

Evaluation of Routing Protocol for Low Power and Lossy Networks: LOADng and RPL

Jiazi Yi, Thomas Clausen

Laboratoire d'Informatique (LIX) – Ecole Polytechnique, France
jiazi@jiaziyi.com Thomas@ThomasClausen.org

Yuichi Igarashi

Hitachi Yokohama Research Lab, Japan
yuichi.igarashi.hb@hitachi.com

Abstract—Routing protocol is a critical component of Low-power and Lossy Networks for Smart Grid. The protocols are used for data forwarding, which includes data acquisition, information dissemination, etc. This paper evaluates two main routing protocols used for Low-power and Lossy Networks: RPL and LOADng, to understand their strengths and limitations. Observations are provided based on analysis of specification and experimental experience, regarding the protocol's routing overhead, traffic pattern, resource requirement, fragmentation, etc. Simulations are further launched to study the performance in different traffic patterns, which include sensor-to-sensor traffic, sensor-to-root traffic and root-to-sensor bidirectional traffic. By evaluating those protocols, the readers could have better understanding of the protocol applicability, and choose the appropriate protocol for desired applications.

I. INTRODUCTION

Advanced Metering Infrastructure (AMI) systems and Smart-Meter networks enable measurement, collection and analysis of energy usage, by way of facilitating fine-grained communication between energy production/distribution entities, and metering (and consumption) entities, represented by electricity meters, heat meters, water meters, gas meters.

A. Background and History

Since the late 90s, the IETF (The Internet Engineering Task Force) has embarked upon a path of developing routing protocols for networks with increasingly more fragile and low-capacity links, with less pre-determined connectivity properties and with increasingly constrained router resources. In '97, by chartering the Mobile Ad hoc NETWORK (MANET) working group, then subsequently in 2006 and 2008 by chartering the 6LowPAN (IPv6 over Low power WPAN) and ROLL (Routing Over Low power and Lossy networks) working groups.

1) *MANET Protocol Developments*: The MANET working group, which aims at standardizing IP routing protocol for ad hoc networks, converged on the development of two protocol families: reactive protocols, including AODV (Ad hoc On-demand Distance Vector Routing) [1], and proactive protocols, including Optimized Link State Routing (OLSR) [2].

After acquiring operational experiences, the MANET working group commenced developing successors to OLSR and AODV, denoted OLSRv2 and DYMO (Dynamic MANET On-demand Routing). Whereas a relatively large and active community around OLSR thus standardized OLSRv2 [3], the

momentum behind DYMO withered in the MANET working group¹.

2) *6LowPAN and ROLL Protocol Developments*: The 6LowPAN working group was chartered for adapting IPv6 for operation over IEEE 802.15.4, accommodating characteristics of that MAC layer, and with a careful eye on resource constrained devices (memory, CPU, energy, ...). Part of the original charter for this working group was to develop protocols for routing in multi-hop topologies under such constrained conditions, and over this particular MAC. Several proposals for routing were presented in 6LowPAN, for each of these philosophies, including LOAD (6LoWPAN Ad Hoc On-demand Distance Vector Routing) [4]. LOAD was a derivative of AODV, but adapted for L2-addresses and mesh-under routing, and with some simplifications over AODV (*e.g.*, removal of intermediate node replies and sequence numbers). However, 6LowPAN was addressing other issues regarding adapting IPv6 for IEEE 802.15.4, such as IP packet header compression, and solving the routing issues was suspended, delegated to a working group ROLL, created in 2008 for this purpose. ROLL produced a routing protocol denoted "Routing Protocol for Low-power lossy networks" (RPL) [5] in 2012.

3) *Towards LOADng*: While LOAD [4] development was suspended by the 6LowPAN working group, pending the results from ROLL and experiences with RPL, AODV derivatives live on: IEEE 802.11s [6] is based on AODV, and the G3-PLC (Power Line Communication) standard [7], published in 2011, specifies the use of [4] at the MAC layer, for providing mesh-under routing for utility (electricity) metering networks. Justifications for using an AODV derivative in preference to RPL include that the former better supports bi-directional data flows such as a request/reply of a meter reading [8], as well as algorithmic and code complexity reasons [9].

