

ROLL Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: 8 January 2026

R. Koutsiamanis, Ed.  
G. Z. Papadopoulos  
N. Montavont  
IMT Atlantique  
P. Thubert  
Cisco  
7 July 2025

Common Ancestor Objective Function and Parent Set DAG Metric Container  
Extension  
draft-ietf-roll-nsa-extension-13

## Abstract

High reliability and low jitter can be achieved by being able to send data packets through multiple paths, via different parents, in a network. This document details how to exchange the necessary information within RPL control packets to let a node better select the different parents that will be used to forward a packet over different paths. This document also describes the Objective Function which takes advantage of this information to implement multi-path routing.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 8 January 2026.

## Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
3. Common Ancestor AP Selection Policies . . . . .	4
3.1. Common Ancestor Strict . . . . .	5
3.2. Common Ancestor Medium . . . . .	6
3.3. Common Ancestor Relaxed . . . . .	6
4. Common Ancestor Objective Function . . . . .	6
4.1. Usage . . . . .	9
5. Node State and Attribute (NSA) object type extension . . . . .	9
5.1. Usage . . . . .	11
6. Controlling PRE . . . . .	12
7. Security Considerations . . . . .	12
8. IANA Considerations . . . . .	12
9. Acknowledgments . . . . .	13
10. References . . . . .	13
10.1. Normative References . . . . .	13
10.2. Informative References . . . . .	13
Appendix A. Implementation Status . . . . .	14
Appendix B. Choosing an AP selection policy . . . . .	17
Authors' Addresses . . . . .	17

## 1. Introduction

Networks in the industrial context must provide stringent guarantees in terms of reliability and predictability, with this domain being one of the main ones addressed by Deterministic Networking [RFC8557]. One of the ways of achieving such guarantees is through Packet Replication and Elimination (PRE) ([RFC9030], Section 4.5.3), a technique which allows redundant paths in the network to be utilized for traffic requiring higher reliability. Another is to have pre-selected backup paths on standby for quick packet retransmission when packet failures occur. Load-balancing can be also used to make sure that not all traffic passes through the same nodes, to more evenly spread the packet forwarding load. Allowing industrial applications to function over wireless networks requires the application of the principles and architecture of Deterministic Networking [RFC8655]. This results in designs that aim at optimizing packet delivery rate

and bounding latency. Additionally, nodes operating on battery need to minimize their energy consumption.

As an example, to meet this goal, IEEE Std. 802.15.4 [IEEE802154] provides Time-Slotted Channel Hopping (TSCH), a mode of operation that uses a common communication schedule based on timeslots to allow deterministic medium access as well as channel hopping to work around radio interference. However, since TSCH uses retransmissions in the event of a failed transmission, end-to-end latency and jitter performance can deteriorate.

Furthermore, the 6TiSCH working group, focusing on IPv6 over IEEE Std. 802.15.4-TSCH, has worked on these issues and produced the "6TiSCH Architecture" [RFC9030] to address that case.

Building a multi-path DODAG can be achieved based on the RPL capability of having multiple parents for each node in a network, a subset of which is used to forward packets. In order to select parents to be part of this subset, the RPL Objective Function (OF) needs additional information. This document describes an OF which implements multi-path routing and specifies the transmission of this specific path information.

This document describes a new Objective Function (OF) called the Common Ancestor (CA) OF (see Section 4). A detailed description is given of how the path information is used within the CA OF and how the subset of parents for forwarding packets is selected. This specification defines a new Objective Code Point (OCP) for the CA OF.

For the path information, this specification focuses on the extensions to the DAG Metric Container [RFC6551] required for supplying to the CA OF a part of the information it needs to operate. This information is the RPL [RFC6550] parent address set of a node and it must be sent to potential children of the node. The RPL DIO Control Message is the canonical way of broadcasting this kind of information and therefore its DAG Metric Container [RFC6551] field is used to append a Node State and Attribute (NSA) object. The node's parent address set is stored as an optional TLV within the NSA object. This specification defines the type value and structure for the parent address set TLV (see Section 5).

