Performance comparative analysis of LOADing-CTP and RPL routing protocols for LLNs

Belghachi Mohammed, Feham Mohamed

***Abstract*—**Low Power and Lossy Networks (LLNs) represent one of the interesting research areas in recent years. The IETF ROLL and 6LoWPAN working groups have developed new IP based protocols for LLNs. In LLNs e.g. 6LoWPANs, heavy data traffic causes congestion which significantly degrades network performance. In this paper we study two routing protocols philosophies for low power and lossy networks (LLNs). This study purposes a detailed evaluation of two routing protocols proposed by IETF, RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) and the reactive LOADng-CTP specified by an IETF draft extended with a collection tree for efficient data acquisition in LLNs. We performed checks on control overhead; End to End Delay and Packet delivery ratio for the two protocols related to multipoint-to-point (MP2P), and point-to-multi point (P2MP) traffic flow.

***Keywords*—**RPL; LOADng; LOADng-CTP; Performance; Simulation; Contiki OS.

I. Introduction

Low power and Lossy Networks (LLNs) are a class of network in which both the routers and their interconnect are constrained [1]: LLN routers typically operate with constraints on (any subset of) processing power, memory and energy (battery), and their interconnects are characterized by (any subset of) high loss rates, low data rates and instability. LLNs are comprised of anything from a few dozen and up to thousands of LLN routers, and support point-to-point traffic (between devices inside the LLN), point-to-multipoint traffic (from a central control point to a subset of devices inside the LLN) and multipoint-to-point traffic (from devices inside the LLN towards a central control point).

In this manuscript we make the following contributions: We evaluate the implementations of LOADng-CTP in Contiki OS and compare it to RPL protocol for bidirectional scenarios in LLNs network architecture. We provide simulation results for the network End-to-End delay, PDR, overhead and show how the implemented LOADng-CTP solution can provide bidirectional data flow scalability [2].

Belghachi Mohammed is with the Computer Science Department, University of Bechar, Bechar 08000, Algeria

Feham Mohamed is with the University of Tlemcen, Algeria

The paper is structured as follows: Section II provides a detailed overview of RPL protocol. Respectively in sections III and IV we describe LOADng and LOADN-CTP protocols specifications. Section V presents detailed performance evaluation and simulations results, in section VI we conclude.

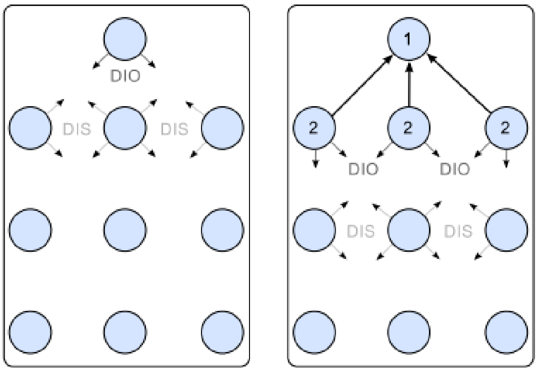
II. RPL Protocol

RPL is distance vector routing protocol for the LLNs which describes a method of construction of a logical topology called DODAG (Destination Oriented Directed Acyclic Graph) seen Fig1 using an objective function and a set of metrics and constraints. The objective function is based on a combination of metrics and constraints to calculate the "better acceptable path» [3].

A. Controls messages

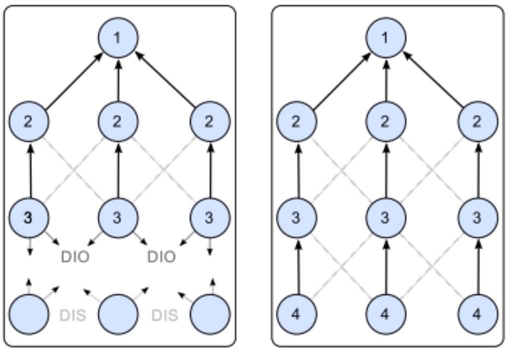
RPL uses the following controls messages:

- DIOs messages (DODAG Information Object) are sent by the RPL nodes to announce a DODAG and its characteristics. The DIOs messages are used for the discovery, the formation and the maintenance of a DODAG, Select all the parents, discover the RPL instance... So, they carry a set of essential information to allow a node to perform these actions.
- DISs messages (DODAG information Solicitation) are RPL messages used to solicit DIOs messages from a RPL node. A RPL node uses DISs to probe the nearest neighbors located in a DODAG.
- The DAOs messages (Advertisement Destination Object) is used to propagate for a destination towards UP in a DODAG. In the case of storage mode, the DAO message is unicasted from the wire to the selected parent. And in the case of non-storage mode [7], the DAO is unicasted toward the root of the DODAG.



