

# **Modelado de Tráfico para Servicios de una Comunidad Académica Virtual sobre IPTV**



**Fabian Fernando Riascos Guerrero  
Andrés Jordano Arévalo Muñoz**

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**

**Departamento de Telemática**

**Línea de Investigación en Servicios Avanzados de Telecomunicaciones**

Popayán, Octubre de 2013

# **Modelado de Tráfico para Servicios de una Comunidad Académica Virtual sobre IPTV**



**Fabian Fernando Riascos Guerrero  
Andrés Jordano Arévalo Muñoz**

**Monografía presentada como requisito para optar por el título de Ingeniero  
en Electrónica y Telecomunicaciones**

Director: M.Sc. Wilmar Yesid Campo Muñoz  
Co-Director: PhD. Jose Luis Arciniegas Herrera

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**

**Departamento de Telemática**

**Línea de Investigación en Servicios Avanzados de Telecomunicaciones**

Popayán, Octubre de 2013

## Tabla de Contenido

	Pág.
<b>CAPITULO 1</b> .....	1
<b>INTRODUCCIÓN</b> .....	1
1.1    CONTEXTO GENERAL .....	1
1.2    ESCENARIO DE MOTIVACIÓN .....	3
1.3    OBJETIVOS.....	3
1.3.1    Objetivo general .....	3
1.3.2    Objetivos específicos.....	3
1.4    ESTRUCTURA DE LA MONOGRAFÍA .....	4
<b>CAPITULO 2</b> .....	5
<b>ESTADO DEL ARTE</b> .....	5
2.1    CONCEPTOS FUNDAMENTALES.....	5
2.1.1    e-Learning.....	5
2.1.2    Comunidad Académica Virtual (CAV).....	6
2.1.2.1    Definición de Comunidad Virtual.....	7
2.1.2.2    Características de una Comunidad Virtual .....	7
2.1.3    Televisión Digital Interactiva (TDi) .....	8
2.1.3.1    Aplicaciones TDi.....	9
2.1.3.2    Contenidos multimedia de TDi .....	9
2.1.3.3    Canal de retorno y canal de interactividad .....	9
2.1.4    Internet Protocol Television (IPTV).....	10
2.1.4.1    Características de IPTV.....	11
2.1.4.2    Calidad de Servicio (QoS) .....	11
2.1.4.3    Calidad de Experiencia (QoE) .....	12
2.1.4.4    Estándares, Arquitectura y Operación de IPTV .....	12
2.1.4.5    MPEG-2 [43] .....	16
2.1.4.6    Multicast.....	17
2.1.5    Modelado de Tráfico .....	18
2.2    TECNOLOGÍAS RELACIONADAS .....	19
2.2.1    LIVE555.....	19

2.2.2	FFMPEG .....	20
2.2.3	OpenRTSP .....	21
2.2.4	VLC Media Player .....	21
2.2.5	Wireshark .....	21
2.2.5	OPNET Modeler .....	21
2.2.6	ns-3.....	22
2.2.7	@RISK [56].....	22
2.3	TRABAJOS RELACIONADOS .....	23
2.3.1	On the Self-similar Nature of Ethernet Traffic [58].....	23
2.3.2	Long-Range Dependence in Variable-Bit-Rate Video Traffic [59].....	23
2.3.3	Experimental Queueing Analysis with Long-Range Dependent Packet Traffic [60] .....	24
2.3.4	Self-similar Traffic Parameter Estimation: A Semi-parametric Periodogram-Based Algorithm [61].....	24
2.3.5	Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level [62].....	24
2.3.6	Self-Similar Traffic and Network Dynamics [63] .....	25
2.3.7	Queueing performance comparison of traffic models for Internet traffic [64] .....	25
2.3.8	MPEG-encoded variable bit-rate video traffic modelling [65] .....	25
2.3.9	Modeling Internet Traffic: Performance Limits [66] .....	25
2.3.10	Modeling of MPEG-4 Video Traffic Using a Multifractal Cascade with Autoregressive Multipliers [67].....	26
2.3.11	Multi-Service traffic models of heterogeneous wireless communication networks [68] ..	26
2.3.12	Using Hidden Markov Chains for Modeling P2P-TV Traffic [69].....	26
2.3.13	Modeling Video on Demand Services taking into account statistical dependences in user behavior [70].....	26
2.3.14	Limitations of Network Emulation with Single-Machine and distributed ns-3 [71] .....	27
<b>CAPITULO 3</b>	.....	<b>28</b>
<b>ARQUITECTURA DE REFERENCIA Y CARACTERIZACIÓN DE TRÁFICO REAL</b>	.....	<b>28</b>
3.1	ARQUITECTURA DEL PROYECTO ST-CAV [72].....	28
3.2	TOPOLOGÍAS PARA CASOS DE ESTUDIO .....	30
3.2.1	Caso de estudio 1. Topología en estrella.....	30
3.2.2	Caso de estudio 2. Topología en árbol .....	31

3.2.3	Caso de estudio 3. Topología en anillo.....	32
3.3	CARACTERIZACIÓN DE TRÁFICO REAL CAPTURADO EN LABORATORIO .....	32
3.3.1	Captura de tráfico real producido por el servicio de VoD de la CAV sobre IPTV.....	32
3.3.1.1	Transmisión y reproducción de video .....	32
3.3.2	Captura de trazas de tráfico .....	36
3.3.3	Análisis del tráfico generado por el servicio de VoD .....	39
3.3.4	Caracterización del tráfico real capturado en laboratorio .....	39
3.3.4.1	Prueba de bondad de ajuste Kolmogorov-Smirnov.....	40
3.3.4.2	Análisis de las componentes de los videos.....	40
3.3.4.3	Caracterización del tráfico de servicios asociados al contenido.....	47
3.4	CUMPLIMIENTO DE OBJETIVO ESPECÍFICO No. 1.....	48
<b>CAPÍTULO 4 .....</b>		<b>49</b>
<b>MODELO DE TRÁFICO DE LOS SERVICIOS DE UNA CAV SOBRE IPTV .....</b>		<b>49</b>
4.1	DESARROLLO DEL MODELO.....	49
4.1.1	Planteamiento del problema.....	49
4.1.2	OPNET Modeler .....	50
4.1.2.1	Descripción del modelo.....	50
4.1.2.2	Validación del modelo.....	56
4.1.3	ns-3 .....	61
4.1.3.1	Descripción del modelo.....	61
4.1.3.2	Validación del modelo.....	64
4.2	CUMPLIMIENTO DEL OBJETIVO ESPECÍFICO No. 2 .....	64
<b>CAPÍTULO 5 .....</b>		<b>65</b>
<b>ANÁLISIS DE RESULTADOS.....</b>		<b>65</b>
5.1	APLICACIÓN DE LOS MODELOS DE SIMULACIÓN A LAS TOPOLOGÍAS PARA CASOS DE ESTUDIO .....	65
5.2	ANÁLISIS DE RESULTADOS .....	66
5.2.1	Topología 1.....	67
5.2.2	Topología 2 .....	70
5.2.3	Topología 3 - Escenario 1 .....	76
5.2.4	Topología 3 – Escenario 2.....	80
5.3	CUMPLIMIENTO DEL OBJETIVO ESPECÍFICO No. 3 .....	85

<b>CAPÍTULO 6</b> .....	87
<b>CONCLUSIONES Y TRABAJOS FUTUROS</b> .....	87
6.1    CONCLUSIONES .....	87
6.1.1    Capítulo 2 .....	88
6.1.2    Capítulo 3 .....	88
6.1.3    Capítulo 4 .....	89
6.1.4    Capítulo 5 .....	90
6.2    TRABAJOS FUTUROS .....	92
<b>AGRADECIMIENTOS</b> .....	94
<b>REFERENCIAS</b> .....	95

## LISTA DE TABLAS

	Pág.
<b>Tabla 3-1.</b> Funciones de distribución de probabilidad (FDP) para las componentes de cada video .....	46
<b>Tabla 3-2.</b> Tamaño de las solicitudes, respuestas y archivos (XML e Imagen) para las aplicaciones de TDi .....	47
<b>Tabla 4-1.</b> FDPs utilizadas en el modelo de OPNET Modeler con los parámetros requeridos por el simulador para poder declararlas .....	53
<b>Tabla 4-2.</b> Resumen de la caracterización realizada con @RISK a los resultados de la simulación del video1 en el modelo desarrollado en OPNET Modeler .....	60
<b>Tabla 4-3.</b> Diferencia entre los parámetros media y variación estándar de las Tablas 3-1 y 4-2 .....	60
<b>Tabla 5-1.</b> Topología1. Valores promedio aproximados del retardo para los clientes 1 y 10 .....	70
<b>Tabla 5-2.</b> Topología1. Valores promedio aproximados del retardo de colas de ejecución entre servidor y switch1, y entre switch1 y cliente1 .....	70
<b>Tabla 5-3.</b> Topología1. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y switch1, y entre switch1 y cliente1 .....	70
<b>Tabla 5-4.</b> Topología1. Valores promedio aproximados del <i>throughput</i> entre servidor y switch1, y entre switch1 y cliente1 .....	70
<b>Tabla 5-5.</b> Topología2. Valores promedio aproximados del retardo para los clientes 1, 5, 6 y 10 .....	75
<b>Tabla 5-6.</b> Topología2. Valores promedio aproximados del retardo de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y switch2 y cliente10 ...	75
<b>Tabla 5-7.</b> Topología2. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y switch2 y cliente10 ...	75
<b>Tabla 5-8.</b> Topología2. Valores promedio aproximados del <i>throughput</i> entre servidor y router1, router1 y switch1 y switch1 y cliente1 .....	75
<b>Tabla 5-9.</b> Topología3 – Escenario1. Valores promedio aproximados del retardo para los clientes 1, 5, 6 y 10 .....	79
<b>Tabla 5-10.</b> Topología3 – Escenario1. Retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10 .....	80
<b>Tabla 5-11.</b> Topología3 – Escenario1. Tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10 .....	80
<b>Tabla 5-12.</b> Topología3 – Escenario1. <i>Throughput</i> entre servidor y router1, router1 y router2 y router3, y switch1 y cliente1 .....	80
<b>Tabla 5-13.</b> Topología3 – Escenario2. Valores promedio aproximados del retardo para los clientes 1, 10, 11 y 20 .....	85
<b>Tabla 5-14.</b> Topología3 – Escenario2. Valores promedio aproximados del retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20 .....	85

**Tabla 5-15.** Topología3 – Escenario2. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20 ..... 85

**Tabla 5-16.** Topología3 – Escenario2. Valores promedio aproximados del *throughput* entre servidor y router1, router2 y router3, y switch1 y cliente1 ..... 85



## LISTA DE FIGURAS

	Pág.
<b>Figura 2-1.</b> Arquitectura básica de IPTV.....	15
<b>Figura 2-2.</b> Tamaños relativos de información para cada tipo de trama MPEG .....	17
<b>Figura 2-3.</b> IP multicast [27] .....	18
<b>Figura 3-1.</b> Arquitectura ST-CAV – Vista de despliegue [72] .....	29
<b>Figura 3-2.</b> Topología en estrella .....	31
<b>Figura 3-3.</b> Topología en árbol .....	31
<b>Figura 3-4.</b> Topología en anillo.....	32
<b>Figura 3-5.</b> Servidor LIVE555 en funcionamiento.....	33
<b>Figura 3-6.</b> Codificación de video con ffmpeg .....	34
<b>Figura 3-7.</b> Proceso de indexación con UniversalIndexer .....	35
<b>Figura 3-8.</b> Carpeta mediaServer.....	36
<b>Figura 3-9.</b> Volcado de red de VLC.....	36
<b>Figura 3-10.</b> Tráfico capturado del video1 por medio de Wireshark.....	37
<b>Figura 3-11.</b> Visualización de las diferentes tramas MPEG-2 por separado, mediante la utilización de filtros en Wireshark .....	38
<b>Figura 3-12.</b> Tráfico producido por la transmisión del video2, y sus respectivas componentes .....	39
<b>Figura 3-13.</b> Caracterización de las componentes de tráfico del video1 .....	42
<b>Figura 3-14.</b> Caracterización de las componentes de tráfico del video2 .....	44
<b>Figura 3-15.</b> Caracterización de las componentes de tráfico del video3 .....	46
<b>Figura 4-1.</b> Modelo básico de simulación a nivel de red.....	50
<b>Figura 4-2.</b> Modelo del servidor a nivel de nodo.....	51
<b>Figura 4-3.</b> Configuración del módulo pt_0 .....	51
<b>Figura 4-4.</b> Proceso asociado al módulo p_1 .....	51
<b>Figura 4-5.</b> Fragmento de código del estado init del proceso del servidor.....	52
<b>Figura 4-6.</b> Modelo del cliente a nivel de nodo .....	54
<b>Figura 4-7.</b> Proceso asociado al módulo p_0 .....	55
<b>Figura 4-8.</b> Fragmento de código del estado init del proceso del cliente.....	55
<b>Figura 4-9.</b> Resumen de la simulación para el video1 .....	57
<b>Figura 4-10.</b> Gráficas de los resultados para las tramas tipo I del video1, arrojados por OPNET Modeler en mode <i>FDP</i> .....	57
<b>Figura 4-11.</b> Caracterización de los resultados de la simulación del video1 mediante la herramienta @RISK.....	60
<b>Figura 4-12.</b> Modelo desarrollado en ns-3 a nivel de red .....	62
<b>Figura 4-13.</b> Creación de ayudantes para la aplicación <i>OnOff</i> y de cernidero .....	62
<b>Figura 4-14.</b> Configuración del atributo <i>Remote</i> para la aplicación <i>OnOff</i> .....	62
<b>Figura 4-15.</b> Configuración de los atributos <i>OffTime</i> , <i>OnTime</i> y <i>MaxBytes</i> para la aplicación <i>OnOff</i> ...	63
<b>Figura 4-16.</b> Instalación de aplicaciones en los nodos.....	63

<b>Figura 4-17.</b> Recepción de datos en los clientes .....	64
<b>Figura 5-1.</b> Topologías de red para casos de estudio .....	66
<b>Figura 5-2.</b> Topología1. Retardo para los clientes 1 y 10 .....	68
<b>Figura 5-3.</b> Topología1. Retardo de colas.....	68
<b>Figura 5-4.</b> Topología1. Tamaño de colas .....	69
<b>Figura 5-5.</b> Topología1. <i>Throughput</i> .....	69
<b>Figura 5-6.</b> Topología2. Retardo .....	71
<b>Figura 5-7.</b> Topología2. Retardo de colas.....	72
<b>Figura 5-8.</b> Topología2. Tamaño de colas .....	73
<b>Figura 5-9.</b> Topología2. <i>Throughput</i> .....	74
<b>Figura 5-10.</b> Topología3 – Escenario1. Retardo.....	76
<b>Figura 5-11.</b> Topología3 – Escenario1. Retardo de colas .....	77
<b>Figura 5-12.</b> Topología3 – Escenario1. Tamaño de colas .....	78
<b>Figura 5-13.</b> Topología3 – Escenario1. <i>Throughput</i> .....	79
<b>Figura 5-14.</b> Topología3 – Escenario2. Retardo.....	81
<b>Figura 5-15.</b> Topología3 – Escenario2. Retardo de colas .....	81
<b>Figura 5-16.</b> Topología3 – Escenario2. Tamaño de colas .....	83
<b>Figura 5-17.</b> Topología3 – Escenario2. <i>Throughput</i> .....	84

## LISTA DE ACRÓNIMOS

<b>AAC:</b>	<i>Advanced Audio Coding</i> (Codificación Avanzada de Audio)
<b>ADSL:</b>	<i>Asymmetric Digital Subscriber Line</i> (Línea de Abonado Digital Asimétrica)
<b>ALSA:</b>	<i>Advanced Linux Sound Architecture</i> (Arquitectura Linux Avanzada de Sonido)
<b>AMR:</b>	<i>Adaptive Multi-Rate</i> (Multi-Tasa Adaptativo)
<b>AODV:</b>	<i>Ad hoc On-Demand Distance Vector Routing</i> (Enrutamiento Ad hoc por Vector de Distancia bajo Demanda)
<b>ATIS:</b>	<i>Automatic Terminal Information Service</i> (Servicio de Información Terminal Automático)
<b>BICAMSA:</b>	Grupo de Biotecnología, Calidad Medioambiental y Seguridad Agroalimentaria
<b>BSD:</b>	<i>Berkeley Software Distribution</i> (Distribución de Software Berkeley)
<b>CAV:</b>	Comunidad Académica Virtual
<b>CD:</b>	<i>Compact Disc</i> (Disco Compacto)
<b>CDF:</b>	<i>Cumulative Distribution Function</i> (Función de Distribución Acumulada)
<b>CSMA:</b>	<i>Carrier Sense Multiple Access</i> (Acceso Múltiple por Detección de Portadora)
<b>DRM:</b>	<i>Digital Rights Management</i> (Gestión Digital de Derechos)
<b>DSL:</b>	<i>Digital Subscriber Line</i> (Línea de Abonado Digital)
<b>DSM-CC:</b>	<i>Digital Storage Media Command and Control</i> (Comando y Control de Almacenamiento Digital de Multimedia)
<b>DV:</b>	<i>Digital Video</i> (Video Digital)
<b>DVB:</b>	<i>Digital Video Broadcasting</i> (Difusión Digital de Video)
<b>DVD:</b>	<i>Digital Video Disc</i> (Disco Digital de Video)
<b>EDiTV:</b>	Educación Virtual basada en Televisión Interactiva
<b>ETSI:</b>	<i>European Telecommunications Standards Institute</i> (Instituto Europeo de Estandarización de Telecomunicaciones)
<b>FBM:</b>	<i>Fractional Brownian Motion</i> (Movimiento Browniano Fraccional)
<b>FTP:</b>	<i>File Transfer Protocol</i> (Protocolo de Transferencia de Archivo)
<b>FTTC:</b>	<i>Fiber To The Curb</i> (Fibra Hasta la Acera)
<b>FTTH:</b>	<i>Fiber To The Home</i> (Fibra Hasta el Hogar)
<b>GICD:</b>	Grupo de Investigación en Comunicación para el Desarrollo
<b>GIPC:</b>	Grupo de Investigación, Pedagogía y Currículo
<b>GIRST:</b>	Grupo de Investigación en Redes y Servicios Telemáticos
<b>GIT:</b>	Grupo de Ingeniería Telemática
<b>GITI:</b>	Grupo de Investigación en Telemática e Informática Aplicada
<b>GOP:</b>	<i>Groups Of Pictures</i> (Grupos de Imágenes)
<b>GPL:</b>	<i>General Public License</i> (Licencia Pública General)
<b>GUI:</b>	<i>Graphical User Interface</i> (Interfaz Gráfica de Usuario)
<b>HD:</b>	<i>High Definition</i> (Alta Definición)
<b>HTTP:</b>	<i>HyperText Transfer Protocol</i> (Protocolo de Transferencia de Hipertexto)
<b>IDC:</b>	<i>Infinitely Divisible Cascade</i> (Cascada Divisible Infinitamente)
<b>IDIS:</b>	Grupo de Investigación y Desarrollo en Ingeniería de Software

<b>IEEE:</b>	<i>Institute of Electrical and Electronics Engineers</i> (Instituto de Ingenieros Eléctricos y Electrónicos)
<b>IGMP:</b>	<i>Internet Group Management Protocol</i> (Protocolo de Gestión de Grupos de Internet)
<b>INRIA:</b>	<i>Institut National de Recherche en Informatique et en Automatique</i> (Instituto Nacional de Investigación en Informática y Automática)
<b>IMS:</b>	<i>IP Multimedia Subsystem</i> (Subsistema Multimedia IP)
<b>IP:</b>	<i>Internet Protocol</i> (Protocolo de Internet)
<b>IPTV:</b>	<i>Internet Protocol Television</i> (Televisión sobre IP)
<b>ITU:</b>	<i>International Telecommunications Union</i> (Unión Internacional de Telecomunicaciones)
<b>K-S:</b>	<i>Kolmogorov-Smirnov</i>
<b>LAN:</b>	<i>Local Area Network</i> (Red de Área Local)
<b>LTE:</b>	<i>Long Term Evolution</i> (Evolución a Largo Plazo)
<b>MHP:</b>	<i>Multimedia Home Platform</i> (Plataforma Multimedia de Hogar)
<b>MMPP:</b>	<i>Markov Modulated Poisson Process</i> (Proceso de Poisson Modulada Markov)
<b>MPLS:</b>	<i>Multi-Protocol Label Switching</i> (Conmutación de Etiqueta Multi-Protocolo)
<b>MOS:</b>	<i>Mean Opinion Score</i> (Puntaje Medio de Opinión)
<b>MPEG:</b>	<i>Moving Pictures Experts Group</i> (Grupo de Expertos en Movimiento de Imágenes)
<b>MUX:</b>	Multiplexor
<b>NDVR:</b>	<i>Network Digital Video Recorder</i> (Red Digital de Grabación de Video)
<b>NGN:</b>	<i>New Generation Networks</i> (Redes de Nueva Generación)
<b>OLSR:</b>	<i>Optimized Link State Routing</i> (Enrutamiento Optimizado de Estado de Enlace)
<b>OSPF:</b>	<i>Open Shortest Path First</i> (Camino Abierto más Corto Primero)
<b>P2P:</b>	<i>Peer to Peer</i> (Punto a Punto)
<b>PDF:</b>	<i>Probability Distribution Function</i> (Función de Distribución de Probabilidad)
<b>PSD:</b>	<i>Power Spectral Density</i> (Densidad Espectral de Potencia)
<b>PSI:</b>	<i>Program and Service Information</i> (Información de Servicios y Programa)
<b>QoE:</b>	<i>Quality of Experience</i> (Calidad de Experiencia)
<b>QoS:</b>	<i>Quality of Service</i> (Calidad de Servicio)
<b>REST:</b>	<i>Representational State Transfer</i> (Estado de Transferencia Representacional)
<b>RTP:</b>	<i>Real-time Transport Protocol</i> (Protocolo de Transporte en tiempo Real)
<b>RTSP:</b>	<i>Real-Time Streaming Protocol</i> (Protocolo de Streaming en Tiempo Real)
<b>SD:</b>	<i>Standard Definition</i> (Definición Estándar)
<b>SDP:</b>	<i>Session Description Protocol</i> (Protocolo de Descripción de Sesión)
<b>SIP:</b>	<i>Session Initiation Protocol</i> (Protocolo de Iniciación de Sesión)
<b>SOAP:</b>	<i>Simple Object Access Protocol</i> (Protocolo de Acceso de Objeto Simple)
<b>SPA:</b>	<i>Semi-parametric Periodogram-Based Algorithm</i> (Algoritmo Basado en Periodograma semi-paramétrico)
<b>STB:</b>	<i>Set-Top Box</i>
<b>TCP:</b>	<i>Transmission Control Protocol</i> (Protocolo de Control de Transmisión)
<b>TDT:</b>	<i>Terrestrial Digital Television</i> (Televisión Digital Terrestre)
<b>TIC:</b>	Tecnologías de la Información y las Comunicaciones

- TISPAN:** *Telecommunications and Internet converged Services and Protocols for Advanced Networking* (Servicios y Protocolos convergentes de Internet y Telecomunicaciones para Networking Avanzado)
- UDP:** *User Datagram Protocol* (Protocolo de Datagrama de Usuario)
- URL:** *Uniform Resource Locator* (Localizador de Fuente Uniforme)
- VCD:** *Video Compact Disc* (Disco Compacto de Video)
- VDSL:** *Very high bit rate Digital Subscriber Line* (Línea de Abonado Digital con tasa de bit muy Alta)
- VoD:** *Video on Demand* (Video bajo Demanda)
- VoIP:** *Voice on IP* (Voz sobre IP)
- WAN:** *Wide Area Network* (Red de Área Amplia)
- WAV:** *Waveform Audio File* (Archivo de Audio de forma de Onda)
- WiFi:** *Wireless Fidelity* (Fidelidad Inalámbrica)
- WiMAX:** *Worldwide Interoperability for Microwave Access* (Interoperabilidad Mundial para Acceso por Microondas)
- XML:** *eXtensible Markup Language* (Lenguaje de Marcas Extensible)



# CAPITULO 1

## INTRODUCCIÓN

### 1.1 CONTEXTO GENERAL

Internet, como el elemento más revolucionario de las Tecnologías de la Información y de las Comunicaciones (TIC), ha provisto a la sociedad en los últimos años la infraestructura tecnológica para el desarrollo de relaciones sociales innovadoras, un ejemplo de esto son las redes sociales, las cuales están en auge en la actualidad con cientos de millones de usuarios alrededor del mundo. Estas redes sociales permiten crear comunidades en las que sus miembros tienen afinidades sociales de algún tipo e intereses personales comunes, tal como se hace en las comunidades físicas, con la ventaja de que en las redes sociales no existen las barreras del tiempo ni el espacio [1].

El uso de las TIC fomenta el desarrollo del aprendizaje electrónico (e-Learning), cuyo objetivo es resolver problemas educativos que van desde el aislamiento geográfico al centro educativo hasta perfeccionar el aprendizaje, ajustado a los adelantos tecnológicos de la sociedad, y considerando el ahorro de tiempo y dinero que esto puede aportar, su objetivo es facilitar el acceso de contenidos educativos a estudiantes en diversos niveles [2]. De modo que algunos servicios de Internet como Video bajo Demanda (*Video On Demand, VoD*), chats, foros, blogs, FTP (File Transport Protocol), etc., se convierten en herramientas que ayudan a la gestión de la enseñanza.

Las Comunidades Académicas Virtuales (CAV) han surgido para brindar apoyo a los procesos de e-Learning. Estas comunidades son versiones académicas de las redes sociales, en las que se almacena documentos (texto, audio, video) pertenecientes a los cursos ofrecidos por las instituciones educativas. Estos documentos pueden ser aportados por las mismas instituciones educativas, por los propios usuarios de la comunidad o también pueden ser extraídos desde páginas de Internet. De acuerdo a [3], una CAV está definida como uno, o varios grupos de individuos que están vinculados por intereses en común, que tienen la capacidad de poseer una fuerza de voluntad autónoma y están comprometidos en un proceso de aprendizaje continuo, cuyo principal objetivo es el de construir conocimientos de forma compartida utilizando las TIC como un medio de expresión, como herramienta de comunicación, como recurso didáctico e incluso como instrumento de gestión.

Las plataformas tecnológicas en las que se pueden implementar servicios de e-Learning van desde los tradicionales sistemas de televisión analógica por cable o por onda terrestre, pasando por Internet, hasta los modernos sistemas de televisión digital que utilizan medios de transmisión avanzados como la fibra óptica. En el Internet, los procesos de e-Learning se han venido desarrollando prácticamente desde su aparición. En la televisión analógica, la cual es el medio masivo de comunicación por excelencia, los procesos de e-Learning no han

tenido tanto éxito como en Internet debido a su falta de interactividad. Esto ha cambiado en los últimos años gracias al desarrollo la Televisión Digital Interactiva (TDi). Este concepto ha permitido desarrollar tecnologías para que la televisión entre en el mundo de la interactividad y permita a sus usuarios participar activamente en el desarrollo de los contenidos que visualizan.

Este trabajo de grado está enmarcado en el proyecto ST-CAV (Servicios de T-Learning<sup>1</sup> para soporte de una Comunidad Académica Virtual) de la Universidad del Cauca con sus grupos de investigación GIT (Grupo de Ingeniería Telemática), GIPC (Grupo de Investigación, Pedagogía y Currículo), BICAMSA (Grupo de Biotecnía, Calidad Medioambiental y Seguridad Agroalimentaria) e IDIS (Grupo de Investigación y Desarrollo en Ingeniería de Software). Este proyecto está avalado por Colciencias (Cód. 1103 489 25425) y cuenta además con la participación de la Universidad Autónoma de Occidente (Cali – Valle del Cauca) con sus grupos de investigación GITI (Grupo de Investigación en Telemática e Informática Aplicada) y GICD (Grupo de Investigación en Comunicación para el Desarrollo), y la Universidad de Oviedo (España) con el grupo de investigación GIRST (Grupo de Investigación en Redes y Servicios Telemáticos). El objetivo general de este proyecto es construir y desarrollar servicios de soporte para la conformación de una CAV, a través de la integración de la TDi y la web [4].

En el mencionado proyecto se abarca el problema del soporte de una CAV desde distintos ámbitos tecnológicos, todos pertenecientes a la TDi. Entre estas tecnologías se encuentran TDT (Televisión Digital Terrestre), Televisión Móvil (*Mobile TV*) e IPTV. Estas tecnologías permiten la transmisión y recepción de datos en una CAV con la ayuda de la web. El presente trabajo de grado se concentra en el estudio del comportamiento del tráfico en una CAV sobre IPTV.

En [5] se define a IPTV como la entrega segura de servicios de entretenimiento relacionados con video, a los usuarios. Estos servicios pueden incluir, por ejemplo, televisión en vivo, VoD y televisión interactiva. Estos servicios son entregados a través de una red conmutada de paquetes agnóstica, que emplea el protocolo IP para el transporte de audio, video y señales de control. En contraste al video sobre Internet público, la seguridad y el rendimiento de la red son estrechamente gestionados para asegurar una experiencia de entretenimiento superior, resultando en un ambiente de negocio irresistible para proveedores de contenido, anunciantes y usuarios.

Esta tecnología utiliza el protocolo IP para la transmisión de contenidos de televisión, esto requiere que la infraestructura en la que se soporte sea la misma que utiliza Internet. Por el hecho de ser parte de la televisión digital, IPTV presenta varias ventajas frente a la televisión analógica, entre ellas están: mayor cantidad de canales con el mismo ancho de banda, mejor robustez ante interferencias y ruido, capacidad de servicios agregados no existentes en la televisión analógica, capacidad de interactividad gracias a un canal de retorno, entre otras

---

<sup>1</sup> El término T-Learning se utiliza con el significado de aprendizaje interactivo a través de un televisor [4].



[6]. Hay que mencionar que paralelamente a las grandes ventajas que ofrece IPTV, se deben cumplir ciertos requerimientos para que la calidad del servicio sea aceptable, un ejemplo de esto es que para utilizar IPTV se requiere mantener información de estado sobre cada comunicación en todos los routers por los que pasan los datos.

Los servicios con los que cuenta una CAV pueden ser muy variados dependiendo del nivel académico de los participantes como también de en qué campo del conocimiento estén involucrados. A pesar de esto hay servicios genéricos con los que toda CAV debe contar, entre ellos se tienen chats, blogs, foros y VoD. Algunos servicios de valor agregado como Telefonía sobre IP (*Voice on IP, VoIP*) y otras aplicaciones, es posible implementarlos en una CAV dependiendo de la capacidad de los servidores con que esta cuente y del ancho de banda disponible en la red.

## **1.2 ESCENARIO DE MOTIVACIÓN**

El presente y el futuro de la educación formal a nivel mundial están basados en la diversificación de los entornos de aprendizaje. Particularmente, este trabajo de grado está enfocado en el estudio del comportamiento del tráfico para un entorno de aprendizaje relativamente nuevo. Este nuevo entorno de aprendizaje hace uso de la televisión como un catalizador cultural que se diferencia de otras plataformas tecnológicas como Internet o la telefonía móvil, en características como la antigüedad y la forma como se despliegan sus servicios asociados, las cuales brindan una gran cantidad de usuarios que reconocen, usan y aceptan su modo de operación [8].

La importancia del presente trabajo de grado está en que al proveer un modelo de tráfico en el que se implementen los servicios de una CAV y a la vez se puedan medir distintos parámetros de rendimiento de red, se permite establecer los requerimientos necesarios para soportar estos servicios, además de los protocolos, estándares y QoS adecuados para tal objetivo, todo esto, antes de implementar físicamente la red y sus servicios, lo que reduce costos y tiempos en caso de que esta implementación no fuera viable.

## **1.3 OBJETIVOS**

### **1.3.1 Objetivo general**

Modelar el tráfico de los servicios de una Comunidad Académica Virtual (CAV) sobre IPTV.

### **1.3.2 Objetivos específicos**

- Caracterizar el comportamiento del tráfico generado por los servicios de una CAV sobre IPTV.
- Obtener un modelo del comportamiento del tráfico generado por los servicios de una CAV sobre IPTV.

- Evaluar mediante simulación el comportamiento del tráfico generado por los servicios de una CAV sobre IPTV.

## 1.4 ESTRUCTURA DE LA MONOGRAFÍA

La estructura de la monografía presenta la siguiente organización de secciones:

*Capítulo 2:* En este capítulo se explora la literatura existente acerca de temas relevantes para el presente proyecto de grado, como IPTV y modelado de tráfico, y se organizan para la contextualización del lector. Además, se presentan distintos artículos y publicaciones relacionadas con el modelado de tráfico.

*Capítulo 3:* Este capítulo muestra la arquitectura del laboratorio de Televisión Digital de la Universidad del Cauca y la descripción de sus distintos componentes. También se describe en este capítulo la caracterización realizada al tráfico producido por tres de los videos disponibles para el proyecto ST-CAV, y se proponen las topologías de red para los casos de estudio.

*Capítulo 4:* Aquí se describe el desarrollo de los modelos de simulación en los dos software de simulación de redes que se utilizan para este proyecto, además de realizar su respectiva validación por medio de la comparación de los resultados arrojados por estos con los datos del tráfico real capturado en laboratorio.

*Capítulo 5:* En este capítulo se presentan los resultados obtenidos para la evaluación de los dos modelos desarrollados sobre las distintas topologías propuestas para casos de estudio en el capítulo 3. Además, se realiza un análisis comparativo entre los dos simuladores de redes.

*Capítulo 6:* En este capítulo se presentan las conclusiones deducidas del presente trabajo de grado. También se proponen distintos trabajos futuros que lo complementen.

Además de los capítulos descritos anteriormente, este trabajo de grado presenta los siguientes anexos en los que se encuentra información complementaria para el proyecto:

*Anexo A:* Conceptos y funcionamiento de los simuladores

*Anexo B:* Análisis de resultados para los videos 2 y 3

*Anexo C:* Instalación de herramientas software

*Anexo D:* Construcción de topologías en ns-3

## CAPITULO 2

### ESTADO DEL ARTE

#### 2.1 CONCEPTOS FUNDAMENTALES

En este apartado se abordan los temas más relevantes del presente trabajo de grado, que son: CAV, IPTV y modelado de tráfico. Esto se realiza mediante la definición de cada uno de ellos como también de algunos otros conceptos y definiciones relacionados, como e-Learning y TDi, que aportan a su adecuada contextualización.

##### 2.1.1 e-Learning

El concepto de e-Learning se define de muchas formas diferentes, fundamentalmente debido a que los actores que de él hacen uso son muy diversos, cada uno con su idiosincrasia y su ámbito de aplicación.

“Desde la perspectiva de su concepción y desarrollo como herramienta formativa, los sistemas de e-Learning tienen una dualidad pedagógica y tecnológica. Pedagógica en cuanto a que estos sistemas no deben ser meros contenedores de información digital, sino que esta debe ser transmitida de acuerdo a unos modelos y patrones pedagógicamente definidos para afrontar los retos de estos nuevos contextos. Tecnológica en cuanto a que todo el proceso de enseñanza-aprendizaje se sustenta en aplicaciones software, principalmente desarrollada en ambientes web, lo que le vale a estos sistemas el sobrenombre de plataformas de formación. Desde la perspectiva de su uso se podría distinguir la visión que tienen sus usuarios finales, que con independencia de su madurez y formación, verán al sistema e-Learning como una fuente de servicios para alcanzar su cometido formativo. No obstante, también es factible diferenciar una visión de organización, en la que se definen el alcance y los objetivos buscados con la formación basada en estos sistemas, distinguiéndose una visión académica y una visión empresarial” [9].

Teniendo en cuenta los aspectos relacionados anteriormente, en [9] se define al e-Learning como “la capacitación no presencial que, a través de plataformas tecnológicas, posibilita y flexibiliza el acceso y el tiempo en el proceso de enseñanza-aprendizaje, adecuándolos a las habilidades, necesidades y disponibilidades de cada discente, además de garantizar ambientes de aprendizaje colaborativos mediante el uso de herramientas de comunicación síncrona y asíncrona, potenciando en suma el proceso de gestión basado en competencias”.

En esta y en todas las demás definiciones existentes de e-Learning que se pueden encontrar en la bibliografía especializada, se acaba haciendo mención explícita o implícita de lo que se conoce como el triángulo del e-Learning.

La tecnología es uno de los lados de este triángulo. Lo que se denominan plataformas on-Line, campus virtuales, etc. Es un elemento básico, que ha pasado de venderse a unos precios astronómicos a encontrarse hoy con acceso libre y gratuito. El desarrollo de software libre permite que miles de profesores de cualquier parte del mundo cuelguen sus cursos en la red y ofrezcan su conocimiento a millones de usuarios que pueden encontrar cualquier materia para estudiar y formarse.

Los contenidos conforman otro de los lados del triángulo. En un primer momento la industria se volcó en la elaboración de contenidos estándar donde la enseñanza de idiomas y de informática/ofimática fueron los cursos más demandados en el sector. Pero con el dominio, conocimiento y uso de estos cursos, las empresas demandan cada vez más contenidos a medida, segmento que está en crecimiento, sobre todo por la globalización en su producción.

Los servicios completan el triángulo. Se entiende por servicios la acción de los tutores, profesores on-Line, la documentación que se encuentra en la biblioteca virtual y todo un amplio abanico de posibilidades desde el chat hasta la gestión de descargas de contenidos, que ayudan a desarrollar los múltiples modelos del e-Learning.

Para que se aseguren los objetivos y las necesidades de los estudiantes, las plataformas de e-Learning deben poseer 4 características fundamentales [11]:

1. **Interactividad:** Es necesaria la realimentación del estudiante con el resto de participantes; formulando preguntas, planteando respuestas, identificando la inconsistencia entre ideas, etc. [11]. En conclusión, todos los estudiantes deben desempeñar un papel activo en su educación utilizando las interfaces del curso para explorar en busca de nuevo conocimiento.
2. **Ubicuidad:** La noción de ubicuidad aparentemente conllevará a servicios ubicuos más completos, como la creciente demanda de acceder a servicios deseados en cualquier momento y en cualquier lugar [12]. Aplicando el término al concepto de e-Learning se genera lo que se conoce como U-Learning. La disponibilidad que esta ofrece sirve como espacio para compartir información en cualquier lugar y en cualquier momento con el uso de cualquier dispositivo con acceso a la red [13].
3. **Disponibilidad:** Es necesario que los usuarios de e-Learning tengan constantemente la posibilidad de acceso a la información. En un entorno e-Learning esto es posible gracias a la digitalización y almacenamiento de los contenidos de los cursos, esto facilita la aprehensión del conocimiento.
4. **Evaluación:** Cada estudiante de e-Learning se convierte en un gestor de aprendizaje, puede resolver ejercicios para autoevaluarse, discutir y solucionar problemas con sus compañeros, transferir datos, trabajar colaborativamente e interactuar con el fin de incrementar sus conocimientos, por lo tanto esta herramienta potencializa el desempeño del estudiante mejorando su atención y rendimiento académico [14].

