



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Máster Universitario
en Tecnologías, Sistemas y
Redes de Comunicaciones

Caracterización y Simulación de Tráfico de Sensores y Comunicaciones en una Smart City

Autor: Antonio Neptalí Calero Guerrero

Director 1: Miguel Ángel Rodríguez Hernández

Cotutor 1: Ángel Aureliano Gómez Sacristán

Fecha de comienzo: 10/04/2018

Lugar de trabajo: E.T.S.I. Telecomunicaciones

Objetivos

El principal objetivo del presente proyecto es definir y caracterizar varios de los principales tipos de servicios asociados a una ciudad inteligente sostenible (Smart Sustainable City - SSC), para ello es necesario definir las fuentes de tráfico asociados a estos servicios de tal manera que se pueda proponer un mecanismo apropiado para brindar una calidad de servicio (QoS) aceptable en las redes de comunicaciones de una SSC sin que esto represente la presencia de silos de información que condicionen el futuro desarrollo de una Smart City y sus servicios asociados.

Trabajaremos con los datos e información correspondiente a los antecedentes y a la vida real de la ciudad de Valencia como un escenario referencial de una Smart City y que permitirá tener una primera estimación del impacto de la implantación de estos nuevos servicios en la ciudad. Para ello, se diseñará la red correspondiente apoyándonos en el software OMNeT++ como una herramienta de simulación que permite realizar un análisis de las características principales de una red como el ancho de banda, retardo extremo a extremo, pérdidas de paquetes, jitter, de esta manera podemos estudiar y determinar los efectos que pueden generar estas fuentes de tráfico sobre la red de comunicaciones existente y así contar con una visión holística de la ciudad de Valencia que permita el normal despliegue de los servicios según sus jerarquías y prioridades.

Metodología

El procedimiento a utilizarse en el presente trabajo será en primera instancia investigar y determinar los sensores adecuados para el despliegue de los diferentes componentes de los servicios relacionados que debe brindar una SSC. Tras la clasificación y caracterización de tráfico de las diferentes fuentes de tráfico heterogéneas correspondiente a los diferentes sensores, se simulará las comunicaciones vinculadas a ellos mediante el uso del simulador OMNET++ el cual nos permitirá realizar un análisis de las principales características del desempeño de la red. Para las simulaciones se utilizarán datos reales de la ciudad de Valencia que permitirán tener una primera estimación del impacto de la implantación de estos nuevos servicios en la ciudad.

Desarrollos teóricos realizados

Realizaremos una revisión de las diferentes opciones de sensores disponibles en el mercado para determinar los dispositivos más apropiados para cumplir las funciones de medición y recolección de datos asociados a los servicios de una SSC, posteriormente se procederá con la caracterización de las diferentes fuentes de tráfico, la clasificación del tipo de tráfico en base a las categorías del servicio MacroLAN y finalmente su configuración en el OMNeT++.

Desarrollo de prototipos y trabajo de laboratorio

Luego de determinar las características y funcionalidades de cada uno de los sensores necesarios para el despliegue de los servicios que se requieren en el proyecto *Desarrollo de la Iniciativa Impulso VLCi*, mediante el uso de la herramienta de simulación OMNeT++, partimos diseñando un modelo de red de acceso Metro – Ethernet, misma que mediante las redes MPLS nos permite realizar las configuraciones de diferentes tipos de servicios (diferentes fuentes de tráfico) para mediante la ejecución de distintas pruebas y errores, simular el comportamiento de la red primero servicio por servicio a fin de encontrar y proponer los valores apropiados

para los mecanismos de QoS y finalmente simulamos todos los servicios en conjunto para evaluar el rendimiento final de la red.

Resultados

Se diseñó y configuró la red de acceso con los elementos necesarios para efectivizar el despliegue de los servicios requeridos y detallados en el proyecto VLCi, se modificó partes de las fuentes en base a las diferentes simulaciones realizadas y se definió los parámetros más apropiados para los mecanismos de QoS de cada servicio. También determinamos como resultados que el throughput es consecuentemente a la suma de todos los servicios, en cambio el retardo y el jitter son mayores que la simulación de cualquier servicio por separado.

Líneas futuras

Las próximas acciones a realizar a partir del presente trabajo de investigación el cual fue creado para establecer un escenario de simulación y análisis de tráfico de los diferentes servicios asociados a una SSC con datos reales de la ciudad de Valencia, estarán las implementaciones y configuraciones físicas por parte del (los) adjudicatario(s) de todos los servicios requeridos conforme se detalla en la Iniciativa Impulso VLCi ^[4].

Abstract

The aim of this project is to analyze, define and characterize the traffic generated by the different sensors associated with the services of a Smart City such as video surveillance cameras, urban waste management, luminaires and public lighting, parking spaces, the efficient management of urban noise and urban environmental monitoring. The first stage of this project includes the learning of the operation of the OMNeT ++ tool and the SSC, the structure of the network and how it works by reviewing documents related to the topic developed by authors belonging to the UPV, as well as external. After, we determine the most appropriate types of sensors for each service according to the requirements of the City Council. Finally, we proceeded with the classification and characterization of the sensors associated with these services, recreating communications related to them through the OMNeT ++ simulator to determine an appropriate quality of service (QoS) mechanism for the communications network. For the simulations, real data from the city of Valencia will be used, which will allow a first estimation of the impact of the implementation of these new services in the city. We test different combinations of committed information rate (CIR) and peak information rate (PIR) for each data source separately in order to analyze the performance of the network through packet loss, throughput, delay and jitter and Finally, we simulate all services together to evaluate their final performance, determining as a result that throughput is the sum of all services but, the delay and jitter are greater than those corresponding to the execution of each service separately.

Autor: Calero Antonio, email: ancague@teleco.upv.es

Director 1: Rodríguez Miguel, email: rmarodrig@upvnet.upv.es

Fecha de entrega: 11-09-18

ÍNDICE

I.	INTRODUCCIÓN.....	2
II.	ANTECEDENTES	3
II.1.	TRABAJOS RELACIONADOS.....	3
II.2.	CIUDAD INTELIGENTE SOSTENIBLE	4
II.3.	SIMULADOR OMNET++	5
II.3.1.	MÓDULOS Y COMPONENTES DE UNA SIMULACIÓN.....	5
II.3.2.	ENTORNO DE EJECUCIÓN.....	7
II.3.3.	ALCANCE Y ANÁLISIS DE RESULTADOS	8
II.4.	DESCRIPCIÓN DEL ESCENARIO.....	9
II.4.1.	RED INTERNA DEL AYUNTAMIENTO	10
II.4.2.	RED MPLS	10
II.4.3.	PLATAFORMA DE SERVIDORES DE LA CIUDAD (CLOUD)	11
II.4.4.	MECANISMOS DE CALIDAD DE SERVICIO (QoS).....	11
III.	DISEÑO E IMPLEMENTACIÓN	11
III.1.	CARACTERIZACIÓN DE LAS FUENTES DE TRÁFICO	11
III.2.	DEFINICIÓN Y CARACTERIZACIÓN DE LOS SERVICIOS DE UNA SMART CITY.....	13
III.3.	DEFINICIÓN DE LOS MECANISMOS DE CALIDAD DE SERVICIO (QoS)	20
III.3.1.	MARCADO (MARKING).....	20
III.3.2.	VIGILANCIA Y MEDICIÓN (POLICING / METERING).....	20
III.3.3.	ENCOLADO Y PROGRAMACIÓN.....	20
III.4.	ESTRATEGIA DE SIMULACIÓN.....	21
IV.	SIMULACIÓN Y ANÁLISIS DE RESULTADOS	21
IV.1.	SIMULACIÓN DE TRÁFICO DE LOS SERVICIOS ASOCIADOS	21
IV.1.1.	GESTIÓN INTELIGENTE DEL PATRIMONIO	21
IV.1.2.	GESTIÓN INTELIGENTE DE RESIDUOS URBANOS	23
IV.1.3.	ALUMBRADO INTELIGENTE	23
IV.1.4.	GESTIÓN EFICIENTE DE PLAZAS DE APARCAMIENTO	24
IV.1.5.	GESTIÓN DE RUIDO URBANO	24
IV.1.6.	MONITORIZACIÓN AMBIENTAL URBANA.....	25
IV.2.	SIMULACIÓN CONSOLIDADA DE TODOS LOS SERVICIOS	26
V.	CONCLUSIONES Y TRABAJO FUTURO	35

I. INTRODUCCIÓN

Una ciudad inteligente sostenible (Smart Sustainable City - SSC) es una ciudad innovadora que utiliza las tecnologías de información y comunicación (TIC) y otros medios para mejorar la calidad de vida de sus ciudadanos, mejorar la eficiencia de la operación, los servicios urbanos y la competitividad, asegurando que satisfice las necesidades de las generaciones presentes y futuras con respeto a aspectos económicos, sociales, ambientales, culturales, de salud, entre otros ^[1].



Fig.1. Smart City: Internet of Things ^[A]

Para una Smart City es muy importante recolectar toda la información de estos aspectos por medio de terminales y dispositivos heterogéneos con el objetivo de realizar una gestión y control de los mismos, para esto, se apoya en las Tecnologías de la Información y Comunicación (TIC) que conforman un conjunto de herramientas que hacen posible la existencia de las Smart Cities de acuerdo a las directrices establecidas por los administradores de la ciudad y con la colaboración imprescindible de sector privado y los ciudadanos. Son las TIC un elemento fundamental para la planificación y gestión económica las ciudades mejorando la calidad de vida y la accesibilidad de sus habitantes y asegurando un desarrollo sostenible económico, social y ambiental en mejora permanente. Las TIC implica dos secciones, la parte física que contiene las instalaciones y conexiones físicas como tal y la parte digital que involucra al Internet de las Cosas (IoT) con las correspondientes configuraciones de red. El presente proyecto se enmarca en la parte digital y solo en una conexión específica, además está basado en una herramienta de simulación de una estructura de red existente para determinar el rendimiento de una red.

El objetivo de este proyecto en primera instancia es determinar cómo definir las fuentes de tráfico de varios servicios verticales que se establecerán en una SSC, en segundo lugar, simular el comportamiento de estos servicios y modificar los enlaces de salida de la red municipal para el exterior con el fin de asegurar una adecuada Calidad de Servicio (QoS).

Para alcanzar esto, primero nos familiarizaremos con el uso de la herramienta de simulación “OMNeT++” que es una herramienta de simulación de red que permite evaluar el comportamiento de una red en operación en función de los resultados obtenidos y visualizados, de tal manera de

realizar modificaciones en caso de requerirse. Para aprender a cerca de esta herramienta de simulación me apoyé en la información de su sitio web oficial y también a través de proyectos anteriores. La simulación usada en este proyecto está basada en la simulación realizada por alumnos de la UPV con anterioridad, basados en la existencia de simulaciones y reportes previos. Estudié y aprendí la estructura de red y los métodos de definición de caracterización de las fuentes de tráfico. A continuación, en base a los requerimientos del ayuntamiento de la ciudad de Valencia sobre los futuros servicios de la ciudad inteligente sostenible que definen las condiciones que deben cumplir los dispositivos y sensores que se requieren para los seis servicios y definimos las caracterizaciones de cada fuente de tráfico. Después de esto, modifiqué la parte de las fuentes en las simulaciones existentes y probé los mecanismos de QoS para determinar los valores que sean los más apropiados para la mejor calidad de los servicios.

En este escenario, se definió la caracterización de los siguientes servicios: Gestión inteligente del patrimonio municipal, gestión inteligente de residuos urbanos, alumbrado inteligente en pedanías, gestión eficiente de plazas de aparcamiento, gestión inteligente del ruido urbano y monitorización ambiental urbana. También definimos los parámetros correspondientes para los mecanismos de QoS de cada servicio. Después de probar muchos grupos de parámetros de mecanismos de QoS, tales como la tasa de información comprometida (Committed Information Rate – CIR), tasa de información máxima (Peak Information Rate – PIR) y el ancho de banda, finalmente definimos la más adecuada para esta estructura basada en resultados que obtuvimos de las simulaciones.

El contenido de este trabajo consta de cinco partes: el capítulo 1 referencia a la introducción de todo el proyecto y este informe, el capítulo 2 trata de trabajos relacionados, así como la comprensión y conocimiento detallado del OMNeT++ como herramienta de simulación usada en el presente proyecto, el capítulo 3 trata del diseño y la implementación que hemos definido para los tipos de tráfico asociadas a las diferentes fuentes y servicios de una Smart City, el capítulo 4 trata de la simulación de las diversas fuentes de tráfico, la obtención de resultados y el análisis relativo de los mismos, finalmente en el capítulo 5, concluye el presente proyecto y lo que se debería tener en cuenta en el futuro para la implementación de los componentes de una Smart City tratados en el presente proyecto.

II. ANTECEDENTES

En la presente sección brindaremos una breve introducción de la ciudad inteligente junto con la herramienta de simulación y el escenario de este proyecto.

II.1. TRABAJOS RELACIONADOS

El desarrollo del presente trabajo está apoyado en documentaciones y artículos relacionadas con el tema como la Evaluación de la calidad del servicio de las comunicaciones en hospitales inteligentes^[2], la simulación de una red de comunicaciones para el control de radiaciones en

aplicaciones médicas^[3], y el pliego de prescripciones técnicas que regirán la realización del contrato de “Desarrollo de la iniciativa impulso VLCi”^[4], entre los principales.

II.2. CIUDAD INTELIGENTE SOSTENIBLE

Ciudad Sostenible Inteligente (Smart Sustainable City - SSC) es la visión holística de una ciudad que aplica las Tecnologías de la Información y Comunicación (TIC) para combinando con la infraestructura existente de una ciudad, ser capaz de converger la administración de todos estos sistemas con el propósito de gestionar y controlar todos estos recursos y de esta manera mejorar la calidad de vida y la accesibilidad de sus habitantes. Una ciudad inteligente, permite a los ciudadanos interactuar con ella de forma multidisciplinar y se adapta en tiempo real a sus necesidades de forma eficiente en calidad y costes, ofreciendo datos abiertos, soluciones y servicios orientados a los ciudadanos como personas, para resolver los efectos del crecimiento de las ciudades en ámbitos públicos y privados a través de la integración innovadora de infraestructuras con sistemas de gestión inteligente, consiguiendo así un ahorro de costos y energía.

El presente proyecto está basado en la arquitectura de una SSC que dispone de accesos fijos y móviles como lo mostrado en la Fig. 2, la arquitectura de red se divide principalmente en tres capas, en la parte superior está el backbone, luego tenemos la capa de red que contiene a los diferentes mecanismos de transmisión ya sean alámbricos o inalámbricos, luego está la capa de acceso en la que tenemos los diferentes tipos de sensores encargados de recolectar las diferentes fuentes de tráfico para ser transmitidos a las capas superiores en las que los enrutadores junto con las puertas de enlace son los encargados de transferir todos estos datos colectados a la red de conmutación de etiquetas multiprotocolo (MPLS) con los servidores correspondientes en la nube, es importante mencionar que los mecanismos de calidad de servicio de QoS, son aplicados en los enrutadores dentro de la capa de red.

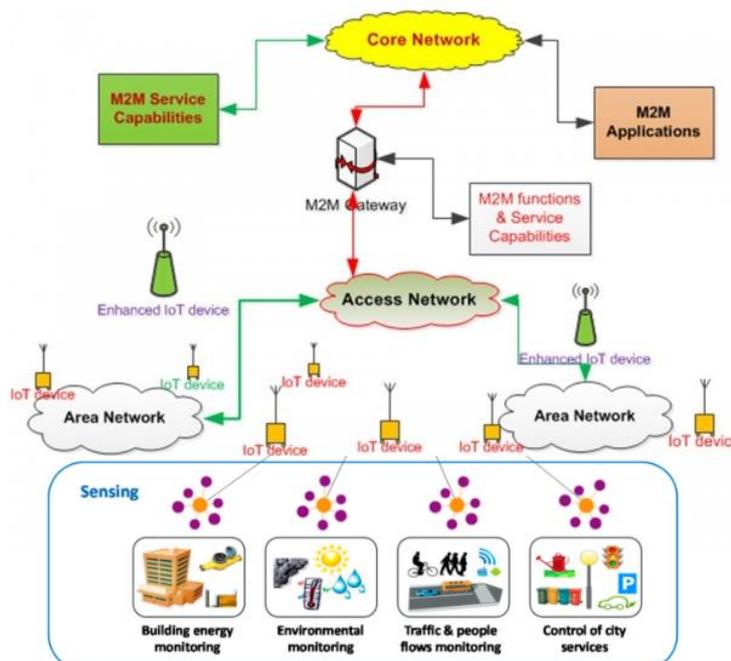


Fig.2. Censado y arquitectura de comunicación en una SSC^[B]

II.3. SIMULADOR OMNET++

OMNeT++ Es un simulador para redes full IP, basado en componentes modulares y extensibles programado y compilado en C++. OMNeT++ ofrece un IDE basado en Eclipse, un entorno gráfico en el tiempo de ejecución, dispone también de extensiones para simulaciones en tiempo real, emulaciones de red, integración de bases de datos, integración de Sistemas C y varias otras funciones^[5].

II.3.1. MÓDULOS Y COMPONENTES DE UNA SIMULACIÓN

Dado que OMNeT++ posee una arquitectura modular, diferentes módulos puedan ser conectados entre sí para formar módulos compuestos. Los módulos se comunican entre sí a través del envío y recepción de mensajes, estos pueden llevar estructuras de datos arbitrarias. Los módulos pueden pasar mensajes a través de las rutas predefinidas vía compuertas o conexiones, o directamente a sus destinos, éste último es muy usado para simulaciones inalámbricas. Los módulos pueden tener parámetros que pueden ser usados para personalizar el comportamiento y/o parametrizar la topología del modelo ^[6].

