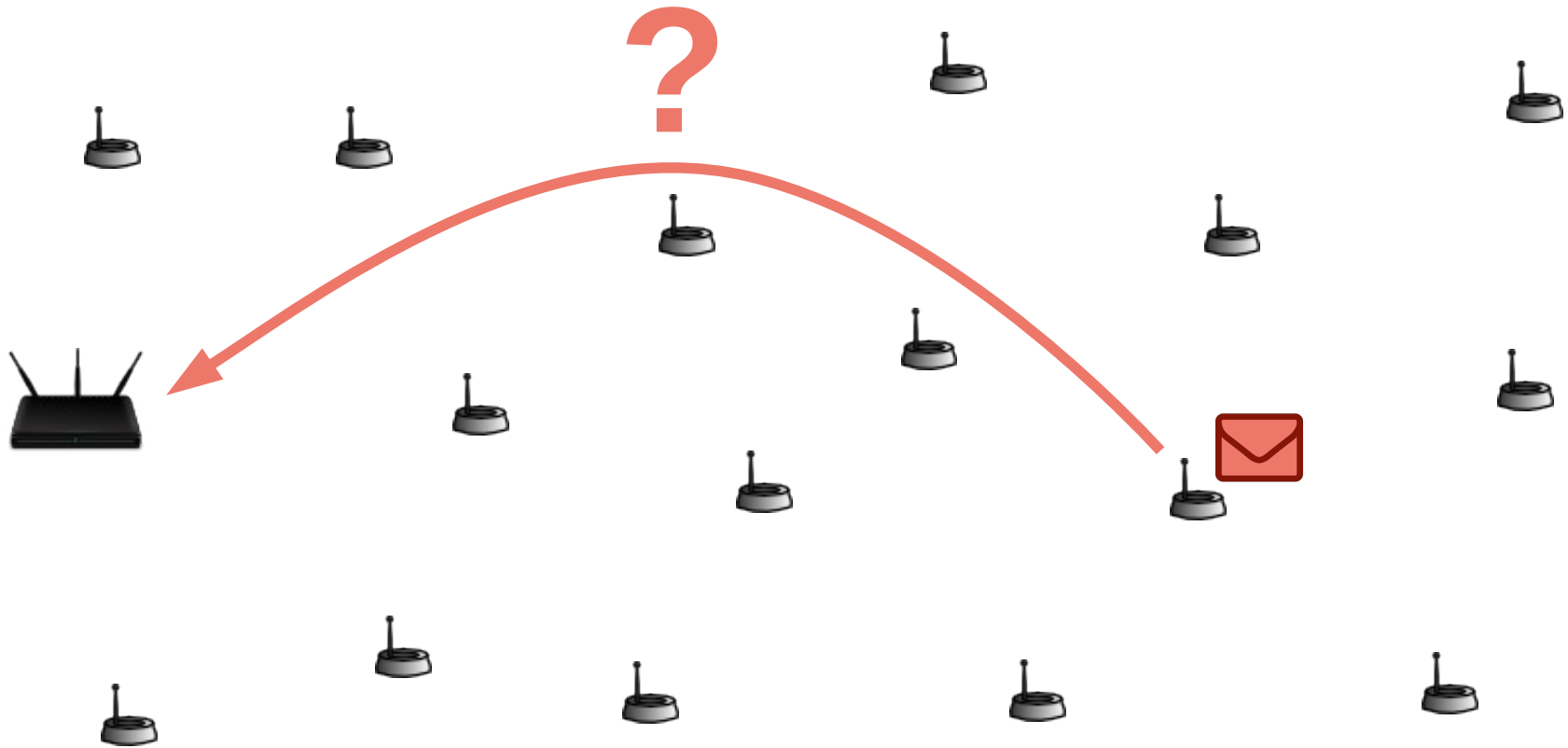
# Routing-Layer Detection of DODAG (Root) Node Failures in Low-Power Wireless Networks

