Routing in Wireless Mesh Nets

- Characteristics
- Concepts
- Metrics
- Protocols
- 802.11s
- Optimisations

Routing in WMNs, Myungchul Kim mck@kaist.ac.kr

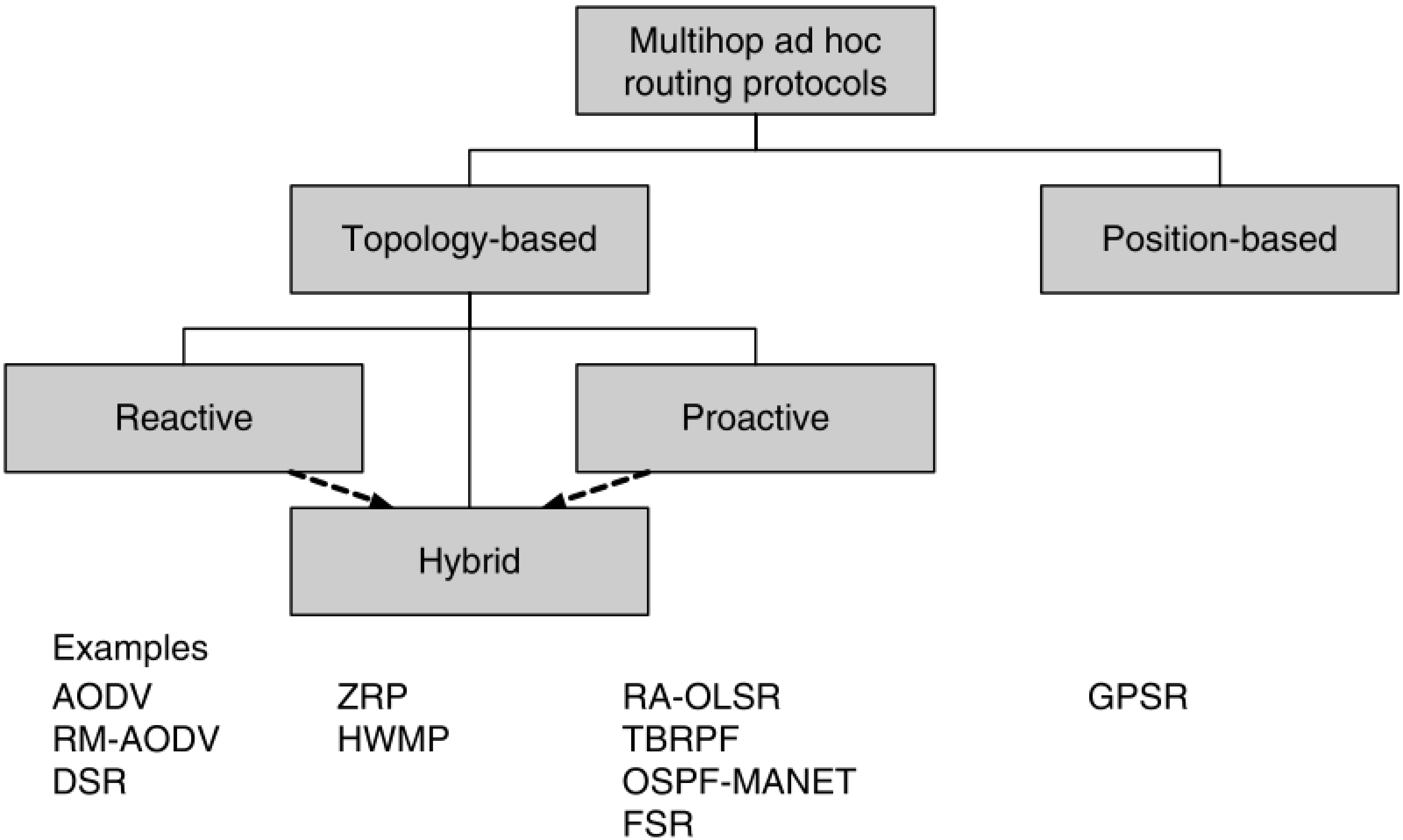
Introduction

- The routing protocols developed for MANETs can be applied to WMNs
- The design of routing protocols for WMNs
  - Either stationary or have min mobility and do not rely on batteries
  - The distance between nodes might be shorted in a WMN -> interference -> bandwidth -> new routing metrics
  - Channel or radio on the path ?
  - Cross-layer design between routing and channel

Special properties of WMNs

- Reliability and network performance
- Types of WMNs
  - Infrastructure mesh networks
  - Client mesh networks
  - Hybrid mesh networks

General concepts of routing protocols



General concepts of routing protocols

- Routing on layer 2
  - Adv: faster access to layer 2 and physical layer, faster forwarding, improvement of media access
  - Disadv:

- Requirements on routing in WMNs
  - Fault tolerance: survivability
  - Load balancing
  - Reduction of routing overhead
  - Scalability
  - QoS support

General concepts of routing protocols

- Multipath routing for load balancing and fault tolerance
  - Resiliency
- QoS routing
  - A route is affected by the interference of other routes in the network
  - Interflow interference vs intraflow interference
  - Intraflow interference is more difficult. Why?

Routing metrics

- Requirements of routing metrics for WMN
  - Ensuring route stability
  - Determined min cost/weight paths have good performance
  - Efficient algorithms
  - Ensuring loop free forwarding
- Hopcount
- Expected Transmission Count(ETC)
- Expected Transmission Time(ETT)
- Weighted Cumulative Expected Transmission Time(WCETT): intraflow interference
- Metric of Interference and Channel-switching(MIC):interflow interference

Routing metrics

- Airtime Link Metric
  - A measure for the consumed channel resources when transmitting a frame over a certain link
  - For IEEE 802.11s WLAN mesh networking

Routing protocols

- Ad hoc on-demand distance vector routing protocol (AODV)
  - Reactive routing protocol
  - Routing overhead and latency
  - Sequence number -> freedom of routing loops and "counting to infinity" problem

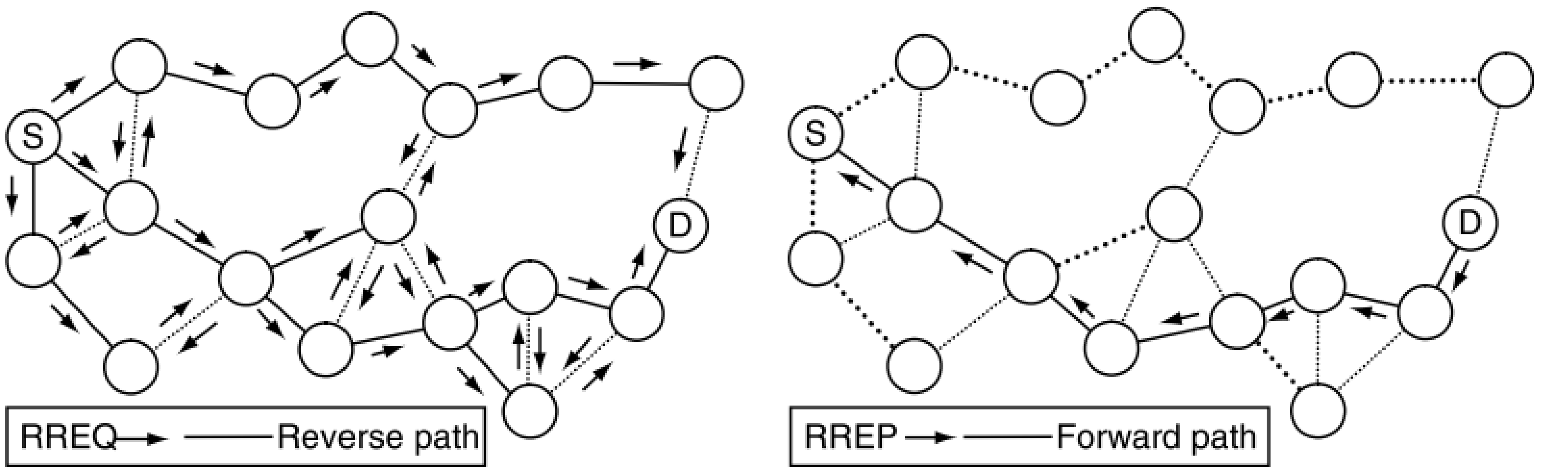


Figure 4.2 **AODV route discovery: route request (*left*) and route reply (*right*).**

Routing protocols

- Ad hoc on-demand distance vector routing protocol (AODV)

When a node receives a RREQ packet, it processes as follows:

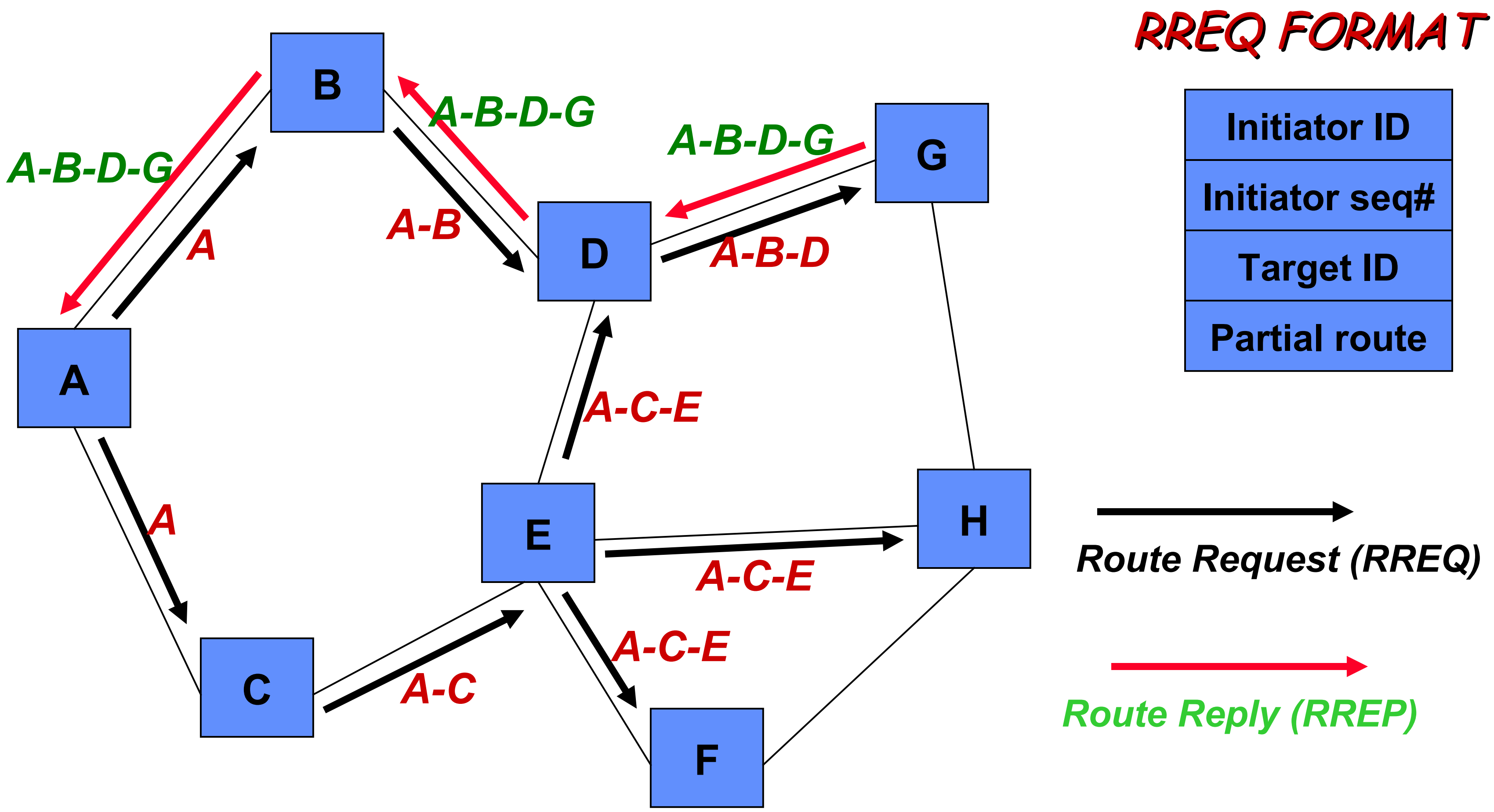
- The route to the previous hop from which the RREQ packet has been received is created or updated.
- The RREQ ID and the originator address are checked to see whether this RREQ has been already received. If yes, the packet is discarded.
- The hopcount is incremented by 1.
- The reverse route to the originator, node *S*, is created or updated.

