



HAL
open science

Converging over deterministic networks for an Industrial Internet

Pascal Thubert

► **To cite this version:**

Pascal Thubert. Converging over deterministic networks for an Industrial Internet. Networking and Internet Architecture [cs.NI]. Ecole nationale supérieure Mines-Télécom Atlantique, 2017. English. NNT : 2017IMTA0011 . tel-01828288

HAL Id: tel-01828288

<https://theses.hal.science/tel-01828288>

Submitted on 3 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

**UNIVERSITE
BRETAGNE
LOIRE**

THÈSE / IMT Atlantique

sous le sceau de l'Université Bretagne Loire

pour obtenir le grade de

DOCTEUR DE IMT Atlantique

Mention : Informatique

École Doctorale Matisse

Présentée par

Pascal Thubert

Préparée au département Réseaux, Sécurité et
Multimédia

Laboratoire Irisa

Converging over Deterministic Networks for an Industrial Internet

Converger sur des réseaux déterministes pour un Internet Industriel

Thèse soutenue le 16 mars 2017

devant le jury composé de :

Mischa Dohler

Professor and Director of the CTR, King's College, London – Angleterre / Président

Isabelle Chrisment

Professeur, TELECOM Nancy, Université de Lorraine / Rapporteur

Thierry Turletti

Directeur de Recherche, INRIA Sophia-Antipolis / Rapporteur

Nicolas Montavont

Professeur, IMT Atlantique / Examineur

Géraldine Texier

Maître de Conférences, IMT Atlantique / Examineur

Laurent Toutain

Maître de Conférences, IMT Atlantique / Directeur de thèse

Latif Ladid

Chair at 5G World Alliance & President, IPv6 Forum, Univ. du Luxembourg / Invité

Remerciements/Thanks

Je tiens à remercier:

Thomas Watteyne et Norm Finn pour l'enseignement qu'ils m'ont apporté au travers des années, et la collaboration sans faille dans nos travaux communs qui ont mené à la création de 6TiSCH et DetNet; en reconnaissance de leur savoir apparemment infini et de leur patience totale - ou presque ;)

Zacharie Brodard, étudiant à l'Ecole Polytechnique, et Hao Jiang, étudiant à Télécom Bretagne, des stagiaires vraiment extraordinaires, qui ont effectué les expérimentations détaillées dans cette thèse dans le cadre de leur stages au laboratoire PIRL de Cisco à Paris durant l'été 2016;

Laurent Toutain, Géraldine Texier et Nicolas Montavont, de l'IMT Atlantique, qui ont dirigé cette thèse, et qui ont su me supporter et me guider tout au long de ce projet;

John Apostolopoulos et Mark Webb, qui ont su m'encourager et m'ont donné les moyens de réaliser ce travail;

Latif Ladid, qui m'a initié au milieu des conférences académiques, et m'a insufflé l'idée de faire cette thèse.

I wish to thank:

Thomas Watteyne and Norm Finn for their patient teaching over the years, and the faultless collaboration that lead to the creation of 6TiSCH and DetNet; in recognition of their bottomless knowledge an an infinite patience - well, almost ;)

Zacharie Brodard, student at l'Ecole Polytechnique, and Hao Jiang, student at Télécom Bretagne, the interns extraordinaires, who performed the experimentation that is detailed in this thesis as part of their internship at the Cisco PIRL lab in Paris during Summer 2016;

Laurent Toutain, Géraldine Texier and Nicolas Montavont, from the IMT Atlantique, who directed this thesis, for their support and guidance all along the project;

John Apostolopoulos and Mark Webb, who encouraged me and granted the necessary time and means to achieve this work;

Latif Ladid, who introduced me to the world of academic conferences, and gave me the idea of doing this thesis.

Acknowledgments

This work took place at the Cisco Paris Innovation and Research Lab (PIRL) and funded in part by the Cisco-Polytechnique Chair.

It was also partly supported by the European Commission's Horizon 2020 Framework Programme, through the H2020 F-Interop and H2020 ARMOUR projects.

Simulations of this work were ran at IMT-Atlantique by Tadanori Matsui.

Summary

Based on time, resource reservation, and policy enforcement by distributed shapers, Deterministic Networking provides the capability to carry specified unicast or multicast data streams for real-time applications with extremely low data loss rates and bounded latency, so as to support time-sensitive and mission-critical applications on a converged enterprise infrastructure.

Deterministic Networking technology allows new Quality of Service (QoS) guarantees of ‘worst-case’ delivery. More precisely, the worst-case data loss and latency can be guaranteed in a consistent fashion as multiple services are deployed, augmenting the load of the network with no measurable impact on existing flows whatsoever.

There are several existing and emerging use-cases for Deterministic Networking. Audio/Video Bridging, (AVB) for the entertainment industry, Professional Media Networking, (PMN) targeted at the broadcast industry and many applications of the Internet of Things (IoT) in the context of the Industrial Internet are three significant categories. In one example, Industrial Internet potentially yields tens of billions of savings in various industries by optimizing industrial processes.

New capabilities are required to drive the connection of billions of things, and make available the vast amounts of data that are generated by IoT applications and do so in accordance with specific application performance requirements beyond our traditional Internet network technologies.

As of today, deterministic Operational Technology (OT) networks are purpose-built, mostly proprietary, typically using serial point-to-point wires, and operated as physically separate networks, which multiplies the complexity of the physical layout and the operational (OPEX) and capital (CAPEX) expenditures, while preventing the agile reuse of the compute and network resources.

Bringing determinism in Information Technology (IT) networks will enable the emulation of those legacy serial wires over IT fabrics and the convergence of mission-specific OT networks onto IP. The IT/OT convergence onto Deterministic Networks will in turn enable new process optimization by introducing IT capabilities, such as the Big Data and the network functions virtualization (NFV), improving OT processes while further reducing the associated OPEX.

Deterministic Networking Solutions and application use-cases require capabilities of the converged network that is beyond existing QOS mechanisms.

Key attributes of Deterministic Networking are:

- Time synchronization on all the nodes, often including source and destination

- The centralized computation of network-wide deterministic paths
- New traffic shapers within and at the edge to protect the network
- Hardware for scheduled access to the media.

This manuscript presents enhancements to existing low power wireless networks (LW-PAN) such as Zigbee, WirelessHARTTM and ISA100.11a to provide those new benefits to wireless OT networks.

Through multiple papers, standard contribution and Intellectual Property publication, the presented work pushes the limits of wireless industrial standards by providing:

1. Complex Track computation based on a novel ARC technology
2. Complex Track signaling and traceability, extending the IETF BIER-TE technology
3. Replication, Retry and Duplicate Elimination along the Track
4. Scheduled runtime enabling highly reliable delivery within bounded time
5. Mix of IPv6 best effort traffic and deterministic flows within a shared 6TiSCH mesh structure

The proposed technology is compatible with low power devices and constrained low-power lossy networks. It was implemented on open-source software and hardware, and evaluated against classical IEEE Std 802.15.4 and 802.15.4 TSCH radio meshes.

This manuscript presents and discusses the experimental results; the experiments show that the proposed technology can guarantee continuous high levels of timely delivery in the face of adverse events such as device loss and transient radio link down.

Résumé en Français

En s'appuyant sur une connaissance précise du temps, sur la réservation de ressources et l'application distribuée de règles d'admission strictes, un réseau déterministe permet de transporter des flux unicast et multicast pré-spécifiés avec un taux de perte extrêmement bas et une latence majorée, ce qui ouvre la voie au support d'applications critiques et/ou temps-réel sur une infrastructure de réseau d'entreprise convergée.

La technologie de réseau déterministe offre de nouvelles garanties de Qualité de Service (QoS) comme par exemple celle de majorer la latence d'un flux en toutes circonstances. Plus précisément, la latence comme le taux de perte peuvent être maintenus en-dessous des bornes garanties alors même que de nouveaux services sont déployés, augmentant la charge globale du réseau sans impact mesurable sur les flux existants.

Il existe de nombreux cas d'usage, soit déjà déployés, soit émergents, pour les réseaux déterministes. Audio/Video Bridging, (AVB) pour l'industrie audio-visuelle, Professional Media Networking (PMN) pour les studios de production professionnels, et de nombreuses variations de l'Internet des objets (IoT) dans le cadre de l'Internet Industriel, sont trois catégories significatives. L'Internet Industriel, en particulier, promet des économies en milliards de dollars dans de multiples industries par l'optimisation des procédures industrielles et des coûts d'installation et d'opération.

De nouvelles capacités sont nécessaires pour permettre l'interconnexion de milliards d'objets, et rendre accessibles les vastes quantités de données générées par les objets connectés, et ce, en accord avec les besoins de performance spécifiques aux diverses applications opérationnelles, qui vont typiquement bien au-delà de ceux de l'informatique de bureau et des technologies classiques de l'Internet.

De nos jours, la Technologie Opérationnelle (OT) s'appuie sur des réseaux déterministes mais conçus à façon, en général propriétaires, utilisant typiquement des liens série spécifiques, et opérés en isolation les uns des autres, ce qui multiplie la complexité physique et les coûts d'achat et de déploiement (CAPEX), ainsi que d'opération et maintenance (OPEX), et empêche l'utilisation agile des ressources.

En apportant le déterminisme dans les réseaux des Technologies de l'Information (IT), une nouvelle génération de réseaux commutés de l'IT va permettre l'émulation de ces liens série et la convergence de réseaux autrefois dédiés sur une infrastructure commune à base d'IP. En retour, la convergence de l'IT et de l'OT permettra de nouvelles optimisations industrielles, en introduisant des technologies héritées de l'IT, comme le BigData et la virtualisation des fonctions du réseaux (NFV), en support des opérations de l'OT, améliorant les rendements tout en apportant une réduction supplémentaire des coûts.

Les solutions de réseaux déterministes réclament des possibilités nouvelles de la part des équipements, possibilités qui vont bien au-delà de celles demandées pour les besoins classiques de la QoS. Les attributs-clé sont :

- la synchronisation précise de tous les nœuds, en incluant souvent la source et la destination des flux
- Le calcul centralisé de chemins de bout en bout à l'échelle du réseau entier
- De nouveaux filtres de mise en forme du trafic à l'intérieur comme à l'entrée du réseau afin de le protéger en tous points
- Des moyens matériels permettant l'accès au medium à des échéances précises.

Ce manuscrit rapporte des améliorations apportées aux techniques existantes des réseaux sans fils à basse puissance (LWPAN) comme Zigbee, WirelessHART™ et ISA100.11a, afin d'amener ces nouveaux bénéfices jusqu'aux réseaux opérationnels sans fil.

Au travers de multiples papiers, de contributions à des standards, et de publication de propriété industrielle, le travail présenté ici repousse les limites des réseaux industriels sans fils en offrant :

1. Le calcul centralisé de chemin complexes (Tracks) basé sur un technologie innovante appelée ARC
2. La signalisation de ces chemins complexes et la traçabilité des paquets par une extension de la technologie BIER-TE
3. Réplication, Renvoi et Elimination des doublons le long de ces chemins complexes
4. Un temps-réel basé sur un échancier qui assure un haut taux de délivrance et garantit une latence bornée
5. La capacité de transporter à la fois des flux déterministes et du trafic IPv6 à multiplexage statistique sur un maillage 6TiSCH partagé

La technologie proposée ici est compatible avec les objets à faible puissance et les réseaux à fortes pertes et contraintes. Elle a été implémentée en programme et sur du matériel open-source, et évaluée face à du IEEE Std 802.15.4 classique (Zigbee), ainsi que du 802.15.4 TSCH, utilisés en topologie maillée.

Ce manuscrit présente et discute les résultats expérimentaux obtenus en cette occasion. L'expérience menée montre que notre nouvelle proposition permet d'éviter les à-coups et de garantir des taux élevés de délivrance, même face à des évènements exceptionnels comme la perte d'un relais ou la dégradation temporaire d'un lien radio.

Introduction

It all started with point-to-point copper wires, transporting analog signals for short messages, then telephone and television, industrial measurements and commands, anything though initially not data. Digital data networks, and then packet networks, came last; but with the advent of determinism, the late comers now show the potential to federate all original forms of wired and wireless communication and lead to the final convergence of all communication networks.

A generic and cheap replacement to serial cables to provide connectivity to all sorts of devices, coupled with resource-sharing meshed networks, are now required to simplify the cabling and drive the costs down in many industries. Dumb as it may seem to emulate the legacy forms of serial communications, reproducing the various aspects of a point-to-point electric cable over a multi-hop packet network is actually the hardest thing to do. Yet, the need is becoming more and more pressing, as 1) managing all the existing sorts of cables and buses has become an increasingly costly hassle in many aspects of our lives and 2) point-to-point wires will not scale to serve the exploding needs of the Internet of Things.

At the Emerson Exchange 2008 conference in Washington, we presented a white paper on *Integrating an Industrial Wireless Sensor Network with your Switched Ethernet and IP Network*¹. The paper discussed how Wireless Sensor Networks (WSNs), which are in essence cheaper and faster to deploy than traditional wired field-buses, could leverage the entire network to connect the sensors to a centralized controlling application located afar on the carpeted floor, for Industrial supervisory control or logging; at the same time, the paper stressed issues that are raised when integrating a classical, often proprietary industrial automation network, with tight response time and availability constraints, into a wider IP network based on packet-switched and Internet technologies. With this and a collection²³⁴⁵ of other papers, the realization is now coming that with techniques such as flow isolation, high availability and a new generation of Quality of Service (QoS), the times of the convergence of these networks are finally approaching.

¹www.controlglobal.com/assets/14WPpdf/140303-Cisco-Emerson-WirelessNetworks.pdf

²www.atos.net/content/dam/global/ascent-whitepapers/ascent-whitepaper-the-convergence-of-it-and-operational-tech.pdf

³www.cisco.com/c/en/us/products/collateral/se/internet-of-everything/white-paper-c11-735380.pdf

⁴www.issource.com/wp-content/uploads/2015/11/111115statseeker-IT-OT-Convergence-White-Paper.pdf

⁵ics.sans.org/media/IT-OT-Convergence-NexDefense-Whitepaper.pdf

Converging Networks for the Industrial Internet

On Operational technology

"Operational technology (OT) is hardware and software that detects or causes a change through the direct monitoring and/or control of physical devices, processes and events in the enterprise"^a.

^a<http://www.gartner.com/it-glossary/operational-technology-ot>

Operational Technology (OT) refers to industrial networks, which focus on highly reliable, secure and Deterministic Networking. In OT environments, Deterministic Networks are characterized as providing a guaranteed bandwidth with extremely low packet loss rates, bounded latency, and low jitter⁶. OT networks are typically used for monitoring systems and supporting control loops, as well as movement detection systems for use in process control (i.e., continuous manufacturing) and factory automation (i.e., discrete manufacturing), and protection systems in the SmartGrid⁷.

Due to its different goals, OT has evolved in parallel but in a manner that is radically different from Information Technology/Information and Communications Technology (IT/ICT), which until now relied on selective queuing and discarding of IP packets to achieve end-to-end flow control over the Internet, and provided limited guarantees in terms of delivery and latency. For that lack of determinism and an associated lack of trust, IT and OT network have been maintained segregated.

Enabling the IT/OT convergence: The motivation behind General Electric's original vision of an Industrial Internet⁸, as depicted in figure 1, was that a single percentile point of operational optimization⁹ would save billions of dollars across many vertical industries. But reaching additional levels of optimization is not an easy task; it would require collecting and processing huge amounts of live measurements by widely distributed OT sensing and IT analytics capabilities¹⁰¹¹¹²¹³. These currently missing measurements comprise all sorts of diagnostics and sensor data that in many cases today are captured but not reported, either because the cost of wiring the devices would be prohibitive, or, if a device has connectivity to a control network, because the control network is kept isolated from the IT network and the wider Internet for security reasons.

⁶www.iab.org/wp-content/IAB-uploads/2013/01/tsn-nfinn-Deterministic-Networking-BOF-0914-v1.pdf

⁷cdn.iotwf.com/resources/10/How-the-Convergence-of-IT-and-OT-Enables-Smart-Grid-Development_2013.pdf

⁸www.ge.com/docs/chapters/Industrial_Internet.pdf

⁹www.geoilandgas.com/our-voice/power-one

¹⁰www.computerweekly.com/opinion/Big-data-to-unlock-value-from-the-Industrial-Internet-of-Things

¹¹sloanreview.mit.edu/case-study/ge-big-bet-on-data-and-analytics/

¹²www.ge.com/digital/blog/everything-you-need-know-about-industrial-Internet-things

¹³www.investors.com/news/ge-courts-silicon-valley-investors-for-digital-industrial-push/

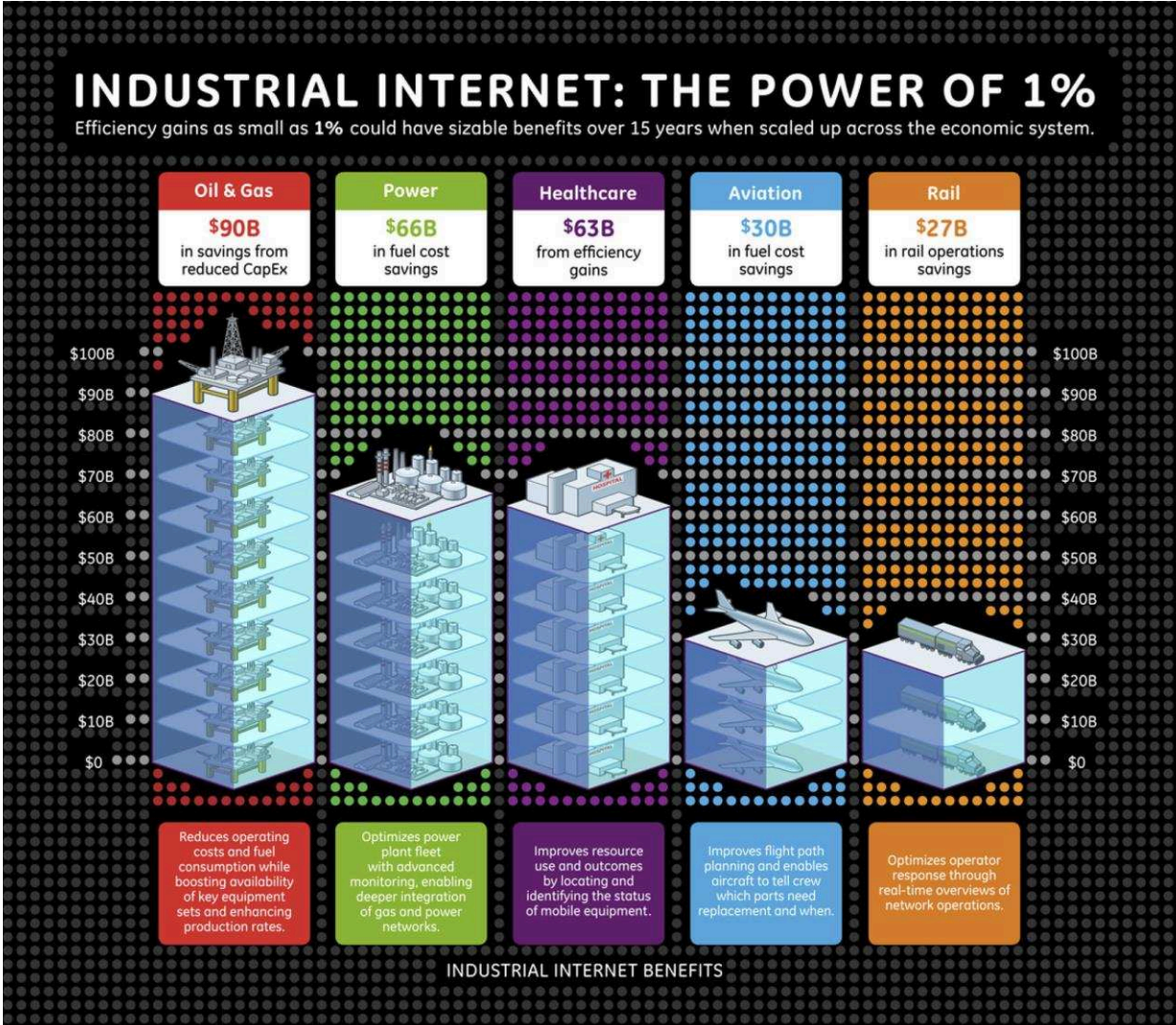


Figure 1 – The Power of One (Source GE estimates)

In order to avoid skyrocketing operational costs in maintaining highly trained teams for multiple solutions, the missing measurements must share the same infrastructure (network and management) as the deterministic OT flows. It results that the Industrial Internet vision can only be achieved by the convergence of IT and OT, whereby the network becomes capable of emulating the properties of deterministic OT circuits in the same fabric that serves traditional best effort IP applications.

New efforts at the IETF and the IEEE¹⁴ will enable traffic that requires bounded latency in a worst case scenario and is generally sensitive to packet loss and/or jitter, for application in a large variety of use cases [26]¹⁵ with a high degree of operational criticality. The upcoming protocols will support a mix of deterministic and classical best-effort traffic to be transported across Ethernet bridges and over IP networks, respectively. This *Deterministic Networking* capability is the key enabler for the convergence of IT and OT networks.

¹⁴www.odva.org/Portals/0/Library/Conference/2015_ODVA_Conference_Ditzel-Didier_TSN.pdf

¹⁵Grossman et al.: *Deterministic Networking Use Cases*

The case of Low-power Lossy Networks: The quality of transmission over IEEE Std 802.15.4 radios is affected by multiple elements, such as the relative location of objects in the environment and interferers of all kinds; these elements may be extremely difficult to control (e.g., radars) and may change brutally (e.g., a door opens); it results that a given channel cannot be expected to remain stable over a long period of time, and that some Channel Agility is required to guarantee a degree of service continuity over a long period of time.

When a radio transmission fails, adding diversity to the transmission characteristics improves the chances to avoid the cause of the failure and thus, those of a successful retransmission. A retransmission over the same channel adds only time diversity, and unless the cause of the loss is really transient, it is bound to fail for the same reason the original transmission did. But, as discussed by Srinivasan [77]¹⁶ in the case for 802.15.4 and 802.11b networks, other forms of diversity can help alleviate the issue.

Considering that those radios are highly sensitive to multipath fading [129]¹⁷, and that in turn multipath fading is highly sensitive to both location and frequency, it makes sense to add channel diversity to the retransmission. Trying this, it was found that switching channels dynamically based on a variation of the link quality also yields transient periods of instability (more on this in section 3.3 on page 20).

So in the years 2003-2007, providing deterministic services over IEEE Wireless Personal Area Networks (WPANs)¹⁸ such as 802.15.4 Low-Rate WPANs (LWPAN) appeared to be extremely challenging. Around that time, Pister developed a novel approach to combine frequency diversity and channel hopping based on time synchronization, and they introduced the quantum leap that brought determinism over low-power wireless, with the Time Synchronized Mesh Protocol (TSMP) [69]¹⁹.

TSMP introduced a scheduled combination of frequency and time diversity that defeats most interferences and can reach wire-equivalent reliability on battery-operated devices. TSMP opened the way to Time-Slotted Channel Hopping (TSCH) [128]²⁰, which was published as an amendment²¹ in 2012, and is now updated and retrofitted in the mainline IEEE Std 802.15.4 standard.

TSCH was immediately adopted by the Process Control industry, and two competing industrial WSN standards were developed in the following years, both leveraging this technique; first came WirelessHARTTM, which shipped with HART7²², and then

¹⁶Srinivasan et al.: “The β -factor: measuring wireless link burstiness”

¹⁷Watteyne et al.: “Mitigating Multipath Fading Through Channel Hopping in Wireless Sensor Networks”

¹⁸standards.ieee.org/about/get/802/802.15.html

¹⁹Pister and Doherty: “TSMP: Time synchronized mesh protocol”

²⁰Watteyne et al.: “Industrial ieee802.15.4e networks: Performance and trade-offs”

²¹standards.ieee.org/findstds/standard/802.15.4e-2012.html

²²en.hartcomm.org/hcp/tech/aboutprotocol/aboutprotocol_specs.html

ISA100.11a [27]²³, which brought a limited support of IPv6.

The above standards are optimized for application in industrial Process Control; they are not designed to access the global Internet. In order to realize the Industrial Internet vision for wireless, there was a need to develop a new series of open standards combining best-effort and time-sensitive IPv6 traffic over TSCH. This work has started in 2013 at the IETF 6TiSCH WG and is now delivering its first round of standards.

But wire-equivalent reliability is still not determinism. For safety and critical control applications, there is a need to guarantee the timely delivery of packets, even in the face of accidental situations such as the failure of a device or an obstacle moving in the way of the transmission. For this, additional forms of diversity, such as spatial diversity, and sometimes implementation and even technology diversities, are required.

Some Context About This Thesis

This manuscript discusses Deterministic Networks in the Industrial Internet with a focus on wireless industrial use cases, and proposes novel techniques to compute deterministic pathways called *Tracks* over an IEEE Std 802.15.4 TSCH-based mesh network.

This work started around 2007 when I contributed as an editor of the specification to the ISA100.11a standardization work. The ISA100.11a products²⁴ shipped in 2011; my main contribution was to enable at least a partial support of IPv6 in the standard.

In order to steer the adoption of IPv6 by the ISA100 committee, there was a need to update 6LoWPAN [55]²⁵ and provide a more efficient, stateful, compression; I jumped in and co-edited the *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks* [93]²⁶, also known as 6LoWPAN Header Compression (HC), and helped re-injecting this work into the ISA100.11a standard.

In parallel, I worked on the standardization of *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks* [133]²⁷, which I co-edited and largely influenced. RPL shipped in 2012 and has become the de-facto standard for routing in LLN.

In 2013, Thomas Watteyne and I lead the formation of the 6TiSCH Working Group (WG) at the IETF, which we have been co-chairing ever since. With its current charter²⁸, 6TiSCH provides a statistically multiplexed network layer based on IPv6 over the TSCH Medium Access Control (MAC) [5]²⁹ layer, leveraging RPL and 6LoWPAN to build energy-efficient mesh networks.

²³Hasegawa et al.: “Industrial wireless standardization—Scope and implementation of ISA SP100 standard”

²⁴www.honeywellprocess.com/library/marketing/whitepapers/ISA100WhitePaper.pdf

²⁵Montenegro et al.: *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*

²⁶Thubert and Hui: *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*

²⁷Winter et al.: *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*

²⁸datatracker.ietf.org/wg/6tisch/charter/

²⁹Bachir et al.: “MAC essentials for wireless sensor networks”

The work on deterministic IP is yet to be achieved, but it was clear in 2014 that the problem was more generic than just IPv6 over TSCH. So I went on and helped create the DetNet Working Group, which covers applications that are found in professional audio/video, industrial process control, safety, electric utilities, and many more [26]³⁰.

At DetNet, I contributed to the *Deterministic Networking Problem Statement* [22]³¹, which documents the issues that are in scope for the group, the *Deterministic Networking Architecture* [92]³², which supports the overall activity and presents in particular the concept of Packet Replication and Elimination, and the study on technical alternatives for the data plane [39]³³.

This manuscript brings it all together; it is organized as follows:

In part I, we study the characteristics of Deterministic Networking, how the concept can be achieved on wired infrastructures, and the direction taken by the standards bodies to enable deterministic properties on packet networks.

In part II, we focus on low-power wireless technology, and consider the specific techniques that are applied today to achieve some degree of determinism on this medium, in particular for use in industrial Process Control networks.

In part III, we walk through our contributions, namely:

- with *BIER-TE-based OAM, Replication and Elimination* [89]³⁴, we introduce a new set to Operations, Administration and Maintenance (OAM) tools [54]³⁵ that is based on Bit Indexed Explicit Replication (BIER) [132]³⁶ Traffic Engineering (BIER-TE) [16]³⁷; this novel approach enables new degrees of control and trace-ability for the Replication and Elimination function in a generic Deterministic Network; an interesting aspect of that innovative technique is the capability to control, on a per-packet basis, which adjacencies (e.g., links) participate to the redundancy scheme, and, in return, to observe which transmission failed for each individual packet; and from that unique capability we derive a protection control loop that limits the redundant transmissions in stable conditions and routes around breakages when they are detected;
- we apply that redundancy control mechanism to 6TiSCH networks; we propose different arrangements to improve the highly predictable properties of the TSCH MAC, leveraging Replication, Retries and Elimination along novel interleaved multi-path structures that we call *Tracks*, to enhance the art of industrial networks and

³⁰Grossman et al.: *Deterministic Networking Use Cases*

³¹Finn and Thubert: *Deterministic Networking Problem Statement*

³²Thubert and Finn: *Deterministic Networking Architecture*

³³Korhonen et al.: *DetNet Data Plane Protocol and Solution Alternatives*

³⁴Thubert, Brodard, and Jiang: *BIER-TE-based OAM, Replication and Elimination*

³⁵Mizrahi et al.: *An Overview of Operations, Administration, and Maintenance (OAM) Tools*

³⁶Wijnands et al.: *Multicast using Bit Index Explicit Replication*

³⁷Eckert et al.: *Traffic Engineering for Bit Index Explicit Replication BIER-TE*

achieve a higher level of reliability and manageability as required for mission-critical operations and safety-related applications;

- in addition, we propose a novel collaborative frame overhearing technique that is intended to balance for the extra transmissions that are associated with Replication and Elimination; an intuition of the proposed technology is that though a transmission from one node to another may have less chances to succeed over a radio than over a wired Ethernet, this might be compensated by the fact that, over the radio, other nodes than the intended recipient may have captured the transmission, and may be in a position to progress it towards the final destination. The collaboration can be applied to improve end to end transmissions as long as a packet that is transmitted can be uniquely identified, and as long as the extra listeners are well-chosen to provide spatial diversity in the rough direction to the destination.

Finally, in part IV, we demonstrate, through experimentation on actual 6TiSCH motes, how the combination of these technologies represents another step towards the ultimate goal of wire-equivalent performances, bringing increased reliability through spatial diversity, and yet controlling the associated energy spending by promiscuous overhearing and dynamic control of the redundancy protection through novel OAM techniques.

Contents

Summary

Résumé en Français

Introduction	i
Converging Networks for the Industrial Internet	ii
About This Thesis	v
I Deterministic Networking: the Vision	1
1 What is deterministic?	3
1.1 Common definitions	3
1.2 A game of convergence	4
1.3 The way to the IT/OT Convergence	4
1.4 A new context for the convergence	5
1.5 Determinism in a network	7
1.5.1 The train analogy (to control loop traffic)	7
1.5.2 The bus analogy (to deterministic circuit switching)	8
1.5.3 The vacation place analogy (to time-sharing)	9
1.5.4 The casino analogy (to statistical effects)	10
1.5.5 All in all	11
2 The precursors	13
2.1 On Fast Reroute	13
2.2 On SDN and Traffic Engineering	14
2.3 On BIER and BIER-TE	15
3 Enabling determinism in communication networks	17
3.1 Expected benefits in wired networks	17
3.2 Can we make Ethernet deterministic?	18
3.3 Can we make wireless deterministic?	20
4 The IETF DetNet architecture	25
4.1 Positioning of work	26
4.2 The architecture in a nutshell	27

4.3	Networking in DetNet	28
4.4	Controlling a Deterministic Network	30
4.4.1	Reporting the topology to the controller	30
4.4.2	Implementing the needs of the application	31
4.4.3	Automating the network operation	32
4.5	Limits and perspectives	33
II	Diving Deeper in Wireless	35
5	The art of low-power wireless sensor network	37
5.1	So how do we make wireless highly predictable?	37
5.2	WSNs in Industrial Process Control	39
5.3	6TiSCH and best effort IPv6	41
6	The vision of 6TiSCH centralized scheduling	45
6.1	PCE vs. 6TiSCH	45
6.2	6TiSCH base elements (time slots, schedule, chunks and bundles)	46
6.3	Applying DetNet to 6TiSCH	52
6.4	Forwarding along 6TiSCH Tracks	53
7	Enabling the convergence	55
III	Novel Approaches for Higher Predictability	57
8	Computing a complex Track: the ARC technology	59
8.1	Introducing the ARC technology	59
9	Centralized diverse paths computation with Replication and Elimination	63
9.1	Running an example of ARC Chain building	63
10	Computing and scheduling time slots	67
10.1	Forwarding rules	67
10.2	Detailed steps	67
10.3	Computation rules	70
11	Controlling Replication and Elimination using BIER headers	71
11.1	Technical solution description	71
11.2	Additional details	76

12 Compensating loss with overhearing: the Leapfrog Collaboration	77
12.1 Related work	77
12.2 Introducing the Leapfrog Collaboration	78
12.3 Running an example of Leapfrog Collaboration	79
12.4 Quick retrospective	82
12.5 Additional perspectives	83
IV Experimentations	85
13 Testing environment	89
13.1 The PIRL	89
13.2 Rover: the test tool	90
13.3 The test layout	92
14 Eliminating jitter with Replication and Elimination	93
14.1 Test description	93
14.2 Experimental results	95
14.3 Analysis	95
15 Saving energy with BIER-TE-based replication automation	97
15.1 Test description	97
15.2 Experimental results	98
15.3 Analysis	101
16 Saving time and energy with the Leapfrog Collaboration	103
16.1 Test description	103
16.2 Experimental results	105
16.3 Analysis	107
17 Conclusion on our experimentation	109
Conclusion and Perspectives	113
Contributions of this thesis	117

PART I

Deterministic Networking: the Vision

What is deterministic?

1.1 Common definitions

In Mathematics and Physics:

Wikipedia: "A deterministic system is a system in which no randomness is involved in the development of future states of the system. A deterministic model will thus always produce the same output from a given starting condition or initial state. "

In Philosophy:

Wikipedia: "A deterministic system is a conceptual model of the philosophical doctrine of determinism applied to a system for understanding everything that has and will occur in the system, based on the physical outcomes of causality. In a deterministic system, every action, or cause, produces a reaction, or effect, and every reaction, in turn, becomes the cause of subsequent reactions. The totality of these cascading events can theoretically show exactly how the system will exist at any moment in time."

A sense of what Deterministic Networking is has emerged as the capability to effectively emulate point-to-point wires on switched networks that were initially designed to serve IT and then IoT applications, such as Ethernet and IEEE Std 802.15.4, and the IETF is now extending the Pseudo-Wire emulation of Ethernet over IP to provide Deterministic Services over larger networks.

Various flavors of this concept can be found around, for instance with the concept of Network Slicing at the 3rd Generation Partnership Project (3GPP)¹. With these methods, an overlay, which is effectively a logical structure of meshed tunnels, would inherit physical properties such as a portion of the available buffers and bandwidth.

In this part, we study the characteristics of Deterministic Networking, how the concept can be achieved on wired infrastructures, and the direction taken by the standards bodies to enable deterministic properties on packet networks.

¹www.3gpp.org/

1.2 A game of convergence

At first, people realized that an on/off electrical signal could be perceived at very long distances and the telegraph was born. They realized that the voltage could be modulated and the analog audio signals, which still available today from devices equipped with those legacy jack and RCA connectors, joined the party.

In the last twenty years, voice, data and video networks have converged to digital over IP. Mail delivery has become quasi-immediate and volumes have multiplied; long distance voice is now mostly free and the videophone is finally a reality; TV is available on-demand and games became interactive and massively multi-player.

The convergence of highly heterogeneous networks to Information Technology (IT) resulted in significant drops in price for the end user while *adding distinct new value* to the related services.

Yet, and even though similar benefits can be envisioned when converging new applications over the Internet, there are still many disjoint branches in the networking family tree, many use cases where mission-specific applications continue to utilize dedicated point-to-point analog and digital technologies for their operations.

And even as a number of industrial protocols are now migrating to open standards such as Ethernet and IP, the typical OT network is kept isolated from the IT network and operated by a different crew of OT specialists, which yields double operational expenses compared to a converged network infrastructure and management.

1.3 The way to the IT/OT Convergence

Forty years ago, OT people found that information encoded as an analog modulation of current could be carried virtually instantly and with no loss over the distance; the basis of industrial control was laid out. Even today, most control signals rely on modulated current, typically between 4mA and 20mA, to report the variation of an observed phenomenon.

Then came digitization, which enabled to multiplex data with the control signal and manage the device, but at the same time introduced latency to industrial processes, the necessary delay to encode a series of bits on a link and transport them along, which in turn may limit the amount of transported information [8]².

The need to save cable and simplify wiring lead to *Time Division Multiplexing* (TDM) of signals from multiple devices over shared digital buses, each signal being granted access to the medium at a fixed period for a fixed duration; with TDM, came more latency, waiting for the next reserved access time.

²Borella et al.: *Methods for determining sendable information content based on a determined network latency*

Statistical multiplexing, with Ethernet and IP, was then introduced to achieve higher speeds at lower cost, and with it finally came jitter and congestion loss.

Some OT applications evolved to compensate for the transport degradation with recovery mechanisms and jitter absorption buffers, at the expense of yet some additional latency. But this did not seem acceptable for all, and in order to avoid those unwanted statistical effects, competing and not interoperable solutions appeared, driven by multiple standard defining organizations (SDOs), consortia and individual vendors.

Notable examples are, in the wired space ³⁴ [44]⁵, PROFINET Isochronous Real-Time (IRT), POWERLINK, SERCOS III, CC-Link IE, and to some degree Modbus TCP and EtherNet/IP which are now converged in the CIP specification, and in the wireless space, WirelessHART™, ISA100.11a, WIA-PA, WIA-FA, iPCF⁶, and WISA⁷.

In the real world today, operational signals are still massively carried as a simple analog modulation of the electrical current, over costly point-to-point wires. In order to replace those wires with a cheaper Ethernet-based switched fabric federating multiple access links, a limited subset of deployed control networks made all the steps towards digital statistical multiplexing; but even those cannot interoperate with IT technology, and the convergence is stalled. The main technical reason for not converging those networks derives from the limits of the protocols that sustain the Internet as we know it today. With multi-hop operations and statistical multiplexing, IP technologies lack “*determinism*”.

1.4 A new context for the convergence

In response to needs from different vertical OT industries, a new trend is emerging to bring determinism in IEEE and IETF technologies that were until then associated to IT:

- A first generation of Ethernet-based standards, called Audio/Video Bridging (AVB)⁸, was developed at the IEEE Std 802.1 for the Professional Media⁹ Networks.
- With TimeSlotted Channel Hopping (TSCH), IEEE Std 802.15.4 evolved into a highly predictable Medium Access Control (MAC)¹⁰ technology.
- The Time-Sensitive Networking (TSN)¹¹ Task Group (TG) at the IEEE and the Deterministic Networking (DetNet) Working Group (WG) at the IETF are now generalizing those methods to transport deterministic flows across Ethernet bridges and over IP networks, respectively.

³www.ethercat.org/2011/italy/download/04_Industrial_Ethernet_Technologies_IT_1102.pdf

⁴ftp://ftp.ni.com/pub/events/nits/presentations/2008/ethernet_protocols.pdf

⁵Lin and Pearson: “An inside look at industrial Ethernet communication protocols”

⁶www.iebmedia.com/index.php?id=8096&parentid=63&themeid=275&showdetail=true

⁷www.millennialnet.com/MillennialNet/media/Resources_Media/WhitePapers/WhitePaper_IntroductiontoWISA_V2.pdf

⁸www.ieee802.org/1/files/public/docs2009/avb-rboatright-p1722-explained-0903.pdf

⁹www.avnu.org/wp-content/uploads/2014/05/AVnu-Pro__White-Paper.pdf

¹⁰www.electronicweekly.com/news/products/analog/low-power-radio-standard-simplifies-sensor-networks-2012-10/

¹¹www.ieee802.org/1/pages/tsn.html

In figure 1.1, the trend to bringing determinism in IT protocols is positioned relative to other trends that lead to it, and ones that in turn it enables.

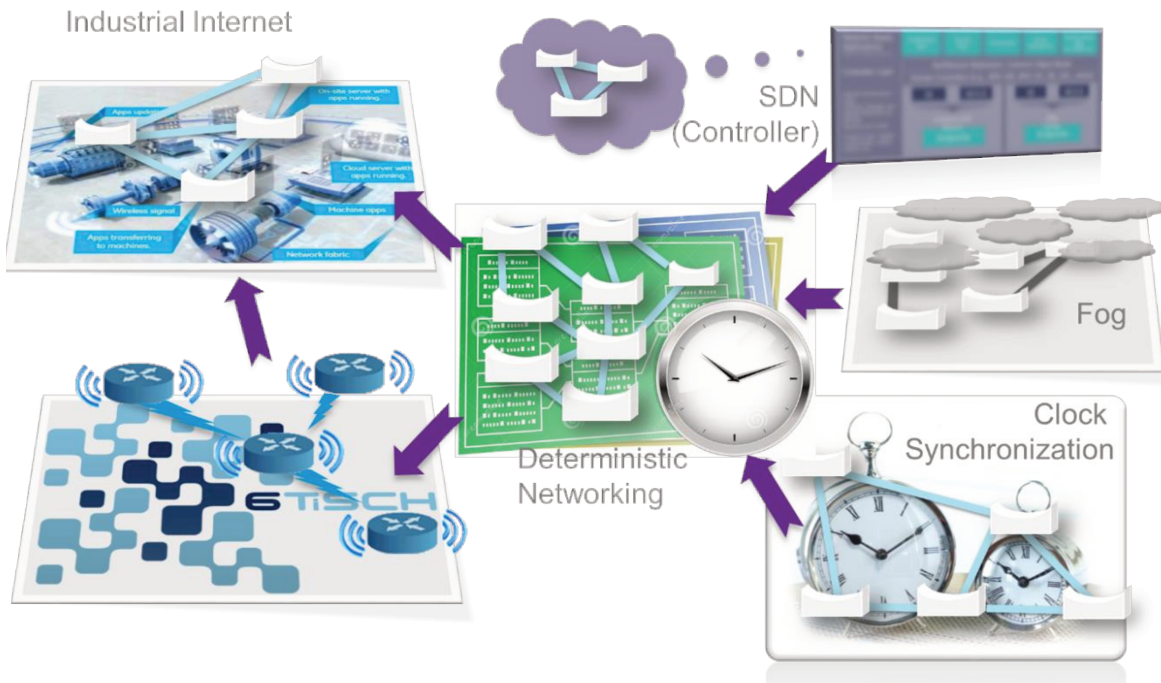


Figure 1.1 – Enabling trends

On the one hand, determinism builds on recent trends of *Clock Synchronization* [78]¹² to control time-shared network resources, and on *Software Defined Networking* (SDN) [70]^{13,14} for the centralized visibility on network resources and capability to achieve a global optimization of the network. On the other hand, Deterministic Networking is a key enabler for a series of new trends that involve the replacement of a serial cable or a multidrop synchronous bus over a switched fabric.

Current trends include the migration to IP of *Professional Media Networks* (PMN) such as found in production studios and concert arenas, the replacement of dedicated automotive buses such as the Controller Area Network (CAN bus) and FlexRay¹⁵ by cheap 100MBps Ethernet over twisted pairs in the new models of cars, and the *Compute platform Disaggregation* whereby the internal bus inside a server such as a Peripheral Component Interconnect (PCI) bus is replaced by an Ethernet switched fabric.

And among those trends, we focus on the *Industrial Internet*; in that case, deterministic signals that are used to control industrial processes must be isolated, and yet share the physical media with other information such as diagnostics and sensory data that are used to monitor and improve the processes.

¹²Sundaraman, Buy, and Kshemkalyani: “Clock synchronization for wireless sensor networks: a survey”

¹³Pujolle: “SDN (Software-Defined Networking)”

¹⁴www.opennetworking.org/sdn-resources/sdn-definition

¹⁵papers.sae.org/2011-01-0453/

1.5 Determinism in a network

Determinism in a network brings the guarantee that a particular information is transported across the network in a tight window of time, and that a periodic process will be repeated identically every time. Determinism is a required property in the power grid, to ensure that high tension lines breakers can be activated within milliseconds, in public transportation to make sure that automated vehicles are operated safely for their passengers, and in industrial automation for control loops.

To further delineate the concepts behind Deterministic Networking, let us use some analogies.

1.5.1 The train analogy (to control loop traffic)

The analog of a congestion loss in the railway system, that is the collision of two trains using the same rails at the same time, is avoided in the real world by fully scheduled operations that repeat, day after day, the predetermined schema that is a train schedule.

Figure 1.2 illustrates the sheer complexity of computing an optimal schedule for multiple trains that will share a same infrastructure, with the goal to minimize the end-to-end time and the constraint to avoid collisions by ensuring that at most one train is present on any section of the rail at any point of time.

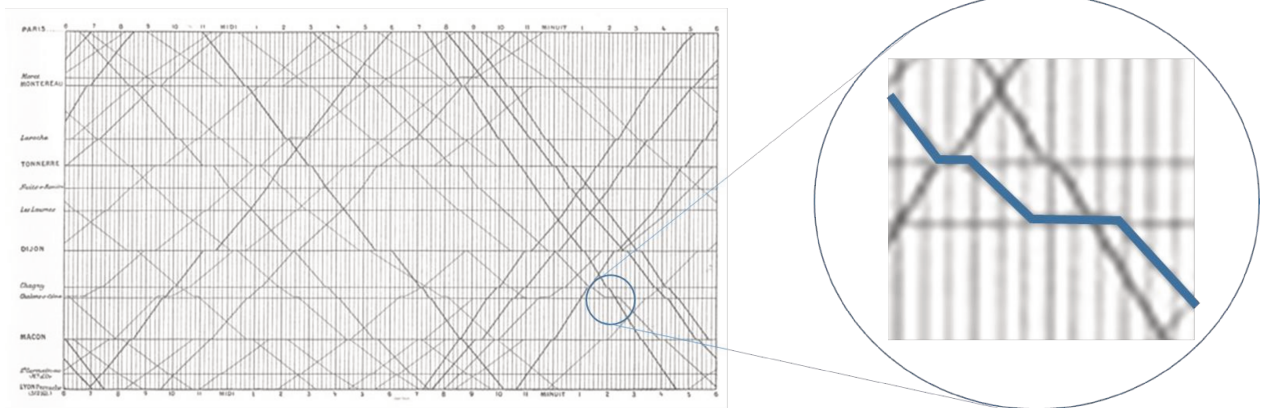


Figure 1.2 – A Railway System, illustrating the complexity of a global schedule

The flat portions of the schedule that are enlarged on the right of figure 1.2 (colored in blue) illustrate that, in order to avoid collisions, a train may be temporarily stopped on a garage way, until the pre-determined time when it can safely leave the station. The overall travel experience may not be as fast as a *hot potato* [80]¹⁶ model whereby the train is expelled as soon as the way is open, but scheduling avoids queues and results in a much more predictable experience for the passengers.

