Using IEEE802.15.4e TSCH in an IoT context:
Overview, Problem Statement and Goals
draft-ietf-6tisch-tsch-05

Abstract

   This document describes the environment, problem statement, and goals
   for using the IEEE802.15.4e TSCH MAC protocol in the context of LLNs.
   The set of goals enumerated in this document form an initial set
   only.

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 http://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 July 12, 2015.

Copyright Notice

Table of Contents

1. Introduction

   IEEE802.15.4e [IEEE802154e] was published in 2012 as an amendment to
   the Medium Access Control (MAC) protocol defined by the
   IEEE802.15.4-2011 [IEEE802154] standard. IEEE802.15.4e will be
   rolled into the next revision of IEEE802.15.4, scheduled to be
   published in 2015. The Timeslotted Channel Hopping (TSCH) mode of
   IEEE802.15.4e is the object of this document.

   This document describes the main issues arising from the adoption of
   the IEEE802.15.4e TSCH in the LLN context, following the terminology
   defined in [I-D.ietf-6tisch-terminology].

   TSCH was designed to allow IEEE802.15.4 devices to support a wide
   range of applications including, but not limited to, industrial ones
   [IEEE802154e]. At its core is a medium access technique which uses
   time synchronization to achieve ultra low-power operation and channel
   hopping to enable high reliability. Synchronization accuracy impacts
   power consumption, and can vary from micro-seconds to milli-seconds
   depending on the solution. This is very different from the "legacy"
   IEEE802.15.4 MAC protocol, and is therefore better described as a
   "redesign". TSCH does not amend the physical layer; i.e., it can
   operate on any IEEE802.15.4-compliant hardware.

   IEEE802.15.4e is the latest generation of ultra-lower power and
   reliable networking solutions for LLNs. [RFC5673] discusses
   industrial applications, and highlights the harsh operating
   conditions as well as the stringent reliability, availability, and
   security requirements for an LLN to operate in an industrial
   environment. In these environments, vast deployment environments
   with large (metallic) equipment cause multi-path fading and
   interference to thwart any attempt of a single-channel solution to be
   reliable; the channel agility of TSCH is the key to its ultra high
   reliability. Commercial networking solutions are available today in
   which nodes consume 10's of micro-amps on average [CurrentCalculator]
   with end-to-end packet delivery ratios over 99.999%
   [doherty07channel].

   IEEE802.15.4e has been designed for low-power constrained devices,
   often called "motes". Several terms are used in the IETF to refer to
   those devices, including "LLN nodes" [RFC7102] and "constrained
   nodes" [RFC7228]. In this document, we use the generic (and shorter)
   term "node", used as a synonym for "LLN node", "constrained node" or
   "mote".

   Bringing industrial-like performance into the LLN stack developed by
   Internet of Things (IoT) related IETF working groups such as 6Lo,
   ROLL and CoRE opens up new application domains for these networks.

Sensors deployed in smart cities [RFC5548] will be able to be
installed for years without needing battery replacement. "Umbrella"
networks will interconnect smart elements from different entities in
smart buildings [RFC5867]. Peel-and-stick switches will obsolete the
need for costly conduits for lighting solutions in smart homes
[RFC5826].

IEEE802.15.4e TSCH focuses on the MAC layer only. This clean
layering allows for TSCH to fit under an IPv6 enabled protocol stack
for LLNs, running 6LoWPAN [RFC6282], IPv6 Routing Protocol for Low
power and Lossy Networks (RPL) [RFC6550] and the Constrained
Application Protocol (CoAP) [RFC7252]. What is missing is a Logical
Link Control (LLC) layer between the IP abstraction of a link and the
TSCH MAC, which is in charge of scheduling a timeslot for a given
packet coming down the stack from the upper layer.

While [IEEE802154e] defines the mechanisms for a TSCH node to
communicate, it does not define the policies to build and maintain
the communication schedule, match that schedule to the multi-hop
paths maintained by RPL, adapt the resources allocated between
neighbor nodes to the data traffic flows, enforce a differentiated
treatment for data generated at the application layer and signaling
messages needed by 6LoWPAN and RPL to discover neighbors, react to
topology changes, self-configure IP addresses, or manage keying
material.