(a) (b)

(a) Step1: Initial DIO Propagation
(b) Step 2: Route Establishment & DIO Propagation



(c) (d)

(c) Step 3: Route Establishment & DIO Propagation
(d) Step 4 : Resulting DODAG

Fig. 1. : Creation of Upward Routes in RPL – The rank of a node is denoted by the number in each circle. Default upward paths between two nodes are depicted by black arrows. Grey striped lines indicate optional paths. (a) Shows the multicast of DIS and the initial DIO propagation. In Figures (b) and (c) nodes join the DODAG, establish routes and propagate DIO messages. (d) Shows the resulting DODAG.

B. *Trickle Timers*

Control plane traffic load is a concern in LLNs where bandwidth and energy are often scarce. Periodic emission of control plane messages is not possible and the use of keep alive for routing adjacency maintenance is not a suitable option. A different approach in RPL consists of controlling the control plane packets frequency update by using adaptive mechanisms controlled by the use of dynamic timers, referred to as trickle timers. DIO messages are "multicasted" on expiry of trickle timers thereby the DIO messages are sent more frequently when a DAG consistency issue is detected to improve the convergence time. As DAG stabilizes, messages are sent less frequently [4].

III. LOADng Protocol

LOADng (LLN On-demand Ad-hoc Distance vector routing protocol – next generation) is a reactive routing protocol; the construction of the topology is based on route discovery by flooding control messages. When a frame needs to be transmitted a router node implementing LOADng executes the following actions: if no route is known to the destination, the node generates a Route Request (RREQ) message containing its address and the address of the destination and broadcast it in its radio neighborhood. This message is propagated hop by hop, by successive broadcastings, So as to flood the network and to be heard by the destination node. Once the RREQ message is received, the destination generates a Route Reply message (RREP) transmitted in unicast on the reverse path followed by the RREQ message. This path can be identified in two ways: by adding the addresses of the nodes traversed by the RREQ message or by storing the address of the node previously traversed at each node of the network. In LOADng, the second solution is retained, so favoring a reduced size of messages at the expense of the memory occupation of the nodes [5].

Once the route is established, the data frame is routed on the determined path and the storage of the source / destination pair at each node of the path makes it possible to avoid re-discovering each new data frame. According to this description, protocols with reactive construction of the topology are preferred for a deployment with few nodes and application traffic that is both low by its volume and by the diversity of sources and destinations, according to their operating hypotheses. Indeed, the establishment of each road requires the flooding of network-wide control messages. Moreover, in order to avoid the flooding of the network at each new application frame, each path discovered requires the installation and storage of the route on each node participating in the routing of the data [6].

IV. LOADng-CTP

LOADng Collection Tree Protocol (LOADng-CTP) is a recent extension of the LOADng protocol using a "collection tree" combined with the LOADng specification. The LOADng-CTP extension aims to build bidirectional routes for the flows MP2P and P2MP traffic in a sensor network, low overhead easy maintenance and better performance [7].

A. *Control messages*

LOADing-CTP uses the same control messages as LOADng with some modification on protocol operations; it introduces two Flag on RREQ messages [8]:

- RREQ COLLECTION TREE TRIGGER: This trigger allows routers to discover bi-directional paths in the vicinity.
- RREQ COLLECTION TREE BUILD: When set, it allows the receiving router to build the path to the root.

HELLO messages: HELLO messages are used to build the collection tree; they are broadcast by the root router and are never forwarded by 1-hop neighbours. These messages are used to identify bidirectional paths [9].