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The draft uses the following Terminology from other RFCs:

Parent Set (PS): Defined in RPL [RFC6550].

Packet Replication and Elimination (PRE): A method that consists of transmitting multiple copies of a packet using multi-path forwarding over a multi-hop network and that consolidates multiple received packet copies to control flooding. See "Complex Track with Replication and Elimination" in [RFC9030], Section 4.5.3 for more details.

The draft introduces the following Terminology:

Alternative Parent (AP): An RPL parent in the parent set of a node is used to forward a packet copy when replicating packets.

Alternative Parent (AP) Selection: The mechanism for choosing the next hop node to forward a packet copy when replicating packets.

Preferred Grand Parent (PGP): The preferred parent of the preferred parent of a node.

### 3. Common Ancestor AP Selection Policies

In the RPL protocol, each node maintains a list of potential parents. When more than one parent is required, as when performing PRE, the RPL DODAG Preferred Parent node is used, as per RPL [RFC6550] parent selection, effectively depending on the OF used. If the CA OF is used, the way this choice is made is described in Section 4. Furthermore, to construct an alternative path toward the root, in addition to the PP node, each node in the network selects one or more parents, called Alternative Parents (APs), from its Parent Set (PS).

There are multiple possible policies for selecting the AP node. This section details three such possible policies.

All three policies defined perform AP selection based on common ancestors, named Common Ancestor Strict, Common Ancestor Medium, and Common Ancestor Relaxed, depending on how restrictive the selection process is. A more restrictive policy will limit flooding but might fail to select an appropriate AP, while a less restrictive one will more often find an appropriate AP but might increase flooding.

All three policies apply their corresponding common ancestor criterion to filter the list of candidate neighbors in the Alternative Parent set.

If after the filtering there are multiple condition-meeting candidate nodes, the node MUST select at least one of them as its AP node. The way this choice is made depends on which OF is used. If the CA OF is used, the way this choice is made is described in Section 4.

### 3.1. Common Ancestor Strict

In the CA Strict OF the node will check if its Preferred Grand Parent (PGP), the PP of its PP, is the same as the PP of the potential AP.

