



TRABAJO FIN DE GRADO  
INGENIERÍA EN TECNOLOGÍAS DE LA  
TELECOMUNICACIÓN

# Análisis de utilización en Redes TSN Asíncronas

---

Rendimiento en Redes TSN Asíncronas para Industria 4.0

**Autor**

Juan Muñoz Villalón

**Directores**

Pablo Ameigeiras Gutiérrez

Jonathan Prados Garzón



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE  
TELECOMUNICACIÓN

Granada, 10 de Julio de 2023









# Análisis de Utilización en Redes TSN Asíncronas

---

Rendimiento en Redes TSN Asíncronas para Industria 4.0

**Autor**

Juan Muñoz Villalón

**Directores**

Pablo Ameigeiras Gutiérrez

Jonathan Prados Garzón



# **Análisis de Utilización en Redes TSN Asíncronas: Rendimiento en Redes TSN Asíncronas para Industria 4.0**

Juan Muñoz Villalón

**Palabras clave:** TSN, TSN Asíncrono, ATS, Utilización, Rendimiento, PCP, Industria 4.0, 5G, Red TSN, URLL, 3GPP.

## **Resumen**

Se espera que TSN (Time-Sensitive Networking) se convierta en la tecnología de red cableada predominante en la Industria 4.0. TSN es un conjunto de estándares que extienden la funcionalidad de las redes Ethernet (IEEE 802.1Q) para permitir el transporte determinista de flujos, es decir, establecer caminos entre puntos finales de modo que haya garantías deterministas de calidad de servicio (QoS, por sus siglas en inglés) extremo a extremo en términos de latencia, variación de retardo (jitter), tasa de pérdida de paquetes y fiabilidad. Actualmente, las redes cableadas en entornos industriales están basadas en Ethernet Industrial, que comprende un conjunto de estándares de propósito específico y propietarios como EtherCAT y Ethernet POWERLINK. Sin embargo, las redes basadas en Ethernet Industrial ofrecen una baja escalabilidad y son difíciles de interoperar. TSN pone fin a estos problemas, además de ofrecer una mayor facilidad de despliegue y operación, traduciéndose en una reducción de costes. Las redes TSN se pueden clasificar en síncronas y asíncronas. Las redes TSN asíncronas se basan en el planificador ATS (Asynchronous Traffic Shaper) para orquestar la transmisión de paquetes en cada puerto de salida de los conmutadores TSN, de modo que se garantice una QoS determinista y tienen una complejidad más reducida, ya que no requieren que sus componentes (conmutadores TSN) compartan una referencia de tiempo común. Además, las redes asíncronas han recibido menos atención en la literatura científica, dado que su especificación es más reciente. En concreto, la capacidad de las redes TSN asíncronas para acomodar tráfico con requisitos de latencia exigentes en la Industria 4.0 no ha sido totalmente abordado hasta la fecha. Por tanto, el objetivo del presente trabajo se centra en la evaluación del rendimiento de las redes TSN asíncronas en el contexto de la Industria 4.0. Para ello, se realiza un estudio de la arquitectura y operación del ATS y se lleva a cabo una extensa evaluación del rendimiento basada en simulación. Este trabajo toma como punto de partida un simulador y un algoritmo de priorización de flujos para redes TSN asíncronas desarrollados en el grupo *Wireless and Multimedia Networking Lab (WiMuNet)* del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada. En este trabajo se han incluido adaptaciones y extensiones a estas herramientas de partida para

ajustarlas a las necesidades de las evaluaciones llevadas a cabo. Entre los principales resultados de este trabajo se incluye una evaluación de la utilización máxima que pueden alcanzar las redes TSN asíncronas para acomodar tráfico realista en la Industria 4.0. Esta cuestión es vital para determinar la viabilidad para adoptar la tecnología TSN asíncrona en la industria.



# Utilization analysis in Asynchronous TSN Networks: Performance in Asynchronous TSN Networks for Industry 4.0

Juan Muñoz Villalón

**Keywords:** TSN, Asynchronous TSN, ATS, Utilisation, Performance, PCP, Industry 4.0, 5G, TSN Network, URLL, 3GPP.

## Abstract

TSN (Time-Sensitive Networking) is expected to become the predominant wired network technology in Industry 4.0. TSN is an open standards set that extends the functionality of Ethernet networks (IEEE 802.1Q) to enable deterministic transport of flows. This means that there are deterministic end-to-end Quality of Service (QoS) guarantees in terms of latency, jitter, packet loss rate and reliability. Currently, wired networks in industrial environments are based on Industrial Ethernet, which comprises a set of special-purpose and proprietary standards such as EtherCAT and Ethernet POWERLINK. However, Industrial Ethernet-based networks offer low scalability and are difficult to interoperate. TSN puts an end to these problems, as well as offering greater ease of deployment and operation, resulting in reduced costs. TSN can be classified into synchronous and asynchronous networks. Asynchronous TSNs rely on the Asynchronous Traffic Shaper (ATS) scheduler to orchestrate the transmission of packets on each output port of the TSN switches to ensure deterministic QoS. They have a lower complexity as they do not require their components (TSN switches) to share a common time reference. Furthermore, asynchronous networks have received less attention in the scientific literature, since their specification is more recent. In particular, the ability of asynchronous TSN to accommodate traffic with demanding latency requirements in Industry 4.0 has not been fully addressed to date. Therefore, the objective of the present work focuses on the performance evaluation of asynchronous TSNs in the context of Industry 4.0. For this purpose, a study of ATS architecture and operation is carried out and an extensive simulation-based performance evaluation is performed. This work takes as a starting point a simulator and a flow prioritization algorithm for asynchronous TSN networks. Both were developed in the Wireless and Multimedia Networking Lab (WiMuNet) group of the Department of Signal Theory, Telematics, and Communications of the University of Granada. This work has included adaptations and extensions to these initial tools in order to adjust them to the needs of the evaluations carried out. The main results of this work include an evaluation of the maximum utilization that asynchronous TSN networks can achieve to accommodate realistic traffic in

Industry 4.0. This question is vital to determine the feasibility of adopting asynchronous TSN technology in industry.

---

Yo, **Juan Muñoz Villalón**, alumno de la titulación Ingeniería de las Tecnologías de la Telecomunicación de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 78224600J, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Juan Muñoz Villalón

Granada a 10 de Julio de 2023.



---

D. **Pablo Ameigeiras Gutiérrez**, Profesor del Área de Ingeniería Telemática del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada.

D. **Jonathan Prados Garzón**, Profesor del Área de Ingeniería Telemática del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada.

**Informan:**

Que el presente trabajo, titulado *Análisis de Utilización en Redes TSN Asíncronas, Rendimiento en Redes TSN Asíncronas para Industria 4.0*, ha sido realizado bajo su supervisión por **Juan Muñoz Villalón**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a 10 de Julio de 2023.

**Los directores:**

**Pablo Ameigeiras Gutiérrez**

**Jonathan Prados Garzón**



# Agradecimientos

Con la finalización de este trabajo, pongo punto final a estos maravillosos cuatro años de carrera y me gustaría agradecer a todas aquellas personas que me han ayudado a seguir adelante y a disfrutar de esta etapa universitaria en Granada.

En primer lugar, agradecer a mis tutores Pablo y Jonathan por ayudarme en todo momento y guiarme durante todo el proyecto y también agradecer a Julia toda la ayuda e información que me ha prestado, la cual ha sido fundamental para mi trabajo. Gracias por la disponibilidad y por confiar en mí para este proyecto.

También quiero agradecer a todos mis profesores que me han instruido hasta ahora y que han conseguido que la carrera me parezca emocionante y fascinante.

Por supuesto, quiero agradecer a mis amigos que han estado conmigo desde el primer año y me han ayudado a que las clases y el estudio sea más ameno y me siento muy afortunado del grupo que formamos desde el primer año porque considero que gracias a la amistad que hemos forjado, la carrera ha sido mucho más llevadera. En general, agradecer a todos los compañeros que he conocido durante estos cuatro años la amabilidad con la que me habéis tratado.

Por último, y no menos importante, quiero agradecer a mi familia por el apoyo constante durante estos 4 años, en especial a mi Madre, la cual me ha ayudado a seguir adelante y me ha ofrecido la oportunidad de irme al extranjero a estudiar y trabajar para formarme.





# Índice general

<b>1. Introducción</b>	<b>19</b>
1.1. Contexto y motivación . . . . .	19
1.2. Objetivos del proyecto . . . . .	21
1.3. Estructura de la memoria . . . . .	21
<b>2. Estado del Arte</b>	<b>23</b>
2.1. TSN . . . . .	23
2.1.1. Fundamentos de TSN . . . . .	24
2.1.2. Arquitectura de la red TSN . . . . .	25
2.2. TSN Síncrono . . . . .	27
2.3. TSN Asíncrono . . . . .	30
2.4. Requerimientos de tráfico en comunicaciones industriales, TSN. . . . .	32
<b>3. Planificación</b>	<b>37</b>
3.1. Planificación de tareas . . . . .	37
3.2. Recursos . . . . .	41
3.2.1. Recursos Humanos . . . . .	41
3.2.2. Recursos Hardware . . . . .	41
3.2.3. Recursos Software . . . . .	41
3.2.4. Horas Empleadas . . . . .	42
3.3. Presupuesto Estimado . . . . .	42
3.3.1. Costes humanos . . . . .	43
3.3.2. Costes Software . . . . .	43
3.3.3. Costes Hardware . . . . .	43
3.3.4. Costes totales . . . . .	43
<b>4. Diseño e Implementación</b>	<b>45</b>
4.1. Introducción al simulador empleado. Caso de una sola instancia ATS . . . . .	45
4.2. Modelo del Sistema a Considerar . . . . .	46
4.3. Definición del Problema . . . . .	47
4.4. Explicación Pseudocódigo del Algoritmo Desarrollado . . . . .	48
4.5. Propuestas de diseño y análisis en el presente proyecto . . . . .	50

4.5.1. Caso de uso del 5G-ACIA. Obtención de porcentajes de tráfico . . . . .	51
4.6. Adición de Tráfico <i>Best-effort</i> . . . . .	52
<b>5. Simulaciones y Resultados</b>	<b>55</b>
5.1. Preparación Experimental de los Escenarios . . . . .	55
5.1.1. Preparación red con varios ATS . . . . .	56
5.2. Análisis del rendimiento del algoritmo por flujos frente al caso por PCP . . . . .	57
5.2.1. Escenario con un solo ATS . . . . .	57
5.2.2. Escenario en una red con varios ATS . . . . .	60
5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico . . . . .	64
5.3.1. Variación en el tráfico <i>Cyclic-Synchronous</i> . . . . .	64
5.4. Análisis de la utilización para el caso con tráfico <i>best-effort</i> . . . . .	71
<b>6. Conclusiones y Trabajos Futuros</b>	<b>73</b>
<b>Bibliografía</b>	<b>77</b>

# Índice de figuras

2.1. Pilares fundamentales (herramientas) de TSN [1] . . . . .	25
2.2. Modelo completamente centralizada. IEEE 802.1 Qcc [2] . . .	26
2.3. Planificación del tráfico: División del tiempo cíclicamente [3] .	28
2.4. Lista de control de puertas que permite la transmisión de la cola de tráfico 7 sólo en T0 y todas las demás colas de tráfico en T1. [3] . . . . .	29
2.5. Red esquematizada con planificación del tráfico [3] . . . . .	29
2.6. Arquitectura del UBS (a) y del ATS (b) [4] . . . . .	31
2.7. Tipos de tráfico de automatización industrial, requisitos de servicio y funciones TSN relacionadas [5] [3] [6] . . . . .	33
3.1. Diagrama de Gantt que recoge la planificación inicial de todo el proyecto . . . . .	40
4.1. Algoritmo de Priorización en el ATS [7] . . . . .	49
5.1. Estructura de la red a analizar . . . . .	57
5.2. Priorización por PCP para un solo ATS . . . . .	58
5.3. Priorización por flujo para un solo ATS . . . . .	59
5.4. Tiempo de ejecución para el caso por PCP para un solo ATS	59
5.5. Tiempo de ejecución para el caso por flujo para un solo ATS	60
5.6. Priorización por PCP para varios ATS . . . . .	61
5.7. Priorización por flujo para varios ATS . . . . .	62
5.8. Tiempo de ejecución para el caso por PCP para varios ATS .	63
5.9. Tiempo de ejecución para el caso por flujo para varios ATS .	63
5.10. Algoritmo de priorización para un solo ATS. Escenario origi- nal (62.35 %) . . . . .	65
5.11. Algoritmo de priorización para un solo ATS. Escenario 30 % .	65
5.12. Algoritmo de priorización para un solo ATS. Escenario 50 % .	66
5.13. Algoritmo de priorización para un solo ATS. Escenario 80 % .	66
5.14. Algoritmo de priorización para varios ATS. Escenario original (62.35 %) . . . . .	67
5.15. Algoritmo de priorización para varios ATS. Escenario 30 % .	68
5.16. Algoritmo de priorización para varios ATS. Escenario 50 % .	69

5.17. Algoritmo de priorización para varios ATS. Escenario 80 %	70
5.18. Utilización obtenida según el porcentaje de tráfico . . . . .	71
5.19. Utilización obtenida en el caso de considerar tráfico <i>Best-Effort</i>	72

# Índice de cuadros

3.1. Horas empleadas en cada una de las tareas . . . . .	42
3.2. Presupuesto desglosado del presente Proyecto . . . . .	43
4.1. Notación usada en el algoritmo de priorización del ATS . . . .	48
4.2. Notación para el cálculo del número de flujos para cada tipo de tráfico . . . . .	51
5.1. Características de los tráficos considerados [7] . . . . .	56
6.1. Valores de utilización para diferentes simulaciones . . . . .	74
2. Resultados obtenidos para la probabilidad original (62.35 %) .	78
3. Resultados obtenidos para la Probabilidad de 30 % . . . . .	79
4. Resultados obtenidos para la Probabilidad 50 % . . . . .	80
5. Resultados obtenidos para la Probabilidad 80 % . . . . .	81



# Capítulo 1

## Introducción

### 1.1. Contexto y motivación

En la última década, los esfuerzos de la academia y la industria de las comunicaciones han traído importantes estándares de 5G de forma que esta tecnología ya es una realidad a día de hoy. Los despliegues iniciales de 5G se pueden considerar una evolución natural de la cuarta generación (4G) basados en el *Release 15* del 3GPP y orientados a comunicaciones eMBB (*enhanced Mobile Broadband*). Sin embargo, la llegada del *Release 16* es el verdadero desafío, el cual introduce nuevas funcionalidades con altas expectativas. Aquí se hablan de servicios con características de latencia y fiabilidad que no se habían visto antes en redes inalámbricas, lo que abre las puertas a nuevos horizontes de aplicación, como puede ser la industria 4.0. En el *Release 16* se hablan de comunicaciones URLLCs (*Ultra Reliable Low Latency Communications*), es decir, comunicaciones ultra fiables y de muy baja latencia para el acceso radio. Se habla de una transformación digital, la cual va a traer aplicaciones y casos de uso innovadores que van a representar una revolución en los próximos años. En esta transformación digital hay varias tecnologías que están llamadas a jugar un papel relevante.

Por un lado, Cloud and Edge Computing, que permitirá ejecutar la inteligencia/procesamiento de robots, sensores, actuadores, controladores desde el Edge/Cloud. Para poder hacerlo hace falta una red que transfiera la información de forma ultrarápida y de forma fiable desde y hacia el cloud.

Por eso, especialmente en el caso de la industria manufacturera, está tomando fuerza la tecnología TSN, que se espera que se convierta en el pilar fundamental de la industria 4.0. TSN (*Time-Sensitive Networking*) es una serie de estándares especificados por el IEEE 802 para la planificación del tráfico de manera determinista, de modo que se cumplan con unas calidades de servicio determinadas, como pueden ser la baja latencia y ultra fiabilidad que se quieren conseguir con 5G inalámbrico.

Por otro lado, tenemos la tecnología 5G/6G que prometen eliminar los

cables y habilitar las aplicaciones y casos de uso mencionados arriba. El objetivo del 5G/6G son las comunicaciones URLLC que van a permitir comunicaciones en tiempo real ultrarrápidas y con latencia mínima que van a abrir nuevas aplicaciones como la comunicación en tiempo real entre vehículos autónomos, sistemas de gestión de tráfico y dispositivos de monitoreo; mejor calidad de las llamadas de voz, videoconferencias más fluidas y una experiencia de usuario mejorada en aplicaciones de mensajería y redes sociales; implementación de fábricas *inteligentes* con maquinaria y robots conectados en tiempo real así como la creación de redes de sensores más extensas y la implementación de soluciones inteligentes en áreas como la agricultura, la salud, la logística y las ciudades inteligentes.

Se ha hablado de TSN (*Time-Sensitive Networking*) como un pilar fundamental para la industria 4.0. Existen 2 tipos de TSN, TSN síncrono y TSN asíncrono. Este trabajo se va a centrar en el caso asíncrono, donde no necesitamos de la sincronización global de la red para el reenvío y la planificación del tráfico, además de ser más sencillo de implementar. Más adelante se describirán las diferencias entre ambos con más detalle.

