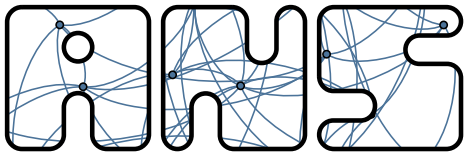


UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

The Optimized Link-State Protocol

Leonardo Maccari

December 1, 2017

Link State Routing[1]

The next slides describe how a link state routing protocol works, not necessarily a protocol for Wireless Mesh Networks (WMNs).

- link-state routers exchange messages to allow each router to learn the entire network topology.
- Based on this learned topology, each router is then able to compute its routing table by using a shortest path computation (with Dijkstra's algorithm).
- For link-state routing, a network is modelled as a directed weighted graph. Each router is a node, and the links between routers are the edges in the graph.
- A positive weight is associated to each directed edge and routers use the shortest path to reach each destination.

[1] http://cnp3book.info.ucl.ac.be/principles/linkstate.html

Link Weights

- In practice, different types of weight can be associated to each directed edge:
  - unit weight. If all links have a unit weight, shortest path routing prefers the paths with the least number of intermediate routers.
  - weight directly or inversely proportional to some link quality metric
- We will see two ways of defining the link weight, ETX and Airtime Metric

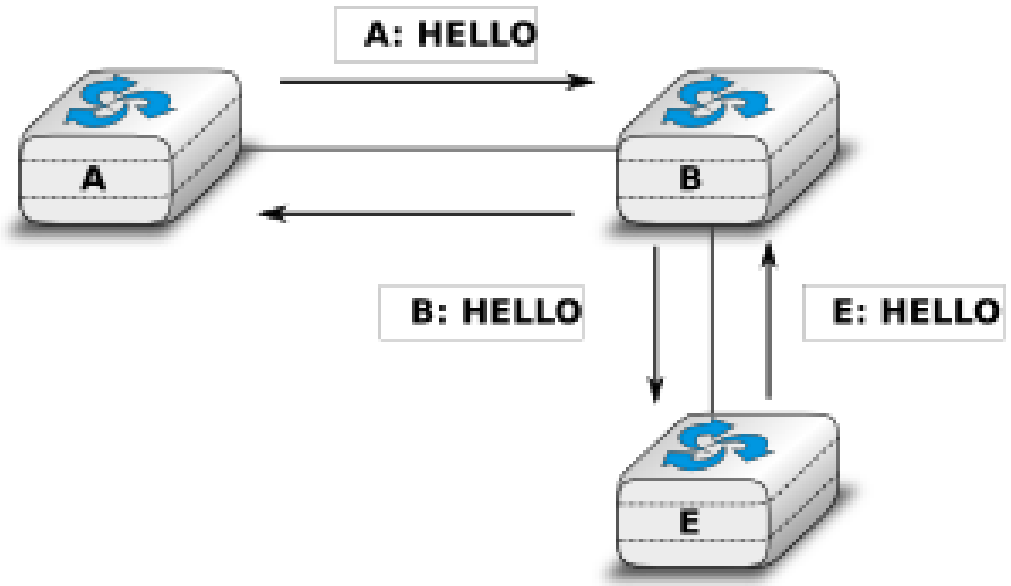
UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- When a link-state router boots, it first needs to discover to which routers it is directly connected.
- Each router picks for itself one unique address from the ones it is configured with.
- it sends a H every $t_{\mathtt{H}}$ seconds on all of its interfaces.
- This message contains the router's address.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Its neighbouring routers also send H messages, the router automatically discovers to which neighbours it is connected.
- H are only sent to neighbours who are directly connected to a router, a router never forwards an H
- H are also used to detect link and router failures.
- A link is considered to have failed if no H has been received from the neighbouring router for a period of k $\times$ t_{H} seconds.

HELLO Exchange



UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Once a router has discovered its neighbours, it must reliably distribute its local links to all routers.
- Each router builds a link-state packet (LSP) containing the following information :
  - LSP.Router: identification (address) of the sender of the LSP
  - LSP.age: age or remaining lifetime of the LSP
  - LSP.seq: sequence number of the LSP (incremented at each LSP)
  - LSP.Links[]: links advertised in the LSP.
    - LSP.Links[i].Id: identification of the neighbour
    - LSP.Links[i].cost: cost of the link

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- These LSPs must be reliably distributed before the routers have a working routing table.
- A Flooding algorithm is used to efficiently distribute the LSPs of all routers.
- Each router that implements flooding maintains a link state database (LSDB) containing the most recent LSP received by each other router.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

