Bounding Degrees on RPL

Fadwa Boubekeur
Sorbonne Universités, UPMC
Université Paris 6, LIP6,
F-75005, Paris, France
fadwa.boubekeur@lip6.fr

Lélia Blin
Université
d'Evry-Val-d'Essonne
F-91000, Evry, France
CNRS, UMR 7606, LIP6,
F-75005, Paris, France
lelia.blin@lip6.fr

Remy Leone
Institut Mines-Télécom /
Télécom ParisTech / CNRS
LTCI UMR 5141 - Thales
Communications & Security
remy.leone@telecom-paristech.fr

Paolo Medagliani
Mathematical and Algorithmic
Sciences Lab, France
Research Center, Huawei
Technologies Co. Ltd.
paolo.medagliani@huawei.com

ABSTRACT

RPL is an open routing protocol standardized by the ROLL group of IETF for constrained IP smart objects. It is one of the emergent protocols dedicated for Low Power and Lossy Networks (LLNs). Unfortunately, RPL suffers from significant packet loss due to the instability of the routes, and from a poor updates. Most of the existing solutions dedicated to solve the routes instability are based on improving the metrics used for constructing the routes. Generally these metrics are based on some evaluation of the radio link quality. In this paper, we adopt a new approach for addressing route instability in RPL, by placing an additional constraint on the maximum number of children a node can accept during tree construction. We call our solution Bounded Degree RPL (BD-RPL). BD-RPL addresses the absence of updating in the downward routes construction. Technically, we use the existing control messages provided by RPL for bounding the node degrees, as well as for updating the downward routes. Therefore, BD-RPL does not generate any additional overhead compared to RPL. Also, BD-RPL does not depend on the radio link quality metric. That is, any improvement of the metric used for RPL will automatically yield an improvement for BD-RPL. We have evaluated BD-RPL using the Cooja simulator, and implemented it on the Iot-lab platform. The experimentation demonstrates an improvement over RPL by an average of 10% in packet delivery, 50% in energy consumption, and 60% in delay.

Keywords

Wireless Sensor Networks; Performance evaluation; RPL protocol; IPV6, Bounded degree; tree

Q2SWinet'15, November 2–6, 2015, Cancun, Mexico.

DOI: http://dx.doi.org/10.1145/2815317.2815339.

1. INTRODUCTION

Smart objects are now a reality in many civilian applications, especially for monitoring remote and harsh environments where human intervention is difficult or, in some cases, impossible. Due to the nature of typical operational scenarios, these networks are often referred to as *Low power and Lossy Networks* (LLNs). Objects, whose commercial cost is around 1$, can be easily deployed in a redundant way in order to guarantee system workability even in case of failure, reducing the cost of maintenance intervention.

However, LLNs devices suffer from a limited computational power and memory storage, and are battery equipped. For this reason, traditional Internet protocols must be adapted in order to run on constrained devices. One of the main critical issues for LLNs is to provide network flexibility and reconfigurability with a very limited overhead. Every exchanged message causes energy consumption due to radio-frequency interface operations and packet processing. For such a reason, the smaller the number of control messages, the longer the network lifetime.

The *Internet Engineering Task Force* (IETF) is making a big effort for the standardization of protocols that are aware of energy and processing limitations within the smart objects. In particular, the IETF working group for *Routing Over Low-power and Lossy networks* (ROLL) focuses on the design of an IPv6 *Routing Protocol for Low power and lossy networks* (RPL) [21]. The working group aims at developing a multi-hop routing protocol for dense wireless networks that scales with network size while maintaining energy efficiency. RPL is an IPv6 distance vector protocol that proactively builds a logical topology based on a *Destination Advertisement Directed Acyclic Graph* (DODAG) tree. The constructed DODAG provides the upward routes from nodes to the DODAG root as well as the downward routes from the DODAG root to the leaves of the tree.

However, some works in [9, 12] pointed out that the topology built by the protocol is not stable, affecting the performance of the protocol. For such a reason, in this paper we focus on a mechanism that provides a higher topology stability in order to reduce packet losses and transmission delay while keeping the energy consumption low. To the best of our knowledge, this is the first work that, in order to guarantee higher performance, aims at reducing the variability of the DODAG tree created by the protocol by taking into account the impact of downward routes creation on the performance of the RPL protocol. In particular, we propose to bound the number of downward routes that each node can accept up to k children. The resulting DODAG is then a k-degree tree. This protocol will be referred to as *Bounded Degree RPL* (BD-RPL).

We leverage on standard RPL control messages that are exchanged by nodes to minimize additional overhead for topology control. Running BD-RPL results in a more stable network topology, as well as a better load distribution. Moreover, the mechanism actually implemented in RPL is not correctly handling the update of downward routes as these are never removed from memory. In BD-RPL, instead, we implement a message exchange scheme to fairly remove the non-used downward routes when a node changes preferred parent.