Existe una propuesta [7] donde se indica como realizar la planificación de paquetes en redes TSN asíncronas proponiendo un algoritmo que se centra en un modelo de retardo para flujos que está incluido en el propio estándar del ATS. (*Asynchronous Traffic Shaper*), el cual es el planificador de paquetes en TSN asíncrono. El algoritmo busca una priorización para los flujos en una instancia ATS, de modo que se cumpla con los requisitos de latencia deterministas de todos los flujos. Sin embargo, dicha propuesta inicial [7] requiere de una serie de cuestiones para ser consolidada y es ahí donde radica la motivación del presente trabajo. Dichas cuestiones son las siguientes:

1. Tomando como punto de partida la solución descrita arriba, analizar la capacidad de las redes TSN asíncronas para acomodar tráfico con requisitos de latencia determinista en la Industria 4.0. En concreto, se evaluará la utilización máxima que alcanzan los enlaces de la red TSN asíncrona para acomodar el tráfico sensible al retardo que se prevé en entornos industriales.
2. Comparar la capacidad de las redes TSN asíncronas para acomodar tráfico sensible al retardo cuando la priorización se lleva a cabo a nivel de PCP y a nivel de flujo. La evaluación en [7] se hizo para un conjunto de tipos de tráfico que no se ajustaban al escenario industrial. Sería conveniente realizar simulaciones con tráficos según un caso de uso.
3. No hay una evaluación del impacto de los distintos parámetros del tráfico en la capacidad máxima.
4. Es muy conveniente saber si la capacidad que queda libre tras acomodar el tráfico URLLC puede ser empleada para tráfico *Best Effort*.



## 1.2. Objetivos del proyecto

Los objetivos principales a abordar en el proyecto son los siguientes:

1. *Análisis de la utilización máxima de los enlaces para el caso de un solo ATS y para el caso de una red con distintos nodos:* Se pretende analizar la utilización que tendríamos en el escenario de un solo enlace para el caso de 7 clases de tráfico típicas de un caso de uso de industria 4.0 con integración de 5G. También se analizará el caso de tener distintos nodos y analizar el rendimiento general de la red. Para ello, se hará uso del algoritmo propuesto en [7] para el análisis del número de flujos que son capaces de redirigirse y, por tanto, las limitaciones que presenta dicho módulo para redirigir tráfico, cumpliendo en todo caso con las calidades de servicio necesarias.
2. *Comparación del escenario anterior con el escenario en que distinguimos por flujo:* en lugar de agregar todos los flujos de una misma clase como se propone en [7]. Se compararán las utilidades obtenidas para ver que escenario es más eficiente.
3. *Análisis de la utilización usando esta vez las probabilidades obtenidas a partir del paper de 5G-ACIA [8]:* Se estudiará con profundidad el caso real propuesto en dicho documento y aplicaremos los porcentajes de flujo de cada clase al modelo propuesto para obtener la utilización y así compararla con los escenarios anteriores.
4. *Simulaciones con barrido en la probabilidad de flujo para la clase Cyclic-Synchronous:* El objetivo es variar el porcentaje del tráfico del tipo *Cyclic-Synchronous*, ya que es el más restrictivo respecto al retardo y es de interés analizar como esta variación del tráfico afecta al rendimiento del sistema, así como a la utilización máxima obtenida.
5. *Simulaciones y análisis del rendimiento para un escenario donde se añadiría tráfico Best-Effort:* así como el estudio y la obtención de conclusiones respecto a la adición de este tipo de tráfico tras acomodar el tráfico URLLC. Ya que existe un elevado porcentaje de la capacidad que no se está utilizando para acomodar tráfico URLLC por lo que sería interesante la opción de poder utilizar dicha capacidad para redirigir tráfico *best-effort*.

## 1.3. Estructura de la memoria

En esta sección se enumeran los distintos capítulos que tiene esta memoria, así como una breve descripción de cada uno de ellos:

1. **Capítulo 1. Introducción:** En esta parte vamos a abordar 3 temas: Un contexto y motivación donde se presenta el objetivo general de la siguiente memoria, unos objetivos del proyecto a cumplir y a estudiar durante el desarrollo del proyecto y la estructura de la memoria que ahora mismo se está presentando.
2. **Capítulo 2. Estado del arte:** Aquí se presentará el contexto teórico en el que se desarrolla la presente memoria, así se mencionaran trabajos e investigaciones realizados con anterioridad y relacionado con el proyecto que se ha llevado a cabo.
3. **Capítulo 3. Planificación:** En esta parte de la memoria, se presenta la planificación y fases que se han llevado a cabo durante el desarrollo del trabajo, así como un diagrama de Gantt y los recursos empleados.
4. **Capítulo 4. Diseño e Implementación:** En este capítulo se introduce y explica el diseño experimental realizado para poder llevar a cabo las evaluaciones del presente trabajo.
5. **Capítulo 5. Simulaciones y resultados:** Aquí se presentan las simulaciones y resultados obtenidos de dichas simulaciones, así como la discusión de los resultados y conclusiones que se extraen de los mismos.
6. **Capítulo 6. Conclusiones y trabajos futuros:** Por último se presenta en esta sección un resumen general del trabajo mostrado, así como posibles líneas de investigaciones relacionadas con las conclusiones obtenidas a partir de los resultados.

## Capítulo 2

# Estado del Arte

En este apartado se va a revisar el estado del arte en el ámbito del problema propuesto. Se analizará de cerca en qué consiste la tecnología TSN (Time Sensitive Networking) así como la diferencia entre TSN síncrono y TSN asíncrono. Para cada uno, se verán sus características clave y como estos funcionan a la hora de reenviar el tráfico por la red. Después se estudiarán los tipos de tráfico que tenemos en TSN, ya que será relevante para nuestra propuesta experimental. Hay que tener en cuenta que tenemos distintos parámetros para los modelos de tráfico existentes en TSN.

### 2.1. TSN

La tecnología Ethernet [9] es capaz de cumplir y satisfacer las necesidades de comunicación en Internet de un gran número de aplicaciones de propósito general y se ha convertido en el medio de transmisión de datos por excelencia, además de ser muy importante en el mundo de Internet de las Cosas (IoT). La mayoría del tráfico IP generado por páginas web, audio/video en streaming, datos de dispositivos, IoT y otros tráficos similares son lo que se denomina *best-effort* y se comporta muy bien sobre Ethernet o Wi-Fi. Sin embargo, hasta hace poco no se hablaba de soporte en tiempo real para el estándar Ethernet. Las comunicaciones en tiempo real son esenciales para la industria y por eso es por lo que surgen varios protocolos como *EtherCAT*, *PROFINET* o *Sercos III* que tratan de cubrir las necesidades de tiempo real que le faltan a Ethernet, sin embargo, no consiguen muy bien del todo la completa conexión entre dispositivos y redes basadas en Ethernet, por lo que sufren de limitaciones de compatibilidad. A raíz de estos problemas que van surgiendo, desde el 2012, el grupo del IEEE encargado del protocolo TSN ha estado activo y trabajando para traer una solución a dichos problemas.

### 2.1.1. Fundamentos de TSN

TSN describe numerosos mecanismos para asegurar y mejorar las comunicaciones en tiempo real sobre Ethernet, además de definir el primer estándar (IEEE 802.1 Qbv) para el reenvío de forma sincronizada de tráfico en redes Ethernet conmutadas. TSN asegura una calidad de servicio (QoS) determinista para las aplicaciones y tráfico en tiempo real sobre Ethernet. La clave está en que se segrega el tráfico en clases con diferentes requerimientos de latencia, *jitter*, pérdida de paquetes o *bit-rate* para que después sean tratados de manera diferente en los conmutadores TSN (TSN *bridges* o *switches*) pertinentes. TSN utiliza un reloj global que sincroniza los distintos dispositivos en la red para poder así planificar de forma síncrona el envío del tráfico a través de los diferentes elementos de la red. De forma que definiendo unas colas, las cuales reenvían el tráfico de acuerdo a un esquema de tiempo, TSN consigue garantizar un retardo determinado para tráficos con unos tiempos de latencia críticos. En esto consiste TSN síncrono. La otra variante es TSN asíncrono, en la cual nos centramos en este trabajo. La diferencia fundamental es que en TSN asíncrono no requiere la sincronización de tiempo de los conmutadores TSN mediante un reloj global. Más adelante, se van a analizar en más profundidad estas dos variantes. Más concretamente, nos centraremos en los planificadores TAS (*Time-Aware Shaper*) y ATS (*Asynchronous Traffic Shaper*), que son los bloques básicos para las redes TSN síncronas y asíncronas, respectivamente.

En la figura 2.1 se pueden apreciar los pilares fundamentales de TSN para proporcionar una QoS determinista, los cuales se describen a continuación:

- **Planificación del tráfico:** Uso de colas para planificar los diferentes tipos de tráfico según las calidades de servicio (QoS).
- **Sincronización del tiempo:** Uso de un reloj global para la red para la sincronización de los planificadores en cada nodo de la red de forma que la latencia sea mínima y se cumplan con los requisitos de *delay*.
- **Gestión de los recursos:** Como se verá más adelante, TSN considera una gestión centralizada, parcialmente centralizada y totalmente distribuida. La que puede tener mayor interés para automatizar su operación es la totalmente centralizada y por eso nos centremos en ella aquí.
- **Fiabilidad:** La alta disponibilidad, como resultado de la ultra fiabilidad, se proporciona mediante la replicación y eliminación de tramas para la Fiabilidad (FRER, de sus siglas en inglés) ([10]) para flujos de datos mediante un mecanismo de fiabilidad por paquete. En otras palabras, TSN permite usar múltiples caminos independientes para enviar el tráfico de un mismo flujo y así aumentar la fiabilidad de la comunicación.

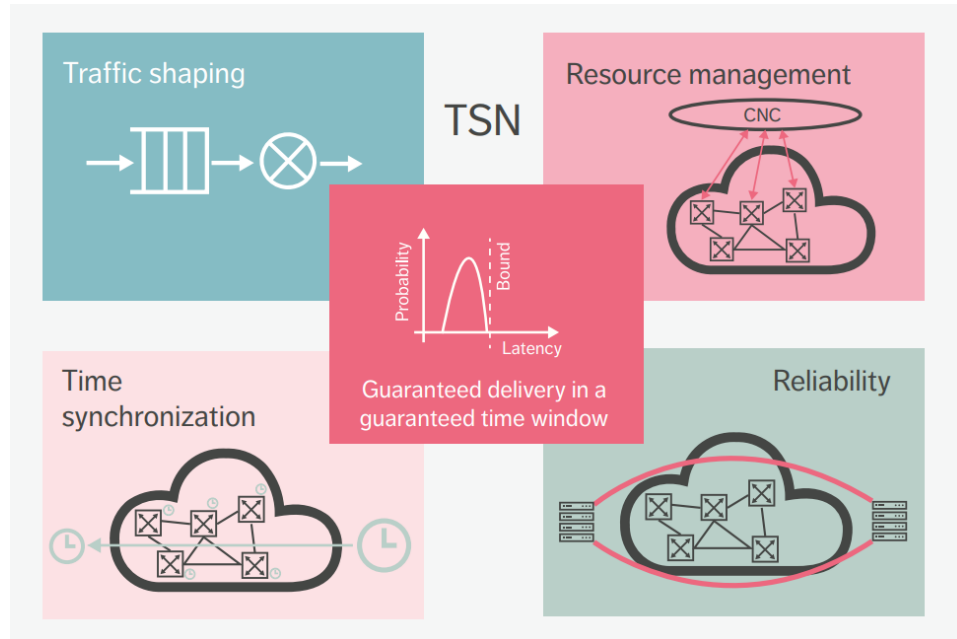


Figura 2.1: Pilares fundamentales (herramientas) de TSN [1]

### 2.1.2. Arquitectura de la red TSN

Como ya hemos comentado, TSN síncrono tiene como objetivo comunicaciones deterministas donde la sincronización del tiempo es crucial. Su objetivo es conseguir latencias acotadas dividiendo el tráfico en clases de tráfico (TC) y asignándoles *slots* de tiempo con repeticiones cíclicas. TSN también introduce reserva dinámica de *streams* para el tráfico determinista. En el estándar IEEE 802.1 Qcc - *Stream Reservation Protocol (SRP) Enhancements and Performance Improvements* [2] se definen diferentes despliegues de arquitectura en TSN, así como una interfaz para solicitudes de *streams* y un módulo para el cálculo de la planificación de los dispositivos.

Podemos diferenciar 3 arquitecturas:

- **Modelo totalmente distribuido:** En este modelo sin entidades centralizadas, las aplicaciones pueden solicitar la reserva de sus *streams* directamente sobre la red, propagando sus peticiones en la topología usando un protocolo UNI. Cada puente TSN se configura él solo a partir de los requerimientos especificados en las peticiones y la información conocida de la red para poder cumplir con los *streams* solicitados.
- **Red centralizada/modelo de usuario distribuido:** Debido a la complejidad computacional que surge por los numerosos dispositivos y por ende, *streams*, surge una entidad denominada *Centralized Network Con-*

*figuration* (CNC). Esta entidad conoce globalmente todos los dispositivos y streams de la red. La diferencia con el modelo anterior es que las peticiones de las aplicaciones finales son reenviadas por los primeros puentes TSN al CNC, el cual configura y gestiona todos los nodos de la red después de terminar la generación de las GLC (Gate Control List) de los nodos. El se encarga de asegurar la calidad de servicio (QoS) requerida para los flujos entrantes y salientes buscando la mejor configuración que aplicar a los *bridges TSN*.

- Modelo totalmente centralizado: Para casos de uso más complejos, donde los usuarios finales también tienen que ser configurados, el modelo CUC (Centralized User Configuration) es introducido. Esta entidad se encarga de descubrir usuarios finales y sus capacidades, de forma que se entera de los requisitos de aplicación que necesitan dichos usuarios y configura características TSN en ellos. El CUC reenvía toda la información hacia el CNC. En la figura 2.2 podemos ver dicha estructura.

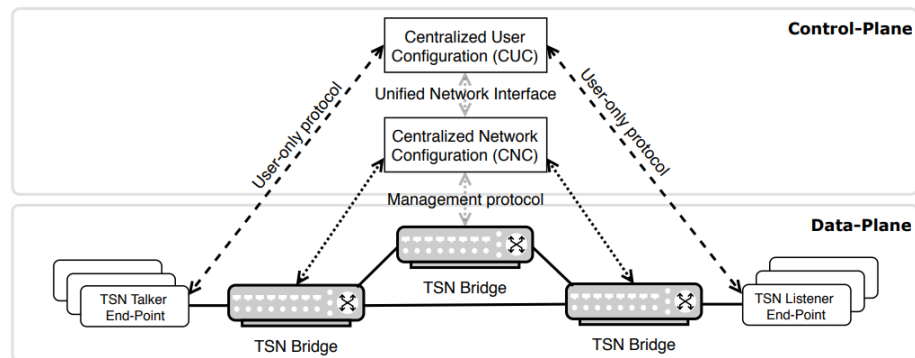


Figura 2.2: Modelo completamente centralizada. IEEE 802.1 Qcc [2]

### A. Separación e integración del plano de datos

Es importante mencionar que gracias a SDN (*Software Defined Networking*) podemos desacoplar el plano de datos del plano de control, los cuales están generalmente en los mismos dispositivos. El plano de datos define como las tramas o paquetes son reenviados en un dispositivo. Este plano de datos se puede integrar con non-TSN Ethernet, aunque la calidad de servicio solo se puede asegurar dentro de la red TSN. Podríamos tener el tráfico URLL transportado sobre la red TSN, mientras que el tráfico *best-effort* podría ir sobre SDN-Ethernet. Eso sí, ambos controladores deben permanecer completamente segregados. En un dominio SDN, todos los conmutadores SDN

están conectados a un controlador SDN lógicamente centralizado. Las reglas de reenvío se especifican en el plano de aplicación, donde las aplicaciones para diferentes propósitos deciden cómo enrutar el tráfico. Si se requiere que parte del tráfico *best-effort* atraviese tanto la red TSN como la red estándar, puede haber aplicaciones virtuales que se ejecuten sobre el controlador SDN para emitir las solicitudes necesarias al controlador TSN. Por tanto, el uso del protocolo OpenFlow [11] es dominante y generalmente, soportado por todos los conmutadores SDN.

## 2.2. TSN Síncrono

Como ya hemos comentado anteriormente, la sincronización del tiempo es crucial para asegurar el comportamiento determinista de los dispositivos finales. El estándar usado para la sincronización del tiempo es el *IEEE 802.1AS generalized Precision Time Protocol (gPTP)* [12] que permite la sincronización del tiempo sobre Ethernet. Más adelante se explicará como la sincronización mediante un reloj global es necesario para la planificación del tráfico, además de que los diferentes conmutadores TSN de la red deben operar sobre el mismo tiempo para no aumentar el retardo añadido. La sincronización del reloj requerida en *gPTP* está definida en el IEEE 1588 [13]: Una instancia *máster PTP* envía información (incluyendo la marca de tiempo actual) a todos los dispositivos conectados a dicho máster usando Ethernet *multicast*. Los dispositivos PTP que reciben dicho tiempo deben añadir la corrección del tiempo de propagación que existe entre el *grand-master* y dichas Instancias PTP.

En TSN síncrono se habla de *Time-Aware Shaper (TAS)*, que es el planificador más sofisticado que tenemos para esta variante de TSN. IEEE 802.1 Qbv Enhancements to Traffic Scheduling [14] persigue la transmisión de tráfico crítico con cero-jitter a través de cada puerto de salida del conmutador. Este estándar habla de dividir el tiempo en *slots* para así poder servir las diferentes clases de tráfico determinadas por su prioridad en el campo PCP (Priority Code Point) del *tag VLAN*.

