



SIMULACIÓN Y CARACTERIZACIÓN DE TRÁFICO DE NUEVOS SERVICIOS  
EN REDES IP

HERNANDO BLANCO CORREA

JOSÉ LUIS GARCÍA GRAZZIANI

UNIVERSIDAD PONTIFICIA BOLIVARIANA  
ESCUELA DE INGENIERÍAS Y ADMINISTRACIÓN

INGENIERÍA ELECTRÓNICA

BUCARAMANGA

2011

SIMULACIÓN Y CARACTERIZACIÓN DE TRÁFICO DE NUEVOS SERVICIOS  
EN REDES IP

HERNANDO BLANCO CORREA

JOSÉ LUIS GARCÍA GRAZZIANI

Trabajo de grado para optar al título de ingeniero Electrónico

Director

JHON JAIRO PADILLA

Doctor en Ingeniería Telemática

UNIVERSIDAD PONTIFICIA BOLIVARIANA  
ESCUELA DE INGENIERÍAS Y ADMINISTRACIÓN  
INGENIERÍA ELECTRÓNICA  
BUCARAMANGA

2011

NOTA DE ACEPTACION

---

---

---

---

---

Firma  
Nombre  
Presidente del jurado

---

Firma  
Nombre  
Presidente del jurado

Bucaramanga, Julio 2011

## Tabla de contenido

OBJETIVOS.....	12
OBJETIVO GENERAL.....	12
OBJETIVOS ESPECÍFICOS .....	12
1  MARCO TEÓRICO.....	16
1.1  Protocolos de internet .....	16
1.1.1  IP (Internet protocol).....	16
1.1.2  TCP (transmission control protocol) .....	17
1.1.3  UDP (user datagram protocol).....	18
1.1.4  FTP: protocolo de transferencia de ficheros.....	18
1.2  Modelado de tráfico .....	19
1.2.1  Criterios de selección de un modelo .....	20
1.3  Distribuciones de probabilidad.....	21
1.3.1  Distribución Normal .....	21
1.3.2  Distribuciones Gamma y exponencial .....	22
1.3.3  Distribución Ji cuadrada .....	24
1.3.4  Distribución de Weibull .....	25
1.3.5  Distribución logarítmica normal .....	26
1.4  Pruebas de bondad de ajuste .....	27
1.5  CONCEPTOS FUNDAMENTALES SOBRE TRÁFICO AUTOSIMILAR ..	28
1.5.1  Autosimilitud determinística.....	28
1.5.2  Autosimilitud estocástica .....	30
1.6  Estimación del parámetro de Hurst .....	31
1.7  OPNET MODELER .....	32
1.7.1  MODELADO DE ALTA FIDELIDAD EN OPNET .....	33
1.7.2  Editor de proyectos .....	33
1.7.3  Editor de nodos.....	33
1.7.4  Editor de procesos .....	33
1.7.5  FUNCIONAMIENTO .....	34

1.8	EASYFIT .....	39
1.8.1	Pruebas de bondad de ajuste .....	40
2	ESTADO DEL ARTE .....	40
2.1	Modelado de tráfico de clientes .....	40
2.1.1	Modelo orientado a página (POM) .....	40
2.1.2	Modelo orientado a tiempos (TOM) .....	41
2.1.3	El modelo orientado a conexión (COM) .....	44
2.2	Tráfico de aplicaciones varias (peer to peer, e-mail y otras) .....	44
2.3	Modelado de Tráfico de Juegos en línea .....	45
2.4	Modelado de Tráfico Redes Sociales .....	46
2.5	Modelado de tráfico FTP.....	46
2.6	El modelado de tráfico de video.....	47
2.7	Modelado de Trafico VoIP .....	49
3	METODOLOGÍA.....	50
3.1	Caracterización de los servicios actuales de internet .....	50
3.2	Comportamiento de los usuarios .....	51
3.3	Manejo del software OPNET .....	51
3.4	Simulación .....	51
3.5	Análisis estadístico .....	52
4	SIMULACION Y RESULTADOS .....	52
4.1	SIMULACIÓN PRUEBA 1.....	52
4.1.1	Objetivo .....	52
4.1.2	Topología prueba 1 .....	52
4.1.3	Configuración de aplicaciones .....	53
4.1.4	Resultado prueba 1 .....	54
4.2	SIMULACION PRUEBA 2.....	55
4.2.1	Objetivo .....	55
4.2.2	Topología prueba 2 .....	55
4.2.3	Configuración de aplicaciones .....	56
4.2.4	Resultados prueba 2 .....	56

4.3	SIMULACION PRUEBA 3.....	57
4.3.1	Objetivo .....	57
4.3.2	Topología prueba 3 .....	57
4.3.3	Configuración de aplicaciones .....	58
4.3.4	Resultados prueba 3 .....	72
4.4	SIMULACIÓN FINAL .....	80
4.4.1	Objetivo .....	80
4.4.2	Topología final.....	80
4.4.3	Configuración de aplicaciones .....	81
4.4.4	Resultados .....	83
5	ANÁLISIS ESTADÍSTICO DE LOS RESULTADOS .....	86
5.1	Estimación del parámetro de hurst .....	86
5.1.1	Metodo RS .....	86
5.1.2	Método de varianza agregada .....	93
5.2	APLICACIÓN DE DISTRIBUCIONES A LOS RESULTADOS.....	98
5.2.1	Distribución Gamma.....	98
5.2.2	Distribución lognormal.....	100
5.2.3	Resultados de las pruebas de bondad de ajuste .....	102
	CONCLUSIONES Y TRABAJOS FUTUROS.....	106
	BIBLIOGRAFÍA.....	108

## LISTA DE FIGURAS

	Pág.
Figura 1 Curva normal .....	21
Figura 2 Copo de nieve de Von Koch .....	29
Figura 3 Variación del copo de nieve de Von Koch .....	30
Figura 4 Número de paquetes que transitan por un segmento de una red Ethernet .....	31
Figura 5 Opnet modeler .....	34
Figura 6 Jerarquía de diseño en OPNET .....	35
Figura 7 Funcionamiento del modelo de Nodos.....	35
Figura 8 Cable par trenzado .....	37
Figura 9 Switch .....	38
Figura 10 Server .....	38
Figura 11 Application definition .....	38
Figura 12 Profile definition .....	38
Figura 13 Easyfit .....	39
Figura 14 FTPdata .....	47
Figura 15 Topología prueba 1 .....	52
Figura 16 Aplicaciones y perfiles en servidor y usuario .....	54
Figura 17 Aplicaciones de opnet.....	54
Figura 18 prueba 1 paquetes/segundo enviados por el usuario .....	55
Figura 19 Topología para la prueba 2 .....	56
Figura 20 Tráfico entre SW1 y SW2 .....	57
Figura 21 Profile definition attributes.....	58
Figura 22 Configuración voip .....	59
Figura 23 Silence length .....	59
Figura 24 Configuración de datos .....	61
Figura 25 Configuración de video .....	62
Figura 26 Modelo matemático frame size .....	62
Figura 27 Configuración peer to peer .....	64
Figura 28 Configuración email .....	66
Figura 29 Configuración ftp.....	67
Figura 30 Configuración http.....	68
Figura 31 Profile attributes .....	69
Figura 32 Profile voip attributes .....	72
Figura 33 Usuario voip .....	73

Figura 34 Usuario database.....	74
Figura 35 Usuario video .....	75
Figura 36 Usuario peer to peer .....	76
Figura 37 Usuario email.....	77
Figura 38 Usuario FTP.....	78
Figura 39 Usuario HTTP .....	79
Figura 40 Total prueba 3.....	80
Figura 41 Esquema de la simulación.....	81
Figura 42 Resultado Final para un usuario en paquetes/segundos .....	84
Figura 43 Resultado Final en paquetes/segundos.....	84
Figura 44 Resultado Final en bits/segundos.....	85
Figura 45 Resultado Final queue delay en segundos .....	85
Figura 46 Exportar datos a excel .....	86
Figura 47 Parámetro de hurst método RS para paquetes/segundos .....	90
Figura 48 Parámetro de hurst método RS para bits/segundos .....	91
Figura 49 Parámetro de hurst método RS para queue dealy.....	92
Figura 50 Parámetro de hurst método varianza agregada para paquetes/segundos .....	96
Figura 51 Parámetro de hurst método varianza agregada para bits/segundos .....	97
Figura 52 Parámetro de hurst método varianza agregada para queue delay .....	98
Figura 53 Histograma y distribución gamma con easyfit para paquetes/segundos .....	100
Figura 54 Histograma y distribución lognormal con easyfit para bits/segundos...	102

## LISTA DE ECUACIONES

	Pág.
Ecuación 1 Función de densidad de la variable aleatoria normal X.....	22
Ecuación 2 Función gamma.....	22
Ecuación 3 Función de densidad dist. Gamma.....	23
Ecuación 4 Media y varianza de la distribución gamma .....	23
Ecuación 5 Función de densidad dist. Exponencial .....	23
Ecuación 6 Media y la varianza de la distribución exponencial .....	24
Ecuación 7 Función de densidad dist. ji cuadrada .....	24
Ecuación 8 La media y la varianza de la distribución ji cuadrada .....	25
Ecuación 9 Función de densidad dist. Weibull.....	25
Ecuación 10 La media y la varianza de la distribución de Weibull.....	26
Ecuación 11 Función de densidad dist. Logarítmica Normal .....	26
Ecuación 12 Media y la varianza de la distribución logarítmica normal .....	27
Ecuación 13 Hipótesis nula.....	27
Ecuación 14 prueba chi cuadrado.....	28
Ecuación 15 fractional Gaussian noise .....	50
Ecuación 16 Ajuste de Hurst método RS.....	87
Ecuación 17 fractional gaussian .....	87
Ecuación 18 Series agregadas .....	93
Ecuación 19 Varianza de la muestra .....	93

## LISTA DE TABLAS

	Pág.
Tabla 1 Módulos del Node Editor.....	36
Tabla 2 Opciones del Process Model .....	37
Tabla 3 Resumen de los parametros del modelo POM propuesto.....	41
Tabla 4 Resumen de los parametros del modelo TOM propuesto.....	43
Tabla 5 Resumen de los parametros del modelo COM propuesto .....	44
Tabla 6 Distribuciones de diferentes servicios y sus parámetros.....	45
Tabla 7 Modelo de una fuente de trafico FTP .....	47
Tabla 8 Tabla de estadísticas de los videos más populares.....	49
Tabla 9 Parámetros de una llamada VOIP .....	50
Tabla 10 Modelo matemático para transaction interarrival time.....	60
Tabla 11 Modelo matemático para transaction size .....	60
Tabla 12 Modelo matemático para frame size .....	61
Tabla 13 Modelo matemático para transaction interarrival time.....	63
Tabla 14 Modelo matemático para transaction size peer to peer.....	63
Tabla 15 Modelo matemático para sent interarrival time .....	64
Tabla 16 Modelo matemático para receive interarrival time.....	65
Tabla 17 Modelo matemático para E-MAIL size .....	65
Tabla 18 Modelo matemático para Download file size.....	66
Tabla 19 Modelo matemático para Inter-arrival time.....	66
Tabla 20 Numero de objetos en http.....	67
Tabla 21 Duración del usuario .....	70
Tabla 22 Modelo matemático para inter repetition time .....	70
Tabla 23 Modelo matemático para duración llamada voip.....	71
Tabla 24 Modelo matemático para llegada de nuevas llamadas en voip.....	71
Tabla 25 Resumen de las distribuciones para application definition.....	82
Tabla 26 Resumen de las distribuciones para profile definition .....	83
Tabla 27 media y varianza para paquetes/segundos.....	99
Tabla 28 Parámetros distribución gamma obtenidos manualmente .....	99
Tabla 29 media y varianza obtenidos en easyfit .....	99
Tabla 30 Parámetros distribución gamma obtenidos con easyfit.....	99
Tabla 31 media y varianza para bits/segundos.....	101
Tabla 32 Parámetros distribución lognormal obtenidos manualmente.....	101

Tabla 33 media y varianza obtenidos en easyfit .....	101
Tabla 34 Parámetros distribución lognormal obtenidos con easyfit .....	101
Tabla 35 Pruebas para Paquetes/segundos alfa= 0.05 .....	103
Tabla 36 Parámetros de las distribuciones para paquetes/segundos .....	103
Tabla 37 Pruebas para bits/segundos alfa= 0.05.....	104
Tabla 38 Parámetros de las distribuciones para bits/segundos .....	104
Tabla 39 Pruebas para queue delay alfa= 0.05 .....	104
Tabla 40 Parámetros de las distribuciones para queue delay .....	105

## **OBJETIVOS**

### **OBJETIVO GENERAL**

- Caracterizar el tráfico de una red IP con aplicaciones actuales tales como juegos en línea, redes sociales, VoIP, IPTV, FTP, e-mail, Web, etc.

### **OBJETIVOS ESPECÍFICOS**

- Seleccionar los modelos matemáticos o de simulación adecuados para el tráfico de los diferentes servicios actuales de internet (redes sociales, VoIP, p2p, FTP, e-mail, Web, etc.).
- Simular cada tipo de tráfico con base en los modelos seleccionados y simular el agregado de tráfico que contiene todos los servicios.
- Hacer un análisis estadístico de los resultados de la simulación del tráfico agregado con el fin de obtener posibles distribuciones que lo representen.