We implemented BD-RPL in Contiki and we evaluated network performance by simulations in Cooja and real experimentations on the Iot-lab testbed. Our results shows that BD-RPL provides for a two-order of magnitude stabler network. As nodes spend less time in trying to adapt to network conditions, the packet delivery ratio is higher, the delay experienced by nodes is smaller, and the energy consumed by nodes is lower as well. In addition, as the topology creation mechanism adopted by BD-RPL is more fair, the resulting network topology is more structured and the energy consumption is more distributed across the nodes in the network. Given that network lifetime is often evaluated as the time before a part of the network becomes disconnected, increasing the lifetime of the most solicited nodes entails a larger overall network lifetime.

The rest of this paper is organized as follows. Section 2 presents related works. Section 3 introduces a brief overview on the RPL protocol and the problem statement. Section 4 explains our solution BD-RPL. The evaluation results are shown in Section 5. Finally, section 6 concludes this paper.

2. RELATED WORK

Since the standardization of the RPL protocol in 2009, several works were proposed to enhance its performance. Some of them focused on the definition of new routing metrics [10, 15]. Authors in [10] design a new objective function that combines several metrics, such as ETX, end-to-end delay, hop count, and battery level of nodes, in order to compute the best route. Authors in [15] propose metrics which guide the interactions between IEEE 802.15.4 MAC mechanism and the RPL protocol. They also propose an approach to select the appropriate routing metric and adapt the corresponding MAC parameters in order to minimize the energy consumption in the network.

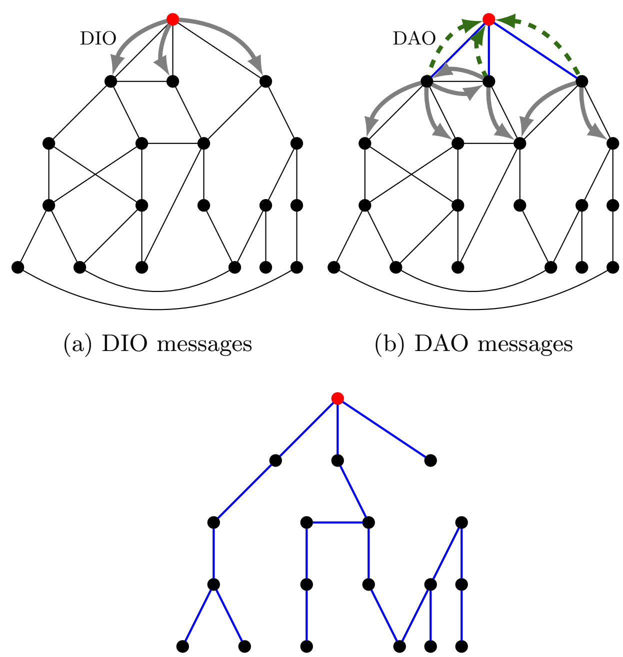
Other works focused, instead, on cross-layer solutions to improve routing efficiency [19, 11]. Authors in [19] adapt the trickle timer to govern the DIO messages to pass routing information about the RPL topology to the synchronous underlying 802.15.4 MAC layer. Authors in [11] consider an asynchronous duty-cycled MAC protocol. In order to reduce the latency, they propose to exploit the DODAG built by RPL to align each node wake-up phase with that of its preferred parent. Authors in [20] propose a preamble-sampling MAC protocol to enhance the performance of the RPL protocol. The proposed scheme is resilient to lossy links because the selection of the preferred parent is dynamic and based on channel conditions and status of the nodes. An opportunistic routing scheme is proposed in [16]. In their solution, before transmitting data, a node dynamically elects its preferred parent according to a radio link quality indicator.

To the best of our knowledge, there is only one work that deals with the excess of downward routes [13]. In this paper, the authors consider that maintaining downward routes to all nodes is expensive in terms of both overhead and memory utilization. They propose to use, according to the specific need, either a proactive scheme or a reactive scheme to build the downward routes. Their approach reduces the overhead and the energy consumption. In this paper, unlike to [13], we entirely conserve the proactive nature of the RPL protocol, i.e. the control messages exchanged by nodes, in order to keep the RPL overhead as low as possible. Moreover, the augmented stability of the tree will enable considerable energy saving as well as delay reduction.

3. ROUTING PROTOCOLS FOR LOW POWER AND LOSSY NETWORKS

The IETF provides solutions for low-power networking in WSNs. Actually the most widely adopted protocols for LLNs is RPL, whose main features will be described in the remaining of this section.

3.1 RPL overview



(a) DIO messages (b) DAO messages

(c) DODAG

Figure 1: RPL overview

RPL is an IPv6 distance vector routing protocol for LLNs that proactively builds a *Destination Oriented Directed Acyclic Graph* (DODAG) tree, as shown in Figure 1(c). RPL leverages on ICMPv6 control messages (DIS, DIO, DAO, DAO-ACK), a metric of reference (ETX, Energy, Hops, . . .), and a set of rules referred to as *objective function*.