In other words, IEEE802.15.4e TSCH is designed to allow optimizations
and strong customizations, simplifying the merging of TSCH with a
protocol stack based on IPv6, 6LoWPAN, and RPL.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in RFC 2119 [RFC2119].

3. TSCH in the LLN Context

To map the services required by the IP layer to the services provided
by the link layer, an adaptation layer is used
[palattella12standardized]. The 6LoWPAN working group started
working in 2007 on specifications for transmitting IPv6 packets over
IEEE802.15.4 networks [RFC4919]. Low-power WPANs are characterized
by small packet sizes, support for addresses with different lengths,
low bandwidth, star and mesh topologies, battery powered devices, low
cost, large number of devices, unknown node positions, high
unreliability, and periods during which communication interfaces are
turned off to save energy. Given these features, it is clear that

the adoption of IPv6 on top of a Low-Power WPAN is not
straightforward, but poses strong requirements for the optimization
of this adaptation layer.

For instance, due to the IPv6 default minimum MTU size (1280 bytes),
an un-fragmented IPv6 packet is too large to fit in an IEEE802.15.4
frame. Moreover, the overhead due to the 40-byte long IPv6 header
wastes the scarce bandwidth available at the PHY layer [RFC4944].
For these reasons, the 6LoWPAN working group has defined an effective
adaptation layer [RFC6282]. Further issues encompass the auto-
configuration of IPv6 addresses [RFC2460][RFC4862], the compliance
with the recommendation on supporting link-layer subnet broadcast in
shared networks [RFC3819], the reduction of routing and management
overhead [RFC6606], the adoption of lightweight application protocols
(or novel data encoding techniques), and the support for security
mechanisms (confidentiality and integrity protection, device
bootstrapping, key establishment, and management).

These features can run on top of TSCH. There are, however, important
issues to solve, as highlighted in Section 4.

Routing issues are challenging for 6LoWPAN, given the low-power and
lossy radio links, the battery-powered nodes, the multi-hop mesh
topologies, and the frequent topology changes due to mobility.
Successful solutions take into account the specific application
requirements, along with IPv6 behavior and 6LoWPAN mechanisms
[palattella12standardized]. The ROLL working group has defined RPL
in [RFC6550]. RPL can support a wide variety of link layers,
including ones that are constrained, potentially lossy, or typically
utilized in conjunction with host or router devices with very limited
resources, as in building/home automation [RFC5867][RFC5826],
industrial environments [RFC5673], and urban applications [RFC5548].
RPL is able to quickly build up network routes, distribute routing
knowledge among nodes, and adapt to a changing topology. In a
typical setting, nodes are connected through multi-hop paths to a
small set of root devices, which are usually responsible for data
collection and coordination. For each of them, a Destination
Oriented Directed Acyclic Graph (DODAG) is created by accounting for
link costs, node attributes/status information, and an Objective
Function, which maps the optimization requirements of the target
scenario.

The topology is set up based on a Rank metric, which encodes the
distance of each node with respect to its reference root, as
specified by the Objective Function. Regardless of the way it is
computed, the Rank monotonically decreases along the DODAG towards
the root, building a gradient. RPL encompasses different kinds of
traffic and signaling information. Multipoint-to-Point (MP2P) is the

dominant traffic in LLN applications. Data is routed towards nodes
with some application relevance, such as the LLN gateway to the
larger Internet, or to the core of private IP networks. In general,
these destinations are the DODAG roots and act as data collection
points for distributed monitoring applications. Point-to-Multipoint
(P2MP) data streams are used for actuation purposes, where messages
are sent from DODAG roots to destination nodes. Point-to-Point (P2P)
traffic allows communication between two devices belonging to the
same LLN, such as a sensor and an actuator. A packet flows from the
source to the common ancestor of those two communicating devices,
then downward towards the destination. RPL therefore has to discover
both upward routes (i.e. from nodes to DODAG roots) in order to
enable MP2P and P2P flows, and downward routes (i.e. from DODAG roots
to nodes) to support P2MP and P2P traffic.

Section 4 highlights the challenges that need to be addressed to use
RPL on top of TSCH.

Several open-source initiatives have emerged around TSCH. The
OpenWSN project [OpenWSN][OpenWSNETT] is an open-source
implementation of a standards-based protocol stack, which aims at
evaluating the applicability of TSCH to different applications. This
implementation was used as the foundation for an IP for Smart Objects
Alliance (IPSO) [IPSO] interoperability event in 2011. In the
absence of a standardized scheduling mechanism for TSCH, a "slotted
Aloha" schedule was used.

