Independent Submission
Request for Comments: 6687
Category: Informational
ISSN: 2070-1721

J. Tripathi, Ed.
J. de Oliveira, Ed.
Drexel University
JP. Vasseur, Ed.
Cisco Systems, Inc.

October 2012

Performance Evaluation of the Routing Protocol for Low-Power and Lossy Networks (RPL)

Abstract

This document presents a performance evaluation of the Routing Protocol for Low-Power and Lossy Networks (RPL) for a small outdoor deployment of sensor nodes and for a large-scale smart meter network. Detailed simulations are carried out to produce several routing performance metrics using these real-life deployment scenarios. Please refer to the PDF version of this document, which includes several plots for the performance metrics not shown in the plain-text version.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6687.

Table of Contents

1. Introduction

Designing a routing protocol for Low-Power and Lossy Networks (LLNs) imposes great challenges, mainly due to low data rates, high probability of packet delivery failure, and strict energy constraints in the nodes. The IETF ROLL Working Group took on this task and specified the Routing Protocol for Low-Power and Lossy Networks (RPL) in [RFC6550].

RPL is designed to meet the core requirements specified in [RFC5826], [RFC5867], [RFC5673], and [RFC5548].

This document's contribution is to provide a performance evaluation of RPL with respect to several metrics of interest. This is accomplished using real data and topologies in a discrete event simulator developed to reproduce the protocol behavior.

The following metrics are evaluated:

- Path quality metrics, such as ETX path cost, ETX path stretch, ETX fractional stretch, and hop distance stretch, as defined in Section 2 ("Terminology");
- Control plane overhead;
- End-to-end delay between nodes;
- Ability to cope with unstable situations (link churns, node dying);
- Required resource constraints on nodes (routing table size).

Some of these metrics are mentioned in the aforementioned RFCs, whereas others have been introduced to consider the challenges and unique requirements of LLNs as discussed in [RFC6550]. For example, routing in a home automation deployment has strict time bounds on protocol convergence after any change in topology, as mentioned in Section 3.4 of [RFC5826]. [RFC5673] requires bounded and guaranteed end-to-end delay for routing in an industrial deployment, and [RFC5548] requires comparatively loose bounds on latency for end-to-end communication. [RFC5548] mandates scalability in terms of protocol performance for a network of size ranging from 10^2 to 10^4 nodes.

Although simulation cannot prove formally that a protocol operates properly in all situations, it can give a good level of confidence in protocol behavior in highly stressful conditions, if and only if real-life data are used. Simulation is particularly useful when theoretical model assumptions may not be applicable to such networks and scenarios. In this document, real deployed network data traces have been used to model link behaviors and network topologies.

2. Terminology

Please refer to [ROLL-TERMS] and [RFC6550] for terminology. In addition, the following terms are specified:

PDR: Packet Delivery Ratio.

CDF: Cumulative Distribution Function.

Expected Transmission Count (ETX Metric): The expected number of transmissions to reach the next hop is determined as the inverse of the link PDR. Consequently, in every hop, if the link quality (PDR) is high, the expected number of transmissions to reach the next hop may be as low as 1. However, if the PDR for the particular link is low, multiple transmissions may be needed.

ETX Path Cost: The ETX path cost metric is determined as the summation of the ETX value for each link on the route a packet takes towards the destination.

ETX Path Cost Stretch: The ETX path cost stretch is defined as the difference between the number of expected transmissions (ETX Metric) taken by a packet traveling from source to destination, following a route determined by RPL and a route determined by a hypothetical ideal shortest path routing protocol (using link ETX as the metric).

ETX Fractional Stretch (fractional stretch factor of link ETX metric against ideal shortest path): The fractional path stretch is the ratio of ETX path stretch to ETX path cost for the shortest path route for the source-destination pair.

Hop Distance Stretch (stretch factor for node hop distance against ideal shortest path): The hop distance stretch is defined as the difference between the number of hops taken by a packet traveling from source to destination, following a route determined by RPL and by a hypothetical ideal shortest path algorithm, both using ETX as the link cost. The fractional hop distance stretch is computed as the ratio of path stretch to count value between a source-destination pair for the hypothetical shortest path route optimizing ETX path cost.

3. Methodology and Simulation Setup

In the context of this document, RPL has been simulated using OMNeT++ [OMNeTpp], a well-known discrete event-based simulator written in C++ and NEtwork Description (NED). Castalia-2.2 [Castalia-2.2] has been used as a Wireless Sensor Network Simulator framework within OMNeT++. The output and events in the simulation are visualized with the help of the Network AniMator, or NAM, which is distributed with the NS (Network Simulator) [NS-2].

Note that no versions of the NS itself are used in this simulation study. Only the visualization tool was borrowed for verification purposes.

In contrast with theoretical models, which may have assumptions not applicable to lossy links, real-life data was used for two aspects of the simulations:

* Link Failure Model: Derived from time-varying real network traces containing packet delivery probability for each link, over all channels, for both indoor network deployment and outdoor network deployment.

* Topology: Gathered from real-life deployment (traces mentioned above) as opposed to random topology simulations.

A 45-node topology, deployed as an outdoor network and shown in Figure 1, and a 2442-node topology, gathered from a smart meter network deployment, were used in the simulations. In Figure 1, links between a most preferred parent node and child nodes are shown in red. Links that are shown in black are also part of the topology but are not between a preferred parent and child node.