El simulador de eventos discretos ONMeT++, consta fundamentalmente de cinco componentes: el archivo de descripción de red (.ned el cual describe la estructura de esta red), el archivo de definición de esta red (.cc el cual define el modo de comunicación), el archivo de definición del mensaje (.msg), la librería núcleo de la simulación y la librería interface del usuario (.lib/.a) y el perfil (.ini). Entre estos, los archivos NED e INI son partes fundamentales de una simulación que necesitamos definir, además, usamos el archivo .XML para añadir algunas características de las fuentes. En el presente trabajo nos enfocaremos principalmente en tres que son:

II.3.1.1. Archivo NED

Este archivo es el más importante dentro de cada simulación, describe la estructura de red dentro de una simulación incluyendo todos los módulos y conexiones. En una simulación siempre hay muchos archivos NED que describen los diferentes módulos en el interior. Podemos importar muchas librerías para definir módulos y herramientas que se requieran, excepto el uso de códigos para definirlos y describirlos, también se puede diseñar la red gráficamente. En la Fig. 3 se presenta la estructura general de la red, misma que también es la vista gráfica del archivo “Network.ned”.

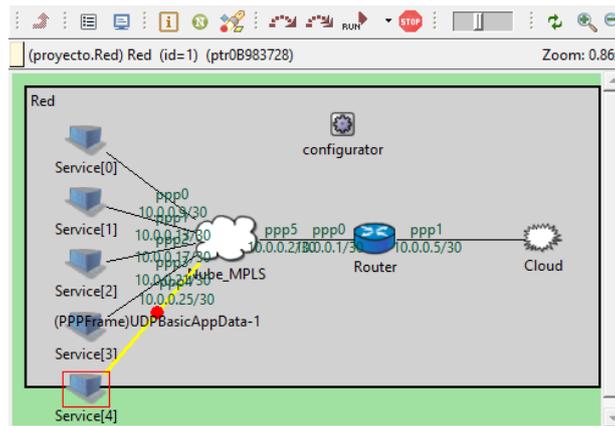


Fig. 3. “Network file” en modo gráfico

II.3.1.2. Archivo INI

Este archivo define la configuración de una simulación y puede haber muchos archivos INI para coincidir con una red, pero solo el de configuración puede ser seleccionado cuando se está ejecutando la simulación. En este archivo todas las configuraciones y limitaciones que se requieran añadir a una implementación pueden ser incluidas dentro, tales como el tiempo de ejecución y los perfiles insertados, además, algunos caracteres de parte de las fuentes y parámetros de los mecanismos de QoS que también son definidos en este archivo.

```

omnetpp.ini | DataSize.xml | ConfigTraffic.xml | Multi.anf
[General]
parallel-simulation = false

[Config Multi]
network = Red

#####
#
# CONFIGURACIÓN GENERAL #
#
#####

##### Duración #####
sim-time-limit = 3600s#1800s

##### Implementación de los módulos de QoS (No tocar) #####
**.RouterHosts.eth[*].ingressCType = "QoS"
**.Router.ppp[0].ingressCType = "QoS"
**.RouterHosts.ppp[*].queueType = "ColasQoS"
**.Router.ppp[1].queueType = "ColasQoS"
**.Nube_MPLS.ppp[*].queueType = "ColasQoS"

##### Fichero configuración de la nube MPLS (No tocar) #####
**.delayer.config = xmldoc("internetCloud.xml")

##### Configuración de los servidores y de parámetros de fuentes (No tocar) #####
**.Server*.numUdpApps = 1
**.Server*.udpApp[*].typename = "UDPSink"
**.Fuente*.localPort=-1 #Puerto efímero

##### Fuentes de tráfico de tiempo real #####
**.Datos.n=12
**.FuenteDatos.H[*].udpApp[*].typename = "UDPDatos"
**.FuenteDatos.H[*].udpApp[*].chooseDestAddrMode = "once"

# Fuente 1 (Datos)
**.FuenteDatos.H[0].udpApp[*].destAddresses = "Cloud.ServerDatos.S[0]"
**.FuenteDatos.H[0].udpApp[*].startTime = uniform(0s,180s)
**.Datos.*[0].udpApp[*].Port = 1001 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = crónico, 1003 = plata)
**.FuenteDatos.H[0].udpApp[*].data_size = xmldoc("DataSize.xml", "//data-size[id='parking-disable']")
**.FuenteDatos.H[0].udpApp[*].config = xmldoc("ConfigTraffic.xml", "//data-profile[id='60x180']")

# Fuente 2 (Datos)
**.FuenteDatos.H[1].udpApp[*].destAddresses = "Cloud.ServerDatos.S[1]"
**.FuenteDatos.H[1].udpApp[*].startTime = uniform(0s,180s)
**.Datos.*[1].udpApp[*].Port = 1003 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = crónico, 1003 = plata)
**.FuenteDatos.H[1].udpApp[*].data_size = xmldoc("DataSize.xml", "//data-size[id='parking-load']")
**.FuenteDatos.H[1].udpApp[*].config = xmldoc("ConfigTraffic.xml", "//data-profile[id='60x180']")

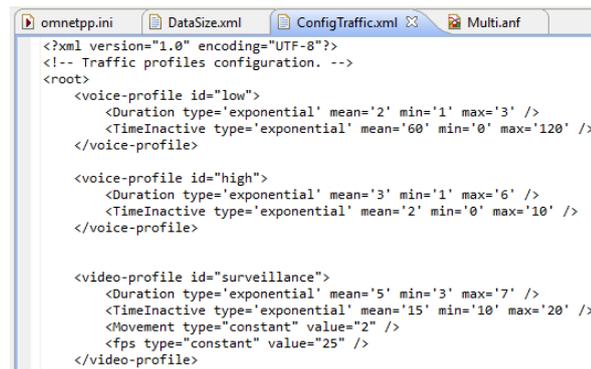
# Fuente 3 (Datos)
**.FuenteDatos.H[2].udpApp[*].destAddresses = "Cloud.ServerDatos.S[2]"
**.FuenteDatos.H[2].udpApp[*].startTime = uniform(0s,180s)
**.Datos.*[2].udpApp[*].Port = 1003 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = crónico, 1003 = plata)

```

Fig. 4. Archivo de simulación “omnetpp.ini”

II.3.1.3. Archivo XML

Estos archivos son usados para añadir algunos complementos a la configuración, podemos emplearlos para definir perfiles de fuentes de tráfico e inclusive algunas clasificaciones, filtros y definiciones de calidad. Los archivos con extensión “.xml” pueden ser insertados en el archivo “.ini” para optimizar y complementar la configuración añadiendo más limitaciones o parámetros a una simulación. Puede haber muchos archivos XML insertados en una simulación y pueden ser añadidos en cualquier posición que se requiera. En la Fig. 5 observamos una parte del archivo “ConfigTraffic.xml” definiendo algunos perfiles de configuración de diferentes fuentes de tráfico.



```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Traffic profiles configuration. -->
<root>
  <voice-profile id="low">
    <Duration type='exponential' mean='2' min='1' max='3' />
    <TimeInactive type='exponential' mean='60' min='0' max='120' />
  </voice-profile>

  <voice-profile id="high">
    <Duration type='exponential' mean='3' min='1' max='6' />
    <TimeInactive type='exponential' mean='2' min='0' max='10' />
  </voice-profile>

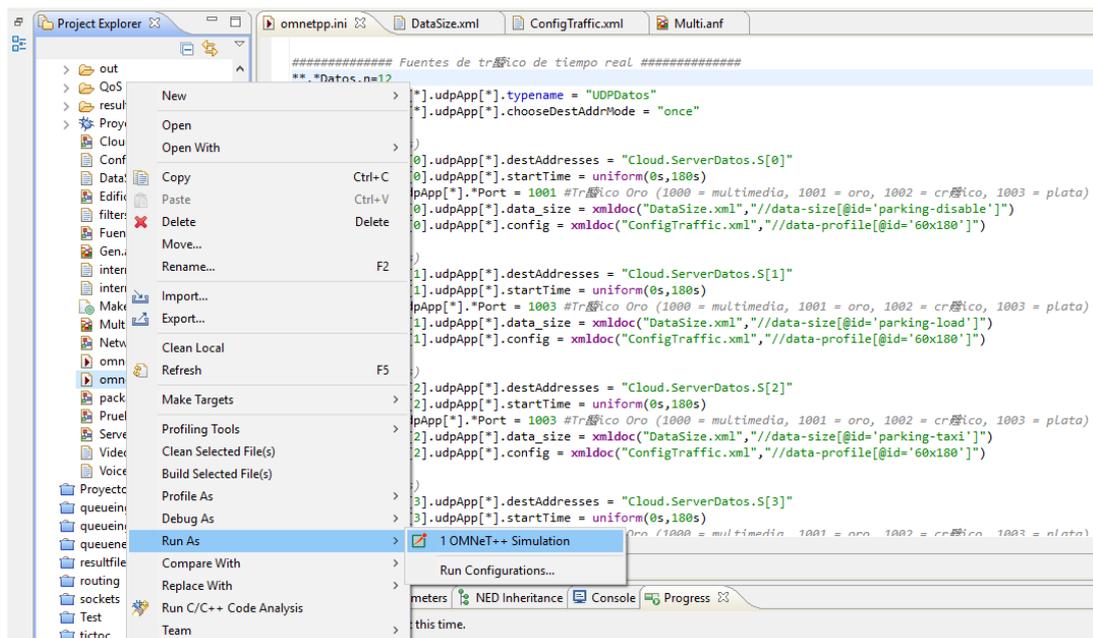
  <video-profile id="surveillance">
    <Duration type='exponential' mean='5' min='3' max='7' />
    <TimeInactive type='exponential' mean='15' min='10' max='20' />
    <Movement type="constant" value="2" />
    <fps type="constant" value="25" />
  </video-profile>
</root>

```

Fig. 5. Archivo “ConfigTraffic.xml”

II.3.2. ENTORNO DE EJECUCIÓN

Una vez definidos y configurados los tres tipos de archivos principales antes mencionados, podemos proceder con la simulación ejecutando el archivo “omnetpp.ini” con el proceso que se muestra en la Fig. 6 (omnetpp.ini → Run As → 1 OMNeT++ Simulation).



```

##### Fuentes de tráfico de tiempo real #####
**.*Datos.n:12
*) .udpApp[*].typename = "UDPDatos"
*) .udpApp[*].chooseDestAddrMode = "once"
)
0].udpApp[*].destAddresses = "Cloud.ServerDatos.S[0]"
0].udpApp[*].startTime = uniform(0s,180s)
pApp[*].Port = 1001 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = cráfico, 1003 = plata)
0].udpApp[*].data_size = xmldoc("DataSize.xml", "//data-size[@id='parking-disable']")
0].udpApp[*].config = xmldoc("ConfigTraffic.xml", "//data-profile[@id='60x180']")
)
1].udpApp[*].destAddresses = "Cloud.ServerDatos.S[1]"
1].udpApp[*].startTime = uniform(0s,180s)
pApp[*].Port = 1003 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = cráfico, 1003 = plata)
1].udpApp[*].data_size = xmldoc("DataSize.xml", "//data-size[@id='parking-load']")
1].udpApp[*].config = xmldoc("ConfigTraffic.xml", "//data-profile[@id='60x180']")
)
2].udpApp[*].destAddresses = "Cloud.ServerDatos.S[2]"
2].udpApp[*].startTime = uniform(0s,180s)
pApp[*].Port = 1003 #Tráfico Oro (1000 = multimedia, 1001 = oro, 1002 = cráfico, 1003 = plata)
2].udpApp[*].data_size = xmldoc("DataSize.xml", "//data-size[@id='parking-taxi']")
2].udpApp[*].config = xmldoc("ConfigTraffic.xml", "//data-profile[@id='60x180']")
)
3].udpApp[*].destAddresses = "Cloud.ServerDatos.S[3]"
3].udpApp[*].startTime = uniform(0s,180s)
)

```

Fig. 6. Inicio y ejecución de la simulación

Para iniciar la simulación, se debe dar clic en el botón “run” o en los botones “fast” y “express” para acelerar la velocidad de ejecución, además se tiene el número de eventos y el estado de los paquetes como se puede apreciar en la pantalla del simulador que se muestra en la Fig. 7. La estructura gráfica de la red también es mostrada en conjunto, adicional también se puede observar el estado de la red.

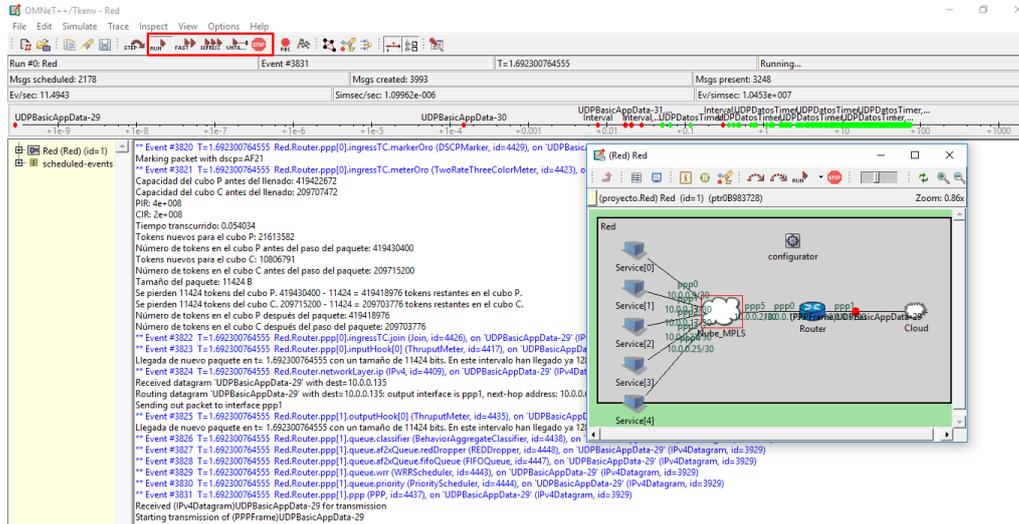


Fig. 7. Pantalla del simulador OMNeT++

II.3.3. ALCANCE Y ANÁLISIS DE RESULTADOS

Cuando la simulación alcanza su tiempo límite, se llamará a la función “finish()” y terminará automáticamente, entonces podemos revisar y hacer el análisis de los resultados obtenidos de la simulación, estos resultados se encuentran listados como se puede apreciar en la Fig. 8. En el mismo listado también podemos encontrar los gráficos de cada variable, incluyendo los tres tipos de fuentes, así como también el gráfico del rendimiento total del enlace de salida de un host mostrado en la Fig. 9.

Folder	File name	Config na...	R	Run id	Module	Name	Count	Mean	Std
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[0].RouterHosts.ppp[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[0].RouterHosts.ppp[0].ppp	bStatevector	75	0.493333333333333...	0.51
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[1].RouterHosts.ppp[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[1].RouterHosts.ppp[0].ppp	bStatevector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[2].RouterHosts.ppp[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[2].RouterHosts.ppp[0].ppp	bStatevector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[3].RouterHosts.ppp[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[3].RouterHosts.ppp[0].ppp	bStatevector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[4].RouterHosts.ppp[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Service[4].RouterHosts.ppp[0].ppp	bStatevector	211005	0.4999976303879055	0.51
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Cloud.RouterServ.ppp[0].queue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Cloud.RouterServ.ppp[0].ppp	bStatevector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Router.ppp[0].queue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Router.ppp[0].ppp	bStatevector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Router.ppp[1].queue.efQueue	queue.lengthvector	1	0.0	n.a.
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Router.ppp[1].ppp	bStatevector	211079	0.49999763121864...	0.51
/Proyec...	Multi-0vec	Multi	0	Multi-0-20180...	Red.Nube.MPLS.ooo[0].queue.efQueue	queue.lengthvector	1	0.0	n.a.

Fig. 8. Listado de resultados

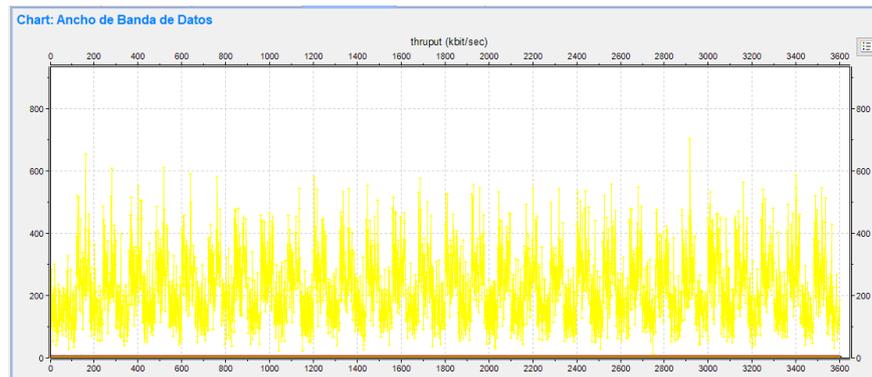


Fig. 9. Rendimiento (throughput) [kbits/s] de un enlace de salida

Podemos revisar cualquier ítem del archivo listado de resultados para conocer más detalles acerca de la simulación y los resultados obtenidos. Además, de requerirse, también se puede exportar los datos desde el OMNeT++ hacia MATLAB para realizar representaciones, análisis e interpretaciones.

II.4. DESCRIPCIÓN DEL ESCENARIO

La funcionalidad de este proyecto se basa en el uso de los dispositivos IoT dentro de la ciudad de Valencia los cuales son capaces de recolectar datos heterogéneos de los dispositivos de las diferentes fuentes de tráfico correspondiente a los componentes del proyecto Impulso VLCi y transferirlos hacia la Plataforma de servidores localizados en la nube, mismos que procesarán esta información y proporcionarán las aplicaciones correspondientes para la gestión de los diferentes servicios. El escenario completo de una Smart City ha sido establecido basado en la red interna del ayuntamiento de la ciudad de Valencia como se muestra de manera consolidada en la Fig. 10.