### **2.1.2 Comunidad Académica Virtual (CAV)**

Internet ha tenido una marcada evolución en los últimos años, facilitando el intercambio de información de manera flexible y ágil gracias a los múltiples servicios disponibles en red. Estos a su vez hacen parte de la tendencia actual de la red de redes por fomentar la colaboración y el trabajo en comunidad; ejemplos de estos son: foros, wikis, blogs, chats, entre otros, estos servicios son más conocidos como herramientas de la Web 2.0. Actualmente estos servicios están integrados en plataformas como las redes sociales [15]. Una parte del desarrollo de las redes sociales, que se ha interesado por la formación académica ha evolucionado en lo que se conoce como Comunidades Académicas Virtuales (CAV).

Las CAV permiten a las personas construir conocimiento a través del uso intensivo de las tecnologías de la información y la comunicación, así como también permiten el acceso a otros servicios (asociados o no a los contenidos). Estas comunidades proveerán un ambiente adecuado para promover el aprendizaje colaborativo [16].

### **2.1.2.1 Definición de Comunidad Virtual**

Las comunidades o grupos de individuos organizados informalmente o semi-formalmente, son un instrumento de gestión del conocimiento [17]. Muchas compañías las consideran como un instrumento estratégico para superar limitaciones pertenecientes a la flexibilidad y la comunicación de las formas organizacionales tradicionales tales como organizaciones orientadas a procesos o funcionales [18].

Las comunidades que están geográficamente separadas requieren alguna especie de sistema de comunicación para crear un puente que acorte las distancias espaciales y temporales. Estas comunidades son llamadas “Comunidades Virtuales”.

En [19] se define a una comunidad virtual como un grupo de personas quienes interactúan en un ambiente virtual, con un propósito específico con soporte tecnológico y guiados por una serie de normas o políticas.

Por otra parte en [20] se define a las comunidades virtuales como agregaciones sociales que emergen de la red cuando un número suficiente de personas entabla discusiones públicas durante un tiempo lo suficientemente largo, con suficiente sentido humano, para formar redes de relaciones personales en el ciberespacio.

Estas definiciones relacionan las comunidades virtuales principalmente con las personas que las conforman, más que con la información y otros tipos de recursos. Son las personas las que finalmente proporcionan un ambiente atractivo, un espacio de relaciones interactivas que generan una atmósfera de confianza, un lugar real que satisface necesidades básicas del ser humano como interés por la información, relación con los otros, y posibilidades de transacción e intercambio [21].

Con el análisis de estas definiciones, en [1] se concluye una nueva definición: “Una comunidad virtual es uno o varios grupos de individuos que están vinculados por intereses en común, que tienen fuerza de voluntad autónoma, están comprometidos con su comunidad y tienen un proceso de aprendizaje continuo. Trabajan de forma colaborativa para satisfacer sus necesidades de información o transacción y construir conocimiento utilizando las tecnologías de la información y comunicación como un medio de expresión.”

### **2.1.2.2 Características de una Comunidad Virtual**

En 1996 un grupo multidisciplinario de académicos identificó las características esenciales de las comunidades virtuales [22]:

- Los miembros tienen una meta compartida, interés, necesidad, o actividad que provee la razón principal para pertenecer a la comunidad.
- Los miembros se comprometen con la comunidad, mantienen la participación activa y hay a menudo intensas interacciones, fuertes lazos emocionales y actividades compartidas por los participantes.
- Los miembros tienen el acceso a los recursos compartidos y hay políticas para determinar el acceso a esos recursos.
- La reciprocidad de información (obtener respuestas y ofrecer apoyo), soporte y servicios entre los miembros, ya que los usuarios ingresan con el objetivo de obtener información clasificada y confiable e incluso participar con su propio conocimiento.
- Hay un contexto compartido de convenciones sociales, lenguaje y protocolos que guían la interacción entre las personas.

Además identificaron las siguientes características, aunque no como esenciales, porque podrían impactar significativamente las interacciones en el medio virtual:

- La evidencia de que las personas tienen diferentes roles; entendiendo como rol un conjunto de permisos definidos para todo el sitio con el propósito de asignar usuarios específicos en contextos específicos.
- La reputación de las personas; en el espacio virtual es imprescindible tener una personalidad para ser reconocido dentro de la misma.
- El conocimiento de las condiciones de la membresía y la identidad del grupo, que permiten crear tejidos complejos de relaciones personales y un sentido creciente de identificación con la comunidad de manera global.
- El criterio de la iniciación para unirse a la comunidad que genera el componente afectivo y los fuertes niveles de interacción entre la persona que ingresa y los miembros que la componen, creando una sensación de pertenencia.
- La historia de la comunidad y la existencia durante un período de tiempo, que contribuye a que persista, aumente con el tiempo y facilite la cohesión entre los miembros.
- Eventos notables o rituales, para conversar y socializar de manera informal a través de comunicación simultánea.
- Los ambientes físicos compartidos y la membresía voluntaria contribuyen a construir significado compartido, confianza compartida y motivación compartida de manera que son distintivas y sensibles a las necesidades crecientes entre los participantes.

### **2.1.3 Televisión Digital Interactiva (TDi)**

TDi es una tecnología que combina el atractivo de la televisión tradicional con características interactivas basadas en tecnología Web, ofreciendo a los televidentes acceso a servicios e información en “un solo clic”. De acuerdo a la información del DSL fórum, el número total de usuarios de TDi en el mundo pasó de 2.95 millones a 8.23 millones de Julio de 2006 a Julio de 2007. También muestra que el crecimiento depende mayormente de los Estados Unidos y los países europeos.

TDi integra muchas funciones multimedia y enriquece el intercambio de fuentes de información. Este nuevo mecanismo interactivo también mejora las actividades comerciales en el campo del servicio de información [23].

Una de las principales ventajas de la llegada de TDi es la posibilidad del incremento de interacción entre el televidente y el terminal de acceso (televisor) a través de servicios y aplicaciones, como juegos, VoD, *t-commerce*, navegación por Internet, entre otros. Sin embargo, debido a ser una tecnología nueva, teniendo pocos estándares establecidos y muchas investigaciones en progreso, la construcción de aplicaciones para TDi todavía es un reto, especialmente para las personas que tienen intereses en sistemas de televisión, pero no dominan tecnologías y herramientas software para el desarrollo de estas aplicaciones. En esta dirección, los conceptos y tecnologías aplicados a los sistemas de computación tradicionales también pueden ser aplicados a los sistemas de televisión interactiva, pero es relevante considerar aspectos específicos de este nuevo entorno, especialmente en términos de la interacción del televidente [24].

### 2.1.3.1 Aplicaciones TDi

Son las responsables de la interacción del televidente con los contenidos y servicios disponibles, permitiéndole ser parte activa de la programación. Bajo el concepto de aplicaciones TDi se tienen *t-government*, *t-commerce*, juegos en línea, aplicaciones de información como estado del tiempo, estado de las carreteras, aeropuertos, publicidad interactiva, entre otros [8].

Con la posibilidad de ofrecer aplicaciones interactivas, los sistemas TDi comienzan a preocuparse no solo con el ciclo de programas, sino también con las aplicaciones. Un programa de televisión sin aplicaciones tiene un ciclo continuo de vida, siendo generado, transmitido y recibido. Cuando se agregan aplicaciones este ciclo continuo de programa es el mismo pero otro ciclo de vida es incluido en el programa. Esto puede ser clasificado como un ciclo pseudo-interactivo o interactivo, incluyendo programas o aplicaciones.

Un programa pseudo-interactivo limita el ciclo de vida al intercambio de información con el *Set-Top Box* (STB), que puede escribir y leer información, pero sin ninguna forma de comunicación externa usando el canal interactivo.

Los programas interactivos tienen las mismas características pseudo-interactivas con la diferencia de la posibilidad de una aplicación de comunicación a través del canal interactivo. Esta característica permite al proveedor de servicio agregar más funcionalidad, permitiendo aplicaciones como *t-commerce*, *t-government* y *t-learning*.

Una nueva forma de comunicación entre el proveedor de servicios y el usuario es posible. Los sistemas de televisión digital deben asegurar el medio para permitir un canal de comunicación interactivo localizado entre los proveedores de servicio y los usuarios para enviar y recibir información [25].

### 2.1.3.2 Contenidos multimedia de TDi

Son textos, imágenes, sonido, animaciones, video, etc., adicionados a lo largo de la transmisión de un programa televisivo, los cuales deben cumplir con una serie de características y especificaciones técnicas para un adecuado despliegue y previa visualización [8].

### 2.1.3.3 Canal de retorno y canal de interactividad

El canal de retorno es el medio de comunicación entre el terminal de acceso del usuario y el proveedor de servicios de interactividad, siendo uno de los componentes más relevantes de la arquitectura de los sistemas de televisión digital interactiva (TDI). La comunicación puede ocurrir a través de una conexión permanente como Ethernet, ADSL (*Asymmetric Digital Subscriber Line*), cable modem u otro, o una conexión temporal, como una red telefónica [24].

El canal de interactividad está diseñado para asegurar dos vías de comunicación entre los proveedores de servicios y los usuarios, para garantizar que la información resultante de la interacción del usuario pueda ser enviada al proveedor de servicios interactivos, y este pueda pedir información relacionada, para producir y entregar contenido nuevo y actualizado a los usuarios a través de programas de televisión o aplicaciones interactivas [25].

### 2.1.4 Internet Protocol Television (IPTV)

IPTV es la convergencia de la televisión digital y el Internet. Provee servicios de video digital por medio del uso del Protocolo de Internet (IP) sobre una infraestructura de red [26]. Los servicios de IPTV incluyen la difusión tradicional de televisión en vivo, VoD, servicios punto a punto (compartición de archivos), red de grabación digital de video (NDVR, Network Digital Video Recording) y programas y propagandas usuales, muchos más que los servicios de la televisión por cable [27].

IPTV ofrece un servicio bidireccional, más interactivo que la televisión tradicional. Esto es gracias a que IPTV tiene una infraestructura diferente a los de la televisión tradicional, en el cual, todos los contenidos son “empujados” hacia los usuarios [26]. La infraestructura de IPTV está basada en una selección personal, dependiendo de los intereses y necesidades de los usuarios. Por lo tanto, más funciones de control están disponibles para el usuario tales como pausa (*pause*), adelantar (*forward*) y regresar (*rewind*).

Las tecnologías detrás de IPTV resultan de la mezcla de tecnología digital y los protocolos IP. Las señales analógicas originales son digitalizadas en una cadena de ceros y unos, los cuales luego son comprimidos para poder ser transmitidos [26]. En la mayoría de sistemas IPTV, los principales protocolos utilizados son: *Internet Group Management Protocol* (IGMP) y *Real-Time Transport Protocol* (RTP) o *Real Time Streaming Protocol* (RTSP) [28].

Es importante tener en cuenta que estos servicios no son lo mismo que transmitir video por el Internet público, donde la información es descargada a otro tipo de equipos, como por ejemplo, computadores, aunque las diferencias claves entre estos dos servicios no están claramente definidas. Televisión por Internet es completamente diferente de IPTV en términos del modelo para el consumidor, la edición y la infraestructura en sí misma. En la Televisión por Internet, el modelo de edición de video es el mismo que el modelo de edición web. La transmisión de video en redes abiertas de IP usa el Internet público, y el acceso al video puede ser realizado desde cualquier computador en todo el mundo. Por el contrario, IPTV hace referencia a la entrega de servicios de video sobre redes cerradas de IP las cuales están soportadas por compañías de telecomunicaciones [26].

Para que IPTV funcione correctamente se necesita realizar cambios en la infraestructura física de las redes IP. IPTV utiliza el protocolo IP para entregar todos los servicios mencionados anteriormente a los usuarios con Calidad de Servicio (QoS, *Quality of Service*) garantizada. IPTV puede ser incorporado con tecnologías de acceso DSL de alta velocidad, tales como ADSL2 (*Asymmetric Digital Subscriber Line*), ADSL2+ y VDSL (*Very high bit-rate Digital Subscriber Line*), como también con Ethernet de alta velocidad y la emergente tecnología LAN (*Local Area Network*) inalámbrica IEEE 802.11n.

IPTV es una integración de servicios de voz, video y datos usando acceso a Internet de gran ancho de banda y gran velocidad. A continuación se relacionan los componentes de IPTV [27]:

- Fuentes de IPTV incluyendo bases de datos para VoD y otros programas.
- Internet de alta velocidad con funciones de multicasting, QoS garantizada, y también, que incluya una red de abonado óptica para servir como núcleo de red IP multicast.
- Redes de acceso de alta velocidad tales como ADSL, ADSL2+, VDSL, una combinación de fibra óptica y DSL como FTTC (*Fiber To The Curbe*) o FTTH (*Fiber To The Home*), Ethernet de alta velocidad, e IEEE 802.11n.



- Dispositivos de usuario IPTV tales como televisores digitales y STB's.

#### 2.1.4.1 Características de IPTV

De acuerdo a [27], las principales características de IPTV son:

- **Selección:** Los usuarios pueden seleccionar su programa de televisión con selección rápida de canal y corto tiempo en el cambio entre los mismos.
- **Almacenamiento:** Los programas de televisión son almacenados en dispositivos locales de almacenamiento para que los usuarios puedan mirarlos en cualquier momento; además, los proveedores de servicio deben almacenar al menos 100 horas de programas de televisión y videos.
- **Qos:** La calidad de servicio debe ser garantizada; una televisión de definición estándar (SD, *Standard Definition*) y una de alta definición (HD, *High Definition*) necesitan anchos de banda de 1 – 4 Mbps y 4 – 12 Mbps respectivamente.
- **Bajo costo:** El costo para el usuario debe ser bajo, esta es la clave del éxito.

#### 2.1.4.2 Calidad de Servicio (QoS)

El despliegue exitoso de servicios de IPTV requiere una excelente QoS para video, voz y datos. Las métricas de QoS para video incluyen jitter, número de paquetes fuera de secuencia, probabilidad de paquetes perdidos, probabilidad de falla de la red, retraso, y otros. Las métricas para voz incluyen MOS (*Mean Opinion Score*), jitter, retraso, tasa de paquetes perdidos de voz, y otros. Las métricas de QoS para servicios IPTV incluyen disponibilidad del canal, tiempo de inicio del canal, tiempo de cambio del canal, tasa de falla de cambio del canal, y otros. Antes de que el sistema sea desplegado se debe realizar una prueba exacta de rendimiento para probar la alta calidad de los servicios IPTV para servicios de televisión, video, voz, como también servicios interactivos [27].

A continuación se definen dos de las métricas más críticas para QoS tanto para video como para audio en IPTV:

##### Jitter

Es la variación del retraso en la llegada de paquetes con respecto al tiempo de retraso promedio. Los paquetes que llegan a un destino a una tasa constante tienen jitter cero. Los paquetes con una tasa irregular de llegada tienen jitter distinto de cero. Después de atravesar la red, separando la fuente de información de la del destino, enrutados y conmutados por varios elementos de red, es probable que los paquetes lleguen al destino con alguna tasa de variación sobre el tiempo. En cualquier evento, si la tasa instantánea de llegada de información no coincide con la tasa a la cual el destino está consumiendo información, los paquetes deben ser almacenados a su llegada. Se debe implementar lo que en inglés se conoce como jitter-buffer, cuya principal función es almacenar paquetes IPTV hasta que todos los paquetes sean transferidos [28].

##### Pérdida de paquetes

Las pérdidas de paquetes ocurren cuando los paquetes no están alcanzando el destino debido a cualquier fallo en el enlace. En las redes actuales de IP, todos los paquetes de video los cuales llevan información de video son tratados como tráfico de datos, por ejemplo, estos paquetes serán omitidos en caso de congestión en los nodos de red como cualquier otro paquete de datos que no sea de tiempo real (por ejemplo www, e-mail, etc.). Pero el

tráfico que no es de tiempo real no tiene requerimientos estrictos en el retraso de llegada tal como el tráfico de video de tiempo real [28].

La pérdida de paquetes MPEG, los cuales son partes de las tramas “I” o “P” disminuirán más la calidad de video que las tramas de tipo “B”. Cualquier paquete de video IP perdido que contenga un máximo de 7 paquetes MPEG afectará a todos los tipos de tramas, pero la pérdida de una trama tipo “I” afectará a todo el Grupo de Imágenes (GOP, *Group of Pictures*) [29].

#### 2.1.4.3 Calidad de Experiencia (QoE)

La Calidad de Experiencia (QoE, *Quality of Experience*) está definida por la ITU [30] como un mérito común para la calidad de un servicio dado para el usuario final. QoE está constituida de una calidad subjetiva experimentada por el usuario final conocida como *Mean Opinion Score* (MOS), y QoS, designada por la ITU (ITU-T E.800) como el efecto objetivo global de un servicio dado, de la red en ejecución [29].

Los proveedores de IPTV deben tener en consideración que los parámetros de codificación más importantes para QoE son los siguientes:

- Largo del GOP: la ventaja de usar un GOP más largo para IPTV es que se reduce la tasa de bit para el flujo de video, pero desde la perspectiva del consumidor esto causará escenas con mala calidad de video.
- Frecuencia de las tramas de video: Las tasas comunes de las tramas de video son 30 – 60 fps (*frames per second*). Si el proveedor de IPTV utiliza tasas de tramas más bajas afectará la calidad del dinamismo de las escenas.
- Estructura GOP: Si un canal está codificado en video con una estructura GOP “BBBP”, entonces habrá mayor posibilidad de pérdida de paquetes pertenecientes a tramas tipo “B” [28].

#### 2.1.4.4 Estándares, Arquitectura y Operación de IPTV

##### Estandarización de IPTV

Al día de hoy muchas áreas separadas de sistemas IPTV están cubiertas por estándares, en un esfuerzo por traer sistemas IPTV interoperables al mercado. La entrega de servicios multimedia interactivos y personalizados incluyendo servicios IPTV sobre redes de próxima generación (NGN, *Next Generation Networks*) ha sido el objetivo de muchos cuerpos de estandarización tales como ITU-T (*International Telecommunication Union - Telecommunication*) [31], ETSI/TISPAN (*Telecoms and Internet Converged Services and Protocols for Advanced Networks*) [32], *Open IPTV Forum* [33], ATIS (*Alliance for Telecommunications Industry Standard*) [34] y 3GPP [35].

Dos arquitecturas principales han sido aprobadas para la entrega de IPTV, a saber, por la ITU-T y ETSI/TISPAN. El primero de estos organismos formó un Grupo Focal para IPTV (FG IPTV) con la misión de coordinar y desarrollar la arquitectura estándar global para IPTV basada en una arquitectura cliente-servidor con la adición de una plataforma de entrega de servicio mientras se tienen en consideración las siguientes áreas claves: Gestión Digital de Derechos (DRM), Calidad de Servicio/Calidad de Experiencia (QoS/QoE), métricas, metadatos e interoperabilidad [36].

La ETSI/TISPAN surgió con la misión de desarrollar especificaciones para las infraestructuras de redes inalámbricas y combinadas de nueva generación, definiendo a IPTV como un servicio NGN que utiliza el subsistema multimedia IP (IMS, *IP Multimedia Subsystem*). Esta arquitectura puede conectar redes por medio de puertas de enlace que forman parte de IMS o cualquier otra infraestructura basada en Protocolo de Inicio de Sesión (SIP, *Session Initiation Protocol*). De otra parte, ATIS ha lanzado un subgrupo llamado *IPTV Interoperability Forum* para desarrollar principalmente una arquitectura completa de referencia para el despliegue de servicios IPTV enfocándose en cuatro áreas principales: equipo de infraestructura, seguridad de contenido, pruebas de interoperabilidad y QoS. Open IPTV Forum tiene el objetivo de trabajar con otros cuerpos de estandarización para definir especificaciones extremo a extremo para la entrega de servicios IPTV a través de una amplia variedad de distintas arquitecturas de red. Los servicios móviles que están basados completamente en IMS están considerados dentro de Open IPTV Forum y son un conjunto de especificaciones de 3GPP dirigidos a la entrega de servicios multimedia IP para usuarios móviles [36].

Para servicios de transmisión de televisión digital, el grupo *Digital Video Broadcast (DVB)* [37] está desarrollando tecnologías que permitan STB interoperables dentro del grupo DVB-IP [38] para proveer una clase de STB *plug and play* recibiendo el servicio IPTV sobre IP. DVB también ha definido la Plataforma Multimedia de Hogar (MHP, *Multimedia Home Platform*) [39], un middleware basado en Java para IPTV el cual es interoperable con algunos middlewares propietarios de IPTV. En la mejora de compresión, descompresión y codificación de contenido, el *Moving Pictures Experts Group (MPEG)* [40] está trabajando en varias especificaciones relevantes para IPTV, tales como el estándar MPEG-E que comprime muchas aplicaciones de interfaz de programa. En paralelo, MPEG-7 representa características específicas de contenido de bajo nivel, como características visuales (textura, movimiento de cámara) o de audio (melodía) y también estructuras de metadatos para describir el contenido. MPEG-21, también llamado marco multimedia, define un marco abierto para mejorar la gestión de consumo e intercambio de recursos de información digital, apuntando a conseguir funciones como creación y distribución de contenido digital, protección de privacidad de usuario, y extracción de recursos de red [36].

### Arquitectura IPTV

La cadena IPTV se basa en cuatro dominios [36]:

1. El dominio del consumidor presentando servicios al usuario final.
2. El dominio del proveedor de red permitiendo la conexión entre el dominio del consumidor y el dominio del proveedor de servicio.
3. El dominio del proveedor de servicio el cual es responsable de proveer servicios a los consumidores.
4. El dominio que posee o es licenciado para vender contenidos.

En un principio IPTV fue implementado en la forma de televisión digital terrestre y televisión digital satelital. La evolución de redes avanzadas y tecnologías de almacenamiento, junto con el incremento del ancho de banda disponible en las redes de acceso y algoritmos de codificación mejorados han permitido la entrega de contenido de televisión sobre un amplio rango de redes móviles y combinadas. *Kingston Communications*, un operador de telecomunicaciones regional en el Reino Unido, lanzó *Kingston Interactive TV* en Septiembre de 1999, un servicio IPTV de televisión interactiva sobre una infraestructura de red DSL de banda ancha, después de varios intentos de gestión de televisión y VoD. Al final de la

década, algunas compañías comenzaron a transmitir programas de televisión en vivo vía web, y después del año 2000, las compañías empezaron a explorar servicios IPTV y VoD junto con telefonía e Internet de alta velocidad, llamando a este conjunto como “*triple play*”. En este caso el contenido de televisión es entregado al usuario final a través de infraestructuras de cable y línea telefónica. Actualmente un gran número de grandes operadores de servicios de telecomunicaciones están ofreciendo servicios tipo *triple play*. AT&T ofrece más de 300 canales en 11 ciudades. Desde entonces un gran número de sistemas IPTV se han desplegado mundialmente, con millones de usuarios IPTV [36].

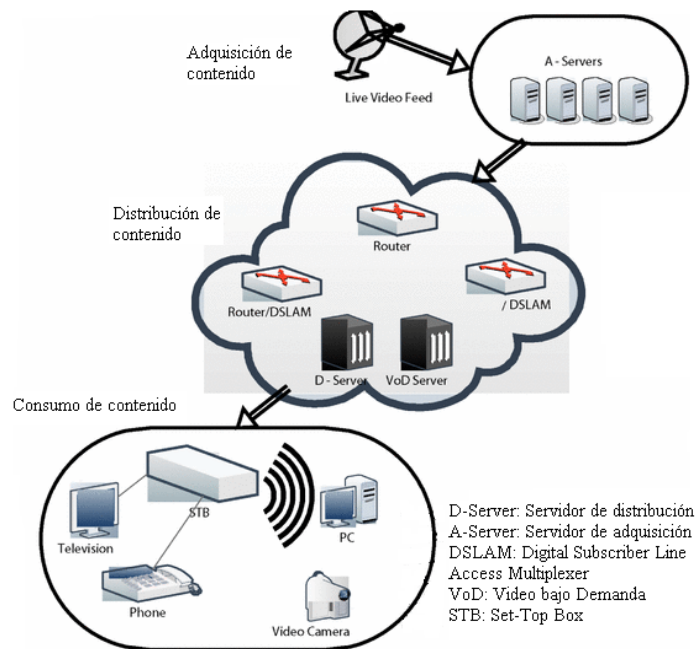
Una infraestructura típica de IPTV consiste de tres grandes bloques, adquisición de contenido, distribución de contenido, y consumo de contenido, construidos en una jerarquía de cobertura nacional, regional y local. Cada parte está implementada con distintos elementos y debe ser capaz de expandirse cuando se necesite. En la Figura 2-1 se muestra una arquitectura básica de IPTV describiendo los grandes componentes de un sistema IPTV típico. Estos componentes incluyen lo siguiente [36]:

1. *Servidores de Adquisición*: Codifican el video y adicionan metadatos.
2. *Servidores de Distribución*: Proveen almacenamiento en caché y control de QoS.
3. *Creadores y servidores de VoD*: Contienen una librería de contenido VoD codificado para proveer servicios VoD.
4. *Routers IP*: Enrutan paquetes IP y proveen un rápido re-direccionamiento en caso de fallas en el enrutamiento.
5. *Puertas de enlace residenciales*: Routers IP para servicios incluidos en casa.
6. *STB*: Un STB es un dispositivo en el lado del consumidor que provee una interfaz al terminal de usuario (TV, PC, Laptop, y otros) por medio de una conexión cable o DSL.

### **Operación de IPTV**

IPTV es actualmente visto como una parte de paquetes *triple play* que son ofrecidos típicamente por operadores de red. IPTV provee los programas tradicionales de televisión lineal en forma de servicios de difusión junto con servicios de VoD y PVR (*Personal Video Recorder*). Más aún, se encuentran algunos servicios de valor agregado como EPG, widgets, información de tráfico vehicular y pronósticos de clima. IPTV permite la entrega de programas de televisión en tiempo real, como también películas, otros tipos de contenido de video interactivo, y servicios multimedia (audio, texto, gráficos, datos) sobre una red basada en IP que provee el nivel requerido de QoS, QoE e interactividad [41].

Un servicio IPTV requiere una cabecera final. Este es el punto en la red en el cual el contenido lineal y bajo demanda es capturado y formateado para distribución sobre la red IP. Típicamente, la cabecera final consume programas lineales vía satélite directamente desde el difusor o programador. Algunos programas también pueden ser consumidos por medio de redes terrestres basadas en fibra óptica. Los codificadores de video y servidores de VoD son las grandes fuentes de contenido de video para servicios IPTV. La cabecera final de video está compuesta de los siguientes componentes [42]:



**Figura 2-1.** Arquitectura básica de IPTV

1. Un codificador de video el cual puede codificar señales analógicas de video en tiempo real desde un proveedor de contenido o una locación de un evento en vivo, a un formato digital basado en una tecnología de compresión de video como MPEG2/4.
2. Un servidor de transmisión de video en vivo el cual encapsula flujos de video que son recibidos con distintos formatos. El servidor también sirve de interfaz para el núcleo de red y transmite la señal de video sobre el núcleo de red hacia la red de acceso.
3. Un servidor de VoD el cual aloja contenido bajo demanda con motores de *streaming* y tiene una gran capacidad de almacenamiento.

Después de codificado, cada canal es encapsulado dentro de paquetes IP y transmitido sobre la red. La red de transporte está dividida en dos grandes partes: *núcleo* y *redes de acceso*. El núcleo conecta a las redes de acceso con el edificio del cliente, y puede ser simplemente una red de distribución nacional sencilla conmutada y varias redes de distribución regionales. El contenido es centralizado y procesado dentro de la red nacional de distribución antes de ser entregado a las diferentes redes de acceso. Las redes de acceso, algunas veces llamadas como "la última milla", y la conexión de banda ancha entre el proveedor de servicio y el usuario final, se pueden llevar a cabo mediante el uso de una variedad de tecnologías (como xDSL y cable híbrido de fibra y coaxial). Las redes IPTV utilizan variantes de DSL asimétrico y DSL de alta velocidad para proveer el ancho de banda requerido para que el cliente pueda correr un servicio IPTV [36].

Dada la limitación del ancho de banda de las redes de acceso, para proveer acceso simultáneo de los canales de televisión a todos los clientes, se ha adoptado ampliamente el multicast para permitir una entrega escalable de información de video para IPTV. La red de hogar distribuye el servicio IPTV a través del hogar. El punto final en la red de hogar, al cual se conectan los dispositivos de usuarios, es el STB. Un STB usualmente está instalado con un middleware cliente para obtener la guía de programación, decodificar información de

video MPEG2 y MPEG4, y desplegarlo en pantalla. Como alternativa, un navegador de Internet puede obtener la guía de programación desde un servidor central. Un STB también puede ser integrado a DSL o cable modem o incluso a un switch IEEE 802.11 para tener acceso a Internet [36].

#### **2.1.4.5 MPEG-2 [43]**

MPEG-2 es uno de los principales estándares para video MPEG. Es usado en una amplia variedad de aplicaciones incluyendo producción para televisión satelital y de cable. Cada día, miles de horas de video MPEG-2 son grabadas, procesadas y reproducidas por emisoras de televisión alrededor del mundo. Además, millones de horas de grabaciones MPEG-2 son vendidas cada día en forma de DVDs.

MPEG-2 soporta señales a 720p y 1080i. También permite la multiplexación de un número de flujos de video y audio, de modo que hace posible aplicaciones multicanal de televisión satelital. MPEG-2 también soporta cinco canales de audio (sonido *surround*) y el estándar *Advanced Audio Coding* (AAC).

Muchos dispositivos MPEG-2, incluyendo codificadores y decodificadores altamente sofisticados, están en su tercera y cuarta generación. Hay literalmente cientos de millones de televisores, STBs, receptores satelitales, y reproductores de DVD instalados en los hogares de consumidores que pueden decodificar señales MPEG-2.

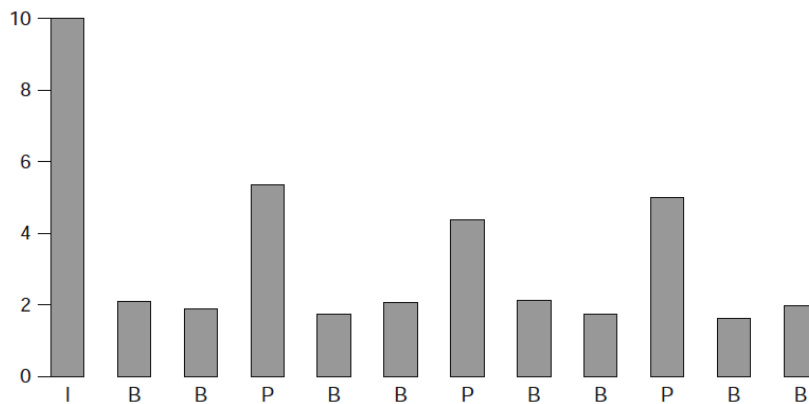
#### **GROUPS OF PICTURES (GOP)**

Los usuarios de cualquier sistema MPEG rápidamente encuentran una variedad de tipos de tramas incluyendo tramas I, P y B, las cuales son llamadas *Grupos de Imágenes* (GOP, *Groups Of Pictures*). Estas tramas describen la forma en la que la información de la imagen está estructurada en un flujo o archivo MPEG.

Una trama es una imagen de una secuencia de video.

- Una trama I es una trama que está comprimida únicamente basada en la información contenida en ella; no hace referencia a ninguna otra de las tramas de video antes o después de ella.
- Una trama P es una trama que ha sido comprimida usando información contenida en la misma trama e información de la trama I o P anterior más cercana.
- Una trama B es una trama que ha sido comprimida usando información de la trama I o P anterior más cercana y de la trama I o P posterior más cercana.
- Un GOP es una serie de tramas consistente de una sola trama I y cero o más tramas P y B. Un GOP siempre comienza con una trama I y termina con la última trama antes de la siguiente trama I. El GOP usualmente es un patrón fijo repetitivo, que está configurado en el dispositivo de compresión. Diferentes proveedores de contenido pueden usar diferentes GOPs para diferentes canales, pero normalmente son fijos dentro de cada canal.

Para entender por qué MPEG utiliza estas diferentes tramas, hay que mirar el tamaño requerido para representar cada tipo de trama. Con una imagen de video de complejidad normal, una trama P tomará de dos a tres veces menos información que una trama I para la misma imagen. Una trama B tomará incluso menos información que una trama P (un factor de entre dos y cinco). La Figura 2-2 muestra los tamaños relativos de información para cada tipo de trama en un GOP típico.



**Figura 2-2.** Tamaños relativos de información para cada tipo de trama MPEG

Un parámetro sobre el cual los proveedores de sistema tienen mucho control es el tamaño del GOP. Escoger el tamaño correcto puede ser un poco controversial.

Hay que recordar que un GOP siempre comienza con una trama I. Para determinar el tamaño de un GOP, simplemente se debe contar el número de tramas B y P entre cada trama I consecutiva y añadir uno para la trama I. Por ejemplo, en la secuencia de tramas mostrada en la Figura 2-2, el tamaño del GOP es 12: una trama I, tres tramas P y ocho tramas B.

Un GOP es considerado pequeño cuando su tamaño está entre tres y cinco. Algunos sistemas utilizan GOPs que son un poco grandes; valores de 15, 30 o incluso 60 se han utilizado en algunas aplicaciones.

Seleccionar un tamaño de GOP adecuado puede tener un gran impacto en una red de video. Muchos factores de rendimiento son afectados por el tamaño del GOP, incluyendo la tasa de bit de flujos codificados, el tiempo en el cambio de canal, y la tolerancia de errores.

#### **2.1.4.6 Multicast**

Multicast entrega información a un grupo de estaciones de destino. IP multicast, mostrado en la Figura 2-3, envía audio, video y datos a múltiples receptores usando protocolos RTP/TCP/IP, con una dirección multicast [27].

Actualmente, IP multicast tiene problemas de escalabilidad cuando hay un gran número de usuarios y grupos. Además, no existe una aplicación multicast totalmente desarrollada en el Internet comercial, con la excepción de algunas aplicaciones limitadas, tales como *Multicast Backbone* (Mbone) y redes IP privadas [27].

Existen tres clases de entrega multicast: multicast denso, multicast disperso, y multicast de fuente específica [27].

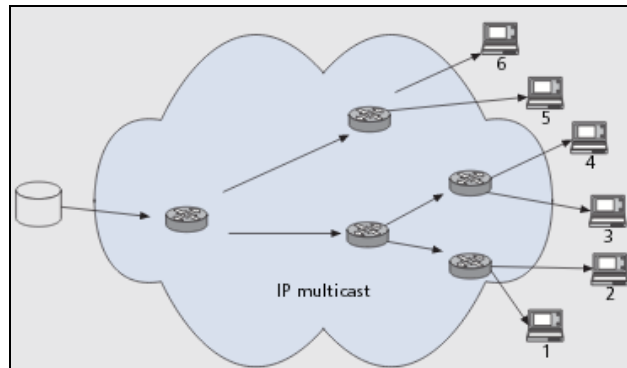


Figura 2-3. IP multicast [27]

- Multicast denso construye un árbol para enviar paquetes a usuarios multicast. Un nodo fuente transmite a todos los routers y nodos.
- Multicast disperso no depende de un protocolo de enrutamiento unicast en particular y se usa para construir un árbol para enviar paquetes a los usuarios multicast. Si un nodo necesita unirse a un grupo multicast envía un mensaje *join/prune* por medio de IGMP a un router, el cual envía paquetes de datos al grupo multicast. Los mensajes *join/prune* son enviados periódicamente a un punto de unión de un grupo específico (RP) por un router designado (DR) para cada grupo activo. Una entrada de ruta es el estado mantenido en un router para representar el árbol de distribución incluyendo dirección de fuente, dirección de grupo, y otros. Este estado crea un árbol de distribución que alcanza a todos los miembros del grupo. Los routers además utilizan un enlace en reversa de reenvío para asegurarse que no existan bucles para el reenvío de paquetes entre routers.
- Multicast de fuente específica entrega paquetes multicast originarios de una dirección de fuente específica a aquellos que lo requieran.

IP multicast es un enfoque para enviar un mensaje simultáneamente a múltiples nodos en vez de solo a uno. Esta tecnología se requiere para proveer servicios IPTV, y ahorrar ancho de banda tanto en las redes núcleo como en las redes de acceso ya que hay una alta probabilidad que más de un usuario mire el mismo programa, y tanto las redes de núcleo como las redes de acceso sean capaces de realizar multicast. Para esta tecnología se requieren tanto IGMP como RTSP [27].

### 2.1.5 Modelado de Tráfico

El modelado de tráfico de una red es el primer paso crítico hacia el entendimiento del potencial y el rendimiento de la misma. Investigaciones extensivas en el área de las redes clásicas tales como Internet, Ethernet, y redes de área local inalámbricas transportando tráfico TCP/IP, HTTP y FTP, han demostrado cómo los modelos de tráfico pueden facilitar el entendimiento de las características del tráfico y manejar una simulación en una etapa temprana para explorar el diseño de una gran red [44]. Un modelo de tráfico describe con qué patrones una fuente genera información (paquetes) que debe ser entregada a través de una red. Con esta descripción de la fuente, una red puede determinar qué tantos recursos de red se necesitan para satisfacer la QoS negociada entre la fuente y la red. Además, el modelado de tráfico es fundamental para el éxito del planeamiento y el diseño de una red [45].



Un flujo de tráfico multimedia generalmente está descrito por un proceso aleatorio estacionario. En la práctica, es difícil desarrollar un modelo de tráfico analítico exacto para su análisis. Solo las estadísticas de tráfico de estado estable, segundo orden y orden superior son apreciables. Las estadísticas de estado estable son definidas por la función de distribución (histograma). Las estadísticas de segundo orden y de orden superior son descritas por funciones de auto-correlación en el estado del tiempo, o por funciones espectrales en el dominio de la frecuencia, tal como el espectro en potencia. Los algoritmos y técnicas conocidas para el procesamiento de señales son útiles para la medición de estadísticas de tráfico. La tarea central de modelado de tráfico es desarrollar metodologías para la construcción de modelos genéricos analíticos que capturen estadísticas importantes de tráfico [46].

Tradicionalmente, el análisis de teletráfico se ha enfocado en obtener una solución analítica para la evaluación del rendimiento de una red (por ejemplo la probabilidad de pérdida de paquetes) [45].

Con el tamaño de las redes creciendo a una tasa asombrosa y el mejor entendimiento del comportamiento estadístico del tráfico de las fuentes existentes (gracias a la disponibilidad de herramientas de medición de tráfico de alta calidad), la aplicabilidad del análisis teletráfico basado en tratabilidad matemática está disminuyendo a una gran velocidad. Consecuentemente, la simulación está creciendo en popularidad como una alternativa para hacer frente al reto de la evaluación de rendimiento, e ingeniería de las redes a gran escala. De hecho, la extrema diversidad de las actuales y futuras aplicaciones de usuario y sus requerimientos de QoS quizás hace a la simulación el único enfoque viable para una ingeniería de tráfico válida y exacta. Comparada a los modelos de tráfico tradicionales, la simulación posee requerimientos considerablemente diferentes para la construcción de modelos de tráfico aceptables [45]:

- *Realista*: El modelo debe ser capaz de representar fuentes de tráfico real. La tratabilidad matemática se mueve a un segundo plano de importancia.
- *Reutilizable*: El modelo debe ser capaz de caracterizar un amplio rango de tráfico variable.
- *Cuidadoso*: El modelo debe ser capaz de capturar características esenciales de fuentes de tráfico que tengan un impacto significativo en el rendimiento de la red con solo un puñado de parámetros.
- *Computacionalmente eficiente*: El modelo debe permitir una rápida simulación.

