



draft-ietf-6tisch-msf-01

simulation campaign

Yasuyuki Tanaka



The 6TiSCH Simulator

<https://bitbucket.org/6tisch/simulator/>

- The new release, [v1.1.6](#), has just been released 🎉
- What protocols are implemented?
 - RPL : [RFC6550](#), [RFC6552](#)
 - Trickle Algorithm: : [RFC6206](#)
 - 6LoWPAN Fragment : [RFC6282](#), [RFC4944](#)
 - Fragment Forwarding : [draft-ietf-6lo-minimal-fragment-00](#)
 - 6top Protocol (6P) : [RFC8480](#) (not draft-ietf-6tisch-6top-protocol any more)
 - 6TiSCH Minimal Security (secure join) : [draft-ietf-6tisch-minimal-security-07](#)
 - 6TiSCH Minimal Configuration : [RFC8180](#)
 - TSCH : [IEEE802.15.4-2015](#)
- And, of course, MSF ([draft-ietf-6tisch-msf-01](#))

Typical Use Case

1. Come up with an idea...



2. Evaluate the idea with the simulator...



3. Test it with physical devices...



Share your great idea with others!



- Iteration with the simulator is much faster than with real hardware
- You can run the simulator on a computer cluster (much faster!)

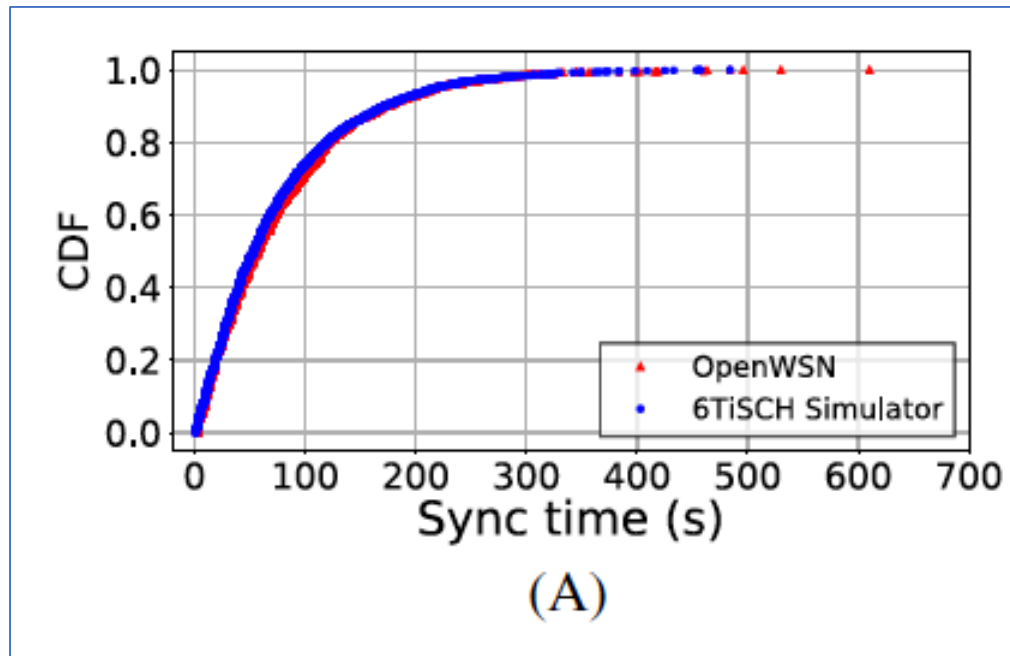
Did you know...?

The 6TiSCH simulator can run with **connectivity trace** which is obtained in a real deployment or a testbed.

Working closely with the real-world



We'll conduct further performance comparison with 6TiSCH stack implementations



E. Municio, et al. "Simulating 6TiSCH Networks", Wiley Transactions on Emerging Telecommunications (ETT), 2018.

We're building 6TiSCH evaluation solutions with testbeds and this simulator

Test environment(s)

- IoT-lab Testbed
 - Saclay, France
 - most-realistic propagation conditions of all IoT-lab sites
 - STM32F + AT86RF231 radio
 - Slow hardware-accelerated encryption -> no link-layer security with default timings
- w-iLab.t Testbed
 - Ghent, Belgium
 - Zolertia Re-Motes (CC2538-based)
- The 6TiSCH Simulator
 - <https://bitbucket.org/6tisch/simulator>
 - Python-based discrete-event simulator
 - Quick performance estimation



w-iLab.t facilities

Mališa Vučinić, et al. "SODA: 6TiSCH Open Data Action", 1st Workshop on Benchmarking Cyber-Physical Networks and Systems (CPSBench), 2018



Lessons on MSF

(from Implementation and preliminary simulations)

- Only a few or no dedicated cells are allocated...
 - NumCellsUsed hardly reaches LIM_NUMCELLSUSED_HIGH when only an autonomous cell is available to communicate with the parent.
 - This is because NumCellsUsed is not incremented during backoff wait delay for the TSCH retransmission algorithm.
- Then, each mote tends to have one TX autonomous cell to its parent and one RX autonomous cell from its children, without dedicated cell
 - Inflow exceeds outflow; TX queue grows; packets are dropped...
 - The frame pending bit feature could make the situation worse.