## RESUMEN GENERAL DE TRABAJO DE GRADO

**TITULO:** SIMULACIÓN Y CARACTERIZACION DE TRÁFICO DE NUEVOS SERVICIOS EN REDES IP

**AUTOR(ES):** HERNANDO BLANCO CORREA  
JOSE LUIS GARCIA GRAZZIANI

**FACULTAD:** Facultad de ingeniería electrónica

**DIRECTOR(A):** JHON JAIRO PADILLA

### RESUMEN

Debido al gran auge de Internet, en la actualidad se alcanzan altos niveles de congestión; esto se debe a los nuevos recursos y aplicaciones que han ido surgiendo en la sociedad actual y al creciente número de usuarios, que cada día aumenta sin parar. Debido a lo mencionado anteriormente, el comportamiento del tráfico actual no es el mismo que el de hace unos años, el cual se limitaba solo a la transmisión de texto e imágenes; ahora han surgido nuevos servicios como las redes sociales, juegos en línea, los videos (youtube), la comunicación a través de VOIP y peer to peer. Dado que, los modelos tradicionalmente usados han demostrado ser insuficientes para caracterizar los nuevos tipos de tráfico que han surgido en la actualidad, en los últimos años la investigación se ha centrado en el modelado de muestras obtenidas en redes reales. El presente proyecto partió de modelos matemáticos existentes propuestos en diferentes investigaciones para nuevas aplicaciones, estos modelos matemáticos se utilizaron para generar el tráfico agregado compuesto por aplicaciones tradicionales (FTP, email, HTTP) y por las nuevas aplicaciones (VOIP, P2P, facebook, youtube). Este tráfico se simuló y a partir del tráfico agregado obtenido se realizaron las pruebas estadísticas pertinentes. Para el desarrollo de la simulación se hizo uso del software OPNET, mediante el cual se llevó a cabo la recolección de los datos (exportación de estos a Excel), los cuales fueron llevados al software EASYFIT para la determinación de los parámetros y realización de 3 pruebas de bondad de ajuste, que permitieron realizar el estudio y validación de su comportamiento dentro de algún modelo. Finalmente se observó el comportamiento del tráfico agregado, concluyendo que este tipo de tráfico puede ser considerado como tráfico autosimilar, pero también puede ser modelado con otras distribuciones que se identifican en este documento.

### **PALABRAS CLAVES:**

OPNET, autosimilar, easyfit, modelado, tráfico, redes.

## RESUMEN GENERAL DE TRABAJO DE GRADO

**TITLE:** SIMULATION AND CHARACTERIZATION OF NEW SERVICES TRAFFIC ON IP NETWORKS

**AUTHORS:** HERNANDO BLANCO CORREA  
JOSE LUIS GARCIA GRAZZIANI

**FACULTY:** Faculty of Engineering electronic

**DIRECTOR:** JHON JAIRO PADILLA

### ABSTRACT

Due to the popularity of the Internet today, currently there are high levels of network congestion, this is due to new resources and applications that have emerged in today's society and the growing number of users that increases every day. In consequence, the behavior of current traffic is different respect to the users traffic few years ago. Older traffic was composed only by the transmission of text and images; today there are new services and applications such as social networks, online games, video (youtube), VoIP and Internet television (IPTV). Since, the traditionally used models have demonstrated to be inadequate to characterize new sources of traffic that have recently emerged, in recent years the research has focused on modeling samples of new services traffic obtained in real networks. This project used the existing mathematical models proposed in various studies for new applications, these mathematical models were used to generate aggregate traffic composed by new (VOIP, P2P, facebook, youtube) and traditional applications (FTP, email, HTTP). This traffic was simulated using the OPNET simulator. Then, obtained results were statistically tested. Data collection obtained with the experiments was exported to Excel, and then, they were introduced to EasyFit software to determine the statistical parameters. Also, we performed 3 goodness fit tests, which allowed the study and validation of the traffic behavior with several statistical models. Finally it was observed the behavior of the converging traffic, concluding that this type of traffic can be considered self-similar traffic.

### PALABRAS CLAVES:

OPNET, self-similar, Easyfit, Network traffic modeling.

## INTRODUCCIÓN

Las redes IP son una parte muy importante en el mundo ya que gracias a sus distintos tipos de información; Estas redes son usadas constantemente por millones de usuarios. Lo que genera un tráfico que deben soportar los equipos actuales.

Es importante la aplicación de modelos matemáticos para explicar la relación que existe entre la capacidad de una red de comunicaciones, la demanda de servicio que los usuarios le imponen y el nivel de desempeño que la red puede alcanzar. Como dicha demanda es de naturaleza estadística, se suele representar mediante algún proceso estocástico adecuado, existen diferentes tráficos de internet a los cuales ya se le han realizado estos análisis definiendo un modelo matemático adecuado para su aproximación. Por otra parte, es importante obtener modelos matemáticos de los agregados de diferentes tipos de tráfico. A finales de los 90's y principios de la década del 2000 se obtuvieron algunos modelos de este tipo de tráfico obteniendo modelos de tipo autosimilar. Sin embargo, en años recientes han aparecido nuevas aplicaciones y servicios tales como Facebook, youtube, skype, etc. Este tipo de aplicaciones no se han incluido dentro de estos agregados de tráfico para obtener nuevos modelos, por lo que este proyecto busca avanzar en este sentido.

# 1 MARCO TEÓRICO

## 1.1 Protocolos de internet (1)

En cualquier arquitectura por capas en que exista una comunicación vertical dentro de cada máquina, los datos que se generan en el nivel superior (aplicación) atraviesan el resto de niveles para “salir” de la máquina por el nivel físico. Cada uno de estos protocolos funciona con unas estructuras de información fundamentales que genéricamente se conocen como PDU (protocol data units). Sin embargo, en cada nivel se utilizan nombres diferentes para las PDUs aunque, de hecho, tiene funciones equivalentes. En el conjunto de protocolos de Internet tenemos las siguientes PDU:

- Las PDU Ethernet o PPP se denominan tramas.
- Las PDU del nivel de interconexión (IP o ARP) se suelen denominar paquetes, aunque las PDU ICMP se suelen denominar mensajes, seguramente porque viajan en paquetes IP.
- En el nivel de transporte, se habla de segmentos en TCP, y de datagramas en UDP.
- En niveles superiores que utilizan UDP, por norma general se utiliza la palabra PDU (SNMP-PDU, por ejemplo). En el caso del TCP, el servicio que proporciona a las aplicaciones es el flujo de bytes sin estructura (byte stream).

El modelo Internet gira en torno a los protocolos TCP/IP. IP es un protocolo que proporciona mecanismos de interconexión entre redes de diferentes tipos y TCP proporciona mecanismos de control de flujo y errores entre los extremos de la comunicación.

### 1.1.1 IP (Internet protocol) (1)

El protocolo IP utiliza una identificación para cada computador en una Internet. Esta identificación se conoce como “Dirección IP”. Las direcciones IP son únicas para cada máquina. Para ser precisos, cada dirección es única para cada una de las interfaces de red IP de cada máquina. Si una máquina dispone de más de una interfaz de red, necesitará una dirección IP para cada una. Las direcciones IP

tienen una longitud de 32 bits .La numeración en IP sigue una filosofía jerárquica. Cada dirección está formada por 2 partes. Una corresponde a la red donde está la estación y la otra, a la propia estación. Para conseguir que no haya ninguna dirección igual, Internet dispone de una organización denominada Internet Network Information Center o InterNIC que se dedica a esta tarea. En la actualidad, esta entidad delega la responsabilidad de la asignación de direcciones a entidades regionales. Las direcciones se asignan por grupos o redes, no individualmente (1).

### **1.1.2 TCP (transmission control protocol)**

Evidentemente, cuando utilizamos una red, no siempre podemos conformarnos con conseguir que la información llegue si la red puede. Aquí interviene el TCP, que es responsable de conseguir que la información llegue en las condiciones de fiabilidad deseadas.

El protocolo UDP no garantiza la entrega de la información que le proporciona una aplicación. Tampoco reordena la información en caso de que llegue en un orden diferente de aquél en que se ha transmitido. Existen aplicaciones que no pueden tolerar dichas limitaciones. Para superarlas, el nivel de transporte proporciona un protocolo llamado TCP. El TCP proporciona fiabilidad a la aplicación; es decir, garantiza la entrega de toda la información en el mismo orden en que ha sido transmitida por la aplicación de origen. Para conseguir esta fiabilidad, el TCP proporciona un servicio orientado a la conexión con un control de flujo y errores.

La fiabilidad de la transmisión que proporciona TCP se consigue gracias a las siguientes estrategias:

- Está orientado a la conexión: tiene una fase de establecimiento de la conexión, una de transmisión de datos y una de desconexión.
- Utiliza el concepto buffered transfer: cuando se transfiere información, TCP divide los flujos de datos (byte stream) que le pasa la aplicación en segmentos del tamaño que le convenga. TCP decide el tamaño de los segmentos tanto si la aplicación genera un byte de información, como si genera flujos de grandes dimensiones. En el primer caso, TCP puede esperar a que la memoria intermedia se llene más antes de transferir la información, o bien la puede transferir de inmediato (mecanismo push). En caso de que los flujos sean muy grandes, TCP puede dividir la información en tamaños más pequeños antes de transferirlos.

- Utiliza una conexión full duplex: la transferencia de información es en ambos sentidos. La aplicación ve dos flujos independientes. En caso de que la aplicación cierre uno de los flujos, la conexión pasa a ser halfduplex. Ello significa que uno de los extremos (el que no ha cerrado la conexión) puede continuar enviando información por el canal, mientras que el otro extremo (el que ha cerrado la conexión) se limita a reconocer la información tomado en (1).

### **1.1.3 UDP (user datagram protocol)**

El UDP es un protocolo no orientado a la conexión, de manera que no proporciona ningún tipo de control de errores ni de flujo, aunque utiliza mecanismos de detección de errores. En caso de detectar un error, el UDP no entrega el datagrama a la aplicación, sino que lo descarta. Conviene recordar que, por debajo, el UDP está utilizando el IP, que también es un protocolo no orientado a la conexión. Las características más importantes del UDP son las siguientes:

- No garantiza la fiabilidad; es decir, no ofrece la seguridad de que cada datagrama UDP transmitido llegue a su destino; es un protocolo best-effort: el UDP hace todo lo posible para transferir los datagramas de su aplicación, pero no garantiza su entrega.
- No preserva la secuencia de la información que le proporciona la aplicación. Como está en modo datagrama y utiliza un protocolo por debajo como el IP, que también está en modo datagrama, la aplicación puede recibir la información desordenada. La aplicación debe estar preparada para que haya datagramas que se pierdan, lleguen con retardo o se hayan desordenado (1).

### **1.1.4 FTP: protocolo de transferencia de ficheros**

Una de las primeras aplicaciones básicas desarrolladas en el entorno de lo que después sería la red Internet fue la transferencia de ficheros entre diferentes sistemas. Al principio de los años setenta ya se elaboraron las primeras especificaciones del protocolo más utilizado para esta finalidad, el FTP. Después de algunas modificaciones y mejoras, la especificación oficial del protocolo se publicó en 1985 en el documento RFC 959. El FTP se basa en el modelo cliente/servidor y permite la transferencia de ficheros tanto del servidor al cliente, como del cliente al servidor. Asimismo, permite que un cliente efectúe transferencias directas de un servidor a otro, con lo que se ahorra la necesidad de

copiar los ficheros del primer servidor al cliente y pasarlos después del cliente al segundo servidor. El protocolo proporciona también operaciones para que el cliente pueda manipular el sistema de ficheros del servidor: borrar ficheros o cambiarles el nombre, crear y borrar directorios, listar sus contenidos, etc. Uno de los objetivos principales de este protocolo consiste en permitir la interoperabilidad entre sistemas muy distintos, escondiendo los detalles de la estructura interna de los sistemas de ficheros locales y de la organización de los contenidos de los ficheros (1).

## **1.2 Modelado de tráfico (2)**

Se entiende por modelo de tráfico una abstracción matemática más o menos compleja que trata de imitar alguna o varias características estadísticas de un tipo de tráfico real o de un flujo concreto en particular. Dado que los modelos poissonianos, tradicionalmente usados con el tráfico telefónico, se han demostrado insuficientes para caracterizar las nuevas fuentes de tráfico que han surgido recientemente, en los últimos años gran parte de la investigación se ha centrado en el modelado de muestras obtenidas en redes reales. En este sentido al investigador le caben dos opciones: muestrear el tráfico en una red a la que tenga acceso y autorización para "espiar" (habitualmente esto sólo es posible en su propia red de área local o LAN) o recurrir a las múltiples bases de datos con muestras de tráfico que ofrece internet. Los ejemplos de este último tipo de muestras son numerosos, tanto para el tráfico Ethernet, tráfico en redes de área metropolitana, tráfico TCP/IP correspondiente a servicios HTTP (la Web) o tráfico de vídeo VBR correspondientes a estándares MPEG y M-JPEG. Por su parte, el muestreo puede ser directo, analizando el tráfico que directamente fluye por el enlace, o indirecto, estudiando de una manera off-line, sobre un servidor, el volumen de información (ficheros en servidores FTP, tamaño de los correos electrónicos, páginas Web, películas MPEG) a la que los clientes tendrán acceso (2).

## **1.2.1 Criterios de selección de un modelo (2)**

### **1.2.1.1 La tratabilidad analítica del modelo**

Es decir, la capacidad intrínseca del modelo de arrojar resultados analíticos, sin necesidad de simulación.