¹⁶Teixeira et al.: “Dynamics of hot-potato routing in IP networks”

Likewise, a deterministic flow may experience short buffering in intermediate hops to guarantee that there are never two deterministic packets scheduled for transmission at overlapping times, in other words, that the link is always available at the scheduled time of transmission; because of the extra fixed latency that is added at each hop, most deterministic flows will incur a higher latency than with best effort hot potato forwarding.

1.5.2 The bus analogy (to deterministic circuit switching)

The goal of the reserved bus lanes in a city is to avoid that the bus is delayed by traffic jams and guarantee a good and repeatable experience for the passengers.



Figure 1.3 – The bus analogy

Say a bus takes 30 minutes to travel from a given bus stop down the street from a user's home, all the way to his office. If there is a bus every 10 minutes, then the transit time for a particular user will always be somewhere between 30 and 40 minutes, depending on how lucky he is on a particular day, and this, regardless of the traffic in the car lanes. This user will certainly not use every bus that passes by, but he knows that there will be one soon enough, and that, 40 minutes later at worst, he will be at his desk.

Because there cannot be as many lines as end-to-end travels, our user may have to hop from one bus to another one or more times before reaching his final destination; switching between lines represents some additional transport time for him, and yields a huge reduction of number of lines and thus of operational complexity, resulting in increased profitability for the bus company.

A bus line is analog to the reserved circuits that can be scheduled in a packet-switched Deterministic Network; regardless of the load, they guarantee a periodic transmission opportunity with a bounded latency for asynchronous commands and alerts. In that analogy, the ultimate realization of a Deterministic Network appears to be a perfect emulation of the good old serial cable over a packet-switched network, transporting the exact same application, at a much lower cost than a full mesh of wires.

1.5.3 The vacation place analogy (to time-sharing)

For those of us who do not have 52 weeks of vacation per year, clever marketing has invented this concept of time sharing acquisition whereby an individual owns a flat in a nice vacation resort for just one week a year. That week, the individual will use the flat as if it was always his, will not find unexpected people in his bed at night, and, provided that all owners are decent enough, he will have no clue whatsoever of how many other owners also live in that flat at other times along the year.

We saw in section 1.5.1 that determinism requires that packets from a particular flow may need to be held in an intermediate node until the scheduled time when the free access to the transmission medium is guaranteed. This means that enough buffers must be available during that period to hold the packet; by analogy to the vacation place, there can be no unwanted occupant in the master bed.

The object of Deterministic Networking, which is to remove chance from the picture, requires tying physical resources to the protected flows. With scheduled operation, a same resource may be affected to different flows at different times, and the duration of this affectation, thus the number of deterministic flows that can make use of a same resource, is directly affected by the precision of the shared sense of time in the network.

It results that the reservation system that locks physical resource to well-dimensioned and identified flows must be aware of the device capabilities such as clock precision and forwarding latency, as well as the amount of buffers, timers and queues that are available to control in that particular device.

With appropriate shaping, a deterministic flow is fully isolated from any influence from other traffic, with no leak, no loss, and no latency whatsoever that could be imputed to other flows, whatever the load on the network. This isolation goes beyond that provided by a Virtual LAN (VLAN)¹⁷ and its Layer 3 equivalent, the Virtual Routing and Forwarding (VRF)¹⁸; those only prevent leak and eavesdropping, but cannot protect against congestion loss and latency induced by some other traffic that happens to share the same physical resources.

In other words, Deterministic Networking brings a new level of isolation and guarantees that are critical to converge OT control flows onto a shared IT infrastructure spanning the campus or the factory. To realize this to its full extension, a strict policing and shaping must be performed that filters out misbehaving devices, whether it is an external attack or a failing network node that may, for instance, repeat the last frame forever at line speed. In turn, the isolation brings a new form of security whereby attempts to influence a flow by injecting another in the same physical infrastructure becomes totally inefficient.

¹⁷www.cisco.com/c/en/us/td/docs/net_mgmt/ciscoverworks_lan_management_solution/4-0/user/guide/configuration_management/cmug/configvlan.pdf

¹⁸www.cisco.com/c/en/us/td/docs/net_mgmt/ciscoverworks_lan_management_solution/4-0/user/guide/configuration_management/cmug/vrf.pdf

1.5.4 The casino analogy (to statistical effects)

The law of large numbers says that long term, the casino wins. Yet, though statistics play against him, there is no bound to the potential gains that a player may achieve. By analogy, statistical multiplexing and arguably buffer bloat yields an unbounded latency.

In wireless, Carrier Sense Multiple Access (CSMA) with collision avoidance (CA) detects a collision and applies a randomized back off to reduce the chances of additional collisions. Though the probability decreases exponentially, there is a theoretical

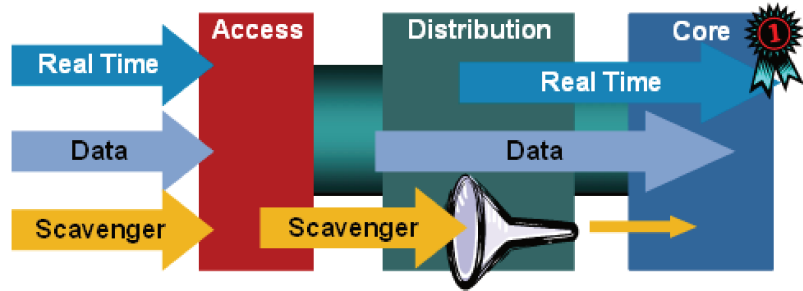


Figure 1.4 – traditional Quality of Service (Source Cisco Systems)

possibility that a frame collides again for a unbounded number of times [136]¹⁹.

Similarly, the hot potato [80]²⁰ forwarding operation in a switch or a router entails a statistical chance of too many flows outgoing a certain link, queues forming and eventually packets being dropped. The imperfect experience of voice over IP generally results from this sort of effect, in particular over Wi-Fi where the data rate may vary considerably and Call Admission Control (CAC) [19]²¹ is very complex to achieve. The only protection in classical routers and switches is to drastically under-utilize the medium, with a very limited amount of critical traffic of the highest Quality of Service (QoS) ²² .

Ever increasing the bandwidth can make it so that statistical effects such as congestion loss and out-of-bound latency become negligible; and in addition, appropriate QoS settings enable to push the unwanted statistical effects onto vulgar packets, and protect relatively low rates of critical flows. This has been the recurrent answer for statistically-multiplexed packet-switched networks against the better groomed behavior of transports designed for circuit mode communications, such as Synchronous Optical Networking (SONET) and its Synchronous Digital Hierarchy (SDH) variant, or T1 Time-Division Multiplexed (T1-TDM) and its Integrated Services Digital Network (ISDN) variant.

So far, this has been a winning strategy for most use-cases, but it falls short when the needs of the critical flows approach the effective throughput of the wires, e.g., for large time-sensitive flows of uncompressed video, where tens of Gbps must be delivered at a precise time across a switched fabric in professional studio.

¹⁹Ziouva and Antonakopoulos: “CSMA/CA performance under high traffic conditions: throughput and delay analysis”

²⁰Teixeira et al.: “Dynamics of hot-potato routing in IP networks”

²¹Fang and Zhang: “Call admission control schemes and performance analysis in wireless mobile networks”

²²www.mixedcriticalityforum.org/about/timing-analysis/

This is why until now, the extreme OT uses cases such as industrial motion control, which operates control loops at several hundred times per second, had to resort to specialized standards or proprietary designs, as detailed in section 1.3 on page 4.

In contrast, scheduled operations can utilize the medium for critical flows up to high loads, limited only by the precision of the clocks and the compute capability of the central controller. Voice calls and industrial control loops are like trains, the traffic is predetermined and with adequate dimensioning and scheduling, the network can be programmed to ensure a seamless and repeatable experience. This requires new shapers, a fine sense of time, and a better control of individual physical resources such as buffers in memory.

A deterministic flow must traverse the network in the same predictable fashion every time, regardless of the load of the network. There can be no observable influence whatsoever from any other flows. A Deterministic Network may be primarily loaded with deterministic flows and still maintain its predictable properties unmodified for each of them. This goes beyond the capabilities of the current QoS-based Internet technologies, where an increased load in one flow generates delays and losses, which are rapidly observable, on adjacent flows.

1.5.5 All in all

Transporting OT traffic requires a network that is reliable and jitter-free, and can be trusted to transport periodic and asynchronous commands with a bounded latency. An ever increasing bandwidth is not always a valid response in the face of high loads of time sensitive flows found in OT applications. This strategy was not accepted by the industrial community, which developed its own adaptations of Ethernet and IP to meet its specific needs. As opposed to QoS-based networks, a Deterministic Network must always retain its properties, even under high loads of critical flows.

To achieve this, it is necessary to schedule the timely operation of the network with the granularity of numerous tiny physical resources. While scheduling enables to provide the required guarantees, it may also yield an additional latency for the transported flows. A precise synchronization of the network must be obtained, so the required resources can be reserved for the individual packets within that particular flow at their precise time of arrival. Over the distance, tiny flows must be aggregated into larger ones that can be processed as a single entity.

With Deterministic Networks, the worst-case data loss and latency can be guaranteed in a consistent fashion when multiple services are deployed on a common converged network infrastructure. A deterministic flow is completely isolated in its own time-shared set of physical resources and cannot be influenced by any other traffic in any observable fashion; from the perspective of the application, a deterministic end-to-end connection appears as a dedicated point-to-point wire.

As Deterministic Networking capabilities are deployed, wired and wireless links that are today limited to provide Internet connectivity can be used to replace any of the cables that we see in day-to-day applications, RCA connectors on the stereo gear, Serial Digital Interface (SDI) in professional video and AES3 in audio, HDMI and S/PDIF, their matches in the consumer domain, RS232 serial cables, fancy bus connectors in cars, 2 and 4-wires cables in industrial control networks, all of them. Wouldn't life be drastically simpler with just one connector?

A shared sense of time enables to transport -in fact to recreate- a clock signal that is used to synchronize both ends of an emulated serial cable.

And the perfect isolation between flows that is obtained to guarantee the required latency is also an improvement on the security side, and a factor of trust in the convergence for OT people.

Awareness of Deterministic Networking is now spreading²³, and the new technology is generally perceived as an evolution of quality of service to bring a new level of guarantees for network-based services. True as it is, what Deterministic also and mostly brings to Networking is more revolutionary; it is the capability, for the first time, to carry any signal that was ever transported across a point-to-point wire, for any form of application that men ever devised, over multi-hop packet networks.

²³www.embedded.com/electronics-blogs/cole-bin/4406659/Deterministic-networking--from-niches-to-the-mainstream-

The precursors

2.1 On Fast Reroute

Use-cases where two non-congruent paths are set up to ensure either a full redundancy, or at least a rapid fail-over, by selecting at the ingress between two pre-computed paths, abound in the art of networking. An example of such, the Linear Protection [62]¹ of the Multi-Protocol Label Switching (MPLS) Transport Profile (MPLS-TP) [6]² [23]³ uses specific OAM frames [2]⁴ to monitor the liveness of the routes and make rapid fail-over decisions.

To avoid the complexity of setting up non-congruent paths, and in order to react even quicker from where the problem actually happens, the IETF has devised the concept of fast-reroute [74]⁵, which means rerouting around a failure from anywhere inside the network.

The IETF has proposed two approaches for IP fast-reroute, the original IP Loop-Free Alternates (LFA) [135]⁶ [10]⁷, which attempts to find a path around a failing node or a failing link, with a variable coverage, and a new technique for IP Label Distribution Protocol (LDP) using Maximally Redundant Trees (MRT) [17]⁸, which draws a pair of trees in any bi-connected topology, each tree connecting every node and both rooted at the destination, in such a fashion that a breakage only blocks one of the trees, so that the other tree offers a path to destination.

One key benefit that is found in both techniques is to be compatible with, and actually leverage, an existing OSPF-based infrastructure; and one limit to both approaches is the lack of control on the generated alternate paths, in particular with respect to their relative costs. The detour may be long and expensive, in particular with MRT which computes global trees.

¹Osborne et al.: *MPLS Transport Profile (MPLS-TP) Linear Protection*

²Beller and Sperber: “MPLS-TP-The New Technology for Packet Transport Networks.”

³Frost, Bryant, and Bocci: *MPLS Transport Profile Data Plane Architecture*

⁴Allan and Busi: *Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks*

⁵Shand and Bryant: *IP Fast Reroute Framework*

⁶Zinin: *Basic Specification for IP Fast Reroute: Loop-Free Alternates*

⁷Bryant et al.: *Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)*

⁸Envedi, Atlas, and Bowers: *An Architecture for IPLDP Fast Reroute Using Maximally Redundant Trees (MRT-FRR)*

2.2 On SDN and Traffic Engineering

Software Defined Networking (SDN)⁹ [59]¹⁰ promotes a model that simplifies the network operation by automating the deployment of network resources from a centralized controller; at the extreme, the networking gear becomes a basic subservient to the master controller in practice, though, a distributed routing plane may still be associated with the SDN operation so as to reach the controller and handle the general purpose bulk of the traffic. Though still highly overhyped¹¹, the momentum started with Openflow [51]¹² is getting traction in campuses and cloud data centers through various vendor incarnations such as the Network Element Drivers¹³ of Cisco’s Network Services Orchestrator and Arista’s Directflow¹⁴.

The art of Traffic Engineering (TE) at the IETF leverages routing protocol extensions to link-state Interior Gateway Protocols (IGP) such as OSPF [134]¹⁵ [4]¹⁶ and IS-IS [7]¹⁷, as well as extensions to BGP [63]¹⁸, to report the topological information to a central routing component, which is implemented by a Path Computation Element (PCE) [20]¹⁹. Relying on that topological information learned, the PCE computes diverse paths and assign flows to those paths. But this technique yields a lot of human intervention and does not yet support deterministic properties.

More closely related to OT applications of IoT and low-power radios, *Industrial Routing Requirements in Low-Power and Lossy Networks* [68]²⁰ discusses the need for the LLN routing protocol to compute multiple forwarding solutions. The centralized approach in TE is echoed by the best practice found in the art of industrial and vehicular networks, operating Time-Triggered Ethernet [38]²¹, WirelessHARTTM and ISA100.11a. The limit of the model is probably the scalability of the controller, considering that the optimization of a Time-Triggered schedule is an NP-complete [13]²² [61]²³ problem with a complexity that grows rapidly with the number of engineered flows.

⁹www.hpc.mil/dren/sdn

¹⁰Nunes et al.: “A survey of software-defined networking: Past, present, and future of programmable networks”

¹¹www.itential.com/blog/overhyped-sdn/

¹²McKeown et al.: “OpenFlow: enabling innovation in campus networks”

¹³www.cisco.com/c/en/us/products/collateral/cloud-systems-management/network-services-orchestrator/datasheet-c78-734669.html

¹⁴www.arista.com/en/um-eos-4172f/eos-section-45-1-introduction#ww1148008

¹⁵Yeung, Katz, and Kompella: *Traffic Engineering (TE) Extensions to OSPF Version 2*

¹⁶Atlas et al.: *OSPF Traffic Engineering (TE) Metric Extensions*

¹⁷Berger, Bartlett, and Harrison: *IPv6 Traffic Engineering in IS-IS*

¹⁸Ould-Brahim and Fedyk: *BGP Traffic Engineering Attribute*

¹⁹Farrel and Ash: *A Path Computation Element (PCE)-Based Architecture*

²⁰Phinney, Thubert, and Dwars: *Industrial Routing Requirements in Low-Power and Lossy Networks*

²¹Kopetz et al.: “The time-triggered ethernet (TTE) design”

²²Craciunas and Oliver: “SMT-based task-and network-level static schedule generation for time-triggered networked systems”

²³Oliver, Craciunas, and Stöger: “Analysis of deterministic ethernet scheduling for the industrial internet of things”

2.3 On BIER and BIER-TE

Bit Indexed Explicit Replication (BIER) [132]²⁴ is a network plane replication technique that was initially intended as a new method for multicast distribution. In a nutshell, a BIER header is added to IPv6 or MPLS packets; this new header includes a bitmap called the *BitString* that explicitly signals the destinations that are intended for a particular packet.

This means that:

1. the source is aware of the individual destinations and
2. the BIER control plane is a simple extension of the unicast routing as opposed to a dedicated multicast data plane, which represents a considerable reduction in OPEX.

The simplicity of the BIER technology makes it very versatile as a network plane signaling protocol. Already, a new Traffic Engineering (BIER-TE) [16]²⁵ variation is emerging that uses bits to signal segments along a TE path.

While the more classical BIER is mainly a multicast technology that typically leverages a unicast distributed control plane through IGP extensions, BIER-TE is mainly a unicast forwarding plane technology that leverages a central computation to setup a path, compute segments and install the mapping in the intermediate nodes.

From the BIER-TE architecture, the key differences between BIER-TE and BIER are as follows:

- BIER-TE replaces in-network autonomous path calculation by explicit paths calculated offpath by the BIER-TE controller host.
- In BIER-TE every *BitPosition* of the BitString of a BIER-TE packet indicates one or more adjacencies - instead of a BFER as in BIER.
- BIER-TE in each Bit-Forwarding Router (BFR) has no routing table but only a BIER-TE Forwarding Table (BIFT) indexed by SI:BitPosition and populated with only those adjacencies to which the BFR should replicate packets to.

The generic abstraction of an adjacency can be implemented as a link, a tunnel or a route along a path segment. With Segment Routing (SR) [21]²⁶ a segment can be signaled as an MPLS label, or an IPv6 Routing Header. A segment may be loosely or strictly source routed, depending on the need for full non-congruence and the confidence that loose routing may still achieve that need.

²⁴Wijnands et al.: *Multicast using Bit Index Explicit Replication*

²⁵Eckert et al.: *Traffic Engineering for Bit Index Explicit Replication BIER-TE*

²⁶Filsfils et al.: *Segment Routing Architecture*

Enabling determinism in communication networks

3.1 Expected benefits in wired networks

A perfectly Deterministic Network would ensure that every packet would reach its destination, and would always do so in due time. In an imperfect world, Deterministic Networking must nearly eliminate packet loss, with the associated goal to guarantee a worst case latency for a packet, and all this whatever the overall network conditions are, effectively emulating the case of a point to point serial cable.

The main cause of data loss in a wired switch fabric is a statistical effect called congestion loss, whereby, at a particular moment, multiple flows entering a switch converge to a same outgoing port, in volumes that exceed the capacity of that port to output the traffic. It results that some of the traffic must be dropped, and one of the desired effects of Quality of Service (QoS) is to intelligently select the frames to be discarded.

Making networks more deterministic eliminates this statistical effect by maintaining at all time the amount of critical packets within the physical capabilities of the hardware. This can be achieved by the use of time-shared resources (bandwidth and buffers) per circuit, and/or by shaping and/or scheduling the packets at every hop.

Equipment failure, such as a switch rebooting, a broken interface adapter, or an unplugged physical wire, is a secondary source of data loss. When a breakage occurs, multiple packets are lost in a row before the flows are rerouted or the system may recover. This is not acceptable for critical applications such as related to safety. A typical process control loop will tolerate an occasional packet loss, but a loss of several packets in a row will cause an emergency stop (typically after 4 packets lost, within a period of 1 second).

Making networks more deterministic improves the resiliency against breakages and statistical transmission loss such as due to cosmic particles, typically by adding redundancy in the network path.

Finally, since the operation of a Deterministic Network rely on precisely applying a tight schedule, and the worst case time of delivery must be guaranteed, there must be a shared sense of time throughout the network that can be exposed to and leveraged by other applications.

3.2 Can we make Ethernet deterministic?

On precise time: A synchronization of clocks in the order of the microsecond can be achieved by a software implementation using the Precision Time Protocol (PTP) [52]¹ IEEE Std 1588, or one of its derived profiles, such as IEEE Std 802.1AS. With hardware assistance, this can be brought down to tens of nanoseconds for use in Smart-grid [34]² and even down to nanoseconds for military [46]³ and 4/5G so-called fronthaul and crosshaul [60]⁴ applications.

With a precise shared sense of time, the switches can guarantee the exact forwarding time at each hop for TSN applications, and in-time delivery at the egress for AVB applications. Precise time is leveraged by Time-Triggered Ethernet (TTE) [38]⁵ to control the exact time of transmission; with TTE, the sender and the listener can agree that a certain packet belongs to a certain deterministic flow just because it is transmitted at a certain precise time, without the need to tag the packet. Keeping the Ethernet frames unmodified enables inter-working between TTE and classical Ethernet switches.

On timely transmission: Another key aspect to provide deterministic latency guarantees is to ensure that the medium is free and can be accessed with no wait when the time comes for sending a deterministic packet.

In the case of full-duplex switched Ethernet, a sender owns the transmit medium, so there is no need to defend against an interfering transmission from other parties; a collision may be avoided either by preventing transmission ahead of the scheduled time by a guard time that is more than the transmission time of the largest frame, or by suspending a frame being transmitted to free the medium for the deterministic packet, which is the solution that the IEEE as selected with 802.1Qbu Frame Preemption.

By construction of the schedule, only one frame may be programmed during any particular window of time sized for one frame of maximum size, and there can never be a contention between deterministic frames.

If a non-deterministic frame is being transmitted at the precise time scheduled for a deterministic frame, its transmission is interrupted and a CRC is attached to validate the partial transmission. The deterministic packet can then be transmitted in time and in full, and then the non-deterministic transmission can resume starting at the offset where the transmission was interrupted, yielding no access latency for the deterministic frame.

¹Meier and Weibel: “IEEE 1588 applied in the environment of high availability LANs”

²Ingram, Schaub, and Campbell: “Use of precision time protocol to synchronize sampled-value process buses”

³MacKay: *Applications and Opportunities for the IEEE 1588 Standard in Military Applications*

⁴Oliva et al.: “An overview of the CPRI specification and its application to C-RAN-based LTE scenarios”

⁵Kopetz et al.: “The time-triggered ethernet (TTE) design”

On redundancy: Fast recovery techniques, in particular leveraging rings, already exist in the IT world, in particular with the ITU-T G.8032 Ethernet Ring Protection Switching (ERPS)⁶ and Cisco’s Resilient Ethernet Protocol (REP)⁷.

But a real Deterministic Network must bring resiliency against physical and logical failures and guarantee the continuity of operations in all conditions; this requires the physical redundancy of each involved piece of networking equipment⁸, and the capability to compute non-congruent paths between source and destination and leverage Frame Replication and Elimination techniques.

Kirmann first introduced the Parallel Redundancy Protocol [36]⁹ (PRP) and then the High-availability Seamless Redundancy [37]¹⁰ (HSR) for that purpose. The International Electrotechnical Commission (IEC) further standardized those methods [33]¹¹ for both the industrial [3]¹² and Smartgrid [79]¹³ applications, and is now evolving the IEC 61850 [29]¹⁴ standard for substation communication networks to include them [25]¹⁵.

On reliability: Finally, critical applications such as Pro Audio¹⁶ demand a packet delivery ratio (PDR) that orders of magnitude better than the capabilities of a simple chain of switches along a path. Additional diversity, such as replication and elimination of a packet over non-congruent paths, enables to reach the required figures.

The maths is simple: If the probability of loss along a path is P , then, if total diversity and independence can be achieved between two parallel non-congruent paths, then the probability of loss with Replication and Elimination over those two paths becomes P^2 . With a loss ratio of an Ethernet fabric that is typically in the order of 10^{-5} , sending redundant copies over two fully diverse paths yields a 10^{-10} loss ratio, which enough to burn a blu-ray disk without a scratch in an AVB studio.

Applying this logic to a highly lossy multi-hop media such as a wireless link with, where a worse than 10% packet error rate (PER) without retries is not uncommon [76]¹⁷, a redundant path still yields at best a PER of 1%, far from the 10^{-5} wire equivalence that industrial networks are after.

⁶www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-G.8032-201508-I!!PDF-E

⁷www.cisco.com/c/en/us/support/docs/lan-switching/ethernet/116384-technote-rep-00.html

⁸www.processonline.com.au/content/industrial-networks-buses/article/redundancy-in-industrial-networks-part-2-71248

⁹Kirmann, Hansson, and Muri: “IEC 62439 PRP: Bumpless recovery for highly available, hard real-time industrial networks”

¹⁰Kirmann et al.: “HSR: Zero recovery time and low-cost redundancy for Industrial Ethernet (High availability seamless redundancy, IEC 62439-3)”

¹¹IEC: *Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)*

¹²Araujo et al.: “High availability automation networks: PRP and HSR ring implementations”

¹³Tan and Luan: “IEC 61850 based substation automation system architecture design”

¹⁴IEC: *IEC 61850 Communication networks and systems for power utility automation - ALL PARTS*

¹⁵Goraj and Harada: “Migration paths for IEC 61850 substation communication networks towards superb redundancy based on hybrid PRP and HSR topologies”

¹⁶avnu.org/proav/

¹⁷Shu et al.: “Packet loss analysis of the IEEE 802.15. 4 MAC without acknowledgements”

3.3 Can we make wireless deterministic?

On transmission reliability: The radio medium, in particular in the industrial, scientific, and medical (ISM) bands that are shared not only between data networks but also with all sorts of interferers such as microwave ovens and radars, is orders of magnitude less reliable than classical wired networks such as Ethernet over copper or fiber; providing deterministic services over wireless appears extremely challenging.

Co-channel interferences are not the only possible causes for a frame loss; for instance, physical obstacles may happen to move in the way of the transmission and block the communication. Multi-Path Fading, which is due to multiple reflections that may reinforce or cancel one another out a few centimeters away, is a major cause of transmission errors; it does not affect all channels in a same fashion, and which channels are impacted is highly sensitive to the relative position of the sender and the receiver, and to their environment.

In short, the quality of a given channel is affected by multiple parameters that may vary brutally, and IEEE Std 802.15.4 transmissions over a fixed channel cannot be expected to remain stable over a long period of time. It results that basic 802.15.4 implementations which operate on a fixed channel will often suffer from intermittent delivery issues, and can only apply to low-end applications for which a consistent reliability is not a concern.

Channel Agility was added to improve the availability of the radio links by permanently sensing the channel, looking for an increase in Bit Error Rate (BER), and switching channel when loss becomes too high. This technique represents a clear improvement from the art of fixed channel, but it only fixes an error condition reactively, and a period of lossy transmissions is experienced before a new channel is selected. This is well suited for applications such as AMI/AMR metering, which do not require a wire equivalent reliability, but, at the same time, this does not provide the deterministic guarantees that industrial applications require.

A predictive technique that would enable to switch to a better channel before the problem even occurs would be ideal but early attempts [28]¹⁸ did not make it to mainstream. Effectively, one can leverage transmission statistics, observe activity on other channels and remember channels with a bad transmission record, so as to black list them. This may be efficient to protect against a stable Wi-Fi co-channel interferer, but there is no way to know if this is effectively the case, and if the problem will last and for how long. The physical phenomenons that are the most common cause of channel degradation, moving an object or starting a process, cannot be fully predicted by the radio device in most practical situations, and making Channel Agility proactive has appeared so far to be a red herring, yielding more complexity than actual benefits.

Time-Slotted Channel Hopping (TSCH) [128]¹⁹ is the best of breed with its simple

¹⁸Hwang: “Energy efficient channel agility utilizing dynamic multi-channel CCA for ZigBee RF4CE”

¹⁹Watteyne et al.: “Industrial ieee802.15.4e networks: Performance and trade-offs”

per-packet channel rotation called channel hopping. TSCH brings Time (by repeating failed transmissions) and Frequency (by switching channel) diversities.

But unless we have infinite time to attempt an infinite number of retries, there is always a limit to the reliability of a system. In short, deterministic delivery within a constrained time cannot be guaranteed over a wireless medium. A realistic goal is to optimize the delivery ratio, and this is best achieved by combining all possible forms of diversity.

On deterministic channel access: The other major difficulty with wireless links is that the medium is shared, meaning that not only this node, but other nodes, may be transmitting at the time intended for a deterministic packet, which would have to wait a variable time till the end of that current transmission. Then, with CSMA/CA [12]²⁰ mechanisms, it would have to introduce a random delay to obtain access to the channel, and then, in case of a collision, wait for an additional exponential backoff time.

In wireless, a technique like the IEEE Std 802.1Qbu - Frame Preemption - is not really workable, because apart from new specific attempts for full duplex-radios, there is no way to interrupt a remote talker that is blinded by its own signal.

Depending on the specific radio technology, packets may be blocked at the MAC or PHY layer and then sent together in a single transmission opportunity; data rates may vary dramatically with the distance and the environment; it results that the exact duration of the current transmission can be very hard to predict, and that a guard time that would guarantee that the medium is empty at scheduled time (see section 3.2 on page 18) on a wireless medium would necessarily be very long and highly wasteful in terms of bandwidth to cope with highly variable transmission duration.

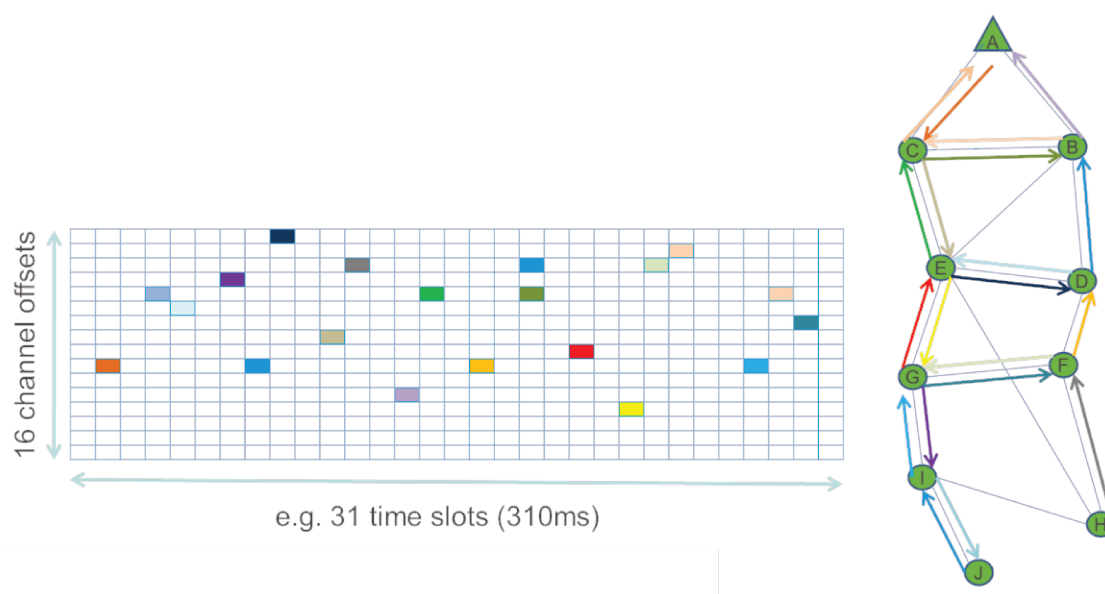


Figure 3.1 – A TSCH schedule (Source T. Watteyne)

²⁰Colvin: “CSMA with collision avoidance”

The only way to approach determinism in wireless transmissions –most people will prefer to use terms such as “highly predictable” instead of deterministic- is to schedule all the transmissions, and that is what TSCH and industrial WSNs do. A schedule such as illustrated with colored codes in figure 3.1 on the preceding page controls at which time and on which channel a frame is forwarded between which pair of nodes.

Scheduling implies that all nodes control the precise time of emission, which in turn requires a shared and precise sense of time. It is, for instance, possible to achieve in the order of tens of microseconds clock synchronization over an 802.15.4 link.

With fully scheduled operations, it is now possible to guarantee the time of delivery for those packets that make it to destination, in a deterministic fashion.

Additional Benefits from scheduling in wireless: In addition to the benefits listed in section 3.1 on page 17, scheduling provides specific value to the wireless medium. On the one hand, scheduling reduces transmission losses and with TSCH and its industrial derivatives, a wire-equivalent loss ratio of 10^{-5} can be obtained: to achieve this, routes are computed so as to enable at least two forwarding solutions for every node, which ensures spatial diversity, whereas retries over the TSCH MAC provide both time and frequency diversity and effectively combats co-channel interference as well as multipath fading.

On the other hand, scheduling optimizes the bandwidth usage: compared to CSMA/CA operations, there is no blank related to IFS and exponential back-off in scheduled operations, though some minimal Clear Channel Assessment may be needed to comply with the local regulations such as ETSI 300-328. And because TSCH time slots provide a full time sharing operation, there is no limit to the ratio of guaranteed critical traffic.

Additionally, the capability to schedule the operation at the scale of a Service Provider (SP) network enables an operator to guarantee Service Level Agreements (SLA) [24]²¹ across customers who use the shared infrastructure.

Finally, scheduling plays a critical role to save energy; in IoT, energy is the foremost concern, and synchronizing sender and listener enables to maintain them in deep sleep at all times when there is no scheduled transmission; this optimizes sleeping periods by avoiding idle listening and long preambles; TSCH enables battery operated nodes that actually forward packets in a mesh topology for multiple years [128]²²

In a nutshell: While scheduling transmissions can guarantee the time of delivery, it is impossible, in the ISM band, to keep all possible interferers at bay. Co-channel interference, as well as the self-inflicted Multi-Path fading, which is due to echoes of the transmission, are unavoidable. So there is no way to guarantee the delivery of all frames.

It takes different mitigation techniques to avoid the different issues that affect wireless

²¹Gaillard: “Managing IoT networks using service level agreements (SLA)”

²²Watteyne et al.: “Industrial IEEE802.15.4e networks: Performance and trade-offs”

transmissions. To combat them all, all possible forms of diversity should be leveraged, in the spatial domain by routing over multipath, in the temporal domain by retrying transmissions or sending copies over parallel paths at distinct times, and in the frequency domain with frequency hopping (within frames) or channel hopping (between frames).

Several times along the way to the destination, the work presented in this manuscript replicates and then eliminates the copies of a packet that are forwarded along parallel disjoint paths, hopping between frequencies with each transmission so that a copy along one path does not interfere with a copy along the other.

The IETF DetNet architecture

On Deterministic Networking

"Deterministic Networking (DetNet) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low data loss rates and bounded latency. Techniques used include: 1) reserving data plane resources for individual (or aggregated) DetNet flows in some or all of the intermediate nodes (e.g. bridges or routers) along the path of the flow; 2) providing explicit routes for DetNet flows that do not rapidly change with the network topology; and 3) distributing data from DetNet flow packets over time and/or space to ensure delivery of each packet's data' in spite of the loss of a path. The capabilities can be managed by configuration, or by manual or automatic network management."

In parallel to industrial applications, the need for determinism in professional and home audio / video markets drove the formation of the Audio/Video Bridging (AVB) standards efforts in IEEE at 802.1. With the demand for connectivity and multimedia in transportation, AVB is being evaluated for application in vehicle head units, rear seat entertainment modules, amplifiers, camera modules, and engine control systems. Automotive AVB networks share the OT requirements for Deterministic Networks characteristics.

This wider application scope for Deterministic Networks has led to the 802.1 AVB Task Group becoming the Time-Sensitive Networking (TSN) Task Group (TG), covering industrial and vehicular applications, and then to the formation of the DetNet¹ WG at the IETF. With DetNet, the networks in consideration can extend beyond the LAN boundaries and secure deterministic forwarding over a composite Layer 2 / Layer 3 network will eventually be enabled.

¹datatracker.ietf.org/wg/detnet/charter/

4.1 Positioning of work

Deterministic Networking refers to the allocation of pre-determined physical resources in the network (queues, buffers, transmission medium) for well-characterized flows that are known a-priori, in order to avoid the statistical effects that lead to a poor bandwidth utilization, uncontrolled jitter, and congestion loss. Bandwidth that is not actually used by deterministic flows is available for use by other traffic.

In more details, a path is nailed down with a particular set of resources at particular times, and the forwarding behavior ensures that the right packets are forwarded at the right time to make use of these resources.

As illustrated in figure 4.1, this requires capabilities for:

- Hop-by-hop synchronization, which enables to restore an apparent end-to-end clock signal that can be associated with the timely distribution of data streams, which in turn can be decoded to reproduce precisely phased analog signals;
- Time-based resource reservation, coupled with enforcement methods, which eliminates collision loss and provides the capability to transport unicast or multicast data streams of predefined characteristics such as data rate and bounded latency.



Figure 4.1 – DetNet Components (Source S. Dhesikan)

Work has started at the DetNet WG to enable the establishment and maintenance of deterministic paths over Layer 2 Bridged and Layer 3 routed segments, by defining a common representation (data models) of the physical resources and network topology, and standardizing the protocol flows and interface mappings that are used to set up the flows.

Work will also be needed for management, with in particular network control frames that are carried along a deterministic path to assess its health and performance - aka Operations, Administration, and Maintenance (OAM)

frames, but also in-band means to trace a packet from the last replication point from which it may have strayed into the network.

4.2 The architecture in a nutshell

The *Deterministic Networking Architecture* [92]² applies a centralized approach for a limited amount of deterministic flows, which share the network with a more classical distributed path computation and statistical multiplexing operation for traffic of lesser criticality and requirements.

In that approach, diverse applications can push their requirements over a Northbound Interface to a centralized controller, which translates these requirements in term of complex routes with replication and elimination capabilities, and pushes the result onto the network over a Southbound Interface, including precise operating schedules and time-based resource reservations.

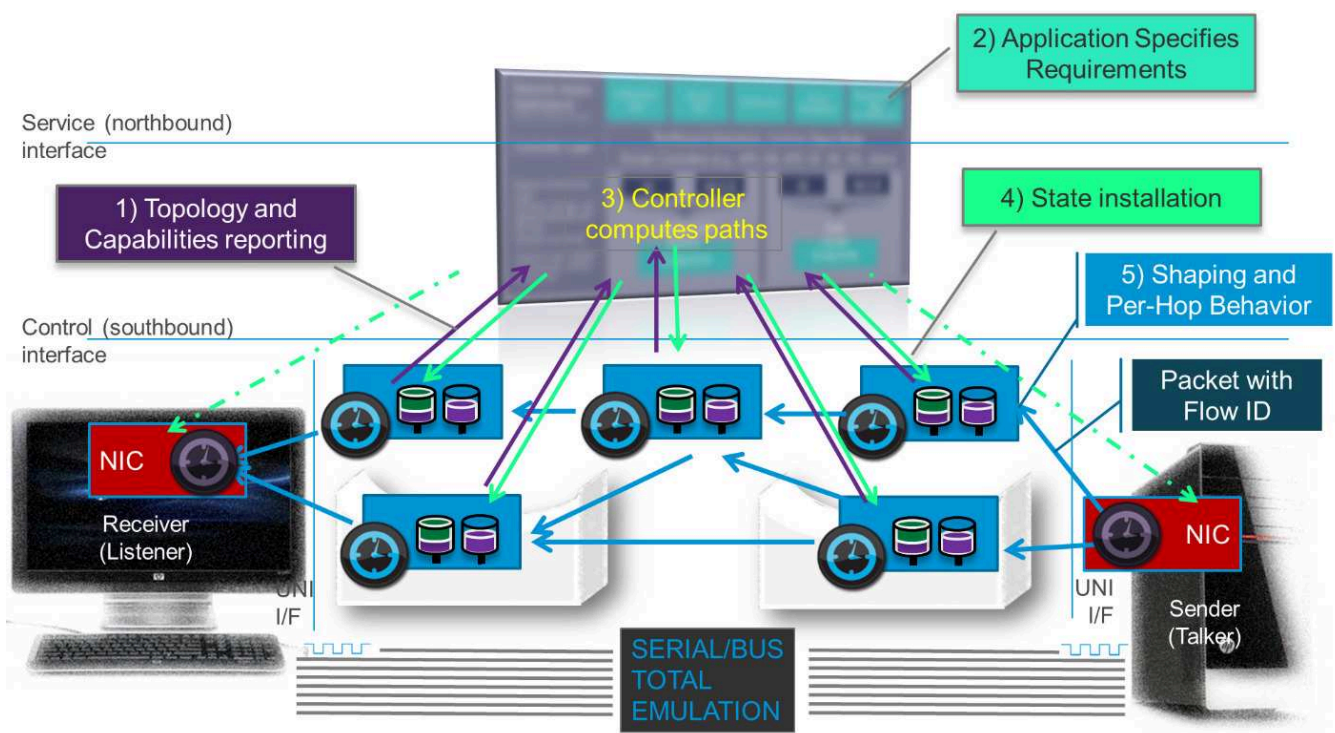


Figure 4.2 – The DetNet Architecture

Figure 4.2 illustrates the steps that take place in order to setup a deterministic path (in blue), and the interfaces (in purple and green) and data models that DetNet should standardize. Following the numbers on that figure, the main steps are:

1. the definition of data models to report the topology and the devices capabilities to the controller which is aware of the application requirements and can perform
2. the computation of a path that matches those needs;
3. the protocol elements to request a path set up for a given flow and configure
4. the Network Interface Card (NIC) in the end nodes, and the time-shared reservation of physical resources in the network nodes along the end-to-end path, and
5. the forwarding behavior for each flow.

²Thubert and Finn: *Deterministic Networking Architecture*

In short, the SDN model comprises three layers, the application which we will not cover in this manuscript, the network which loses some of its distributed features, and the controller that seats in between and implements the needs of the former using the capabilities of the latter.

The next sections explore the DetNet abstractions for the network and the controller.

4.3 Networking in DetNet

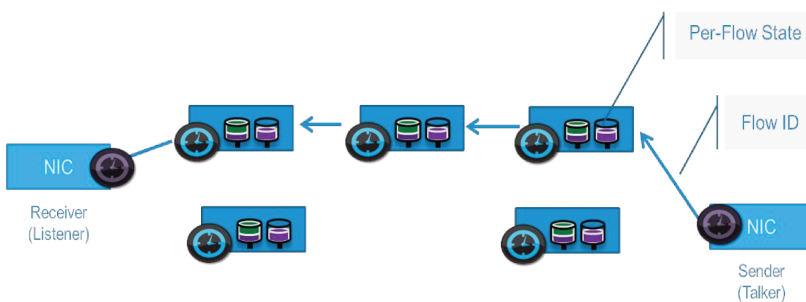


Figure 4.3 – Single Path

The art of L2 with AVB, or of L3 with IntServ (RSVP), is already capable to establish a serial path in a distributed fashion as depicted in figure 4.3, with end-to-end latency guarantees [75]³ at the expense of wasted bandwidth.

So arguably, a distributed architecture could be used by DetNet as well.

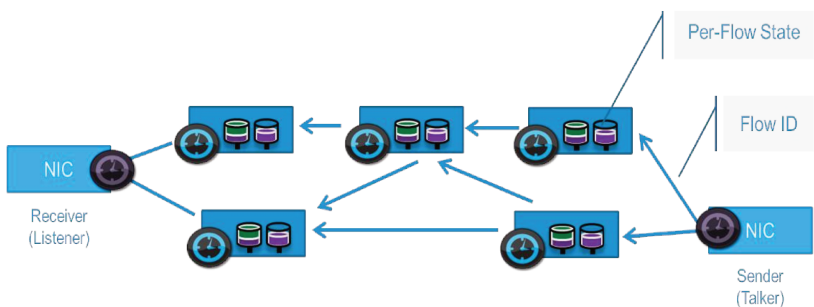


Figure 4.4 – Complex Track

But to reach a higher reliability, DetNet adds Replication, Retry (wireless) and Elimination of individual packets to the problem, making the path more complex and more global, as illustrated in figure 4.4. The complexity of the DetNet path

makes solving the problem of minimizing the overall latency and the amount of resources even more difficult. It was determined from the art of Traffic Engineering that optimally distributing a number of known flows across a shared network is an NP-Complete problem and that the most advanced optimization will be obtained by leveraging a central PCE with a global view of the network.

This is why, as opposed to IntServ, the DetNet architecture relies on a controller with a total view of the system capabilities and application requirements to optimize globally the allocation of resources and install the related state transactionally.

³Shenker: *Specification of Guaranteed Quality of Service*

Figure 4.5 represents a classical view of a network as perceived by a networking engineer, with switches that forward frames at Layer 2, routers that route packets at Layer 3, over both wired and wireless connectivity, with all sort of medium between routers. From the perspective of that engineer, the elements of concern are means of logical connectivity, e.g., Virtual LANs, and communication layers, L3 routing vs. L2 switching.