4. Problems and Goals

As highlighted in Appendix A, TSCH differs from traditional low-power
MAC protocols because of its scheduled nature. TSCH defines the
mechanisms to execute a communication schedule, yet it is the entity
that sets up that schedule which controls the topology of the
network. This scheduling entity also controls the resources
allocated to each link in that topology.

How this entity should operate is out of scope of TSCH. The
remainder of this section highlights the problems this entity needs
to address. For simplicity, we refer to this entity by the generic
name "LLC". Note that the 6top sublayer, currently being defined in
[I-D.wang-6tisch-6top-sublayer], can be seen as an embodiment of this
generic "LLC".

Some of the issues the LLC needs to target might overlap with the
scope of other protocols (e.g., 6LoWPAN, RPL, and RSVP). In this
case, it is entailed that the LLC will profit from the services
provided by other protocols to pursue these objectives.

4.1. Network Formation

The LLC needs to control the way the network is formed, including how
new nodes join, and how already joined nodes advertise the presence
of the network. The LLC needs to:

1. Define the Information Elements included in the Enhanced Beacons
    advertising the presence of the network.

2. For a new node, define rules to process and filter received
    Enhanced Beacons.

3. Define the joining procedure. This might include a mechanism to
    assign a unique 16-bit address to a node, and the management of
    initial keying material.

4. Define a mechanism to secure the joining process and the
    subsequent optional process of scheduling more communication
    cells.

4.2. Network Maintenance

Once a network is formed, the LLC needs to maintain the network's
health, allowing for nodes to stay synchronized. The LLC needs to:

1. Manage each node's time source neighbor.

2. Define a mechanism for a node to update the join priority it
    announces in its Enhanced Beacon.

3. Schedule transmissions of Enhanced Beacons to advertise the
    presence of the network.

4.3. Multi-Hop Topology

RPL, given a weighted connectivity graph, determines multi-hop
routes. The LLC needs to:

1. Define a mechanism to gather topological information, node and
    link state, which it can then feed to RPL.

2. Ensure that the TSCH schedule contains cells along the multi-hop
    routes identified by RPL.

3. Where applicable, maintain independent sets of cells to transport
    independent flows of data.

4.4. Routing and Timing Parents

At all times, a TSCH node needs to have a time source neighbor it can
synchronize to. The LLC therefore needs to assign a time source
neighbor to allow for correct operation of the TSCH network. A time
source neighbors could, or not, be taken from the RPL routing parent
set.

4.5. Resource Management

A cell in a TSCH schedule is an atomic "unit" of resource. The
number of cells to assign between neighbor nodes needs to be
appropriate for the size of the traffic flow. The LLC needs to:

1. Define a mechanism for neighbor nodes to exchange information
    about their schedule and, if applicable, negotiate the addition/
    deletion of cells.

2. Allow for an entity (e.g., a set of devices, a distributed
    protocol, a PCE, etc.) to take control of the schedule.

4.6. Dataflow Control

TSCH defines mechanisms for a node to signal it cannot accept an
incoming packet. It does not, however, define the policy which
determines when to stop accepting packets. The LLC needs to:

1. Define a queuing policy for incoming and outgoing packets.

2. Manage the buffer space, and indicate to TSCH when to stop
    accepting incoming packets.

3. Handle transmissions that have failed. A transmission is
    declared failed when TSCH has retransmitted the packet multiple
    times, without receiving an acknowledgment. This covers both
    dedicated and shared cells.

4.7. Deterministic Behavior

As highlighted in [RFC5673], in some applications, data is generated
periodically and has a well understood data bandwidth requirement,
which is deterministic and predictable. The LLC needs to:

1. Ensure timely delivery of such data.

2. Provide a mechanism for such deterministic flows to coexist with
    bursty or infrequent traffic flows of different priorities.