## 2.2 TECNOLOGÍAS RELACIONADAS

A continuación se realiza la descripción de algunas de las tecnologías necesarias para el completo desarrollo del presente trabajo de grado.

### 2.2.1 LIVE555

El servidor multimedia LIVE555 es un conocido y completo servidor RTSP de fuente abierta y utiliza los protocolos RTSP, RTP y SDP (*Session Description Protocol*) para el streaming multimedia y es compatible con reproductores multimedia como VLC y QuickTime. LIVE555 es una aplicación de fuente abierta, su código fuente está disponible y puede ser modificado para cumplir con requerimientos específicos [47].

LIVE555 puede generar varios tipos de flujos de archivos multimedia como son [48]:

- TransportStream de MPEG (.ts)
- WebM o Matroska (.webm o .mkv)
- MPEG-1, MPEG-2 (.mpeg)
- MPEG-4 (.m4e)
- H.264 (.264)
- DV (.dv)
- MP3 (.mp3)
- WAV (.wav)
- AMR (.amr)
- AAC (.aac)

Estos flujos pueden ser recibidos y/o reproducidos por cualquier cliente multimedia RTSP/RTP que obedezca a los estándares, entre ellos se incluyen [48]:

- VLC Media Player
- QuickTime Player
- Amino Set-Top Boxes (solamente para la reproducción de archivos .ts)
- OpenRTSP (el cual recibe y/o almacena datos de flujos sin tener que reproducirlos)

### 2.2.2 FFMPEG

FFmpeg es un sistema multimedia de fuente abierta (disponible para que cualquier persona interesada pueda ayudar en su desarrollo) que permite decodificar, codificar, transcodificar, multiplexar, demultiplexar, transmitir, filtrar y reproducir archivos de audio y video. El proyecto FFmpeg intenta proveer la mejor solución técnica posible para desarrolladores de aplicaciones y usuarios finales [49].

FFmpeg provee varias herramientas [49]:

- ffmpeg es una herramienta que permite convertir archivos multimedia entre formatos.
- ffmpeg es un servidor de transmisión multimedia para transmisiones en vivo.
- ffmpeg es un sencillo reproductor multimedia basado en SDL<sup>1</sup> (*Simple Directmedia Layer*) y librerías FFmpeg.
- ffmpeg es un sencillo analizador de flujo multimedia.

Y librerías:

- libavutil es una librería que contiene funciones para simplificar la programación, incluyendo generadores de números aleatorios, estructuras de datos, rutinas matemáticas, utilidades multimedia, y muchas más.
- libavcodec es una librería que contiene decodificadores y codificadores para codecs de audio/video.
- libavformat es una librería que contiene demultiplexores y multiplexores para formatos contenedores multimedia.
- libavdevice es una librería que contiene dispositivos de entrada y salida para la grabación desde muchos software multimedia de entrada/salida como Video4Linux, Video4Linux2, Vfw, y ALSA.
- libavfilter es una librería que contiene filtros multimedia.

## ffmpeg

Para propósitos del presente trabajo de grado únicamente se utilizará la herramienta ffmpeg. A continuación se presenta una breve descripción de esta herramienta.

ffmpeg es un rápido convertidor de audio y video que también puede grabar desde una fuente en vivo de audio/video. También puede convertir entre tasas de muestreo arbitrarias y redimensionar archivos de video.

ffmpeg puede leer un número arbitrario de archivos de entrada (los cuales pueden ser archivos regulares, pipes, flujos de red, dispositivos de grabación, etc.), y escribir a un número arbitrario de archivos de salida, los cuales son especificados por un nombre de archivo de salida [50].

### 2.2.3 OpenRTSP

OpenRTSP es un programa de línea de comandos que puede ser utilizado para abrir, generar, recibir y opcionalmente grabar flujos multimedia que son especificados por un URL RTSP, por ejemplo, un URL que comience con rtsp:// [51].

### 2.2.4 VLC Media Player

VLC es un reproductor de multimedia gratuito y de fuente abierta que es capaz de reproducir la mayoría de archivos multimedia tales como DVD, CD Audio, VCD y varios protocolos de streaming [52].

### 2.2.5 Wireshark

Wireshark es un analizador de protocolos gratuito y de fuente abierta. Es utilizado para la solución de problemas de redes, análisis de software y desarrollo de protocolos de comunicación, entre otros. Su nombre original era *Ethereal*, pero por problemas de *trademark*, en Mayo de 2006 cambiaron su nombre a Wireshark.

Wireshark es un software multiplataforma, el cual es posible ejecutar en varios sistemas operativos Unix como Linux, OS X, BSD y Solaris, además de Microsoft Windows. Wireshark, al igual que los demás programas distribuidos junto con él como TShark, son gratuitos, publicados bajo los términos de la licencia GNU *General Public License* [53].

### 2.2.5 OPNET Modeler

OPNET Modeler es un software para el análisis y diseño de redes, dispositivos, protocolos, y aplicaciones de comunicaciones. Los usuarios pueden analizar redes simuladas para comparar el impacto de diferentes tecnologías sobre el comportamiento extremo a extremo. OPNET Modeler incorpora un amplio rango de protocolos y tecnologías, e incluye un ambiente de desarrollo que permite desarrollar toda clase de redes y tecnologías incluyendo [54]:

- VoIP
- TCP
- OSPFv3
- MPLS
- IPv6

## Principales Características [54]

- Motor de simulación de eventos discretos más rápido
- Cientos de protocolos y modelos de dispositivos con código fuente
- Modelado orientado a objetos
- Ambiente de modelado jerárquico
- Simulación de eventos discretos, híbrida, y opcionalmente analítica
- Kernel de simulación de 32 y 64 bits completamente paralelos
- Soporte computado de red para simulación distribuida
- System-in-the-loop opcional para interfaz de simulaciones con sistemas en vivo
- Aplicación de modelado y análisis realista
- Interfaz abierta para integrar archivos externos, librerías y otros simuladores
- Depuración y análisis basados en Interfaz Gráfica de Usuario (GUI, *Graphic User Interface*)

### 2.2.6 ns-3

ns-3 es un simulador de redes de eventos discretos, cuyo principal objetivo es la investigación y el uso educacional. ns-3 es un software gratuito, licenciado bajo la licencia GNU GPLv2, y está disponible públicamente para investigación, desarrollo y uso.

La meta del proyecto ns-3 es desarrollar un ambiente de simulación abierto preferente para la investigación de redes: debe estar alineado con las necesidades de simulación de investigación de redes modernas y debe fomentar la contribución en comunidad, revisión de puntos, y validación del software [55].

### Modelos de Simulación

El proyecto ns-3 está comprometido a construir un núcleo de simulación sólido que esté bien documentado, fácil de usar y depurar, y que abastezca las necesidades de todo el flujo de trabajo de simulación, desde la configuración de la simulación hasta la colección y análisis de trazas.

Además, la infraestructura del software ns-3 fomenta el desarrollo de modelos de simulación los cuales son suficientemente realistas para permitir que ns-3 sea usado como un emulador de red en tiempo real, interconectado con el mundo real y el cual permite muchas implementaciones de protocolos del mundo real para ser rehusadas dentro de ns-3.

El núcleo de simulación ns-3 soporta investigación tanto en redes basadas en IP como en no basadas en IP. Sin embargo, la gran mayoría de sus usuarios se enfocan en simulaciones inalámbricas/IP lo cual involucra modelos para Wi-Fi, WiMAX, o LTE para las capas 1 y 2 y una variedad de protocolos de enrutamiento estáticos y dinámicos tales como OLSR (*Optimized Link State Routing*) y AODV (*Ad hoc On-Demand Distance Vector Routing*) para aplicaciones basadas en IP [54].

### 2.2.7 @RISK [56]

@RISK es un software que realiza análisis de riesgo usando la simulación para mostrar múltiples resultados posibles en un modelo de hoja de cálculo, e indica qué probabilidad hay de que se produzcan.

@RISK es un programa con licencia, propiedad de la empresa *PALISADE* (fabricante de software de análisis de riesgo y de decisiones). Para este trabajo de grado se ha utilizado la versión de prueba por 30 días de @RISK disponible en [56].

Esta herramienta va a ser utilizada en el desarrollo del presente proyecto con el fin de obtener la Funciones de Distribución de Probabilidad asociadas a los datos del tráfico capturado en laboratorio. Esto se verá en detalle en el capítulo 4.

## 2.3 TRABAJOS RELACIONADOS

A continuación se describen los trabajos relacionados con el campo del modelado de tráfico que tienen mayor relevancia en el desarrollo del presente trabajo de grado.

Se ha prestado especial interés en los trabajos de investigación de Walter Willinger, un miembro de los laboratorios de investigación en informática y sistemas software de AT&T en Florham Park, New Jersey, USA, debido a que es uno de los pioneros en cuanto al desarrollo de procesos matemáticos que describen el comportamiento del tráfico en una red a gran escala. Este investigador ha tenido una gran influencia en el mundo del teletráfico gracias a trabajos como “*On the Self-Similar Nature of Ethernet Traffic*”, que está incluido en “*The Best of the Best- Fifty Years of Communications and Networking Research*”, un libro del año 2007 de la sociedad de comunicaciones IEEE (*Institute of Electrical and Electronics Engineers*) que compila los artículos más importantes publicados en el campo de las comunicaciones y el estudio de redes en los últimos 50 años [57].

Para la descripción de cada uno de los siguientes trabajos relacionados se hará uso del siguiente patrón:

- **Nombre:** Nombre completo de la publicación en su idioma original.
- **Año:** Año de la publicación.

### 2.3.1 On the Self-similar Nature of Ethernet Traffic [58]

**Año:** 1994

**Aportes:** En este artículo se demuestra que el tráfico en una red de área local (LAN) es estadísticamente auto similar. Además, ninguno de los modelos usados anteriormente fue capaz de capturar este comportamiento fractal, tal comportamiento tiene serias implicaciones para el diseño, el control y el análisis de alta velocidad, redes basadas en células y flujos que se agregan típicamente intensifican la auto-similaridad (explosividad) en vez de suavizarlo.

De este artículo se extrajo la definición y estudio de auto-similaridad para tráfico de redes, con lo que se estudió la viabilidad de implementar un algoritmo que midiera este parámetro en el simulador de redes.

### 2.3.2 Long-Range Dependence in Variable-Bit-Rate Video Traffic [59]

**Año:** 1995

**Aportes:** En este artículo se analizan videos de tasa de bit variable, generados por una variedad de codecs diferentes y siendo representados en varios escenarios. Al desarrollar

las pruebas estadísticas y gráficas, se concluye que la dependencia de rango largo (*long-range dependent processes*) es una característica inherente de los videos de tasa de bit variable, independientemente del escenario (video llamada, video conferencia, etc.) y del códec utilizado.

De este artículo se tuvo en cuenta el estudio realizado al tráfico de videos de tasa de bit variable y con esto hacer un análisis comparativo con los videos utilizados para el trabajo de grado.

### 2.3.3 Experimental Queueing Analysis with Long-Range Dependent Packet

#### Traffic [60]

Año: 1996

**Aportes:** En este documento se demuestra empíricamente que mas allá de su significado estadístico en la medición de tráfico, la dependencia de rango largo tiene un impacto considerable en la realización de encolado y es una característica dominante de un número de problemas de ingeniería de tráfico de paquetes (buffer sizing, control de admisión y control de tasa).

Se tomaron de este artículo los resultados obtenidos para el análisis de los problemas producidos en una red por el encolado de tráfico.

### 2.3.4 Self-similar Traffic Parameter Estimation: A Semi-parametric

#### Periodogram-Based Algorithm [61]

Año: 1996

**Aportes:** En este artículo se hace uso de un algoritmo basado en periodograma semi-paramétrico (SPA, *Semi-parametric Periodogram-Based Algorithm*) para estimar el parámetro de Hurst y los picos. El algoritmo está basado en el hecho que la densidad espectral de potencia (PSD, *Power Spectral Density*) de procesos con dependencia de rango largo obedecen a la ley de las potencias cerca del origen. El SPA es aplicado para estimar los parámetros de las trazas FBM (*Fractional Brownian Motion*) tanto como de las trazas reales. Los resultados obtenidos del análisis de medida de paquetes muestran que el tráfico de paquetes exhibe propiedades fractales tales como la auto-similaridad, los cuales son fundamentalmente diferentes de las características encontradas en el tráfico de voz por conmutación de circuitos y capturados comúnmente mediante modelos de tráfico de paquetes tales como el *Batch-Poisson* y *Markov Modulated Poisson Process* (MMPP).

### 2.3.5 Self-Similarity Through High-Variability: Statistical Analysis of Ethernet

#### LAN Traffic at the Source Level [62]

Año: 1997

**Aportes:** En este artículo se expone una explicación física plausible para la ocurrencia de la auto-similaridad en tráfico en una red de área local (LAN, *Local Area Network*). La explicación está basada en la convergencia de nuevos resultados para los procesos que exhiben alta variabilidad (infinita varianza) y es soportada por análisis estadísticos detallados de medidas de tráfico de tiempo real desde una red de área local Ethernet al nivel de fuentes

individuales. Adicionalmente se desarrollan resultados matemáticos concernientes a la superposición de fuentes ON/OFF estrictamente alternantes.

### 2.3.6 Self-Similar Traffic and Network Dynamics [63]

**Año:** 2002

**Aportes:** En este artículo se revisa lo que se conoce en el momento acerca de la auto-similaridad de tráfico de red y su significancia. Se considera un asunto de estudio actual, porque la forma en la cual las dinámicas de red (especialmente, las dinámicas del protocolo de control de transporte, TCP), puede afectar la auto-similaridad observada. Para este fin, primero se discuten algunos de los problemas asociados con aplicar técnicas tradicionales de evaluación de desempeño para redes a gran escala y altamente cargadas como Internet.

Los artículos [61], [62] y [63] se estudiaron con la intención de determinar que tan factible es realizar mediante simulación la medición de la auto-similaridad de los datos obtenidos para el tráfico de los videos utilizados en el proyecto de grado.

### 2.3.7 Queuing performance comparison of traffic models for Internet traffic [64]

**Año:** 1998

**Aportes:** En este artículo se compara la predicción de desempeño de una variedad amplia de modelos de tráfico para un número de medidas de desempeño que son comunes en la ingeniería de tráfico. En particular se investigan los modelos de Piosson, 2-state MMPPP, AR(1), Weibull, Pareto y fBm (*fractional Brownian motion*). Los resultados numéricos llevan a la conclusión de que la elección de un modelo en particular no solo depende del tipo de tráfico sino también del la aplicación/dispositivo considerado en términos de buffering.

Este artículo se utilizó como argumento para la escogencia de las distintas funciones de distribución para las componentes de los videos del trabajo de grado.

### 2.3.8 MPEG-encoded variable bit-rate video traffic modelling [65]

**Año:** 2005

**Aportes:** Este artículo está enfocado para proponer un modelo de tráfico de video acertado para video de tasa de bit variable codificado con MPEG. Un método multinomial es propuesto para capturar la correlación entre los tipos de tramas. La utilidad de este método se demuestra de dos maneras separadas. La primera usa una combinación del método multinomial y el proceso de renovación espacial para modelar el tráfico de video MPEG VBR. La segunda manera usa el método multinomial para mejorar la desempeño del modelo existente.

### 2.3.9 Modeling Internet Traffic: Performance Limits [66]

**Año:** 2006

**Aportes:** En este artículo se presenta la metodología de modelado de tráfico mediante la agregación de bytes usando el proceso M/G/infin para la captura de correlaciones de tráfico. Utiliza trazas de tráfico para construir modelos de tráfico M/G/infin, y evalúa el tráfico generado tanto estadísticamente como también mediante una simulación en un ambiente de

red. También muestra las limitaciones de los modelos de agregación de bytes, y concluye algunos aspectos para obtener modelos de tráfico más exactos.

### **2.3.10 Modeling of MPEG-4 Video Traffic Using a Multifractal Cascade with Autoregressive Multipliers [67]**

**Año:** 2011

**Aportes:** En este artículo se propone un modelo de tráfico de video basado en una cascada multi-fractal multi-aplicativa. Se asume que los multiplicadores en cascada pueden ser eficientemente descritos por modelos auto-regresivos en cada escala de la cascada. Este modelo de cascada multiplicativa basado en proceso autoregresivo es usado para el modelado de tráfico al nivel del GOP (*Group of Pictures*) como también al nivel de tramas. Con la validación del modelo, se realizó un análisis estadístico y de comportamiento de colas de los procesos de tráfico de video generados usando el modelo propuesto. También se verificó el rendimiento del modelo en la captura de características de tráfico real de video MPEG-4 comparándolo con otros modelos multi-fractales.

### **2.3.11 Multi-Service traffic models of heterogeneous wireless communication networks [68]**

**Año:** 2008

**Aportes:** Un buen modelo de tráfico generalmente mejora la eficiencia del espectro en redes inalámbricas. Los modelos de tráfico convencionales, como la distribución Poisson, solo se adaptan para tráficos tradicionales, como el de voz. Con la emergencia de varias clases de tráfico, especialmente los servicios de datos de paquetes conmutados de 3G/B3G en redes de comunicaciones inalámbricas heterogéneas, han aparecido nuevos modelos de tráfico. La dependencia de rango largo o de auto-similaridad y multi-fractalidad son las distintas características de estos nuevos tráficos. En este artículo se analiza el principio y las diferencias de estos modelos y propone un algoritmo óptimo de modelado de tráfico, llamado Modelo de Cascadas Divisibles Infinitamente (IDC, *Ininitely Divisible Cascades*). Este artículo presenta también temas adicionales en el modelado de tráfico multi-servicio para redes de comunicaciones inalámbricas heterogéneas.

### **2.3.12 Using Hidden Markov Chains for Modeling P2P-TV Traffic [69]**

**Año:** 2010

**Aportes:** En este artículo se estudia el tráfico generado por PPLive y SopCast, que están entre las aplicaciones más populares P2P-TV de hoy en día, y se proponen cadenas ocultas Markov para modelar el tráfico que ellos generan. Los resultados muestran que los modelos son bastante exactos y pueden ser usados efectivamente en muchas tareas de red como en el análisis de rendimiento de redes, dimensionamiento y planeamiento de redes e ingeniería de tráfico.

### **2.3.13 Modeling Video on Demand Services taking into account statistical dependences in user behavior [70]**

**Año:** 2013



**Aportes:** Este artículo presenta la implementación de un modelo de simulación de un servicio de VoD. El modelo está desarrollado teniendo en cuenta tanto el comportamiento estadístico del usuario como la carga de trabajo del servidor. Mientras el primero es analizado a través de un servicio real de VoD, el segundo es analizado por medio de experimentos en laboratorio. Sin embargo, la principal contribución del modelo es que también considera las dependencias entre interacciones de usuarios. El modelo ha sido validado usando información real, y evaluado en diferentes situaciones. Los resultados muestran que el rendimiento es influenciado tanto por accesos masivos a los contenidos, como por la estructura de dependencia entre interacciones de usuarios.

Los artículos [8], [9], [10], [11], [12] y [13] se usaron como base para el desarrollo de una metodología para la creación del modelo de tráfico propuesto en el trabajo de grado.

### **2.3.14 Limitations of Network Emulation with Single-Machine and distributed ns-3 [71]**

**Año:** 2010

**Aportes:** En este artículo se examina el simulador/emulador de redes ns-3, con el fin de determinar su capacidad en la evaluación de servicios a larga escala. Para este propósito, se desarrolla un servicio real de video cliente/servidor sobre una red emulada. El servicio se escala progresivamente mediante el aumento del número de clientes en una máquina sencilla. En adición, se extiende ns-3 para soportar una arquitectura distribuida de nodos de red, de esta manera, se repiten los experimentos con una configuración distribuida.

De este artículo se tomó parte del trabajo realizado en el simulador de redes ns-3, la cual fue útil para la selección de la aplicación utilizada en el modelo desarrollado para este simulador.

## **CAPITULO 3**

# **ARQUITECTURA DE REFERENCIA Y CARACTERIZACIÓN DE TRÁFICO REAL**

En el presente capítulo se describe la arquitectura del laboratorio de Televisión Digital de la Universidad del Cauca en el cual se lleva a cabo el desarrollo del proyecto ST-CAV. Además se hace referencia a la caracterización estadística del tráfico real capturado en laboratorio para el escenario de IPTV.

Dentro del proyecto ST-CAV se desplegaron dos escenarios para la transmisión de contenidos por Internet. El primer escenario, desplegado por los ejecutores de este proyecto, corresponde a un escenario de IPTV sobre el que se hicieron pruebas de transmisión de contenidos mediante el servidor Live555, por ser este servidor capaz de soportar MPEG-2, que es uno de los estándares de codificación de audio y video utilizados para IPTV [33] (además de ser soportado por el STB Amino, el cual no soporta el estándar MPEG-4), y el protocolo RTSP, usado para el inicio y control de la sesión de streaming en el servicio de VoD sobre IPTV. El segundo escenario es el de transmisión de contenidos y consumo de servicios mediante HTTP, el cual no cumple con los estándares de IPTV al no usar el protocolo RTSP ni los estándares de codificación de audio y video propios de esta tecnología, además de tratar a los contenidos multimedia como cualquier recurso más accedido por HTTP.

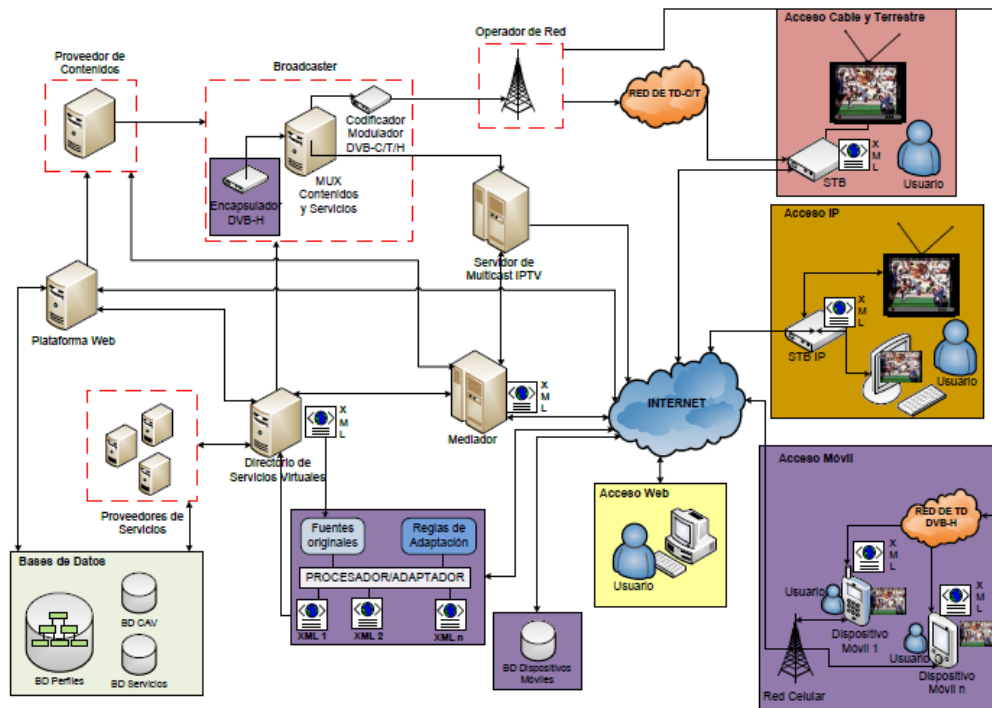
### **3.1 ARQUITECTURA DEL PROYECTO ST-CAV [72]**

La Figura 3-1 muestra el diagrama de la vista de despliegue de la arquitectura del Laboratorio de Televisión Digital de la Universidad del Cauca. De acuerdo a como se definió en los requisitos del proyecto ST-CAV [4], este tiene dos escenarios: Broadcast e IPTV con el servicio de VoD, además del acceso web en el cual también se implementó el servicio de VoD.

En el primero de los escenarios, los contenidos que son transmitidos a través del canal de broadcast, son almacenados y gestionados por el proveedor de contenidos. En él están disponibles los contenidos generados por los usuarios y la programación a transmitir. Los servicios y aplicaciones con los cuales los usuarios podrán interactuar están disponibles en un conjunto de proveedores de servicios. Para que estos servicios puedan ser utilizados por ST-CAV, deben ofrecerse a través de servicios WEB o con estándares XML, es decir, todas las respuestas de invocación a los servicios deben cumplir con el formato XML siguiendo la estructura definida en el proyecto ST-CAV para que los dispositivos puedan interpretar y representar la información según corresponda. Cuando estos servicios y/o aplicaciones son asociados al contenido, la información de los mismos es adicionada como un flujo que es

multiplexado en el *carrusel de objetos*<sup>2</sup>, permitiendo que los usuarios puedan acceder a ellos desde los diferentes terminales (STB-C/T, dispositivo móvil). Después de que los diferentes flujos de audio, video y datos son multiplexados entran en una etapa de modulación que los convierte en una señal adecuada para cada uno de los medios (cable, terrestre y móvil).

Para el caso de IPTV la señal no es modulada, después de ser multiplexada entra a una adaptación y empaquetamiento para enviar la información a través de un multicast IP.



**Figura 3-1.** Arquitectura ST-CAV – Vista de despliegue [72]

Cuando los servicios y/o aplicaciones no están asociados al contenido, los usuarios podrán acceder a ellos a través del canal de retorno. Esta petición llega al Mediador, el cual está encargado de interpretar los mensajes XML que recibe de los usuarios y solicitar el servicio al Directorio Virtual de Servicios. Además, mantiene una relación directa con sus proveedores para acceder a los servicios y genera el archivo XML adecuado que permitirá a los usuarios acceder al servicio.

<sup>2</sup> El carrusel de objetos DSM-CC (*Digital Storage Media-Command Control*), es un estándar de difusión de datos que permite tanto transmitir aplicaciones y datos MHP (*Multimedia Home Platform*) vía un canal de broadcast en forma de carrusel, como sincronizar estas aplicaciones con el audio y el video de un servicio mediante los flujos de eventos.

El acceso web permite a un usuario acceder a las funcionalidades de la plataforma web, en la cual son creados los contenidos interactivos de televisión digital, se genera la parrilla de programación y se realiza toda la gestión de las CAV. La plataforma web presta todas las funcionalidades para gestionar las CAV, los usuarios y sus perfiles, las recomendaciones de contenidos, comunidades, etc. Algunas de estas funcionalidades están disponibles en la web y otras son accedidas por los usuarios desde los entornos de televisión por cable, terrestre, móvil e IP.

En el servicio de VoD los usuarios buscan los contenidos que desean visualizar por medio del canal de retorno. Estos contenidos son enviados por el mismo canal, no se utiliza multicast IPTV. Este servicio únicamente está disponible para el escenario de IPTV.

## **3.2 TOPOLOGÍAS PARA CASOS DE ESTUDIO**

En el presente proyecto se lleva a cabo el estudio del tráfico generado por los servicios de una CAV. Este tráfico es tomado de la experimentación real en laboratorio para luego poder evaluar este mismo tráfico en distintas topologías de red por medio del uso de herramientas de simulación. Teniendo en cuenta las características de los equipos disponibles en el laboratorio y los recursos necesarios para el correcto funcionamiento de las simulaciones en cada software de simulación, se han diseñado tres distintas topologías de red en las cuales se evaluarán parámetros de rendimiento de red al hacer fluir en ellas el tráfico real conseguido con anterioridad. El desarrollo de esta experimentación se documentará en detalle en el Capítulo 4 del presente documento.

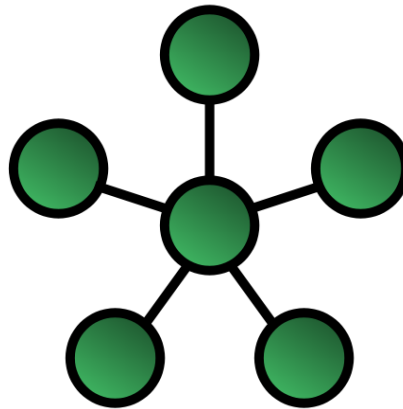
A continuación se describen las topologías diseñadas para cada uno de los casos de estudio.

### **3.2.1 Caso de estudio 1. Topología en estrella**

Para el primer caso de estudio se utilizará una topología en estrella, tal como se puede observar en la Figura 3-2.

Una red con topología en estrella es una LAN en la cual todos los nodos (computadores u otros dispositivos) están directamente conectados a un computador, router o switch central. Cada nodo de la red está indirectamente conectado con todos los demás a través del nodo central.

Las topologías en estrella previenen la transmisión de paquetes a través de un excesivo número de nodos. La mayoría del tiempo, en la comunicación entre dos nodos, tan solo están involucrados un máximo de tres dispositivos y dos enlaces.

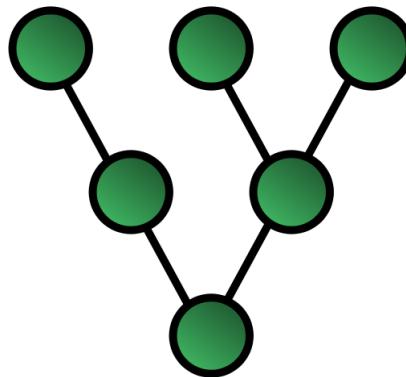


**Figura 3-2.** Topología en estrella

Debido a que en esta topología el nodo central es el “cuello de botella”, al incrementar su capacidad, o conectar dispositivos adicionales al mismo, el tamaño de la red es incrementado fácilmente. La centralización también permite la inspección del tráfico a través de toda la red, esto facilita el análisis del tráfico y la detección de comportamiento sospechoso [73].

### **3.2.2 Caso de estudio 2. Topología en árbol**

Para el segundo caso de estudio se ha seleccionado la topología en árbol mostrada en la Figura 3-3.

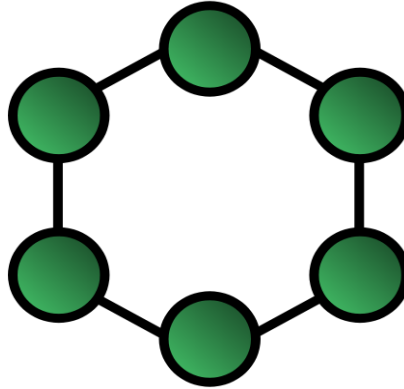


**Figura 3-3.** Topología en árbol

Este tipo particular de topología de red está basado en nodos hereditarios. El nivel más alto de una red en árbol consiste de un nodo “raíz”, este nodo está conectado a uno o más nodos en el nivel inferior siguiente por medio de enlaces punto a punto. Estos a su vez están conectados a uno o varios nodos en el nivel inmediatamente inferior por medio de enlaces punto a punto. Las redes en árbol no están limitadas a un número de niveles, sin embargo, al ser redes basadas en la topología de red tipo bus, son propensas a fallas de red debido al mal funcionamiento de nodos en los niveles superiores [73].

### 3.2.3 Caso de estudio 3. Topología en anillo

En este último caso de estudio se utiliza una topología de red tipo anillo. En la Figura 3-4 se muestra una representación de la misma.



**Figura 3-4.** Topología en anillo

Una topología de red en anillo se caracteriza por estar configurada en forma circular. En una topología de red en anillo la información viaja alrededor del anillo en una dirección y cada dispositivo actúa como un repetidor para mantener el nivel de potencia de la señal. Cada dispositivo incorpora un receptor para la señal de entrada y un transmisor para enviar la información al siguiente dispositivo en el anillo. La red depende de la capacidad de la señal para viajar a través de cada dispositivo. Cada nodo es un enlace crítico [73].

## 3.3 CARACTERIZACIÓN DE TRÁFICO REAL CAPTURADO EN LABORATORIO

En el presente apartado se describe el proceso realizado para la caracterización del tráfico capturado en laboratorio, para el servicio de VoD. Además se muestran los resultados encontrados.

Para este proyecto se utilizaron tres videos alojados en el servidor de contenidos de la arquitectura relacionada en el apartado 3.1. Estos videos poseen características distintas en cuanto a duración, movilidad, etc., para poder realizar un análisis comparativo entre ellos.

### 3.3.1 Captura de tráfico real producido por el servicio de VoD de la CAV sobre IPTV

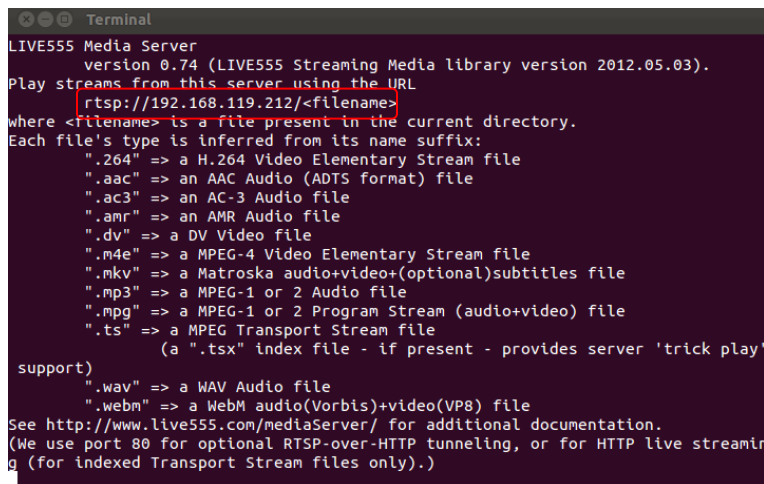
#### 3.3.1.1 Transmisión y reproducción de video

El desarrollo del presente proyecto se ha venido realizando en el Laboratorio de Televisión Digital de la Universidad del Cauca. En este laboratorio se han desarrollado distintos proyectos (entre ellos, ST-CAV) que han dado como resultado la creación de varios servicios y aplicaciones que soportan una CAV. Uno de estos servicios (siendo el más importante en cuanto a consumo de recursos se refiere) es el de VoD; este servicio está disponible para

ser consumido vía Internet por medio del uso de una plataforma web o también vía IPTV directo a un televisor con el uso de un STB (para más detalles ver el apartado 3.1), en el presente proyecto se utilizará tanto el acceso IPTV como el web (ver Figura 3-1).

Para lograr capturar el tráfico producido por el servicio de VoD en una red IPTV, es necesario tener un servidor y un usuario. El servidor se encarga de transmitir la información para que esta pueda ser consumida por el cliente, el cuál tiene acceso a la red a la que está conectado el servidor. Por su parte, el cliente se encarga de consumir el servicio desde el servidor. Esto se logra mediante el uso del protocolo de transporte RTSP.

Para el caso de acceso web, se necesita tener dos equipos. Uno de ellos con un software que permita consumir el servicio del servidor. Para el presente proyecto se ha utilizado el reproductor multimedia VLC, este reproductor es bastante útil para este propósito por su versatilidad y facilidad de utilización. El otro equipo debe tener instalado un software que permita la transmisión de los datos de audio y video para que puedan ser consumidos por los clientes. Este software debe ser útil para que el servicio pueda ser consumido tanto vía web como IPTV ya que este servidor se utilizará para ambos casos. Para este propósito se utilizó el servidor LIVE555 (este software está descrito en detalle en el Capítulo 2, en la sección “Tecnologías relacionadas”, además en el Anexo C se encuentra detallada su instalación y configuración sobre un sistema operativo Ubuntu), el cual se puede descargar gratuitamente desde su página web [50], sobre un sistema operativo Ubuntu (se escogió este software debido a que soporta formatos de IPTV como MPEG-2, e incluso H.264). En la Figura 3-5 se muestra una imagen del funcionamiento del servidor LIVE555 en una consola de Ubuntu. En esta imagen se puede observar que el servidor despliega la IP del equipo en el que está alojado y el protocolo por medio del cuál realiza la transmisión; esta información es importante para que los clientes puedan conectarse satisfactoriamente al servidor y consumir la información que este está transmitiendo.



```
Terminal
LIVE555 Media Server
  version 0.74 (LIVE555 Streaming Media library version 2012.05.03).
Play streams from this server using the URL
  rtsp://192.168.119.212/<filename>
where <filename> is a file present in the current directory.
Each file's type is inferred from its name suffix:
  ".264" => a H.264 Video Elementary Stream file
  ".aac" => an AAC Audio (ADTS format) file
  ".ac3" => an AC-3 Audio file
  ".amr" => an AMR Audio file
  ".dv"  => a DV Video file
  ".m4e" => a MPEG-4 Video Elementary Stream file
  ".mkv" => a Matroska audio+video+(optional)subtitles file
  ".mp3" => a MPEG-1 or 2 Audio file
  ".mpg" => a MPEG-1 or 2 Program Stream (audio+video) file
  ".ts"  => a MPEG Transport Stream file
           (a ".tsx" index file - if present - provides server 'trick play'
           support)
  ".wav" => a WAV Audio file
  ".webm" => a WebM audio(Vorbis)+video(VP8) file
See http://www.live555.com/mediaServer/ for additional documentation.
(We use port 80 for optional RTSP-over-HTTP tunneling, or for HTTP live streaming
  (for indexed Transport Stream files only).)
```

Figura 3-5. Servidor LIVE555 en funcionamiento

Para el consumo de este servicio por medio de una red IPTV se utilizó un STB de marca Amino conectado a un televisor. Este STB en particular necesita que los archivos consumidos estén en un formato llamado MPEG-TS (*MPEG-Transport Stream*) cuya extensión es .ts. Para obtener este tipo de archivo se hizo uso del software ffmpeg (descrito en detalle en la sección “Tecnologías relacionadas” del Capítulo 2 y en el Anexo C). Con este software se logra cambiar la codificación y el formato del audio y video del archivo así como también otros parámetros como resolución, relación de aspecto, tasa de bit, etc. Mediante el uso de este software se cambiaron los codecs de los archivos para experimentación de tal forma que coincidieran con los utilizados en IPTV y además soportados por el STB utilizado, estos son: mpeg-2 para video y aac para audio. En la Figura 3-6 se muestra una captura de pantalla tomada al proceso de codificación y compresión realizado a un archivo de video llamado Video1.wmv mediante el uso de ffmpeg. Se puede apreciar el cambio de codecs, en video de wmv2 a mpeg2video, y en audio, de wmv2 a libfaac, además se empaqueta el archivo en un flujo de transporte TS. Todo esto se ha realizado en un sistema operativo Ubuntu.