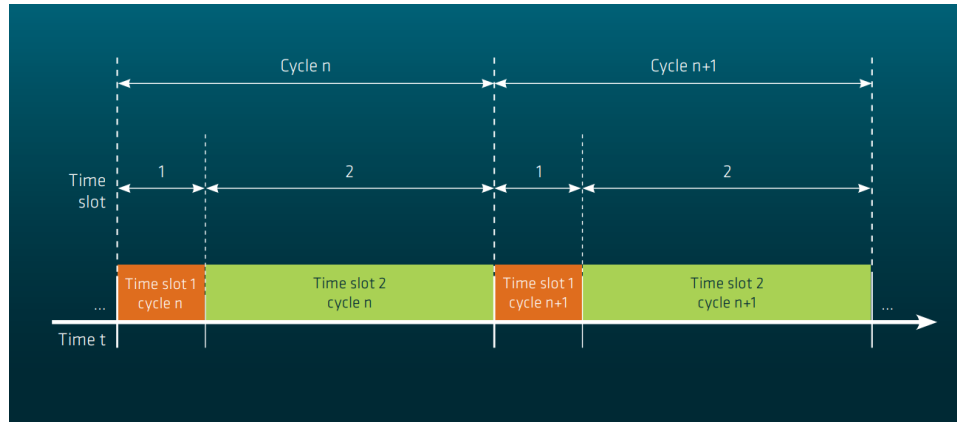


Figura 2.3: Planificación del tráfico: División del tiempo cíclicamente [3]

En la figura 2.3 se aprecia como el tiempo se divide en diferentes *slots* que se repiten cíclicamente. Una clase de tráfico se puede seleccionar para que solo se transmite en dicho *slot* dicha clase de tráfico. Las tramas que pertenecen a otras clases de tráfico deben esperar a que se les permita transmitir. De esta forma, en cada conmutador se planifica el tiempo de diferente manera, de forma que si se coordinan los planificadores de los diferentes conmutadores, se puede conseguir y asegurar que las clases de tráfico con requisitos de latencia más críticos puedan transmitir sin sufrir ningún retardo en colas. El mecanismo de bloqueo de colas siguiendo una planificación temporal con una puerta para cada cola se muestra en la figura 2.4.

En la figura 2.5 se puede observar esta planificación del tráfico en diferentes conmutadores. En cada conmutador tenemos un ciclo que se repite y se puede apreciar como las tramas de diferentes colores, de acuerdo a sus prioridades se transmiten en diferentes *slots* de dicho ciclo.



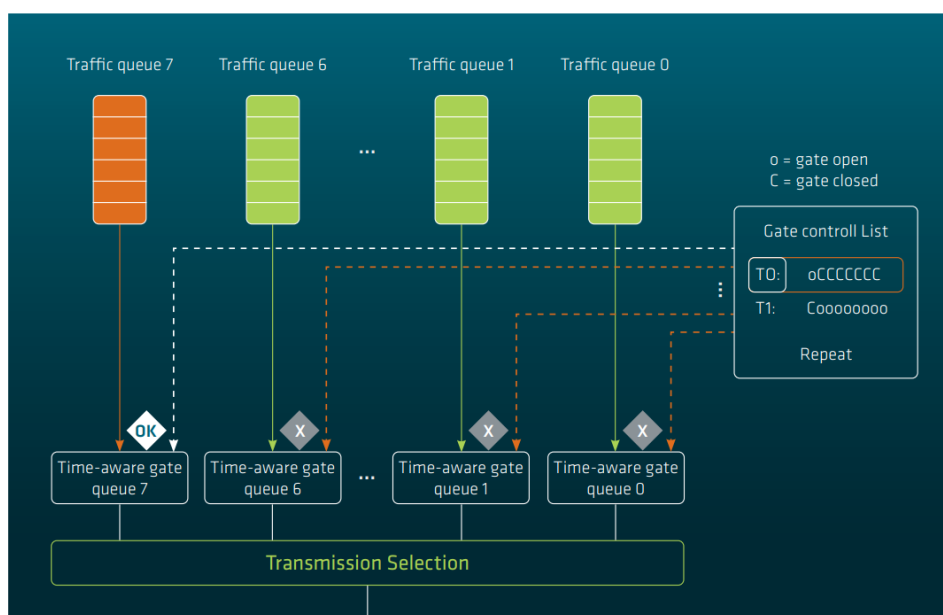


Figura 2.4: Lista de control de puertas que permite la transmisión de la cola de tráfico 7 sólo en T0 y todas las demás colas de tráfico en T1. [3]

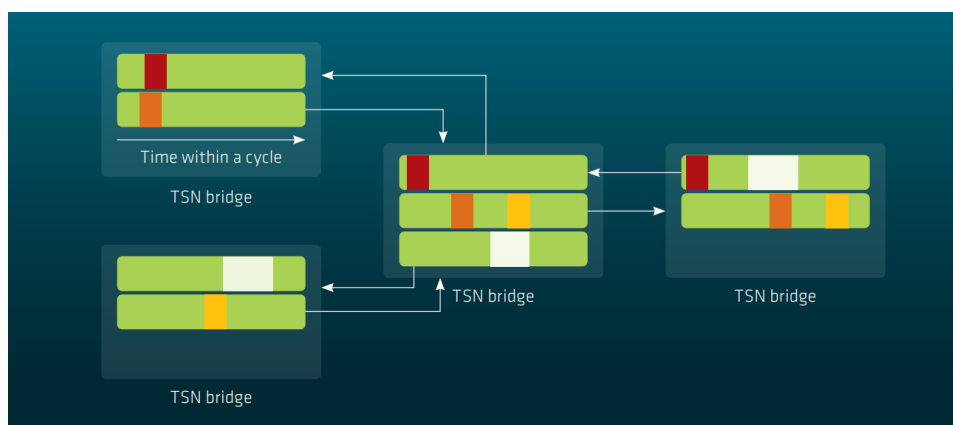


Figura 2.5: Red esquematizada con planificación del tráfico [3]

En cuanto a los inconvenientes de TSN síncrono, los podemos resumir en los siguientes puntos:

- Problema en cuanto a la sincronización. No es del todo sencillo implementar la sincronización necesaria para el funcionamiento del modelo y el problema es mayor a la hora de integrar TSN y 5G.

- Planificación bastante compleja, lo cual a su vez hace que sea difícilmente escalable.
- Solo aplicable al tráfico periódico y muy regular.
- Presenta dificultades para integrarse con 5G debido al jitter que introduce 5G al transferir los paquetes.

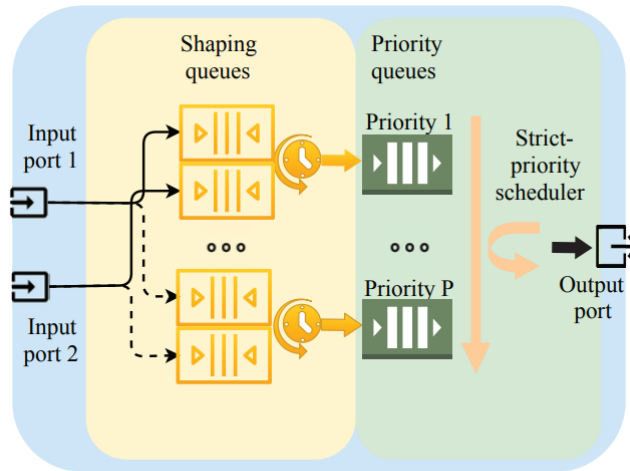
## 2.3. TSN Asíncrono

En la sección previa se ha comentado el TSN síncrono y aquí se va a analizar su variante, el TSN asíncrono. A diferencia del TSN síncrono, en este no se necesita una coordinación y sincronización del tiempo en la red para planificar el tráfico mediante un reloj global, como comentábamos en el apartado anterior. Esto mejora nuestra escalabilidad y la utilización del canal, aunque se aumenta la latencia en comparación con la arquitectura síncrona. TSN asíncrono es ideal para el transporte de tráfico esporádico con diferentes requisitos de tiempo real.

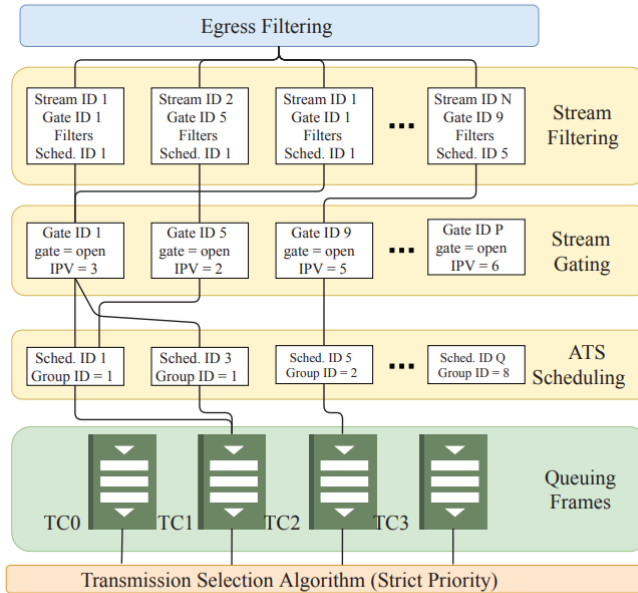
El bloque del TSN asíncrono es el IEEE 802.1Qcr Asynchronous Traffic Shaper (ATS) [15], el cual se basa en el UBS (Urgency-Based Shaper) propuesto por *Specht* y *Samii* [16]. El ATS define un método asíncrono para planificar el reenvío de las tramas en los puertos de salida de los conmutadores TSN[17]. En la figura 2.6 se esquematiza ambos UBS y ATS. En la figura 2.6a se muestra el modelo de encolado del UBS. Solo se muestra un puerto de salida por simplicidad, pero se debe apuntar que existe una instancia UBS por puerto de salida. El UBS consiste de 2 etapas, tenemos una primera etapa de colas de conformación de tráfico y una segunda etapa con colas de prioridades. Todas estas siguen una disciplina FCFS (First-Come First-Served).

La conformación intercalada del tráfico es lo novedoso y fundamental que subyace al ATS para lograr una regulación del tráfico por flujo y por salto de forma rentable. Se utiliza una sola cola (cola de conformación) para realizar la regulación del tráfico de una serie de flujos, cada uno con sus diferentes requisitos. Para conseguir este fin, solo se comprueba la elegibilidad de la trama HOL (head-of-line), es decir, que se examina si la trama HOL puede ser transmitida de acuerdo a las limitaciones de regulación de su tipo de tráfico. Lo realmente interesante es que *Specht* y *Samii* demuestran que la conformación intercalada no aumenta la latencia en el UBS [16] [18]. UBS además soporta el uso de conformación por flujo mediante cubo de fugas (*leaky bucket*) para asegurar la velocidad de datos y el tamaño de ráfaga para cada flujo.

La asignación de buffers sigue una serie de reglas denominadas *queue allocation rules* (QARs) y que en resumen dicen que cada cola de conformación puede ser asociada a un solo puerto de entrada (QAR1) para evitar la



(a)



(b)

Figura 2.6: Arquitectura del UBS (a) y del ATS (b) [4]

propagación de sobrecarga de tráfico no conformador, a un nivel de prioridad en el salto anterior (QAR2) y a un nivel de prioridad interno (QAR3). Estas últimas 2 reglas son necesarias para la QoS. Estas reglas definen el número de colas de conformación necesarias para  $P$  niveles de prioridad. Como forma de ilustración, un ATS que recibe tráfico desde  $N$  puertos de entrada y que tiene  $P$  niveles de prioridad internos, necesita  $P * N$  colas de conformación.

La segunda etapa en la arquitectura UBS consiste en una cola FCFS

por cada nivel de prioridad en el planificador. Cada cola recoge el tráfico de salida de todas las colas de conformación asignadas al mismo nivel de prioridad. El algoritmo de selección para transmitir en esta etapa es el de prioridades estrictas, es decir, una clase de tráfico tiene prioridad a la hora de transmitir sobre clases de tráfico con menor prioridad.

La figura 2.6b muestra la arquitectura del ATS y su modo de operación. El ATS es la implementación de un UBS en el estándar de TSN y el proceso de reenvío consiste de los siguientes pasos:

- *Stream filtering.* Esto se refiere al paso previo de filtrado y clasificación de los paquetes que entran al ATS. Cada trama es *mapeada* a un filtro usando su prioridad y otros campos para que después cada clase de tráfico tenga asociado un planificador ATS.
- *Stream Gating.* Este paso es opcional y asigna un nivel de prioridad interno (IPV) que sobrescribe al nivel de prioridad actual de una trama en el paso 4. El objetivo es ajustar la latencia por salto en el ATS.
- *Planificador ATS.* En este paso se realiza la conformación explicada anteriormente.
- *Encolado y transmisión.* Como en el paso 2 del UBS, las tramas que provienen del mismo filtro (clase de tráfico) son encoladas en sus respectivos niveles de prioridad para ser transmitidas de acuerdo a una política de prioridad estricta.

## 2.4. Requerimientos de tráfico en comunicaciones industriales, TSN.

Una red de comunicaciones industriales basada en TSN consiste en una red que mezcla y porta diferentes tipos de tráfico con calidades de servicio (QoS) diversas. Se puede tener desde tráfico *best-effort* hasta tráfico crítico de baja latencia como puede ser URLLC. Varias organizaciones (3GPP[19], IEC/IEEE[5], IIC[6], etc) han definido una serie de tipos de tráfico junto con sus requerimientos en casos de industria 4.0 y se han recogido en la tabla 1. Estas clases de tráfico se han tenido en cuenta a la hora de la experimentación en el diseño propuesto.

Traffic types	Periodic / Sporadic	Typical period	Data delivery guarantee	Tolerance to jitter	Tolerance to loss	Typical data size (Byte)	Criticality	Traffic priorities (VLAN PCP)
<b>Isochronous</b>	P	100 $\mu$ s ~ 2 ms	Deadline	0	None	Fixed: 30 ~ 100	High	6
<b>Cyclic -Synchronous</b>	P	500 $\mu$ s ~ 1 ms	latency bound ( $\tau$ )	$\leq \tau$	None	Fixed: 50 ~ 1000	High	5
<b>Cyclic -Asynchronous</b>	P	2 ms ~ 20 ms	latency bound ( $\tau$ )	$\leq \tau$	1 ~ 4 Frames	Fixed: 50 ~ 1000	High	5
<b>Events: control</b>	S	10 ms ~ 50 ms	latency bound ( $\tau$ )	n.a.	Yes	Variable: 100 ~ 200	High	4
<b>Events: alarm &amp; operator com-mands</b>	S	2 s	latency bound ( $\tau$ )	n.a.	Yes	Variable: 100 ~1500	Medium	3
<b>Network control</b>	P	50 ms ~ 1 s	throughput	Yes	Yes	Variable: 50 ~ 500	High	7
<b>Configuration &amp; diagnostics</b>	S	n.a.	throughput	n.a.	Yes	Variable: 500 ~ 1500	Medium	2
<b>Video</b>	P	Frame Rate	throughput	n.a.	Yes	Variable: 1000 ~ 1500	Low	1
<b>Audio/Voice</b>	P	Sample Rate	throughput	n.a.	Yes	Variable: 1000 ~1500	Low	1
<b>Best effort</b>	S	n.a.	None	n.a.	Yes	Variable: 30 ~ 1500	Low	0

Figura 2.7: Tipos de tráfico de automatización industrial, requisitos de servicio y funciones TSN relacionadas [5] [3] [6]

El tipo de tráfico puede ser esporádico, donde por ejemplo una aplicación puede enviar un mensaje o no cada cierto tiempo o periódico, donde los mensajes se transmiten de forma regular siguiendo un patrón cíclico.

*Data delivery guarantee* sirve de guía para seleccionar la mejor QoS en Ethernet para la transmisión de datos de la aplicación en cuestión. Un paquete debe cumplir con esta garantía mínima para que no se considere perdido o descartado. Tenemos 3 tipos si nos fijamos en la tabla:

- **Deadline:** Los paquetes deben de llegar con una latencia determinada crítica. Los tipos de tráfico con esta característica no toleran de ninguna forma variaciones en el retardo y se debe cumplir el margen que se indica de latencia.
- **Latencia:** Aquí se define unos márgenes de retardo y que los paquetes deben cumplir para que no se descarten.
- **Throughput:** La entrega de paquetes está garantizada si se cumple con el límite de *throughput* establecido.

La tolerancia a *jitter* determina la tolerancia de una aplicación a la variación de retardo extremo a extremo en la transmisión de sus tramas. En general, esta suele ser menor que la latencia a garantizar.

La tolerancia a pérdidas indica el grado de pérdida consecutiva de paquetes, o paquetes que no cumplen con la latencia garantizada.

El tamaño típico de paquetes hace referencia al tamaño del mensaje que se encapsula en el *payload* de la trama Ethernet.

El PCP (Priority Code Point) es un campo muy importante en el trabajo propuesto y hace referencia al campo del *tag VLAN* que indica la prioridad del paquete. Se tienen 8 valores para dicha prioridad, donde el 0 es el menos prioritario y el 7 el mayor. Dependiendo de las QoS de cada clase de tráfico, su PCP variará.

La criticidad en la tabla hace referencia a como de crítico es el retardo o *delay* de una clase de tráfico determinada. Tenemos 3 tipos:

- **Alto (ej. PCP 4 a 7):** Para tipos de tráfico que pertenecen bien a aplicaciones o servicios de red que necesitan de una latencia crítica de operación. Son aquellos tipos de tráfico donde la pérdida de paquete puede causar problemas en el funcionamiento del sistema y no se contempla la repetición o retransmisión por parte de la fuente.
- **Medio (ej. PCP 2 y 3):** Aquí tenemos aquellas clases de tráfico que son relevantes pero no continuamente necesarias para la parte crítica de operación del sistema. La pérdida de paquetes puede ocasionar retrasos y degradación en el funcionamiento del sistema, pero se tolera ciertas pérdidas.

- Bajo (ej. PCP 0 y 1): Para tipos de tráfico que no son relevantes a la hora del funcionamiento crítico del sistema y donde se tolera perfectamente la pérdida de datos. Aquí podemos tener tráfico *best-effort*





## Capítulo 3

# Planificación

En este capítulo, primero se expondrá la planificación de las tareas llevadas a cabo durante el proyecto presente y después se expondrán los recursos utilizados para la realización de dicho proyecto.

