



Labeled Distance Routing Protocol

J.J. Garcia-Luna-Aceves

Marc Mosko

U.C. Santa Cruz

Charles E Perkins

Nokia Research

Labeled Distance Routing

- LDR – what is it?
 - Ad hoc: wireless networks with no fixed infrastructure and mobility.
 - On-demand: nodes maintain routes as needed.
 - Loop-free: the routing tables at nodes never form a loop.

Other protocols

- DSR
 - Uses source routing for loop-freedom, but may still have temporary loops if using 'savaging'.
- OLSR
 - Pro-active link-state protocol, can have loops.
- TORA
 - Requires synchronized clocks.

Other protocols (cont)

■ AODV

- Uses a sequence number for loop-freedom.
- LDR is based on AODV, but the loop-free algorithms are very different.

■ ROAM

- On-demand versions of DUAL (next slide). Require reliable communications and some use diffusing computations.
- Use a *feasible distance* invariant for loop-freedom.

DUAL

■ Diffusing Update Algorithm

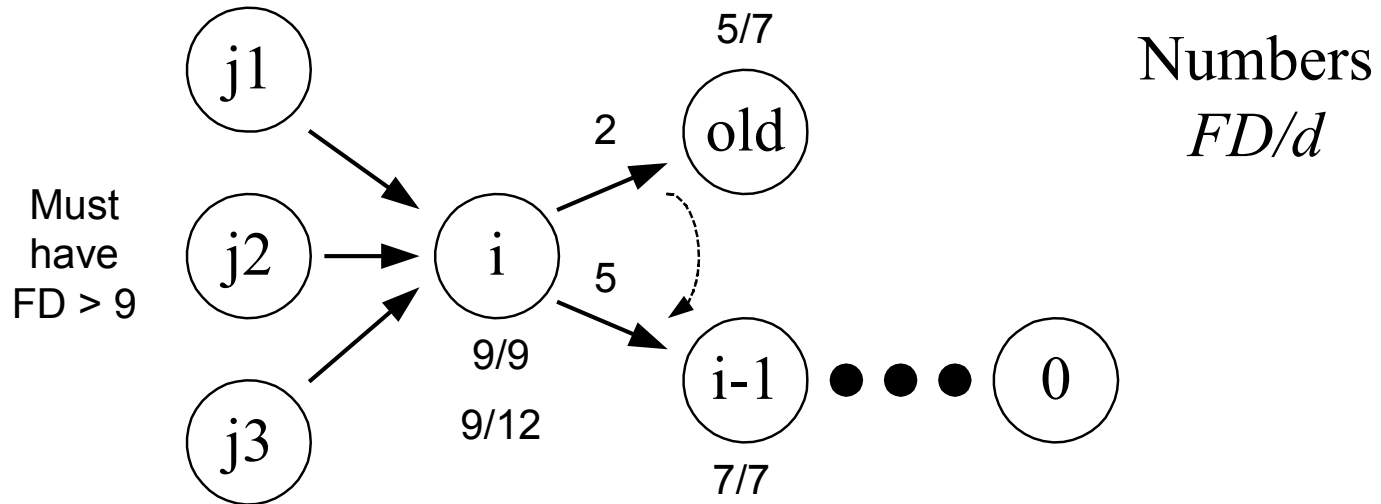
- At a given node N for destination D , the feasible distance fd_D^N is the minimum measured distance d ever known.
- Along a path $\{v_k, \dots, v_0\}$, for node v_i to add or change successor to v_0 , it must satisfy double bound where S is all predecessors of v_i for v_0 :

$$\text{New successor} \rightarrow fd_{v_0}^{v_{i-1}} < fd_{v_0}^{v_i} \wedge (\forall j \in S) fd_{v_0}^{v_i} < fd_{v_0}^{v_j} \leftarrow \text{Existing predecessors}$$

DUAL (cont)

- Because of double bound, a choice of feasible successor maintains *downstream* and *upstream* ordering.
- The ordering prevents loops, because a node cannot be on a chosen successor path.
- To reset a FD, DUAL uses a diffusing computation to force predecessors to choose a new successor or invalidate their routes.

FD Example



Node i switches successors from old to $i-1$. The distance at $i-1$ is 7, which is $< FD$ 9 at i , so it is feasible successor.

