              The Minimum Rank with Hysteresis Objective Function

Abstract

   The Routing Protocol for Low-Power and Lossy Networks (RPL)
   constructs routes by using Objective Functions that optimize or
   constrain the routes it selects and uses. This specification
   describes the Minimum Rank with Hysteresis Objective Function
   (MRHOF), an Objective Function that selects routes that minimize a
   metric, while using hysteresis to reduce churn in response to small
   metric changes. MRHOF works with additive metrics along a route, and
   the metrics it uses are determined by the metrics that the RPL
   Destination Information Object (DIO) messages advertise.

Table of Contents

1. Introduction

   An Objective Function specifies how RPL [RFC6550] selects paths. For
   example, if an RPL instance uses an Objective Function that minimizes
   hop count, RPL will select paths with a minimum hop count. RPL
   requires that all nodes in a network use a common Objective Function;
   relaxing this requirement may be a subject of future study.

   The nodes running RPL might use a number of metrics to describe a
   link or a node [RFC6551] and make these metrics available for route
   selection. RPL advertises metrics in RPL Destination Information
   Object (DIO) messages with a Metric Container suboption. An
   Objective Function can use these metrics to choose routes.

   To decouple the details of an individual metric or Objective Function
   from forwarding and routing, RPL describes routes through a value
   called Rank. Rank, roughly speaking, corresponds to the distance
   associated with a route. RPL defines how nodes decide on paths based

on Rank and advertise their Rank. An Objective Function defines how
nodes calculate Rank, based on the Rank of its potential parents,
metrics, and other network properties.

This specification describes the Minimum Rank with Hysteresis
Objective Function (MRHOF), an Objective Function for RPL, which uses
hysteresis while selecting the path with the smallest metric value.
The metric that MRHOF uses is determined by the metrics in the DIO
Metric Container. For example, the use of MRHOF with the latency
metric allows RPL to find stable minimum-latency paths from the nodes
to a root in the Directed Acyclic Graph (DAG) instance [RFC6550].
The use of MRHOF with the Expected Transmission Count (ETX) metric
[RFC6551] allows RPL to find the stable minimum-ETX paths from the
nodes to a root in the DAG instance. In the absence of a metric in
the DIO Metric Container or of a DIO Metric Container, MRHOF defaults
to using ETX to compute Rank, as described in Section 3.5.

Because MRHOF seeks to minimize path costs as described by metrics,
it can only be used with additive metrics. MRHOF does not support
metrics that are not additive.

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 RFC
2119 [RFC2119].

The terminologies used in this document are consistent with the
terminologies described in [ROLL-TERM] and [RFC6551].

The terminologies used in this document are also consistent with the
terminologies described in [RFC6550], except the term Rank. In this
document, Rank refers to the value of the Rank field, not DAGRank as
in [RFC6550].

This document introduces three terms:

Selected metric: The metric chosen for path selection by the network
    operator. MRHOF supports using a single metric for path
    selection. The decision to use a metric (other than ETX) as the
    selected metric is indicated by the presence of the chosen metric
    in the DIO Metric Container. The selection of the ETX metric is
    indicated by the absence of the Metric Container, in which case
    ETX is advertised as Rank.

Path cost: Path cost quantifies a property of an end-to-end path.
   Path cost is obtained by each node summing up the selected link
   metric to the path cost advertised by the parent. Path cost can
   be used by RPL to compare different paths.

Worst parent: The node in the parent set with the largest path cost.

3. The Minimum Rank with Hysteresis Objective Function

The Minimum Rank with Hysteresis Objective Function, MRHOF, is
designed to find the paths with the smallest path cost while
preventing excessive churn in the network. It does so by using two
mechanisms. First, it finds the minimum cost path, i.e., path with
the minimum Rank. Second, it switches to that minimum Rank path only
if it is shorter (in terms of path cost) than the current path by at
least a given threshold. This second mechanism is called
"hysteresis".

MRHOF may be used with any additive metric listed in [RFC6551] as
long as the routing objective is to minimize the given routing
metric. Nodes MUST support at least one of these metrics: hop count,
latency, or ETX. Nodes SHOULD support the ETX metric. MRHOF does
not support non-additive metrics.

