

AODVjr

A Simplified Version of AODV

Luke Klein-Berndt & Ian Chakeres

National Institute of Standards and Technology
University of California at Santa Barbara



Ad hoc On-Demand Distance Vector (AODV) Routing

AODV is an ad-hoc routing protocol. It is able to discover a route through a network of computers.

- Creates routes on-demand
- Built for mobile networks
- Loop free with quick convergence
- Can scale to handle a few hundred nodes
- Can be integrated into the existing protocol stack

When AODV was designed it incorporated many features designed to maximize performance at the cost of added complexity.



Simplicity

The goal of AODVjr is simplicity. Unlike AODV it does not require:

- RERR messages
- Sequence numbers
- Precursor lists
- Gratuitous RREP
- Hop count
- Hello messages

It is able to do this by requiring only destinations to reply to RREQ and uses end-to-end hello messages to maintain routes.

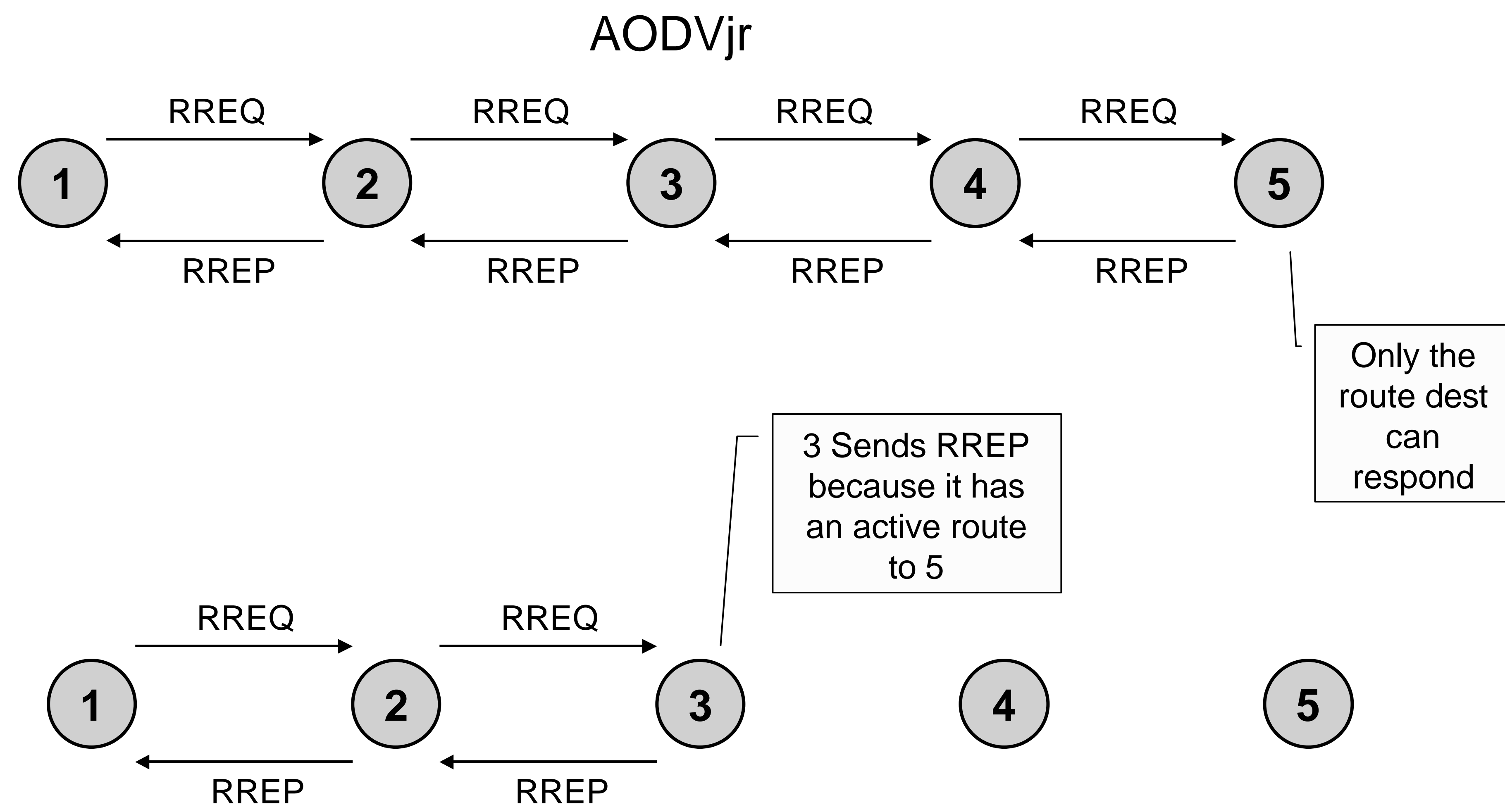


Link Layer Detection

- Link layer detection allows a sending node to detect if a unicast packet is successfully received.
- In simulations AODV using link layer detection provides amazing performance.
- Currently it is impossible to access link layer feedback information in off the shelf hardware.
- Current implementations use periodic hello messages to detect local link connectivity.
- Hello messages cause a large amount of control overhead. Each node must periodically send broadcast packets. Each receiving node must also process them.