LDR and Loop-freedom

- Uses two features

- DUAL-based feasible distance.

- Allows nodes to find new routes in their neighborhood, even after link breaks. AODV requires an increased sequence number which generally sends RREQ to destination.

- AODV-based sequence number

- This replaces the reliable communications and diffusing computations common to other DUAL-based protocols. Functions as a path reset.

LDR loop-freedom

- Ordering (SN,fd) properties
 - Stored per destination, on-demand
 - $(sn_2, fd_2) \succ (sn_1, fd_1)$ if $sn_1 > sn_2$ or $sn_1 = sn_2$ and $fd_1 < fd_2$
 - Read as “node #1 is a feasible successor to #2” or as “Node #1 is in-order for node #2”.
 - $\text{Min}\{(sn_1, fd_1), (sn_2, fd_2)\} = \#1$ if $(sn_2, fd_2) \succ (sn_1, fd_1)$; else #2.
 - (sn_1, fd_1) is a non-increasing function over time.
 - The sequence number never decreases or for the same SN, the feasible distance never increases.
 - SN is a 64-bit number, so it may increase without bound for foreseeable lifetime.

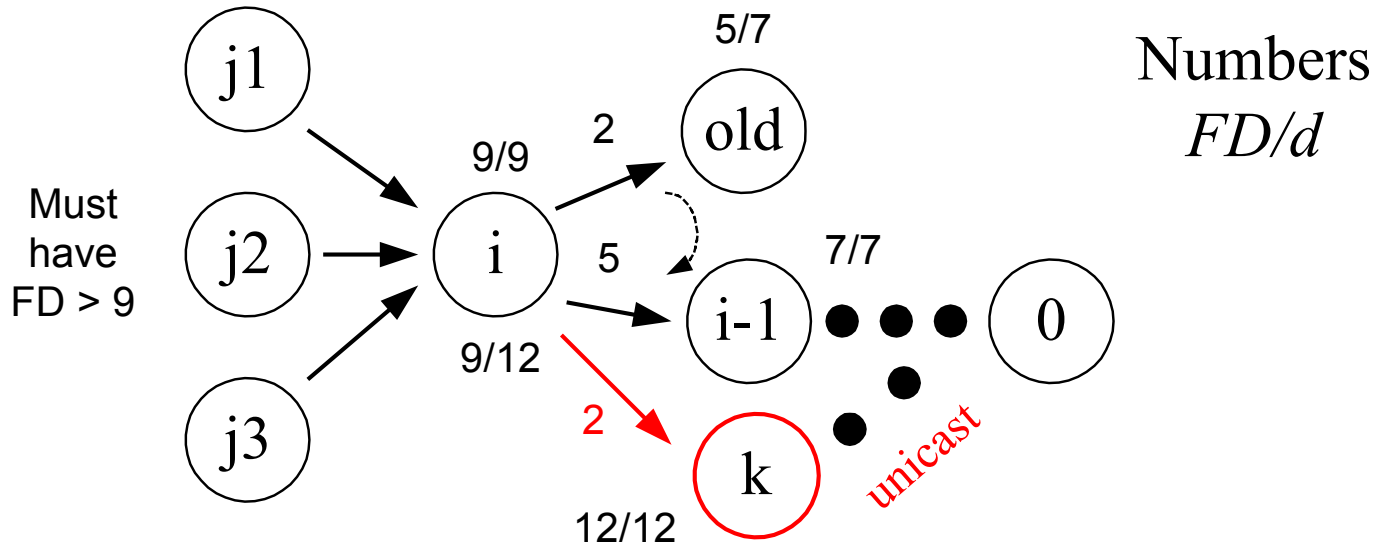
LDR Update Algorithm

- Solicitation process (route requests)
 - Node puts stored (SN,FD) for destination T in RREQ & broadcasts
 - There are “unassigned” nodes, not discussed in talk.
 - As request goes hop-by-hop, each relay node puts minimum of RREQ’s (SN,FD) or its own stored (SN,FD) for T in to packet.
 - RREQ records the minimum ordering seen by packet for T .
 - A node may reply if its (SN,d) pair for T in-order for RREQ. A node may always reply to a RREQ for itself. d is measured distance (next slide).