Konrad Iwanicki

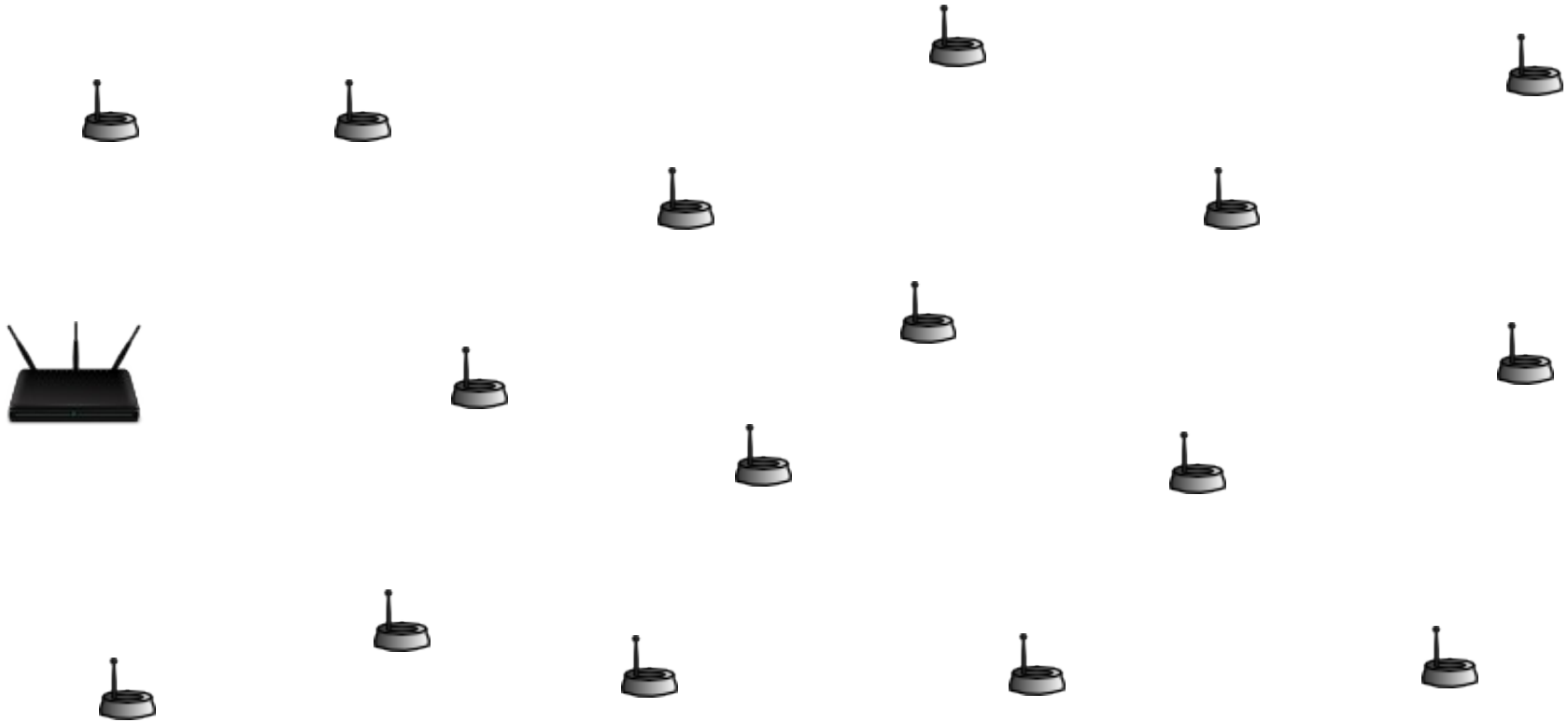
*University of Warsaw*



# Routing



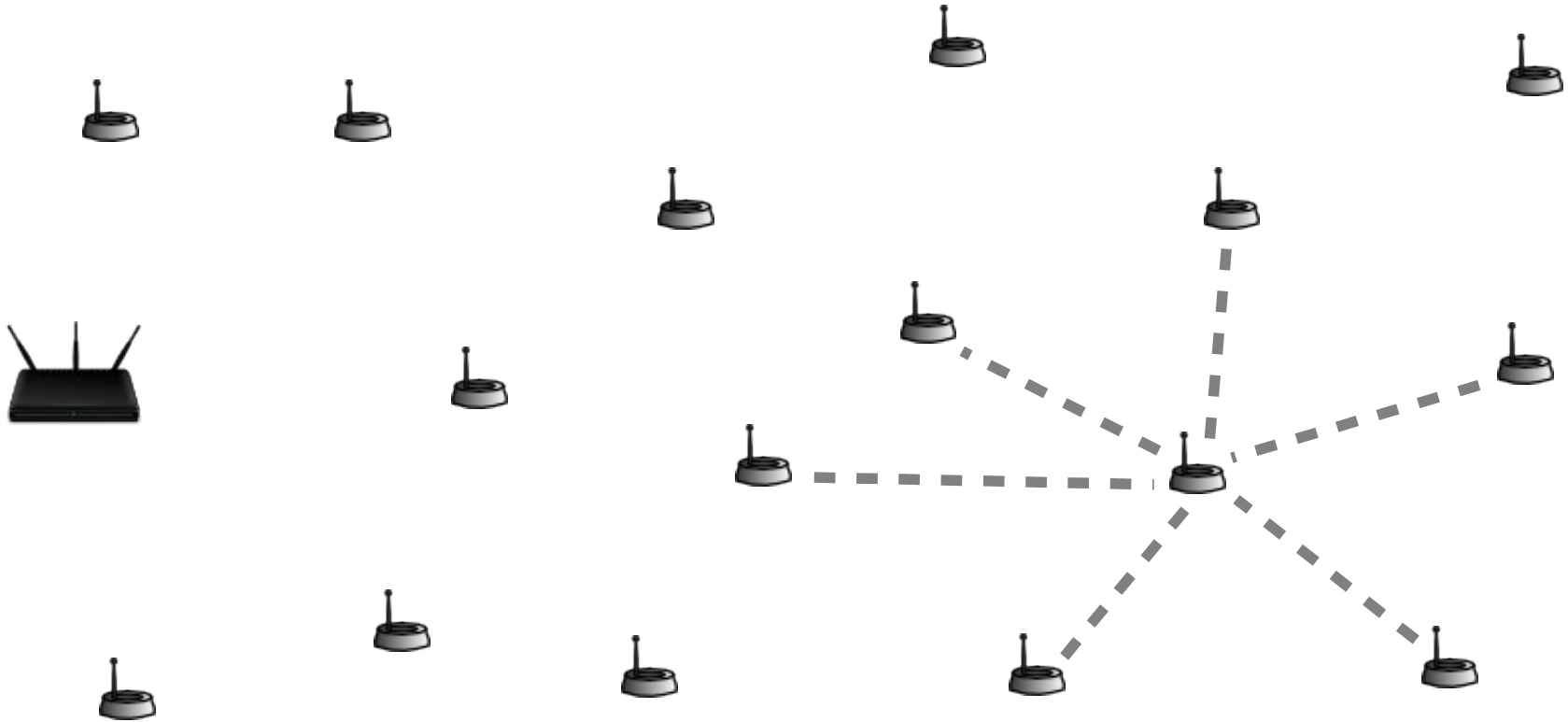
# DODAG



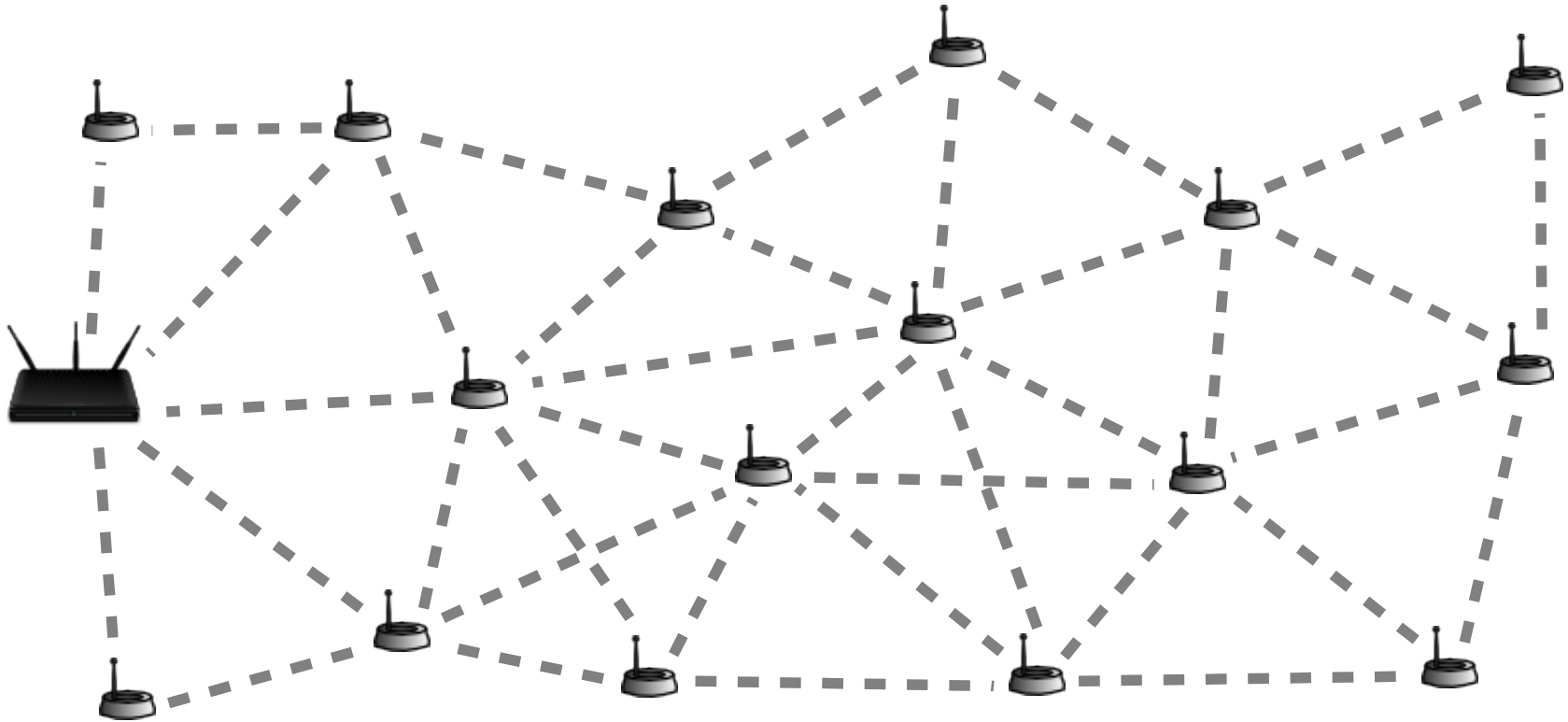
# DODAG



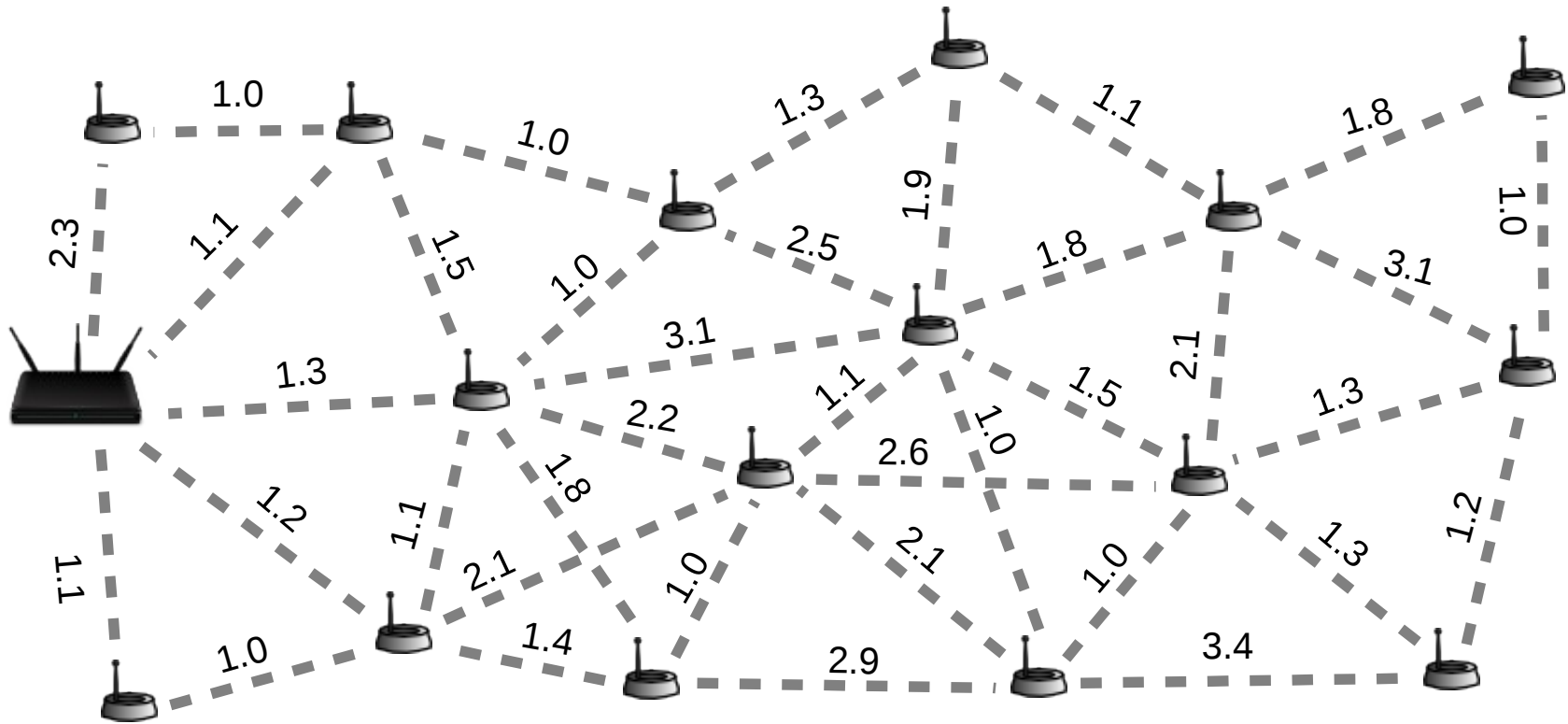
# DODAG



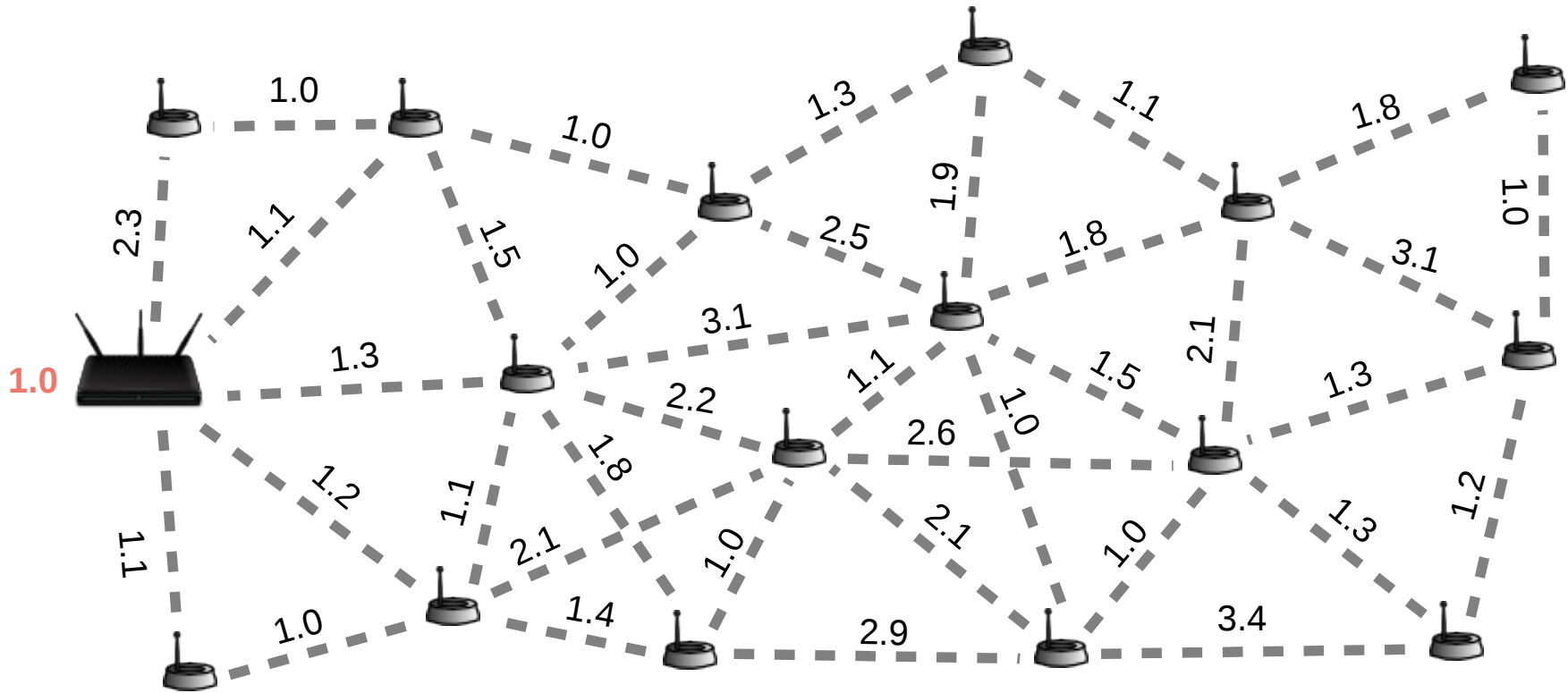
# DODAG



# DODAG

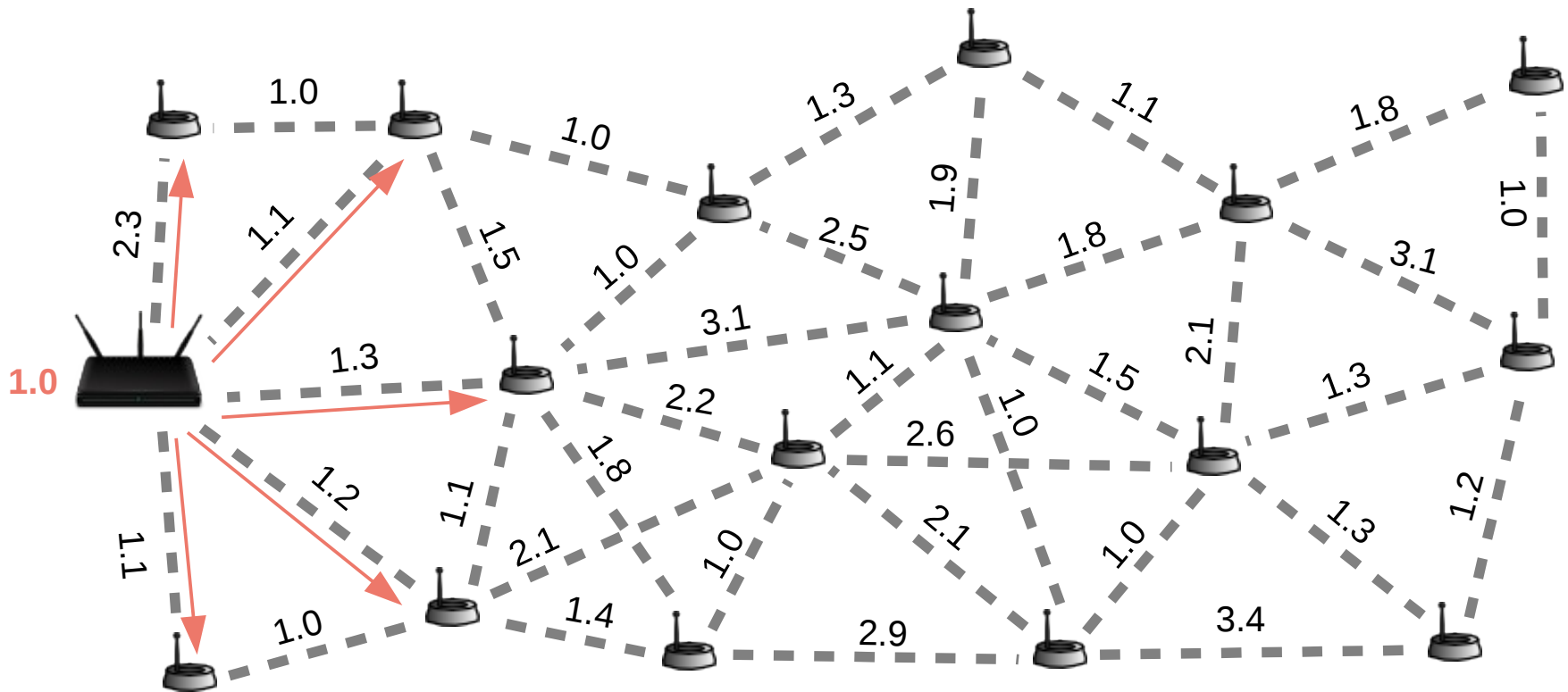


# DODAG

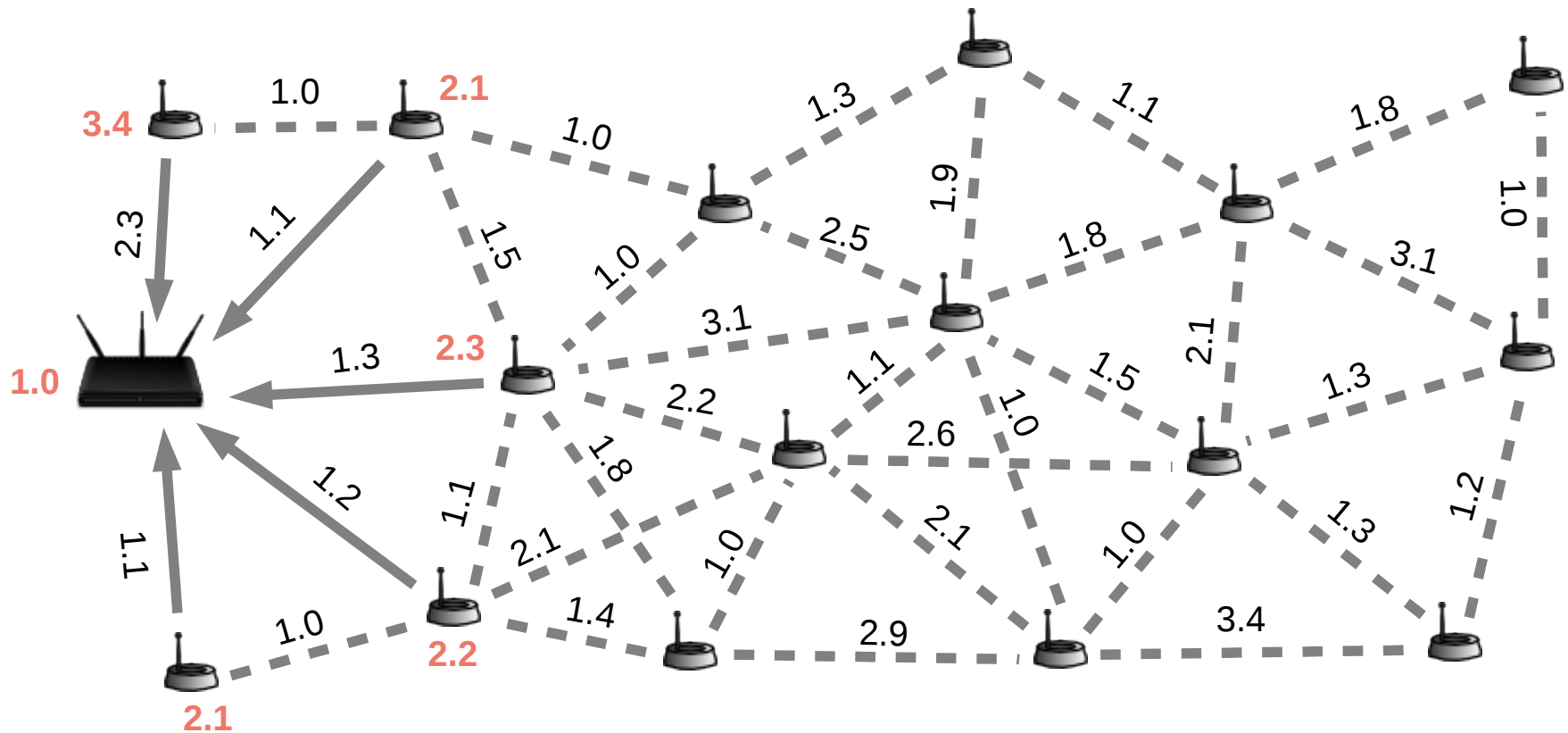




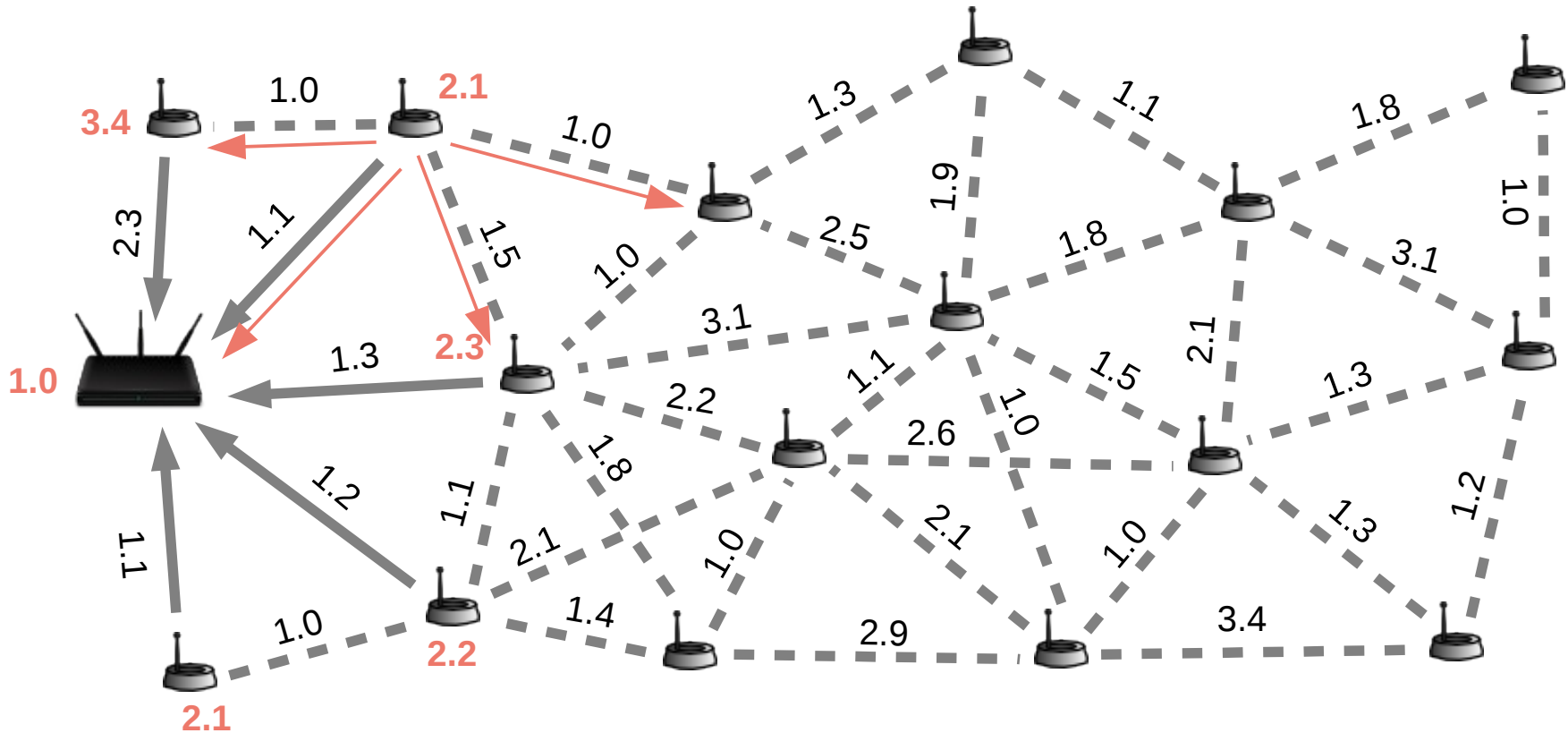
# DODAG



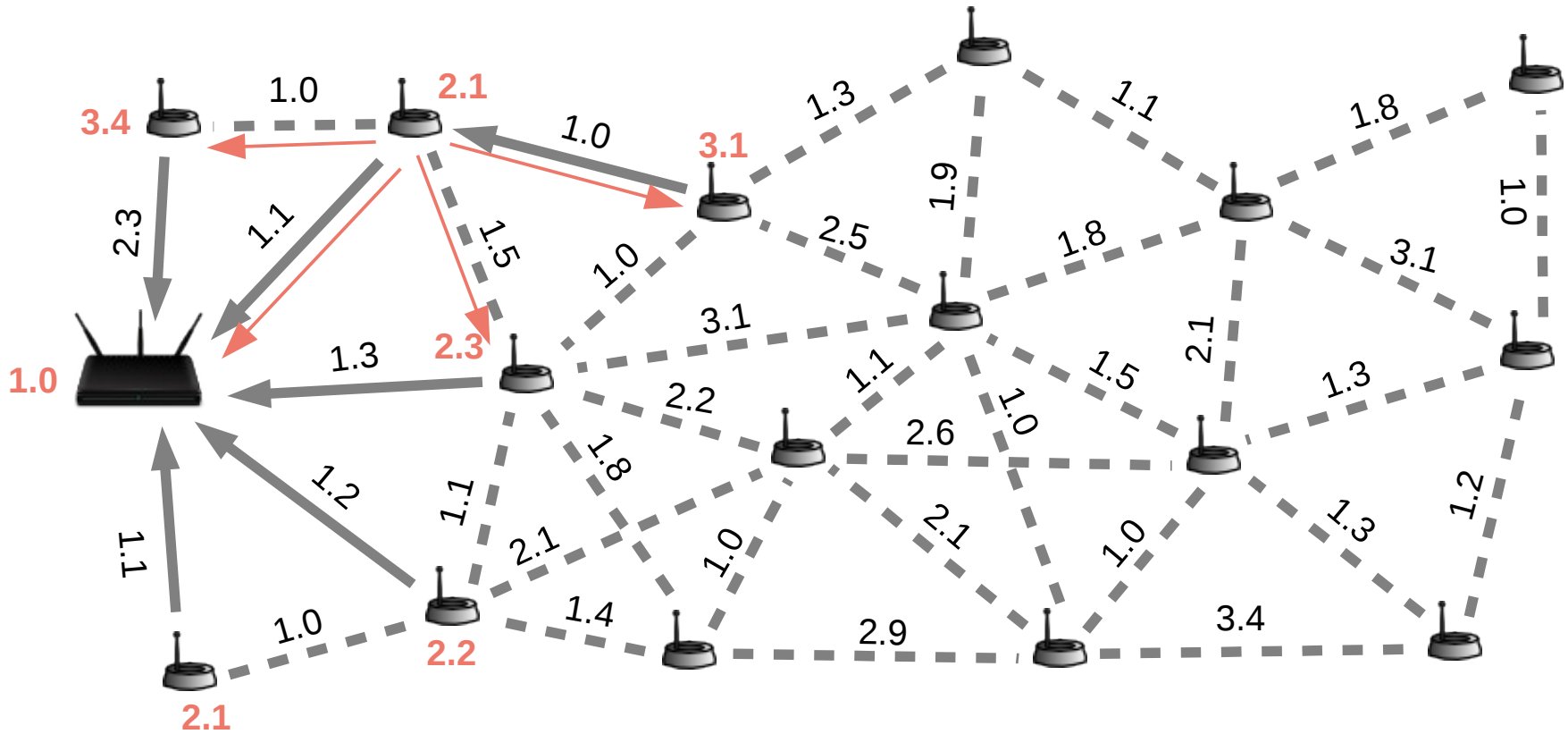
# DODAG



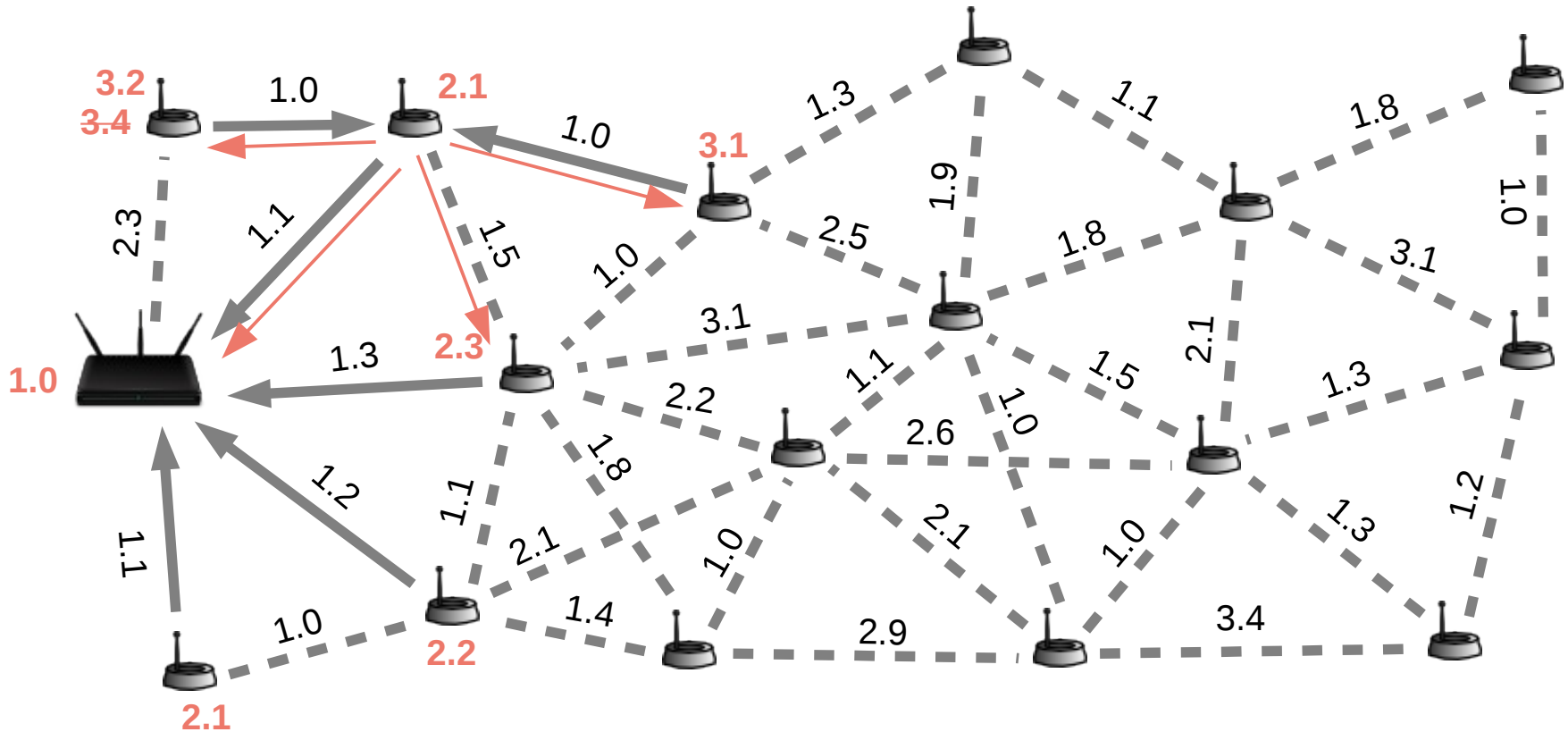
# DODAG



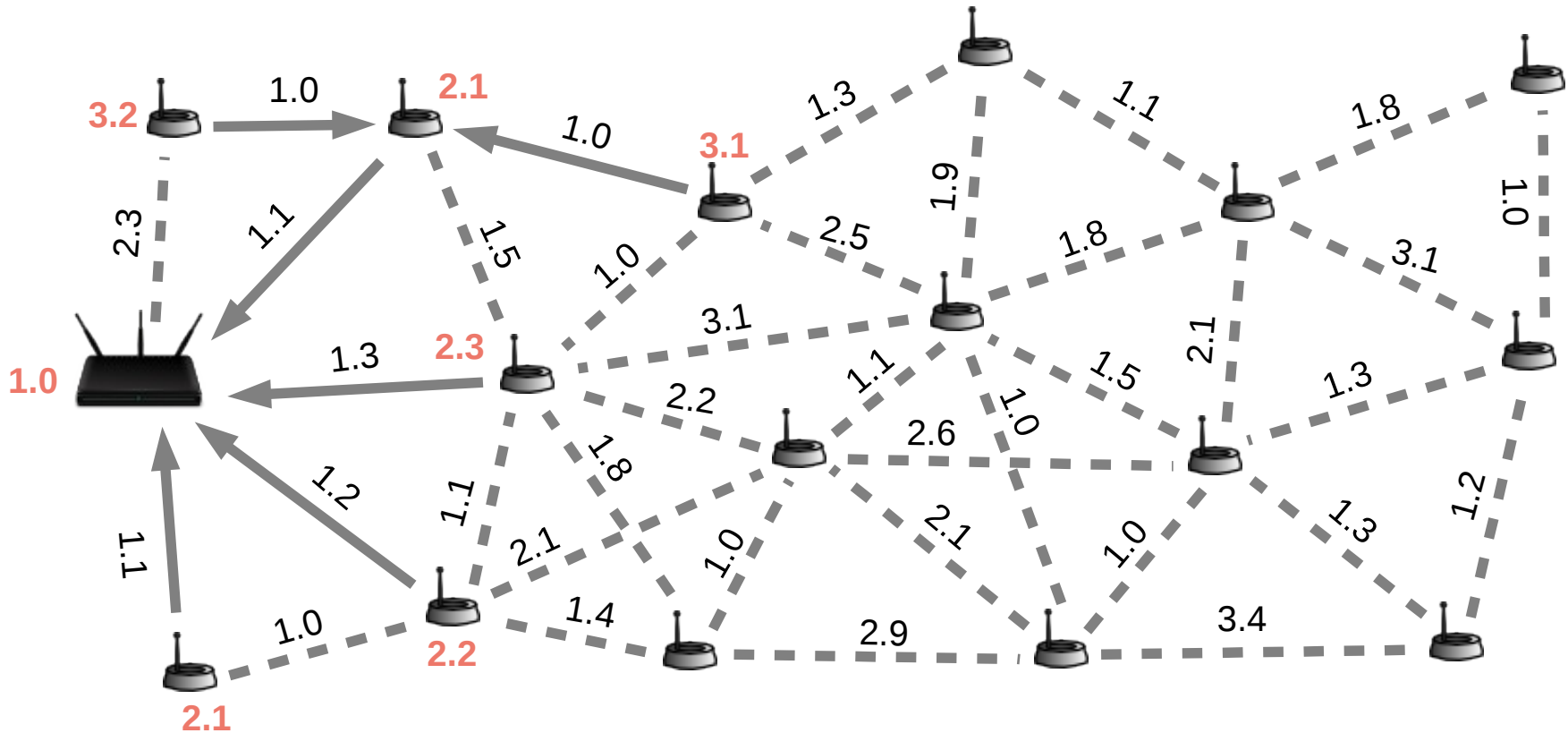
# DODAG



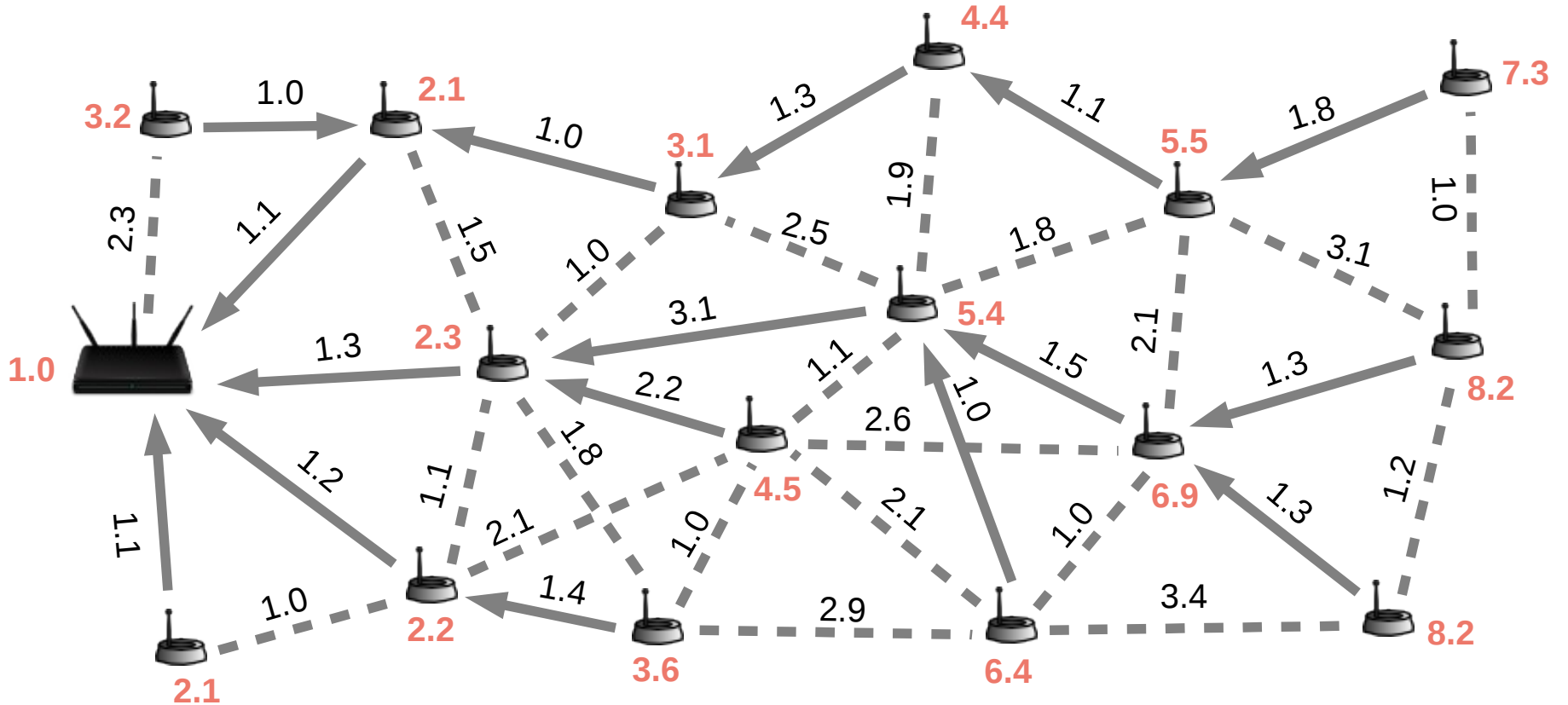
# DODAG



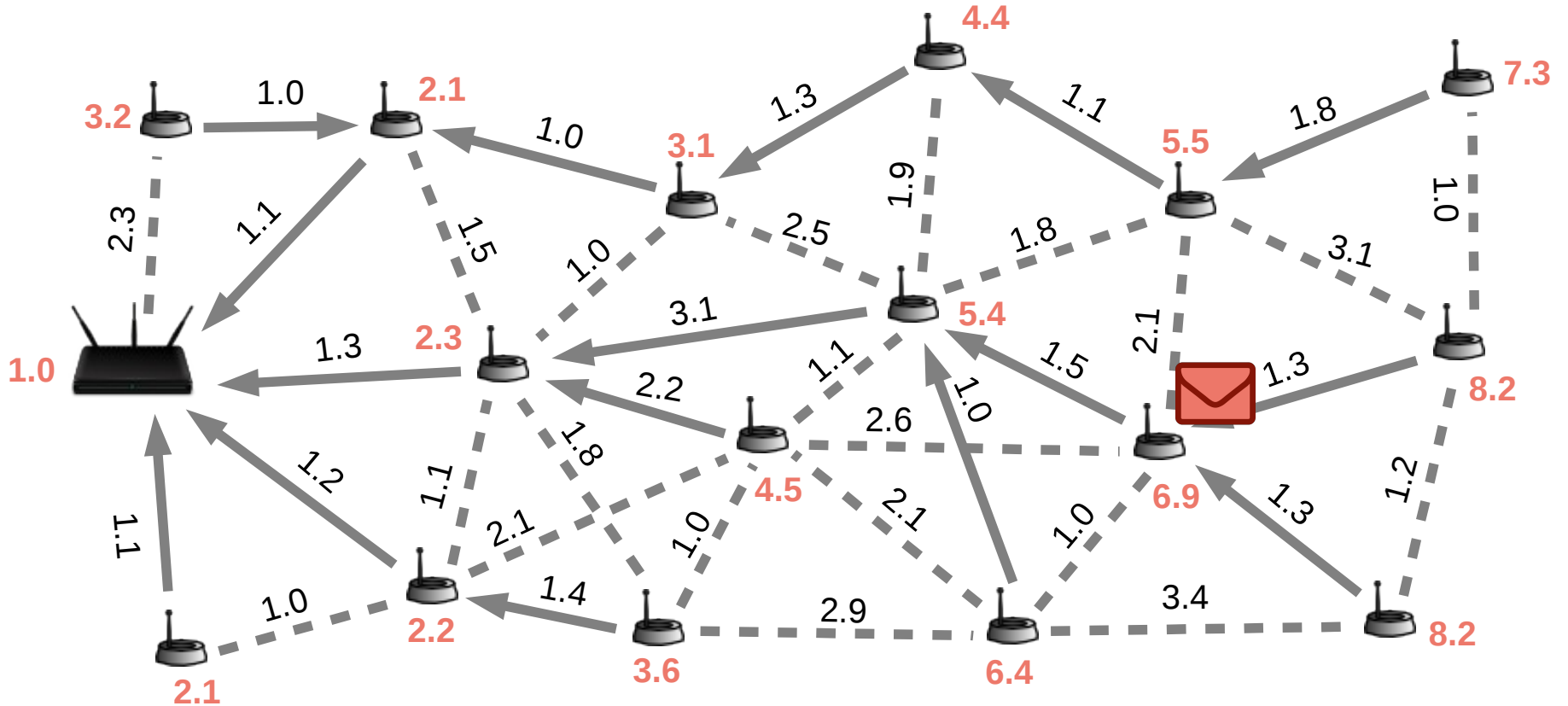
# DODAG



# DODAG

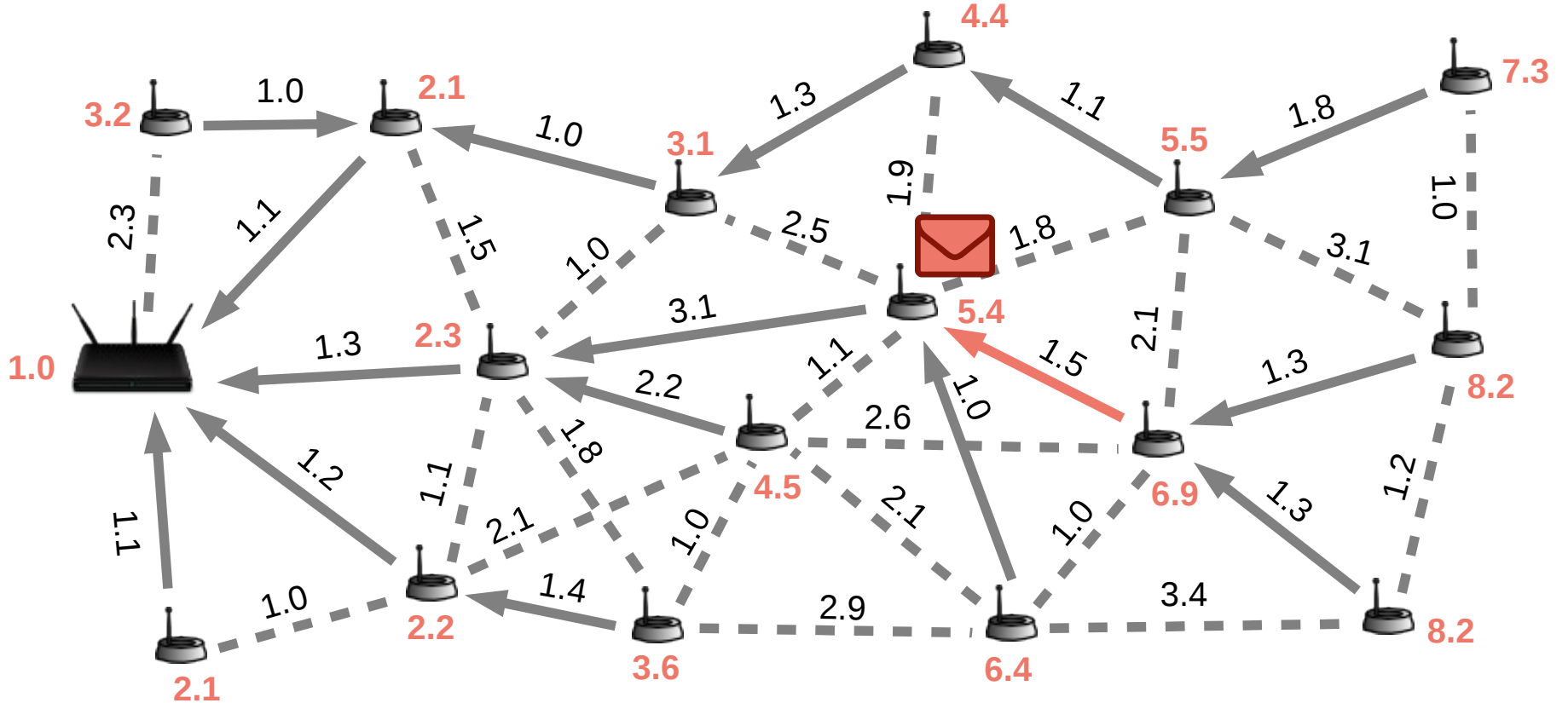


# DODAG

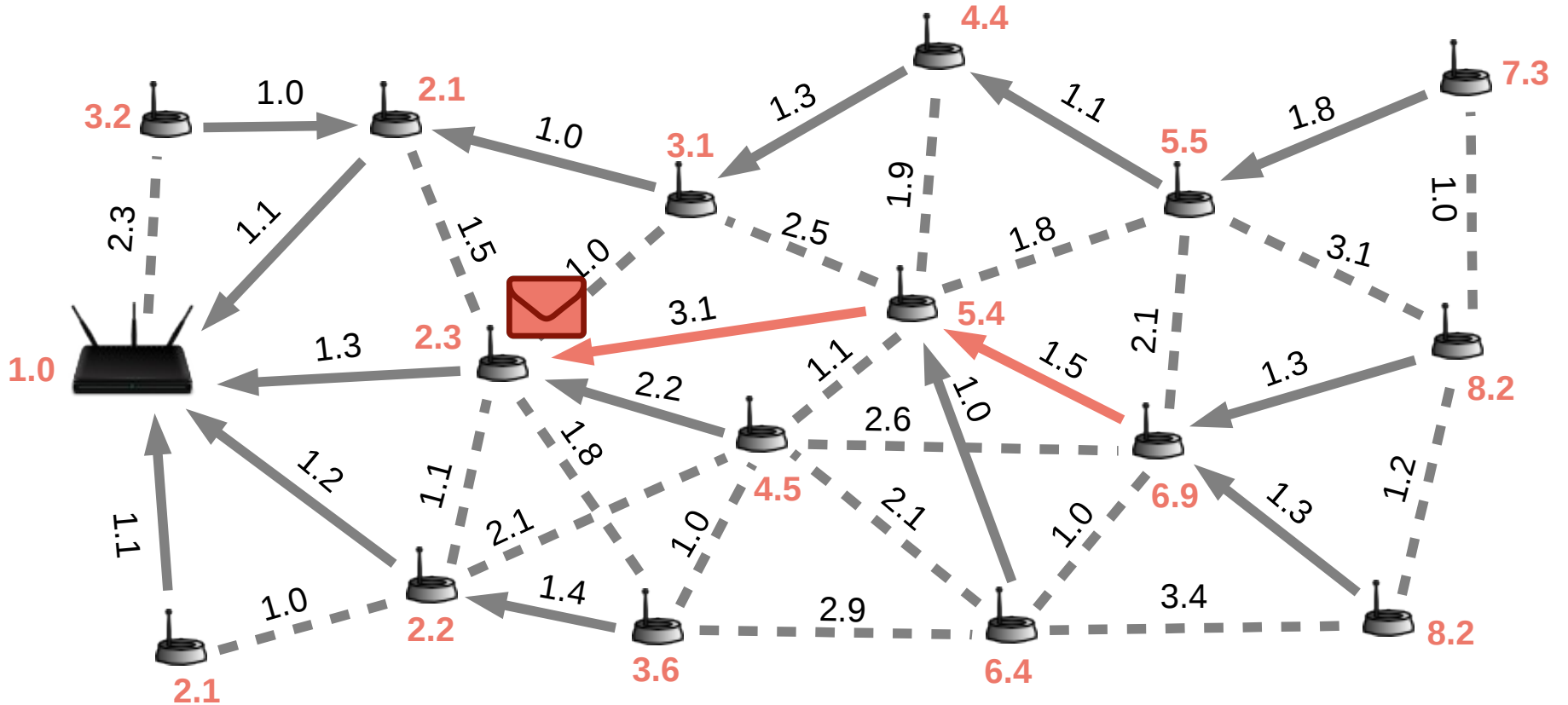




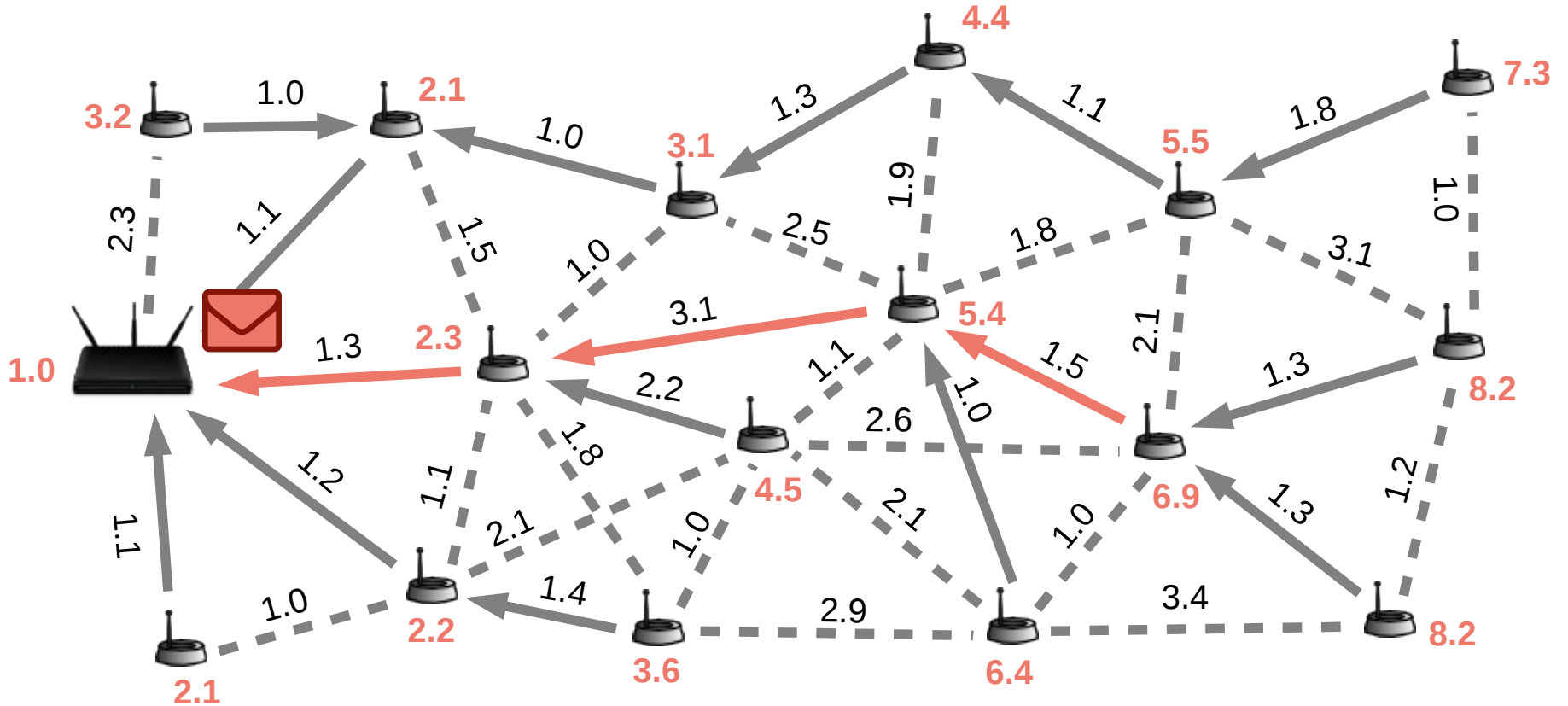
# DODAG



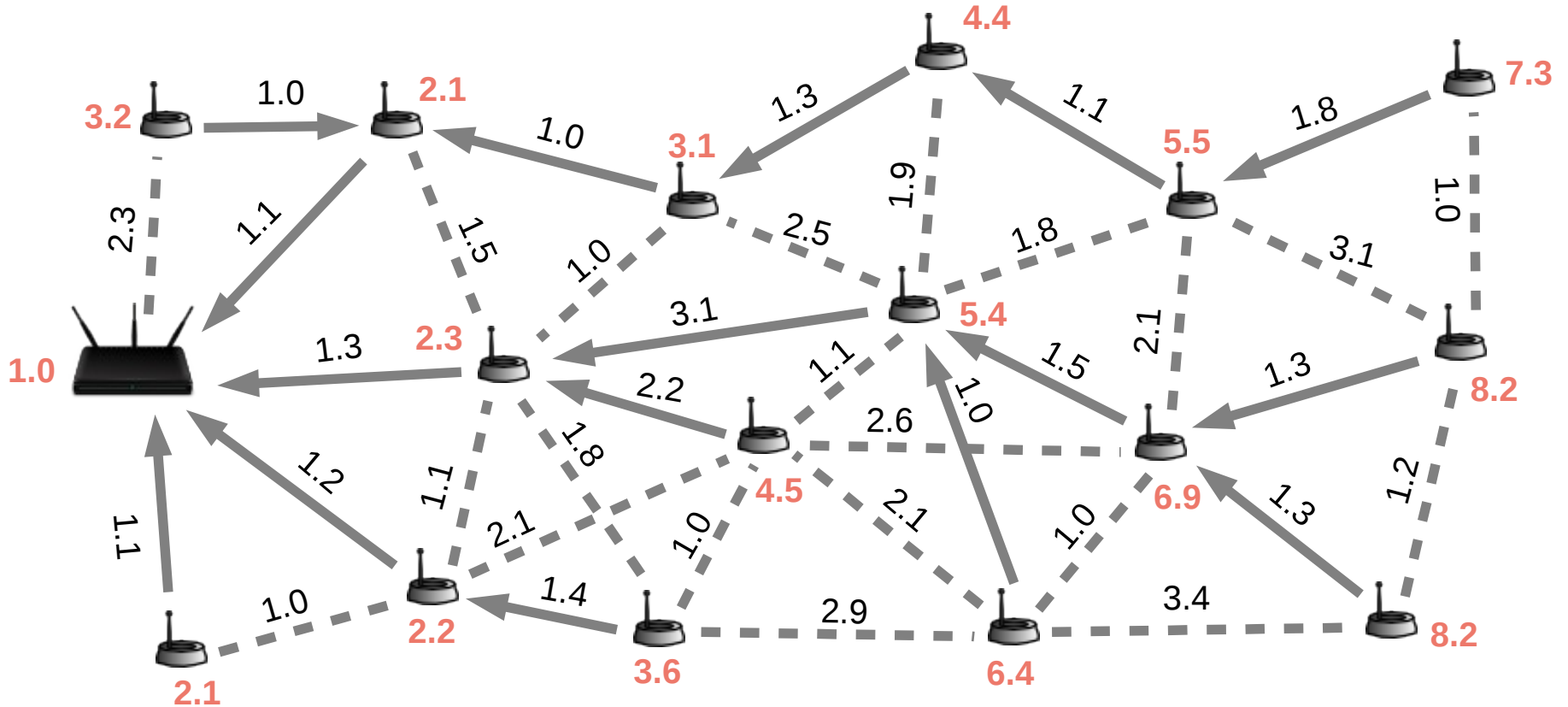
# DODAG



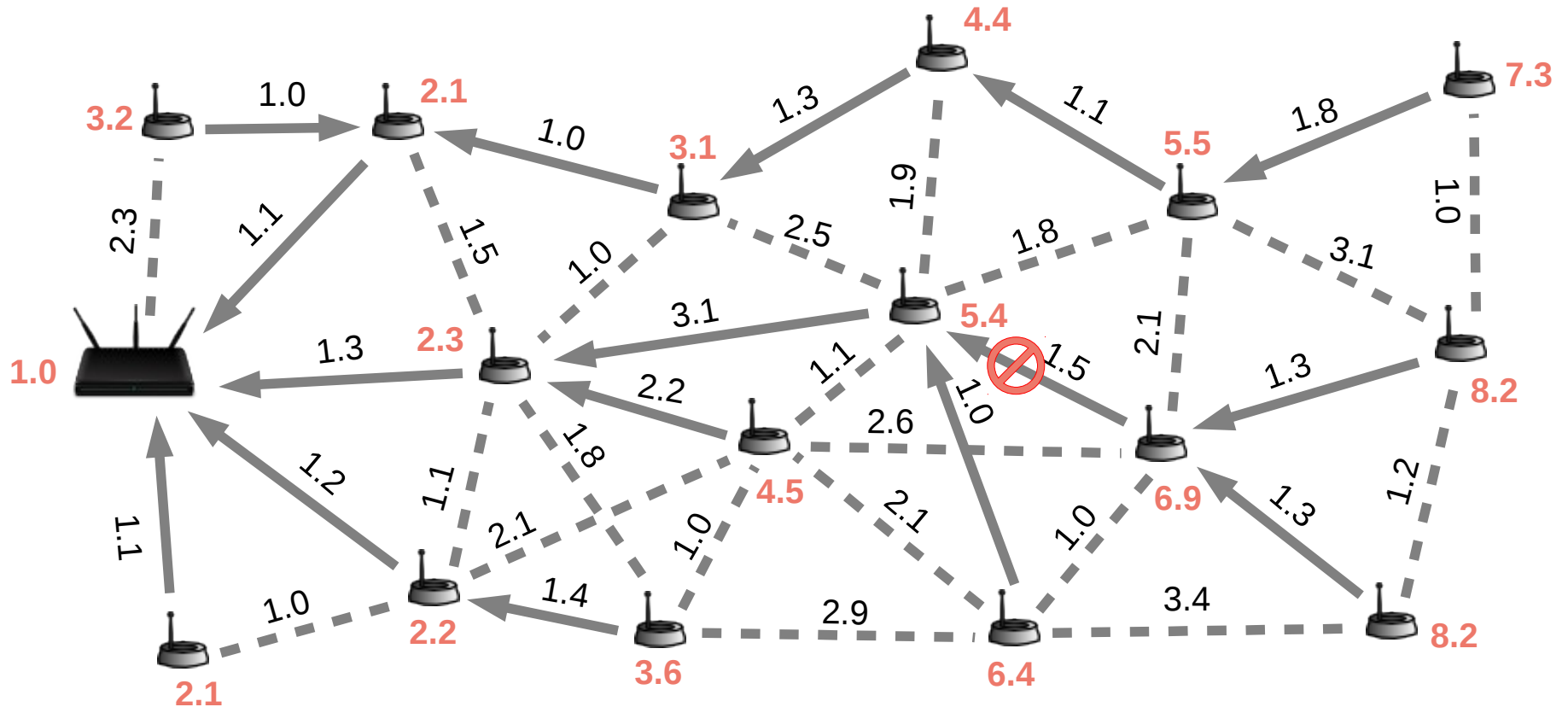
# DODAG



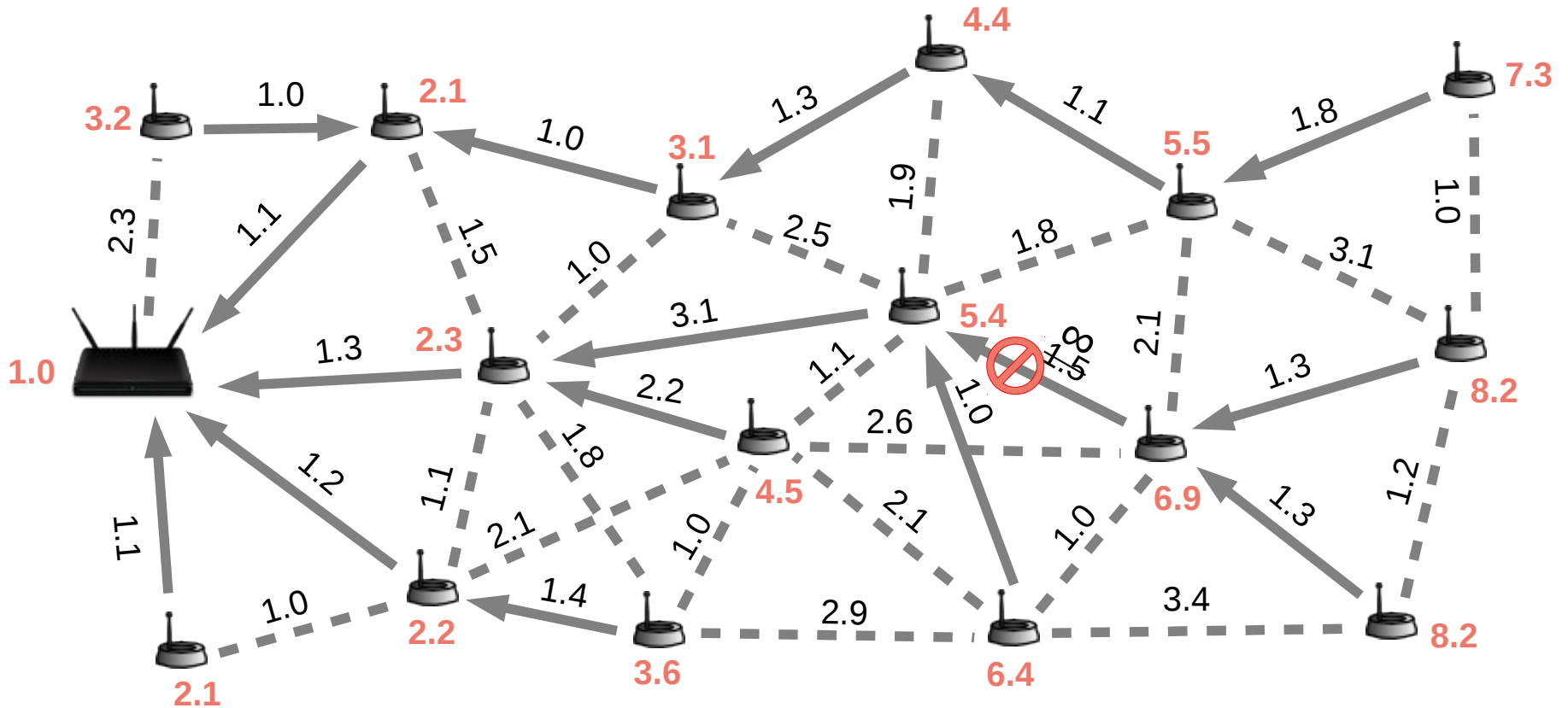
# DODAG under Failures



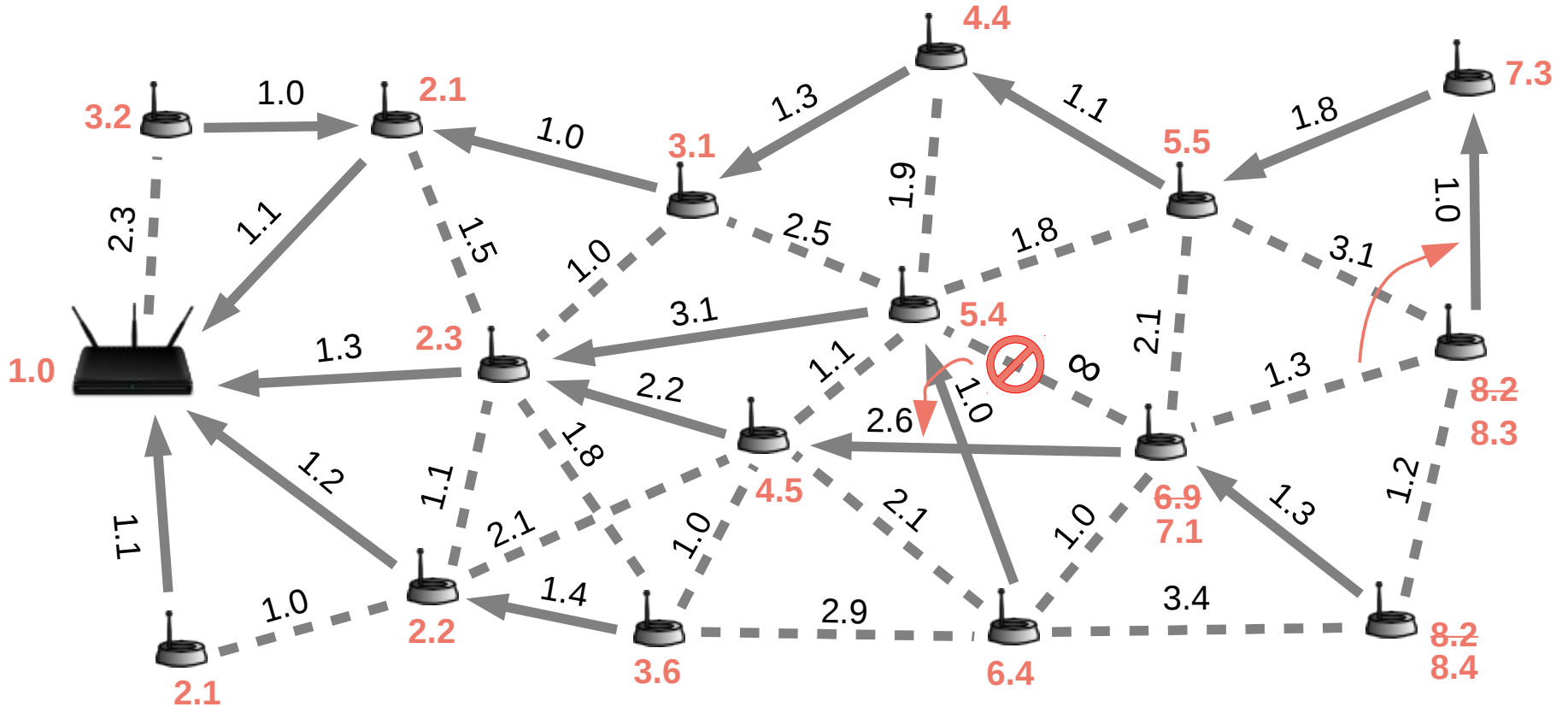
