6TiSCH D. Dujovne, Ed.
Internet-Draft Universidad Diego Portales
Intended status: Standards Track LA. Grieco
Expires: May 4, 2017 Politecnico di Bari
MR. Palattella
Luxembourg Institute of Science and Technology (LIST)
N. Accettura
LAAS-CNRS
October 31, 2016

6TiSCH 6top Scheduling Function Zero (SF0)
draft-ietf-6tisch-6top-sf0-02

Abstract

This document defines a Scheduling Function called "Scheduling Function Zero" (SF0). SF0 dynamically adapts the number of allocated cells between neighbor nodes, based on the amount of currently allocated cells and the neighbor nodes' cell requirements. Neighbor nodes negotiate in a distributed neighbor-to-neighbor basis the number of cell(s) to be added/deleted. SF0 uses the 6P signaling messages to add/delete cells in the schedule. This function selects the candidate cells from the schedule, defines which cells will be added/deleted and triggers the allocation/deallocation process.

Requirements Language

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].

Status of This Memo

This Internet-Draft will expire on May 4, 2017.

Copyright Notice

Table of Contents

1. TEMPORARY EDITORIAL NOTES

This document is an Internet Draft, so it is work-in-progress by nature. It contains the following work-in-progress elements:

- "TODO" statements are elements which have not yet been written by the authors for some reason (lack of time, ongoing discussions with no clear consensus, etc). The statement does indicate that the text will be written at some time.
- "TEMPORARY" appendices are there to capture current ongoing discussions, or the changelog of the document. These appendices will be removed in the final text.
- "IANA_" identifiers are placeholders for numbers assigned by IANA. These placeholders are to be replaced by the actual values they represent after their assignment by IANA.
- The string "REMARK" is put before a remark (questions, suggestion, etc) from an author, editor of contributor. These are on-going discussions at the time to writing, NOT part of the final text.
- This section will be removed in the final text.

2. Introduction

This document defines a minimal Scheduling Function for the 6top sublayer [I-D.ietf-6tisch-6top-protocol], called "Scheduling Function Zero" (SF0). SF0 is designed to offer the minimal set of functionalities to be usable in a wide range of applications. SF0 defines two algorithms: The Scheduling Algorithm defines the number of cells to allocate/delete between two neighbours and the relocation algorithm defines when to relocate a cell.

The Scheduling Algorithm: A number of TX and/or RX cells must be allocated between neighbor nodes in order to enable data transmission among them. From the allocated cells, a part of them can be under effective use by the neighbours, while the rest of cells are over-provisioned to detect an increase in cell usage without packet loss. The Scheduling Algorithm collects the cell allocation/deletion requests from the neighbors and the number of cells which are currently under usage. First, a Cell Estimation Algotithm calculates the number of required cells by adding the collected values and second, the calculated value is given to the Allocation Policy, which provides stability by adding hystheresis and overprovisioning by deciding when to schedule the new number of cells, according to a threshold. In order to reduce consumption, this algorithm is triggered only when there is a change on the number of effectively used cells or if there is a change on the number of requested cells from a particular node.

The Relocation Algorithm: Allocated cells may experiment packet loss from different sources, such as noise, interference or cell collision (after the same cell is allocated by other nodes in range on the network). In order to avoid this problem, Packet Delivery Rate (PDR) is monitored periodically for each allocated cell. A relocation is triggered when the PDR value drops below a certain threshold, compared to the average PDR of the rest of allocated cells. The destination location on the schedule is defined randomly.

To synthesize, a node running SF0 determines when to add/delete cells in a three-step process:

1. It waits for a triggering event (Section 4).
2. It applies the Cell Estimation Algorithm (CEA) for a particular neighbor to determine how many cells are required to that neighbor (Section 5).
3. It applies the Allocation Policy to compare the number of required cells to the number of already scheduled cells, and determines the number of cells to add/delete (Section 6).