LDR Update Algorithm

- Advertisement (route reply)
 - When a node N issues a RREP for T , it puts its (SN, d) pair for T in RREP.
 - d is the measured distance to destination.
 - Because $fd \leq d$, this will maintain ordering.
 - Hop-by-hop, each node tests feasibility and sets/updates routes, then issues new RREP with current (SN, d) for T .
 - RREP takes reverse path of broadcast flood
 - Ensures that reply is usable along reverse path because RREP sent based on *minimum* ordering of path.

LDR Example



Node i switches successors from old to $i-1$. The distance at $i-1$ is 7, which is $< FD$ 9 at i , so it is feasible successor. The RREQ that goes through k is infeasible and RR is set. Path built via k would have $9/14 \rightarrow 12/12 \rightarrow \dots$ otherwise.

LDR proof outline

■ Control packets

- Do not loop because they form a tree (RREQ) or path in tree (RREP)

■ Loop-freedom

- Ordering is non-increasing function with time. If node A chooses B as a successor, then $A \succ B$ is always true as long as B is A's successor.
 - If $A(t_0) \succ \dots \succ B(t_0)$, then
 - $\neg(B(t_1) \succ A(t_1) \succ \dots \succ B(t_1))$
 - $B(t_1) \succ A(t_1) \succ \dots \succ \neg B(t_1)$ could be okay

LDR proof outline

■ Termination

- Calculation: initiation of a RREQ until timer expires or receives any feasible RREP.
 - RREP not necessarily in response to RREQ, could be other cross-traffic.
- Assuming no changes in network during time of calculation except those caused by calculation, node starting calculation guaranteed successful termination if destination is reachable.
 - This is because RREQ records minimum necessary condition to establish feasible path or forces path reset.

LDR Summary

■ Protocol

- SN is master reset
- FD orders nodes for localized repairs

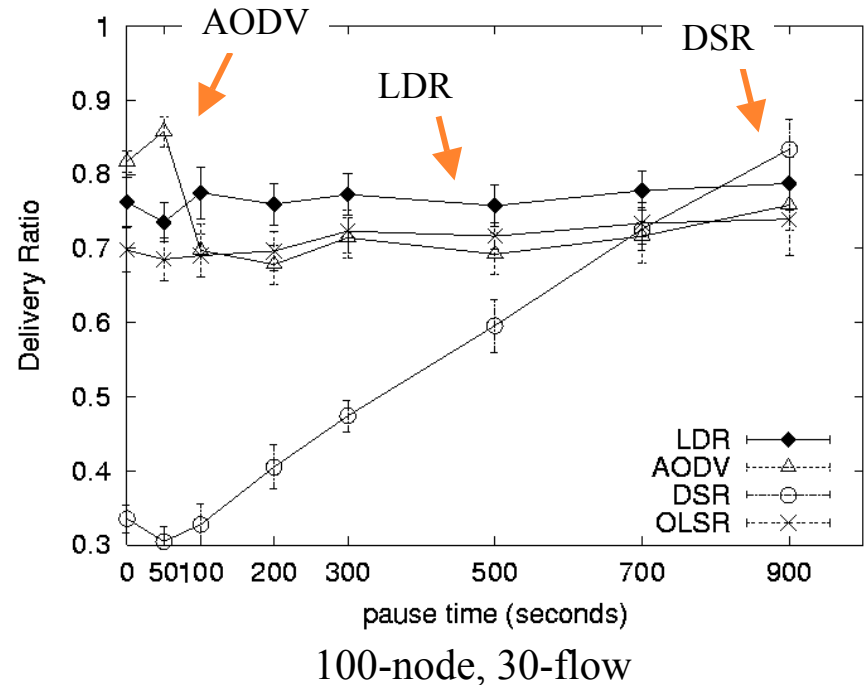
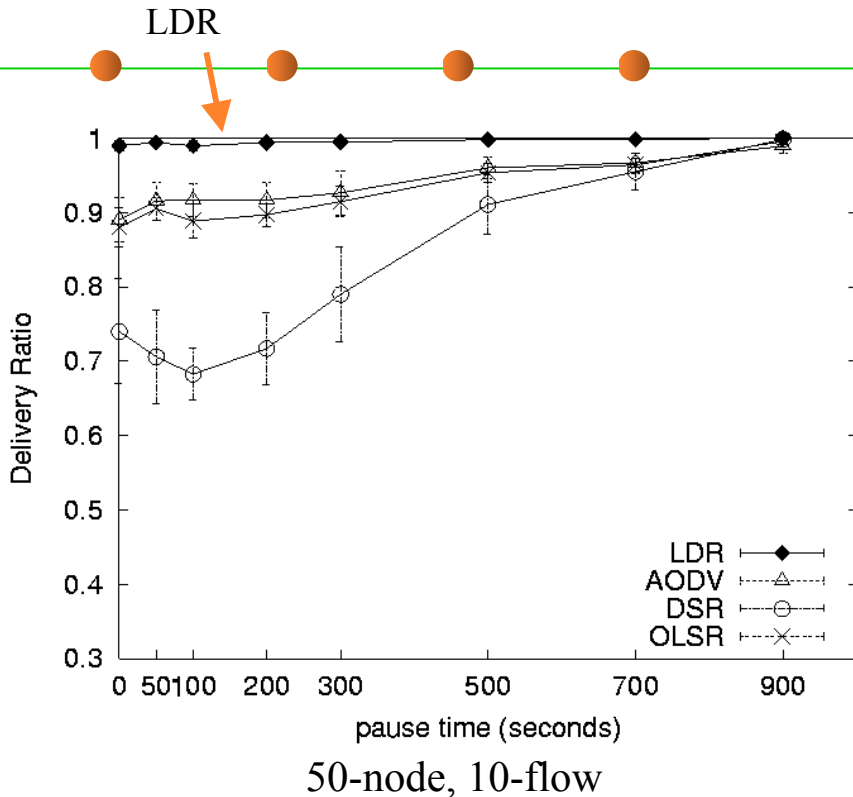
■ Update Algorithm

