

Low Overhead Loop-Free Routing in Wireless Sensor Networks

Henry-Joseph **Audéoud**, Michał **Król**, Martin **Heusse**, and
Andrzej **Duda**

Grenoble Institute of Technology,
CNRS Grenoble Informatics Laboratory UMR 5217,
Grenoble, France



Low Overhead Loop-Free Routing in Wireless Sensor Networks — Talk outline

- LRP: Lightweight Routing Protocol
- Based on a collection tree
- Downward host routes

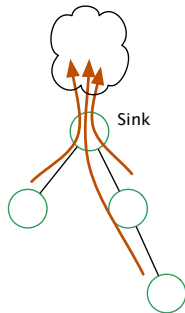
- Loop-free local repair mechanism
- Loop-free IP packets routing
- Low Overhead

- Evaluations experiments

Presentation of LRP

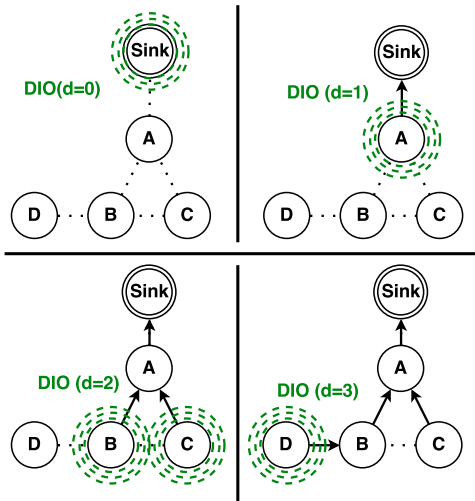
LRP — Collection Tree

- Collect outgoing traffic
- Default routes directed to the root
- Construction similar to RPL



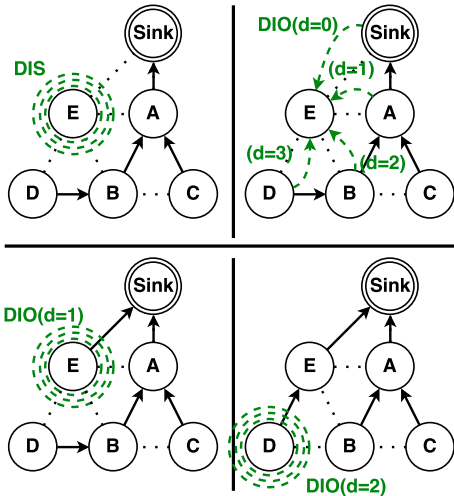
Collection Tree Construction,

using DODAG Information Object (DIO) messages



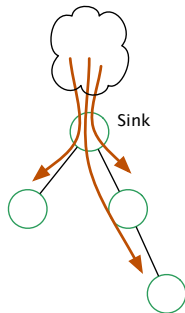
New Node Association

using DODAG Information Solicitation (DIS) messages



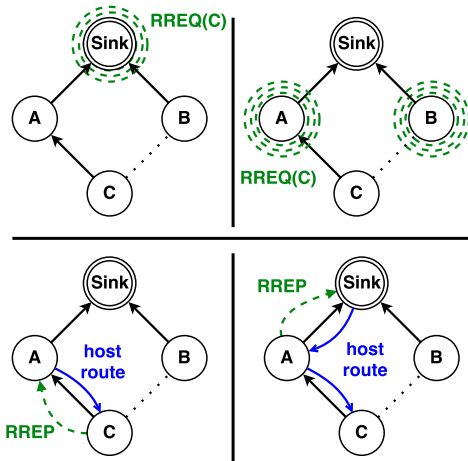
LRP — Downward Routes

- Host routes may be created **reactively**...
 - ▶ Find a host into the network
 - ▶ Repair a broken host route
- ... or **proactively**
 - ▶ e.g. a newly associated node
 - ▶ Decrease energy consumption



