If the routing protocol is so smart, why should neighbour discovery be so dumb?

Nicolas Déjean, Coronis,
Montpellier, France
nicolas.dejean@coronis.com

Position statement

Our interest is in Smart Metering infrastructures, made of multi-hop wireless networks of resource-constrained devices.

Because of the wireless communication medium, and in particular in areas of high node density, an end-point has link to many routers. This is different from the traditional wired networks.

It thus makes sense that a leaf node has an ability to select a "good" router (or a few good routers) to attach to, among all available. For this purpose, the node has to understand at least a subset of the routing protocol and the metrics in use in the network.

Energy consumption and the use of the communication medium are of primary concern in resource-constrained wireless networks. A node should quickly arrive at a reasonably good choice of router, and without triggering too much activity in the network. It therefore makes sense to optimize the initial connection to the network compared to traditional ND.

Since all nodes (including leaf nodes) within the network know at least a subset of the routing protocol, the routing protocol itself can be used for ND, eliminating much of the bootstrapping control traffic.

Background

From listening to our customer requirements and from our experience of having deployed and operated dozens of Smart Metering networks including some comprising more than 100k nodes, we have come up with the following two key principles:

- When a new node is added to a live Smart Metering network, the connection must happen and a confirmation must be returned to the technician within a few minutes.
- Due to high node density, a naïve joining process can easily trigger control traffic worth several years of steady network operation energy.

In the Water Metering market for instance, where the whole network (including routers) is often battery operated, we generally operate at a signalling period comprised between one hour and several days in order to reach the expected 10 to 15 years lifetime of the nodes. For the first principle to be met, the connection process must be much quicker than the steady control traffic rate, and therefore it must happen in a pro-active manner, with the new node explicitly requesting information from the neighbour routers.

In addition, in order to reduce collision rates and also to limit the power consumption induced in the network by the overhearing phenomenon, the new node shall qualify its request with some parameters, such as minimal expected metric value, so that the population of routers that are authorized to answer is limited. With a narrow enough selection of router authorized to answer, the usual medium access contention mechanisms perform well. If no router answers the call, an iterative algorithm lowers the requirements until an answer is received. Energy-wise, this is much better than a single "catch-all" request.

Finally, modern wireless PHY/MAC layers can avoid most of the overhearing when unicast addressing is used. The reception is aborted as soon as the destination MAC address is received and detected to be different from that of the node. Using unicast addressing as soon as possible will thus allow maximum energy conservation during the connection process. In a typical Smart Metering setting, the endpoints only reach out to a few routers and endpoints. Conversely, the routers, situated on elevated locations, are heard by a large number of nodes. As a consequence, broadcast addressing can be safely used for network solicitation requests, but unicast addressing should be used for answering to them.

This kind of algorithm has been demonstrated in the field for years now, albeit with a proprietary protocol, long before IETF developed an interest in low-power wireless networks. We are looking forward to reaching the same performance level with RPL and a variation of 6LoWPAN over our own radio link technology.

Appendix: details on Coronis proprietary protocol

The goal is to achieve both low energy expenditure and quick connection of a new node into a live network. We assume that the new node knows the metrics in use within the targeted network.

The node runs an iterative algorithm by which it requests (broadcast) network information from routers belonging to the desired network ID and which match the request parameters. The requirements are relaxed at each round until one or several answers are received, or when the maximum number of rounds is reached. The answers by the routers can contain the values of other metrics in addition to those by which the request was qualified, to that the router selection can take place based on more metrics.

The following example is based on two simple metrics in the requests (hop count and link quality level) and relies on the transmission of a third metric into the answers (routing load).
In this example, the router hop count is limited to 3 and the link quality level has three possible values. Consequently, up to twelve frames can be broadcast, in this order:

- Soliciting info from router with Hop Count = 0, Link Quality Level >= 75%
- Soliciting info from router with Hop Count = 0, Link Quality Level >= 50%
- Soliciting info from router with Hop Count = 0, Link Quality Level >= 25%
- Soliciting info from router with Hop Count = 1, Link Quality Level >= 75%
- Soliciting info from router with Hop Count = 1, Link Quality Level >= 50%
- Soliciting info from router with Hop Count = 1, Link Quality Level >= 25%
- Soliciting info from router with Hop Count = 2, Link Quality Level >= 75%
- Soliciting info from router with Hop Count = 2, Link Quality Level >= 50%
- Soliciting info from router with Hop Count = 2, Link Quality Level >= 25%
- Soliciting info from router with Hop Count = 3, Link Quality Level >= 75%
- Soliciting info from router with Hop Count = 3, Link Quality Level >= 50%
- Soliciting info from router with Hop Count = 3, Link Quality Level >= 25%

Any answer stops the iteration. The new node then selects its router, based on the routing load and the link quality level, base on the following algorithm:

- First filter out asymmetric link (link quality on new node receive side does not match the request parameter)
- Then favour the routers that advertise the minimal routing load (load balancing)
- In case of equality, favour the routers that present the best link quality level.