The RPL protocol supports three models of data forwarding: (i) Multipoint To Point Model (MP2P), (ii) Point To Multipoint Model (P2MP), and (iii) Point to Point. In this work, we focus on MP2P and P2MP models. In MP2P, each node in the graph needs to send packets to the root of the DODAG using the upward routes that constitute the DODAG topology. In order to build the DODAG, the root sends multicast messages, referred to as *Destination Information Object* (DIO), to its neighbors (solid arrows in Figure 1(a)). The DIO messages include the routing metric, the rank value of the node, and the value of the objective function. In particular, the rank of a node is a measure of the distance of a node from the root of the DODAG according to the given metric. Being additive, the rank increases from the root to the leaves. When a node receives a DIO message from several potential parents, it chooses its parent, referred to as *preferred parent*, according to value computed by the objective function on the given metric. Its rank is then computed starting from the rank of its parent. The node also keeps track of a set of alternative parents to replace its preferred parent in case this latter is not reachable or when the metric of the link to its preferred parent becomes poor. After a node has computed its rank and its preferred parent, it broadcasts updated DIO messages to its neighbors that will repeat this process until when the full DODAG is built.

RPL also supports the P2MP model where the root sends packets to leaves nodes in the DODAG through downward routes. The downward routes are established via ICMPv6 control messages, referred to as *Destination Advertisement Object* (DAO). The DAO messages go through the DODAG (as shown by dashed lines in Figure 1(b)) and are used to advertise destination information towards the leaves. For the construction of downward routes, RPL can operate in two modes, as reported in the DIO messages: *storing* mode and *non-storing* mode. In storing mode, the non-leaf and non-root nodes can store a routing table. On the other hand, in non-storing mode, all RPL nodes (except for the root) don't support this feature. In this case, only the root is responsible for storing the complete set of downward routes. In this paper, we will only consider storing mode.

The construction of downward routes in storing mode is carried out as follow. Let consider a node v after the selection of its preferred parent p in the DODAG. v sends a DAO message to p with its prefix information. When p receives the DAO message, it adds the route of v to its routing table and sends an acknowledgment, referred to as *Destination Advertisement Object ACKnowledgment* (DAO-ACK), to v. Then, p generates a DAO with the updated route and sends, in its turn, the DAO message to its own preferred parent. The process continues until the DAO reaches the root. As a result, the root stores the complete route toward v.

It is important to note that RPL transmits the DIO and DAO messages using a mechanism referred to as *Trickle timer* [14] that regulates the frequency of transmission of control messages to reduce overhead. When the DODAG is stable, the frequency of control messages is reduced.

3.2 Problem statement

Several performance evaluations have been made on the RPL protocol by simulations [18, 17, 12, 1] and by experimentations [9, 2]. These studies aim at understanding the behavior of the RPL protocol and evaluating its performance. Most of these studies have shown that, amongst the protocols available for LLNs, RPL has the best performance in terms of delay, energy consumption, and overhead. On the other hand, RPL seems to be one of the protocols that has the worst packet reception rate. The authors in [12] also showed that RPL has an unstable topology due to the non-optimal metric.

3.2.1 Impact of the instability of the topology

RPL topology instability is due to several causes. First, the nature of the metrics does not take into account the presence of other protocols running inside wireless nodes. In fact, due to the non-optimal metric, node v may decide to change its preferred parent even if this latter is still reachable. The node can take this decision because the quality of the metric of the preferred parent has decreased. This may happen, for instance, when some packets between v and its parent are lost or delayed. However, in beaconless mode this can be the result of the sleep phase of the radio interfaces of the nodes, introduced to increase the energy efficiency in the 802.15.4 standard. This means that the receiving node may not be available for packet reception at the moment of the transmission. Moreover, the change of parent has a domino effect. When a node v changes its parent, it chooses a parent with a worse metric. This metric is sent by v to each associated child w. If w has a candidate parent with better metric, w will then change its parent.

To understand the impact of this instability on the performance of RPL, we focus on performance indicators such as the number of parent changes and the number of packets lost. In particular, the former indicates the stability of the topology. To prove this behavior, we made a preliminary simulation of the RPL protocol in the Cooja simulator for a network with 50 nodes. Results are shown in Figure 2. Performance indicators are expressed as a function of simulation time.

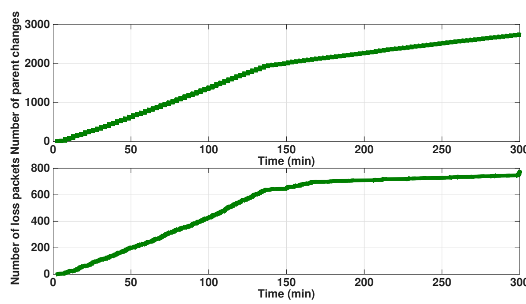


Figure 2: Number of parent changes and packet loss performance as a function of simulation time.

The number of packets lost in the second plot increases proportionately to the number of parent changes in the first plot. In addition, the two curves increase along the simulation even after the convergence of the DODAG, that is a few minutes of simulation roughly. The curves bring to light the strong relationship between the instability of the DODAG topology built by the RPL and the packets lost.

3.2.2 Impact of unbounded degree nodes

The topology construction does not take into account the number of children for one node, namely the degree of the node in the DODAG. However, this has an effect on the performance of the protocol in terms of: (i) packets lost,

(ii) energy consumption, and (iii) transmission delay. The number of children has an impact on the loss of packets because the collisions of packets sent to a node v increase with the number of children of v. The amount of energy consumption is due to the memory used by the node, as well as the number of packet transmissions and receptions. In this way, the higher the number of children of node v, the larger its memory occupation to store the routing table of its children, and the higher the number of received and sent messages. Moreover, the higher the number of children, the larger the delay because the number of collisions to send packets to parent p increases, thus transmission is slowed down. In this paper, our main objective is to control the stability of the DODAG by bounding the degree of nodes. The benefit of this approach is an enhancement of packets reception rate, an increased control of the topology and a better repartition of the energy consumption over nodes.