3.1. Computing the Path Cost

Root nodes (Grounded or Floating) set the variable cur_min_path_cost
to the metric value that computes to a Rank of MinHopRankIncrease.

If a non-root node does not have metrics to compute the path cost
through any of the candidate neighbors, it MUST join one of the
candidate neighbors as a RPL Leaf.

Otherwise, nodes compute the path cost for each candidate neighbor
reachable on an interface. The path cost of a neighbor represents
the cost of the path, in terms of the selected metric, from a node to
the root of the Destination-Oriented DAG (DODAG) through that
neighbor. A non-root node computes a neighbor's path cost by adding
two components:

1. If the selected metric is a link metric, the selected metric for
    the link to the candidate neighbor. If the selected metric is a
    node metric, the selected metric for the node.

   2. The value of the selected metric in the Metric Container in the
       DIO sent by that neighbor. In case the Metric Container is
       empty, ETX is the selected metric -- use the Rank advertised by
       that neighbor as the second component. See Section 3.5 for
       details on how an ETX metric is used in MRHOF.

   A node SHOULD compute the path cost for the path through each
   candidate neighbor reachable through an interface. If a node cannot
   compute the path cost for the path through a candidate neighbor, the
   node MUST NOT select the candidate neighbor as its preferred parent.
   However, if the node cannot compute the path cost through any
   neighbor, it may join the candidate neighbor as a Leaf, as described
   above.

   If the selected metric is a link metric and the metric of the link to
   a neighbor is not available, the path cost for the path through that
   neighbor SHOULD be set to MAX_PATH_COST. This cost value will
   prevent this path from being considered for path selection.

   If the selected metric is a node metric, and the metric is not
   available, the path cost through all the neighbors SHOULD be set to
   MAX_PATH_COST.

   The path cost corresponding to a neighbor SHOULD be recomputed each
   time any of the following conditions are met:

   1. The selected metric of the link to the candidate neighbor is
       updated.

   2. The selected metric is a node metric and the metric is updated.

   3. A node receives a new metric advertisement from the candidate
       neighbor.

   This computation SHOULD also be performed periodically. While it is
   harmless to delay this computation up to a minimum Trickle interval
   [RFC6550], longer delays in updating the path cost after the metric
   is updated or a new metric advertisement is received can lead to
   stale information.

3.2. Parent Selection

   After computing the path cost for all the candidate neighbors
   reachable through an interface for the current DODAG iteration
   [RFC6550], a node selects the preferred parent. This process is
   called "parent selection". To allow hysteresis, parent selection
   maintains a variable, cur_min_path_cost, which is the path cost of
   the current preferred parent.

3.2.1. When Parent Selection Runs

    A MRHOF implementation SHOULD perform Parent Selection each time:

    1. The path cost for an existing candidate neighbor, including the
        preferred parent, changes. This condition can be checked
        immediately after the path cost is computed.

    2. A new candidate neighbor is inserted into the neighbor table.

    If, despite the above, it is necessary to defer the parent selection
    until a later time (e.g., up to the Trickle minimum interval
    [RFC6550]), note that doing so can delay the use of better paths
    available in the network.

3.2.2. Parent Selection Algorithm

    If the selected metric for a link is greater than MAX_LINK_METRIC,
    the node SHOULD exclude that link from consideration during parent
    selection.

    A node MUST select the candidate neighbor with the lowest path cost
    as its preferred parent, except as indicated below:

    1. A node MAY declare itself as a Floating root, and hence have no
        preferred parent, depending on system configuration.

    2. If cur_min_path_cost is greater than MAX_PATH_COST, the node MAY
        declare itself as a Floating root.

    3. If the smallest path cost for paths through the candidate
        neighbors is smaller than cur_min_path_cost by less than
        PARENT_SWITCH_THRESHOLD, the node MAY continue to use the current
        preferred parent. This is the hysteresis component of MRHOF.

    4. If ALLOW_FLOATING_ROOT is 0 and no neighbors are discovered, the
        node does not have a preferred parent and MUST set
        cur_min_path_cost to MAX_PATH_COST.