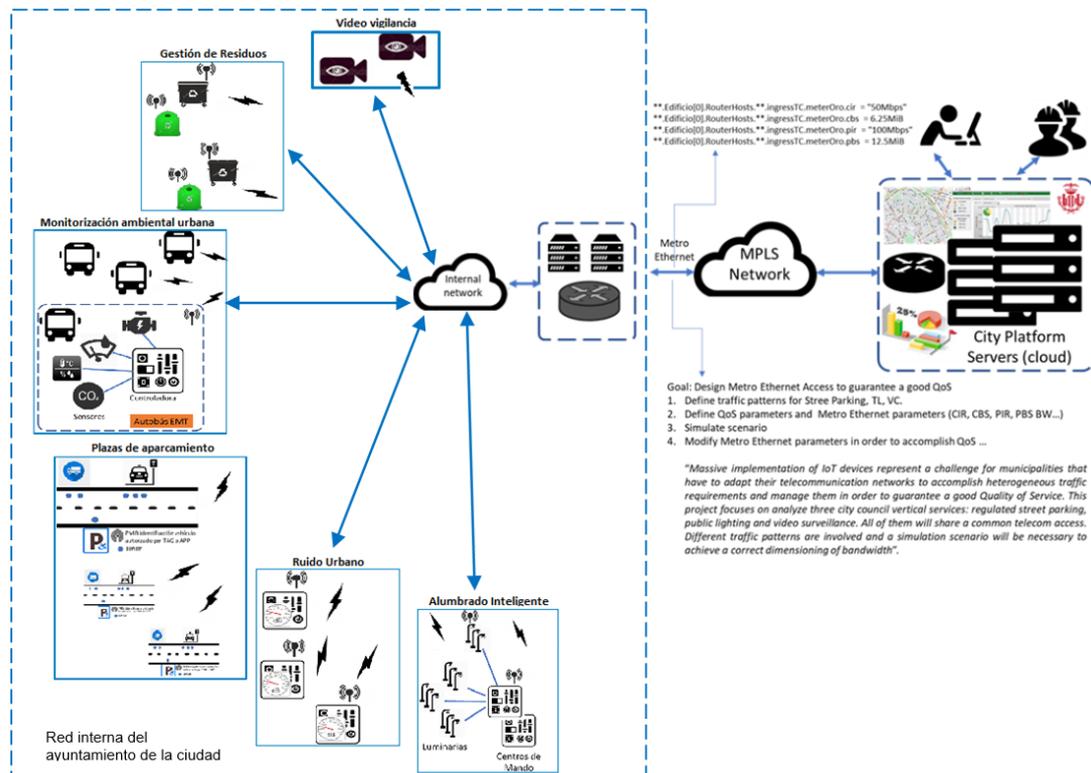


Fig. 10. Escenario del proyecto

La Fig. 10 es una vista funcional de la comunicación de la Smart City del Ayuntamiento de Valencia. En este proyecto, la red interna del Ayuntamiento de la ciudad es la responsable de recolectar todos los datos de los diferentes dispositivos IoT y transferirlos a través de un acceso común a la red MPLS, misma que reenviará los paquetes a la plataforma de servidores de la ciudad localizados en la nube. El detalle que realiza cada parte del escenario, la revisamos a continuación.

II.4.1. RED INTERNA DEL AYUNTAMIENTO

La red interna incluye todos los terminales y toda la infraestructura de red. Los terminales y sensores son responsables de capturar los datos del estado de la ciudad, mientras que la infraestructura formada por los conmutadores y los enrutadores se encargan de recolectar los datos en conjunto y transferirlos a la plataforma de servidores localizados en la nube. Para el Ayuntamiento de la ciudad, hay un gran número de diferentes fuentes de datos que deberán ser recolectados juntos para su procesamiento, sin embargo, en este proyecto me enfocaré en seis tipos de servicios: servicios de cámaras de video vigilancia, gestión de residuos urbanos, luminarias y alumbrado público, plazas de aparcamiento, gestión eficiente del ruido urbano, monitorización ambiental urbana.

La red interna del Ayuntamiento puede ser dividido en dos partes, una es la parte de la fuente de datos y la otra es la parte correspondiente al enlace. En la simulación de este proyecto, terminales y sensores conforman la parte de la fuente de datos, que al mismo tiempo son la parte inicial de la simulación de esta red. Todas las fuentes pueden ser clasificadas en tres tipos: voz, video y datos. Basados en esta clasificación, futuras definiciones serán hechas y mencionadas en este proyecto.

La puerta de enlace de este proyecto recolecta todos los datos generados y los transfiere hacia adelante hacia el exterior de la red que es la red MPLS. Los conmutadores y enrutadores que están en esta red se encargan de coleccionar todos estos datos heterogéneos en únicamente un acceso para toda la red interna. Esta parte es esencial para asegurar la calidad del servicio (QoS) de toda la red, además la QoS es siempre medida en la parte de los enrutadores para evaluar el rendimiento de la red.

II.4.2. RED MPLS

La red MPLS es una red de conmutación de etiquetas de paquetes multiprotocolo la cual es usada para la comunicación entre hosts y servidores ^[7]. La red MPLS es responsable de la transmisión de los paquetes recibidos desde la puerta de enlace de la red interna del Ayuntamiento para los correspondientes servidores remotos ya que tiene la capacidad para conectar entre dos puntos cualquiera, es muy adecuado para este escenario, además, la red MPLS soporta el uso de mecanismos QoS, mismos que son la base para la implementación de este proyecto.

II.4.3. PLATAFORMA DE SERVIDORES DE LA CIUDAD (CLOUD)

La plataforma de servidores de la ciudad es una plataforma integrada para controlar todos los servicios de manera conjunta. Está dividida en tres partes: el servidor de video, el servidor de voz y el servidor de paquetes de datos, mismos que son transmitidos desde los diferentes tipos de servicios antes mencionados, especialmente para el servidor de datos que es quien recibe los datos de todas las fuentes y trata con ellos en conjunto para al final disponer de información de utilidad.

II.4.4. MECANISMOS DE CALIDAD DE SERVICIO (QoS)

Estos mecanismos de QoS son aplicados para asegurar que los paquetes de servicios importantes no sean retrasados demasiado tiempo o borrados de la red y de esta manera se asegura que la red está operando en los niveles apropiados de eficiencia y calidad. Al definir la prioridad y limitación de descarte de paquetes, diferentes tipos de flujos son proporcionados con diferentes calidades de servicio. En total son cinco pasos: primero: filtros y medidores clasifican el tráfico en cuatro tipos (multimedia, oro, administración y plata) acorde a su caracterización. Segundo: Dos tasas y tres marcas de colores (trTCM ^[8]) están implicados a marcar el tráfico con tres diferentes colores (verde, amarillo y rojo) mostrando así si el tráfico puede ser transmitido satisfactoriamente o no, por lo que, en la parte de la administración de las colas, el tráfico es clasificado en tres clases (EF, AFxy que involucra 4 subniveles y BE) ^[9] Tercero: Re-envío acelerado (Expedited Forwarding EF) tiene la más alta prioridad para ser transmitida hacia adelante, mientras que el cuarto es el Mejor Esfuerzo (Best Effort BE) que tiene la prioridad más baja. Quinto: El Re-envío Garantizado (Assured Forwarding AF) tiene una prioridad media, en consecuencia, el ancho de banda es distribuido más equitativamente alcanzando una mejor calidad de la comunicación.

III. DISEÑO E IMPLEMENTACIÓN

La caracterización de las tres partes principales fuentes (video, voz y datos) son las primeras en ser introducidas, a continuación, irán las caracterizaciones específicas de los servicios antes mencionados y analizados en conjunto con la definición en este proyecto. Después de esto, lo siguiente en ser analizado será los mecanismos de calidad de servicio (QoS) para finalmente en la instancia final de este proyecto, revisar la simulación del servicio y los pasos para la optimización de la calidad del servicio.

III.1. CARACTERIZACIÓN DE LAS FUENTES DE TRÁFICO

Las fuentes de tráfico pueden ser divididas en tres clases: voz, video y datos. Las fuentes de voz y video pertenecen a la comunicación tipo humano (Human Type Communications - HTC) que tiene tiempos activos e inactivos. Por otra parte, la fuente de datos como tal, pertenece a la comunicación

tipo maquina (Machine Type Communications - MTC) que tiene intervalos tipo ráfagas de datos y un amplio número de fuentes.

En la definición de una fuente, hay cuatro partes en total: el tiempo de inicio, la clase de servicio, el codificador (o calidad de video, tamaño de datos para los datos) y el perfil. El tiempo de inicio es el tiempo cuando el servicio empieza a transmitir la información, comúnmente una variable aleatoria en un rango definido. La clase de servicio es la prioridad del servicio, es decir, es su importancia. En este proyecto hemos definido en tres tipos: multimedia, oro (gold) y plata (silver). La clase multimedia es el de más alta prioridad, mientras que la de plata es la más baja. La codificación, la calidad y el tamaño de los datos define el tamaño del paquete de las fuentes. Para la voz y video, hay muchos estándares de calidad y tamaño. En OMNeT++, los perfiles almacenan la configuración de una fuente. En la Tabla 1 se muestra las caracterizaciones que hemos definido para estos tres tipos de tráfico.

Parámetro	Tipo	Descripción
Tamaño del recurso de texto	Tamaño de datos	Tamaño de cada paquete de texto (en bytes)
Tamaño del recurso de imagen		Tamaño de cada paquete de imagen (en bytes)
Número de recursos de texto		Número de recurso de texto en el archivo
Número de recursos de imagen		Número de recurso de imagen en el archivo
Número de paquetes	Perfil	Número de paquetes en una sesión
Intervalo de envío		Tiempo entre dos paquetes
Número de ráfagas		Número de sesiones en total
Intervalo de ráfagas		Tiempo entre dos sesiones

Tabla 1: Caracterización fuente de datos

En este proyecto, las fuentes de datos son los principalmente usados y provienen de la mayoría de los servicios y sus sensores asociados. Los datos de forma conjunta consiguen representar la información de la vida real, en este caso, los servicios asociados a una ciudad. Para los datos tipo texto, por ejemplo, las configuraciones arriba mencionadas necesitaban ser definidas como un patrón de tráfico. La información del tamaño de los datos son todos almacenados en el archivo “*DataSize.xml*” y la información del perfil son todos guardados en el archivo “*ConfigTraffic.xml*”.

Parámetro	Tipo	Descripción
Res X	Calidad de video	Ancho del frame (en píxeles)
Res Y		Altura del frame (en píxeles)
Duración	Perfil	Duración de un video (minutos)
Tiempo inactivo		Tiempo cuando no es generado ningún video (minutos)
Movimiento		El nivel de movimiento de un video (1 a 4)
Frame por segundo (fps)		Número de frames en 1 segundo

Tabla 2: Caracterización fuente de video

Parámetro	Tipo	Descripción
-----------	------	-------------

Tamaño del paquete	Códec voz	Tamaño de cada paquete (in bytes)
Intervalo de envío		Tiempo entre dos paquetes (minutos)
Duración	Perfil	Duración de un archivo de voz (minutos)
Tiempo inactivo		Tiempo cuando no es generado ningún archivo de voz (minutos)

Tabla 3: Caracterización fuente de voz

De la misma manera que las fuentes de datos, las configuraciones de las fuentes de video y voz, son almacenadas por separado en dos archivos. La definición sobre el tamaño del archivo es guardada en los archivos *videoQuality.xml* y *voiceCode.xml* y el detalle sobre el patrón de la información es almacenado en el archivo *ConfigTraffic.xml*. En este proyecto, la fuente de voz, no está incluida en ningún servicio y la fuente de video se incluye solamente en un tipo de servicio.

III.2. DEFINICIÓN Y CARACTERIZACIÓN DE LOS SERVICIOS DE UNA SMART CITY

En esta sección, definiremos la caracterización de seis servicios de una Smart City en base a los requerimientos dispuestos por cada uno de componentes del proyecto VLCi del Ayuntamiento de Valencia. Tenemos sensores correspondientes para cada servicio y definimos la caracterización de los sensores en base a la hoja de datos de cada uno de los sensores teniendo en cuenta las condiciones en la vida real. A continuación, las caracterizaciones que definimos para todos los servicios.

III.2.1. PATRIMONIO MUNICIPAL INTELIGENTE

Este servicio es la recolección de tres tipos de sensores de datos y las cámaras de video vigilancia. El propósito de este servicio es proveer una protección inteligente del patrimonio arqueológico y cultural de esta ciudad. Al proporcionar un sistema de vigilancia y una estación meteorológica, el Ayuntamiento de la ciudad puede obtener información en tiempo real del estado y las condiciones del patrimonio, de tal manera por ejemplo de si se requiere, podría hacer un control de la temperatura, la humedad y otros estados del patrimonio en cualquier momento. Para el sistema de video vigilancia, se compararon varios tipos de cámaras en base a los requerimientos y las condiciones establecidas por el Ayuntamiento de la ciudad. En la Tabla 4 se presenta esta comparación.

Parámetro	CÁMARA			
	IP2M-846EB	IP2M-850E	TV-IP450P	TV-IP450PI
Iluminación mínima	Color: 0.05 Lux B / W: 0.005Lux	Color: 0.05Lux B / W: 0.005Lux	Color: 0.05Lux B / W: 0.01Lux	Color: 0.05Lux B / W: 0.01Lux
velocidad de obturación	1/1 - 1 / 30,000	1/3 - 1 / 30,000	1/1 - 1 / 10,000 s	1/1 - 1 / 10,000 s
Enfoque	Digital: 16x Óptica: 4x	Digital: 16x Óptico: 20x	Digital: 16x Óptico: 20x	Digital: 16x Óptico: 20x
Ángulo de visión	H: 116.5 ° - 34.5 ° V: N / A	H: 54.1 ° - 3.2 ° V: N / A	H: 54 ° V: N / A	H: 54 ° V: N / A
Longitud focal	2.7 mm - 11 mm	4.7mm - 94mm	4.7mm - 94mm	4.7mm - 94mm
Apertura máxima	F 1.6 a 2.8	F 1.4 a 2.6	F1.4 - 3.5	F1.4 - 3.5
Ángulo de rotación / inclinación	Pan 0 - 360 ° Inclinación: 0 - 90 °	Pan 0 - 360 ° Inclinación: -15 - 90 °	Pan 0 - 360 ° Inclinación -5 - 90 °	Pan 0 - 360 ° Inclinación -5 - 90 °

Día /noche	Sí	Sí	Sí	Sí
Resolución	1280x720	1280x720	1280x960	1280x960
Codificación de video	H.264 / MJPEG	H.264 / MJPEG	H.264 / MJPEG	H.264 / MJPEG
Taza frame máx.	30 fps	30 fps	30 fps	30 fps
Detección de movimiento	N / A	Sí	Sí	N / A
Manejo de alarmas	2/1 de canal de entrada / salida	2/1 de canal de entrada / salida	Alarma externa	Alarma externa
Detección de audio	Sí	Sí	Sí	Sí
Entrada de micrófono	N / A	N / A	Externo	Externo
Puerto de red	RJ-45	RJ-45	100 Base-T	100 Base-T

Tabla 4: Comparación de tipos de cámaras

Para este servicio hemos comparado cuatro tipos de cámaras correspondientes a dos fabricantes y aunque ninguna de ellas cumple todos los requerimientos dispuestos por el Ayuntamiento, acorde a la importancia de cada función, sugerimos como la mejor opción es la tercera cámara, la “*TP-IP450P*” por las siguientes razones: 1) La detección de movimiento es sin duda una de las más importantes funciones a tener en cuenta para este servicio. 2) El requerimiento dispuesto para la velocidad de obturación es de 1/1 a 1/1000 s, y la “*TP-IP450P*” es la más cercana ya que de las otras opciones son demasiado. Seleccionado el tipo, definiremos la caracterización para la video vigilancia en base a la hoja de datos del dispositivo y consideraciones previas del proyecto.

Video vigilancia					
Número	20				
Hora de inicio	Uniforme (0 s, 180 s)				
Clase de servicio	Oro				
Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	1			
	Intervalo de ráfagas	3600			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 1000 B	Min: 100 B	Max: 2000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0		Fin: 10

Tabla 5: Caracterización fuente de video vigilancia

En la estación meteorológica, hay tres sensores instalados para conocer las condiciones en tiempo real, siendo el sensor de temperatura y humedad, el sensor de iluminación y ultravioleta, y el sensor de gas. Estos sensores son usados para detectar las condiciones en tiempo real dentro de las locaciones de cada patrimonio, asegurando así que exista una protección total. En base a los requerimientos dispuestos, seleccionamos algunas opciones para finalmente definir tres sensores como se muestran en la Tabla 6.

Para la temperatura y humedad, seleccionamos un sensor que puede detectar los dos requerimientos a la vez, revisando los rangos de medición y su relación con los requisitos, seleccionamos el sensor "*Transmitter EE33-M*"^[10], este sensor es el que mayor precisión presenta en

base a los requerimientos planteados. Para el sensor de iluminación y luz ultravioleta, deberíamos ser capaces de detectar el brillo ambiental y la fuerza de la luz solar directamente en la superficie, en base a estas necesidades, seleccionamos el sensor “*UVB+UVA Sensor PMA1107*”^[11]. Para el sensor de gas, seleccionamos uno que puede realizar medidas de NO₂, O₃, CO y SO₂ en el medioambiente y además también puede medir las partículas PM_{2.5} y PM₁₀, además debido a que este sensor debería ser usado en interiores para protección del patrimonio, seleccionamos el que tiene las más altas prestaciones de medida y exactitud dispone, siendo el “*Vaisala Air Quality Transmitter AQT400*”^[12]. Las siguientes tablas muestran la caracterización que se han definido basándonos en sus especificaciones técnicas y aplicaciones para estos servicios.