4. BD-RPL

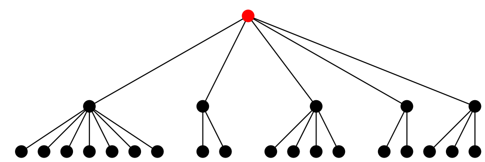
In this section, we will describe the proposed *Bounded Degree RPL* (BD-RPL) protocol. The network is modeled as an undirected connected graph $G(V, E)$. We define a constant $k < |V|$, where $|V|$ is the number of nodes in the graph. This constant number k is known by each node and represents the bound on the number of children can be accepted. In other words, k is the maximum degree of the DODAG. Remember that, the root is not bounded by k.

During the construction of the DODAG, each node v in the DODAG selects one preferred parent p and a set of potential alternative parents to build the upward route. After that, v sends a DAO to p in order to build the downward route. Our solution for bounding the degree of the DODAG intervenes at this level: each node v in the DODAG bounds up to k the number of children it can accept. This limitation only affects the number of next-hop children, while the number of children traversed in the whole downward routes is not bounded by BD-RPL.

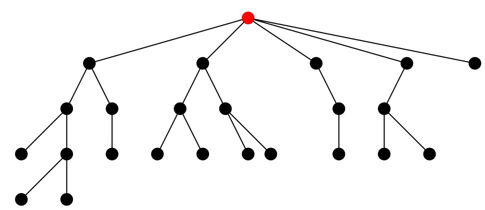
The protocol leverages on standard RPL control messages, such as DAO and DAO-ACK, to implement the bounded-degree feature. We modified the DAO-ACK messages by adding a new field that contains the denial or the acceptance of the parent. Therefore, our solution leverages on the existing control messages provided by RPL. We only introduce a minor overhead for the removal of the unused routes within the previous parent. When a node receives a deny from its parent, it sends a new message to another preferred parent. Our mechanism, executed by every node in the DODAG except the root, works as follow:

- As soon as a node v selects its preferred parent p in the DODAG, it sends a DAO message to p to establish the downward routes.
- When p receives the DAO message, it verifies the number of associated children. If this value is lower than k, it accepts v as child and it adds the route to v in its routing table. Then, p sends a DAO-ACK to v to acknowledge the association. However, if the number of children is already k, p does not accept v by sending a deny DAO-ACK with to v.
- When v receives the DAO-ACK message, if it is a DAO-ACK with an acceptance, v considers the upward route to p as established and stops sending DAO control messages. On the other hand, if the DAO-ACK message contains a deny, v chooses a new preferred parent p' in the set of its feasible parents and sends a new DAO message to p'.

It is necessary to clarify that bounding the degree of the node is a NP-hard problem [8], that may lead to partially connected networks. However, as in many practical LLNs networks the topology is sufficiently dense, full connectivity is guaranteed. In fact, as verified via both simulation and testbed, this problem never happens. Thus, the nodes have always a parent in a degree-bounded DODAG. Of course, the mechanism of bounding the downward routes also affects the upward routes. We point out that with BD-RPL the parent p chosen by node v is not necessarily the neighbor u with the best metric, as u may have denied v.



(a) RPL



(b) BD-RPL

Figure 3: RPL versus BD-RPL

Figure 3 presents an illustrative example of our solution. In Figure 3(a), where basic RPL is executed, the degree of nodes in the DODAG is not bounded. In Figure 3(b), instead, BD-RPL is executed. In this case, with $k = 2$, the topology is distributed more uniformly across the nodes in the network. In this way, although the number of packet relayed by nodes close to the sink is not changing, the impact of collisions and packets lost is reduced, resulting in a considerable energy saving. Note that the depth of the DODAG can increase in BD-RPL compared to RPL.

4.1 Impact of children bounding

The RPL protocol does not provide a mechanism to remove the non-used downward routes. When a parent node adds a downward route to its routing table, this route remains registered even when the child changes the preferred parent. This has an impact on network performance:

- The number of downward routes increases each time a parent node adds a new child. Effectively, for each added child, many downward routes can be registered, namely a downward route per child and others downward routes for the children of each child. However, previous non-used downward routes are not removed. This significant number of downward routes increases the size of used memory and, consequently, the energy consumption.
- With BD-RPL, when a parent node bounds its children to a maximum degree k, it only registers the k

first next-hop children and the downward routes associated with them. Then, new potential incoming children with new downward routes are not accepted as the maximum number of children has already been reached. So, only the downward routes associated with the k first children will be added.
- The number of parent changes reduces significantly. In fact, children nodes may conserve the same preferred parent for long time. This is because other potential parents may refuse them as they have already associated k children. In this way, only parents with less than k associated children will accept new nodes. Hence, this will prevent nodes to choose a new parent with better metric.