Figure 1 [See the PDF.]

Figure 1: Outdoor Network Topology with 45 Nodes.

Note that this is just a start to validate the simulation before using large-scale networks.

A set of time-varying link quality data was gathered from a real network deployment to form a database used for the simulations. Each link in the topology randomly 'picks up' a link model (trace) from the database. Each link has a Packet Delivery Ratio (PDR) that varies with time (in the simulation, a new PDR is read from the database every 10 minutes) according to the gathered data. Packets are dropped randomly from that link with probability (1 - PDR). Each time a packet is about to be sent, the module generates a random number using the Mersenne Twister random number generation method. The random number is compared to the PDR to determine whether the packet should be dropped. Note that each link uses a different random number generator to maintain true randomness in the simulator and to avoid correlation between links. Also, the packet drop applies to all kinds of data and control packets (RPL), such as the DIO, DAO, and DIS packets defined in [RFC6550]. Figure 2 shows a typical temporal characteristic of links from the indoor network traces used in the simulations. The figure shows several links with perfect connectivity, some links with a PDR as low as 10%, and several for which the PDR may vary from 30% to 80%, sharply changing back and forth between a high value (strong connectivity) and a low value (weak connectivity).

Figure 2 [See the PDF.]

Figure 2: Example of Link Characteristics.

In the RPL simulator, the LBR (LLN Border Router) or the Directed Acyclic Graph (DAG) root first initiates sending out DIO messages, and the DAG is gradually constructed. RPL makes use of trickle timers: the protocol sets a minimum time period with which the nodes start re-issuing DAOs, and this minimum period is denoted by the trickle parameter Imin. RPL also sets an upper limit on how many times this time period can be doubled; this is denoted by the parameter DIOIntervalDoublings, as defined in [RFC6550]. For the simulation, Imin is initially set to 1 second and DIOIntervalDoublings is equal to 16, and therefore the maximum time between two consecutive DIO emissions by a node (under a steady network condition) is 18.2 hours. The trickle time interval for emitting DIO messages assumes the initial value of 1 second and then changes over simulation time, as mentioned in [RFC6206].

Another objective of this study is to give insight to the network administrator on how to tweak the trickle values. These recommendations could then be used in applicability statement documents.

Each node in the network, other than the LBR or DAG root, also emits DAO messages as specified in [RFC6550], to initially populate the routing tables with the prefixes received from children via the DAO messages to support Point-to-Point (P2P) and Point-to-Multipoint (P2MP) traffic in the "down" direction. During these simulations, it is assumed that each node is capable of storing route information for other nodes in the network (storing mode of RPL).

For nodes implementing RPL, as expected, the routing table memory requirement varies according to the position in the DODAG (Destination-Oriented DAG). The (worst-case) assumption is made that there is no route summarization (aggregation) in the network. Thus, a node closer to the DAG will have to store more entries in its routing table. It is also assumed that all nodes have equal memory capacity to store the routing states.

For simulations of the indoor network, each node sends traffic according to a Constant Bit Rate (CBR) to all other nodes in the network, over the simulation period. Each node generates a new data packet every 10 seconds. Each data packet has a size of 127 bytes including 802.15.4 PHY/MAC headers and RPL packet headers. All control packets are also encapsulated with 802.15.4 PHY/MAC headers. To simulate a more realistic scenario, 80% of the packets generated by each node are destined to the root, and the remaining 20% of the

packets are uniformly assigned as destined to nodes other than the root. Therefore, the root receives a considerably larger amount of data than other nodes. These values may be revised when studying P2P traffic so as to have a majority of traffic going to all nodes as opposed to the root. In the later part of the simulation, a typical home/building routing scenario is also simulated, and different path quality metrics are computed for that traffic pattern.

The packets are routed through the DODAG built by RPL according to the mechanisms specified in [RFC6550].

A number of RPL parameters are varied (such as the packet rate from each source and the time period for emitting a new DAG sequence number) to observe their effect on the performance metric of interest.

4. Performance Metrics

4.1. Common Assumptions

As the DAO messages are used to feed the routing tables in the network, they grow with time and size of the network. Nevertheless, no constraint was imposed on the size of the routing table nor on how much information the node can store. The routing table size is not expressed in terms of Kbytes of memory usage but measured in terms of the number of entries for each node. Each entry has the next-hop node and path cost associated with the destination node.

The link ETX (Expected Transmission Count) metric is used to build the DODAG and is specified in [RFC6551].

4.2. Path Quality

Hop Count: For each source-destination pair, the number of hops for both RPL and shortest path routing is computed. Shortest path routing refers to a hypothetical ideal routing protocol that would always provide the shortest path in terms of ETX path cost (or whichever metric is used) in the network.

The Cumulative Distribution Function (CDF) of the hop count for all paths (n * (n - 1) in an n-node network) in the network with respect to the hop count is plotted in Figure 3 for both RPL and shortest path routing. One can observe that the CDF corresponding to 4 hops is around 80% for RPL and 90% for shortest path routing. In other words, for the given topology, 90% of the paths have a path length of 4 hops or less with an ideal shortest path routing methodology, whereas in RPL P2P routing, 90% of the paths will have a length of no more than 5 hops. This result indicates that despite having a

non-optimized P2P routing scheme, the path quality of RPL is close to an optimized P2P routing mechanism for the topology under consideration. Another reason for this may relate to the fact that the DAG root is at the center of the network; thus, routing through the DAG root is often close to an optimal (shortest path) routing. This result may be different in a topology where the DAG root is located at one end of the network.