# DODAG under Failures



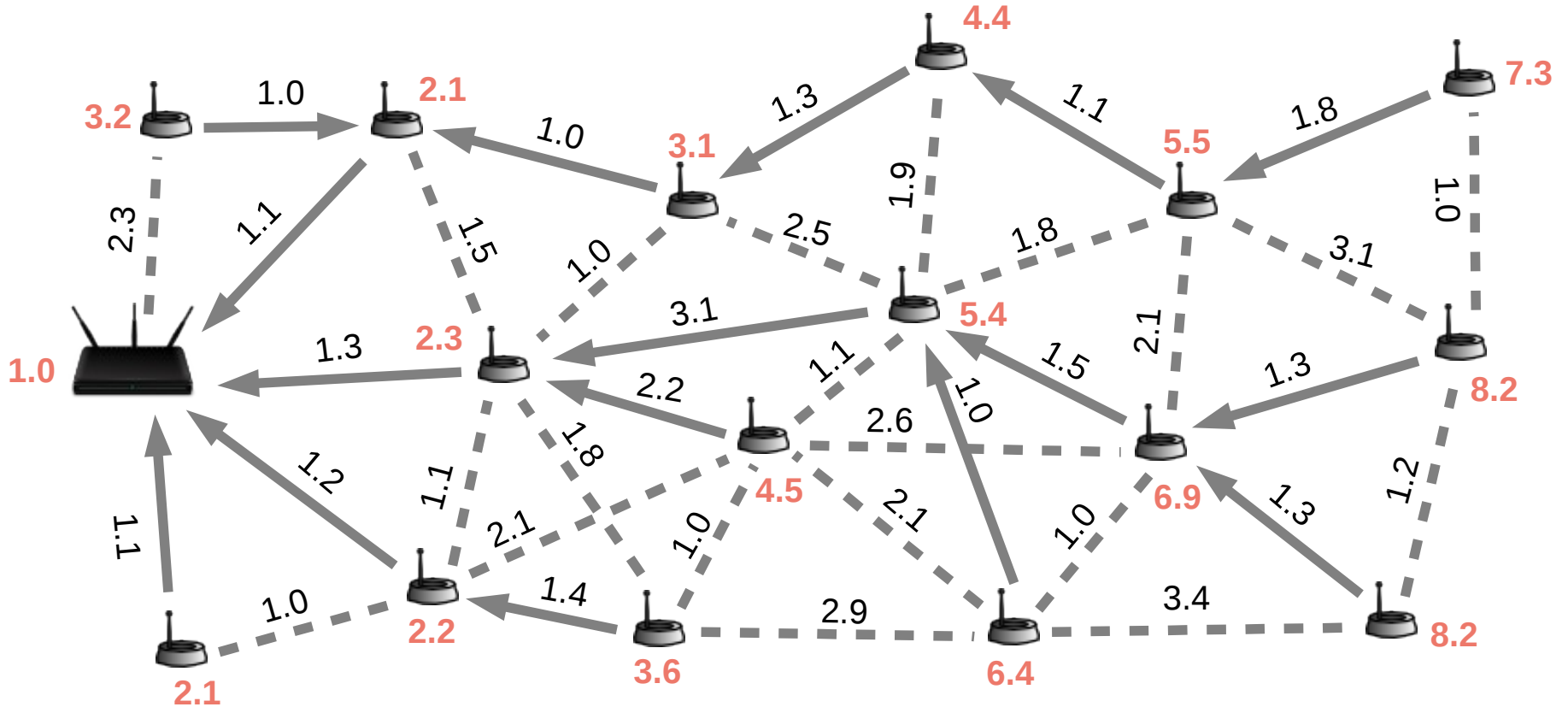
# DODAG under Failures



# DODAG under Failures

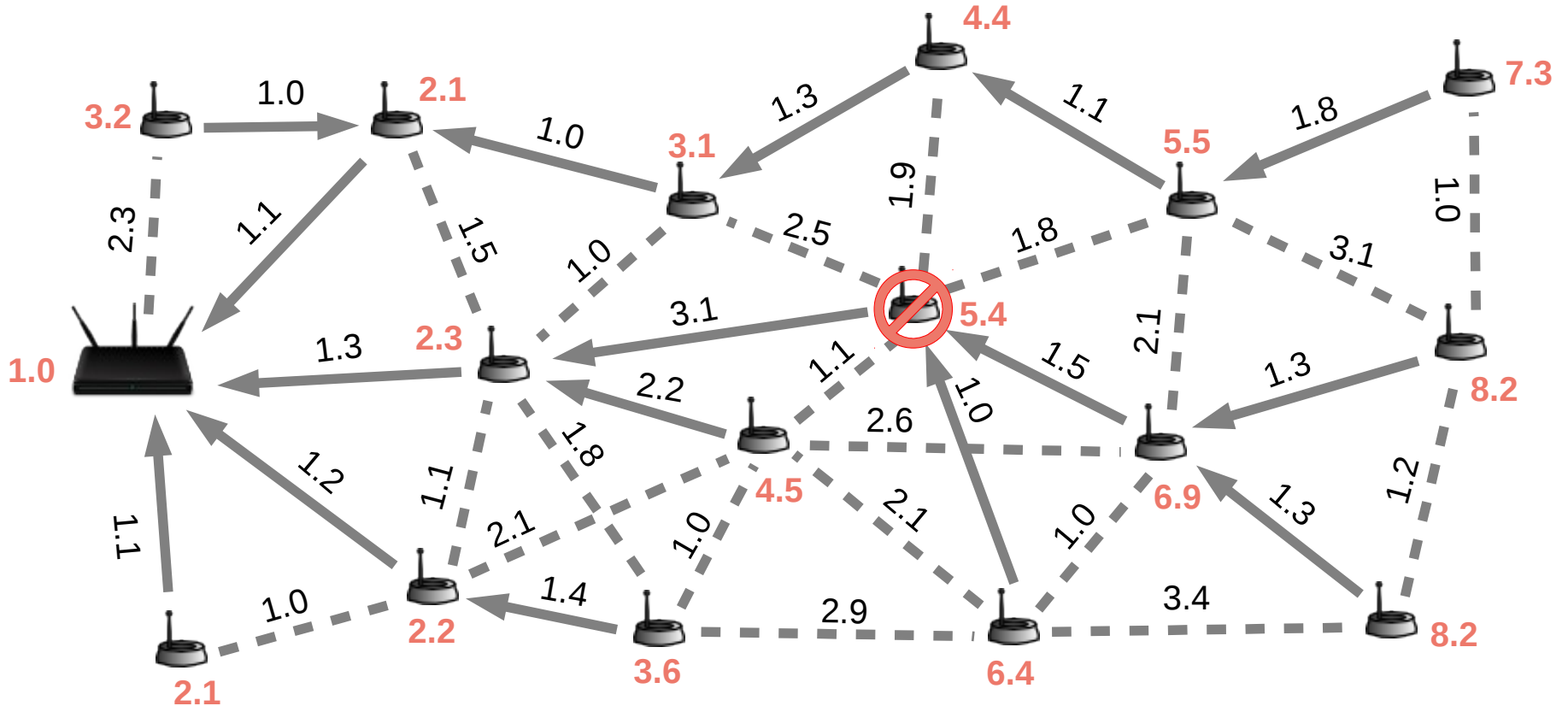


# DODAG under Failures

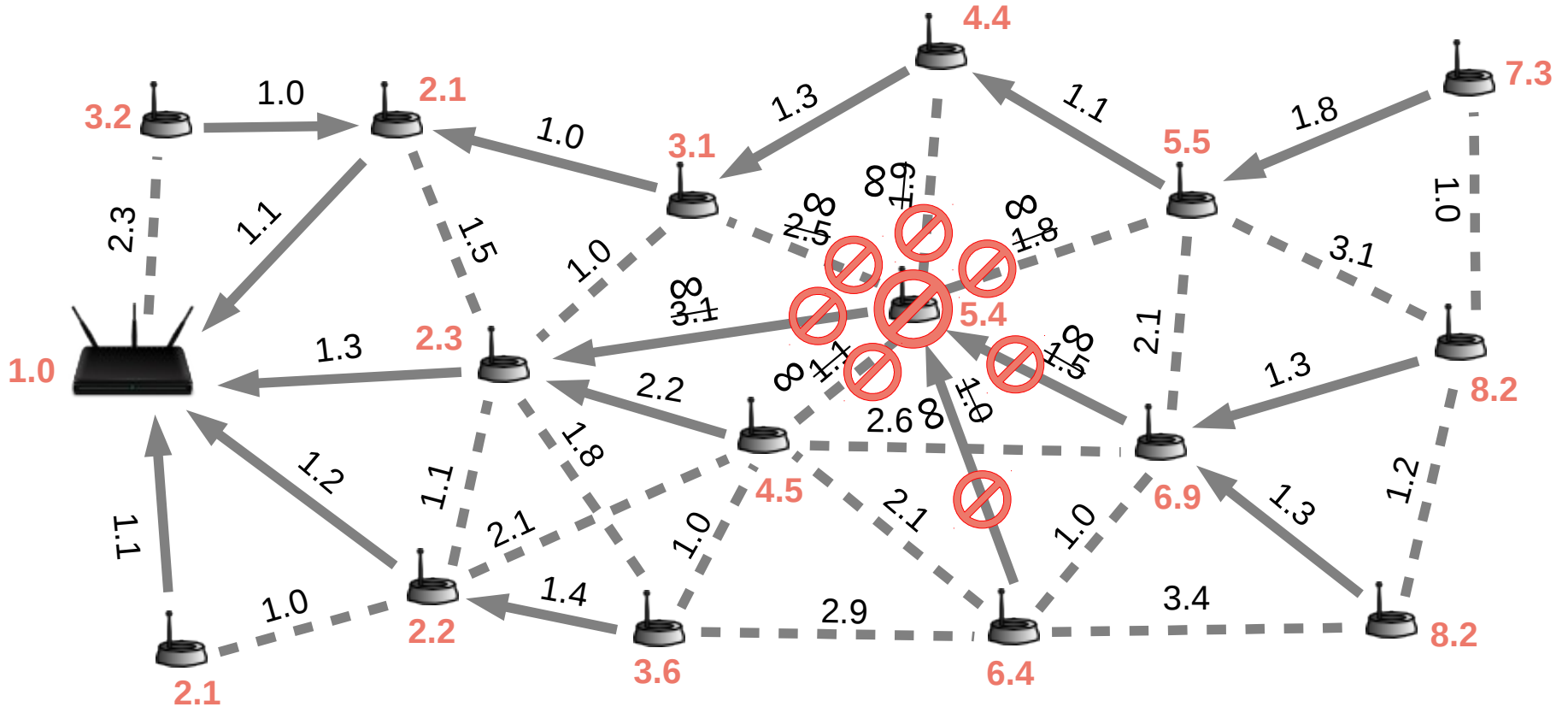




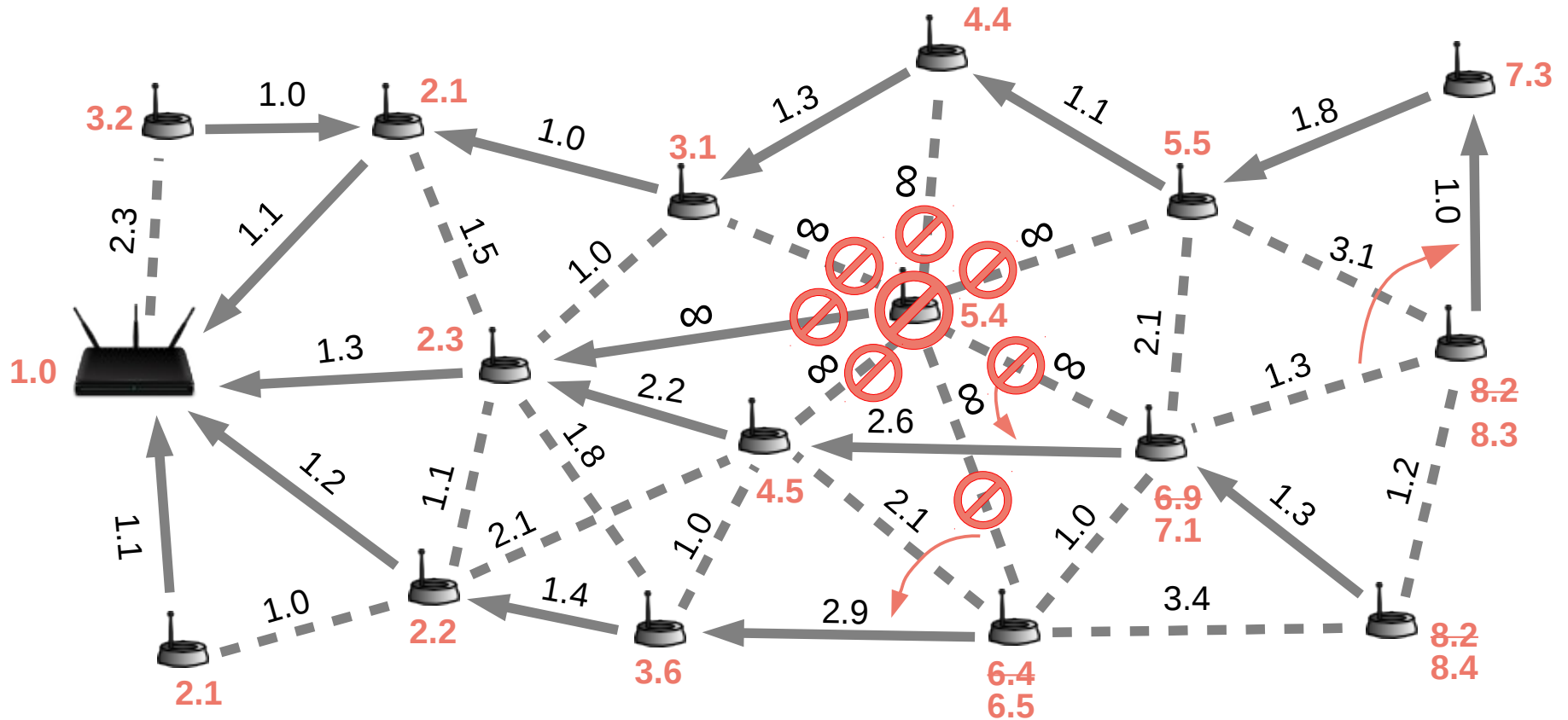
# DODAG under Failures



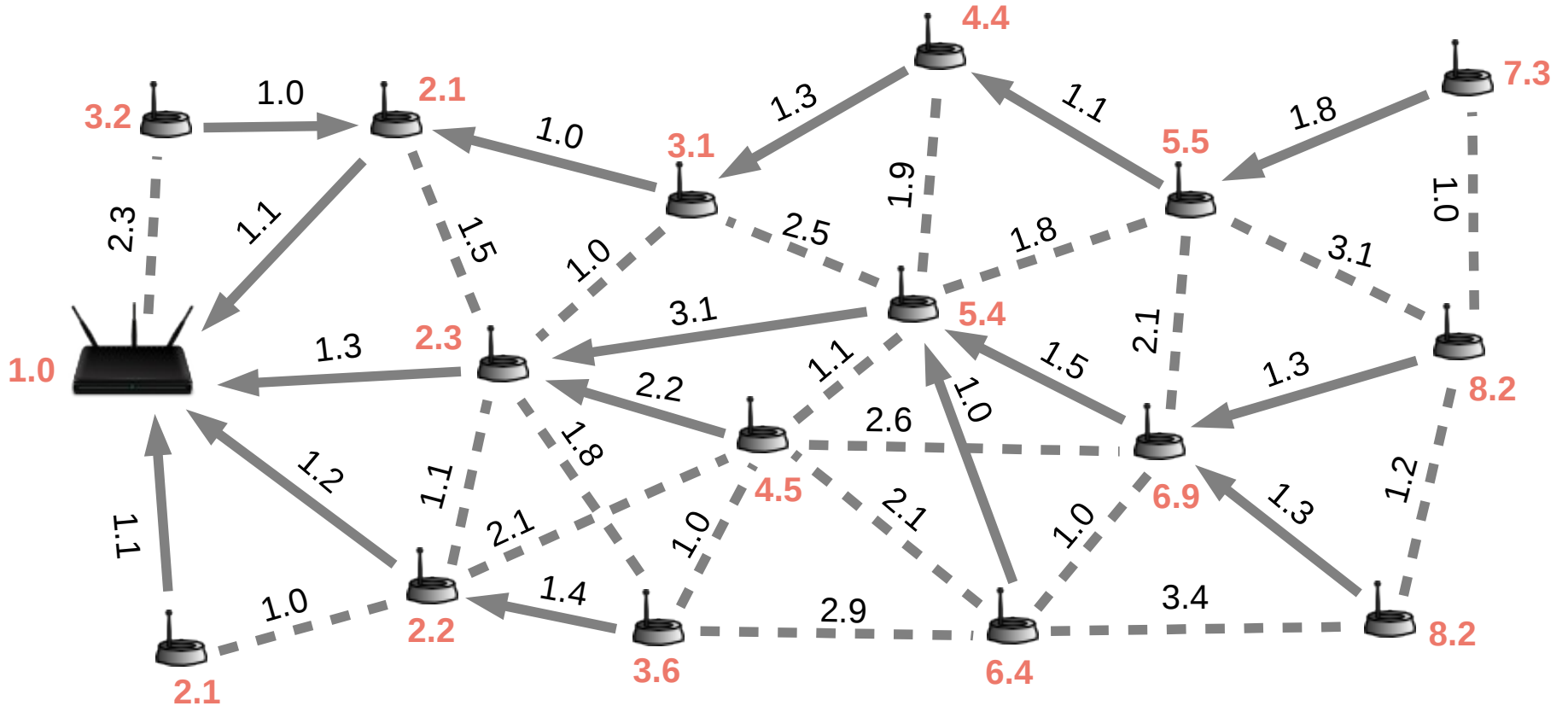
# DODAG under Failures



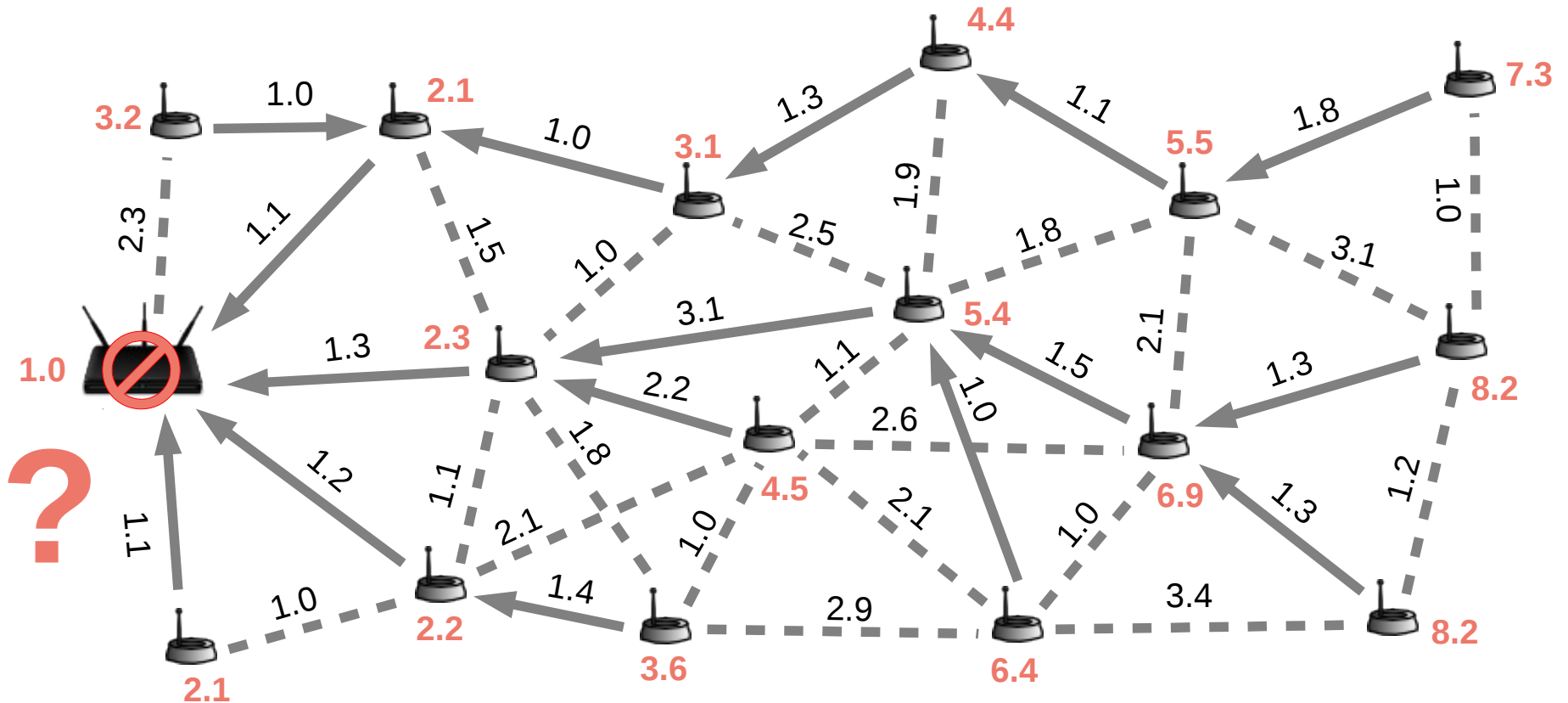
# DODAG under Failures



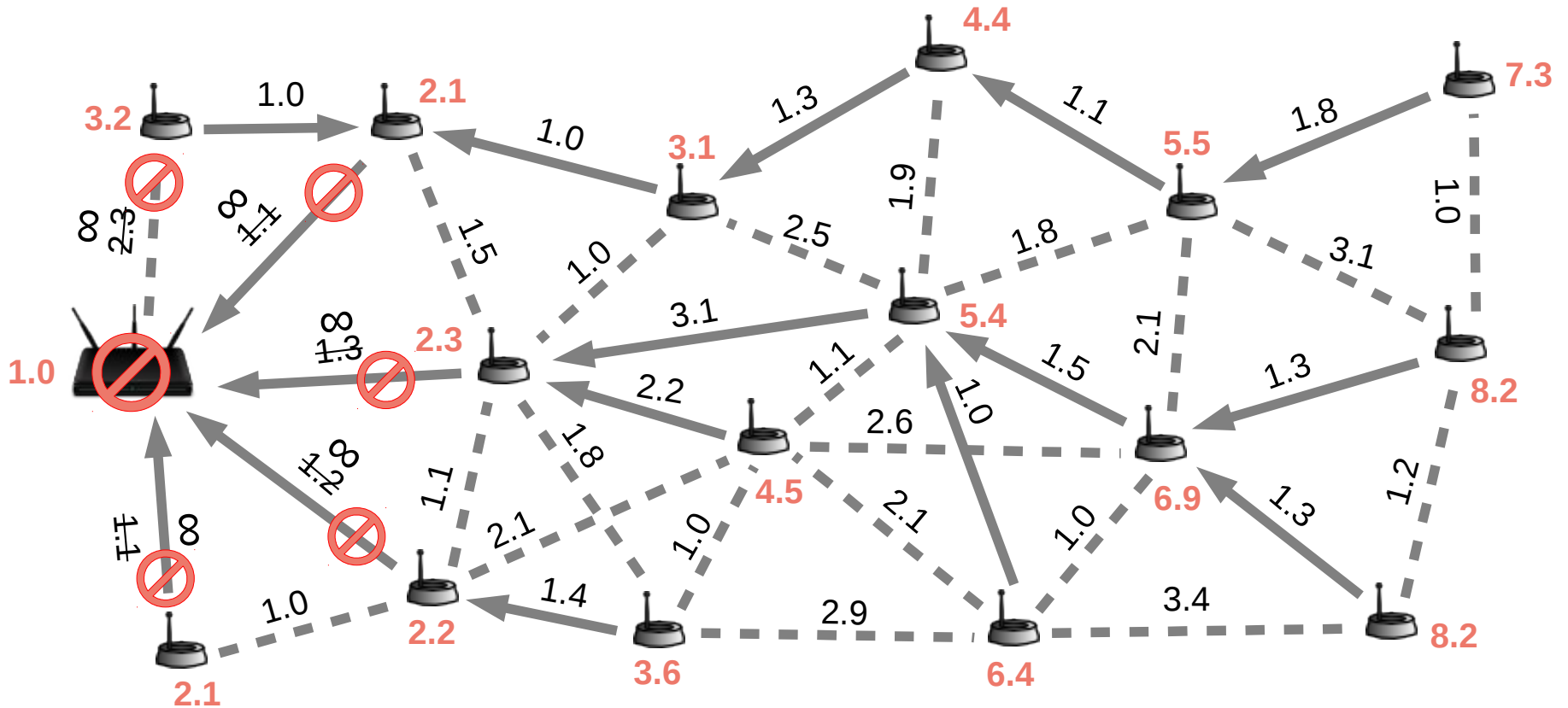
# DODAG under Failures



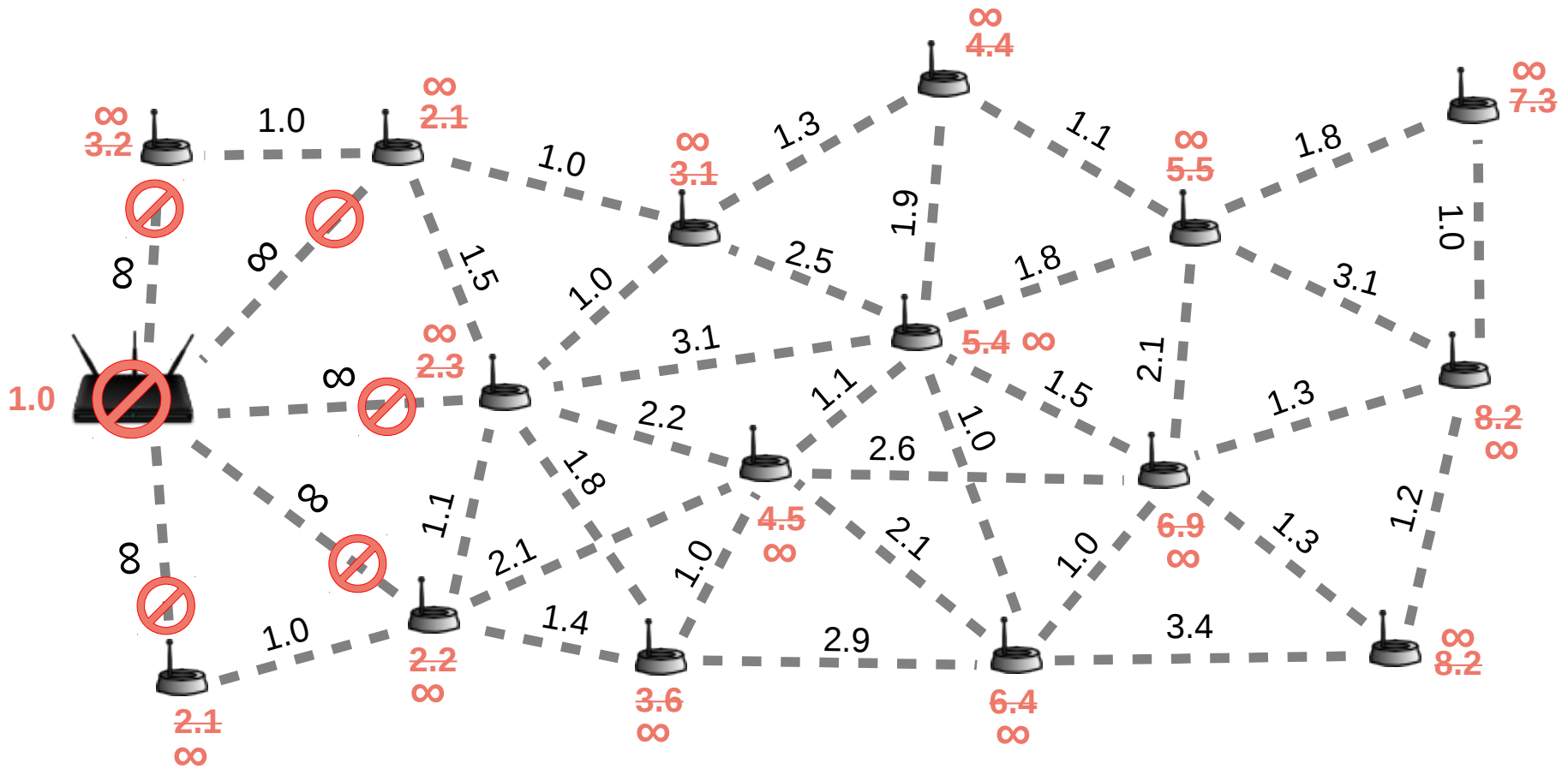
# DODAG under Failures



# DODAG under Failures



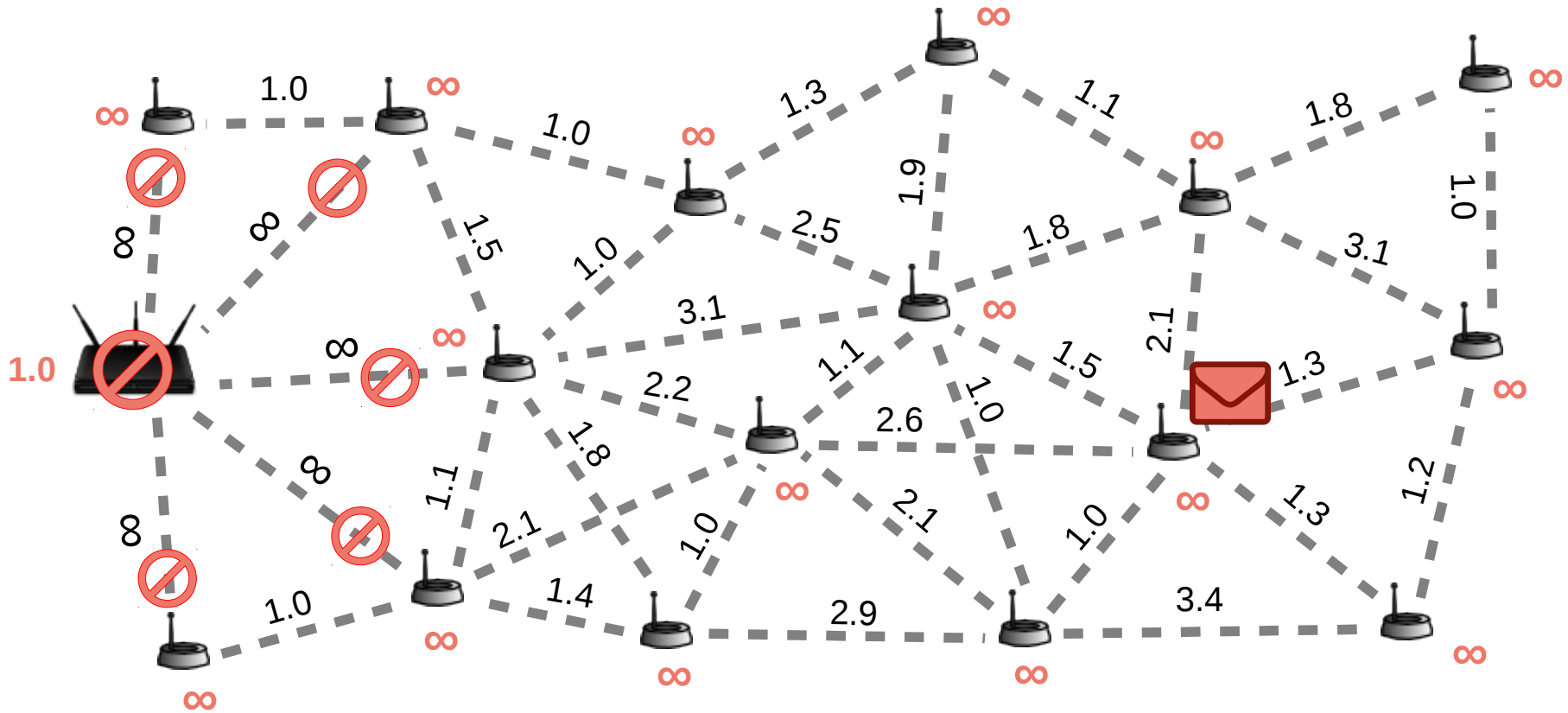
# DODAG under Failures



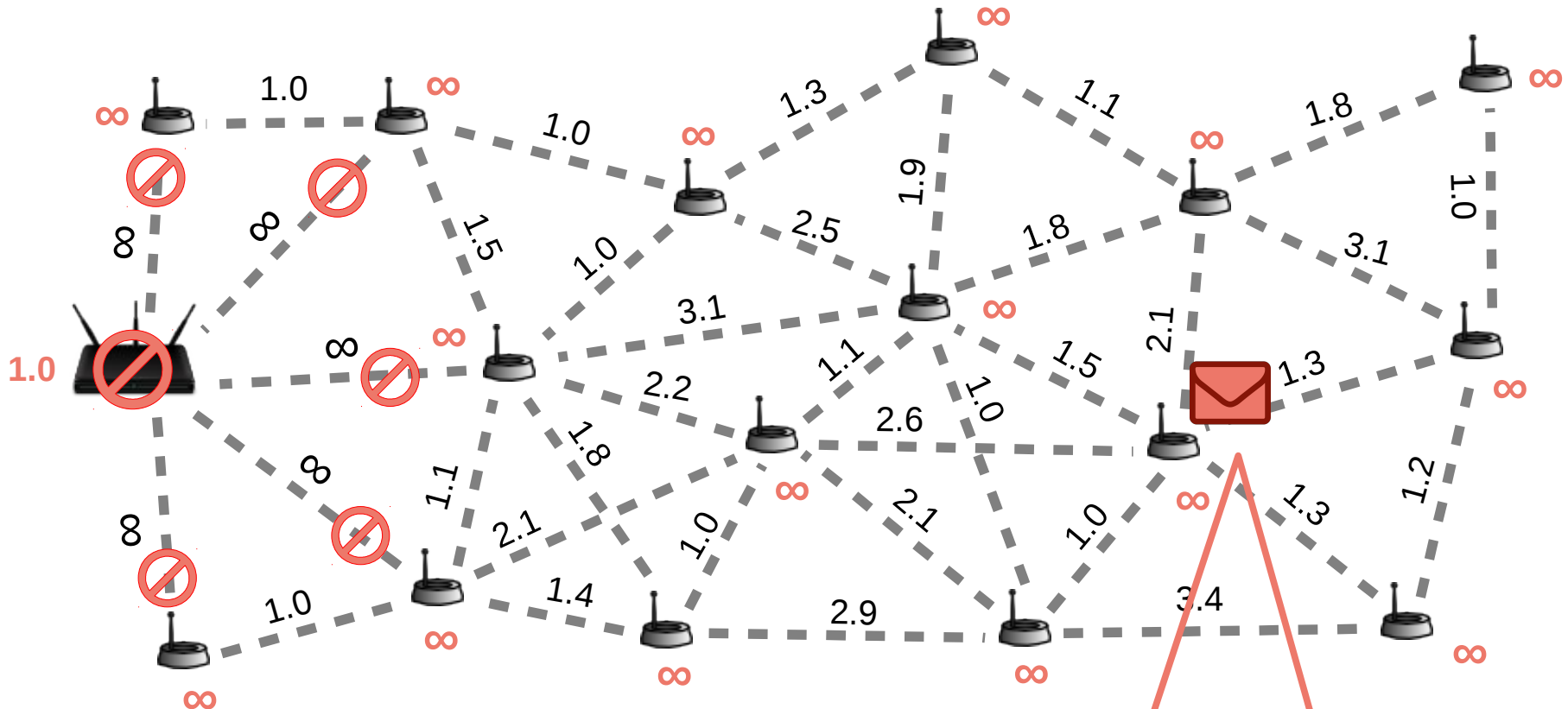




# DODAG under Failures



# DODAG under Failures



ICMPv6 No Route is signaled immediately to the higher layer (i.e., without any forwarding).

# Routing Protocols under Failures

Behavior of three representative approaches to route maintenance:

- Hierarchical routing protocol (IPSN 2009)
- Collection Tree Protocol, CTP (SenSys 2009)
- RPL (RFC 6550) – ContikiRPL & TinyRPL