Routing protocols

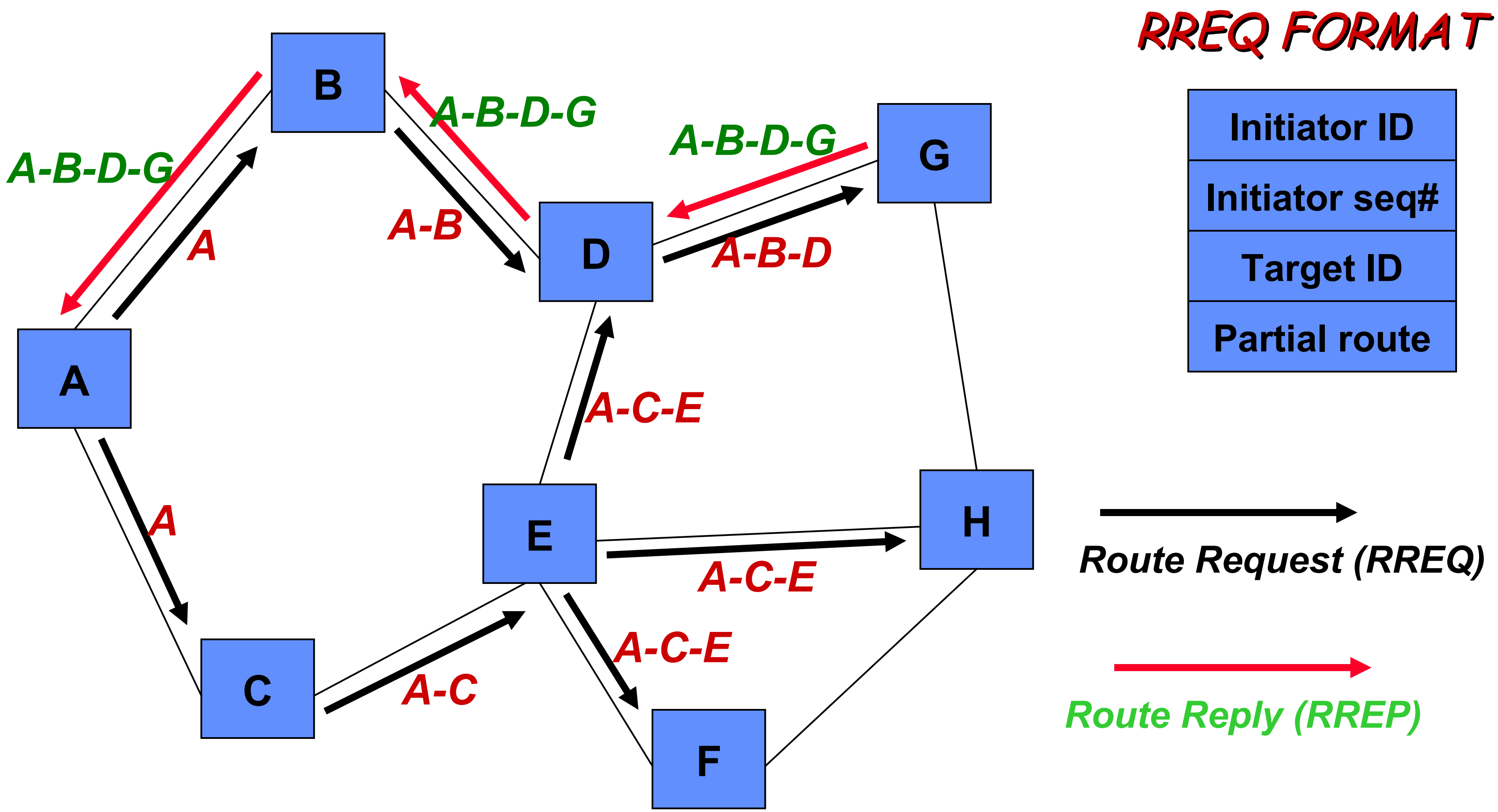
- If the node is the requested destination, it generates a route reply (RREP) and sends the RREP packet back to the originator along the created reverse path to the source node *S*.
- If the node is not the destination but has a valid path to *D*, it issues a RREP to the source depending on the *destination only flag*.

  If intermediate nodes reply to RREQs, it might be the case that the destination will not hear any RREQ, so that it does not have a back route to the source. If the *gratuitous* RREP flag is set in the RREQ, the replying intermediate node will send a gratuitous RREP to the destination. This sets the path to the originator of the RREQ in the destination.
- If the node does not generate a RREP, the RREQ is updated and rebroadcast if TTL is ≥ 1.

Route Discovery in DSR

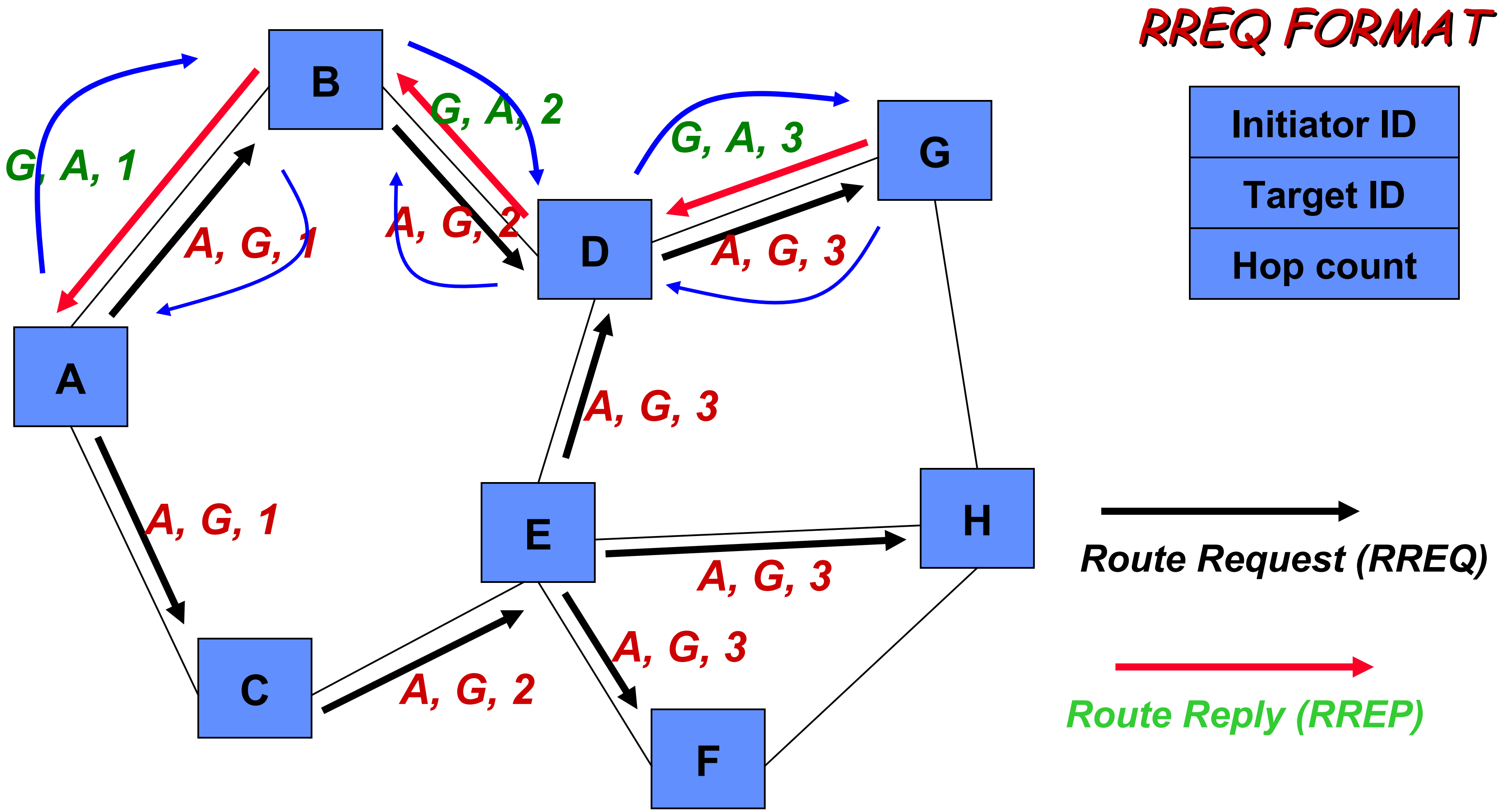


- **Each node** **appends own identifier** **when forwarding RREQ**



- **Each node appends own identifier when forwarding RREQ**

Route Discovery in AODV



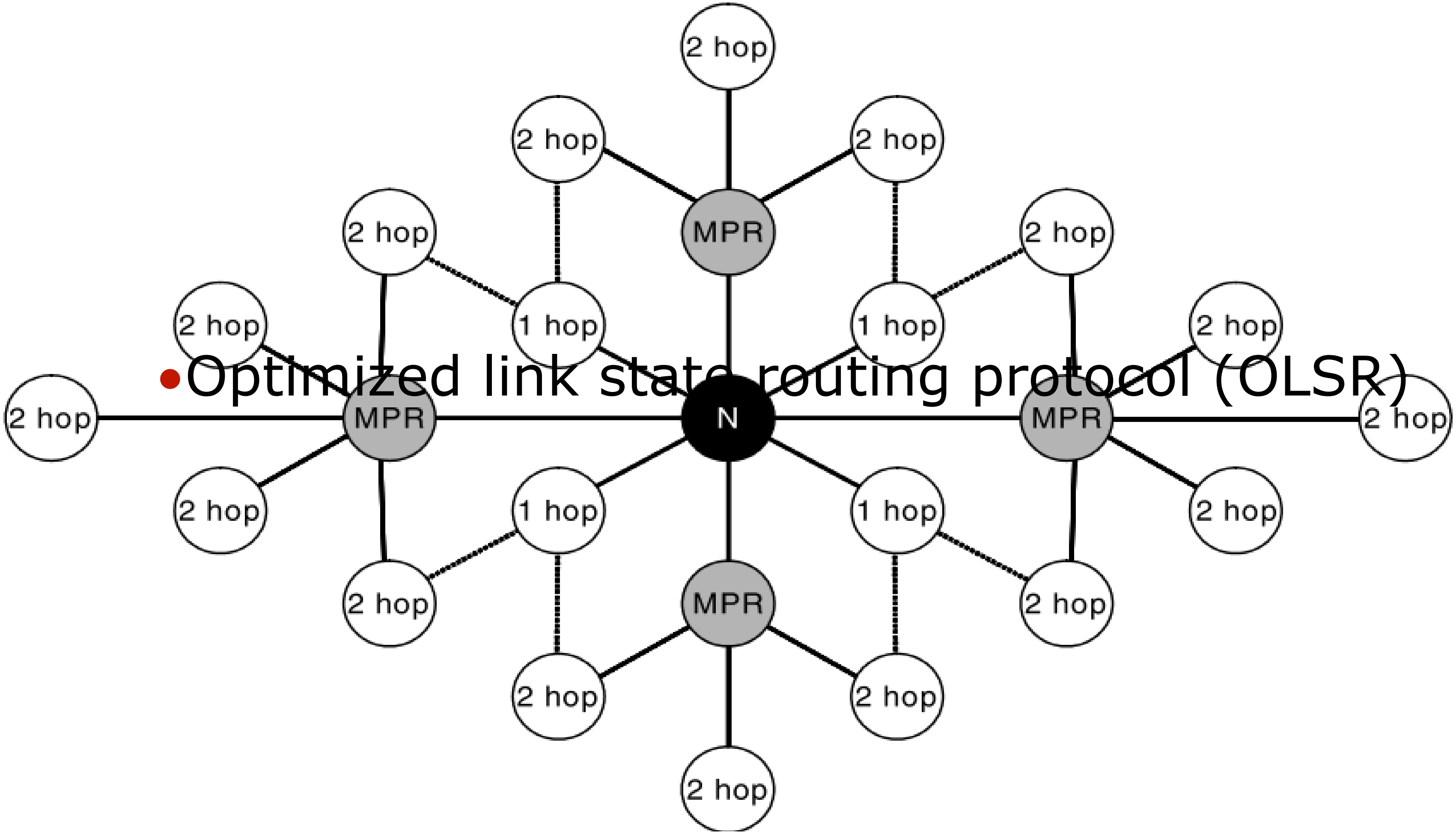
- **Each node remembers where the packet came from (reverse path) during RREQ.**
- **It set up forward path during RREP.**

Routing protocols

- Optimized link state routing protocol (OLSR)
  - Proactive routing protocol
  - The shortest path algorithm based on the hop count metric
  - Optimized broadcast mechanism for the network-wide distribution of the necessary link-state information
  - Multipoint relays (MPRs): 2-hop neighbors receive broadcast messages even if only the MPRs rebroadcast the message
  - MPR selection
  - Topology control messages: link state information

Routing protocols

- Optimized link state routing protocol (OLSR)



Routing protocols

- Cross-layer routing approach
  - Message routing strategy (MRS): find high throughput paths with reduced interference and increased reliability by optimally controlling transmission power.