### 3.1. Planificación de tareas

- **Fase de estudio y análisis inicial:**

1. *Estudio de TSN y el estado del arte:* Como primera tarea, fue imprescindible estudiar e informarse acerca de la tecnología TSN y el contexto en el que se implementa, así como su relación con 5G y la industria 4.0.
2. *Estudio del simulador y el algoritmo priorización de flujos en instancias ATS scheduler:* Esta tarea se refiere a la de familiarizarse con el código MATLAB del simulador de redes TSN asíncronas y el algoritmo de priorización usados como punto de partida en este trabajo. Esto incluye lanzar simulaciones preliminares para comprender los parámetros de entrada y salida proporcionados por estas herramientas.

- **Fase de diseño e implementación del algoritmo por flujos para un solo enlace:**

1. *Modificación del algoritmo para el análisis y priorización a nivel de flujos en instancias ATS:* El algoritmo de priorización tomado como punto de partida llevaba a cabo una priorización por clase de tráfico. En este trabajo se incluyeron las modificaciones necesarias para que soportase una priorización a nivel de flujo.
2. *Implementación de la utilización:* Modificación del código para el estudio de la utilización obtenida para las distintas cantidades de flujos que se barren. Es decir, la modificación del código para

obtener la recolección y el volcado de las estadísticas necesarias para medir la utilización.

3. *Análisis y estudio del caso de uso del informe técnico de 5G-ACIA [8]*: Estudio del caso de uso propuesto en el documento mencionado para la implementación de las probabilidades de cada tipo de tráfico, así como el mapeo de las clases de tráfico del documento a las clases de tráfico propuestas en el proyecto.
4. *Implementación de los porcentajes de tráfico obtenidos a partir del 5G-ACIA [8]*: Modificación del código para obtener el número de flujos para cada clase de tráfico a partir del número de flujos total introducido como parámetro y basándonos en los porcentajes del 5G-ACIA [8].

■ **Fase de simulaciones:**

1. *Simulaciones para el caso de análisis por clase de tráfico*: Simulaciones iniciales para obtener la utilización cuando agregamos los tráficos en una misma clase y realizamos la priorización basándonos en estas clases de tráfico.
2. *Simulaciones para el caso de análisis por flujo*: En este caso se realiza una priorización por flujo y comparamos los resultados obtenidos en el caso anterior (priorización por clase de tráfico).
3. *Simulaciones con barrido en el porcentaje de tráfico para la clase Cyclic-Synchronous*: Realizamos distintas simulaciones para 3 casos diferentes: porcentajes de 30, 50 y 80 % para la clase de tráfico *Cyclic-Synchronous* y basándose en el modelo del 5G-ACIA [8].
4. *Simulaciones y análisis de una red TSN asíncrona con múltiples nodos y enlaces*: En este caso, las evaluaciones se extienden al caso de tener varios nodos conectados entre sí considerando diversas topologías (estrella, anillo y conexión en cadena (-*daisy chain*-)) y los resultados se comparan con los de una única instancia ATS para estudiar la degradación en la utilización máxima alcanzable en función del número de saltos.
5. *Simulaciones y análisis del rendimiento para un escenario donde se añadiría tráfico Best-Effort*: Por último, se considera la opción de añadir una cola de tráfico *best-effort* para utilizar la capacidad restante una vez se ha encaminado el tráfico URLLC.

- **Fase de escritura de la memoria:** Esta será la última fase del proyecto y consiste en elaborar la memoria del mismo, recogiendo por escrito toda la información recopilada previa al diseño, así como los resultados que se han obtenido de las simulaciones realizadas. También

se incluyen las conclusiones de lo que se ha aprendido, así como líneas futuras del proyecto.

A continuación, se va a mostrar un diagrama de Gantt donde se recoge de forma visual la planificación inicial de todo el proyecto.

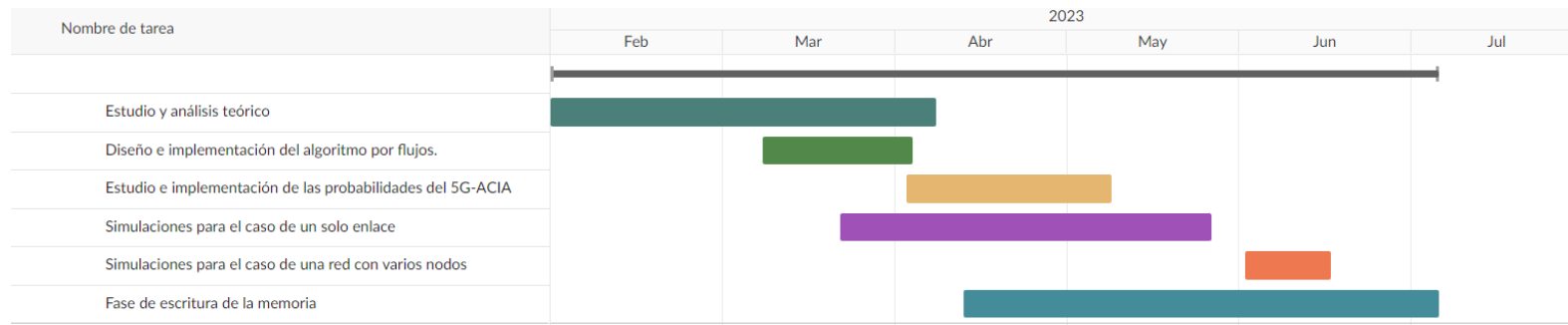


Figura 3.1: Diagrama de Gantt que recoge la planificación inicial de todo el proyecto

## 3.2. Recursos

En esta sección se van a comentar y analizar los recursos, así como las horas empleadas en el desarrollo del trabajo.

### 3.2.1. Recursos Humanos

Los recursos humanos son en sí las personas que han intervenido en el desarrollo del proyecto. En este trabajo concretamente encontramos:

- **Pablo Ameigeiras Gutiérrez**, Profesor Titular del Área de Telemática del Departamento Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada, como tutor.
- **Jonathan Prados Garzón**, Profesor Ayudante Doctor del Área de Telemática del Departamento Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada, como cotutor.
- **Juan Muñoz Villalón**, alumno del Grado de Ingeniería de Telecomunicación impartido en la Escuela Técnica Superior de Ingenierías Informática y de Telecomunicaciones de la Universidad de Granada.

### 3.2.2. Recursos Hardware

Los recursos hardware son aquellos dispositivos utilizados para los diseños y simulaciones del proyecto. Los recursos han sido:

- **Ordenador Portátil del alumno:** TOSHIBA SATELLITE P50-A-14G con sistema operativo Windows 10 Home, procesador Intel Core i7-4700WQ 2.40GHz y memoria RAM de 8 GB.
- **PC del alumno:** MSI MS-7918 con sistema operativo Windows 10 Pro, procesador i7-4790K de 4.00 GHz y memoria RAM de 32 GB.

### 3.2.3. Recursos Software

Los recursos software hacen referencia a todas las aplicaciones o programas informáticos usados durante la realización del proyecto. Estas han sido:

- Entorno de desarrollo MATLAB R2020b.
- Editor de textos LaTeX online, Overleaf.
- Editor de hojas de cálculo Excel.

### 3.2.4. Horas Empleadas

En esta subsección se va a incluir una tabla que recoge las horas empleadas en cada una de las fases mencionadas en la sección 3.1.

En la tabla 3.1 podemos encontrar un pequeño resumen de las horas empleadas en el desarrollo de cada una de las tareas anteriores para la posterior estimación de los costes.

Tareas	Horas
Estudio de TSN y el estado del arte	25
Estudio del simulador para el algoritmo del ATS scheduler	5
Modificación del código para el análisis por flujos	4
Implementación de la utilización	5
Análisis y estudio del caso de estudio del <i>paper</i> de 5G-ACIA[8]	25
Implementación de las probabilidades mencionadas en el código	8
Simulaciones análisis por flujo vs análisis por PCP	30
Simulaciones con barrido en la probabilidad de flujo para la clase <i>Cyclic-Synchronous</i>	90
Simulaciones de una red con distintos nodos y análisis del algoritmo en varios enlaces	30
Simulaciones y análisis del rendimiento para un escenario donde se añadiría tráfico <i>Best-Effort</i>	5
Redacción de la memoria	75
<b>Total</b>	<b>302</b>

Cuadro 3.1: Horas empleadas en cada una de las tareas

## 3.3. Presupuesto Estimado

En esta sección se expone el presupuesto estimado a partir de los recursos empleados y las horas de trabajo dedicadas a cada una de las partes comentadas en la sección anterior.

### 3.3.1. Costes humanos

El salario medio de un ingeniero junior en España es de unos 12.82 €/hora [20]. Teniendo en cuenta que las horas empleadas en el proyecto son unas 302 horas, podemos estimar un precio en costes humanos de:  $12.82 \cdot 302 = 3871.64$  €.

En cuanto al coste del trabajo invertido por el tutor y cotutor del proyecto, el sueldo medio de un profesor titular de Universidad en España es de, 39600 € anuales y el de profesor ayudante Doctor de, 26400 € anuales. Suponiendo que son 12 pagas y 22 días al mes laborales con jornadas de 8 horas, se tiene que los sueldos medios a la hora son 18.57 € y 12.5 €, respectivamente. Estimando unas 50 horas de trabajo por parte de ambos tutor y cotutor, podemos hacer una estimación del coste del trabajo de ambos:  $50 \cdot (18.57 + 12.5) = 1553.5$  €

Por tanto, la estimación de los costes humanos total en el proyecto es de  $3871.64 + 1553.5 = 5425.14$  €

### 3.3.2. Costes Software

Las herramientas utilizadas que requieren de una licencia son MATLAB y el Microsoft Office 365 para el uso de Excel, ya que Overleaf es gratis. El precio de la licencia de MATLAB es de aproximadamente 800 € y el precio de Microsoft Office 365 es de 69 €/año. Teniendo en cuenta estos datos, la suma de costes de recursos software es de 869€.

### 3.3.3. Costes Hardware

En cuanto a los costes hardware, como se ha indicado, se han utilizado 2 dispositivos. El precio del portátil TOSHIBA es de unos 800€ y el PC de sobremesa 1000€, por lo que la suma es de 1800€.

### 3.3.4. Costes totales

Por último, en la tabla (3.2) se recogen todos los costes anteriormente mencionados, siendo el coste estimado total del proyecto 8094.14€.

Recursos	Coste (€)
Recursos Humanos	5425.14
Recursos Software	869
Recursos Hardware	1800
<b>Total</b>	<b>8094.14</b>

Cuadro 3.2: Presupuesto desglosado del presente Proyecto





## Capítulo 4

# Diseño e Implementación

### 4.1. Introducción al simulador empleado. Caso de una sola instancia ATS

En este capítulo se va a abordar los cambios introducidos en el simulador dado para la realización del proyecto. Hay que mencionar que en mi caso parto de un simulador que se me ha dado en el cual yo introduzco una serie de cambios para adaptarlo al caso de estudio del presente proyecto.

El simulador diseñado en MATLAB se centra en buscar la forma más óptima de a minimizar el número de niveles de prioridad utilizados para un cierto número de flujos según un algoritmo de priorización de flujos de un ATS (Asynchronous Traffic Shaper [15]), el cual se explica con más detalle en la siguiente sección. Este criterio de optimalidad se elige, ya que el número de niveles de prioridad utilizados está relacionado indirectamente con el coste de producción de un conmutador TSN. El simulador permite hacer un barrido de flujos, como por ejemplo, de 100 a 1000 con saltos de 100, y además permite realizar varios escenarios para cada cantidad de flujos y varias repeticiones por escenario para así obtener mejor estabilidad estadística. Como salida, el simulador obtiene, como ya hemos comentado, la mejor configuración, si existe, y nos ofrece un porcentaje de escenarios en los que se consigue llegar a un resultado óptimo según el número de flujos. Además, el simulador nos ofrece como salida la cantidad de prioridades que se generan para satisfacer un número de flujos determinado.

El simulador contempla 7 clases de tráfico, cada una con un PCP determinado y unas características de comunicación determinadas que son:

- *Rate* del flujo.
- Tamaño de ráfaga (*burstiness*) del flujo.
- Retardo máximo del flujo.
- Tamaño del flujo.

Para dividir los  $X$  número de flujos que tenemos en una instancia de la simulación entre las diferentes 7 clases de tráfico, se asignan porcentajes a cada clase de tráfico, de manera que los *rates* agregados de las diferentes clases de tráfico (sumando los *rate* de cada flujo en esa clase) sean constantes. Comentar que a cada flujo que se asocia a una clase de tráfico, se le asignan aleatoriamente de entre unos valores tabulados (vector con límite inferior y superior), las 4 características que hemos comentado anteriormente. Sin embargo, una característica que no es deseada y que realiza este simulador es que no soporta una priorización por flujo, que se explicará posteriormente, sino por clases de tráfico. Esto es que todos los flujos pertenecientes a una misma clase de tráfico son planificados como un solo *flujo agregado* y, por ende, todos ellos compartirán la misma priorización. Esto se realizaba así para reducir el tiempo de ejecución. Sin embargo, esta característica no es deseable en el presente proyecto porque se quiere explorar los límites máximos de utilización ofrecidos por una red TSN asíncrona. Como se verá en el capítulo de resultados, realizar una priorización más granular a nivel de flujo resulta en una mayor utilización con respecto a una priorización por clase de tráfico. Es por ello que esta mejora fue uno de los principales cambios a introducir en el simulador de partida.

## 4.2. Modelo del Sistema a Considerar

El algoritmo de priorización empleado en el simulador se basa en el propuesto en el artículo [7], el cual se va a explicar a continuación para una mejor comprensión de la herramienta. No obstante, recomendamos al lector interesado en detalles más concretos de dicho algoritmo a la citada fuente. Vamos a considerar una red de TSN asíncrono que comprende una serie de *switches* TSN usando instancias ATS en sus puertos de salida para ofrecer garantías deterministas de QoS. Vamos a suponer que existen una serie de flujos sensibles al retardo que deben transportarse a través de la red, cuyo tráfico está limitado por  $r^*t + b$  [21], donde  $r$  y  $b$  son, respectivamente, el *data rate* y el tamaño de ráfaga (*burstiness*). Todos estos requisitos son conocidos de antemano, como suele ser el caso en redes industriales. Por ejemplo, podemos tener flujos críticos para comunicar eventos de alarma, controlar el movimiento de los dispositivos operativos y dirigir los robots móviles por la fábrica.

Cada *switch* TSN incluye una instancia ATS en cada puerto de salida para gestionar la transmisión de los paquetes en el enlace correspondiente de acuerdo a la operación descrita en el capítulo de *Estado del arte*. Cada instancia ATS incluye  $P$  niveles de prioridad y estamos asumiendo que hay suficientes colas de conformación para usar todos los niveles de prioridad sin importar la configuración de la red TSN asíncrona. También vamos a asumir que cada nivel de prioridad es asociado con un índice entero  $p$  y que

los niveles inferiores significan más prioridad. De esta forma, la prioridad 1 es el nivel de mayor prioridad. Los niveles de 1 a  $P - 1$  se utilizan para acomodar el tráfico según sus requisitos de *delay*, mientras que el nivel  $P$  está destinado al tráfico *best-effort*.

Dadas las rutas elegidas por flujo y la distribución de los requisitos de retardo entre los saltos, existe un conjunto de flujos sensibles al retardo  $F$  que van a ser priorizados en cada instancia ATS. En este caso se asume que existe un requisito de peor caso de *delay*, denominada  $D_f$  para cada flujo  $f \in F$  en la instancia ATS determinada. De esta forma, podemos definir el requisito WCQD (*Worst-Case Queuing Delay*) por cada flujo  $f$  como  $R_f = D_f - l_f/C$ , donde  $l_f$  es el tamaño máximo de paquete del flujo  $f$  y  $C$  es la capacidad nominal del enlace gestionado por la instancia ATS. Se va a asumir que cada flujo  $f \in F$  es asociado con un índice  $i$  de acuerdo a su requisito WCQD  $R_f$ , por lo que se quedaría como  $R_i$ . Índices más pequeños es igual a requisitos WCQD más estrictos:  $R_{i-1} \leq R_i \leq R_{i+1}$  siendo, por tanto, los flujos con índices 1 y  $F = |F|$  aquellos con requisitos WCQD más y menos estrictos, respectivamente.

Digamos que  $F_p$  es el conjunto de flujos asociados a un nivel de prioridad  $p$ . Por tanto, el WCQD  $Q_p$  experimentado por cada flujo asociado a ese nivel de prioridad  $p$  es limitado superiormente como sigue [16] [18] [2]:

$$Q_p = \frac{\sum_{f \in F_1 \cup \dots \cup F_p} b_f + \max_{f \in F_{p+1} \cup \dots \cup F_8} l_f}{C - \sum_{f \in F_1 \cup \dots \cup F_{p-1}} r_f} \quad (4.1)$$

Donde  $r_f$  y  $b_f$  son el *data rate* y el tamaño de ráfaga (*burstiness*) para el flujo  $f$ , respectivamente. En el cuadro 4.1 se puede consultar la notación considerada en esta sección.

### 4.3. Definición del Problema

El objetivo de este algoritmo es encontrar una solución factible de priorización para una instancia ATS específica y por tanto cumplir con los requisitos de retardo para todo flujo  $f \in F$  minimizando el numero de niveles de prioridad usados. A continuación se formula el problema en cuestión:

$$\text{minimize} \{ \max_{f \in F} P_f : P_f \in [1, 1 - P] \cap \mathbb{R} \forall f \in F (C1);$$

$$Q_p \leq R_f \forall f \in F_p, p \in [1, P - 1] (C2);$$

$$\sum_{f \in F} r_f \leq C (C3). \} \quad (4.2)$$

Notación	Descripción
$F$	Conjunto de todos los flujos que se van a gestionar en una instancia ATS.
$F_p$	Conjunto de todos los flujos asociados a un nivel de prioridad $p$ en la instancia ATS.
$C$	Capacidad del enlace en la instancia ATS a considerar.
$r_f, b_f, y l_f$	<i>Data rate</i> , tamaño de ráfaga ( <i>burstiness</i> ) y el tamaño máximo de paquete para el flujo $f$ , respectivamente.
$R_f$ y $D_f$	WCQD y requisitos de retardo para el flujo $f$ en la instancia ATS a considerar, siendo $R_f = D_f - l_f/C$ .
$Q_p$ y $Q_f$	WCQD para el nivel de prioridad $p$ y experimentado por el flujo $f$ .
$R_1$ y $R_F$	WCQD más estrictos e indulgentes de entre todos los flujos $F$ .
$P$	Número máximo de niveles de prioridad (colas) disponibles (implementadas) en la instancia ATS.