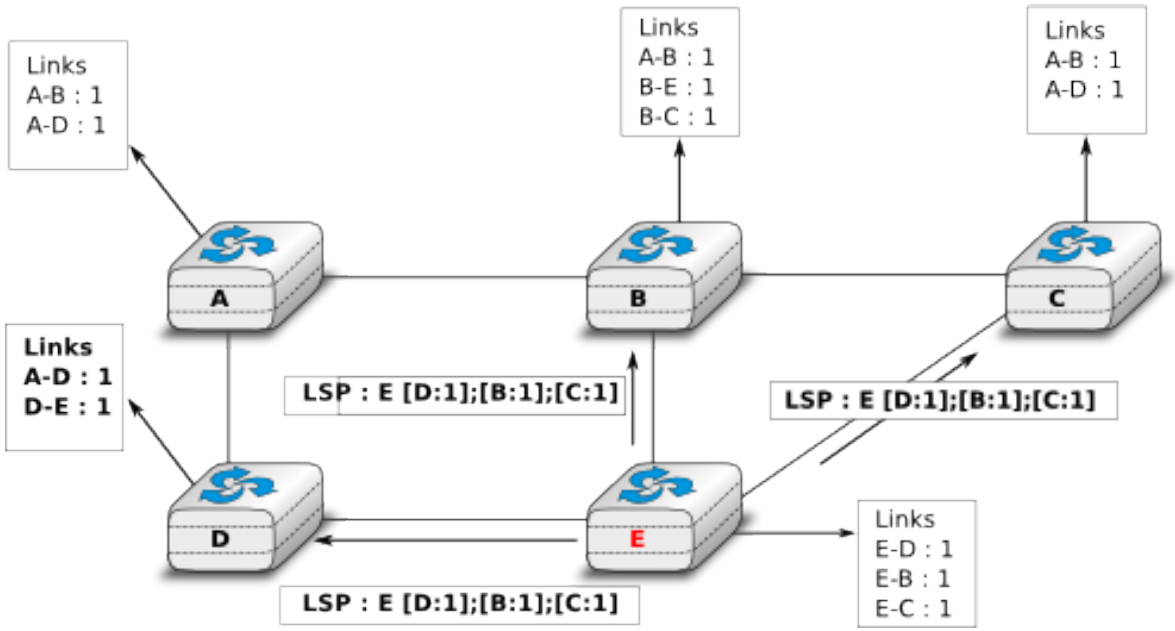
- When a router receives an LSP, it first verifies whether this LSP is already stored inside its LSDB using the sequence number.
- If so, the router has already distributed the LSP earlier and it does not need to forward it.
- Otherwise, the router forwards the LSP on all links except the link over which the LSP was received.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

```
# links is the set of all links on the router
# Router R LSP arrival on link l
if newer(LSP, LSDB(LSP.Router)) :
  LSDB.add(LSP)
  for i in links :
    if i!=l :
      send(LSP,i)
else:
 # LSP has already been flooded
 pass
```

H flooding

- E learns by H its neighbors D, B and C.
- E sends LSP with seq 0 containing links E->D, E->B and E->C.
- E sends LSP on all links
- routers D, B and C insert the LSP in their LSDB and forward it over their other links.



UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- To ensure that all routers receive all LSPs, even when there are transmissions errors, link state routing protocols use reliable flooding.
- routers use acknowledgements and if necessary retransmissions to ensure that all link state packets are successfully transferred to all neighbouring routers.

Failure Detection

- When a link fails, the two routers attached to the link detect the failure by the lack of H messages received in the last k $\times$ t_H seconds.
- Once a router has detected a local link failure, it generates and floods a new LSP that no longer contains the failed link and the new LSP replaces the previous LSP in the network.
- As the two routers attached to a link do not detect this failure exactly at the same time, some links may be announced in only one direction.
- uni-directional links are not considered for routing

Criticalities

Using link-state routing in WMN requires modifications due to critical issues:

- Wireless links are not necessarily symmetric

Criticalities

Using link-state routing in WMN requires modifications due to critical issues:

- Wireless links are not necessarily symmetric
- The generated control traffic is high

Criticalities

Using link-state routing in WMN requires modifications due to critical issues:

- Wireless links are not necessarily symmetric
- The generated control traffic is high
- There is not way to estimate link quality

Criticalities

Using link-state routing in WMN requires modifications due to critical issues:

- Wireless links are not necessarily symmetric
- The generated control traffic is high
- There is not way to estimate link quality
- There is no way to estimate path quality

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

OLSR is a routing protocol that tries to solve the previous issues. Its two main features are:

- Link state detection using `H` messages to avoid asymmetric links
- Proactive control message diffusion using MPR nodes to reduce control messages
- OLSRv2 was introduced by RFC 7181 in 2014, it does not change the basic behaviour but fixes some issues and improves some other ones.

Some definitions:

N: number of nodes in the network

D_i: set of 1-hop neighbors of node n_i

D_i^2: set of 2-hop neighbors of node n_i

M_i: set of MPR nodes chosen by node n_i

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- To avoid broken links each node will include in `H` its full 1-hop neighbor set.
- For each neighbor in the `H` a status is added that may be SYM/ASYM
- When node n_i receives an `H` by neighbor node n_j, n_i will include n_j in its `H` messages as an ASYM neigh.
- If n_j is able to receive `H` from n_i it will do the same
- When n_i receives `H` messages from n_j containing address of n_i in the neighbors, n_i will sponsor n_j as a symmetric neighbor. n_j will do the same.
- There are hysteresis mechanisms in order to avoid wireless fluctuations.

Topology Control messages (TC)

- In OLSR, LSP messages are named Topology Control (TC) messages
- TC contain only symmetric links

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

OLSR Step 1

Every node must include in Hs the IP address of nodes in D_i, with the SYM/ASYM bit set. From now on we will simply call neighbors only the SYM neighbors.

Two main consequences:

- The HELLO messages are larger than naive approach
- Using HELLO messages each node knows not only its 1-hop neighborhood but its 2-hop neighborhood

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- OLSR does not impose the generation of a new TC when a link fails
- This is (probably) because a wireless link can be extremely flaky, and oscillate much more than a wired link
- In this case, n_i may repeatedly send `TC`s when the link breaks down and when it comes back up
- Recall that a `TC` not only occupies airtime, but it also triggers re-computation of routing tables on other nodes.
- It is wiser to let the link have a bad quality, and just let the routing metric take care of it.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

Each node of a network with N nodes with in average d neighbors periodically generates:

- Hs: few bytes containing own address (size s) and neighbour addresses, generated every $t_{\mathtt{H}}$ (defaults to 2 sec).
- TCs: contains the set of IPs of all the neighbors (size $s * d$) at interval $t_{\mathtt{TC}}$ (defaults to 5 sec), re-broadcast by every node in the network (multiply by N).