- Initially, through neighbor discovery protocol, each node explores its neighborhood, calculates the metrics, such as transmission rate, interference and PER, and determines the local transmission. After that, local links are advertised.
- Whenever an event triggers a change in the routing metrics of one or more links, the power optimization is performed on the concerned link and the route update process is started.
- Once the best metric is identified (which is a relatively stable condition), the link is advertised. The MRS routing protocol selects optimal paths to reach any other wireless mesh router of the network by a distance vector approach.

Routing protocols

- Bandwidth aware routing
  - If the traffic for connection request is splitable, i.e., using multiple routes to satisfy the bandwidth request, linear programming formulation.
  - If the traffic for a single route, identify the max bottleneck capacity for a route.
- Multi-Radio Link-Quality Source Routing (MR-LQSR)

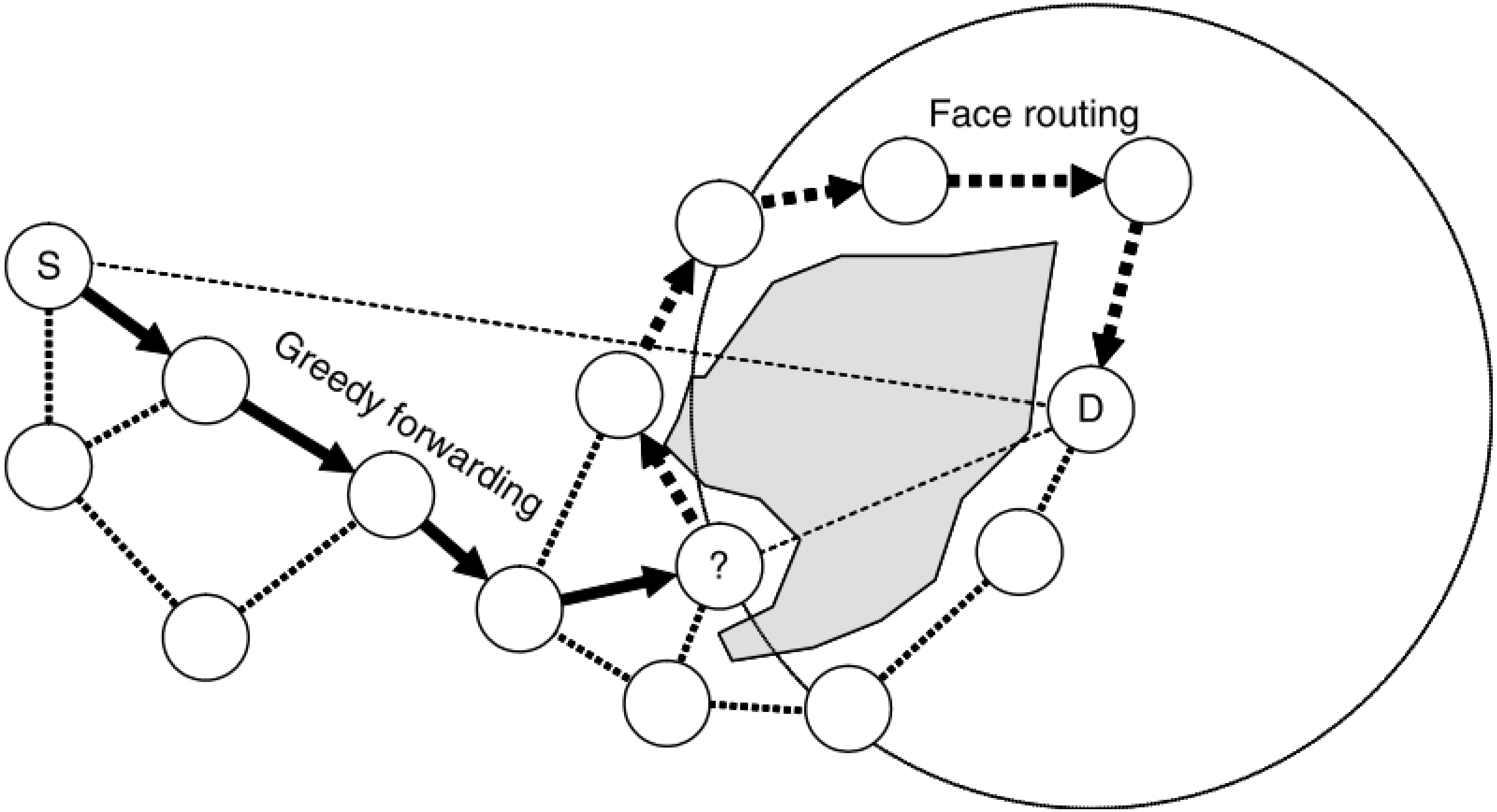
$$\mathrm{WCETT} = (1-\beta) \times \sum_{i=1}^{n} \mathrm{ETT}_i + \beta \times \max_{1 \le j \le k} X_j$$

  - WCETT
  - Balance trade-off between throughput and delay

Routing protocols

- Other topology-based routing protocols for WMN
  - TBRPF: proactive
  - DYMO: reactive and similar to AODV
  - OSPF-MANET: the flooding of link sate advertisement is reduced.
  - FSR: proactive
  - ZRP: proactive and reactive
- Position-based routing protocols
  - Geographical positions
  - Greedy Perimeter Stateless Routing (GPRS): greedy forwarding with face routing as fallback
  - Landmark guided forwarding: topology-based proactive routing for nearby nodes with position-based forwarding for faraway nodes.

Routing protocols



Proposed routing for IEEE 802.11s WLAN Mesh Networking

- IEEE 802.11s
    - Started in 2004 and still in progress
    - 50 mesh nodes
    - Routing, MAC enhancements and security
    - Routing at layer 2
    - Legacy IEEE 802.11 devices can connect to a mesh through mesh access points using the conventional methods of the standard.

- Airtime routing metric
    - Reflect the amount of channel resources consumed for transmitting a frame over a particular link
    - The path with the smallest sum of airtime link metrics is the best path

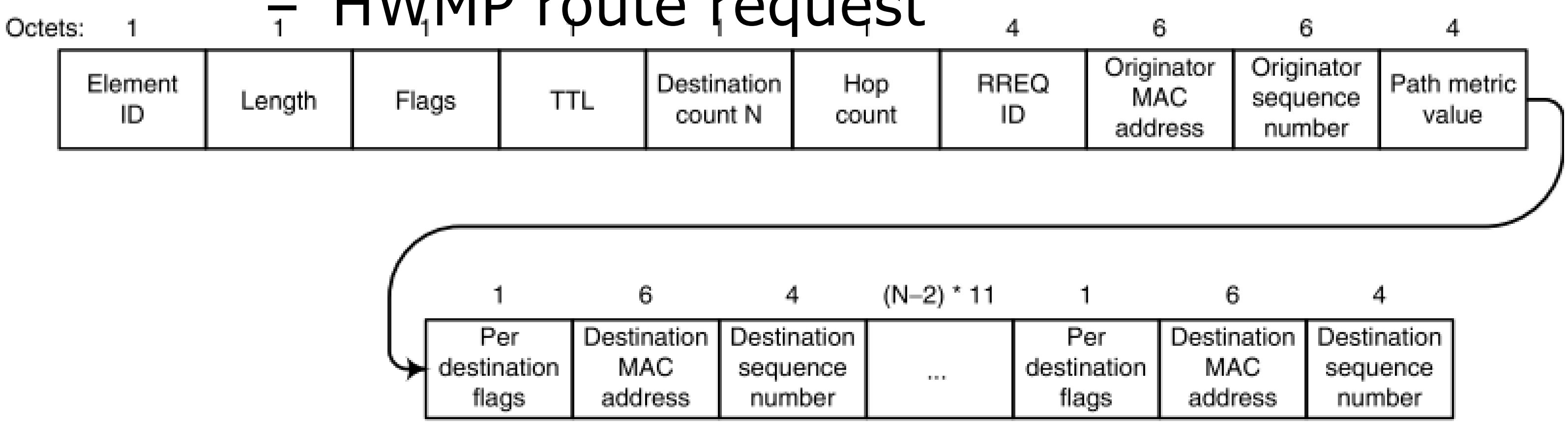
Proposed routing for IEEE 802.11s WLAN Mesh Networking

$$c_a = \left[O_{ca} + O_p + \frac{B_t}{r} \right] \frac{1}{1 - e_{fr}}$$

The channel access overhead O_{ca}, the MAC protocol overhead O_p, and the number of bits B_t in a test frame are constants whose values depend on the used IEEE 802.11 transmission technology. The transmission bit rate r in Mbit/s is the rate at which the mesh node would transmit a frame of size B_t based on the current conditions with the frame error rate e_{fr}.

Proposed routing for IEEE 802.11s WLAN Mesh Networking

- Hybrid wireless mesh protocol (HWMP)
  - The default routing protocol for WMAN
  - Radio metric AODV
  - Use MAC addresses as a layer 2 routing protocol
  - Radio-aware metrics
  - Multiple destination in RREQ
  - HWMP route request

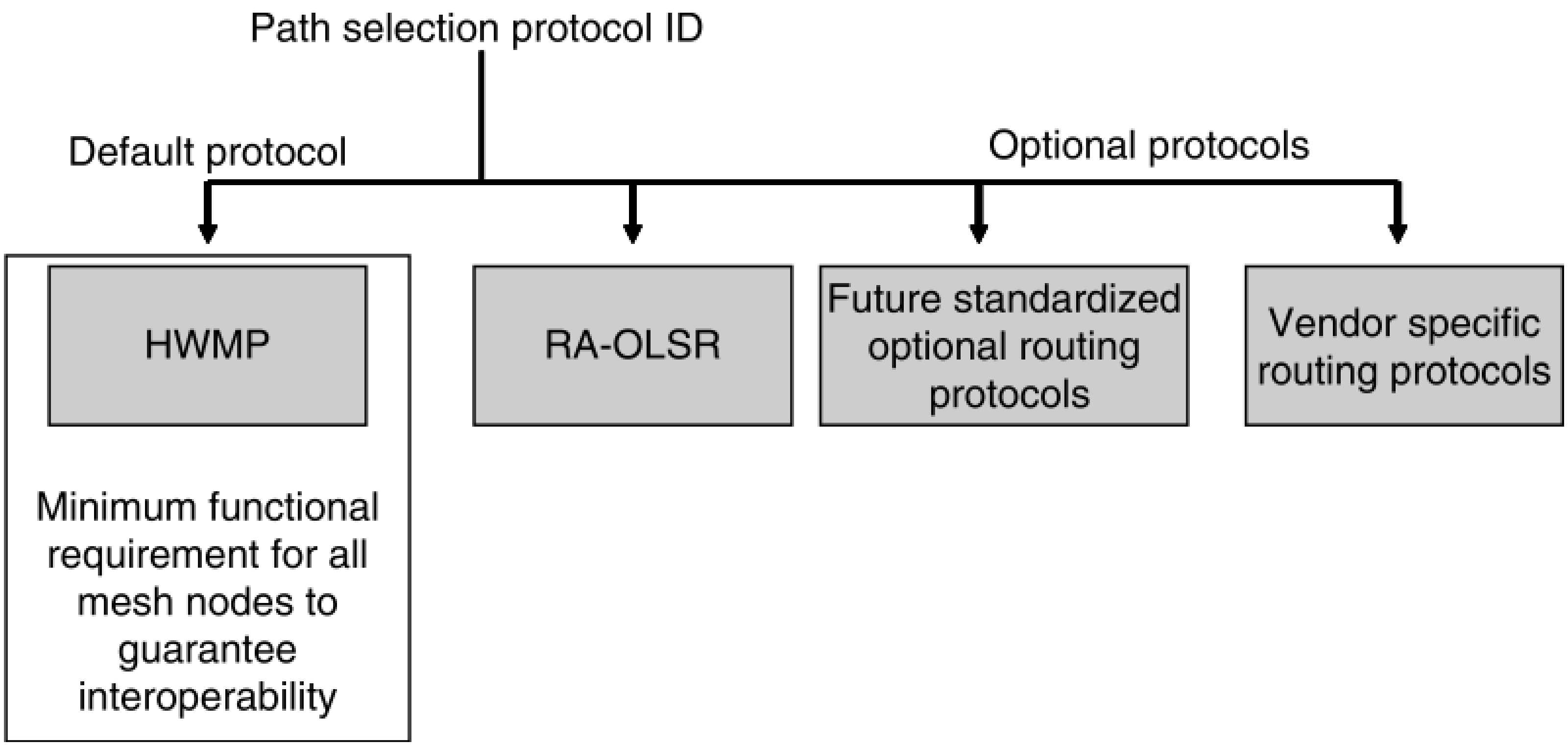


Flags

Proposed routing for IEEE 802.11s WLAN Mesh Networking

- Radio aware optimized link state routing (RA-OLSR)
  - An optional proactive routing protocol of the emerging IEEE 802.11s
  - Use MAC address
  - Shortest path computation
- Extensibility
  - Not a single routing protocol that is optimal in every useful scenario
  - Each mesh network announces its used routing protocol and routing metric to new nodes by corresponding IDs.