B. *LOADng-CTP parameters*

LOADng-CTP uses the following parameters:

- NET TRAVERSAL TIME: is the maximum allowed time (end-to-end) for a packet when moving in the network.
- RREQ MAX JITTER: is the maximum jitter of the RREQ message transmission
- HELLO MIN JITTER: is the minimum jitter for the transmission of HELLO messages. With the following condition HELLO MIN JITTER> 2 × RREQ MAX JITTER
- HELLOMAXJITTER: is the maximum jitter for HELLO message transmission.
- RREPREQUIRED: is a Flag to set if a RREP message is required to create paths from the root to the sensors when receiving a RREQ BUILD message.

C. Protocol operation

The basic operation of the protocol can be represented by four steps [10] seen Fig2:

- Step 1: At the beginning all flags are initiated and defined by the root router. The collection tree is created by the root node of the collection tree. The root of the collection tree generates RREQ TRIGER set to 1 [11]. The initiator and destination of the RREQ TRIGGER are set to the address of the root. When a RREQ TRIGGER is generated, a RREQ message with a RREQ BUILD flag is provided to be sent in 2 × NET TRAVERSAL TIME [12].
- Step 2: This step summarizes the neighbours' discovery phase; each router acquires the list of neighbours with bidirectional or unidirectional links and updates the routing table.
- Step 3: This step describes the process of building the collection tree. When a router receives a RREQ BUILD message, if this message has been received from a neighbour with a bidirectional the message is validated, the collection tree is constructed, the routing table is created and the Protocol are updated, otherwise the message is ignored.
- Step 4: The procedure for building the Root path to sensors is detailed in this step. By exchanging RREQ TRIGGER and RREQ BUILD messages, all nodes in the collection tree get a bidirectional path to the root. In fact the paths from the root to the sensors are constructed by using the Flag RREP REQUIRED defined by true and transmitted in unicast to the root and the routing tables are updated according to the established bidirectional path.

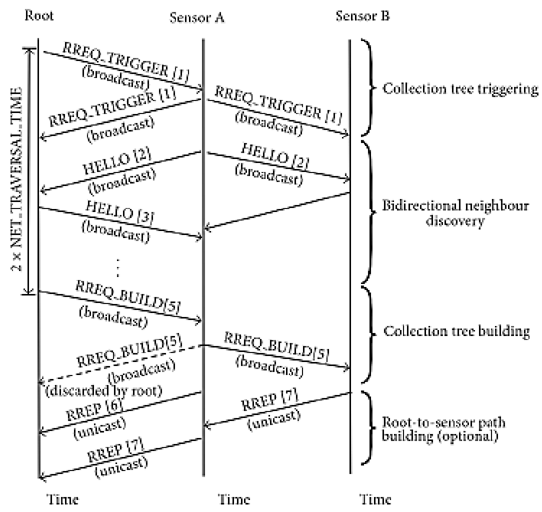


Fig. 2. Message exchange of LOADng-CTP between root and sensors.

V. PERFORMANCE EVALUATION

A. Simulation Settings

We evaluated the two routing protocols (RPL and LOADng-CTP) in terms of packet Delivery ratio (PDR), latency in order to predict how it behave in larger networks, and overhead to describe its power consumption and memory management. The detailed settings of the scenarios studied are detailed in table I the values have been averaged over 10 runs;

TABLE I. COOJA SIMULATOR PARAMETER SETUP

Settings Transport layer UDP	**Value**
Wireless channel model	UDGM Model with Distance Loss
Communication range	250m
Distance to the Concentrator	Variable [50-500] Meters
Grid Size	1000*1000 m2
Number of routers	Variable [20/300]
Mote type	Tmote Sky
Network layer	μIPv6 6LoWPAN
MAC layer	CSMA ContikiMAC
Radio interface	CC2420 2.4 GHz IEEE 802.15.4
Simulation time	8h

Simulations were completed in a field of 1000 × 1000 meters, with variable amounts of routers positioned randomly , were realistic conditions were approached and different nodes are suggested to interferences. The network scenarios are substance of two different traffic patterns: multipoint-to-point (MP2P), where all routers generate CBR traffic flow by periodic reporting of 512-byte data packet with 60 seconds interval and acknowledgment of each received frame in upward direction of 16 bytes payload, for which the destination always is the sink. And point-to- multipoint (P2MP) traffic with two messages types, acknowledgment of data frames in downward direction every data arrival of 12 bytes payloads and configuration data packet with CBR traffic flow by periodic message of 61-byte payload with 300 seconds interval in downward direction.