Además de esto, el archivo .ts tiene que ser indexado para que el usuario pueda pausar, adelantar y retroceder el video. Para tal fin se utilizó el software Universal Indexer de LIVE555. En la Figura 3-7 se muestra una captura de pantalla tomada al proceso de indexación realizado al video relacionado anteriormente.

```
root@stcav-Studio-540: ~/live/mediaServer
kb/s
[buffer @ 0x2865ac0] w:1920 h:1080 pixfmt:yuv420p tb:1/1000000 sar:0/1 sws_param:
[scale @ 0x28c2440] w:1920 h:1080 fmt:yuv420p -> w:720 h:576 fmt:yuv420p flags:0x4
[mpeg2video @ 0x28c1b20] Warning min_rate > 0 but min_rate != max_rate isn't recomme
nded!
[mpeg2video @ 0x28c1b20] impossible bitrate constraints, this will fail
[mpegts @ 0x28cbe20] muxrate VBR, pcr every 3 pkts, sdt every 200, pat/pmt every 40
pkts
Output #0, mpegts, to 'video5.ts':
  Metadata:
    encoder      : Lavf53.32.100
  Stream #0:0: Video: mpeg2video, yuv420p, 720x576 [SAR 64:45 DAR 16:9], q=2-31, 5
700 kb/s, 90k tbn, 30 tbc
  Stream #0:1: Audio: aac, 48000 Hz, 2 channels, s16, 150 kb/s
Stream mapping:
  Stream #0:0 -> #0:0 (wmv2 -> mpeg2video)
  Stream #0:1 -> #0:1 (wmav2 -> libfaac)
Press [q] to stop, [?] for help
frame= 26 fps= 0 q=2.5 size= 708kB time=00:00:00.76 bitrate=7564.5kbits/s dup
frame= 64 fps= 63 q=2.6 size= 1732kB time=00:00:02.03 bitrate=6976.6kbits/s dup
frame= 101 fps= 66 q=2.6 size= 2640kB time=00:00:03.26 bitrate=6620.2kbits/s dup
frame= 130 fps= 64 q=2.7 size= 3424kB time=00:00:04.23 bitrate=6625.9kbits/s dup
frame= 164 fps= 65 q=3.1 size= 4286kB time=00:00:05.36 bitrate=6542.1kbits/s dup
frame= 197 fps= 65 q=3.1 size= 5166kB time=00:00:06.46 bitrate=6544.0kbits/s dup
=32 drop=0
```

Figura 3-6. Codificación de video con ffmpeg



```

root@stcav-Studio-540: ~/live/mediaServer
frame= 2789 fps= 60 q=2.5 size= 68407kB time=00:01:32.86 bitrate=6034.4kbts/s dup
frame= 2822 fps= 60 q=2.5 size= 69094kB time=00:01:33.84 bitrate=6031.4kbts/s dup
frame= 2836 fps= 60 q=2.5 Lsize= 69479kB time=00:01:33.93 bitrate=6059.5kbts/s du
p=472 drop=0
video:62355kB audio:1720kB global headers:0kB muxing overhead 8.433170%
root@stcav-Studio-540:~/live/mediaServer# ./UniversalIndexer mediaServer/video1.ts
bash: ./UniversalIndexer: No such file or directory
root@stcav-Studio-540:~/live/mediaServer# cd ..
root@stcav-Studio-540:~/live# ./UniversalIndexer mediaServer/video1.ts
Starting indexing of [mediaServer/video1.ts]
Index file created will be [mediaServer/video1.tsx]
Searching the first PSI-PMT packet...
* PMT PID is 4096 (0x1000) - 2 Packets skipped since beginning
* PCR PID is 256 (0x100) - We assume it is also the video PID
Searching the first PCR...
* First PCR value is 0.667s - 3 Packets skipped since beginning
Searching PES sections start for indexing...
* 378440 TS packets were read
* 2836 PES sections were found
* 1654 PES sections were filtered
* Last PCR value is 95.167s
* Movie length estimation is 94.500s
* Indexing running time was 1s
End of indexing.
root@stcav-Studio-540:~/live#
    
```

Figura 3-7. Proceso de indexación con UniversalIndexer

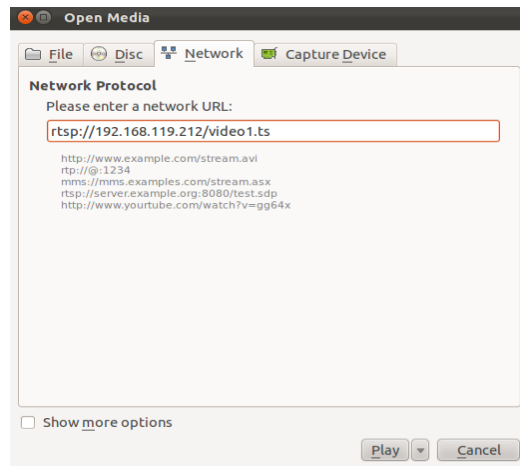
Después de la ejecución de este comando (`./UniversalIndexer`) se genera un archivo de extensión `.tsx`. En la Figura 3-8 se ha tomado una captura de pantalla de la carpeta `mediaServer`, que se encuentra alojada dentro de la carpeta de instalación del servidor LIVE555, llamada `live`, y en la cual se encuentran almacenados todos los archivos que estarán disponibles para transmitir y que podrán ser solicitados por cualquier usuario conectado por red al servidor, cuando esté en ejecución. En esta figura se pueden distinguir las tres versiones del archivo `video1`, `.wmv`, `.ts` y `.tsx`.

La Figura 3-9 es una captura de pantalla tomada a la ventana de volcado de red del reproductor VLC, como se puede observar, se ha configurado de tal manera que pueda consumir el archivo `Video1.ts` del servidor de VoD. En esta configuración se han especificado la IP del servidor y el protocolo de streaming utilizado.

```

root@stcav-Studio-540: ~/live/mediaServer
root@stcav-Studio-540:~/live/mediaServer# ls
1.ts          elephant_dreams.mp4  Spiderman4.ts      trailer2.ts
1.tsx        elephant_dreams.ts  Spiderman4.tsx    trailer2.tsx
COPYING      elephant_dreams.tsx Spidermann4.mp4   trailer3.ts
dasbeste.mp4 elephant_dreamsx1.ts Spidermann4.ts    trailer3.tsx
dasbeste.ts  ffmpeg2pass-0.log  Spidermann4.tsx  trailer4.ts
dasbeste.tsx ffmpeg2pass-0.log.mbt Spidermann.mp4   trailer4.tsx
dynamicRTSPServer.cpp ipod.mp4           The_Hobbit2.ts   trailer5.ts
dynamicRTSPServer.hh ipod.ts            The_Hobbit3.ts   trailer5.tsx
dynamicRTSPServer.o ipod.tsx           The_Hobbit3.tsx  trailer6.ts
elef.ts      Live555MediaServer The_Hobbit4.ts   trailer6.tsx
elef.tsx    live555MediaServer.cpp The_Hobbit4.tsx  trailer.ts
elephant_dreams2.ts live555MediaServer.o The_Hobbit5.ts   trailer.tsx
elephant_dreams2.tsx Makefile           The_Hobbit5.tsx  version.hh
elephant_dreams3.ts Makefile.head      The_Hobbit6.ts  video1.ts
elephant_dreams3.tsx Makefile.tail      The_Hobbit6.tsx video1.tsx
elephant_dreams4.ts Speed.mp4          The_Hobbit7.ts  video1.wmv
elephant_dreams4.tsx Speed.ts           The_Hobbit7.tsx video2.ts
elephant_dreams5.ts Speed.tsx          The_Hobbit7.ts  video2.tsx
elephant_dreams5.tsx Spiderman42.ts     The_Hobbit8.ts  video2.wmv
elephant_dreams6.ts Spiderman42.tsx    The_Hobbit.ts   video3.ts
elephant_dreams7.ts Spiderman43.ts     The_Hobbit.tsx  video3.tsx
elephant_dreams7.tsx Spiderman43.tsx   trailer10.ts    video3.wmv
elephant_dreams8.ts Spiderman43.tsx   trailer10.tsx
elephant_dreams8.tsx Spiderman4.mp4    trailer10.tsx
root@stcav-Studio-540:~/live/mediaServer#
    
```

**Figura 3-8.** Carpeta mediaServer



**Figura 3-9.** Volcado de red de VLC

El STB tiene un teclado por medio del cual es posible configurar la ventana de navegación del mismo para que pueda consumir el video del servidor de VoD. Una vez el STB ha hecho la petición al servidor VoD (LIVE555) y después de un tiempo de espera de aproximadamente dos segundos, se reproduce el video en el televisor, esto mismo ocurre en el reproductor multimedia VLC.

### 3.3.2 Captura de trazas de tráfico

Para esta parte del experimento se tomará en cuenta únicamente el consumo del servicio mediante el STB Amino debido a que el evento de transmisión y consumo es el mismo para el caso de plataforma web, la única diferencia consiste en la utilización de la red en cada momento del experimento, pero esto es algo que está totalmente fuera de contexto y que no interesa para el desarrollo del experimento.

Teniendo esto en cuenta, el paso a seguir es lograr capturar las trazas tráfico entre el servidor VoD LIVE555 y el cliente STB Amino mientras este último está consumiendo el servicio de VoD (archivo: Video1.ts).

Para realizar esta tarea se ha utilizado el software Wireshark (descrito con más detalle en la sección “Tecnologías relacionadas” del Capítulo 2). Esta herramienta es capaz de capturar todo el tráfico que fluye a través del dispositivo al que está conectado. En este punto es necesario recordar (como se mencionó en el capítulo 2, apartado 2.1.4.5) que el estándar de codificación MPEG-2, el cuál se ha utilizado para codificar los videos utilizados para la captura de tráfico, crea un flujo de video mediante cuatro tipos de datos, estos son: cuadros intra, cuadros posteriores predecibles, cuadros predecibles bi-direccionales y audio (de ahora en adelante se hará referencia a estos datos como tramas I, P, B y de audio).

El tráfico capturado de la red por medio de Wireshark, contiene todo tipo de información de control, relleno, sincronización, etc., entre los dos equipos, además de los distintos tipos de tramas que conforman el video en cuestión, y que al final, son los datos que interesan para el experimento. Por lo tanto, es necesario utilizar filtros en Wireshark para que únicamente muestre los datos de interés. En la Figura 3-10 se observa una captura de pantalla de Wireshark en la cual se ha capturado el tráfico del Video1. En esta imagen, no se ha aplicado ningún filtro a Wireshark por lo cual se puede observar toda la información compartida entre el servidor y el cliente, que en este caso es el STB Amino.

En la Figura 3-11 se aprecian los datos mostrados por Wireshark (tramas I, P, B y audio respectivamente) al aplicar distintos filtros para separar las tramas del video.

Es necesario recordar que las tramas que conforman un flujo de video MPEG-2 son demasiado grandes para ser empaquetadas en una trama Ethernet, por lo que son divididas al momento de la transmisión en muchas partes del tamaño de una trama Ethernet, con el fin de que puedan ser volcadas en el medio de transporte. Al hacer esta división, la primera trama Ethernet transmitida es la que contiene la cabecera de la trama MPEG-2 (I, P, B o audio). En esta cabecera se encuentra toda la información relacionada con la trama, como tipo de trama, tamaño y *timestamp* (en este caso corresponde al tiempo en el que se envía la trama en relación al momento en el que se empezó a capturar el tráfico con Wireshak). Las tramas Ethernet restantes que completan una trama MPEG-2 corresponden a la carga útil de la trama, esto es, la información codificada del video.

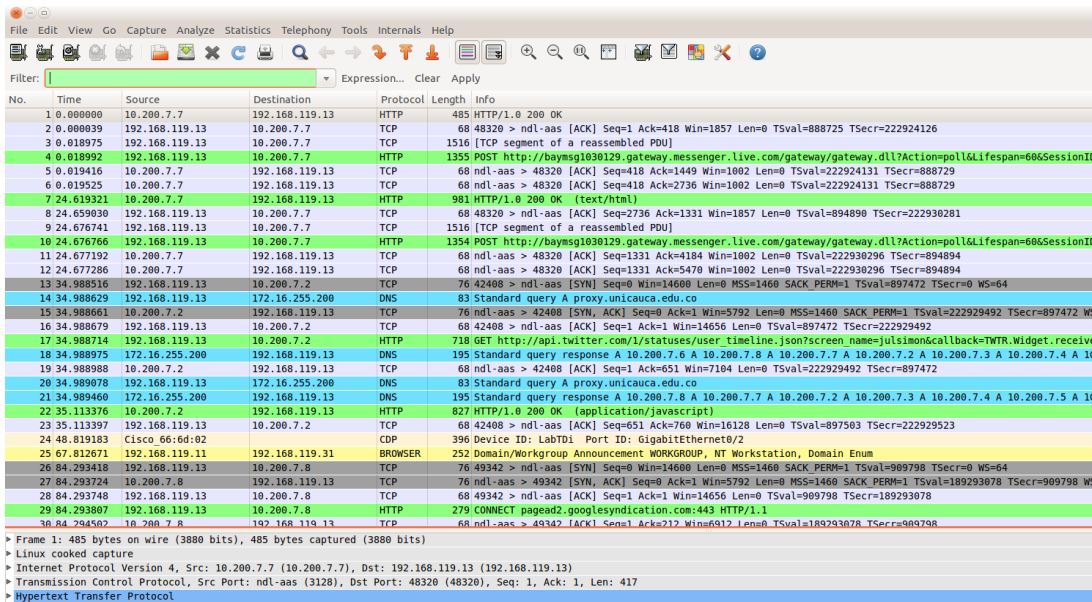


Figura 3-10. Tráfico capturado del video1 por medio de Wireshark

Filter: `mpeg-pes.dts and lmpeg-pes.frame_type==2` Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
269	704.156120	DTS 1.366666666	PTS 1.400000000	MPEG PES	1360	sequence-header
490	704.309703	DTS 1.700000000	PTS 1.800000000	MPEG PES	1360	video-stream
755	704.785469	DTS 2.100000000	PTS 2.200000000	MPEG PES	1360	video-stream
1022	705.215613	DTS 2.500000000	PTS 2.600000000	MPEG PES	1360	video-stream
1281	705.616481	DTS 2.900000000	PTS 3.000000000	MPEG PES	1360	video-stream
1522	706.035713	DTS 3.300000000	PTS 3.400000000	MPEG PES	1360	video-stream
1726	706.417927	DTS 3.700000000	PTS 3.800000000	MPEG PES	1360	video-stream
1956	706.825272	DTS 4.100000000	PTS 4.200000000	MPEG PES	1360	video-stream
2200	707.198954	DTS 4.500000000	PTS 4.600000000	MPEG PES	1360	video-stream
2462	707.635717	DTS 4.900000000	PTS 5.000000000	MPEG PES	1360	video-stream
2736	708.040884	DTS 5.300000000	PTS 5.400000000	MPEG PES	1360	sequence-header
2960	708.403144	DTS 5.700000000	PTS 5.800000000	MPEG PES	1360	sequence-header
3187	708.799044	DTS 6.100000000	PTS 6.200000000	MPEG PES	1360	video-stream
3432	709.231807	DTS 6.500000000	PTS 6.600000000	MPEG PES	1360	video-stream
3667	709.624828	DTS 6.900000000	PTS 7.000000000	MPEG PES	1360	sequence-header
3909	710.055547	DTS 7.300000000	PTS 7.400000000	MPEG PES	1360	sequence-header
4153	710.440580	DTS 7.700000000	PTS 7.800000000	MPEG PES	1360	sequence-header
4396	710.818131	DTS 8.100000000	PTS 8.200000000	MPEG PES	1360	sequence-header
4643	711.240666	DTS 8.500000000	PTS 8.600000000	MPEG PES	1360	sequence-header
4884	711.598182	DTS 8.900000000	PTS 9.000000000	MPEG PES	1360	video-stream

a) Tramas tipo I

Filter: `mpeg-pes.frame_type==2` Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
315	704.156868	DTS 1.400000000	PTS 1.500000000	MPEG PES	1360	p-frame
380	704.170540	DTS 1.500000000	PTS 1.600000000	MPEG PES	1360	p-frame
450	704.248013	DTS 1.600000000	PTS 1.700000000	MPEG PES	1360	p-frame
577	704.486410	DTS 1.800000000	PTS 1.900000000	MPEG PES	1360	p-frame
647	704.612032	DTS 1.900000000	PTS 2.000000000	MPEG PES	1360	p-frame
710	704.708571	DTS 2.000000000	PTS 2.100000000	MPEG PES	1360	p-frame
835	704.923042	DTS 2.200000000	PTS 2.300000000	MPEG PES	1360	p-frame
885	705.002530	DTS 2.300000000	PTS 2.400000000	MPEG PES	1360	p-frame
979	705.151737	DTS 2.400000000	PTS 2.500000000	MPEG PES	1360	p-frame
1104	705.340962	DTS 2.600000000	PTS 2.700000000	MPEG PES	1360	p-frame
1149	705.406666	DTS 2.700000000	PTS 2.800000000	MPEG PES	1360	p-frame
1236	705.547278	DTS 2.800000000	PTS 2.900000000	MPEG PES	1360	p-frame
1362	705.745046	DTS 3.000000000	PTS 3.100000000	MPEG PES	1360	p-frame
1398	705.799836	DTS 3.100000000	PTS 3.200000000	MPEG PES	1360	p-frame
1454	705.902915	DTS 3.200000000	PTS 3.300000000	MPEG PES	1360	p-frame
1606	706.169561	DTS 3.400000000	PTS 3.500000000	MPEG PES	1360	p-frame
1633	706.213491	DTS 3.500000000	PTS 3.600000000	MPEG PES	1360	p-frame
1687	706.334157	DTS 3.600000000	PTS 3.700000000	MPEG PES	1360	p-frame
1819	706.600574	DTS 3.800000000	PTS 3.900000000	MPEG PES	1360	p-frame

b) Tramas tipo P

Filter: `mpeg-pes.frame_type==3` Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
351	704.159695	192.168.119.13	PTS 1.433333333	MPEG PES	1360	b-frame
366	704.161353	192.168.119.13	PTS 1.466666666	MPEG PES	1360	b-frame
425	704.215829	192.168.119.13	PTS 1.533333333	MPEG PES	1360	b-frame
437	704.228171	192.168.119.13	PTS 1.566666666	MPEG PES	1360	b-frame
478	704.290742	192.168.119.13	PTS 1.633333333	MPEG PES	1360	b-frame
489	704.307531	192.168.119.13	PTS 1.666666666	MPEG PES	1360	b-frame
549	704.431220	192.168.119.13	PTS 1.733333333	MPEG PES	1360	b-frame
562	704.459431	192.168.119.13	PTS 1.766666666	MPEG PES	1360	b-frame
627	704.576359	192.168.119.13	PTS 1.833333333	MPEG PES	1360	b-frame
646	704.610541	192.168.119.13	PTS 1.866666666	MPEG PES	1360	b-frame
683	704.665718	192.168.119.13	PTS 1.933333333	MPEG PES	1360	b-frame
698	704.688070	192.168.119.13	PTS 1.966666666	MPEG PES	1360	b-frame
741	704.761511	192.168.119.13	PTS 2.033333333	MPEG PES	1360	b-frame
754	704.783708	192.168.119.13	PTS 2.066666666	MPEG PES	1360	b-frame
813	704.886550	192.168.119.13	PTS 2.133333333	MPEG PES	1360	b-frame
823	704.903984	192.168.119.13	PTS 2.166666666	MPEG PES	1360	b-frame
872	704.981857	192.168.119.13	PTS 2.233333333	MPEG PES	1360	b-frame
884	705.000922	192.168.119.13	PTS 2.266666666	MPEG PES	1360	b-frame
938	705.088238	192.168.119.13	PTS 2.333333333	MPEG PES	1360	b-frame

c) Tramas tipo B

Filter: `mpeg-pes.stream==audio-stream` Expression... Clear Appl

No.	Time	Source	Destination	Protocol	Length	Info
423	704.213766	192.168.119.13	PTS 1.400000000	MPEG	1360	
486	704.302938	192.168.119.13	PTS 1.528000000	MPEG	1360	
575	704.482813	192.168.119.13	PTS 1.656000000	MPEG	1360	
680	704.661236	192.168.119.13	PTS 1.784000000	MPEG	1360	
752	704.780311	192.168.119.13	PTS 1.912000000	MPEG	1360	
870	704.978672	192.168.119.13	PTS 2.061333333	MPEG	1360	
976	705.147284	192.168.119.13	PTS 2.210666666	MPEG	1360	
1091	705.321866	192.168.119.13	PTS 2.381333333	MPEG	1360	
1202	705.493475	192.168.119.13	PTS 2.552000000	MPEG	1360	
1278	705.611672	192.168.119.13	PTS 2.722666666	MPEG	1360	
1393	705.791711	192.168.119.13	PTS 2.893333333	MPEG	1360	
1505	706.002515	192.168.119.13	PTS 3.064000000	MPEG	1360	
1593	706.159015	192.168.119.13	PTS 3.213333333	MPEG	1360	
1631	706.208981	192.168.119.13	PTS 3.341333333	MPEG	1360	
1723	706.411522	192.168.119.13	PTS 3.490666666	MPEG	1360	
1817	706.597803	192.168.119.13	PTS 3.640000000	MPEG	1360	
1858	706.654532	192.168.119.13	PTS 3.768000000	MPEG	1360	
1942	706.798815	192.168.119.13	PTS 3.896000000	MPEG	1360	
2025	706.954726	192.168.119.13	PTS 4.024000000	MPEG	1360	

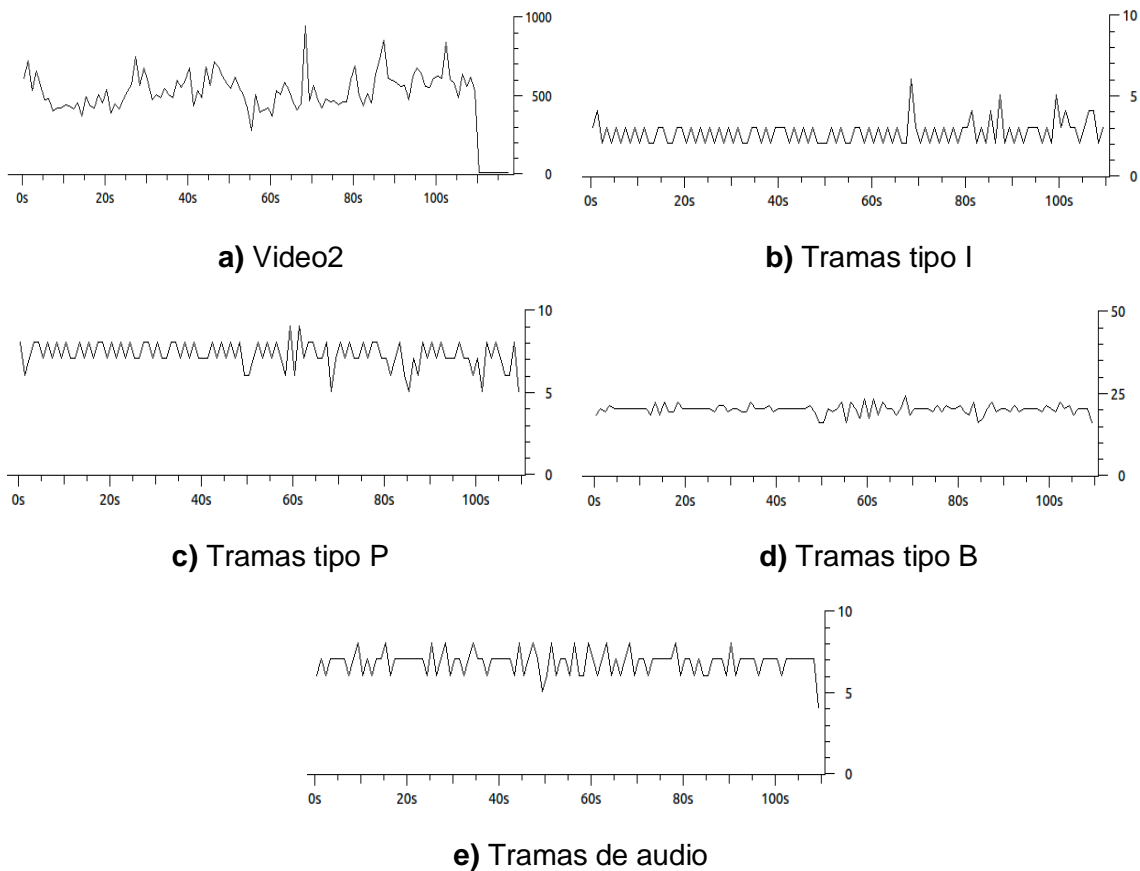
d) Tramas de audio

Figura 3-11. Visualización de las diferentes tramas MPEG-2 por separado, mediante la utilización de filtros en Wireshark

Para poder realizar la caracterización del comportamiento del tráfico de cada uno de los tipos de tramas MPEG-2 del video, se necesita caracterizar las dos componentes principales de ellas que son tamaño y tiempo entre tramas (el tiempo entre tramas se logra obtener por medio de la comparación del *timestamp* de cada trama con la de la trama anterior), por lo tanto, se debe abstraer de la cabecera de cada trama MPEG-2 la información correspondiente a estas dos componentes. Esta tarea fue realizada con la ayuda del software `awk` de Linux, el cual es un lenguaje de programación diseñado para procesar datos basados en texto [74].

### 3.3.3 Análisis del tráfico generado por el servicio de VoD

En este apartado se muestra los resultados obtenidos del comportamiento del tráfico para cada componente del video2. Hay que anotar que las gráficas mostradas en la Figura 3-12 (generadas con la ayuda de Wireshark) representan el comportamiento del tráfico en función del tiempo del video2 (más adelante se detallan las características de cada video), esto se hace únicamente con propósitos ilustrativos, ya que esta información, tabulada en este formato no es de utilidad cuando se trata de obtener una conducta basada en variabilidad con cierto comportamiento probabilístico [75].



**Figura 3-12.** Tráfico producido por la transmisión del video2, y sus respectivas componentes

### 3.3.4 Caracterización del tráfico real capturado en laboratorio

Una vez obtenidos los archivos que contienen las características de cada una de las tramas de video (I, P, B y audio) por separado, de cada uno de los tres videos, el siguiente paso en la realización del modelo de tráfico que representa el comportamiento real del tráfico de estos videos sobre una red IPTV, es la caracterización de cada una de estas tramas por medio de funciones de distribución de probabilidad (FDP, *Probability Distribution Function*). En este punto cabe recordar que para poder describir una trama de video por medio de una

función de distribución, es necesario determinar el comportamiento estadístico de cada una de sus componentes, estas son: tiempo entre paquetes y tamaño de paquetes; por lo tanto para describir el comportamiento estadístico de una trama de video, se necesitan dos funciones de distribución.

#### **3.3.4.1 Prueba de bondad de ajuste Kolmogorov-Smirnov**

Con el objetivo de encontrar las funciones de distribuciones de probabilidad que describan con mayor fidelidad el comportamiento del tráfico generado por cada una de las componentes del video, es posible utilizar la prueba de ajuste Kolmogorov-Smirnov [75]. En [76] se describe la metodología a seguir para realizar esta prueba a los datos de cualquier componente de video.

#### **3.3.4.2 Análisis de las componentes de los videos**

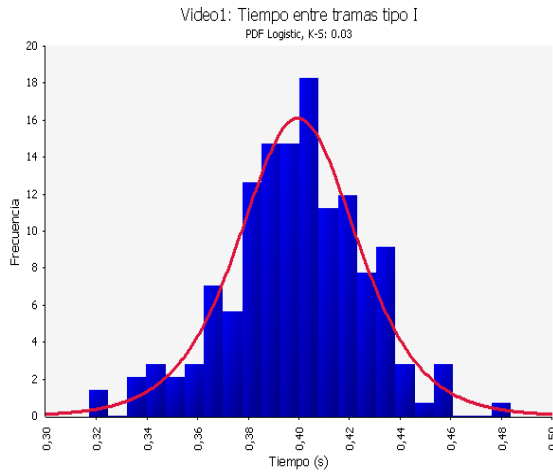
En este apartado se muestra gráficamente, por medio de histogramas de densidad de probabilidad, qué función de distribución de probabilidad (y sus respectivos parámetros) valida la hipótesis para cada componente de los tres videos. Para esto se tomarán en cuenta las funciones que cumplan con un nivel de confianza de al menos 90% al realizarles la prueba de Kolmogorov-Smirnov. Para esta tarea se utilizó el software @RISK [56] en su versión de prueba. Este software realiza el histograma de densidad de probabilidad de los datos ingresados, lo contrasta con las FDP que más se acerquen a su comportamiento, realiza la prueba de bondad de ajuste Kolmogorov-Smirnov y determina el parámetro K-S. Este parámetro tiene valores entre 0 y 1 y mide el nivel confianza de que los datos ingresados provengan de una determinada FDP (donde 0 representa el 100% de confianza y 1 el 0%). Para este caso de estudio se tendrán en cuenta las FDP cuyo parámetro K-S se encuentre entre 0 y 0.1.

#### **Video 1**

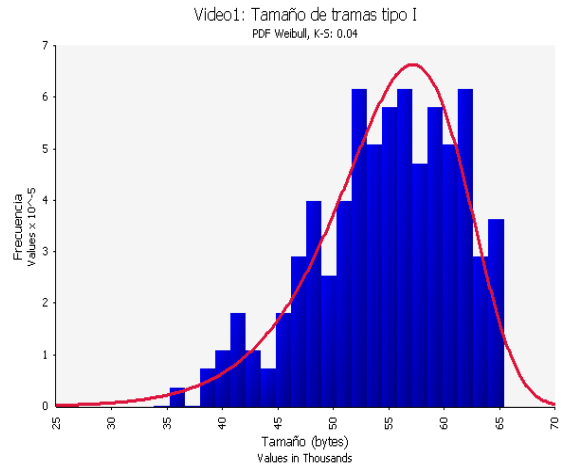
Características del video:

- Duración: 94s
- Tamaño: 62,4MB
- Tasa de bit: 5,2Mbps
- Complejidad: Este es un video con movimiento apreciable durante toda su duración
- Número de tramas I, P, B y de audio respectivamente: 201, 699, 1866 y 637

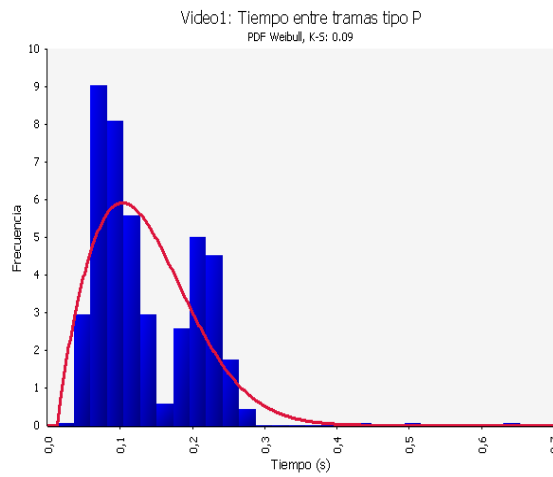
En la Figura 3-13 se observan los histogramas (barras azules) correspondientes a la densidad de probabilidad de cada uno de los componentes del video 1, además se aprecia la FDP (línea roja) que valida la hipótesis para la prueba de Kolmogorov-Smirnov en cada uno de los casos. En la Tabla 3-1 se ha hecho un resumen con toda esta información, además de la información de los dos videos restantes, la cual va a ser presentada a continuación.



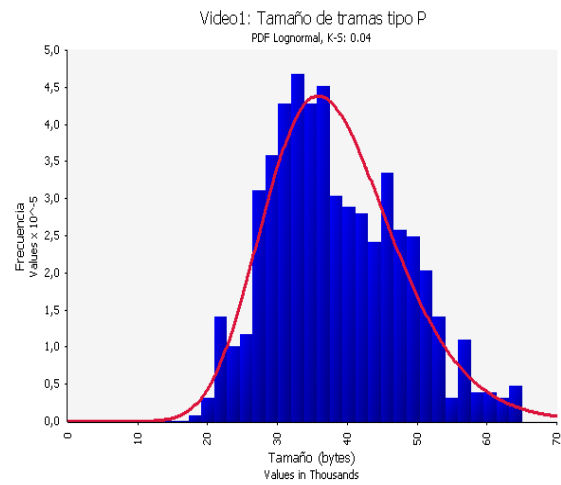
a) Tiempo entre tramas tipo I. Logistic



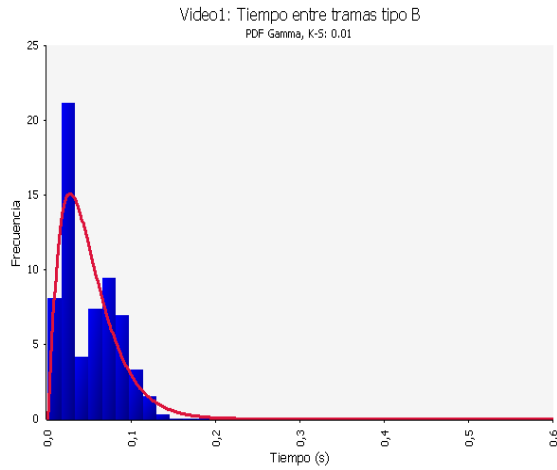
b) Tamaño de tramas tipo I. Weibull



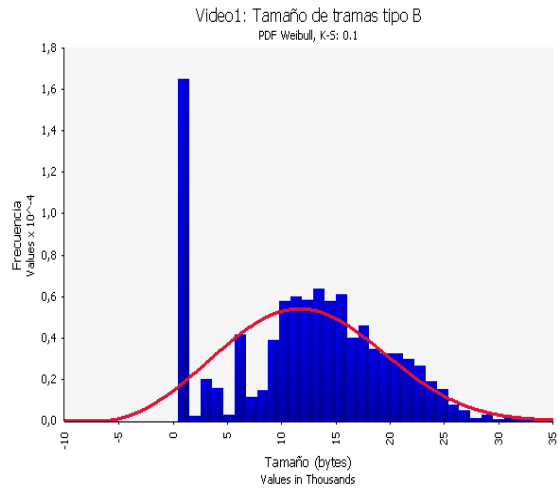
c) Tiempo entre tramas tipo P. Weibull



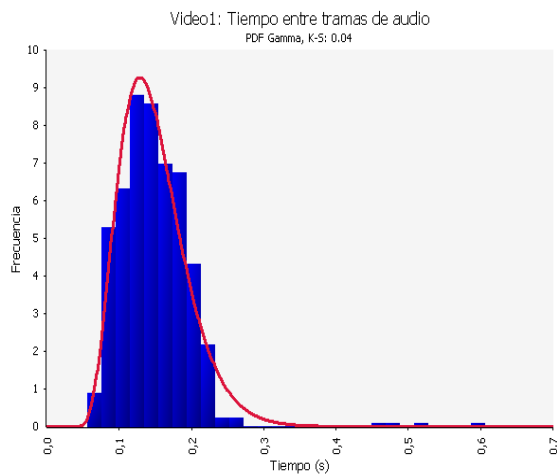
d) Tamaño de tramas tipo P. Lognormal



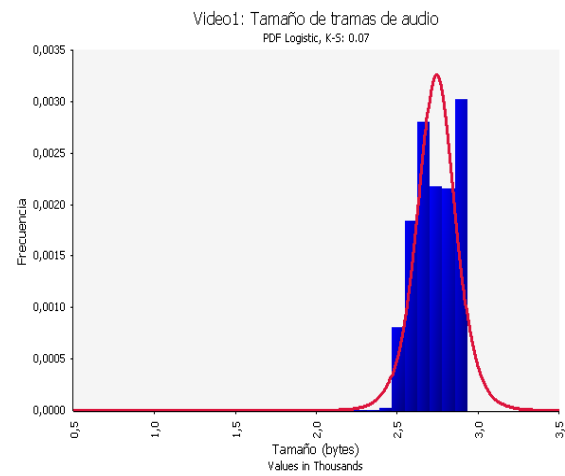
e) Tiempo entre tramas tipo B. Gamma



f) Tamaño de tramas tipo B. Weibull



g) Tiempo entre tramas de audio. Gamma



h) Tamaño de tramas de audio. Logistic

**Figura 3-13.** Caracterización de las componentes de tráfico del video1

## Video 2

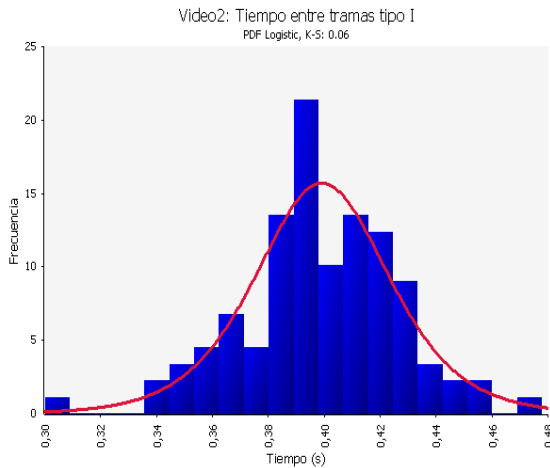
Características del video:

- Duración: 109s
- Tamaño: 58,9MB
- Tasa de bit: 4,3Mbps

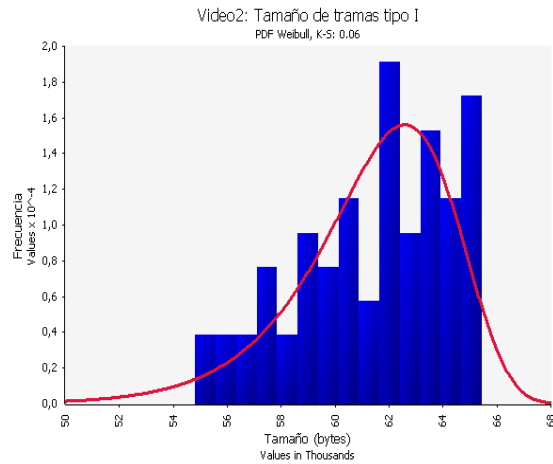


- Complejidad: Este es un video más extenso que el video 1, pero tiene la particularidad de que la cantidad de movimiento es más baja durante toda su duración.
- Número de tramas I, P, B y de audio respectivamente: 123, 790, 2188 y 748.

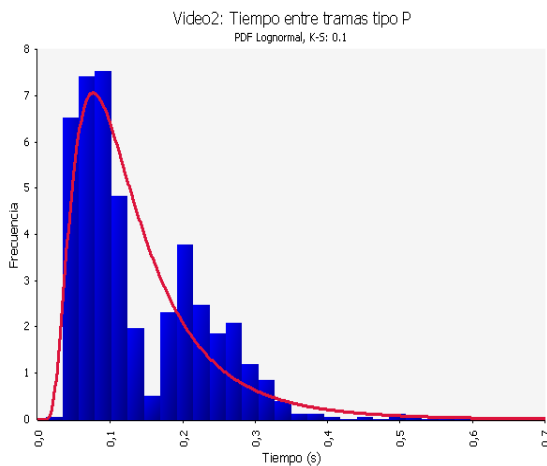
En la Figura 3-14 se puede apreciar los histogramas correspondientes a las componentes del video 2 y sus respectivas FDPs, al igual que se hizo con el video 1. Estos datos están resumidos en la Tabla 3-1.