Sensor de temperatura y humedad		
Número		21
Hora de inicio		Uniforme (0 s, 180 s)
Clase de servicio		Plata
Perfil de datos	Número de paquetes	1
	Intervalo de envío	1
	Número de ráfaga	1
	Intervalo de ráfagas	3600
Tamaño de datos	Tamaño del recurso de texto	20 B
	Recursos numéricos	2

Tabla 6: Caracterización del sensor de temperatura y humedad

Iluminación y UV		
Número		8
Hora de inicio		Uniforme (0s, 180s)
Clase de servicio		Plata
Perfil de datos	Número de paquetes	1
	Intervalo de envío	1
	Número de ráfaga	1
	Intervalo de ráfagas	3600
Tamaño de datos	Tamaño del recurso de texto	20
	Recursos numéricos	1

Tabla 7: Caracterización del sensor de iluminación y ultravioleta

Sensor de gases		
Número		8
Hora de inicio		Uniforme (0s, 180s)
Clase de servicio		Plata
Perfil de datos	Número de paquetes	1
	Intervalo de envío	1
	Número de ráfaga	1
	Intervalo de ráfagas	3600
Tamaño de datos	Tamaño del recurso de texto	20
	Recursos numéricos	4

Tabla 8: Caracterización del sensor de gas

III.2.2. GESTION INTELIGENTE DE RESIDUOS

A la gestión inteligente de residuos urbanos en Valencia le interesa conocer el nivel de llenado de todos los contenedores de la ciudad, de esta manera el Ayuntamiento puede organizar sus recursos para gestionar los mismos en los tiempos apropiados. Para esta gestión solo necesitamos un tipo de sensor, el mismo que debería ser capaz de detectar el nivel de llenado del contenedor y transmitir esos datos al servidor cada vez que se requiera. Acorde a éste y otros requerimientos, el sensor más apropiado es el “*U-Dump M2M waste management sensor*”^[13].

Este sensor detecta el nivel porcentual de llenado del contenedor cada hora y envía al servidor esta información de interés en tiempo real. Estos datos pueden ser transmitidos a través de las redes GPRS / 3G hacia la nube, además, debe ser de fácil instalación. La caracterización para este sensor, al igual que los anteriores está basado en su hoja de datos más las condiciones en la vida real.

Gestión de residuos					
Número	5154				
Hora de inicio	Uniforme (0 s, 180 s)				
Clase de servicio	Plata				
Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	1			
	Intervalo de ráfagas	3600			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 1000 B	Min: 100 B	Max: 2000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0		Fin: 10

Tabla 9: Caracterización del sensor de gestión de residuos

III.2.3. ILUMINACIÓN INTELIGENTE

El objetivo de este servicio es proveer de inteligencia a los centros de mando de iluminación del Ayuntamiento de la ciudad de Valencia, especialmente en aquellos distritos donde se necesita minimizar la existencia de averías provocadas por robos de cobre. El Ayuntamiento requiere controlar la luz ambiental para conocer si alguna persona realiza alguna aproximación a las luminarias o sus dispositivos complementarios. Para esto necesita controlar remotamente la iluminación y así generar alertas oportunas. Con el fin de combinar ambas funciones, para la iluminación inteligente definimos el sensor “*TMD2725*”^[14], la razón es que este sensor puede detectar la luz ambiente y la proximidad juntos, y también puede proveer una reducción de potencia. La caracterización para este sensor de luz ambiente (ALS) y proximidad se muestra en la Tabla 10.

Iluminación inteligente	
Número	2000
Hora de inicio	Uniforme (0 s, 180 s)
Clase de servicio	Plata

Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	120			
	Intervalo de ráfagas	120			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 1000 B	Min: 0 B	Max: 2700 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0	Fin: 5	

Tabla 10: Caracterización para el sensor de luz ambiente y proximidad

III.2.4. GESTION DE PLAZAS DE APARCAMIENTO

Este servicio trata de la implementación y gestión inteligente de los espacios de aparcamiento en Valencia, estos espacios están divididos en tres tipos: espacios para personas con discapacidad, espacios para carga y descarga y espacios para taxis; cada tipo tiene sus propios requerimientos y caracterizaciones. Sin embargo, debido a las funciones similares en la gestión de espacios, los tres tipos de aparcamientos pueden usar un mismo sensor, pero con tres diferentes configuraciones. En nuestra investigación, definimos dos tipos de sensores que satisfacen estos requerimientos, uno es del fabricante “*Urbiotica*” y el otro es del fabricante “*Libelium*”, en la Tabla 11 se muestra la comparación.

Parámetro	Urbiótica	Libelium
Comunicación	IEEE 802.15.4	LoRaWAN, SigFox
Detección	Magnética	Magnética
Impermeabilidad	IP 67	IP 68
Tiempo de vida	10 años	10 años
Flexibilidad de configuración	Tasa de medición	mas parámetros pueden ser configurados

Tabla 11: Caracterización sensores plazas aparcamiento

De las comparaciones realizadas, el sensor de “*Libelium*” fue el seleccionado debido a sus mejores opciones de configuración. Ejemplo, primero, el uso de LoRaWAN como método de comunicación proporciona una gama más amplia y un menor consumo de energía. Segundo, el sensor de “*Libelium*” dispone de varios modos de funcionamiento y una muy alta flexibilidad de configuración. Tercero, el sensor de “*Libelium*” puede ser instalado sobre la superficie o bajo la misma en caso de requerirse sin problemas. Por lo tanto, para este proyecto seleccionamos el sensor de “*Libelium*”^[15] por ser el de mejores prestaciones. En la Tabla 12, definimos las caracterizaciones para este sensor en base a su hoja de datos y los requerimientos dispuestos por el Ayuntamiento para este servicio.

Aparcamiento para personas con discapacidad		
Número	425	
Hora de inicio	Uniforme (0 s, 180 s)	
Clase de servicio	Plata	
Perfil de datos	Número de paquetes	1
	Intervalo de envío	1
	Número de ráfaga	60

	Intervalo de ráfagas	180			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 500 B	Min: 10 B	Max: 2000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0		Fin: 6
Aparcamiento para carga y descarga					
Número	125				
Hora de inicio	Uniforme (0 s, 180 s)				
Clase de servicio	Plata				
Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	60			
	Intervalo de ráfagas	180			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 300 B	Min: 10 B	Max: 1000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0		Fin: 6
Aparcamiento para taxis					
Número	126				
Hora de inicio	Uniforme (0 s, 180 s)				
Clase de servicio	Plata				
Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	60			
	Intervalo de ráfagas	180			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 300 B	Min: 10 B	Max: 1000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0		Fin: 6

Tabla 12: Caracterización para los sensores de aparcamiento de personas con movilidad reducida, carga y descarga y taxis.

III.2.5. GESTIÓN DE RUIDO URBANO

El objetivo de este servicio es establecer un sistema de control de los niveles de ruido en las calles, plazas, áreas o zonas más propensas a estos escenarios, de esta manera los lugares donde más ruido se genere, puedan ser controlados con el fin de conservar la mantener la salud ambiental especialmente en las noches. Para este propósito, el sensor debe detectar los niveles de ruido y activar alarmas cuando exceda los valores máximos permitidos acorde a su capacidad y precisión, alcance y facilidad de instalación y comunicación. En este proyecto hemos seleccionado dos sensores como candidatos, a continuación, la comparación.

Dispositivo	Rango de medición	Resolución	Comunicación	Temperatura de operación	Protección ambiental
U-sound	40 - 110 dB	0.1 dB	WiFi, Ethernet, 3G	[-33°C; -65°C]	Viento y lluvia
TA120	35 - 120 dB	0.1 dB	WiFi, Ethernet, GPRS / 3G	[-10°C; -50°C]	Lluvia, nieve, viento y pájaros

Tabla 13: Comparación de dos sensores de ruido

De los presentados, para este servicio seleccionamos “TAI20” del fabricante “CESVA”^[16] debido a que tiene un mayor rango de medida y una mayor protección frente a agentes externos. La caracterización para este para este servicio con este sensor, se muestra en la Tabla 14.

Gestión del ruido					
Número	500				
Hora de inicio	Uniforme (0 s, 180 s)				
Clase de servicio	Plata				
Perfil de datos	Número de paquetes	1			
	Intervalo de envío	1			
	Número de ráfaga	20			
	Intervalo de ráfagas	1800			
Tamaño de datos	Tamaño del recurso de texto	Tipo: Exponencial	Media: 1000 B	Min: 100 B	Max: 2000 B
	Recursos numéricos	Tipo: Uniforme	Inicio: 0	Fin: 10	

Tabla 14: Caracterización para el sensor de ruido

III.2.6. GESTIÓN DE MONITORIZACIÓN AMBIENTAL URBANA

En la actualidad Valencia cuenta con este servicio, pero corresponden a sensores fijados en pocas locaciones estáticas, por lo que el objetivo de este servicio es monitorizar y evaluar la calidad del aire a nivel de comunicación y derivados del tráfico, pero mediante una monitorización en movimiento, por lo que esta medición se realizará a través de sondas instaladas en vehículos municipales. Este servicio incluye la monitorización de los principales contaminantes que afectan la calidad del aire como gases CO, NO₂ y O₃, partículas de suspensión PM_{2.5}, temperatura y humedad en el exterior inclusive, de tal manera de asegurar el funcionamiento de cualquier condición meteorológica. Debido a la similitud de requerimientos que presente este servicio, hemos seleccionado el sensor de gases mencionado en la sección III.1.1.1. con la diferencia que en este caso será instalado en medios móviles. La caracterización para este servicio con este sensor, se muestra en la Tabla 15.

Sensor de monitorización ambiental urbana		
Número	40	
Hora de inicio	Uniforme (0s, 180s)	
Clase de servicio	Plata	
Perfil de datos	Número de paquetes	1
	Intervalo de envío	1
	Número de ráfaga	1
	Intervalo de ráfagas	3600
Tamaño de datos	Tamaño del recurso de texto	20
	Recursos numéricos	10

Tabla 15: Caracterización para el sensor de monitorización ambiental urbana

III.3. DEFINICIÓN DE LOS MECANISMOS DE CALIDAD DE SERVICIO (QoS)

Los mecanismos de calidad de servicio (QoS) contienen cinco fases en total: clasificación, marcado, vigilancia o supervisión, encolado y programación^[17]. El primer paso es clasificar todos los servicios dentro de diferentes *Clases de Servicios* (CoS) los cuales han sido hechos en el paso previo de la definición de caracterización de las fuentes de tráfico. En este proyecto, usamos tres clases definidas basadas en una Macrolan (protocolo usado por Telefónica – Movistar para grandes clientes) aplicadas al servicio de un hospital inteligente^[2] donde se tienen: multimedia, oro y plata, después de esto se definieron las siguientes cuatro fases.

III.3.1. MARCADO (MARKING)

En esta fase, todos los campos organizados en la etapa de clasificación, se les dará un valor en IP o MPLS a las cabeceras de los paquetes^[18], por consiguiente, todo el tráfico puede ser identificado posteriormente en base a su cabecera. En este proyecto, el módulo de marcado DSCP es usado para esta etapa. El marcador DSCP es usado para diferenciar la prioridad de cada tráfico por medio de las diferentes codificaciones en las cabeceras. Luego, en los mecanismos de QoS, el tráfico con más alta prioridad será transmitido primero. La definición de prioridad está basada en la parte de clasificación (multimedia, oro o plata).

III.3.2. VIGILANCIA Y MEDICIÓN (POLICING / METERING)

Esta etapa es usada para determinar el ancho de banda ocupado por cada tipo de servicio después de la clasificación. Si la velocidad de datos de un servicio excede los límites pre-definidos, los paquetes serán procesados especialmente, como también pueden ser descartados o serán dados otros valores DSCP^[19]. En este proyecto, dos tasas representadas mediante tres colores (Three Colour Meters - trTCM) son usados como vigilancia.

Al igual que su nombre, trTCM tiene dos tipos de tasas (Tasa de información máxima – Peak Informatio Rate PIR y la Tasa de información comprometida – Committed Information Rate CIR). La PIR es la tasa más alta que esta conexión puede obtener y la CIR es la tasa mínima prometida por esta conexión. Después de comparar entre PIR y el CIR, los paquetes son marcados con tres diferentes colores, así, si un paquete es marcado como rojo, probablemente será dado de baja; si es marcado como amarillo, será transmitido tan pronto como sea posible; y si es marcado como verde, es casi imposible que sea descartado. En este proyecto, hay cuatro valores definidos para una fuente en supervisión: PIR, CIR, PBS (Peak Burst Size – Tamaño máximo de ráfaga) y el CBS (Committed Burst Size – Tamaño de Ráfaga Comprometida), el valor aproximado intentaremos mostrarlo en el ítem IV.

III.3.3. ENCOLADO Y PROGRAMACIÓN

El encolado es el proceso responsable de la transmisión de paquetes que están en la cola de salida. En un puerto de salida, hay varios tipos de colas debido a las diferentes prioridades^[20] por lo que

alcanzarán diferentes estrategias de transmisión. En este proyecto, las colas están clasificadas en cuatro tipos: re-envío acelerado (Expedited Forwarding - EF), re-envío garantizado (Assured Forwarding – AF1x, AF2x) y el mejor esfuerzo (Best Effort – BE). EF tiene la prioridad más alta y BE la más baja.

La programación es la última etapa del mecanismo de calidad de servicio (QoS). El planificador usado en este proyecto es el ponderado Round Robin (Weighted Round Robin – WRR), una especie de programador ponderado de ancho de banda responsable de organizar colas no prioritarias cuando la priorización de alguna esté inactiva ^[21].

III.4. ESTRATEGIA DE SIMULACIÓN

En el presente proyecto la implementación de esta simulación está dividida en dos etapas: primero implementamos la simulación de cada tipo de fuente de tráfico por separado y la segunda es la ejecución de la simulación de todos los servicios en conjunto. El primer paso está dirigido a obtener el ancho de banda necesario para cada servicio y verificar sus comportamientos. El segundo paso es alcanzar la calidad de servicio de la red agregada. El detalle de todas las etapas de la implementación de la simulación se menciona en el ítem IV.

IV. SIMULACIÓN Y ANÁLISIS DE RESULTADOS

El presente segmento se divide en dos partes: la primera es donde todos los servicios son probados por separado para chequear su comportamiento y encontrar los rangos apropiados de desempeño para cada una de ellas. La segunda es donde todos los servicios son implementados en conjunto, a su vez, esta parte puede ser dividida en dos subprocesos. 1) Encontrar los valores apropiados de CIR y PIR para cada clase. 2) Encontrar el valor apropiado de ancho de banda total para todos los servicios. En el primer subproceso, lo dividiremos en base a tres condiciones: a) $CIR < PIR < \text{máximo rendimiento}$. b) $CIR < \text{máximo rendimiento} < PIR$. c) $\text{máximo rendimiento} < CIR < PIR$ para determinar la influencia de CIR y PIR y la condición más apropiada para determinar el rendimiento final de la red. En el segundo subproceso, se probarán tres valores: 10 Mbps, 100 Mbps y 1Gbps con el fin de determinar cuál es el valor más apropiado para la red interna del Ayuntamiento de la ciudad de Valencia.

IV.1. SIMULACIÓN DE TRÁFICO DE LOS SERVICIOS ASOCIADOS

En esta sección, implementaremos y simularemos los seis servicios por separado para analizar su comportamiento y definir un patrón de tráfico.

IV.1.1. GESTIÓN INTELIGENTE DEL PATRIMONIO

En este servicio, existen 4 fuentes en total: una corresponde al video y tres a fuentes de datos como tal (temperatura y humedad, luz y ultravioleta, gases), los de datos, no son sensibles al tiempo por lo

que los hemos clasificado como clase “*plata*”, mientras que la video vigilancia es en tiempo real por lo que la clasificamos como clase tipo “*oro*”. En la Fig. 11 y 12 se presenta los resultados de las fuentes de video y de los tres tipos de fuentes de datos.

1) VIDEO VIGILANCIA – CLASE “ORO”

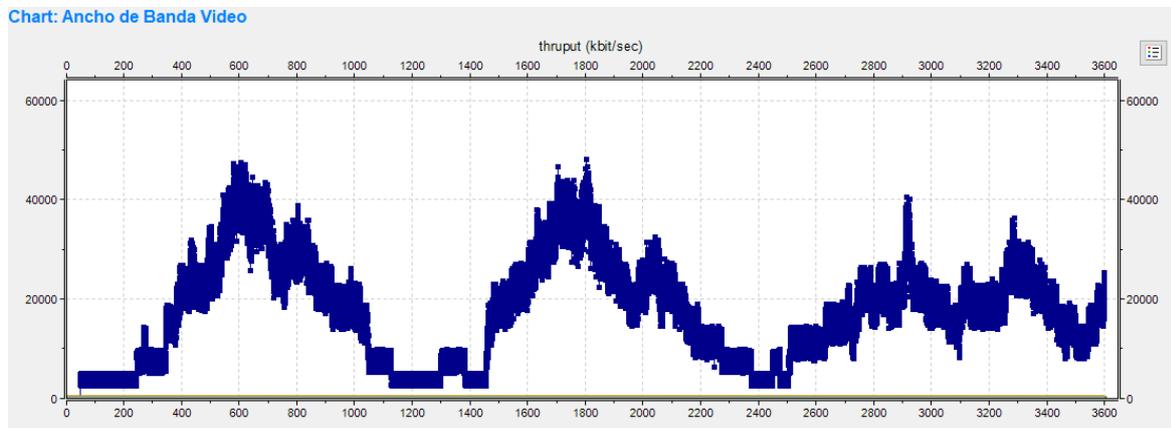


Fig. 11. Rendimiento de 20 cámaras juntas

En la Figura 11 se presenta el desempeño de 20 fuentes de video juntas después de simular por al menos 3 horas. De lo observado, podemos definir que el rango de desempeño o rendimiento va desde aproximadamente los 0 Mbps hasta alrededor 48 Mbps y el rendimiento medio es aproximadamente 28 Mbps. A partir de esto podemos determinar que el ancho de banda permitido para las fuentes de video debe por lo menos ser superior a los 30 Mbps.