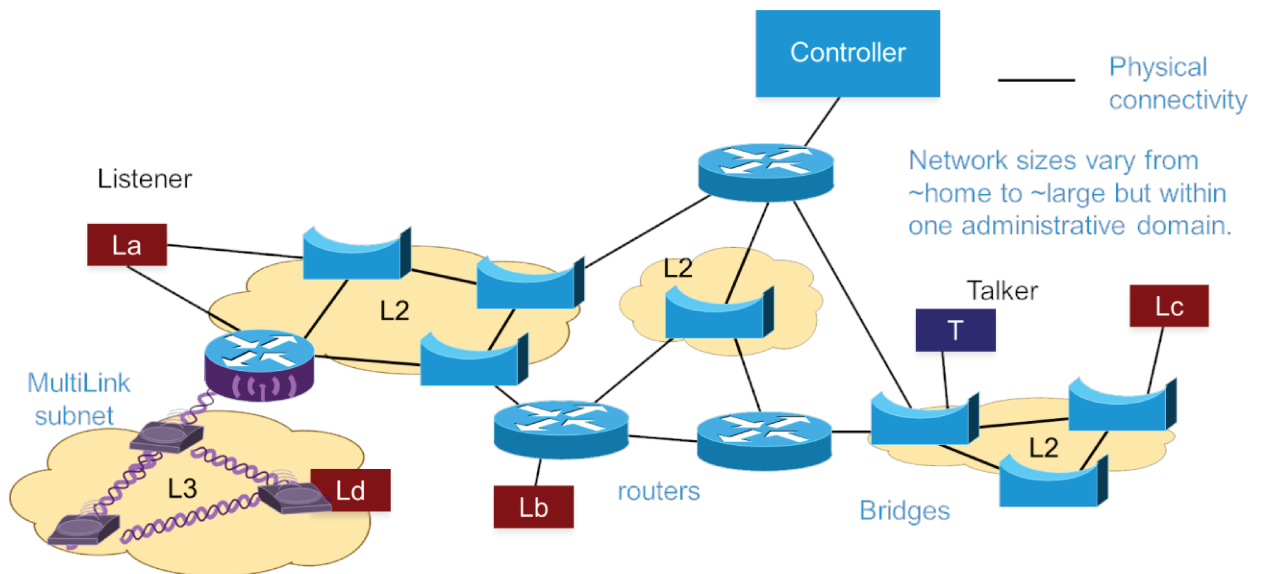


Figure 4.5 – Perspective of a networking engineer (Source N. Finn)

In contrast, figure 4.6 represents the view of a Deterministic Networking Controller.

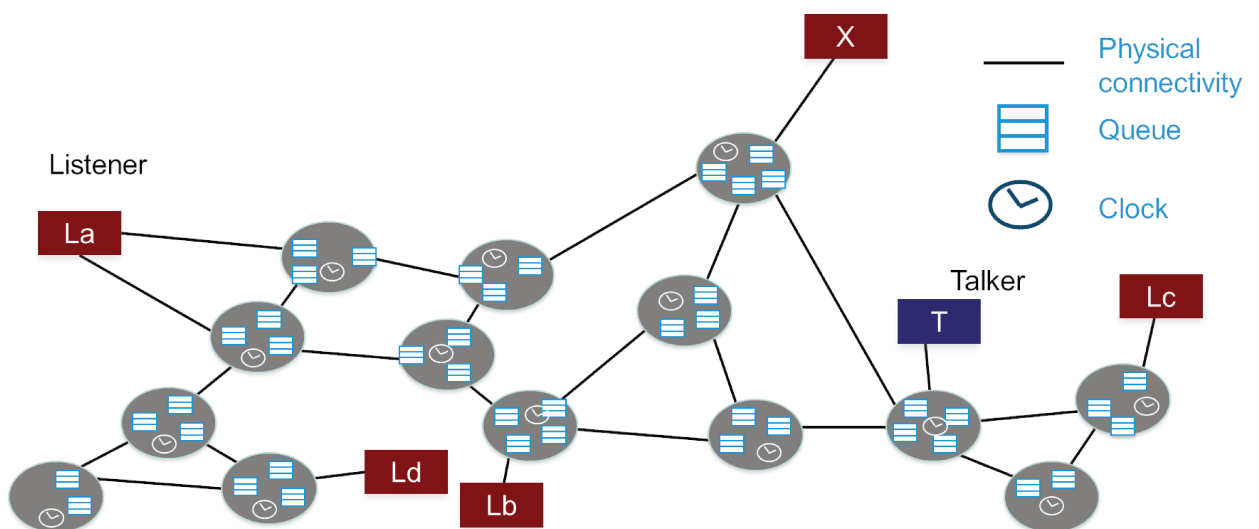


Figure 4.6 – Perspective of the controller (Source N. Finn)

From the Deterministic Networking perspective, the network is a collection of interconnected boxes and wires, regardless of whether they are implementing as Layer 2 or Layer 3 boxes. What counts is the physical capabilities in those devices to store reliably some amount of packets and resend them at a precise time along a scheduled path.

4.4 Controlling a Deterministic Network

The previous section presented the simplified view of the network from the DetNet controller perspective. This section elaborates on the steps that are presented in figure 4.2 on page 27, of reporting the topology to the controller, implementing the needs of the application in the network, and automating the network operation.

4.4.1 Reporting the topology to the controller

In order to compute a deterministic path, the controller needs to learn not only the connectivity between the networking pieces of equipment, but also their capabilities down to amounts of buffers and timers that can be reserved, which types of shapers are available on the device, and with which precision the device is synchronized to the rest of the network.

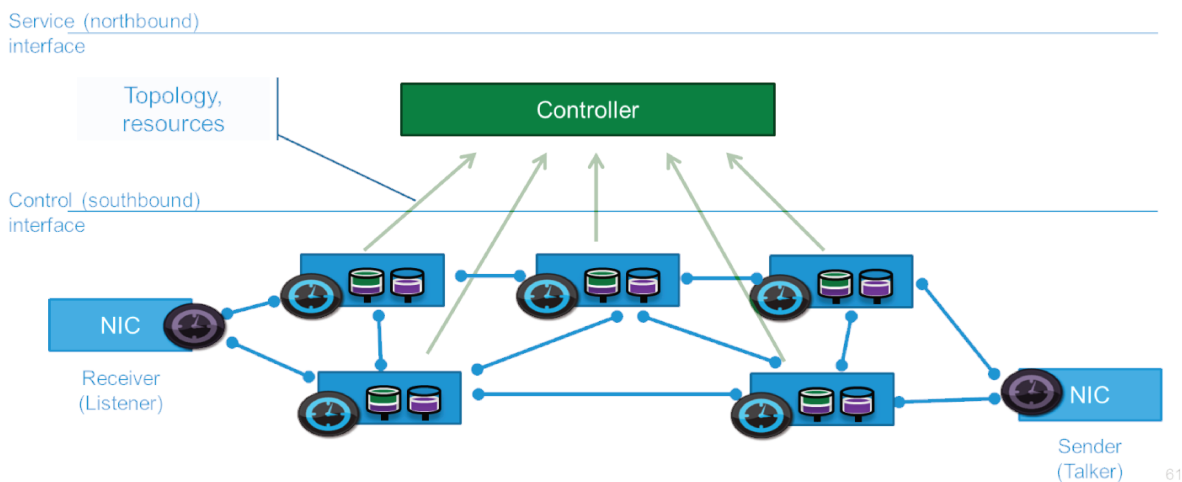


Figure 4.7 – Reporting capabilities and topology to the Controller

The most probable outcome is that DetNet will inherit from work done in the art of TE at the Traffic Engineering Architecture and Signaling (TEAS)⁴ working group to report the topology and the device capabilities to the controller over a control interface called the Southbound Interface. This means that DetNet needs to extend the data models that exist in TE to report, in particular, the physical characteristic that are relevant to the deterministic path computation, which in turn may differ from one medium to another; for instance, IEEE Std 802.15.4 TSCH has a concept of time slots associated to frequency hopping that does not exist in Ethernet.

⁴datatracker.ietf.org/wg/teas/charter/

4.4.2 Implementing the needs of the application

The creation or the modification of a flow is triggered when an application, residing on an end-device, a management console like an Industrial Human Machine Interface (HMI), or a Broadcast Control System in Professional Media Networking (PMN), needs a deterministic communication channel to, say, transport the content of a movie to a bluray disk burner for mastering.

A time sensitive application residing on the end-device (a mobile phone) or some third party hardware (a central HMI) provides a Traffic Specification (TSPEC) that reflects the transport requirements from its perspective, expressed as volumes of data, sensitivity to jitter, loss, and latency. The application makes its request to the controller over a service interface called the Northbound Interface, which is defined by the Open Networking Foundation (ONF)⁵, providing the TSPEC as parameter.

The controller translates those requirements in terms that are actionable by the network devices. In many cases, the controller will schedule a complex path across the network, which can be seen as an evolved form of a circuit.

The relevant state is pushed in the end and intermediate systems, indicating how to identify a particular flow and the operation to be executed on that flow, like the precise time of forwarding and the behavior of the shaper, e.g., credit-based vs. time-triggered.

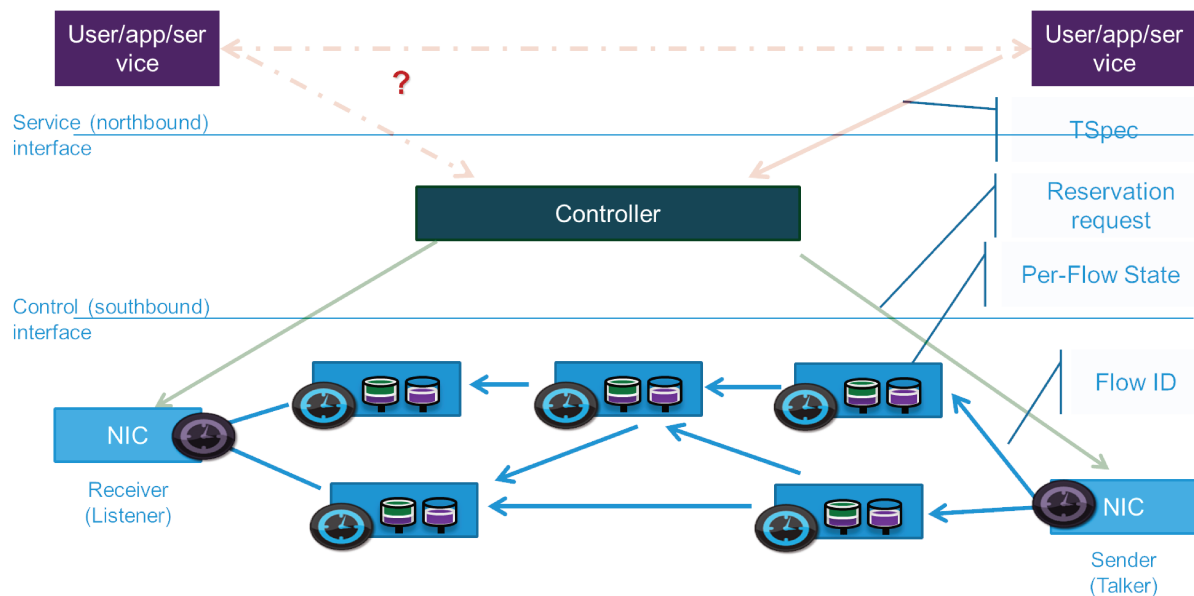


Figure 4.8 – Setting up a new flow

As shown in 4.8, DetNet needs to define the new data models that are required to set up a path that supports new features such as packet Replication, Retry and Elimination, and the definition of tagging elements (i.e., Flow ID, and packet marking) to be used to identify the flow as it is being forwarded along the complex path.

⁵www.sdxcentral.com/sdn/definitions/who-is-open-networking-foundation-onf/

4.4.3 Automating the network operation

The SDN model used in DetNet enables a degree of automation that help reduce the operational expenses associated to the deterministic network. To make that happen, the controller must participate to a permanent control loop that observes the network to validate whether it still matches user needs, triggers actions when that is not the case, performs an update computation (that’s the PCE’s piece), and then reprograms the network, which is done by the combination of a driver for the particular network element and a protocol such as the PCE Protocol (PCEP), a REST or a Command Line Interface.

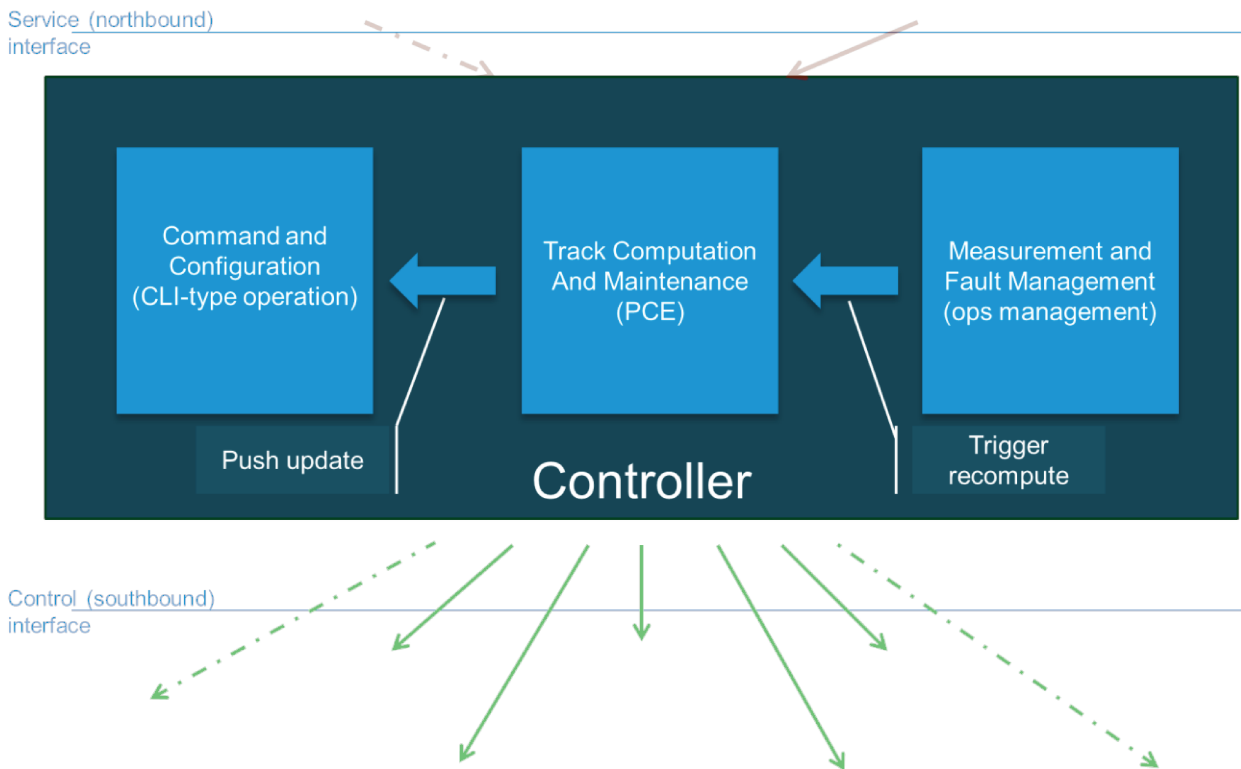


Figure 4.9 – Inside the controller

Figure 4.9 represents the innards of a controller, which effectively is composed of a classical Path Computation Element (PCE) that figures out the optimized layout for the complex paths, a management component that supports the life-cycle of the paths, and Command and Configuration drivers that trigger the required operation over the interfaces that the particular networking hardware support.

This process may be triggered in an open loop, for instance by a user who configures a new device on some HMI, or in a close loop as an automated reaction to an event that is processed in the measurement and fault management entity.

4.5 Limits and perspectives

Deterministic Networking is not a matter of standard bodies and networking layers, (L2 vs. L3); nor is it about virtualization. What it is really about is a tight control of physical operations and scheduling of real buffers and queues, receiving, shaping and resending real packets at precise times with dedicated Hardware.

Such operation requires a network that is precisely synchronized; DetNet will inherit the precise clock synchronization from work done at other bodies, such as but not limited to, GPS, IEEE802.1AS, the Precision Time Protocol (PTP) IEEE Std 1588, or the Network Time Protocol (NTP) [57]⁶. The precision of the synchronization limits the capability to tightly control the resources and thus the amount of deterministic traffic that can be applied onto the network.

Other limits of the centralized approach are on the one hand the complexity of the computation that hinders the scalability of the solution, and on the other hand the control plane overhead and the delays that are required to install a new path, or modify an existing path, from the controller across the network to all the intermediate nodes.

More of a constraint than a limit, the DetNet operation at Layer 3 will require services from lower layers to achieve the required properties end-to-end; the DetNet group will collaborate with 802.1 TSN, which is responsible for Layer 2 operations, to define a common design to support deterministic applications. A number of abstractions such as the end-host operation should be defined in a fashion that is agnostic to the choice of network used for the connectivity. This common design should ensure that the definition of the southbound and northbound interfaces is kept homogeneous between Layer 2 and Layer 3 so as to enable various inter-working models, such as DetNet transporting TSN on a deterministic pseudo-wire, or a mapping interface between a TSN network and a DetNet network at the edge of a deterministic L2 fabric.

The weakest link along a path will limit the whole chain, and an imperfect mapping at the interconnection between two networks with slightly different operations may induce a reduction of end-to-end results. The interconnection between a wired and a wireless network, in particular, can be expected to yield complex mapping issues.

The following chapters explore how deterministic can be applied on wired and wireless media, and which particular benefits are expected in either case. We'll see that determinism can be practically achieved, but through different methods and with different capabilities. The interconnection and the enablement of end-to-end deterministic capabilities are still a complete green field, open for further research.

⁶Neagoe, Cristea, and Banica: "NTP versus PTP in computer networks clock synchronization"

PART II

Diving Deeper in Wireless

The art of low-power wireless sensor network

5.1 So how do we make wireless highly predictable?

The industrial WSNs listed in section 1.3 on page 4 apply the principles discussed in section 3.3 on page 20 and schedule all possible transmissions to minimize collision losses. Scheduling is achieved through TDM, by slicing time and affecting time slots to particular transmissions. This scheduled mode of operation is particularly adapted to well known, periodic flows for which a schedule can be computed in advance.

TSCH combines TDM with channel agility in order to defeat interferences, in particular Multi-Path Fading, which generally affects 2 to 4 channels out of the 16 available with IEEE Std 802.15.4 in the 2.4GHz band, and co-channel interference, in particular when it is located in a limited number of adjacent channels.

TSCH recognizes that smarts in channel selection do not pay off in practice, and all channels that are not black-listed are equally used; they are tried in a pseudo-random order, hopping between non-adjacent channels at each transmission; and though typically, at any given point of time, several channels present a high BER for a particular pair of devices, a series of retransmissions over a sequence of alternate channels eventually bypasses the issue after a few attempts, and an industrial-class reliability can effectively be achieved.

The complexity in TSCH is elsewhere; it comes from the need to ensure that the receiver is tuned to the same channel as the sender at the precise time of the transmission; this requires additional protocol elements to synchronize the network and schedule the transmissions.

A TSCH schedule can be viewed as the program of a mechanical piano that would play the same tune over and over, whereby the channels used would be the music. But the analogy stops here, since at the next iteration of a schedule, all transmissions are rotated by a pseudo random number, with a sequence such as below:

5	6	12	7	15	4	14	11	8	0	1	2	13	3	9	10
---	---	----	---	----	---	----	----	---	---	---	---	----	---	---	----

An example is given in Figure 5.1 with 3 iterations of a schedule, and 4 possible transmissions ($A \mapsto T$), ($B \mapsto A$), ($D \mapsto A$) and ($C \mapsto E$), the first 3 being best effort, to be prioritized in A and possibly colliding between B and D, while the last is protected.

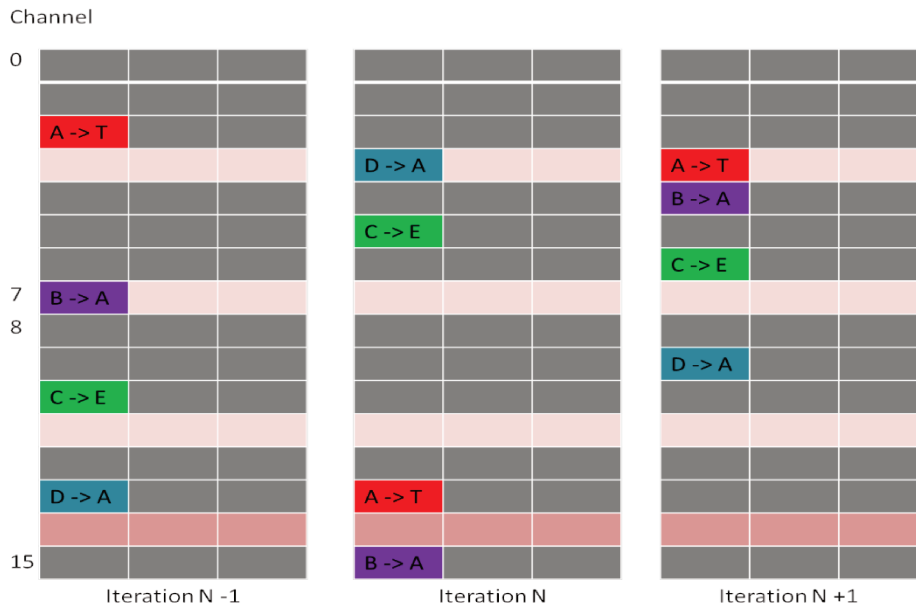


Figure 5.1 – TSCH iterations

Say that a strong interferer causes loss on channel 14 (marked brick red), then any transmission on that channel is mostly doomed. But then say that the particular transmission from node D to node A ($D \mapsto A$) experiences multipath fading on channels 3, 7 and 10, which represented in pink in Figure 5.1. At iteration N, the transmission will fail, but a retransmission at iterations N+1 of the schedule will succeed, because it is on a different channel.

The strategy is effective in combating both co-channel interference and multipath fading, and reach wire-equivalent reliability [126]¹ [129]², given enough time to perform all the required retransmissions.

But it falls short in the face of physical and logical failures such as equipment breakage, reboot, or network desynchronization, and the latency incurred in fast reroute may be too high to guarantee in-time delivery; in contrast, the work presented in this thesis provides multi-path redundancy which addresses these failure cases, and reduces the jitter incurred in retries and rerouting for a better determinism in end-to-end transmissions, arguably at the expense of additional energy spending for which we propose some remediation.

¹Watteyne, Mehta, and Pister: “Reliability through frequency diversity: why channel hopping makes sense”

²Watteyne et al.: “Mitigating Multipath Fading Through Channel Hopping in Wireless Sensor Networks”

5.2 WSNs in Industrial Process Control

ISA100.11a [32]³ and WirelessHART™ [30]⁴ are the major industrial WSN technologies in use today in Process Control networks [58]⁵ [67]⁶; WIA-PA [31]⁷ was developed in parallel in China, also for Process Automation applications[43]⁸. Interestingly, WIA offers a faster FA version for Factory Automation, trading IEEE Std 802.15.4 for an IEEE Std 802.11 Physical (PHY) layer.

IEC 62591 	IEC 62734 ISA100.11a 	IEC 62601 WIA-PA 
IEC based on HART 7.1.	IEC based on 2011 revision	
TDMA	TDMA+CSMA	Alternate from China
fixed time slots (10ms)	Var. time slots	Star, mesh and hybrid topology
Mesh only, no Layer-3	Star, mesh and hybrid topology	Standardization work started in 2006
Shipped YE-2008.	Shipped mid-2010	Work on WIA-FA using 802.11 PHY.
Vendor driven standard	Mostly user driven	
Emerson , E&H, ABB, Siemens	Honeywell , Yokogawa, Invensys	

Figure 5.2 – Industrial Standards for Wireless Process Control

In order to avoid collisions and ensure the transmission of a packet at an exact time, deterministic radio operations require a fully scheduled MAC such as TSCH and LTE/5G. Both ISA100.11a and WirelessHART™ use variations of the TSCH MAC, which is optimized for ultra-low power activities and is a natural match for low-frequency periodic flows, such as control loops, and both are IEC standards, IEC 62734 and IEC 62591, respectively.

Also, both ISA100.11a and WirelessHART™ rely on the centralized routing model promoted by SDN. A Controller called System Manager, or Network Manager, respectively, computes all the routes in the mesh network. Those routes are generally multipath, so as to augment the spatial diversity that is offered to the transported flows and to fast-route

³IEC: *IEC 62734 Industrial networks - Wireless communication network and communication profiles - ISA 100.11a*

⁴IEC: *IEC 62591 Industrial networks - Wireless communication network and communication profiles - WirelessHART™*

⁵Nixon and Round Rock: “A Comparison of WirelessHART™ and ISA100. 11a”

⁶Petersen and Carlsen: “WirelessHART versus ISA100. 11a: the format war hits the factory floor”

⁷IEC: *IEC 62601 Industrial networks - Wireless communication network and communication profiles - WIA-PA*

⁸Liang et al.: “Survey and experiments of WIA-PA specification of industrial wireless network”

around the interference and breakages dynamically. Due to the complexity of solving the NP-complete problem of multi-path route optimization, those networks are not designed to scale beyond roughly 100 nodes, and are generally too costly to efficiently address large scale monitoring applications such as required for the Industrial Internet.

In spite of their common roots and design points, each of these standards is defined from the PHY layer all the way to the application as a monolithic silo, with no desire to interoperate with one another beyond the regulatory capabilities (ETSI, FCC) to share the spectrum with other technologies.

Because a needful device may only exist in one of those standards, practical use cases often need to deploy more than one of those standards, which means that different hardware must be installed and maintained by different OT specialists, which multiplies the operational expenses (OPEX), and appears to be a major limitation that to their wider deployment.

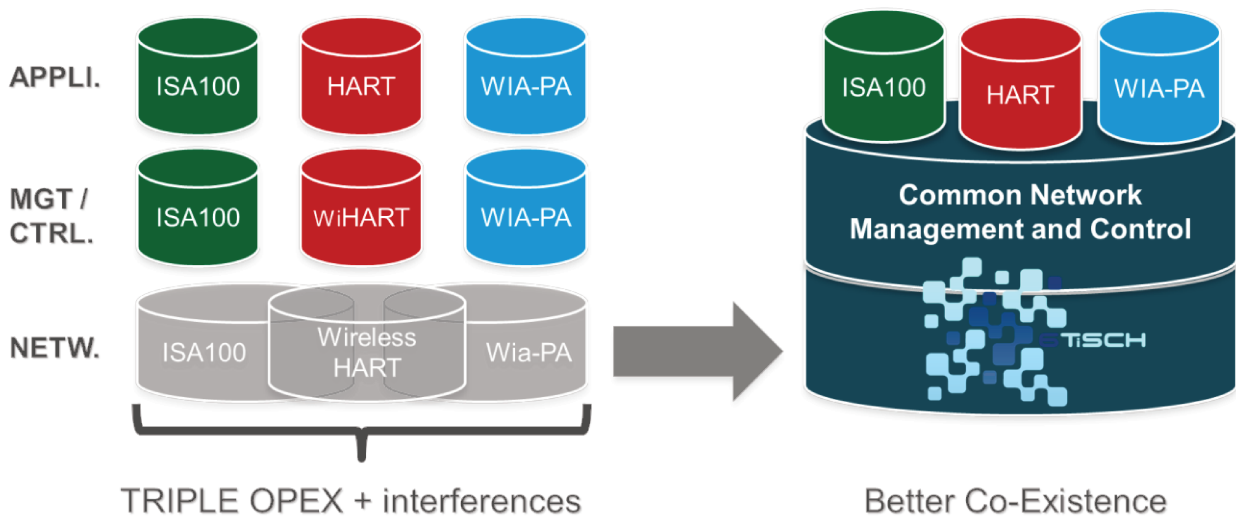


Figure 5.3 – Triple OPEX

As illustrated in figure 5.3, this contrasts with the end-to-end principle that guides the Internet designs, with a network that is agnostic to the applications and can be shared between multiple existing and any upcoming ones.

It is usual that a green field starts with highly specialized proprietary or semi-proprietary solutions; this tends to maintain the prices high and limits the adoption of the new technology. Soon enough, open standards based on the Internet Protocol (IP) -that would be IPv6 for the IoT- and the end-to-end principle eventually take over. Bringing IP eliminates the need for gateways to provide connectivity to a wider network such as the Internet, and enables a common network infrastructure and a shared management for all applications, which drives the OPEX down and results in a larger acceptance. This transition can be expected for industrial WSNs as well.

But the Industrial Internet is also - and a lot - about reporting non-critical data such

as diagnostics and for which the incumbent protocols are not a cost-efficient solution. The next problem for industrial wireless is thus to extend highly predictable WSN technologies to support IPv6, and in the process to share bandwidth and other physical resources with non-deterministic traffic, reaching higher scales at lower costs.

5.3 6TiSCH and best effort IPv6

The application of wireless technology in the operational space has enabled a variety of new devices to get interconnected, at a very low marginal cost per device, at any distance ranging from Near Field to interplanetary, and in circumstances where wiring may not be practical, for instance on fast-moving or rotating devices.

While critical monitoring was by and large the main initial application, large scale/best effort capture of missing measurements for analytics purposes is now the fastest growing application of wireless technology in OT; a study by ABI Research shown in figure 5.4 indicates that it should represent more than the half of the deployed devices within the next 3 years.

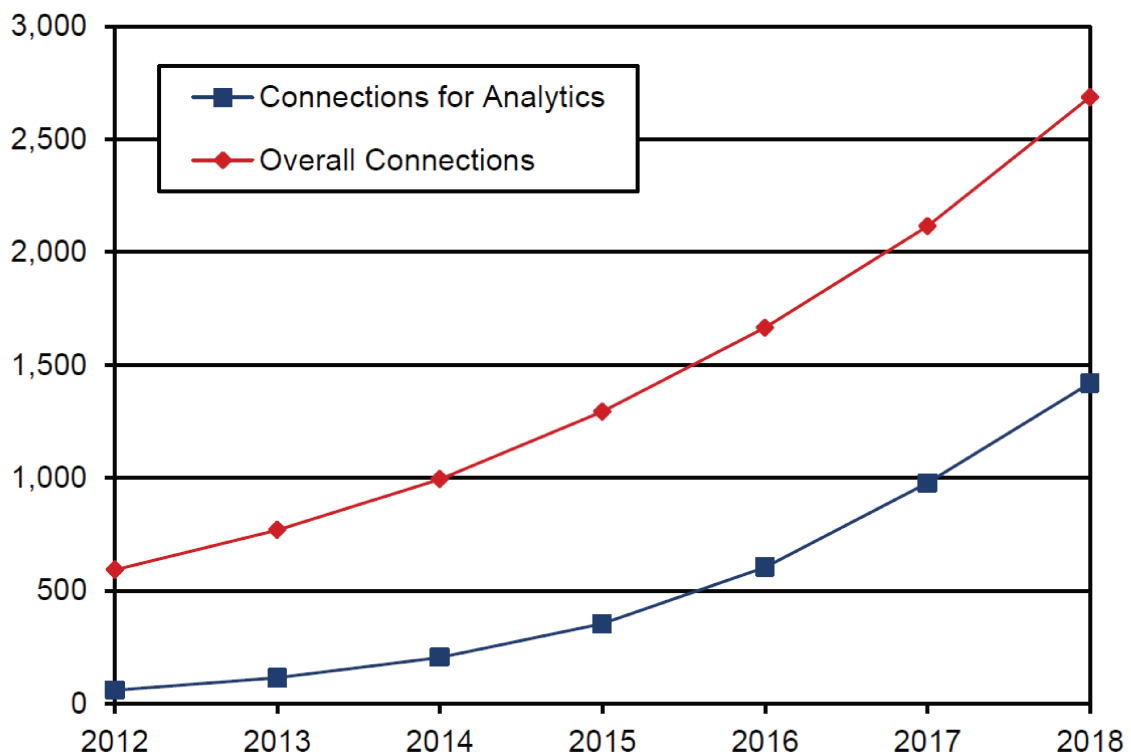


Figure 5.4 – Growth in analytics-related measurements. Source: ABI Research

To support this growing class of traffic, standardization work has started at the IETF 6TiSCH WG that enables IPv6 over the TSCH MAC technology for best effort statistically multiplexed traffic. This way, a large number of rare transmissions can share TSCH

resources that are left unused by deterministic control traffic, without interfering with it. The challenge is to adapt to the dynamics of these transmissions and preserve energy.

The 6TiSCH Architecture discusses techniques for allocating resources in the form of chunks of time slots that allow the physical separation of the two types of traffic, enabling an effective coexistence whereby the highly predictable properties of time-sensitive flows are maintained in the presence of stochastic traffic.



6TiSCH addresses this additional challenge and allows for a mix of stochastic (best effort) IPv6 flows with such well-known deterministic flows while preserving the deterministic properties regardless of the load imposed by other flows. While the work on a protocol stack [65]⁹ for best effort is well-advanced at the IETF, and though the vision is clearly to apply the methods defined at the IETF DetNet WG, it remains to be ensured that the way a path is signaled in wired networks is fit for the wireless medium as well.

The pillar of this stack, 6LoWPAN, has emerged as the suite of protocols that enables IPv6 on any low-power personal-area device, however small. Multiple Internet-Drafts¹⁰ are available from the IETF that specify 6LoWPAN over such Low-power Lossy Networks (LLN)s as powerline (IEEE Std 1901.2-2013), BACnet, NFC, ZWave, Bluetooth Low Energy (BTLE) and even IEEE Std 802.11ah.

To clarify a common misunderstanding, 6LoWPAN does not compare with a full solution standard such as WirelessHARTTM and ISA100.11a; 6LoWPAN is the standard way to support IPv6 over LLNs, meant to be included in a full solution such as the above. 6LoWPAN provides an Adaptation Layer and a novel registration-based IPv6 Neighbor Discovery (ND) operation for duplicate address detection (DAD) that saves inefficient multicast operations.

The 6TiSCH architecture [82]¹¹ combines 6LoWPAN for IPv6, RPL for routing between constrained devices, and a suite of IETF protocols, for application over IEEE Std 802.15.4 TSCH; in a fashion, 6TiSCH generalizes ISA100.11a and WirelessHARTTM to en-

⁹Palattella et al.: “Standardized protocol stack for the internet of (important) things”

¹⁰tools.ietf.org/wg/6lo/

¹¹Thubert: *An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4*

able Industrial Internet use case over 802.15.4 TSCH, though it must be noted that for the most part, the 6TiSCH architecture is NOT specific to that MAC.

The extent of the problem space for 6TiSCH encompasses one or more LLNs, which are reachable over one or more LLN Border Routers (6LBRs). The LLNs may be federated through a common high speed backbone link such as an Ethernet switched fabric operating classical IPv6 ND. 6TiSCH introduces the concept of a Backbone Router (6BBR) [83]¹² functionality that performs routing and proxy operations to aggregate the LLNs and the backbone link into a single Multi-Link Subnet.

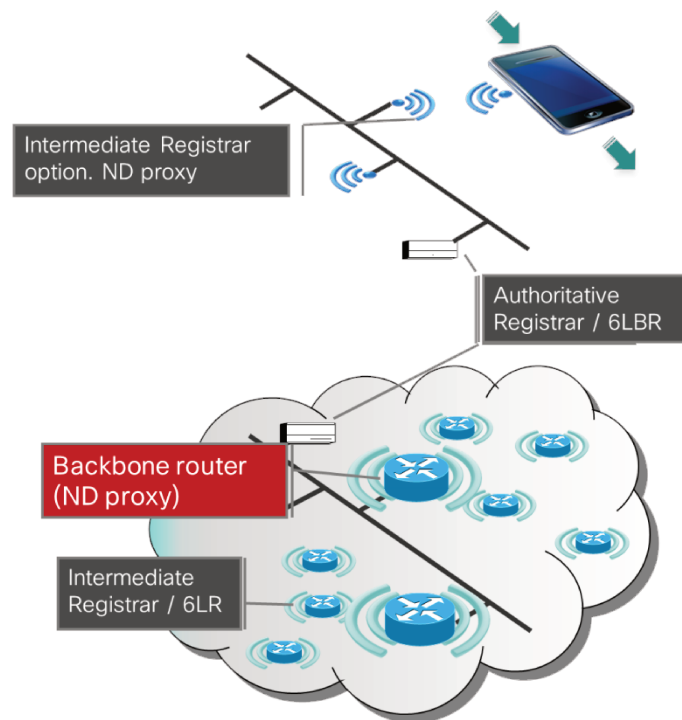


Figure 5.5 – Backbone Router

With the currently defined operation and as illustrated in figure 5.5, the backbone router proxies on Neighbor Discovery (ND) operation over the backbone on behalf of the LLN devices, enabling them to remain asleep as they are looked up from backbone devices. As a result, the BBR attracts the packets from the backbone and routes them to the destination LLN devices through the operation of RPL.

The first deliverable of the WG, the 6TiSCH Minimal Support [121]¹³, describes a slotted-aloha operation over a static schedule composed of shared timeslots, which means that any node can transmit or listen on these slots; the Minimal Support specification is complete at the time of this writing, going through the IESG review process as a best practice document.

A companion document 6TiSCH Minimal Security [124]¹⁴ is on the works to enable

¹²Thubert: *IPv6 Backbone Router*

¹³Vilajosana, Pister, and Watteyne: *Minimal 6TiSCH Configuration*

¹⁴Vucinic, Simon, and Pister: *Minimal Security Framework for 6TiSCH*

the initial settings of Layer 2 keys and a secure connectivity for the configuration of the device. Since the Minimal Security assumes the one-touch manual setting of Pre-Shared Keys (PSK) on each individual device, there is an additional desire to enable fully autonomic operations and avoids the manual intervention.

Based on certificates installed by the vendor on the device, and a backend collaboration between the vendor and the target domain, the 6TiSCH Secure Join Protocol [71]¹⁵ will cover the phase that precedes the Minimal Security and enable the initial trust and key exchange between the device and the domain, eliminating the need for a PSK.

The group is now addressing the challenge to make the schedule dynamic, which involves a Scheduling Function [14]¹⁶ that allocates and releases time slots dynamically for parent/child unicast communication along the RPL graph, based on the observed needs of the current flows, and a peer-to-peer protocol [125]¹⁷ between adjacent nodes to negotiate the time slots.

When this work is complete, 6TiSCH may recharter to work on adapting DetNet to wireless and ultimately enable the IT/OT convergence for WSNs.

¹⁵Richardson: *6tisch Secure Join protocol*

¹⁶Dujovne et al.: *6TiSCH 6top Scheduling Function Zero (SF0)*

¹⁷Wang and Vilajosana: *6top Protocol (6P)*

The vision of 6TiSCH centralized scheduling

The wireless convergence

In order to fully realize its architecture, the 6TiSCH WG has to make work together:

- A Distributed Routing for large scale monitoring (RPL) to enable the co-existence of low-criticality flows of *IPv6-based Industrial Internet*, ensuring the separation of resources between deterministic and stochastic and leveraging IEEE/IETF standards (IPv6, IEEE Std 802.15.4 TSCH, 6LoWPAN, etc. . .), and this work is well underway;
- A Centralized Routing for *Time-Sensitive flows* for mission-critical data streams such as monitoring, control loops, diagnostics and alerts; a deterministic reach back to Fog or Cloud based application must be provided for virtualized loops and measurement files, following the work initiated at DetNet for the abstractions that are common to all networks.

6.1 PCE vs. 6TiSCH

With 6TiSCH, a PCE controls the network via a Command and Configuration interface that implements specific device drivers, e.g. CLI, NETCONF or CoMi. This PCE inherits from the overall DetNet design but requires some specific awareness for such notions as channels, which must be added to the common DetNet design.

A similar concept arises when implementing DetNet on fiber optics, which requires the awareness of the light wavelength -the *lambda*- that is used for multiplexing. In that context, (MPLS) was already generalized as G-MPLS [49]¹ by adding the implicit context of the lambda to make a switching decision.

¹Mannie: “Generalized multi-protocol label switching (GMPLS) architecture”

In other words, as opposed to switching based only on octets in a frame header, G-MPLS uses the physical context of the packet, here the lambda at which it was received, to make the switching decision.

In “Label switching over IEEE802.15.4e networks,” Morell, Vilajosana, Vicario, and Watteyne [56]² observe that with scheduled IEEE Std 802.15.4 TSCH, a frame can be switched based on the particular time and the particular channel at which it was received. This can effectively be considered as a form of G-MPLS, and enables to save octets when transmitting on a constrained medium.

6.2 6TiSCH base elements (time slots, schedule, chunks and bundles)

Based on IEEE Std 802.15.4 TSCH, 6TiSCH defines a new concept that is global to the network, called a *Channel distribution/usage* (CDU) matrix, as illustrated in figure 6.1; a CDU matrix is composed of so-called *cells*, each with an equal duration of one network *timeslot*.

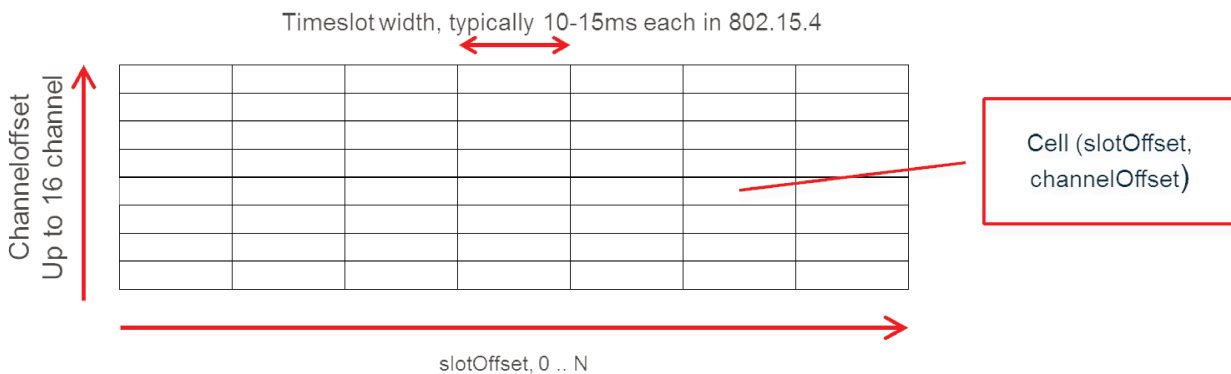


Figure 6.1 – 6TiSCH CDU matrix

The CDU matrix provides the characteristics of the network, such as the physical properties of the cells, the channels being used vs. blacklisted, and the definitions therein have a global impact on the network, for instance the size of the cells indicates the duration of a *timeslot* in the *schedule* of each participating device.

The height of the matrix equals the number of available channels (indexed by *ChannelOffset*), and its width is the period of the iterative network scheduling operation (indexed by *slotOffset*).

The width of the CDU matrix must be in phase with the period of the application. If a node needs to support different applications with incompatible periods, then multiple CDU matrices must be defined to accommodate those different periods, and the

²Morell et al.: “Label switching over IEEE802.15.4e networks”

transmissions associated to different CDU matrices may periodically overlap.

The CDU matrix also describes how cells are grouped in *Chunks* of similar properties. A Chunk is a list of cells, well-distributed in time and frequency, within the CDU matrix. In other words, a chunk represents a unit of bandwidth and can be seen as the generalization of a transmission channel in the time/frequency domain.

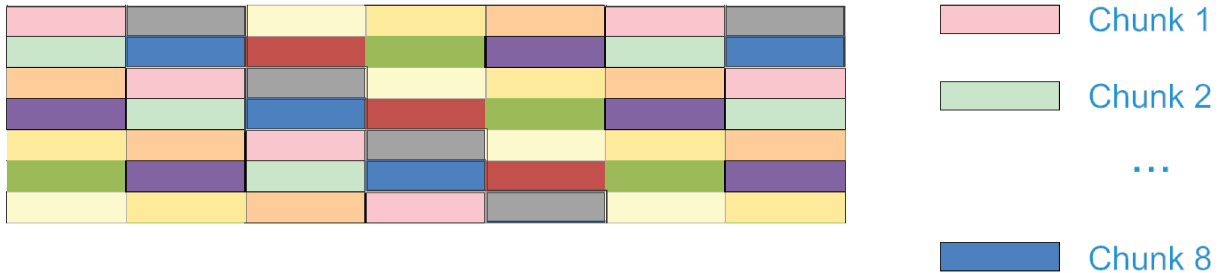


Figure 6.2 – Example CDU matrix partitioned in Chunks

The partition of the CDU in chunks is globally known by all the nodes in the network that participate to an appropriation process that is left to be fully defined, by which a node gets ownership of the cells in the Chunk. A node that appropriates a Chunk gets to decide which transmissions will occur over the cells in the chunk within its interference domain.

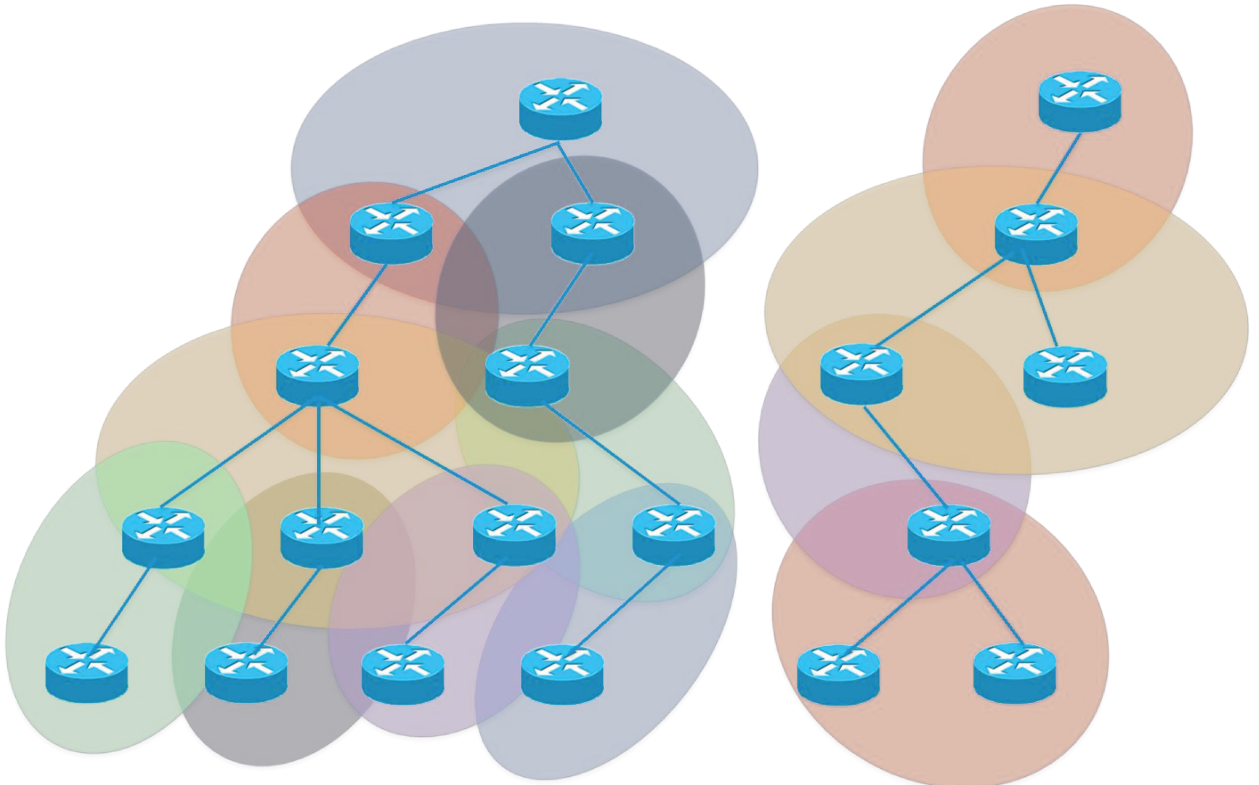


Figure 6.3 – Spatial application of Chunks

The idea is that a node, typically a RPL parent, appropriates a chunk, and then uses it to communicate with its children. The appropriation process will enable a parent to grab a whole chunk and validate that this chunk is not used within its interference domain.