Bounding the degree of a node should affect as little as possible the choice of a link with a suitable metric. However, a parent with a good link quality could remain without children because it has already registered the k first children which, in a consecutive moment, joined other parents. The solution, described in Subsection 4.2, is to remove routes that are no longer used to leave room for nodes to associate with the most suitable available parent.

4.2 Update downward routes

In order to solve to the above-mentioned problems, we add into BD-RPL the update of downward routes. BD-RPL allows the parents nodes to update the list of their children. For that purpose, BD-RPL exploits the DAO control messages provided by the RPL protocol to specify a non-used downward routes. So, we modified RPL as follows:

- When a node v changes its preferred parent from p to p', it sends a DAO message with a non-used downward route to the parent p.
- When the node p receives the DAO message, it removes the child v and all the downward routes relying on v.

In this way, parent nodes remove the previous children and have free space to add new children with the associated downward routes. Our mechanism enhances the packet delivery in the downward direction because the messages are sent on updated routes. Furthermore, our mechanism allows the topology to evolve with the radio link variations. Nodes can, therefore, change parent when the link to the previous parent becomes poor. Finally, our mechanism removes all the non-used downward routes, thus reducing memory utilization and energy consumption.

5. EVALUATION

In this section, we evaluate the BD-RPL protocol with degree of the DODAG set to 3 through a Contiki-based implementation. . First, we run simulations on Cooja simulator. Then, we validate our results by experimentations on the Iot-lab platform. Note that we do not compare the performance of the root node, because, with BD-RPL, the root node does not bound its children. We compare the performance of BD-RPL with RPL in terms of:

Topology stability: number of parent changes per node.

Radio energy consumption: estimation of the radio consumed energy to send and receive messages.

Delay of transmission: time that takes for a packet to reach its destination. In the simulation, we calculate the average delay of transmission in the upward direction from DODAG nodes to the root and in the downward direction from the root node to other nodes. As in experimentation, the clock of nodes are not synchronized, we only consider end-to-end delay, computed at each node as the time required to send a packet and receive the associated acknowledgment.

Packet reception ratio: ratio between the number of packets received and the number of packets sent. This indicates the reliability of the routing protocol.

5.1 Set up

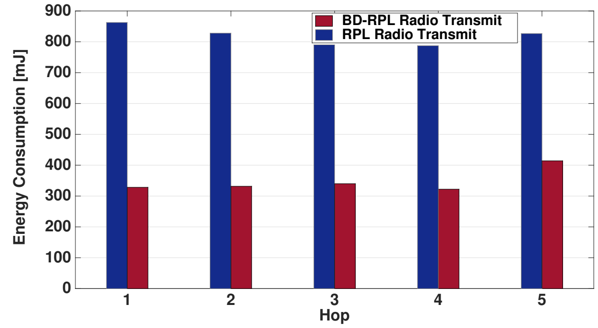
We have implemented BD-RPL on Contiki [5], an open-source operating system specific for constrained LLNs networks. Contiki is designed to use a small amount of memory while supporting a full-IP network stack, a 6LowPAN adaptation layer and an implementation of the RPL protocol. Contiki also supports ContikiMAC [4], a radio duty-cycling mechanism used to make nodes sleep most of the time and periodically wake-up, in order to carry out energy-aware listen and transmit operations. ContikiMAC runs on top of the beaconless IEEE 802.15.4 protocol. The metric of routing for the RPL protocol is the standard ETX [3]. All nodes in the network are configured in storing mode. We assume that nodes have a high transmission power. Thus, each node has several neighbors.

5.2 Simulation

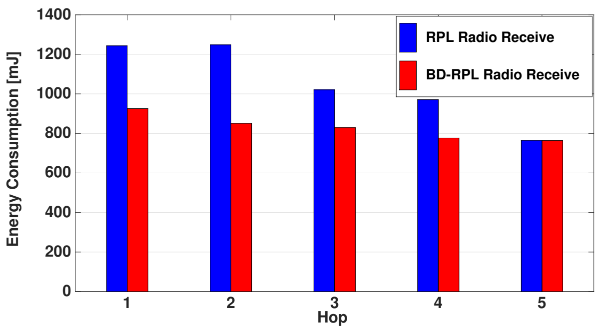
We run our simulation in COOJA [7], a simulator for wireless sensor networks. COOJA simulates each node as a TMOTE Sky platform based on a 16-bit MSP430 micro-controller and a CC2420 Chipcon radio interface at 2.4 GHz. The Radio Duty Cycling channel check rate is set to 4 Hz. In our setup, we consider 50 randomly-deployed nodes forming a fully connected network. The propagation model is a Unit Disk Graph Model with a transmission range and an interference range of 30 meters with a non lossy medium. Nodes begin the transmission of data packets after 2 minutes of simulation to let the DODAG tree be built. After this setup period, the nodes start sending UDP packets to the root node every 2 minutes. In addition, the root node acknowledges successful transmissions every 3 minutes. Each simulation is run 5 times to reduce statistical fluctuations and each run lasts for 5 hours.