4.8. Scheduling Mechanisms

Several scheduling mechanisms can be envisioned, and possibly coexist in the same network. For example, [I-D.phinney-roll-rpl-industrial-applicability] describes how the allocation of bandwidth can be optimized by an external Path Computation Element (PCE). Another centralized (PCE-based) traffic-aware scheduling algorithm is defined in [TASA-PIMRC]. Alternatively, two neighbor nodes can adapt the number of cells autonomously by monitoring the amount of traffic, and negotiating the allocation to extra cell when needed. An example of decentralized algorithm is provided in [tinka10decentralized]. This mechanism can be used to establish multi-hop paths in a fashion similar to RSVP. The LLC needs to:

1. Provide a mechanism for two 6TiSCH devices to negotiate the allocation and deallocation of cells between them.

2. Provide a mechanism for device to monitor and manage the 6TiSCH capabilities of a node several hops away.

3. Define an mechanism for these different scheduling mechanisms to coexist in the same network.

4.9. Secure Communication

Given some keying material, TSCH defines mechanisms to encrypt and authenticate MAC frames. It does not define how this keying material is generated. The LLC needs to:

1. Define the keying material and authentication mechanism needed by a new node to join an existing network.

2. Define a mechanism to allow for the secure transfer of application data between neighbor nodes.

3. Define a mechanism to allow for the secure transfer of signaling data between nodes and the LLC.

5. IANA Considerations

This memo includes no request to IANA.

6. Security Considerations

This memo is an informational overview of existing standards, and does define any new mechanisms or protocols.

It does describe the need for the 6TiSCH WG to define a secure
solution. In particular, Section 4.1 describes security in the join
process. Section 4.9 discusses data frame protection.

7. Acknowledgments

Special thanks to Dominique Barthel, Patricia Brett, Guillaume
Gaillard, Pat Kinney, Ines Robles, Timothy J. Salo, Jonathan Simon,
Rene Struik, Xavi Vilajosana for reviewing the document and providing
valuable feedback. Thanks to the IoT6 European Project (STREP) of
the 7th Framework Program (Grant 288445).

8. References

8.1. Normative References

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

8.2. Informative References

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained
           Application Protocol (CoAP)", RFC 7252, June 2014.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for
           Constrained-Node Networks", RFC 7228, May 2014.

[RFC7102] Vasseur, JP., "Terms Used in Routing for Low-Power and
           Lossy Networks", RFC 7102, January 2014.

[RFC6606] Kim, E., Kaspar, D., Gomez, C., and C. Bormann, "Problem
           Statement and Requirements for IPv6 over Low-Power
           Wireless Personal Area Network (6LoWPAN) Routing", RFC
           6606, May 2012.

[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.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6
           Datagrams over IEEE 802.15.4-Based Networks", RFC 6282,
           September 2011.

[RFC5867] Martocci, J., De Mil, P., Riou, N., and W. Vermeylen,
           "Building Automation Routing Requirements in Low-Power and
           Lossy Networks", RFC 5867, June 2010.

[RFC5826]   Brandt, A., Buron, J., and G. Porcu, "Home Automation
            Routing Requirements in Low-Power and Lossy Networks", RFC
            5826, April 2010.

[RFC5673]   Pister, K., Thubert, P., Dwars, S., and T. Phinney,
            "Industrial Routing Requirements in Low-Power and Lossy
            Networks", RFC 5673, October 2009.

[RFC5548]   Dohler, M., Watteyne, T., Winter, T., and D. Barthel,
            "Routing Requirements for Urban Low-Power and Lossy
            Networks", RFC 5548, May 2009.

[RFC4944]   Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler,
            "Transmission of IPv6 Packets over IEEE 802.15.4
            Networks", RFC 4944, September 2007.

[RFC4919]   Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6
            over Low-Power Wireless Personal Area Networks (6LoWPANs):
            Overview, Assumptions, Problem Statement, and Goals", RFC
            4919, August 2007.

[RFC4862]   Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless
            Address Autoconfiguration", RFC 4862, September 2007.

[RFC3819]   Karn, P., Bormann, C., Fairhurst, G., Grossman, D.,
            Ludwig, R., Mahdavi, J., Montenegro, G., Touch, J., and L.
            Wood, "Advice for Internet Subnetwork Designers", BCP 89,
            RFC 3819, July 2004.

[RFC2460]   Deering, S. and R. Hinden, "Internet Protocol, Version 6
            (IPv6) Specification", RFC 2460, December 1998.

[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-02 (work in
            progress), July 2014.

[I-D.wang-6tisch-6top-sublayer]
            Wang, Q., Vilajosana, X., and T. Watteyne, "6TiSCH
            Operation Sublayer (6top)", draft-wang-6tisch-6top-
            sublayer-01 (work in progress), July 2014.