Route Discovery

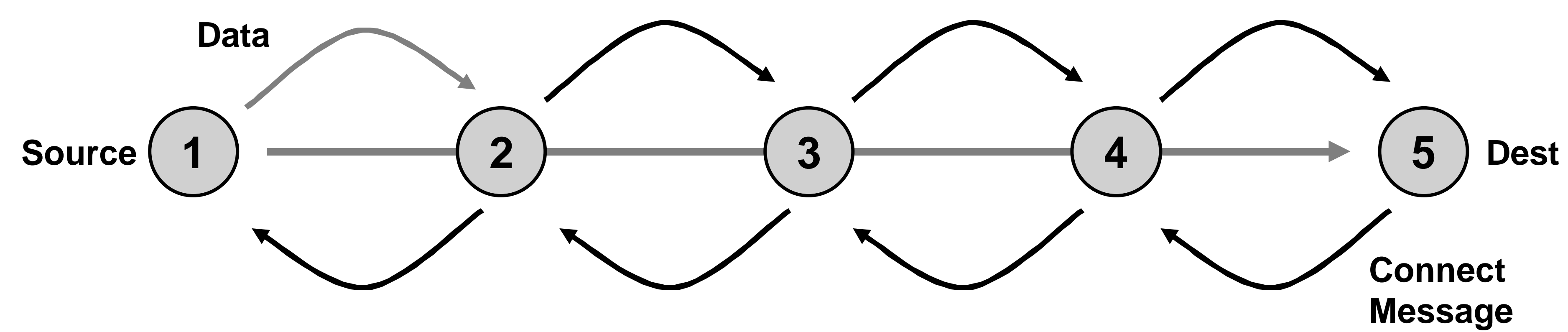


MOMENT UCSB

NIST
National Institute of Standards and Technology

Route Maintenance

AODVjr



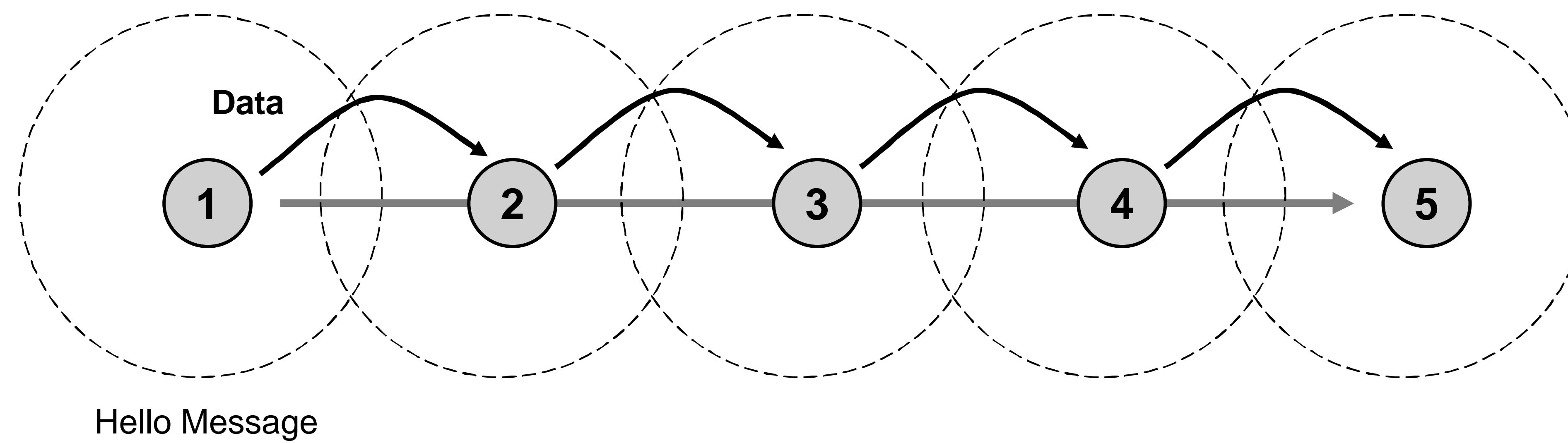
- The *source route* is kept alive by data traveling along it.
- If the destination has received data in the past few seconds and has not sent data to the source, it sends a Connect message to the source, maintaining the *reverse route*.