Energy consumption Figure 4 presents the average radio energy consumption as a function of the hop distance from the sink for RPL and BD-RPL. Energy consumption is reduced with BD-RPL during both listen and transit phases. In particular, the energy consumption during the listening phase is reduced as a node has less associated children, while the energy consumption in the transmission phase is reduced as there are less collisions to transmit to the preferred parent. The advantage of BD-RPL is particularly prominent for the nodes closer to the sink. In addition, from Figure 4 it can be seen that with BD-RPL the energy consumed by the nodes is distributed among the nodes in the network, resulting in a more fair energy consumption and in a longer network lifetime.

Delay Figure 5 presents the delay of transmission in the upward direction as a function of the hop distance from the sink for RPL and BD-RPL. The upward delay is reduced because the average load on nodes is reduced as well. Consequently, the packets experience less collisions and wait for a lower time interval before being inserted in the wireless



(a) Radio Energy Transmit



(b) Radio Energy Receive

Figure 4: Radio Energy Consumption as a function of the hop distance from the sink for RPL and BD-RPL.

medium. As nodes have a limited number of children, the transmission to the parent become less competitive and the delay is reduced.

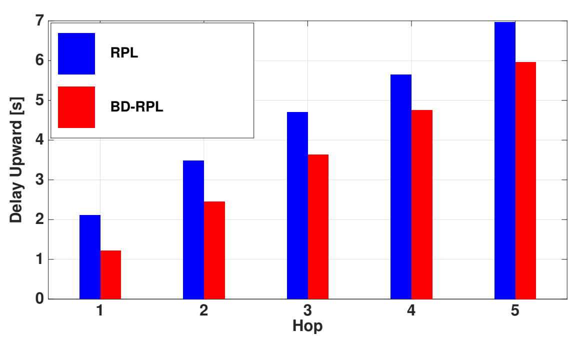


Figure 5: Upward Average Delay as a function of the hop distance from the sink for RPL and BD-RPL.

In Figure 6, we evaluate downward delay as a function of the hop distance from the sink for RPL and BD-RPL. With BD-RPL, downward delay is comparable to RPL for nodes close to the sink (up to 3 hops). For nodes deeper in the tree, instead, BD-RPL shows a better downward delay performance, as the number of collisions is lower with BD-RP.

Upward packet delivery ratio Figure 7 shows packet reception rate as a function of the hop distance from the sink for RPL and BD-RPL. With BD-RPL, the packet reception rate is enhanced by almost 10 % compared to original RPL.

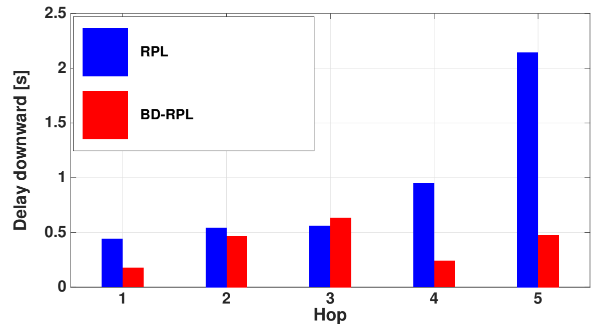


Figure 6: Average Delay Downward as a function of the hop distance from the sink for RPL and BD-RPL.

As for the delay, reducing the load on nodes results in a smaller loss of packets due to parent unavailability. In fact, according to the CSMA/CA protocol run on nodes, after a given number of unsuccessful transmission attempts, the packet is discarded. If the load of the receiving node is low, this will be most probably available for packet reception, consequently increasing packet delivery ratio.

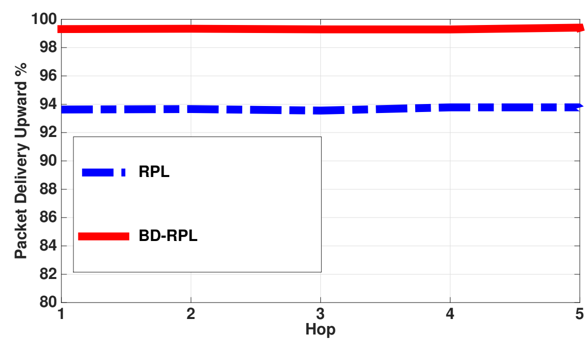


Figure 7: Upward Packet Delivery Ratio as a function of the hop distance from the sink for RPL and BD-RPL.

5.3 Experimentation

This subsection shows BD-RPL results with the Iot-lab platform. To avoid repeating, we will exclusively point out the differences between simulations and experimentations.

Setup and hypothesis For our experimentation of BD-RPL in IoT-lab, we considered 50 WSN430 static nodes. FIT IoT-lab [6] is a testbed designed to handle large-scale WSN experiments. Its main goal is to offer an accurate open-access multi-user scientific tool to support design, development, tuning, and experimentation related to IoT. IoT-lab features different kind of nodes such as WSN430, ARM Cortex M3 and ARM A8. Each node includes a gateway that provides connection to the global infrastructure of the IoT-LAB to flash, control and monitor the nodes both at runtime and before an experiment takes place.

In our experiments, we considered the CC2420 radio communication chipset, operating at 2.4 GHz and implementing the unslotted IEEE 802.15.4 standard. The Radio Duty Cycling channel check rate is set to 4 Hz. The transmission power of nodes is set to -1 dBm. The nodes start transmitting their data packets after 2 minutes to let the protocol build the tree. Every 2 minutes nodes send UDP packets to the sink node. The sink node acknowledges each received data packet. We run each experiment for 3 hours and we repeat it 6 times with the same node distribution.