### **1.2.1.2 La facilidad de implementación**

Ya sea vía software para proceder a simulaciones o vía hardware con el objetivo de disponer de un generador de tráfico sintético en tiempo real. Esta característica exige simplicidad no sólo a la propia estructura del modelo sino también a la algoritmia y a los costes computacionales que exige su implementación.

### **1.2.1.3 La "parsimonia" del modelo**

Con este concepto, que recurre al significado original de la palabra parsimonia, se describe la capacidad del modelo para ser descrito mediante un conjunto reducido de parámetros. De igual importancia resulta la posible significación física de los parámetros. Se debe ser consciente de que al usuario final hay que ofrecerle una representación de la realidad comprensible, flexible pero con una variabilidad acotada y controlable mediante unos parámetros que posean una conexión evidente con la realidad (por ejemplo: velocidad media, velocidad de pico, tamaño de ráfaga).

### **1.2.1.4 Capacidad de modelar otros tipos de tráfico y, en especial, tráfico agregado**

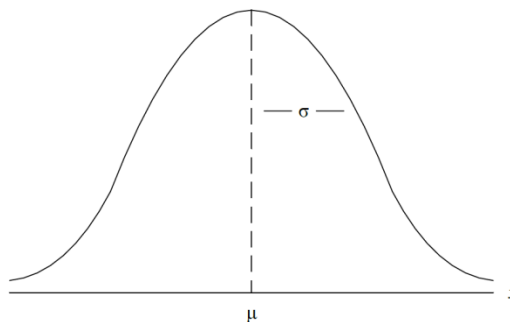
En numerosas ocasiones se puede exigir de un modelo, en especial de los generalistas, la ductilidad suficiente para que, mediante ciertos cambios de sus parámetros, este sea capaz de imitar otros tipos de tráfico o incluso el propio tráfico agregado que resulta de multiplexar sobre un canal varias fuentes individuales.

## 1.3 Distribuciones de probabilidad

### 1.3.1 Distribución Normal (3)

La distribución continua de probabilidad más importante en todo el campo de la estadística es la distribución normal. Su gráfica se denomina curva normal (Figura 1 Curva normal), es la curva con la forma de campana, la cual describe aproximadamente muchos fenómenos que ocurren en la naturaleza, la industria y la investigación. Las mediciones físicas en áreas como los experimentos meteorológicos, estudios de lluvia y mediciones de partes fabricadas a menudo se explican mejor con una distribución normal. Los errores en las mediciones científicas se aproximan extremadamente bien mediante una distribución normal.

**Figura 1 Curva normal**



Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye,K Myers, *Probabilidad y estadística para ingenieros* (3)

La distribución normal a menudo se denomina distribución gaussiana, en honor de Karl Friedrich Gauss. Una variable aleatoria continua  $X$  que tiene la distribución en forma de campana se llama variable aleatoria normal. La ecuación matemática para la distribución de probabilidad de la variable normal depende de los dos parámetros  $\mu$  y  $\sigma$ , su media y desviación estándar. De aquí se denotan los valores de la densidad de  $X$  con  $n(x;\mu,\sigma)$  (3).

La función de densidad de la variable aleatoria normal  $X$ , con media  $\mu$  y varianza  $\sigma^2$ , es:

### **Ecuación 1 Función de densidad de la variable aleatoria normal X**

$$n(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad -\infty < x < +\infty$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros (3)*

Donde  $\pi = 3.14159\dots$  y  $e = 2.71828\dots$

Una vez que se especifican  $\mu$  y  $\sigma$ , la curva normal queda determinada por completo.

### **1.3.2 Distribuciones Gamma y exponencial (3)**

Las distribuciones gamma y exponencial juegan un papel importante en teoría de colas y problemas de confiabilidad. Los tiempos entre llegadas en instalaciones de servicio, y tiempo de falla de componentes y sistemas eléctricos, a menudo quedan bien modeladas con la distribución exponencial. La distribución exponencial es un caso especial de la distribución gamma. La distribución gamma deriva su nombre de la bien conocida función gamma, que se estudia en muchas áreas de las matemáticas.

La función gamma se define como:

#### **Ecuación 2 Función gamma**

$$\Gamma(a) = \int_0^{\infty} x^{a-1} e^{-x} dx \quad a > 0$$

$$\Gamma(1) = 1 \quad \Gamma(n) = (n-1)!$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros (3)*

La variable aleatoria continua X tiene una distribución gamma, con parámetros  $\alpha$  y  $\beta$ , si su función de densidad está dada por

### Ecuación 3 Función de densidad dist. Gamma

$$f(x) = \begin{cases} \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}} & x > 0 \\ 0 & \text{otro caso} \end{cases}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros* (3)

Donde  $\alpha > 0$  y  $\beta > 0$ .

La media y la varianza de la distribución gamma son:

### Ecuación 4 Media y varianza de la distribución gamma

$$\mu = \alpha\beta \quad \sigma^2 = \alpha\beta^2$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros* (3)

Cuando la distribución gamma tiene  $\alpha = 1$  se llama distribución exponencial. La variable aleatoria continua  $X$  tiene una distribución exponencial, con parámetro  $\beta$ , si su función de densidad está dada por:

### Ecuación 5 Función de densidad dist. Exponencial

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-\frac{x}{\beta}} & x > 0 \\ 0 & \text{otro caso} \end{cases}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros* (3)

Donde  $\beta > 0$ .

La media y la varianza de la distribución exponencial son:

**Ecuación 6 Media y la varianza de la distribución exponencial**

$$\mu = \beta \quad \sigma^2 = \beta^2$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros* (3)

**1.3.3 Distribución Ji cuadrada (3)**

Otro caso especial importante de la distribución gamma se obtiene al hacer  $\alpha = v/2$  y  $\beta = 2$ , donde  $v$  es un entero positivo. Este resultado se llama distribución ji cuadrada. Esta distribución tiene un solo parámetro,  $v$ , llamado grados de libertad. Esta distribución juega un papel muy importante en la inferencia estadística. Tiene una aplicación considerable en la metodología y en la teoría. Además, es un componente importante de la prueba de hipótesis y la estimación estadística.

La variable aleatoria continua  $X$  tiene una distribución ji cuadrada, con  $v$  grados de libertad, si su función de densidad está dada por:

**Ecuación 7 Función de densidad dist. ji cuadrada**

$$f(x) = \begin{cases} \frac{1}{2^{\frac{v}{2}} \Gamma\left(\frac{v}{2}\right)} x^{\frac{v}{2}-1} e^{-\frac{x}{2}} & x > 0 \\ 0 & \text{otro caso} \end{cases}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros* (3)

Donde  $v$  es un entero positivo.

La media y la varianza de la distribución ji cuadrada son

### **Ecuación 8 La media y la varianza de la distribución ji cuadrada**

$$\mu = v \quad \sigma^2 = 2v$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye,K Myers, *Probabilidad y estadística para ingenieros (3)*

#### **1.3.4 Distribución de Weibull (3)**

Como la distribución gamma y exponencial, la distribución de Weibull también se aplica a problemas de confiabilidad y de prueba de vida como los tiempos de falla o duración de vida de un componente, medido en algún tiempo específico hasta que falla. El tiempo de falla se representa mediante la variable aleatoria continua T, con función de densidad de probabilidad f(t), que es una distribución de Weibull.

La variable aleatoria continua X tiene una distribución de Weibull, con parámetros  $\alpha$  y  $\beta$ , si su función de densidad está dada por:

#### **Ecuación 9 Función de densidad dist. Weibull**

$$f(x) = \begin{cases} \alpha\beta x^{\beta-1} e^{-\alpha x^\beta} & x > 0 \\ 0 & \text{otro caso} \end{cases}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye,K Myers, *Probabilidad y estadística para ingenieros (3)*

Donde  $\alpha > 0$  y  $\beta > 0$ .

La media y la varianza de la distribución de Weibull son:

### Ecuación 10 La media y la varianza de la distribución de Weibull

$$\mu = \alpha^{-\frac{1}{\beta}} \Gamma\left(1 + \frac{1}{\beta}\right) \quad \sigma^2 = \alpha^{-\frac{2}{\beta}} \left\{ \Gamma\left(1 + \frac{2}{\beta}\right) - \left[ \Gamma\left(1 + \frac{1}{\beta}\right) \right]^2 \right\}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye,K Myers, *Probabilidad y estadística para ingenieros (3)*

#### 1.3.5 Distribución logarítmica normal (3)

La distribución logarítmica normal se utiliza en una amplia variedad de aplicaciones. La distribución se aplica en casos donde una transformación de logaritmo natural tiene como resultado una distribución normal.

La variable aleatoria continua X tiene una distribución logarítmica normal si la variable aleatoria Y= ln(X) tiene una distribución normal con media  $\mu$  y desviación estándar  $\sigma$ . La función de densidad de X que resulta es

### Ecuación 11 Función de densidad dist. Logarítmica Normal

$$f(x) = \begin{cases} \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} & x \geq 0 \\ 0 & \text{otro caso} \end{cases}$$

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye,K Myers, *Probabilidad y estadística para ingenieros (3)*

La media y la varianza de la distribución logarítmica normal

### Ecuación 12 Media y la varianza de la distribución logarítmica normal

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

Fuente: R.E. Walpole, R.H. Myers, and S.L.&Ye, K Myers, *Probabilidad y estadística para ingenieros (3)*

#### 1.4 Pruebas de bondad de ajuste (4)

Una prueba de bondad de ajuste se emplea para decidir cuándo un conjunto de datos se apega a una distribución de probabilidad dada. Considérese una muestra aleatoria de tamaño  $n$  de la distribución de una variable aleatoria  $X$  dividida en  $k$  clases exhaustivas y mutuamente excluyentes, y sea  $N_i$ ,  $i=1,2,\dots, k$ , el número de observaciones en la  $i$ -ésima clase. Considérese la verificación de la hipótesis nula

#### Ecuación 13 Hipótesis nula

$$H_0: F(x) = F_0(x),$$

Fuente: Douglas C.MONTGOMERY, *Probabilidad y estadística aplicadas a la ingeniería (4)*

En donde el modelo de probabilidad propuesto  $F_0(x)$  se encuentra especificado, de manera completa, con respecto a todos los parámetros. De esta forma la hipótesis nula es sencilla. Dado que se especifica  $F_0(x)$  de manera completa, se puede obtener la probabilidad de  $p_i$  de obtener una observación en la  $i$ -ésima clase bajo  $H_0$ , en donde necesariamente  $\sum_{i=1}^k p_i = 1$ .

Sea  $n_i$  la realización de  $N_i$  para  $i= 1, 2,\dots,k$  de manera tal que  $\sum_{i=1}^k n_i = n$ . La probabilidad de tener, de manera exacta,  $n_i$  observaciones en la  $i$ -ésima clase es  $p_i^{n_i}$  para  $i= 1, 1 \dots,k$ . Dado que existe  $k$  categorías mutuamente excluyentes con probabilidades  $p_1, p_2, \dots, p_k$ , entonces bajo la hipótesis nula la probabilidad de la muestra agrupada es igual a la función de probabilidad de una distribución multinomial.

La estadística tiene aproximadamente una distribución chi-cuadrada con un grado de libertad conforme  $n$  va tomando valores cada vez más grandes.

Si se sigue este tipo de razonamiento, puede demostrarse que para  $k \geq 2$  categorías distintas, la estadística

#### **Ecuación 14 prueba chi cuadrado**

$$\sum_{i=1}^k \frac{(N_i - np_i)^2}{np_i}$$

Fuente: Douglas C.MONTGOMERY, *Probabilidad y estadística aplicadas a la ingeniería*  
(4)

Tiene una distribución, en forma aproximada, chi-cuadrada con  $k-1$  grados de libertad, si  $n$  tiene un valor suficientemente grande. Nótese que  $n_i$  es la frecuencia observada en la  $i$ -ésima clase, y  $np_i$  es la referencia correspondiente que se esperaba bajo la hipótesis nula. De acuerdo con lo anterior, la estadística es la suma sobre todas las  $k$  clases de los cocientes de los cuadrados de las diferencias entre las frecuencias observada y esperada, y la frecuencia esperada. La estadística dada por (1) recibe el nombre de prueba de bondad de ajuste chi-cuadrada de Pearson.

### **1.5 CONCEPTOS FUNDAMENTALES SOBRE TRÁFICO AUTOSIMILAR (5)**

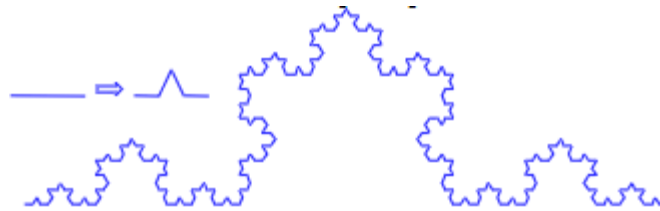
La autosimilitud y la fractalidad describen el fenómeno en el que cierta propiedad de un objeto se preserva con respecto a la escalización en el tiempo o en el espacio. Este fenómeno sucede en imágenes naturales, en el subdominio de convergencia de ciertos sistemas dinámicos y en muchas series de tiempo (como los procesos de tráfico que nos interesan). En un objeto autosimilar o fractal, sus partes magnificadas se asemejan a la forma del objeto completo, donde la semejanza se mide en algún sentido adecuado.

#### **1.5.1 Autosimilitud determinística (5)**

Mediante la iteración de cierto procedimiento se puede obtener, por simple construcción, la forma más sencilla de autosimilitud. Un buen ejemplo es el copo de nieve de Von Koch, que se construye, dividiendo cada línea en tres segmentos iguales y reemplazando el segmento de la mitad por dos segmentos iguales, como

en un triángulo equilátero. Si el procedimiento se repite para cada nuevo segmento indefinidamente, cualquier pequeña porción de la curva de Von Koch puede magnificarse para reproducir, exactamente, una porción mayor.