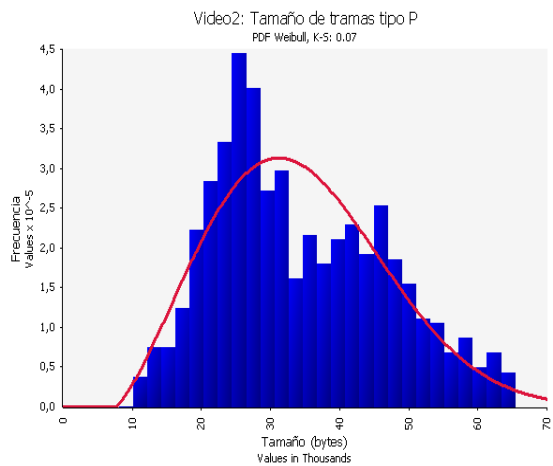
a) Tiempo entre tramas tipo I. Logistic



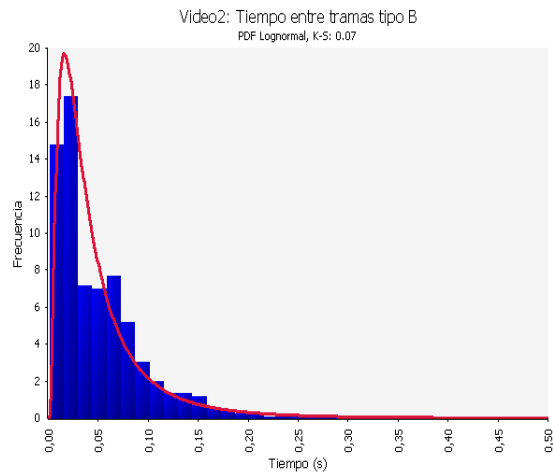
b) Tamaño de tramas tipo I. Weibull



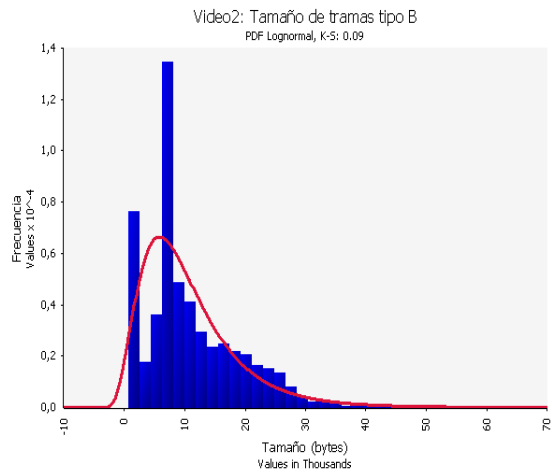
c) Tiempo entre tramas tipo P. Lognormal



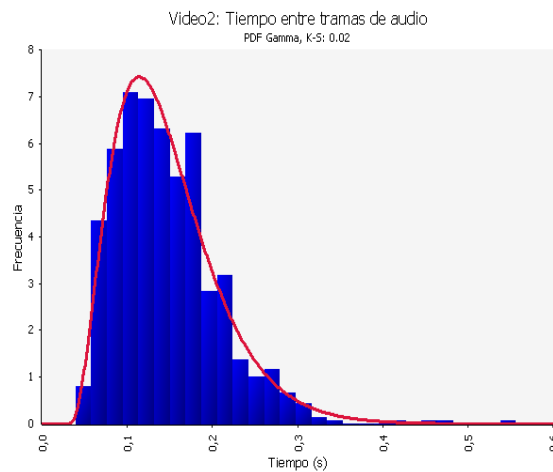
d) Tamaño de tramas tipo P. Weibull



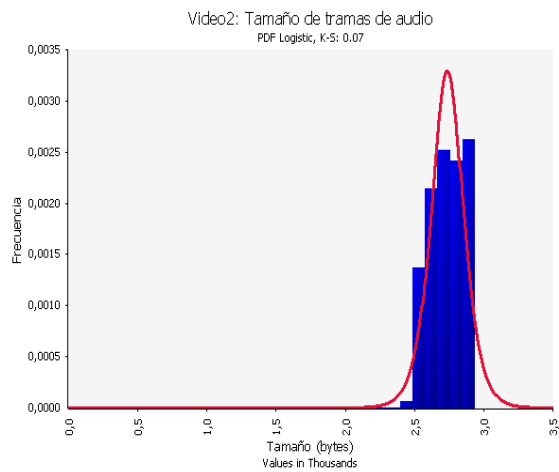
e) Tiempo entre tramas tipo B. Lognormal



f) Tamaño de tramas tipo B. Lognormal



g) Tiempo entre tramas de audio. Gamma



h) Tamaño de tramas de audio. Logistic

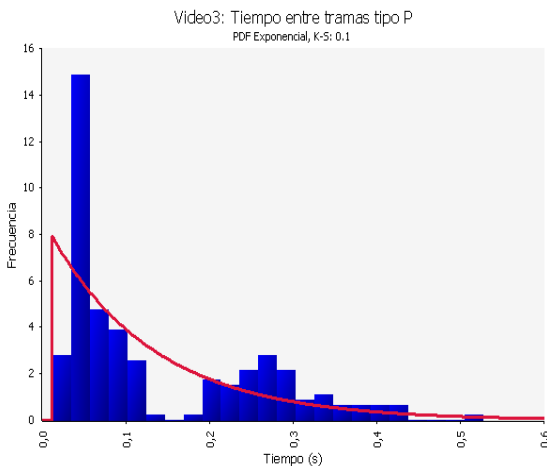
**Figura 3-14.** Caracterización de las componentes de tráfico del video2

### Video 3

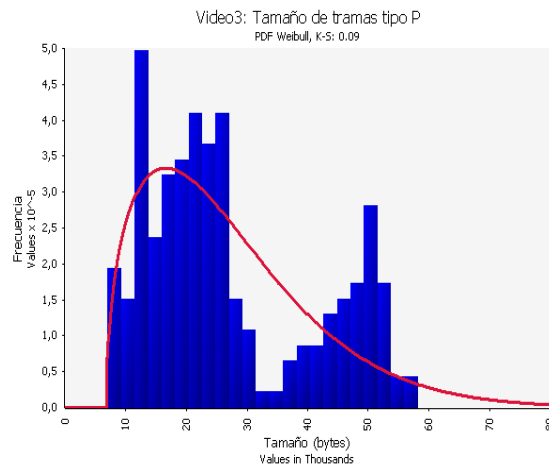
Características del video:

- Duración: 28s
- Tamaño: 9,7MB
- Tasa de bit: 2,8Mbps
- Complejidad: Video corto con muy baja movilidad.
- Número de tramas I, P, B y de audio respectivamente: 2, 208, 579 y 198.

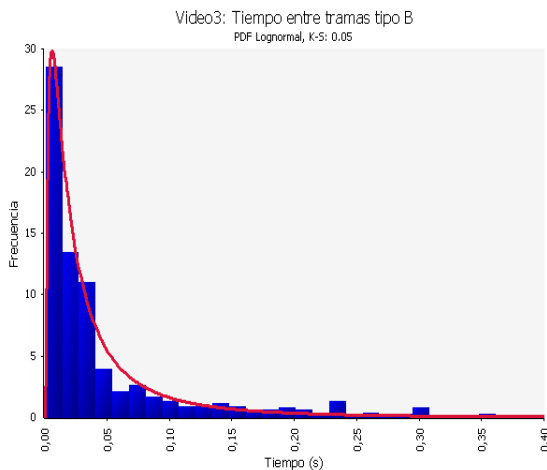
Como se puede observar en la Figura 3-15 se han realizado los histogramas de densidad de probabilidad para las componentes del video 3, junto con la FDP que valida la hipótesis de la prueba Kolmogorov-Smirnov, con sus respectivos parámetros de media ( $\mu$ ) y variación estándar ( $\sigma$ ), menos para las tramas tipo I. Esto se debe a que debido a las características del video (corta duración y baja movilidad) solamente se hallaron dos tramas I en todo el video, por lo tanto no es posible hallar una distribución estadística con tan pocos datos. Para solucionar este inconveniente es necesario crear las dos tramas independientemente para el funcionamiento del modelo, esto se verá detalladamente en el siguiente capítulo.



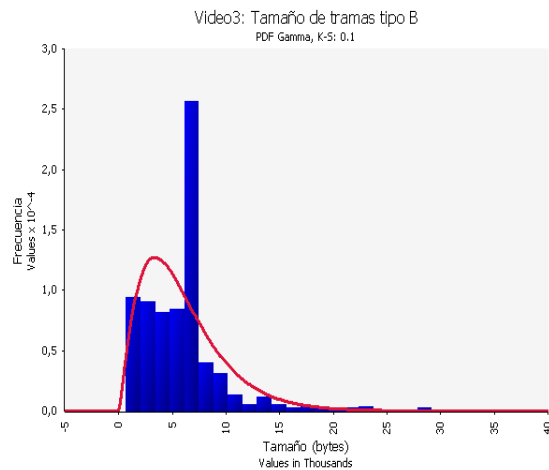
a) Tiempo entre tramas tipo P. Exponencial



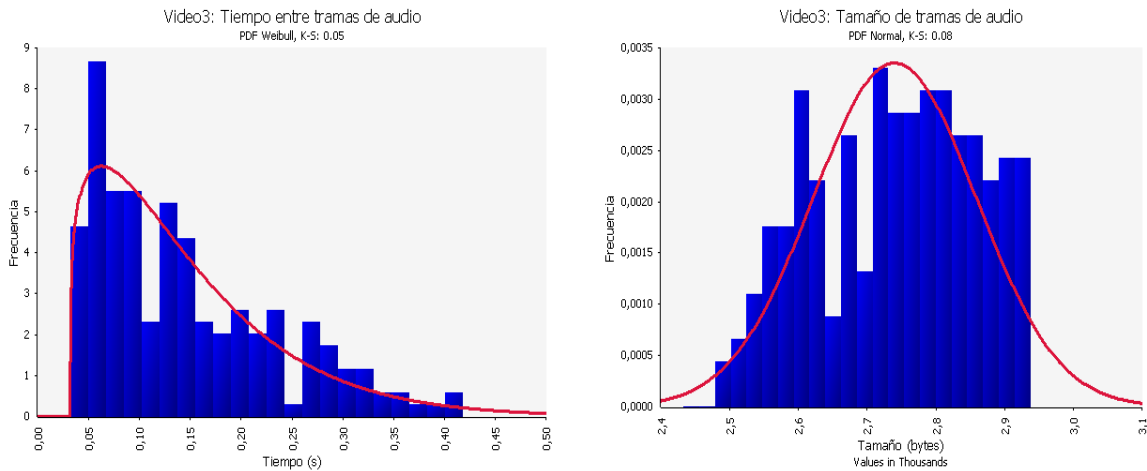
b) Tamaño de tramas tipo P. Weibull



c) Tiempo entre tramas tipo B. Lognormal



d) Tamaño de tramas tipo B. Gamma



e) Tiempo entre tramas de audio. Weibull

f) Tamaño de tramas de audio. Normal

**Figura 3-15.** Caracterización de las componentes de tráfico del video3

La información recogida del video 3 está resumida en la Tabla 4-1, al igual que la de los videos 1 y 2.

	Video1		Video2		Video3	
	Tiempo entre tramas	Tamaño de tramas	Tiempo entre tramas	Tamaño de tramas	Tiempo entre tramas	Tamaño de tramas
<b>Tramas tipo I</b>	Logistic $\mu=0,3995$ $\sigma=0,00283$	Weibull $\mu=57177,5$ $\sigma=6592,3$	Logistic $\mu=0,3993$ $\sigma=0,0289$	Weibull $\mu=61379,2$ $\sigma=2910,3$	N/A	N/A
<b>Tramas tipo P</b>	Weibull $\mu=0,1358$ $\sigma=0,0688$	Lognormal $\mu=38446,8$ $\sigma=9491,8$	Lognormal $\mu=0,1396$ $\sigma=0,0972$	Weibull $\mu=3447,4$ $\sigma=12343$	Exponencial $\mu=0,1375$ $\sigma=0,1261$	Weibull $\mu=26923,8$ $\sigma=14113,4$
<b>Tramas tipo B</b>	Gamma $\mu=0,0508$ $\sigma=0,0338$	Weibull $\mu=12270,8$ $\sigma=7051,2$	Lognormal $\mu=0,0514$ $\sigma=0,0559$	Lognormal $\mu=10706,5$ $\sigma=8272,7$	Lognormal $\mu=0,05$ $\sigma=0,0911$	Gamma $\mu=59314$ $\sigma=3895,4$
<b>Tramas de audio</b>	Gamma $\mu=0,1484$ $\sigma=0,0466$	Logistic $\mu=2743,1$ $\sigma=139,1$	Gamma $\mu=0,1468$ $\sigma=0,0614$	Logistic $\mu=2738,3$ $\sigma=137,6$	Weibull $\mu=0,1457$ $\sigma=0,0933$	Normal $\mu=2740$ $\sigma=118,9$

**Tabla 3-1.** Funciones de distribución de probabilidad (FDP) para las componentes de cada video

En la Tabla 3-1 se muestran todas las FDP para cada una de las componentes de los tres videos excepto para las tramas I del tercero. Esto se debe a que debido a la poca movilidad que tiene el video y su corta duración (como se miró en el capítulo 2), solo hicieron falta dos tramas tipo I para su codificación.

Cabe anotar que debido a las distintas características de los videos, no existe un patrón definido para ninguna de las componentes de los videos, aunque es posible observar que

existe mucha similitud entre el audio del primero y el segundo video y entre las tramas I de los mismos.

### 3.3.4.3 Caracterización del tráfico de servicios asociados al contenido

Existen otros servicios que ofrece una CAV aparte de VoD, estos por lo general son asociados al contenido (ver capítulo 2).

El proyecto ST-CAV [4] surgió como la continuación de un proyecto anterior desarrollado por la Línea de Investigación en Servicios Avanzados de Telecomunicaciones del departamento de Telemática de la Universidad del Cauca, denominado EDiTV (Educación Virtual basada en Televisión Interactiva para apoyar Procesos Educativos a Distancia) [77], cuyo objetivo fue fomentar la utilización de la televisión como un medio adicional para llegar a los estudiantes haciendo uso de las ventajas de la televisión interactiva. Al igual que el proyecto ST-CAV, este proyecto también tuvo la participación de la Universidad Autónoma de Occidente (Cali) y la Universidad de Oviedo (España).

En este proyecto se desarrollaron aplicaciones asociadas al contenido para TDi, las cuales han sido usadas para el desarrollo del proyecto ST-CAV. En [78] se realizó un proceso de investigación en el cual se caracterizó el comportamiento del tráfico generado por las aplicaciones de “solicitud de información adicional” y “respuestas de selección múltiple”. Estas aplicaciones y los tamaños de los archivos generados por cada una de ellas se resumen en la tabla 3-2.

Aplicación de solicitud de información adicional				Aplicación de respuestas de selección múltiple	
Solicitud de información	Archivo XML1	Archivo XML2	Imagen	Envío de Información	Respuesta del Servidor
256 bytes	1084 bytes	8443 bytes	10053 bytes	400 bytes	200 bytes

**Tabla 3-2.** Tamaño de las solicitudes, respuestas y archivos (XML e Imagen) para las aplicaciones de TDi

Para el presente únicamente se tendrá en cuenta la aplicación de solicitud de información adicional debido a que es la que genera el tráfico más crítico para el servicio de VoD.

En [78] se realizó la caracterización del comportamiento de los usuarios para dicha aplicación. A pesar de que el tiempo de respuesta del servidor ante una petición de un usuario es aleatorio, este es muy pequeño en comparación con el tiempo entre peticiones, por lo que es válido tomar el tiempo entre peticiones como el tiempo entre la transmisión de información relacionada con la aplicación. El resultado de esta caracterización es que la FDP que describe el comportamiento del tiempo entre peticiones de un usuario para la aplicación es Exponencial con:

$$\text{Media}(\mu) = 28,5262s, \text{ y } \text{variación estándar} (\sigma) = 25,1136s$$

### **3.4 CUMPLIMIENTO DE OBJETIVO ESPECÍFICO No. 1**

Con lo realizado en el capítulo 3, se da cumplimiento al objetivo específico No. 1 propuesto en el anteproyecto de este trabajo de grado, el cual es, “Caracterizar el comportamiento del tráfico generado por los servicios de una CAV sobre IPTV”.

## **CAPÍTULO 4**

# **MODELO DE TRÁFICO DE LOS SERVICIOS DE UNA CAV SOBRE IPTV**

En el presente capítulo, se describe el proceso de modelado, mediante simulación, del tráfico real capturado en laboratorio, y caracterizado mediante FDPs (ver apartado 3.3). Para esto se utilizan dos software distintos de simulación de redes, OPNET Modeler [54] y ns-3 [55]. Esto se hace con el fin de investigar en el uso de herramientas de fuente abierta como ns-3, y comparar su rendimiento con un software que es ampliamente usado en el mundo de la ingeniería de tráfico, el cual es OPNET Modeler.

El objetivo del modelado mediante simulación, es reproducir el comportamiento observado del tráfico real capturado en laboratorio, de los servicios de una CAV sobre IPTV, para poder dimensionar diferentes escenarios de red y evaluar este tráfico sobre ellas.

### **4.1 DESARROLLO DEL MODELO**

Con el desarrollo de este proyecto se ha querido prestar especial interés a la utilización de software libre que sea una alternativa al software privado (en este caso OPNET Modeler) utilizados comúnmente para la simulación de redes. Es por esto que se ha desarrollado un proceso de investigación en la herramienta de simulación de redes ns-3, la cual se ha descrito con detalle en el capítulo 2. Esto ha dado como resultado la creación de dos modelos de simulación, uno por cada simulador de redes. En el apartado 5.3, se realiza un análisis comparativo entre las dos herramientas para resaltar las ventajas y desventajas de una con respecto a la otra, y la efectividad del ns-3, ya que este es un software que no está desarrollado totalmente.

Por lo expuesto anteriormente, este apartado se divide en dos partes, en la primera de ellas se expone el proceso de desarrollo del modelo en el simulador OPNET Modeler, y en la segunda, se hace lo propio con ns-3.

#### **4.1.1 Planteamiento del problema**

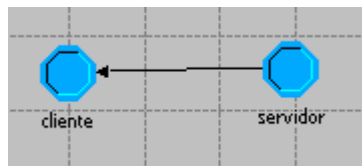
Teniendo en cuenta lo expuesto en el apartado 3.3.1.1, es necesario crear en el software de simulación, un servidor que sea capaz de producir tráfico de VoD con igual comportamiento que el tráfico producido por el servidor real del laboratorio, así como también el tráfico producido por las aplicaciones asociadas al contenido, las cuáles se detallarán más adelante. Además, se debe crear un cliente que sea capaz de consumir estos servicios y pueda generar estadísticas de interés para el modelo.

#### 4.1.2 OPNET Modeler

Para el desarrollo del presente proyecto se utilizó el simulador de redes OPNET Modeler 11.5. En el Anexo C, se describe la forma de instalación y configuración de esta versión de OPNET Modeler en un sistema operativo Windows XP Service Pack 3. Además, en el Anexo A se describen también los distintos componentes y abstracciones del simulador, además de su funcionamiento.

##### 4.1.2.1 Descripción del modelo

En la Figura 4-1 se muestra el modelo básico desarrollado, a nivel de red. Como se puede apreciar existe solamente un servidor conectado directamente a un cliente para que corresponda con el experimento realizado en el laboratorio, esto se hace para poder realizar la validación del modelo de simulación, mediante la comparación de los resultados arrojados por el mismo, con los del escenario real.



**Figura 4-1.** Modelo básico de simulación a nivel de red

A continuación se describen los dos componentes del modelo a nivel de red: Servidor y Cliente.

##### **Servidor**

Como se ha mencionado anteriormente, el propósito del servidor en el modelo es difundir contenido multimedia y aplicaciones asociadas al contenido como lo hacen los servidores reales del laboratorio.

En la Figura 4-2, se observa el servidor a nivel de nodo, así mismo se puede observar que hay dos módulos. El módulo denominado p\_1, es un procesador al cual se le ha asignado el proceso descrito párrafos más adelante. El módulo denominado pt\_0, es un módulo transmisor existente en el editor de nodos de OPNET Modeler, este módulo se encarga de volcar la señal en el medio de transmisión (enlace). Los dos módulos están unidos por medio de un flujo de señal, el cual posee un código para poder ser utilizado dentro de los procesos de los módulos. En la Figura 4-3 se aprecia la configuración del módulo transmisor, este está configurado para transmitir a una tasa de bit de 100Mbps de acuerdo a la tecnología existente en el laboratorio, la cual es *Fast Ethernet*.





Figura 4-2. Modelo del servidor a nivel de nodo

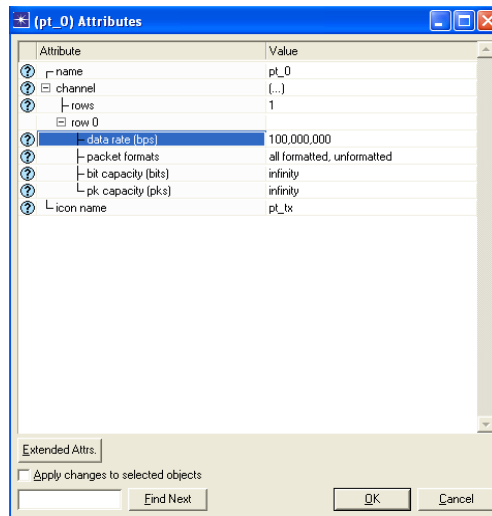


Figura 4-3. Configuración del módulo pt\_0

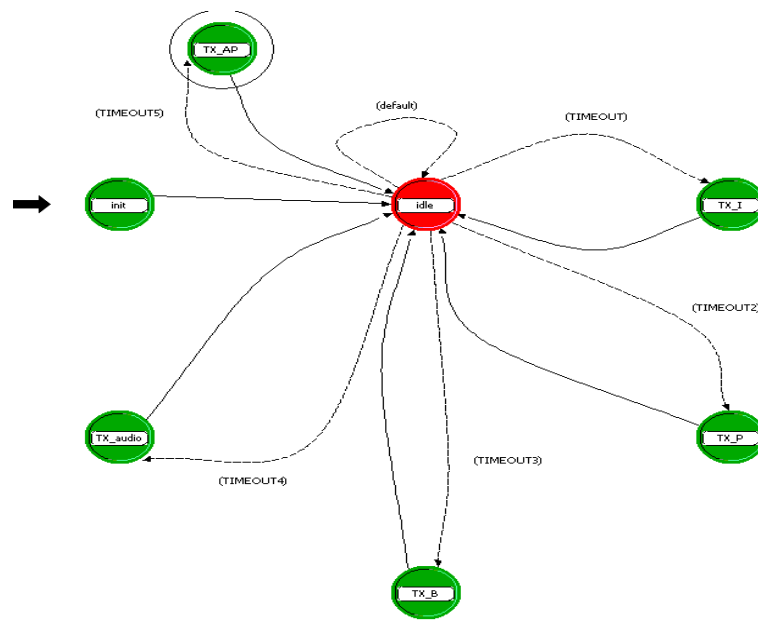


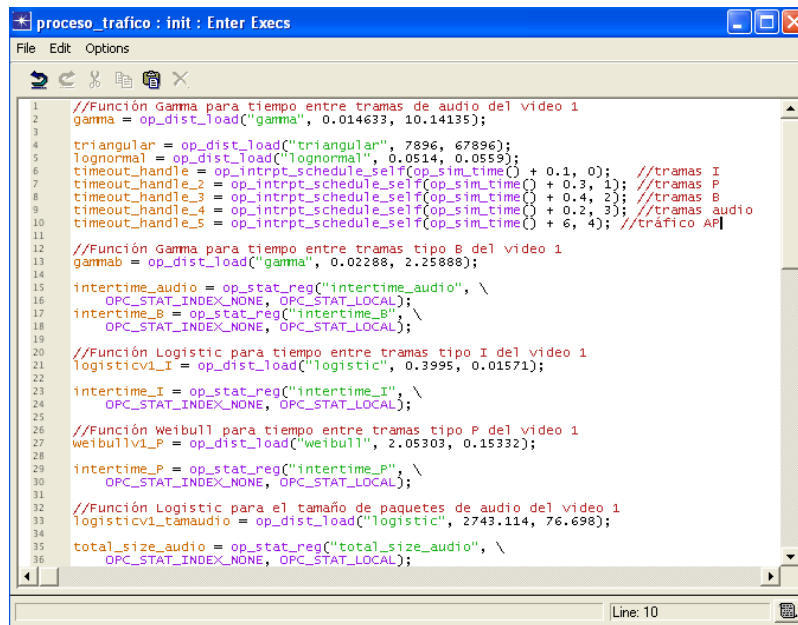
Figura 4-4. Proceso asociado al módulo p\_1

La Figura 4-4 representa el proceso asociado al módulo p\_1 del modelo del servidor. El propósito de este proceso es crear los diferentes tipos de tramas MPEG-2 de cada video y transmitir las de acuerdo a las FDPs halladas para el tiempo entre tramas de cada una de ellas. El tamaño de estas tramas también tendrá asociadas las FDPs correspondientes.

A continuación se describe cada uno de los estados del proceso asociado al módulo p\_1 del nodo servidor.

### Init:

Este es el estado inicial del proceso, cuyo código se ejecuta una sola vez en toda la simulación, en la Figura 4-5 se observa una captura de pantalla de una parte de este código, el cuál pertenece a la ejecución de entrada del estado. Aquí están declarados los manejadores de las interrupciones, los cuales se describirán posteriormente. Además como se observa en la Figura 4-5, también están declaradas las funciones de cada trama MPEG-2 y de audio de cada video, con sus respectivos parámetros. En la Tabla 4-1 se aprecia cada una de las funciones utilizadas para la simulación con los parámetros requeridos por OPNET Modeler [54].



```
1 //Función Gamma para tiempo entre tramas de audio del video 1
2 gamma = op_dist_load("gamma", 0.014633, 10.14135);
3
4 triangular = op_dist_load("triangular", 7896, 67896);
5 lognormal = op_dist_load("lognormal", 0.0514, 0.0559);
6 timeout_handle = op_intrpt_schedule_self(op_sim_time() + 0.1, 0); //tramas I
7 timeout_handle_2 = op_intrpt_schedule_self(op_sim_time() + 0.3, 1); //tramas P
8 timeout_handle_3 = op_intrpt_schedule_self(op_sim_time() + 0.4, 2); //tramas B
9 timeout_handle_4 = op_intrpt_schedule_self(op_sim_time() + 0.2, 3); //tramas audio
10 timeout_handle_5 = op_intrpt_schedule_self(op_sim_time() + 6, 4); //tráfico AP
11
12 //Función Gamma para tiempo entre tramas tipo B del video 1
13 gammab = op_dist_load("gamma", 0.02288, 2.25888);
14
15 intertime_audio = op_stat_reg("intertime_audio", \
16     OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
17 intertime_B = op_stat_reg("intertime_B", \
18     OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
19
20 //Función Logistic para tiempo entre tramas tipo I del video 1
21 logisticv1_I = op_dist_load("logistic", 0.3995, 0.01571);
22
23 intertime_I = op_stat_reg("intertime_I", \
24     OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
25
26 //Función Weibull para tiempo entre tramas tipo P del video 1
27 weibullv1_P = op_dist_load("weibull", 2.05303, 0.15332);
28
29 intertime_P = op_stat_reg("intertime_P", \
30     OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
31
32 //Función Logistic para el tamaño de paquetes de audio del video 1
33 logisticv1_tamaudio = op_dist_load("logistic", 2743.114, 76.698);
34
35 total_size_audio = op_stat_reg("total_size_audio", \
36     OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
```

Figura 4-5. Fragmento de código del estado init del proceso del servidor

<b>Gamma</b>	scale	shape
<b>Logistic</b>	mean	scale
<b>Weibull</b>	shape	scale
<b>Lognormal</b>	mean	variance
<b>Normal</b>	mean	variance

**Tabla 4-1.** FDPs utilizadas en el modelo de OPNET Modeler con los parámetros requeridos por el simulador para poder declararlas

Para poder declarar las funciones en el simulador es necesario calcular los parámetros requeridos por el mismo. Para esto existen ecuaciones matemáticas, las cuales se describen a continuación.

Para las siguientes ecuaciones, *mean* ( $\mu$ ) = *media* y *sd* ( $\sigma$ ) = *standard deviation* (desviación estándar).

**FDP Gamma (*scale* ( $\beta$ ), *shape* ( $\alpha$ )):**

$$mean = \alpha\beta, sd = \beta\sqrt{\alpha} \quad (1)$$

**FDP Logistic (*mean* ( $\mu$ ), *scale* ( $\beta$ )):**

$$sd = \frac{\pi\beta}{\sqrt{3}} \quad (2)$$

**FDP Weibull (*shape* ( $\alpha$ ), *scale* ( $\beta$ )):**

$$mean = \beta\Gamma\left(1 + \frac{1}{\alpha}\right), sd = \beta\sqrt{\Gamma\left(1 + \frac{2}{\alpha}\right) - \Gamma\left(1 + \frac{1}{\alpha}\right)^2} \quad (3)$$

Donde  $\Gamma(z)$  es la función Gamma, la cual está definida como

$$\Gamma(1 + z) = \int_0^{\infty} t^z e^{-t} dt \quad (4)$$

**FDP Lognormal (*normal mean* ( $\mu$ ), *variance*):**

$$mean = e^{\mu + \frac{\sigma^2}{2}}, variance = (e^{\sigma^2} - 1)e^{2\mu + \sigma^2} \quad (5)$$

**FDP Normal (*mean* ( $\mu$ ), *variance*):**

$$variance = \sigma^2 \quad (6)$$

En la Figura 4-5 aparecen algunas de las funciones para las tramas del video1, con sus respectivos parámetros ya calculados. Hay que mencionar que el orden en el que se ingresan estos parámetros corresponde al de la Tabla 4-1 [54].

### Idle:

Este es un estado estacionario (el único no forzado), que sirve como espera para que el proceso cambie a cualquiera de los cuatro estados forzados por medio de la activación de interrupciones. Cuando se activa alguna de las interrupciones, el proceso inmediatamente cambiará de estado dependiendo de cuál de ellas sea. Además de las líneas de interrupción que hacen cambiar de estado al proceso, el estado *init* también posee una interrupción interna denominada *default*, la cual se utiliza con el fin de prevenir bucles.

### TX\_I, TX\_P, TX\_B, TX\_audio y TX\_AP:

Estos cinco estados son los encargados de crear, configurar y transmitir los paquetes contenedores de las tramas correspondientes al servicio de VoD y a la aplicación asociada al contenido (TX\_AP; de la cual se describió su caracterización en el apartado 3.3.4.3). Cada uno de estos estados lo activa una interrupción diferente, la cual es de tipo *timeout* (tiempo de espera) y está configurada de acuerdo al tiempo entre paquetes de cada trama, y declarada en el bloque HB. En este bloque se declara el nombre de la interrupción, tipo de interrupción y un código (número) para poder acceder a ella.

En este punto hay que aclarar que el servidor se puede configurar para almacenar un video a la vez, por lo que las interrupciones serán distintas para cada video. Cada interrupción tiene un “manejador” (ver Figura 4-5), el cual se encarga de configurar el tiempo de espera de la interrupción (el proceso es realmente interrumpido por el manejador, indicando al proceso que debe atender una interrupción de tipo *timeout* con código x). Estos manejadores están declarados en el código de entrada del estado *init* y están configurados para inicializar en un tiempo determinado. Además, también están inicializados en el bloque SV como variables de tipo *Evhandle*.

### Cliente

En la Figura 4-6 se puede apreciar el cliente a nivel de nodo, el cuál es muy parecido al nodo servidor, la diferencia radica en que el nodo cliente posee un módulo de recepción (*pr\_0*), y este está conectado con un flujo de señal que va hacia el procesador (*p\_0*).



Figura 4-6. Modelo del cliente a nivel de nodo

El módulo pr\_0 está configurado para trabajar a una tasa de bit de 100Mbps. Por su parte, el módulo p\_0 ha sido configurado para trabajar con el proceso mostrado en Figura 4-7. El objetivo de este nodo es recibir los paquetes provenientes del nodo de transmisión, identificar qué tipo de trama contiene (I, P, B, audio o aplicación asociada al contenido), y registrar estadísticas de control. En la Figura 4-7 se aprecia que el proceso asociado al módulo p\_0 del nodo cliente, posee tres estados distintos, los cuales se describen a continuación.

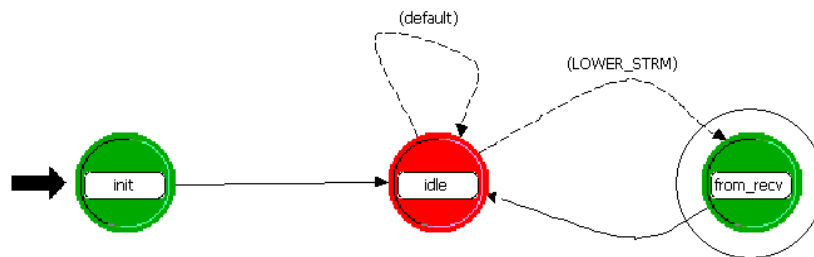


Figura 4-7. Proceso asociado al módulo p\_0

#### Init:

Este es el estado inicial del proceso. Aquí se han declarado variables de estado correspondientes a las estadísticas tomadas por el cliente. En la Figura 4-8 se observa una captura de pantalla de un fragmento del código de este estado.

```
1 header_st = op_stat_reg("header_st", \
2   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
3
4 total_size = op_stat_reg("total_size", \
5   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
6
7 payload_st = op_stat_reg("payload_st", \
8   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
9
10 intertimeI = op_stat_reg("intertimeI", \
11   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
12
13 intertimeP = op_stat_reg("intertimeP", \
14   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
15
16 intertimeB = op_stat_reg("intertimeB", \
17   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
18
19 intertime_audio = op_stat_reg("intertime_audio", \
20   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
21
22 total_size_P = op_stat_reg("total_size_P", \
23   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
24
25 total_size_B = op_stat_reg("total_size_B", \
26   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
27
28 total_size_audio = op_stat_reg("total_size_audio", \
29   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
30
31 intertime_tcp = op_stat_reg("intertime_tcp", OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
32
33 total_size_tcp = op_stat_reg("total_size_tcp", \
34   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
35
36 delay = op_stat_reg("delay", \
37   OPC_STAT_INDEX_NONE, OPC_STAT_LOCAL);
```

Figura 4-8. Fragmento de código del estado init del proceso del cliente

**Idle:**

Al igual que en el proceso del servidor, este es un estado estacionario que sirve de espera para que el proceso cambie al estado *from\_rcv* en el momento en el que se active la interrupción *LOWER\_STRM*. Además de esta interrupción, también posee una interrupción del tipo *default* para prevenir bucles.

**from\_rcv:**

Este estado es el encargado de recibir los paquetes provenientes del módulo *pr\_0* del nodo cliente y procesarlos. Para que el proceso pueda entrar en el estado *from\_rcv*, existe una interrupción denominada *LOWER\_STRM*, la cual al activarse, permite el cambio de estado entre *idle* y *from\_rcv*. Esta interrupción está definida en el bloque HB y es del tipo *stream* (flujo).

**4.1.2.2 Validación del modelo**

El objetivo de la validación del modelo, es mostrar que los datos transmitidos por parte del servidor simulado en OPNET Modeler generan un tráfico que representa el comportamiento del tráfico real capturado del servidor del laboratorio. Para esto se contrastarán las funciones obtenidas en el apartado 3.3.4.2 con los resultados arrojados por OPNET Modeler; en este apartado se mostró que es posible caracterizar los datos obtenidos de la captura de tráfico, para cada trama MPEG-2 de cada video, por medio de FDPs. En la Figura 3-14 se muestran estas FDPs para el video1, junto con los histogramas correspondientes a la densidad de probabilidad de cada una de las tramas de cada video.

En OPNET Modeler es posible observar los resultados de las simulaciones gráficamente. Existen además, opciones en el software que permiten mostrar estas gráficas en *FDP*, *CDF* (*Cumulative Distribution Function*), *average* (promedio), tal como son los datos, etc. Para poder hacer un análisis comparativo de los resultados de las simulaciones con las FDPs obtenidas en el apartado 3.3.4.2, se utilizará la opción de visualización FDP.

Además de las gráficas obtenidas en OPNET Modeler, se utilizará también el software @RISK para poder corroborar estos resultados, mediante la aproximación de los datos obtenidos en el simulador, a FDPs por medio de la prueba de Kolmogorov-Smirnov. Para tal objeto se utilizará la opción de exportación de gráfica de OPNET Modeler, la cual produce un archivo de texto correspondiente a la gráfica que se está exportando. Este archivo de texto posee dos columnas, una por cada eje de la gráfica. Debido a motivos de extensión, únicamente se mostrará en este documento la validación para el video 1. El proceso de validación para los demás videos es exactamente igual.

En la Figura 4-9 se aprecia una captura de pantalla que muestra el resumen de la simulación. Se puede observar que el tiempo de simulación es de 1m 34s, lo cual corresponde a 94s, que es el tiempo que dura el video 1 (ver apartado 3.3.4.2, características del video1). Además, la Figura 4-9 muestra otros datos que corresponden al número de paquetes (tramas I, P, B y audio) transmitidos por el servidor, los cuales se pueden contrastar con las características del video1.

En la Figura 4-10 se muestran los resultados obtenidos de la simulación para las tramas tipo I del video1, tomadas directamente de OPNET Modeler, mientras que en la Figura 4-11 se aprecian los resultados de todas las tramas MPEG-2 para la simulación del video1, tabulados con la herramienta @RISK. Aquí se puede ver que al igual que en el apartado 3.3.4.2, los datos han sido representados por medio de histogramas de densidad de frecuencia (barras azules) y las FDPs (líneas rojas) a las que más se acerca la distribución de estos datos.