MOMENT UCSB

NIST
National Institute of Standards and Technology

Route Maintenance

AODV

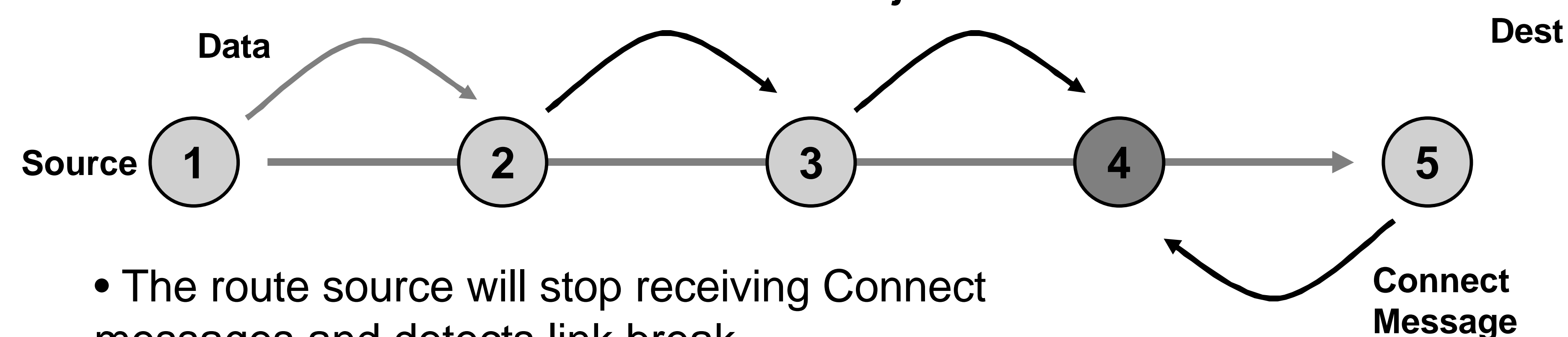


- Periodic Hello messages are used to detect link connectivity along the *source route*.
- In simulation it is possible to use link layer feedback to sense link breaks.
- The *source route* is kept alive by data traversing it.



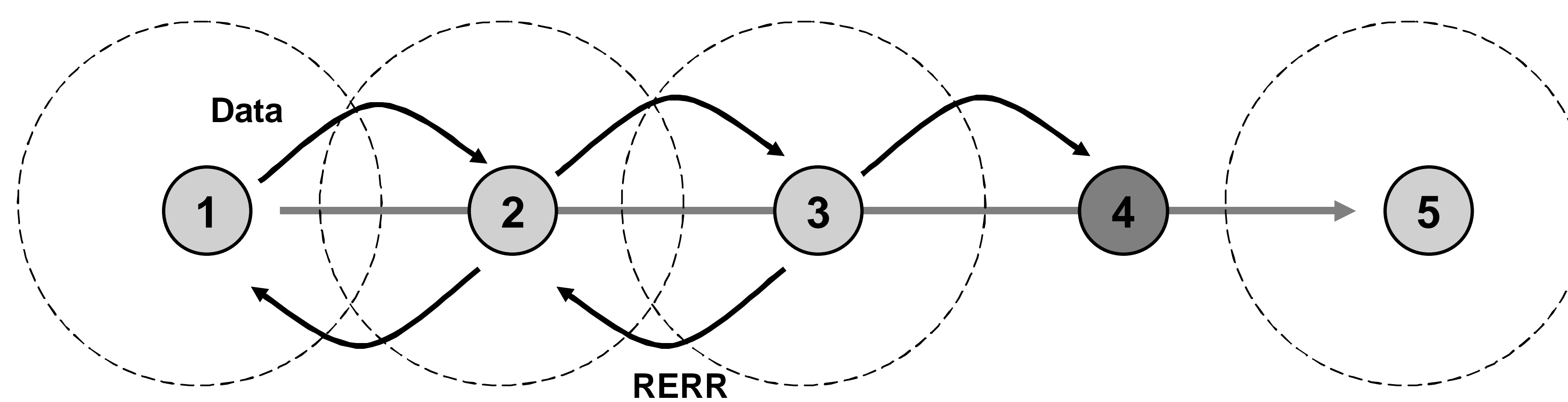
Route Maintenance

AODVjr



- The route source will stop receiving Connect messages and detects link break.