In November 2012, ITU G3-PLC Alliance approved unanimously to supersede LOAD, the routing algorithm described in the recommendation ITU-T G.9903, with Lightweight On-demand Ad hoc Distance-vector Routing (LOADng). Resulting from this decision, ITU-T has ratified the recommendation ITU-T G.9903 Amendment 1 in May 2013 [12] which includes LOADng in a specific normative annex.

¹<http://tools.ietf.org/wg/manet/minutes?item=minutes81.html>

B. Statement of Purpose

This paper evaluates the two routing protocols, LOADng and RPL - two of the major protocols that are standardized and to be used in AMI networks. Their applicability, strength and limitations are illustrated. The following aspects are specially studied, which are also of the great concerns of LLNs in constrained environments:

- The number of routing protocol messages used to maintain the routing tables; this is one important factor in energy consumption, and a lower routing protocol overhead is thus beneficial to prolong the network lifetime.
- Support for different kinds of traffic patterns, which include sensor-to-root (multipoint-to-point) traffic, root-to-sensor (point-to-multipoint) traffic, and sensor-to-sensor (point-to-point) traffic [16].
- Support of bi-directional routes. Some application traffic in AMI networks is bi-directional (*e.g.*, requiring a message exchange between two nodes in the network, typically between the controller and a meter). Furthermore, some upper-layer protocols, not specifically conceived for AMI but for LLNs in general, require acknowledgement message. For example, CoAP (Constrained Application Protocol) proposed by IETF [13] and DLMS/COSEM (Device Language Message specification / Companion Specification for Energy Metering) [14] both supports request/response type application.
- Length of the paths discovered by routing protocols. The routing protocol is expected to obtain optimal paths (the shortest one according to link metric used).
- Likelihood of packet fragmentation. In LLNs, the layer 2 frame size is often limited - *e.g.*, for IEEE 802.15.4, the maximum frame size is 127 octets. This requires not only small message size for routing control messages, but also low overhead in the data packet to reduce the risk of packet fragmentation.

[15] provides initial simulation results on the performance of RPL and LOADng, in a centralized architecture (communication related to the root) and home automation scenarios (less than 50 nodes). However, the traffic patterns, as well as size of the network tested are still limited, and some of the concerns listed above are not addressed.

Based on the previous efforts and experiences, this paper provides some general observations of RPL and LOADng, and evaluates the performance of the two routing protocols in different traffic patterns, and large scale simulation scenarios (up to 500 nodes).

C. Paper Outline

The remainder of this paper is organized as follows. Section II presents a selection of observations of RPL and LOADng, to understand their properties and characteristics when being used in LLNs. In section III, simulations are performed to study the performance of the routing protocols with different traffic patterns. Section IV concludes this paper .

II. RPL AND LOADng: OBSERVATION

In section I-B, some basic requirements to AMI routing protocols were listed. This section presents an analysis of RPL and LOADng with respect to these requirements. The analysis and observations are based on the protocol specifications, and will – as far as possible – be verified by the performance study in presented in section III.

A. RPL Observations

RPL is a proactive distance-vector protocol. The basic construct in RPL is a Destination Oriented Directed Acyclic Graph (DODAG), with a single RPL Route acting as DODAG root. The DODAG Root has responsibilities in addition to those of other RPL Routers, including initiating, configuring, and managing the DODAG, and (in some cases) acting as a central relay for traffic through and between other RPL Routers in the network. The DODAG is constructed by generation and transmission of DIO (DODAG Information Object) message, indicating the router's rank in the DODAG. Thus, RPL provides “upward” or “multipoint-to-point” routes from sensors inside the network towards the DODAG root. “Downward routes” are enabled by having sensors issue Destination Advertisement Object (DAO) messages, propagating as unicast via preferred parents towards the DODAG Root. “Point-to-point routes”, for communication between devices inside the network and where neither of the communicating devices are the DODAG Root, are as default supported by having the source sensor transmit a data packet, via its default route to the DODAG Root (*i.e.*, using the upward routes), which will then, depending on the “Mode of Operation” for the DODAG, either add a source-route to the received data packet for reaching the destination sensor (downward routes in non-storing mode), or simply use hop-by-hop routing (downward routes in storing mode) for forwarding the data packet.