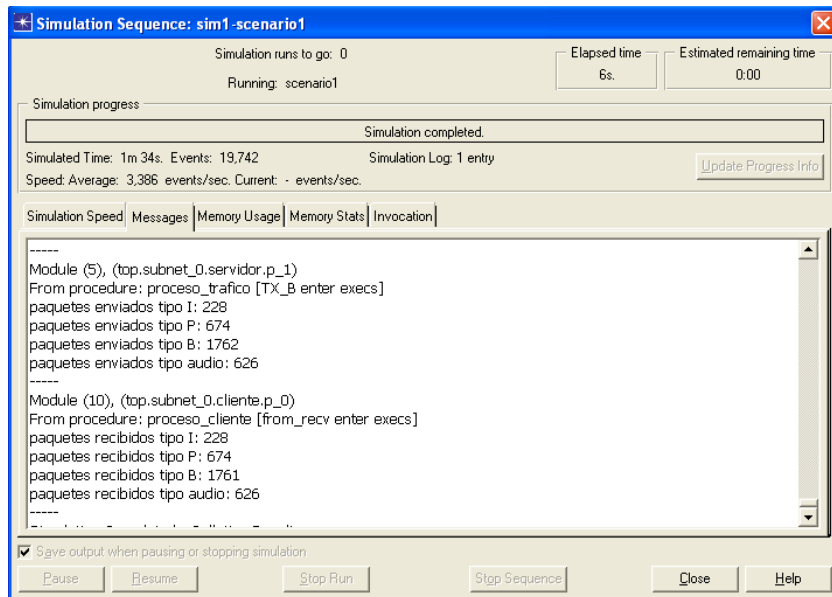
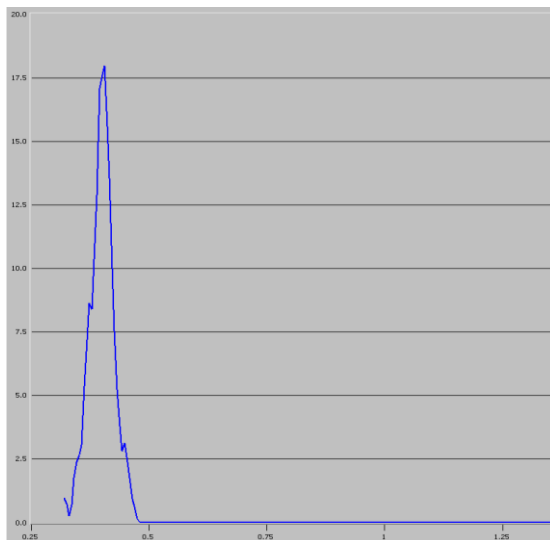
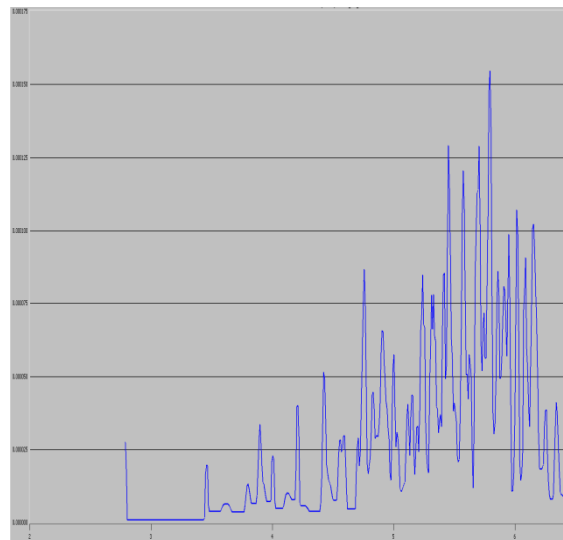


Figura 4-9. Resumen de la simulación para el video1



a) Tiempo entre tramas tipo I



b) Tamaño de tramas tipo I

Figura 4-10. Gráficas de los resultados para las tramas tipo I del video1, arrojados por OPNET Modeler en mode FDP

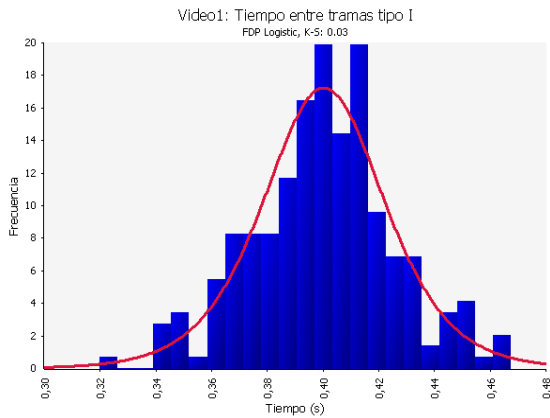
En la Figura 4-10, para las gráficas de tiempo entre paquetes, el eje x corresponde al tiempo entre paquetes y el eje y a la frecuencia de los valores del eje x. Para las gráficas de tamaño de paquetes, el eje x corresponde al tamaño de paquetes y el eje y a la frecuencia de los valores del eje x.

Como se puede observar en la Figura 4-10, la opción de visualización de gráficas *FDP* de OPNET Modeler, muestra la densidad de probabilidad no con un histograma como comúnmente se hace, sino con una línea continua. Teniendo estos resultados, es posible compararlos directamente con las *FDPs* resultantes en la Figura 3-14 del apartado 3.3.4.2. A pesar de que las gráficas arrojadas por OPNET Modeler no determinan las *FDPs* a las cuales se aproximan los datos, es posible encontrar un gran parecido entre las gráficas de la Figura 4-10 y las correspondientes gráficas de la Figura 3-14; esto indica que el comportamiento del tráfico del video1 transmitido por el servidor del modelo desarrollado en OPNET Modeler es muy parecido al comportamiento del tráfico del video1 transmitido por el servidor real del laboratorio. Para corroborar lo anterior, se exportaron los datos de las gráficas de la Figura 4-10 (además de las tramas restantes del video1) en forma de texto, para luego poder tabularlos en la herramienta @RISK y determinar las *FDPs* que se aproximan a ellas. Las gráficas de la Figura 4-11 muestran los resultados obtenidos para lo expuesto anteriormente.

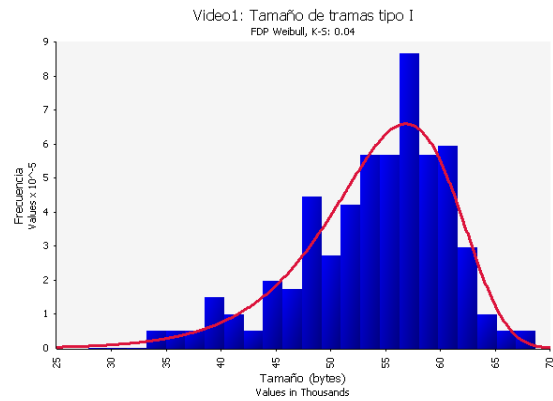
En la Figura 4-11 se puede apreciar que para el video 1, el tráfico transmitido por el servidor del modelo desarrollado en OPNET Modeler se comporta de manera similar al tráfico transmitido por el servidor de VoD del laboratorio. El hecho de que el parámetro K-S sea en general menor para los datos recogidos de la transmisión del servidor en el modelo que los del servidor real, se debe a que el tráfico transmitido por el servidor real es aleatorio pero su comportamiento se puede representar mediante *FDPs*. Al contrario de esto, el comportamiento del tráfico transmitido por el servidor del modelo, proviene directamente de *FDPs*, de tal manera que se aproxima mucho más a estas. Si la simulación tuviera un tiempo lo suficientemente grande, los datos de la transmisión de tráfico se asemejarían mucho más a las *FDPs* de las que provienen, por lo tanto, el parámetro K-S tendería a cero.

La Tabla 3-1 muestra un resumen de la caracterización realizada a los datos de cada una de las tramas de los tres videos transmitidos por el servidor real de VoD. En esta tabla se observa que se han colocado los parámetros correspondientes a la media y la variación estándar de cada una de las funciones. En la Tabla 4-2 se ha hecho un resumen de los resultados de la caracterización realizada a los datos obtenidos del servidor del modelo desarrollado en OPNET Modeler para el video 1. Además, se muestran los parámetros media y variación estándar de cada función. Como se observa, las *FDPs* son las mismas para las Tablas 3-1 (video1) y 4-2.

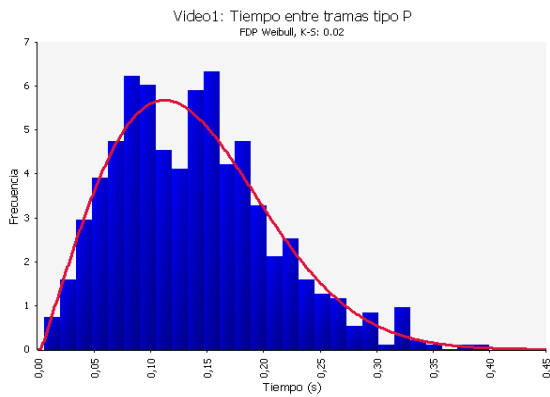




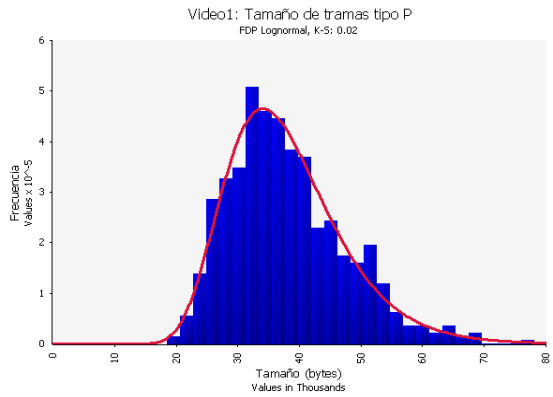
a) Tiempo entre tramas tipo I. Logistic



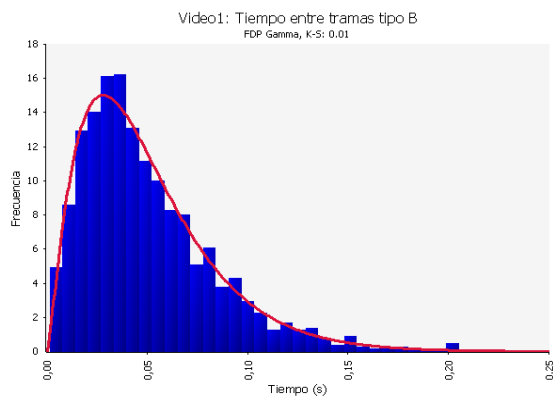
b) Tamaño de tramas tipo I. Weibull



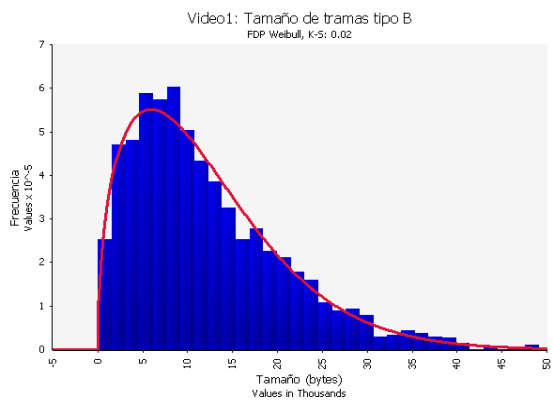
c) Tiempo entre tramas tipo P. Weibull



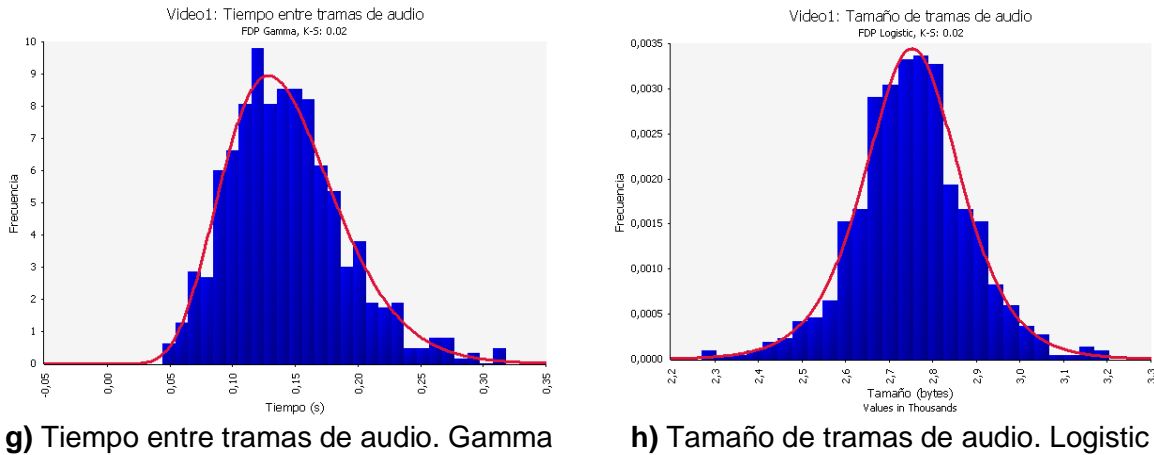
d) Tamaño de tramas tipo P. Lognormal



e) Tiempo entre tramas tipo B. Gamma



f) Tamaño de tramas tipo B. Weibull



**Figura 4-11.** Caracterización de los resultados de la simulación del video1 mediante la herramienta @RISK

	Tiempo entre tramas	Tamaño de tramas
<b>Tramas I</b>	Logistic, $\mu=0,4$ , $\sigma=0,0263$	Weibull, $\mu=54066,3$ , $\sigma=6837$
<b>Tramas P</b>	Weibull, $\mu=0,1381$ , $\sigma=0,0697$	Lognormal, $\mu=37960,5$ , $\sigma=9544,3$
<b>Tramas B</b>	Gamma, $\mu=0,0505$ , $\sigma=0,0334$	Weibull, $\mu=12167,2$ , $\sigma=9000,8$
<b>Tramas de audio</b>	Gamma, $\mu=0,1439$ , $\sigma=0,0466$	Logistic, $\mu=2753,2$ , $\sigma=131,7$

**Tabla 4-2.** Resumen de la caracterización realizada con @RISK a los resultados de la simulación del video1 en el modelo desarrollado en OPNET Modeler

En la Tabla 4-2 también se puede apreciar que los parámetros media y variación estándar de las FDPs determinadas para las componentes del video1, mediante la simulación del modelo en OPNET Modeler, son bastante similares a los parámetros de las FDPs para el video1, transmitido por medio del servidor real de VoD en laboratorio. En la Tabla 4-3 están descritas las diferencias de cada parámetro entre la Tabla 4-2 y la Tabla 3-1, entre paréntesis se encuentra la diferencia porcentual con los datos de la Tabla 3-1.

	Tiempo entre tramas (s)		Tamaño de tramas (bytes)	
	Media ( $\mu$ )	Variación estándar ( $\sigma$ )	Media ( $\mu$ )	Variación estándar ( $\sigma$ )
<b>Tramas I</b>	0,0005 (0,1%)	0,002 (7%)	645,2 (1%)	244,7 (3%)
<b>Tramas P</b>	0,0023 (1%)	0,0009 (1%)	486,3 (1%)	52,5 (0,5%)
<b>Tramas B</b>	0,0003 (0,5%)	0,0004 (1%)	103,6 (0,8%)	1949,6 (20%)
<b>Tramas de audio</b>	0,0045 (3%)	0 (0%)	10,1 (0,3%)	7,4 (5%)

**Tabla 4-3.** Diferencia entre los parámetros media y variación estándar de las Tablas 3-1 y 4-

Se puede apreciar en la Tabla 4-3 que las diferencias de los parámetros, entre los datos reales y los simulados, de cada FDP de las componentes para el video1 tienen un promedio de 2,86% con picos de 20% para la variación estándar de las tramas tipo B y 0,1% para la media de las tramas tipo I. Con estos resultados se puede concluir que el modelo desarrollado en el simulador de redes OPNET Modeler representa al conjunto servidor – enlace – cliente, con un margen de error de 2,86%.

### 4.1.3 ns-3

ns-3 no es solo una herramienta para simulación y emulación de redes sino que es en sí misma un proyecto de desarrollo de software en el cual participan reconocidas Universidades en el mundo como INRIA, la Universidad de Washington (como miembros fundadores), la Universidad Técnica de Berlín, entre muchas otras; además de empresas y grupos de investigación orientados a redes . Este proyecto nació alrededor del año 2006 y tiene una expectativa de trabajo de por lo menos 14 años más. La herramienta ns-3 ya dio sus primeros frutos con la implementación de redes P2P, CSMA, WiFi, LTE y MESH, y con cada versión entregada se crean más módulos, se fortalecen los existentes o se corrigen errores. La herramienta de simulación Ns-3 es un software libre, por lo que los usuarios pueden crear o modificar los modelos existentes. En el Anexo A, se detallan los conceptos más importantes para la creación de modelos de tráfico en ns-3, además de las distintas soluciones propuestas para afrontar el problema planteado en el presente proyecto.

#### 4.1.3.1 Descripción del modelo

##### Red simulada con la aplicación *OnOffApplication*

Esta Aplicación es un generador de tráfico que sigue un patrón *On/Off*. Después de que la aplicación empieza a correr, los estados *On* y *Off* se alternan. La duración de cada uno de estos estados está determinada por el valor de las variables *OnTime* y *OffTime*. Durante el estado *Off* no hay tráfico generado, mientras que durante el estado *On* si se genera tráfico.

Con el análisis sobre las trazas de tráfico real capturadas con Wireshark, en donde el escenario era el servidor de VoD y el cliente STB Amino; se obtuvieron las FDPs que describen el comportamiento de cada una de las componentes de los 3 videos (ver apartado 3.3.4.2).

##### Atributos de la aplicación

A continuación se detallan los atributos que se necesitan configurar para poder desarrollar el modelo mostrado en la Figura 4-12.

- **DataRate**: tasa de bit cuando la aplicación esta en el estado *On*.
- **PacketSize**: tamaño de los paquetes enviados en el estado *On*.
- **Remote**: dirección de destino.
- **OnTime**: variable usada para determinar la duración del estado *On*.
- **OffTime**: variable usada para determinar la duración del estado *Off*.
- **MaxBytes**: tamaño total del tráfico en bytes, que la aplicación envía.

- **Protocol:** protocolo a usar.

De las funciones obtenidas, el tamaño de tramas (I, P, B y audio) corresponde al estado *On* de la aplicación y el tiempo entre tramas corresponde al estado *Off*.

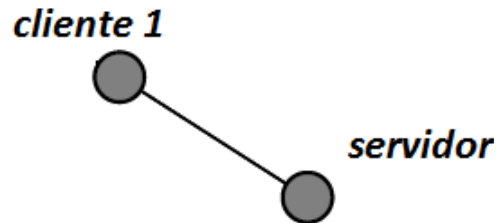


Figura 4-12. Modelo desarrollado en ns-3 a nivel de red

### Creación de ayudantes para la aplicación *OnOff* y de sumidero (*sink*)

En el código mostrado en la Figura 4-13 se define un ayudante para aplicaciones *OnOff* llamado *OnOffHelper*. Este es el encargado de configurar y manejar este tipo de aplicaciones. El constructor del ayudante recibe parámetros importantes que determinan las características de la aplicación. El primer parámetro define un socket para transmisión UDP (RTSP utiliza al protocolo UDP para la transmisión del contenido multimedia, y TCP para datos de control) y el segundo ya es el socket como tal; una dirección IP y un puerto.

Además de lo anterior, se define un ayudante de cernideros llamado *PacketSinkHelper* para recibir los datos enviados, también el constructor de este ayudante recibe los mismos parámetros que recibe el constructor del ayudante de topologías *OnOff*; una cadena de caracteres especificando un socket para transmisión UDP y el socket como tal.

```
OnOffHelper onoff ("ns3::UdpSocketFactory", Address (InetSocketAddress (Ipv4Address::GetAny (), port)));  
PacketSinkHelper sink ("ns3::UdpSocketFactory", Address (InetSocketAddress (Ipv4Address::GetAny (), port)));  
ApplicationContainer app;
```

Figura 4-13. Creación de ayudantes para la aplicación *OnOff* y de cernidero

### Configuración de los atributos de la Aplicación *OnOff*

Con el método *SetAttribute*, que se aprecia en la Figura 4-14, el ayudante de aplicaciones *OnOff*, fija los atributos de la aplicación. Este método recibe dos parámetros; el primero es una cadena de caracteres especificando el atributo de la aplicación que se desea fijar o modificar, y el segundo es el atributo como tal.

```
onoff.SetAttribute ("Remote",AddressValue (InetSocketAddress (Ipv4Address (csmaInterfaces.GetAddress(i)), port)));
```

Figura 4-14. Configuración del atributo *Remote* para la aplicación *OnOff*

En el código mostrado en la Figura 4-15 se fijan los atributos *Offtime*, *OnTime* y *Maxbytes*.

```
onoff.SetAttribute("OffTime",StringValue("ns3::LogNormalRandomVariable[Mu=0.4026,Sigma=0.0352]"));
onoff.SetAttribute("OnTime",StringValue("ns3::WeibullRandomVariable[Shape=10.2063,Scale=57480.9]"));
onoff.SetAttribute("MaxBytes",UintegerValue(10989173));
```

**Figura 4-15.** Configuración de los atributos *OffTime*, *OnTime* y *MaxBytes* para la aplicación *OnOff*

Los atributos *OffTime* y *OnTime* de la aplicación, reciben como parámetro por medio del método *SetAttribute*, una función probabilística, como se ve en la Figura 4-15. Esta función se obtuvo de la caracterización de las componentes de los videos. Para este caso particular la función Lognormal representa el tiempo entre tramas tipo I del video1, mientras que la función Weibull representa el tamaño de estas. Como se puede apreciar, estas FDPs no son iguales que las utilizadas para el modelo en OPNET Modeler, esto se debe a que únicamente un cierto número de funciones son aceptadas en ns-3, entre las que no se encuentran en el software está Logistic, de tal forma que se tuvo que cambiar las que no fueran soportadas por ns-3. Esto influye en el parámetro K-S registrado, que es el que mide la cercanía de los datos con la FDP a evaluar. "MaxBytes" recibe un valor entero del total de bytes que componen la trama I.

De esta manera, se describe en términos generales la implementación y configuración de un video por medio de la Aplicación *OnOff* con las características obtenidas del análisis de tráfico. Para los video 2 y 3 el proceso es análogo.

### Instalación de aplicaciones en los nodos

Con las instrucciones mostradas en la Figura 4-16, se extrae una aplicación del contenedor de aplicaciones, se la configura con los parámetros establecidos anteriormente en el ayudante de aplicaciones *OnOff*, se la instala en el nodo servidor y se fija el tiempo en el cual la función empieza a enviar datos.

```
app = onoff.Install (terminals3.Get (1));
app.Start (Seconds (1.0));
```

**Figura 4-16.** Instalación de aplicaciones en los nodos

### Recepción de datos en los clientes

Para que los datos sean recibidos en los nodos cliente es necesario adherir una aplicación cernidero (ver Figura 4-17). De manera similar, como en el caso del ayudante de aplicaciones *OnOff*, el ayudante de sumideros *SinkHelper*, por medio de su método *SetAttribute*, puede configurar los nodos cliente para recibir datos. El atributo *Local* hace referencia al nodo donde los datos se reciben, y el siguiente parámetro del método es un socket. Posteriormente se toma una aplicación del contenedor de aplicaciones y se la instala en el nodo que va a recibir la aplicación. Finalmente se fija el tiempo en que el nodo está listo para recibir datos.

```
sink.SetAttribute ("Local", AddressValue (InetSocketAddress (Ipv4Address (csmaInterfaces.GetAddress(i)), port)));  
app = sink.Install (terminals.Get (i));  
app.Start (Seconds (0.0));
```

**Figura 4-17.** Recepción de datos en los clientes

#### 4.1.3.2 Validación del modelo

La validación del modelo no fue posible realizarse en la herramienta de simulación ns-3 porque la aplicación utilizada no puede generar tramas como tal; si no que genera el grupo de bytes que la componen. Por ejemplo si dos tramas B se componen de 5KB cada una, la aplicación solamente generará 5KB y después de un tiempo los otros 5KB; sin saber que se trata de dos tramas B. Partiendo de este hecho no existe la forma de saber, qué grupo de bytes pertenece a que trama, y por consiguiente tampoco es posible conocer el tiempo entre ellas.

En el Anexo A se explica el por qué se escogió la aplicación *OnOff* para el desarrollo del modelo de simulación en ns-3. Para resumir lo dicho en el Anexo A, se puede decir que la utilización de contenedores virtuales (máquinas virtuales de Linux) para la creación de una red emulada, fue la opción que podía haber presentado mejores resultados, y que por trabajar con máquinas virtuales las cuales generan tráfico real entre ellas, habría sido fácil su captura con un analizador de protocolos, y por ende, se podía llevar a cabo la validación del modelo. Los problemas encontrados para la asociación de las máquinas virtuales con ns-3 (descritos en el Anexo A), es posible solucionarlos, pero hacerlo implicaría un tiempo demasiado extenso, que debido a los plazos acordados para el desarrollo del trabajo de grado, no se tiene a disposición. De tal forma, que esta tarea puede convertirse en un trabajo futuro que complementa el presente proyecto.

Por lo dicho anteriormente, se decidió proseguir con la investigación de ns-3, pero con la aplicación *OnOff* descrita en este capítulo. El hecho de que no se haya podido realizar la validación al modelo en ns-3, indica que este no es concluyente, es decir, no es posible comprobar que el mismo corresponda a la realidad. Sin embargo, uno de los intereses del desarrollo de este proyecto, fue explorar esta herramienta para dejar de precedente el uso de alternativas de software libre, por tal motivo, en el Anexo B se exponen los resultados obtenidos de la evaluación del modelo desarrollado en ns-3, en los distintos escenarios propuestos para casos de estudio en el apartado 3.2.

## 4.2 CUMPLIMIENTO DEL OBJETIVO ESPECÍFICO No. 2

Con lo realizado en el apartado 4.1.2, se da cumplimiento al objetivo específico No. 2 propuesto en el anteproyecto de este trabajo de grado, el cual es, "Obtener un modelo del comportamiento del tráfico generado por los servicios de una CAV sobre IPTV".

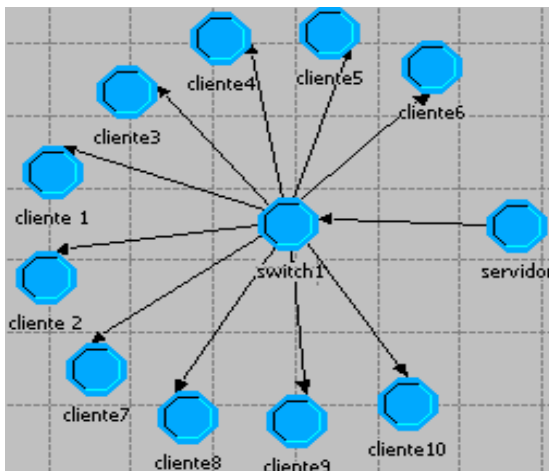
## CAPÍTULO 5

### ANÁLISIS DE RESULTADOS

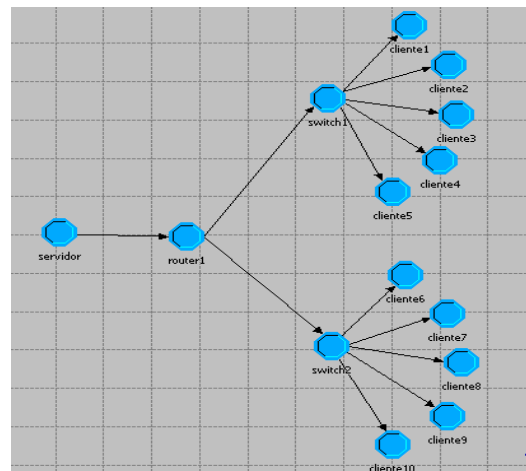
#### 5.1 APLICACIÓN DE LOS MODELOS DE SIMULACIÓN A LAS TOPOLOGÍAS PARA CASOS DE ESTUDIO

En el capítulo 3, apartado 3.2, se establecieron tres topologías distintas para realizar el análisis del modelo de simulación. Estas topologías fueron escogidas porque son de bastante uso en la implementación de redes LAN, pero también con la intención de establecer diferencias entre ellas en cuanto a parámetros de rendimiento de red como *throughput* y *delay*. En la Figura 5-1 aparecen las distintas topologías de red utilizadas para casos de estudio, diseñadas en OPNET Modeler. Como se puede apreciar, en la topología 3 hay dos escenarios; el primero de ellos está diseñado para una red con 10 usuarios, y el segundo para 20 usuarios, para poder analizar el rendimiento de la red cuando existe una sobrecarga de información. Las otras dos topologías tienen 10 usuarios cada uno, este número de usuarios se decidió ya que, analizando las características de los videos, es un número que pueden soportar las tres topologías

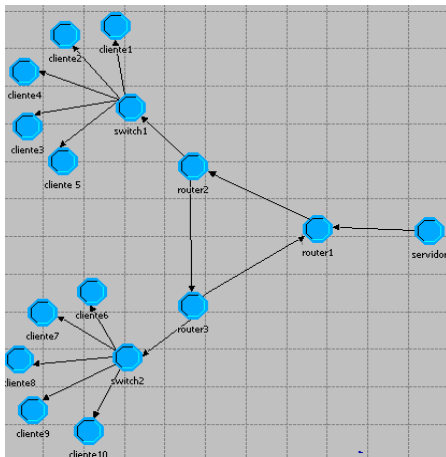
En todas las topologías mostradas a continuación, se utilizan switches para la conexión de nodos (clientes). Estos clientes están numerados de tal manera que el cliente 1 es el primero que es atendido por cada switch y el cliente 5 es el último (orden de conmutación). En el caso de la topología 3, escenario 2, y de la topología 1, se aplica el mismo concepto pero con 10 usuarios por cada switch.



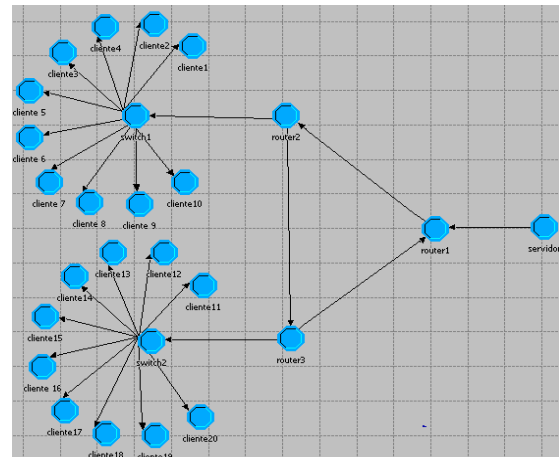
e) Topología1 – OPNET Modeler



f) Topología2 – OPNET Modeler



g) Topología3, escenario1 – OPNET Modeler



h) Topología3, escenario2 – OPNET Modeler

**Figura 5-1.** Topologías de red para casos de estudio

El objetivo de la simulación es obtener los resultados de los parámetros de rendimiento de redes descritos en el apartado 5.2 para el peor de los casos en cada topología. Esto es, en el hipotético caso de que todos los usuarios hagan la petición del mismo video al mismo tiempo. Con esto es posible determinar los recursos de red necesarios para atender a un número de usuarios en el peor de los casos. Por lo tanto, la simulación comienza en el momento en el que el servidor empieza a transmitir los datos requeridos por el usuario y termina cuando el video ha finalizado. En el Anexo D, se ha resumido el procedimiento para la construcción de topologías en ns-3.

## 5.2 ANÁLISIS DE RESULTADOS

En la Figura 5-1 se muestran las topologías para los casos de estudio. En este apartado se presenta gráfica y textualmente los resultados obtenidos para las simulaciones de cada video en cada topología, para luego hacer un análisis comparativo entre las topologías. Debido a lo extenso de la información, los resultados gráficos presentados en este apartado corresponden únicamente a la simulación del video1 sobre las topologías descritas anteriormente. Para los otros dos videos se presentan tablas con el resumen de los resultados obtenidos para ellos. Los resultados gráficos obtenidos para los videos 2 y 3, se presentan en el Anexo B.

Para realizar el análisis de los resultados, se midieron los siguientes parámetros de red: *throughput*, retardo, retardo de colas y tamaño de colas. No se muestran los resultados de pérdida de paquetes y BER (*Bit Error Rate*) debido a que estos tuvieron un valor de cero para todos los casos.



Todas las gráficas mostradas a continuación fueron obtenidas de OPNET Modeler y exportadas en forma de texto para poder tabularlas en Excel y que tengan una mejor visualización.

Antes de proseguir con la presentación de los resultados es prudente definir los parámetros de rendimiento de red que se están midiendo [43]:

**Throughput.** Medida de la capacidad o uso de una red (cantidad de información transmitida por unidad de tiempo), usualmente se mide en Mbps o Kbps.

**Retardo:** Tiempo transcurrido entre el momento en el que el servidor “coloca” un paquete en el medio de transmisión, hasta que este llega a su destino. En redes LAN se considera un retardo aceptable menos de 200ms.

**Retardo de colas:** Tiempo que una tarea en la red espera en una cola de un dispositivo de red hasta ser ejecutada. Este retardo es proporcional a la cantidad y velocidad de procesamiento de los dispositivos de red.

**Tamaño de colas:** Tamaño (usualmente en bytes) de las colas de espera para la ejecución de tareas en un dispositivo de red. Los switches modernos tienen un espacio de hasta 300MB para colas.

### **5.2.1 Topología 1**

La primera topología a usar es una en estrella (ver apartado 3.2 y Figura 5-1). Se utiliza esta topología debido a su sencillez y a que es comúnmente utilizada en redes LAN. A continuación se describen las características de esta topología.

Tipo de topología: Estrella

Ancho de banda: 100Mbps

Número de clientes: 10

A continuación se muestran las gráficas de los resultados obtenidos para los parámetros de rendimiento de red, para la simulación del video1 en la topología1 (ver Figura 5-1).

En la Figura 5-2 se observa la gráfica del promedio de retardo para los clientes 1 y 10. Como se aprecia, el retardo para el cliente 10 es significativamente mayor que para el cliente 1, aproximadamente de 15ms (aproximadamente 42ms para el cliente1 y 57ms para el cliente10) en promedio. Esta diferencia de retardos se debe al procesamiento realizado por el switch, además de otros factores como las colas de ejecución.

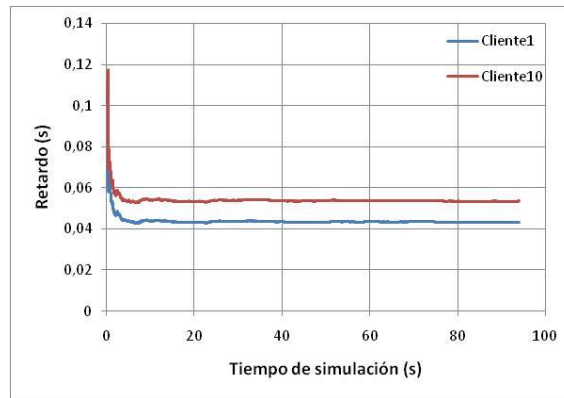
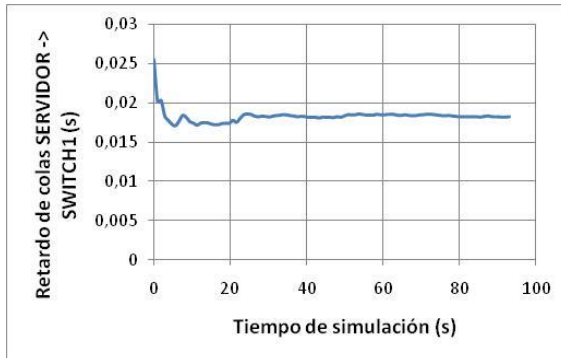
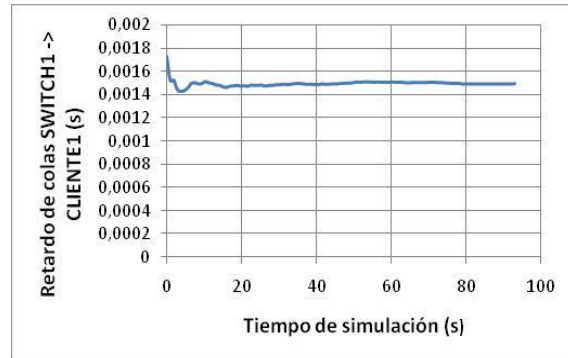


Figura 5-2. Topología1. Retardo para los clientes 1 y 10



a) Retardo de colas entre servidor y switch1



b) Retardo de colas entre switch1 y cliente1

Figura 5-3. Topología1. Retardo de colas

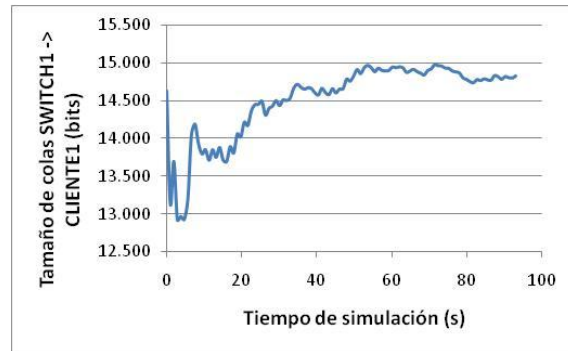
La Figura 5-3 muestra el retardo de colas de dos secciones importantes de la red, en la parte a) se observa el promedio del retardo de colas existente entre el servidor y el switch1. Este valor se encuentra en promedio en los 18ms. En la parte b) se aprecia el promedio del retardo de colas que hay entre el switch1 y el cliente1, como se observa, este valor está en promedio en 1,5ms. Esta diferencia de retardo de colas se debe a que para que el servidor transmita la información del video 1 al switch1, este debe procesar 10 veces (debido al número de clientes) la información necesaria para que el switch1 transmita el video al cliente1. Lo dicho anteriormente se puede verificar observando que la diferencia entre los dos retardos es de aproximadamente 10 veces.

La Figura 5-4 muestra las gráficas de tamaño de colas entre el servidor y el switch1 y entre el switch1 y el cliente1. Se observa que el tamaño de colas entre el servidor y el switch1 es en promedio de aproximadamente 1,2Mbits, mientras que entre el switch1 y el cliente1 es de aproximadamente 14,5Kbits. Al igual que para las gráficas de retardo de colas, la diferencia entre la gráfica de las partes a) y b) de la Figura 5-4 se debe a que el servidor debe procesar mucha más información para transmitir el video 1 al switch1, que la que tiene que procesar el

switch1 para transmitir el video 1 al cliente1. Esta vez, la diferencia no es de 10 veces, como para el retardo de colas, sino que es de aproximadamente 100 veces, de lo cual se puede deducir que el tamaño de colas aumenta linealmente a medida que aumentan los usuarios pero con un factor de 10.



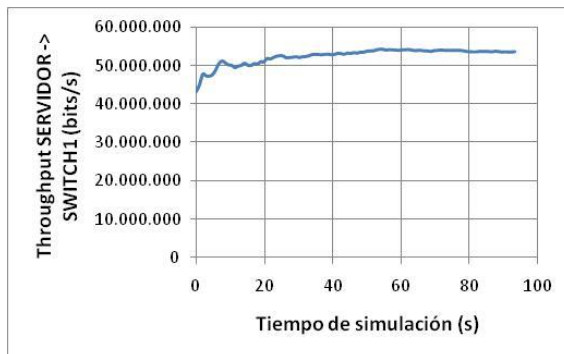
a) Tamaño de colas entre servidor y switch1



b) Tamaño de colas entre switch1 y cliente1

**Figura 5-4.** Topología1. Tamaño de colas

En la Figura 5-5 se muestran los resultados de la simulación para el *throughput* entre el servidor y el switch1 (parte a), y entre el switch1 y el cliente1 (parte b). El valor del *throughput* entre el servidor y el switch1 es de aproximadamente 52Mbps, mientras que entre el switch1 y el cliente1 es de aproximadamente 5,2Mbps. En este caso la diferencia es de exactamente 10 veces debido a que el servidor debe soportar la carga de 10 usuarios al tiempo para la transmisión del video1 al switch1, mientras que este último debe soportar la carga de tan solo un usuario para transmitir el video1 al cliente1.



a) *Throughput* entre servidor y switch1



b) *Throughput* entre switch1 y cliente1

**Figura 5-5.** Topología1. *Throughput*

## Resumen de resultados para la topología 1

En las Tablas 5-1, 5-2, 5-3 y 5-4, se han resumido los resultados obtenidos (valores aproximados) para los parámetros de rendimiento de red, de la simulación de cada video para la topología1. En el Anexo B se presentan los resultados gráficos obtenidos para la simulación de los videos 2 y 3 sobre la topología1.