AODV

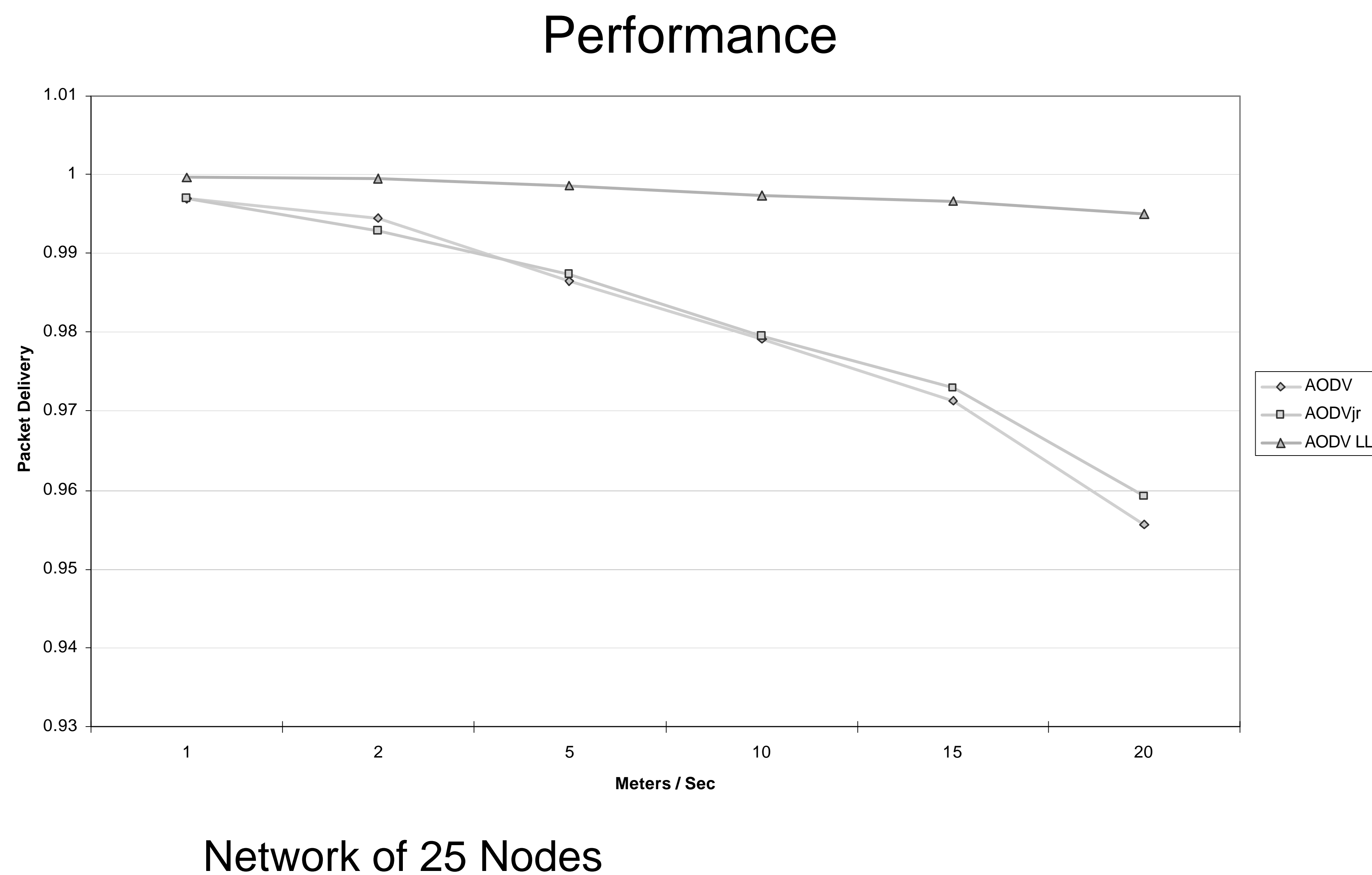


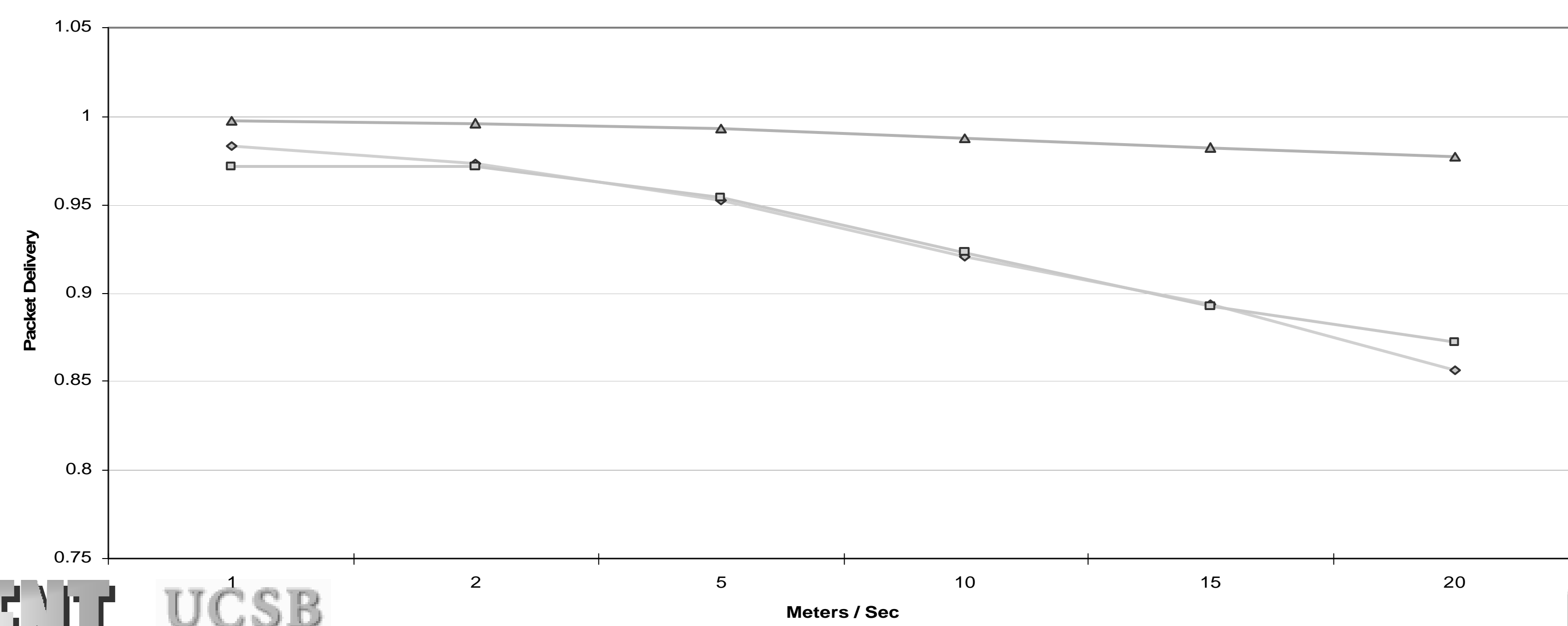