A total of

$$\frac{(s * (d + 1)) * N}{t_{\mathtt{H}}} + \frac{(s * d) * N^2}{t_{\mathtt{TC}}}$$

bytes per second generated on the whole network (plus protocol overhead). Since the number of links instead grows linearly with N, control traffic *per link* scales $\propto N$.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Consider that all these messages are sent to the broadcast address.
- Broadcast messages at layer 2 are sent at the basic rate, which can be 100 times slower than the maximum rate (for instance in 802.11n broadcast is sent at 6mb/s Vs 600 mb/s maximum rate).
- It means that every time you send a broadcast message, you occupy the airtime for the equivalent time of 100 unicast packets of the same size

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

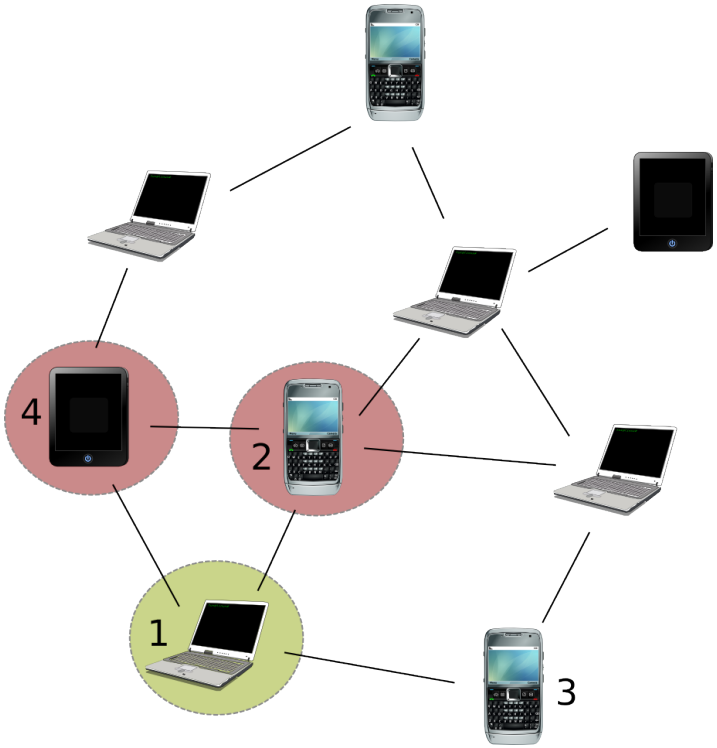
MPR set definition

The MPR set M_i of a node n_i is an arbitrary subset of its symmetric 1-hop neighborhood D_i which satisfies the following condition: every node in the 2-hop neighborhood D_i^2 of n_i must have at least a symmetric link towards a node in M_i

- Basically, M_i is a subset of the neighbors of n_i that can be used to reach any node in D_i^2 (i.e. they *cover* all the nodes in D_i^2)

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

Node 2 and 4 are MPR for node 1 and node 1 is a *selector* for node 2 and 4. Node 3 is not necessary to cover n_1^2.

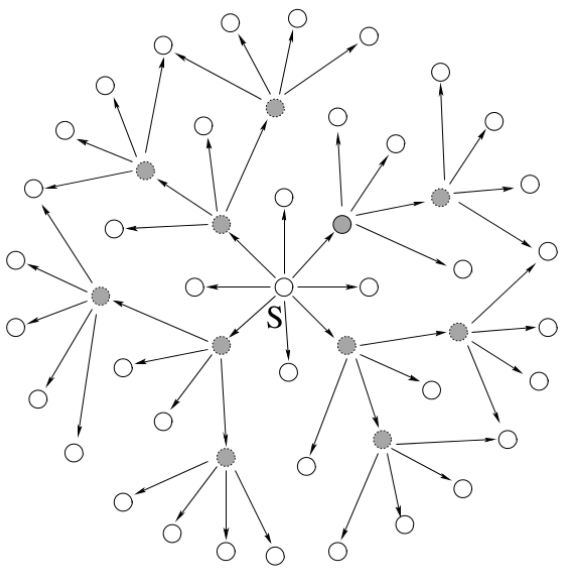


UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- When n_i needs to broadcast a message to all the nodes in D_i^2, it will send it in MAC layer broadcast with IP Time-To-Live set to 1.
- If only nodes in M_i rebroadcast the message, it will reach all nodes in D_i^2.
- The smaller M_i compared to D_i the less packets are sent, compared to naive broadcast.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- If n_i does a network-wide broadcast, it sends a packet to the broadcast MAC address and broadcast IP address, with TTL>1
- Every node in D_i receive it
- If only nodes in M_i broadcast it again, and the process follows on if there are no losses every node n_j receives the packet
  - MPR guarantes broadcast communication to all the network



using only a subset of nodes

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Two-hop distance is preserved: Choosing MPRs does not change the connectivity of the network.
- If n_j is in D_i^2 its distance from n_i remains 2
- There exist a shortest path from n_i to n_k made of MPR nodes
- If every MPR generates TCs, n_k knows every MPR and their neighborhood
- Since every node in the network (excluding trivial topologies) must be selector of at least one MPR node, n_k can compute a shortest path to any other node

$$\rightarrow \text{shortest paths are preserved.}$$

TC Size Reduction

- There is no need for a MPR to include in TC messages its full neighborhood
- Since every node is a selector of at least one MPR, then an MPR includes in its TCs not its full neighborhood, but its selectors only.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

OLSR Step 2

Each node must select a subset M_i as small as possible. In the `Hs` for each neighbor another bit (MPR/NOT_MPR) is added, so each MPR knows its selector set. Only the MPRs re-broadcast flooded messages, and only from their selectors.

Step 3

Only MPR nodes generate `TCs`. Each `TC` includes the selector set of the MPR that generated it

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

With MPRS the network-wide average overhead passes from:

$$\frac{s * N}{t_{\mathrm{H}}} + \frac{(s * d) * N^2}{t_{\mathrm{TC}}}$$

to

$$\frac{s * N}{t_{\mathrm{H}}} + \frac{(s * \mathbf{m}) * \mathbf{M}^2}{t_{\mathrm{TC}}}$$

where m = average size of the selector set

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

OLSR Step 1

Every node must include in Hs the IP address of nodes in D_i, with the SYM/ASYM bit set.

OLSR Step 2

Each node must select a subset M_i as small as possible. In the Hs for each neighbor another bit (MPR/NOT_MPR) is added, so each MPR knows its selector set. Only the MPRs re-broadcast flooded messages, and only from their selectors.