Proposed routing for IEEE 802.11s WLAN Mesh Networking



Joint routing and channel assignment

- Load-aware channel assignment: routing first and followed by channel assignment since the load in each link is given.
- Interference-aware routing: channel assignment first and followed by routing since the interference among links is determined.
- In a multiradio multichannel WMN, routing and channel assignment are interdependent.

Joint routing and channel assignment

- Combined load-aware routing and channel assignment
  - An initial estimation of the expected load on each virtual link -> channel assignment and routing algorithms

- Step 1: The routing algorithm computes the initial routes for each node pair given a set of node pairs and the expected traffic load between each node pair.
- Step 2: Given the input from the routing step, the radio channel assignment algorithm assigns a radio channel to each interface such that the available bandwidth at each virtual link is no less than its expected load.
- Step 3: The new channel assignment is fed back to the routing algorithm to reach more informed routing decisions.
- Step 4: Recalculate the link load based on the routing information. If some of the link loads are more than their capacities, go to Step 2, otherwise, Stop.

Joint routing and channel assignment

- Joint LP-based routing and channel assignment
  - Graph
  - Decisions to make
    - Assign a set of wireless channels to each link
    - Determine whether each (link, channel) pair is active for each time slot and
    - Assign the communication traffic to the active (link, channel) pairs over different time slots.
  - Integer programming formulation and linear programming relaxation.

Evaluation of mesh routing protocols in Wireless Community Networks

Babel

- Bellman-Ford
- Application of routing updates (seq #)
- Neighbour discovery
  - Hello: mcast /4 sec, estimate *rxcost* of link
  - I Heard You: bidir, unicast /12s, share *rxcost*
- Topology dissemination
  - Sharing routing table /16 or async update

BMX6

- Distance-vector, proactive
- Compression
- Neighbour discovery
  - Hello: mcast /0.5 sec, w/ seq#
  - Report: bidir, report #Hello msgs → cost
- Topology dissemination
  - OGM: announce presence and re-sent if good choice /5s

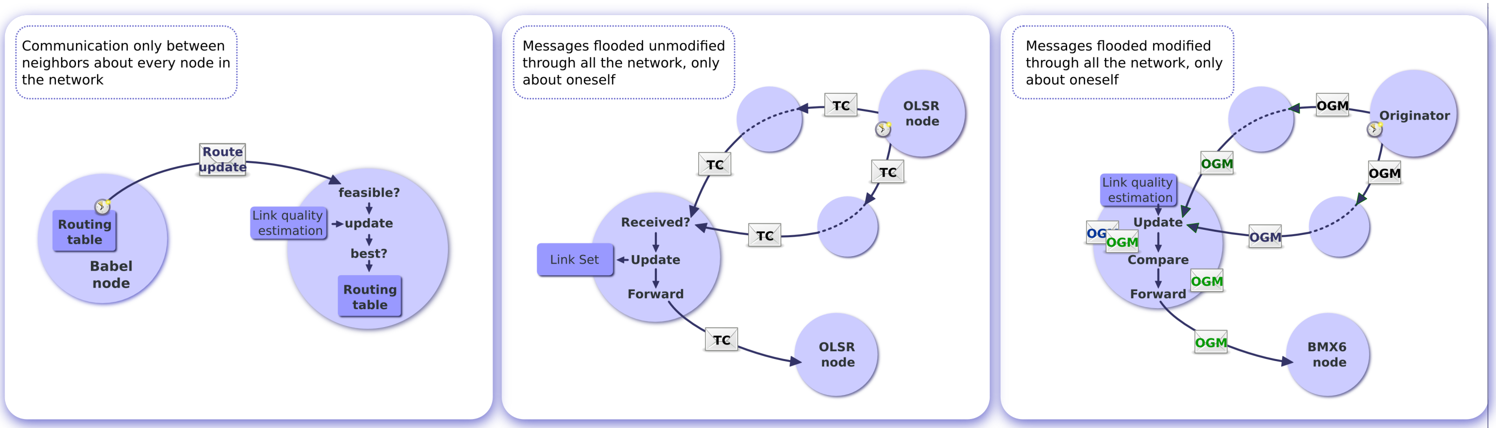
OLSR

- Optimized Link-State: only multi-point relays (MPR) retransmit msgs
- Neighbour discovery
  - Hello: mcast /2 sec, w/ seq#
  - Report: bidir, report #Hello msgs → cost
- Topology dissemination
  - Topology Control: list all nodes reachable + cost in both directions /5s
- Tricks:
  - Fish-eye extension: less TC updates to far away nodes

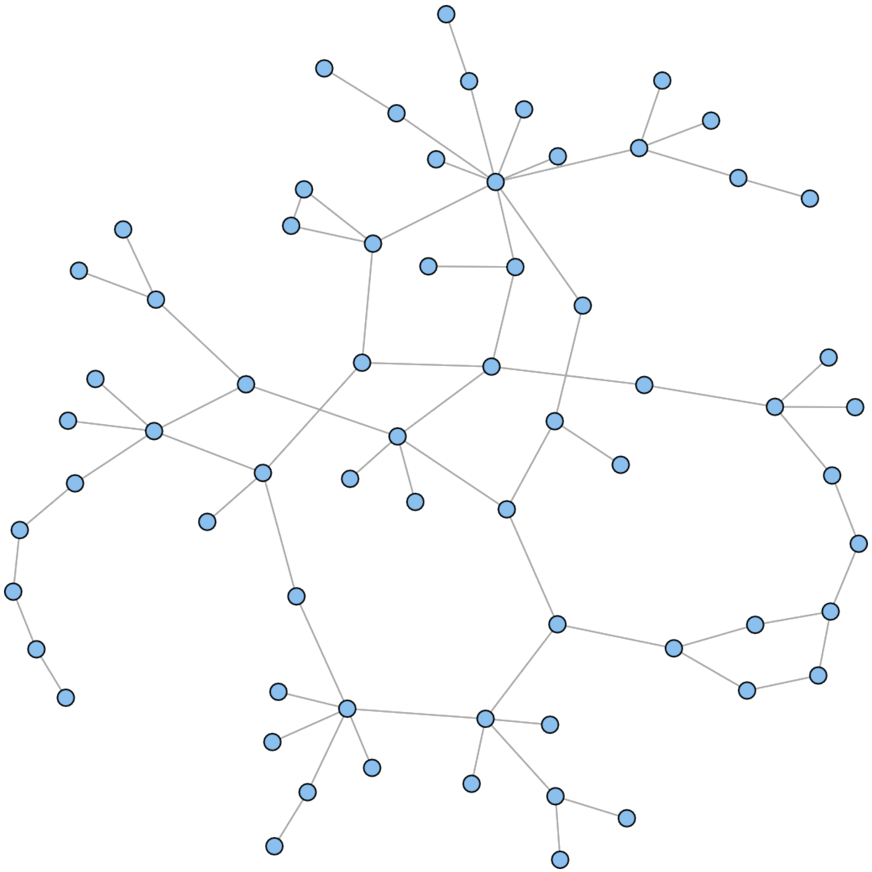
Topology dissemination mechanisms

Table 1: Periodic Messages

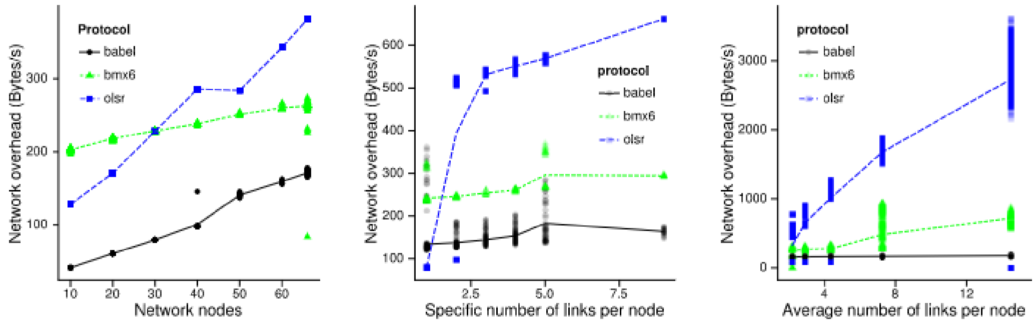
	Message	Size (B)	Interval	Messages/node
Babel	Hello	8	4 s	1
	IHU	16	12 s	$1 \times n$
	Update	$[\,12, 28\,]$	16 s	$1 \times route$
BMX6	Hello	$[\,4, 6\,]$	0.5 s	1
	RP	1	0.5 s	$1 \times n$
	OGM	4	5 s	1
OLSR	Hello	$[\,28, 32\,] + 20 \times n$	2 s	1
	TC	$28 + 20 \times n$	5 s	1