Topology routes changes Figure 8 compares the average parent changes as a function of the hop distance from the sink. Results show that with BD-RPL we reduce the average number of parent changes by two orders of magnitude. We saw on this figure that, with BD-RPL, nodes that are one hop away from the root significantly reduce their number of parent changes. This is due to the refusal carried out by new potential preferred parent, forcing them to stay with their current parents. This confirms our first intuition that the excess of parent changes is not always a good behavior for the performance of the protocol.

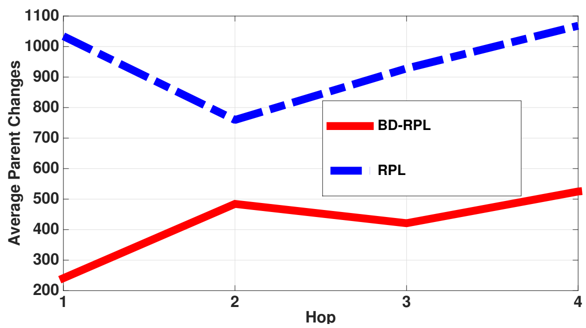
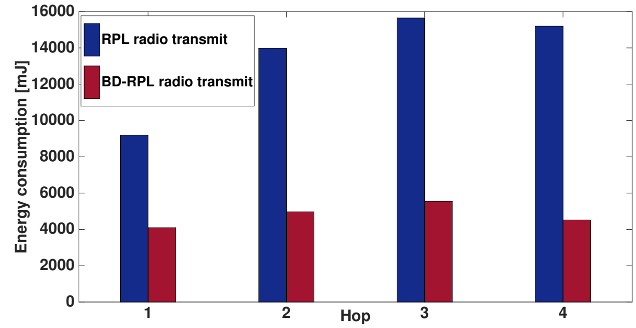


Figure 8: Average Parent Changes as a function of the hop distance from the sink for RPL and BD-RPL.

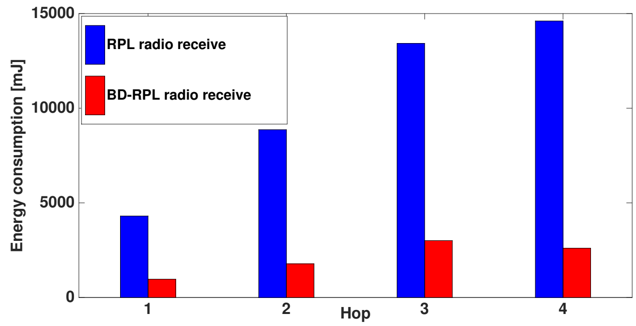
Energy consumption Figure 9 shows the energy consumption of nodes to send and receive packets as a function of the hop distance from the sink for RPL and BD-RPL. In the real testbed, the energy consumption is 10 times higher than in the simulation for both RPL and BD-RPL. Furthermore, as already shown in Figure 9, the energy consumption with BD-RPL is reduced by a factor of 2 compared to RPL. We remark that the nodes close to sink reduce their radio energy consumption in listening because they have less children to manage. The energy consumption in the transmission phase, instead, is reduced as there are less collisions to transmit to the preferred parent.

Delay of transmission Figure 10 represents the average end-to-end transmission delay as a function of the hop distance from the sink for RPL and BD-RPL. This figure shows that, as with simulation, BD-RPL reduces the end-to-end delay with respect to RPL. This effect is more evident for nodes that are close to the sink because they have less children to manage and packets are sent without additional waiting time.

Upward packet delivery ratio Results in Figure 11 shows that as in simulation, we enhance the upward packet delivery ratio. This is even more prominent than in simulations as the improvement is around 10%. However, we acknowledge that the loss of packets in real testbed is most important than the loss of packets in simulation due to higher interference from the external world that cannot be captured by simulators.



(a) Radio Energy Transmit



(b) Radio Energy Receive

Figure 9: Radio Energy Consumption as a function of the hop distance from the sink for RPL and BD-RPL.

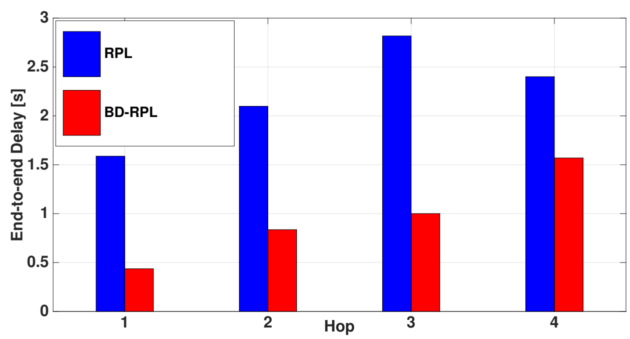


Figure 10: Delay as a function of the hop distance from the sink for RPL and BD-RPL.