Step 3

Only MPR nodes generate TCs. Each TC includes the selector set of the MPR that generated it

OLSR: summing up

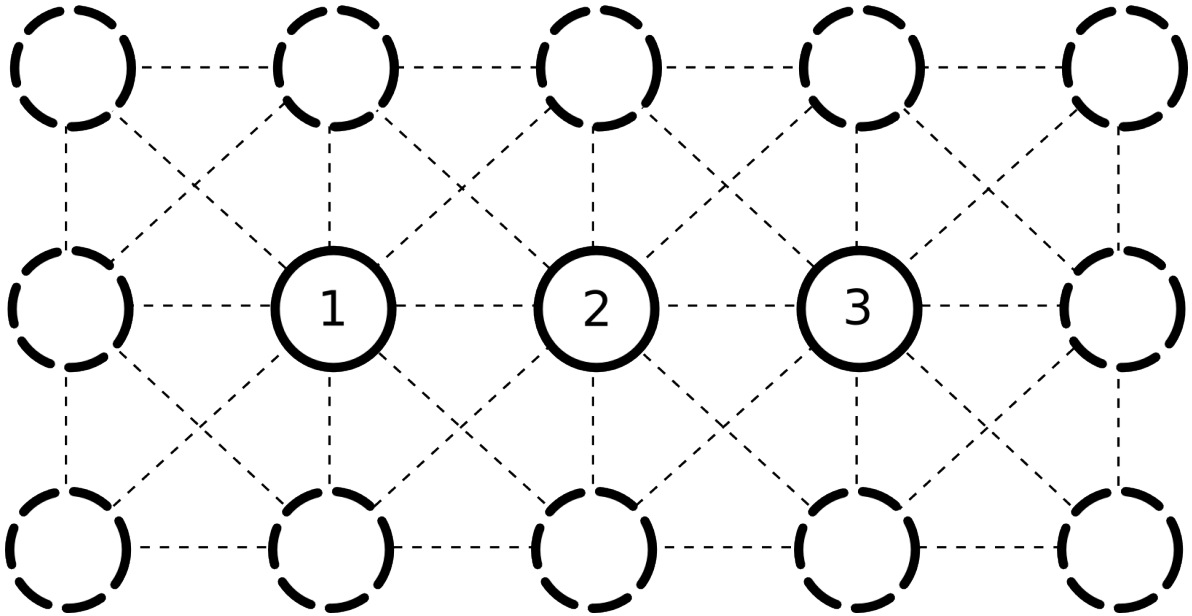
- If all the nodes apply step 1,2,3 and there is no loss, than all the nodes have enough information to compute the shortest path to any node in the network.
- Since only the MPR generate TC messages and each TC includes only a fraction of the neighborhood of the MPR that generated it the overall control traffic is reduced
- It is very important that each node selects a minimal MPR set, since M depends on that (not that easy).

Examples[2]

- Only 3 nodes over 5 in the network generate TC messages
- Each TC message is forwarded exactly 2 times

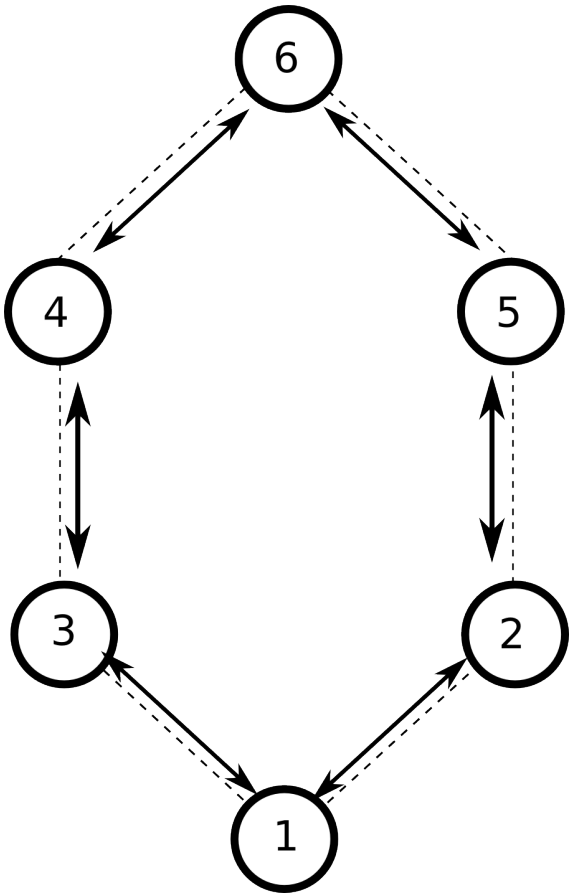
[2] dashed edges are wireless links, arrows go from a selector to the MPR (only the most important ones). Solid vertexes are MPRs.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science



- Only 3 nodes over 15 in the network generate TC messages
- Each TC message is forwarded exactly 2 times

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science



- All the nodes in the network generate TC messages
- Each TC message is forwarded exactly 5 times

Not Enough

- OLSR so far solved the asymmetrical link problem, and alleviated the overhead problem, compared to a classical link state.
- We still need to improve on the quality
- We still have room for improvement in overhead

OLSR Extensions

- OLSR has been extended by developers:
  - Introduction of a link-quality measure → ETX
  - Further reduction of overhead → Fisheye