1) *Data traffic flow assumption:* RPL makes a-priori assumptions of data traffic types, and explicitly defines three such traffic types: sensor-to-root data traffic (multipoint-to-point, MP2P) is predominant, root-to-sensor data traffic (point-to-multipoint, P2MP) is rare and sensor-to-sensor (point-to-point, P2P) data traffic is extremely rare. While not specifically called out thus in the RPL specification [5], the resulting protocol design, however, reflects these assumptions in that the mechanism constructing MP2P routes is efficient in terms of control traffic generated and state required, P2MP route construction much less so - and P2P routes subject to potentially significant route stretch (routes going through the DODAG Root in non-storing mode) and over-the-wire overhead from using source routing (from the DODAG Root to the destination) – or, in case of storing mode, considerable memory requirements in all routers also inside the network.

In real applications, there are scenarios where P2P traffic is a common occurrence. For example, in the remote control in building automation, more than 30% percent of traffic can be P2P [16]. This can cause more network congestion and energy consumption by using RPL.

2) *DODAG root requirement*: When the Internet was designed, it was with the principle of avoiding a single-point-of-failure: even if one or several routers cease operation, the other routers in the network should be able to make the global network continue to function.

In RPL, the DODAG Root has both a special responsibility and is subject to special requirements. The DODAG Root is responsible for determining and maintaining the configuration parameters for the DODAG, and for initiating DIO emissions. The DODAG Root is also responsible (in both storing and non-storing mode) for being able to, when downward routes are supported, maintain sufficient topological information to be able to construct routes to all destinations in the network.

Although in principle, RPL provides “Floating DODAGs” to provide internal connectivity in case the administratively provisioned DODAG fails, because of the special memory/function requirement of DODAG root, the other routers in the network either may not have enough resources to act as DODAG roots – or, will all have to be provisioned with resources sufficient to act as DODAG root (resources which will be unused during normal network operation). Therefore, the DODAG root represents a potential single-point-of-failure in an RPL routed network.

3) *Fragmentation*: It is desired that applications in an AMI network originate small packets, in order to avoid link-layer fragmentation [17]. In 802.15.4 [18] the frame size is only 127 octets, of which up to 25 octets may be used for frame overhead, and another 21 octets for link layer security, leaving 81 octets for layer-2 payload. When using the IPv6 compression mechanism supplied by the compressed IPv6 header consumes (at least) 2 octets [19], leaving (at best) 79 octets for layer 3 data payload – such as routing protocol signals and application data – before fragmentation occurs.

While 79 octets may seem sufficient to carry RPL control messages, consider the following: RPL control messages are carried in ICMPv6, and the mandatory ICMPv6 header consumes 4 octets. The DIO base object of RPL consumes another 24 octets. If link metrics are used, that consumes at least another 8 octets – and this, when using a simple hop count metric; other metrics may require more. The DODAG Configuration Object consumes up to a further 16 octets, for a total of 52 octets. Adding a Prefix Information Object for address configuration consumes another 32 octets, for a total of 84 octets, thus exceeding the 79 octets available for layer 3 data payload and causing link-layer fragmentation of such a DIO.

For data traffic, RPL may further increase the probability of fragmentation in downward traffic for storing mode, due to the use of source-routing: for each data packet a fixed source routing overhead of 8 octets is imposed, plus a variable number of octets for entries in the source route – the exact number of octets depending on the address length and number hops to be traversed. This would reduce the already limited space available for data payload, and increase the risk of fragmentation. Of further note is, that the DODAG root adding a source routing header means that applications cannot know

in advance what to expect of the MTU, and thus applications cannot properly design their data traffic to “fit” within an unfragmented frame.

4) *Link Bi-directionality*: When a DODAG is constructed by RPL, parents (and the preferred parent) are selected based on receipt of DIOs. This, alone, does not guarantee the ability of an RPL Router to successfully communicate with the parent. However, the basic use of links is for “upward” routes, i.e., for the RPL Router to use a parent (the preferred parent) as relay towards the DODAG Root - in the opposite direction of the one in which the DIO was received.