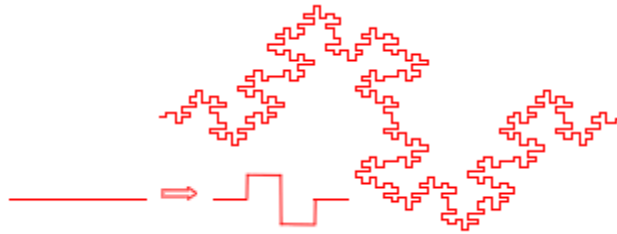
**Figura 2 Copo de nieve de Von Koch**



Fuente: Monroy, Marco Aurelio Alzate. *Introducción al Tráfico Autosimilar en Redes de Comunicaciones*. (5)

Esta propiedad se conoce como “Autosimilitud exacta”. Obsérvese que un segmento de línea es un objeto unidimensional autosimilar pues puede dividirse en  $N$  segmentos idénticos, cada uno de ellos escalizado por el factor  $r = N^{-1}$ . Igualmente, un cuadrado es un objeto bidimensional autosimilar pues puede dividirse en  $N$  cuadrados idénticos, donde el lado de cada uno de ellos se ha escalizado por un factor  $r = N^{-1/2}$ . Finalmente, un cubo es un objeto tridimensional autosimilar que puede dividirse en  $N$  cubos idénticos, donde el lado de cada uno de ellos se ha escalizado por un factor  $r = N^{-1/3}$ . Generalizando, un objeto  $D$ -dimensional autosimilar puede dividirse en  $N$  copias más pequeñas de sí mismo donde el lado de cada copia se ha escalizado por un factor  $r = N^{-1/D}$ . Así pues, la dimensión de un objeto autosimilar compuesto por  $N$  partes idénticas en las que cada lado ha sido escalizado por un factor  $r$  es  $D = \log(N) / \log(1/r)$ . En el caso de la curva de Von Koch, cada segmento de línea se convierte en  $N = 4$  subsegmentos, cada uno de los cuales se escala por un factor  $r = 1/3$ , de manera que su dimensión es  $D = \log(4)/\log(3) = 1.26$ . Como esta es una dimensión fraccional, a estos objetos autosimilares se les llama también fractales (Fractional Dimensional). La Figura 3 muestra una variación del copo de nieve de Von Koch de dimensión fractal 1.5.

**Figura 3 Variación del copo de nieve de Von Koch**

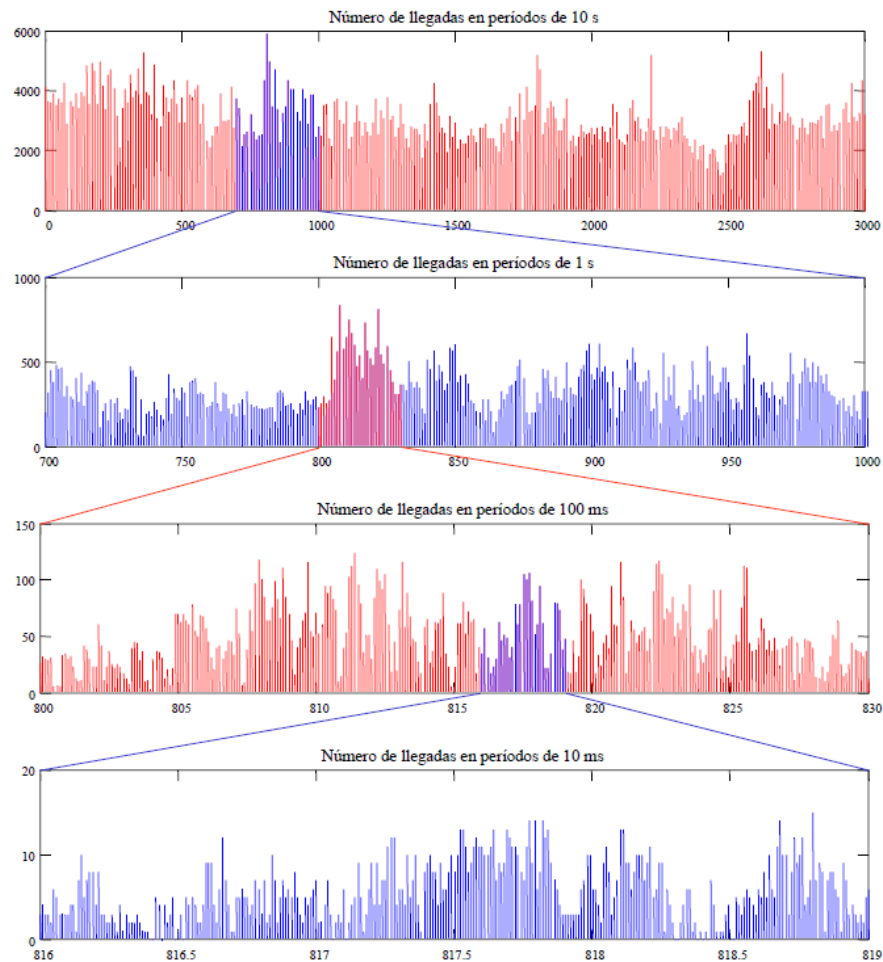


Fuente: Monroy, Marco Aurelio Alzate. *Introducción al Tráfico Autosimilar en Redes de Comunicaciones*. (5)

### 1.5.2 Autosimilitud estocástica (5)

La autosimilitud estocástica, como la determinística, se puede ilustrar visualmente. En la Figura 4 observamos el número de paquetes que transitan por un segmento de una red Ethernet, en períodos de 10 s, esto es, la barra que aparece en el instante  $t$  de la gráfica corresponde al número de paquetes que pasaron por ese segmento de la red en el intervalo  $[t, t+10)$ . En Figura 4 (b) se magnifico por un factor 10 una porción de la Figura 4 (a), de manera que se redujo la granularidad a 1 s. Este proceso se repite en la Figura 4 (c) y (d), que tienen granularidades de 0.1 y 0.01 segundos respectivamente, esto es, la barra que aparece en el instante  $t$  de la Figura 4 (d) corresponde al número de paquetes que pasaron por ese mismo segmento de la red en el intervalo  $[t, t+0.01)$ . Debemos admitir que las cuatro gráficas de la Figura 4 “se parecen mucho” entre sí. Sin embargo, a diferencia de los objetos fractales determinísticos, cada gráfica no es una reproducción exacta de la gráfica anterior. Por supuesto, sería mucho pedir si esperásemos dicha fractalidad en un proceso tan aleatorio como la llegada de paquetes a una red Ethernet. Pero si consideramos que el tráfico observado es una traza muestral de un proceso estocástico y restringimos la similitud a ciertas estadísticas de las series de tiempo reescaladas, descubriremos fácilmente autosimilitud exacta en los objetos matemáticos abstractos y autosimilitud aproximada en las realizaciones específicas, como la de la Figura 4.

**Figura 4 Número de paquetes que transitan por un segmento de una red Ethernet**



Fuente: Monroy, Marco Aurelio Alzate. *Introducción al Tráfico Autosimilar en Redes de Comunicaciones*. (5)

### 1.6 Estimación del parámetro de Hurst (6)

Al simular tráfico como movimiento browniano fraccional, debemos ajustar el valor medio, la varianza y el parámetro de  $H$  para que correspondan con los respectivos estimadores estadísticos tomados de mediciones de tráfico real. Más allá de la lenta convergencia de los estimadores tradicionales para la media y la varianza, no existe mayor dificultad en obtener valores adecuados para estos dos

parámetros (claro, si la ergodicidad es una suposición válida). Sin embargo, la estimación del parámetro H es mucho más fácil.

Este estimador es utilizado para determinar la autosimilitud de las muestras en el tiempo; los procesos autosimilares producen un incremento del índice de dispersión de la forma  $m^{-1}t^{2(H-1)}$ , graficando  $\log(\text{var}(t))$  contra  $\log(t)$ , donde  $\text{var}$  es la varianza y  $t$  es el tiempo. El diagrama varianza-tiempo, determina el parámetro Hurts (H), que dependerá de la alineación de la pendiente  $m$ .

Se debe estar en un rango entre  $0.5 < H < 1$ , donde un valor de  $H=0.5$  indica ausencia de autosimilitud. Cuanto más cerca de 1 esté H, mayor será el grado de persistencia y de la dependencia de largo plazo.

## 1.7 OPNET MODELER (7)

Los diseñadores de redes están constantemente desafiados por la creciente complejidad de los protocolos de comunicaciones y el aumento de los despliegues de red. Debido a esto necesitan un software de simulación robusto el cual les permita hacer una simulación del comportamiento de los protocolos de red de forma fácil e intuitiva, además que sea capaz de analizar de manera eficiente su desempeño en los modelos de infraestructura de una red de escala real.

OPNET Modeler ® acelera el proceso para el análisis y diseño de redes de comunicación, dispositivos, protocolos y aplicaciones. Los usuarios pueden analizar las redes simuladas para comparar el impacto de los diseños de diferentes tecnologías. Además OPNET Modeler ® incluye un amplio conjunto de protocolos, tipos de redes y tecnologías como lo son:

- VoIP
- TCP
- OSPFv3
- MPLS
- IPv6
- Others

## **1.7.1 MODELADO DE ALTA FIDELIDAD EN OPNET (8)**

### **Modelado de Arquitectura**

El diseño del modelador se basa en una serie de editores jerárquicos paralelos a las redes reales, equipos y protocolos.

### **1.7.2 Editor de proyectos**

Este representa gráficamente la topología de una red de comunicaciones.

El usuario crea un nodo y enlaza objetos para representar los elementos de una topología de red y configurarlos rápidamente usando cuadros de diálogo. Los proyectos pueden contener varios escenarios para facilitar las comparaciones de diseño.

### **1.7.3 Editor de nodos**

Este captura la arquitectura de un dispositivo de red o sistema que representa el flujo de datos entre elementos funcionales, llamados "módulos".

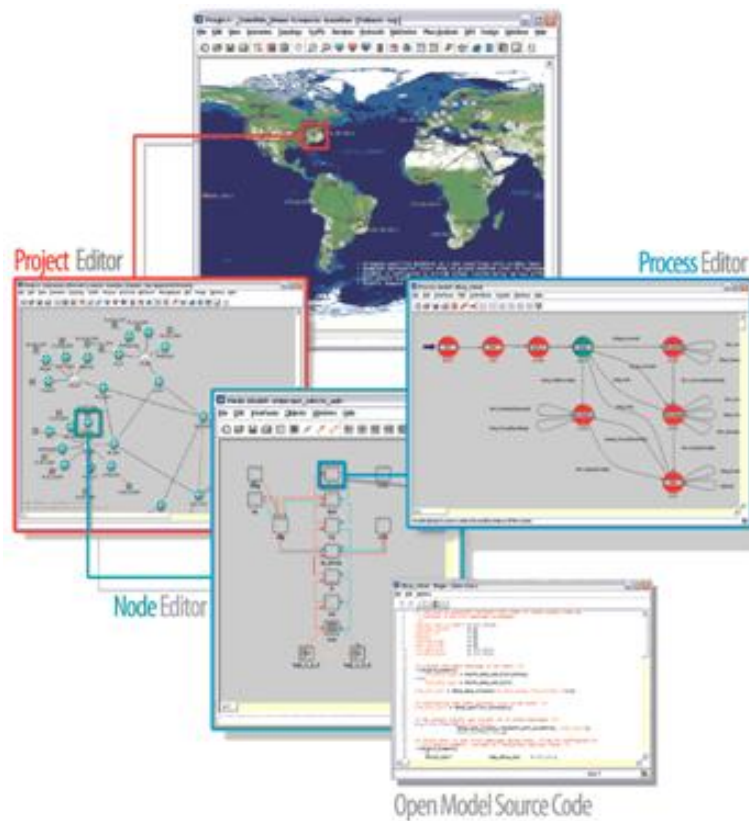
Los módulos suelen representar los protocolos de red o algoritmos y se asignan los modelos de procesos (desarrollado en el editor de procesos) para lograr el comportamiento requerido.

### **1.7.4 Editor de procesos**

Este utiliza una potente máquina de estados finitos (FSM), la cual está enfocada en apoyar la especificación detallada de los protocolos, recursos, aplicaciones y algoritmos.

Las máquinas de estados finitos son dinámicas y pueden ser generadas durante la simulación en respuesta a eventos específicos. El código C / C++ que gobierna cada estado de un modelo de proceso puede ser rápidamente modificado para requisitos particulares. El núcleo de procedimiento APIs de OPNET existe para facilitar el desarrollo y apoyar los mecanismos de comunicación comunes, tales como paquetes, las colas y el tráfico.

**Figura 5 Opnet modeler**



Fuente: OpnetModeller home page.

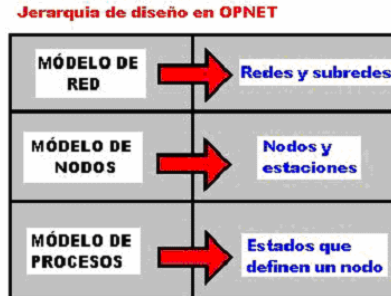
[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html) (8)

### 1.7.5 FUNCIONAMIENTO (7)

Como veremos a continuación OPNET es un simulador que posee una interfaz muy seductora para los usuarios. Esto es debido a que incluye varias librerías de modelos. El código fuente de estas librerías es accesible si se dispone de la versión OPNET Modeler, esto consigue que el programador se pueda familiarizar más rápidamente con toda la jerarquía interna del programa. Para ser utilizado, primero el usuario tiene que comprender la jerarquía que se utiliza para poder plantear las simulaciones.