Figure 3 [See the PDF.]

Figure 3: CDF of Hop Count versus Hop Count.

ETX Path Cost: In the simulation, the total ETX path cost (defined in the Terminology section) from source to destination for each packet is computed.

Figure 4 shows the CDF of the total ETX path cost, both with RPL and shortest path routing. Here also one can observe that the ETX path cost from all sources to all destinations is close to that of shortest path routing for the network.

Figure 4 [See the PDF.]

Figure 4: CDF of Total ETX Path Cost along Path versus ETX Path Cost.

Path Stretch: The path stretch metric encompasses the stretch factor for both hop distance and ETX path cost (as defined in the Terminology section). The hop distance stretch, which is determined as the difference between the number of hops taken by a packet while following a route built via RPL and the number of hops taken by shortest path routing (using link ETX as the metric), is computed. The ETX path cost stretch is also provided.

The CDF of both path stretch metrics is plotted against the value of the corresponding path stretch over all packets in Figures 5 and 6, for hop distance stretch and ETX path stretch, respectively. It can be observed that, for a few packets, the path built via RPL has fewer hops than the ideal shortest path where path ETX is minimized along the DAG. This is because there are a few source-destination pairs where the total ETX path cost is equal to or less than that of the ideal shortest path when the packet takes a longer hop count. As the RPL implementation ignores a 20% change in total ETX path cost before switching to a new parent or emitting a new DIO, it does not necessarily provide the shortest path in terms of total ETX path cost. Thus, this implementation yields a few paths with smaller hop counts but larger (or equal) total ETX path cost.

Figure 5 [See the PDF.]

Figure 5: CDF of Hop Distance Stretch versus Hop Distance Stretch Value.

Figure 6 [See the PDF.]

Figure 6: CDF of ETX Path Stretch versus ETX Path Stretch Value.

The data for the CDF of the hop count and ETX path cost for the ideal shortest path (SP) and a path built via RPL, along with the CDF of the routing table size, is given below in Table 1. Figures 3 to 7 relate to the data in this table.

CDF (%age)	Hop (SP)	Hop (RPL)	ETX Cost (SP)	ETX Cost (RPL)	Routing Table Size
0	1.0	1.0	1	1.0	0
5	1.0	1.03	1	1.242	1
10	2.0	2.0	2	2.048	2
15	2.0	2.01	2	2.171	2
20	2.0	2.06	2	2.400	2
25	2.0	2.11	2	2.662	3
30	2.0	2.42	2	2.925	3
35	2.0	2.90	3	3.082	3
40	3.0	3.06	3	3.194	4
45	3.0	3.1	3	3.41	4
50	3.0	3.15	3	3.626	4
55	3.0	3.31	3	3.823	5
60	3.0	3.50	3	4.032	6
65	3.0	3.66	3	4.208	7
70	3.0	3.92	4	4.474	7
75	4.0	4.16	4	4.694	7
80	4.0	4.55	4	4.868	8
85	4.0	4.70	4	5.091	9
90	4.0	4.89	4	5.488	10
95	4.0	5.65	5	5.923	12
100	5.0	7.19	9	10.125	44

Table 1: Path Quality CDFs.

Overall, the path quality metrics give us important information about the protocol’s performance when minimizing the ETX path cost is the objective to form the DAG. The protocol, as explained, does not always provide an optimum path, especially for peer-to-peer communication. However, it does end up reducing the control overhead

cost, thereby reducing unnecessary parent selection and DIO message forwarding events, by choosing a non-optimized path. Despite this specific implementation technique, around 30% of the packets travel the same number of hops as an ideal shortest path routing mechanism, and 20% of the packets experience the same number of attempted transmissions to reach the destination. On average, this implementation costs only a few extra transmission attempts and saves a large number of control packet transmissions.

4.3. Routing Table Size

The objective of this metric is to observe the distribution of the number of entries per node. Figure 7 shows the CDF of the number of routing table entries for all nodes. Note that 90% of the nodes need to store less than 10 entries in their routing table for the topology under study. The LBR does not have the same power or memory constraints as regular nodes do, and hence it can accommodate entries for all the nodes in the network. The requirement to accommodate devices with low storage capacity has been mandated in [RFC5673], [RFC5826], and [RFC5867]. However, when RPL is implemented in storing mode, some nodes closer to the LBR or DAG root will require more memory to store larger routing tables.

Figure 7 [See the PDF.]

Figure 7: CDF of Routing Table Size with Respect to Number of Nodes.

4.4. Delay Bound for P2P Routing

For delay-sensitive applications, such as home and building automation, it is critical to optimize the end-to-end delay. Figure 8 shows the upper bound and distributions of delay for paths between any two given nodes for different hop counts between the source and destination. Here, the hop count refers to the number of hops a packet travels to reach the destination when using RPL paths. This hop distance does not correspond to the shortest path distance between two nodes. Note that each packet has a length of 127 bytes, with a 240-kbps radio, which makes the transmission delay approximately 4 milliseconds (ms).

Figure 8 [See the PDF.]

Figure 8: Comparison of Packet Latency, for Different Path Lengths, Expressed in Hop Count.