Cuadro 4.1: Notación usada en el algoritmo de priorización del ATS

donde  $R$  es el conjunto de los numeros naturales. La variable de decisión  $P_f$  hace referencia al nivel de prioridad asignado al flujo  $f \in F$ . Esta variable es un entero y toma valores en el conjunto de niveles disponibles en la instancia ATS en concreto, como se especifica en la condición 1 (C1).

El objetivo del problema es minimizar el número de niveles de prioridad necesarios para redirigir los flujos entrantes en el ATS. El porqué de usar este criterio de optimización se debe a que el coste de una red TSN asíncrona está directamente relacionada o depende directamente del número de niveles de prioridad en las instancias ATS. Cuanto mayor sea el número de prioridades, mayor será el coste de los *bridges* TSN basados en ATS, además de que es más difícil de gestionar la red.

La tercera condición (C3) indica que se debe asegurar que el tráfico agregado que atraviesa el ATS es menor que la capacidad nominal. De hecho, esto es algo que se deriva de la ecuación (4.1). Por otro lado, los requisitos WCQD para todos los flujos se deben cumplir, eso es la condición 2 (C2).

#### 4.4. Explicación Pseudocódigo del Algoritmo Desarrollado

El algoritmo utilizado se especifica en la figura 4.1. Como hemos comentado, el objetivo es encontrar una solución factible de priorización de una cantidad concreta de flujos  $F$  en una instancia ATS determinada, de acuerdo al criterio de optimización explicado. De esta forma, el algoritmo va iteran-

---

```

1: Problem_Solved = 1;                                ▷ BC_O1
2: No_Solution = 2;                                    ▷ BC_O2
3: Searching_Solution = 3;                            ▷ BC_O3
4: Initialize  $\mathcal{F}_1 \leftarrow \mathcal{F}$ ;  $\mathcal{F}_2 \leftarrow \emptyset$ ;  $k = 1$ ;
5: function PrioritizeFlows( $\mathcal{F}_1, \mathcal{F}_2, k$ )
6:   prob_status = Searching_Solution;
7:   while prob_status == Searching_Solution do
8:     if  $Q_2 \leq R_f \forall f \in \mathcal{F}_2$  then
9:       if  $Q_1 \leq R_f \forall f \in \mathcal{F}_1$  then
10:        return Problem_Solved;
11:       end if
12:        $k++$ ;  $\mathcal{F}_k = \{\}$ ;
13:        $\mathcal{F}_p \leftarrow \mathcal{F}_{p-1} \forall p = [2, k]$ ;  $\mathcal{F}_1 \leftarrow \emptyset$ ;
14:     end if
15:      $f^* \leftarrow \arg \min_{f \in \mathcal{F}_2} R_f$ ;
16:      $\mathcal{F}_2 \leftarrow \mathcal{F}_2 \setminus f^*$ ;  $\mathcal{F}_1 \leftarrow \mathcal{F}_1 \cup \{f^*\}$ ;
17:     if  $\mathcal{F}_2 == \emptyset$  then
18:       return No_Solution;
19:     end if
20:   end while
21: end function

```

---

Figura 4.1: Algoritmo de Priorización en el ATS [7]

do (líneas 7-20) hasta que una solución se encuentre, es decir, los requisitos de retardo se cumplen para todos los flujos, mientras que la utilización se mantiene por debajo de 100 % (línea 10), o no se alcance una solución (línea 17).

En cada iteración, primero el algoritmo comprueba si se cumplen los requisitos WCQD para todos los flujos asociados al segundo nivel de prioridad  $\mathcal{F}_2$  (línea 8). Si la condición se cumple, que se verifica usando la ecuación (4.1), entonces el algoritmo comprueba de nuevo si los requisitos WCQD para todos los flujos alojados en  $\mathcal{F}_1$  son cumplidos. En caso de que sea así, el algoritmo ha encontrado una solución factible y termina. En el caso contrario, el algoritmo crea un nuevo nivel  $\mathcal{F}_k$  y decrementa el nivel de prioridad para todos los flujos en 1, dejando el nivel  $\mathcal{F}_1$  vacío (líneas 12-13).

La razón por la que se hace esto es que, una vez que el algoritmo encuentra una priorización satisfactoria para los flujos asignados a los niveles de prioridad actuales 2 a  $k$ , el nivel de prioridad más alto puede dividirse para encontrar una solución factible sin afectar a los WCQD de los niveles de prioridad actuales 2 a  $k$ .

Si las condiciones en las líneas 8 o 9 no se cumplen, entonces el algoritmo mueve el flujo  $f^*$  con el requisito WCQD más estricto en el nivel 2 al nivel 1. Igualmente, si el flujo  $f^*$  resulta que es el último flujo en  $\mathcal{F}_2$ , entonces el algoritmo no tiene solución.

## 4.5. Propuestas de diseño y análisis en el presente proyecto

En la referencia [7] se propone realizar la priorización por PCP. Es decir, todos los flujos pertenecientes a un mismo PCP se priorizan conjuntamente. Para hacer esa priorización, todos los flujos de esa clase se tratan como si fueran un flujo conjunto en el que el *rate* se calcula como el *rate* agregado de todos los flujos de dicha clase. El requisito de retardo es el más estricto de todos los flujos de dicho conjunto de flujos. En cuanto al *burstiness*, se calcula como la suma de los *burstiness* de los distintos flujos y por último, el tamaño de paquete se calcula como el máximo de entre el tamaño de paquete máximo de todos los flujos.

Hacer la priorización por PCP es muy escalable porque solo hay 8 posibles PCPs (o sea que solo hay que priorizar 8 posibles superflujos). Pero tiene un inconveniente claro, si un flujo de dicho conjunto tiene un requisito muy estricto, todos los flujos tendrán dicho requisito tan estricto. En cambio, puede haber algunos flujos de dicho PCP que tengan un requisito de retardo más laxo.

Con respecto a la referencia [6], este TFG propone hacer la priorización flujo a flujo, cada uno con su requisito de retardo. Cabe esperar que el tiempo de computación del algoritmo sea mayor, pero que con esto consigamos acomodar un mayor número de flujos, y, por tanto, incrementar la capacidad máxima de tráfico que se puede soportar.

A diferencia de la implementación de [7], vamos a usar un modelo de tráfico similar al propuesto por el organismo 5G-ACIA para un escenario industrial [8], el cual es más realista al estar basado en un caso de uso real. La principal diferencia que existe con el modelo planteado en [7] es que dicho modelo no propone una repartición del número de flujos entre las distintas clases basada en un caso real, sino que se asignan tantos flujos como sea necesario a cada clase para hacer que todos los textitrates agregados (de cada clase de tráfico) sean del mismo tamaño.

También, en este TFG se va a evaluar experimentalmente el impacto en la capacidad de las principales características del tráfico. Dado que el tráfico tiene unas cuantas características (*rate*, *burstiness*, retardo, tamaño de paquete y duración del flujo) no podemos variarlas todas simultáneamente porque nos saldrían muchos casos. Por ello, lo que vamos a variar es el porcentaje del tráfico de un tipo u otro de tráfico en función de si tiene un requisito dominante de retardo, *rate*, etc. En este caso, se va a variar el tráfico con requisito dominante en retardo.

#### 4.5.1. Caso de uso del 5G-ACIA. Obtención de porcentajes de tráfico

Un inconveniente del modelo sintético de tráfico compuesto que incluía originalmente el simulador de partida es que no capturaba de forma fidedigna la demanda de tráfico en la Industria 4.0. Por tanto, el cambio sugerido en el proyecto es basarnos en un caso de uso más real, de modo que se capture una demanda de tráfico típica de la Industria 4.0. Tras una profunda revisión bibliográfica, se halló con un caso de uso realista especificado en un informe técnico de 5G-ACIA [8]. En él, se introduce un caso de uso de un modelo de 5G para uso industrial, con grupos de tráfico similares a los considerados en nuestro proyecto y que se discutirán en el capítulo de simulaciones y resultados.

A partir de este caso de uso, se obtienen los porcentajes para cada clase de tráfico, dividiendo el *rate* en DL (bajada) existente en la red de cada tipo de tráfico entre el *rate* total de DL (bajada). A partir de estos porcentajes de tráfico del 5G-ACIA, podemos obtener el número de flujos para cada clase de tráfico en nuestro simulador para cada simulación (número de flujos total), en definitiva, obtener los porcentajes de tráfico en el simulador (que son distintos de los del 5G-ACIA).

Variable	Descripción
$N_t$	Número de flujos del tipo de servicio $t$
$E[r_t]$	Rate (committed data rate en terminología TSN) medio por flujo. Si el <i>rate</i> de cada flujo se muestrea de una distribución uniforme definida con unos valores mínimo $r_t^{(min)}$ y máximo $r_t^{(max)}$ , entonces, $E[r_t] = \frac{r_t^{(min)} + r_t^{(max)}}{2}$
$P_{ACIA}^{(t)}$	Porcentaje del rate agregado que representa el tipo de servicio (tráfico) $t$ de acuerdo con los casos de uso definidos en los informes técnicos de 5G-ACIA.
$C$	Capacidad nominal del enlace
$U$	Utilización del enlace. Es la utilización en promedio del canal físico. Ejemplo: si en nuestro escenario tenemos un rate agregado de 700 Mbps que es servido por un enlace de 1 Gbps, la utilización será del 70 %.

Cuadro 4.2: Notación para el cálculo del número de flujos para cada tipo de tráfico

Expliquemos como se obtiene el número de flujos necesario para cada tipo de tráfico. Suponiendo que tenemos  $T$  tipos de tráfico y considerando la notación definida en el cuadro 4.2, el *rate* agregado que servirá el enlace con capacidad  $C$  vendrá dado por:

$$\sum_{t=1}^T N_t \cdot E[r_t] = U \cdot C \quad (4.3)$$

Esto es porque por definición  $U = E[r_{agg}]/C$ , donde  $E[r_{agg}]$  es el *rate* agregado promedio que generarán todos los tipos de servicios considerados. A su vez  $E[r_{agg}]$  será igual (parte izquierda de la igualdad de arriba) a la suma del *rate* medio generado por cada tipo de servicio, que no es otra cosa que el número de flujos del correspondiente servicio por su *rate* medio por flujo. Por otro lado, queremos garantizar que el *rate* agregado que genera un tipo de servicio  $t$  suponga el porcentaje dado en los casos de uso definidos por 5G-ACIA. Esto es:

$$\frac{N_t \cdot E[r_t]}{\sum_{k=1}^T N_k \cdot E[r_k]} = P_{ACIA}^{(t)} \quad (4.4)$$

Según la ecuación (4.3), el denominador del término de la izquierda en la igualdad (4.4) es igual a  $U \cdot C$ . Por lo que:

$$\frac{N_t \cdot E[r_t]}{U \cdot C} = P_{ACIA}^{(t)} \Rightarrow N_t = \frac{P_{ACIA}^{(t)} \cdot U \cdot C}{E[r_t]}$$

De este modo podemos determinar el número de flujos (es un valor promedio) necesarios a generar en una configuración dada, de modo que en promedio se cumplan con los porcentajes de tráfico de los casos de uso de 5G-ACIA.

## 4.6. Adición de Tráfico *Best-effort*

Cuando se observen las simulaciones en el siguiente capítulo, se podrá apreciar que en todas ellas no se alcanza una utilización máxima excesivamente elevada y, por tanto, se puede afirmar que la capacidad restante que queda después de acomodar los servicios con requisitos de retardo se puede utilizar para encaminar tráfico *best-effort*. En concreto, se añadiría una cola adicional de prioridad mínima para acomodar dicho tráfico. Al ser el tamaño máximo de paquetes igual a 1500 Bytes (suele venir dado por la MTU de las redes Ethernet) el único impacto que tiene el tráfico *best-effort* sobre el tráfico sensible a retardo, lo único que hay que cambiar en la configuración de las simulaciones es que todos los niveles de prioridad que acomodan flujos con requisitos de retardo perciban un tamaño máximo de trama asociado a



la operación *non-preemptive* de 1500 Bytes. De esta forma, se representaría el tamaño máximo de paquetes que habría en la cola de prioridad *best-effort*.



## Capítulo 5

# Simulaciones y Resultados

### 5.1. Preparación Experimental de los Escenarios

Como ya se ha comentado previamente, se va a estudiar y analizar la capacidad máxima de las redes TSN asíncronas para acomodar tráfico sensible al retardo en escenarios industriales. Dicho de otro modo, se estudiará la utilización máxima alcanzable en los enlaces de la red TSN asíncrona para acomodar dicho tráfico. Para ello se usarán el simulador y algoritmo de priorización de flujos TSN presentados en el capítulo anterior para diferentes escenarios. Primero, se comparará la utilización máxima alcanzable cuando se realiza una priorización por clase de tráfico o PCP frente a una priorización por flujos. Por otro lado, se estudiará el impacto del número de saltos o instancias ATS en la utilización máxima alcanzable. Por último, se analizará el impacto del porcentaje de tráfico cíclico síncrono en la utilización máxima alcanzable. Estas cuestiones son claves para determinar la viabilidad de adoptar la tecnología TSN asíncrona en la Industria 4.0. Antes de pasar a discutir los resultados de las evaluaciones llevadas a cabo, se debe primero explicar y comentar que para los diferentes escenarios a estudiar se van a tener en consideración 7 clases de tráfico diferentes cuyos parámetros vienen recogidos en la Tabla 5.1. La configuración de las mismas está basada en [22] [23] de modo que se recogen unos perfiles de tráfico y requisitos de retardo típicos para escenarios industriales.

Ahora bien, en cada simulación se hará un barrido desde un determinado número de flujos a otro con saltos de  $X$  flujos para analizar la respuesta del algoritmo de priorización en diferentes escenarios. Este número de flujos es el número de flujos total y como hemos comentado anteriormente, se deberá sacar el número de flujos de cada clase de tráfico según lo comentado en la sección 4.3.1 a partir de ciertos porcentajes de tráfico obtenidos según el documento del 5GACIA acerca de un caso de uso específico en un escenario industrial [8]. En este caso se presentan *rates* para cada clase de tráfico y a partir de estos *rates*, se obtienen los porcentajes de 5GACIA de cada

Clase de Tráfico	$r_q$ (Mbps)	$b_q$ (paquete)	$D_q$ (ms)	$l_Q$ (KBytes)	PCP
Cyclic-Synchronous	8 - 0.8	1 - 4	1 - 0.5	1 - 0.05	6
Mobile Robots	<10	1 - 4	500 - 1	0.25 - 0.04	3
Cyclic-Asynchronous	0.2 - 4E-3	1 - 4	20 - 2	1 - 0.05	5
Events: Control	>12	1 - 4	50 - 10	0.2 - 0.1	4
Augmented Reality	20 - 10	1 - 4	10	1.5 - 0.03	2
Network Control	8E-3 - 4E-3	1 - 4	1000 - 50	0.5 - 0.05	7
Config. and Diagnostics	2	1 - 4	100 - 10	1.5 - 0.5	1

Cuadro 5.1: Características de los tráfico considerados [7]

tráfico en la red. Los tipos de tráfico presentados en dicho artículo son *mapeados* a las clases de tráfico que se consideran en el presente trabajo según la similitud en sus parámetros de red. Además, está la opción de realizar diversos escenarios para cada número de flujos, donde el reparto de flujos entre las clases de tráfico y los parámetros de estos cambian entre escenarios. También, para cada escenario se realizan  $X$  repeticiones donde en cada una de ellas, se repite el algoritmo para ese determinado número de flujos.

#### 5.1.1. Preparación red con varios ATS

Se van a realizar ciertas simulaciones tanto para un ATS individual como para una red con varias instancias de ATS. En este apartado se va a explicar la estructura de dicha red, ya que va a ser la misma para las diferentes simulaciones. Al igual que para el caso de un solo ATS, se van a utilizar los tipos de tráfico de la Tabla 5.1. La red en cuestión a analizar consiste en una red en anillo (topología típica en entornos industriales) con 5 nodos, como se muestra en la figura 5.1. Para los diferentes escenarios que se van a analizar hay que tener en cuenta que el tráfico se va a generar desde los nodos 2 y 5 y van a tener como destino los nodos 3 y 4. La capacidad de los enlaces es de 1 Gbps.

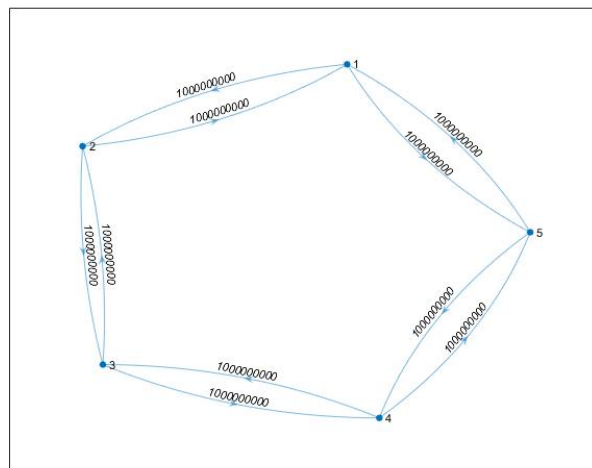


Figura 5.1: Estructura de la red a analizar

## 5.2. Análisis del rendimiento del algoritmo por flujos frente al caso por PCP

### 5.2.1. Escenario con un solo ATS

En esta sección se va a analizar el rendimiento del algoritmo comentado en el capítulo anterior para 2 casos: Diferenciación por clase de tráfico donde todos los flujos se juntan en uno, el cual representa a dicho tipo de tráfico y el caso donde diferenciamos cada flujo con sus características a la hora de ejecutar el algoritmo.