# Routing Protocols under Failures

Behavior of three representative approaches to route maintenance:

- Hierarchical routing protocol (IPSN 2009)
- Collection Tree Protocol, CTP (SenSys 2009)
- RPL (RFC 6550) – ContikiRPL & TinyRPL

**The conclusion: Failure handling in these approaches leaves a lot of room for improvement in terms of traffic and latency.**

# Problems

# Problems

- Failed links are removed sequentially.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.



# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

- Coordinate failure detection.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

- Coordinate failure detection.
- Remove failed links in parallel.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

- Coordinate failure detection.
- Remove failed links in parallel.
- Do not probe all links.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

- Coordinate failure detection.
- Remove failed links in parallel.
- Do not probe all links.

A framework that **complements** instead of **replacing** existing route maintenance.

# Problems

- Failed links are removed sequentially.
- All failed links must be removed.
- Link removal is uncoordinated.

# Solutions

- Coordinate failure detection.
- Remove failed links in parallel.
- Do not probe all links.

A framework that **complements** instead of **replacing** existing route maintenance.

**RNFD** = **R**outing-layer **N**ode **F**ailure **D**etector

# RNFD

- Each node explicitly tracks the current condition of the root.

# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.



# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.
- Detecting the failure is largely the task of the root's neighbors.

# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.
- Detecting the failure is largely the task of the root's neighbors.



coordination

# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.
- Detecting the failure is largely the task of the root's neighbors.
- When a neighbor suspects the root to be down, it triggers other neighbors to help verify if the root is down by forwarding packets.



coordination

# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.
- Detecting the failure is largely the task of the root's neighbors.