[I-D.phinney-roll-rpl-industrial-applicability]
            Phinney, T., Thubert, P., and R. Assimiti, "RPL
            applicability in industrial networks", draft-phinney-roll-
            rpl-industrial-applicability-02 (work in progress),
            February 2013.

8.3. External Informative References

   [IEEE802154e]
              IEEE standard for Information Technology, "IEEE std.
              802.15.4e, Part. 15.4: Low-Rate Wireless Personal Area
              Networks (LR-WPANs) Amendment 1: MAC sublayer", April
              2012.

   [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", June 2011.

   [OpenWSN] "Berkeley's OpenWSN Project Homepage",
              <http://www.openwsn.org/>.

   [OpenWSNETT]
              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.

   [IPSO]   "IP for Smart Objects Alliance Homepage",
              <http://www.ipso-alliance.org/>.

   [CurrentCalculator]
              Linear Technology, "Application Note: Using the Current
              Calculator to Estimate Mote Power", August 2012,
              <http://cds.linear.com/docs/en/application-note/
              Application_Note_-
              _Using_the_Current_Calculator_to_Estimate_Mote_Power.pdf>.

   [doherty07channel]
              Doherty, L., Lindsay, W., and J. Simon, "Channel-Specific
              Wireless Sensor Network Path Data", IEEE International
              Conference on Computer Communications and Networks (ICCCN)
              2008, 2007.

   [tinka10decentralized]
              Tinka, A., Watteyne, T., and K. Pister, "A Decentralized
              Scheduling Algorithm for Time Synchronized Channel
              Hopping", Ad Hoc Networks 2010, 2010, <
              http://robotics.eecs.berkeley.edu/~pister/
              publications/2008/TSMP%20DSN08.pdf>.

   [watteyne09reliability]
              Watteyne, T., Mehta, A., and K. Pister, "Reliability
              Through Frequency Diversity: Why Channel Hopping Makes
              Sense", International Conference on Performance Evaluation
              of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-
              WASUN) 2009, Oct. 2009, <http://www.ietf.org/mail-
              archive/web/roll/current/pdfa_EzmuDIv3.pdf>.

   [TASA-PIMRC]
              Palattella, MR., Accettura, N., Dohler, M., Grieco, LA.,
              and G. Boggia, "Traffic Aware Scheduling Algorithm for
              Multi-Hop IEEE 802.15.4e Networks", IEEE PIMRC 2012, Sept.
              2012, < http://www.cttc.es/resources/
              doc/120531-submitted-tasa-25511.pdf>.

   [palattella12standardized]
              Palattella, MR., Accettura, N., Vilajosana, X., Watteyne,
              T., Grieco, LA., Boggia, G., and M. Dohler, "Standardized
              Protocol Stack For The Internet Of (Important) Things",
              IEEE Communications Surveys and Tutorials 2012, Dec. 2012,
              < http://www.cttc.es/resources/doc/121025-
              completestackforiot-clean-4818610916636121981.pdf>.

Appendix A. TSCH Protocol Highlights

   This appendix gives an overview of the key features of the
   IEEE802.15.4e Timeslotted Channel Hopping (TSCH) amendment. It makes
   no attempt at repeating the standard, but rather focuses on the
   following:

   o Concepts which are sufficiently different from traditional
      IEEE802.15.4 networking that they may need to be defined and
      presented precisely.

   o Techniques and ideas which are part of IEEE802.15.4e and which
      might be useful for the work of the 6TiSCH WG.

A.1. Timeslots

   All nodes in a TSCH network are synchronized. Time is sliced up into
   timeslots. A timeslot is long enough for a MAC frame of maximum size
   to be sent from node A to node B, and for node B to reply with an
   acknowledgment (ACK) frame indicating successful reception.

   The duration of a timeslot is not defined by the standard. With
   IEEE802.15.4-compliant radios operating in the 2.4GHz frequency band,
   a maximum-length frame of 127 bytes takes about 4ms to transmit; a
   shorter ACK takes about 1ms. With a 10ms slot (a typical duration),

this leaves 5ms to radio turnaround, packet processing and security
operations.