We expect additional SFs, offering more functionalities for a more specific use case, to be defined in future documents. SF0 addresses the requirements for a scheduling function listed in Section 5.2 [TODO: update if needed] from [I-D.ietf-6tisch-6top-protocol], and follows the recommended outline listed in Section 5.3 [TODO: update if needed] of [I-D.ietf-6tisch-6top-protocol]. This document follows the terminology defined in [I-D.ietf-6tisch-terminology].

3. Scheduling Function Identifier

The Scheduling Function Identifier (SFID) of SF0 is IANA_SFID_SF0.

4. SF0 Triggering Events

We RECOMMEND SF0 to be triggered at least by the following events:

1. If there is a change in the current number of used cells
2. If there is a successful cell allocation/deallocation process with the neighbour.

This allows SF0 to be triggered by any change in locally generated or incoming traffic. The exact mechanism of when SF0 is triggered is implementation-specific.

5. SF0 Cell Estimation Algorithm

The Cell Estimation Algorithm takes into account the new incoming cell requirements from the neighbor node and the current outgoing number of effectively used cells. This allows the algorithm to estimate a new number of cells to be allocated. As a consequence, the Cell Estimation Algorithm for SF0 follows the steps described below:

1. Collect the new incoming cell requirements from the neighbor
2. Collect the current number of effectively used cells
3. Calculate the new number of cells to be allocated by adding the current number of effectively used cells and the new incoming cells requirement
4. Submit the request to the allocation policy as REQUIREDCELLS
5. Return to step 1 and wait for a triggering event.

A new incoming cell requirement is the result of a successful allocation process from the neighbor. TODO/REMARK: Add a number of cells to the required cells as OVERPROVISION to guarantee that the growth on the effectively used cells can be identified without packet loss. This value is implementation specific. A value of OVERPROVISION equal to zero leads to queue growth and possible packet loss, since there are no overprovisioned cells to detect if there is a growth of effectively used cells.

6. SF0 Allocation Policy

The "Allocation Policy" is the set of rules used by SF0 to decide when to add/delete cells to a particular neighbor to satisfy the cell requirements.

SF0 uses the following parameters:

SCHEDULEDCELLS: The number of cells scheduled from the current node to a particular neighbor.
REQUIREDCELLS: The number of cells calculated by the Cell Estimation Algorithm from the current node to that neighbor.
SF0THRESH: Threshold parameter introducing cell over-provisioning in the allocation policy. It is a non-negative value expressed as number of cells. The definition of this value is implementation-specific. A setting of SF0THRESH>0 will cause the node to allocate at least SF0THRESH cells to each of its' neighbors.

The SF0 allocation policy compares REQUIREDCELLS with SCHEDULEDCELLS and decides to add/delete cells taking into account SF0THRESH. This is illustrated in Figure 1.

```
                   SCHEDULEDCELLS
       <--------------------------------->
      +---+---+---+---+---+---+---+---+---+
      |   |   |   |   |   |   |   |   |   |
      +---+---+---+---+---+---+---+---+---+
                      |<----------------->|
                      |     SF0THRESH     |
                      |                   |
      REQUIREDCELLS   |                   |
      +---+---+       |                   |       DELETE
      |   |   |       |                   |       ONE/MORE
      +---+---+       |                   |       CELLS
                      |                   |
          REQUIREDCELLS                   |
      +---+---+---+---+---+---+           |       DO
      |   |   |   |   |   |   |           |       NOTHING
      +---+---+---+---+---+---+           |
                      |                   |
                  REQUIREDCELLS           |
      +---+---+---+---+---+---+---+---+---+---+   ADD
      |   |   |   |   |   |   |   |   |   |   |   ONE/MORE
      +---+---+---+---+---+---+---+---+---+---+   CELLS
```

Figure 1: The SF0 Allocation Policy

1. If REQUIREDCELLS<(SCHEDULEDCELLS-SF0THRESH), delete one or more cells.
2. If (SCHEDULEDCELLS-SF0THRESH)<=REQUIREDCELLS<=SCHEDULEDCELLS, do nothing.
3. If SCHEDULEDCELLS<=REQUIREDCELLS, add one or more cells.