Para ambos casos se va a barrer el número de flujos de 50 en 50 para 5 escenarios diferentes y repitiendo cada escenario 5 veces, para obtener una mejor estabilidad estadística. En cada escenario se generan parámetros diferentes para los flujos que se están simulando en ese momento, mientras que las repeticiones son solo del algoritmo a analizar.

En la figura 5.2 se puede apreciar el porcentaje de escenarios que cumplen con el algoritmo de priorización para un número determinado de flujos, así como el número medio de prioridades que se están usando en el caso de analizar por PCP, mientras que en la figura 5.3 se pueden observar los mismos datos pero para el caso de diferenciación por flujos.

Todas las gráficas de priorización en este capítulo siguen el mismo formato. Dichas gráficas tienen varios indicadores:

- El porcentaje de flujos que pueden ser admitidos en la red (y que cumplen sus requisitos de retardo) frente al total de flujos generados.

Lo ideal de este indicador es que sea del 100 %. La red no podrá ser usada en aquellos rangos de carga en los que este porcentaje caiga significativamente por debajo del 100 %, ya que implicaría que muchos flujos no pueden ser admitidos.

- Porcentaje de realizaciones que se pueden acomodar con un solo nivel o con 2 niveles de priorización
- Utilización del enlace promediado a lo largo de todas las realizaciones para cada número de flujos.

Por otro lado, en la figura 5.4 y 5.5 se puede analizar el tiempo de ejecución medio para los 5 escenarios y 5 repeticiones por escenario para ambos casos y por número de flujos.

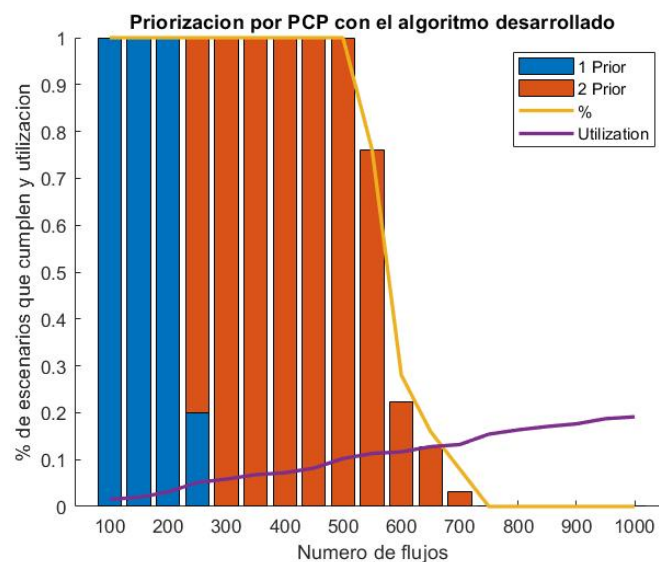


Figura 5.2: Priorización por PCP para un solo ATS

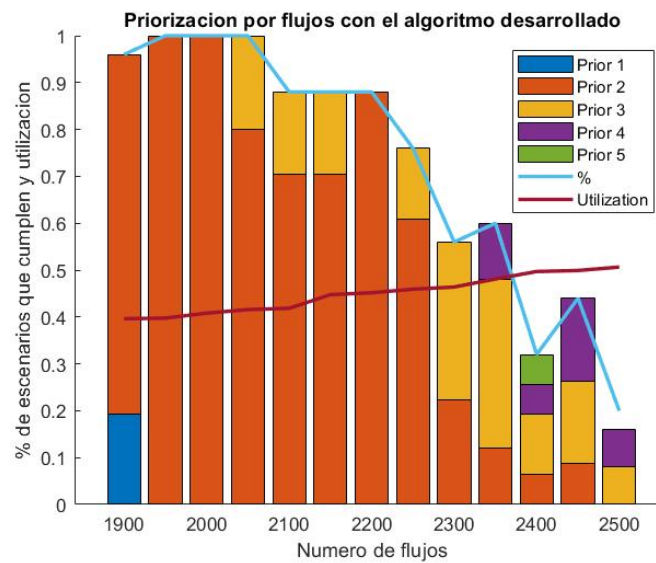


Figura 5.3: Priorización por flujo para un solo ATS

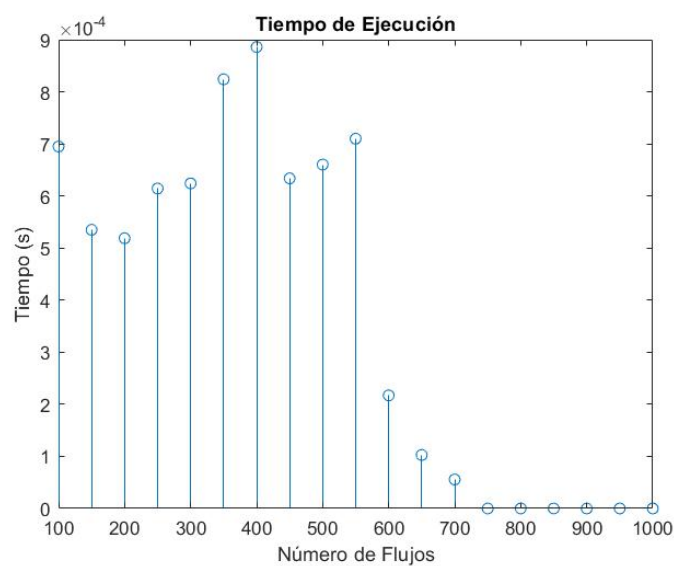


Figura 5.4: Tiempo de ejecución para el caso por PCP para un solo ATS

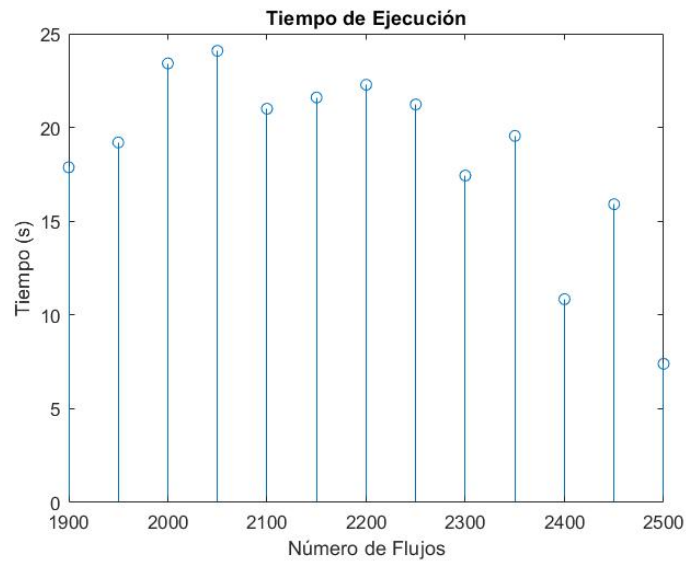


Figura 5.5: Tiempo de ejecución para el caso por flujo para un solo ATS

Como conclusión de este par de simulaciones, se puede afirmar que el uso del algoritmo de priorización por flujos en el ATS nos permite obtener un resultado factible y correcto para una mayor cantidad de flujos que en el caso de diferenciación por clase de tráfico, además de que se obtiene una mayor utilización máxima (en torno a 40 % para el caso de por flujos frente a un 10 % aproximadamente para el otro caso). Sin embargo, este aumento de la capacidad de redirección en el ATS tiene un coste asociado y este es el tiempo de ejecución. Si uno se fija en las figuras 5.4 y 5.5 podrá determinar que el tiempo de ejecución del algoritmo para el caso de priorización por flujos es aproximadamente cinco ordenes de magnitud mayor que en el caso de priorización por clase de tráfico.

### 5.2.2. Escenario en una red con varios ATS

Ahora se va a realizar la misma comparación pero en un escenario con diferentes saltos, es decir, flujos TSN atravesando múltiples instancias de ATS. En ambos casos se va a barrer el número de flujos de 100 en 100 para cada escenario y se van a realizar 10 repeticiones por escenario. De nuevo se puede ver en las figuras 5.6 y 5.7 el porcentaje de escenarios que cumplen y la mayor utilización conseguida de entre los diferentes enlaces para la diferenciación por flujos y por PCP en el algoritmo de priorización. Por otro lado, en las figuras 5.8 y 5.9 se puede apreciar el tiempo de ejecución para ambos casos.



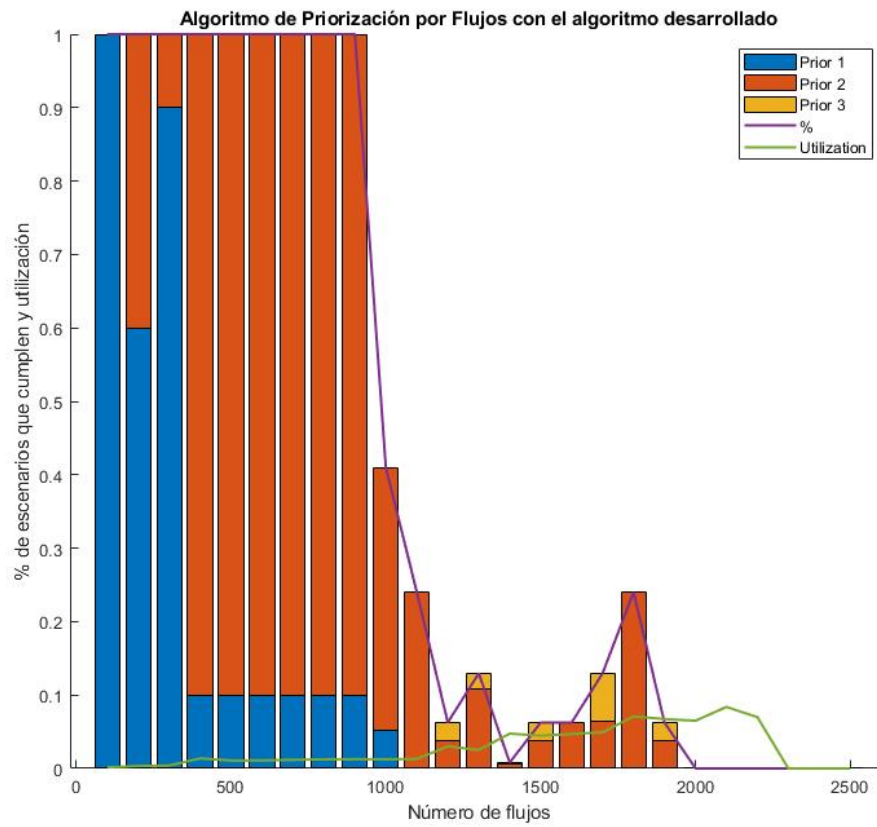


Figura 5.6: Priorización por PCP para varios ATS

## 5.2. Análisis del rendimiento del algoritmo por flujos frente al caso por PCP

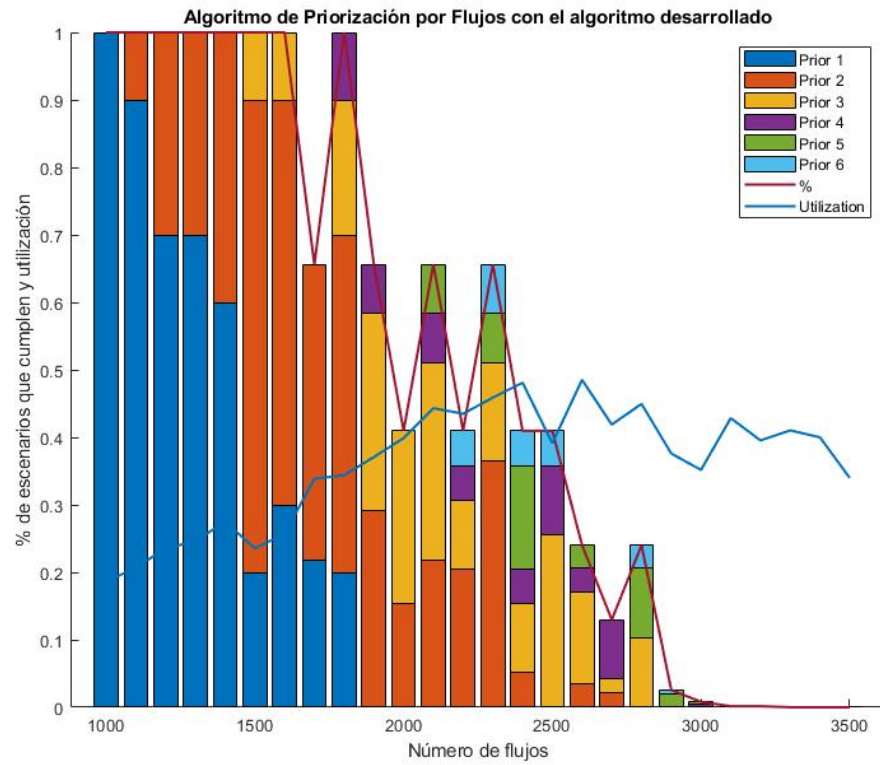


Figura 5.7: Priorización por flujo para varios ATS

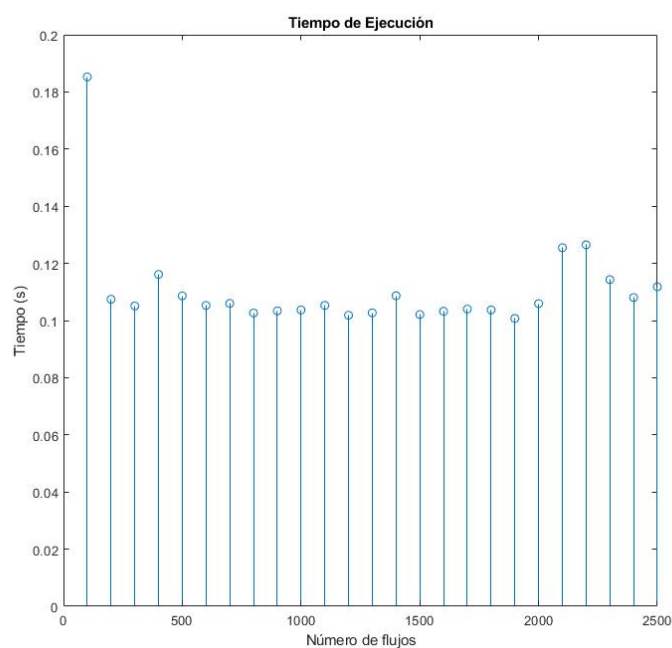


Figura 5.8: Tiempo de ejecución para el caso por PCP para varios ATS

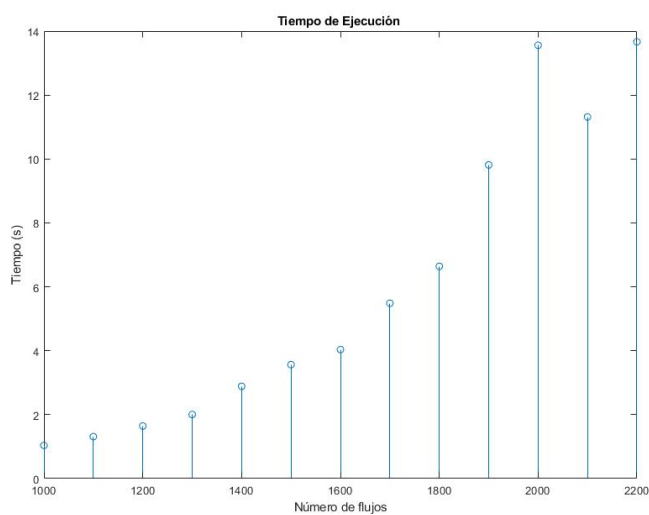


Figura 5.9: Tiempo de ejecución para el caso por flujo para varios ATS

De nuevo, las conclusiones son las mismas para el caso de varios ATS. Cuando se usa la diferenciación por flujo en lugar de por PCP, se obtienen soluciones para un mayor número de flujos, así como una mayor utilización

### 5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico

---

(aproximadamente 34 %), aunque el tiempo de ejecución es mayor que en el caso de por PCP.

### 5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico

A continuación se van a llevar a cabo simulaciones para el algoritmo de priorización del ATS en el caso de por flujos, variando el porcentaje de tráfico de distintas clases de tráfico.

#### 5.3.1. Variación en el tráfico *Cyclic-Synchronous*

Se va a variar el porcentaje de tráfico del tipo *Cyclic-Synchronous* para analizar como afecta la cantidad de tráfico con un PCP estricto y por ende requisitos de retardo críticos al rendimiento del algoritmo de priorización y estudiar la utilización que se alcanza para cada porcentaje así como el número de flujos que se pueden reenviar sin problema y, por tanto, el porcentaje de escenarios que cumplen. Comentar que se van a estudiar 3 casos diferentes del escenario original (porcentajes de tráfico basados en el 5GACIA [8]): 30 %, 50 % y 80 % de tráfico *Cyclic-Synchronous*. En realidad, lo que se está realizando es cambiar el *rate* total de dicho tipo de tráfico a analizar para que se obtenga el porcentaje deseado y mantener las tasas del resto sacadas del caso del 5GACIA para así obtener el resto de porcentajes según el aumento o la disminución para esta nueva tasa de tráfico modificada.