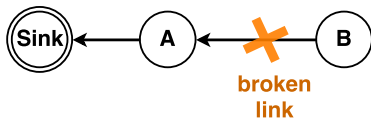
Reactive Host Route Establishment,

using Route REQuest and Route REPLY messages



Contributions

First Contribution — Loop-Free Local Repair



Aim **Re-association** to the network after link break

Existing solutions

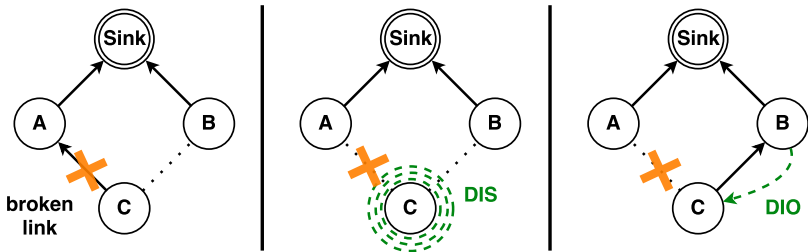
TORA: Must not lose routing packets & transient routing loops

RPL: Limited Count-to-Infinity situations

Local Repair,

direct re-association

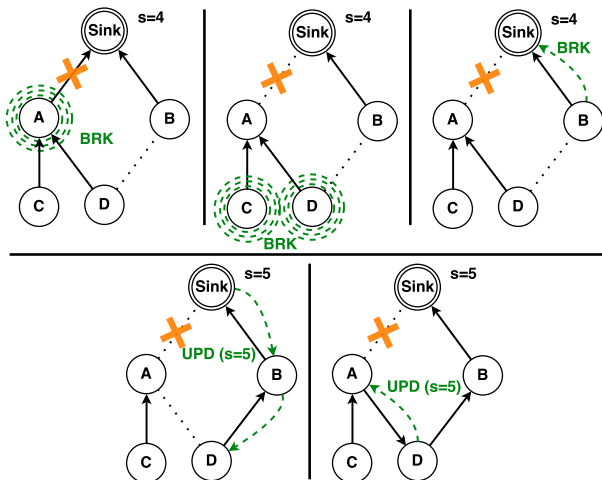
Triggered by Neighbor Unreachability Detection algorithm



Problem To avoid loops, a node **must not move away** from the sink

Local Repair,

using Link Reversal mechanism: BReaK and UPDate messages

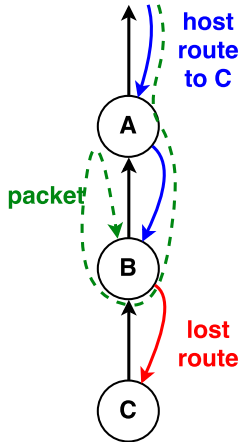


Second Contribution — Datapath Validation

Aim **Detect loops** created by the
coexistence between **host** and
default routes

Existing solutions

- Babel: Remember deleted routes, to ensure they are not used again
- RPL: Add information (and IP header !) to all IP packets

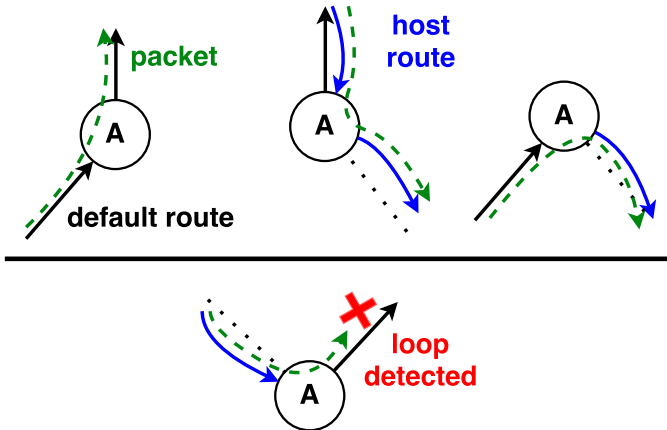


LRP's order

Solution in LRP Use routes accordingly to an order

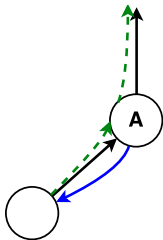
- More precise (prefix length)
 - Newer (sequence number)
 - Closer (metric)

Loop Detection on Packet Routing

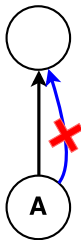


Previously used route...

host or default route?