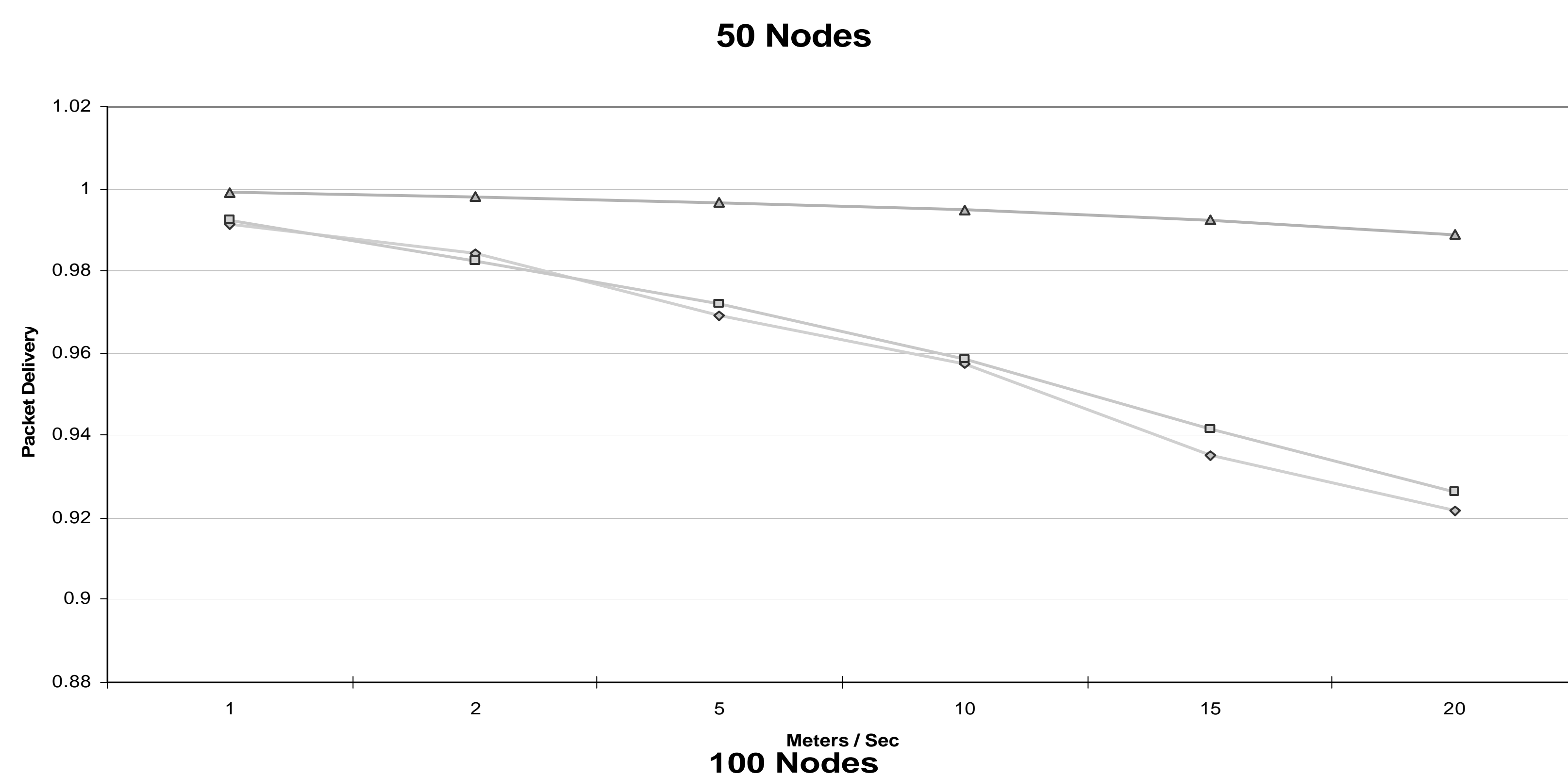
- Upstream node detects a link break and sends a RERR.