- RREQ records strongest metrics along path
- RREP satisfies strongest metrics and follows same reverse path
- Feasibility tested hop-by-hop

Simulations

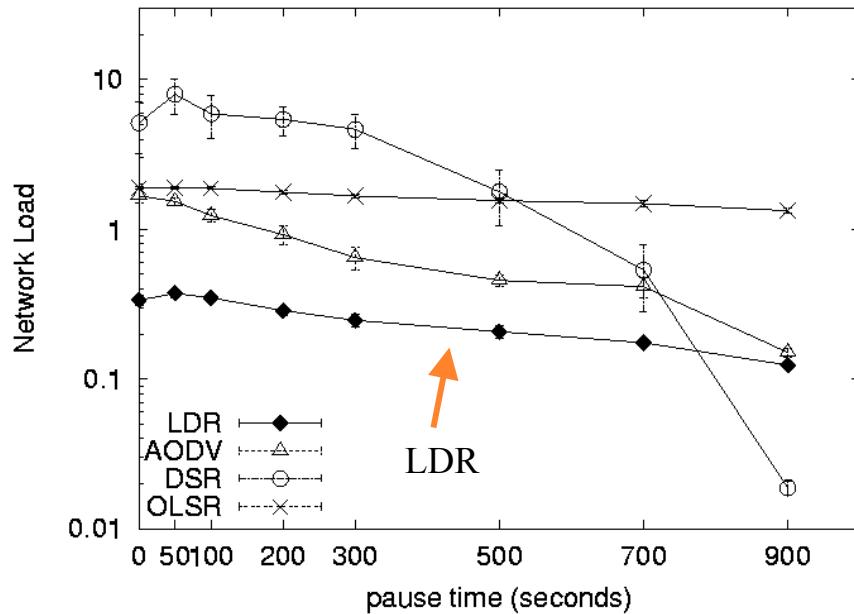
- Mobile-node scenarios
 - 50-node on 1500x300 m topology
 - 100-node on 2500x600 m topology
 - GloMoSim, 802.11, random-waypoint
 - 512-byte CBR, 4 pps, 10-flow & 30-flow
 - Flow length mean 60s, exp. variate length
 - Compare LDR, AODV, DSR, and OLSR