Lessons on MSF (cont'd)

- Other comments can be found on 6TiSCH WG ML
 - <https://www.ietf.org/mail-archive/web/6tisch/current/msg06067.html>
 - <https://www.ietf.org/mail-archive/web/6tisch/current/msg06098.html>
 - <https://www.ietf.org/mail-archive/web/6tisch/current/msg06099.html>
- Now we're evaluating MSF with the latest simulator release. Will share the result.

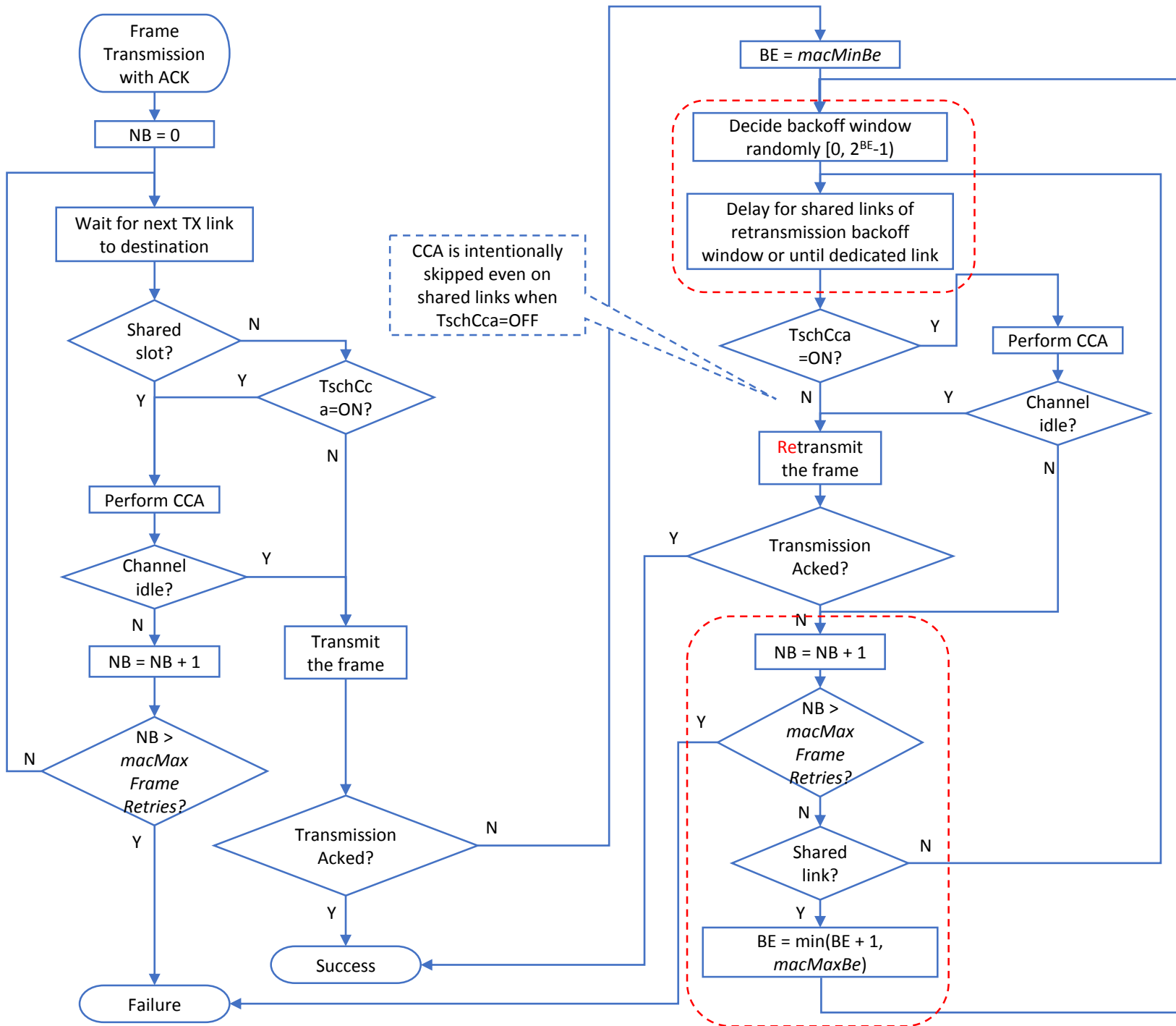
Conclusion



- Introducing the autonomous cell may not be as easy as we expected
- Frame pending bit is interesting!
 - However, it's underspecified...; looking forward to new text by TGm4d 😊
 - https://mentor.ieee.org/802.15/documents?is_group=04md
 - A task to rewrite Section 7.2.1.3 of IEEE802.15.4-2015 is tracked as CID 93
 - Not sure what we should do...
 1. when a frame is lost in an unscheduled slot
 2. when a frame having the pending bit on is received successfully but there is no memory to handle another frame
- You want to try by yourself? Get [the 6TiSCH Simulator!](#)



A complete flowchart of TSCH Transmission for unicast frames without PCA (updated, November of 2018)
 For further information, see this thread:
<https://www.ietf.org/mail-archive/web/6tisch/current/msg05936.html>



note:
 modifications to the original flowchart
 BE is a global variable
 NB and backoff window are per-frame variables