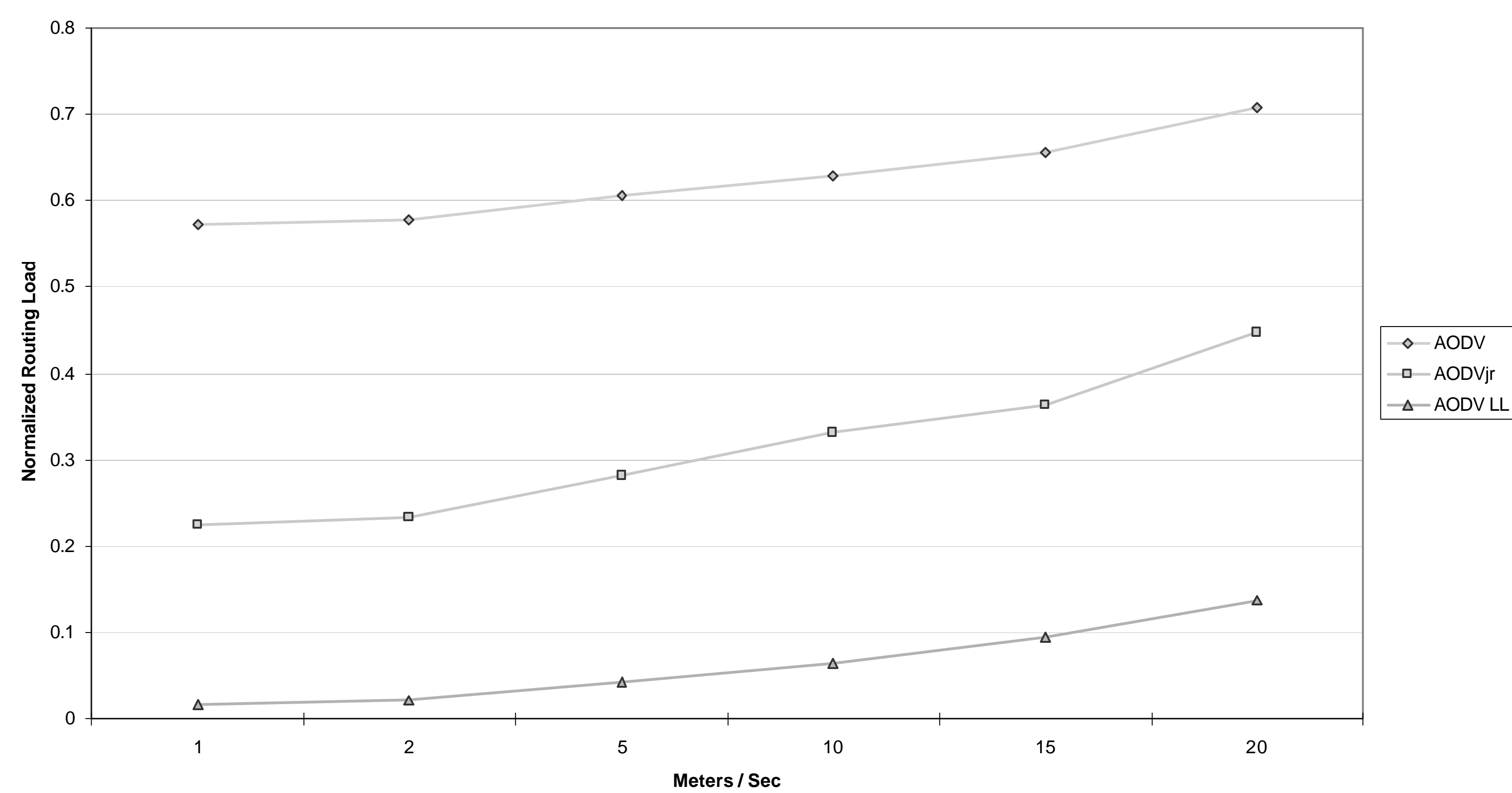
Simulations

- NS-2
- AODV from Mahesh Marina
- AODVjr developed by us
- Random waypoint
- 802.11 2Mbps
- 900 seconds
- Five runs averaged
- Ten sources (unless otherwise noted), five 512-byte packets per second





Normalized Routing Load



In a network of 25 nodes, with 10 sources, AODVjr maintains a much lower normalized routing load compared to AODV with hello messages.



Control Overhead

AODV with hello messages

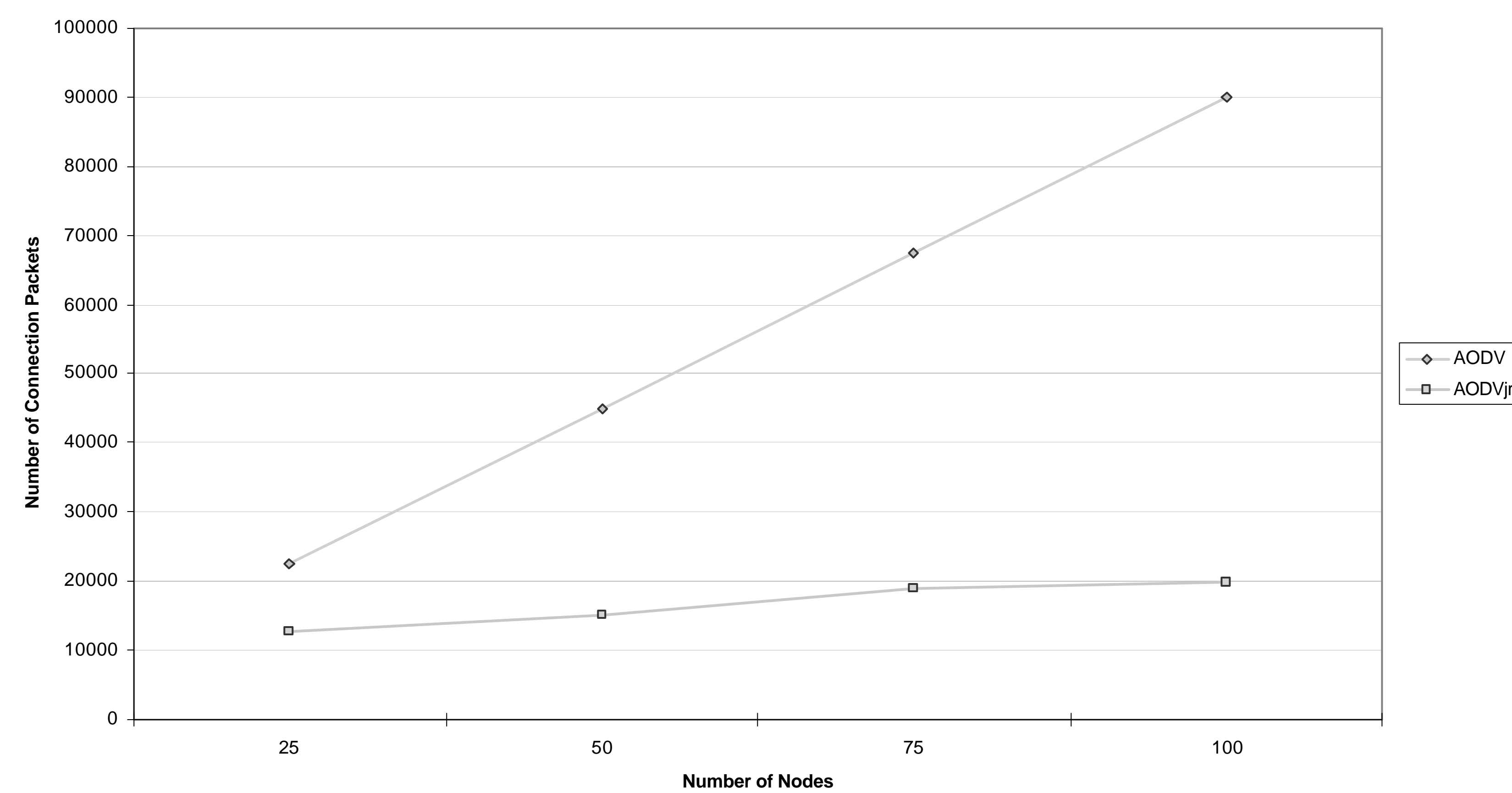
- Every node in the network must broadcast a Hello message and every receiving node must process it. Routing overhead is controlled by the total number of nodes.
- If a route breaks a RERR message must be sent to each node using that route, eventually reaching the source.