Topology of guifi.net Barcelona county

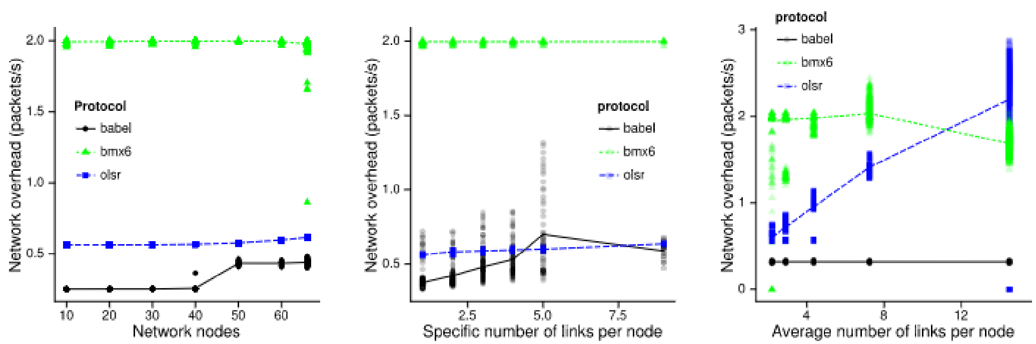


Network overhead

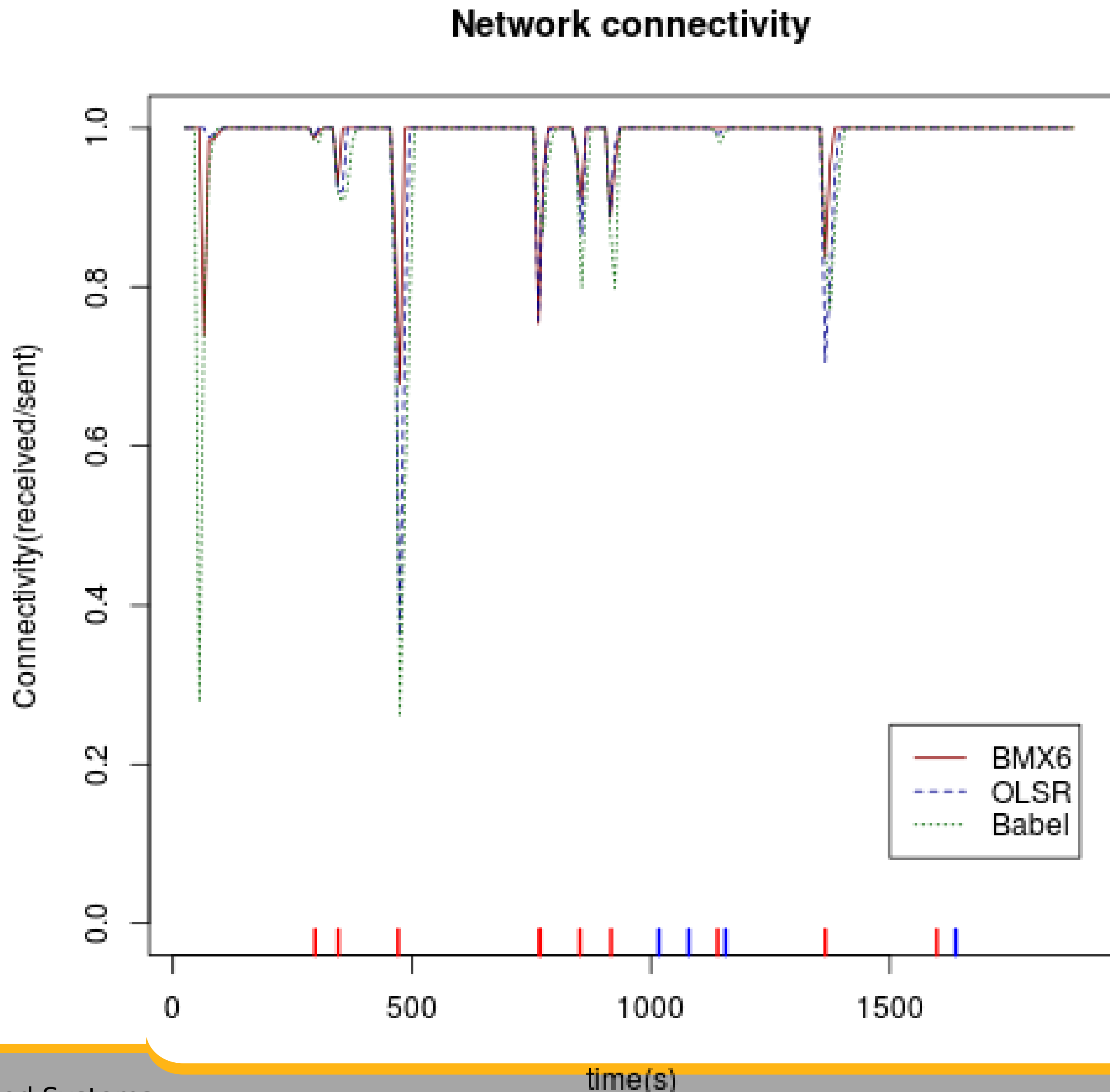


(a) Bytes vs. nodes (b) Bytes vs. specific links (c) Bytes vs. average links

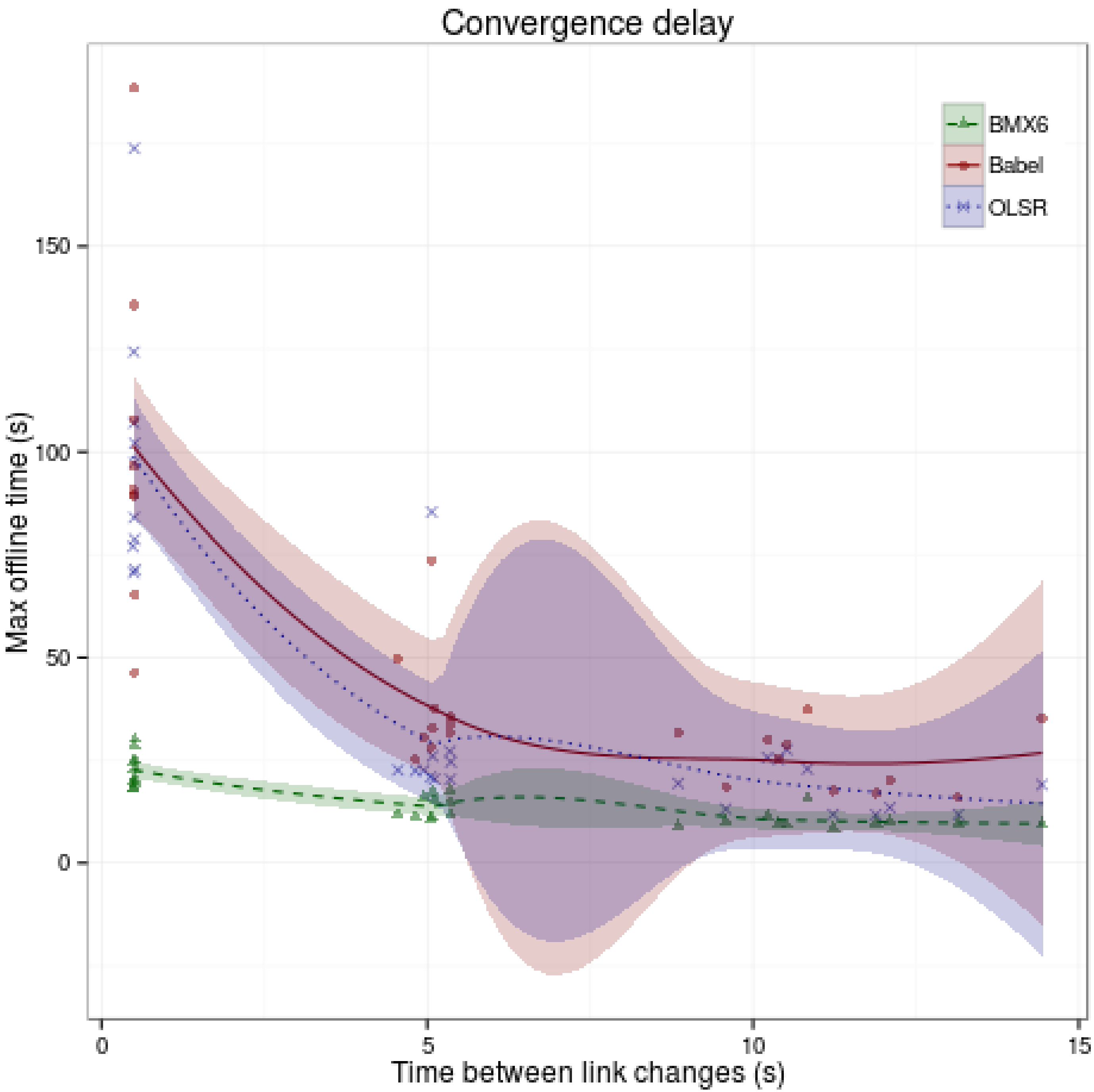
Figure 3: Network overhead in bytes



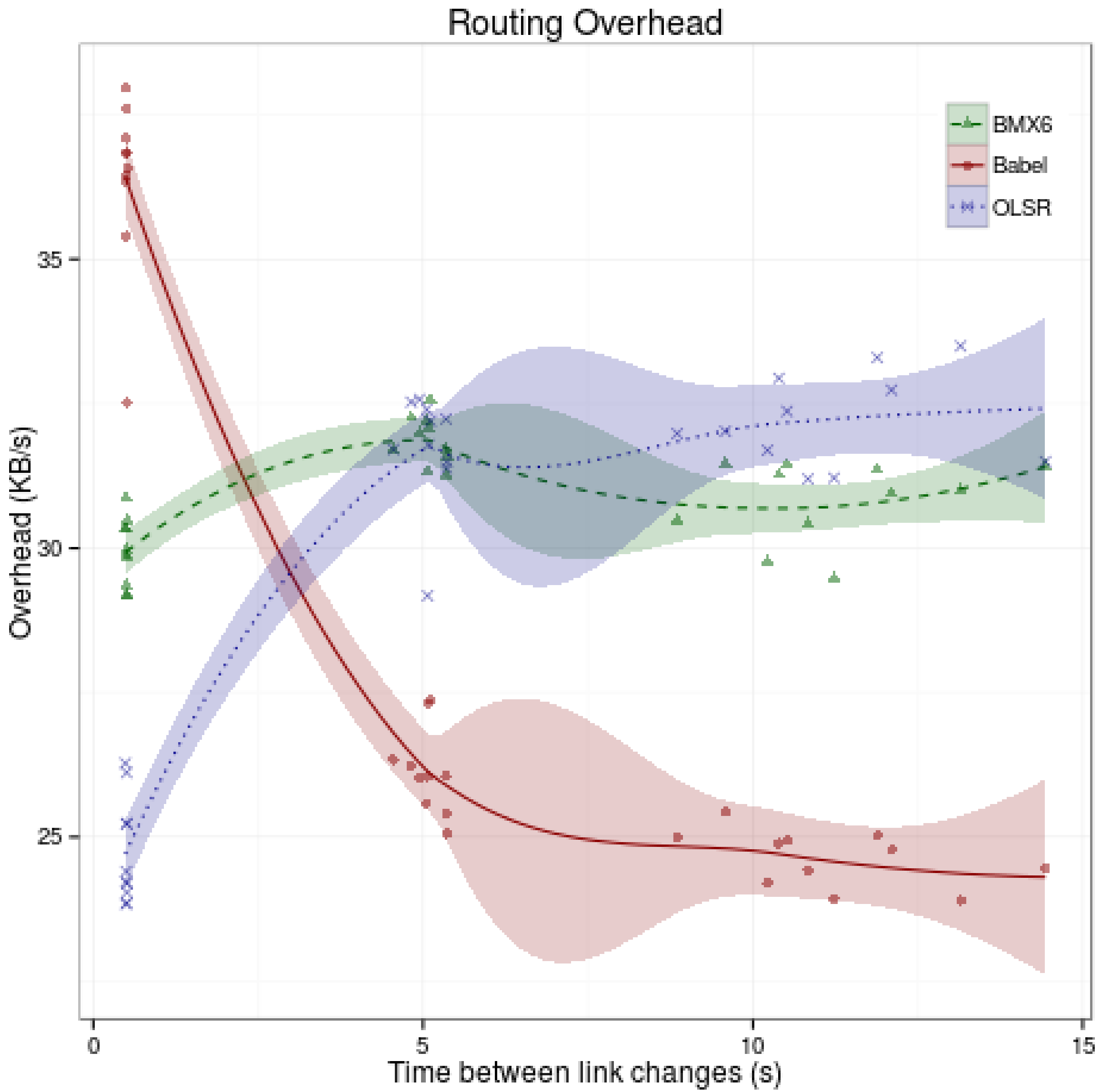
(a) Packets vs. nodes (b) Packets vs. specific links (c) Packets vs. average links