- When a neighbor suspects the root to be down, it triggers other neighbors to help verify if the root is down by forwarding packets.

coordination

parallel link  
removal

# RNFD

- Each node explicitly tracks the current condition of the root.
- It synchronizes this knowledge with other nodes through control beacons.
  - Independent of the current DODAG shape.
- Detecting the failure is largely the task of the root's neighbors.
- When a neighbor suspects the root to be down, it triggers other neighbors to help verify if the root is down by forwarding packets.
- When sufficiently many (not necessarily all) neighbors suspect the root, all nodes globally consent that the root is down.

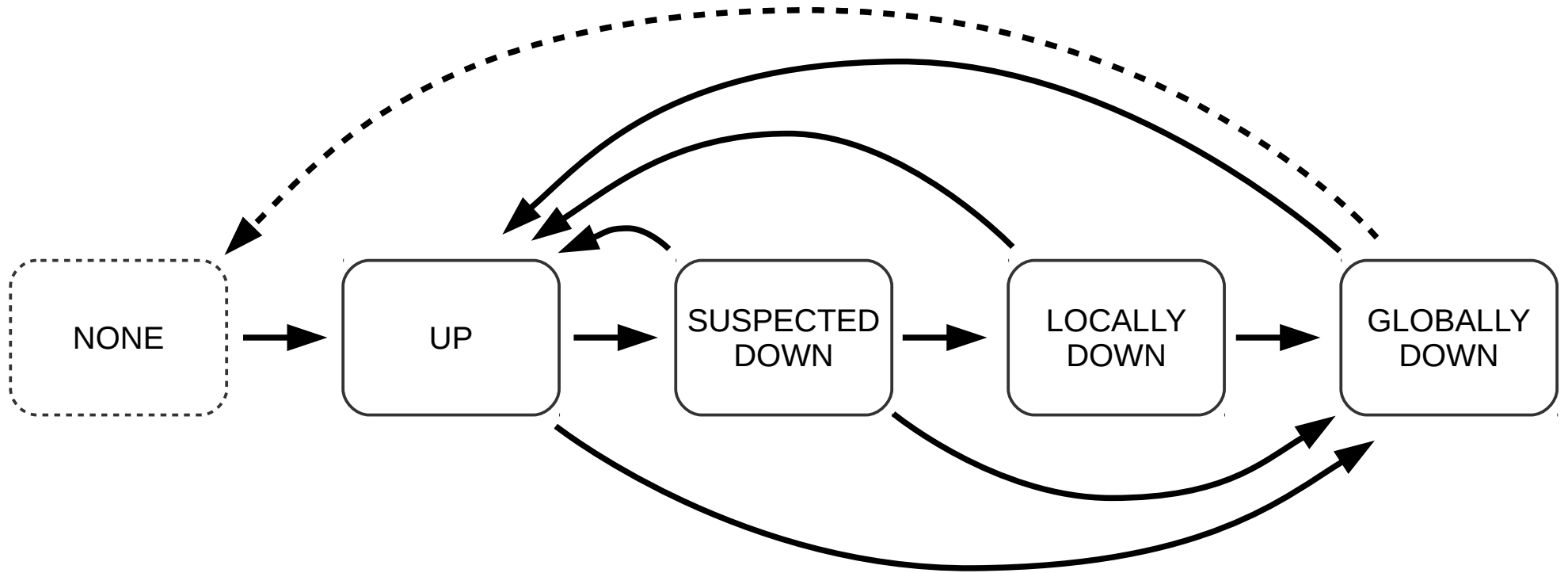
coordination

parallel link  
removal

# RNFD

- Each node explicitly tracks the current condition of the root.
  - It synchronizes this knowledge with other nodes through control beacons.
    - Independent of the current DODAG shape.
  - Detecting the failure is largely the task of the root's neighbors.