A.2. Slotframes

Timeslots are grouped into one of more slotframes. A slotframe
continuously repeats over time. TSCH does not impose a slotframe
size. Depending on the application needs, these can range from 10s
to 1000s of timeslots. The shorter the slotframe, the more often a
timeslot repeats, resulting in more available bandwidth, but also in
a higher power consumption.

A.3. Node TSCH Schedule

A TSCH schedule instructs each node what to do in each timeslot:
transmit, receive or sleep. The schedule indicates, for each
scheduled (transmit or receive) cell, a channelOffset and the address
of the neighbor to communicate with.

Once a node obtains its schedule, it executes it:

o For each transmit cell, the node checks whether there is a packet
   in the outgoing buffer which matches the neighbor written in the
   schedule information for that timeslot. If there is none, the
   node keeps its radio off for the duration of the timeslot. If
   there is one, the node can ask for the neighbor to acknowledge it,
   in which case it has to listen for the acknowledgment after
   transmitting.

o For each receive cell, the node listens for possible incoming
   packets. If none is received after some listening period, it
   shuts down its radio. If a packet is received, addressed to the
   node, and passes security checks, the node can send back an
   acknowledgment.

How the schedule is built, updated and maintained, and by which
entity, is outside of the scope of the IEEE802.15.4e standard.

A.4. Cells and Bundles

Assuming the schedule is well built, if node A is scheduled to
transmit to node B at slotOffset 5 and channelOffset 11, node B will
be scheduled to receive from node A at the same slotOffset and
channelOffset.

A single element of the schedule characterized by a slotOffset and
channelOffset, and reserved for node A to transmit to node B (or for

node B to receive from node A) within a given slotframe, is called a
"scheduled cell".

If there is a lot of data flowing from node A to node B, the schedule
might contain multiple cells from A to B, at different times.
Multiple cells scheduled to the same neighbor can be equivalent, i.e.
the MAC layer sends the packet on whichever of these cells shows up
first after the packet was put in the MAC queue. The union of all
cells between two neighbors, A and B, is called a "bundle". Since
the slotframe repeats over time (and the length of the slotframe is
typically constant), each cell gives a "quantum" of bandwidth to a
given neighbor. Modifying the number of equivalent cells in a bundle
modifies the amount of resources allocated between two neighbors.

A.5. Dedicated vs. Shared Cells

By default, each scheduled transmit cell within the TSCH schedule is
dedicated, i.e., reserved only for node A to transmit to node B.
IEEE802.15.4e allows also to mark a cell as shared. In a shared
cell, multiple nodes can transmit at the same time, on the same
frequency. To avoid contention, TSCH defines a back-off algorithm
for shared cells.

A scheduled cell can be marked as both transmitting and receiving.
In this case, a node transmits if it has an appropriate packet in its
output buffer, or listens otherwise. Marking a cell as
[transmit,receive,shared] results in slotted-Aloha behavior.

A.6. Absolute Slot Number

TSCH defines a timeslot counter called Absolute Slot Number (ASN).
When a new network is created, the ASN is initialized to 0; from then
on, it increments by 1 at each timeslot. In detail:

$$ASN = (k*S+t)$$

where k is the slotframe cycle (i.e., the number of slotframe
repetitions since the network was started), S the slotframe size and
t the slotOffset. A node learns the current ASN when it joins the
network. Since nodes are synchronized, they all know the current
value of the ASN, at any time. The ASN is encoded as a 5-byte
number: this allows it to increment for hundreds of years (the exact
value depends on the duration of a timeslot) without wrapping over.
The ASN is used to calculate the frequency to communicate on, and can
be used for security-related operations.

A.7. Channel Hopping

   For each scheduled cell, the schedule specifies a slotOffset and a
   channelOffset. In a well-built schedule, when node A has a transmit
   cell to node B on channelOffset 5, node B has a receive cell from
   node A on the same channelOffset. The channelOffset is translated by
   both nodes into a frequency using the following function:

   frequency = F {(ASN + channelOffset) mod nFreq}

   The function F consists of a look-up table containing the set of
   available channels. The value nFreq (the number of available
   frequencies) is the size of this look-up table. There are as many
   channelOffset values as there are frequencies available (e.g. 16 when
   using IEEE802.15.4-compliant radios at 2.4GHz, when all channels are
   used). Since both nodes have the same channelOffset written in their
   schedule for that scheduled cell, and the same ASN counter, they
   compute the same frequency. At the next iteration (cycle) of the
   slotframe, however, while the channelOffset is the same, the ASN has
   changed, resulting in the computation of a different frequency.