Retardo (s)			
	Video1	Video2	Video3
<b>Cliente1</b>	42	42	36
<b>Cliente10</b>	57	57	41

**Tabla 5-1.** Topología1. Valores promedio aproximados del retardo para los clientes 1 y 10

Retardo de colas (ms)			
	Video1	Video2	Video3
<b>Servidor -&gt; Switch1</b>	18	16,5	9
<b>Switch1 -&gt; Cliente1</b>	1,5	1,35	1

**Tabla 5-2.** Topología1. Valores promedio aproximados del retardo de colas de ejecución entre servidor y switch1, y entre switch1 y cliente1

Tamaño de colas (Kbits)			
	Video1	Video2	Video3
<b>Servidor -&gt; Switch1</b>	1200	1100	500
<b>Switch1 -&gt; Cliente1</b>	14,5	12	6,5

**Tabla 5-3.** Topología1. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y switch1, y entre switch1 y cliente1

Throughput (Mbps)			
	Video1	Video2	Video3
<b>Servidor -&gt; Switch1</b>	52	44	28
<b>Switch1 -&gt; Cliente1</b>	5,2	4,4	2,8

**Tabla 5-4.** Topología1. Valores promedio aproximados del *throughput* entre servidor y switch1, y entre switch1 y cliente1

### 5.2.2 Topología 2

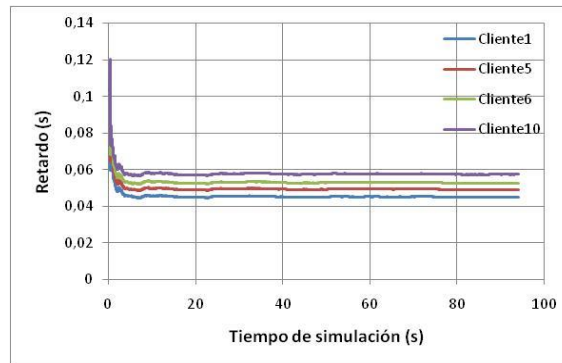
En el apartado 3.2 se definió la topología 2 como una topología tipo árbol (ver Figura 5-1). Al igual que la topología en estrella, la topología en árbol también es utilizada comúnmente en redes LAN. Como se ve en la Figura 5-1, esta topología consta de 2 switches, con cinco usuarios cada uno, conectados a un router, el cual está directamente conectado al servidor. A continuación se describen las características de esta topología.

Tipo de topología: Árbol

Ancho de banda: 100Mbps

Número de clientes: 10

Las gráficas que se muestran a continuación pertenecen a los resultados obtenidos, para los parámetros de rendimiento de red, de la simulación del video1 sobre la topología2.



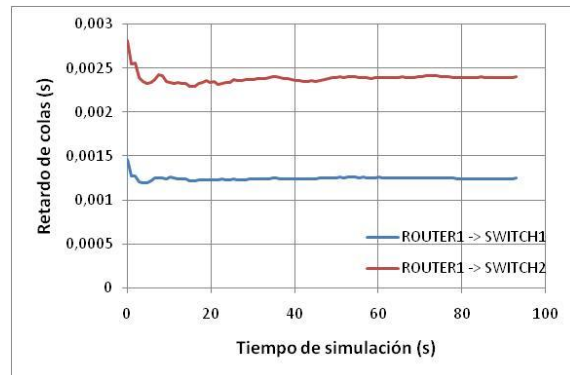
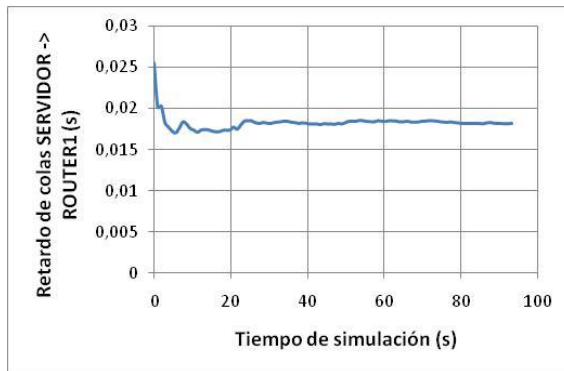
**Figura 5-6.** Topología2. Retardo

La Figura 5-6 muestra los retardos de paquetes correspondientes a los clientes 1 y 5 del switch1, y 6 y 10 del switch2. Como se puede apreciar, el retardo es mayor a medida que se avanza en el número del cliente. Esta diferencia de retardo se debe a que el servidor está configurado para que atienda un cliente a la vez, comenzando por el cliente1 y terminando en el cliente10. En la topología1 se vio que sucedía lo mismo, pero en los resultados mostrados para ella, solo se apreció el retardo para dos clientes. En este caso, se lo hace para cuatro clientes para apreciar el rendimiento en cada uno de los switches.

El promedio de estos resultados es de aproximadamente es de 45ms para el cliente1, 50ms para el cliente5, 54ms para el cliente6 y 59ms para el cliente10. Si se toman en cuenta los retardos del cliente1 y el cliente10, se puede observar que estos son ligeramente mayores (aproximadamente 3ms y 2ms respectivamente) a los resultados obtenidos para el retardo de los clientes 1 y 10 de la topología1. Esto indica que el aumento de procesamiento en la topología2, debido a que hay un router y dos switches, ha aumentado en promedio, aproximadamente 2,5ms el retardo apreciado.

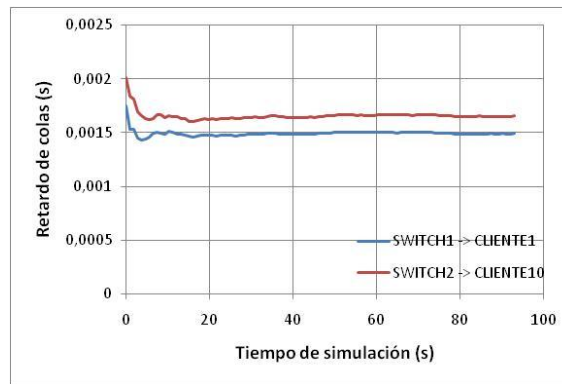
En la Figura 5-7 se muestran los valores promedio obtenidos para el retardo de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y switch2 y cliente10. Estos valores son en promedio, aproximadamente de 18ms entre servidor y router1 (valor idéntico al del video1 para la topología1), 1,25ms entre router1 y switch1, 2,4ms entre router1 y switch2, 1,5ms entre switch1 y cliente1, y 1,7ms entre switch2 y cliente10. Se puede apreciar que el valor promedio de retardo de colas de ejecución entre

router1 y switch1 es menor al valor promedio entre switch1 y cliente1, además de entre switch2 y cliente10.



a) Retardo de colas entre servidor y router1

b) Retardo de colas para router1



c) Retardo de colas para switch1 y para switch2

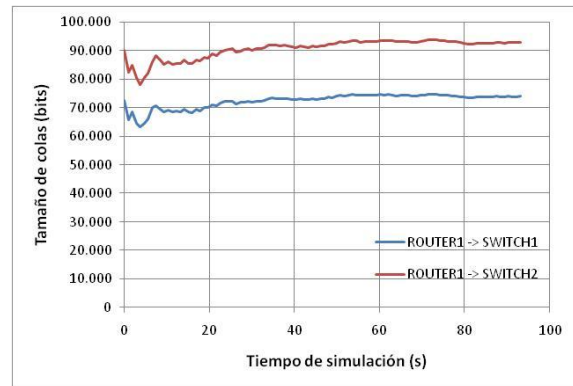
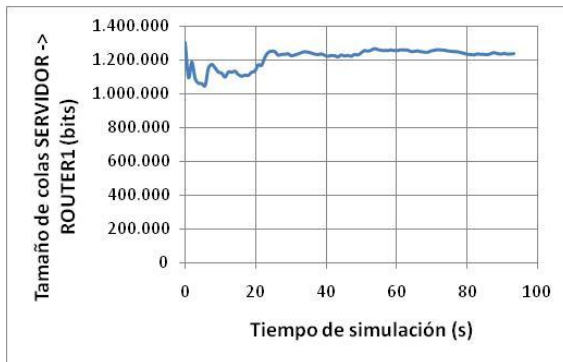
**Figura 5-7.** Topología2. Retardo de colas

La diferencia de retardos existente entre router1 y switch1, y router1 switch2, se debe a que el servidor está configurado para atender primero los clientes del switch1, por lo tanto, el router1 transmite primero la información correspondiente al switch1, mientras que la información correspondiente al switch2 espera en cola.

En OPNET Modeler existe una aplicación en la cual se puede observar el tránsito de paquetes por la red. En esta aplicación se pudo observar que al principio de la simulación los paquetes transmitidos por el switch2, tienen un tiempo de espera en relación a los paquetes transmitidos por el switch1. Este tiempo es apreciable en la diferencia de retardo de colas de ejecución entre switch1 y cliente1, y switch10 y cliente10, ya que los paquetes llegan al switch2 pero son encolados por un tiempo de aproximadamente 2ms.

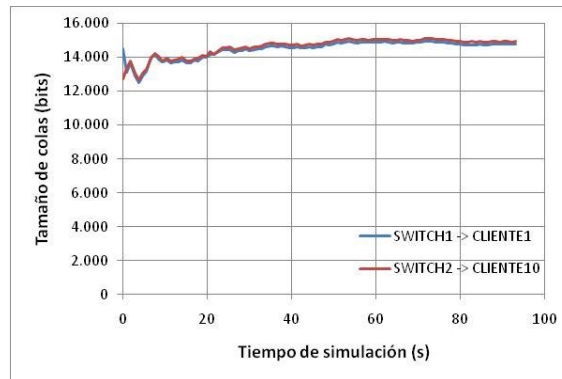
La Figura 5-8 muestra los resultados obtenidos para el promedio del tamaño de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y

switch2 y cliente10. El tamaño de colas de ejecución entre servidor y router1 es de 1,2Mbits, igual a lo que se observó para este mismo video en la topología1.



a) Tamaño de colas entre servidor y router1

b) Tamaño de colas para router1



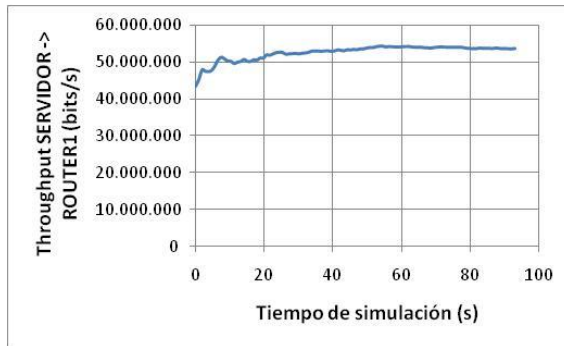
c) Tamaño de colas para switch1 y para switch2

**Figura 5-8.** Topología2. Tamaño de colas

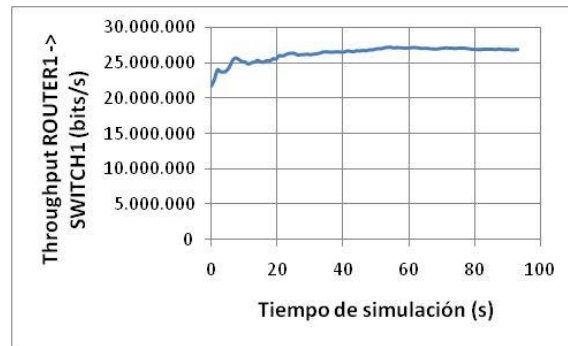
También se aprecia en la Figura 5-8 que el valor promedio para el tamaño de colas de ejecución es de aproximadamente 71Kbits entre router1 y switch1, y 90Kbits entre router1 y switch2. Mientras que el tamaño de colas de ejecución entre switch1 y cliente1, y switch2 y cliente10 es igual, con un valor de aproximadamente 14,5Kbits, valor que es idéntico al encontrado para el video1 en la topología1. La diferencia existente entre los tamaños de colas de ejecución entre router1 y switch1, y router1 y switch2, se debe al mismo argumento dado para la diferencia entre estos dos enlaces para el retardo de colas.

Haciendo una comparación con los resultados obtenidos para el video1 en la topología1 (ver Tabla 5-3), se observa que el tamaño de colas de ejecución entre switch1 y cliente1, y entre switch2 y cliente10, es el mismo que el encontrado entre switch1 y cliente1 para la topología1.

La Figura 5-9 muestra los resultados del promedio del *throughput* entre servidor y router1, router1 y switch1, y switch1 y cliente1. Para el caso del *throughput*, las gráficas entre router1 y switch2, y entre switch2 y cliente10, son exactamente iguales a las gráficas entre router1 y switch1, y switch1 y cliente1, por tal motivo, solo se exponen estas últimas. Los resultados obtenidos para el *throughput* entre servidor y router1, y entre switch1 y cliente1, son iguales a las obtenidas para servidor y switch1, y switch1 y cliente1 del video1, topología1 (aproximadamente 52Mbps y 5,2Mbps respectivamente).



a) *Throughput* entre servidor y router1



b) *Throughput* entre router1 y switch1



c) *Throughput* entre switch1 y cliente1

**Figura 5-9.** Topología2. *Throughput*

La parte b) de la Figura 5-9 muestra el resultado del *throughput* entre router1 y switch1. Como se puede apreciar, este tiene un promedio de aproximadamente 26Mbps, lo que corresponde a 5 veces la tasa de bit del video, lo cual es de esperarse debido a que cada switch transmite el video a 5 usuarios.



## Resumen de resultados para la topología 2

En las Tablas 5-5, 5-6, 5-7 y 5-8, se han resumido los resultados obtenidos (valores aproximados) para los parámetros de rendimiento de red, de la simulación de cada video para la topología2. Los resultados gráficos de los videos 2 y 3 se encuentran en el Anexo B.

Retardo (ms)			
	Video1	Video2	Video3
<b>Cliente1</b>	45	45	39
<b>Cliente5</b>	50	49	40
<b>Cliente6</b>	54	53	41
<b>Cliente10</b>	59	58	42

**Tabla 5-5.** Topología2. Valores promedio aproximados del retardo para los clientes 1, 5, 6 y 10

Retardo de colas (ms)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	18	17	8,2
<b>Router1 -&gt; Switch1</b>	1,25	1,1	0,75
<b>Router1 -&gt; Switch2</b>	2,4	2,3	1,2
<b>Switch1 -&gt; Cliente1</b>	1,5	1,35	0,9
<b>Switch2 -&gt; Cliente10</b>	1,7	1,5	1

**Tabla 5-6.** Topología2. Valores promedio aproximados del retardo de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y switch2 y cliente10

Tamaño de colas (Kbits)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	1200	1100	400
<b>Router1 -&gt; Switch1</b>	71	60	25
<b>Router1 -&gt; Switch2</b>	90	75	31
<b>Switch1 -&gt; Cliente1</b>	14,5	12	5
<b>Switch2 -&gt; Cliente10</b>	14,5	12	5

**Tabla 5-7.** Topología2. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y router1, router1 y switch1, router1 y switch2, switch1 y cliente1, y switch2 y cliente10

Throughput (Mbps)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	52	44	28
<b>Router1 -&gt; Switch1</b>	26	22	14
<b>Switch1 -&gt; Cliente1</b>	5,2	4,4	2,8

**Tabla 5-8.** Topología2. Valores promedio aproximados del *throughput* entre servidor y router1, router1 y switch1 y switch1 y cliente1

### 5.2.3 Topología 3 - Escenario 1

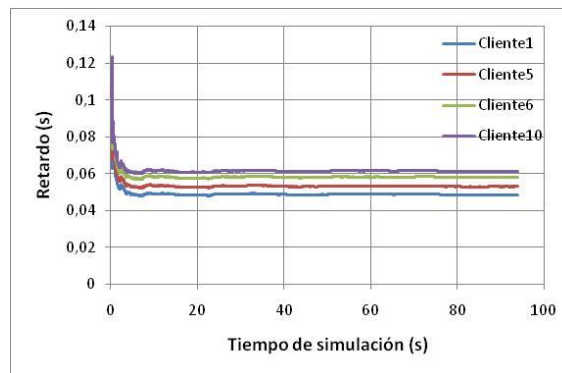
En el apartado 5.1 (Figura 5-1) se definieron dos escenarios distintos para la topología3 definida en el apartado 3.2. En este apartado se recogerán los resultados obtenidos para la simulación de los tres videos en el primer escenario. Este tipo de topología (anillo) es una de las topologías más usadas para redes LAN. A continuación se describen sus características.

Tipo de topología: Anillo

Ancho de banda: 100Mbps

Número de clientes: 10

A continuación se muestran los resultados obtenidos para los parámetros de rendimiento de red obtenidos mediante la simulación del video1 en la topología3 – escenario1.

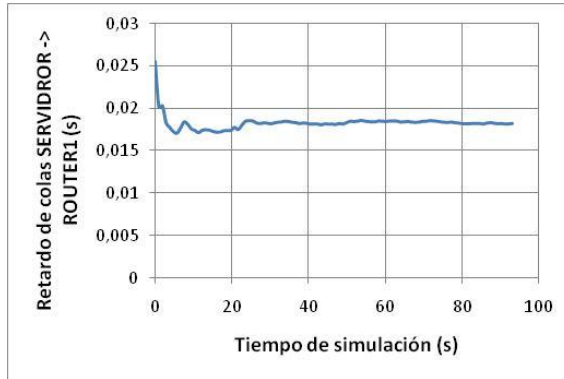


**Figura 5-10.** Topología3 – Escenario1. Retardo

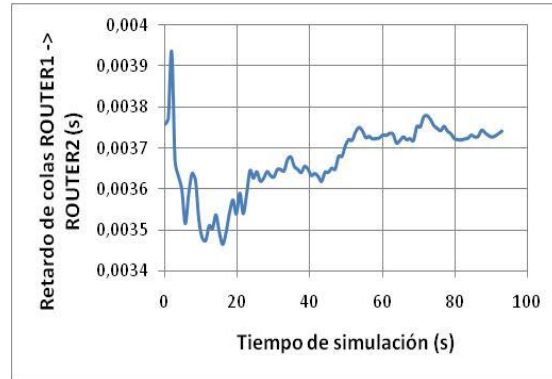
La Figura 5-10 muestra los resultados obtenidos para los retardos de los clientes 1, 5, 6 y 10. Como se puede apreciar, los valores para estos retardos son en promedio, de aproximadamente 50ms para el cliente1, 55ms para el cliente5, 60ms para el cliente6 y 61ms para el cliente10. Hasta este punto, el retardo producido en los clientes para el video1 en la topología3 – escenario1, ha sido mayor que para el mismo video en las otras dos topologías vistas anteriormente, con un aumento promedio con respecto a la topología2 de alrededor de 4,5ms.

Los resultados obtenidos para el retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10, se muestran en la Figura 5-11. Estos valores son en promedio, de aproximadamente 18ms entre servidor y router1, 3,65 ms entre router1 y router2, 2,3ms entre router2 y router3, 1,5ms entre switch1 y cliente1, y 1,7ms entre switch2 y cliente10. Los valores para el retardo de colas de ejecución entre servidor y router1, y entre switch1 y cliente1, coinciden con los obtenidos para el video1 en la topología2 y también en la topología1, lo que indica que el cambio de topología no ha afectado el retardo de colas de ejecución en estos enlaces. Al igual que sucedió para la topología2, se observa que existe una diferencia de retardos entre los enlaces para switch1 y

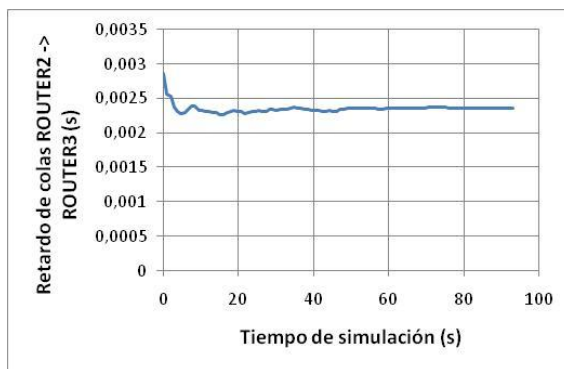
cliente1, y para switch2 y cliente10. La explicación a este fenómeno se dio anteriormente para la topología2 y también aplica para la topología que es objeto de estudio en este apartado.



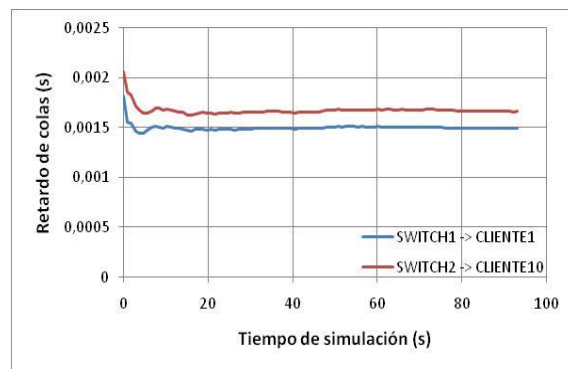
a) Retardo de colas entre servidor y router1



b) Retardo de colas entre router1 y router2



c) Retardo de colas entre router2 y router3



d) Retardo de colas para switch1 y para switch2

**Figura 5-11.** Topología3 – Escenario1. Retardo de colas

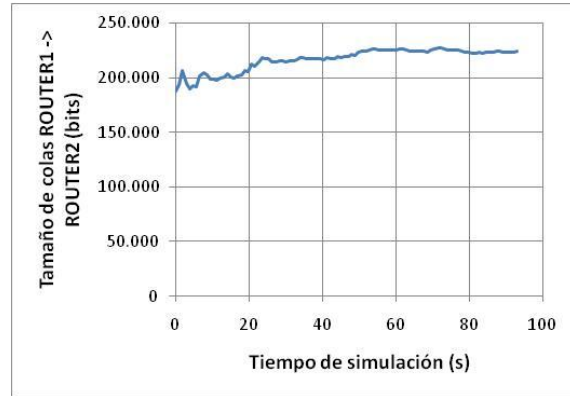
La Figura 5-12 muestra los resultados obtenidos para el tamaño promedio de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10. Los valores de estos resultados son de aproximadamente 1,2Mbits entre servidor y router1, 215Kbits entre router1 y router2, 85Kbits entre router2 y router3, y 14,5Kbits para los enlaces correspondientes a switch1 y cliente1, y switch2 y cliente10. Los valores para servidor y router1, y para switch1, y cliente1 y switch2 y cliente10 son iguales que los obtenidos para la topología2 y también para la topología1, lo cual demuestra que en el caso del tamaño de colas de ejecución, al igual que pasa con el retardo de colas de ejecución, no ha sido afectado por el cambio de topologías.

Para terminar con los resultados del video1 sobre la topología3 – escenario1, la Figura 5-13 muestra los resultados obtenidos para el *throughput* entre servidor y router1, router2 y

router3, y switch1 y cliente1. Se puede observar que los resultados obtenidos son idénticos a los de la topología2, lo que es un resultado esperado, ya que se están haciendo mediciones en enlaces que están soportando la misma cantidad de clientes para ambas topologías.



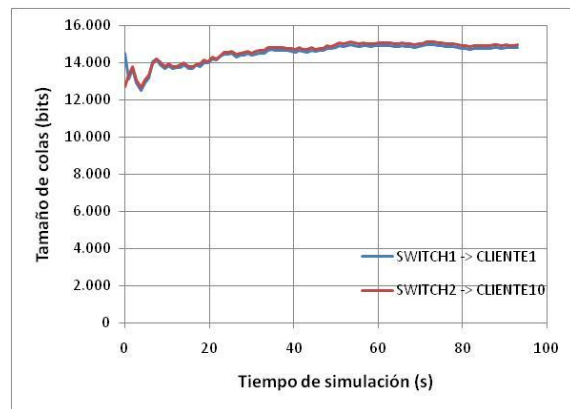
a) Tamaño de colas entre servidor y router1



b) Tamaño de colas entre router1 y router2

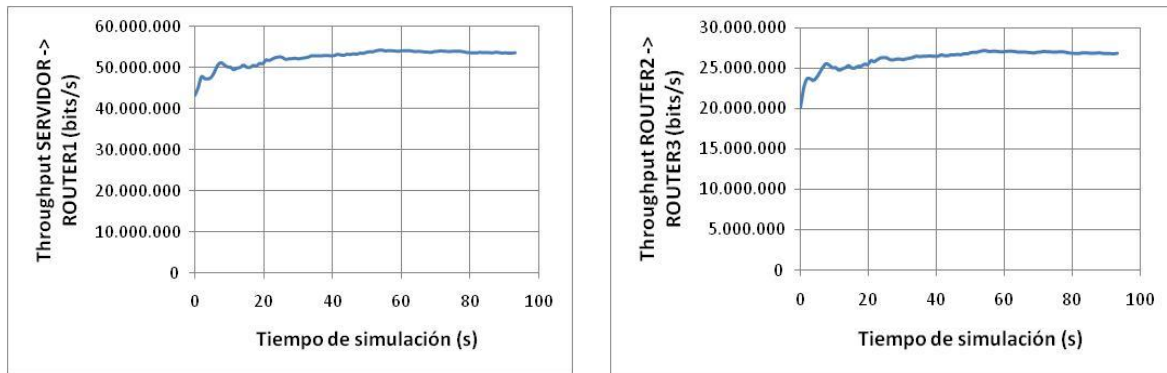


c) Tamaño de colas entre router2 y router3



d) Tamaño de colas para switch1 y para switch2

**Figura 5-12.** Topología3 – Escenario1. Tamaño de colas



a) *Throughput* entre servidor y router1

b) *Throughput* entre router2 y router3



c) *Throughput* entre switch1 y cliente1

**Figura 5-13.** Topología3 – Escenario1. *Throughput*

**Resumen de resultados para la topología 3 – escenario 1**

En las Tablas 5-9, 5-10, 5-11 y 5-12, se han resumido los resultados obtenidos (valores aproximados) para los parámetros de rendimiento de red, de la simulación de cada video para la topología3 – escenario1. Los resultados gráficos de los videos 2 y 3 se encuentran en el Anexo B.

	Retardo (ms)		
	Video1	Video2	Video3
<b>Cliente1</b>	50	50	40
<b>Cliente5</b>	55	55	42
<b>Cliente6</b>	60	59	44
<b>Cliente10</b>	61	60	45

**Tabla 5-9.** Topología3 – Escenario1. Valores promedio aproximados del retardo para los clientes 1, 5, 6 y 10

Retardo de colas (ms)			
	Video1	Video2	Video3
Servidor -> Router1	18	17	8,3
Router1 -> Router2	3,65	3,5	1
Router2 -> Router3	2,3	2,2	1,3
Switch1 -> Cliente1	1,5	1,35	0,9
Switch2 -> Cliente10	1,7	1,5	1

**Tabla 5-10.** Topología3 – Escenario1. Retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10

Tamaño de colas (Kbits)			
	Video1	Video2	Video3
Servidor -> Router1	1200	1100	500
Router1 -> Router2	215	175	100
Router2 -> Router3	85	75	40
Switch1 -> Cliente1	14,5	12	6,5
Switch2 -> Cliente10	14,5	12	6,5

**Tabla 5-11.** Topología3 – Escenario1. Tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente10

Throughput (Mbps)			
	Video1	Video2	Video3
Servidor -> Router1	52	44	28
Router2 -> Router3	26	22	14
Switch1 -> Cliente1	5,2	4,4	2,8

**Tabla 5-12.** Topología3 – Escenario1. *Throughput* entre servidor y router1, router1 y router2 y router3, y switch1 y cliente1

#### 5.2.4 Topología 3 – Escenario 2

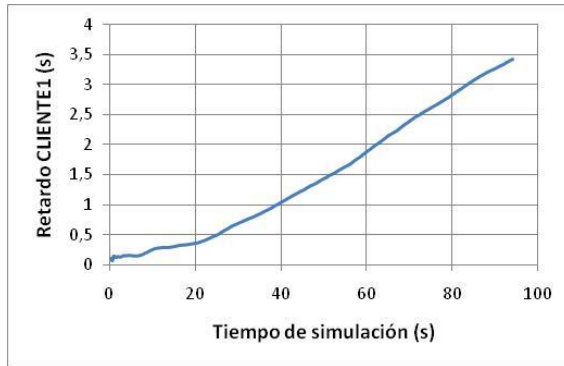
Hasta el momento se han mostrado los resultados de las simulaciones de los tres videos en distintas topologías, todas con 10 usuarios. Esto se hace debido a que es un número de usuarios no crítico, el cual pueden manejar las tres topologías con un ancho de banda de 100Mbps. En este apartado se mostrarán los resultados obtenidos para la simulación de los tres videos en la topología3, pero esta vez con 20 usuarios (ver Figura 5-1), para poder analizar qué pasa con los parámetros de rendimiento de red cuando esta se satura. Hay que advertir que este escenario es ideal, es decir, los dispositivos de red pueden funcionar a su mayor rendimiento posible (velocidad de transmisión completa y capacidad infinita de colas), algo que no es factible en la realidad. Sin embargo, con una cantidad de 20 usuarios se asegura que exista una saturación evidente al menos para el video1. A continuación se muestran las características de la topología3 – escenario2.

Tipo de topología: Anillo

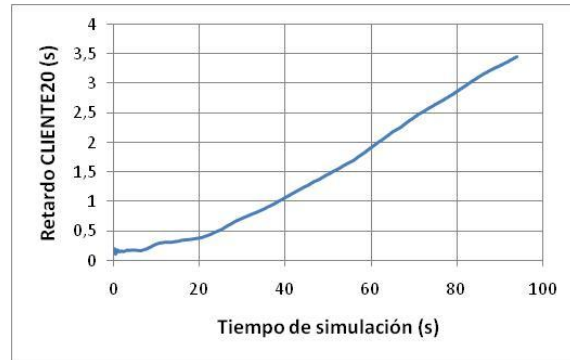
Ancho de banda: 100Mbps

Número de usuarios: 20

Las figuras relacionadas a continuación, muestran los resultados obtenidos para los parámetros de rendimiento de red, para la simulación del video1 sobre la topología3 – escenario2.

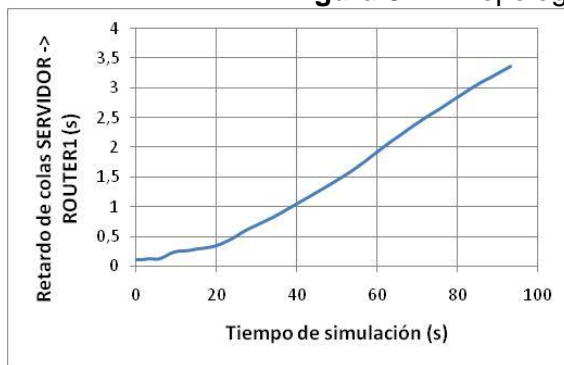


a) Retardo para el cliente1

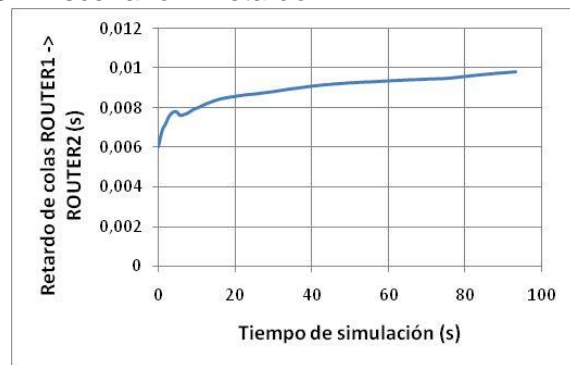


b) Retardo para el cliente20

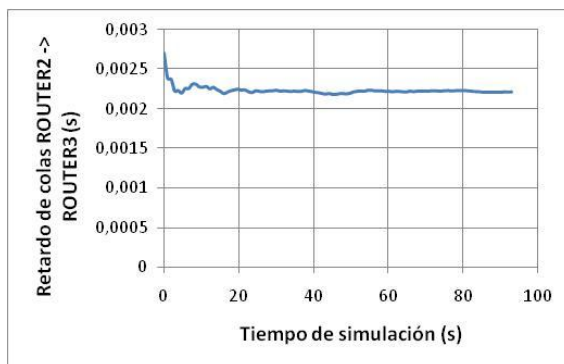
**Figura 5-14. Topología3 – Escenario2. Retardo**



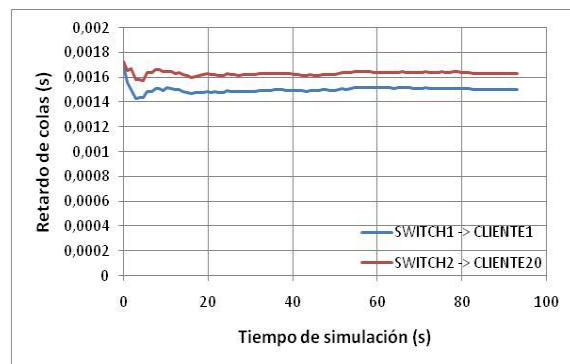
a) Retardo de colas entre servidor y router1



b) Retardo de colas entre router1 y router2



c) Retardo de colas entre router2 y router3



d) Retardo de colas para switch1 y para switch2

**Figura 5-15. Topología3 – Escenario2. Retardo de colas**

La Figura 5-14 muestra el retardo obtenido para los usuarios 1 y 20. Se puede observar que en esta ocasión no se mostró el retardo de varios clientes en una sola gráfica. Esto se hizo debido a que el retardo promedio fue el mismo para todos los clientes, por lo tanto solo se muestran los casos más extremos vistos en las otras topologías, para indicar este hecho.

Como se puede ver, a diferencia de lo que ocurre en la simulación del video1 en las otras topologías, esta vez el promedio del retardo no toma un valor estable, sino que aumenta progresivamente, esto indica, que efectivamente hubo una saturación de tráfico en el servidor y que los paquetes se fueron encolando cada vez más a medida que avanzaba el video. El valor promedio de estos retardos es de aproximadamente 1,5s para cada cliente, lo cual es un tiempo en extremo grande para el rendimiento aceptable de una red LAN, en la que el tiempo máximo de retardo aceptable es de 200ms.

En la Figura 5-15 se muestran los resultados de retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20.

En la Figura 5-15 se observan los resultados obtenidos para el retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20. Estos resultados tienen valores promedio de aproximadamente 1,5s entre servidor y router1, 9ms entre router1 y router2, 2,3ms entre router2 y router3, 1,5ms entre switch1 y cliente1, y 1,6ms entre switch2 y cliente20.

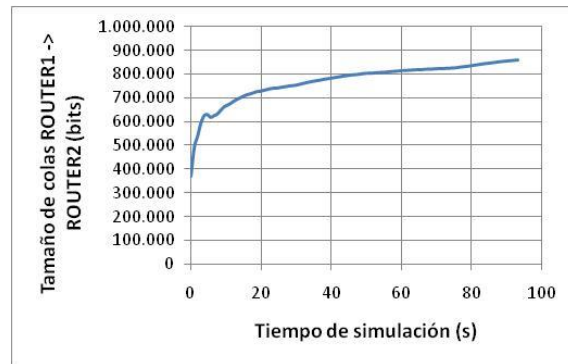
El retardo de colas de ejecución registrado para el enlace entre servidor y router1 es de un tiempo muy grande en comparación con los resultados obtenidos para las anteriores tres topologías para este mismo video. Sin embargo después del router1, el tiempo de retardo de colas baja significativamente e iguala los tiempos de retardo de colas de ejecución observados para este mismo video en la topología3 – escenario1, para los enlaces correspondientes a router2 y router3, switch1 y cliente1, y switch2 y cliente20 (ver Tabla 5-10). Esto indica que cuando el servidor trabaja a su máxima velocidad de transmisión, satura sus colas de ejecución, lo cual impide que este envíe suficiente información como para que el router1 tenga que saturar también sus colas de ejecución, lo cual se puede apreciar en la Figura 5-15, ya que a pesar de que el retardo de colas de ejecución ha bajado considerablemente para el enlace entre router1 y router2, este aún es tres veces mayor que el medido para el mismo enlace en la topología3 – escenario1. De tal manera que a medida que se avanza en la red, la cantidad de información transmitida por cada dispositivo disminuye hasta estabilizarse. De todas maneras, esto no quiere decir que la información esté llegando a tiempo, ya que solo en el servidor, el retardo de colas de ejecución es de 1,5s.

La Figura 5-16 muestra los resultados obtenidos para el tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20. Los valores promedio de estos resultados son aproximadamente 175Mbits entre servidor y router1, 750Kbits entre router1 y router2, 150Kbits entre router2 y router3, y 12,8Kbits para los enlaces de switch1 y cliente1, y de switch2 y cliente20.





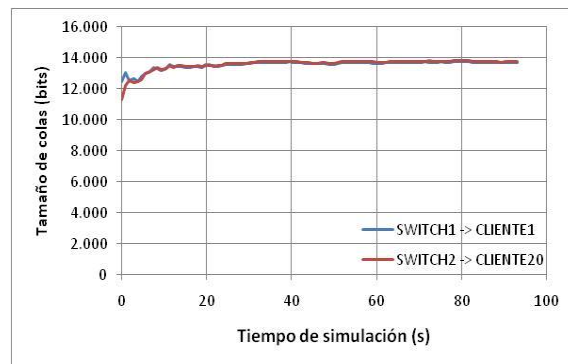
a) Tamaño de colas entre servidor y router1



b) Tamaño de colas entre router1 y router2



c) Tamaño de colas entre router2 y router3

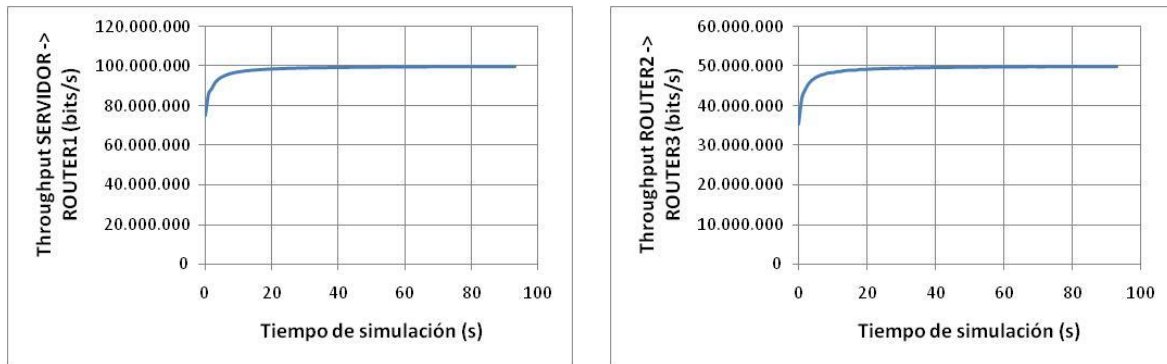


d) Tamaño de colas para switch1 y para switch2

**Figura 5-16.** Topología3 – Escenario2. Tamaño de colas

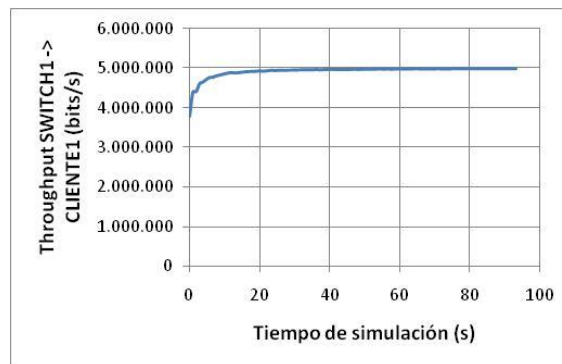
La Figura 5-16 muestra los resultados obtenidos para el tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20. Los valores promedio de estos resultados son aproximadamente 175Mbits entre servidor y router1, 750Kbits entre router1 y router2, 150Kbits entre router2 y router3, y 12,8Kbits para los enlaces de switch1 y cliente1, y de switch2 y cliente20.