Esta jerarquía de diseño se muestra en la figura 3:

**Figura 6 Jerarquía de diseño en OPNET**



Fuente: OpnetModeller home page.

[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html) (7)

Tenemos un modelo de red donde irán definidas las redes y subredes de la simulación. Seguidamente disponemos de un modelo de nodos donde se define la estructura interna de éstos y por último tenemos el modelo de procesos donde se definen los estados que definen un nodo. En la correcta simulación de un proyecto se debe haber entendido los aspectos anteriores; puestos que de lo contrario la simulación saldría errónea y no serviría. Para un mayor entendimiento profundizaremos en este aspecto. Como hemos mencionado con anterioridad, poseemos un Modelo de nodos. Este modelo de nodos funciona como muestra la figura 4:

**Figura 7 Funcionamiento del modelo de Nodos.**













Fuente: OpnetModeller home page.

[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html) (8)

Un nodo puede tener en su interior varios módulos. Éste tiene una función definida en su interior, así, un módulo llamado receptor tendrá la función de recibir los paquetes de otro. Para realizar esta parte de la simulación necesitamos tener el Node editor que podremos encontrar en el OPNET Modeler.

Los módulos que nos ofrece el node editor son los siguientes:

**Tabla 1 Módulos del Node Editor**

	Icono estándar para un módulo de procesado. Su función principal es construir bloques de modelo de nodo.
	Icono estándar usado para un módulo de cola. Lo que le hace diferente del resto de módulos es que el módulo de colas tiene recursos adicionales internos llamados subcolas.
	Icono usado para un módulo de transmitir. Transmisor Punto a Punto.
	Icono usado para un módulo de recibir. Receptor Punto a Punto.
	Icono usado para un módulo de transmitir. Transmisor tipo bus.
	Icono usado para un módulo de recibir. Receptor tipo bus.
	Icono usado para un módulo de transmitir. Transmisor por radiofrecuencia.
	Icono usado para un módulo de recibir. Receptor de radiofrecuencia.
	Icono estándar usado para las comunicaciones via satélite.
	Icono estándar para un módulo esys.

Fuente: OpnetModeller home page.











[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html) (7)

Para crear modelos de procesos, el software ofrece un editor llamado *process editor*. En éste se pueden definir los estados y las transiciones de los estados. Una vez se ha creado el modelo de proceso, el siguiente paso es la compilación y la verificación. En el momento de realizar una simulación el programador tendrá que efectuar los siguientes pasos:

- Descripción del modelo
- Modelo de red
- Modelo de nodos
- Modelo de procesos
- Resultado de la simulación

El process model dispone de los siguientes iconos de configuración:

**Tabla 2 Opciones del Process Model**

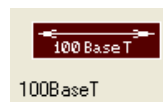
	Create State: Crea un estado dentro del process model.
	Create Transition: Crea transiciones entre los diferentes estados.
	Set Initial State: Establece el Estado Inicial.
	Edit State Variables: Establece los estados variables de la simulación.
	Edit Temporary Variables: Instaura Estados Temporales para el process model
	Edit Header Block: Instaura el Header Block donde se declaran las variables, macros, tipos de datos, etc., incluidos en el process model.
	Edit Function Block: Instaura el Function Block en el cual se crean nuevas funciones en C++ para el process model
	Edit Diagnostic Block: Establece el Diagnostic Block que contiene funciones sobre el estado de la simulación.
	Edit Termination Block: Establece el Termination Block donde se establecen unas funciones que son ejecutadas antes de terminar la simulación.
	Compile Process Model: Compilación del código.

Fuente: OpnetModeller home page.

[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html) (7)

## Cable

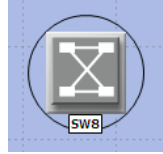
**Figura 8 Cable par trenzado**



Fuente: opnet modeler.

## Switch

**Figura 9 Switch**



Fuente: opnet modeler.

## Server

**Figura 10 Server**



Fuente: opnet modeler.

**Figura 11 Application definition**



Fuente: opnet modeler

**Figura 12 Profile definition**

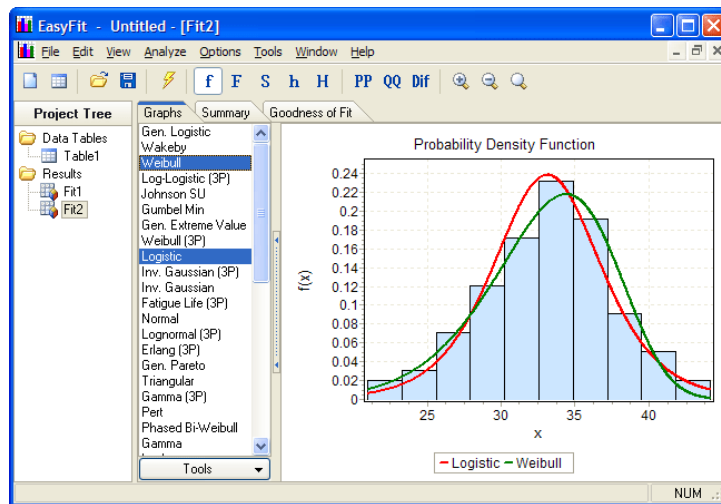


Fuente: opnet modeler.

## 1.8 EASYFIT (9)

EasyFit permite ajustar automáticamente las distribuciones a los datos de la muestra y seleccionar el mejor modelo en unos segundos. Está diseñado para hacer el análisis de datos lo más fácil posible, dejar los detalles técnicos complejos entre bastidores y permitirle que se enfoque en los objetivos de su proyecto.

**Figura 13 Easyfit**



Fuente: [http://www.mathwave.com/es/products/easyfit\\_features.html](http://www.mathwave.com/es/products/easyfit_features.html) (9)

EasyFit permite seleccionar fácil y rápidamente la distribución de probabilidades que mejor se ajuste a los datos. Las características principales son las siguientes:

- compatibilidad con más de 55 distribuciones
- modo eficaz de ajuste automático de datos
- capacidad de ajuste manual de distribución
- gráficos interactivos
- pruebas de bondad de ajuste

Una variedad de características únicas hace que esta potente herramienta de análisis de datos sea una solución muy flexible. Con EasyFit, usted puede:

- analizar sus conjuntos de datos aun cuando sean muy grandes
- desarrollar modelos mejores usando distribuciones avanzadas
- tomar decisiones correctas y rápidas con informes hipertextuales

Además, EasyFit permite ver gráficos de distribuciones y examinar sus propiedades sin introducir datos, generar números aleatorios, calcular estadísticas descriptivas y mucho más.

### **1.8.1 Pruebas de bondad de ajuste (9)**

Las pruebas de bondad de ajuste miden la compatibilidad de una muestra aleatoria con una función teórica de distribución de probabilidades. En otras palabras, estas pruebas demuestran qué bien la distribución seleccionada se ajusta a sus datos. EasyFit es compatible con las pruebas Kolmogorov-Smirnov, Anderson-Darling y Chi-cuadrado

## **2 ESTADO DEL ARTE**

Para el desarrollo del presente proyecto de investigación se realizó una búsqueda exhaustiva en diferentes bases de datos, partiendo de modelos matemáticos existentes propuestos en diferentes investigaciones para las nuevas aplicaciones. Estos modelos matemáticos se utilizarán para generar el tráfico agregado compuesto por las aplicaciones tradicionales (FTP, email etc.) y por las nuevas aplicaciones (VoIP, Iptv, p2p, youtube, etc).

### **2.1 Modelado de tráfico de clientes (10)**

Es importante saber el comportamiento que tienen los usuarios cuando inician una sesión en internet, ya que cada uno lo hace de manera diferente, es decir, no todos comienzan a navegar a la misma hora, no duran el mismo tiempo en línea, no consultan el mismo tipo de información, ni realizan las mismas actividades en la red. Para esto en (10) se realiza una caracterización (detallada y sistemática del tráfico de clientes WWW en dos entornos: uno académico y otro empresarial) a partir de la cual proponen 3 modelos con las distribuciones de probabilidad totalmente definidas y parametrizadas.

#### **2.1.1 Modelo orientado a página (POM) (10)**

Se trata de un modelo que incluye los niveles de sesión, página y paquete, obviando el nivel de conexión. Se pretende con él generar paquetes que serán inyectados en el enlace, por lo que no es necesario considerar el nivel de conexión, ya que en el nivel de página se concentra la mayor parte del rafagueo.

En este caso, el efecto de la red, los protocolos y los servidores consultados están incluidos dentro de la caracterización de los parámetros del modelo, por lo que será necesario diferenciar los dos sentidos de la comunicación en los parámetros de los niveles de página y paquete. De esta manera, se diseñaron dos modelos: uno para el sentido descendente y otro para el ascendente, que compartirán el nivel de sesión. Así pues, los parámetros involucrados en este modelo serán por tanto los que se relacionan a continuación:

**Tabla 3 Resumen de los parametros del modelo POM propuesto**

PARÁMETRO	DISTRIBUCIÓN	AJUSTE EN EL ENTORNO EMPRESARIAL	AJUSTE EN EL ENTORNO ACADÉMICO
Número de páginas por sesión	Log-Normal	$\mu=23$ Pág. $\sigma=61,2$ Pág.	$\mu=7,4$ Pág. $\sigma=17$ Pág.
Tiempo entre páginas	Gamma	$\mu=42,2$ s $\sigma=154,3$ s	$\mu=63,6$ s $\sigma=206,8$ s
Tamaño de las páginas en el sentidos descendente	Pareto	$\mu=35984$ bytes $\alpha=1,66$	$\mu=26789$ bytes $\alpha=1,47$
Tamaño de las páginas en el sentidos ascendente	Pareto	$\mu=4370$ $\alpha=1,81$	$\mu=4007$ $\alpha=1,79$
Tamaño de los paquetes en el sentidos descendente	Determinista	1500 bytes	1500 bytes
Tiempo entre paquetes descendentes dentro de las páginas	Log-Normal	$\mu=0,64$ s $\sigma=10,88$ s	$\mu=1,29$ s $\sigma=43,08$ s
Tiempo entre paquetes ascendentes dentro de las páginas	Log-Normal	$\mu=1,22$ s $\sigma=51,52$ seg	$\mu=3,49$ s $\sigma=314,2$ seg

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10)

### 2.1.2 Modelo orientado a tiempos (TOM) (10)

Al igual que el modelo POM, incluye los niveles de sesión, página y paquete, obviando el nivel de conexión. Todo lo dicho anteriormente acerca del efecto de la red, los protocolos y los servidores consultados es igualmente válido en este caso. Esos efectos están incluidos dentro de la caracterización de los parámetros del

modelo, por lo que será necesario diferenciar los dos sentidos de la comunicación en los parámetros de los niveles de página y paquete. De esta manera se diseñaron dos modelos, uno para el sentido descendente y otro para el ascendente, que compartirán el nivel de sesión. Los parámetros involucrados en este modelo serán por tanto los que se relacionan a continuación:

Tabla 4 Resumen de los parametros del modelo TOM propuesto

PARÁMETRO	DISTRIBUCIÓN	AJUSTE EN EL ENTORNO EMPRESARIAL	AJUSTE EN EL ENTORNO ACADÉMICO
Número de páginas por sesión	Log-Normal	$\mu=23$ Pág. $\sigma=61,2$ Pág.	$\mu=7,4$ Pág. $\sigma=17$ Pág.
Tiempo entre páginas	Gamma	$\mu=42,25$ $\sigma=154,3$ s	$\mu=63,6$ s $\sigma=206,8$ s
Tamaño de las páginas en el sentidos descendente	Pareto	$\mu=35984$ bytes $\alpha=1,66$	$\mu=26789$ bytes $\alpha=1,47$
Tamaño de las páginas en el sentidos ascendente	Pareto	$\mu=4370$ $\alpha=1,81$	$\mu=4007$ $\alpha=1,79$
Duración de las páginas	Pareto	$\mu=34,29$ s $\alpha=1,59$	$\mu=125,76$ s $\alpha=1,14$
Tamaño de los paquetes en el sentidos descendente	Determinista	1500 bytes	1500 bytes
Tiempo entre paquetes descendentes dentro de las páginas	Log-Normal	$\mu=\frac{Duraciónpag.emp(s)(1500bytes)}{Tamaño pag.emp(bytes)}$ $\sigma=\mu*17,78$	$\mu=\frac{Duraciónpag.acad(s)(1500bytes)}{Tamañoacad.emp(bytes)}$ $\sigma=\mu*33,5$
Tiempo entre paquetes ascendentes dentro de las páginas	Log-Normal	$\mu=\frac{Duraciónpag.emp(s)(1500bytes)}{Tamaño pag.emp(bytes)}$ $\sigma=\mu*35,28$	$\mu=\frac{Duraciónpag.acad(s)(1500bytes)}{Tamañoacad.emp(bytes)}$ $\sigma=\mu*89,98$

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*.  
[Universidad de malaga] Malaga (10)