RFCs 5673 [RFC5673] and 5548 [RFC5548] mention a requirement for the end-to-end delivery delay to remain within a bounded latency. For instance, according to the industrial routing requirement,

non-critical closed-loop applications may have a latency requirement that can be as low as 100 ms, whereas monitoring services may tolerate a delay in the order of seconds. The results show that about 99% of the end-to-end communication (where the maximum hop count is 7 hops) is bounded within the 100-ms requirement, for the topology under study. It should be noted that due to poor link condition, there may be packet drops triggering retransmission, which may cause larger end-to-end delivery delays. Nodes in the proximity of the LBR may become congested at high traffic loads, which can also lead to higher end-to-end delay.

4.5. Control Packet Overhead

The control plane overhead is an important routing characteristic in LLNs. It is imperative to bound the control plane overhead. One of the distinctive characteristics of RPL is that it makes use of trickle timers so as to reduce the number of control plane packets by eliminating redundant messages. The aim of this performance metric is thus to analyze the control plane overhead both in stable conditions (no network element failure overhead) and in the presence of failures.

Data and control plane traffic comparison for each node: Figure 9 shows the comparison between the amount of data packets transmitted (including forwarded packets) and control packets (DIO and DAO messages) transmitted for all individual nodes when link ETX is used to optimize the DAG. As mentioned earlier, each node generates a new data packet every 10 seconds. Here one can observe that a considerable amount of traffic is routed through the DAG root itself. The x axis indicates the node ID in the network. Also, as expected, the nodes that are closer to the DAG root and that act as routers (as opposed to leaves) handle much more data traffic than other nodes. Nodes 12, 36, and 38 are examples of nodes next to the DAG root, taking part in routing most of the data packets and hence having many more data packet transmissions than other nodes, as observed in Figure 9. We can also observe that the proportion of control traffic is negligible for those nodes. This result also reinforces the fact that the amount of control plane traffic generated by RPL is negligible on these topologies. Leaf nodes have comparable amounts of data and control packet transmissions (they do not take part in routing the data).

Figure 9 [See the PDF.]

Figure 9: Amount of Data and Control Packets Transmitted against Node Id Using Link ETX as Routing Metric.

Data and control packet transmission with respect to time: In Figures 10, 11, and 12, the amount of data and control packets transmitted for node 12 (low rank in DAG, closer to the root), node 43 (in the middle), and node 31 (leaf node) are shown, respectively. These values stand for the number of data and control packets transmitted for each 10-minute interval for the particular node, to help understand what the ratio is between data and control packets exchanged in the network. One can observe that nodes closer to the DAG root have a higher proportion of data packets (as expected), and the proportion of control traffic is negligible in comparison with the data traffic. Also, the amount of data traffic handled by a node within a given interval varies largely over time for a node closer to the DAG root, because in each interval the destination of the packets from the same source changes, while 20% of the packets are destined to the DAG root. As a result, the pattern of the traffic that is handled changes widely in each interval for the nodes closer to the DAG root. For the nodes that are farther away from the DAG root, the ratio of data traffic to control traffic is smaller, since the amount of data traffic is greatly reduced.

The control traffic load exhibits a wave-like pattern. The amount of control packets for each node drops quickly as the DODAG stabilizes, due to the effect of trickle timers. However, when a new DODAG sequence is advertised (global repair of the DODAG), the trickle timers are reset and the nodes start emitting DIOs frequently again to rebuild the DODAG. For a node closer to the DAG root, the amount of data packets is much larger than that of control packets and somewhat oscillatory around a mean value. The amount of control packets exhibits a 'saw-tooth' behavior. In the case where the ETX link metric is used, when the PDR changes, the ETX link metric for a node to its child changes, which may lead to choosing a new parent and changing the DAG rank of the child. This event resets the trickle timer and triggers the emission of a new DIO. Also, the issue of a new DODAG sequence number triggers DODAG re-computation and resets the trickle timers. Therefore, one can observe that the number of control packets attains a high value for one interval and comes down to lower values for subsequent intervals. The interval with a high number of control packets denotes the interval where the timers to emit a new DIO are reset more frequently. As the network stabilizes, the control packets are less dense in volume. For leaf nodes, the amount of control packets is comparable to that of data packets, as leaf nodes are more prone to face changes in their DODAG rank as opposed to nodes closer to the DAG root when the link ETX value in the topology changes dynamically.

Figure 10 [See the PDF.]

Figure 10: Amount of Data and Control Packets Transmitted for Node 12.

Figure 11 [See the PDF.]

Figure 11: Amount of Data and Control Packets Transmitted for Node 43.

Figure 12 [See the PDF.]

Figure 12: Amount of Data and Control Packets Transmitted for Node 31.

4.6. Loss of Connectivity

Upon link failures, a node may lose its parents -- preferred and backup (if any) -- thus leading to a loss of connectivity (no path to the DAG root). RPL specifies two mechanisms for DODAG repairs, referred to as global repair and local repair. In this document, simulation results are presented to evaluate the amount of time data packets are dropped due to a loss of connectivity for the following two cases: a) when only using global repair (i.e., the DODAG is rebuilt thanks to the emission of new DODAG sequence numbers by the DAG root), and b) when using local repair (poisoning the sub-DAG in case of loss of connectivity) in addition to global repair. The idea is to tune the frequency at which new DODAG sequence numbers are generated by the DAG root, and also to observe the effect of varying the frequency for global repair and the concurrent use of global and local repair. It is expected that more frequent increments of DODAG sequence numbers will lead to a shorter duration of connectivity loss at a price of a higher rate of control packets in the network. For the use of both global and local repair, the simulation results show the trade-off in amount of time that a node may remain without service and total number of control packets.