    If there are multiple neighbors that share the smallest path cost, a
    node MAY use a different selection criteria to select which of these
    neighbors should be considered to have the lowest cost.

    A node MAY include up to PARENT_SET_SIZE-1 additional candidate
    neighbors in its parent set. The cost of the path through the nodes
    in the parent set is smaller than or equal to the cost of the paths
    through any of the nodes that are not in the parent set. If the cost

of the path through the preferred parent and the worst parent is too
large, a node MAY keep a smaller parent set than PARENT_SET_SIZE.

Once the preferred parent is selected, the node sets its
cur_min_path_cost variable to the path cost corresponding to the
preferred parent. The value of the cur_min_path_cost is carried in
the Metric Container corresponding to the selected metric when DIO
messages are sent.

3.3. Computing Rank

DAG roots set their Rank to MinHopRankIncrease.

Once a non-root node selects its parent set, it can use the following
table to covert the path cost of a parent (written as Cost in the
table) to a Rank value:

```
            +------------------+------------+
            | Node/link Metric |    Rank    |
            +------------------+------------+
            |     Hop-Count    |    Cost    |
            |     Latency      | Cost/65536 |
            |       ETX        |    Cost    |
            +------------------+------------+
```

Table 1: Conversion of Metric to Rank

If MRHOF is used with other metrics, the Rank is undefined. If the
Rank is undefined, the node must join one of the neighbors as a RPL
Leaf node according to [RFC6550].

MRHOF uses this Rank value to compute the Rank it associates with the
path through each member of the parent set. The Rank associated with
a path through a member of the parent set is the maximum of two
values. The first is the corresponding Rank value calculated with
the table above, the second is that nodes' advertised Rank plus
MinHopRankIncrease.

A node sets its Rank to the maximum of three values:

1. The Rank calculated for the path through the preferred parent.

2. The Rank of the member of the parent set with the highest
    advertised Rank, rounded to the next higher integral Rank, i.e.,
    to MinHopRankIncrease * (1 + floor(Rank/MinHopRankIncrease)).

3. The largest calculated Rank among paths through the parent set,
    minus MaxRankIncrease.

The first case is the Rank associated with the path through the
preferred parent. The second case covers requirement 5 of Rank
advertisements in Section 8.2.1 of [RFC6550]. The third case ensures
that a node does not advertise a Rank, which then precludes it from
using members of its parent set.

Note that the third case means that a node advertises a conservative
Rank value based on members of its parent set. This conservative
value might be significantly higher than the Rank calculated for the
path through the preferred parent. Accordingly, picking a parent set
whose paths have a large range of Ranks will likely result in
suboptimal routing: nodes might not choose good paths because they are
advertised as much worse than they actually are. The exact selection
of a parent set is an implementation decision.

3.4. Advertising the Path Cost

Once the preferred parent is selected, the node sets its
cur_min_path_cost variable to the path cost corresponding to its
preferred parent. It then calculates the metric it will advertise in
its metric container. This value is the path cost of the member of
the parent set with the highest path cost. Thus, while
cur_min_path_cost is the cost through the preferred parent, a node
advertises the highest cost path from the node to the root through a
member of the parent set. The value of the highest cost path is
carried in the metric container corresponding to the selected metric
when DIO messages are sent.

If ETX is the selected metric, a node MUST NOT advertise it in a
metric container. Instead, a node MUST advertise an approximation of
its ETX in its advertised Rank value, following the rules described
in Section 3.3. If a node receives a DIO with a Metric Container
holding an ETX metric, MRHOF MUST ignore the ETX metric value in its
Rank calculations.

DODAG Roots advertise a metric value that computes to a Rank value of
MinHopRankIncrease.

3.5. Working without Metric Containers

In the absence of a Metric Container, MRHOF uses ETX as its metric.
It locally computes the ETX of links to its neighbors and adds this
value to their advertised Rank to compute the associated Rank of
routes. Once parent selection and rank computation is performed
using the ETX metric, the node advertises the Rank and MUST NOT
include a metric container in its DIO messages. While assigning Rank
in this case, use the representation of ETX described in [RFC6551],
i.e., assign Rank equal to ETX * 128.