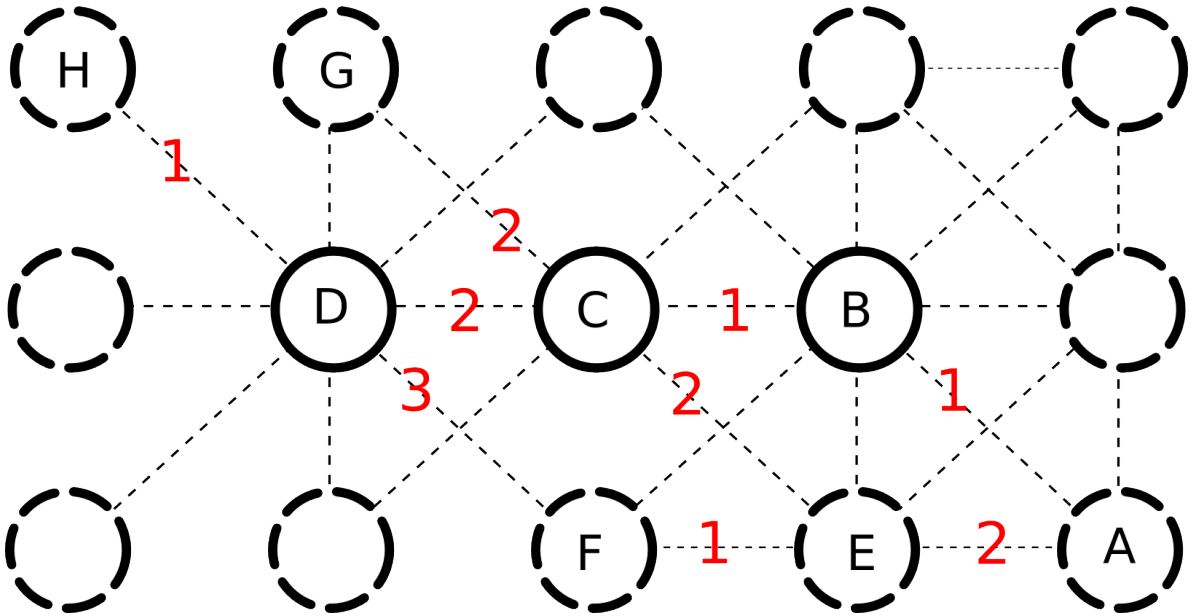
Problem

- Is it always better to use a "short" path?
- In some cases, a very bad link (say, with 50% loss) is there.
- Probably it is better to avoid that link, even if this means taking a longer path.

QOLSR

- Each node makes an estimation of the link quality for its links (a numeric value).
- Each node includes this value in its HELLO and possibly TC messages.
- The adjacency matrix now has weights on the edges and distance can be computed on a weighted graph.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science



- Without weights, path A-B-C-D-H and A-E-C-D-H have the same cost
- With weights A-B-C-D-H costs 5 while A-E-C-D-H costs 7

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

ETX (Expected Transmission Count) is the simplest, yet one of the most used metrics to estimate link quality. Take two neighbor nodes n_i and n_j

- In each `H` n_i includes the value of $t_{\texttt{H}}$
- n_j now knows the estimated time of arrival of the next `H`
- n_j now can sense the loss of a `H`
- n_j keeps a time-window in which it counts received and missed Hs.
- The ratio received/lost is called the Link Quality (LQ) measured by n_j for neighbor n_i.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- In each `H`, n_j includes the LQ towards n_i, same does n_j.
- When n_j receives `Hs` from n_i it receives LQ computed by n_i: the reverse link quality (RLQ)
- Finally, ETX metric is given by:

$$ETX = \frac{1}{LQ * RLQ}$$

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

Note:

- LQ: probability to successfully send a packet from n_i to n_j
- RLQ: probability to successfully send a packet from n_j to n_i
- Since each 802.11 packet to be successfully transmitted requires two packets (data packet + ACK in the other direction) then the probability of successfully transmit a packet is estimated by RLQ*LQ.
- ETX thus is the inverse of this probability, that is, the average number of transmission attempts n_j needs to do to successfully send a packet to n_i
- ETX is symmetric

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.
  - **ETX is optimistic**

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.
  - **ETX is optimistic**
- Unicast packets are ACKed and retransmitted, so the loss is lower than the same packet, at the same rate, in broadcast

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.
  - **ETX is optimistic**
- Unicast packets are ACKed and retransmitted, so the loss is lower than the same packet, at the same rate, in broadcast
  - **ETX is pessimistic**

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.
  - **ETX is optimistic**
- Unicast packets are ACKed and retransmitted, so the loss is lower than the same packet, at the same rate, in broadcast
  - **ETX is pessimistic**
- ETX does not say anything about other quality features of the link: capacity, delay etc. . .

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- Broadcast packets are sent at the base rate, unicast packets are sent at the negotiated rate, the highest the rate, the highest the loss.
- Broadcast packets are (tentatively) small packets, so the chance of collision is smaller than traffic packets.
  - **ETX is optimistic**
- Unicast packets are ACKed and retransmitted, so the loss is lower than the same packet, at the same rate, in broadcast
  - **ETX is pessimistic**
- ETX does not say anything about other quality features of the link: capacity, delay etc. . .
  - **ETX is just imprecise**

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

On the other hand

- ETX does not require any cross-layer interaction, it is based only on layer 3
- It can be safely implemented in any driver/hardware/OS
- It is larger or equal to one, and thus can be used as an additive metric