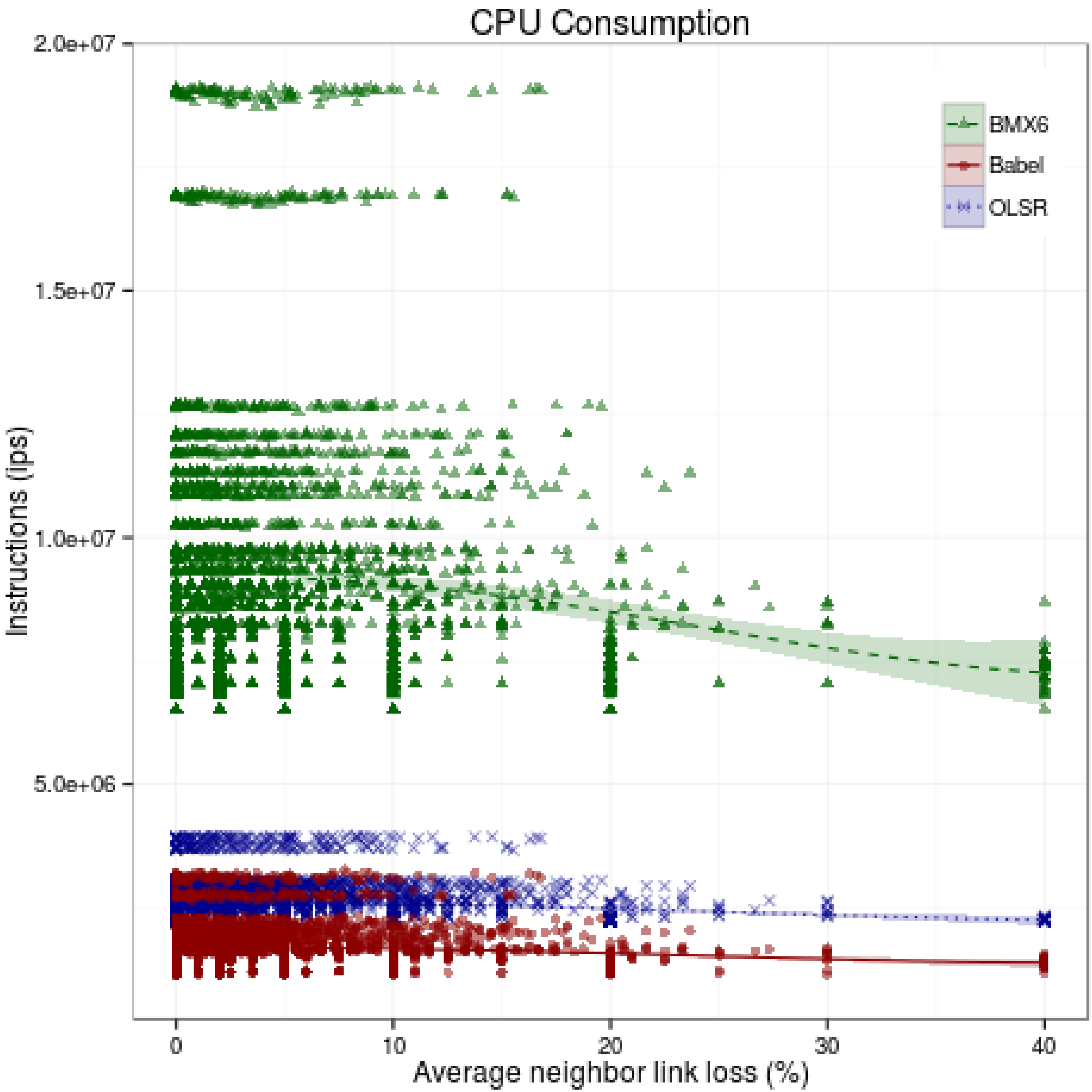


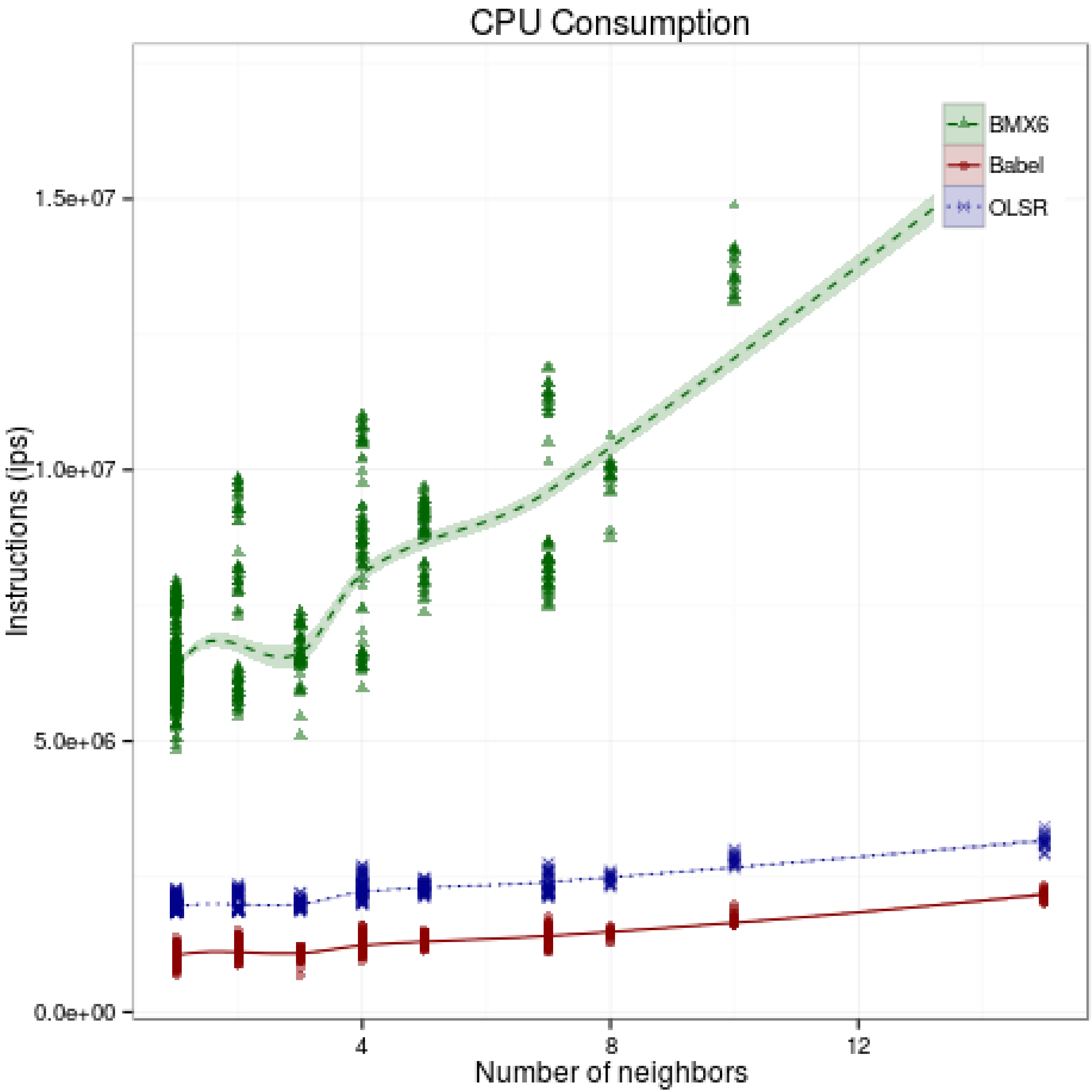
Network connectivity
Connectivity(received/sent)
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
time(s)
BMX6
OLSR
Babel

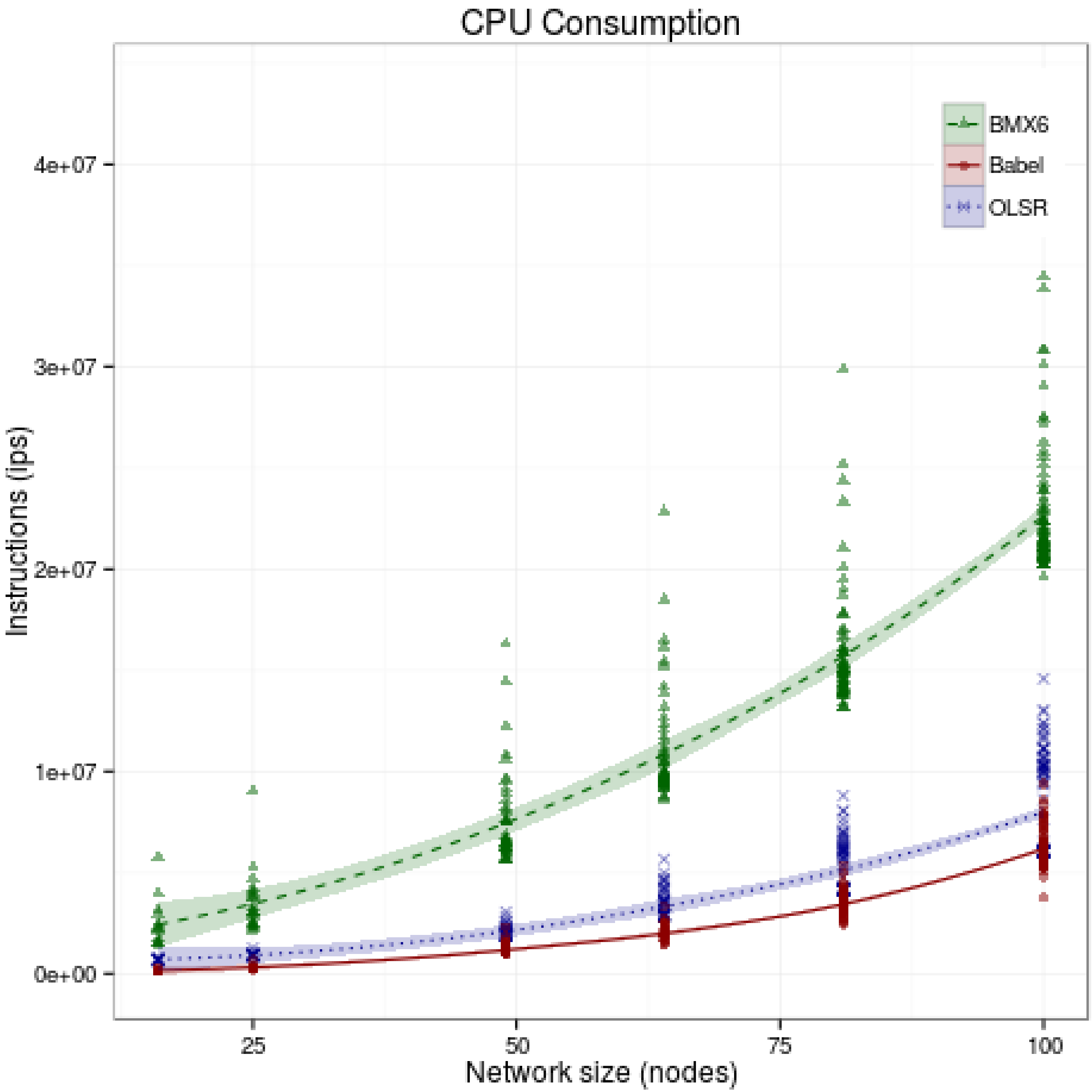


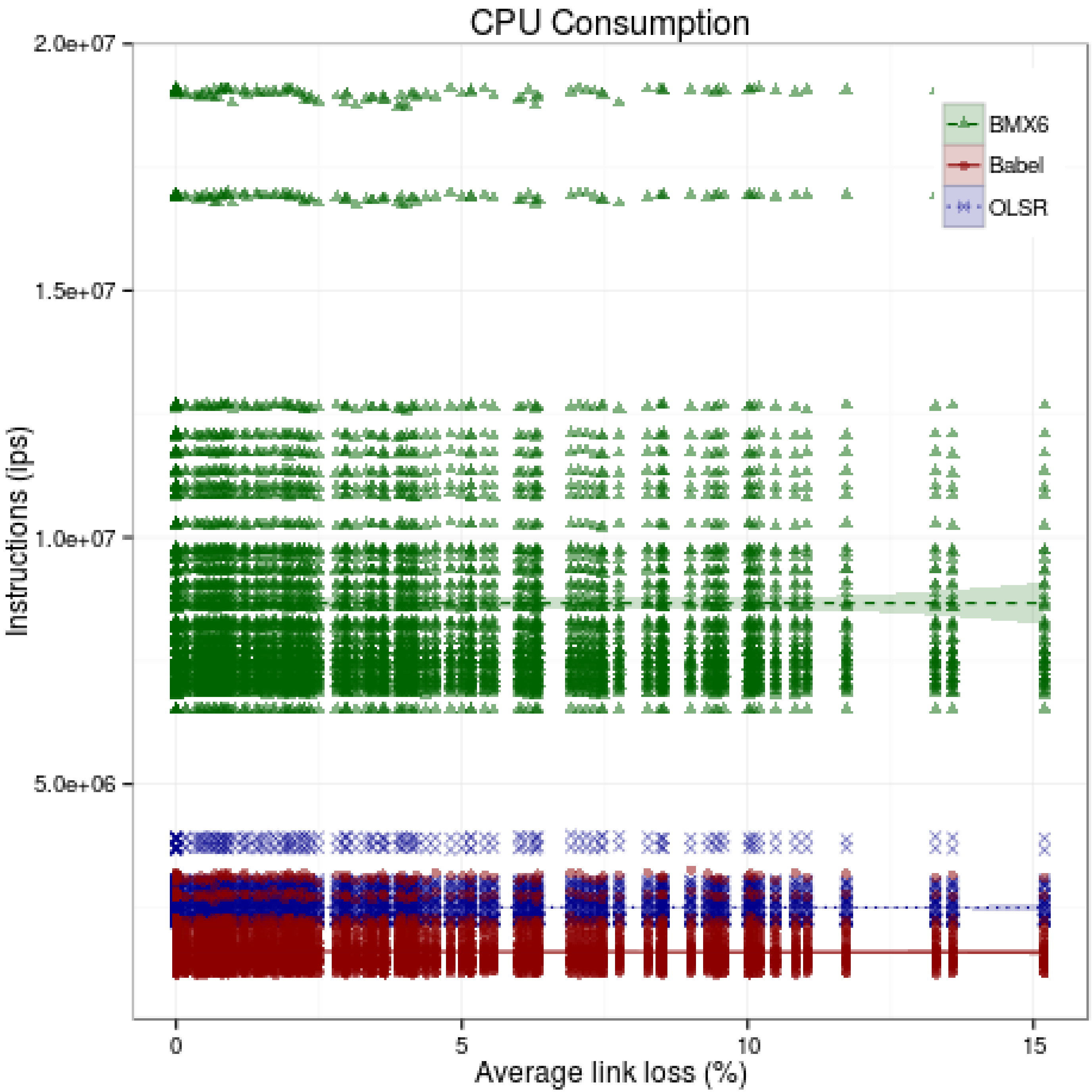
Convergence delay
BMX6
Babel
OLSR
Max offline time (s)
Time between link changes (s)
0
50
100
150
0
5
10
15


Routing Overhead
BMX6
Babel
OLSR
35
30
25
Overhead (KB/s)
0
5
10
15
Time between link changes (s)