A *schedule*, on the other hand, is a MAC-level abstraction that is local to each node; a node’s schedule represents the minimal knowledge that this node needs to participate to the network. To express a schedule, TSCH defines the concept of *slotframes*, which represent the timely layout of *timeslots*, and *bundles*, which represent their logical relationship (e.g. they belong to a same IPv6 Link).

A *slotframe* comprises a series of timeslots of equal length and priority, indexed by *SlotOffset*; a timeslot represents the activity of this node at that slotOffset, e.g. which cell is being used, and whether it is a reception or a transmission. The duration of a slotframe is aligned with that of a CDU matrix as illustrated in figure 6.4.

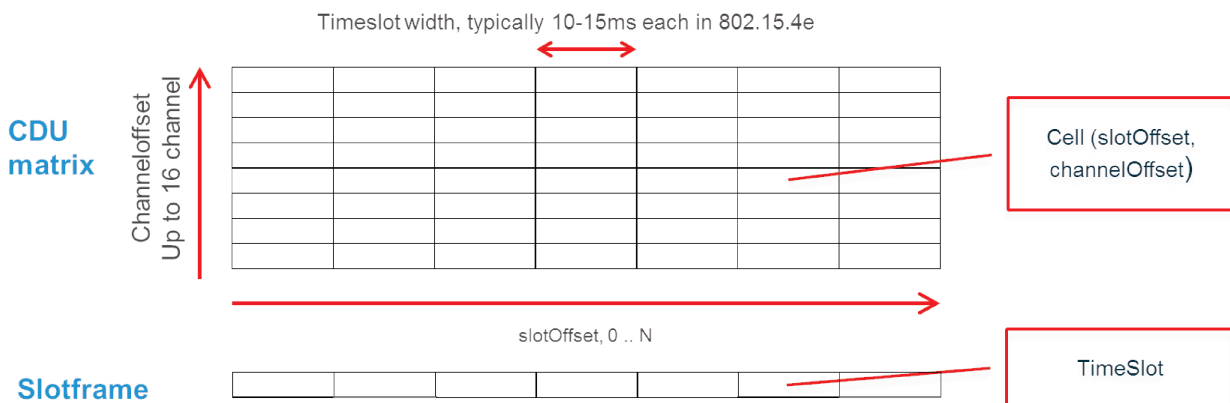


Figure 6.4 – 6TiSCH Slotframe

A node may need to support concurrent applications, and these applications may need to transmit or receive at a same time. To resolve such collisions, the application flows with different priorities are scheduled on different slotframes, ordered by application priority.

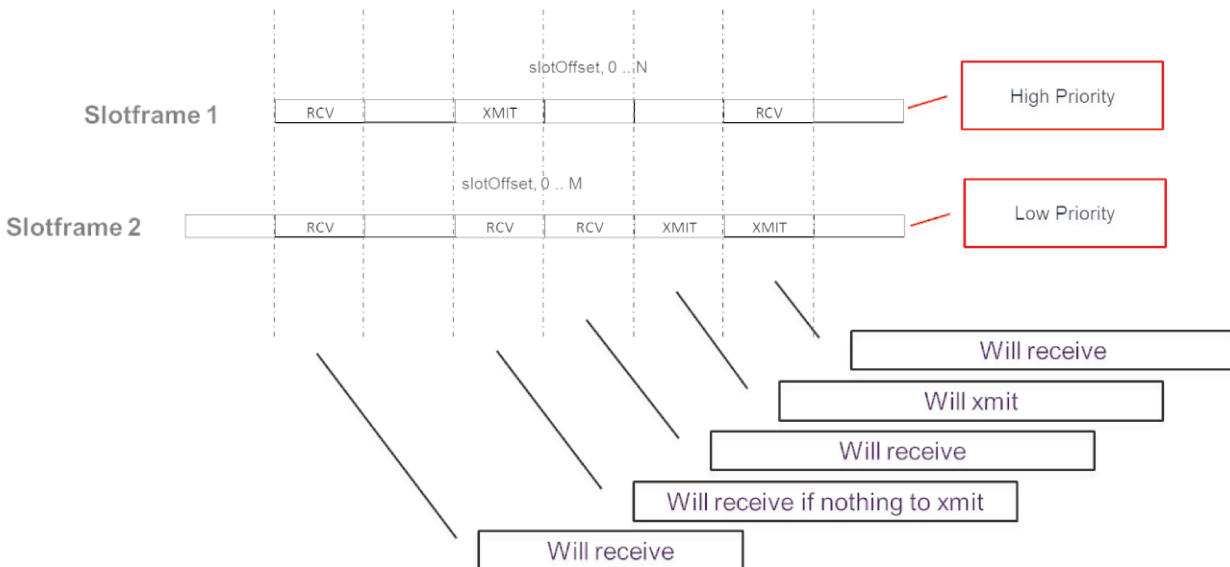


Figure 6.5 – 6TiSCH Slotframe prioritization and schedule execution

Decision to transmit / receive is made at the beginning of a timeslot based on existing frames in queues and slotframe priorities, as illustrated in figure 6.5.

We proposed an appropriation mechanism based on monitoring an energy pattern in the first cell of the chunk [98]³. With that method, the interference can be avoided beyond the range of carrier detection, and previously undetected duplication of appropriation can be observed and eliminated.

With TSCH, one node wakes up to receive a frame when another wakes up to transmit, on a timeslots of their respective schedules that are associated with a same cell of the CDU. They wake up at the exact same time -modulo a small guard time to cover for the clock drift- and on the exact channel, which it rotat-

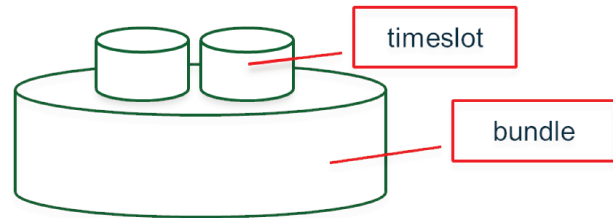


Figure 6.6 – A 6TiSCH Bundle

ing with the same pseudo-random algorithm on both ends. This is how power-constrained devices can stay in low-energy deep sleep mode when not involved in a communication.

With 6TiSCH, the bandwidth that is allocated for a particular purpose is represented as a *bundle* of timeslots. A node that owns of a chunk of cells will associate them to timeslots in matching bundles on both ends. When applied to Layer 3, a bundle participates to an IP Link between adjacent nodes, whereas at Layer 2, a bundle participates to a path between a source node and a destination node and this node may only be an intermediate forwarder.

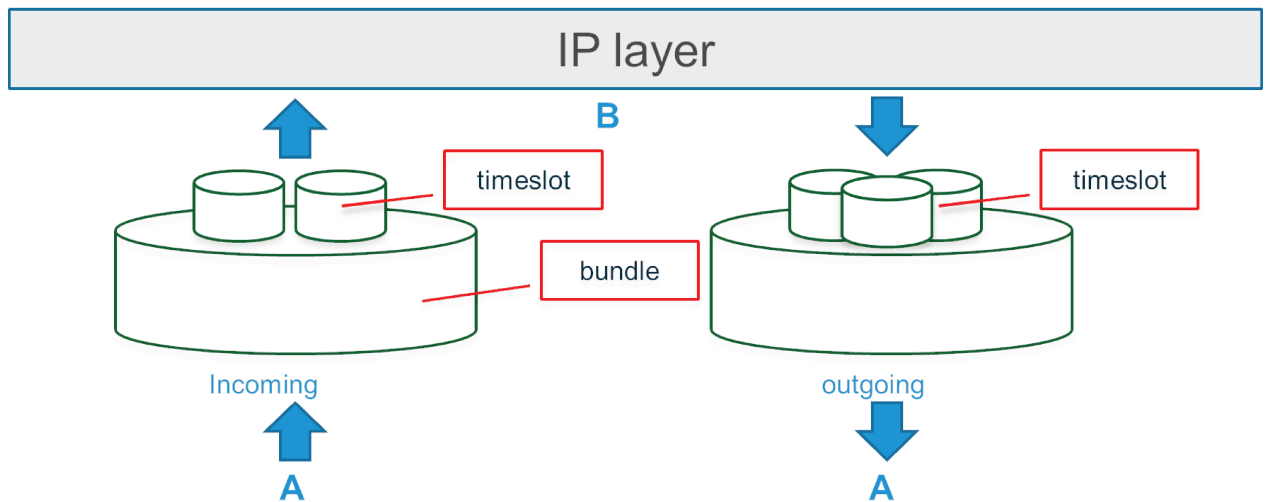


Figure 6.7 – A 6TiSCH Layer 3 Bundle

Figure 6.7 illustrates that the establishment of a Layer 3 Link abstraction for IPv6 communication between a node A and a node B requires setting up a pair of bundles, one in each direction. If the traffic is asymmetrical, it may be that the number of timeslots in the respective bundles is not the same in both directions.

³Thubert, Vasseur, and Wetterwald: *DISTRIBUTED APPROPRIATION AND DEFENSE OF TIME SLOT OWNERSHIP IN A CHANNEL HOPPING SCHEDULE*

For Layer 2 communications, 6TiSCH introduces the concept of a *Track*, which we elaborate upon in section 6.4 on page 53, and that of Track switching. The simplest form of Track is a serial switched path such as shown in figure 6.8 with a purple serial Track from a node A to a node U, and a green one from a node B to a node V.

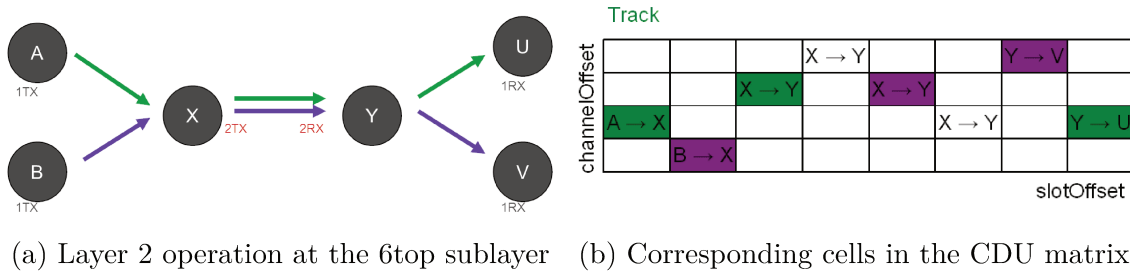


Figure 6.8 – Forwarding along 2 simple serial Tracks, A to U and B to V

In that model, as illustrated in Figure 6.9, the bundle that corresponds to receive slots in a node B for transmission from a node A is now paired in the schedule of node B with a bundle of transmit slots from B to C. This sequence of paired bundles from node to node determines a circuit with a particular amount of bandwidth between a source node and a destination node along a serial path.

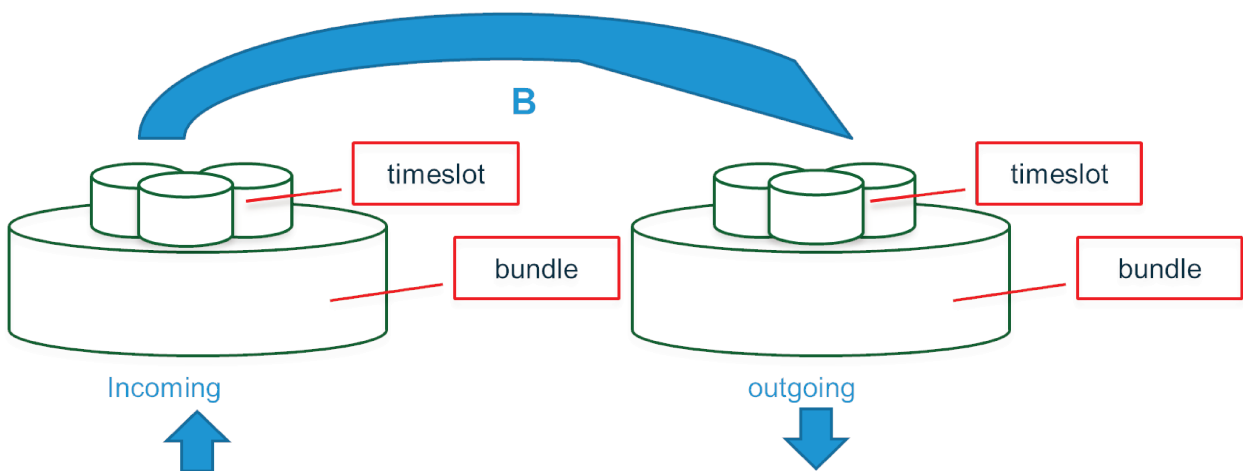


Figure 6.9 – A 6TiSCH Layer 2 Bundle

But for the more general case of 6TiSCH Tracks, more complex constructs are required to express the full semantics of Frame Replication, Retries and Elimination. In our experimentation, we used bundles of equivalent timeslots, so the next timeslot in a transmit bundle could be used for a retransmission of a same frame, and a different transmit bundle was used to represent the Replication. A single receive bundle was used to represent the Elimination of multiple copies of a same frame, regardless of the previous hop.

In both Layer 2 and Layer 3 cases, the timeslots in the receive bundle in node A must use the same cell as the matching timeslot used for reception in node B, but the direction, transmit vs. receive, is opposite.

This can be ensured in a distributed fashion by the 6top Protocol [125]⁴, which leverages a new IEEE Std 802.15.4 Information Element (IE) in 802.15.4 frames to update the cells that are allocated between adjacent 6TiSCH nodes; that new IE was delegated by the IEEE Assigned Numbers Authority⁵ (ANA) to the IETF and allocations will be handled by the Internet Corporation for Assigned Names and Numbers (ICANN)⁶ Internet Assigned Numbers Authority (IANA)⁷.

The schedule is enforced by the 6top sublayer; 6top resides at the upper Layer 2, above the IEEE802.15.4 Medium Access Control (MAC) but below the 6LoWPAN sublayers for compression and fragmentation, and the IP(v6) Layer above it.

As illustrated in Figure 6.10, a classical IoT device will operate all layers, and may either switch or route a packet. Operating in 6top is akin to MPLS switching, which is sometimes seen as an intermediate Layer, like a Layer 2.5.

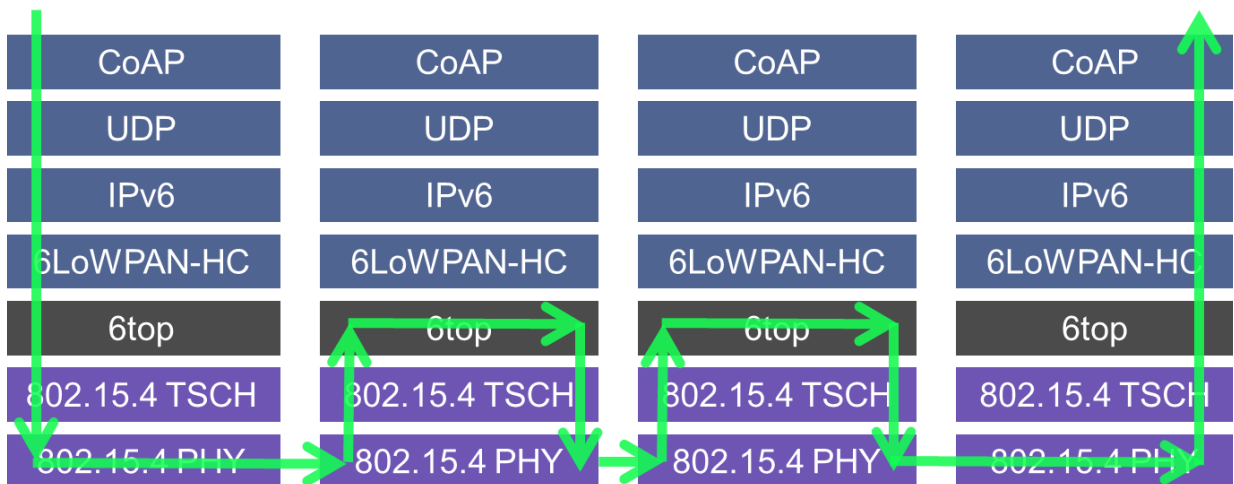


Figure 6.10 – G-MPLS switching along Tracks

It is worth noting that, because the timeslot information is sufficient for the 6top sublayer to make a switching decision, and this, regardless of the payload of the frame, 6LoWPAN fragments can be forwarded along the G-MPLS-switched path without the need of reassembly in intermediate nodes. This was true for any frames in the art of 802.15.4 TSCH and remains true for 6TiSCH Tracks.

This contrasts with the classical 6LoWPAN fragment operation in Layer 3 -aka *route-over-* mesh networks, where fragments must be reassembled at each hop, in order to enable the IP routing operation based on the IP header that is only present in the first fragment. Work on *LLN Fragment Forwarding and Recovery* [94]⁸ is ongoing to improve that situation and forward individual fragments all the way.

⁴Wang and Vilajosana: *6top Protocol (6P)*

⁵www.ieee802.org/15/ANA.html

⁶<https://www.icann.org/resources/pages/welcome-2012-02-25-en>

⁷www.ietf.org/iana.html

⁸Thubert and Hui: *LLN Fragment Forwarding and Recovery*

6.3 Applying DetNet to 6TiSCH

This section shows how a variation of the Available Routing Construct (ARC) technology can be leveraged to form the ladder, and then how timeslots can be scheduled and ACKs optionally suppressed to save additional energy.

In a nutshell, the major steps for 6TiSCH centralized operations as controlled by a PCE are as follows:

- Discovering and exposing the 6TiSCH topology to the PCE/controller
- Programming a full schedule per node vs. TE-like path setup
- Retransmission vs. elimination and replication
- Detecting and rerouting around interferences

The WG has documented which kind of operations may be required from a central computer to establish a deterministic path over the 6TiSCH network. Some of this text was integrated in the *Deterministic Networking Use Cases* [26]⁹ Work Group document, in section 5 *Wireless for Industrial*¹⁰.

In order to avoid paraphrasing, that section is integrated here by reference.

A more complete description of the 6TiSCH requirements for DetNet and the concept of a Track can be found in the *6TiSCH requirements for DetNet*, which is also integrated by reference to this thesis.

In that document, we detail the dependencies on DetNet and PCE controller to express topologies and capabilities, as well as abstract state that the controller must be able to program into the network devices to enable deterministic forwarding operations.

In particular, we elaborate on the design of a 6TiSCH Track. In order to cope with the high loss ratio on the wireless medium, there is a need to support the concept of Replication and Elimination, as combined with that of Retransmission, which is more specific to radio links. It results that a Track may be a lot more complex than a serial sequence of hops, and involve correlating multiple transmissions and reception as relating to a same packet.

In other words, a node may need to:

- Expect more than one copies of the same frame at some subsequent slotOffset/channelOffset and eliminate the duplicate;
- Forward replicated copies of that frame at multiple slotOffset/channelOffset;
- Perform retries when the copies are not received, which is a concept that is not present in wired Deterministic Networks.

⁹Grossman et al.: *Deterministic Networking Use Cases*

¹⁰<https://tools.ietf.org/html/draft-ietf-detnet-use-cases-11#section-5>

6.4 Forwarding along 6TiSCH Tracks

In this thesis, we suggest to extend the concept of G-MPLS so that the physical properties of a transmission indicate not simply a next hop but a full context for Replication, Retries and Elimination, all done without the need to add information inside the frame itself, and leverage that concept to form complex Tracks.

The concept of a 6TiSCH Track is yet to be fully elaborated, but this work, in particular with the *6TiSCH requirements for DetNet* [81]¹¹, has largely contributed to shape it in its current form.

A deterministic flow is qualified with information such as:

- *bandwidth requirements*, which translates in a number of cells in a bundle;
- *precise time of packet transmission*, which translates in the slotOffset in the schedule for those cells, and a
- *maximum latency*, which translates in the alignment of schedules along a path, which must be computed by a PCE so as to meet the end-to-end deadline.

In order to closely indicate the deterministic scheduling of every individual packet, this work suggests a number of operational rules for 6TiSCH Tracks:

- The IEEE Std 802.15.4 destination MAC address in the frames is always set to multicast (0xFFFF), meaning that all nodes that have a Track programmed to listen to that particular timeslot are expected to accept the frame and handle it;
- one (or more than one) bundle are assigned uniquely to each frame in the flow; in other words, all the cells in the bundle carry a copy of a same frame;
- Multiple cells in a bundle indicate that a packet is to be retried in case of a loss; an acknowledgement is the indication that the other cells in the bundle will not be used. This denotes an OR operation (more on this in section 10.1 on page 67) between the cells in the bundle. The experimental phase of this thesis relies on multi-cell bundles to experiment with Retries associated with retransmissions. It can be noted that, from the standpoint of the transmitting node, it would not matter whether the receiving node is the same for all tries; a single bundle can be used to try to send to node A, and if that fails to node B, etc...
- Multiple transmit bundles denote a Replication. Regardless of the success or failure of the transmissions over bundle A, bundle B will be tried. This denotes an AND operation between the cells in the bundle. The experimental phase of this thesis relies on multiple one-cell bundles to experiment with Replication and Elimination while keeping the loss ratio high enough by avoiding retransmissions.

¹¹Thubert: *6TiSCH requirements for DetNet*

- Multiple receive bundles indicate an Elimination. In some cases, there is no point in listening to the next bundles once the frame is received. With the BIER-TE-based technique that is presented in chapter 11 on page 71, this thesis relies on receiving all the possible copies and performing an AND operation between the bitmaps of each copy so as to determine which transmissions worked or failed along the Track.

Enabling the convergence

In the use case of Industrial Internet, industries are after the next percentile point of operational efficiency to reduce down-times and operational expenditures (OPEX), and save natural resources and energy. A new form of optimization is emerging, which requires collecting and processing of live “big data”, that’s huge amounts of missing measurements, by widely distributed sensing and analytics (Fog) capabilities.

For reasons of cost and complexity of operation, as well as available spectrum, there is a need of convergence of deterministic industrial networks acting as silos onto IP and to share bandwidth with non-deterministic traffic, reaching higher scales at lower costs.

Type of traffic	Deterministic (e.g. Control Loops)	Stochastic (e.g. classical IP)
Type of MAC		
Deterministic (e.g. 802.15.4 TSCH)	<p>Good fit Adapted to centralized routing and fully scheduled operation All industrial protocols are here</p>	<p>Difficult but achievable: requires dynamic allocation of transmission resources (6TiSCH)</p>
Stochastic (e.g. Zigbee, Wi-Fi)	<p>Problems with channel access (guard time) Lead to gross over- provisioning CSMA alone cannot provide hard guarantees</p>	<p>Good fit Adapted for IP traffic, distributed routing and statistical multiplexing with RED</p>

Figure 7.1 – Matching MAC technology and type of traffic

In the art of OT, industrial networks are deployed to transport deterministic flows (e.g. control loops) over Deterministic Networks (serial, TDM buses). In the IT art, best effort networks (Ethernet, Wi-Fi) transport statistically multiplexed traffic (IP). In both cases, there is a natural fit and multiple technologies have been successfully deployed over the last 40 years.

Converging deterministic and stochastic flows on a same network requires breaking that natural coupling. Either the network is deterministic and the challenge is to transport best effort traffic, or the network is deterministic and the challenge is the reverse.

As it goes, both have been tried. Applying QoS (IEEE Std 802.11e or derivatives) and with a widely underutilized physical medium, it is possible to make a stochastic packet-switched network transport Deterministic Flows. But under load from best effort traffic, this solution fails rapidly, faster on Wi-Fi than on Ethernet since 802.11e is a statistical QoS operation [48]¹.

The 6TiSCH architecture defines the other way around, with the challenge to enable stochastic IP traffic over a deterministic TSCH MAC. The idea is that a PCE reserves *hard cells* from the time/frequency matrix CDU matrix for deterministic flows; those cells cannot be reused or displaced; they are allocated when a flow is established, along complex paths called *Tracks*. The reservation ensures that a flow that is placed on a Track cannot be influenced whatsoever by stochastic IP flows, which can only use the unreserved cells.

What's free in the CDU matrix is partitioned in chunks of *soft cells* that are made available for best effort traffic. This Time and Frequency Division Multiplexing technique is how 6TiSCH ensure the co-existence of deterministic and best effort traffic on a same medium.

¹Mangold et al.: “Analysis of IEEE 802.11 e for QoS support in wireless LANs”

PART III

Novel Approaches for Higher Predictability

Computing a complex Track: the ARC technology

8.1 Introducing the ARC technology

The Available Routing Constructs (ARCs) technology was introduced with an IETF Internet-Draft [85]¹ and a number of associated patents filed at USPTO [86]² [115]³ [116]⁴; it was further discussed in academics papers [53]⁵.

ARCs brings a new conceptual model that reconsiders the traditional greedy approach of a serial path that must always progress towards a destination; instead, ARCs enables novel forwarding behaviors whereby it becomes possible for a packet to backtrack within a limited number of routers and then follow another shortest path, which, by construction, avoids the problem, whether it is a breakage or a simple congestion.

As shown in figure 8.1, an ARC is a two-edged routing construct that forms its own fault isolation and recovery domain. An ARC has a high point that we call a cursor, and 2 exits that we call edges; the edges of an ARC are directed outwards and links within the ARC are oriented from the cursor towards the edges.

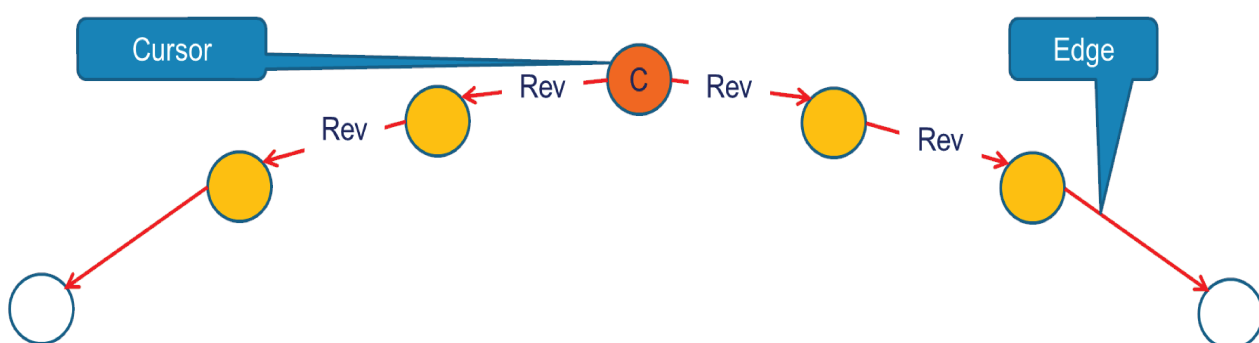


Figure 8.1 – An ARC, [Re] for Reversible Link

A key property is that the links in an ARC are reversible; they are returned by moving the cursor, between the new position and the old position.

¹Thubert and Bellagamba: *Available Routing Constructs*

²Thubert and Bellagamba: *Generating a loop-free routing topology based on merging buttressing arcs into routing arcs*

³Thubert et al.: *Repair of Failed Network Routing Arcs Using Data Plane Protocol*

⁴Thubert et al.: *Resilient forwarding of packets in an ARC chain topology network*

⁵Melakessou, Palattella, and Engel: “Towards a new way of reliable routing: Multiple paths over ARCs”

An ARC constitutes its own domain for load balancing and failure recovery; one failure *per ARC* can be recovered, and from there, an alternate shortest path to avoid a congestion or a breakage can be taken; in contrast, MRT builds global blue and red trees and thus is limited to only one reroute [91]⁶ over the whole topology.

Using an algorithm such as Open Lowest ARC First (OLAF) [85]⁷, for each destination, the topology of nodes and links is formatted as a Directed Acyclic Graph (DAG) of ARCs, called an *ARCset*, which terminates at the destination, as shown in figure 8.2.

In order to form the DAG, both edges of an ARC end either in another, downstream, ARC, or at one of the destinations, which we collectively call Omega. Omega may be a node, a border router, or a collection of such.

It results that a packet that exits an ARC reaches another ARC that is nearer to the destination Omega, regardless of the edge that the packet uses to exit the ARC. So routing between ARCs is effectively greedy whereas rerouting inside an ARC may increase the cost metric.

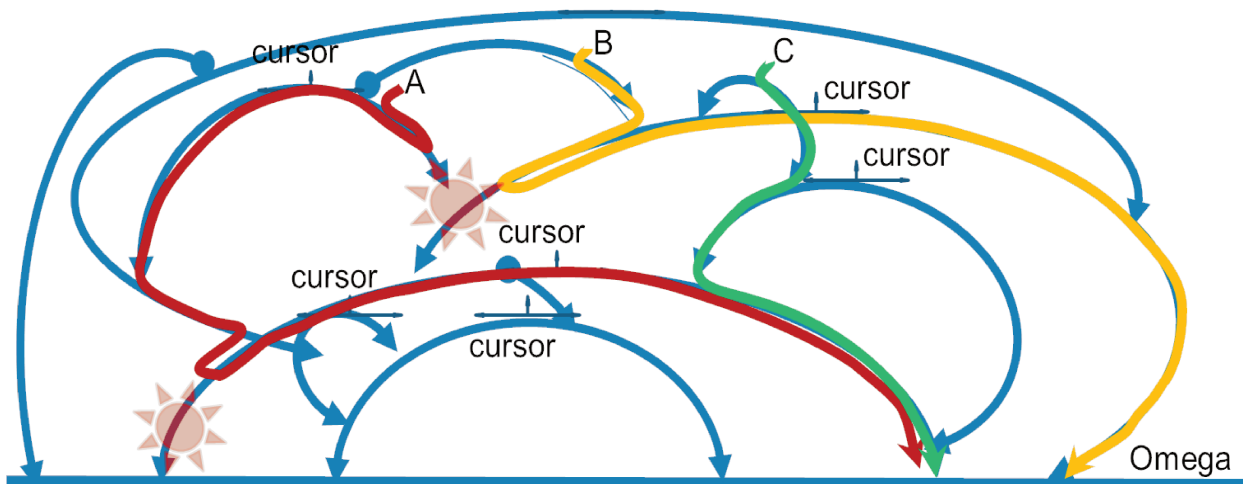


Figure 8.2 – An ARC Multi-Path to Omega

Figure 8.2 also shows the paths followed by three packets, respectively from nodes A, B, and C, towards Omega. Each path is a cascade of ARCs along the ARCset, with the possibility to exit at either edge of the ARC on each ARC.

Depending on the source and accidents along the way, the path followed by a packet may be different from one packet to another, and even end at different members of Omega.

In this example, the packet from node C follows the green path on the right and does not experience an issue, but the packets from A and B, respectively following the red and the yellow paths on the left, face a forwarding plane failure and have to turn around along their respective ARCs, towards the alternate edge of the ARC.

⁶Thubert, Enyedi, and Ramasubramanian: *ARC vs. MRT*

⁷Thubert and Bellagamba: *Available Routing Constructs*

In both cases, the packets exit on the other end of the ARC and from there take the shortest path to reach the destination.

The yellow packet continues to the destination Omega, but the red packet faces a second breakage in another ARC further down the road, and is turned again within that other ARC. Since each ARC is its own recovery domain, the second breakage is corrected independently and the red packet also makes it to the destination.

Technically-speaking, an ARC can be implemented as a sequence of hops with two ends. It results that one ARC is, configuration-wise, very similar to an MPLS-TP path; rerouting along an ARC can also be seen as a particular application of existing fast-reroute.

An ARCset is typically computed by a separate Network Management System or an SDN controller. ARCs extends the concept of SDN centralized routing computation to any traffic, and proposes a new balance of responsibilities between, on the one hand, the slow-response centralized routing computation, and, on the other hand, the quick-response forwarding-plane operations, which are still distributed and can be automated.

An ARC deployment in an IPv6 or an MPLS network may leverage Segment Routing [21]⁸ to direct packets towards the appropriate edge.

ARCs offers a new model and a number of issues that were difficult to solve with greedy technologies appear a lot simpler from that new perspective. We explored some of those capabilities in past work (published as US patents):

- (i) *Shared Risk Link Groups* (SRLG) refers to groups of links that may fail together because they share physical properties. An example of SRLG may be a bundle of optical fibers that may be cut at the same time by a digging engine. [115]⁹ evaluates how ARCs and a novel data plane protocol can recursively solve the problem using Link Reversal techniques
- (ii) *Ring Topologies* are widely used in LANs and Service Provider network to sustain network redundancy. [87]¹⁰ shows how ARCs can be laid over rings and provide several layers of redundancy.
- (iii) *Autonomous Systems* (AS) are the building blocks of the Internet, typically running an Interior gateway protocol (IGP) inside and the Border Gateway Protocol (BGP) between them. The capability to balance load between egress border routers is a key function to avoid congestion inside to at the egress of the AS. Because Omega can be composed of multiple nodes, ARCs can be used to compute a redundant topology, and, combined with Segment Routing [21]¹¹, enable load balancing over multiple paths to multiple exits in the AS.

⁸Filsfils et al.: *Segment Routing Architecture*

⁹Thubert et al.: *Repair of Failed Network Routing Arcs Using Data Plane Protocol*

¹⁰Thubert, Bellagamba, and Anteuinis: *Hierarchical label distribution and route installation in a loop-free routing topology using routing arcs at multiple hierarchical levels for ring topologies*

¹¹Filsfils et al.: *Segment Routing Architecture*

Table 8.3 summarizes the additional benefits over distributed routing and classical SDN techniques:

	IS-IS / OSPF	IS-IS / OSPF + LFA / MRT	SDN/MPLS-TP Segment Routing	ARCs
Route Computation	Distributed	Distributed	Centralized	Centralized
Routing model	Source Oriented. Routes computed from this node to all destinations.	Source Oriented. Re-route computed per node per destination	TP is Path Centric. Alternate path computed per path.	(Anycast) Destination Centric. Routes are computed per set of equivalent destinations. Re-route included in computation.
Fast Reroute	None	1 breakage per network	TP solves 1 breakage per alternate path	Multiple breakages per path (1 breakage per ARC, more with OAM control protocol)
Microloops	Yes	No	No	Loopless on multiple failures
Load Management	Routing cost update and re-convergence	Routing cost update and re-convergence	Engineered SR per flow at ingress	Automated in forwarding plane per ARC. No impact on routing plane.
SRLG	No	Need to configure dependency group	No	Automatic
Optim. for Rings	No	No	No	Yes
Multi-path / Multi-Exit transit	No	No	No	Yes

Figure 8.3 – Comparing Fast Reroute technologies

Centralized diverse paths computation with Replication and Elimination

As discussed in chapter 8, there are a number of techniques in the art that can be leveraged to build a pair of non-congruent paths between a source and a destination. But in order to enable replication and elimination inside the network, there is an additional need to provide a number of bridges, or rendez-vous points, between the diverse paths, and new technology must be introduced.

Since we must avoid single points of failure, we want to build a topology with the general shape of a ladder based on two non-congruent paths, as opposed to that of chain of a sausages for instance. To achieve this, a number of interconnection paths must be setup to transport the frames between the two non-congruent paths.

9.1 Running an example of ARC Chain building

The application of ARCs to bi-casting for replication and elimination was presented in *Applying Available Routing Constructs to bicasting* [106]¹ and an associated patent [88]².

This section presents a variation of the OLAF [85]³ technique that is designed to form complex Tracks, which, in their simplest form, adopt the general shape of a ladder; the ladder has two parallel disjoint paths between a source and a destination, and then steps that connect the two paths and enable replication and elimination several times along the way.

This new mechanism is complemented in section 10.2 by one that schedules timeslots in an appropriate order, and whereby ACKs are optionally suppressed to save energy.

In figure 9.1 and 9.2, the adapted OLAF technique is applied to build an *ARC Chain* to protect a Track between A and Z with multiple replication and elimination points.

¹Thubert and Wijnands: *Applying Available Routing Constructs to bicasting*

²Thubert, Bellagamba, and Wijnands: *Bicasting using non-congruent paths in a loop-free routing topology having routing arcs*

³Thubert and Bellagamba: *Available Routing Constructs*

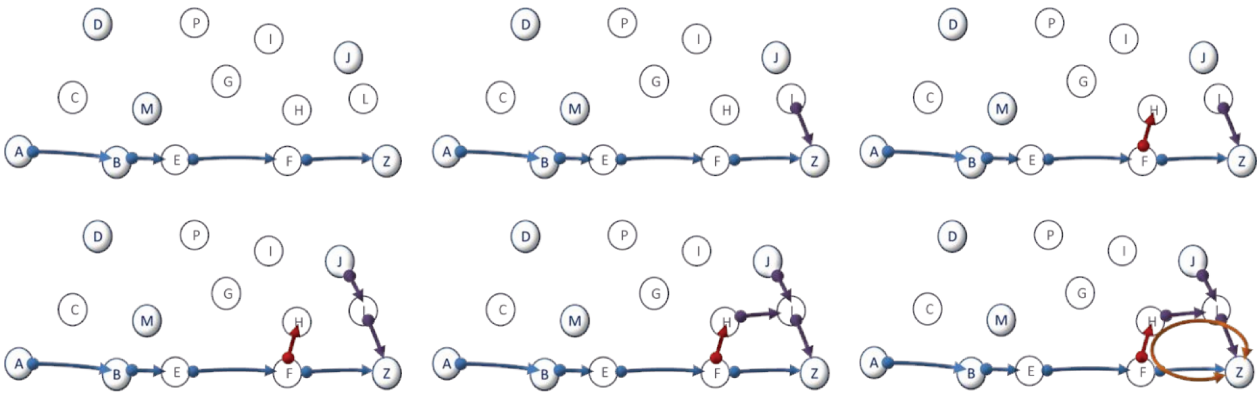


Figure 9.1 – building the first ARC

To start with, in figure 9.1, we build first ARC that terminates in node Z.

To build that ARC, we consider node F, which is the hop right before Z in the original sequential path from A to Z. We want to protect the link F to Z by an ARC that provides an alternate solution for F to Z.

With the adapted OLAF algorithm, we compute an (incoming, reverse) Shortest Path First (SPF) tree off the destination Z using incoming routing metrics.

In parallel, we grow F's SPF tree using outgoing metrics. As we run the SPF algorithm, we pick next the node that is closest to either F or Z and place it in the relevant tree.

We continue building the trees till they meet or the outgoing tree meets another existing ARC (but there's none at this stage).

Figure 9.1 depicts the trees as they grow, always picking first the node that is nearest to one of the trees, till finally, if the graph is effectively bi-connected, the trees eventually connect.

At that point, a node belongs to the two trees; in this case, this common node is node H. We can thus assemble our first ARC F, H, L making links FH and HL reversible.

The ARC Cursor may be placed on the node on the original path that is protected by the new ARC, node F in this example. It makes more sense to place it initially on the node that connects the trees, since it is roughly equidistant from both edges. In case of a breakage, a packet can be sent back towards the cursor to exit on the other side of the ARC, effectively crossing the steps of the ladder.

The set of *safe nodes* is now F, H, L, Z. These are nodes that, with original OLAF [85]⁴ technique, are already part of an ARCset, and thus can reach the destination even if there is a downstream breakage in the network.

From there, the process recurses, building trees from the nodes that are in the ARC we just formed, and connecting them to a tree off the previous hop in the original sequential path to Z.

⁴Thubert and Bellagamba: *Available Routing Constructs*

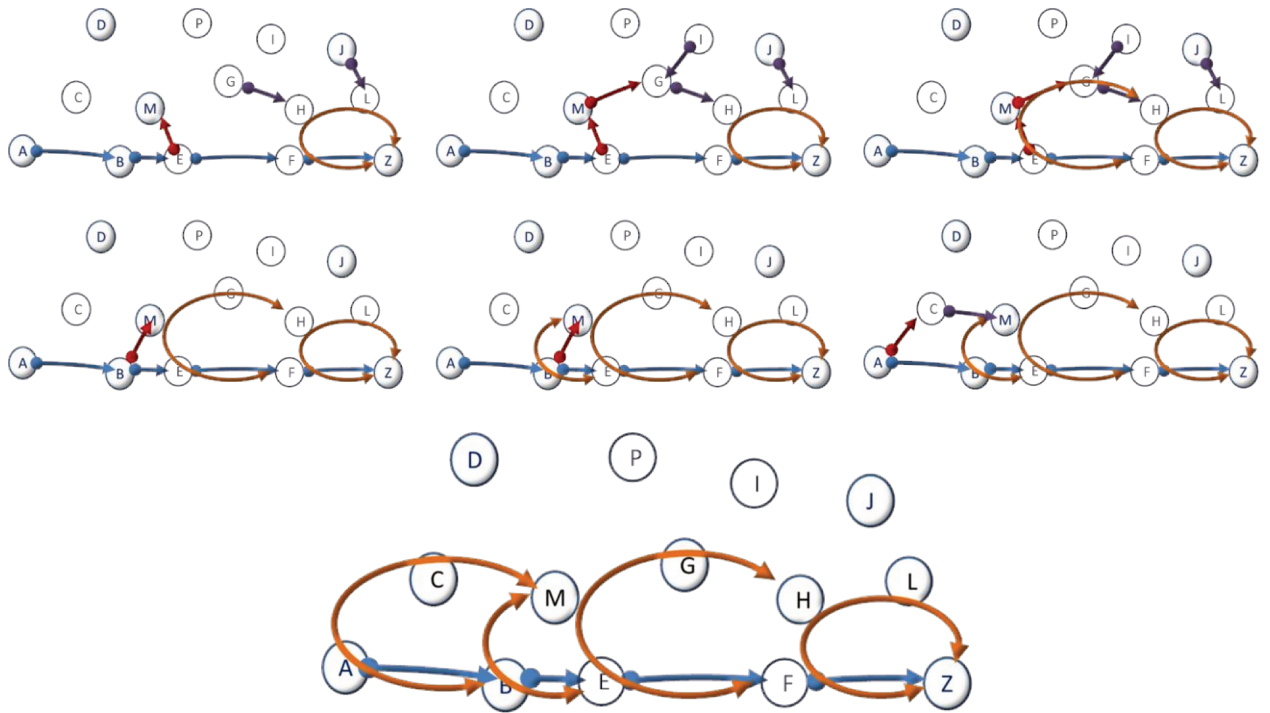


Figure 9.2 – Completing the ARC chain

This is illustrated in figure 9.2, where we continue building ARC after ARC, each ARC designed to safely reach different points in existing ARCs.

At this step, we keep the nodes from the previous ARC that are not on the original path from A to Z, H and L. We start again growing incoming trees from them, while also building outgoing tree off node E.

With this, we can build the second ARC E, M, G, H making links EM and MG reversible.

Then we iterate for B to E; again we keep the nodes from the previous ARC that are not on the path, M, G, and H, in this case, and grow incoming trees.

In fact, B connects to M right away so we have a collapsed ARC, meaning an ARC made of a single node with multiple safe possibilities.

Next we iterate for A to B; again we keep the nodes from the previous ARC that are not on the path, which means M only, this time. The process completes when the source node A is a safe node, meaning that we have formed an ARC chain from A to Z.

Note: In the algorithm, we do not need to limit the construction of a new ARC to trees rooted at nodes that belong to the most recently built one. We can also keep nodes in earlier ARCs if no solution is found rapidly rooting trees at the last ARC only. For instance, there might have been a node B' such that A to E is a lot shorter via B' than it is via B. Starting a tree off C and E would have allowed to bypass B.

Above a certain cost threshold it can be admitted that there is no backup for the node in the original path and in that case we include it as well as a root for a tree.

Computing and scheduling time slots

10.1 Forwarding rules

The expectation is that a central computer, such as a Network Manager or a System Manager used in existing industrial deterministic WSNs, will compute the ARC chain and then set up forwarding steps in each node in the form of a schedule. The schedule in the node may be complex, and will be represented in the form of logical ‘AND’ and ‘OR’ operation between time slots, ‘OR’ meaning that a successful completion for one of the OR’ed steps removes the need to perform the other OR’ed steps.

Due to operations such as retries and elimination, a same packet may be received at any of several time slots in each iteration of the schedule, resulting in an ‘OR’ operation. That packet is then forwarded at some other time slots, using ‘OR’ operations between equivalent time slots for retries, and ‘AND’ operations between timeslots that reach different next hops and are used for replication.

This way 6TiSCH uses that concept is discussed in more details in section 4.1.2. *Replication, Retries and Elimination of 6TiSCH requirements for DetNet* [81]¹ and section 6.4 on page 53 details how bundles of cell can be related to these logical operations.

If the cause of a loss is linked to the transmission channel, for instance in case of a cross-channel or a multi-path interference, a retry in the same conditions (same peers, channels, etc. . .) has an increased probability to fail as well, for the exact same reasons that caused the first loss. TSCH restores some of the chances by adding channel diversity, and the proposal below uses an additional spatial diversity to increase the chances again, using replication and elimination inside ARCs along an ARC chain.

10.2 Detailed steps

Figures 10.1 and 10.2 depict the operation for the first ARC computed in section 9.1 on page 63 (the last in the chain) using rules that are further detailed in section 10.3.