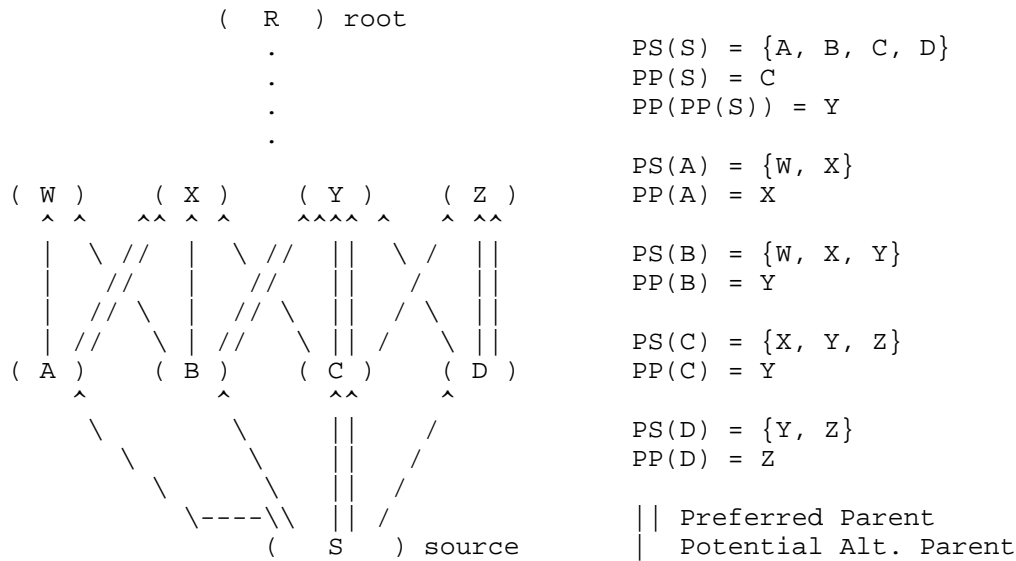


Figure 1: Example Common Ancestor Strict Alternative Parent Selection policy

For example, in Figure 1, the source node S must know its grandparent sets through nodes A, B, C, and D. The Parent Sets (PS) and the Preferred Parents (PP) of nodes A, B, C, and D are shown on the side of the figure. The CA Strict parent selection policy will select an AP for node S for which  $PP(PP(S)) = PP(AP)$ . Given that  $PP(PP(S)) = Y$ :

- \* Node A:  $PP(A) = X$  and therefore it is different than  $PP(PP(S))$
- \* Node B:  $PP(B) = Y$  and therefore it is equal to  $PP(PP(S))$
- \* Node D:  $PP(D) = Z$  and therefore it is different than  $PP(PP(S))$

Therefore, node S MUST select node B as its AP node, since  $PP(PP(S)) = Y = PP(B)$ .

### 3.2. Common Ancestor Medium

In the CA Medium OF the node will check if its Preferred Grand Parent (PGP), the PP of its PP, is contained in the PS of the potential AP.

Using the same example, in Figure 1, the CA Medium parent selection policy will select an AP for node S for which  $PP(PP(S))$  is in  $PS(AP)$ . Given that  $PP(PP(S)) = Y$ :

- \* Node A:  $PS(A) = \{W, X\}$  and therefore  $PP(PP(S))$  is not in the set

- \* Node B:  $PS(B) = \{W, X, Y\}$  and therefore  $PP(PP(S))$  is in the set

- \* Node D:  $PS(D) = \{Y, Z\}$  and therefore  $PP(PP(S))$  is in the set

Therefore, S MUST select at least one node among B and D as its AP node.

### 3.3. Common Ancestor Relaxed

In the CA Relaxed OF the node will check if the Parent Set (PS) of its Preferred Parent (PP) has a node in common with the PS of the potential AP.

Using the same example, in Figure 1, the CA Relaxed parent selection policy will select an AP for node S for which  $PS(PP(S))$  has at least one node in common with  $PS(AP)$ . Given that  $PS(PP(S)) = \{X, Y, Z\}$ :

- \* Node A:  $PS(A) = \{W, X\}$  and the common nodes are  $\{X\}$

- \* Node B:  $PS(B) = \{W, X, Y\}$  and the common nodes are  $\{X, Y\}$

- \* Node D:  $PS(D) = \{Y, Z\}$  and the common nodes are  $\{Y, Z\}$

Therefore, S MUST select at least one node among A, B, and D as its AP node.

## 4. Common Ancestor Objective Function

An OF which allows the multiple paths to remain correlated is detailed here. More specifically, when using this OF a node will select an AP node "close" to its PP node to allow the operation of overhearing between parents. Closeness here is not strictly defined, however, the premise is that those candidate parent nodes that have common parents themselves have a higher probability of being within each other's radio range, though it's of course not guaranteed. For more details about overhearing and its use in this context see the "Complex Track with Replication and Elimination" in [RFC9030],

Section 4.5.3. If multiple potential APs match this condition, one of the APs with the lowest rank will be registered, with the choice between multiple nodes with the same lowest rank being implementation-specific.

The OF described here is an extension of The Minimum Rank with Hysteresis Objective Function (MRHOF) [RFC6719]. The CA OF does not update [RFC6719]. Rather, it uses the existing definition of MRHOF in [RFC6719] to build a new OF (with a new Objective Code Point (OCP)) which provides additional functionality, while maintaining compatibility by retaining the existing functionality of MRHOF for the preferred parent. To be precise, this OF extends MRHOF by specifying how an AP is selected while the selection and switching of the PP remain unaltered. Importantly, the calculation of the rank of the node through each candidate neighbor and the selection of the PP is kept the same as in MRHOF.