   This results in "channel hopping": even with a static schedule, pairs
   of neighbors "hop" between the different frequencies when
   communicating. A way of ensuring communication happens on all
   available frequencies is to set the number of timeslots in a
   slotframe to a prime number. Channel hopping is a technique known to
   efficiently combat multi-path fading and external interference
   [watteyne09reliability].

A.8. Time Synchronization

   Because of the slotted nature of communication in a TSCH network,
   nodes have to maintain tight synchronization. All nodes are assumed
   to be equipped with clocks to keep track of time. Yet, because
   clocks in different nodes drift with respect to one another, neighbor
   nodes need to periodically re-synchronize.

   Each node needs to periodically synchronize its network clock to
   another node, and it also provides its network time to its neighbors.
   It is up to the entity that manages the schedule to assign an
   adequate time source neighbor to each node, i.e., to indicate in the
   schedule which of neighbor is its "time source neighbor". While
   setting the time source neighbor, it is important to avoid
   synchronization loops, which could result in the formation of
   independent clusters of synchronized nodes.

   TSCH adds timing information in all packets that are exchanged (both
   data and ACK frames). This means that neighbor nodes can

resynchronize to one another whenever they exchange data. In detail,
two methods are defined in IEEE802.15.4e-2012 for allowing a device
to synchronize in a TSCH network: (i) Acknowledgment-Based and (ii)
Frame-Based synchronization. In both cases, the receiver calculates
the difference in time between the expected time of frame arrival and
its actual arrival. In Acknowledgment-Based synchronization, the
receiver provides such information to the sender node in its
acknowledgment. In this case, it is the sender node that
synchronizes to the clock of the receiver. In Frame-Based
synchronization, the receiver uses the computed delta for adjusting
its own clock. In this case, it is the receiver node that
synchronizes to the clock of the sender.

Different synchronization policies are possible. Nodes can keep
synchronization exclusively by exchanging EBs. Nodes can also keep
synchronized by periodically sending valid frames to a time source
neighbor and use the acknowledgment to resynchronize. Both method
(or a combination thereof) are valid synchronization policies; which
one to use depends on network requirements.

A.9. Power Consumption

There are only a handful of activities a node can perform during a
timeslot: transmit, receive, or sleep. Each of these operations has
some energy cost associated to them, the exact value depends on the
the hardware used. Given the schedule of a node, it is
straightforward to calculate the expected average power consumption
of that node.

A.10. Network TSCH Schedule

The schedule entirely defines the synchronization and communication
between nodes. By adding/removing cells between neighbors, one can
adapt a schedule to the needs of the application. Intuitive examples
are:

o Make the schedule "sparse" for applications where nodes need to
   consume as little energy as possible, at the price of reduced
   bandwidth.

o Make the schedule "dense" for applications where nodes generate a
   lot of data, at the price of increased power consumption.

o Add more cells along a multi-hop route over which many packets
   flow.

A.11. Join Process

   Nodes already part of the network can periodically send Enhanced
   Beacon (EB) frames to announce the presence of the network. These
   contain information about the size of the timeslot used in the
   network, the current ASN, information about the slotframes and
   timeslots the beaconing node is listening on, and a 1-byte join
   priority. The join priority field gives information to make a better
   decision of which node to join. Even if a node is configured to send
   all EB frames on the same channel offset, because of the channel
   hopping nature of TSCH described in Appendix A.7, this channel offset
   translates into a different frequency at different slotframe cycles.
   As a result, EB frames are sent on all frequencies.

   A node wishing to join the network listens for EBs. Since EBs are
   sent on all frequencies, the joining node can listen on any frequency
   until it hears an EB. What frequency it listens on is
   implementation-specific. Once it has received one or more EBs, the
   new node enables the TSCH mode and uses the ASN and the other timing
   information from the EB to synchronize to the network. Using the
   slotframe and cell information from the EB, it knows how to contact
   other nodes in the network.

   The IEEE802.15.4e TSCH standard does not define the steps beyond this
   network "bootstrap".

A.12. Information Elements

   TSCH introduces the concept of Information Elements (IEs). An
   information element is a list of Type-Length-Value containers placed
   at the end of the MAC header. A small number of types are defined
   for TSCH (e.g., the ASN in the EB is contained in an IE), and an
   unmanaged range is available for extensions.