AODVjr

- Connect packets are unicast from destination to source along active routes; routing overhead is affected by the number of active routes.



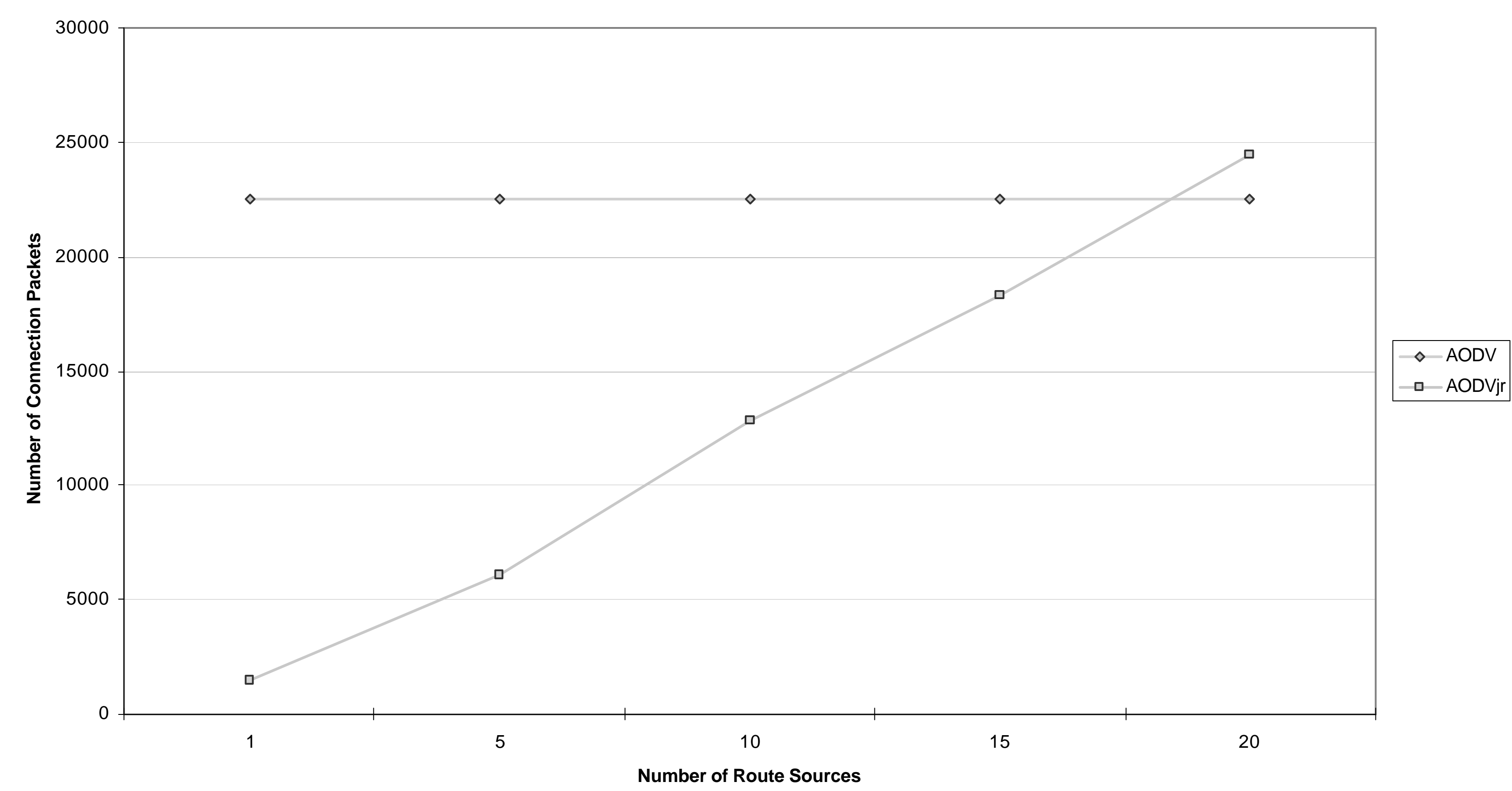
Route Maintenance Packets



The number of nodes in the network determines the number of route maintenance packets in AODV (with hello messages) generates.