In fact, unidirectional links are no rare occurrence in wireless networks. Although “*RPL operations requires bidirectional links*” [5], RPL does not specify exactly what mechanisms should be used, and how they should be used to avoid unidirectional links. There are some solutions mentioned in [5], such as Bidirectional Forwarding Detection (BFD) or Neighbor Unreachability Detection (NUD).

BFD is explicitly called out as “often not desirable” as it uses a proactive approach, and is not considered (by RPL) as adapted to LLNs. For NUD, it is only invoked when a transmission fails. An RPL Router may detect that its preferred parent is lost by way of NUD, when trying to communicate to the DODAG Root. If that RPL Router has no other parents in its parent set, all it can do is wait: RPL does not provide other mechanisms for an RPL Router to react to such an event. Therefore, RPL adopts neither those mechanisms, and does not specify an alternative mechanism.

5) *Loops*: RPL “*guarantees neither loop free route selection nor tight delay convergence times, but can detect and repair a loop as soon as it is used. RPL uses this loop detection to ensure that packets make forward progress [...] and trigger repairs when necessary*” [5]. This implies that a loop may only be detected and fixed when data traffic is sent through the network. When a loop is detected and route repair is triggered, data packets have to be buffered. However, with limited memory constraints of LLNs, buffering incoming packets during the route repair may not be possible for all incoming data packets, leading to dropped packets. Depending on the transport protocol, these data packets must be retransmitted by the source (increasing the network load) or be definitely lost.

B. LOADng Observations

LOADng [10] is derived from AODV. It includes a core specification and different extensions to improve the performance in specific scenarios (e.g., collection trees for more efficient data acquisition [20], smart route request [21] and expanding ring search [22] to reduce overhead). As a reactive protocol, the basic operations of LOADng include generation of Route Requests (RREQs) by a LOADng Router (originator) for when discovering a route to a destination, forwarding of such RREQs until they reach the destination LOADng Router, generation of Route Replies (RREPs) upon receipt of an RREQ by the indicated destination, and unicast hop-by-hop forwarding of these RREPs towards the originator.

1) *Data traffic flow assumption*: LOADng makes no assumption of data traffic type used with the protocol, but provides a general point-to-point mechanism, which is more close to the typical Internet architecture: in the Internet, no single router acts as relay for all traffic. To have more efficient performance when subject to other traffic patterns, such as data acquisition or broadcast, this is supported by way of the protocol being extensible (in a backwards/forward compatible fashion).

2) *No Root - No Requirements*: No router in a LOADng network plays any special role. All routers exhibit the same behavior (signaling and processing). Even when using a collection tree extension, while there exists a conceptual “root”, the router which acts as that conceptual “root” has no additional signaling or processing requirements when compared to other routers in the network.

3) *Fragmentation*: To reduce the overhead of routing control messages, LOADng makes use of a generalized message format, specially designed for ad hoc networks [23]. It can not only reduce the overall size of the control packets generated by LOADng, but also permits protocol extensions to be developed. Additionally, by being frugal in included information in LOADng control messages, a typical LOADng control message size is around 30 octets (depending on the length of address used), which easily fits into a single 802.15.4 frame.

For data traffic, LOADng does not require routing-related information (such as a source routing header, or fields to indicate the “direction” of the packet) to be added to the IP header. Thus applications can properly and predictably design their data traffic to “fit” within an unfragmented frame.

4) *Link Bi-directionality*: LOADng is designed to work in networks wherein uni-directional links occur. In the core specification of LOADng, a bi-directionality check is performed by using “blacklist”. When a *Route Reply* message is sent by a LOADng router, by default, it is expecting a *Route Reply Acknowledgment* message. If the acknowledgement is not received on time, the “next hop” of this *Route Reply* is blacklisted, and is not supposed to join the route in a subsequent route discovery process. By doing so, only bi-directional links are used for forwarding data packets in LOADng.