Como se puede observar, para el tamaño de colas de ejecución sucede algo parecido a lo que pasa con el retardo de colas de ejecución. El tamaño de las colas entre servidor y router1 es demasiado grande en comparación con las otras topologías, pero después este tamaño va disminuyendo (lo argumentado para el retardo de colas de ejecución también se aplica para este caso), hasta incluso tener un tamaño menor que la topología3 – escenario1 para los enlaces de switch1 y cliente1, y de switch2 y cliente20. Esto indica que la información que llega a los switches no alcanza a tener la misma tasa de bit original del video.



a) *Throughput* entre servidor y router1

b) *Throughput* entre router2 y router3



c) *Throughput* entre switch1 y cliente1

**Figura 5-17.** Topología3 – Escenario2. *Throughput*

En la Figura 5-17 se observan los resultados obtenidos para el *throughput* entre servidor y router1, router2 y router3, y switch1 y cliente1. Se puede apreciar que el *throughput* entre servidor y router1 es de 100Mbps la mayoría del tiempo de simulación, lo que indica que el servidor está trabajando a su máxima capacidad. En el capítulo 4 se vio que la tasa de bit para el video1 es de 5,2Mbps, si se multiplica este valor por 20, que es el número de usuarios de la topología3 – escenario2, la velocidad de transmisión atender las 20 peticiones al mismo tiempo es de 104Mbps, lo que explica por qué el *throughput* entre router2 y router3 es de exactamente 50Mbps, ya que debería ser de 52Mbps; y por qué el *throughput* entre switch1 y cliente1 es de exactamente 5Mbps y no llega a ser de 5,2Mbps, que es la tasa de bit del video1. Esto muestra que transcurridos los 94s de simulación para el video1, no se ha transmitido toda la información del video.

### Resumen de resultados para la topología 3 – escenario 2

En las Tablas 5-13, 5-14, 5-15 y 5-16, se han resumido los resultados obtenidos (valores aproximados) para los parámetros de rendimiento de red, de la simulación de cada video para la topología3 – escenario2. Los resultados gráficos de los videos 2 y 3 se encuentran en el Anexo B.

Retardo (ms)			
	Video1	Video2	Video3
<b>Cliente1</b>	1500	250	70
<b>Cliente10</b>	1500	260	75
<b>Cliente11</b>	1500	270	78
<b>Cliente20</b>	1500	280	100

**Tabla 5-13.** Topología3 – Escenario2. Valores promedio aproximados del retardo para los clientes 1, 10, 11 y 20

Retardo de colas (ms)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	1500	270	40
<b>Router1 -&gt; Router2</b>	9	15	2,8
<b>Router2 -&gt; Router3</b>	2,3	2,2	1,3
<b>Switch1 -&gt; Cliente1</b>	1,5	1,28	1
<b>Switch2 -&gt; Cliente20</b>	1,6	1,5	1

**Tabla 5-14.** Topología3 – Escenario2. Valores promedio aproximados del retardo de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20

Tamaño de colas (Kbits)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	175000	25000	3000
<b>Router1 -&gt; Router2</b>	750	1300	200
<b>Router2 -&gt; Router3</b>	150	140	40
<b>Switch1 -&gt; Cliente1</b>	12,8	12	5,5
<b>Switch2 -&gt; Cliente20</b>	12,8	12	5,5

**Tabla 5-15.** Topología3 – Escenario2. Valores promedio aproximados del tamaño de colas de ejecución entre servidor y router1, router1 y router2, router2 y router3, switch1 y cliente1, y switch2 y cliente20

Throughput (Mbps)			
	Video1	Video2	Video3
<b>Servidor -&gt; Router1</b>	100	88	56
<b>Router2 -&gt; Router3</b>	50	44	28
<b>Switch1 -&gt; Cliente1</b>	5	4,4	2,8

**Tabla 5-16.** Topología3 – Escenario2. Valores promedio aproximados del *throughput* entre servidor y router1, router2 y router3, y switch1 y cliente1

### 5.3 CUMPLIMIENTO DEL OBJETIVO ESPECÍFICO No. 3

Con lo realizado en el apartado 5.2.1, se da cumplimiento al objetivo específico No. 2, propuesto en el anteproyecto de este trabajo de grado, el cual es, “Evaluar mediante

simulación, el comportamiento del tráfico generado por los servicios de una CAV sobre IPTV”. Con el cumplimiento de este objetivo, también se cumple el objetivo general del proyecto, el cual es, “Modelar el tráfico de los servicios de una Comunidad Académica Virtual (CAV) sobre IPTV”.

## **CAPÍTULO 6**

### **CONCLUSIONES Y TRABAJOS FUTUROS**

En el presente capítulo se presentan las conclusiones extraídas del desarrollo del presente trabajo de grado. Además se plantean trabajos futuros para la continuación del estudio de tráfico en los proyectos de T-Learning, en particular, en los desarrollados en la Universidad del Cauca.

#### **6.1 CONCLUSIONES**

Esta tesis desarrolla un modelo de tráfico que tiene en cuenta los aspectos enumerados a continuación, y que responden a la pregunta de investigación: “¿Cuál es el modelo de tráfico para los servicios de una CAV, sobre IPTV?”.

- Videos consumidos de manera real en el laboratorio del ST-CAV.
- Descripción del tráfico de dos aplicaciones asociadas al contenido.
- Tráfico individual por componentes de los videos.
- Protocolo RTSP para VoD.
- Modelado de transmisión de los videos.
- Validación de los resultados.
- Modelo completo de red en OPNET y ns-3.
- Modelo de VoD mediante simulación por eventos discretos, esto es, construcción mediante lenguaje de programación de las características telemáticas de los videos (donde se recoge la duración de los videos, el protocolo RTSP, la complejidad de los videos, etc.) y de un aplicación asociada al contenido. En OPNET Modeler esto se realizó mediante la programación en C/C++ de los estados de una máquina de estados finitos.
- Generación de 4 diferentes topologías de red, donde se desplego el modelo de VoD y análisis de los resultados.
- Congestión de la red mediante el incremento de usuarios.

A continuación se presentan las conclusiones extraídas de cada capítulo de la monografía, por lo tanto este apartado se divide en cuatro secciones, una por cada capítulo.

### **6.1.1 Capítulo 2**

En el capítulo 2 se exploró el estado del arte de conceptos importantes para el presente trabajo de grado, por lo tanto, las conclusiones citadas a continuación se extraen de estos conceptos.

- En IPTV la tecnología multicast es clave, ya que permite la difusión de señales de datos a múltiples usuarios sin la necesidad de utilizar grandes porciones de ancho de banda. Esta tecnología es la principal característica para que IPTV pueda ser competente en el servicio de televisión interactiva y constantemente está en desarrollo para que pueda ser escalable fácilmente.
- El modelado de tráfico de una red es el primer paso crítico para el diseño y posterior implementación de una red. Este proceso permite identificar fortalezas y debilidades, y delimitar el alcance de la red. Con esto es posible corregir problemas antes de implementar la red, lo cual conlleva a un ahorro tanto de tiempo como de dinero.
- Una parte muy importante del modelado de tráfico es la parte de simulación de la red. Es necesario contar con un software que permita realizar esta tarea con relativa facilidad ya que es aquí donde se creará una red lo suficientemente grande como para analizar el comportamiento de la misma con el tráfico capturado con anterioridad.

### **6.1.2 Capítulo 3**

- El servidor de streaming de video LIVE555, mostró ser un software competitivo para la transmisión de información de video a través de una red LAN, ya que no se presentaron inconvenientes en la transmisión y reproducción de videos de distintas tasas de bit. Su modo de operación está basado en el protocolo de transporte RTSP, por medio del cual vuelca el contenido del video en la red.
- A través del software libre ffmpeg, es posible editar un video para que este concuerde con las necesidades de transporte de IPTV (codecs MPEG-2 para video, y aac para audio), como lo son formatos y codecs. Además, con ffmpeg se puede empaquetar los archivos de video en flujos de transporte (MPEG-TS), para que estos puedan ser reproducidos en algunos dispositivos, como STBs, que requieren este formato en la información que consumen.
- Para que un usuario pueda utilizar funciones como adelanto, retroceso y pausa, en la reproducción de un video consumido de un servidor de VoD, este video debe estar indexado. Para tal fin, el software UniversalIndexer demostró ser eficiente y efectivo, gracias a su fácil usabilidad.
- El analizador de protocolos Wireshark es una herramienta capaz de capturar todo el tráfico que fluye a través del dispositivo al que está conectado. Con la ayuda de los filtros adecuados, es posible separar el tráfico capturado en distintos tipos de tramas o flujos, para poder analizarlos por separado. Para el caso de este proyecto estas tramas son I, P, B y audio.

- Debido a su tamaño, las tramas MPEG-2 (I, P, B y audio) son divididas en secciones que correspondan al tamaño del campo payload o "cara útil" de una trama Ethernet, con el fin de ser transportadas a través del canal correspondiente.
- Para la correcta caracterización del tráfico de un video en formato MPEG-2, se deben caracterizar cada una de sus componentes, las cuales son: tramas I, P, B y audio. Como se mostró en el presente proyecto de grado, hacer esto da como resultado una mayor aproximación al comportamiento real del tráfico. Esto es clave para el desarrollo de modelos de simulación más precisos, y por ende, para un mejor planeamiento de la implementación de servicios que formatos IPTV.
- Para caracterizar el comportamiento del tráfico generado por un tipo de trama MPEG-2 es necesario caracterizar por separado, tanto el tamaño, como el tiempo entre tramas. Para esto es necesario conocer además del tamaño de cada una de estas tramas, el tiempo de diferencia entre la transmisión de ellas.
- Una Función de Distribución de Probabilidad (FDP), puede describir el comportamiento del tráfico producido por las componentes de una trama MPEG-2 (tamaño y tiempo entre tramas), si los datos de estas componentes se acercan al definido por la FDP. Esto es posible calcularlo matemáticamente con la ayuda del test de Kolmogorov-Smirnov, el cual entrega el parámetro K-S, el cual mide la distancia máxima entre las Funciones de Distribución Acumulada (CDF) entre las FDPs teóricas y los datos de las componentes de la trama MPEG-2 con la cual se las está relacionando.

### **6.1.3 Capítulo 4**

- En el nivel de abstracción del nodo, este puede contener distintos módulos, los cuales se pueden asociar a distintos procesos. Estos módulos se conectan por medio de "líneas de flujo", las cuales transmiten información de un módulo a otro. Estas "líneas de flujo" tienen un código tanto para su entrada (cola), como para su salida (punta). Estos códigos son útiles cuando existe más de una línea saliendo o entrando en un módulo. Estos códigos pueden ser utilizados en el código de los estados de los procesos, para hacer referencia a un paquete de entrada o de salida.
- La validación realizada para el video1 arrojó datos de número de tramas para la simulación en OPNET Modeler, que difieren en un 3,3% de los capturados en laboratorio. Esta diferencia se explica debido a la naturaleza aleatoria del tráfico producido por un video con formato MPEG-2. A diferencia de esto, los datos arrojados por la simulación del video en el modelo, no son tan aleatorios, ya que estos provienen directamente de FDPs.
- La validación para el modelo desarrollado en OPNET Modeler, mostró que este difiere en un 2,86% de la realidad, lo cual es un valor aceptable para el modelo. Esto muestra la eficiencia del simulador en la creación de este tipo de modelos.
- Ns-3 es un simulador de red, el cual requiere fundamentos bastantes buenos en el lenguaje de programación C. si bien el simulador cuenta con su propio lenguaje de programación, este lenguaje es basado en C.

- Ns-3 todavía es un proyecto joven que necesita madurar, lo que quiere decir que se deben adelantar trabajos no solamente para investigación de topologías sino que también trabajos para mejorar el simulador en sí.
- Debido a que no es posible hacer una validación del modelo creado en ns-3 mediante la aplicación *OnOff*, no se puede comprobar que este modelo represente la realidad.

#### 6.1.4 Capítulo 5

- Para el retardo, en las simulaciones en OPNET Modeler, se presentaron los menores tiempos en la topología1, debido a que esta es la que menos procesamiento debe realizar, ya que solo posee un switch. A diferencia de esto, la topología2 debe realizar el procesamiento de un router y dos switches, mientras que la topología3 lo debe hacer para tres routers.
- El nivel de procesamiento de cada topología influyo en los parámetros de rendimiento de retardo, retardo de colas y tamaño de colas. En estos, se vio que ha medida que el nivel de procesamiento aumenta, también aumentan sus valores.
- Se aprecia que los parámetros de red medidos entre el servidor y el primer dispositivo al que este se conecta, y entre un switch y un cliente, se mantienen constantes a pesar del cambio de topologías, lo que hace concluir, que como es de esperarse, el tipo de topología no influye en estos enlaces.
- Para OPNET Modeler, el *throughput* medido en los distintos enlaces, fue aproximadamente igual al *throughput* esperado, dependiendo del número de usuarios y la tasa de bit del video1. Esto no ocurrió para ns-3 debido a que su tuvo que utilizar FDPs distintas para la simulación en este.
- A diferencia del retardo de colas, el tamaño de colas no se vio alterado en su valor en el enlace final para el cliente, es decir, para lo dos enlaces entre switch y cliente que se midió este parámetro en cada topología, se presentó el mismo valor.
- Teniendo en cuenta los resultados obtenidos para la topología3 - escenario2 en OPNET Modeler, se puede concluir que no es posible que una red con un ancho de banda de 100Mbps, soporte la petición de 20 usuarios, de un video con tasa de bit de 5,2Mbps. Esto quiere decir, que a pesar de que se puede implementar un servicio de VoD en esta red, potencialmente podría presentar fallos de disponibilidad.
- Los resultados obtenidos para la topología3 - escenario2 en OPNET Modeler, muestra que con 20 usuarios y un ancho de banda de 100Mbps, el servidor trabaja a su máxima capacidad el 80% del tiempo, y a pesar de esto no logra transmitir toda la información a tiempo, lo cual influye en el retardo para los clientes, y también en el retardo de colas y tamaño de colas para el servidor, haciendo que estas describan una cura hacia arriba que se extiende a lo largo de la duración del video.
- En la topología3 - escenario2, en OPNET Modeler, se ve que a medida que se avanza en la red, los valores de retardo de colas y tamaño de colas disminuyen drásticamente, lo cual lleva a pensar, que debido a las colas que forma cada router, el siguiente debe soportar mucha menos carga. De esto se deduce que existe un cuello de botella entre servidor y router1, en el que se está congestionando el tráfico.



- En las gráficas de *throughput* para la topología3 - escenario2 en OPNET Modeler, se ve que el enlace entre router2 y router3 soporta una carga de 50Mbps, mientras que el enlace entre switch1 y cliente1 soporta 5Mbps, esto demuestra que la información no está llegando a tiempo, debido a que estos valores deberían ser de aproximadamente 52Mbps y 5Mbps.
- Para ns-3, los valores de retardo en las topologías 1 y 2 se mantuvieron por debajo del milisegundo. En la topología 3 y 10 clientes se vio aumento en el retardo causado principalmente por la adición los routers y los enlaces entre estos. También se le suma el retardo de los enlaces entre los enrutadores y el procesamiento de los paquetes dispositivos de red.
- En ns-3, en la topología3 – escenario2, se vio un porcentaje de pérdida de paquetes de aproximadamente del 10% del total. Y el valor de retardo sobrepasaba los 10 milisegundos. Se concluye que la topología3 – escenario2 no es capaz de soportar el consumo de los videos al mismo tiempo.
- Los quipos de red de ns-3 tienen velocidades de procesamiento diferentes a las de los dispositivos reales, esto se refleja en los valores retardo encontrados para los paquetes enviados.
- En la topología3 – escenario2, en ns-3, la congestión de la red mostro similitud con el mundo real. En el mundo real las redes nunca llegan a funcionar al cien por ciento de la capacidad, sino que por el contrario existe un factor de pérdida de rendimiento en los equipos reales.
- La pérdida de paquetes se presentó porque en tamaño de las colas que pueden soportar los dispositivos en ns-3 es insignificante. En el mundo real los quipos de red tienen buffers que almacenan mucha información en las colas para prevenir la pérdida de paquetes. El tamaño de estos buffers puede variar según sea el modelo o la marca del equipo de red.
- ns-3 es un software de simulación de redes que aún necesita mucho desarrollo, pero que tiene gran potencial como herramienta de fuente abierta.
- OPNET Modeler es una herramienta bastante versátil y poderosa, que permite que un desarrollador cree modelos de simulación del mundo real con gran exactitud.
- El número de clientes soportados por una red que utiliza IPTV para difusión de un servicio de VoD, depende de las características de los videos a transmitir. En el análisis de resultados se vio que para un video de las características del video 3, una red con 100Mbps fácilmente puede soportar 20 usuarios sin que haya congestión, sin embargo para las características de los videos 1 y 2 no sucedió lo mismo. En estos se presentó congestión y hubieron valores de retardo de paquetes de hasta 7 veces lo recomendado para una red LAN.
- Para una CAV como la del proyecto ST-CAV resulta útil un servidor como LIVE555 para prestar el servicio de VoD sobre IPTV. El problema con este servidor es que no se pueden implementar otros servicios asociados al contenido, ni tampoco los no asociados al contenido, por lo tanto se deberían implementar dos servidores distintos para este caso.

## 6.2 TRABAJOS FUTUROS

- Desarrollar un modelo del comportamiento del tráfico generado por los usuarios que realizan peticiones a una CAV. Este trabajo complementaría el trabajo de grado de este documento, con la implementación de un canal de retorno, por medio del cual los usuarios realicen peticiones de videos y aplicaciones, dependiendo de FDPs con las que se haya caracterizado su comportamiento. Esto solamente es posible mediante el análisis del comportamiento de usuarios que utilicen los servicios de una CAV totalmente implementada.
- Estudiar el comportamiento del tráfico de los servicios de una CAV sobre otras infraestructuras tecnológicas, como TV Móvil. Este trabajo es importante debido a que el uso de los servicios de una CAV, y de la televisión digital en general, mayormente se lleva a cabo mediante otras infraestructuras tecnológicas, que no son IPTV.
- Estudiar y caracterizar el comportamiento del tráfico generado por el servicio de VoD, mediante la utilización de protocolos que no sean de IPTV, como la entrega de video mediante servidores Flash. Esto podría brindar un método de comparación para la eficiencia de estos protocolos.
- Investigar un servidor (de fuente abierta como LIVE555) que permita implementar servicios de valor agregado (asociados o no al contenido) aparte del servicio de VoD. Esto ayudaría al desarrollo de las CAV (entre ellas, el proyecto ST-CAV) al disminuir costos y tiempos tanto en desarrollo como en implementación y haría más viable el desarrollo de distintas aplicaciones y/o servicios en ámbitos tecnológicos como IPTV, en el que por el momento no existe esta posibilidad debido a que servidores como LIVE555 sólo se utilizan para servicios de streaming de audio y video.

Los trabajos futuros enunciados a continuación están orientados únicamente al mejoramiento de la herramienta Ns-3. Esto es debido a que existen aún problemas con la herramienta y por consiguiente se cree conveniente plantear soluciones a estos problemas técnicos.

- Las emulaciones con máquinas virtuales en ns-3 son posibles, pero no existe una validación y/o un funcionamiento óptimo de los módulos que permiten la conexión entre la maquina virtual y la red ns-3. Como casos particulares, hay problemas de enrutamiento, y los datos que pasan por la red son alterados o dañados. Como trabajo futuro se propone corregir y validar el modulo de compatibilidad principalmente en estos dos problemas enunciados.
- La aplicación *UdpTraceClientServerApplication* es de mucho interés cuando se trata de caracterizar el tráfico de video, lastimosamente existe un problema de generación de tráfico cuando la tramas de video son demasiado grandes. Como trabajo futuro se propone corregir y modificar la aplicación para que funcione con tamaños de tramas de video más grandes.
- Las funciones de distribución que se usan para modelar flujos de datos en Ns-3, son escasas. Se propone como trabajo futuro implementar más funciones de distribución que serían de utilidad para posteriores trabajos de modelado de tráfico con la aplicación *OnOffApplication*.

- La obtención de estadísticas en ns-3 no es una tarea simple y por este motivo muchos trabajos de investigación no se realizan sobre esta herramienta. Como se vio en este trabajo, la herramienta *Flowmonitor* entrega estadísticas con valores medios al final de la simulación. Como trabajo futuro se propone la implementación de un módulo para obtener estadísticas en función del tiempo y que en conjunto con la herramienta *Gnuplot* entreguen también estas estadísticas con sus respectivas graficas.

## **AGRADECIMIENTOS**

Agradecemos especialmente al Magister Wilmar Yesid Campo Muñoz por la dirección de este trabajo de grado, así como por su dedicación y paciencia para la exitosa culminación del mismo. Ofrecemos también nuestros agradecimientos al PhD. Jose Luis Arciniegas por su co-dirección en el proyecto, como también por la gestión realizada para facilitarnos los distintos elementos existentes en el Laboratorio de Televisión Digital para realizar los experimentos necesarios para el desarrollo del trabajo de grado.

A todo el equipo de trabajo del proyecto ST-CAV por brindarnos su ayuda cuando fue necesaria. A nuestras familias, por el apoyo y paciencia brindados durante todo nuestro proceso formativo en la Universidad del Cauca. A nuestros amigos y colegas de la Facultad de Ingeniería Electrónica y Telecomunicaciones por su valiosa amistad, consejos y conocimientos compartidos.

Agradecemos también al PhD. Roberto García Fernández, profesor titular del Departamento de Informática de la Universidad de Oviedo (España) por su soporte en el uso de la herramienta de simulación OPNET Modeler para llevar a cabo este trabajo, el cual esta en marcado dentro del proyecto de Doctorado “Modelo de Tráfico para Servicios Interactivos de una Comunidad Académica Virtual, con contenidos de Audio y Video de Alta Calidad” del cual él es su Co-Director.

Finalmente, a profesores, administrativos y toda la gente que forma parte de la Facultad de Ingeniería Electrónica y Telecomunicaciones por su valioso servicio a la sociedad, del cual nos hemos visto beneficiados.

## REFERENCIAS

- [1] Pacheco, Ibeth A.; Tibaquirá, Nestor M., “Marco de referencia para la creación y consolidación de Comunidades Académicas Virtuales en la Universidad del Cauca”, Tesis de pregrado, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, 2008
- [2] Gómez, Fredy L., “Estrategias de virtualidad en la educación rural el reto del e-learning 2.0 en los procesos de educación superior”, revista digital “La educ@ción”, Junio de 2010.
- [3] Blando, M., 2003. “Comunidades Académicas Virtuales: Compartir para mejorar”, (México, 2003).
- [4] <http://www.unicauca.edu.co/stcav>
- [5] *ATIS IPTV Exploratory Group Report and Recommendation to the TOPS Council*, Alliance for Telecommunication Industry Solutions, 2006-07.
- [6] Oliver, Budda; “Benefits of Internet Protocol Television (IPTV)”, 2007. Online. Disponible en [http://ezinearticles.com/?Benefits-of-Internet-Protocol-Television-\(IPTV\)&id=892172](http://ezinearticles.com/?Benefits-of-Internet-Protocol-Television-(IPTV)&id=892172)
- [7] Jokipelto, P. A., “T-learning Model for Learning via Digital TV”. Universidad de Tecnología de Helsinki. Finlandia. 2005.
- [8] Zemante, Gaby; Tique, Johan; “Arquitectura para la planificación dinámica de servicios y contenidos multimedia de t-learning”, Tesis de pregrado, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, 2011.
- [9] García P., Francisco J.; “Estado actual de los sistemas de e-Learning”, Teoría de la educación: Educación y Cultura en la sociedad de la información, ISSN 1138-9737, Vol. 6, 2005.
- [10] Galera Lozano, José; “El triángulo del e-Learning” (2004); artículo disponible en <http://www.noticias.com/el-triangulo-del-e-learning.36815>
- [11] C. Digital, “Aprendizaje virtual en un mundo real,” 2011. Disponible en <http://www.cencadedigital.com/7>
- [12] Meurant, R.C.;, "OntheObliquity of Ubiquity," *Multimedia and Ubiquitous Engineering*, 2007. *MUE '07. International Conference on* , vol., no., pp.862-867, 26-28 April 2007 doi: 10.1109/MUE.2007.166
- [13] Mejía, Juan S.; Samboní, John J.; “Alternativa de integración de un LMS con una pantalla interactiva de bajo coste”, Tesis de pregrado, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, 2012.
- [14] JianfengZhu, “Study on e-learning education model based on web intelligence,” in *International Conference on e-Education, e-Business, e-Management, and e-Learning*, 2010. IC4E '10. IEEE,2010, pp. 554–557. 8, 18
- [15] Chanchí G., Gabriel E.; Campo, Wilmar Y.; Amaya, Juan P.; Arciniegas, Jose L.; “Esquema de servicios para Televisión Digital Interactiva, basados en el protocolo REST-JSON”. Artículo. Congreso Ibero-americano de Telemática (CITA 2011), Gramado RS, Brasil. Mayo de 2011.

- [16] Duran, D.F.; Cerón Martínez, R.A.; Arciniegas Herrera, J.L.; , "Architecture for the support of the video on demand service for virtual academic communities on IPTV," Computing Congress (CCC), 2011 6th Colombian , vol., no., pp.1-7, 4-6 May 2011doi: 10.1109/COLOMCC.2011.5936300
- [17] Geib, M.; Braun, C.; Kolbe, L.; Brenner, W.; , "Measuring the utilization of collaboration technology for knowledge development and exchange in virtual communities," System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on , vol., no., pp. 10 pp., 5-8 Jan. 2004doi: 10.1109/HICSS.2004.1265091
- [18] Skyrme, D. J., Knowledge Networking: Creating the Collaborative Enterprise, Reed Educational and Professional Publishing, Woburn, 1999.
- [19] Preece, J. 2000. "Online Communities: Designing usability, supporting sociability". chapter 1. University of Maryland, Baltimore County. John Wileyand Sons, United Kingdom.
- [20] Rheingold, H. 1996. "The Virtual Community". Capítulo 1: The Heart of the WELL [en línea] Disponible en: <http://www.rheingold.com/vc/book/>. pp. 57-58. Consultado: Agosto27 de 2012
- [21] Hagel III, John, Armstrong, Arthur G. Net Gain. 1997. Harvard Business School Press, chapter 1. Disponible en [http://www.google.com.co/books?id=PI5mfbP5JgC&pg=PT45&source=gbs\\_toc\\_r&cad=0\\_0&sig=ACfU3U2dwZaMtB4BleAI4JaXfjXIGKDDsA#PPT193,M1](http://www.google.com.co/books?id=PI5mfbP5JgC&pg=PT45&source=gbs_toc_r&cad=0_0&sig=ACfU3U2dwZaMtB4BleAI4JaXfjXIGKDDsA#PPT193,M1)
- [22] Preece, J., Maloney-Krichmar, D. 2003 "Online Communities: Focusing on sociability and usability". In J. Jacko and A. Sears, A. (Eds.) Handbook of Human-Computer Interaction, Lawrence Erlbaum Associates Inc. Publishers. Mahwah: NJ. 596-620. University of Maryland, Baltimore County..
- [23] Jiang Wu; Junwei Zuo; , "Universal design for interactive digital television in China," Computer-Aided Industrial Design & Conceptual Design (CAIDCD), 2010 IEEE 11th International Conference on , vol.1, no., pp.688-691, 17-19 Nov. 2010doi: 10.1109/CAIDCD.2010.5681256
- [24] Benedito dos Santos Junior, J.B.; Braga, K.P.; de Souza, D.; Vilela, G.; Lima, D.; Diogo, W.; , "Towards interactivity for citizenship: An approach to interactive digital television," Consumer Communications and Networking Conference (CCNC), 2012 IEEE , vol., no., pp.723-728, 14-17 Jan. 2012doi: 10.1109/CCNC.2012.6180999
- [25] Prado, G.M.; Zorzo, S.D.; , "Interactive Service Provider Architecture for Interactive Digital Television systems," *Computer Information Systems and Industrial Management Applications (CISIM), 2010 International Conference on* , vol., no., pp.541-546, 8-10 Oct. 2010
- [26] Geng Yu; Westholm, T.; Kihl, M.; Sedano, I.; Aurelius, A.; Lagerstedt, C.; Odling, P.; , "Analysis and characterization of IPTV user behavior," *Broadband Multimedia Systems and Broadcasting, 2009. BMSB '09. IEEE International Symposium on* , vol., no., pp.1-6, 13-15 May 2009
- [27] Yang Xiao, Xiaojiang Du, Jingyuan Zhang, Fei Hu, Sghaier Guizani, "Internet Protocol Television (IPTV): The Killer Application or the Next Generation Internet", *IEEE Communications Magazine*, Vol. 45, Issue 11, pp 126-134, 2007.
- [28] Janevski, T.; Vanevski, Z.; , "Quality of Service analysis for IPTV provisioning," Telecommunications, 2009. ConTEL 2009. 10th International Conference on , vol., no., pp.443-450, 8-10 June 2009
- [29] Joseph Weber and Tom Newberry, *IPTV Crash Course*, McGraw-Hill, 2007.

- [30] Tim Rahrer, Nortel, Riccardo Fiandra, FastWeb, Steven Wright, BellSouth “ TR-126 Triple-play Services Quality of Experience (QoE) Requirements and Mechanisms For Architecture & Transport” , DSL Forum, February 21, 2006.
- [31] International Telecommunication Union-Telecommunication. Disponible en <http://www.itu.int/ITU-T>
- [32] Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN), European Telecommunications Standards Institute, Sophia Antipolis, France [Online]. Disponible en <http://www.etsi.org/tispan>
- [33] Open IPTV Forum. Disponible en <http://www.oipf.tv>
- [34] Alliance for Telecommunications Industry Solutions. Disponible en <http://www.atis.org>.
- [35] 3GPP. Disponible en <http://www.3gpp.org>
- [36] Zeadally, S.; Moustafa, H.; Siddiqui, F.; "Internet Protocol Television (IPTV): Architecture, Trends, and Challenges," *Systems Journal, IEEE* , vol.5, no.4, pp.518-527, Dec. 2011 doi: 10.1109/JSYST.2011.2165601
- [37] DVB. Digital Video Broadcast Project. Disponible en <http://www.dvb.org>
- [38] DVB-IPTV. DVB-IPTV: Standards and Blue Books. Disponible en <http://www.dvb.org/technology/standards>
- [39] Digital Video Broadcasting (DVB): Multimedia Home Platform (MHP) Specification 1.0.3, European Telecommunications Standards Institute, Sophia Antipolis, France, Tech. Rep. ETSI TS 201 812 V1.1.1 (2007-09).
- [40] MPEG [Online]. Disponible en <http://www.chiariglione.org/mpeg>
- [41] G. O'Deiscoll, Next Generation IPTV Services and Technologies. New York: Wiley, 2008.
- [42] J. She, F. Hou, P. Ho, and L. Xie, "IPTV over WiMAX: Key success factors, challenges, and solutions," *IEEE Commun. Mag.*, vol. 45, no. 8, pp. 87–93, Aug. 2007.
- [43] Simpson, W.; Greenfield, H.; "IPTV and Internet Video. Expanding the Reach of Television Broadcasting – 2<sup>nd</sup> ed." ISBN 978-0-240-81245-8
- [44] Soteriou, V.; Hangsheng Wang; Peh, L.; , "A Statistical Traffic Model for On-Chip Interconnection Networks," *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2006. MASCOTS 2006. 14th IEEE International Symposium on* , vol., no., pp. 104- 116, 11-14 Sept. 2006 doi: 10.1109/MASCOTS.2006.9
- [45] Bo Ryu; , "Modeling and simulation of broadband satellite networks. II. Traffic modeling," *Communications Magazine, IEEE* , vol.37, no.7, pp.48-56, Jul 1999 doi: 10.1109/35.774880
- [46] Kulkarni, L.A.; San-Qi Li; , "Traffic modeling: matching the power spectrum and distribution," *Global Telecommunications Conference, 1995. GLOBECOM '95., IEEE* , vol.3, no., pp.1701-1706 vol.3, 14-16 Nov 1995 doi: 10.1109/GLOCOM.1995.502700
- [47] Vun, N.; Ansary, M.; , "Implementation of an embedded H.264 live video streaming system," *Consumer Electronics (ISCE), 2010 IEEE 14th International Symposium on* , vol., no., pp.1-4, 7-10 June 2010 doi: 10.1109/ISCE.2010.5523699
- [48] Página web del servidor LIVE555. Disponible en <http://www.live555.com>
- [49] Página web del software ffmpeg. Disponible en <http://www.ffmpeg.org>
- [50] Manual de ffmpeg. Disponible en <http://ffmpeg.org/ffmpeg.html>
- [51] Página web de openRTSP. Disponible en <http://www.live555.com/openRTSP/>
- [52] Página web del reproductor multimedia VLC. Disponible en <http://www.videolan.org/vlc/>
- [53] Página web del analizador de protocolos Wireshark. Disponible en <http://www.wireshark.org/>

- [54] Página web del simulador OPNET Modeler. Disponible en [http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html)
- [55] Página web del simulador ns-3. Disponible en <http://www.nsnam.org/>
- [56] Página web de la herramienta software @RISK. Disponible en <http://www.palisade-lta.com/risk/>
- [57] Página web de A&T. Disponible en [http://www.research.att.com/people/Willinger\\_Walter?fbid=56jFTrtIVYP](http://www.research.att.com/people/Willinger_Walter?fbid=56jFTrtIVYP)
- [58] Leland, W.E.; Taqqu, M.S.; Willinger, W.; Wilson, D.V.; , "On the self-similar nature of Ethernet traffic (extended version) ," *Networking, IEEE/ACM Transactions on* , vol.2, no.1, pp.1-15, Feb 1994 doi: 10.1109/90.282603
- [59] Beran, J.; Sherman, R.; Taqqu, M.S.; Willinger, W.; , "Long-range dependence in variable-bit-rate video traffic," *Communications, IEEE Transactions on* , vol.43, no.234, pp.1566-1579, Feb/Mar/Apr 1995 doi: 10.1109/26.380206
- [60] Erramilli, A.; Narayan, O.; Willinger, W.; , "Experimental queueing analysis with long-range dependent packet traffic," *Networking, IEEE/ACM Transactions on* , vol.4, no.2, pp.209-223, Apr 1996 doi: 10.1109/90.491008
- [61] Wing-Cheong Lau; Erramilli, A.; Wang, J.L.; Willinger, W.; , "Self-similar traffic parameter estimation: a semi-parametric periodogram-based algorithm," *Global Telecommunications Conference, 1995. GLOBECOM '95., IEEE* , vol.3, no., pp.2225-2231 vol.3, 14-16 Nov 1995 doi: 10.1109/GLOCOM.1995.502797
- [62] Willinger, W.; Taqqu, M.S.; Sherman, R.; Wilson, D.V.; , "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," *Networking, IEEE/ACM Transactions on* , vol.5, no.1, pp.71-86, Feb 1997 doi: 10.1109/90.554723
- [63] Erramilli, A.; Roughan, M.; Veitch, D.; Willinger, W.; , "Self-similar traffic and network dynamics," *Proceedings of the IEEE* , vol.90, no.5, pp.800-819, May 2002 doi: 10.1109/JPROC.2002.1015008
- [64] Huebner, F.; Liu, D.; Fernandez, J.M.; , "Queueing performance comparison of traffic models for Internet traffic," *Global Telecommunications Conference, 1998. GLOBECOM 1998. The Bridge to Global Integration. IEEE* , vol.1, no., pp.471-476 vol.1, 1998 doi: 10.1109/GLOCOM.1998.775774
- [65] Liew, C.H.; Kodikara, C.K.; Kondoz, A.M.; , "MPEG-encoded variable bit-rate video traffic modelling," *Communications, IEE Proceedings-* , vol.152, no.5, pp. 749- 756, 7 Oct. 2005 doi: 10.1049/ip-com:20045014
- [66] Hassan, H.; Garcia, J.M.; Bockstal, C.; , "Modeling Internet Traffic: Performance Limits," *Internet Surveillance and Protection, 2006. ICISP '06. International Conference on* , vol., no., pp.7, 26-28 Aug. 2006 doi: 10.1109/ICISP.2006.20
- [67] Coelho Rocha, F.G.; Teles Vieira, F.H.; , "Modeling of MPEG-4 Video Traffic Using a Multifractal Cascade with Autoregressive Multipliers," *Latin America Transactions, IEEE (Revista IEEE America Latina)* , vol.9, no.5, pp.860-867, Sept. 2011 doi: 10.1109/TLA.2011.6031001
- [68] Debin Zou; Xing Zhang; Wenbo Wang; , "Multi-service traffic models of heterogeneous wireless communication networks," *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on* , vol., no., pp.495-498, 25-27 June 2008
- [69] Garcia, M.A.; da Silva, A.P.C.; Meo, M.; , "Using Hidden Markov Chains for Modeling P2P-TV Traffic," *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE* , vol., no., pp.1-6, 6-10 Dec. 2010



- [70] Lobo, A.; García, R.; Pañeda, X.; Melendi, D.; Cabrero, S.; “Modeling Video on Demand Services taking into account Statistical Dependences in User Behavior”. Simulation Modelling Practice and Theory, Volume 31, Pages 96-115. University of Oviedo. 2013.
- [71] Alvarez, A.; Orea, R.; Cabrero, S; Pañeda, X.; García, R.; Melendi, D.; “Limitations of Network Emulation with Single-Machine and Distributed ns-3” SimuTools '10. Proceedings of the 3<sup>rd</sup> International ICST Conference on Simulation tools and Techniques. Article No. 67. 2010.
- [72] Acevedo, Claudia P.; Cerón, Gineth M.; Arciniegas, José L.; Durán, Diego F.; Cerón, Rodrigo; “Arquitectura de ST-CAV, *Proyecto Servicios de T-learning para el Soporte de una Comunidad Académica Virtual*”, Documento de referencia proyecto ST-CAV, Abril 2012.
- [73] Página web. Disponinble en <http://searchnetworking.techtarget.com>
- [74] Página web de AWK. Disponible en <http://www.gnu.org/software/gawk/>
- [75] Astaiza, E., Bermúdez, H. F., & Muñoz, P. A. (2007). *Simulación de sistemas de telecomunicaciones*. Armenia: Arte Imagen.
- [76] Campo, W.; “Análisis de Tráfico para una Red HFC Generado por la Distribución de Contenidos Educativos para la Televisión Digital Interactiva”, Tesis de Maestría en Ingeniería Telemática. Departamento de Telemática, Línea de Investigación en Servicios Avanzados de Telecomunicaciones, FIET, Universidad del Cauca. 2009.
- [77] Proyecto EDiTV, “*Educación Virtual basada en Televisión Digital Interactiva para apoyar procesos educativos a distancia*”. Disponible en <http://www.unicauca.edu.co/EDiTV>
- [78] Rueda, Diego F.; Taimal, Iván R.; “Análisis y evaluación de la red de acceso HFC para la distribución de contenidos educativos de iTV con QoS”, Tesis de pregrado, Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, 2009.