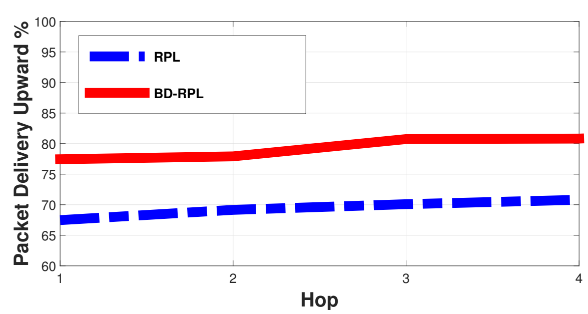


Figure 11: Packet Delivery Ratio Upward per Hop

6. CONCLUSION

As traditional RPL suffers from significant packet losses due to DODAG tree instability, in this paper we presented BD-RPL. This modification of the RPL protocol tackles route instability in LLNs by introducing a bound on the maximum number of children a node can accept during tree construction. Moreover, BD-RPL addresses the absence of updating in the downward routes construction.

In particular, we leverage on the existing control messages provided by RPL to bound nodes degree, as well as to update downward routes. Thus, BD-RPL adds a marginal additional overhead compared to RPL. In addition, as BD-RPL is agnostic of the considered routing metric, any improvement of the metric used for RPL will automatically yield an improvement for BD-RPL.

The simulations and the experiments proved an improvement over RPL by an average of 10% in packet delivery, 50% in energy consumption, and 60% in delay. We have evaluated BD-RPL using both the Cooja simulator and the Iot-lab platform.

Future research directions will take into account the evaluation of the protocol on larger scenarios (i.e., up to 1000 nodes) and the introduction of a mechanism that allows to orchestrate children transmissions to further reduce collisions and energy consumption.

7. REFERENCES

[1] N. Accettura, L.A. Grieco, G. Boggia, and P. Camarda. Performance Analysis of the RPL Routing Protocol. In *ICM*, 2011.

[2] T. H. Clausen, U. Herberg, and M. Philipp. A critical evaluation of the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL). In *WiMob*, pages 365–372, 2011.

[3] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A High-throughput Path Metric for Multi-hop Wireless Routing. In *MobiCom*, pages 134–146, 2003.

[4] A. Dunkels. The ContikiMAC Radio Duty Cycling Protocol, 2011.

[5] A. Dunkels, B. Gronvall, and T. Voigt. Contiki-a lightweight and flexible operating system for tiny networked sensors. In *LCN*, pages 455–462, 2004.

[6] E.Fleury, N.Mitton, T.Noel, and C.Adjih. FIT IoT-LAB: The Largest IoT Open Experimental Testbed. *ERCIM News*, page 14, 2015.

[7] J. Eriksson, F. Österlind, N. Finne, N. Tsiftes, A. Dunkels, T. Voigt, R. Sauter, and P. Marrón. COOJA/MSPSim: interoperability testing for wireless sensor networks. In *ICSTT*, page 27, 2009.

[8] M. Fürer and B. Raghavachari. Approximating the Minimum-Degree Steiner Tree to within One of Optimal. *J. Algorithms*, 17(3):409–423, 1994.

[9] O. Gaddour and A. Koubaa. RPL in a nutshell: A survey. *Computer Networks*, 56(14):3163–3178, 2012.

[10] O. Gaddour, A. Koubaa, N. Baccour, and M. Abid. OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol. In *WiOpt*, pages 365–372, 2014.

[11] P. Gonizzi, P. Medagliani, G. Ferrari, and J. Leguay. RAWMAC: A routing aware wave-based MAC protocol for WSNs. In *WiMob*, pages 205–212, 2014.

[12] O. Iova, F. Theoleyre, and T. Noël. Stability and efficiency of RPL under realistic conditions in Wireless Sensor Networks. In *PIMRC*, pages 2098–2102, 2013.

[13] CA. La, M. Heusse, and A. Duda. Link reversal and reactive routing in Low Power and Lossy Networks. In *PIMRC*, pages 3386–3390, 2013.

[14] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko. The Trickle Algorithm, 2011.

[15] P. Di Marco, C. Fischione, G. Athanasiou, and PV. Mekikis. MAC-aware routing metrics for low power and lossy networks. In *INFOCOM*, pages 13–14, 2013.

[16] B. Pavkovic, F. Theoleyre, and A. Duda. Multipath opportunistic RPL routing over IEEE 802.15.4. In *MSWiM*, pages 179–186, 2011.

[17] I. E. Radoi, A. Shenoy, and D. K. Arvind. Evaluation of Routing Protocols for Internet-Enabled Wireless Sensor Networks, 2012.

[18] J. Tripathi, J. Cavalcante de Oliveira, and JP. Vasseur. A performance evaluation study of RPL: Routing Protocol for Low power and Lossy Networks. In *CISS*, pages 1–6, 2010.

[19] M. Vucinic, G. Romaniello, L. Guelorget, B. Tourancheau, F. Rousseau, O. Alphand, A. Duda, and L. Damon. Topology construction in RPL networks over beacon-enabled 802.15.4. *CoRR*, 2014.

[20] T. Watteyne, M. R. Akhavan, and A. H. Aghvami. Enhancing the performance of RPL using a receiver-based MAC protocol in lossy WSNs. In *ICT*, pages 191–194, 2011.

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