On the top right of the picture, a summary of the arc and incoming links from upper ARCs are represented; arrows are represented in the same direction on that picture and in the schedule below that is progressing stepwise below, each step a timeslot.

¹Thubert: *6TiSCH requirements for DetNet*

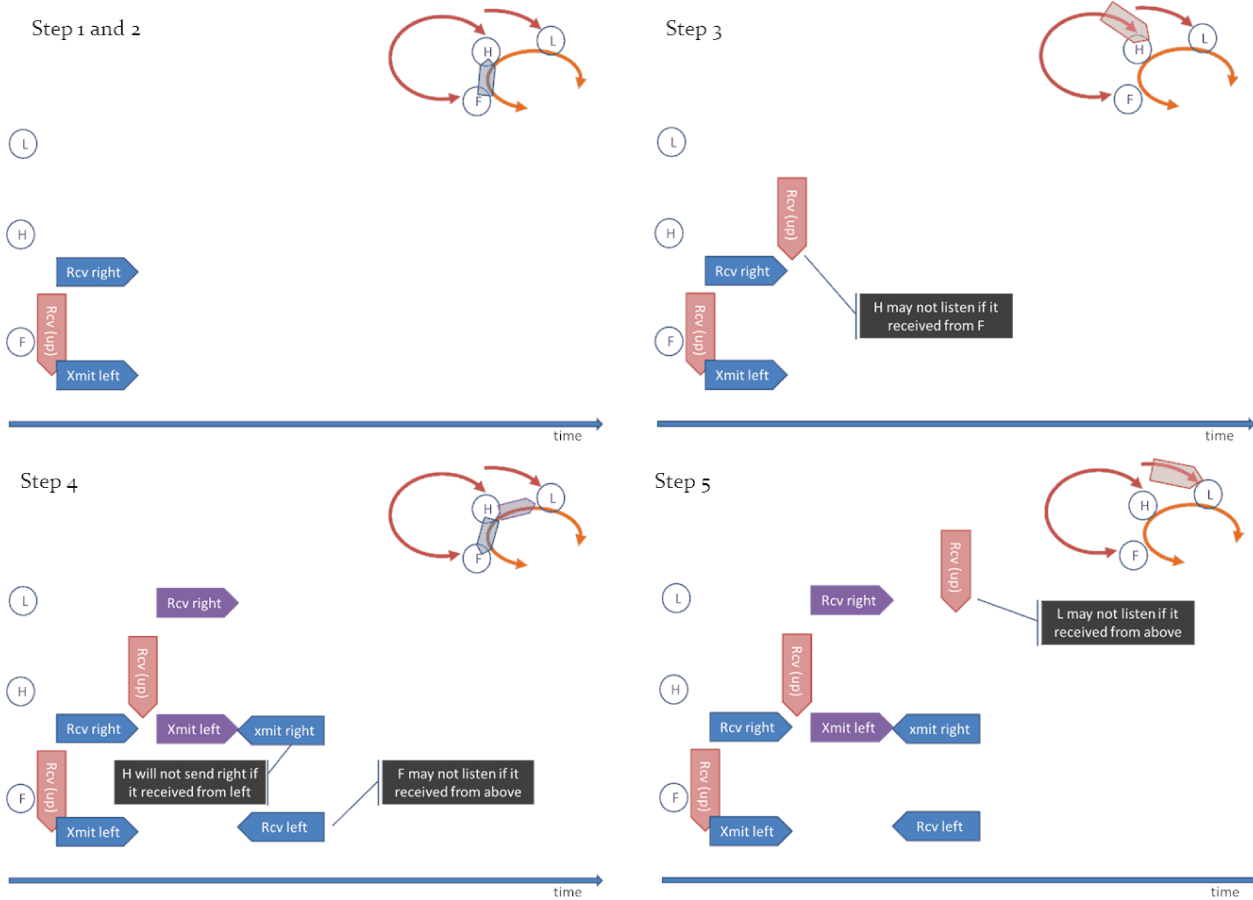


Figure 10.1 – Assigning time Slots steps 1 to 5

Two timeslots can actually be collapsed, either because a single transmission may serve two listeners (e.g. at T4), or because two different transmissions may happen at the same time but on different channels (e.g. at T8).

Step 1: node F receives a first copy of the packet; say that in the order of the schedule, F receives that copy at a timeslot T1, from a parent node in an upper ARC.

Step 2: node F forwards the packet along the ARC; F being at the right of its ARC, it is only programmed to send one copy (to the left) at time T2. This means that from the perspective of node H, H is scheduled to receive the frame (from its right in the figure) at the timeslot when F transmits the frame (to its left).

Step 3: H receives a first copy of the packet at timeslot T3; Node H may or may not have received F's copy, depending on interferences, multipath fading etc. . . If it did, then it may save the cost of listening to the next copies ('OR' operation in the schedule).

Step 4: H forwards the packet along the ARC at timeslot T4 (and T4bis). Being in the middle of the ARC, H has a transmit schedule on both sides. If the timeslot is used in shared mode, T4 and T4bis could effectively be the same timeslot.

Step 5: L receives a first copy of the packet at timeslot T5. In order to reduce the latency, timeslot 5 could be scheduled at the very same time as timeslot T4bis, as long as it is using a different channel; alternatively.

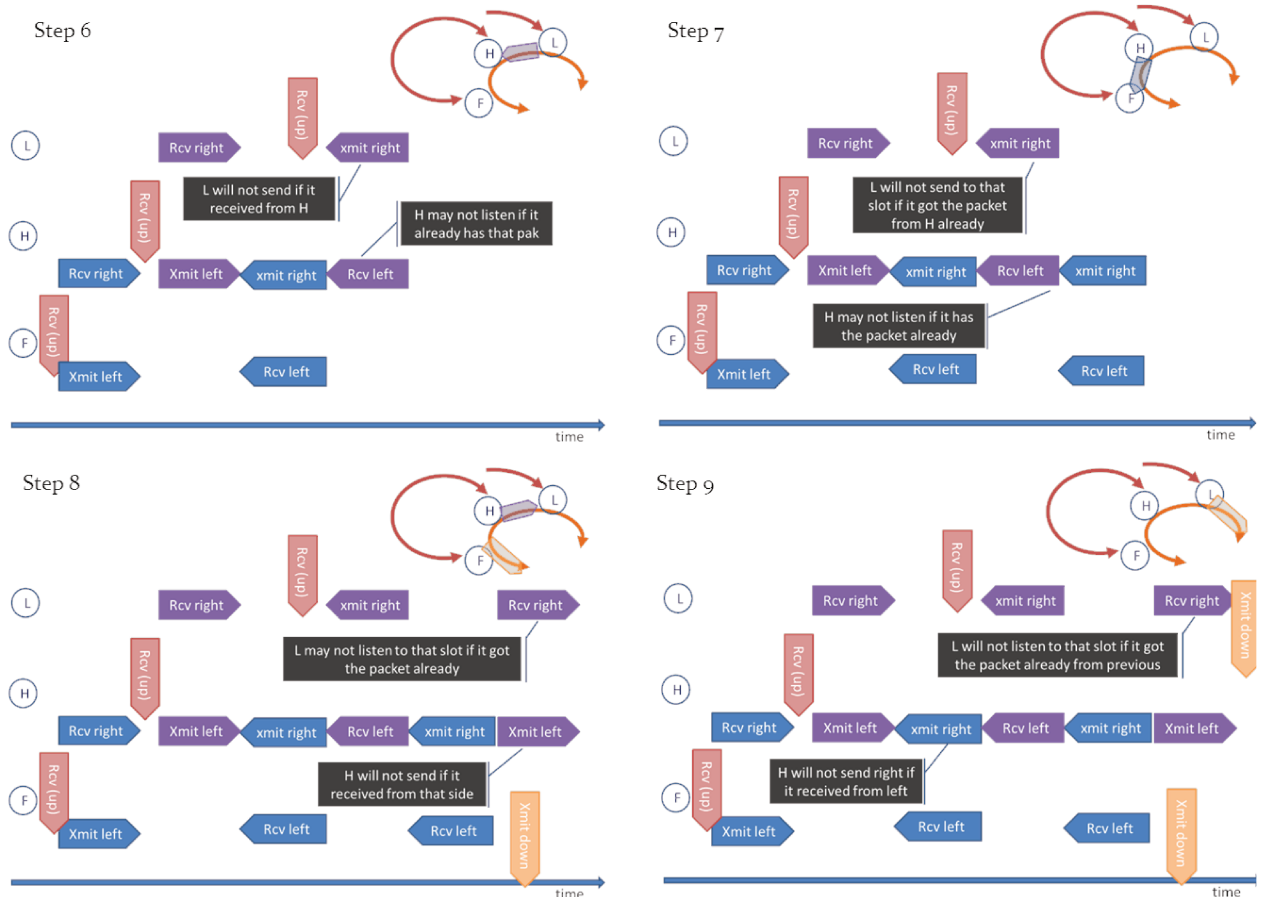


Figure 10.2 – Assigning time Slots steps 6 to 9

Step 6: L forwards to H at timeslot T6 L refrains from forwarding to H if it already got the same frame from H, which would have happened in timeslot T4. This suppression is the effect of the new rules which are introduced with this new technique.

Step 7: H forwards right, if still needed, at timeslot T7

Step 8: H forwards left, if still needed, at timeslot T8 At time Slot T8bis, which can be the same as timeslot T8, but for a transmission on a different channel, and since all of its receive opportunities are consumed, F can send down to destination Z (in an intermediate ARC that would be the next ARC).

Step 9: L can also send down, at timeslot T9 At timeslot T9, since all of its receive opportunities are consumed, node L can send down to destination Z.

This schedule can be expressed for each node in the ARC as follows:

Node	T1	T2	T3	T4(bis)	T5	T6	T7	T8(bis)	T9
Upper	XMIT		XMIT		XMIT				
L				RCV	RCV	XMIT		RCV	XMIT
H		RCV	RCV	XMIT		RCV	XMIT	XMIT	
F	RCV	XMIT		RCV			RCV	XMIT	
Z								RCV	RCV

10.3 Computation rules

The exact times or exact channels that are being used at each hop can actually be chosen by the central computer - a PCE. Depending on the number of flows that have to be accommodated, the PCE has more or less degrees of liberty to find a solution that fits all the requirements for all the paths. The overall problem is similar to that of Time Triggered Ethernet in the wired space, known to be NP-Complete.

The sequence has to be computed so that the packets progress towards the destination, and to ensure that all possible chances of reception have happened before attempting transmission; and so is the overall maximal latency for a certain frame, which is implemented as the delay between the first injection of the frame inside the ARC chain by A, and the last chance that Z had to receive the frame, in this scenario from L.

With this patented technique [101]², the central computer respects the rules summarized below to compute the schedule in each node (and the ‘AND’ and ‘OR’ operations):

- A scheduled frame or packet is associated to a number of “receive” and “transmit” time slots at any given node;
- From the perspective of a node, the first slot is a “receive” timeslot and the last is a “transmit” timeslot;
- There may be intermixed “receive” and “transmit” time slots, though having all the “receive” time slots first is preferable;
- A node in an ARC copies a frame received from a parent ARC as soon as possible inside the ARC, left and right, not down;
- But it does not copy in one direction if it already got the frame from that direction;
- And it may not listen on a “receive” timeslot if the frame was already received from any direction along the ARC.

If there is no retry scheduled, the Layer 2 ACK may be omitted, so both sender and receiver can go to sleep immediately after the transmission, which saves additional energy.

²Thubert, Vasseur, and Wetterwald: *Installation of time slots for sending a packet through an ARC chain topology network*

Controlling Replication and Elimination using BIER headers

This seminal idea for this chapter was presented as *OAM and Time Slot Control in a Deterministic ARC Chain Topology Network* [97]¹. It was then adapted to in-band signaling based on *Traffic Engineering for Bit Index Explicit Replication (BIER)*, which is introduced in section 2.3 on page 15, and in that shape, was proposed to the IETF Design Team for the DetNet data plane. This section was finally exported and published at the IETF as *BIER-TE-based OAM, Replication and Elimination* [89]².

11.1 Technical solution description

We leverage BIER-TE to activate the Replication and Elimination functions in a manner that is abstract to the data plane forwarding information. An adjacency, which is represented by a bit in the BIER header, can correspond in the data plane to an Ethernet hop, a Label Switched Path, a network segment [21]³, or it can correspond to an IPv6 loose or strict source routed path.

In a nutshell, BIER-TE is used as follows:

1. A controller computes a complex path, which 6TiSCH calls a Track. The complex path takes the general form of a ladder to enable Replication and Elimination between disjoint path with no single point of failure, as illustrated in figure 11.1 on the following page.

The analog of the steps of a ladder are the adjacencies that enable to copy packets from one path onto the other and recover intermediate losses, whereas the side rails are the main disjoint paths.

2. The steps, and the segments of side rails in-between steps, are the adjacencies. With the methods proposed in this chapter, the adjacencies can be activated on-demand and on a per-packet basis by setting or resetting bits in the BIER header.

¹Thubert, Vasseur, and hui: *OAM and Time Slot Control in a Deterministic ARC Chain Topology Network*

²Thubert, Brodard, and Jiang: *BIER-TE-based OAM, Replication and Elimination*

³Filsfils et al.: *Segment Routing Architecture*

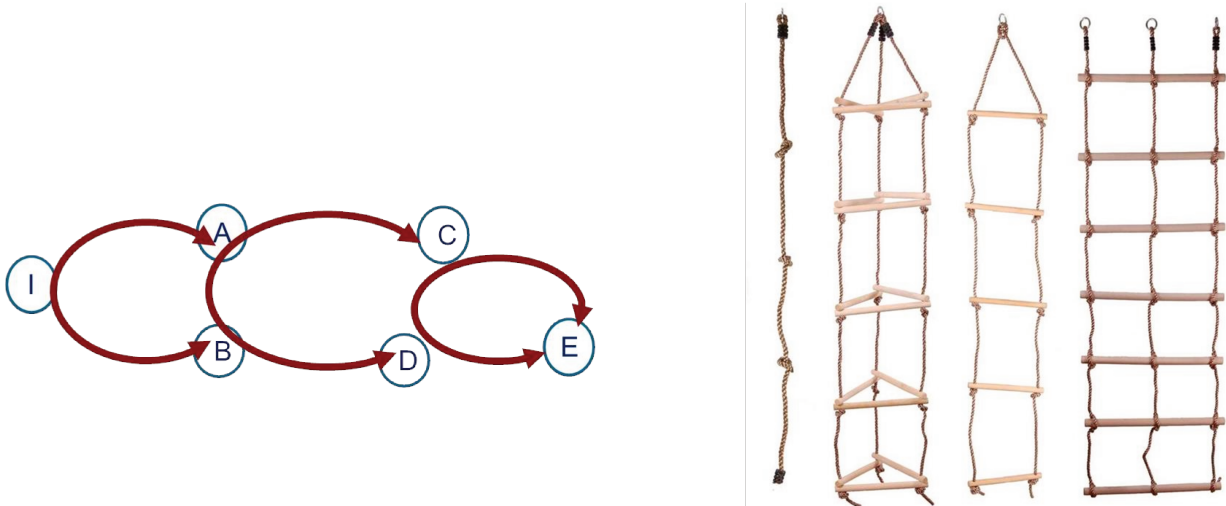


Figure 11.1 – The analogy of a ladder

3. The controller assigns a BIER domain, and inside that domain, assigns bits to the adjacencies as illustrated in figure 11.2. The controller assigns each bit to a replication node that sends towards the adjacency, for instance the ingress router into a segment that will insert a routing header in the packet.

A single bit may be used in both directions of a step of the ladder, indicating the other end of the adjacency.

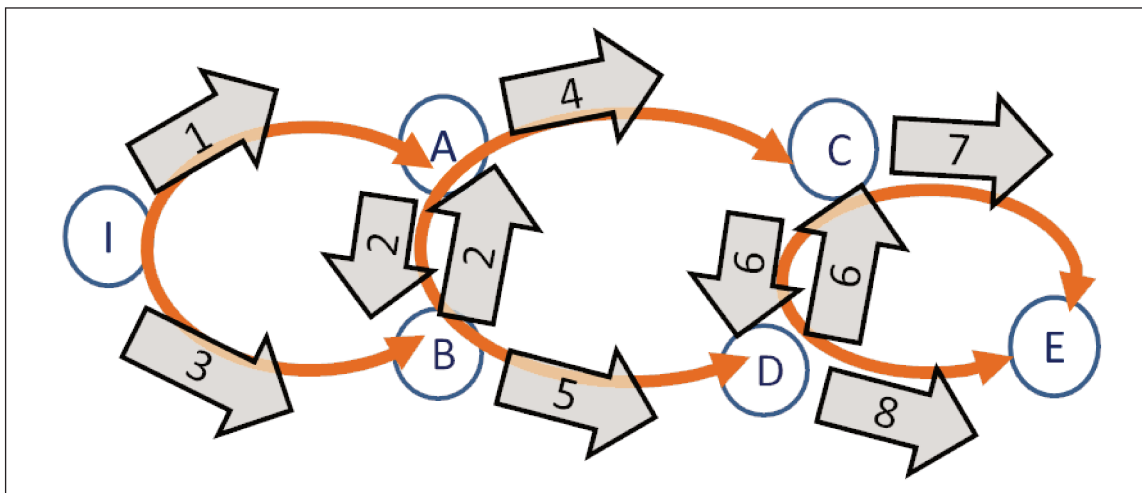


Figure 11.2 – Assigning bits to control Replication and Elimination

4. The controller activates the replication by deciding the setting of the bits associated with the adjacencies. This decision can be modified at any time, but takes the latency of a controller round trip to effectively take place.

Next is an example that uses Replication and Elimination to protect the $A \mapsto C$ adjacency.

Table 11.1 presents a setting whereby this method is used to protect segment $A \mapsto C$ by enabling a replication along $A \mapsto B$, $B \mapsto D$, $D \mapsto C$, and elimination in C.

Bit#	Adjacency	Owner	Example Bit Setting
1	$I \mapsto A$	I	1
2	$A \mapsto B$	A	1
	$B \mapsto A$	B	
3	$I \mapsto B$	I	0
4	$A \mapsto C$	A	1
5	$B \mapsto D$	B	1
6	$C \mapsto D$	C	1
	$D \mapsto C$	D	
7	$C \mapsto E$	C	1
8	$D \mapsto E$	D	0

Table 11.1 – Assigning bits to protect segment $A \mapsto C$.

The BIER-TE header with the controlling BitString is injected in the packet by the ingress node of the deterministic path. That node may act as a replication point, in which case it may issue multiple copies of the packet; in the example below, the bit settings make A and C Replication Points and C and D Elimination Points. But in the example, Elimination at D and Replication at C are pointless.

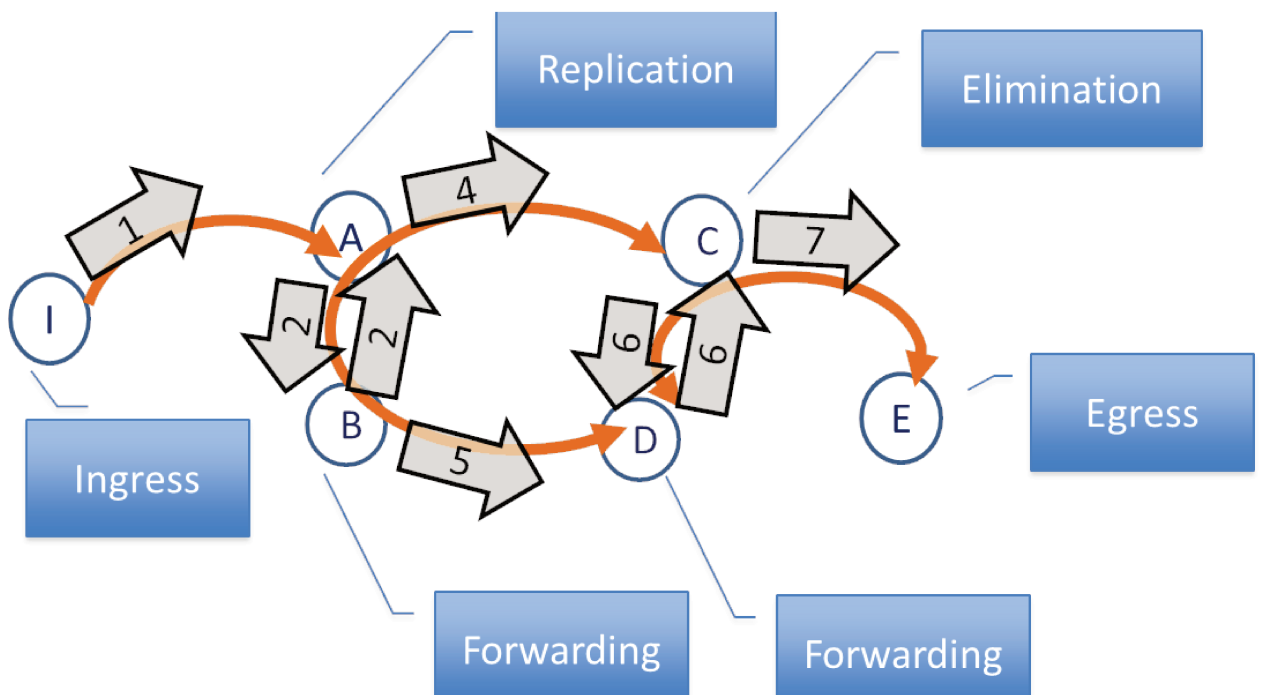


Figure 11.3 – Enabled adjacencies with bits 1, 2, 4, 5, 6 and 7 set

For each of its bits that is set in the BIER-TE header, the owner replication point resets the bit and transmits towards the associated adjacency as illustrated in figure 11.4.

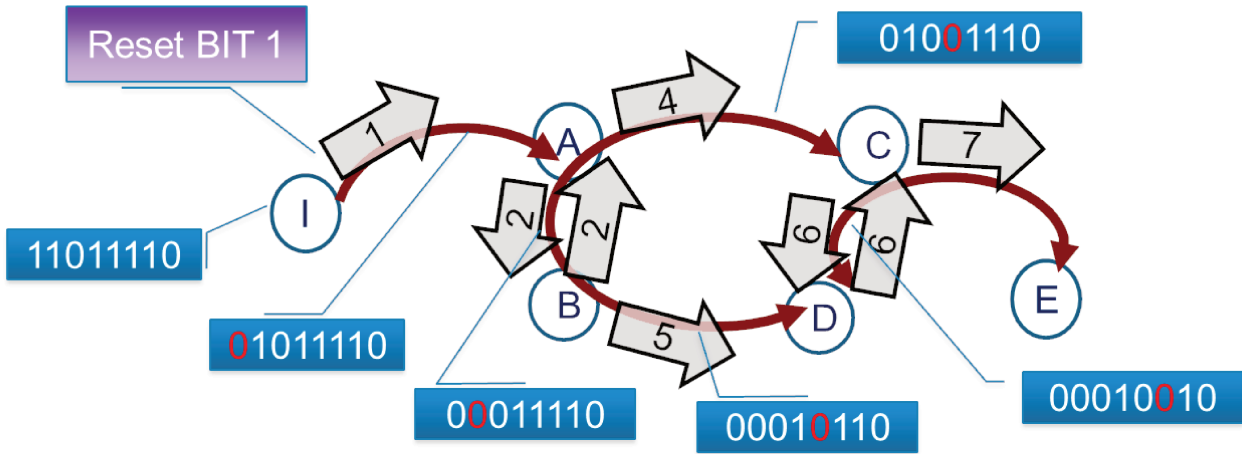


Figure 11.4 – Bitwise operation down the path

To achieve this, the replication point copies the packet and inserts the relevant data plane information, such as a source route header, towards the adjacency that corresponds to the bit.

For the example at hand, the table below indicates the BitString in BIER Header as the packet progresses along the ladder:

Adjacency	BIER BitString
I ↦ A	01011110
A ↦ B	00011110
B ↦ D	00010110
D ↦ C	00010010
A ↦ C	01001110

It can be noted that each replica of the packet has a unique BitString which can be used as a signature of the replication point and the intent for that particular replica. This feature is a key enabler for trace-ability of the replication operation inside the network, e.g., if a packet strays away from the Track, this information can be used as an indication of the source of the last alteration to the packet.

Adversely, an elimination node on the way strips the data plane information and performs a bitwise AND on the BitStrings from the various copies of the packet that it has received, before it forwards the packet with the resulting BitString.

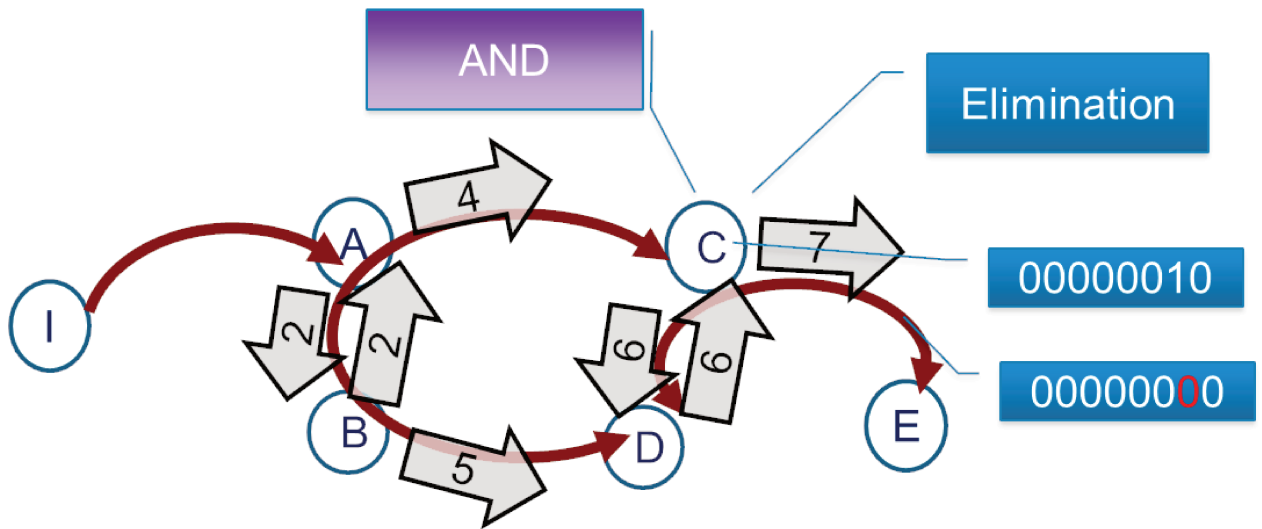


Figure 11.5 – AND operation at the elimination point

The table below illustrates the BitString Processing at Elimination Point C:

Operation	BIER BitString
D → C	00010010
A → C	01001110
AND in C	00000010
C → E	00000000

In this example, all the transmissions succeeded and the BitString at arrival has all the bits reset - note that the egress may be an Elimination Point in which case this is evaluated after this node has performed its AND operation on the received BitStrings).

But if a transmission failed along the way, one (or more) bit is never cleared. The table below provides the possible outcomes of a transmission:

Failing Adjacency	BIER BitString at Egress
I → A	Frame Lost
I → B	Not Tried
A → C	00010000
A → B	
B → D	01001100
D → C	
C → E	Frame Lost
D → E	Not Tried

If the frame is lost, then it is probably due to a failure in either $I \mapsto A$ or $C \mapsto E$, and the controller should enable segments $I \mapsto B$ and $D \mapsto E$ to isolate the problem with the precision of an adjacency.

A BitString of 00010000 indicates unequivocally a transmission error on the $A \mapsto C$ adjacency, whereas a BitString of 01001100 indicates a loss in either $A \mapsto B$, $B \mapsto D$ or $D \mapsto C$. Enabling $D \mapsto E$ on the next packets may provide more information for the determination of the problem.

11.2 Additional details

The BIER header is of variable size, and a DetNet network of a limited size can use a model with 64 bits if 64 adjacencies are enough, whereas a larger deployment may be able to signal up to 256 adjacencies for use in very complex paths.

For the DetNet data plane, a replication point is an ingress point for more than one adjacency, and an elimination point is an egress point for more than one adjacency.

A pre-populated state in a replication node indicates which bits are served by this node and to which adjacency each of these bits corresponds. With DetNet, the state is typically installed by a controller entity such as a PCE. The way the adjacency is signaled in the packet is fully abstracted in the bit representation and must be provisioned to the replication nodes and maintained as a local state, together with the timing or shaping information for the associated flow.

The DetNet data plane uses BIER-TE to control which adjacencies are used for a given packet. This is signaled from the path ingress, which sets the appropriate bits in the BIER BitString to indicate which replications must happen. The replication point clears the bit associated to the adjacency where the replica is placed, and the elimination points perform a logical AND of the BitStrings of the copies that it gets before forwarding.

As is apparent in the examples above, clearing the bits enables to trace a packet to the replication points that made any particular copy. BIER-TE also enables to detect the failing adjacencies or sequences of adjacencies along a path and to activate additional replications to counter balance the failures.

Finally, using the same BIER-TE bit for both directions of the steps of the ladder enables to avoid replication in both directions along the crossing adjacencies. At the time of sending along the step of the ladder, the bit may have been already reset by performing the AND operation with the copy from the other side, in which case the transmission is not needed and does not occur (since the control bit is now off).

Compensating loss with overhearing: the Leapfrog Collaboration

So far we studied ways to build a redundant path with Replication and Elimination, program it in a TSCH network and then signal dynamically which adjacencies are to be walked through on a per packet bases. This provides a deployable solution to realize a Deterministic Network over a wireless TSCH medium, but is energy greedy and some optimization should be found in order to address a larger scope of application.

This chapter proposes a novel variation of promiscuous listening *-overhearing-* techniques that allows two parallel paths to protect one another within one Track. In other words, the chances of loss in radio transmission along one path are counterbalanced by another radio property which produces redundancy on the receiver side.

12.1 Related work

Experimentation has proven that simple transmission over IEEE Std 802.15.4 without retries, though it is permitted by the standards, yields a high packet error rate [76]¹.

As shown in previous studies [47]², the mechanisms used to improve reliability in WSNs are typically either based on retransmission or on redundancy. The most widespread paradigm is retransmission and acknowledgement, aka ARQ, for Automatic Repeat re-Quest, also known as Automatic Repeat Query. ARQ at Layer 2 is compatible with the use of an IP stack above, but care must be taken on how its implementation at various layers of the network interact with one another [18]³. The principle remains at all layers to detect the transmission failure (often by a gap between transmissions or from an acknowledgement timer expiration) and try that transmission again, until a positive acknowledgement is received, or a maximum amount of retransmission attempts have failed, in which case the connection is dropped.

Although ARQ is a simple and quite efficient mechanism, the technique falls short for deterministic WSNs. On the one hand, retransmission leads to additional delays and jitter that may affect the transport protocol or the application. On the other hand, re-

¹Shu et al.: “Packet loss analysis of the IEEE 802.15. 4 MAC without acknowledgements”

²Mahmood, Seah, and Welch: “Reliability in wireless sensor networks: A survey and challenges ahead”

³Fairhurst and Wood: *Advice to link designers on link Automatic Repeat reQuest (ARQ)*

transmitting a packet consumes energy for both the sender and the receiver and utilizes the spectrum without ensuring that this sending will be more successful than the previous one.

In order to alleviate the energy consumption of retransmission mechanisms, several pieces of work used overhearing to implicitly acknowledge the packet reception. For example, Lee and Huh [41]⁴ use implicit acknowledgement by overhearing when the receiver in turn forwards the packet. Maalel, Roux, Kellil, and Bouabdallah [45]⁵ enhanced the mechanism with spatial diversity by maintaining an ordered list of neighboring node susceptible of retransmitting the packet.

In his PhD thesis, Lampin [40]⁶ suggests to leverage overhearing along a wireless graph with a particular addressing to skip one hop if the next hop already heard a packet. A similar technique was envisioned for application in Wi-Fi in *Path shortening in a wireless mesh network* [114]⁷. The two concepts that were missing at the time were the 6TiSCH Track to define the overhearing nodes and the schedule to optimize the operation by having only the right nodes listen at the right time as the packet progresses along the Track.

Overhearing is also widely used to address security issues as shown by Alghamdi, Wu, Zheng, and Kanhere [1]⁸.

12.2 Introducing the Leapfrog Collaboration

In our model, a controller computes (at least) two parallel or interleaved paths for one Track, such that the nodes on one path may promiscuously overhear transmissions along the other path. For a scheduled transmission from a node X to a node Y, a listen-only time slots is scheduled in at least one alternate node Z that is also in the complex Track and preferably closer than X to the destination. This entry is added to Z's normal forwarding schedule to give it an additional chance to receive a frame by overhearing transmissions on the other path before its turn to transmit.

It results that globally, the schedules in the devices are organized with more receive slots than transmit slots, and that the extra receive slots are configured to refrain from generating an acknowledgement, since the extra receiver is overhearing and is not the intended recipient, and as optional, so the node listens only if it did not already receive the packet in a prior receive slot.

⁴Lee and Huh: “Reliable data transfer using overhearing for implicit ack”

⁵Maalel et al.: “Adaptive reliable routing protocol for wireless sensor networks”

⁶Lampin: “Urban wireless sensors networks : Applications and protocols”

⁷Thubert et al.: *Path shortening in a wireless mesh network*

⁸Alghamdi et al.: “Constructing a Shortest Path Overhearing Tree with Maximum Lifetime in WSNs”

12.3 Running an example of Leapfrog Collaboration

In the example illustrated in figure 12.1, we assume that due to radio transmission range and instabilities, the chances to receive decrease rapidly with the distance, e.g., B receives roughly 80% of A's transmissions, but C only gets 35% of them, D receives roughly 75% of C's transmissions, but E only gets 40% of them, etc...

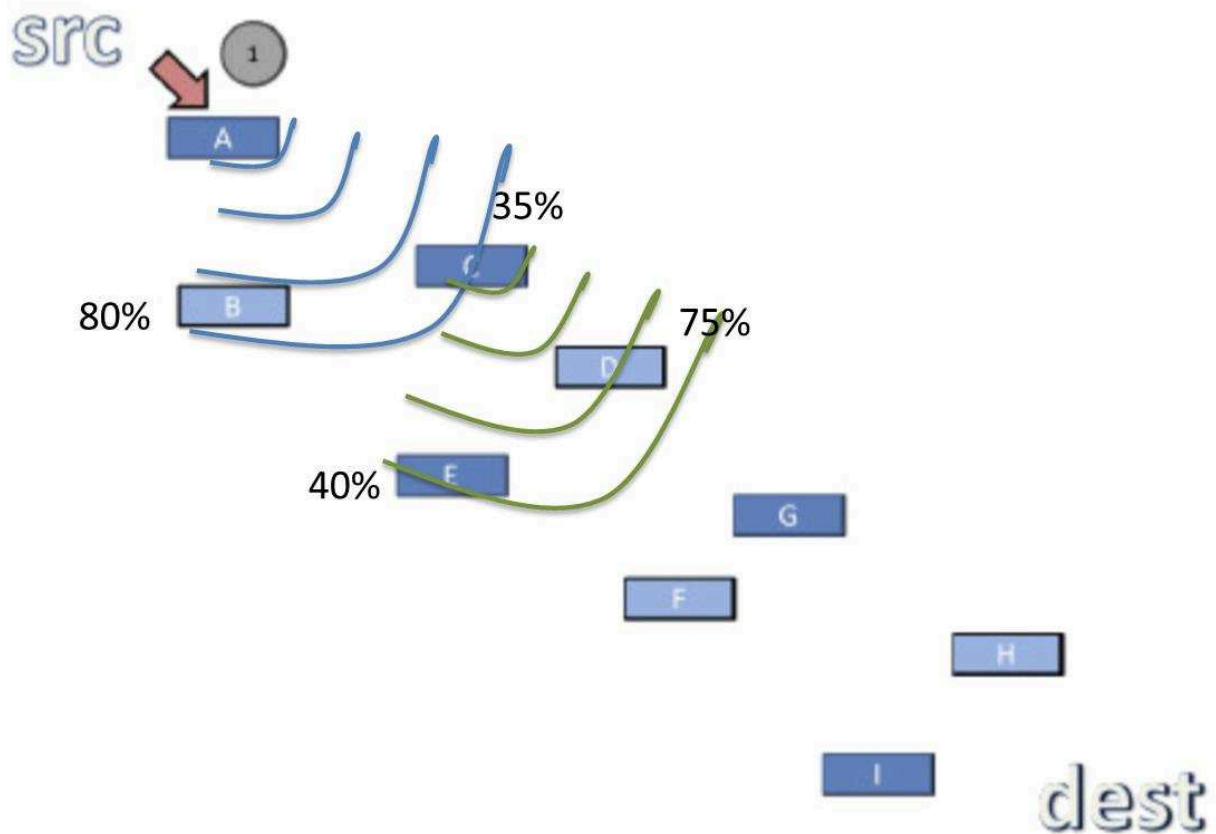


Figure 12.1 – Example PDR for a LeapFrog Track

In this example, our Leapfrog Collaboration builds two parallel paths in a Track between a source *src* and a destination *dest*, via intermediate nodes A, C, E, G, I and via B, D, F and H, respectively. As illustrated in the top left of figure 12.2 on the following page, we compute the ladder as a double helix, making it so that the transmission across the steps is actually better than that along the rails. Whereas the art of frame Replication and Elimination would blindly forward a copy of the frame on each of these paths, the proposed Leapfrog Collaboration takes advantage to the shared properties of the medium to compensate for the additional loss that is incurred with radios by leveraging overhearing by other nodes than the expected destination of the transmission.

A schedule is set up with promiscuous listening between the paths, one path being operated on even time slots and the other on odd time slots. In order to save additional energy, a policy is set up to send on an even timeslot only upon detection of a failure on the preceding odd timeslot, as observed from a missing overheard L2 Ack.

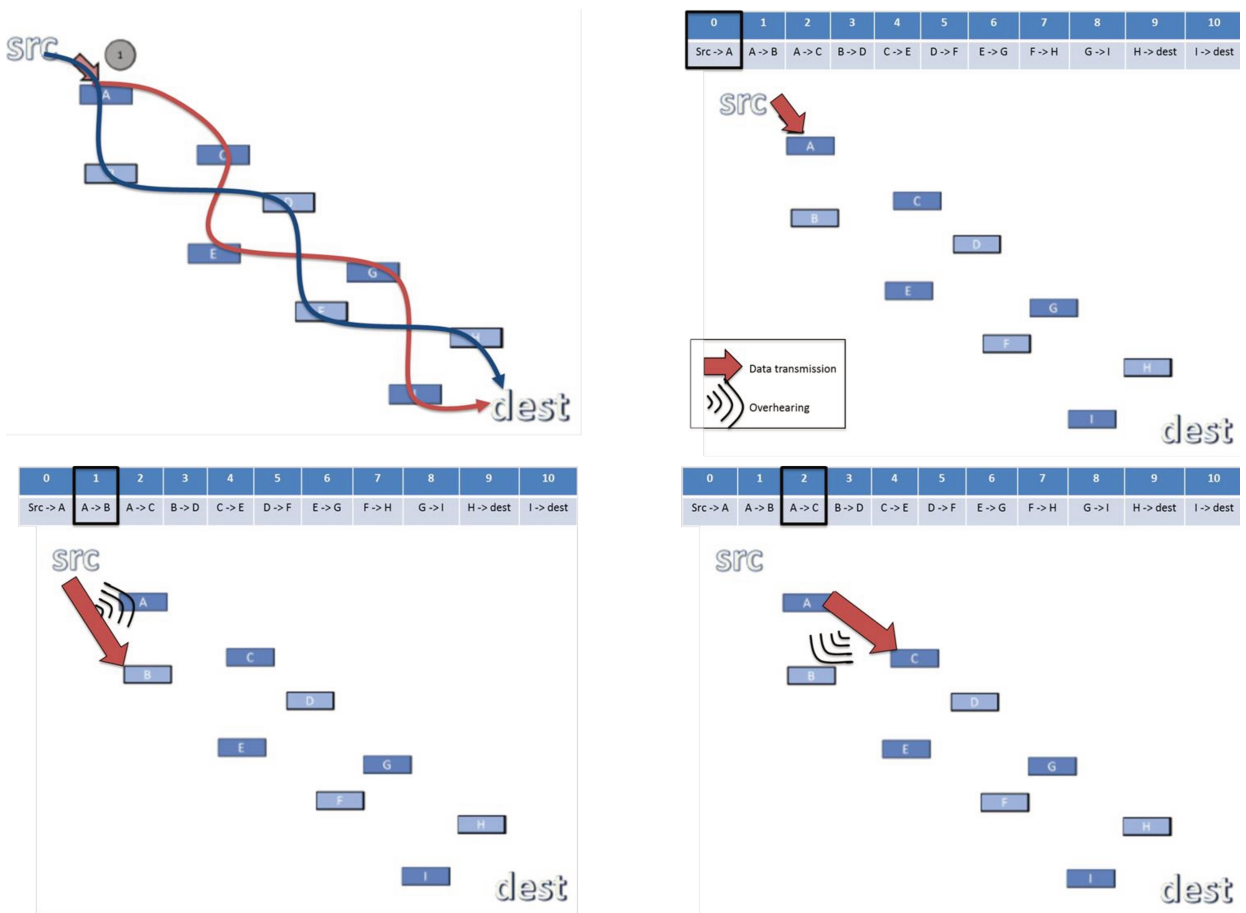


Figure 12.2 – LeapFrog: Complex Track and initial forwarding steps

Figures 12.2, 12.3 on the facing page and 12.4 on page 82 illustrate the double helix and the progress of the packet for each timeslot in the schedule from timeslot 0 to 10.

Timeslot 0; src \mapsto A:) A first copy is injected by the source to node A, which is the node nearest to the source and the ingress point for one of the paths. Transmission on timeslot 0 is optional, considering that A has great chances to overhear the transmission from the source to B, and that there is no node overhearing node for it. Alternatively, B may be informed of that transmission and listen promiscuously for it.

Timeslot 1; src \mapsto B (alt. A): In this step, A is aware of schedule and if it did not get the frame in optional slot 1, so if A failed to receive the packet in Timeslot 0, then it listens promiscuously to the transmission from source to B. It is a second chance for A to get the packet but does not cost an extra transmission from src as a normal ARQ would.

Timeslot 2; A \mapsto C (alt. B): Then the process continues. A sends to C which is afar. Optionally, it is possible to save that transmission if A heard promiscuously the acknowledgement from B to the source, since B is expected to be nearer than A from C. Also, B will listen to that transmission only if it failed to receive in timeslot 1.

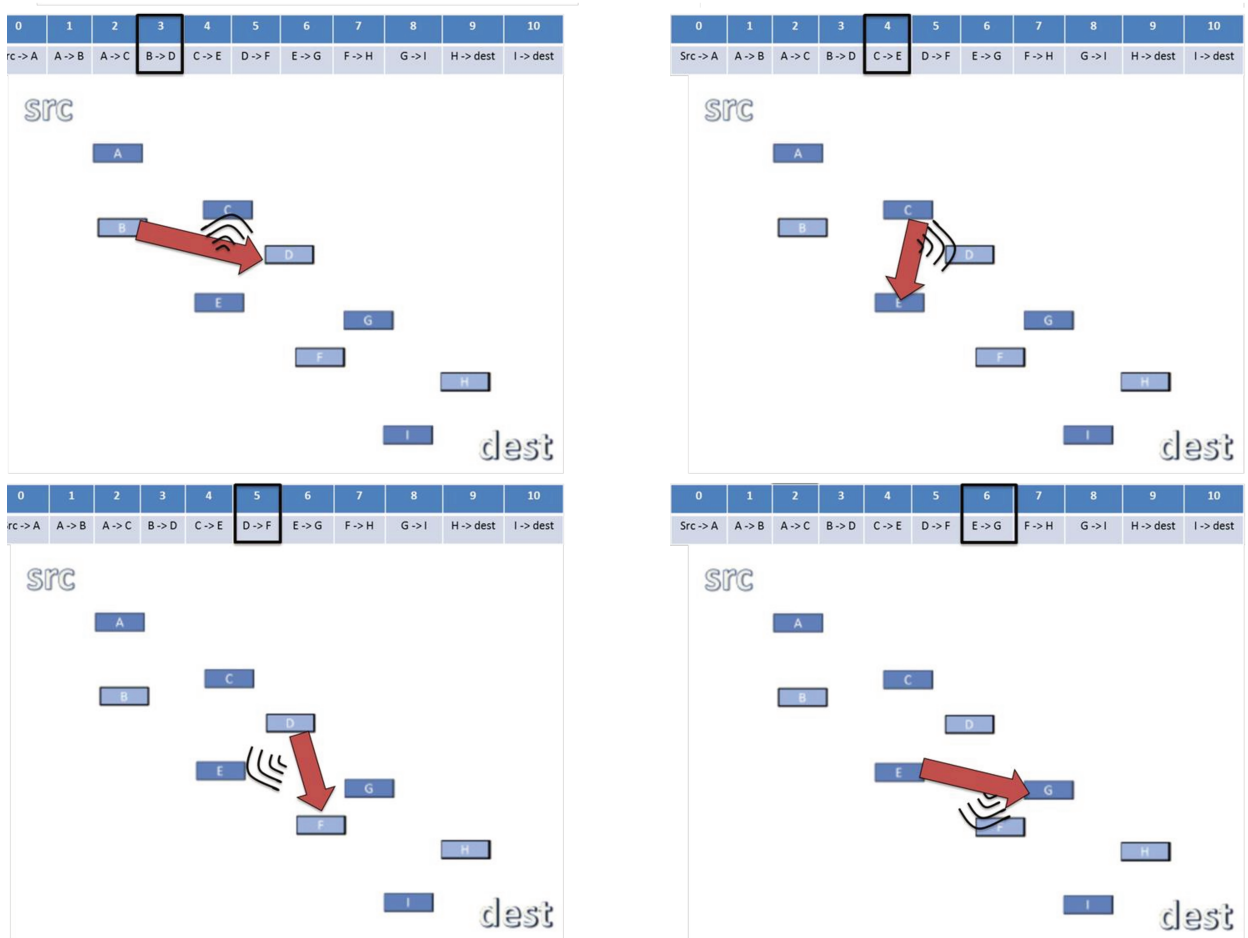


Figure 12.3 – LeapFrog: Intermediate forwarding steps

Timeslot 3; $B \mapsto D$ (alt. C): The same occurs in the parallel path: C is aware of schedule and listens promiscuously to the transmission from B to D.