How the CA OF differs from MRHOF in a section-by-section manner follows in detail:

[RFC6719], Section 2: "Terminology". Term "Selected Metric":

The CA OF uses only one metric, like MRHOF, for rank calculation, with the same MRHOF semantics. For selecting the AP, the PS TLV (stored in the DIO Metric Container Node State and Attribute (NSA) object body, see Section 5) is used. This additional NSA metric is disregarded for rank calculation.

[RFC6719], Section 3 "The Minimum Rank with Hysteresis Objective Function":

Same as MRHOF extended to AP selection. Minimum Rank path selection and switching apply correspondingly to the AP with the extra CA requirement of having some match between ancestors, according to one of the Common Ancestor AP selection policies defined in Section 3.

[RFC6719], Section 3.1 "Computing the Path Cost":

Same as MRHOF extended to AP selection. If a candidate neighbor does not fulfill the CA requirement then the path cost through that neighbor MUST be set to MAX\_PATH\_COST, the same value used by MRHOF. As a result, the node MUST NOT select the candidate neighbor as its AP.

[RFC6719], Section 3.2 "Parent Selection":

Same as MRHOF extended to AP selection. To allow hysteresis, AP selection maintains a variable, `cur_ap_min_path_cost`, which is the path cost of the current AP.

[RFC6719], Section 3.2.1 "When Parent Selection Runs":

Same as MRHOF.

[RFC6719], Section 3.2.2 "Parent Selection Algorithm":

Same as MRHOF extended to AP selection. If the smallest path cost for paths through the candidate neighbors is smaller than `cur_ap_min_path_cost` by less than `PARENT_SWITCH_THRESHOLD` (the same variable as MRHOF uses), the node MAY continue to use the current AP. Additionally, if there is no PP selected, there MUST NOT be any AP selected either. Finally, as with MRHOF, a node MAY include up to `PARENT_SET_SIZE-1` additional candidate neighbors in its Alternative Parent set. The value of `PARENT_SET_SIZE` is the same as in MRHOF.

[RFC6719], Section 3.3 "Computing Rank":

Same as MRHOF.

[RFC6719], Section 3.4 "Advertising the Path Cost":

Same as MRHOF.

[RFC6719], Section 3.5 "Working without Metric Containers":

The CA OF can work without metric containers identically to MRHOF. Nodes that transmit DIO messages without the Metric Container will never be selected as an AP by the CA OF of another node but can be selected as the PP as per the operation of MRHOF. Effectively, the lack of Metric Containers is equivalent to operating with a Parent Set TLV where there are no PS IPv6 addresses and the PS Length is 0.

[RFC6719], Section 4 "Using MRHOF for Metric Maximization":

Same as MRHOF.

[RFC6719], Section 5 "MRHOF Variables and Parameters":

Same as MRHOF extended to AP selection. The CA OF operates like MRHOF for AP selection by maintaining separate:

AP: Corresponding to the MRHOF PP. Hysteresis is configured for AP with the same `PARENT_SWITCH_THRESHOLD` parameter as in MRHOF. The AP MUST NOT be the same as the PP.

Alternative parent set: Corresponding to the MRHOF parent set. The size is defined by the same `PARENT_SET_SIZE` parameter as in MRHOF. The Alternative parent set MUST be a strict subset of the parent set.

`cur_ap_min_path_cost`: Corresponding to the MRHOF `cur_min_path_cost` variable. To support the operation of the hysteresis function for AP selection.



[RFC6719], Section 6 "Manageability":  
Same as MRHOF.

[RFC6719], Section 6.1 "Device Configuration":  
Same as MRHOF.

[RFC6719], Section 6.2 "Device Monitoring":  
Same as MRHOF.