When SF0THRESH equals 0, any discrepancy between REQUIREDCELLS and SCHEDULEDCELLS triggers an action to add/delete cells. Positive values of SF0THRESH reduce the number of 6P Transactions.

7. Rules for CellList

There are two methods to define the CellList: The Whitelist method, which fills the CellList with the number of proposed cells to the neighbour, and the Blacklist, which fills the CellList with the cells which cannot be used by the neighbour. The rule to select the method is implementation-specific. When issuing a 6top ADD Request, SF0 executes the following sequence:

Whitelist case:

The Transaction Source node: Prepares the CellList field by selecting randomly the required cells, verifying that the slot

offset and channel offset are not occupied and choose channelOffset randomly for each cell.
The Transaction Destination node: Goes through the cells in the CellList in order, verifying whether there are no slotOffset conflicts.

Blacklist case:

The Transaction Source node: Prepares the CellList field by building a list of currently scheduled cells into the CellList.
The Transaction Destination node: Selects randomly the required cells from the unallocated cells on the schedule, verifying that the slot offset and channel offset are not occupied from the ones on the CellList.

8. 6P Timeout Value

The general timeout equals the equivalent time of the number of slots until the next scheduled cell. TODO/REMARK: The exact calculation is currently under discussion on the Mailing List.

9. Meaning of Metadata Information

The Metadata 16-bit field is used as follows:

BITS 0-7 [SLOTFRAME] are used to identify the slotframe number
BITS 8-14 are RESERVED
BIT 15 [WBLIST] is used to indicate that the CellList provided is a Whitelist (value=0) or a Blacklist (value=1).

10. Node Behavior at Boot

In order to define a known state after the node is restarted, a CLEAR command is issued to each of the neighbor nodes to enable a new allocation process. The 6P Initial Timeout Value provided by SF0 should allow for the maximum number of TSCH link-layer retries, as defined by Section 4.3.4 of [I-D.ietf-6tisch-6top-protocol]. TODO/REMARK: The initial timeout is currently under discussion.

11. Relocating Cells

SF0 uses Packet Delivery Rate (PDR) statistics to monitor the currently allocated cells for cell relocation (by changing their slotOffset and/or channelOffset). When the PDR of one or more softcells is below 20% of the average PDR of the rest of the scheduled cells, SF0 relocates the cell(s) to a number of available cells selected randomly. REMARK: This criteria is currently under discussion; simulation/experimentation is required to either adjust the threshold or to change the process.

12. Forced Cell Deletion Policy

TODO: When all the cells are scheduled, we need a policy to free cells, for example, under alarm conditions or if a node disappears from the neighbor list.

13. 6P Error Handling

A node implementing SF0 handles a 6P Response depending on the Return Code it contains:

RC_SUCCESS:
If the number of elements in the CellList is the number of cells specified in the NumCells field of the 6P ADD Request, the operation is complete. The node does not take further action. If the number of elements in the CellList is smaller (possibly 0) than the number of cells specified in the NumCells field of the 6P ADD Request, the neighbor has received the request, but less than NumCells of the cells in the CellList were. In that case, the node MAY retry immediately with a different CellList if the amount of storage space permits, or build a new (random) CellList.

RC_ERR_VER: The node MUST NOT retry immediately. The node MAY add the neighbor node to a blacklist. The node MAY retry to contact this neighbor later.

RC_ERR_SFID: The node MUST NOT retry immediately. The node MAY add the neighbor node to a blacklist. The node MAY retry to contact this neighbor later.

RC_ERR_GEN: The node MUST issue a CLEAR command tot he neighbour.

RC_ERR_BUSY: Wait for a timeout and restart the scheduling process.

RC_ERR_NORES: Wait for a timeout and restart the scheduling process.

RC_ERR_RESET: Abort 6P Transaction

RC_ERR: Abort 6P Transaction. The node MAY retry to contact this neighbor later.

14. Examples

TODO

15. Implementation Status

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC6982]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was

supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC6982], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

OpenWSN: This specification is implemented in the OpenWSN project [OpenWSN]. The authors of this document are collaborating with the OpenWSN community to gather feedback about the status and performance of the protocols described in this document. Results from that discussion will appear in this section in future revision of this specification.