Figure 13 shows the CDF of time spent by any node without service, when the data packet rate is one packet every 10 seconds and a new DODAG sequence number is generated every 10 minutes. This plot reflects the property of global repair without any local repair scheme. When all the parents are temporarily unreachable from a node, the time before it hears a DIO from another node is recorded, which gives the time without service. We define the DAG repair timer as the interval at which the LBR increments the DAG sequence number,

thus triggering a global re-optimization. In some cases, this value might go up to the DAG repair timer value, because until a DIO is heard, the node does not have a parent and hence no route to the LBR or other nodes not in its own sub-DAG. Clearly, this situation indicates a lack of connectivity and loss of service for the node.

Figure 13 [See the PDF.]

Figure 13: CDF: Loss of Connectivity with Global Repair.

The effect of the DAG repair timer on time without service is plotted in Figure 14, where the source rate is 20 seconds/packet and in Figure 15, where the source sends a packet every 10 seconds.

Figure 14 [See the PDF.]

Figure 14: CDF: Loss of Connectivity for Different Global Repair Period, Source Rate 20 Seconds/Packet.

Figure 15 [See the PDF.]

Figure 15: CDF: Loss of Connectivity for Different Global Repair Period, Source Rate 10 Seconds/Packet.

The data for Figures 13 and 15 can be found in Table 2. The table shows how the CDF of time without connectivity to the LBR increases while we increase the time period to emit new DAG sequence numbers, when the nodes generate a packet every 10 seconds.

CDF (%age)	Repair Period 10 Minutes	Repair Period 30 Minutes	Repair Period 60 Minutes
0	0.464	0.045	0.027
5	0.609	0.424	0.396
10	1.040	1.451	0.396
15	1.406	3.035	0.714
20	1.934	3.521	0.714
25	2.113	5.461	1.856
30	3.152	5.555	1.856
35	3.363	7.756	6.173
40	4.9078	8.604	6.173
45	8.575	9.181	14.751
50	9.788	21.974	14.751
55	13.230	30.017	14.751
60	17.681	31.749	16.166
65	29.356	68.709	16.166
70	34.019	92.974	302.459
75	49.444	117.869	302.459
80	75.737	133.653	488.602
85	150.089	167.828	488.602
90	180.505	271.884	488.602
95	242.247	464.047	488.602
100	273.808	464.047	488.602

Table 2: Loss of Connectivity Time, Data Rate - 10 Seconds / Packet.

The data for Figure 14 can be found in Table 3. The table shows how the CDF of time without connectivity to the LBR increases while we increase the time period to emit new DAG sequence numbers, when the nodes generate a packet every 20 seconds.

CDF (%age)	Repair Period 10 Minutes	Repair Period 30 Minutes	Repair Period 60 Minutes
0	0.071	0.955	0.167
5	0.126	2.280	1.377
10	0.403	2.926	1.409
15	0.902	3.269	1.409
20	1.281	16.623	3.054
25	2.322	21.438	5.175
30	2.860	48.479	5.175
35	3.316	49.495	10.30
40	3.420	93.700	25.406
45	6.363	117.594	25.406
50	11.500	243.429	34.379
55	19.703	277.039	102.141
60	22.216	284.660	102.141
65	39.211	285.101	328.293
70	63.197	376.549	556.296
75	88.986	443.450	556.296
80	147.509	452.883	1701.52
85	154.26	653.420	2076.41
90	244.241	720.032	2076.41
95	518.835	1760.47	2076.41
100	555.57	1760.47	2076.41

Table 3: Loss of Connectivity Time, Data Rate - 20 Seconds / Packet.

Figure 16 shows the effect of the DAG global repair timer period on control traffic. As expected, as the frequency at which new DAG sequence numbers are generated increases, the amount of control traffic decreases because DIO messages are sent less frequently to rebuild the DODAG. However, reducing the control traffic comes at a price of increased loss of connectivity when only global repair is used.

Figure 16 [See the PDF.]

Figure 16: Amount of Control Traffic for Different Global Repair Periods.

From the above results, it is clear that the time the protocol takes to re-establish routes and to converge, after an unexpected link or device failure happens, is fairly long. [RFC5826] mandates that "the routing protocol MUST converge within 0.5 seconds if no nodes have moved". Clearly, implementation of a repair mechanism based on new DAG sequence numbers alone would not meet the requirements. Hence, a local repair mechanism, in the form of poisoning the sub-DAG and issuing a DIS, has been adopted.

The effect of the DAG repair timer on time without service when local repair is activated is now observed and plotted in Figure 17, where the source rate is 20 seconds/packet. A comparison of the CDF of loss of connectivity for the global repair mechanism and the global + local repair mechanism is shown in Figures 18 and 19 (semi-log plots, x axis in logarithmic scale and y axis in linear scale), where the source generates a packet every 10 seconds and 20 seconds, respectively. For these plots, the x axis shows time in log scale, and the y axis denotes the corresponding CDF in linear scale. One can observe that using local repair (with poisoning of the sub-DAG) greatly reduces loss of connectivity.