### 2.1.3 El modelo orientado a conexión (COM) (10)

En [7] se propone un modelo de nivel de aplicación que denominaremos modelo orientado a conexión o COM (Connection Oriented Model). Se trata de un modelo que incluye los niveles de sesión, página y conexión. Obviamente, en este caso, no se modela el nivel de paquete, ya que la generación de éstos corresponderá a los protocolos TCP/IP. Se pretende por tanto con este modelo generar conexiones que serán cursadas por los protocolos TCP/IP. Los parámetros que habría que tener en cuenta para este modelo son:

**Tabla 5 Resumen de los parámetros del modelo COM propuesto**

PARÁMETRO	DISTRIBUCIÓN	AJUSTE EN EL ENTORNO EMPRESARIAL	AJUSTE EN EL ENTORNO ACADÉMICO
Páginas por sesión	Log-Normal	$\mu=14$ Pág. $\sigma=36$ Pág.	$\mu=9,8$ Pág. $\sigma=26$ Pág.
Tiempo entre páginas	Gamma	$\mu=46,8$ s. $\sigma=168,6$ s	$\mu=76,4$ s. $\sigma=256,8$ s
Conexiones por página	Log-Normal	$\mu=5,3$ conex. $\sigma=12$ conex.	$\mu=4,5$ conex. $\sigma=12,7$ conex.
Tiempo entre conexiones	Gamma	$\mu=2,3$ s. $\sigma=4,5$ s.	$\mu=2,8$ s. $\sigma=5,1$ s.
Tamaño de las conexiones	Pareto	$\mu=5616$ bytes $\alpha=1,77$	$\mu=4616$ bytes $\alpha=1,44$
Tamaño de las peticiones	Log-Normal	$\mu=364$ bytes. $\sigma=101$ bytes	$\mu=349$ bytes. $\sigma=91$ bytes
Número de peticiones	Geométrica	$\mu=1,3$ paquetes	$\mu=1,1$ paquetes

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10)

### 2.2 Tráfico de aplicaciones varias (peer to peer, e-mail y otras) (11)

Según (11) para modelar el tamaño de las solicitudes y el tiempo entre llegadas de los diferentes tipos de tráfico, es necesaria la realización de un estudio previo de los datos reales facilitados por el operador de red. Debido a la enorme cantidad de datos procesados, por un lado, los análisis realizados con el tráfico acumulado en cada controlador de la red y por otra parte, un análisis detallado por cada aplicación se lleva a cabo. Estos análisis servirán como base para la construcción del modelo de tráfico.

En este estudio, además del tráfico capturado en los controladores de una red de fibra, el tráfico en el router de acceso Internet está disponible. Por medio de la

información del programa Flowscan, que utiliza la identificación del puerto, pudieron diferenciar el tráfico por servicios, protocolos, redes de origen y las redes de destino. Se procesaron las aplicaciones más importantes, con más volumen de tráfico. Por lo tanto, se ha analizado en detalle ftp, http, smtp, bittorrent, edonkey, kazaa, nntp. El resto de los servicios proporcionan un volumen insignificante de tráfico.

Con el fin de hacer un modelo preciso del tráfico, es necesario determinar las funciones de distribución de probabilidad de los servicios implicados. Para este fin, mediante el procesamiento de la información de tráfico en el router de acceso a Internet, es posible determinar el tamaño de paquetes del tráfico por aplicación. En la siguiente tabla se indican las distribuciones utilizadas por servicio y sus parámetros (11).

**Tabla 6 Distribuciones de diferentes servicios y sus parámetros**

Service	Distribution	Parameter
http	Pareto	Loc = 2025.7833 Shape = 5.1430
ftp	Lognormal	$\mu = 7.6554$ $\sigma = 0.4305$
smtp	Weibull	$\alpha = 5.1838$ $\beta = 2931.1571$
nntp	Lognormal	$\mu = 8.5973$ $\sigma = 0.0879$
Bittorrent	Lognormal	$\mu = 8.2762$ $\sigma = 0.0355$
Kazaa	Weibull	$\alpha = 11.2685$ $\beta = 5569.3851$
Edonkey	Weibull	$\alpha = 81.2422$ $\beta = 2602.2035$

Fuente: Roberto Garcíá, Victor Garcíá, Xabiel G. Pañeda ,David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load*. [Computer Science Department, University of Oviedo] septiembre de 2005 (11)

### 2.3 Modelado de Tráfico de Juegos en línea (12)

En este estudio se encontró que los juegos tipo MMORPG (Massively multiplayer online role-playing game) y FPS (First-Person Shooting) son similares en que ambos generan pequeños paquetes y requieren anchos de banda bajos. En la práctica, el requisito de ancho de banda de los es menor en los MMORPG debido a que es el que menos juego en tiempo real tiene. Las características más distintivas son la periodicidad fuerte, localidad temporal, la irregularidad, y la auto-

similitud observada en el tráfico de los MMORPG. La auto-semejanza del tráfico agregado se debe a las actividades de ocio de los jugadores individuales. Por otra parte, se muestra que la sesiones de llegada del juego dentro de una hora pueden ser modeladas mediante el modelo de Poisson, mientras que la duración de las sesiones del juego es de cola pesada como se observa en (12).

La información encontrada en este artículo no se utilizó para el desarrollo de nuestro proyecto debido a la falta de parámetros para realizar la simulación de tráfico de juego en línea en OPNET.

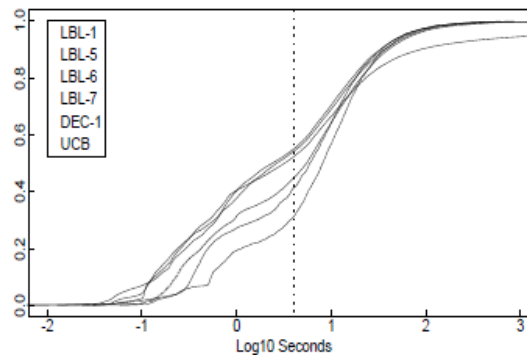
## **2.4 Modelado de Tráfico Redes Sociales**

No se encontró ningún estudio o investigación acerca de la caracterización o modelado de este tipo de tráfico, que permitiera la simulación de su comportamiento para nuestro proyecto, pero se asume que su comportamiento es similar al del tráfico HTTP (del cual si se tienen los parámetros) ya que este es el protocolo usado en cada transacción de la World Wide Web (www), a través de la World Wide Web y Con un navegador web, un usuario puede visualizar páginas web que contengan texto, videos e imágenes y navegar a través de ellas usando hiperenlaces.

## **2.5 Modelado de tráfico FTP**

Modelar el trafico FTP es particularmente importante porque las conexiones FTPDATA actualmente llevan la mayor parte de los bytes de datos en las redes. Se han hecho estudios en (13) que han encontrado que los paquetes de llegada FTPDATA están lejos de ser exponenciales, esto no es sorprendente, ya que los factores de la red conducen a un proceso muy diferente de llegadas sin memoria.

**Figura 14 FTPdata**



Fuente: Floyd, Vern Paxson and Sally. *Wide-Area Traffic: The Failure of Poisson Modeling*. [Lawrence Berkeley Laboratory and EECS Division, University of California] (13)

La figura 1 muestra que en cada caso la cola superior de la distribución es mucho más pesada que el caso exponencial, y es mejor aproximación una distribución lognormal o log-logistic.

Según (14), en el tráfico IP medido, la aplicación de usuario es FTP y es ejecutada a través de TCP / IP, donde la distribución del tamaño de archivos de descarga y su valor medio son los siguientes:

**Tabla 7 Modelo de una fuente de tráfico FTP**

<b>Variable</b>	<b>Distribution</b>	<b>Mean</b>
<i>Inter-arrival time (s)</i>	exponential	30
<i>Download file size(bytes)</i>	exponential	10000

Fuente: Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN*. (14)

## 2.6 El modelado de tráfico de video

El modelado de tráfico de video VBR es bastante complejo, porque el perfil de utilización de ancho de banda se caracteriza por un proceso al azar con dependencia a largo plazo. El proceso aleatorio que describe este tipo de perfil de tráfico puede ser modelado como clásicamente con un proceso auto-similar, un proceso de Markov, o bien un proceso  $M / G / \infty$ . El análisis de seguimiento de

vídeo comprimido real de archivos confirma la característica de auto-similitud, que es el resultado del alto rango de dependencia observado en la función de auto correlación del proceso aleatorio como se muestra en (15).

La caracterización de los patrones de carga de trabajo de la Web 2.0 es fundamental para planificar, diseñar y construir servicios más eficientes para compartir videos. Se analizó el sitio Web 2.0 más popular para descarga de videos, YouTube. La popularidad y las características únicas de YouTube, presentan nuevos desafíos para medir su capacidad de ampliación a largo plazo. Para hacer frente a estos desafíos, se estudió el tráfico de YouTube en los vídeos más populares para observar y estudiar sus características (16) .

Debido a que YouTube es considerado una enorme videoteca, su esquema de acceso tiene un impacto importante en la distribución del tráfico en Internet. Por lo tanto, la comprensión de las características de YouTube y sitios similares para compartir videos es fundamental para analizar el tráfico de red (16).

Para obtener un resumen de las estadísticas de los videos más populares, en [15] se incluyeron archivos de vídeo único, número de visitas, duración y tamaño de archivo de 4100 videos que fueron recogidos a diario. Se repitió el mismo procedimiento para obtener los datos semanales de los videos más populares. Se observó que los cambios en la lista diaria de los 100 mejores videos ocurren muy a menudo, pero los cambios en la lista semanal de los videos son poco frecuentes.

Finalmente, el resultado muestra que los videos populares tienen duraciones muy por debajo del máximo de diez minutos. Este hecho se puede observar en los valores promedio y la mediana de la duración de vídeo en los datos semanales y diarios establecidos, que van desde 3,89 hasta 4,41 minuto, respectivamente (16).

Las últimas filas de la tabla consideran las características de los distintos tamaños de archivo en el conjunto de datos. Como era de prever, los tamaños de archivo de contenidos de vídeo son de gran tamaño. También se observó que el tamaño promedio y la mediana de los conjuntos de datos diarios y semanales están cerca unos de otros. Por otra parte, el coeficiente de variación (COV) de tamaño de los archivos al día es menor que uno, lo que sugiere el tamaño de los archivos no son muy variables.

**Tabla 8 Tabla de estadísticas de los videos más populares**

Time Frame	Daily	Weekly
Unique IDs	3,605	700
View Count		
Mean	52,759.18	271,090.60
Median	27,382.50	190,827.50
COV	1.74	1.01
Rating		
Mean	4.25	3.96
Median	4.60	4.32
COV	0.21	0.23
Duration ( Min)		
Mean	4.42	3.89
Median	3.41	3.11
COV	0.93	1.08
Model	Negative Binomial p = 0.0037 S = 1	Negative Binomial p = 0.0038 s = 1
File Sizes		
Mean (KB)	10,631.41	9,269.17
Median (KB)	8,330.83	7,297.01
COV	0.93	1.11
Model	Weibull a = 1.144 B = 11,193	Weibull a = 1.133 b = 11,212

Fuente: Mojgan Soraya, Masood Zamani, Abdolreza Abhari. *MODELING OF MULTIMEDIA FILES ON THE WEB 2.0*. [Ryerson University, Toronto, Canadá] (16)

## 2.7 Modelado de Trafico VoIP

Uno de los servicios más importantes hoy en día en las redes de comunicación es el servicio de voz. Hay un número de beneficios de Voz sobre IP (VoIP), que subrayan su importancia: comunicación reducida costo, el uso de la infraestructura integrada de IP, que participa en una aplicación multimedia, etc. VoIP también puede aprovechar las ventajas de las redes de paquetes conmutados. Conociendo las propiedades de auto-similares del tráfico de VoIP, existen muchos procesos auto-similares que se pueden aplicar para modelar este tráfico. Se propone, por ejemplo, uno de las formas más sencillas de modelar tráfico auto-similar usando fractional Gaussian noise (fGn) como se muestra en (17), este modelo está dado por:

### Ecuación 15 fractional Gaussian noise

$$|X(t) = m + \sigma Z_H(t),|$$

Fuente: Trang Dinh Dang, Balázs Sonkoly, Sándor Molnár. *Fractal Analysis and Modeling of VoIP Traffic*. [High Speed Networks Lab, Dept. of Telecommunications & Media Informatics] 2002 (17)

En [35], para el modelado de una llamada VoIP se consideraron cinco factores: el tiempo de llegada entre llamada, duración de la llamada, la tasa de rechazo de llamada, la decisión de destinatario de la llamada y el contenido de llamadas. La Tabla muestra las diferencias entre una llamada ordinaria y una llamada spam.

Según las estadísticas, los usuarios envían llamadas por 175-205 minutos y reciben llamadas por 107-110 min. Además, generalmente, los usuarios envían 7,5 llamadas y reciben 8,6 llamadas diarias en promedio. Con base en estas estadísticas, se han definido los atributos principales del modelo para una llamada ordinaria. El promedio del tiempo de llegada entre llamadas (5310) se deriva de la hipótesis de que 7,5 llamadas se generan en 11 h (8 a.m.-7 p.m.). Además, se obtuvo la media de la duración de una llamada basada en el promedio del número de llamadas diarias y el tiempo de envío de llamadas mensuales (18).