#### 4.1. Usage

All the Common Ancestor AP Selection Policies (Section 3) apply their corresponding criterion to filter the list of candidate neighbors in the Alternative Parent set. The AP is then selected from the Alternative Parent set based on Rank and using hysteresis as is done for the PP in MRHOF. It is noteworthy that the OF uses the same Objective Code Point (OCP): (TBD1) for all policies used.

The PS information can be used by any of the described AP selection policies or other ones not described here, depending on requirements. It is optional for all nodes to use the same AP selection policies. Different nodes may use different AP selection policies since the selection policy is local to each node. For example, using different policies can be used to vary the transmission reliability in each hop. Some suggestions are provided in Appendix B.

#### 5. Node State and Attribute (NSA) object type extension

In order to select their AP node, nodes need to be aware of their grandparent node sets. Within RPL [RFC6550], the nodes use the DODAG Information Object (DIO) Control Message to broadcast information about themselves to potential children. However, RPL [RFC6550], does not define how to propagate information related to the parent set, which is what this document addresses.

DIO messages can carry multiple options, out of which the DAG Metric Container option [RFC6551] is the most suitable structurally and semantically to carry the parent set. The DAG Metric Container option itself can carry different nested objects, out of which the Node State and Attribute (NSA) [RFC6551] is appropriate for transferring generic node state data. Within the Node State and Attribute, it is possible to store optional TLVs representing various node characteristics. As per the Node State and Attribute (NSA) [RFC6551] description, no TLV has been defined for use. This document defines one TLV for transmitting a node's parent set.

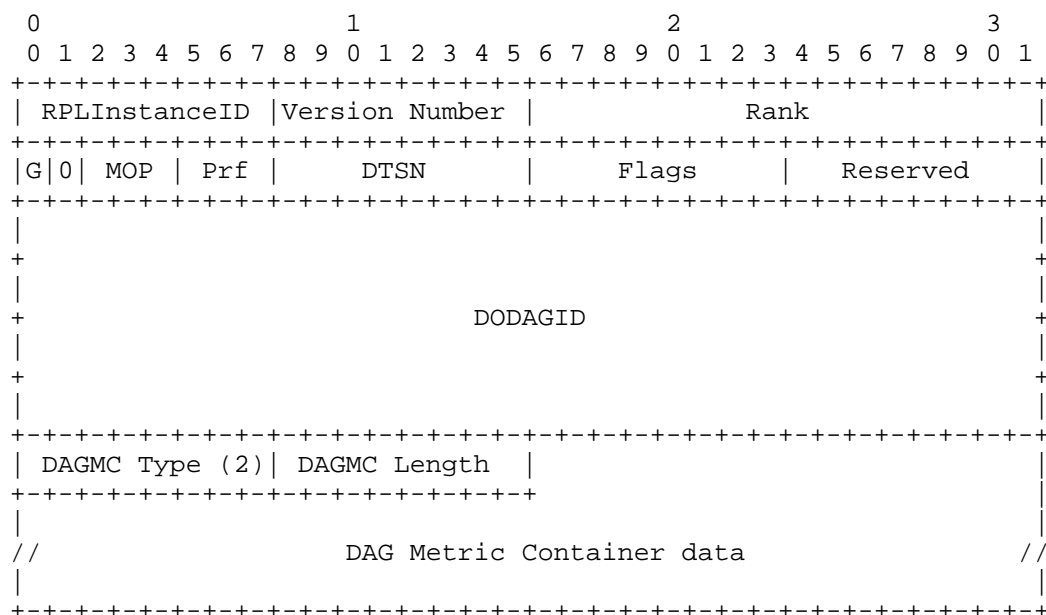


Figure 2: Example DIO Message with a DAG Metric Container option

Figure 2 shows the structure of the DIO Control Message when a DAG Metric Container option is included. The DAG Metric Container option type (DAGMC Type in Figure 2) has the value 0x02 as per the IANA registry for the RPL Control Message Options, and is defined in [RFC6550]. The DAG Metric Container option length (DAGMC Length in Figure 2) expresses the DAG Metric Container length in bytes. DAG Metric Container data holds the actual data and is shown expanded in Figure 3.