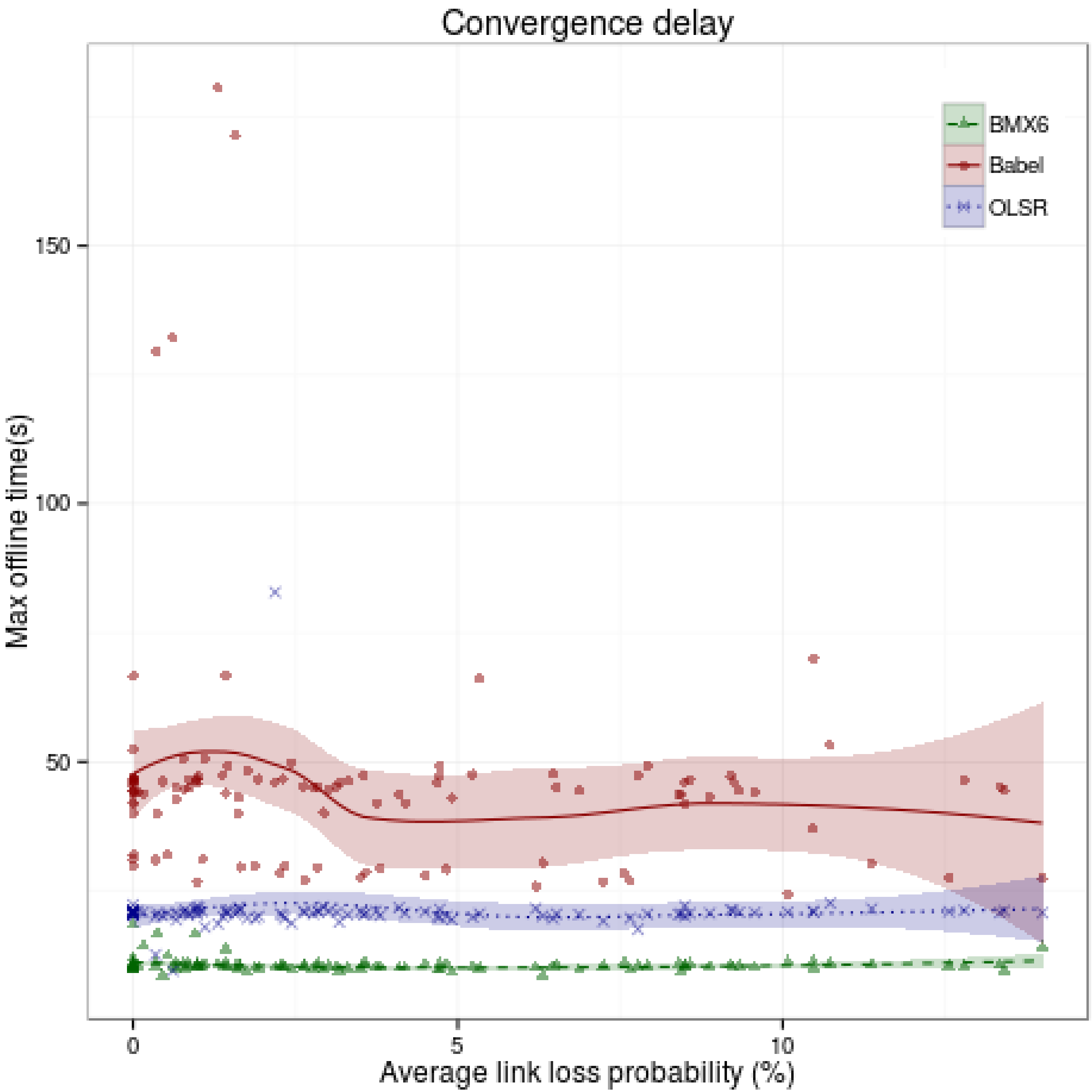

CPU Consumption
2.0e+07
1.5e+07
1.0e+07
5.0e+06
Instructions (ips)
BMX6
Babel
OLSR
0
10
20
30
40
Average neighbor link loss (%)

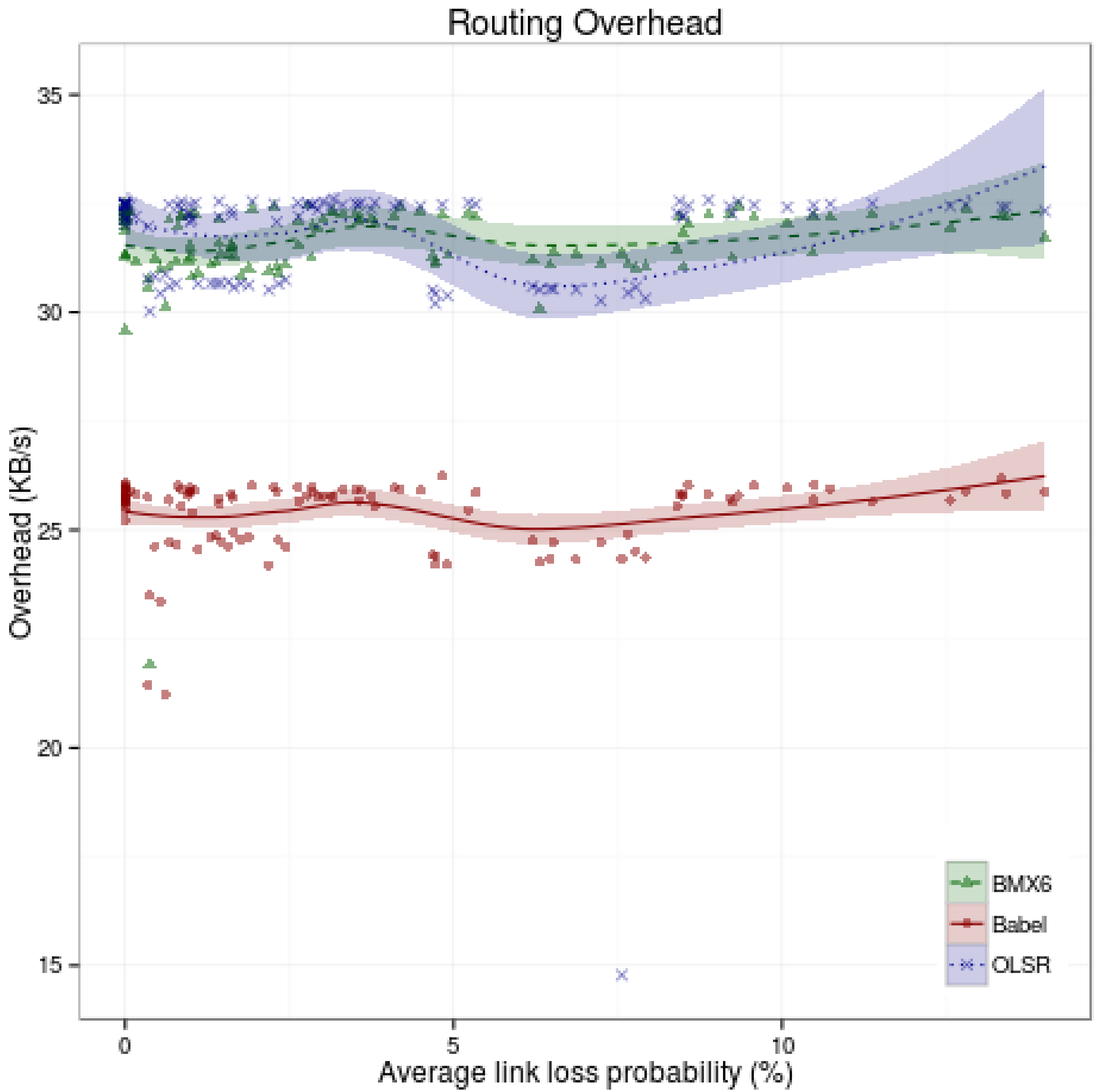

CPU Consumption
BMX6
Babel
OLSR
1.5e+07
1.0e+07
5.0e+06
0.0e+00
Instructions (ips)
4
8
12
Number of neighbors

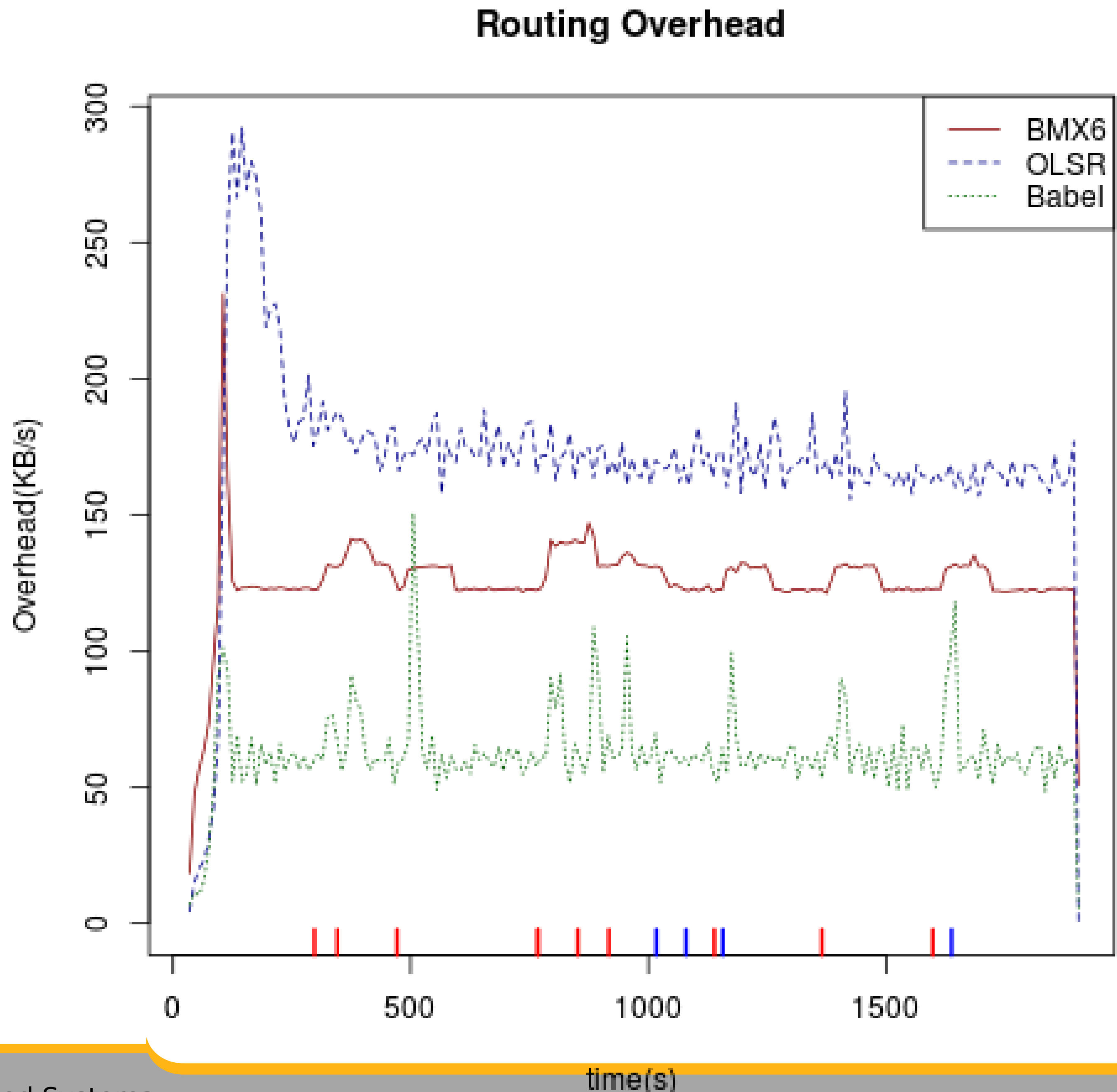

CPU Consumption
BMX6
Babel
OLSR
Instructions (ips)
4e+07
3e+07
2e+07
1e+07
0e+00
25
50
75
100
Network size (nodes)