  - When a neighbor suspects the root to be down, it triggers other neighbors to help verify if the root is down by forwarding packets.
  - When sufficiently many (not necessarily all) neighbors suspect the root, all nodes globally consent that the root is down.
- 
- coordination
- parallel link removal
- partial link probing

# RNFD



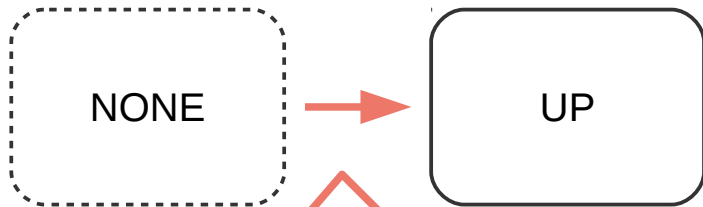
# RNFD



NONE

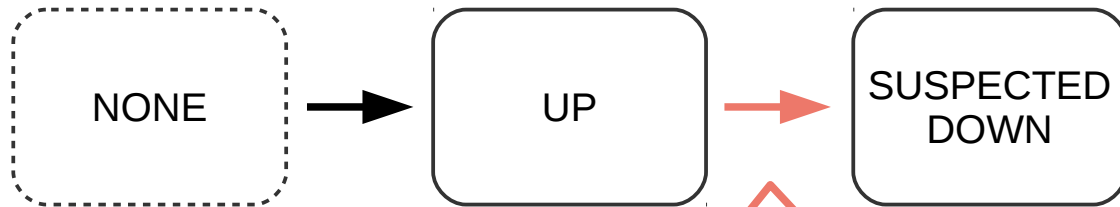


# RNFD



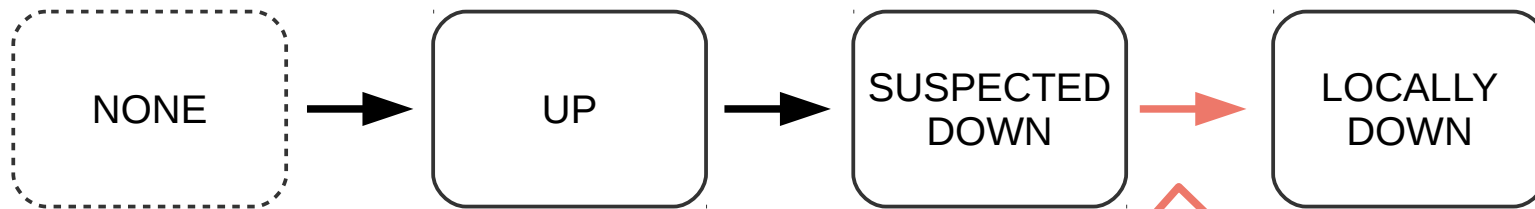
- The node learns about the DODAG and joins the DODAG.
- Its subsequent behavior depends on whether the node is the root's neighbor or not.

# RNFD



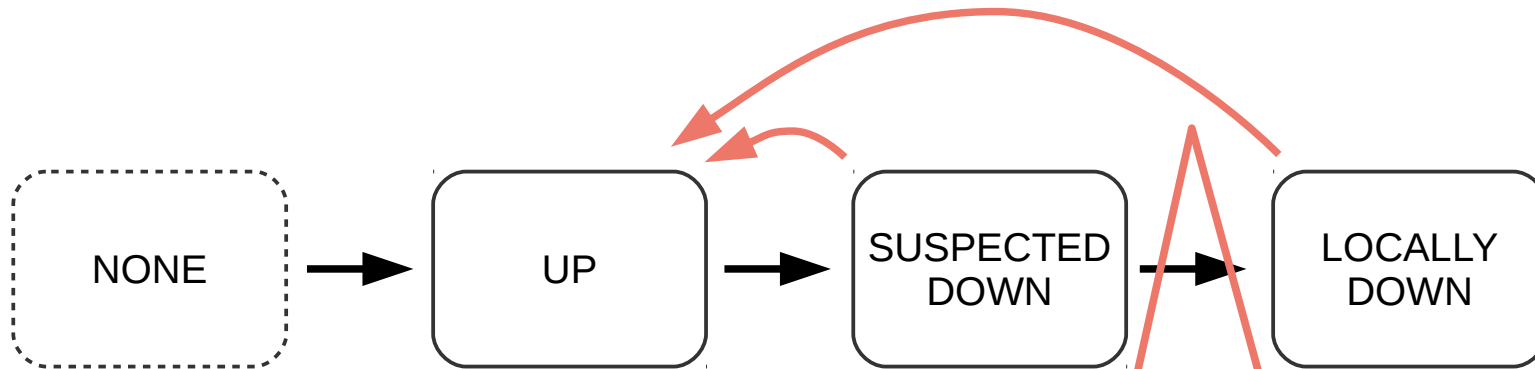
- The node – being the root's neighbor – starts suspecting the root to be down by observing:
  - the data plane or
  - the control plane.
- The neighbor may decide to verify its observations.

# RNFD



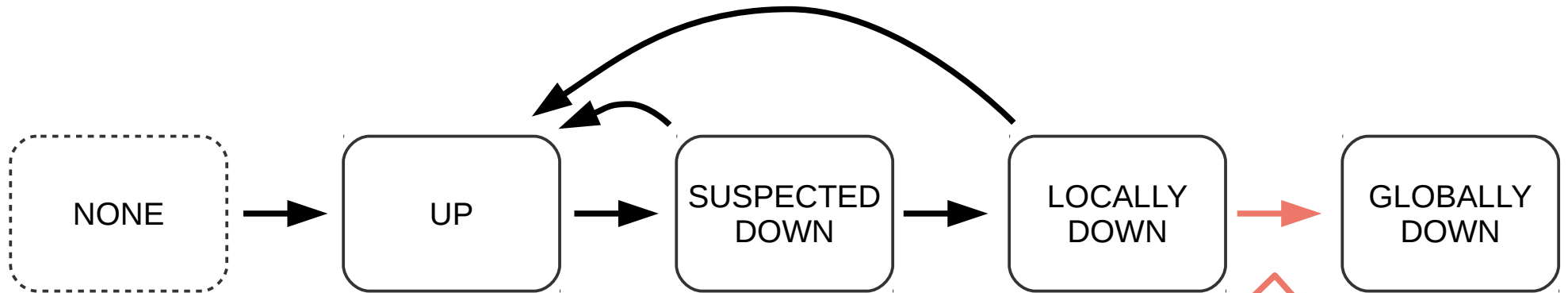
- The node – being the root's neighbor – is convinced that the root is down.
- It tries to trigger the root's other neighbors to check their links to the root by mechanisms at:
  - the data plane and
  - the control plane.