5) *Loops*: The routing messages in LOADng like *Route Request* and *Route Reply* generated by a given LOADng Router share a single unique, monotonically increasing sequence number. Meanwhile, only destination is permitted to respond to a *Route Request*, which ensures loop freedom.

Table I summarizes the general features of RPL and LOADng.

III. LOADNG AND RPL: PERFORMANCE EVALUATION

This section presents the simulation results for LOADng and RPL (non-storing mode) performance evaluation. Different traffic patterns, including P2P, MP2P, P2MP are studied.

A. Simulation Setup

Simulations were made with varying numbers of routers from 63 to 500, equipped with IEEE 802.11 wireless interfaces. Note that it is not suggested that IEEE 802.11 be used for LLNs, but the general behavior of routing protocols (notably, observing frame sizes, path-lengths, control signal counts etc) can be concluded from simulations using IEEE 802.11. Routers were placed statically uniformly in a square field while maintaining a consistent network density, *i.e.*, the simulation field grows as the number of routers increases: 1100m×1100m for 63 nodes, 1580m×1580m for 125 nodes, 2230m×2230m for 250 nodes, 3160m×3160m for 500 nodes. Each CBR (Constant Data Rate) traffic emits a data packet with 512 bytes every 5 seconds. For each datapoint, the values have been averaged over 10 runs.

RPL is used in non-storing mode. The reason that storing mode is not tested is because each RPL Router has to store routes for destinations in its sub-DODAG. With constrained memory of LLN devices, it is hard to scale in storing mode. In an experimental testbed, [24] argues that the network should be limited to 30 routers in RPL storing mode.

B. Simulation Results

1) *Point-to-point scenarios*: Figure 1 gives the performance of LOADng and RPL in a P2P scenario with 30 concurrent CBR traffic flows. Every CBR traffic flow is originated from a random device (except the root) in the network, to another random device (except the root) in the network. In this scenario, LOADng exhibits a significantly lower overhead, compared to RPL. This is mainly due to the re-active nature of LOADng: the route discovery is initiated only when there is a data packet to be delivered. For RPL, which is a pro-active protocol, control packet transmissions always exist to maintain a routing table to every possible destinations in the network, even there is no data traffic in the network. For those same reasons, LOADng presents slightly higher delay than RPL.

Figure 1(c) illustrates the average path length of the routes discovered by the two protocols. In P2P scenarios, all the data traffic in RPL non-storing mode need to travel through the root (except the destination happens to be somewhere on the route to the root), which results in that the paths offered by RPL are about 60% longer than those offered by LOADng. Because of unnecessary path stretch and higher overhead, RPL has higher loss rate in larger scale scenarios (500 nodes).

	RPL	LOADng
Basic mechanism	Proactive	Reactive
Support of general traffic pattern	No	Yes
Special requirement for the root	Yes	No
Bi-directional route check	No	Yes
Loop freedom	No	Yes
Risk of fragmentation	Yes	No
Constant data packet header	No (non-storing mode)	Yes
Metric support	Yes	Yes

Table I
GENERAL FEATURES OF RPL AND LOADNG

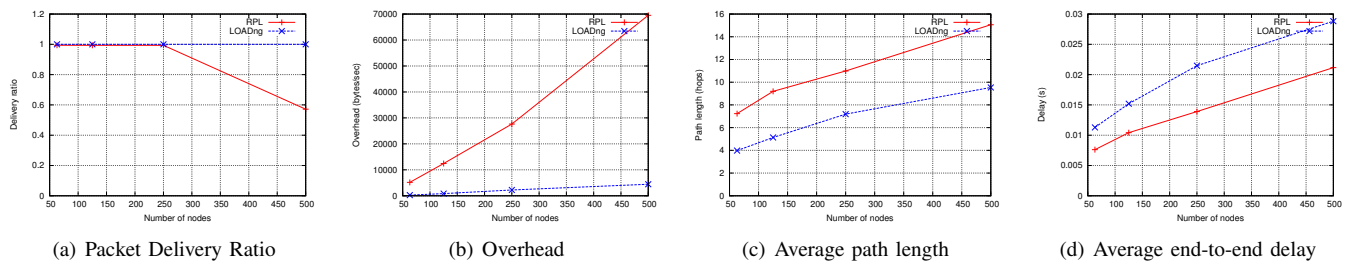


Figure 1. Simulation results of P2P scenarios, 30 CBR sources

2) *Multipoint-to-point Scenarios*: In MP2P scenarios, there is a single root in the network, to which all other devices in the network send data packets, *i.e.*, there are $n - 1$ CBR traffics in the network, where n is the number of devices. Three protocol settings are simulated in MP2P scenarios:

- RPL: RFC 6550 [5] with only DIO message enabled;
- LOADng: the LOADng core specification [10];
- LOADng-CTP: LOADng with collection tree extension [20].

LOADng-CTP is an extension module specially designed for LOADng, and which is compatible with the LOADng core specification.

The performance of different settings is shown in Figure 2. LOADng has the highest overhead because it needs to initiate a route discovery for every route to the root. However, with the collection tree extension, LOADng-CTP greatly reduces the overhead, even when compared to RPL.

3) *P2MP and bi-direction scenarios*: This scenario simulates a bi-directional “polling application” is simulated, where the root collects information from all devices in the network within in 24 hours – a fairly typical real-world application, say, when a power company needing to inquire all power meters on the last day of each month. The root sends a single request with one data packet (11 octets payload) to each sensor, and the sensor replies with a single data packet (100 octets payload).

The simulated time was 25 hours for LOADng and 270 seconds for RPL. The reason for not choosing 24 hours for RPL as well, is quite simply the required amount of time that would be required for conducting the simulation: as DAO messages are sent periodically² in the simulation (5 seconds per packet), even if no data traffic is present in the network, the required amount of time to conduct the simulation would have been several days for a single simulation run, and trace files would grow to hundreds of Gigabytes per simulation. As in LOADng no control traffic is sent when no data traffic is sent, the simulation runs considerably faster, allowing to simulate the 24 hours interval. Despite this difference of simulation time, the results are comparable, as the transmitted data traffic stays the same (0.1 seconds per data request for RPL, and 30 seconds per data request for LOADng). The control traffic overhead of RPL can be extrapolated from 270s to 1 day, as

²One might argue that sending DAOs periodically is not smart, and that this should not be done. Unfortunately, the RPL specification does not indicate any schedule or events, indicating when DAOs are to be sent.

DAO messages are sent periodically.

Figure 3 gives the simulation results of P2MP bi-directional scenarios. It can be concluded from 3(b) that even when RPL only runs for only 270s, it generates a higher overhead than does LOADng. Since the simulation time was much lower for RPL, and DAOs are sent periodically, the expected amount of control traffic for a duration of 1 day can be easily extrapolated, where the overhead of RPL is two orders of magnitude higher than the overhead of LOADng.

Figure 3(d) gives the delay of the two protocols. For LOADng downward traffic, because LOADng needs to wait for the route discovery process before sending out the data traffic, it presents a higher delay than does RPL, as well as a higher delay than LOADng presents for upward traffic. However, for most LLN scenarios where delay is not a crucial issue, this may be acceptable.

IV. CONCLUSION

This paper has evaluated RPL and LOADng, both routing protocols for LLNs, for their applicability also for AMI networks. Based on specification analysis, experimental experience and simulation studies of typical traffic patterns for AMI networks, the strengths and limitations of the two protocols have been explored.

RPL is a proactive protocol, and is optimized for sensor-to-root (MP2P) communication, in which case it provides low delay (if no loop occurs). However, RPL also presents several limitations. First, because the protocol is designed with an implicit assumption of MP2P being the predominant traffic pattern, it is not efficient for root-to-sensor (P2MP) traffic, and not appropriate for sensor-to-sensor (P2P) traffic. Indeed, the control signals (DAOs) for supporting P2MP and P2P traffic appears an “afterthought” by way of their emission schedule being unspecified in the RPL specification. Secondly, RPL imposes a special requirement on the *root* node in the network, which makes it a potential single-point-of-failure, a bottle-neck for traffic, and causes systemic route stretch for P2P traffic. Thirdly, there is a high likelihood of fragmentation and loops when packets are routed by RPL. Finally, because RPL is lack link bi-directionality check mechanisms, it is not appropriate for networks with uni-directional links.