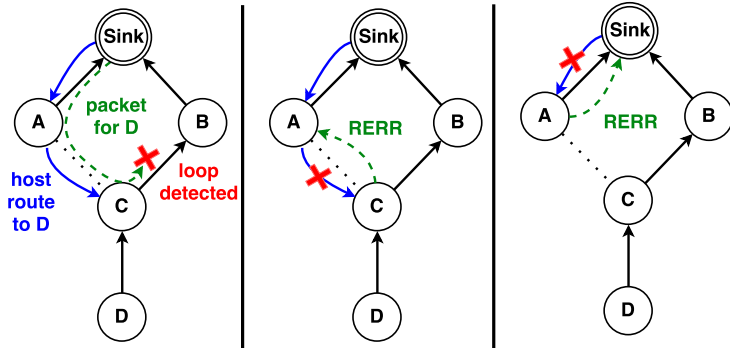
Identify predecessors



Ensure using default route to
successor

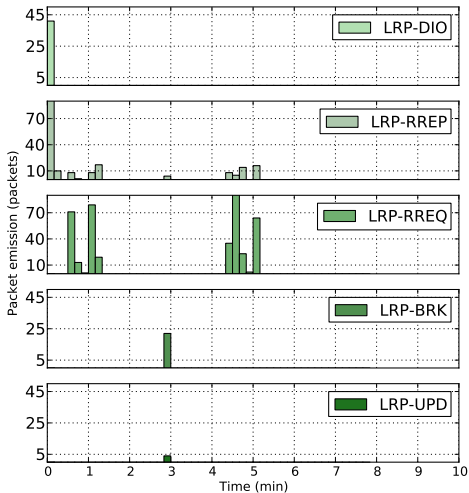
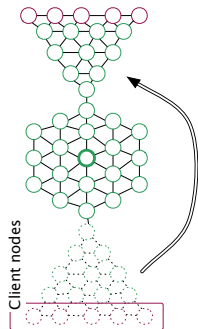
Looping Route Erasure,

using Route ERror messages

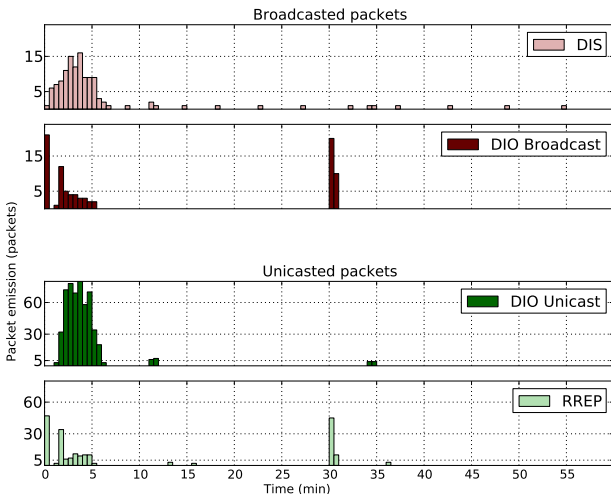


Evaluations experiments

Emulation in Cooja —

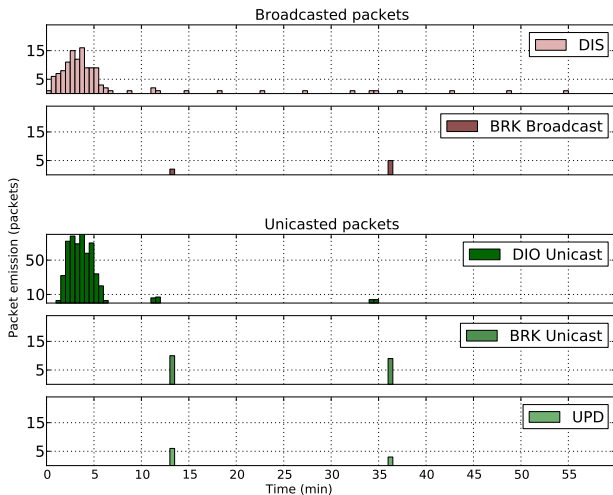


Experimentation on IoT-lab — Collection Tree Construction



Experimentation on IoT-lab —

Local Repair



Conclusion

LRP:

- Loop-free local repair mechanism
- Loop-free IP packets routing
- Loop-free, at all time, low overhead

Perspectives

- Which metric use?
- Include distance into DIS message
- Expanding ring search for local repair
- Explicit predecessor declaration

Thank you !