# RNFD



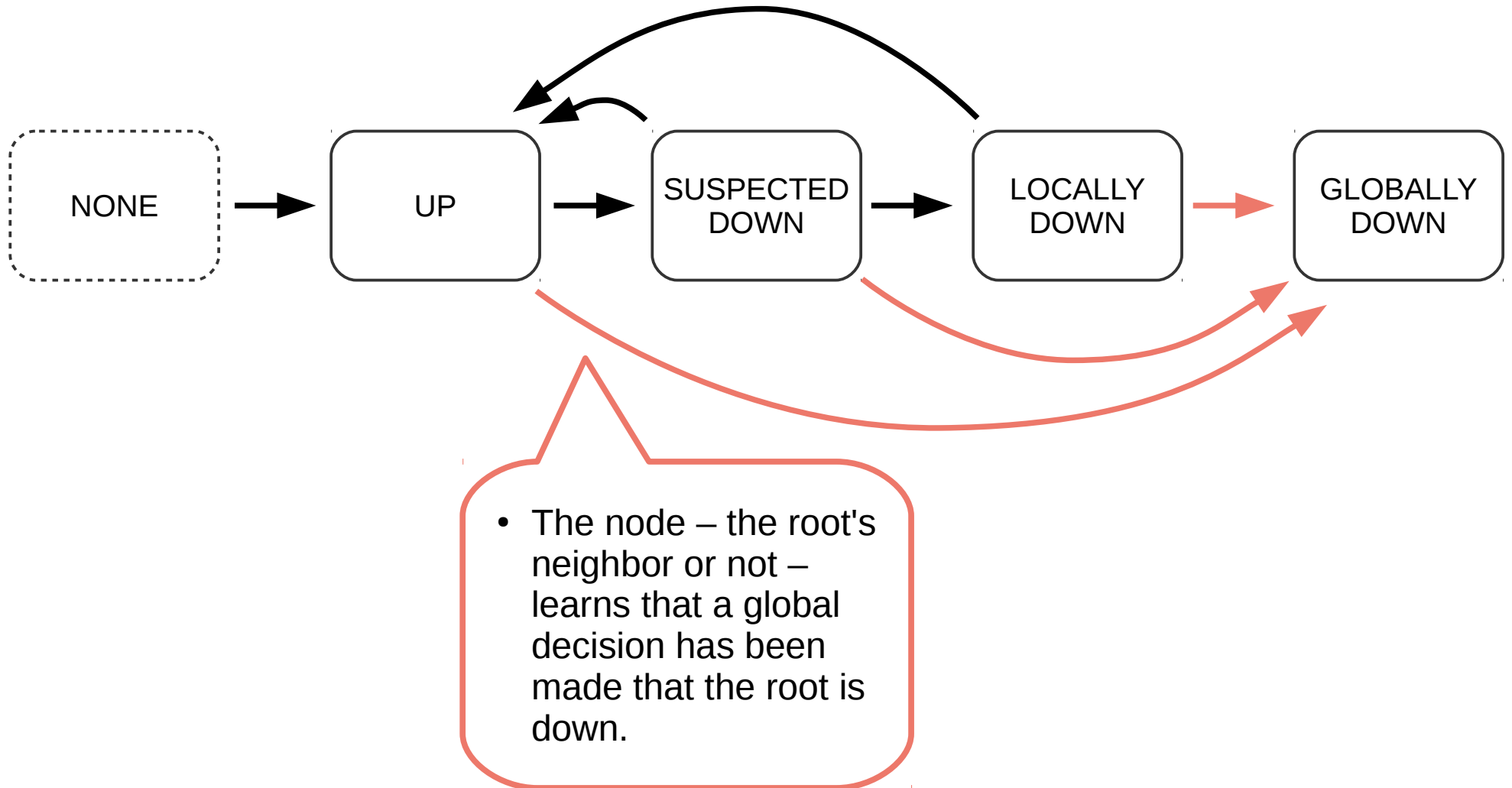
- The node – being the root's neighbor – learns at:
  - the data plane or
  - the control plane
- that its suspicion of the root is a mistake.

# RNFD

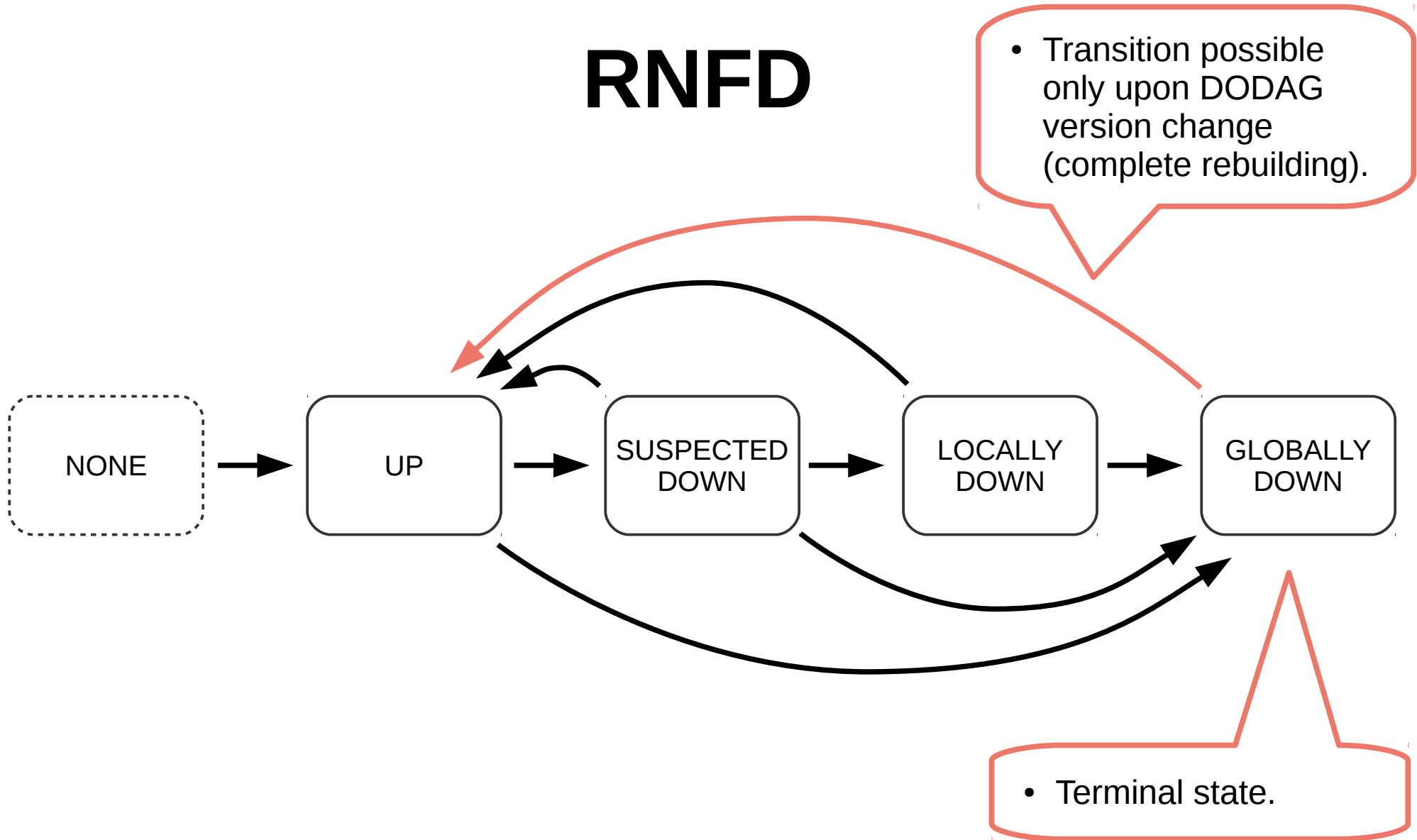


- The node – the root's neighbor or not – discovers that sufficiently many nodes consider the root as failed.
- It makes a global decision that the root is indeed down.

# RNFD

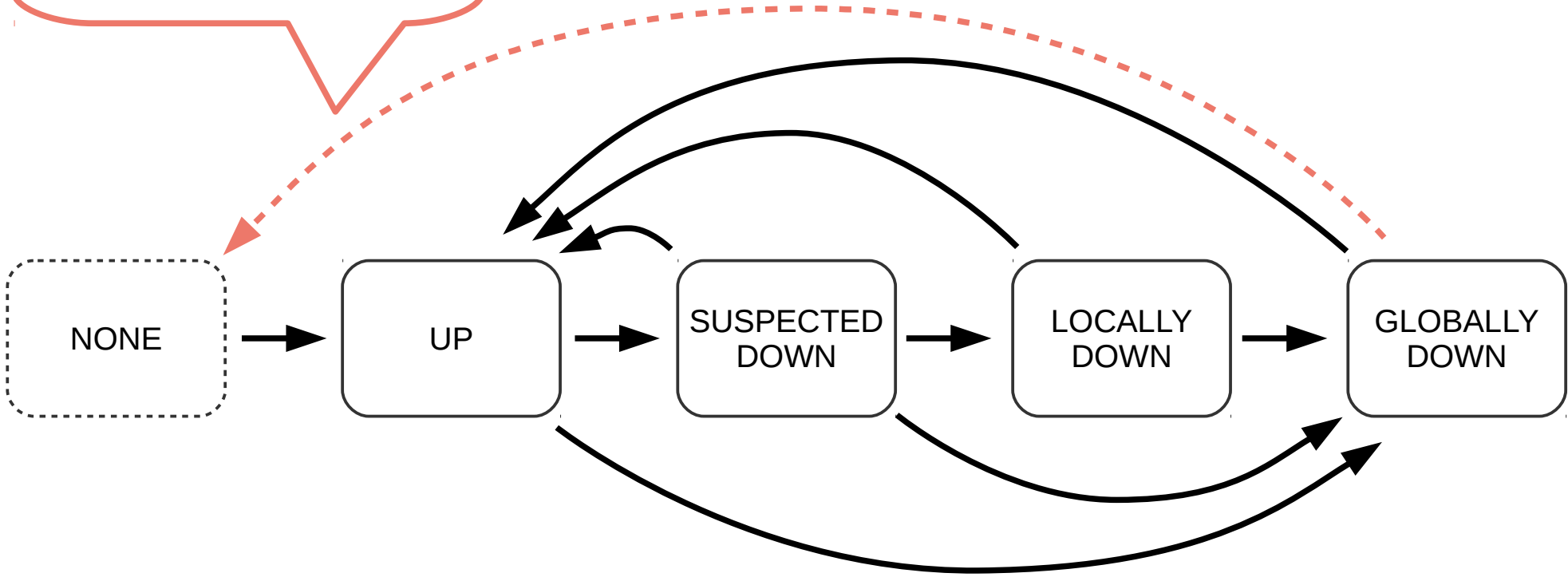


# RNFD



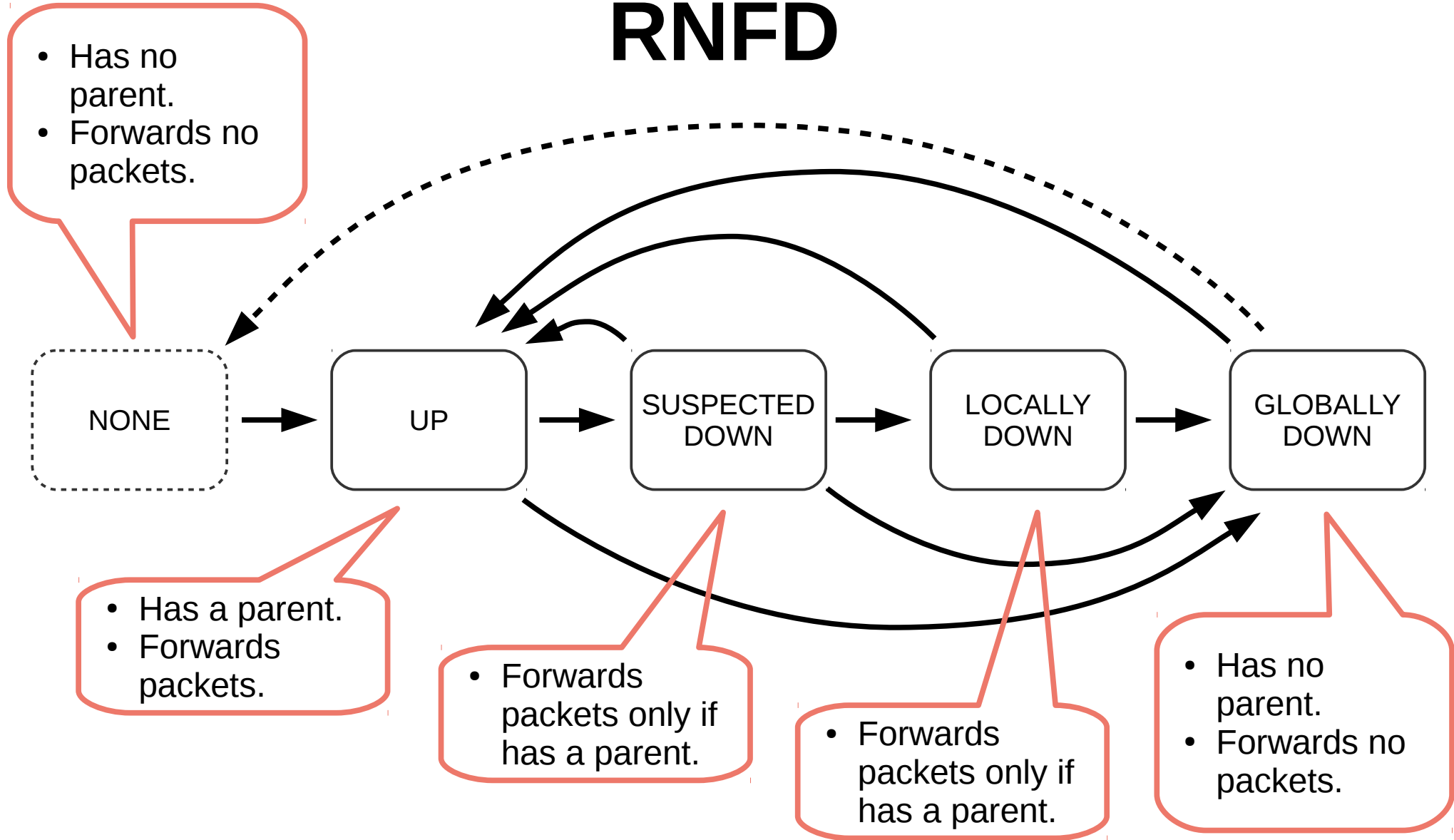
# RNFD

- Optional transition (garbage-collecting the dead DODAG).



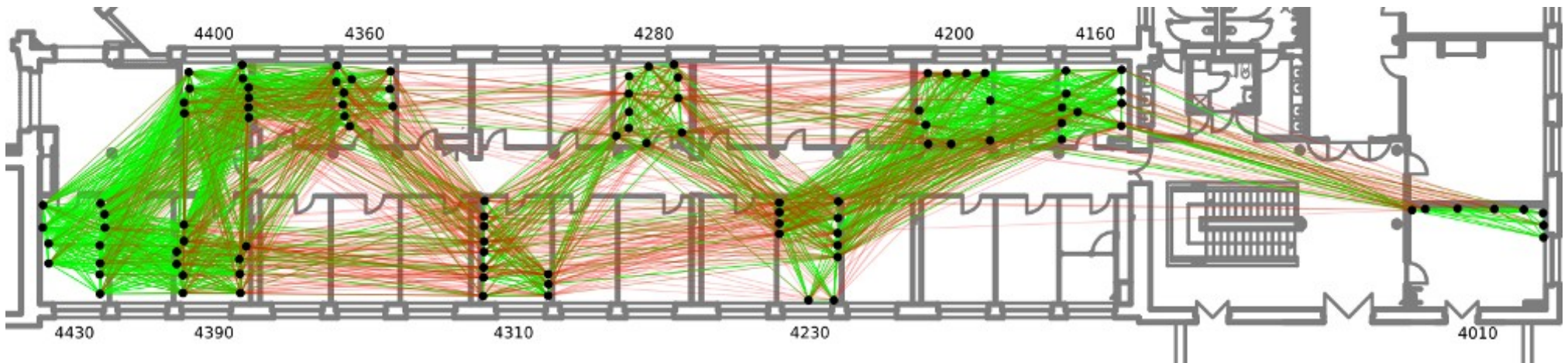


# RNFD



# Evaluation

- RNFD was implemented in TinyOS.
- It was evaluated in:
  - Low-level simulator, TOSSIM
    - various configurations
    - realistic gain values and noise traces
  - Two testbeds:
    - Testbed A: 32 nodes @ 2.4GHz
    - Testbed B: 102 nodes (76 active) @ 868MHz



# Evaluation

- RNFD detects failures one-two orders of magnitude faster than sole RPL.
- It requires less data traffic and control traffic to this end.
- It need not generate more traffic in a stable state.
- It introduces little information overhead.

# Conclusions and Future Work

- By:
  - coordinating failure detection,
  - removing dead links in parallel, and
  - not probing all links,

RNFD can significantly improve RPL's handling of DODAG roots' failures.

- It can also be used to handle failures of other “important” nodes.
- It works along (not instead of) RPL's existing route maintenance mechanisms: it should be easy to integrate.
- I would like to have the protocol standardized in an RFC as a companion protocol for RPL.

# Thank you

# Questions?

The presented research was supported by the National Center for Research and Development (NCBR) in Poland under grant no. LIDER/434/L-6/14/NCBR/2015.



The National Centre  
for Research and Development