LOADng-CTP were implemented with C in contiki OS. The settings for RPL are listed in Table II, and for LOADng-CTP in Table III.

TABLE II. RPL PARAMETERS

Parameter	Value
Mode of operation	non-storing
Rank metric	hop count
DIOIntervalMin	2 s
DIOIntervalDoublings	20
DIORedundancyConstant	1
DAOInterval	15s

TABLE III. LOADNG-CTP PARAMETERS

Parameter	Value
RREQ jitter	0 - 0.5 s
RREQMAXJITTER	1 s
NETTRAVERSALTIME	10 s
HELLO MIN JITTER	3 s
HELLO MAX JITTER	5 s
Route lifetime	15 s
Routing	Mesh routing

B. Simulation Results

The packet delivery ratio is the ratio of packets successfully received to the total sent. Fig 3 shows the delivery ratio of the tow protocols. Both LOADng-CTP and RPL obtain delivery ratios close to 100%, regardless of number of nodes.

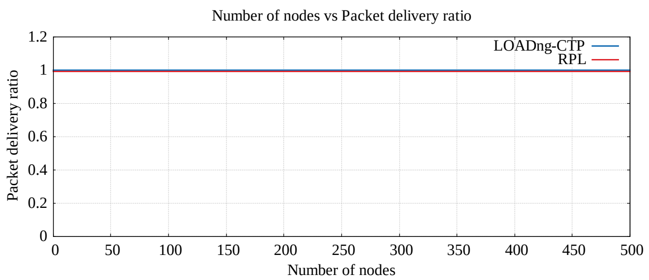


Fig. 3. Packet delivery ratio

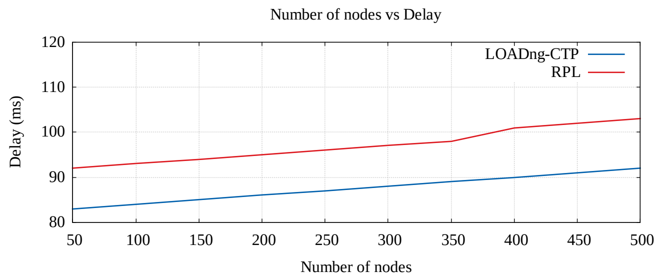


Fig. 4. Average end-to-end delay

Fig 4 shows that LOADng-CTP is very efficient in terms of average latency which is equal to 85 ms compared to RPL 94 ms. While when the network is with increasing number of nodes is variable latency is equal to 81 ms and 95 ms for LOADng-CTP and RPL respectively.

For the sensor networks, the routing overhead is also a crucial consideration. Fig 5 and Fig 6 shows the number of overhead packets per router and average overhead of network (bytes/second) respectively, which the networks are needed to converge to a stable state, i.e., every router has a route to the root.

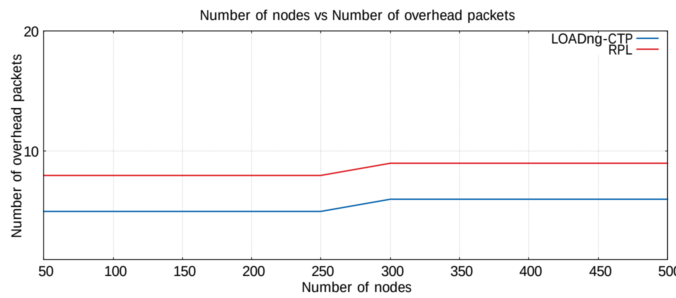


Fig. 5. Number of overhead packets transmitted by each router

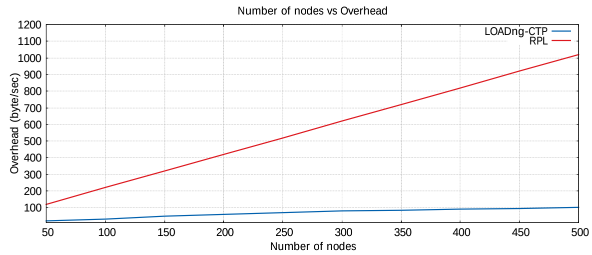


Fig. 6. Overhead bytes per second in the whole network

The overhead packets of LOADng-CTP and RPL grow linearly with RPL sending twice as many packets as LOADng-CTP, and RPL sending 10 times more bytes/s as compared to LOADng-CTP, due to the RPL control packets (mainly, the DIOs) being bigger: a DIO packet 5 takes up to 40 octets in these scenarios, whereas a LOADng-CTP RREQ and RREP packet typically is 10 octets. The overhead of LOADng grows exponentially as the number of nodes increases, up to 700,000 packets for scenarios of 500 nodes (not drawn in the figure).