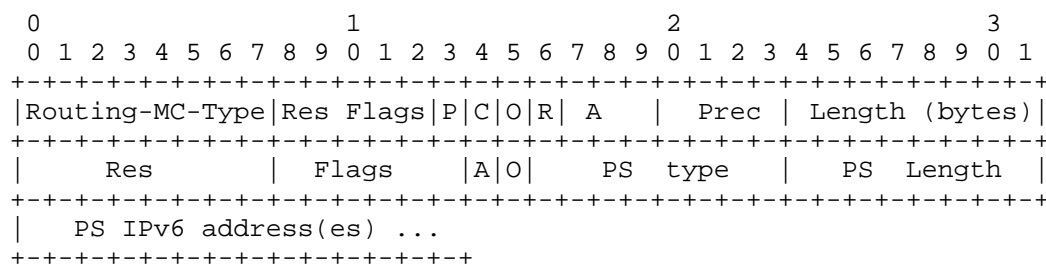


Figure 3: DAG Metric Container (MC) data with Node State and Attribute (NSA) object body and a TLV

The structure of the DAG Metric Container data in the form of a Node State and Attribute (NSA) object with a TLV in the NSA Optional TLVs field is shown in Figure 3. The first 32 bits comprise the DAG Metric Container header and all the following bits are part of the Node State and Attribute object body, as defined in [RFC6551]. This document defines a new TLV, which MUST be carried in the Node State and Attribute (NSA) object Optional TLVs field within the context of the use of the CA OF. The TLV is named Parent Set and is abbreviated as PS in Figure 3.

PS type: The type of the Parent Set TLV. The value is (TBD2).

PS Length: The total length of the TLV value field (PS IPv6 address(es)) in bytes (0 included). The length is an integral multiple of 16, the number of bytes in an IPv6 address.

PS IPv6 address(es): One or more 128-bit IPv6 addresses, without any separator between them. The field consists of one IPv6 address per parent in the parent set. The parent addresses are listed in decreasing order of preference and not all parents in the parent set need to be included. The selection of how many parents from the parent set will be included is left to the implementation. The number of parent addresses in the PS IPv6 address(es) field can be deduced by dividing the length of the PS IPv6 address(es) field in bytes by 16, the number of bytes in an IPv6 address.

### 5.1. Usage

The PS is used in the process of parent selection, and especially in AP selection since it can help the alternative path to not significantly deviate from the preferred path. The Parent Set is information local to the node that broadcasts it.

The PS is used only within NSA objects configured as a metric, therefore the DAG Metric Container field "C" MUST be 0. Additionally, since the information in the PS needs to be propagated downstream but cannot be aggregated, the DAG Metric Container field "R" MUST be 1. Finally, since the information contained is by definition partial, specifically just the parent set of the DIO-sending node, the DAG Metric Container field "P" MUST be 1.

The presence of incorrectly configured flags MUST render the Parent Set TLV invalid. This case MUST be handled equivalently to operating with a Parent Set TLV where there are no PS IPv6 addresses and the PS Length is 0.

The presence of a PS Length value that is not a multiple of 16 or larger than 240 MUST render the Parent Set TLV invalid. This case MUST be handled equivalently to operating with a Parent Set TLV where there are no PS IPv6 addresses and the PS Length is 0.

## 6. Controlling PRE

PRE is very helpful when the aim is to increase reliability for a certain path, however, its use creates additional traffic as part of the replication process. It is conceivable that not all paths have stringent reliability requirements. Therefore, a way to control whether PRE is applied to a path's packets SHOULD be implemented. For example, a traffic class label can be used to determine this behavior per flow type as described in Deterministic Networking Architecture [RFC8655].

## 7. Security Considerations

All the security considerations from [RFC6550], [RFC6551], and [RFC6719] apply.