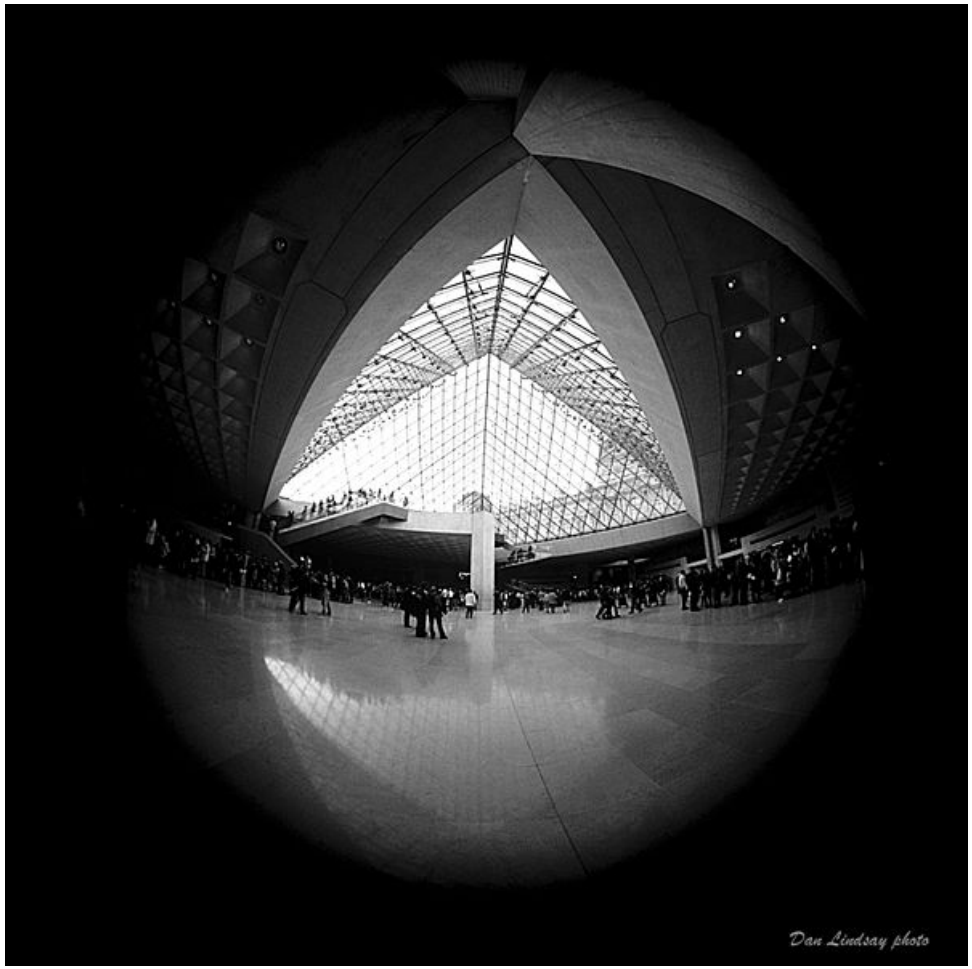
Fisheye

- A node needs only the necessary information to decide what is the next hop on the shortest path to the destination.
- We have seen that nodes already have a limited view of the topology outside their 2-hop neighborhood
- This approach can be enhanced with fisheye strategies

Fisheye

- The word fisheye refers to the view of the world a fish has under the water
- If you look up from below the water you see the whole 180 degrees horizon compressed in a cone with a smaller angle
- Basically, you have better vision on the close thinks and a fading compressed vision on objects far away. The same happens if you use a *fisheye* lens on a camera

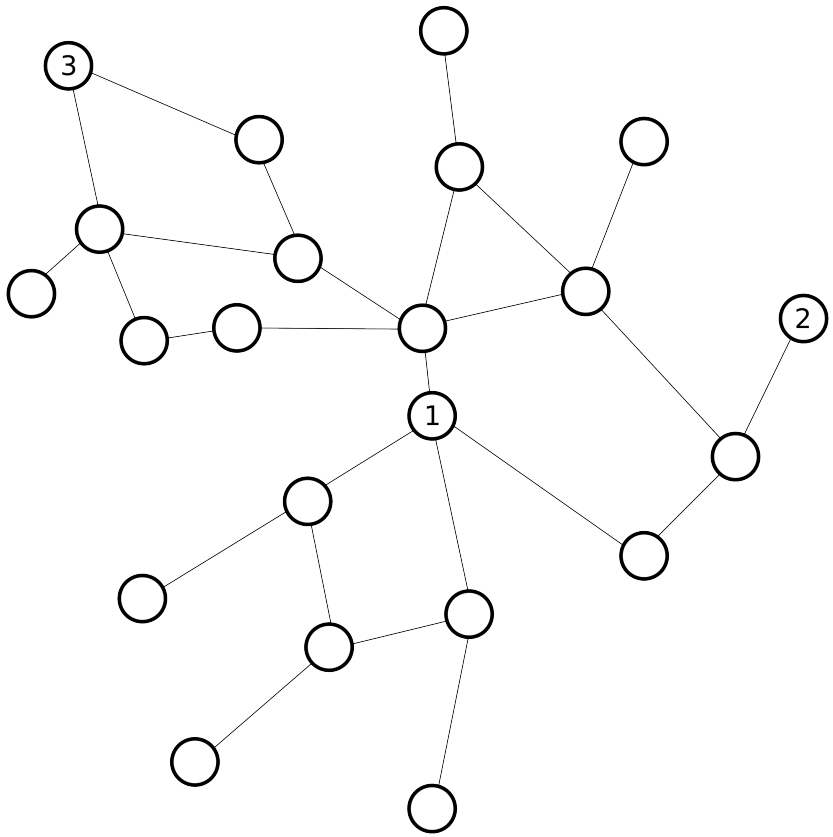
Fisheye [3]



[3] http://en.wikipedia.org/wiki/File:Museumfisheye.jpg

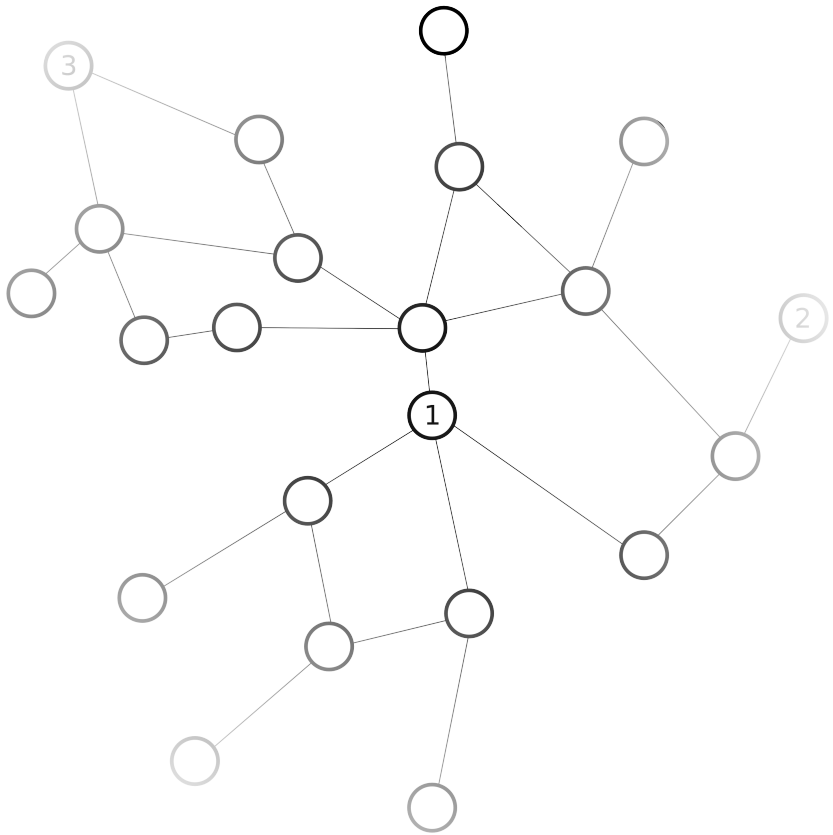
UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