LOADng supports more general traffic patterns, making no a-priori assumptions. Therefore, no node plays a “special role” in the network, from the point of view of the routing protocol. LOADng provides a compressed and flexible packet format,

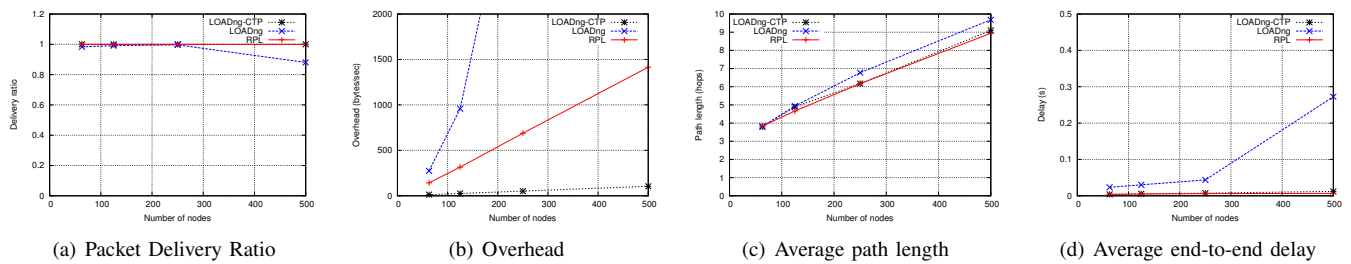


Figure 2. Simulation results of MP2P scenarios

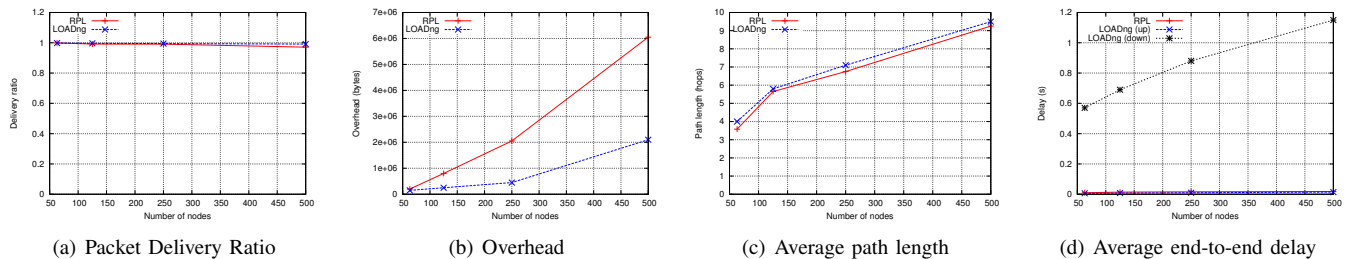


Figure 3. Simulation results of P2MP bi-directional scenarios

with no risk of control packet fragmentation, and also does not impose source routing (or other signals) in data packets, providing applications with a known, fixed MTU – allowing applications to design their data packets to “fit” within a single frame. Also, LOADng offers loop-free routes, and supports networks with uni-directional links. As a reactive protocol, LOADng has possible higher delay in the route discovery phase, and presents higher overhead if there are too many concurrent P2P traffic flows in the network.