Figure 17 [See the PDF.]

Figure 17: CDF: Loss of Connectivity for Different DAG Repair Timer Values for Global+Local Repair, Source Rate 20 Seconds/Packet.

Figure 18 [See the PDF.]

Figure 18: CDF: Loss of Connectivity for Global Repair and Global+Local Repair, Source Rate 10 Seconds/Packet.

Figure 19 [See the PDF.]

Figure 19: CDF: Loss of Connectivity for Global Repair and Global+Local Repair, Source Rate 20 Seconds/Packet.

A comparison between the amount of control plane overhead used for global repair only and for the global plus local repair mechanism is shown in Figure 20, which highlights the improved performance of RPL in terms of convergence time at very little extra overhead. From Figure 19, in 85% of the cases the protocol finds connectivity to the LBR for the concerned nodes within a fraction of seconds when local repair is employed. Using only global repair leads to repair periods of 150-154 seconds, as observed in Figures 13 and 14.

Figure 20 [See the PDF.]

Figure 20: Number of Control Packets for Different DAG Sequence Number Period, for Both Global Repair and Global+Local Repair.

5. RPL in a Building Automation Routing Scenario

Unlike the previous traffic pattern, where a majority of the total traffic generated by any node is destined to the root, this section considers a different traffic pattern, which is more prominent in a home or building routing scenario. In the simulations shown below, the nodes send 60% of their total generated traffic to the physically 1-hop distant node and 20% of traffic to a 2-hop distant node; the other 20% of traffic is distributed among other nodes in the network. The CDF of path quality metrics such as hop count, ETX path cost, average hop distance stretch, ETX path stretch, and delay for P2P routing for all pairs of nodes is calculated. Maintaining a low delay bound for P2P traffic is of high importance, as applications in home and building routing typically have low delay tolerance.

5.1. Path Quality

Figure 21 shows the CDF of the hop count for both RPL and ideal shortest path routing for the traffic pattern described above. Figure 22 shows the CDF of the expected number of transmissions (ETX) for each packet to reach its destination. Figures 23 and 24 show the CDF of the stretch factor for these two metrics. To illustrate the stretch factor, an example from Figure 24 will be given next. For all paths built by RPL, 85% of the time, the path cost is less than the path cost for the ideal shortest path plus one.

Figure 21 [See the PDF.]

Figure 21: CDF of End-to-End Hop Count for RPL and Ideal Shortest Path in Home Routing.

Figure 22 [See the PDF.]

Figure 22: CDF of ETX Path Cost Metric for RPL and Ideal Shortest Path in Home Routing.

Figure 23 [See the PDF.]

Figure 23: CDF of Hop Distance Stretch from Ideal Shortest Path.

Figure 24 [See the PDF.]

Figure 24: CDF of ETX Metric Stretch from Ideal Shortest Path.

5.2. Delay

To get an idea of maximum observable delay in the above-mentioned traffic pattern, the delay for different numbers of hops to the destination for RPL is considered. Figure 25 shows how the end-to-end packet latency is distributed for different packets with different hop counts in the network.

Figure 25 [See the PDF.]

Figure 25: Packet Latency for Different Hop Counts in RPL.

For this deployment scenario, 60% of the traffic has been restricted to a 1-hop neighborhood. Hence, intuitively, the protocol is expected to yield path qualities that are close to those of ideal shortest path routing for most of the paths. From the CDF of the hop count and ETX path cost, it is clear that peer-to-peer paths are more often closer to an ideal shortest path. The end-to-end delay for distances within 2 hops is less than 60 ms for 99% of the delivered packets, while packets traversing 5 hops or more are delivered within 100 ms 99% of the time. These results demonstrate that for a normal routing scenario of an LLN deployment in a building, RPL performs fairly well without incurring much control plane overhead, and it can be applied for delay-critical applications as well.

6. RPL in a Large-Scale Network

In this section, we focus on simulating RPL in a large network and study its scalability by focusing on a few performance metrics: the latency and path cost stretch, and the amount of control packets. The 2442-node smart meter network with its corresponding link traces was used in this scalability study. To simulate a more realistic scenario for a smart meter network, 100% of the packets generated by each node are destined to the root. Therefore, no traffic is destined to nodes other than the root.

6.1. Path Quality

To investigate RPL’s scalability, the CDF of the ETX path cost in the large-scale smart meter network is compared to a hypothetical ideal shortest path routing protocol that minimizes the total ETX path cost (Figure 26). In this simulation, the path stretch is also calculated for each packet that traverses the network. The path stretch is determined as the difference between the path cost taken by a packet

while following a route built via RPL and a path computed using an ideal shortest path routing protocol. The CDF of the ETX fractional stretch, which is determined as the ETX metric stretch value over the ETX path cost of an ideal shortest path, is plotted in Figure 27.

The fractional hop distance stretch value, as defined in the Terminology section, is shown in Figure 28.