**Tabla 9 Parámetros de una llamada VOIP**

	Ordinary caller	Spam caller 1	Spam caller 2	Spam caller 3
Call inter-arrival time (s)	Exponential (5310)	Constant (60)	Constant (130)	Constant (310)
Call duration (s)	Exponential (90)	Lognormal (30,9)	Exponential (50)	Exponential (50)

Fuente: Hyung-Jong Kim a, Myuhng Joo Kim a, Yoonjeong Kim a, Hyun Cheol Jeong b. *DEVS-Based modeling of VoIP spam callers' behavior for SPIT level calculation*. (18)

## 3 METODOLOGÍA

### 3.1 Caracterización de los servicios actuales de internet

Con el fin de dar respuesta a los objetivos propuestos para esta investigación se abordó la caracterización de los servicios actuales de internet, para lo cual se recurrió a las múltiples bases de datos con muestras de tráfico que ofrece internet,

donde recolecto información de los modelos utilizados para diferentes servicios (Voip, youtube, p2p, etc.) y se reconoció cuáles son las distribuciones que más de asemejan a su comportamiento.

### **3.2 Comportamiento de los usuarios**

Se recolectó información acerca de los diferentes comportamientos de un usuario en una sesión de una red IP. En esta fase se determinaron las distribuciones estadísticas que modelan la entrada y salida de usuarios en una red según diferentes tipos de comportamiento.

### **3.3 Manejo del software OPNET**

Se estudió la herramienta de simulación OPNET MODELER, sus principales tipos de tráfico, dispositivos y herramientas de análisis (necesarias para la simulación).

### **3.4 Simulación**

Antes del desarrollo de la simulación se llevaron a cabo 3 pruebas, mediante las cuales se fortalecieron los conocimientos en OPNET y se fue construyendo paso a paso la simulación final.

La primera prueba se desarrolló para profundizar los conocimientos en OPNET MODELER simulando una red sencilla, utilizando en esta red la configuración por defecto que el software brinda para las diferentes aplicaciones.

La segunda prueba consistió en agregar varios tipos de servicios como son data base, email, http, voip, ftp, video y p2p en una misma simulación utilizando la configuración o valores por defecto de opnet y se observaron sus resultados.

En la prueba 3 se implementaron los modelos matemáticos investigados para cada uno de los servicios y observo su comportamiento de manera independiente y en conjunto

Finalmente se realizó una simulación de una red con 300 usuarios, los cuales utilizan diferentes servicios de internet aleatoriamente y tienen sesiones de duración variable.

### **3.5 Análisis estadístico**

Se realizó el análisis estadístico con ayuda de la herramienta Easyfit, con la cual se interpretaron los datos obtenidos en la simulación y se obtuvo una caracterización para el comportamiento del tráfico agregado.

## **4 SIMULACION Y RESULTADOS**

En esta sección se observara el proceso llevado a cabo en el desarrollo de la simulación, el cual fue dividido en diferentes pruebas, donde la configuración fue mejorando a medida que se avanzaba, además, se observaron los resultados obtenidos en cada prueba, para así llegar a una simulación final del tráfico de internet, a la cual se le realizó un análisis estadístico para obtener las distribuciones que más se ajustaban a las variables resultantes (paquetes/segundos, bits/segundos y queue delay).

### **4.1 SIMULACIÓN PRUEBA 1**

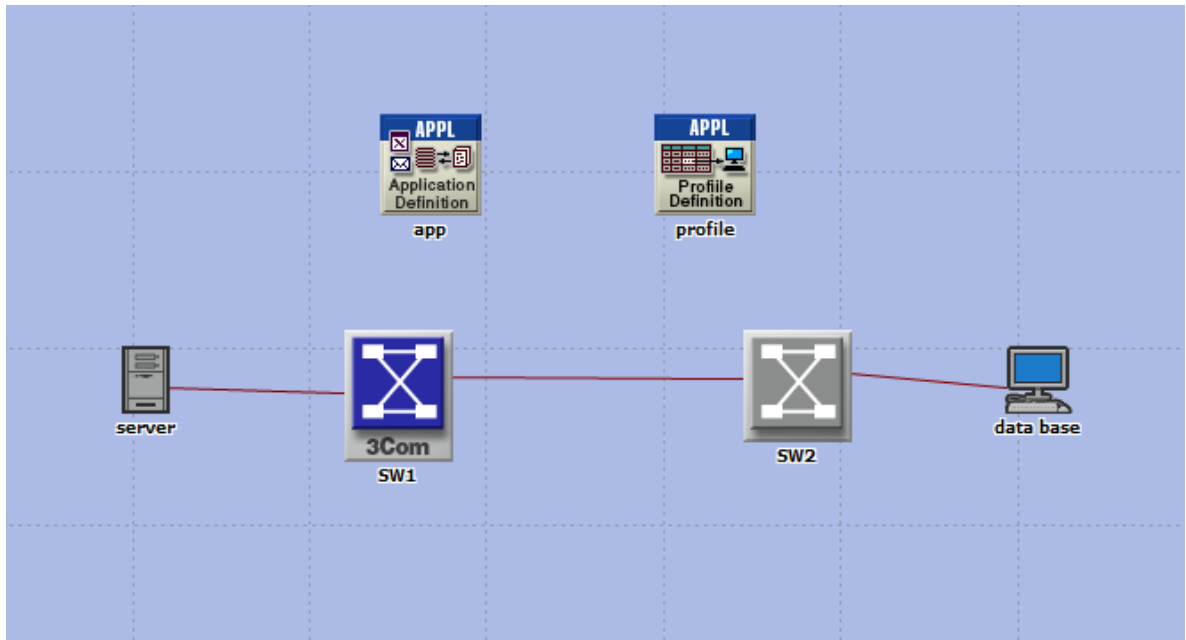
#### **4.1.1 Objetivo**

- Profundizar los conocimientos en OPNET MODELER realizando la simulación de una red sencilla, utilizando en esta red la configuración por defecto que el software brinda para las diferentes aplicaciones.

#### **4.1.2 Topología prueba 1**

En la Figura 15 se puede observar la topología de red usada en la simulación, la cual cuenta con 2 switches, los cuales se encuentran conectados a un usuario y un serve

**Figura 15 Topología prueba 1**



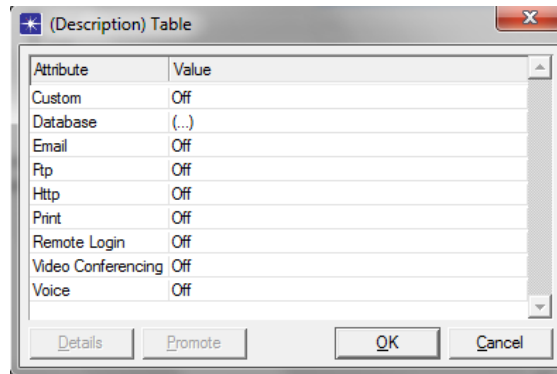
### 4.1.3 Configuración de aplicaciones

Una vez realizada la topología, se cargó una aplicación por defecto de OPNET MODELER en el servidor y en el usuario, en este caso se utilizó una base de datos como aplicación. Esto se realizó dando click derecho en el icono aplicación definition (ver Figura 11) que se encuentra en el escenario de la simulación, seguido de esto en la lista de opciones se elige *edit attributes* (ver Figura 21), esta opción permite nombrar y seleccionar la aplicación a utilizar de una lista que brinda OPNET (ver Figura 17). También fue necesario cargar un perfil para la aplicación de base de datos, ya que ahí se define lo siguiente: Tiempo de inicio de la simulación, tiempo entre repeticiones y duración de la transmisión. Esto se realizó dando click derecho en el icono *profile definition* (ver Figura 12) que se encuentra en el escenario de la simulación, seguido de esto en la lista de opciones se elige *edit attributes* (ver Figura 31), esta opción abre una ventana donde se nombra y se carga la aplicación de base de datos seleccionada anteriormente. Ya creada la aplicación, esta se carga en la opción *edit attributes* del servidor y del usuario, como se muestra en la Figura 16.

**Figura 16 Aplicaciones y perfiles en servidor y usuario**



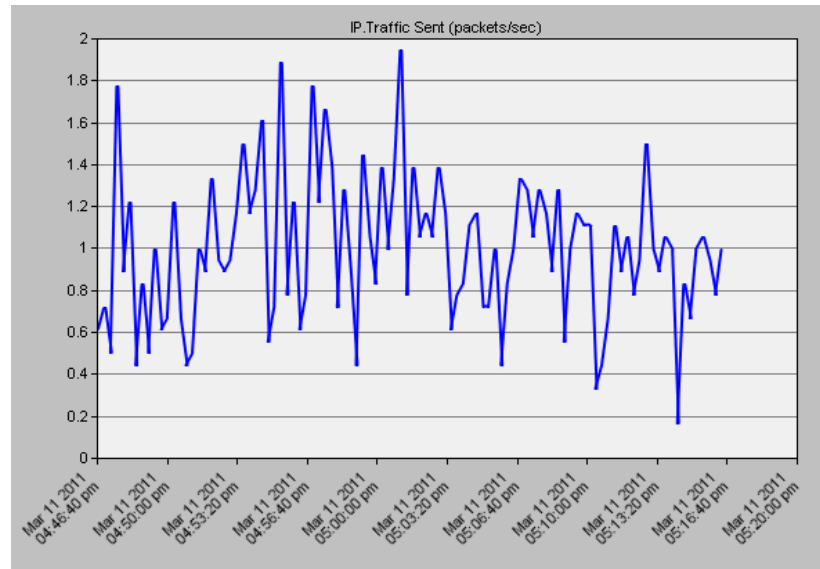
**Figura 17 Aplicaciones de opnet**



#### 4.1.4 Resultado prueba 1

Para observar los resultados de la simulación, es necesario ubicarse en el punto de la red donde se desea ver el flujo de tráfico, ya seleccionado este punto se da click derecho y se escoge *choose individual DES statistics* en el menú de opciones. En esta opción se escogió el trafico enviado por el usuario (paquetes/segundo) con protocolo IP (ver Figura 18).

**Figura 18 prueba 1 paquetes/segundo enviados por el usuario**



## 4.2 SIMULACION PRUEBA 2

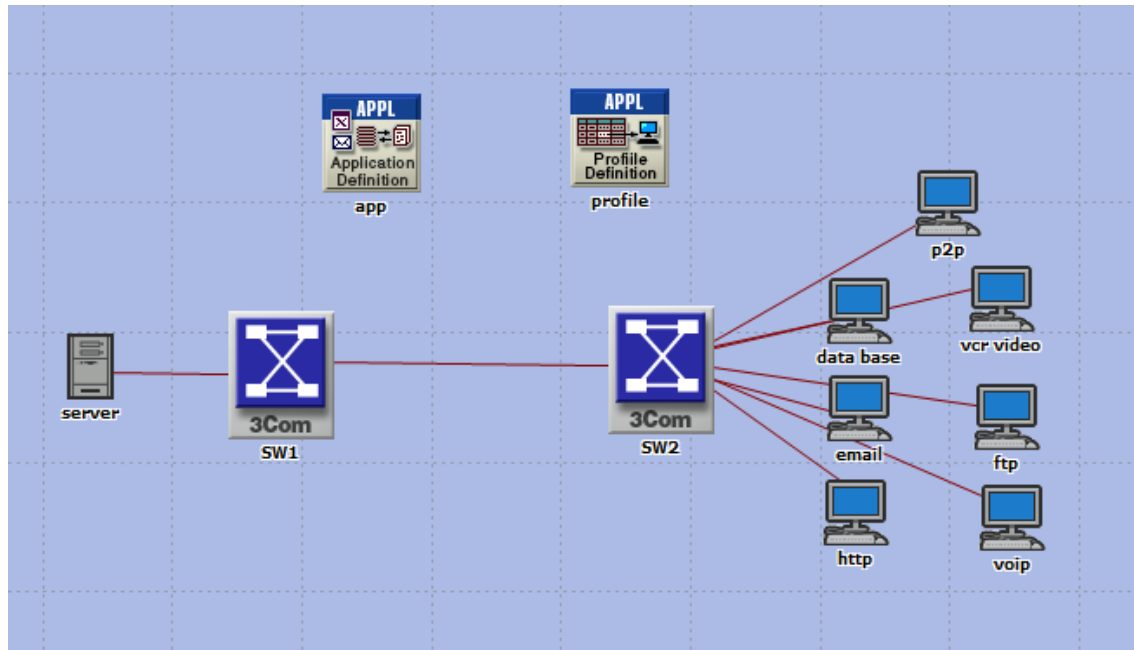
### 4.2.1 Objetivo

- Agregar varios tipos de servicios como son data base, email, http, voip, ftp, video y p2p en una misma simulación utilizando la configuración por defecto de OPNET y observar sus resultados.

### 4.2.2 Topología prueba 2

Para esta simulación se utilizó una topología muy similar a la de la prueba 1 (Figura 15), adicionándole de más usuarios, cada uno de estos con un servicio diferente como se observa en la Figura 19.