#### A. Escenario con un solo ATS

Se empezará con las simulaciones para un solo ATS. En la figura 5.10 se muestra el caso original, el cual es el mismo escenario que la figura 5.3. Se están realizando de nuevo 5 escenarios y 5 repeticiones por escenario y saltos de 100 flujos entre cada escenario.

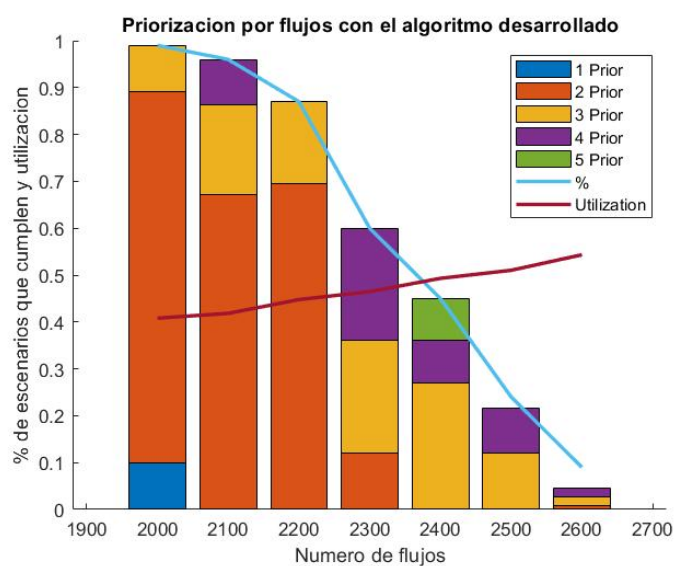


Figura 5.10: Algoritmo de priorización para un solo ATS. Escenario original (62.35 %)

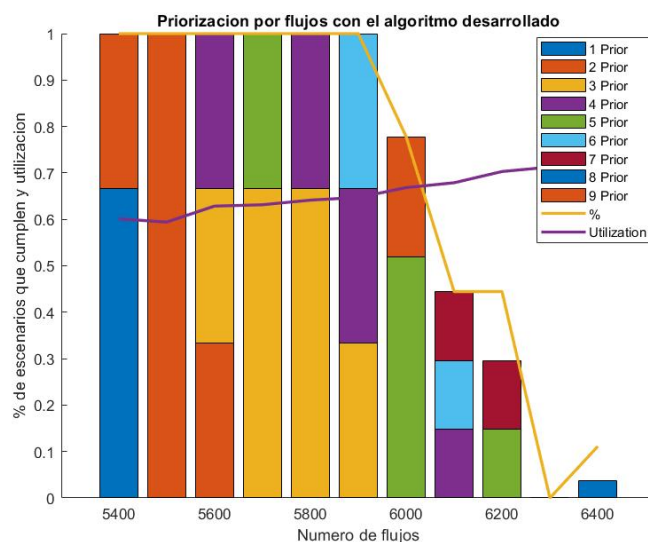


Figura 5.11: Algoritmo de priorización para un solo ATS. Escenario 30 %

### 5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico

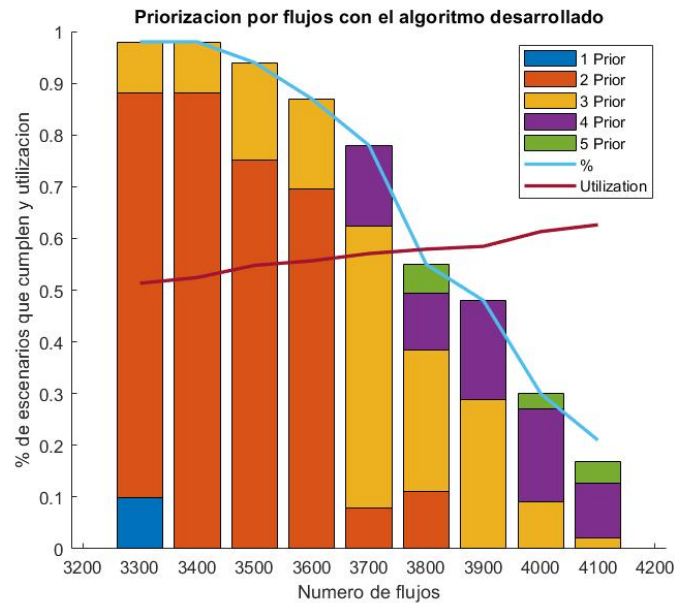


Figura 5.12: Algoritmo de priorización para un solo ATS. Escenario 50 %

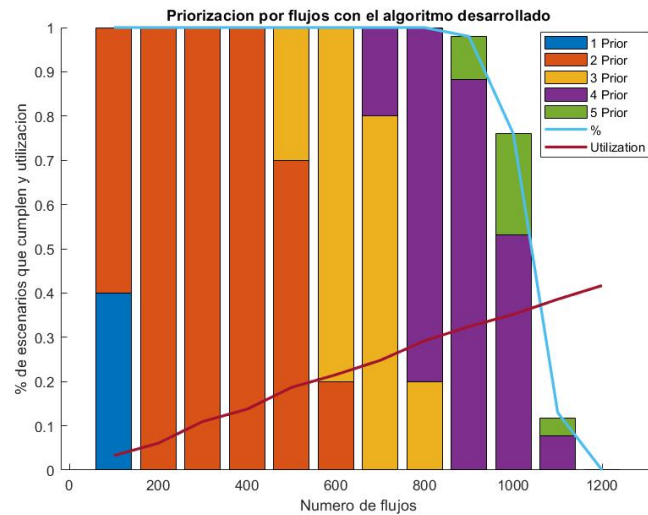


Figura 5.13: Algoritmo de priorización para un solo ATS. Escenario 80 %

Si se consulta el apéndice A, se puede observar resultados con datos para las 4 simulaciones realizadas para un solo ATS (variando la probabilidad). Lo que se muestra son los datos obtenidos para una determinada simulación a X flujos, exactamente el número de flujos donde la utilización es máxima antes de que empiece a disminuir. Se pueden observar las siguientes variables:

número de flujos, los porcentajes del 5G-ACIA [8] (estos son los porcentajes de tráfico obtenidos a partir del documento y, por tanto, los porcentajes que varían al modificar el porcentaje del tráfico *Cyclic-Synchronous*), los porcentajes “nuestros” que son los obtenidos al dividir el número de flujos de cada clase entre el total, el rate existente en dicha simulación para cada clase de tráfico y la división entre dicho rate de cada clase de tráfico y el rate agregado de todas las clases (total). Este último dato es bastante interesante, ya que teóricamente demuestra que la operación matemática realizada para obtener el número de flujos de cada clase a partir de los porcentajes de tráfico del 5G-ACIA es correcta, ya que se puede apreciar que dichos valores coinciden aproximadamente con los porcentajes del 5G-ACIA.

### B. Escenario en la red con varios ATS

Para el caso de varios ATS se llevan a cabo las mismas simulaciones también con saltos de 100 flujos entre cada escenario y realizando 10 repeticiones por cada uno de ellos.

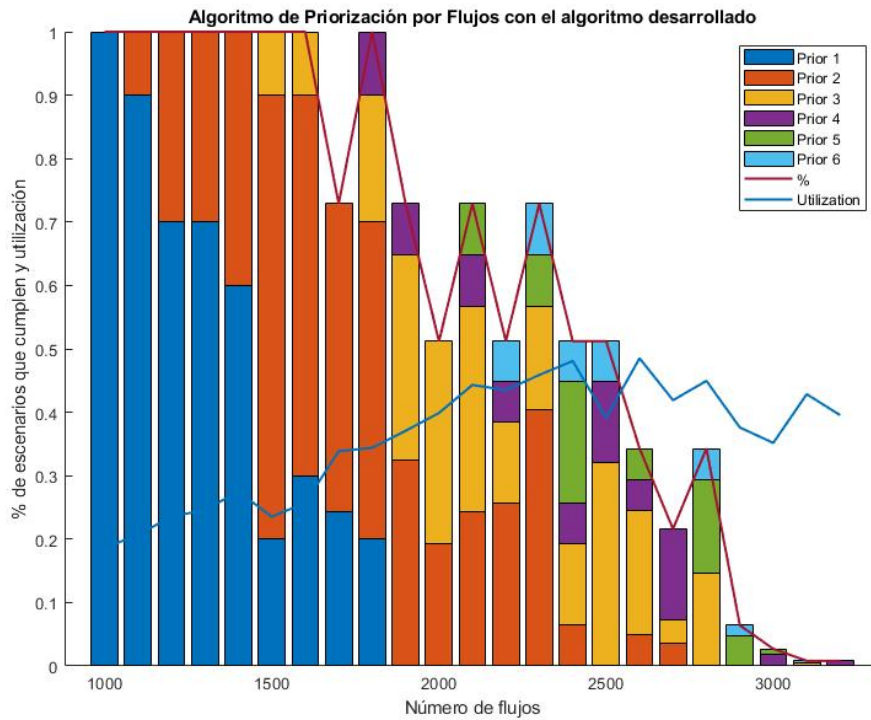


Figura 5.14: Algoritmo de priorización para varios ATS. Escenario original (62.35 %)

### 5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico

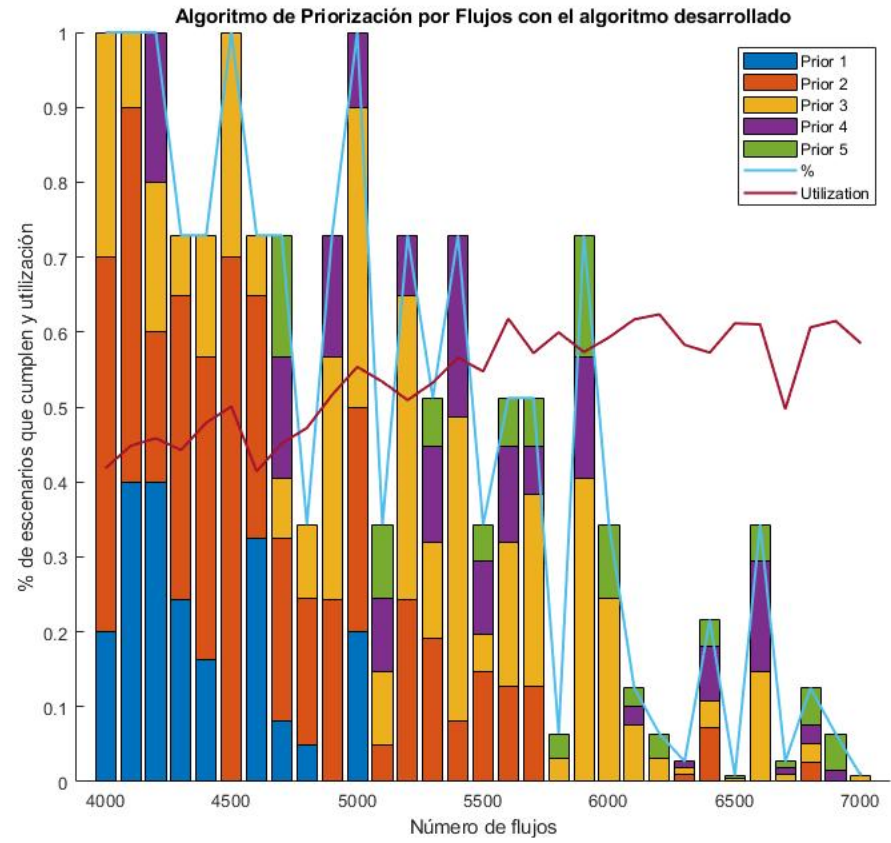


Figura 5.15: Algoritmo de priorización para varios ATS. Escenario 30 %



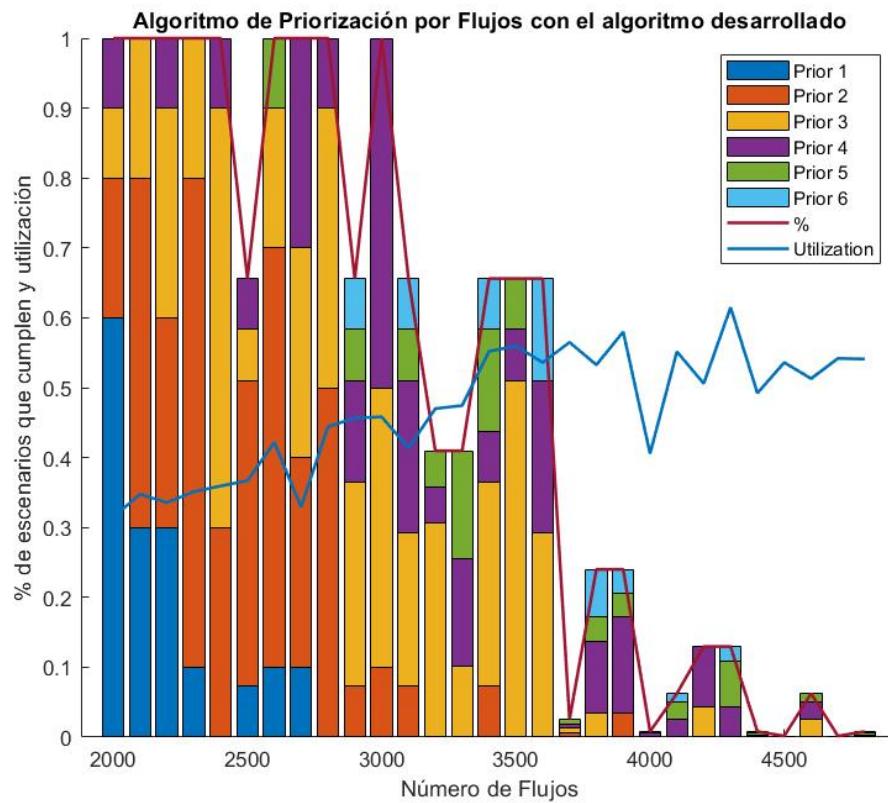


Figura 5.16: Algoritmo de priorización para varios ATS. Escenario 50 %

### 5.3. Análisis de la utilización máxima para diferentes porcentajes de clase de tráfico

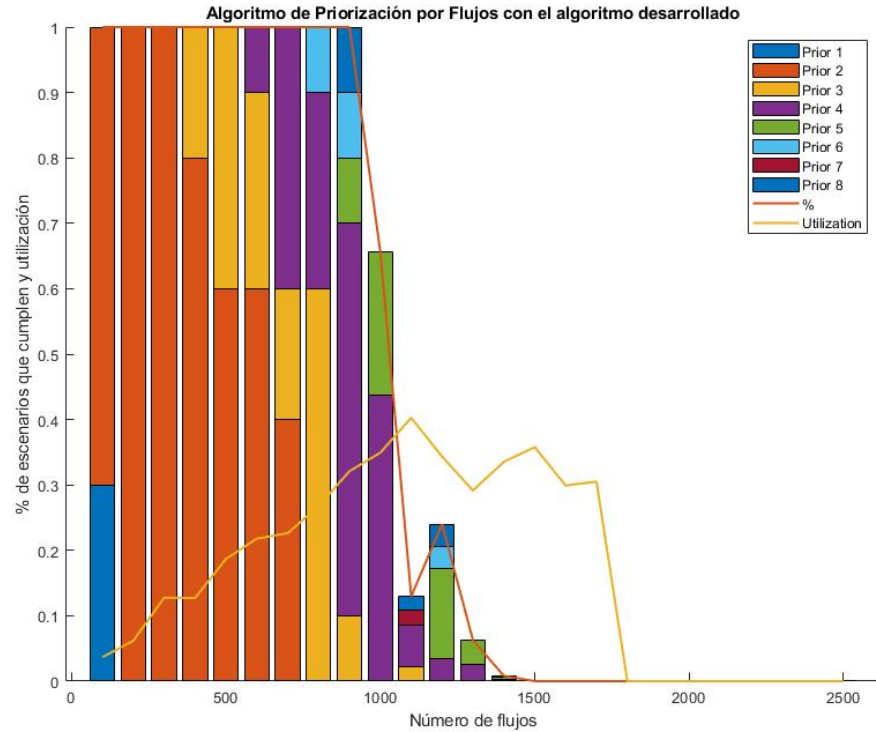


Figura 5.17: Algoritmo de priorización para varios ATS. Escenario 80 %

Como conclusión de las simulaciones mostradas arriba, se puede afirmar y apreciar que al aumentar la cantidad de tráfico de tipo crítico respecto a sus requisitos de retardo, la utilización máxima que se obtiene es cada vez menor, ya que el número de flujos para el cual el algoritmo encuentra solución es menor. En la figura 5.18 se puede observar las utilidades máximas obtenidas en cada escenario. Esto se debe a que al aumentar la cantidad de tráfico sensible al retardo, como es el tráfico *Cyclic-Synchronous* (es el que tiene unos requisitos de retardo más exigentes), al algoritmo le resulta más difícil cumplir con las calidades de servicio y los requisitos de retardo de todos los flujos y, por tanto, para cantidades de flujos cada vez más elevadas, el algoritmo no encuentra solución.