Looking at the path quality plots, it is obvious that RPL works in a non-optimal fashion in this deployment scenario as well. However, on average, for each source-destination pair, the ETX fractional stretch is limited to 30% of the ideal shortest path cost. This fraction is higher for paths with shorter distances and lower for paths where the source and destination are far apart. The negative stretch factor for the hop count is an interesting feature of this deployment and is due to RPL's decision to not switch to another parent where the improvement in path quality is not significant. As mentioned previously, in this implementation, a node will only switch to a new parent if the advertised ETX path cost to the LBR through the new candidate parent is 20% better than the old one. The nodes tend to hear DIOs from a smaller hop count first, and later do not always shift to a larger hop count and smaller ETX path cost. As the traffic is mostly to the DAG root, some P2P paths built via RPL do yield a smaller hop count from source to destination, albeit at a larger ETX path cost.

As observed in Figure 26, 90% of the packets transmitted during the simulation have a (shortest) ETX path cost to destination less than or equal to 12. However, via RPL, 90% of the packets will follow paths that have a total ETX path cost of up to 14. Though all packets are destined to the LBR, it is to be noted that this implementation ignores a change of up to 20% in total ETX path cost. Figures 27 and 28 indicate that all paths have a very low ETX fractional stretch factor as far as the total ETX path cost is concerned, and some of the paths have lower hop counts to the LBR or DAG root as well when compared to the hop count of the ideal shortest path.

Figure 26 [See the PDF.]

Figure 26: CDF of Total ETX Path Cost versus ETX Path Cost.

Figure 27 [See the PDF.]

Figure 27: CDF of ETX Fractional Stretch versus ETX Fractional Stretch Value.

Figure 28 [See the PDF.]

Figure 28: CDF of Fractional Hop Count Stretch.

6.2. Delay

Figure 29 shows how end-to-end packet latency is distributed for different hop counts in the network. According to [RFC5548], Urban LLNs (U-LLNs) are delay tolerant, and the information, except for critical alarms, should arrive within a fraction of the reporting interval (within a few seconds). The packet generation for this deployment has been set higher than usual to incur high traffic volume, and nodes generate data once every 30 seconds. However, the end-to-end latency for most of the packets is condensed between 500 ms and 1 s, where the upper limit corresponds to packets traversing longer (greater than or equal to 6 hops) paths.

Figure 29 [See the PDF.]

Figure 29: End-to-End Packet Delivery Latency for Different Hop Counts.

6.3. Control Packet Overhead

Figure 30 shows the comparison between data packets (originated and forwarded) and control packets (DIO and DAO messages) transmitted by each node (link ETX is used as the routing metric). Here one can observe that in spite of the large scale of the network, the amount of control traffic in the protocol is negligible in comparison to data packet transmission. The smaller node ID for this network actually indicates closer proximity to the DAG root, and nodes with high ID numbers are actually farther away from the DAG root. Also, as expected, we can observe in Figures 31, 32, and 33 that the (non-leaf) nodes closer to the DAG root have many more data packet transmissions than other nodes. The leaf nodes have comparable amounts of data and control packet transmissions, as they do not take part in routing the data. As seen before, the data traffic for a child node has much less variation than the nodes that are closer to the DAG root. This variation decreases with increase in DAG depth. In this topology, Nodes 1, 2, and 3, etc., are direct children of the LBR.

Figure 30 [See the PDF.]

Figure 30: Data and Control Packet Comparison.

Figure 31 [See the PDF.]

Figure 31: Data and Control Packets over Time for Node 1.

Figure 32 [See the PDF.]

Figure 32: Data and Control Packets over Time for Node 78.

Figure 33 [See the PDF.]

Figure 33: Data and Control Packets over Time for Node 300.

In Figure 34, the effect of the global repair period timer on control packet overhead is shown.

Figure 34 [See the PDF.]

Figure 34: Numbers of Control Packets for Different Global Repair Timer Periods.

7. Scaling Property and Routing Stability

An important metric of interest is the maximum load experienced by any node (CPU usage) in terms of the number of control packets transmitted by the node. Also, to get an idea of scaling properties of RPL in large-scale networks, it is also key to analyze the number of packets handled by the RPL nodes for networks of different sizes.

In these simulations, at any given interval, the node with maximum control overhead load is identified. The amount of maximum control overhead processed by that node is plotted against time for three different networks under study. The first one is Network 'A', which has 45 nodes and is shown in Figure 1 (Section 3); the second is Network 'B', which is another deployed outdoor network with 86 nodes; and the third is Network 'C', which is the large deployed smart meter network with 2442 nodes as noted previously in this document.

In Figure 35, the comparison of maximum control loads is shown for different network sizes. For the network with 45 nodes, the maximum number of control packets in the network stays within a limit of 50 packets (per 1-minute interval), where for the networks with 86 and 2442 nodes, this limit stretches to 100 and 2 * 10^3 packets per 1-minute interval, respectively.

Figure 35 [See the PDF.]

Figure 35: Scaling Property of Maximum Control Packets Processed by Any Node over Time.

For a network built with low-power devices interconnected by lossy links, it is of the utmost importance to ensure that routing packets are not flooded in the entire network and that the routing topology stays as stable as possible. Any change in routing information, especially parent-child relationships, would reset the timer, leading to emitting new DIOs, and would hence change the node's path metric to reach the root. This change will trigger a series of control plane messages (RPL packets) in the DODAG. Therefore, it is important to carefully control the triggering of DIO control packets via the use of thresholds.