REFERENCES

- [1] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” Experimental RFC 3561, July 2003.
- [2] T. Clausen and P. Jacquet, “Optimized Link State Routing Protocol (OLSR),” Experimental RFC 3626, October 2003.
- [3] T. Clausen, C. Dearlove, and P. Jacquet, “The Optimized Link State Routing Protocol version 2,” Internet Draft, draft-ietf-manet-olsrv2-11, work in progress, April 2010.
- [4] K. Kim, S. D. Park, G. Montenegro, S. Yoo, and N. Kushalnagar, “6LoWPAN Ad Hoc On-Demand Distance Vector Routing,” June 2007, Internet Draft, work in progress, draft-daniel-6lowpan-load-adhoc-routing-03.
- [5] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, and J. Vasseur, “RPL: IPv6 Routing Protocol for Low power and Lossy Networks,” March 2012, IETF RFC 6550.
- [6] G. Hiertz, S. Max, R. Zhao, D. Denteneer, and L. Berlemann, “Principles of ieee 802.11s,” Honolulu, Hawaii, USA: Proceedings of WiMAN, Aug 2007, p. 6.
- [7] “ITU-T G.9956: Narrow-Band OFDM power line communication transceivers - Data link layer specification,” November 2011, telecommunication Standardization Section of ITU.
- [8] U. Herberg and T. Clausen, “A comparative performance study of the routing protocols load and rpl with bi-directional traffic in low-power and lossy networks (lln),” Proceedings of the 8th ACM PE-WASUN, October 2011.
- [9] T. Clausen, U. Herberg, and M. Philipp, “A critical evaluation of the “ipv6 routing protocol for low power and lossy networks”,” Proceedings of the 5th IEEE WiMob, October 2011.
- [10] T. Clausen, A. C. de Verdiere, J. Yi, A. Niktash, Y. Igarashi, H. Satoh, and U. Herberg, “The lln on-demand ad hoc distance-vector routing protocol - next generation,” The Internet Engineering Task Force, October 2011, internet Draft, work in progress, draft-clausen-lln-loadng.
- [11] T. Clausen, A. Camacho, J. Yi, A. C. de Verdiere, Y. Igarashi, H. Satoh, and Y. Morii, “Experience with the loadng routing protocol for llns,” The Internet Engineering Task Force, October 2011, internet Draft, work in progress, draft-lavenu-lln-loadng-interoperability-report.
- [12] “ITU-T G.9903: Narrow-band orthogonal frequency division multiplexing power line communication transceivers for G3-PLC networks: Amendment 1,” May 2013, telecommunication Standardization Section of ITU.
- [13] Z. Shelby, K. Hartke, and C. Bormann, “Constrained Application Protocol (CoAP),” May 2013, Internet Draft, work in progress, draft-ietf-core-coap-16.
- [14] “Electricity metering - Data exchange for meter reading, tariff and load control - Part 53: COSEM application layer,” 2006, iEC.
- [15] M. Vucinic, B. Tourancheau, and A. Duda, “Performance comparison of the rpl and loadng routing protocols in a home automation scenario,” Proceedings of IEEE WCNC, 2013.
- [16] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, “Building Automation Routing Requirements in Low-Power and Lossy Networks,” June 2010, IETF RFC 5867.
- [17] N. Kushalnagar, G. Montenegro, and C. Scumacher, “6LoWPANs: Overview, Assumptions, Problem Statement, and Goals,” August 2007, IETF RFC 4919.
- [18] “IEEE Std. 802.15.4-2003,” October 2003, Computer Society, IEEE.
- [19] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks,” September 2007, Standards Track RFC 4944.
- [20] J. Yi, T. Clausen, and A. C. de Verdiere, “Efficient data acquisition in sensor networks: introducing (the) loadng collection tree protocol,” Proceedings of the 8th IEEE International Conference on Wireless Communications, Networking and Mobile Computing., October 2011.
- [21] J. Yi, T. Clausen, and A. Bas, “Smart route request for on-demand route discovery in constrained environments,” Proceedings of the IEEE International Conference on Wireless Information Technology and Systems, September 2012.
- [22] A. Bas, J. Yi, and T. Clausen, “Expanding ring search for route discovery in loadng routing protocol,” Proceedings of The 1st International Workshop on Smart Technologies for Energy, Information and Communication, September 2012.
- [23] T. Clausen, C. Dearlove, J. Dean, and C. Adjih, “Generalized MANET Packet/Message Format,” Std. Track RFC 5444, February 2009.
- [24] J. Ko, S. Dawson-Haggerty, D. Culler, and A. Terzis, “Evaluating the performance of rpl and 6lowpan in tinyns,” Proceedings of the Workshop on Extending the Internet to Low power and Lossy Networks, 2011.