Timeslot 4; $C \mapsto E$ (alt. D): As for all even timeslots, this transmission can be made optional, depending on an energy-conservation policy; if node C heard the Layer 2 Acknowledgement from D, then it knows that D received the frame and there is no point resending if D is configured to only listen if it did not already receive the frame from B.

Timeslot 5; $D \mapsto F$ (alt. E):

Timeslot 6; $E \mapsto G$ (alt. F):

Timeslot 7; $F \mapsto H$ (alt. G):

Timeslot 8; $G \mapsto I$ (alt. H): These steps show the 2 copies progressing towards the destination on the parallel path following the same logic as above.

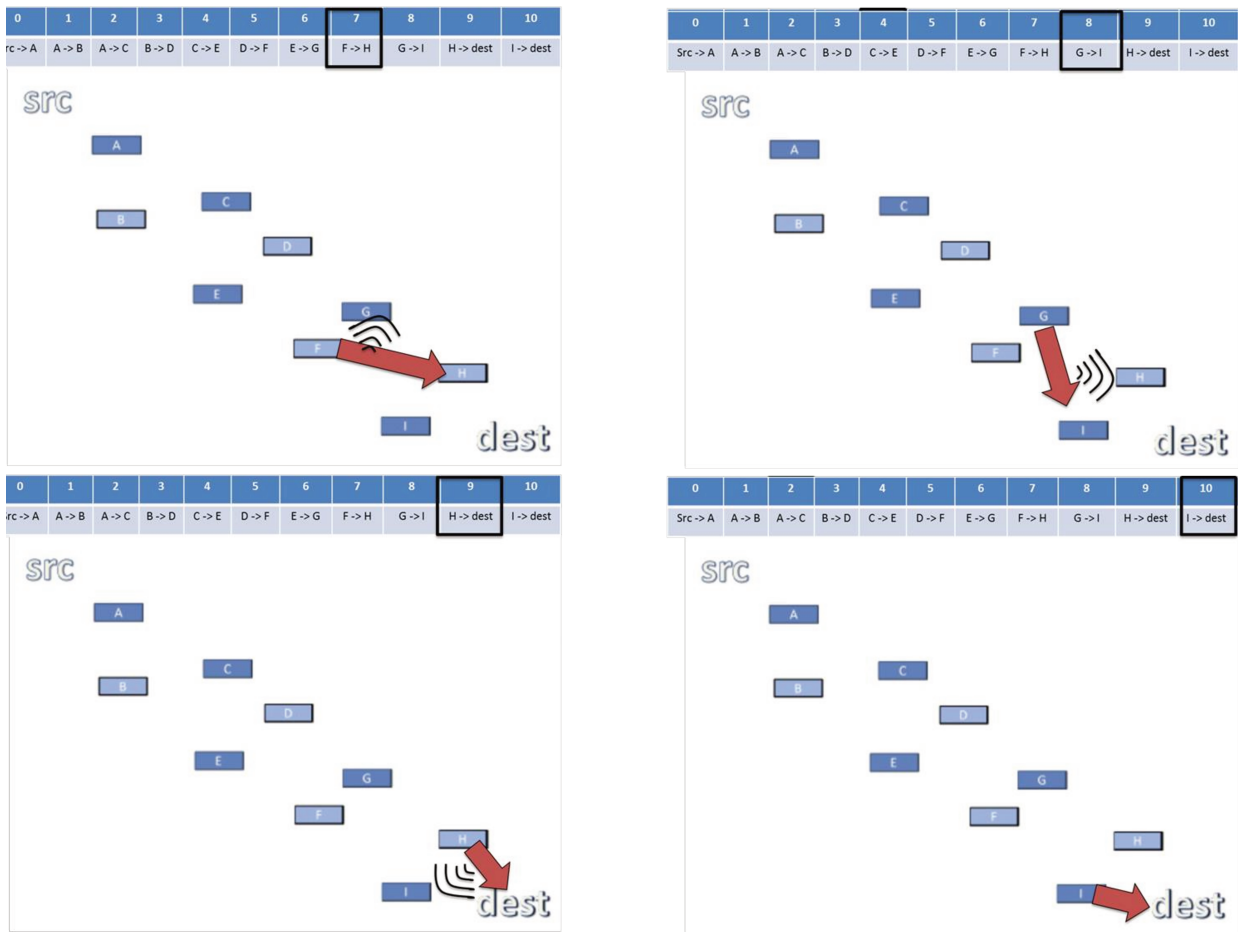


Figure 12.4 – LeapFrog: Final forwarding steps and delivery

Timeslot 9; H \mapsto dest (alt. I): The process continues until at some point the frame gets a first chance to promiscuously reach the destination dest.

Timeslot 10; I \mapsto dest: If node I did not overhear Destination dest’s acknowledgement to H on timeslot 9, then it sends its own copy of the frame to dest, and dest only listens if it did not get the frame from H at the earlier step.

12.4 Quick retrospective

Compared to an ARQ retry, the LeapFrog collaboration mechanism avoids both the jitter that is incurred with the variable number of transmissions, and the energy spent in retransmission. This is replaced by additional nodes overhearing promiscuously, and re-sending the snooped frame on behalf of the transmitter to improve the diversity, thus the chances of successful reception. In one embodiment, there are two parallel path snooping on one another; this mechanism yields higher chances of success but also a higher energy budget. In another, more energy-conservative embodiment, the overhearing node only repeats in case of a frame loss which is identified by a missing Acknowledgment.

12.5 Additional perspectives

In the track represented in the top left of figure 12.2 on page 80, there are only 2 parallel paths, and this corresponds to an alternate view on figure 12.5, but a more complex collaboration may be built as illustrated in figure 12.6.

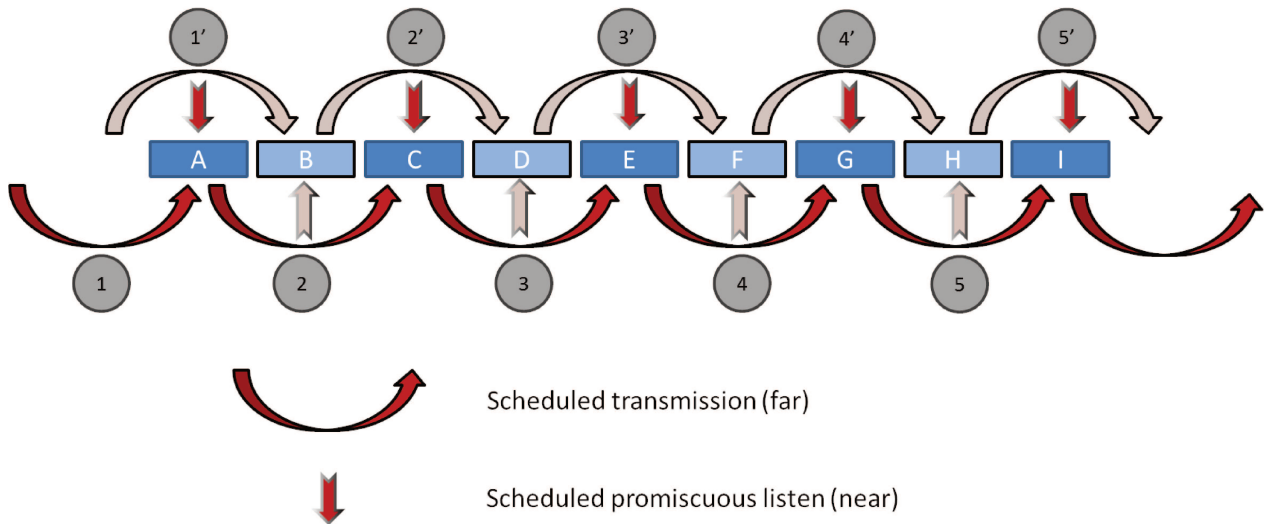


Figure 12.5 – Alternate view with only a plan B

Figure 12.6 illustrates an alternate solution with two overhearing nodes instead of one providing an additional redundancy. The benefit is an increase transmission success, at the expense of additional energy spent in promiscuous overhearing, plus possibly a longer utilization of a particular buffer if a frame is now received sooner that it was with the previous example.

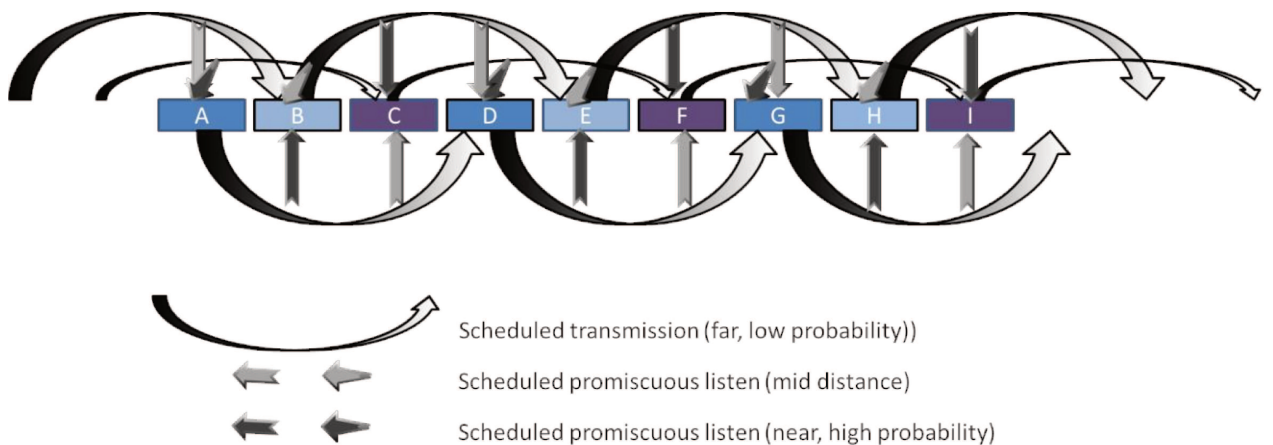


Figure 12.6 – Alternate view with an additional redundancy

PART IV

Experimentations

A warning on loss ratio

With the art of IEEE802.14.5 TSCH, we can build radio meshes that exhibit a packet delivery ratio similar to that of an Ethernet switched path, typically with a loss in the order of 10^{-5} . In order to experiment anything, we need a certain amount of loss in the transmission. This amount must be realistic, and yet, in order to run our test in reasonable times, we had to find ways to keep them orders of magnitude above the normal levels that would be experienced in a real industrial deployment. So we voluntarily placed our devices in crowded areas, and applied different types of errors and anomalies, which appear clearly in the resulting graphs. In turn, these anomalies affect some of the results, and in particular give the wrong image of the actual delivery ratio that can be obtained with the technologies that are being demonstrated here. This should be considered when reading this part.

Testing environment

13.1 The PIRL

This series of tests took place at the Paris Innovation Labs (PIRL) during summer 2016 and was performed by Zacharie Brodard and Jiang Hao under my lead, as part of their internship at Cisco. The physical layout is represented in figure 13.1.

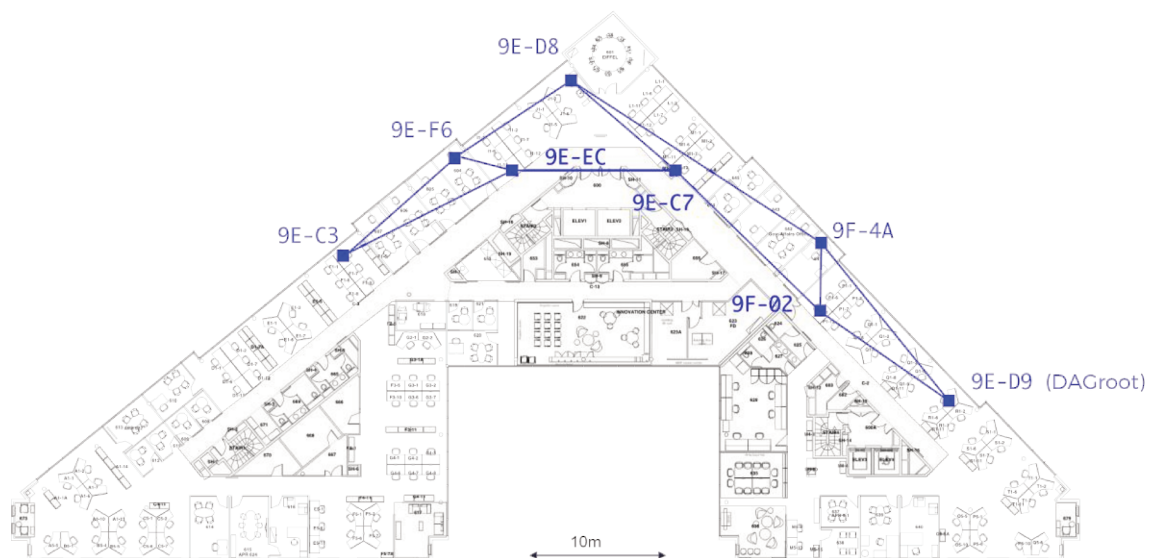


Figure 13.1 – PIRL, 6th Floor at Cisco’s ILM office near Paris

PIRL is an active office space with dense usage of Wi-Fi over the 2.4GHz band, we leveraged this as one of the high yet realistic sources of loss ratio that we are after. We run some tests during the day to generate loss that is due to co-channel interference, and others during the night to better control the causes of loss.

In any case, we keep the transmission power levels high, even if the size of the testbed is relatively limited. The reason for this is that even though maintaining the power low would generate some loss from the limit of the range, it would also and mostly remove the multipath fading, which is probably a more realistic a source of loss in a well-engineered deployment such as found in industrial Process Control.

13.2 Rover: the test tool

The experimentation was conducted on the testing device shown in figure 13.2; it is made of an OpenMote-cc2538 [123]¹² running OpenWSN [130]³⁴ and mounted on a Raspberry PI, which is connected to the lab network and is used to control and program the mote.

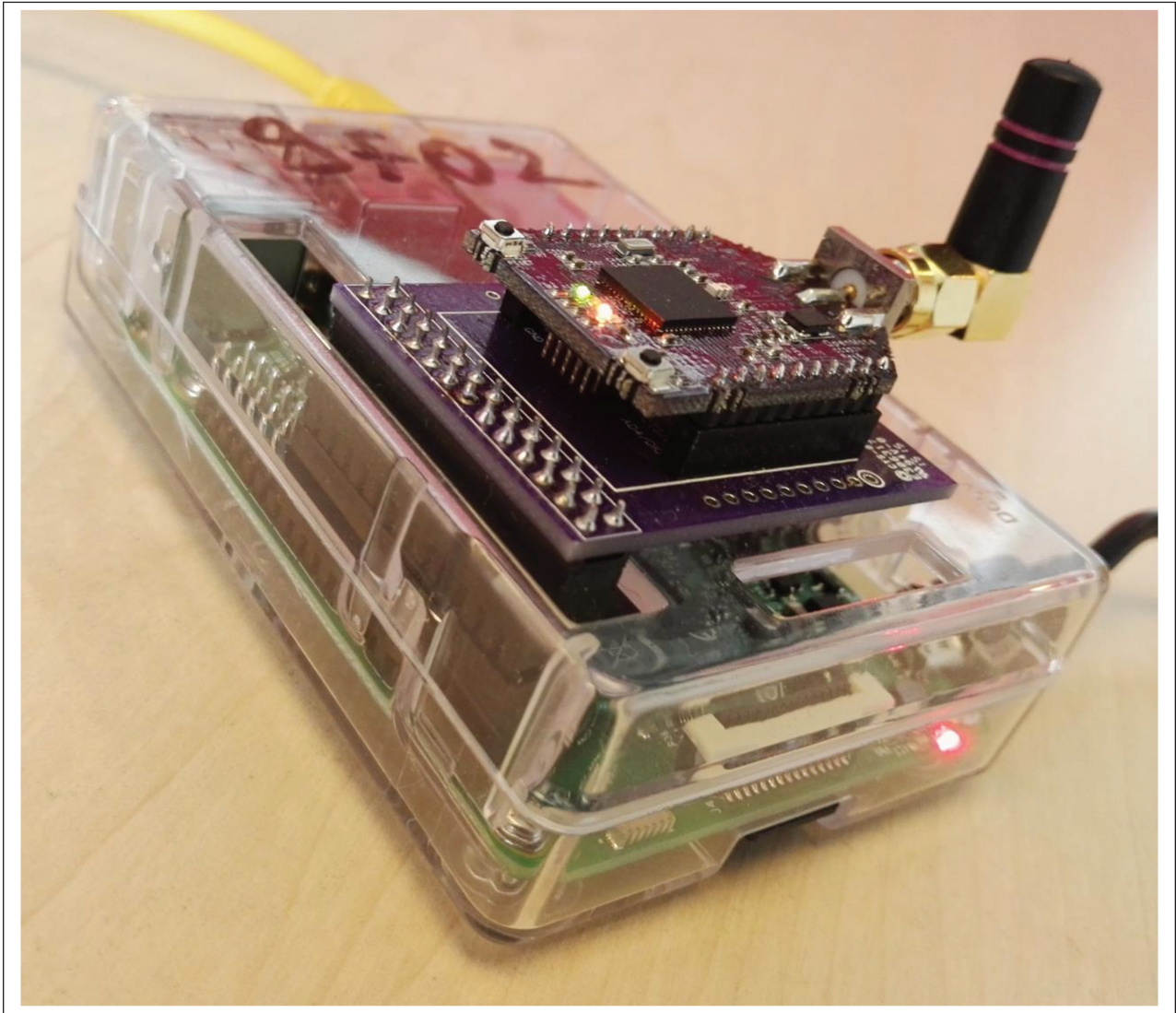


Figure 13.2 – Rover: an OpenMote mounted on a RPi

Both OpenMote and OpenWSN are open-source projects; and both took their root in Kris Pister’s lab at UC Berkeley, under the lead of Thomas Watteyne, Xavi Vilajosana and Pere Tuset.

OpenMote provides a relatively cheap and open platform for experimentation, and a number of OS were ported on it, including Contiki and OpenWSN.

¹Vilajosana et al.: “OpenMote: Open-Source Prototyping Platform for the Industrial IoT”

²www.openmote.com/hardware/openmote-cc2538-en.html

³Watteyne et al.: “OpenWSN: a Standards-based Low-Power Wireless Development Environment”

⁴openwsn.atlassian.net/wiki/pages/viewpage.action?pageId=688187

OpenWSN boasts an active support community and is maintained up to date with the latest 6lo, ROLL and 6TiSCH specifications; it was ported on a variety of software and hardware platforms including OpenMote and FreeRTOS. The OpenMote/OpenWSN combination was used at ETSI interop tests⁵ which took place in the recent years.

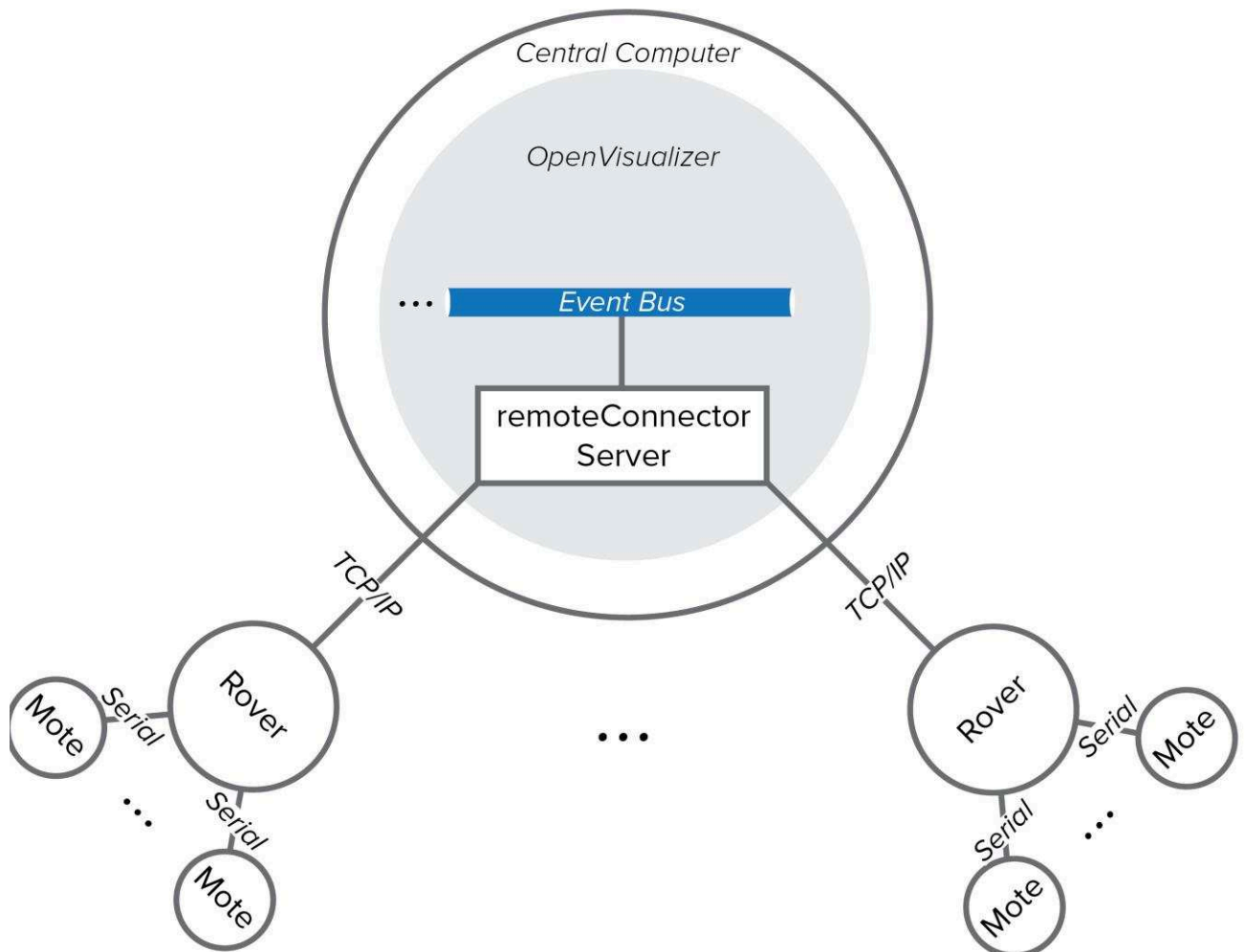


Figure 13.3 – Test bed: test control flows

OpenWSN comes with a Python test tool called OpenVizualiser (OV). With Rover, we split the OV software so that a central computer running the full-fledged master code can control a number of slaves running on the remote Raspberry Pis over IP, providing a single system image for the user and the test tools, as shown in figure 13.3.

We described the test bed in full details in “Rover: Poor (but Elegant) Man’s Testbed” [9]⁶.

Zacharie Brodard modified the OpenWSN software that processes the 6LoWPAN Routing Header (6LoRH) [109]⁷ to support the BIER 6LoRH encoding that we described as a new draft proposal [89]⁸.

⁵www.etsi.org/technologies-clusters/technologies/testing/10-news-events/events/942-6tisch-plugtests

⁶Brodard et al.: “Rover: Poor (but Elegant) Man’s Testbed”

⁷Thubert et al.: *6LoWPAN Routing Header*

⁸Thubert, Brodard, and Jiang: *BIER-TE-based OAM, Replication and Elimination*

The semantics of the BIER bits was directly encoded as a schedule in the individual devices and programmed through Rover.

Jiang Hao then added support to read the RPL topology and generate the track automatically. He also added the automation that is experimented with in chapter 15 on page 97.

In summary, Rover is an IoT testbed which combines OpenWSN software and OpenMote hardware in an easy-to-use, flexible and cheap solution. It offers an ideal real-world experimentation environment for both academic and industry R&D activities, and will help shape the future of the Internet of Things.

Rover improves to the OpenVisualizer software by allowing node to connect remotely. Through the Rover bases, we could reprogram or reflash the OpenMotes, install new schedules and export all the statistics and results if the tests we ran.

13.3 The test layout

Using Rover, we ran the series of tests described in this part with a kit of eight Rover bases, interconnected over the Lab Ethernet wires over IP, as described in figure 13.4.

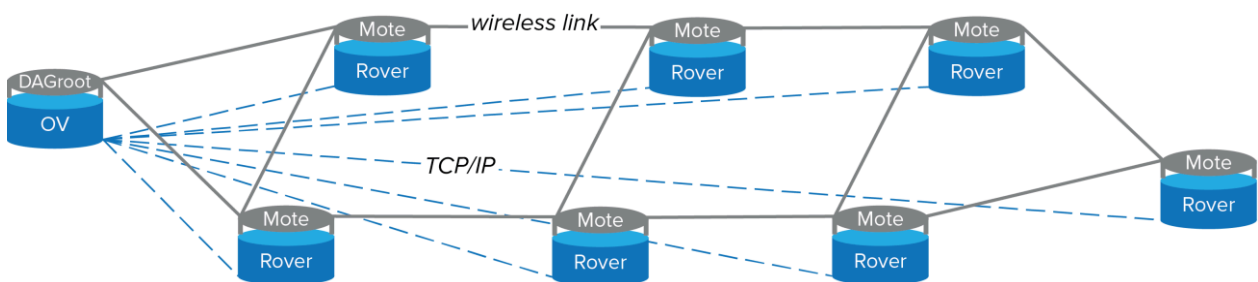


Figure 13.4 – Test bed: layout

The exact positioning of the devices in the PIRL lab is visible in figure 13.1 on page 89.

Eliminating jitter with Replication and Elimination

In this test, we study the pros and cons of Deterministic Replication and Elimination over complex 6TiSCH Tracks as discussed in section 6.4 on page 53 versus the art of retransmission along a serial path.

14.1 Test description

The logical topology that we used for the experimentation is represented in figure 14.1. We established three 6TiSCH Tracks, an upper way (red) and a lower way (blue) with TSCH and ARQ retries, and a complex Track that uses replication and elimination over both paths in parallel, but no ARQ.

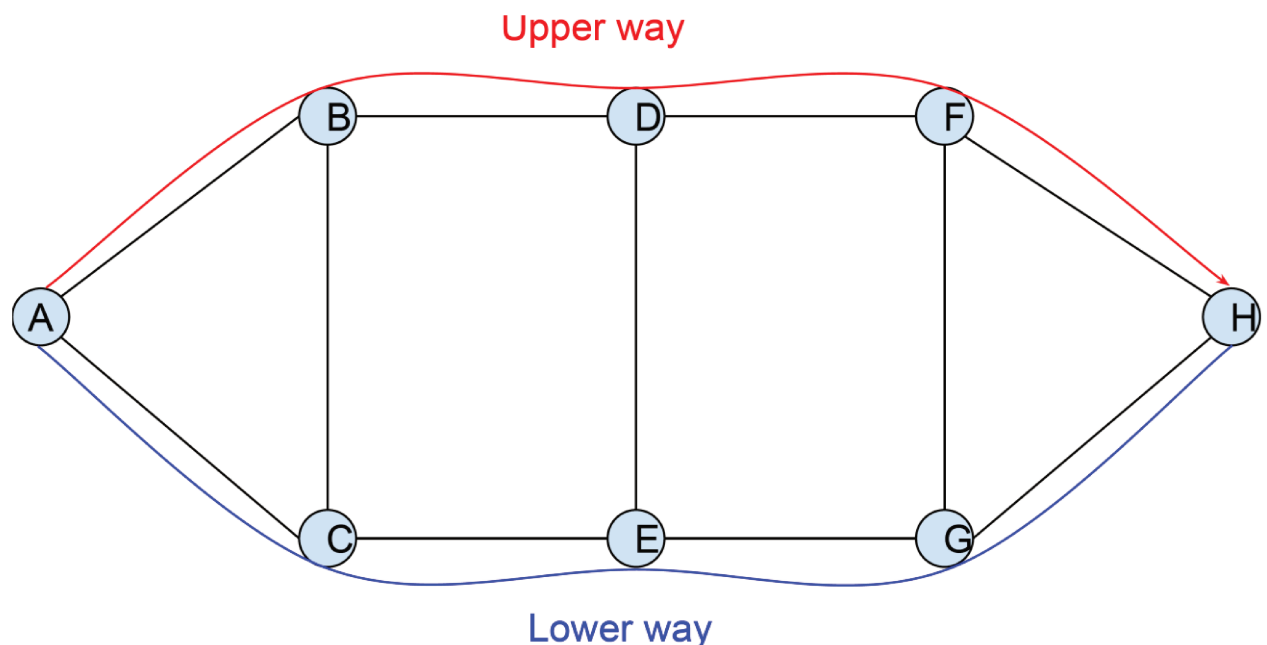


Figure 14.1 – Logical topology for the test

The bundles that represent the complex Track are made of a single cell, and replication and elimination are configured as AND operations between bundles, as discussed in section 6.2 on page 46.

The reasons for multiple receive bundles is that we want to receive all the copies of a same packet and perform the AND operation between the BIER bitmap, as discussed in chapter 11 on page 71, so as to detect and resolve link failures.

For both the blue and the red Tracks, a bundle is set up for transmission over contiguous 16 of a total of 20 time slots, shifted by one at each hop, denoting an OR operation between cells in the bundle, and a maximum of 20 transmissions to reach destination.

We experimented the blue and red Tracks with a constant channel and then TSCH; the former was unstable and for all of the tests presented in this part, we apply TSCH over all 16 channels; we force some anomalies to demonstrate the extra capabilities of BIER-TE and experiment with our control loops.

We send three packets, a red packet along the upper path, a blue packet along the lower path, both forwarded in a hot potato fashion, and then a green packet along the complex Track with replication and elimination, all in a continuous loop, to compare the behaviors.

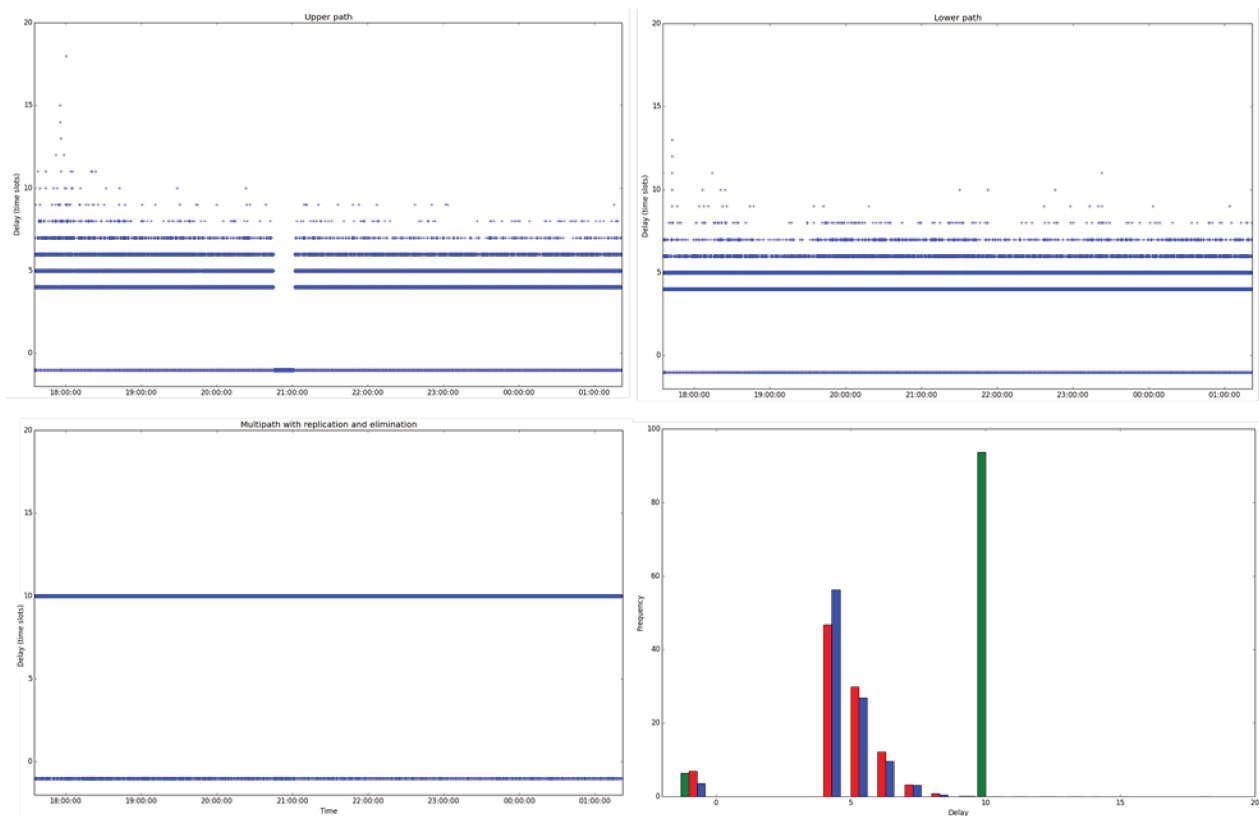


Figure 14.2 – Test1: Replication and Elimination vs. Serial Path

Around 21:00 (9PM) we force the reboot of a device on the red path, causing period of continuous loss, to demonstrate the value of Replication and Elimination when a breakage happens. This affects the PDR along the red path, which fails to deliver appropriately for a period of time, and creates a surge of loss that is observable in the aggregated result.

14.2 Experimental results

The top plots on Figure 14.2 on the facing page represent, over the course of one experiment, with time as the x-axis, the number of time slots, represented as the y-axis, that it took to reach destination. Each packet is represented by a blue cross. The packets that did not make it to the destination are represented with a latency of -1 , which shows as a negative number of time slots.

The third plot, bottom left, represents the same thing for the deterministic Track, and the fourth on the bottom right has the aggregated results, by number of time slots used on the x-axis, and percentage of successful deliveries on the y-axis, the green column representing the deterministic Track.

14.3 Analysis

The top figures show the expected statistical effect of ARQ, whereby depending on the number of retries the packets suffer a variable delay with a minimum observed latency of 4 slots (all transmissions successful) and a maximum of 19 slots (very rare, observed at 18:00 on the red path).

This statistical behavior explains the roughly Poissonian shape of the envelope of the blue and red columns on the aggregated view, bottom right. The most significant result in this experiment appears with the comparison, on a same graph, of this envelope with the green peak that represents the latency along the complex Track. We confirm observations that were made with TTE and the theory that we developed in the train analogy in section 1.5.1 on page 7, that deterministic traffic usually -but not always- experiments a higher latency than hot potato traffic, but also that this latency is a constant.

Indeed, the bottom left figure shows that the green packets experience a constant latency of exactly 10 time slots observed along the complex Track, which is explained by the fully scheduled operation and the total absence of retries along that Track.

We saw in section 3.2 on page 18 that if the probability of loss along one hop on the red or the blue path is roughly P then the probability of a loss on both of the parallel hops is P^2 . This explains why, though rare, we still see packet losses on the green Track.

Comparing the blue and the green Tracks, we see that for most packets, a linear path with TSCH takes the packet faster and with a better energy budget than the complex Track. This is expected since the extra transmissions are performed reactively, only when a first transmission failed, whereas in the case of the complex Tracks, duplicate transmissions happen even if there is no loss. The channel rotation in TSCH preserves equal chances of success for a retransmission. Now, comparing the red and the green Tracks, we see that for the cases that TSCH does not compensate, e.g., the device that reboots at 21:00, the complex Track brings a superior behavior, maintaining a rate of delivery at all times.

Saving energy with BIER-TE-based replication automation

In this test, we evaluate the benefits of controlling Replication and Elimination using BIER headers as discussed in chapter 11 on page 71 in order to save energy. With our BIER-TE approach, each bit in the transported BIER bitmap encodes an adjacency; when a given bit is set, the corresponding adjacency is traversed at its scheduled time, otherwise it is not used for this packet. Note that the schedule is the same regardless of the bit settings in individual packets, so the latency is constant regardless of the portion of the Track that is being used.

15.1 Test description

We saw in section 14.3 on page 95 that with TSCH operation along a serial path and the classic hot potato operation, most though not all packets make it to destination faster and at a lower energy budget than with full replication and elimination, which consumes extra time and energy even when there is no loss. This shows that TSCH coupled with ARQ is indeed an excellent technique, to the point that apart from extreme use cases such as safety, the energy budget that is consumed in replication and elimination appears difficult to justify.

For our test, we reuse the logical topology that was described in section 14.1 on page 93 and make it so that the lower path on the top left experiences a higher loss rate than the top path on the right. We establish a control loop whereby a serial path is used till there is a loss; upon a loss, the source is instructed to set all the bits in the BIER bitmap, so as to establish where the problem comes from. If a link is found to be faulty, a new serial path is computed that routes around the problem, and this path is used for a while. Periodically, the normal path is retested and if it is operative again, then the initial bitmap with the minimum number of bits set is restored.

As illustrated in figure 14.1 on page 93, the shortest serial path has four hops, $A \mapsto C$, $C \mapsto E$, $E \mapsto G$, and $G \mapsto H$, so it can be encoded with only four of the eleven bits that are required to represent all the adjacencies in that graph. When the full replication and elimination is used, a transmission requires almost three times as much energy.

15.2 Experimental results

Figure 15.1 represents the spectrum of latency that is experienced by the packets along the different Tracks.

Both the yellows and the green bars correspond to Replication and Elimination, but the yellow one leverages the BIER bitmap to discover faulty radio hops, and dynamically adapt the forwarding plane to route around them without affecting the routing or the scheduling that are maintained the same during the whole experiment.

As before, the red and blue bars represent the serial paths with standard reactive retransmission upon loss (ARQ).

For all transmissions, we leverage TSCH over 16 channels so the frequencies that are being used are rotated for each packet.

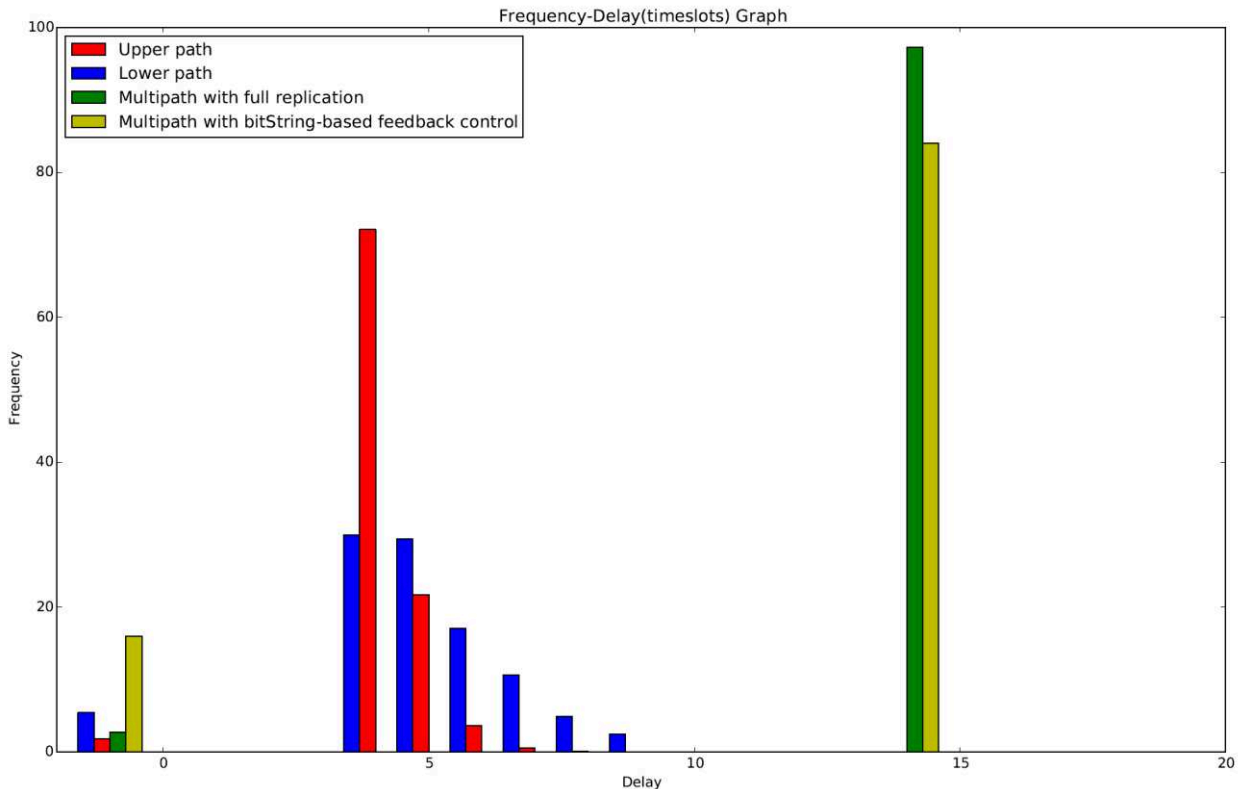


Figure 15.1 – Latency spectrum

Figure 15.2 on the facing page represents the delivery of the various Tracks over time.

Figure 15.3 on the next page represents the energy spent per packet over a small window of time where events of interest take place.

Finally, figures 15.4 on page 100 and 15.5 on page 100 represent the number of losses in a row, the latter being zoomed to at least two losses in a row so that the number of occasions of four losses is still visible.