The peer-to-peer based basic LOADng mechanism is not optimized for sensor-to-root traffic.

VI. CONCLUSION

This paper has offered a detailed protocols comparison of LOADng-CTP to RPL routing protocol on behalf for MP2P and P2MP traffic types. Our results expose the performance of the implementation of LOADng-CTP protocols in Contiki OS. One of its most significant aspects is the considerable reduction of routing overhead due to the smart RREQ used, RREQ flags and the unicast of RREP and its effects in the substantial drop of the messages number and size to maintain routing tables. The implemented LOADng-CTP extension permit on demand collection trees construction supporting upward traffic from sensors to root in bidirectional traffic scenarios. Our study reveals that the LOADng-CTP extension harvests better performance than LOADng: higher data delivery ratios, lower delays and lower overhead and LOADng-CTP is comparable to RPL: same data delivery ratios, same delays and lower overhead for bidirectional data traffic which make it a better solution than RPL for LLNs networks.

In our future works we will concentrate on optimizing upward and downward End-to-End Delay for LOADng-CTP protocol to enhance its performance.

REFERENCES

[1] S. Gusmeroli, S. Haller, M. Harrison, K. Kalaboukas, M. Tomasella, O. Vermesan, H. Vogt, and K. Wouters, Vision and Challenges for Realising the Internet of Things. European Commission, 2010.

[2] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, ―Building Automation Routing Requirements in Low-Power and Lossy Networks,‖ RFC 5867, IETF, June 2010.

[3] A. Brandt, J. Buron, and G. Porcu, ―Home Automation Routing Requirements in LowPower and Lossy Networks,‖ RFC 5826, IETF, April 2010.

[4] Jiazi Yi; Clausen, T.; de Verdiere, A.C., "Efficient Data Acquisition in Sensor Networks: Introducing (the) LOADng Collection Tree Protocol," Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference on, vol., no., pp.1, 4, 21- 23 Sept. 2012.

[5] Tripathi, J.; de Oliveira, J.C., "Proactive versus reactive revisited: IPv6 routing for Low Power Lossy Networks," Information Sciences and Systems (CISS), 2013 47th Annual Conference on, vol., no., pp.1, 6, 20- 22 March 2013.

[6] Jiazi Yi and Thomas Clausen, ―Collection Tree Extension of Reactive Routing Protocol for Low-Power and Lossy Networks,‖ International Journal of Distributed Sensor Networks, vol. 2014, Article ID 352421, 12 pages, 2014.

[7] IETF. Web site. http://www.ietf.org

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

[9] T. Clausen, A Colin, J Yi, A. Niktash, Y Igarashi, H Satoh, and U Herberg. The LLN On-demand Ad hoc Distance-vector Routing Protocol- Next Generation (LOADng), July, 2014. Web site. http://tools.ietf.org/html/draft-clausen-lln-loadng-11

[10] O.Gnawali,R.Fonseca,K.Jamieson,D.Moss,andP.Levis,―The Collection Tree Protocol (CTP),‖ TinyOS, Tech. Rep., 2009, http://sing.stanford.edu/pubs/sing-09-01.pdf.

[11] T. Clausen, A. C. de Verdiere, J. Yi, U. Herberg, and Y. Igarashi,―Experienceswith RPL:IPv6 Routing Protocol for Low ,power and Lossy Networks,‖ The Interne t Engineering Task Force, internet Draft ,work in progress ,draft-clausen-lln-rpl- experiences, February 2013.

[12] T. Clausen, A. Camacho, J. Yi et al., ―Interoperability report for the lightweight on-demand ad hoc distance-vector routing protocol—next generation (loadng),‖ The Internet Engineering Task Force, internet Draft ,work in progress ,draft-lavenu-lln- loadng-interoperability-report, December 2012.