16. Security Considerations

TODO

17. IANA Considerations

- IANA_SFID_SF0

18. 6P Compliance

- MUST specify an identifier for that SF. OK
- MUST specify the rule for a node to decide when to add/delete one or more cells to a neighbor. OK
- MUST specify the rule for a Transaction source to select cells to add to the CellList field in the 6P ADD Request. OK
- MUST specify the rule for a Transaction destination to select cells from CellList to add to its schedule. OK
- MUST specify a value for the 6P Timeout, or a rule/equation to calculate it.
- MUST specify a meaning for the "Metadata" field in the 6P ADD Request. OK
- MUST specify the behavior of a node when it boots. OK
- MUST specify what to do after an error has occurred (either the node sent a 6P Response with an error code, or received one). OK
- MUST specify the list of statistics to gather. An example statistic if the number of transmitted frames to each neighbor. In case the SF requires no statistics to be gathered, the specific of the SF MUST explicitly state so.

- SHOULD clearly state the application domain the SF is created for. OK
- SHOULD contain examples which highlight normal and error scenarios.
- SHOULD contain a list of current implementations, at least during the I-D state of the document, per [RFC6982].
- SHOULD contain a performance evaluation of the scheme, possibly through references to external documents.
- MAY redefine the format of the CellList? field.

19. Acknowledgments

Thanks to Kris Pister for his contribution in designing the default Bandwidth Estimation Algorithm. Thanks to Qin Wang and Thomas Watteyne for their support in defining the interaction between SF0 and the 6top sublayer.

This work is partially supported by the Fondecyt 1121475 Project, the Inria-Chile "Network Design" group, and the IoT6 European Project (STREP) of the 7th Framework Program (Grant 288445).

20. References

20.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

20.2. Informative References

[RFC6982] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", RFC 6982, DOI 10.17487/RFC6982, July 2013, <http://www.rfc-editor.org/info/rfc6982>.

[I-D.ietf-6tisch-terminology] Palattella, M., Thubert, P., Watteyne, T., and Q. Wang, "Terminology in IPv6 over the TSCH mode of IEEE 802.15.4e", draft-ietf-6tisch-terminology-07 (work in progress), March 2016.

[I-D.ietf-6tisch-6top-protocol] Wang, Q. and X. Vilajosana, "6top Protocol (6P)", draft-ietf-6tisch-6top-protocol-02 (work in progress), July 2016.

[OpenWSN] Watteyne, T., Vilajosana, X., Kerkez, B., Chraim, F., Weekly, K., Wang, Q., Glaser, S., and K. Pister, "OpenWSN: a Standards-Based Low-Power Wireless Development Environment", Transactions on Emerging Telecommunications Technologies , August 2012.

Appendix A. [TEMPORARY] Changelog

- draft-ietf-6tisch-6top-sf0-02

  * Editorial changes (figs, typos, ...)

Authors' Addresses

Diego Dujovne (editor)
Universidad Diego Portales
Escuela de Informatica y Telecomunicaciones
Av. Ejercito 441
Santiago, Region Metropolitana
Chile

Phone: +56 (2) 676-8121
Email: diego.dujovne@mail.udp.cl

Luigi Alfredo Grieco
Politecnico di Bari
Department of Electrical and Information Engineering
Via Orabona 4
Bari 70125
Italy

Phone: 00390805963911
Email: a.grieco@poliba.it

Maria Rita Palattella
Luxembourg Institute of Science and Technology (LIST)
Department 'Environmental Research and Innovation' (ERIN)
41, rue du Brill
Belvaux L-4422
Grand-duchy of Luxembourg

Phone: +352 275 888-5055
Email: mariarita.palattella@list.lu

Nicola Accettura
LAAS-CNRS
7, avenue du Colonel Roche
Toulouse 31400
France

Phone: +33 5 61 33 69 76
Email: nicola.accettura@laas.fr