2) FUENTE DE DATOS – CLASE “PLATA”

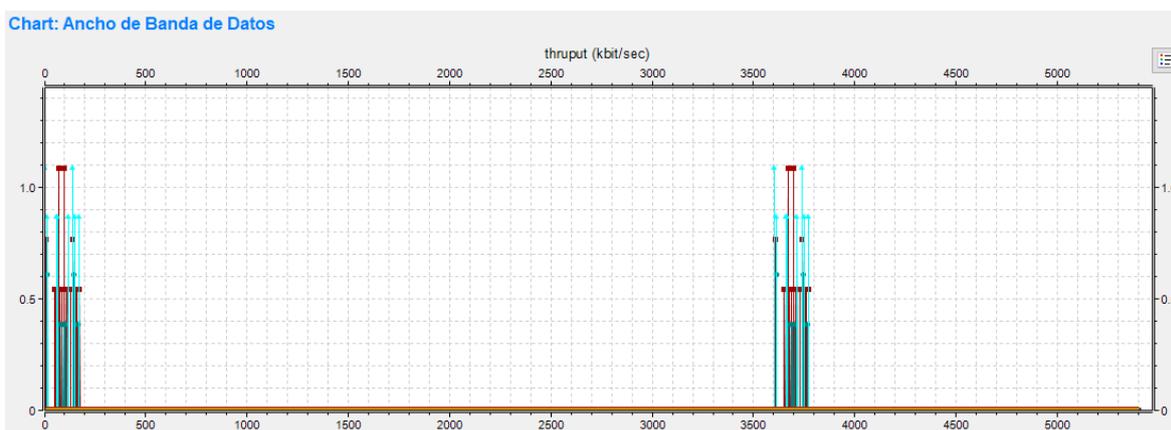


Fig. 12. Rendimiento de tres sensores tipo de datos

La Figura 12 muestra el rendimiento de tres tipos de sensores juntos los cuales incluyen 21 sensores de humedad y temperatura, 8 de luz visible y ultravioleta y 8 sensores de gas. En la misma figura se puede observar que rendimiento máximo de un sensor es aproximadamente 1,1 Kbps por lo que el rendimiento total es menos de 2 Kbps.

IV.1.2. GESTIÓN INTELIGENTE DE RESIDUOS URBANOS

En este servicio, existe únicamente un tipo de sensor para detectar el nivel de llenado de los contenedores, por lo tanto, hay una sola fuente de datos, lo que significa que únicamente necesitamos definir el rendimiento de un tipo de fuente.

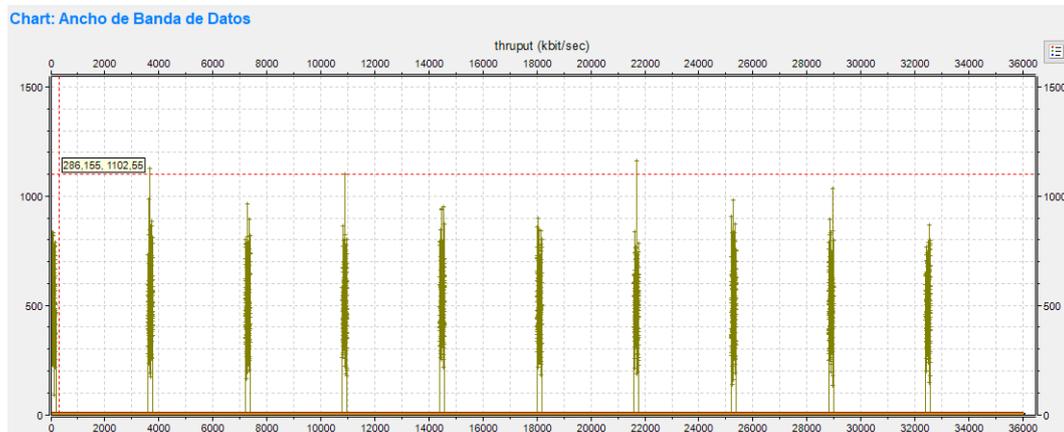


Fig. 13. Rendimiento de los sensores de gestión de residuos

La Figura 13, muestra el rendimiento de 3154 sensores en total después de ejecutar la simulación durante 30 minutos aproximadamente. Definimos que el rendimiento está en un rango que va desde 200 Kbps hasta 1,1 Mbps teniendo una máxima concentración en el rango de los 250 Kbps a 800 Kbps y la duración entre dos apariciones de tráfico es de 1 hora aproximadamente. La fuente de datos de gestión de residuos está clasificada como clase “plata” porque no es sensitivo todo el tiempo.

IV.1.3. ALUMBRADO INTELIGENTE

Este servicio puede ser cubierto solo con un tipo de sensor, mismo que debe ser para control inteligente de iluminación, 2000 sensores son usados para este servicio. La iluminación inteligente está clasificada dentro de la clase “oro” acorde a sus propiedades en tiempo real. Debido a que hay un solo tipo de sensor, el origen de este servicio es también una sola fuente. A continuación, en la Fig. 14 se muestra el desempeño de esta fuente de datos.

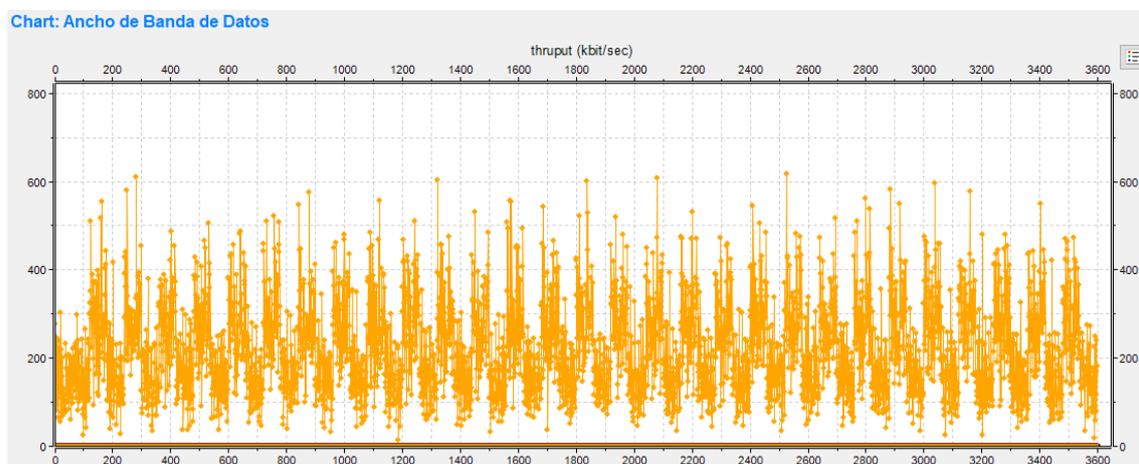


Fig. 14. Rendimiento del sensor fuente de luz

Después de simular este escenario, obtuvimos su rendimiento como se muestra en la Fig.14, en donde determinamos que el rendimiento del sensor de iluminación está en el rango de los 50 Kbps a 600 Kbps, concentrándose en su mayoría desde 100 Kbps hasta 400 Kbps.

IV.1.4. GESTIÓN EFICIENTE DE PLAZAS DE APARCAMIENTO

Este servicio también puede ser cubierto con el empleo de un solo tipo de sensor que detectará la presencia de vehículos. Este sensor debe ser dividido y configurado en tres patrones para este servicio, por lo tanto, en total hay tres tipos de fuentes de datos aplicados en este servicio, los cuales son: aparcamientos para personas con discapacidad, para carga y descarga y para taxis. Dado que estas aplicaciones son del mismo tipo, las hemos clasificado juntas dentro de la misma clase “plata”, esto es porque el sensor no requiere ser sensible en el tiempo. A continuación, se presentan las propiedades del rendimiento para estos tres tipos de fuentes.

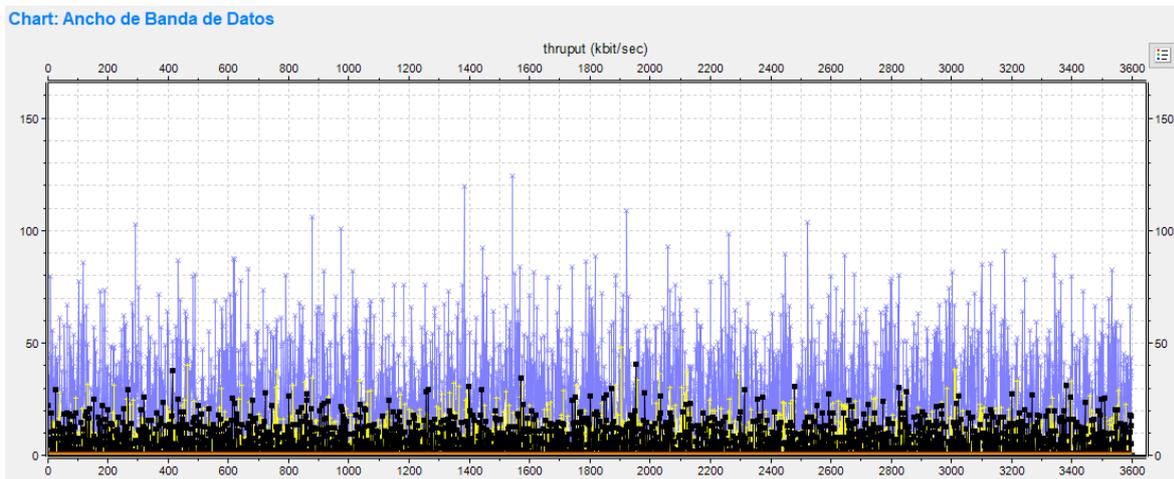


Fig. 15. Rendimiento de los tres tipos de sensores de aparcamiento

La Figura 15 muestra el rendimiento de tres fuentes de datos por separado. El color azul muestra el rendimiento de los 425 sensores de aparcamiento para personas con discapacidad, el color amarillo corresponde a los 125 sensores de aparcamiento de carga y descarga y el color negro corresponde a los 126 sensores de las paradas para taxis. De la misma Figura 15, definimos que el rendimiento para el aparcamiento para personas discapacitadas va desde los 0 Kbps hasta los 100 Kbps (con algunos picos superiores) y para los otros dos tipos de sensores va desde los 0 Kbps a los 35 Kbps que son mucho menores a los de aparcamientos de las personas con discapacidad.

IV.1.5. GESTIÓN DE RUIDO URBANO

Este servicio requiere solo de un tipo de sensor por lo tanto tenemos una sola fuente. Hemos clasificado dentro de la clase “plata” porque que no es sensible al tiempo. En la Fig. 16 se puede observar el rendimiento de 500 sensores de nivel de ruido urbano juntos.

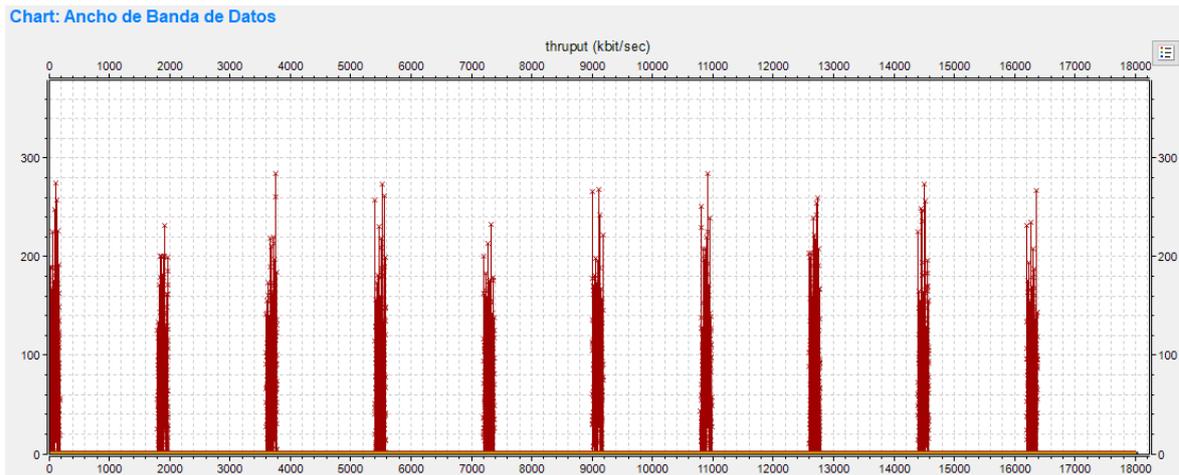


Fig. 16. Rendimiento de 500 sensores de ruido urbano

La Figura. 16 muestra el resultado del rendimiento de este escenario que va en el rango de los 0 Kbps a los 280 Kbps, centrándose principalmente bajo el rango de los 200 Kbps. El tiempo de duración entre los tráficoes es de media hora aproximadamente.

IV.1.6. MONITORIZACIÓN AMBIENTAL URBANA

Este servicio requiere de dos tipos de sensores, uno para la monitorización de los gases y otro para el monitoreo de la temperatura y humedad (éste último realizado por un solo sensor), por lo que tendremos dos fuentes de datos. Las dos fuentes las hemos clasificado como clase “plata”. En la Fig. 17 se presenta el rendimiento o desempeño de los 40 sensores de temperatura – humedad y de gases juntos.

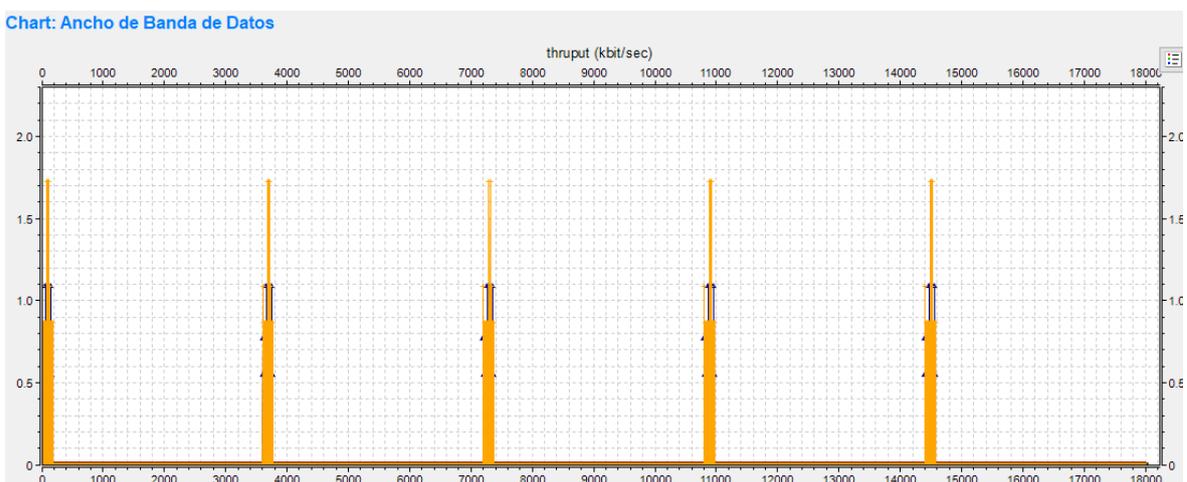


Fig. 17. Rendimiento de 40 sensores de monitorización ambiental urbana

La Figura 17 muestra el resultado del rendimiento de este servicio, mismo que va en el rango de los 0 Kbps hasta los 1,73 Kbps teniendo su mayor concentración de los 0 kbps hasta los 0,9 Kbps. El tiempo de duración entre los tráficoes es de 1 hora aproximadamente.

IV.2. SIMULACIÓN CONSOLIDADA DE TODOS LOS SERVICIOS

En esta sección, simularemos todos los servicios de manera conjunta. El trabajo incluido en esta parte puede ser dividido en dos segmentos: el primero es probar los valores de los mecanismos de QoS para diferentes clases, en el segundo es probar los diferentes anchos de banda de los enlaces de salida para todos los servicios juntos. La Tabla 16 muestra la estructura de esta simulación, la cual se ejecutó en cada condición durante 3 horas aproximadamente.

1) Prueba de diferentes valores de CIR

En este proyecto, se asignó tres clases en total para dividir todos los servicios que deberían ser incluidos en una Smart City, pero en base a los seis servicios tratados en este proyecto, ninguno es una fuente de voz que siempre vayan a estar clasificado como clase “multimedia”. Por lo tanto, es necesario definir los valores del CIR para dos clases: “oro” y “plata”. La Tabla 16 muestra cada clase de servicio y su rango de rendimiento.

Servicios	Gestión del patrimonio (video)	Gestión del patrimonio (datos)	Gestión de residuos urbanos	Gestión de Iluminación	Gestión de plazas de aparcamiento	Gestión del ruido urbano	Monitorización ambiental urbana
Clase	oro	plata	plata	oro	plata	plata	plata
Max throughput	48 Mbps	1,1 Kbps	1,1 Mbps	0,4 Mbps	0,12 Mbps	0,28 Mbps	1,7 Kbps

Tabla 16: Asignación de clases y rendimiento

De las simulaciones realizadas con cada fuente, se encontró que el rendimiento de las fuentes “oro” se encuentra bajo los 48 Mbps y el rendimiento de las fuentes “plata” está bajo los 1,1 Mbps. Entonces establecimos las siguientes tres condiciones para probar los valores apropiados de CIR y PIR para las dos clases. Las condiciones pueden ser clasificadas en tres estados: 1) PIR y CIR son ambos muy pequeños. 2) PIR es lo suficientemente grande, pero CIR es muy pequeño. 3) PIR y CIR, son a la vez lo suficientemente grande para una buena comunicación. Se probó las dos clases juntas en las tres condiciones, los valores empleados en estas condiciones se presentan en la Tabla 17.