4. Using MRHOF for Metric Maximization

   MRHOF cannot be directly used for parent selection using metrics that
   require finding paths with a maximum value of the selected metric,
   such as path reliability. It is possible to convert such a metric
   maximization problem to a metric minimization problem for some
   metrics and use MRHOF provided:

      There is a fixed and well-known maximum metric value corresponding
      to the best path. This is the path cost for the DAG root. For
      example, the logarithm of the best link reliability has a value of
      0.

      The metrics in the maximization problem are all negative. The
      logarithm of the link reliability is always negative.

   For metrics meeting the above conditions, the problem of maximizing
   the metric value is equivalent to minimizing the modified metric
   value, e.g., logarithm of link reliability. MRHOF is not required to
   work with these metrics.

5. MRHOF Variables and Parameters

   MRHOF uses the following variable:

      cur_min_path_cost: The cost of the path from a node through its
      preferred parent to the root computed at the last parent
      selection.

   MRHOF uses the following parameters:

      MAX_LINK_METRIC: Maximum allowed value for the selected link
      metric for each link on the path.

      MAX_PATH_COST: Maximum allowed value for the path metric of a
      selected path.

      PARENT_SWITCH_THRESHOLD: The difference between the cost of the
      path through the preferred parent and the minimum cost path in
      order to trigger the selection of a new preferred parent.

      PARENT_SET_SIZE: The number of candidate parents, including the
      preferred parent, in the parent set.

      ALLOW_FLOATING_ROOT: If set to 1, allows a node to become a
      floating root.

The parameter values are assigned depending on the selected metric.
The best values for these parameters are determined by the
requirement of the specific RPL deployment. For instance, if we use
ETX as the selected metric and UDP as the transport protocol, we
should use a small MAX_LINK_METRIC (e.g., ETX of 1.1) so that link-
layer retransmissions are sufficient to provide a good chance of end-
to-end reliability.

The working group has extensive experience routing with the ETX
metric [Hui08b]. Based on those experiences, the following values
are RECOMMENDED when ETX is the selected metric:

   MAX_LINK_METRIC: 512. Disallow links with greater than 4 expected
   transmission counts on the selected path.

   MAX_PATH_COST: 32768. Disallow paths with greater than 256
   expected transmission counts.

   PARENT_SWITCH_THRESHOLD: 192. Switch to a new path only if it is
   expected to require at least 1.5 fewer transmissions than the
   current path.

   PARENT_SET_SIZE: 3. If the preferred parent is not available, two
   candidate parents are still available without triggering a new
   round of route discovery.

   ALLOW_FLOATING_ROOT: 0. Do not allow a node to become a floating
   root.

6. Manageability

   Section 18 of [RFC6550] depicts the management of RPL. This
   specification inherits from that section and its subsections, with
   the exception that metrics as specified in [RFC6551] are not used and
   do not require management.

6.1. Device Configuration

   An implementation SHOULD allow the following parameters to be
   configured at installation time: MAX_LINK_METRIC, MAX_PATH_COST,
   PARENT_SWITCH_THRESHOLD, PARENT_SET_SIZE, and ALLOW_FLOATING_ROOT.
   An implementation MAY allow these parameters to be configured
   dynamically at run time once a network has been deployed.

   A MRHOF implementation MUST support the DODAG Configuration option as
   described in [RFC6550] and apply the parameters it specifies. Care
   should be taken in the relationship between the MRHOF
   PARENT_SWITCH_THRESHOLD parameter and the RPL MaxRankIncrease

parameter. For example, if MaxRankIncrease is smaller than
PARENT_SWITCH_THRESHOLD, a RPL node using MRHOF could enter a
situation in which its current preferred parent causes the node's
Rank to increase more than MaxRankIncrease but MRHOF does not change
preferred parents. This could cause the node to leave the routing
topology even though there may be other members of the parent set
that would allow the node's Rank to remain within MaxRankIncrease.