   A data bit in the MAC header indicates whether the frame contains
   IEs. IEs are grouped into Header IEs, consumed by the MAC layer and
   therefore typically invisible to the next higher layer, and Payload
   IEs, which are passed untouched to the next higher layer, possibly
   followed by regular payload. Payload IEs can therefore be used for
   the next higher layers of two neighbor nodes to exchange information.

A.13. Extensibility

   The TSCH standard is designed to be extensible. It introduces the
   mechanisms as "building block" (e.g., cells, bundles, slotframes,
   etc.), but leaves entire freedom to the upper layer to assemble
   those. The MAC protocol can be extended by defining new Header IEs.

   An intermediate layer can be defined to manage the MAC layer by
   defining new Payload IEs.

Appendix B.   TSCH Gotchas

   This section lists features of TSCH which we believe are important
   and beneficial to the work of 6TiSCH.

B.1.   Collision Free Communication

   TSCH allows one to design a schedule which yields collision-free
   communication. This is done by building the schedule with dedicated
   cells in such a way that at most one node communicates with a
   specific neighbor in each slotOffset/channelOffset cell. Multiple
   pairs of neighbor nodes can exchange data at the same time, but on
   different frequencies.

B.2.   Multi-Channel vs. Channel Hopping

   A TSCH schedule looks like a matrix of width "slotframe size", S, and
   of height "number of frequencies", nFreq. For a scheduling
   algorithm, cells can be considered atomic "units" to schedule. In
   particular, because of the channel hopping nature of TSCH, the
   scheduling algorithm should not worry about the actual frequency
   communication happens on, since it changes at each slotframe
   iteration.

B.3.   Cost of (continuous) Synchronization

   When there is traffic in the network, nodes which are communicating
   implicitly re-synchronize using the data frames they exchange. In
   the absence of data traffic, nodes are required to synchronize to
   their time source neighbor(s) periodically not to drift in time. If
   they have not been communicating for some time (typically 30s), nodes
   can exchange an dummy data frame to re-synchronize. The frequency at
   which such messages need to be transmitted depends on the stability
   of the clock source, and on how "early" each node starts listening
   for data (the "guard time"). Theoretically, with a 10ppm clock and a
   1ms guard time, this period can be 100s. Assuming this exchange
   causes the node's radio to be on for 5ms, this yields a radio duty
   cycle needed to keep synchronized of 5ms/100s=0.005%. While TSCH does
   requires nodes to resynchronize periodically, the cost of doing so is
   very low.

B.4. **Topology Stability**

   The channel hopping nature of TSCH causes links to be very "stable".
   Wireless phenomena such as multi-path fading and external
   interference impact a wireless link between two nodes differently on
   each frequency. If a transmission from node A to node B fails,
   retransmitting on a different frequency has a higher likelihood of
   succeeding that retransmitting on the same frequency. As a result,
   even when some frequencies are "behaving bad", channel hopping
   "smoothens" the contribution of each frequency, resulting in more
   stable links, and therefore a more stable topology.

B.5. **Multiple Concurrent Slotframes**

   The TSCH standard allows for multiple slotframes to coexist in a
   node's schedule. It is possible that, at some timeslot, a node has
   multiple activities scheduled (e.g. transmit to node B on slotframe
   2, receive from node C on slotframe 1). To handle this situation,
   the TSCH standard defines the following precedence rules:

   1. Transmissions take precedence over receptions;

   2. Lower slotframe identifiers take precedence over higher slotframe
       identifiers.

   In the example above, the node would transmit to node B on slotframe
   2.

Authors' Addresses

   Thomas Watteyne (editor)
   Linear Technology
   32990 Alvarado-Niles Road, Suite 910
   Union City, CA 94587
   USA

   Phone: +1 (510) 400-2978
   Email: twatteyne@linear.com

Maria Rita Palattella
University of Luxembourg
Interdisciplinary Centre for Security, Reliability and Trust
4, rue Alphonse Weicker
Luxembourg L-2721
LUXEMBOURG

Phone: +352 46 66 44 5841
Email: maria-rita.palattella@uni.lu

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

Phone: +39 08 05 96 3911
Email: a.grieco@poliba.it