Condiciones	Clase "Oro"				Clase "Plata"				Paquetes perdidos
	CIR (Mbps)	CBS (MiB)	PIR (Mbps)	PBS (MiB)	CIR (Mbps)	CBS (MiB)	PIR (Mbps)	PBS (MiB)	
CIR < PIR < Max throughput	0,5	0,0625	10	1,25	0,05	0,00625	0,1	0,0125	22056080 (video) 12500 (residuos)
CIR < Max throughput < PIR	1	0,125	50	6,25	0,1	0,0125	30	3,75	2024400 (video)
Max throughput < CIR < PIR	100	7,5	200	12,5	80	5	200	12,5	0

Tabla 17: Parámetros configurables bajo diferentes condiciones

De los resultados mostrados en la Tabla 17, se encontró que la pérdida de paquetes (video y datos) ocurrió con gran notoriedad bajo la condición “CIR < PIR < Max throughput”. Con el fin de conocer

futuras comparaciones, en la Fig. 18 se muestran los resultados del rendimiento (throughput), retardos y jitter para las dos clases bajo las tres diferentes condiciones.

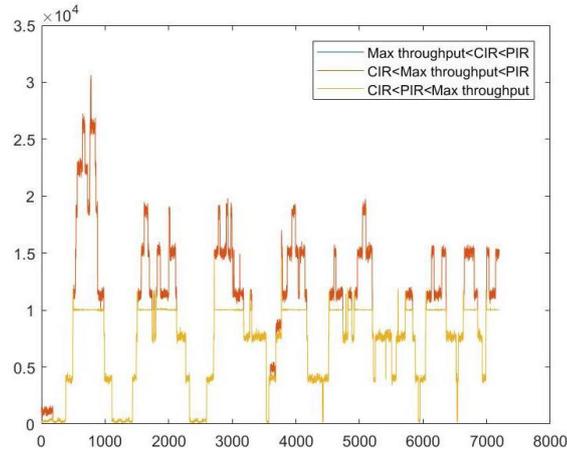


Fig. 18. Rendimiento total de la red bajo las tres condiciones

La Figura 18 muestra el desempeño o rendimiento de todas las fuentes bajo las tres condiciones establecidas donde se puede ver que bajo la condición “CIR < PIR < Max throughput”, el desempeño (throughput) está limitado alrededor de los 10 Mbps, lo cual es casi igual a la suma del PIR en las dos clases. Bajo las condiciones “CIR < Max throughput < PIR” y “Max throughput < CIR < PIR”, el rendimiento es casi el mismo, está en un rango inferior a los 32 Mbps.

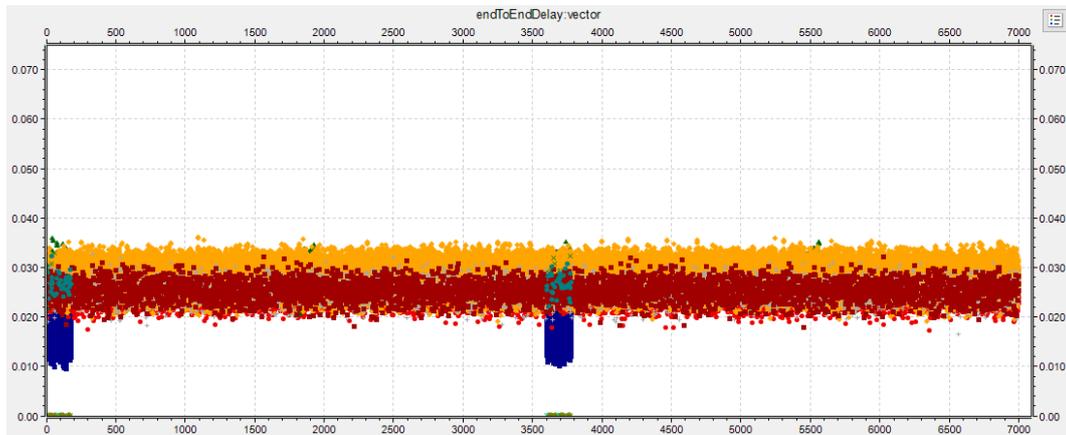


Fig. 19. Retardo de datos bajo “Max throughput < CIR < PIR”

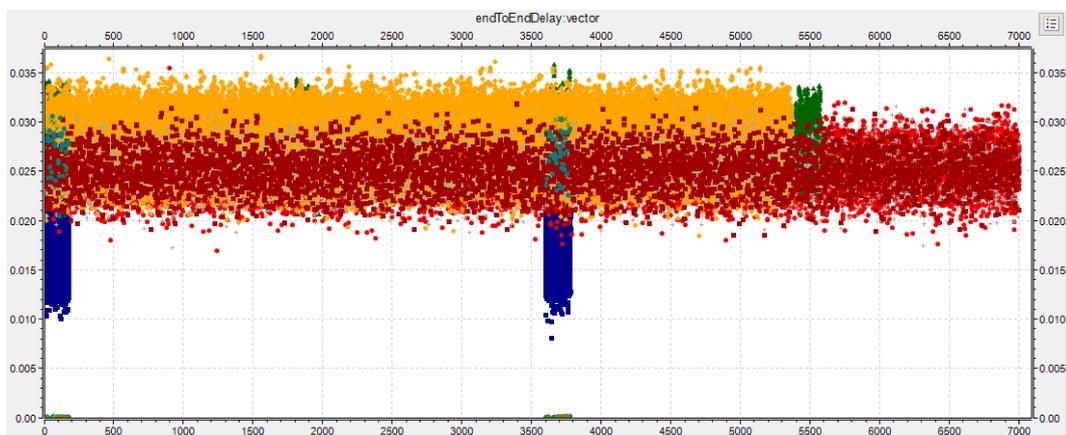


Fig. 20. Retardo de datos bajo “CIR < Max throughput < PIR”

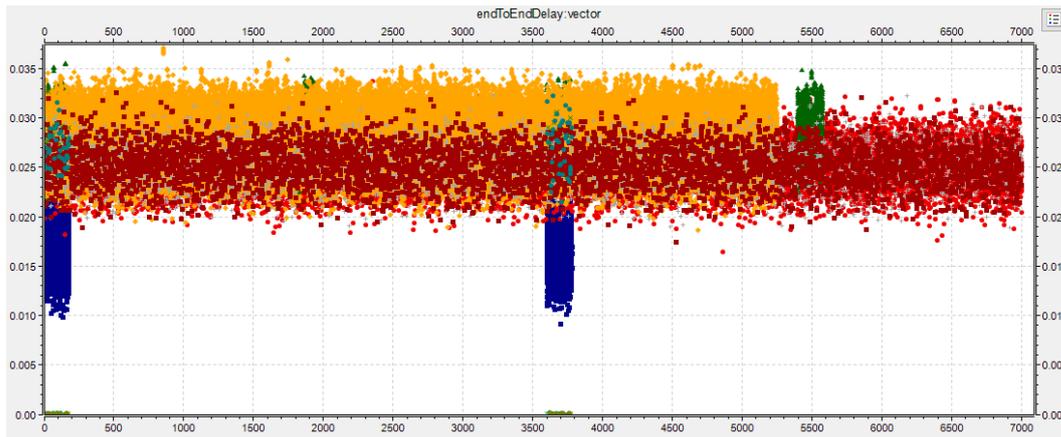


Fig. 21. Retardo de datos bajo “CIR < PIR < Max throughput”

Las Figuras 19, 20 y 21, muestran el retardo de todas las fuentes de datos bajo las tres diferentes condiciones. Los valores para las tres imágenes se concentraron bajo 34 ms, algunos exceden, pero son inferiores a 36 ms. En la Fig. 20 la distribución es similar pero un poco mayor que la de la Fig. 19 aún dentro del rango bajo 34 ms. En la Fig. 21 el valor es ligeramente inferior a los 34 ms para todos los puntos, aunque, la distribución se encuentra todavía de forma similar.

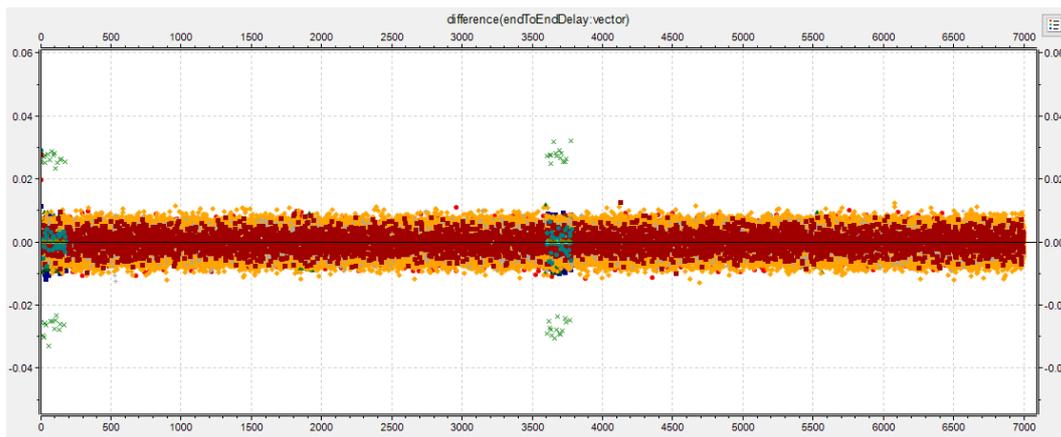


Fig. 22. Jitter de fuentes de datos bajo “Max throughput < CIR < PIR”

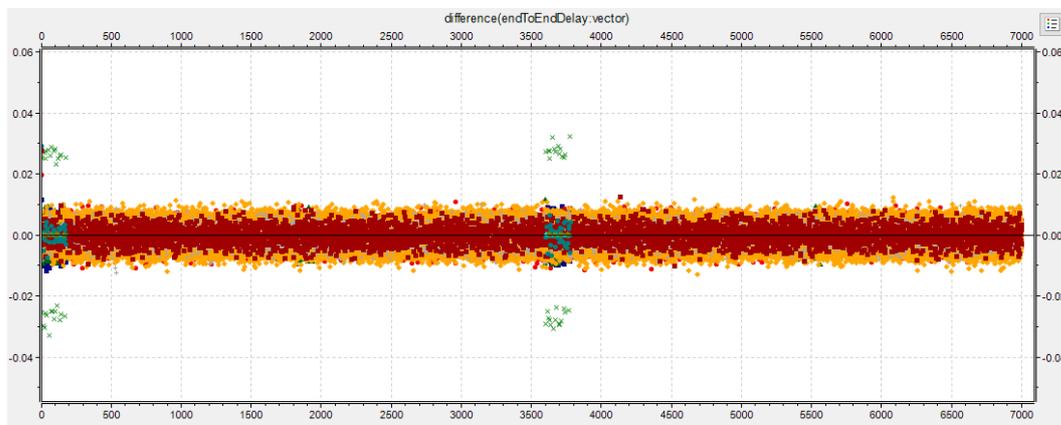


Fig. 23. Jitter de fuentes de datos bajo “CIR < Max throughput < PIR”

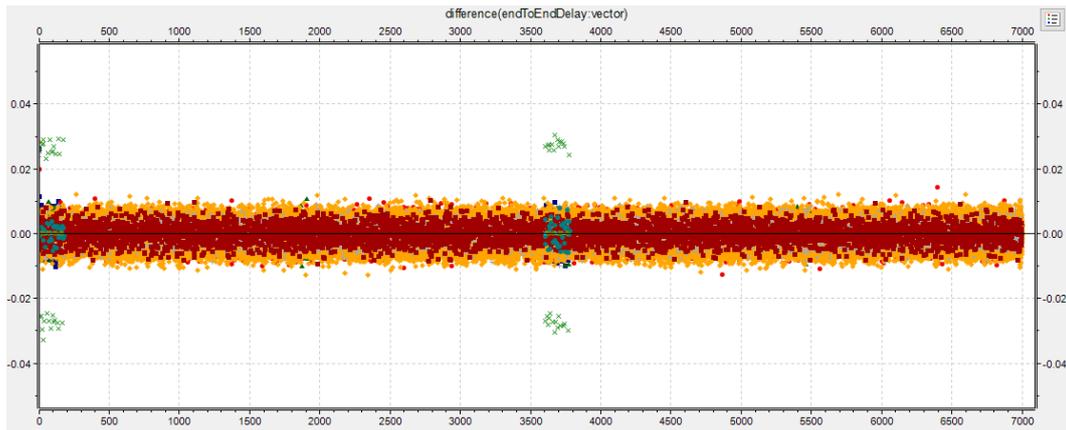


Fig. 24. Jitter de fuentes de datos bajo “CIR<PIR<Max throughput”

Las Figuras 22, 23 y 24, muestran el jitter de todas las fuentes de datos bajo las tres condiciones mencionadas. En las Figs. 22 y 23, el valor está concentrado aproximadamente en el rango de $\pm 0,01$ s, aunque se tienen algunos picos abruptos que llegan hasta los $\pm 0,03$ s con intervalos de cada hora. Entre las dos primeras figuras prácticamente no existe diferencias. En la Fig. 24, los valores son ligeramente inferiores al rango $\pm 0,01$ s, y también existe la presencia de algunos picos que superan este rango llegando hasta los $\pm 0,03$ s en las partes superior e inferior.

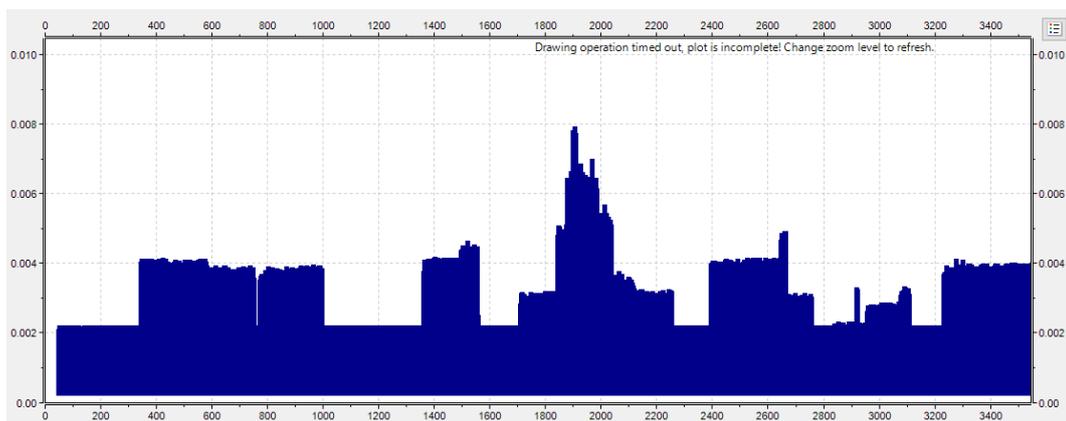


Fig. 25a) Retardo de video bajo “Max throughput < CIR < PIR”

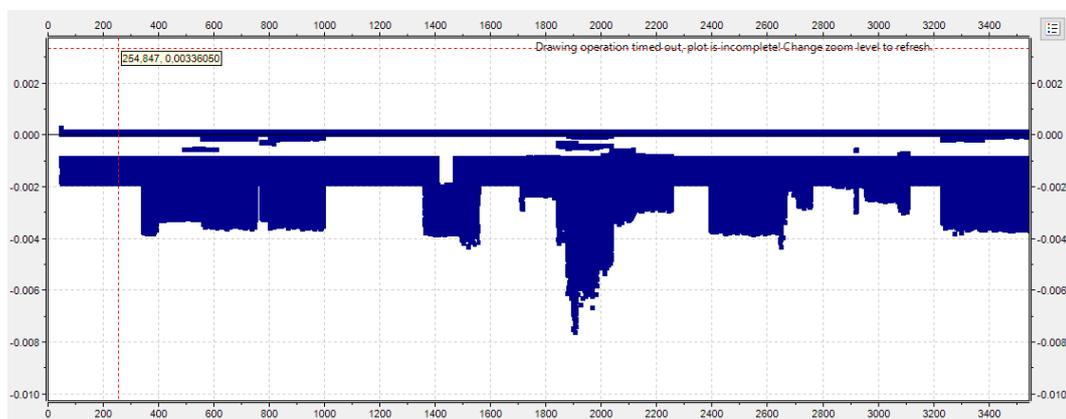


Fig. 25b) Jitter de video bajo “Max throughput < CIR < PIR”

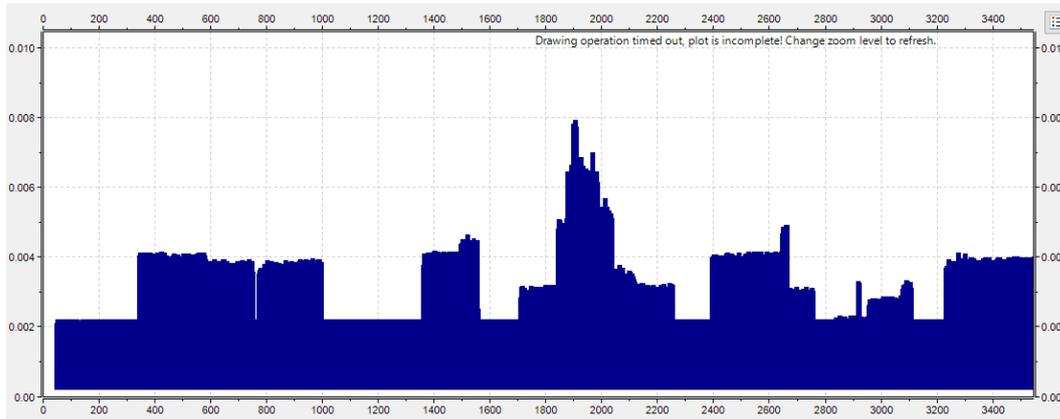


Fig. 26a). Retardo de video bajo “CIR< Max throughput<PIR”