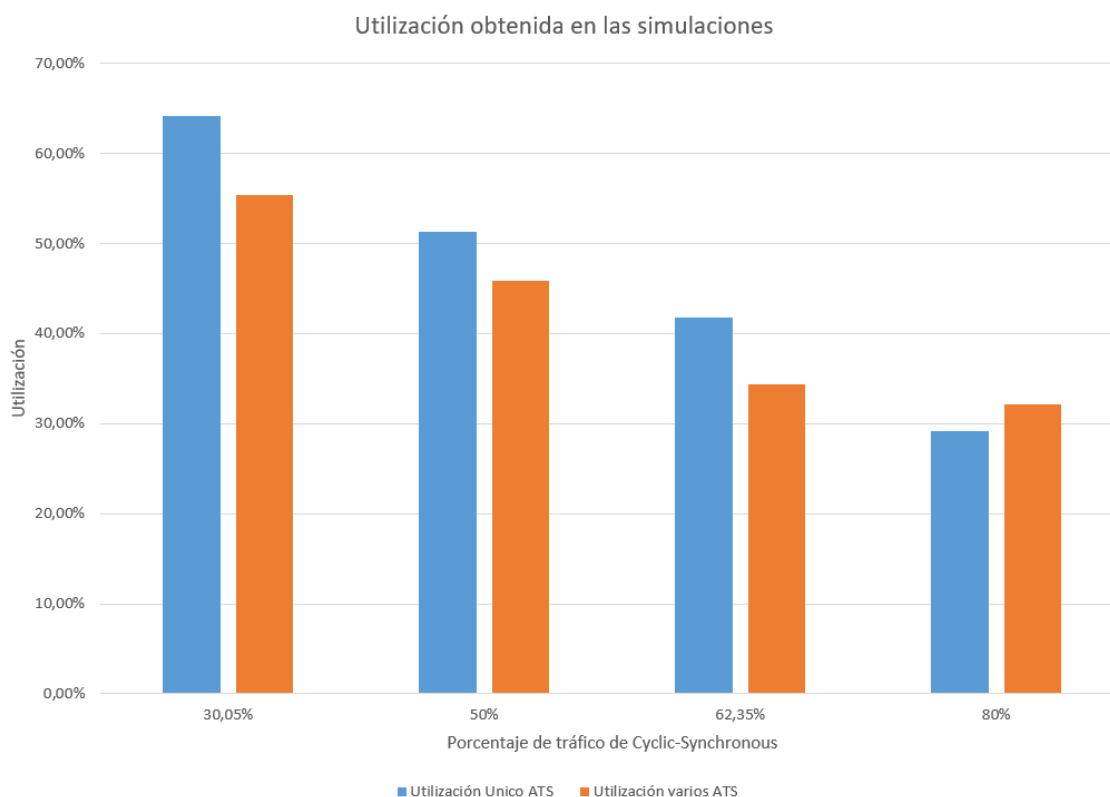


Figura 5.18: Utilización obtenida según el porcentaje de tráfico

#### 5.4. Análisis de la utilización para el caso con tráfico *best-effort*

En la figura 5.19 se puede observar los resultados para el algoritmo de priorización con la configuración comentada para *best-effort*. Se trata de una simulación desde 1800 a 2800 flujos con saltos de 50 flujos entre cada escenario. Si se compara con la figura 5.3 o 5.10, se podrá apreciar que apenas hay diferencia entre dichas simulaciones donde la utilización máxima es más o menos la misma (en torno a 41 %) y el algoritmo deja de encontrar solución en torno a 2500 flujos. El escenario es el de un solo ATS. Por tanto, se puede afirmar que utilizar la capacidad restante para tráfico *best-effort* no afecta significativamente a la calidad de servicio percibida por los flujos sensibles al retardo.

#### 7.4. Análisis de la utilización para el caso con tráfico *best-effort*

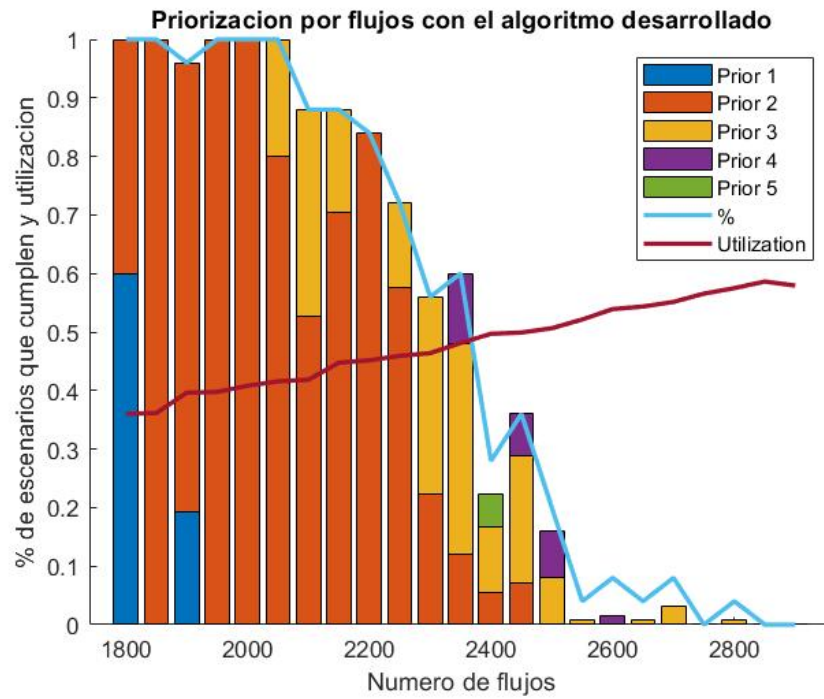


Figura 5.19: Utilización obtenida en el caso de considerar tráfico *Best-Effort*

## Capítulo 6

# Conclusiones y Trabajos Futuros

En resumen, en el presente trabajo se ha revisado teóricamente en que consiste la tecnología TSN (Time Sensitive Networking), pero poniendo el foco en la variante TSN asíncrona. Se ha estudiado el funcionamiento del planificador ATS (Asynchronous Traffic Shaper), el cual es el elemento básico de las redes TSN asíncronas, y se ha propuesto y comentado un algoritmo para la priorización de flujos en instancias ATS. También se ha presentado un simulador que emula el funcionamiento de una red TSN asíncrona para poder llevar a cabo las pertinentes evaluaciones de interés. Se han realizado diversas simulaciones donde se ha puesto a prueba la capacidad y rendimiento de encaminamiento y planificación de dicho módulo ATS de forma individual frente a diversos tipos de tráfico con requisitos de retardo diversos. Por otro lado, se han realizado las mismas pruebas para numerosas instancias de ATS en una red en anillo donde tenemos los mismos tipos de tráfico que circulan por la red en diversos flujos. Las conclusiones que se han obtenido se pueden recoger en los siguientes puntos:

- Como primera conclusión, se pueden recoger los valores de utilización máxima obtenidos en las diversas simulaciones. En la siguiente tabla (6.1) se recogen los valores para las simulaciones que comparan la priorización por PCP frente a por flujos, así como la simulación para el tráfico *Best-Effort*. Los valores para las simulaciones para diferentes porcentajes de tráfico *Cyclic-Synchronous* se pueden apreciar en la figura 5.18.

Simulación	Utilización (%)
Priorización por PCP para un solo ATS	10.23
Priorización por flujo para un solo ATS	41.57
Priorización por PCP para varios ATS	1.27
Priorización por flujo para varios ATS	34.4
Priorización por flujo para un solo ATS (Caso Best-Effort)	41.57

Cuadro 6.1: Valores de utilización para diferentes simulaciones

- La utilización máxima y por ende el número de flujos máximo que el ATS es capaz de encaminar correctamente es mayor si dividimos y tratamos cada flujo por separado en el algoritmo desarrollado en lugar de agrupar los distintos flujos de cada clase de tráfico en un macro-flujo cuyas características de retardo son la media de los flujos individuales que conforman dicho macro-flujo. Aunque hay que afirmar que el tiempo de ejecución del algoritmo es mucho mayor para el caso de diferenciación por flujos individuales.
- Al aumentar el porcentaje de un tipo de tráfico con unos requisitos de retardo elevados, como puede ser el tráfico *Cyclic-Synchronous*, la utilización máxima que se obtiene es menor, ya que el algoritmo no es capaz de obtener solución para encaminar un elevado número de flujos. Ver la figura 5.18 donde se recogen las utilizaciones obtenidas para dichas simulaciones.
- Por último, se ha estudiado el caso de añadir tráfico *best-effort*, ya que cuando el algoritmo encuentra solución para un determinado número de flujos, la utilización que se está empleando es para tráfico que necesita ser planificado para el funcionamiento correcto de TSN, ya que requieren de unos requisitos de retardo determinados, pero, sin embargo, el resto de la capacidad del enlace puede ser utilizada para enviar tráfico *best-effort*, el cual no requiere de requisitos de retardo. Tan solo, sería añadir una cola destinada a este tipo de tráfico.

Como trabajo futuro queda estudiar el comportamiento de la red frente a la variación del porcentaje de tráfico de otra clase, como puede ser la clase cuyo rate por flujo sea el más elevado para ver así como afecta dicho tráfico a la utilización. Futuras investigaciones también pueden ir en la línea de estudiar el rendimiento del módulo TAS del TSN síncrono, así como la integración de TSN y 5G para la revolución de la industria 4.0.

# Bibliografía

- [1] György Miklós Joachim Sachs János Farkas, Balázs Varga. 5g-tsn integration meets networking requirements for industrial automation. *Ericsson Technology*, Agosto 27, 2019.
- [2] IEEE. IEEE 802.1Qcc-2018 - IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks – Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements. Technical Report 802.1Qcc, IEEE, 2018.
- [3] Integration of 5g with time sensitive networking for industrial communications. White Paper, Feb. 2021.
- [4] Jonathan Prados-Garzon and Tarik Taleb. Asynchronous time-sensitive networking for 5g backhauling. *IEEE Network*, 35(2):144–151, 2021.
- [5] IEC/IEEE. Time-Sensitive Networking Profile for Industrial Automation. <https://www.ieee802.org/1/files/private/60802-d1-2.pdf>, June 2020.
- [6] Industrial Internet Consortium (IIC). Time Sensitive Networks for Flexible Manufacturing Testbed - Characterization and Mapping of Converged Traffic Types. [https://www.iiconsortium.org/pdf/IIC\\_TSN\\_Testbed\\_Char\\_Mapping\\_of\\_Converged\\_Traffic\\_Types\\_Whitepaper\\_20180328.pdf](https://www.iiconsortium.org/pdf/IIC_TSN_Testbed_Char_Mapping_of_Converged_Traffic_Types_Whitepaper_20180328.pdf), March 2019.
- [7] Julia Caleyá-Sánchez, Jonathan Prados-Garzon, Lorena Chinchilla-Romero, Pablo Muñoz, and Pablo Ameigeiras. Flow prioritization for tsn asynchronous traffic shapers. Presentadoo en el 3rd International Workshop on Time-Sensitive and Deterministic Networking (TEN-SOR), June 2023.
- [8] A 5g traffic model for industrial use cases. White Paper, Nov. 2019.
- [9] Ieee standard for ethernet. *IEEE Std 802.3-2022 (Revision of IEEE Std 802.3-2018)*, pages 1–7025, 2022.

- [10] Ieee standard for local and metropolitan area networks–frame replication and elimination for reliability. *IEEE Std 802.1CB-2017*, pages 1–102, 2017.
- [11] Open networking foundation. openflow switch specification version 1.5.1. <https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf>.
- [12] Ieee standard for local and metropolitan area networks – timing and synchronization for time-sensitive applications. *IEEE Standard 802.1AS-2020*, June 2020.
- [13] Ieee standard for a precision clock synchronization protocol for networked measurement and control systems. *IEEE Std 1588-2019 (Revision of IEEE Std 1588-2008)*, pages 1–499, 2020.
- [14] Ieee standard for local and metropolitan area networks – bridges and bridged networks - amendment 25: Enhancements for scheduled traffic. *IEEE Std 802.1Qbv-2015 (Amendment to IEEE Std 802.1Q-2014 as amended by IEEE Std 802.1Qca-2015, IEEE Std 802.1Qcd-2015, and IEEE Std 802.1Q-2014/Cor 1-2015)*, pages 1–57, 2016.
- [15] IEEE Draft Standard for Local and metropolitan area networks–Bridges and Bridged Networks Amendment: Asynchronous Traffic Shaping. *IEEE P802.1Qcr/D2.1, Feb. 2020*, pages 1–152, 2020.
- [16] Johannes Specht and Soheil Samii. Urgency-based scheduler for time-sensitive switched ethernet networks. In *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*, pages 75–85, 2016.
- [17] Ahmed Nasrallah, Akhilesh S. Thyagaturu, Ziyad Alharbi, Cuixiang Wang, Xing Shao, Martin Reisslein, and Hesham ElBakoury. Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research. *IEEE Communications Surveys Tutorials*, 21(1):88–145, 2019.
- [18] Jean-Yves Le Boudec. A theory of traffic regulators for deterministic networks with application to interleaved regulators. *IEEE/ACM Transactions on Networking*, 26(6):2721–2733, 2018.
- [19] 3gpp technical specification 22.104, service requirements for cyber-physical control applications in vertical domains, v17.3.0. [https://www.3gpp.org/ftp/Specs/archive/22\\_](https://www.3gpp.org/ftp/Specs/archive/22_), published July 2020.
- [20] Salario de ingeniero junior en españa. <https://es.talent.com/salary?job=ingeniero+junior#:~:text=%C2%BFCu%C3%A1nto%20gana%20un%20Ingeniero%20junior%20en%20Espa%C3%B1a%3F&>



- [text=El%20salario%20ingeniero%20junior%20promedio,hasta%20%E2%82%AC%2031.500%20a1%20a%C3%B1o.](#), Fecha de acceso: 2023-06-16.
- [21] Jean-Yves Le Boudec and Patrick Thiran. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer-Verlag, Berlin, Heidelberg, 2001.
  - [22] 3GPP. TS22.104 V17.4.0. (2020) Service Requirements for Cyber-Physical Control Applications in Vertical Domains. Technical report, 3rd Generation Partnership Project (3GPP), 2020.
  - [23] Industrial Internet Consortium (IIC). Time Sensitive Networks for Flexible Manufacturing Testbed Characterization and Mapping of Converged Traffic Types. Standard Rev. 1.0, March 2019.
  - [24] Jonathan Prados-Garzon, Lorena Chinchilla-Romero, Pablo Ameigeiras, Pablo Muñoz, and Juan M. Lopez-Soler. Asynchronous time-sensitive networking for industrial networks. In *2021 Joint European Conference on Networks and Communications 6G Summit (EuCN-C/6G Summit)*, pages 130–135, 2021.
  - [25] Youhwan Seol, Doyeon Hyeon, Junhong Min, Moonbeom Kim, and Jeongyeup Paek. Timely survey of time-sensitive networking: Past and future directions. *IEEE Access*, 9:142506–142527, 2021.

## Apéndice A. Resultados obtenidos para las simulaciones del apartado 5.3 para un solo ATS

En este apéndice se incluyen datos tomados para las simulaciones de la sección 5.3. Cada tabla contiene una serie de datos sobre las clases de tráfico a analizar para un número de flujos determinado en un escenario de porcentaje determinado. Exactamente, se han recabado los datos para el número de flujos en los que la utilización es máxima.

Simulación a 2100 flujos							
Tipo de Tráfico	Config and Diagnostic	Augmented Reality	Mobile Robots	Events-Control	Cyclic-Asynchronous	Cyclic-Synchronous	Network Control
PCP	1	2	3	4	5	6	7
Número de Flujos	0	2	1	4	330	59	1704
Porcentajes 5G-ACIA	2.68E-06	0.77	0.0301	0.1645	0.0805	0.6235	0.0245
Porcentajes Nuestras	0	9.52E-04	4.76E-04	0.0019	0.1571	0.0281	0.8114
Rate (Mbps)	0	31.5	29.6	64.4	33	264	10.2
Rate (clase tráfico) /Rate Agregado (Total)	0	0.0728	0.0684	0.1488	0.0763	0.61	0.0237

Cuadro 2: Resultados obtenidos para la probabilidad original (62.35 %)

Simulación a 5800 flujos							
Tipo de Tráfico	Config and Diagnostic	Augmented Reality	Mobile Robots	Events-Control	Cyclic-Asynchronous	Cyclic-Synchronous	Network Control
PCP	1	2	3	4	5	6	7
Número de Flujos	0	6	2	13	928	43	4808
Porcentajes 5G-ACIA	4.98E-06	0.143	0.0559	0.3056	0.1495	0.3005	0.0455
Porcentajes Nuestras	0	0.001	3.45E-04	0.0022	0.16	0.0074	0.829
Rate (Mbps)	0	92.1	43	219	96.8	190	28.8
Rate (clase tráfico) /Rate Agregado (Total)	0	0.1376	0.0642	0.3272	0.1445	0.2836	0.0429

Cuadro 3: Resultados obtenidos para la Probabilidad de 30 %

Simulación a 3300 flujos							
Tipo de Tráfico	Config and Diagnostic	Augmented Reality	Mobile Robots	Events-Control	Cyclic-Asynchronous	Cyclic-Synchronous	Network Control
PCP	1	2	3	4	5	6	7
Número de Flujos	0	4	1	7	523	57	2708
Porcentajes 5G-ACIA	3.56E-06	0.1022	0.04	0.2184	0.1068	0.5	0.0325
Porcentajes Nuestras	0	0.0012	0.0003030	0.0021	0.1585	0.0173	0.8206
Rate (Mbps)	0	55.3	22.4	116	55	256	16.2
Rate (clase tráfico) /Rate Agregado (Total)	0	0.1061	0.0429	0.2233	0.1054	0.4913	0.031

Cuadro 4: Resultados obtenidos para la Probabilidad 50 %

Simulación a 800 flujos							
Tipo de Tráfico	Config and Diagnostic	Augmented Reality	Mobile Robots	Events-Control	Cyclic-Asynchronous	Cyclic-Synchronous	Network Control
PCP	1	2	3	4	5	6	7
Número de Flujos	0	1	0	2	121	52	624
Porcentajes 5G-ACIA	1.42E-06	0.0409	0.016	0.0874	0.0427	0.8	0.013
Porcentajes Nuestras	0	0.0013	0	0.0025	0.1513	0.065	0.78
Rate (Mbps)	0	11.1	0	30.2	12.4	227	3.80
Rate (clase tráfico) /Rate Agregado (Total)	0	0.039	0	0.1063	0.0437	0.7976	0.0133

Cuadro 5: Resultados obtenidos para la Probabilidad 80 %