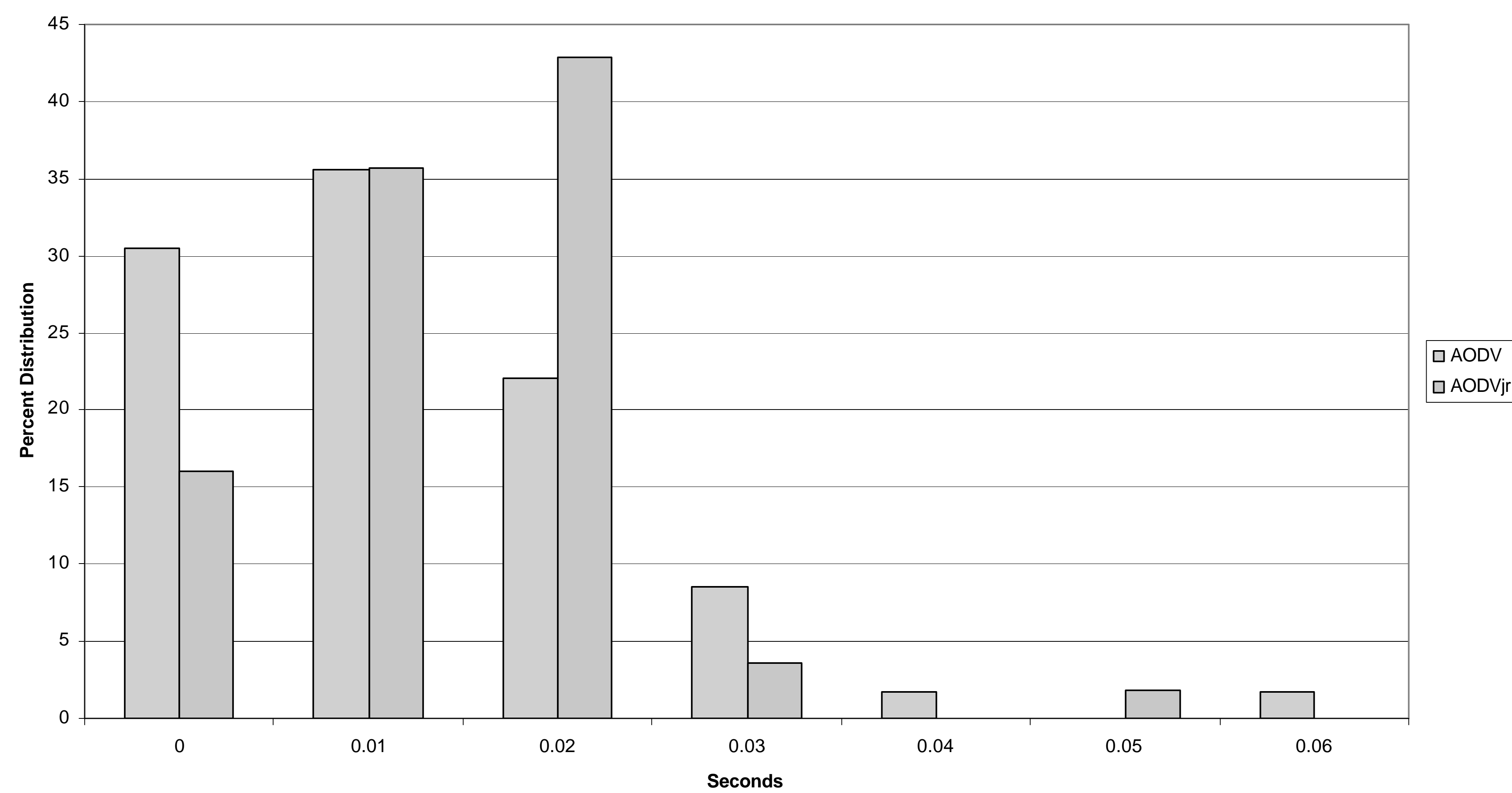
Route Maintenance Packets



In AODVjr the number of route maintenance packets is directly related to the number of active routes.



Route Discovery Latency



Since AODVjr discovers the entire route, it has a slightly higher route discovery latency compared to AODV.



Conclusion

- With currently available hardware AODVjr is as good as AODV with hello messages.
- Routing overhead in AODVjr is related to the number of active routes, not the number of nodes.
- RERR messages, sequence numbers and other optimizations could be added optionally to further improve performance.
- Security can easily be added as there are no mutable fields in the control packets.
- AODVjr is much simpler than AODV.