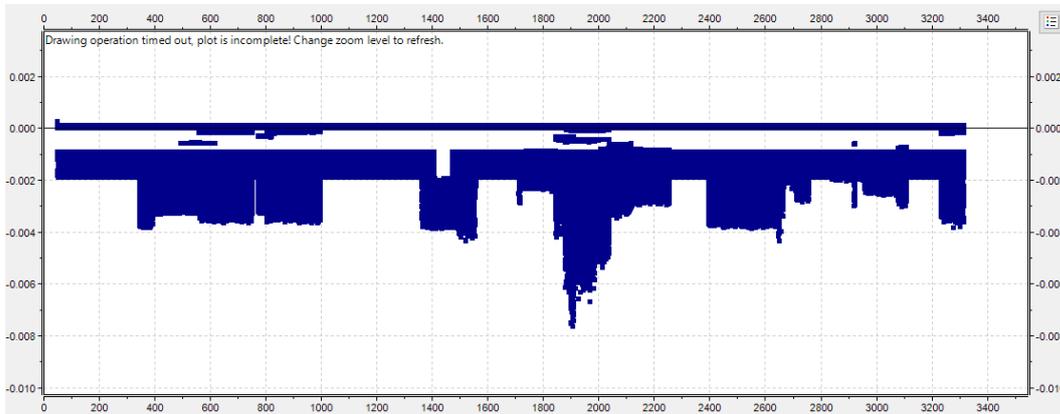


Fig. 26b). Jitter de video bajo “CIR< Max throughput<PIR”

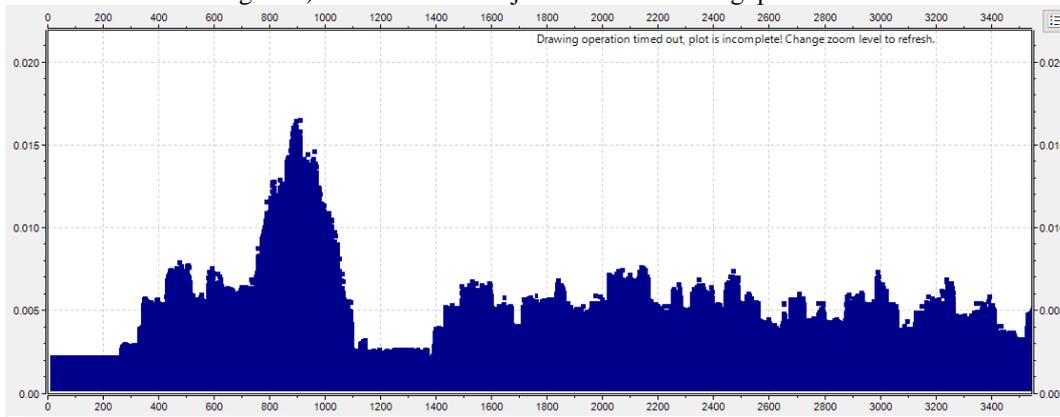


Fig. 27a). Retardo de video bajo “CIR<PIR< Max throughput”

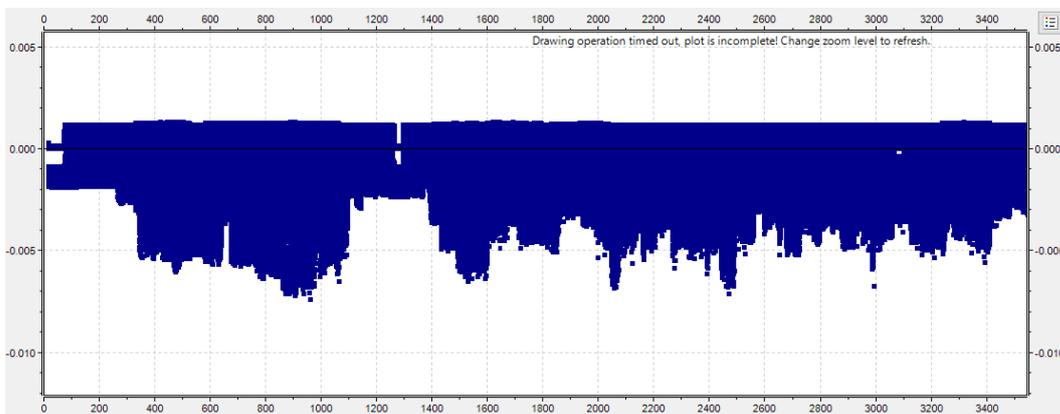


Fig. 27b). Jitter de video bajo “CIR<PIR< Max throughput”

En las gráficas presentadas en las Figs 25, 26 y 27, se muestran el retardo y el jitter de todas las fuentes bajo las tres condiciones. De las Fig. 25a) y 26a), el valor del retardo en su mayoría está concentrado por debajo de los 4ms, aunque se tiene algunos picos que pueden llegar hasta cerca de los 8ms inclusive, y en las Figs. 25b) y 26b) el valor del jitter su mayoría se concentra entre - 4 ms (hay algunos picos que parten de - 7ms) a 0,25ms. Los valores de las Figs. 25 y 26 son bastante similares, prácticamente son coincidentes. En la Fig. 27a), los valores del retardo son similares, en su mayoría se concentran bajo los 6ms, pero con picos que pueden llegar hasta los 16ms. Los valores del jitter mostrados en la Fig. 27b), están entre en un rango desde los -5ms (con picos desde los - 7ms) a 1ms, valores que están en un rango mayor que de las Figs. 25 y 26.

De todas las pruebas, el resultado bajo la condición “Max throughput < CIR < PIR” es el que tiene el mejor rendimiento que los otros dos por lo que definimos el valor del CIR y PIR como esta condición.

2) Pruebas de Diferentes Anchos de Banda

En esta sección, el objetivo es encontrar y proponer el ancho de banda adecuado para el trabajo de la red interna del Ayuntamiento con los seis servicios juntos. Se ejecutó la simulación por 4 horas aproximadamente en los tres diferentes anchos de banda: 10 Mbps, 100 Mbps y 1 Gbps. 10 Mbps es notablemente menor que el requerimiento para cada servicio por separado, 100 Mbps puede ser suficiente para cada servicio, pero es inferior al total de ancho de banda asignado, finalmente, 1 Gbps es más grande que la suma de todos los anchos de banda.

Por lo tanto, los rendimientos (throughput) bajo el ancho de banda de 100 Mbps y 1 Gbps son casi los mismos, no hay pérdidas de paquetes bajo estos dos anchos de banda, sin embargo, bajo 10 Mbps, el throughput está forzosamente limitado a los 10 Mbps disponibles por lo que los valores picos que excedan este rango, serán eliminados teniendo pérdidas de paquetes de video y de datos lo cual indica que 10 Mbps no sería la selección apropiada.

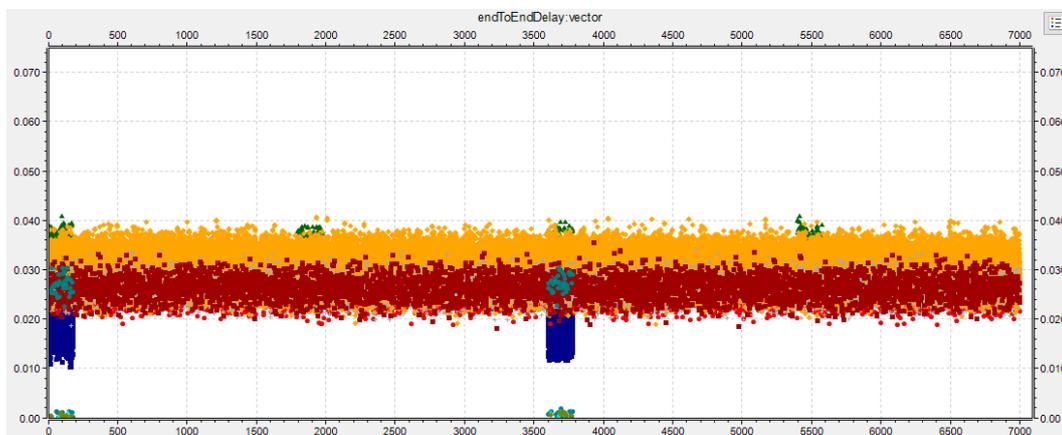


Fig. 29. Retardo de la fuente de datos bajo 10 Mbps

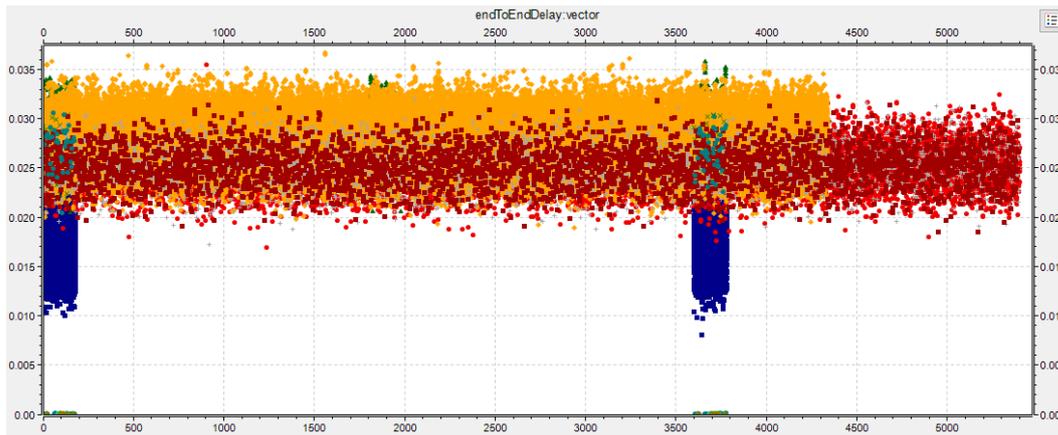


Fig. 30. Retardo de la fuente de datos bajo 100 Mbps

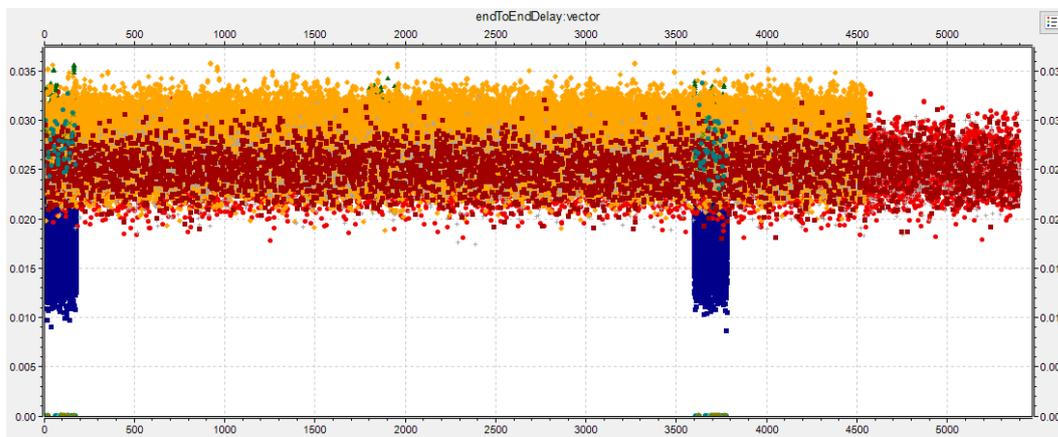


Fig. 31. Retardo de la fuente de datos bajo 1 Gbps

Las Figuras 29, 30 y 31 muestran el retardo de todas las fuentes de datos bajo los tres anchos de banda: 10 Mbps, 100 Mbps y 1 Gbps. El retardo en la Fig. 29 es de 0s a 40ms con una concentración entre los 22ms y los 38ms, valores que son altos. En las Fig. 30 y 31 los resultados son muy similares, los puntos van desde 0s a 34ms con una mayor concentración entre los 21ms hasta los 33ms con unos picos que parten desde los 11ms aproximadamente. De lo revisado el retardo bajo los 10 Mbps es mucho mayor que los otros dos anchos de banda en todos los servicios. Después, comparando el retardo de las Figs. 30 y 31, encontramos que el valor de la última es ligeramente menor que la de la que le antecede, aunque la diferencia es menor de 1ms.

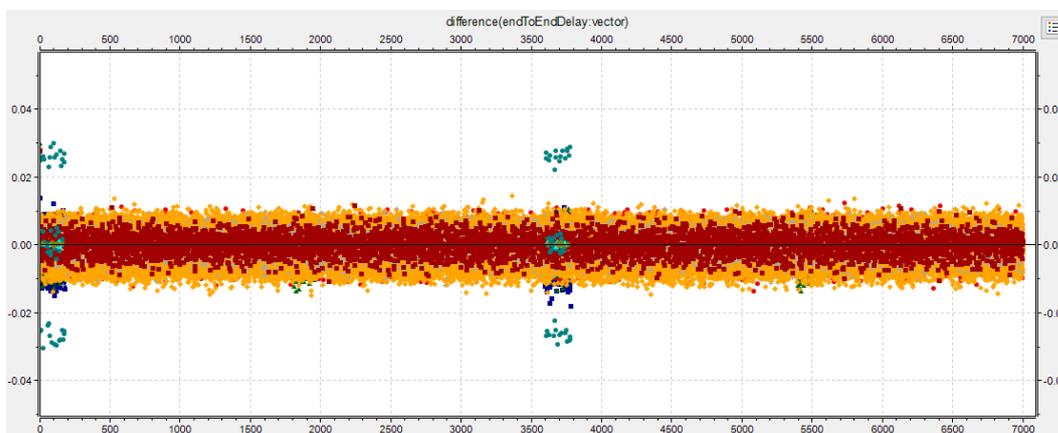


Fig. 32. Jitter de las fuentes de datos bajo 10 Mbps

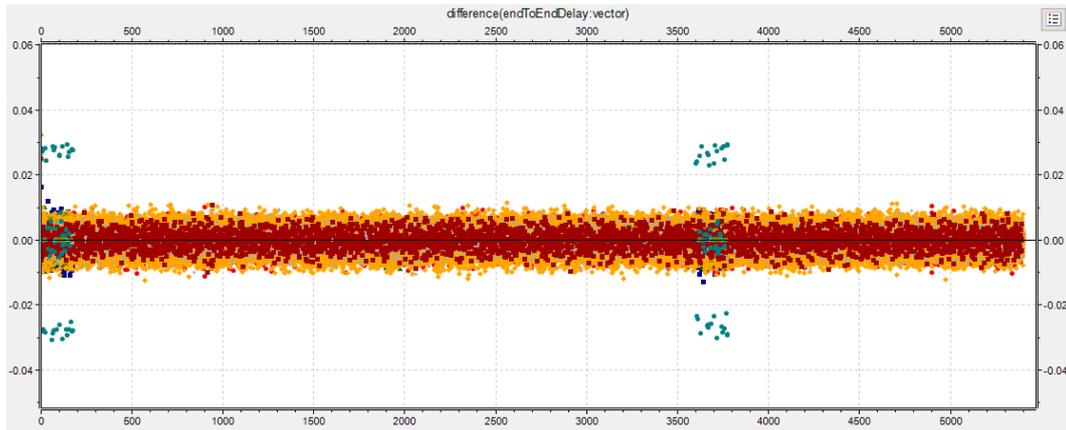


Fig. 33. Jitter de las fuentes de datos bajo 100 Mbps

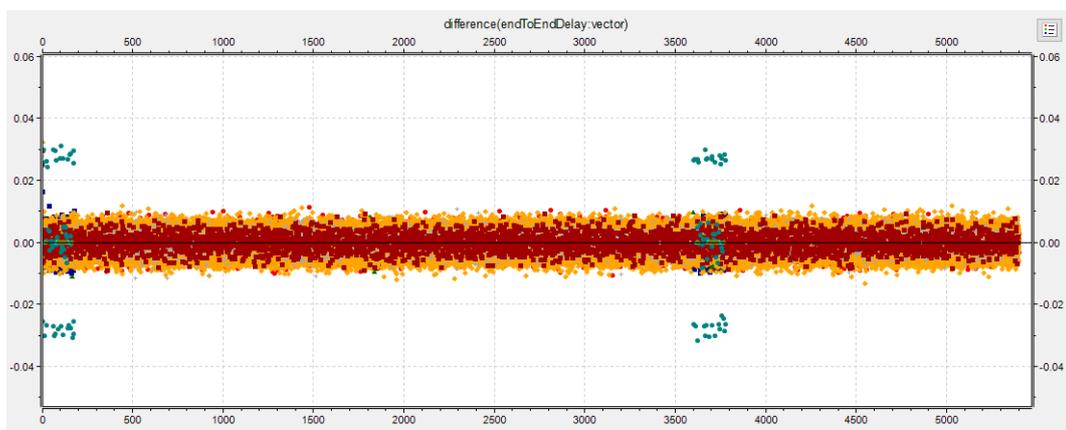


Fig. 34. Jitter de las fuentes de datos bajo 1 Gbps

Las tres figuras arriba mostradas son el jitter de las fuentes de datos bajo los tres anchos de banda, tienen una relación muy parecida al comportamiento de los retardos. El jitter bajo los 10 Mbps es de $-0,014s$ a $0,01s$; bajo los 100 Mbps es de $\pm 0.01s$, sin embargo, bajo 1 Gbps es más pequeño, va desde $-0,09s$ a $0,08s$ aproximadamente. Los resultados del jitter son similares a los de los retardos, bajo 10 Mbps el jitter es mayor, bajo los 100 Mbps el jitter disminuye y bajo 1 Gbps el jitter es el menor de las tres condiciones.