Fisheye transforms the network from this . . .



UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

. . . to this



UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

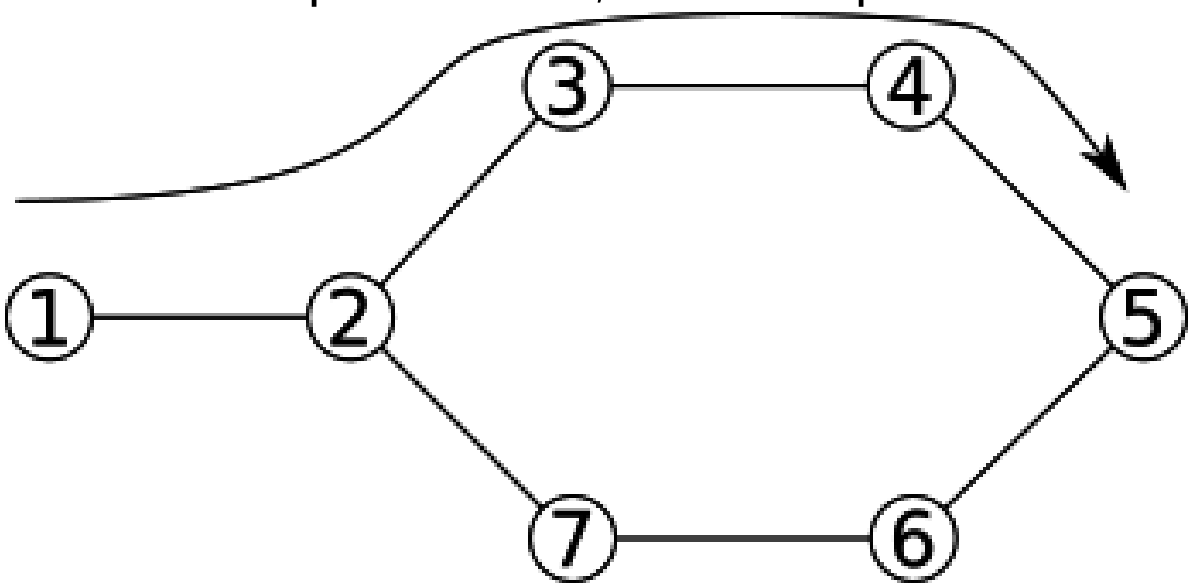
- To obtain this effect, the TTL of the TC messages is not always to the maximum.
- a sequence of TTL can be used to control the diffusion of TC messages, for instance {2,4,16}
- Doing this, the first TC message will be sent only to nodes that are 2-hops away, the second to nodes that are 4-hops away, the third to all the nodes in the network
- Periodically every node receives enough information to build all the routing table, but the information is more accurate (it is fresher) for closer nodes

Fisheye

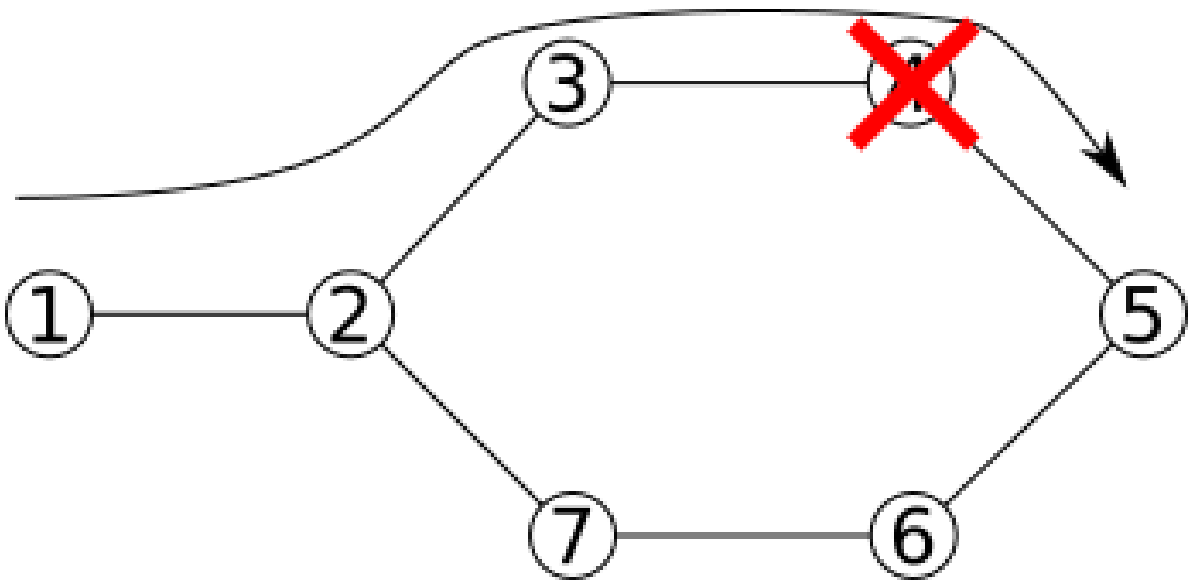
- With Fisheye, the total overhead cease to be *linear* with the number of MPR nodes
- How to define the TTL sequence is an open issue