Delivery Ratio

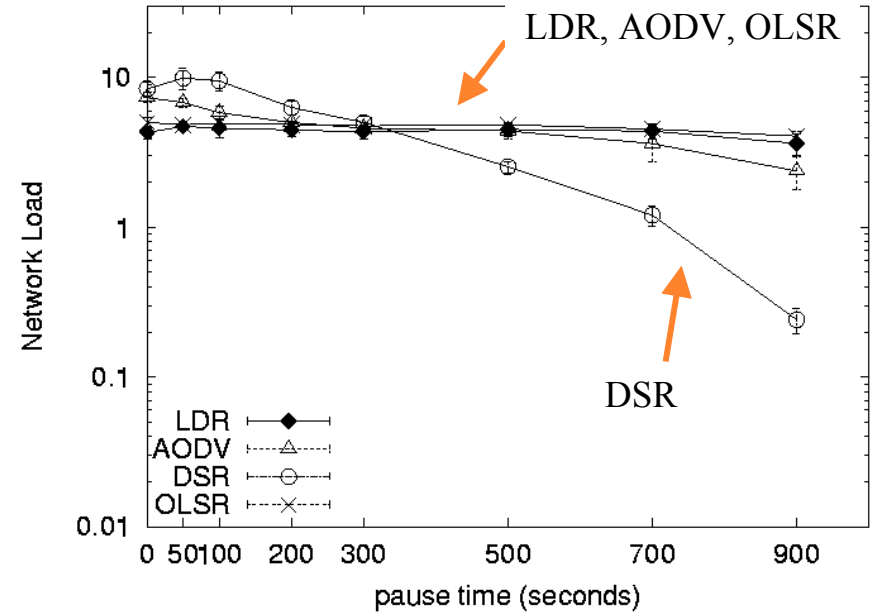


At low load, LDR has significantly higher delivery ratio. At high load, LDR best except at no mobility and high mobility.

Control Overhead



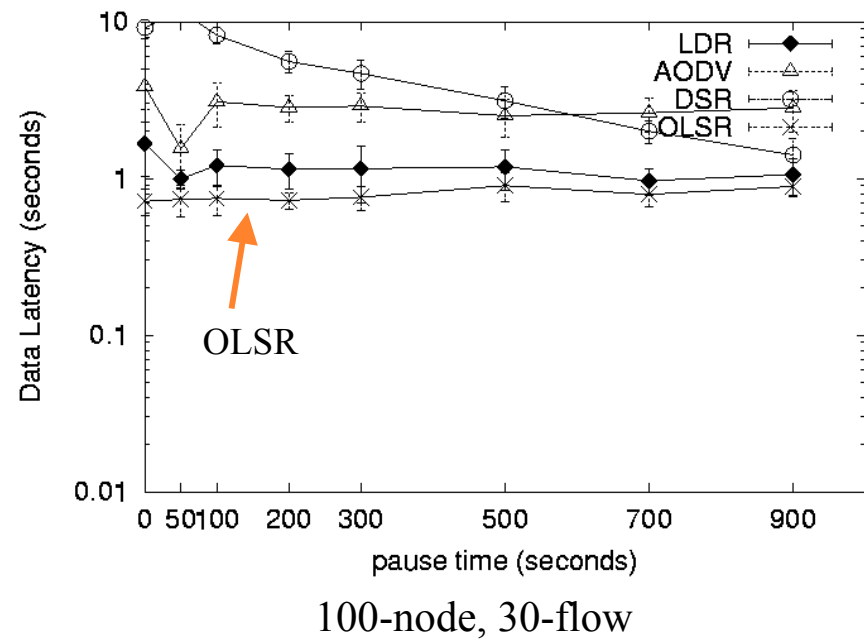
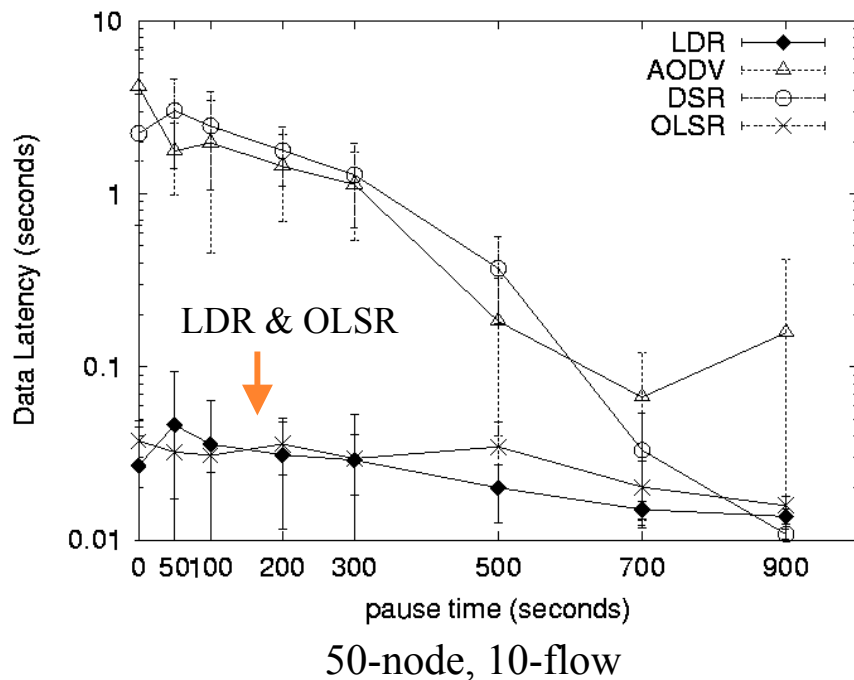
50-node, 10-flow