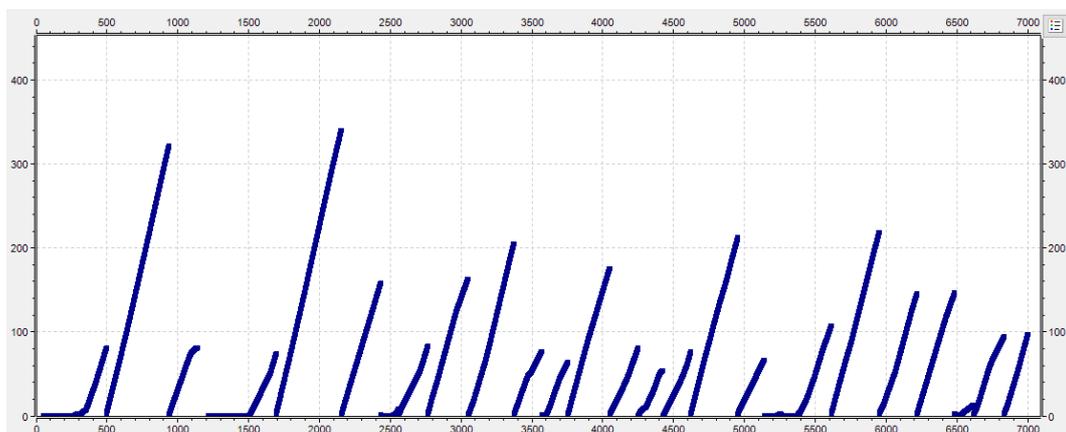


Fig. 35a). Retardo de las fuentes de video bajo 10 Mbps

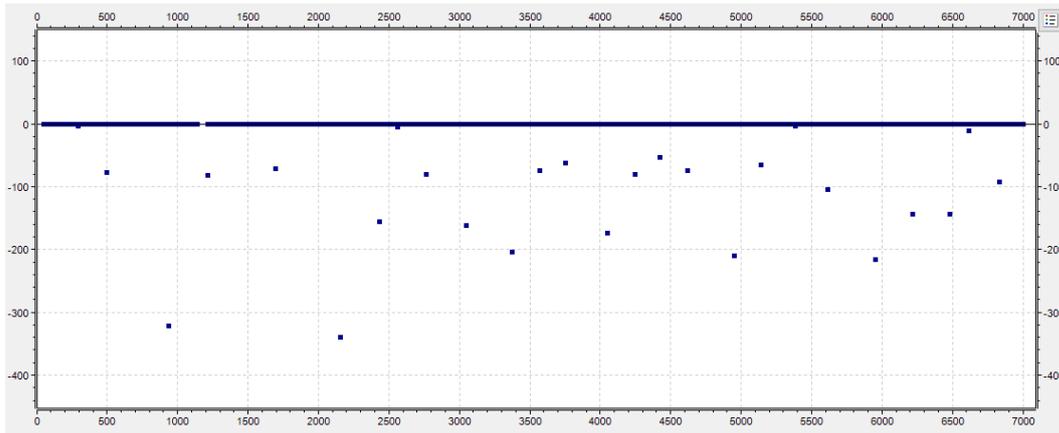


Fig. 35b). Jitter de las fuentes de video bajo 10 Mbps

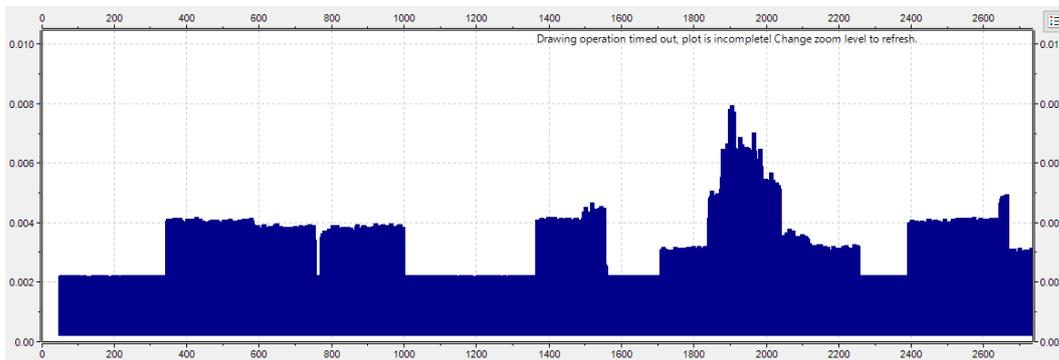


Fig. 36a). Retardo de las fuentes de video bajo 100 Mbps

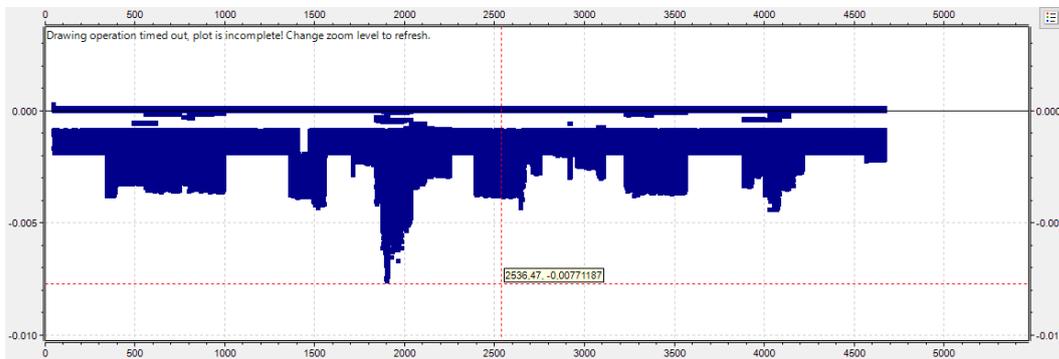


Fig. 36b). Jitter de las fuentes de video bajo 100 Mbps

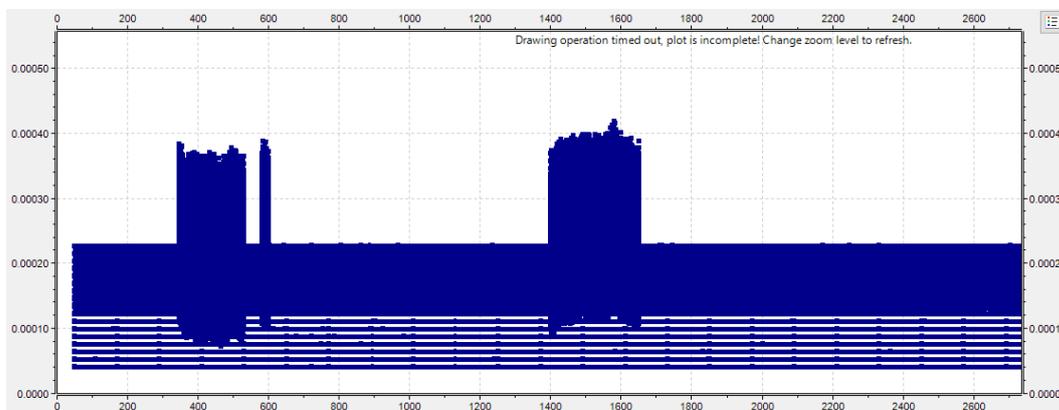


Fig. 37a). Retardo de las fuentes de video bajo 1 Gbps

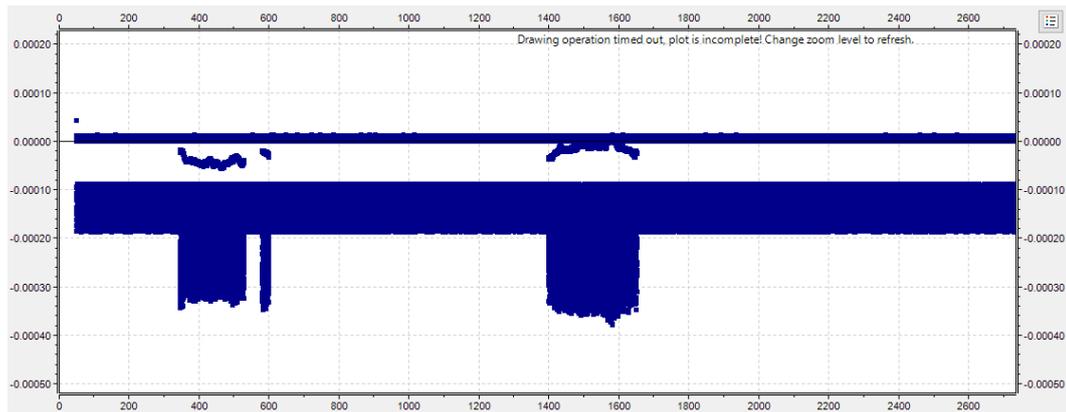


Fig. 37b). Jitter de las fuentes de video bajo 1 Gbps

Las 6 figuras anteriores muestran el retardo y el jitter de las fuentes de video bajo las tres condiciones dadas. El valor del retardo y jitter bajo 10 Mbps son claramente muy grandes, el valor del retardo puede alcanzar los 340s y del jitter puede alcanzar los -340s. Bajo los 100 Mbps, el retardo va de los 0s a los 0,008s (8ms); el jitter va de los -0,007s (-7ms) a los 0,0003s (0.3ms). Bajo el 1 Gbps, el retardo va desde los 0,00004s (0,04ms) a los 0,00038s (0,38ms) con algunos picos que pueden llegar hasta los 0,40ms; el jitter va de los -0,00036s (-0,36ms) a los 0,00002s (0,02ms), todos estos valores son menores que el de 100 Mbps y mucho menores que los de 10 Mbps.

De todos los resultados anteriores, podemos asegurar en primera instancia que los 10 Mbps no son suficientes para la red del ayuntamiento. Para los 100 Mbps y 1 Gbps, la diferencia del retardo de datos entre ellos es menor que 0,001s y la diferencia del retardo de video es menor que 0,007s. Para el jitter la diferencia es inclusive más pequeña. Acorde con el rendimiento (throughput) su máximo valor está en el orden de los 31 Mbps. Con el propósito de ahorrar recursos de ancho de banda, pensamos que 100 Mbps es una mejor opción que 1 Gbps.

V. CONCLUSIONES Y TRABAJO FUTURO

De las pruebas de simulación realizadas, establecimos algunas conclusiones acerca de los mecanismos de QoS y los anchos de banda. Los mecanismos de QoS y la velocidad del enlace de salida tienen influencia directa en el rendimiento y calidad de servicio de la red.

- a. Un bajo PIR influye en el rendimiento de las fuentes causando pérdidas de paquetes, pero también causa bajos retardos y altos tiempos de jitter para cada fuente. Esto es porque el PIR es aplicado para limitar el máximo flujo de una fuente en una unidad de tiempo, muchos paquetes son descartados directamente en lugar de que permanezcan en la cola cuando el PIR no es lo suficientemente grande, entonces el tiempo de espera decrece causando más diferencias entre tiempos de espera.
- b. Un bajo CIR influye en el retardo y el jitter de la comunicación. Un bajo CIR causa una cola extensa y permite que los paquetes esperen más tiempo, pero si el tamaño del paquete o el número

de fuentes son demasiado pequeños, la influencia de un bajo CIR es muy pequeño, inclusive casi no ejerce ninguna influencia.

- c. Un bajo ancho de banda causará un gran retardo y jitter solo si es apenas mayor que el rendimiento de las fuentes. A mayor ancho de banda de los servicios, menor retardo y jitter en la calidad de los servicios, pero si el ancho de banda es menor que el rendimiento de las fuentes, irrevocablemente causará pérdida de paquetes.
- d. Al agregar múltiples fuentes, el ancho de banda es compartido para todas las fuentes juntas, por lo tanto, el rendimiento en el enlace de salida es menor que la suma de todos los picos máximos de rendimiento en cada fuente, además, esto es debido a un mayor tiempo de simulación donde los sensores tienen mayores tiempos disponibles y en consecuencia pueden recolectar datos durante tiempos más largos para evitar picos de rendimiento que puedan desaprovechar el ancho de banda.
- e. Con las pruebas realizadas, podemos determinar que un ancho de banda de 10 Mbps no es lo suficiente para cubrir los requerimientos de red asociados a los servicios de la ciudad de Valencia como ciudad inteligente. Los anchos de banda de 100 Mbps y 1 Gbps son las mejores opciones, pero teniendo en cuenta que la diferencia del retardo y jitter entre estas dos opciones es mínima y que el rendimiento (throughput) está por el orden de los 35 Mbps y con el propósito de ahorrar recursos en todos sus aspectos, pensamos que 100 Mbps es una mejor opción.

Las siguientes fases de este proyecto podrían incluir la definición de más servicios que deberían ser incluidos en una ciudad inteligente sostenible de tal manera que las simulaciones sean la más cercanas posibles a la vida real de una ciudad inteligente antes de su implementación.

AGRADECIMIENTOS

- A Dios, a mi tía Mariana, a mis padrinos Bárbara y Ulrich Rohn, a mis padres Mercedes y Neptalí, a mi familia, a Peter Van Dijk y otros amigos, quienes estuvieron presentes en los momentos cruciales de mi vida, prestos a brindarme su apoyo en sus diferentes tipos y formas con el fin de que pueda alcanzar este objetivo mientras sobrellevaba difíciles momentos de mi vida.
- A España por medio de la UPV por abrirme sus puertas y brindarme la oportunidad de estudiar un postgrado que me permite crecer a nivel profesional y personal.
- A mi tutor, Miguel Ángel Rodríguez por sus enseñanzas, guía, soporte y entereza para el desarrollo del presente proyecto.

REFERENCIAS

[A] Smart city concept with different icon and elements; Disponible en: https://www.123rf.com/photo_52445942_stock-vector-smart-city-concept-with-different-icon-and-elements-modern-city-design-with-future-technology-for-li.html

- [B] Enabling Technologies for Effective Deployment of Internet of Things (IoT) Systems; Disponible en: <https://telsoc.org/ajtde/2014-12-v2-n4/a65>
- [1] Ciudades Inteligentes Disponibles - Tecnologías de Información y Comunicación (TIC) [online] Disponible en: <https://www.itu.int/en/ITU-T/ssc/united/Pages/default.aspx>
- [2] Gómez-Sacristan, A., Rodríguez-Hernández, M., and Sempere, V. (2015). Evaluation of Quality of Service in Smart-Hospital Communications. *Journal of Medical Imaging and Health informatics*, pp 1-7
- [3] Jiménez Peralta, F., Rodríguez-Hernández, M., (2017) Simulación de una Red de Comunicaciones para el Control de Radiaciones en Aplicaciones Médicas.
- [4] PPT 113-17-SP VALENCIA - IMPULSO- VLCi, Pliego de peticiones técnicas que regirán la realización del contrato de “Desarrollo de la iniciativa Impulso VLCi” / Procedimiento abierto, Exp: 113/17-SP.
- [5] Qué es OMNeT++ [oline] disponible en: <https://omnetpp.org/intro>
- [6] OMNeT++ simulación manual [online] disponible en: <https://www.omnetpp.org/doc/omnetpp/manual/#sec:introduction:what-is-omnetpp>
- [7] Heron, G. Martini, L. (2008). 'MPLS'. In: Kasim, A. *Delivering carrier ethernet*. The McGraw-Hill Companies.
- [8] Heinanen, J., Guerin, R. (1999), 'A Two Rate Three Color Marker'. *Network Working Group*. IETF RFC 2698.
- [9] Frahim, E., Froom, R., Sivasubramanian, B. (2004). *CCNP Auto-aprendizaje: Comoprensión e implementación de la Calidad de Servicio en Redes Conmutadas Multicapa CISCO*. 2da Edición USA: Cisco Press.
- [10] *Humidity and Temperature Transmitter for High-end Meteorological Applications*. Disponible en: <https://www.epluse.com/en/products/humidity-instruments/industrial-humidity-transmitters/ee33-m/>
- [11] *Model PMA1107 Analog Non-Weighted UVA+B Sensor*; Disponible en: <https://solarlight.com/product/uvauvbsensorpma1107/>
- [12] *VAISALA Air Quality Transmitter AQT400 Series for Measuring Pollution Gases and Particles*; Disponible en: <https://www.vaisala.com/sites/default/files/documents/AQT400-Series-Datasheet-B211581EN.pdf>
- [13] *Urbiotica U-Dump M2M waste management sensor*; Disponible en: <http://www.urbiotica.com/en/product/u-dump-m2m-2/>
- [14] *TMD2725 Ambient Light Sensor*; Disponible en: <https://ams.com/tmd2725>
- [15] *Sensor Smart Parking*; Disponible en: <http://www.libelium.com/products/smart-parking/>
- [16] *TA120 Noise measuring sensor for Smart Solutions*; Disponible en: <https://www.cesva.com/en/products/sensors-terminals/TA120/>
- [17] Evans, J., Filsfils, C. (2007). 'Introduction to QoS Mechanisms and Architectures' In: *Deploying IP and MPLS QoS for multiservice networks*. USA: Elsevier Inc.
- [18] Evans, J., Filsfils, C. (2007). 'Data Plane QoS Mechanisms', pp 98-100, In: *Deploying IP MPLS QoS for multiservice networks*. USA: Elsevier Inc.
- [19] Evans, J., Filsfils, C. (2007). 'Data Plane QoS Mechanisms', pp 100-112, In: *Deploying IP MPLS QoS for multiservice networks*. USA: Elsevier Inc.
- [20] Evans, J., Filsfils, C. (2007). 'Data Plane QoS Mechanisms', pp 112-116, In: *Deploying IP MPLS QoS for multiservice networks*. USA: Elsevier Inc.
- [21] Gong, X. (2006). *Principles of Broadband Communication Network*. CN: Beijing University of Posts and Telecommunications Press.