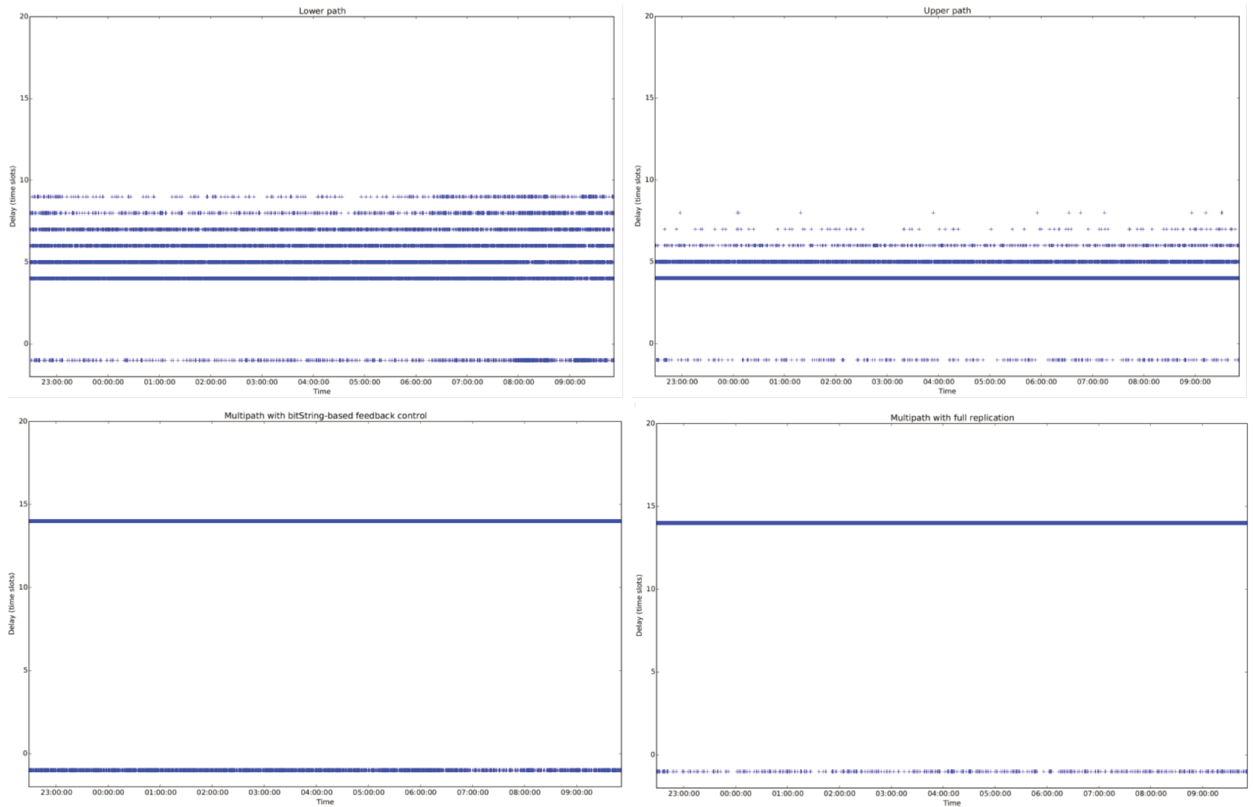


Figure 15.2 – Transmission Reports for various Tracks

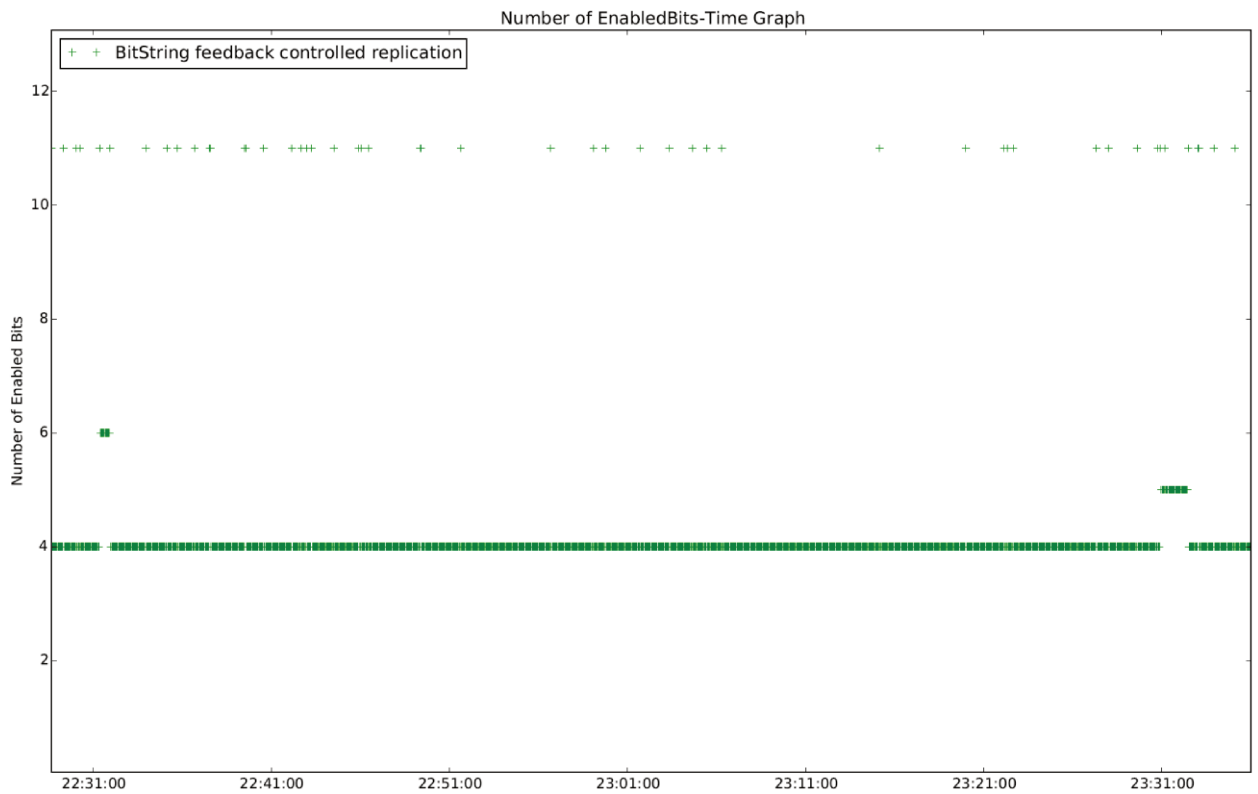


Figure 15.3 – Energy spent per packet

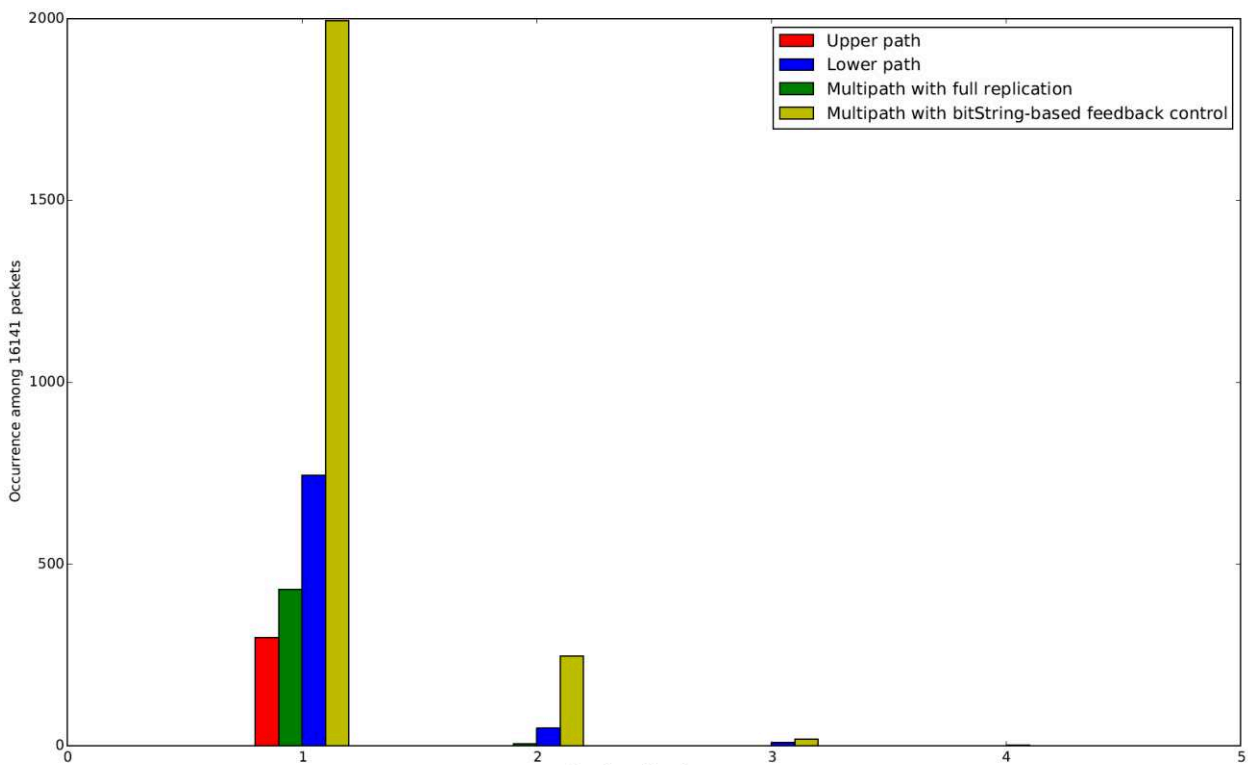


Figure 15.4 – Number of losses in a row (zoomed)

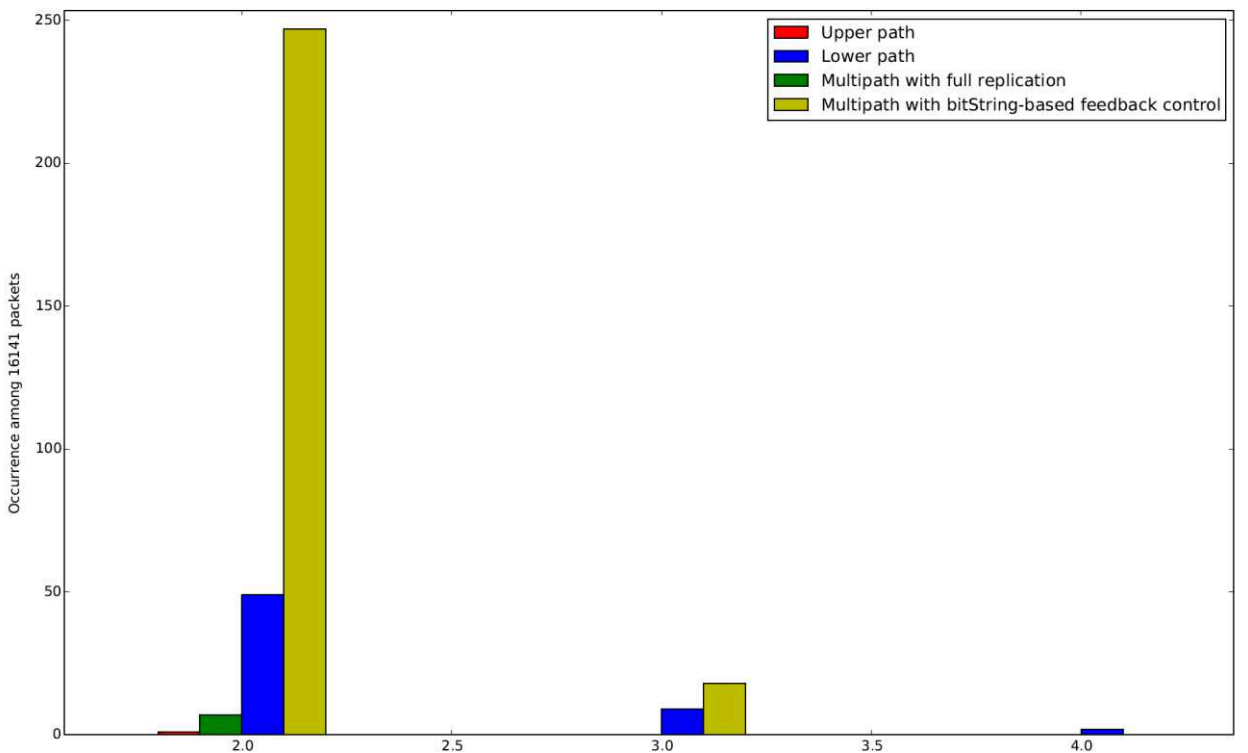


Figure 15.5 – Number of losses in a row, zooming

15.3 Analysis

Looking at the loss that is represented for a latency of -1 in figure 15.1 on page 98, we see that the delivery on the fully replicated Track on the bottom right inherits from the good properties of the top path, whereas the automated energy-saving Track on the bottom left, which is laid out along the lower path, experiences an even higher loss since it has no scheduled retransmission.

This is also illustrated in the left of figure 15.1 on page 98 with the yellow bar associated to latency -1 , which is much higher than the blue or the red.

Comparing the latency spectrum of the full replication in green, with that of the automated energy-saving Track in yellow, we see that the success rate drops from 95% to 83% in our experiment. This is the trade off that we are making in this model to save energy.

Indeed, figure 15.3 on page 99, that illustrates the energy consumption per packet, shows that almost all the packets are sent at the minimal energy cost. Figure 15.3 on page 99 is a zoomed window of time to make each transmission visible. Packets that are sent with the full bitmap require eleven transmissions, so they appear as a green cross at $y = 11$. In contrast, packets that make it along the shortest path appear as a green cross at $y = 4$. We see small bursts of rerouted packets with one additional hop at 22:31:00, and a slightly longer one at 23:31:00.

In the former case, we blocked two links and the workable path involved two additional hops, so the rerouted packets appear as green crosses at $y = 6$, whereas in the latter case, we rebooted one device and only one additional hop was required for the time necessary for the node to rejoin the network.

Figure 15.5 on the facing page shows that the rapid reaction to the loss effectively permits to avoid the critical case of four losses in a row, even with the high degree of loss that is seen in this case. But it must be noted that this requires an efficient back channel for communication. If the back channel is implemented with 6TiSCH minimal support [121]¹, it may be that the latency involved delays the reaction to the problem, adding complexity to the set up.

In summary, the control loop that we put in place excels at saving energy and avoids the condition of four losses in a row, which effectively meets the requirements that we considered for this experiment. On the other hand, this technique yields a high loss ratio, and the return channel to the controller may be problematic to operate in a low power lossy network without a reliable back channel.

¹Vilajosana, Pister, and Watteyne: *Minimal 6TiSCH Configuration*

Saving time and energy with the Leapfrog Collaboration

We saw in section 15 on page 97 that we can save energy on the replication and elimination process by activating it only reactively to errors, and that even then we could avoid four losses in a row, which is a condition for applicability to industrial Process Control.

But then, compared to retries and TSCH, this solution still yields a latency that is far from optimal and may limit the applicability in critical processes.

In this test, we demonstrate the benefits of the Leapfrog Collaboration that is described in section 12.1 on page 77, to save both time and energy, and then again compare with the classical operation of TSCH along a path.

16.1 Test description

For this test, we use RPL Non-Storing Mode [133]¹ in order to learn the topology at the root of the topology. In that Mode, RPL enables devices to report the child-parent relationship along a (DODAG) that is directed towards the root.

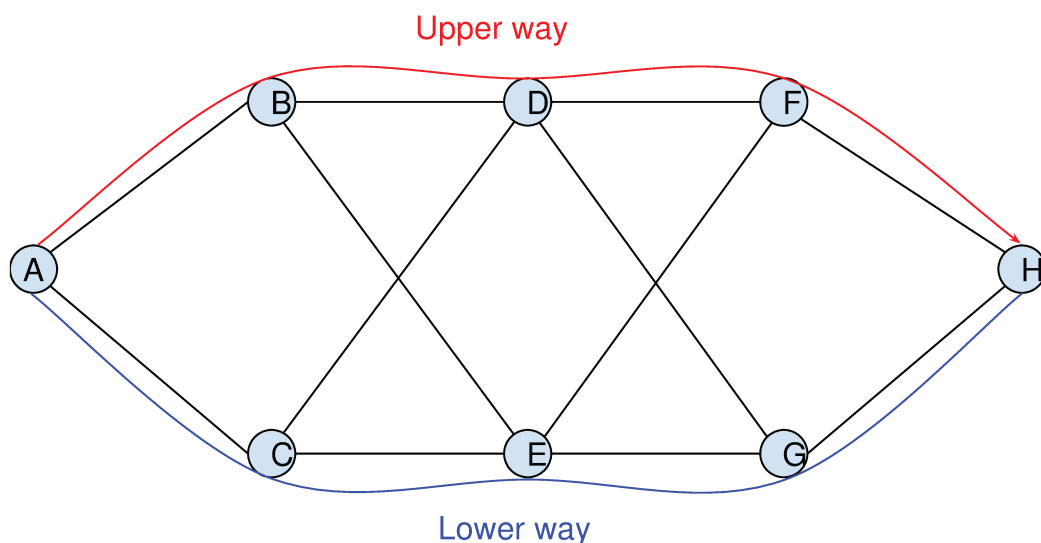


Figure 16.1 – Non-storing RPL topology

¹Winter et al.: *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*

With that information, the root can build a source-routed path to the target device by starting from the target, looking up one of its parents, add it to the tail of the source route path, and recurse till it reaches self.

In our example, Node A is the root, and node H is the target of the messages. With RPL, H advertises to the root that it has 2 parents, F, and G; G advertises D and E, etc... C and B have a single parent, that is A.

Since RPL cannot advertise siblings, we end up with a slightly different vision of the topology, and instead of building the ladder-shape logical structure like the one represented in figure 14.1 on page 93, we end up with braided paths instead, as illustrated in figure 16.1 on the preceding page.

In addition to the red and blue paths that use TSCH with retries, we introduce a purple Track that has the full replication and elimination, a green Track that is only composed of the upper and lower paths operating in parallel but with neither intermediate replication nor retry, and a yellow Track that similar to the green one but which enables overhearing of the transmissions on the other path.

The green Track exhibits a low latency by avoiding to schedule the crossing transmissions between the paths. But since there is no retry, it is highly lossy. On the other hand, the purple Track is reliable but slow. We apply the schedule that is represented in figure 16.2 on that topology and run it in a noisy environment.

Multipath with full replication:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A->B	A->C	B->E	B->D	C->E	C->D	D->G	D->F	E->G	E->F	F->H	G->H			

Multipath in parallel with snooping:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A->>B,C	A->>C,B	B->>E,D	C->>E,D	D->>G,F	E->>G,F	F->>H	G->>H							

Multipath in parallel without snooping:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A->>B	A->>C	B->>D	C->>E	D->>F	E->>G	F->>H	G->>H							

TSCH Schedule:



Figure 16.2 – Overhearing Schedule

The objective of this test is to demonstrate how close we can be to the best of both worlds by overhearing the activity of one path from devices along the other. We need an extremely high error rate for our experiment. So we place it in an environment with a high level of co-channel interference from Wi-Fi users, and place obstacles on the line of sight path.

In several instances during the test, alternating on the upper and lower paths, we shield the antenna of intermediate nodes for short periods of time, in an effort to represent a passing opaque obstacle that would occasionally block the communication on that side.

16.2 Experimental results

Figure 16.3 represents the distribution of delivery, time in the x-axis, and latency in the y-axis. As before, each packet is a tiny blue cross, and as before, we use a latency of -1 to represent a lost packet.

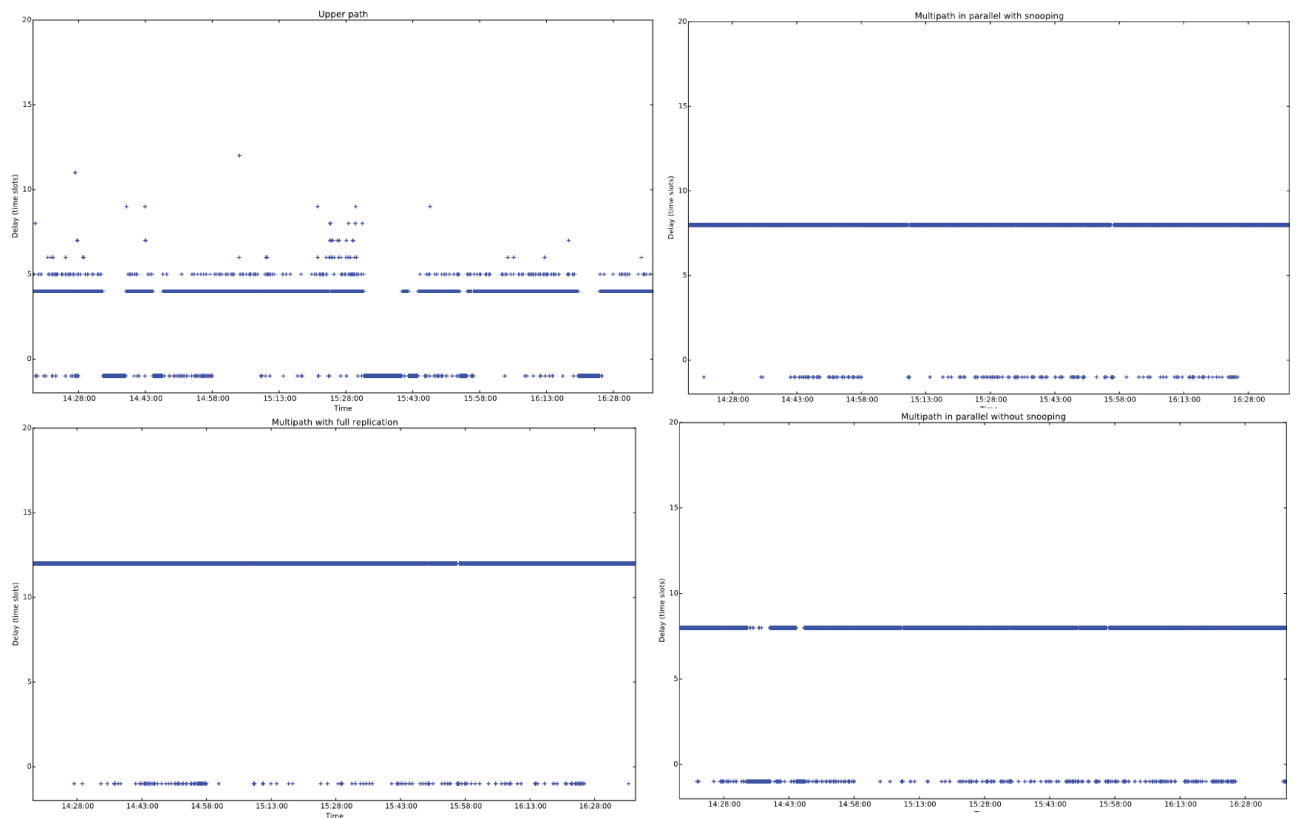


Figure 16.3 – Delivery latency over time

The upper path is represented on the top left, and we can see the statistical distribution of latency; the periods of dense blue lines for a pseudo latency of -1 are indicative of the errors that we inject by covering the antenna of intermediate nodes.

The result of the experiment is plotted on figure 16.4 on the next page, whereas figure 16.5 on the following page represents the number of losses in a row that were observed during that time.

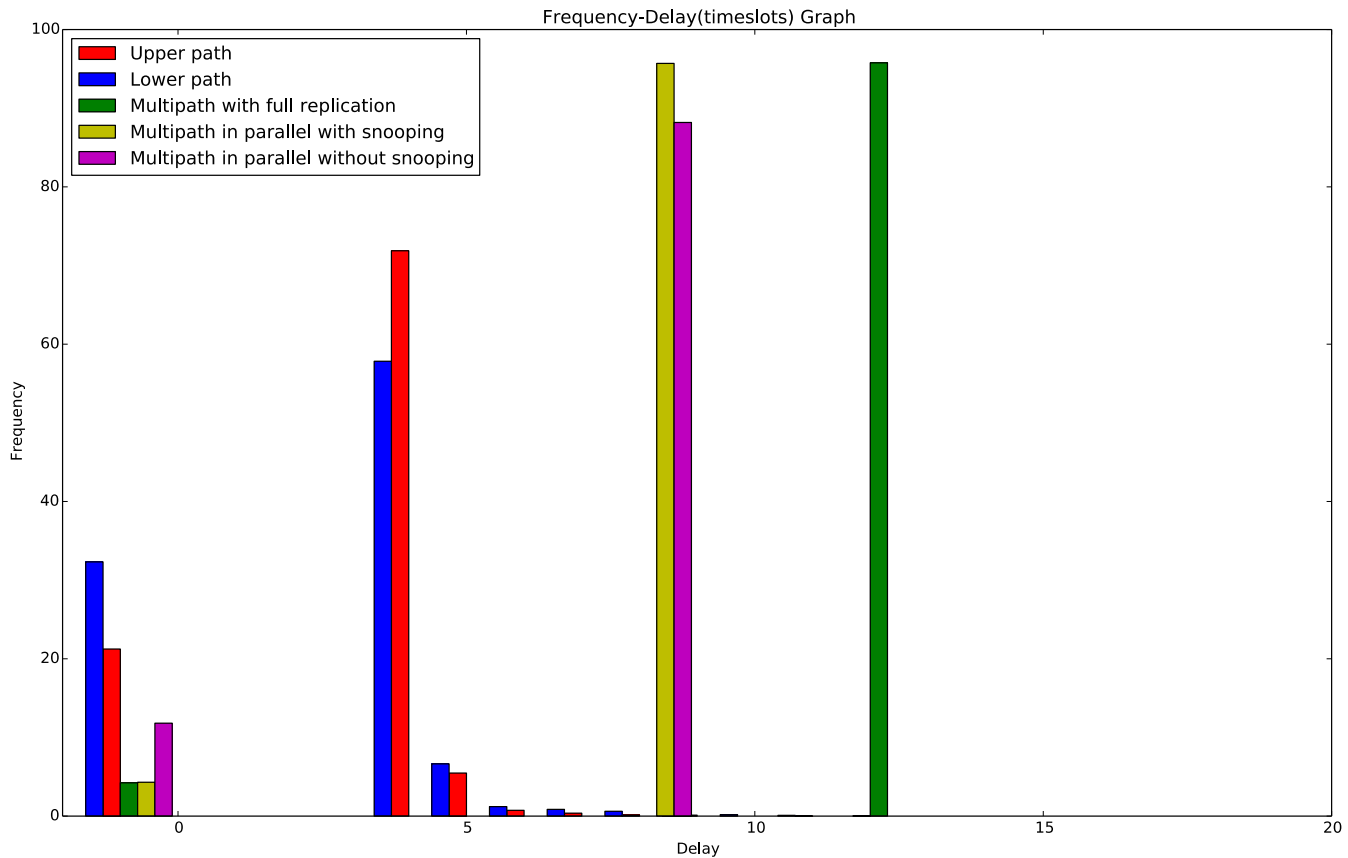


Figure 16.4 – Percentage of occurrence per type of Track

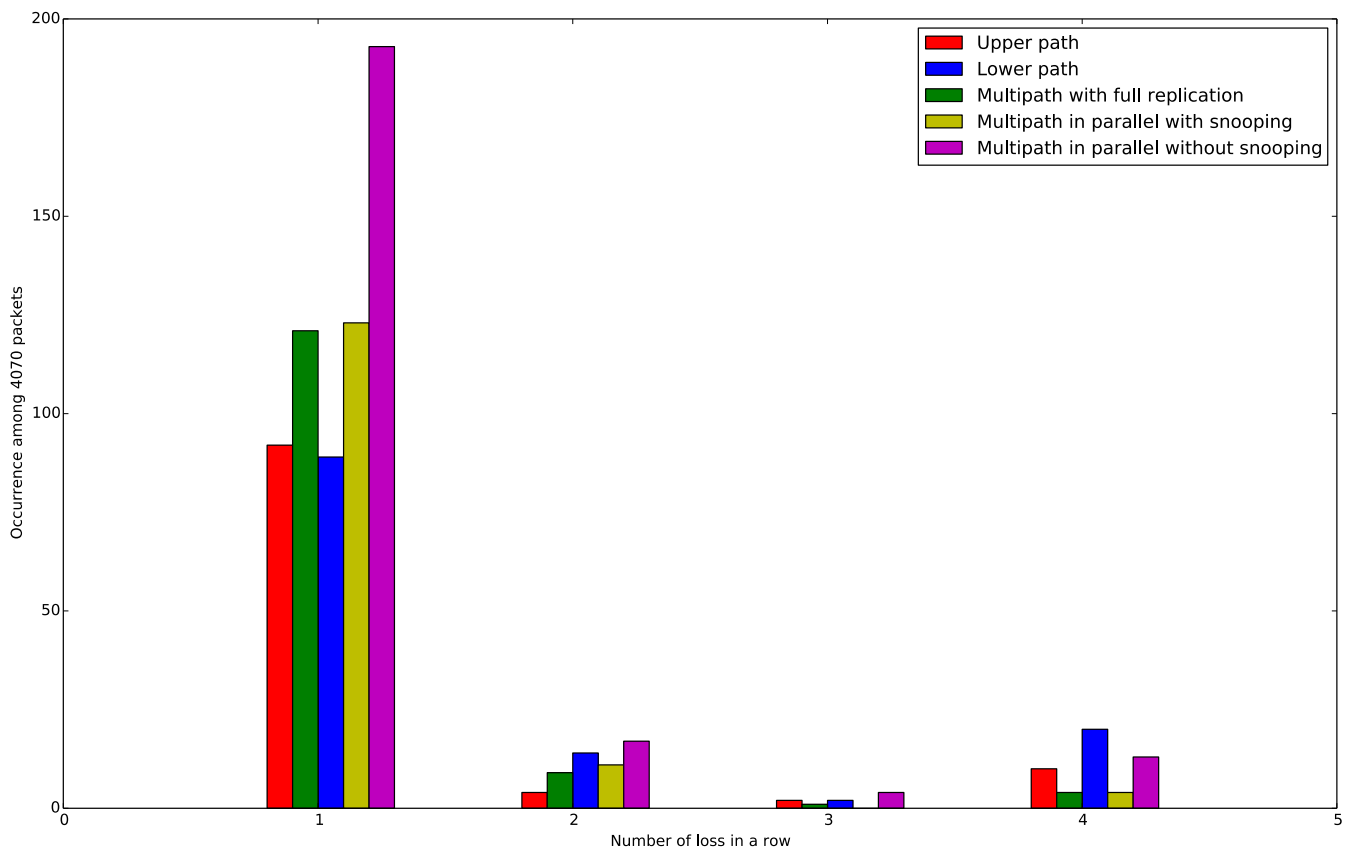


Figure 16.5 – Losses in a row

16.3 Analysis

Though this test involves a much higher loss rate than the previous ones, the envelope of the red and blue plots, as compared to the purple peak, reproduces the results that we have already seen and analyzed in section 14.3 on page 95; the purple peak representing replication and elimination is much shifted to the right compared to the serial paths with retries in red and blue, indicating more transmissions thus more latency and energy spending.

The green Track, which corresponds to full replication and elimination only at the edges and forwarding on the blue and red paths in between, is certainly an improvement in terms of latency and energy spending, but it yields a loss rate that is in the same order as that experienced by either blue or red.

The yellow bar in figures 16.4 on the facing page and 16.5 on the preceding page, which corresponds to the Leapfrog Collaboration, effectively seems to approach the best of both worlds; the latency is reduced in the exact same fashion for the yellow Track as it is for the green Track, but the delivery ratio is increased by almost thirty percent.

The energy consumption $[122]^2$ for the Leapfrog Collaboration Track is somewhere in between the green and the purple cases, since, compared to the green case, we spend additional energy for listening, but not for sending. Since the energy spent on listening is roughly equivalent to that spent in transmitting, we can estimate the energy budget of the green Track as roughly 16 such units, that of the purple Track as 30 and that of the yellow Track as 21, so the Leapfrog Collaboration in this case consumes roughly 30% more energy than the green Track, whereas the purple Track is 50% more.

Considering the huge amounts of errors that we forced, it is not surprising that four losses in a row happened on all types of Tracks, and this manuscript should not be used to dismiss the proposed techniques for industrial application, all the contrary; what we wanted to illustrate is visible in figure 16.5 on the facing page, which shows that compared to serial paths with retries or in parallel, the full replication and the Leapfrog Collaboration both provide a three to four times improvement.

At the same time, we see that the Leapfrog Collaboration is only a few percentile points below the full replication, for the considered criteria of reliable delivery, as visible on the left of figure 16.4 on the preceding page, and avoidance of four losses in a row, as visible on the right of figure 16.5 on the facing page, within a latency that is reduced to 8 time slots.

²Vilajosana et al.: “A realistic energy consumption model for TSCH networks”

Conclusion on our experimentation

We saw that hopping channels with TSCH is mandatory to run our tests reliably and used TSCH at all times, for all types of Tracks. Then we applied different techniques to optimize the two critical properties by which we can rate the applicability of a wireless solution to industrial applications, reliable delivery within a bounded latency and maximum number of losses in a row.

Here is what we found:

- we saw that a serial nailed down path with retries is actually challenged if packets must be delivered reliably within a bounded latency; retransmissions introduce a delay such that some packets arrive too late and are considered as good as lost; using two parallel paths aggregated into a complex Track in place of ARQ enabled to deliver the packets timely, but the loss ratio appeared too high;
- for the criterion that the loss of one packet is acceptable, but four losses in a row are not, we saw that replication and elimination along a complex Track enables the expected user experience at all times, but also that it introduces a latency that is sometimes not acceptable; moreover, the extra energy budget may reduce the applicability of the approach.

the Leapfrog Collaboration appears to be nearing the best of both worlds, passing the reliability and latency criteria and yet doing so within reasonable time and energy budgets.

Compared to hot potato with ARQ along a serial path, Leapfrog provides a guarantee against accidental link or device failures, and a bounded latency for all packets. A serial path with retries has a better energy budget and average latency, and if this the foremost criterion, then it can be selected; but when the trail of late packets is considered as good as lost, or if the delivery of packets must be ensured at all times including rare failures, then the Leapfrog Collaboration should be preferred.

Compared to full replication and elimination, the Leapfrog Collaboration provides slightly lower yet similar benefits for both reliable and timely delivery of packets, as well as avoidance of four losses in a row. The energy consumption and the latency are greatly improved, nearing the budget obtained with two parallel paths and no retry, yet still significantly higher than that of a serial path.

Wrapping up

Conclusion

A trend towards determinism in IT networks: Over the last forty years, most of the communication technologies that we use on a daily basis, mail, books, music, voice and video, have converged onto digital networks, and, by and large, the Internet, not only bringing costs down but also adding unprecedented new value such as immediate service and improved interactivity.

In a similar fashion, converging IT and OT technology on a shared network yields vastly unrealized benefits in multiple vertical industries, including but not limited to manufacturing, commercial sector, building automation, vehicles, and the power grid; despite the huge potential, the convergence is not happening.

In some cases, the operational resistance comes from a lack of trust in the technology and between different professional groups; only time, probably decades, will change that. In other cases, the limitation is technical and the lack of determinism in IT networks now appears as the gating factor.

Both wired and wireless networks are evolving towards more determinism, in particular with work done at the IEEE 802.1 for bridged Ethernet networks, and at 802.15 for Low-power Wireless PANs, but the techniques used in wired and wireless environments are largely different; the DetNet group at the IETF is now considering the establishment of end-to-end paths with Deterministic properties from the perspective of Layer 3, and this thesis proposes improvements to the work done at 6TiSCH for the particular case of LWPANs.

Applicability of deterministic IT networks to solve OT problems: The applicability of the various techniques in the art, and of those proposed in this manuscript, really depends on the use case:

- If an industrial application has no degree of liberty in terms of acceptable jitter and latency whatsoever, then the traditional technique of modulating current over a point to point wire is probably still the best, if not the only option. But the cost and operational complexity of deploying new cables in an existing production facility hinders the addition of new devices for upgrades and enhancements.
- If a high rate and an ultra-precise determinism is required, but some limited latency is acceptable, then a solution based on high speed deterministic Ethernet could be considered; but even if that does not require end-to-end wiring but only from device to the switched fabric, deploying Ethernet comes at the incremental cost for

deploying wires the devices, which may or may not be doable in a particular environment, or may be hardly affordable, depending on the use case. When applicable, for instance for low speed Process Control applications, huge savings in incremental deployment time and cost of goods may be achieved by relying of wireless solutions as opposed to wires.

- If energy is critical and some rare interruptions of service are acceptable, for instance for non-critical monitoring applications, then a hot potato forwarding along a serial TSCH path will be optimal, providing both the best average delivery time and the lowest energy consumption. If the additional latency is acceptable, then a controlled BIER-TE based scheduling associated with a limited number of retries will reduce the jitter at a minimal energy expenditure, and increase the reliability and the availability of the system.
- If more determinism in delivery time is required, or if the application requires availability at all times, then the proposed Leapfrog Collaboration scheme optimizes the PDR/energy ratio.
- But if the prominent goal is to get every packet to the destination and energy is only secondary to it, then a full Replication and Elimination offers the best chances of success. If the latency budget permits, one might then balance an even more complex Track and more spatial diversity with a mix replication and retransmission.

In any case, the proposed BIER-TE bitmap technique provides an efficient encoding to control the forwarding operation along complex Tracks on a per packet basis, which can adapted to meet the required Quality of Service for each individual packet.

Based on the TimeSlotted Channel Hopping MAC technology, the combination of this novel technique and the capabilities offered by the 6TiSCH architecture enables all these possible arrangements and can provide a deterministic delivery for IPv6 packets even in the face of the stochastic behavior of the radio medium.

Perspectives and next research on deterministic wireless networking: Our research opens new grounds for providing determinism in wireless networks, leveraging physical properties of the medium as opposed to simply emulating the operations that are already being studied and standardized for the wired space, in particular for Ethernet. The property that we have been focusing on is the capability by multiple downstream nodes to receive a packet that is destined to only one of them, and we found novel ways to leverage that to compensate the lack of reliability of unicast radio transmissions.

Other properties of the radio medium may be considered. For instance, the degradation of the unicast transmission that is due to multipath fading may be compensated by

sectorized or beam-forming antennas and more complex receive chains, enabling Replication and Elimination of a same physical transmission between the same pair of nodes, as received over different sectors of the antenna, having travelled across different (reflected) physical paths and arriving with microseconds of offset.

As time passes and technologies progress, the energy budget of memory and computation will go down while the physics of radio propagation will not change. It results that more and more, we'll attempt to rely on more complex computation to avoid costly retransmissions and save as much on Replication and Elimination as we can. An alternate form of diversity that we have not evaluated yet, the use of network coding at intermediate points along a Track, may help drive the transmission budget farther down.

Some less ambitious research may also be conducted to propose improvement to the LeapFrog Collaboration, in particular:

- using the Leapfrog on a parallel transmission only upon the detection of a lack of acknowledgement on the main path
- coupling LeapFrog with a routing protocol such as RPL, for instance having a snooping child retransmit on behalf of a first child if the acknowledgement of a parent appears to be missing, thereby providing additional spatial diversity

Enabling an end-to-end DetNet: The DetNet WG at the IETF aims at solving the deterministic networking problem at Layer 3, across technologies. Initially, people consider this as an encapsulation problem, e.g. adding new control fields to pseudo-wires to provide information such as timing and flow identification. But at some point, the Layer 3 abstraction has to be instantiated on the physical network, and if people can figure how IEEE Std 802.1 TSN and DetNet can interact, applying DetNet on other physical layers will bring its own variety of problems.

In particular, enabling determinism on wired and wireless networks separately is certainly not the end of the journey for DetNet. And it can be foreseen that maintaining deterministic properties at the interconnection of wired and wireless networks will be problematic; the protocols used to synchronize, the typical throughput and the methods used to schedule and access the medium, everything is widely different, yielding non-homogeneous capabilities that will be difficult to abstract and account for at an upper layer.

It results that DetNet across wired and wireless links is a complete green field for future research; but this is certainly not barring from deploying the emerging technology in homogeneous wired or wireless control networks. The vision of an IT/OT convergence in an Industrial Internet is now becoming a reality, with new standards such as TSN and 6TiSCH becoming available for production.

Contributions of this thesis

Contribution to the ETSI IPv6 Industry Specification Group

The ETSI IPv6 Industry Specification Group (IP6 ISG)¹ is composed of industry experts in IPv6 and produces highly regarded and widely publicized best practice and recommendation documents to ease and accelerate the deployment of IPv6 in the enterprise, in service provider networks, academia and education networks, datacenter, IoT, SDN and NFV, 5G mobile wireless access and core, and more.

I adapted this manuscript for publication as a contribution to IP6 ISG as ETSI Group Report (GR) IP6 0009 *IPv6-Based Industrial Internet leveraging 6TiSCH Technology*, the produced document largely echoing this manuscript.

Being published as one of those documents is a clear recognition for this work by eminent peers, and a fulfillment for the author.

Contribution to 6TiSCH

The IETF 6TiSCH WG was initially chartered to develop an architecture [82]² and the minimal support [121]³ to provide an IPv6 stack over TSCH, in order to enable stochastic best effort) IPv6 traffic over a basic schedule.

The second round of 6TiSCH still considers stochastic traffic, but adds capabilities to adapt the schedule dynamically to adapt to the variations of the traffic. The work also includes a secure join process, and in preparation for a potential next phase, requirements [81]⁴ for the co-existence with deterministic flows.

The most probable next steps, should the work continue, will involve deterministic IPv6 flows, with Tracks that are either controlled through RPL [96]⁵, or through Yang models yet to be standardized. This thesis comes in preparation for that work; outcomes include multiple papers describing 6TiSCH activities in relation with determinism, patents, and Internet-Drafts at the IETF, some of which resulting in Standard Track RFCs.

¹www.etsi.org/technologies-clusters/technologies/transition-to-ipv6

²Thubert: *An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4*

³Vilajosana, Pister, and Watteyne: *Minimal 6TiSCH Configuration*

⁴Thubert: *6TiSCH requirements for DetNet*

⁵Thubert and Pylakutty: *Root initiated routing state in RPL*

A number of co-authored papers were published in the course of this work and are included in this thesis.

- Book chapters
 - Palattella, Thubert, Vilajosana, Watteyne, Wang, and Engel: “6tisch wireless industrial networks: Determinism meets IPv6” [64] (award winning)
- Papers published in journals
 - Dujovne, Watteyne, Vilajosana, and Thubert: “6TiSCH: deterministic IP-enabled industrial internet (of things)” [15]
 - Thubert: “Low power wireless protocols boost industrial networking” [84]
- Papers published in conferences
 - Thubert, Watteyne, Palattella, Vilajosana, and Wang: “IETF 6tsch: Combining IPv6 connectivity with industrial performance” [112]
 - Thubert, Palattella, and Engel: “6TiSCH centralized scheduling: When SDN meet IoT” [95]
 - Watteyne and Thubert: “Efficient 6LoWPAN Neighbor Discovery applied to Multilink IoT subnets” [127]
 - Brodard, Jiang, Chang, Watteyne, Vilajosana, Thubert, and Texier: “Rover: Poor (but Elegant) Man’s Testbed” [9]
- Papers accepted for future conferences
 - Matsui, Papadopoulos, Thubert, Watteyne, and Montavont: “Poster: 4th Industrial Revolution: Toward Deterministic Wireless Industrial Networks” [50] accepted at ACM EWSM 2017
 - Jiang, Brodard, Ahmed Bouabdallah, Montavont, Texier, Thubert, Watteyne, and Papadopoulos: “Competition: Controlled Replication for Higher Reliability and Predictability in Industrial IoT Networks” [35] accepted at ACM EWSM 2017
 - Papadopoulos, Matsui, Thubert, Texier, Watteyne, and Montavont: “Leapfrog Collaboration: Toward Deterministic and Predictable in Industrial-IoT applications” [66] accepted at IEEE ICC 2017 Conference
- Papers submitted or in progress
 - Brodard, Thubert, Jiang, Watteyne, Chang, Texier, Papadopoulos and Montavont: “Multipath Redundancy for Wireless Time-Slotted Networks” aimed at the IEEE Transactions on Industrial Informatics

I am also the editor of a number of IETF standards that participate to the 6TiSCH overall design and to the 6TiSCH ecosystem, in particular at the 6lo WG which continues and extends the work done at 6LoWPAN, and the ROLL⁶ WG where RPL [133]⁷ was initially developed.

The following contributions are closely related to this thesis though not included therein:

- Informational Internet-Drafts
 - Thubert: *An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4* [82]
 - Thubert: *6TiSCH requirements for DetNet* [81]
- Standard Track Internet-Drafts
 - Chakrabarti, Thubert, and Nordmark: *An Update to 6LoWPAN ND* [11]
 - Thubert: *IPv6 Backbone Router* [83]
 - Sarikaya, Thubert, and Sethi: *Address Protected Neighbor Discovery for Low-power and Lossy Networks* [73]
 - Thubert and Pylakutty: *Root initiated routing state in RPL* [96]
 - Robles, Richardson, and Thubert: *When to use RFC 6553, 6554 and IPv6-in-IPv6* [72]
- RFCs
 - Thubert and Cragie: *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch* [90]
 - Thubert, Bormann, Toutain, and Cragie: *6LoWPAN Routing Header* [109]
- Prior related RFCs
 - Thubert and Hui: *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks* [93]
 - Winter, Thubert, Brandt, Vasseur, Hui, Pister, Levis, Struik, Kelsey, and Clausen: *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks* [133]

⁶datatracker.ietf.org/wg/roll/charter/

⁷Winter et al.: *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*

Contribution to DetNet and BIER

A mere two years after 6TiSCH, the DetNet WG was formed in 2015 to define Deterministic Networking in a general Internet context. I contributed to the creation of the WG, and the foundation documents listed below. I was also a member of the forwarding plane alternatives Design Team.

A roughly the same time, the BIER⁸ WG (for Bit Index Explicit Replication) formed to study a new model for directed multicast flows, whereby bits in a special header indicate which individual destinations are targeted for a particular packet.

The BIER technique was later extended for the purpose of Traffic Engineering (TE); in the BIER-TE [16]⁹ approach, a position in the BIER bitmap effectively indicates an adjacency (say, a segment in a complex path) as opposed to a destination.

I am the principal inventor of US Patent No. 9,456,444 [97]¹⁰ which is a seminal work for BIER-TE, and is included in this thesis. I then extended this work with the BIER-based Replication and Elimination Internet-Draft listed below.

- Informational Internet-Drafts
 - Grossman, Farkas, Varga, Vilajosana, Gunther, Thubert, Wetterwald, Raymond, Korhonen, Kaneko, Das, Zha, Goetz, Schmitt, Mahmoodi, Spirou, and Vizarreta: *Deterministic Networking Use Cases* [26]
 - Thubert and Finn: *Deterministic Networking Architecture* [92]
 - Finn and Thubert: *Deterministic Networking Problem Statement* [22]
 - Korhonen, Farkas, Mirsky, Thubert, Zhuang, and Berger: *DetNet Data Plane Protocol and Solution Alternatives* [39]

- Standard Track Internet-Drafts
 - Thubert, Brodard, and Jiang: *BIER-TE-based OAM, Replication and Elimination* [89]

The discussion and the experimental work in this thesis are focusing on 6TiSCH networks where transmission errors are more readily observable, but the work has a more global applicability to Deterministic Networks in general.

For that reason, the method is being proposed at DetNet and BIER for standardization with a generic applicability to all types of networks. The proposal was initially submitted as a section of a draft version of the DetNet dataplane alternatives, and was then moved to the BIER draft listed above.

⁸datatracker.ietf.org/wg/bier/charter/

⁹Eckert et al.: *Traffic Engineering for Bit Index Explicit Replication BIER-TE*

¹⁰Thubert, Vasseur, and hui: *OAM and Time Slot Control in a Deterministic ARC Chain Topology Network*

Patents issued or still pending

On IPR

It must be noted that I work in the industry for a private company; as many such companies do, Cisco encourages the production of Intellectual Property Rights (IPR) as part of the innovation process within the company. It results that patents, as opposed to academic papers, are the most common form for the seminal work presented in this thesis. Papers and Standards usually follow, though not all are encumbered by IPR.

We developed and patented a number of novel techniques to provide improvements over the art of industrial quasi-deterministic wireless networks. Some of these methods are detailed in this thesis, with experimental and simulation-based evaluations.

Seminal Work: The following patents are seminal work for this thesis and included by reference:

- Issued Patents
 - Thubert, Vasseur, and hui: *OAM and Time Slot Control in a Deterministic ARC Chain Topology Network* [97]
 - Thubert, Vasseur, and Wetterwald: *Installation of time slots for sending a packet through an ARC chain topology network* [101]
- Patents Pending
 - Thubert, Wijnands, and Shepherd: *USING BIT INDEX EXPLICIT REPLICATION (BIER) IN LOW-POWER AND LOSSY NETWORKS* [108]
 - Thubert, Wetterwald, and Vasseur: *LEAPFROG COMMUNICATIONS IN A TSCH NETWORK* [105]

Related contributions: Also considered as part of the work done for this thesis are the following patents, either still pending or already issued:

- Issued Patents
 - Thubert, Wetterwald, Ribiere, Moon, and Kruys: *Path shortening in a wireless mesh network* [114]
 - Thubert, Vasseur, Wetterwald, and Mundenmaney: *Dynamic installation of local storing mode paths in a non-storing low-power and lossy network* [110]
 - Thubert, Vasseur, and Wetterwald: *Timeslot distribution in a distributed routing protocol for deterministic wireless networks* [104]
 - Thubert, Bellagamba, Anteunis, and Levy-Abegnoli: *Generating a loop-free routing topology using routing arcs* [111]
 - Thubert, Vasseur, and Wetterwald: *Reassignment of unused portions of a transmission unit in a network* [102]
 - Vasseur, Thubert, and Wetterwald: *Efficient network probing for deterministic wireless networks* [118]
 - Wetterwald, Vasseur, and Thubert: *Configuring new paths in a wireless deterministic network* [131]
 - Thubert, Vasseur, and Wetterwald: *Synchronized routing updates for TSCH networks* [103]
 - Thubert, Vasseur, and Wetterwald: *Distributed rescheduling of bounded flows in a time sensitive network* [99]
 - Thubert, Vasseur, and Wetterwald: *Exclusive and overlapping transmission unit allocation and use in a network* [100]
- Patents Pending
 - Thubert, Wijnands, and Shepherd: *Reliable multicast in low-power and lossy networks* [107]
 - Vasseur, Wetterwald, and Thubert: *DISTRIBUTED ARBITRATION OF TIME CONTENTION IN TSCH NETWORKS* [120]
 - Levy-Abegnoli, Thubert, and Wetterwald: *Establishing deterministic multicast paths in a network* [42]
 - Thubert, Vasseur, and Wetterwald: *DISTRIBUTED APPROPRIATION AND DEFENSE OF TIME SLOT OWNERSHIP IN A CHANNEL HOPPING SCHEDULE* [98]

-
- Vasseur, Thubert, and Wetterwald: *PREDICTIVE TIME ALLOCATION SCHEDULING FOR TSCH NETWORKS* [119]
 - Thubert, Wetterwald, Vasseur, and Levy-Abegnoli: *Reverse directed acyclic graph for multiple path reachability from origin to identified destination via multiple target devices* [117]
 - Thubert, Vasseur, Levy-Abegnoli, and Wetterwald: *Parent device allocation of retransmit slot to child network device on behalf of peer child device in a deterministic network* [113]

In short

Table 17.1 provides a summary of the main contributions listed in this chapter.