100-node, 30-flow

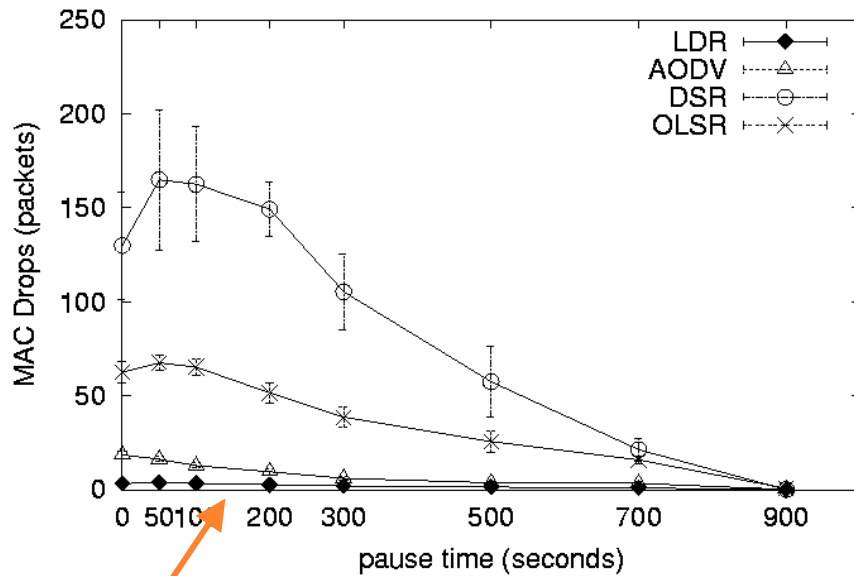
At low load, LDR generally least overhead. At high load, LDR, AODV, & LSR same, DSR lowest with low mobility.

Data Packet Latency

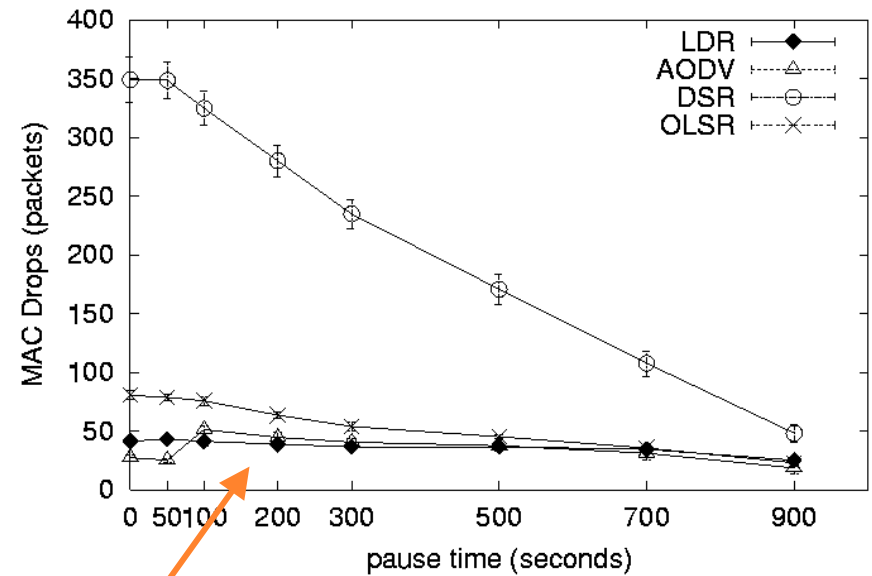


At low load, LDR & OLSR indistinguishable. At high load, OLSR slightly lower latency, and it is a pro-active protocol.