In this study, the effect of the tolerance value that is considered before emitting a DIO reflecting a new path cost is analyzed. Four cases are considered:

- No change in DAG depth of a node is ignored;
- The implementation ignores a 10% change in the ETX path cost to the DAG root. That is, if the change in total path cost to the root/LBR -- due to DIO reception from the most preferred parent or due to shifting to another parent -- is less than 10%, the node will not advertise the new metric to the root;
- The implementation ignores a 20% change in ETX path cost to the DAG root for any node before deciding to advertise a new depth;
- The implementation ignores a 30% change in the total ETX path cost to the DAG root of a node before deciding to advertise a new depth.

This decision does affect the optimum path quality to the DAG root. As observed in Figure 36, for 0% tolerance, 95% of paths used have an ETX fractional stretch factor of less than 10%. Similarly, for 10% and 20% tolerance levels, 95% of paths will have a 15% and 20% ETX fractional path stretch. However, the increased routing stability and decreased control overhead are the profit gained from the 10% extra increase in path length or ETX path cost, whichever is used as the metric to optimize the DAG.

Figure 36 [See the PDF.]

Figure 36: ETX Fractional Stretch Factor for Different Tolerance Levels.

As the above-mentioned threshold also affects the path taken by a packet, this study also demonstrates the effect of the threshold on routing stability (number of times P2P paths change between a source and a destination). For Network 'A' (shown in Figure 1) and the

large smart meter network 'C', the CDF of path change is plotted in Figures 37 and 38, respectively, against the fraction of path change for different thresholds (triggering the emission of a new DIO upon path cost change).

If X packets are transferred from source A to destination B, and out of X times, Y times the path between this source-destination pair is changed, then we compute the fraction of path change as Y/X * 100%. This metric is computed over all source-destination pairs, and the CDF is plotted in the y axis.

Figure 37 [See the PDF.]

Figure 37: Distribution of Fraction of Path Change for Network A.

Figure 38 [See the PDF.]

Figure 38: Distribution of Fraction of Path Change for Large Network C.

This document also compares the CDF of the fraction of path change for three different networks -- A, B, and C. Figure 39 shows how the three networks exhibit a change of P2P path when a 30% change in metric cost to the root is ignored before shifting to a new parent.

Figure 39 [See the PDF.]

Figure 39: Comparison of Distribution of Fraction of Path Change.

8. Comments

All the simulation results presented in this document corroborate the expected protocol behavior for the topologies and traffic model used in the study. For the particular discussed scenarios, the protocol is shown to meet the desired delay and convergency requirements and to exhibit self-healing properties without external intervention, incurring negligible control overhead (only a small fraction of data traffic). RPL provided near-optimum path quality for most of the packets in the scenarios considered here and is able to trade off control overhead for path quality via configurable parameters (such as decisions on when to switch to a new parent), as per the application and device requirements; thus, RPL can trade off routing stability for control overhead as well. Finally, as per the requirement of urban LLN deployments, the protocol is shown to scale to larger topologies (several thousand nodes), for the topologies considered in this implementation.

9. Security Considerations

This document describes investigations performed in the Castalia wireless sensor network simulator; it does not consider packets on the Internet. [RFC6550] describes security considerations for RPL networks.

10. Acknowledgements

The authors would like to acknowledge Jerald P. Martocci, Mukul Goyal, Emmanuel Monnerie, Philip Levis, Omprakash Gnawali, and Craig Partridge for their valuable and helpful suggestions over metrics to include and overall feedback.

11. Informative References

[Castalia-2.2] Boulis, A., "Castalia: Revealing pitfalls in designing distributed algorithms in WSN", Proceedings of the 5th international conference on Embedded networked sensor systems (SenSys'07), pp. 407-408, 2007.

[NS-2] "The Network Simulator version 2 (ns-2)", <http://www.isi.edu/nsnam/ns/>.

[OMNeTpp] Varga, A., "The OMNeT++ Discrete Event Simulation System", Proceedings of the European Simulation Multiconference (ESM'2001), June 2001.

[RFC5548] Dohler, M., Ed., Watteyne, T., Ed., Winter, T., Ed., and D. Barthel, Ed., "Routing Requirements for Urban Low-Power and Lossy Networks", RFC 5548, May 2009.

[RFC5673] Pister, K., Ed., Thubert, P., Ed., Dwars, S., and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks", RFC 5673, October 2009.

[RFC5826] Brandt, A., Buron, J., and G. Porcu, "Home Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5826, April 2010.

[RFC5867] Martocci, J., Ed., De Mil, P., Riou, N., and W. Vermeylen, "Building Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5867, June 2010.

[RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko, "The Trickle Algorithm", RFC 6206, March 2011.

[RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, March 2012.

[RFC6551] Vasseur, JP., Ed., Kim, M., Ed., Pister, K., Dejean, N., and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks", RFC 6551, March 2012.

[ROLL-TERMS] Vasseur, JP., "Terminology in Low power And Lossy Networks", Work in Progress, September 2011.

Authors' Addresses

Joydeep Tripathi (editor)
Drexel University
3141 Chestnut Street 7-313
Philadelphia, PA 19104
USA

EMail: jt369@drexel.edu

Jaudelice C. de Oliveira (editor)
Drexel University
3141 Chestnut Street 7-313
Philadelphia, PA 19104
USA

EMail: jau@coe.drexel.edu

JP. Vasseur (editor)
Cisco Systems, Inc.
11, Rue Camille Desmoulins
Issy Les Moulineaux 92782
France

EMail: jpv@cisco.com