In this document, the structure of the DIO control message is extended, within the pre-defined DIO options. The additional information is the list of IPv6 addresses of the parent set of the node transmitting the DIO. This use of this additional information can have the following additional potential consequences:

- \* A malicious node that can send DIOs can use the parent set extension to convince neighbors to route through itself, instead of the normal preferred parent they would use. However, this is already possible with other OFs (like OF0 [RFC6552] and MRHOF [RFC6719]) by reporting a fake rank value in the DIO, thus masquerading as the DODAG root.

## 8. IANA Considerations

This document requests the allocation of a new value (TBD1) from the "Objective Code Point (OCP)" registry in the "Routing Protocol for Low Power and Lossy Networks (RPL)" registry group. The Description field should have the value "Common Ancestor Objective Function (CAOF)".

This document also requests the allocation of a new value (TBD2) for the "Parent Set" TLV from the "Routing Metric/Constraint TLVs" registry in the "Routing Protocol for Low Power and Lossy Networks (RPL) Routing Metric/Constraint" registry group. The Description field should have the value "Parent Set".

## 9. Acknowledgments

We are very grateful to Dominique Barthel, Rahul Jadhav, Fabrice Theoleyre, Diego Dujovne, Derek Jianqiang Hou, Michael Richardson, and Alvaro Retana for their comments, feedback, and support which lead to many improvements to this document. We would also like to thank Tomas Lagos Jenschke very much for helping in the implementation and evaluation of this document.

## 10. References

### 10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.
- [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, DOI 10.17487/RFC6550, March 2012, <<https://www.rfc-editor.org/rfc/rfc6550>>.
- [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, DOI 10.17487/RFC6551, March 2012, <<https://www.rfc-editor.org/rfc/rfc6551>>.
- [RFC6719] Gnawali, O. and P. Levis, "The Minimum Rank with Hysteresis Objective Function", RFC 6719, DOI 10.17487/RFC6719, September 2012, <<https://www.rfc-editor.org/rfc/rfc6719>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.

### 10.2. Informative References

## [IEEE802154]

IEEE standard for Information Technology, "IEEE Std. 802.15.4, Part. 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks", n.d.,  
<<http://standards.ieee.org/findstds/standard/802.15.4-2015.html>>.

[RFC6552] Thubert, P., Ed., "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)", RFC 6552, DOI 10.17487/RFC6552, March 2012,  
<<https://www.rfc-editor.org/rfc/rfc6552>>.

[RFC8557] Finn, N. and P. Thubert, "Deterministic Networking Problem Statement", RFC 8557, DOI 10.17487/RFC8557, May 2019,  
<<https://www.rfc-editor.org/rfc/rfc8557>>.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019,  
<<https://www.rfc-editor.org/rfc/rfc8655>>.

[RFC9030] Thubert, P., Ed., "An Architecture for IPv6 over the Time-Slotted Channel Hopping Mode of IEEE 802.15.4 (6TiSCH)", RFC 9030, DOI 10.17487/RFC9030, May 2021,  
<<https://www.rfc-editor.org/rfc/rfc9030>>.

## Appendix A. Implementation Status

A research-stage implementation of the PRE mechanism using the proposed extension as part of a 6TiSCH IOT use case was developed at IMT Atlantique, France by Tomas Lagos Jenschke and Remous-Aris Koutsiamanis. It was implemented on the open-source Contiki OS and tested with the Cooja simulator. The DIO DAGMC NSA extension is implemented with a configurable number of parents from the parent set of a node to be reported.

( R )

(11) (12) (13) (14) (15) (16)

(21) (22) (23) (24) (25) (26)

(31) (32) (33) (34) (35) (36)

(41) (42) (43) (44) (45) (46)

(51) (52) (53) (54) (55) (56)

( S )

Figure 4: Simulation Topology

The simulation setup is:

Topology: 32 nodes structured in a regular grid as shown in Figure 4. Node S (source) is the only data packet sender and sends data to node R (root). The parent set of each node (except R) is all the nodes in the immediately higher row, the immediately above 6 nodes. For example, each node in {51, 52, 53, 54, 55, 56} is connected to all of {41, 42, 43, 44, 45, 46}. Nodes 11, 12, 13, 14, 15, and 16 have a single upwards link to R.

