draft-ietf-6tisch-6top-sf0-01

D. Dujovne
LA. Grieco
MR. Palattella
N. Accetura
Status

• **Goal:** To describe the On-The-Fly scheduling function as the default SF for the 6tisch stack: Scheduling Function Zero (SF0)

• Updated changes since IETF95: Bandwidth Estimation Algorithm, Allocation Policy, Whitelist/Blacklist, Timeout, Behavior at Boot, Errors.

• Next:
  - Adapt to new changes on 6P.
  - ToDo questions at the end.
Bandwidth Estimation Algorithm

- Selected the **alternative algorithm** from two presented at IETF95.
Cell Allocation Policy

Establish a high SF0THRESH for Overprovisioning

Figure 1: The SF0 Allocation Policy
The **Estimated Bandwidth** is converted to **Required Cells** for the Allocation Policy (according to each of the cells’ PDR).

A relocation request is treated as **new incoming bandwidth**.
Whitelist

- **Transaction Source Node**
  - Select slotOffset Randomly
  - Verify slotOffset Free
  - Choose channelOffset Randomly

  For each Cell, repeat until `NumCell`

- **Transaction Destination Node**
  - CellList, NumCell
  - Verify if slotOffset is free
    - Yes: Allocate Cell
    - No: Check next Cell from CellList

  For each Cell, repeat until `((NumCell) or (CellList empty))`
Blacklist

- **Transaction Source Node**

- **Transaction Destination Node**

For each Cell, repeat until \((\text{NumCell})\) or \((\text{CellList empty})\)

1. Verify if slotOffset is NOT in BlackList
2. Random Cell from empty cells
3. Allocate Cell
5. 6P Timeout Value

The general timeout equals the equivalent time of the number of slots until the next scheduled cell.

- Steady-state timeout: No news until **next scheduled cell** (time difference between timeslots)
Node behaviour at boot

7. 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 neighbour nodes to enable a new allocation process. The 6P Initial Timeout Value provided by SF0 allows the maximum number of TSCH link-layer retries. Given the TSCH parameters for the backoff mechanism, macMinBE and macMaxBE, and the length in seconds of the minimal Slotframe, SM, the timeout value is computed as: \[
\text{timeout} = (2^{(\text{macMaxBE}+1)} - 2^{\text{macMinBE}}) \times \text{SM}
\]

- **Initial timeout** to apply during the bootstrap process. (When do we consider bootstrap finished?)
Errors

- **RC_VER_ERR**: The node MUST NOT retry immediately. The node MAY add the neighbor node on a blacklist. The node MAY retry to contact this neighbor later.
- **RC_SFID_ERR**: The node MUST NOT retry immediately. The node MAY add the neighbor node on a blacklist. The node MAY retry to contact this neighbor later.
- **RC_BUSY**: Wait for a timeout and restart the scheduling process.
- **RC_RESET**: Abort 6P Transaction
- **RC_ERR**: Abort 6P Transaction. The node MAY retry to contact this neighbor later.

• **RC_BUSY** supports **concurrent transactions**.

• Proposal from Qin Wang: use one bit from metadata to differentiate between "**no processing resources**" from "**ongoing concurrent transaction**"
ToDo / Schedule Generations

• Modify according to the new Schedule Generations from 6P.
• Change the behavior at boot to take advantage of this feature and recover from a node crash or disconnection?
Todo / Relocation

- Cell relocation: is 20% of difference below the **average PDR** is a reasonable number to trigger relocation?
- Which is the **right monitoring period** for the relocation process?
ToDo / Cell deletion

• Do we add to the scope of the draft the deletion of cells after a **timeout from a neighbour**? (house cleaning)
• What to do when cell stock is **depleted**?
Questions?