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

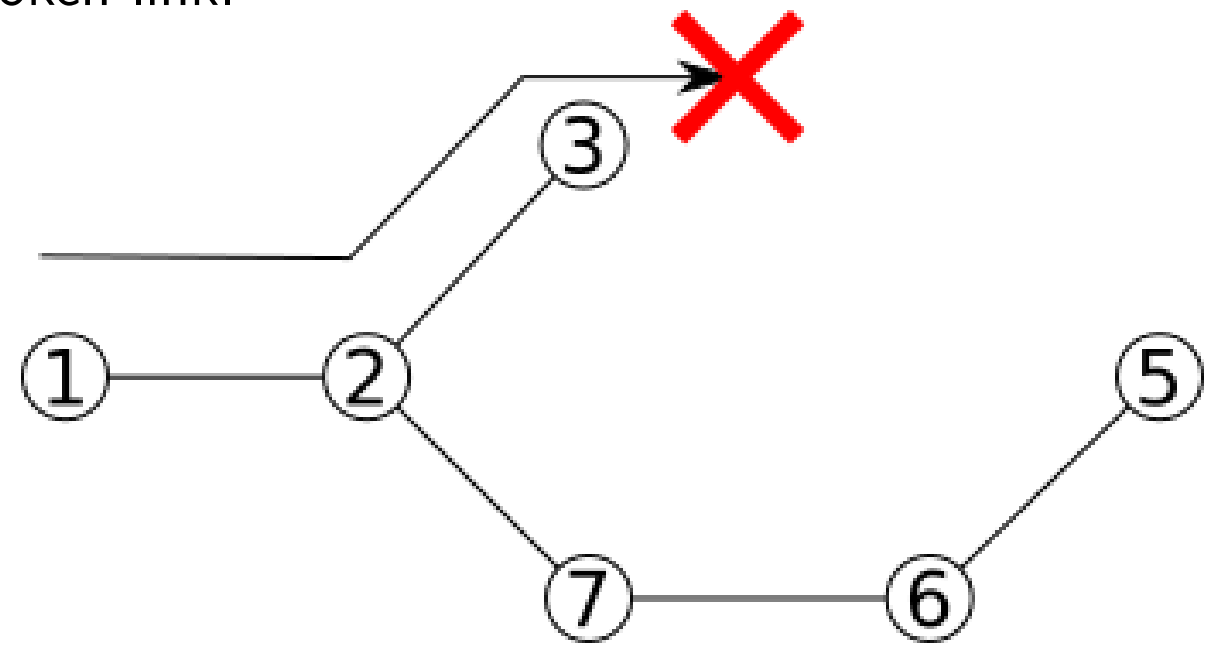
Let's consider this simple network, with the path 1-5 as shown:



Imagine node 4 breaks

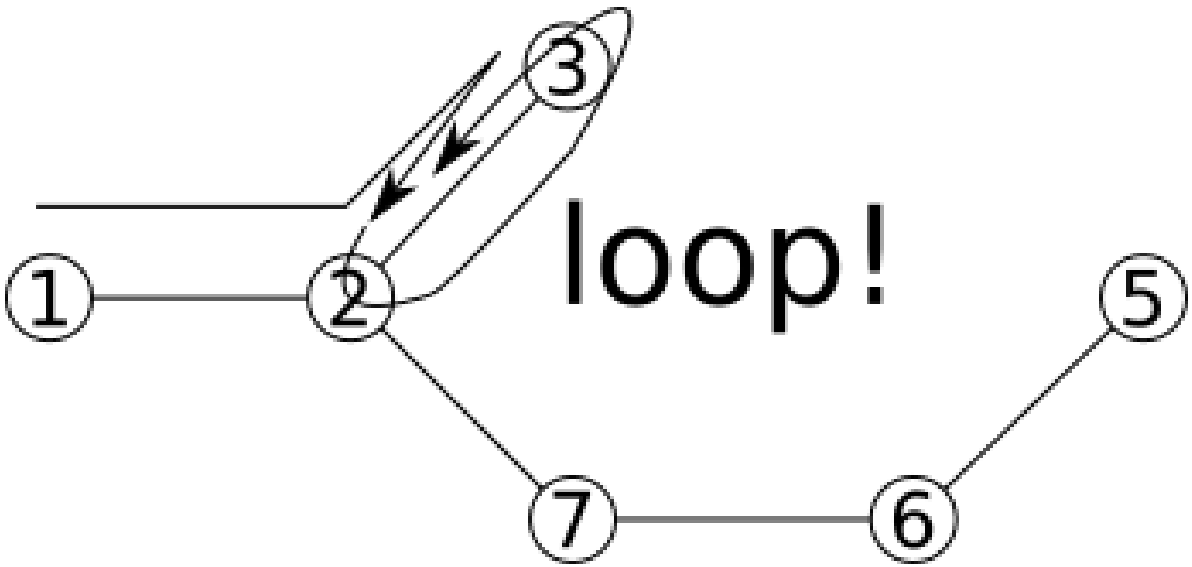


Up to when node 3 does not sense the broken link, packets are lost on the broken link.



UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

When node 3 looses enough HELLOs from 4, it will recalculate its own path to node 5.



Node 2 still doesn't know that node 4 died, a loop is created!

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- The loop is due to the fact that node 3 and node 2 have a different knowledge of the network topology
- Node 3 in fact receives HELLOs from node 4 with a higher frequency than node 2 receives TCs.

UNIVERSITY OF TRENTO
Department of Information Engineering and Computer Science

- But this is what Fisheye does all the time!
- Fisheye create regions of nodes with different (more or less updated) knowledge of the topology.
- Nodes in the border between regions can create loops.