MAC: TSCH with 1 retransmission

Platform: Cooja

Schedule: Static, 2 timeslots per link from each node to each parent in its parent set, 1 broadcast EB slot, 1 sender-based shared timeslot (for DIO and DIS) per node (total of 32).

Simulation lifecycle: Allow link formation for 100 seconds before starting to send data packets. Afterward, S sends data packets to R. The simulation terminates when 1000 packets have been sent by S.

Radio Links: Every 60 s, a new Packet Delivery Rate is randomly drawn for each link, with a uniform distribution spanning the 70% to 100% interval.

Traffic Pattern: CBR, S sends one non-fragmented UDP packet every 5 seconds to R.

PS extension size: 3 parents.

Routing Methods:

- \* RPL: The default RPL non-PRE implementation in Contiki OS.
- \* 2nd ETX: PRE with a parent selection method which picks as AP the 2nd best parent in the parent set based on ETX.
- \* CA Strict: As described in Section 3.1.
- \* CA Medium: As described in Section 3.2.

Simulation results:

Routing Method	Average Packet Delivery Rate (%)	Average Traversed Nodes/packet (#)	Average Duplications/packet (#)
RPL	82.70	5.56	7.02
2nd ETX	99.38	14.43	31.29
CA Strict	97.32	9.86	18.23
CA Medium	99.66	13.75	28.86

Table 1: Simulation results

Links:

- \* Contiki OS DIO DAGMC NSA extension (draft-koutsiamanis-roll-nsa-extension branch) (<https://github.com/ariskou/contiki/tree/draft-koutsiamanis-roll-nsa-extension>)
- \* Wireshark dissectors (for the optional PS TLV) - currently merged / in master (<https://code.wireshark.org/review/gitweb?p=wireshark.git;a=commit;h=e2f6ba229f45d8ccae2a6405e0ef41f1e61da138>)



## Appendix B. Choosing an AP selection policy

The manner of choosing an AP selection policy is left to the implementation, for maximum flexibility.

For example, a different policy can be used per traffic type. The network configurator can choose the CA Relaxed policy to increase reliability (thus producing some flooding) for specific, extremely important, alert packets. On the other hand, all normal data traffic uses the CA Strict policy. Therefore, an exception is made just for the alert packets.

Another option would be to devise a new disjoint policy, where the paths are on purpose non-correlated, to increase path diversity and resilience against whole groups of nodes failing. The disadvantage may be increased jitter.

Finally, a network configurator may provide the CA policies with a preference order of Strict > Medium > Relaxed as a means of falling back to more flood-prone policies to maintain reliability.

## Authors' Addresses

Remous-Aris Koutsiamanis (editor)  
IMT Atlantique  
Office B220  
4 rue Alfred Kastler, CS 20722  
44307 Nantes Cedex 3  
France  
Email: aris@ariskou.com

Georgios Papadopoulos  
IMT Atlantique  
Office B00 - 114A  
2 Rue de la Chataigneraie  
35510 Cesson-Sevigne - Rennes  
France  
Phone: +33 299 12 70 04  
Email: georgios.papadopoulos@imt-atlantique.fr

Nicolas Montavont  
IMT Atlantique  
Office B00 - 106A  
2 Rue de la Chataigneraie  
35510 Cesson-Sevigne - Rennes  
France

Phone: +33 299 12 70 23

Email: nicolas.montavont@imt-atlantique.fr

Pascal Thubert

Cisco Systems

Building D

45 Allee des Ormes - BP1200

3244 MOUGINS - Sophia Antipolis

France

Phone: +33 497 23 26 34

Email: pthubert@cisco.com