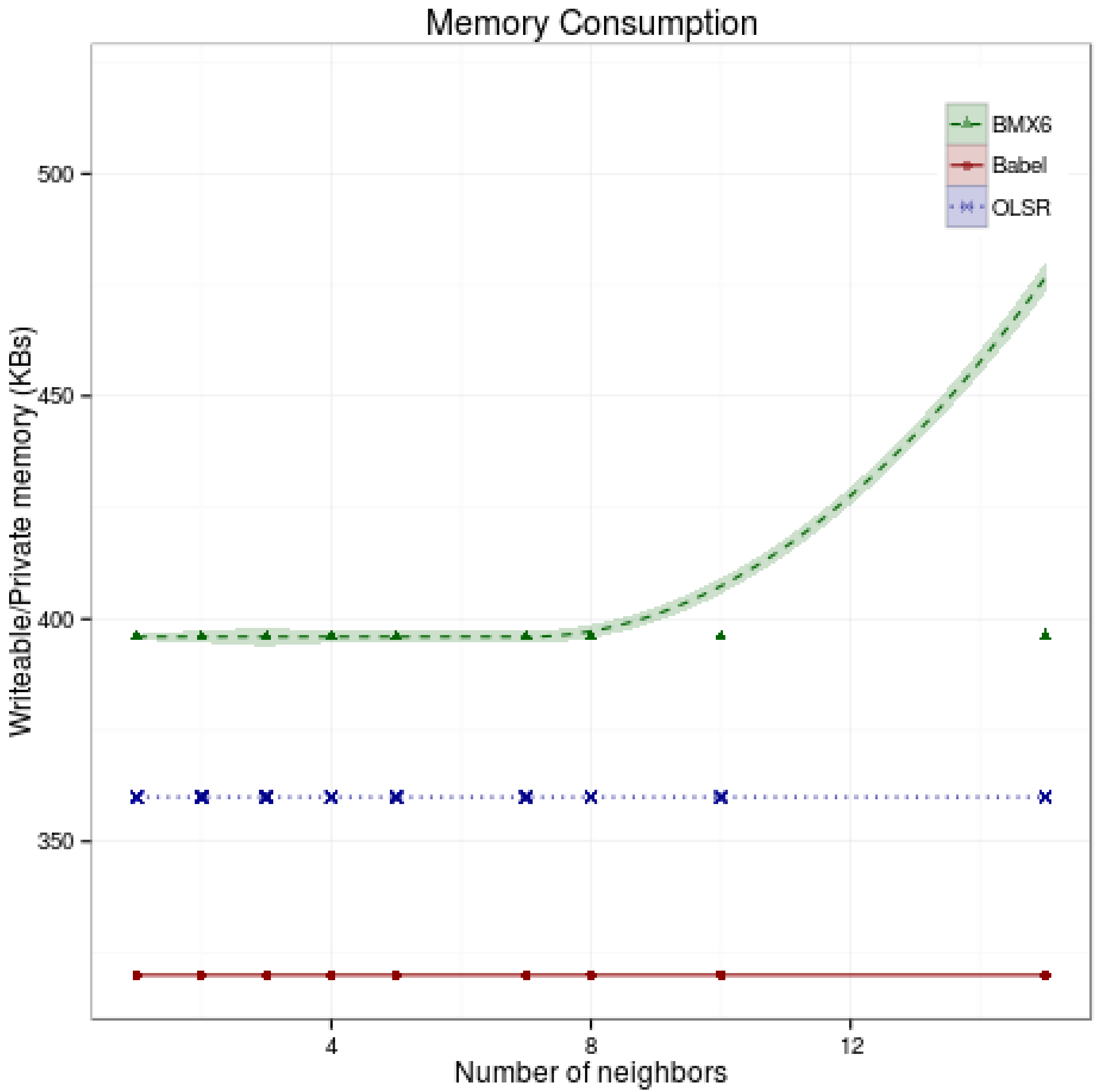


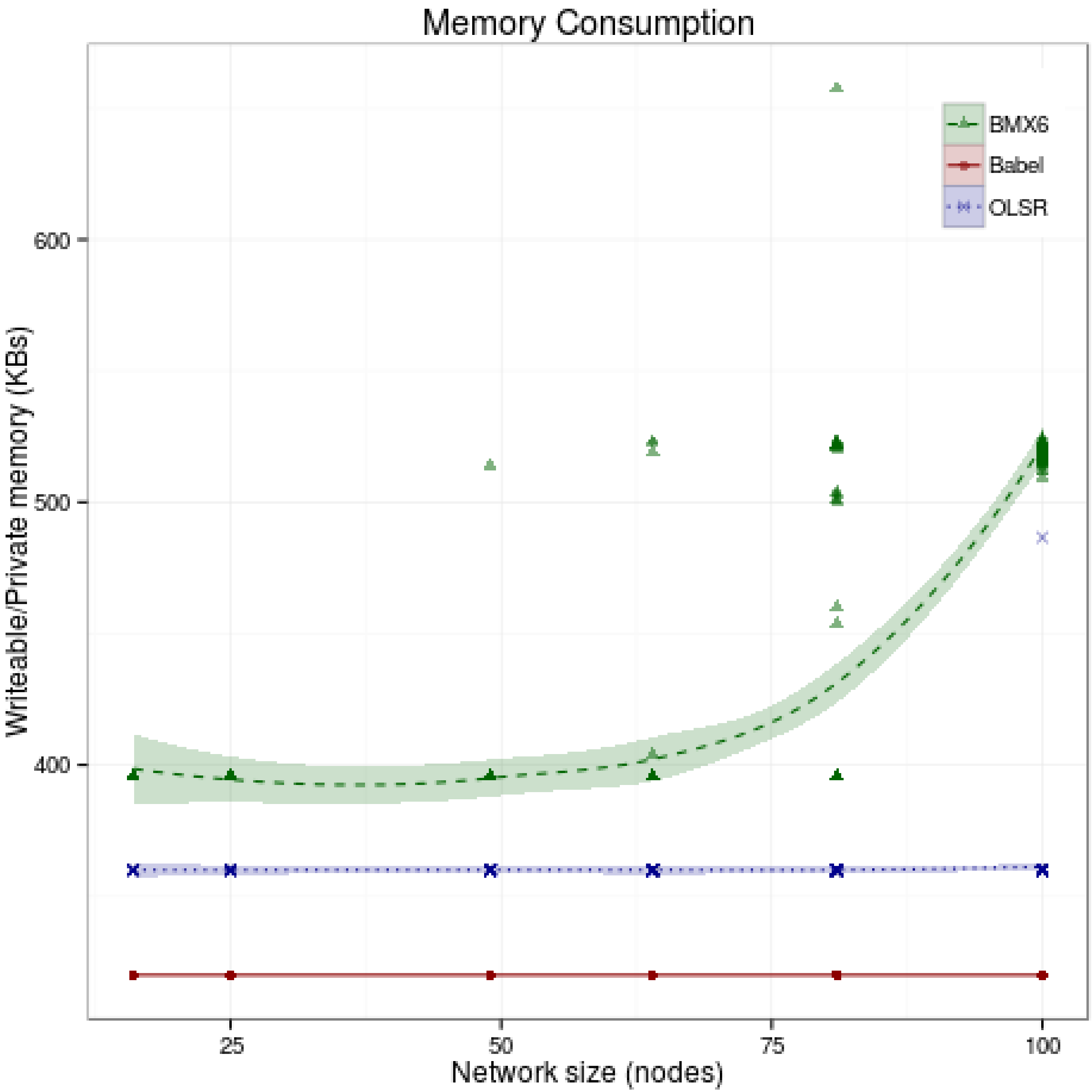
CPU Consumption
Instructions (ips)
2.0e+07
1.5e+07
1.0e+07
5.0e+06
BMX6
Babel
OLSR
0
5
10
15
Average link loss (%)

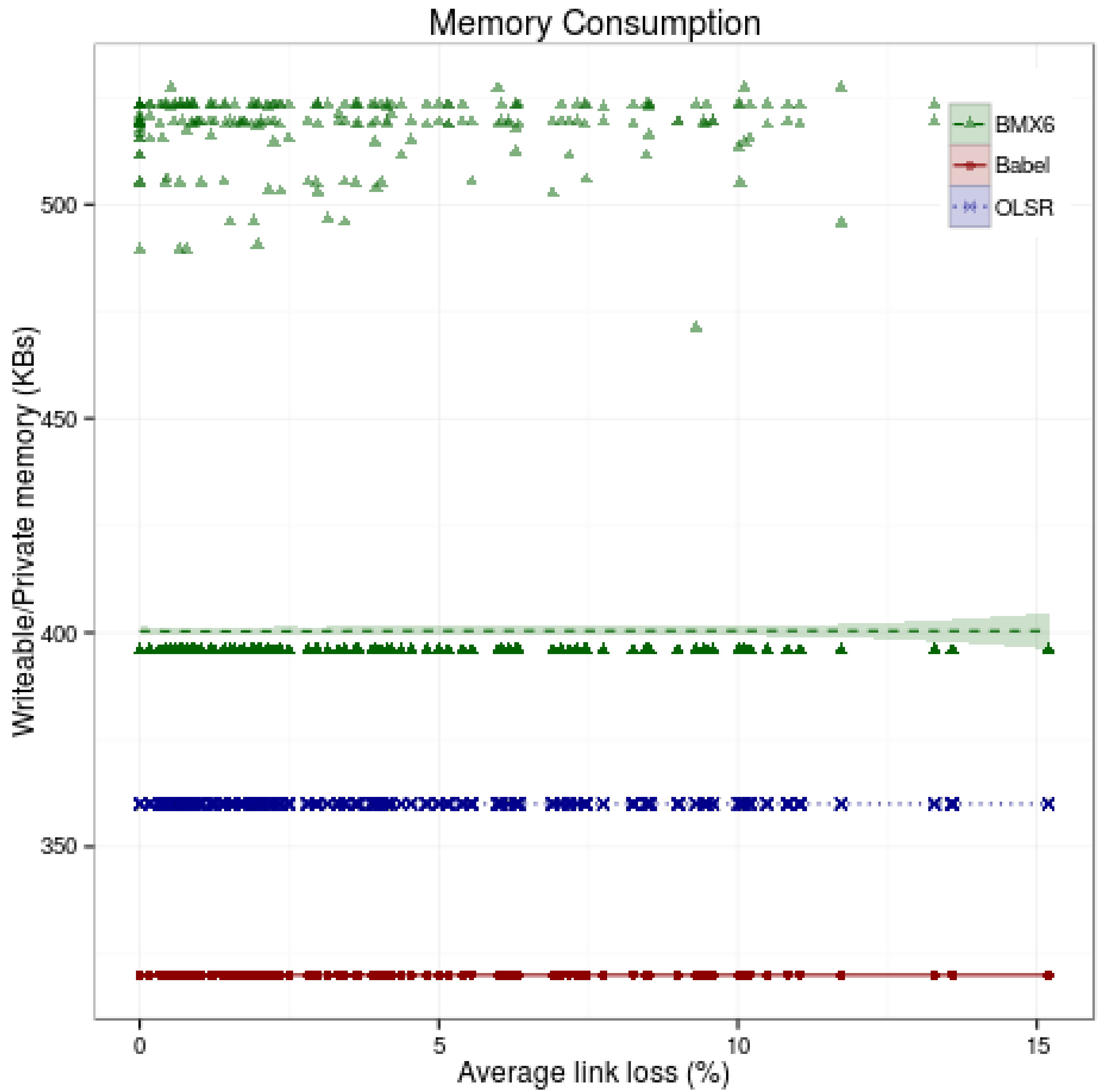

Convergence delay
BMX6
Babel
OLSR
150
100
50
Max offline time(s)
0
5
10
Average link loss probability (%)


Routing Overhead
35
30
25
20
15
Overhead (KB/s)
0
5
10
Average link loss probability (%)
BMX6
Babel
OLSR


Routing Overhead
BMX6
OLSR
Babel
Overhead(KB/s)
0
50
100
150
200
250
300
0
500
1000
1500
time(s)


Memory Consumption
BMX6
Babel
OLSR
Writeable/Private memory (KBs)
500
450
400
350
4
8
12
Number of neighbors


Memory Consumption
BMX6
Babel
OLSR
600
500
400
Writeable/Private memory (KBs)
25
50
75
100
Network size (nodes)



Memory Consumption
Writeable/Private memory (KBs)
500
450
400
350
0
5
10
15
Average link loss (%)
BMX6
Babel
OLSR

Characteristic	OLSR	BMX6	Babel
Increase of protocol overhead depending on size, density, and dynamics			
Size:	high linear	low linear	moderate linear
Density (low density):	high linear	low linear	lowest
Density (high wireless density):	logarithmic	low linear	in non-linear steps
Topology dynamics due to interference from TCP user traffic:	negative	unimpaired	highly susceptible with typical strong growth
Increase of memory usage depending on size and density			
Size:	low linear	low linear	lowest, unaffected
Density:	linear, acceptable	low linear	low, unaffected
Comment: Non-critical given the studied range of size, density, and dynamics			
Increase of CPU usage depending on size and density			
Size:	low, total max $< 0.5\%$	low linear, total max $< 0.2\%$	low linear, total max $< 0.2\%$
Density:	varying, total max $< 1.5\%$	varying, total max $< 1\%$	varying, total max $< 2\%$
Comment: Non-critical given the studied range of size, density, and dynamics			
End-to-end path-healing performance due to topology changes			
Off time versus path length:	high, increasing slope with jump between 7 and 9 hops, $avg \sim 35s$	low, unaffected, $avg \sim 8s$	medium, unaffected, $avg \sim 16s$
Outage versus changing rate:	BMX6 shows least outage in highly changing environments All protocols equally good at low changing rates		