Unless configured otherwise, a MRHOF implementation SHOULD use the
default parameters as specified in Section 5.

Because of the partially coupled relationship between Rank and metric
values, networks using MRHOF require care in setting
MinHopRankIncrease. A large MinHopRankIncrease will cause MRHOF to
be unable to select paths with different hop counts but similar
metric values. If MinHopRankIncrease is large enough that its
increment is greater than that caused by link cost, then metrics will
be used to select a preferred parent, but the advertised Rank will be
a simple hop count. This behavior might be desirable, but it also
might be unintended; care is recommended.

With ETX as the selected metric, RPL's Rank advertisement rules can
require a DODAG Root to advertise a Rank higher than its
corresponding ETX value, as a DODAG Root advertises a Rank of
MinHopRankIncrease. Because all DODAG Roots within a DODAG Version
advertise the same Rank, this constant value typically does not
affect route selection. Nevertheless, it means that if a DODAG
Version has a MinHopRankIncrease of M and a path has an advertised
ETX of E, then the actual ETX of the path is likely closer to a value
of E-M than a value of E.

6.2. Device Monitoring

A MRHOF implementation should provide an interface for monitoring its
operation. At a minimum, the information provided should include:

   DAG information as specified in Section 6.3.1 of [RFC6550],
   including the DODAGID, the RPLInstanceID, the Mode of Operation,
   the Rank of this node, the current Version Number, and the value
   of the Grounded flag.

   A list of neighbors indicating the preferred parent. The list
   should indicate, for each neighbor, the Rank, the current Version
   Number, the value of the Grounded flag, and associated metrics.

7. Acknowledgements

   Thanks to Antonio Grilo, Nicolas Tsiftes, Matteo Paris, JP Vasseur,
   and Phoebus Chen for their comments. Thanks to Barry Leiba, Brian
   Haberman, Martin Stiemerling, Ralph Droms, Robert Sparks, Russ
   Housley, Stephen Farrell, Wesley Eddy, Miguel A. Garcia, Mukul Goyal,
   and Michael Richardson for their feedback during the publication
   phase of this document.

8. IANA Considerations

   Per this document, IANA has allocated value 1 from the "Objective
   Code Point (OCP)" sub-registry of the "Routing Protocol for Low Power
   and Lossy Networks (RPL)" registry.

9. Security Considerations

   This specification makes simple extensions to RPL and so is
   vulnerable to and benefits from the security issues and mechanisms
   described in [RFC6550] and [ROLL-SEC]. This document does not
   introduce new flows or new messages, and thus requires no specific
   mitigation for new threats.

   MRHOF depends on information exchanged in a number of RPL protocol
   elements. If those elements were compromised, then an implementation
   of MRHOF might generate the wrong path for a packet, resulting in it
   being misrouted. Therefore, deployments are RECOMMENDED to use RPL
   security mechanisms if there is a risk that routing information might
   be modified or spoofed.

10. References

10.1. Normative References

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC6550]   Winter, T., Thubert, P., 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., Kim, M., Pister, K., Dejean, N., and D.
               Barthel, "Routing Metrics Used for Path Calculation in
               Low-Power and Lossy Networks", RFC 6551, March 2012.

10.2. Informative References

   [Hui08b]   Hui, J. and D. Culler, "IP is dead, long live IP for
                wireless sensor networks", Proceedings of the 6th ACM
                Conference on Embedded Networked Systems SenSys 2008,
                November 2008,
                <http://portal.acm.org/citation.cfm?id=1460412.1460415>.

   [ROLL-SEC]   Tsao, T., Alexander, R., Dohler, M., Daza, V., and A.
                Lozano, "A Security Framework for Routing over Low Power
                and Lossy Networks", Work in Progress, January 2012.

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

Authors' Addresses

   Omprakash Gnawali
   University of Houston
   PGH 577, University of Houston
   Houston, TX 77204
   USA

   Phone: +1 713 743 3356
   EMail: gnawali@cs.uh.edu

   Philip Levis
   Stanford University
   412 Gates Hall, Stanford University
   Stanford, CA 94305
   USA

   EMail: pal@cs.stanford.edu