Type of contribution	Number of contributions
ETSI Group Report	1
Issued patents	2
Patents pending	2
Book chapter	1
Journal papers	2
Conference papers	6
IETF Informational I-Ds	6
IETF Standard Track I-Ds	4
IETF Standard Track RFCs	2

Table 17.1 – My various contribution during this thesis

Glossary

- 3GPP** 3rd Generation Partnership Project. 3
- 6LoWPAN** IETF IPv6 over LWPANs WG. v, 42, 45, 51, 91, 118
- 6TiSCH** IETF IPv6 over the TSCH mode of IEEE Std 802.15.4 WG. v–vii, 41–54, 56, 67, 71, 78, 91, 93, 101, 113–115, 117–120
- 6lo** IETF IPv6 over Networks of Resource-constrained Nodes WG (continues 6LoWPAN). 91, 118
- AMI** Advanced Metering Infrastructure. 20
- AMR** Automatic Meter Reading. 20
- ARC** Available Routing Construct. 52, 59–65, 67–70
- ARQ** Automatic Repeat reQuest (or Query). 77, 80, 82, 93, 95, 97, 98, 109
- AS** Autonomous System. 61
- AVB** Audio/Video Bridging. 5, 18, 19, 25, 28
- BBR** Backbone Router. 43
- BER** Bit Error Rate. 20, 37
- BIER** Bit Indexed Explicit Replication. vi, 15, 54, 71–76, 91, 92, 94, 97, 98, 100, 102, 114, 119, 120
- CAC** Call Admission Control. 10
- CAN** Controller Area Network. 6
- CDU** Channel Distribution/Usage. 46–50, 56
- CSMA/CA** Carrier Sense Multiple Access with Collision Avoidance. 21, 22
- DAG** Directed Acyclic Graph. 60
- DetNet** IETF Deterministic Networking WG. vi, 5, 25–33, 42, 44, 45, 52, 71, 76, 113, 115, 119, 120

DODAG Destination Oriented Directed Acyclic Graph. 103

ERPS Ethernet Ring Protection Switching. 19

G-MPLS Generalized MPLS. 45, 46, 51, 53

HSR High-availability Seamless Redundancy. 19

ICT Information and Communication Technologies. ii

IEC International Electrotechnical Commission. 19, 39

IEEE Institute of Electrical and Electronics Engineers. iii–v, 3, 5, 18, 20, 21, 25, 30, 33, 37, 39, 42, 45, 46, 51, 53, 56, 77, 87, 113, 115, 118

IETF Internet Engineering Task Force. iii, v, 3, 5, 13, 14, 25, 26, 28, 30, 32, 41, 42, 45, 51, 59, 71, 113, 115, 117, 118, 122

IoT Internet of Things. 3, 14, 22, 40, 51, 92, 117

IP Internet Protocol. i–iii, v, 3–6, 10, 11, 13, 40, 49, 51, 55, 56, 77, 91, 92

IPv6 Internet Protocol Version 6. v, 15, 40–43, 45, 48, 49, 61, 71, 114, 117

ISDN Integrated Services Digital Network. 10

ISM Industrial, Scientific, and Medical. 20, 22

IT Information Technology. ii, iii, 3–5, 9, 18, 44, 55, 113, 115

LAN Local Area Network. 9, 25, 29, 61

LDP Label Distribution Protocol. 13

LFA Loop-Free Alternates. 13

LLN Low-Power and Lossy Network. v, 14, 42, 43

LWPAN Low-power Wireless Personal Area Network. iv, 113

MAC Medium Access Control. v, vi, 5, 21, 22, 39, 41, 43, 48, 51, 53, 55, 56, 114

MPLS MultiProtocol Label Switching. 13, 15, 45, 51, 61

MPLS-TP MPLS Transport Profile. 13, 61

NFV Network Functions Virtualization. 117

OAM Operations, Administration and Maintenance. vi, vii, 13, 26

OLAF Open Lowest ARC First. 60, 63

OPEX Operational Expenditures. 15, 40, 55

OT Operational Technology. ii, iii, 4, 5, 9, 11, 12, 14, 25, 40, 41, 43, 44, 55, 113, 115

PAN Personal Area Network. 113

PCE Path Computation Element. 14, 28, 32, 45, 52, 53, 56, 70, 76

PCI Peripheral Component Interconnect. 6

PER Packet Error Rate. 19

PIRL Cisco's Paris Innovation Labs. 89, 92

PMN Professional Media Networking. 6, 31

PRP Parallel Redundancy Protocol. 19

PSK Pre-Shared Keys. 44

PTP Precision Time Protocol. 18, 33

QoS Quality of Service. i, 10, 11, 17, 56

REP Resilient Ethernet Protocol. 19

RFC Request For Comment (as an IETF standard). 117, 119, 122

ROLL IETF Routing Over Low power and Lossy networks WG. 91, 118

RPL IPv6 Routing Protocol for Low-Power and Lossy Networks. v, 42–45, 47, 92, 103, 104, 117, 118

SDH Synchronous Digital Hierarchy. 10

SDN Software-Defined Networking. 6, 14, 28, 32, 39, 61, 62, 117

SONET Synchronous Optical Networking. 10

SP Service Provider. 22

SPF Shortest Path First. 64

SR Segment Routing. 15

TDM Time Division Multiplexing. 4, 10, 37, 55

TE Traffic Engineering. vi, 14, 15, 30, 39, 52, 54, 71, 73, 76, 94, 97, 98, 100, 102, 114, 119

TSCH Time-Slotted Channel Hopping. iv–vi, 5, 20–22, 30, 37–39, 41–43, 45, 46, 48, 49, 51, 56, 67, 77, 87, 93–95, 97, 98, 103, 104, 109, 114, 117

TSMP Time Synchronized Mesh Protocol. iv

TSN Time-Sensitive Networking. 5, 18, 25, 33, 115

TTE Time-Triggered Ethernet. 18, 95

VLAN Virtual Local Area Network. 9

WG Working Group. v, 5, 25, 26, 41–43, 45, 52, 115, 117–119

WPAN Wireless Personal Area Network. iv

WSN Wireless Sensor Network. i, iv, 22, 37, 39–41, 44, 67, 77

List of Figures

1	The Power of One (Source GE estimates)	iii
1.1	Enabling trends	6
1.2	A Railway System, illustrating the complexity of a global schedule	7
1.3	The bus analogy	8
1.4	traditional Quality of Service (Source Cisco Systems)	10
3.1	A TSCH schedule (Source T. Watteyne)	21
4.1	DetNet Components (Source S. Dhesikan)	26
4.2	The DetNet Architecture	27
4.3	Single Path	28
4.4	Complex Track	28
4.5	Perspective of a networking engineer (Source N. Finn)	29
4.6	Perspective of the controller (Source N. Finn)	29
4.7	Reporting capabilities and topology to the Controller	30
4.8	Setting up a new flow	31
4.9	Inside the controller	32
5.1	TSCH iterations	38
5.2	Industrial Standards for Wireless Process Control	39
5.3	Triple OPEX	40
5.4	Growth in analytics-related measurements. Source: ABI Research	41
5.5	Backbone Router	43
6.1	6TiSCH CDU matrix	46
6.2	Example CDU matrix partitioned in Chunks	47
6.3	Spatial application of Chunks	47
6.4	6TiSCH Slotframe	48
6.5	6TiSCH Slotframe prioritization and schedule execution	48
6.6	A 6TiSCH Bundle	49
6.7	A 6TiSCH Layer 3 Bundle	49
6.8	Forwarding along 2 simple serial Tracks, A to U and B to V	50
6.9	A 6TiSCH Layer 2 Bundle	50
6.10	G-MPLS switching along Tracks	51
7.1	Matching MAC technology and type of traffic	55

8.1	An ARC, [Re] for Reversible Link	59
8.2	An ARC Multi-Path to Omega	60
8.3	Comparing Fast Reroute technologies	62
9.1	building the first ARC	64
9.2	Completing the ARC chain	65
10.1	Assigning time Slots steps 1 to 5	68
10.2	Assigning time Slots steps 6 to 9	69
11.1	The analogy of a ladder	72
11.2	Assigning bits to control Replication and Elimination	72
11.3	Enabled adjacencies with bits 1, 2, 4, 5, 6 and 7 set	73
11.4	Bitwise operation down the path	74
11.5	AND operation at the elimination point	75
12.1	Example PDR for a LeapFrog Track	79
12.2	LeapFrog: Complex Track and initial forwarding steps	80
12.3	LeapFrog: Intermediate forwarding steps	81
12.4	LeapFrog: Final forwarding steps and delivery	82
12.5	Alternate view with only a plan B	83
12.6	Alternate view with an additional redundancy	83
13.1	PIRL, 6 th Floor at Cisco's ILM office near Paris	89
13.2	Rover: an OpenMote mounted on a RPi	90
13.3	Test bed: test control flows	91
13.4	Test bed: layout	92
14.1	Logical topology for the test	93
14.2	Test1: Replication and Elimination vs. Serial Path	94
15.1	Latency spectrum	98
15.2	Transmission Reports for various Tracks	99
15.3	Energy spent per packet	99
15.4	Number of losses in a row (zoomed)	100
15.5	Number of losses in a row, zooming	100
16.1	Non-storing RPL topology	103
16.2	Overhearing Schedule	104
16.3	Delivery latency over time	105
16.4	Percentage of occurrence per type of Track	106
16.5	Losses in a row	106

List of Tables

11.1 Assigning bits to protect segment $A \mapsto C$	73
17.1 My various contribution during this thesis	123



Bibliography

- [1] Alghamdi et al., “Constructing a Shortest Path Overhearing Tree with Maximum Lifetime in WSNs,” *in: 2016 49th Hawaii International Conference on System Sciences (HICSS)*, IEEE, 2016, pp. 5858–5867.
- [2] Allan and Busi, *Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks*, RFC 6371, Sept. 2011, DOI: 10.17487/rfc6371, URL: <https://rfc-editor.org/rfc/rfc6371.txt>.
- [3] Araujo et al., “High availability automation networks: PRP and HSR ring implementations,” *in: Industrial Electronics (ISIE), 2012 IEEE International Symposium on*, IEEE, 2012, pp. 1197–1202.
- [4] Atlas et al., *OSPF Traffic Engineering (TE) Metric Extensions*, RFC 7471, Mar. 2015, DOI: 10.17487/rfc7471, URL: <https://rfc-editor.org/rfc/rfc7471.txt>.
- [5] Bachir et al., “MAC essentials for wireless sensor networks,” *in: IEEE Communications Surveys & Tutorials* 12.2 (2010), pp. 222–248.
- [6] Beller and Sperber, “MPLS-TP-The New Technology for Packet Transport Networks.,” *in: DFN-Forum Kommunikationstechnologien*, vol. 149, 2009, pp. 81–92.
- [7] Berger, Bartlett, and Harrison, *IPv6 Traffic Engineering in IS-IS*, RFC 6119, Feb. 2011, DOI: 10.17487/rfc6119, URL: <https://rfc-editor.org/rfc/rfc6119.txt>.
- [8] Borella et al., *Methods for determining sendable information content based on a determined network latency*, US Patent 6,182,125, Jan. 2001.
- [9] Brodard et al., “Rover: Poor (but Elegant) Man’s Testbed,” *in: ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN)*, 2016.
- [10] Bryant et al., *Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)*, RFC 7490, Apr. 2015, DOI: 10.17487/rfc7490, URL: <https://rfc-editor.org/rfc/rfc7490.txt>.
- [11] Chakrabarti, Thubert, and Nordmark, *An Update to 6LoWPAN ND*, Internet-Draft draft-ietf-6lo-rfc6775-update-01, Work in Progress, Internet Engineering Task Force, Jan. 2017, 22 pp., URL: <https://tools.ietf.org/html/draft-ietf-6lo-rfc6775-update-01>.
- [12] Colvin, “CSMA with collision avoidance,” *in: Computer Communications* 6.5 (1983), pp. 227–235.

-
- [13] Craciunas and Oliver, “SMT-based task-and network-level static schedule generation for time-triggered networked systems,” *in: Proceedings of the 22nd International Conference on Real-Time Networks and Systems*, ACM, 2014, p. 45.
- [14] Dujovne et al., *6TiSCH 6top Scheduling Function Zero (SF0)*, Internet-Draft draft-ietf-6tisch-6top-sf0-02, Work in Progress, Internet Engineering Task Force, Oct. 2016, 12 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-6top-sf0-02>.
- [15] Dujovne et al., “6TiSCH: deterministic IP-enabled industrial internet (of things),” *in: IEEE Communications Magazine* 52.12 (2014), pp. 36–41.
- [16] Eckert et al., *Traffic Engineering for Bit Index Explicit Replication BIER-TE*, Internet-Draft draft-eckert-bier-te-arch-04, Work in Progress, Internet Engineering Task Force, July 2016, 28 pp., URL: <https://tools.ietf.org/html/draft-eckert-bier-te-arch-04>.
- [17] Envedi, Atlas, and Bowers, *An Architecture for IPLDP Fast Reroute Using Maximally Redundant Trees (MRT-FRR)*, RFC 7812, June 2016, DOI: 10.17487/rfc7812, URL: <https://rfc-editor.org/rfc/rfc7812.txt>.
- [18] Fairhurst and Wood, *Advice to link designers on link Automatic Repeat reQuest (ARQ)*, RFC 3366, Sept. 2002, DOI: 10.17487/rfc3366, URL: <https://rfc-editor.org/rfc/rfc3366.txt>.
- [19] Fang and Zhang, “Call admission control schemes and performance analysis in wireless mobile networks,” *in: IEEE Transactions on vehicular technology* 51.2 (2002), pp. 371–382.
- [20] Farrel and Ash, *A Path Computation Element (PCE)-Based Architecture*, RFC 4655, Aug. 2006, DOI: 10.17487/rfc4655, URL: <https://rfc-editor.org/rfc/rfc4655.txt>.
- [21] Filsfils et al., *Segment Routing Architecture*, Internet-Draft draft-ietf-spring-segment-routing-10, Work in Progress, Internet Engineering Task Force, Nov. 2016, 29 pp., URL: <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-10>.
- [22] Finn and Thubert, *Deterministic Networking Problem Statement*, Internet-Draft draft-ietf-detnet-problem-statement-01, Work in Progress, Internet Engineering Task Force, Sept. 2016, 11 pp., URL: <https://tools.ietf.org/html/draft-ietf-detnet-problem-statement-01>.
- [23] Frost, Bryant, and Bocci, *MPLS Transport Profile Data Plane Architecture*, RFC 5960, Aug. 2010, DOI: 10.17487/rfc5960, URL: <https://rfc-editor.org/rfc/rfc5960.txt>.
- [24] Gaillard, “Managing IoT networks using service level agreements (SLA),” Theses, INSA Lyon, Dec. 2016, URL: <https://hal.inria.fr/tel-01429025>.

-
- [25] Goraj and Harada, “Migration paths for IEC 61850 substation communication networks towards superb redundancy based on hybrid PRP and HSR topologies,” *in: Developments in Power Systems Protection, 2012. DPSP 2012. 11th International Conference on*, IET, 2012, pp. 1–6.
- [26] Grossman et al., *Deterministic Networking Use Cases*, Internet-Draft draft-ietf-detnet-use-cases-11, Work in Progress, Internet Engineering Task Force, Oct. 2016, 79 pp., URL: <https://tools.ietf.org/html/draft-ietf-detnet-use-cases-11>.
- [27] Hasegawa et al., “Industrial wireless standardization—Scope and implementation of ISA SP100 standard,” *in: SICE Annual Conference (SICE), 2011 Proceedings of*, IEEE, 2011, pp. 2059–2064.
- [28] Hwang, “Energy efficient channel agility utilizing dynamic multi-channel CCA for ZigBee RF4CE,” *in: IEEE Transactions on Consumer Electronics* 57.1 (2011), pp. 113–119.
- [29] IEC, *IEC 61850 Communication networks and systems for power utility automation - ALL PARTS*, 2003-2016, URL: <https://webstore.iec.ch/publication/6028>.
- [30] IEC, *IEC 62591 Industrial networks - Wireless communication network and communication profiles - WirelessHART™*, 2016, URL: <https://webstore.iec.ch/publication/24433>.
- [31] IEC, *IEC 62601 Industrial networks - Wireless communication network and communication profiles - WIA-PA*, 2015, URL: <https://webstore.iec.ch/publication/23902>.
- [32] IEC, *IEC 62734 Industrial networks - Wireless communication network and communication profiles - ISA 100.11a*, 2014, URL: <https://webstore.iec.ch/publication/7409>.
- [33] IEC, *Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)*, 2016, URL: <https://webstore.iec.ch/publication/24447>.
- [34] Ingram, Schaub, and Campbell, “Use of precision time protocol to synchronize sampled-value process buses,” *in: IEEE Transactions on Instrumentation and Measurement* 61.5 (2012), pp. 1173–1180.
- [35] Jiang et al., “Competition: Controlled Replication for Higher Reliability and Predictability in Industrial IoT Networks,” *in: Proceedings of the International Conference on Embedded Wireless Systems and Networks (EWSN)*, 2017.

-
- [36] Kirrmann, Hansson, and Muri, “IEC 62439 PRP: Bumpless recovery for highly available, hard real-time industrial networks,” *in: 2007 IEEE Conference on Emerging Technologies and Factory Automation (EFTA 2007)*, IEEE, 2007, pp. 1396–1399.
- [37] Kirrmann et al., “HSR: Zero recovery time and low-cost redundancy for Industrial Ethernet (High availability seamless redundancy, IEC 62439-3),” *in: Proceedings of the 14th IEEE international conference on Emerging technologies & factory automation*, IEEE Press, 2009, pp. 203–206.
- [38] Kopetz et al., “The time-triggered ethernet (TTE) design,” *in: Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC’05)*, IEEE, 2005, pp. 22–33.
- [39] Korhonen et al., *DetNet Data Plane Protocol and Solution Alternatives*, Internet-Draft draft-ietf-detnet-dp-alt-00, Work in Progress, Internet Engineering Task Force, Oct. 2016, 51 pp., URL: <https://tools.ietf.org/html/draft-ietf-detnet-dp-alt-00>.
- [40] Lampin, “Urban wireless sensors networks : Applications and protocols,” Theses, INSA de Lyon, Jan. 2014, URL: <https://tel.archives-ouvertes.fr/tel-01127316>.
- [41] Lee and Huh, “Reliable data transfer using overhearing for implicit ack,” *in: ICCAS-SICE, 2009*, IEEE, 2009, pp. 1976–1979.
- [42] Levy-Abegnoli, Thubert, and Wetterwald, *Establishing deterministic multicast paths in a network*, US Patent App. 14/816,108, Aug. 2015.
- [43] Liang et al., “Survey and experiments of WIA-PA specification of industrial wireless network,” *in: Wireless Communications and Mobile Computing 11.8* (2011), pp. 1197–1212, ISSN: 1530-8677, DOI: 10.1002/wcm.976, URL: <http://dx.doi.org/10.1002/wcm.976>.
- [44] Lin and Pearson, “An inside look at industrial Ethernet communication protocols,” *in: White Paper Texas Instruments* (2013).
- [45] Maalel et al., “Adaptive reliable routing protocol for wireless sensor networks,” *in: ICWMC 2013, The Ninth International Conference on Wireless and Mobile Communications*, 2013, pp. 52–55.
- [46] MacKay, *Applications and Opportunities for the IEEE 1588 Standard in Military Applications*, tech. rep., DTIC Document, Jan. 2007, URL: <http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA474269>.
- [47] Mahmood, Seah, and Welch, “Reliability in wireless sensor networks: A survey and challenges ahead,” *in: Computer Networks* 79 (2015), pp. 166–187.
- [48] Mangold et al., “Analysis of IEEE 802.11 e for QoS support in wireless LANs,” *in: IEEE wireless communications* 10.6 (2003), pp. 40–50.

-
- [49] Mannie, “Generalized multi-protocol label switching (GMPLS) architecture,” *in: Interface* 501 (2004), p. 19.
- [50] Matsui et al., “Poster: 4th Industrial Revolution: Toward Deterministic Wireless Industrial Networks,” *in: Proceedings of the International Conference on Embedded Wireless Systems and Networks (EWSN)*, 2017.
- [51] McKeown et al., “OpenFlow: enabling innovation in campus networks,” *in: ACM SIGCOMM Computer Communication Review* 38.2 (2008), pp. 69–74.
- [52] Meier and Weibel, “IEEE 1588 applied in the environment of high availability LANs,” *in: 2007 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, IEEE, 2007, pp. 100–104.
- [53] Melakessou, Palattella, and Engel, “Towards a new way of reliable routing: Multiple paths over ARCs,” *in: Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on)*, IEEE, 2014, pp. 168–174.
- [54] Mizrahi et al., *An Overview of Operations, Administration, and Maintenance (OAM) Tools*, RFC 7276, June 2014, DOI: 10.17487/rfc7276, URL: <https://rfc-editor.org/rfc/rfc7276.txt>.
- [55] Montenegro et al., *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*, RFC 4944, Sept. 2007, DOI: 10.17487/rfc4944, URL: <https://rfc-editor.org/rfc/rfc4944.txt>.
- [56] Morell et al., “Label switching over IEEE802.15.4e networks,” *in: Transactions on Emerging Telecommunications Technologies* 24.5 (2013), pp. 458–475.
- [57] Neagoe, Cristea, and Banica, “NTP versus PTP in computer networks clock synchronization,” *in: 2006 IEEE International Symposium on Industrial Electronics*, vol. 1, IEEE, 2006, pp. 317–362.
- [58] Nixon and Round Rock, “A Comparison of WirelessHART™ and ISA100. 11a,” *in: Whitepaper, Emerson Process Management* (2012), pp. 1–36.
- [59] Nunes et al., “A survey of software-defined networking: Past, present, and future of programmable networks,” *in: IEEE Communications Surveys & Tutorials* 16.3 (2014), pp. 1617–1634.
- [60] de la Oliva et al., “An overview of the CPRI specification and its application to C-RAN-based LTE scenarios,” *in: IEEE Communications Magazine* 54.2 (2016), pp. 152–159.
- [61] Oliver, Craciunas, and Stöger, “Analysis of deterministic ethernet scheduling for the industrial internet of things,” *in: 2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, IEEE, 2014, pp. 320–324.

-
- [62] Osborne et al., *MPLS Transport Profile (MPLS-TP) Linear Protection*, RFC 6378, Oct. 2011, DOI: 10.17487/rfc6378, URL: <https://rfc-editor.org/rfc/rfc6378.txt>.
- [63] Ould-Brahim and Fedyk, *BGP Traffic Engineering Attribute*, RFC 5543, May 2009, DOI: 10.17487/rfc5543, URL: <https://rfc-editor.org/rfc/rfc5543.txt>.
- [64] Palattella et al., “6tisch wireless industrial networks: Determinism meets IPv6,” *in: Internet of Things*, Springer, 2014, pp. 111–141.
- [65] Palattella et al., “Standardized protocol stack for the internet of (important) things,” *in: IEEE Communications Surveys & Tutorials* 15.3 (2013), pp. 1389–1406.
- [66] Papadopoulos et al., “Leapfrog Collaboration: Toward Deterministic and Predictable in Industrial-IoT applications,” *in: Proceedings of the IEEE International Conference on Communications (ICC)*, 2017.
- [67] Petersen and Carlsen, “WirelessHART versus ISA100. 11a: the format war hits the factory floor,” *in: IEEE Industrial Electronics Magazine* 5.4 (2011), pp. 23–34.
- [68] Phinney, Thubert, and Dwars, *Industrial Routing Requirements in Low-Power and Lossy Networks*, RFC 5673, Oct. 2009, DOI: 10.17487/rfc5673, URL: <https://rfc-editor.org/rfc/rfc5673.txt>.
- [69] Pister and Doherty, “TSMP: Time synchronized mesh protocol,” *in: IASTED Distributed Sensor Networks* (2008), pp. 391–398.
- [70] Pujolle, “SDN (Software-Defined Networking),” *in: Software Networks* (), pp. 15–48.
- [71] Richardson, *6tisch Secure Join protocol*, Internet-Draft draft-ietf-6tisch-dtsecurity-secure-join-00, Work in Progress, Internet Engineering Task Force, Dec. 2016, 20 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-dtsecurity-secure-join-00>.
- [72] Robles, Richardson, and Thubert, *When to use RFC 6553, 6554 and IPv6-in-IPv6*, Internet-Draft draft-ietf-roll-useofrplinfo-10, Work in Progress, Internet Engineering Task Force, Dec. 2016, 35 pp., URL: <https://tools.ietf.org/html/draft-ietf-roll-useofrplinfo-10>.
- [73] Sarikaya, Thubert, and Sethi, *Address Protected Neighbor Discovery for Low-power and Lossy Networks*, Internet-Draft draft-ietf-6lo-ap-nd-00, Work in Progress, Internet Engineering Task Force, Nov. 2016, 17 pp., URL: <https://tools.ietf.org/html/draft-ietf-6lo-ap-nd-00>.
- [74] Shand and Bryant, *IP Fast Reroute Framework*, RFC 5714, Jan. 2010, DOI: 10.17487/rfc5714, URL: <https://rfc-editor.org/rfc/rfc5714.txt>.
- [75] Shenker, *Specification of Guaranteed Quality of Service*, RFC 2212, Sept. 1997, DOI: 10.17487/rfc2212, URL: <https://rfc-editor.org/rfc/rfc2212.txt>.

-
- [76] Shu et al., “Packet loss analysis of the IEEE 802.15. 4 MAC without acknowledgements,” *in: IEEE communications letters* 11.1 (2007), pp. 79–81.
- [77] Srinivasan et al., “The β -factor: measuring wireless link burstiness,” *in: Proceedings of the 6th ACM conference on Embedded network sensor systems*, ACM, 2008, pp. 29–42.
- [78] Sundararaman, Buy, and Kshemkalyani, “Clock synchronization for wireless sensor networks: a survey,” *in: Ad hoc networks* 3.3 (2005), pp. 281–323.
- [79] Tan and Luan, “IEC 61850 based substation automation system architecture design,” *in: 2011 IEEE Power and Energy Society General Meeting*, 2011.
- [80] Teixeira et al., “Dynamics of hot-potato routing in IP networks,” *in: ACM SIGMETRICS Performance Evaluation Review*, vol. 32, 1, ACM, 2004, pp. 307–319.
- [81] Thubert, *6TiSCH requirements for DetNet*, Internet-Draft draft-thubert-6tisch-4DetNet-01, Work in Progress, Internet Engineering Task Force, June 2015, 21 pp., URL: <https://tools.ietf.org/html/draft-thubert-6tisch-4DetNet-01>.
- [82] Thubert, *An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4*, Internet-Draft draft-ietf-6tisch-architecture-11, Work in Progress, Internet Engineering Task Force, Jan. 2017, 53 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-architecture-11>.
- [83] Thubert, *IPv6 Backbone Router*, Internet-Draft draft-ietf-6lo-backbone-router-03, Work in Progress, Internet Engineering Task Force, Jan. 2017, 29 pp., URL: <https://tools.ietf.org/html/draft-ietf-6lo-backbone-router-03>.
- [84] Thubert, “Low power wireless protocols boost industrial networking,” *in: Industrial Ethernet Book Issue 83/24* (2014), URL: <http://www.iebmedia.com/index.php?id=10339&parentid=74&themeid=255&hft=83&showdetail=true>.
- [85] Thubert and Bellagamba, *Available Routing Constructs*, Internet-Draft draft-thubert-rtgwg-arc-03, Work in Progress, Internet Engineering Task Force, Jan. 2015, 21 pp., URL: <https://tools.ietf.org/html/draft-thubert-rtgwg-arc-03>.
- [86] Thubert and Bellagamba, *Generating a loop-free routing topology based on merging buttressing arcs into routing arcs*, US Patent 9,413,638, Aug. 2012.
- [87] Thubert, Bellagamba, and Anteunis, *Hierarchical label distribution and route installation in a loop-free routing topology using routing arcs at multiple hierarchical levels for ring topologies*, US Patent 9,338,086, May 2016.
- [88] Thubert, Bellagamba, and Wijnands, *Bicasting using non-congruent paths in a loop-free routing topology having routing arcs*, US Patent 9,112,788, Aug. 2015.

-
- [89] Thubert, Brodard, and Jiang, *BIER-TE-based OAM, Replication and Elimination*, Internet-Draft draft-thubert-bier-replication-elimination-00, Work in Progress, Internet Engineering Task Force, Sept. 2016, 11 pp., URL: <https://tools.ietf.org/html/draft-thubert-bier-replication-elimination-00>.
- [90] Thubert and Cragie, *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch*, RFC 8025, Nov. 2016, DOI: 10.17487/rfc8025, URL: <https://rfc-editor.org/rfc/rfc8025.txt>.
- [91] Thubert, Enyedi, and Ramasubramanian, *ARC vs. MRT*, Internet-Draft, Work in Progress, Internet Engineering Task Force, Jan. 2015, 20 pp., URL: <https://tools.ietf.org/html/draft-thubert-rtwg-arc-vs-mrt-01>.
- [92] Thubert and Finn, *Deterministic Networking Architecture*, Internet-Draft draft-ietf-detnet-architecture-00, Work in Progress, Internet Engineering Task Force, Sept. 2016, 32 pp., URL: <https://tools.ietf.org/html/draft-ietf-detnet-architecture-00>.
- [93] Thubert and Hui, *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*, RFC 6282, Sept. 2011, DOI: 10.17487/rfc6282, URL: <https://rfc-editor.org/rfc/rfc6282.txt>.
- [94] Thubert and Hui, *LLN Fragment Forwarding and Recovery*, Internet-Draft draft-thubert-6lo-forwarding-fragments-03, Work in Progress, Internet Engineering Task Force, Oct. 2016, 16 pp., URL: <https://tools.ietf.org/html/draft-thubert-6lo-forwarding-fragments-03>.
- [95] Thubert, Palattella, and Engel, “6TiSCH centralized scheduling: When SDN meet IoT,” in: *Standards for Communications and Networking (CSCN), 2015 IEEE Conference on*, IEEE, 2015, pp. 42–47.
- [96] Thubert and Pylakutty, *Root initiated routing state in RPL*, Internet-Draft draft-ietf-roll-dao-projection-00, Work in Progress, Internet Engineering Task Force, Dec. 2016, 14 pp., URL: <https://tools.ietf.org/html/draft-ietf-roll-dao-projection-00>.
- [97] Thubert, Vasseur, and hui, *OAM and Time Slot Control in a Deterministic ARC Chain Topology Network*, US Patent 9,456,444, Sept. 2016.
- [98] Thubert, Vasseur, and Wetterwald, *DISTRIBUTED APPROPRIATION AND DEFENSE OF TIME SLOT OWNERSHIP IN A CHANNEL HOPPING SCHEDULE*, US Patent 20,160,020,864, Jan. 2016.
- [99] Thubert, Vasseur, and Wetterwald, *Distributed rescheduling of bounded flows in a time sensitive network*, US Patent 9,319,332, Apr. 2016.
- [100] Thubert, Vasseur, and Wetterwald, *Exclusive and overlapping transmission unit allocation and use in a network*, US Patent 9,258,098, Feb. 2016.

-
- [101] Thubert, Vasseur, and Wetterwald, *Installation of time slots for sending a packet through an ARC chain topology network*, US Patent 9,320,036, Apr. 2016.
- [102] Thubert, Vasseur, and Wetterwald, *Reassignment of unused portions of a transmission unit in a network*, US Patent 9,356,759, May 2016.
- [103] Thubert, Vasseur, and Wetterwald, *Synchronized routing updates for TSCH networks*, US Patent 9,485,157, Nov. 2016.
- [104] Thubert, Vasseur, and Wetterwald, *Timeslot distribution in a distributed routing protocol for deterministic wireless networks*, US Patent 9,510,3479, Nov. 2016.
- [105] Thubert, Wetterwald, and Vasseur, *LEAPFROG COMMUNICATIONS IN A TSCH NETWORK*, US Patent Pub. 20,160,020,979, Jan. 2016.
- [106] Thubert and Wijnands, *Applying Available Routing Constructs to multicasting*, Internet-Draft draft-thubert-rtgwg-arc-bicast-01, Work in Progress, Internet Engineering Task Force, Oct. 2013, 8 pp., URL: <https://tools.ietf.org/html/draft-thubert-rtgwg-arc-bicast-01>.
- [107] Thubert, Wijnands, and Shepherd, *Reliable multicast in low-power and lossy networks*, US Patent App. 14/661,301, Mar. 2015.
- [108] Thubert, Wijnands, and Shepherd, *USING BIT INDEX EXPLICIT REPLICATION (BIER) IN LOW-POWER AND LOSSY NETWORKS*, US Patent Pub. 20,160,142,248, May 2016.
- [109] Thubert et al., *6LoWPAN Routing Header*, Internet-Draft draft-ietf-roll-routing-dispatch-05, Work in Progress, Internet Engineering Task Force, Oct. 2016, 36 pp., URL: <https://tools.ietf.org/html/draft-ietf-roll-routing-dispatch-05>.
- [110] Thubert et al., *Dynamic installation of local storing mode paths in a non-storing low-power and lossy network*, US Patent 9,479,421, Oct. 2014.
- [111] Thubert et al., *Generating a loop-free routing topology using routing arcs*, US Patent 9,088,502, July 2015.
- [112] Thubert et al., "IETF 6tsch: Combining IPv6 connectivity with industrial performance," in: *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2013 Seventh International Conference on*, IEEE, 2013, pp. 541–546.
- [113] Thubert et al., *Parent device allocation of retransmit slot to child network device on behalf of peer child device in a deterministic network*, US Patent App. 14/642,066, Mar. 2015.
- [114] Thubert et al., *Path shortening in a wireless mesh network*, US Patent 8,111,684, Feb. 2012.
- [115] Thubert et al., *Repair of Failed Network Routing Arcs Using Data Plane Protocol*, US Patent 9,485,136, Nov. 2016.

-
- [116] Thubert et al., *Resilient forwarding of packets in an ARC chain topology network*, US Patent 9,226,292, Dec. 2015.
- [117] Thubert et al., *Reverse directed acyclic graph for multiple path reachability from origin to identified destination via multiple target devices*, US Patent App. 14/643,190, Mar. 2015.
- [118] Vasseur, Thubert, and Wetterwald, *Efficient network probing for deterministic wireless networks*, US Patent 9,350,635, May 2016.
- [119] Vasseur, Thubert, and Wetterwald, *PREDICTIVE TIME ALLOCATION SCHEDULING FOR TSCH NETWORKS*, US Patent 20,160,021,011, Jan. 2016.
- [120] Vasseur, Wetterwald, and Thubert, *DISTRIBUTED ARBITRATION OF TIME CONTENTION IN TSCH NETWORKS*, US Patent Pub. 20,160,021,006, Jan. 2016.
- [121] Vilajosana, Pister, and Watteyne, *Minimal 6TiSCH Configuration*, Internet-Draft draft-ietf-6tisch-minimal-21, Work in Progress, Internet Engineering Task Force, Feb. 2017, 27 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-minimal-21>.
- [122] Vilajosana et al., “A realistic energy consumption model for TSCH networks,” *in: IEEE Sensors Journal* 14.2 (2014), pp. 482–489.
- [123] Vilajosana et al., “OpenMote: Open-Source Prototyping Platform for the Industrial IoT,” *in: 7th EAI International Conference on Ad Hoc Networks (AdHocNets)*, San Remo, Italy: Springer, Jan. 2015, pp. 211–222.
- [124] Vucinic, Simon, and Pister, *Minimal Security Framework for 6TiSCH*, Internet-Draft draft-ietf-6tisch-minimal-security-01, Work in Progress, Internet Engineering Task Force, Feb. 2017, 17 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-minimal-security-01>.
- [125] Wang and Vilajosana, *6top Protocol (6P)*, Internet-Draft draft-ietf-6tisch-6top-protocol-03, Work in Progress, Internet Engineering Task Force, Oct. 2016, 32 pp., URL: <https://tools.ietf.org/html/draft-ietf-6tisch-6top-protocol-03>.
- [126] Watteyne, Mehta, and Pister, “Reliability through frequency diversity: why channel hopping makes sense,” *in: Proceedings of the 6th ACM symposium on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*, ACM, 2009, pp. 116–123.
- [127] Watteyne and Thubert, “Efficient 6LoWPAN Neighbor Discovery applied to Multilink IoT subnets,” *in: 2015 IEEE International Conference on Communications (ICC)*, IEEE, 2015, pp. 642–647.

-
- [128] Watteyne et al., “Industrial ieee802.15.4e networks: Performance and trade-offs,” *in: 2015 IEEE International Conference on Communications (ICC)*, IEEE, 2015, pp. 604–609.
- [129] Watteyne et al., “Mitigating Multipath Fading Through Channel Hopping in Wireless Sensor Networks,” *in: International Conference on Communications (ICC)*, IEEE, Cape Town, South Africa, May 2010.
- [130] Watteyne et al., “OpenWSN: a Standards-based Low-Power Wireless Development Environment,” *in: Transactions on Emerging Telecommunications Technologies (ETT) 23.5* (2012), pp. 480–493.
- [131] Wetterwald, Vasseur, and Thubert, *Configuring new paths in a wireless deterministic network*, US Patent 9,258,097, Feb. 2016.
- [132] Wijnands et al., *Multicast using Bit Index Explicit Replication*, Internet-Draft draft-ietf-bier-architecture-05, Work in Progress, Internet Engineering Task Force, Oct. 2016, 36 pp., URL: <https://tools.ietf.org/html/draft-ietf-bier-architecture-05>.
- [133] Winter et al., *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, RFC 6550, Mar. 2012, DOI: 10.17487/rfc6550, URL: <https://rfc-editor.org/rfc/rfc6550.txt>.
- [134] Yeung, Katz, and Kompella, *Traffic Engineering (TE) Extensions to OSPF Version 2*, RFC 3630, Oct. 2003, DOI: 10.17487/rfc3630, URL: <https://rfc-editor.org/rfc/rfc3630.txt>.
- [135] Zinin, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*, RFC 5286, Sept. 2008, DOI: 10.17487/rfc5286, URL: <https://rfc-editor.org/rfc/rfc5286.txt>.
- [136] Ziouva and Antonakopoulos, “CSMA/CA performance under high traffic conditions: throughput and delay analysis,” *in: Computer communications 25.3* (2002), pp. 313–321.

En s'appuyant sur une connaissance précise du temps, sur la réservation de ressources et l'application distribuée de règles d'admission strictes, un réseau déterministe permet de transporter des flux pré-spécifiés avec un taux de perte extrêmement bas et une latence maximale majorée, ouvrant la voie au support d'applications critiques et/ou temps-réel sur une infrastructure de réseau convergée.

De nos jours, la Technologie Opérationnelle (OT) s'appuie sur des réseaux déterministes mais conçus à façon, en général propriétaires, utilisant typiquement des liens série spécifiques, et opérés en isolation les uns des autres, ce qui multiplie la complexité physique et les coûts d'achat et de déploiement (CAPEX), ainsi que d'opération et de maintenance (OPEX), et empêche l'utilisation agile des ressources.

En apportant le déterminisme dans les réseaux des Technologies de l'Information (IT), une nouvelle génération de réseaux commutés de l'IT va permettre l'émulation de ces liens série et la convergence de réseaux autrefois dédiés sur une infrastructure commune à base d'IP. En retour, la convergence de l'IT et de l'OT permettra de nouvelles optimisations industrielles, en introduisant des technologies héritées de l'IT, comme le BigData et la virtualisation des fonctions du réseau (NFV), en support des opérations de l'OT, améliorant les rendements tout en apportant une réduction supplémentaire des coûts.

Les solutions de réseaux déterministes réclament des possibilités nouvelles de la part des équipements, possibilités qui vont bien au-delà de celles demandées pour les besoins classiques de la QoS.

Les attributs-clé sont :

- la synchronisation précise de tous les nœuds, en incluant souvent la source et la destination des flux
- Le calcul centralisé de chemins de bout en bout à l'échelle du réseau
- De nouveaux filtres de mise en forme du trafic à l'intérieur comme à l'entrée du réseau afin de le protéger en tous points
- Des moyens matériels permettant l'accès au medium à des échéances précises.

Au travers de multiples papiers, de contributions à des standards, et de publication de propriété industrielle, le travail présenté ici repousse les limites des réseaux industriels sans fils en offrant :

1. Le calcul centralisé de chemin complexes basé sur une technologie innovante appelée ARC
2. La signalisation de ces chemins complexes et la traçabilité des paquets par une extension de la technologie BIER-TE
3. Réplication, Renvoi et Elimination des doublons le long de ces chemins complexes
4. Un temps-réel basé sur un échéancier qui assure un haut taux de délivrance et garantit une latence bornée
5. La capacité de transporter à la fois des flux déterministes et du trafic IPv6 à multiplexage statistique sur un maillage 6TiSCH partagé

Ce manuscrit rapporte des améliorations apportées aux techniques existantes des réseaux sans fils à basse puissance (LoWPAN) comme Zigbee, WirelessHART™ et ISA100.11a, afin d'amener ces nouveaux bénéfices jusqu'aux réseaux opérationnels sans fil. Elle a été implémentée en programme et sur du matériel open-source, et évaluée face à du IEEE Std. 802.15.4 classique ainsi que du 802.15.4 TSCH, utilisés en topologie maillée. L'expérience menée montre que notre nouvelle proposition permet d'éviter les à-coups et de garantir des taux élevés de délivrance, même face à des événements exceptionnels comme la perte d'un relais ou la dégradation temporaire d'un lien radio.

Mots-clés: DetNet, Deterministic Networking, 6TiSCH, 6LoWPAN, IEEE Std. 802.15.4, WSN, IOT, Industrial Internet.

Based on time, resource reservation, and policy enforcement by distributed shapers, Deterministic Networking provides the capability to carry specified unicast or multicast data streams for real-time applications with extremely low data loss rates and bounded latency, so as to support time-sensitive and mission-critical applications on a converged enterprise infrastructure.

As of today, deterministic Operational Technology (OT) networks are purpose-built, mostly proprietary, typically using serial point-to-point wires, and operated as physically separate networks, which multiplies the complexity of the physical layout and the operational (OPEX) and capital (CAPEX) expenditures, while preventing the agile reuse of the compute and network resources.

Bringing determinism in Information Technology (IT) networks will enable the emulation of those legacy serial wires over IT fabrics and the convergence of mission-specific OT networks onto IP. The IT/OT convergence onto Deterministic Networks will in turn enable new process optimization by introducing IT capabilities, such as the Big Data and the network functions virtualization (NFV), improving OT processes while further reducing the associated OPEX.

Deterministic Networking Solutions and application use-cases require capabilities of the converged network that is beyond existing QOS mechanisms.

Key attributes of Deterministic Networking are:

- Time synchronization on all the nodes, often including source and destination
- The centralized computation of network-wide deterministic paths
- New traffic shapers within and at the edge to protect the network
- Hardware for scheduled access to the media.

Through multiple papers, standard contribution and Intellectual Property publication, the presented work pushes the limits of wireless industrial standards by providing:

1. Complex Track computation based on a novel ARC technology
2. Complex Track signaling and traceability, extending the IETF BIER-TE technology
3. Replication, Retry and Duplicate Elimination along the Track
4. Scheduled runtime enabling highly reliable delivery within bounded time
5. Mix of IPv6 best effort traffic and deterministic flows within a shared 6TiSCH mesh structure

This manuscript presents enhancements to existing low power wireless networks (LoWPAN) such as Zigbee, WirelessHART™ and ISA100.11a to provide those new benefits to wireless OT networks. It was implemented on open-source software and hardware, and evaluated against classical IEEE Std. 802.15.4 and 802.15.4 TSCH radio meshes. This manuscript presents and discusses the experimental results; the experiments show that the proposed technology can guarantee continuous high levels of timely delivery in the face of adverse events such as device loss and transient radio link down.

Keywords: DetNet, Deterministic Networking, 6TiSCH, 6LoWPAN, IEEE Std. 802.15.4, WSN, IOT, Industrial Internet.



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

N° d'ordre : 2017IMTA0011

IMT Atlantique Bretagne-Pays de la Loire - www.imt-atlantique.fr

Campus de Brest
Technopôle Brest-Iroise
CS 83818
29238 Brest Cedex 03
T +33 (0)2 29 00 11 11
F +33 (0)2 29 00 10 00

Campus de Nantes
4, rue Alfred Kastler - La Chantrerie
CS 20722
44307 Nantes Cedex 3
T +33 (0)2 51 85 81 00
F +33 (0)2 51 85 81 99

Campus de Rennes
2, rue de la Châtaigneraie
CS 17607
35576 Cesson Sévigné Cedex
T +33 (0)2 99 12 70 00
F +33 (0)2 99 12 70 08