MAC layer unicast drops



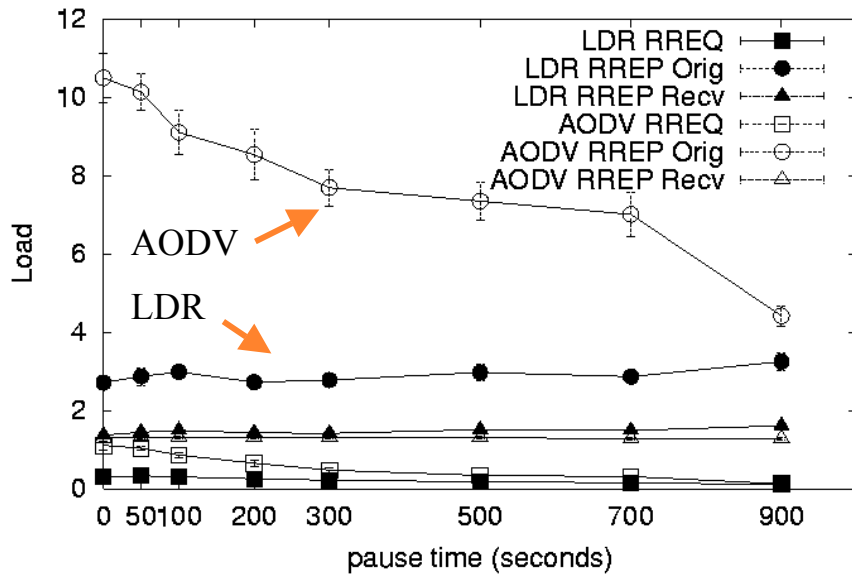
LDR 50-node, 10-flow



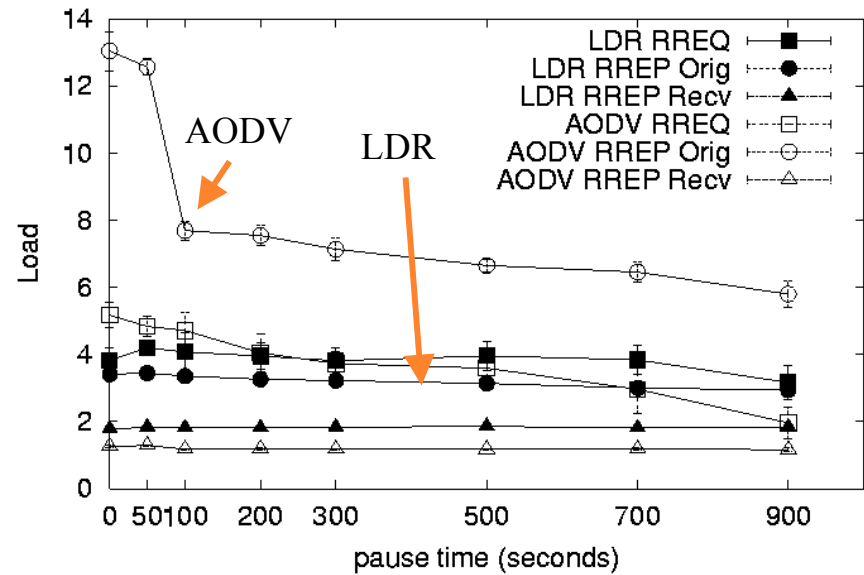
AODV & LDR 100-node, 30-flow

LDR & AODV have fewest MAC-layer drops. MAC-layer drops caused by 802.11 multiple retry failure from collisions.

RREQ Load



50-node, 10-flow



100-node, 30-flow

LDR originates far fewer RREPs for roughly the same number of RREQs. LDR has more locality of repair.

Conclusion

- LDR uses two invariants
 - Feasible distance from DUAL
 - Sequence number from AODV
- Performance
 - Better delivery ratio and network load than other protocols
 - Latency almost as good as OLSR
 - Few MAC-layer drops
 - Many fewer RREPs for same # RREQs than AODV