**Figura 19 Topología para la prueba 2**



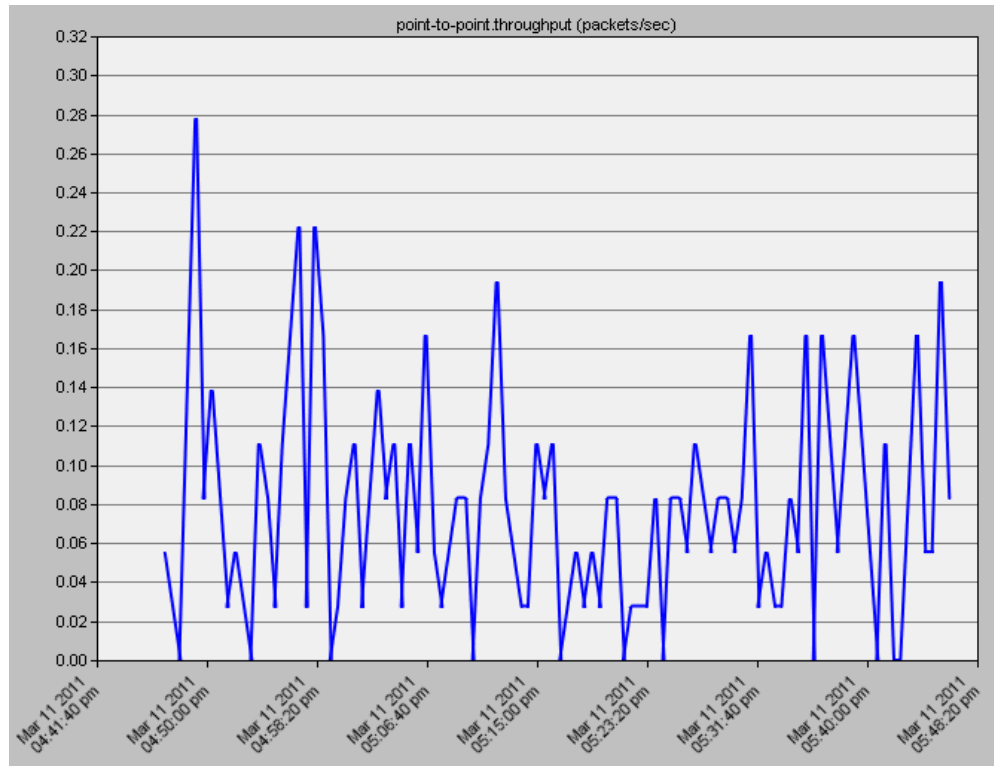
### 4.2.3 Configuración de aplicaciones

Partiendo de la prueba 1 se realizó el mismo procedimiento para cargar otras aplicaciones como lo son: email, http, voip, ftp, video y p2p (ver Figura 17). El cual consistió en agregar tanto en el servidor como en cada uno de los nuevos usuarios la aplicación que se buscaba observar.

### 4.2.4 Resultados prueba 2

Se observó el flujo de tráfico generado por estos usuarios, cada uno con un servicio distinto. Esto se hizo tomando los resultados de tráfico ip en la unión del switch 1 con el switch 2 (ver Figura 19).

**Figura 20 Tráfico entre SW1 y SW2**



### 4.3 SIMULACION PRUEBA 3

#### 4.3.1 Objetivo

- Implementar los modelos matemáticos investigados para cada uno de los servicios y observar sus resultados de manera independiente y en conjunto.

#### 4.3.2 Topología prueba 3

Esta topología fue la misma implementada en la prueba 2 (ver Figura 19), cada uno de estos utilizando un servicio diferente aleatoriamente.

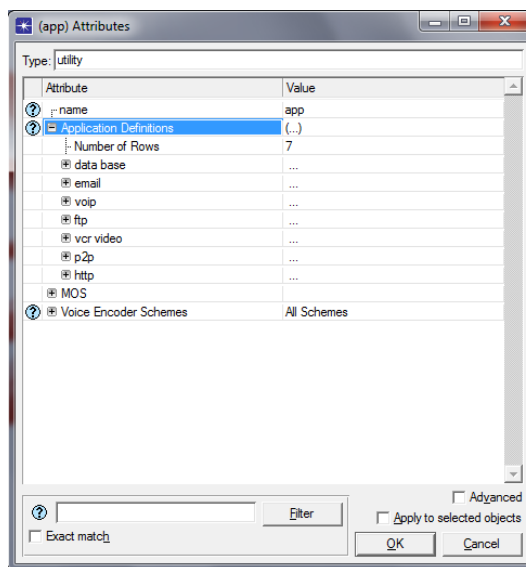
### 4.3.3 Configuración de aplicaciones

Existen dos aplicaciones usadas en OPNET MODELER para la configuración de servicios como lo son *Application definition* (ver Figura 11) y *profile definition* (ver Figura 12).

#### 4.3.3.1 APPLICATION DEFINITION

Para la simulación se implementaron diferentes modelos matemáticos previamente investigados estos fueron necesarios para representar el tráfico de diferentes servicios como voip, datos, video, peer to peer, e-mail, ftp y http. Estos modelos matemáticos se definen en esta aplicación (ver Figura 21).

Figura 21 Profile definition attributes



Los modelos matemáticos usados fueron los siguientes:

#### 4.3.3.1.1 VOIP

El modelo matemático utilizado para simular el tráfico voip (ver Figura 23) fue el que tiene por defecto la herramienta OPNET, debido a que en esta sección no se pudo configurar el tiempo de llegada, ni la duración de los usuarios. Como veremos más adelante en la configuración de *profile definition*, donde si usaremos un modelo matemático que está hecho con estas características. En esta aplicación permite modificar el tiempo en silencio y el tiempo de charla en la llamada (ver Figura 22).

Figura 22 Configuración voip

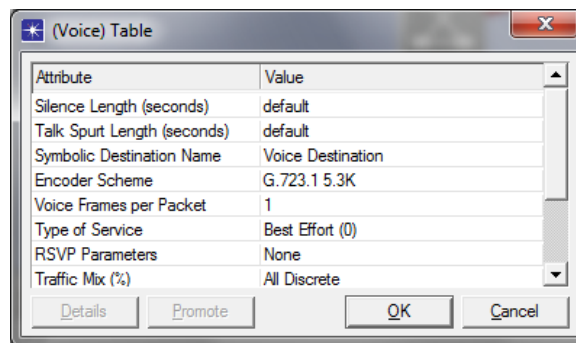
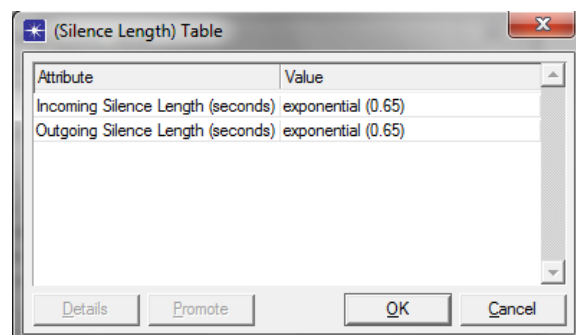


Figura 23 Silence length



#### 4.3.3.1.2 DATOS

Para simular el tráfico de datos se utilizó el modelo POM (MODELO ORIENTADO A PÁGINA) el cual mide el tiempo entre paquetes ascendentes dentro de las páginas (ver Tabla 10) (10). Este modelo está relacionado directamente con todos los datos enviados por el usuario en el momento de la navegación, debido a que mide el tiempo entre paquetes, por esto fue utilizado para representar en OPNET el tiempo entre transacciones, dado en segundos (ver Figura 24).

**Tabla 10 Modelo matemático para transaction interarrival time**

Parámetros	Lognormal
$\mu$	1.22
$\sigma$	51.52

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

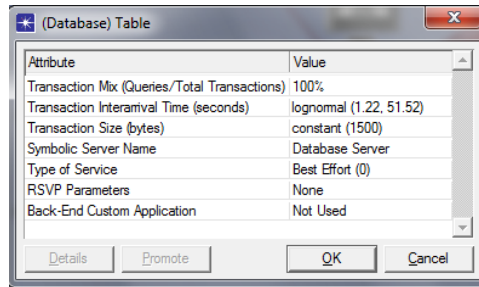
Se estudió el tamaño de los paquetes IP, encontrándose una serie de tamaños típicos de los que se estimaron sus correspondientes proporciones en cada una de las trazas. No obstante, en el caso del sentido descendente, se concluyó que la tendencia clara era el uso mayoritario de paquetes de 1500 bytes (10). Este fue utilizado en el parámetro *transaction size* en OPNET (ver Figura 24).

**Tabla 11 Modelo matemático para transaction size**

Parámetros	Constante
Valor	1500

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

**Figura 24 Configuración de datos**



#### 4.3.3.1.3 VIDEO

La principal página de videos en internet es youtube. En (16) se realizaron mediciones de la transferencia en esta página durante un día o una semana para los videos más vistos, a estos datos se les aplicó un modelo matemático (ver Tabla 12) el cual los representa, este modelo fue el que se utilizó para dar valor a el parámetro *frame size information* en OPNET (ver

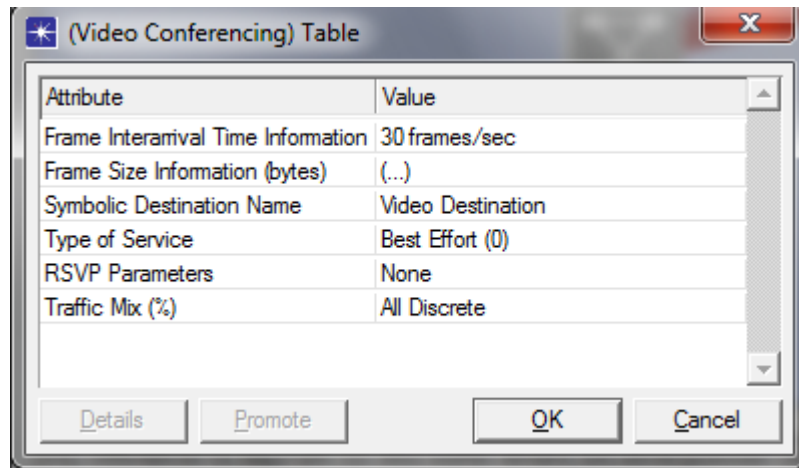
Figura 26). También se utilizó una velocidad de 30 frames/sec (ver Figura 25).

**Tabla 12 Modelo matemático para frame size**

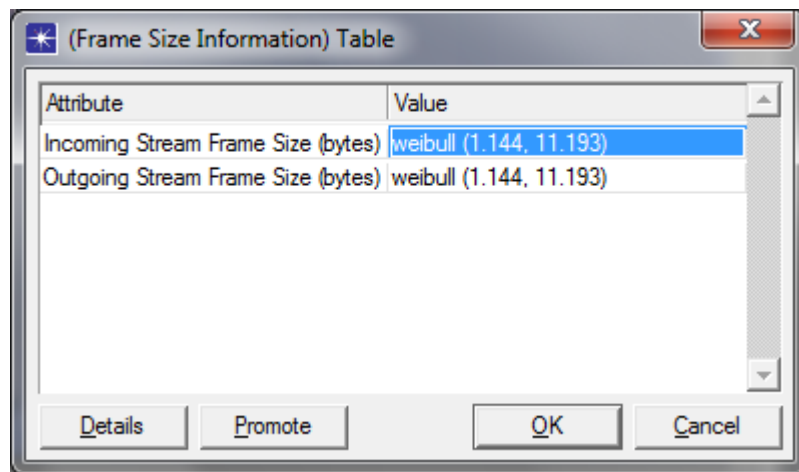
Parámetros	Weibull
A	1.144
B	11.193

Fuente: Mojgan Soraya, Masood Zamani, Abdolreza Abhari. *MODELING OF MULTIMEDIA FILES ON THE WEB 2.0*(16)

**Figura 25 Configuración de video**



**Figura 26 Modelo matemático frame size**



#### 4.3.3.1.4 PEER TO PEER

Se puede decir que el tiempo entre llegadas de FTP es similar al de los servicios *peer to peer*, por esto se utilizó este modelo (ver Tabla 13) para representarlo en OPNET en la configuración de la aplicación *peer to peer* transaction interarrival time. En (14) se puede ver este modelo matemático para FTP.

**Tabla 13 Modelo matemático para transaction interarrival time**

Parámetros	Exponencial
$\mu$	30

Fuente: Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN*.(14)

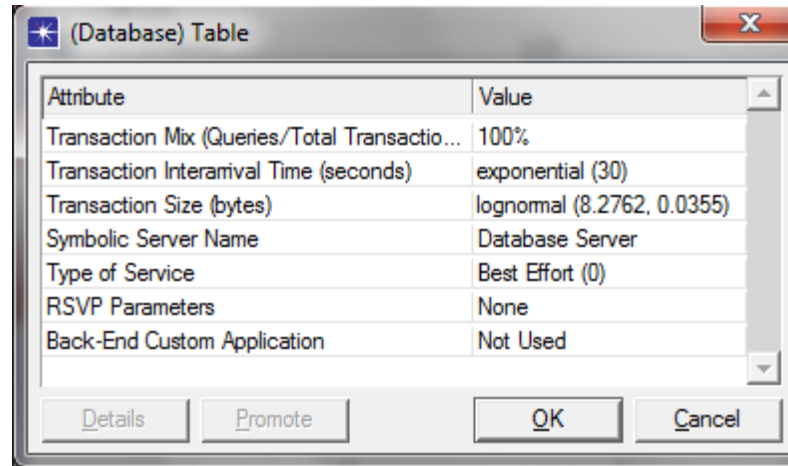
Existen diferentes programas que utilizan este sistema (p2p) para el intercambio directo de información en diferentes formatos, entre los ordenadores interconectados. Uno conocido y que aún se utiliza es el bittorrent, por lo que en (11) se realizaron análisis para ese y otros programas *peer to peer*. Por ser bittorrent uno de los más destacados se utilizó su modelo para la simulación. Este modelo matemático (ver Figura 14) representa la transferencia en bits de este programa. Por esto mismo, fue aplicado en opnet en transaction size (ver Figura 27).

**Tabla 14 Modelo matemático para transaction size peer to peer**

Parámetros	Lognormal
$\mu$	8.2762
$\Sigma$	0.0355

Fuente: Roberto Garcíá, Victor Garcíá, Xabiel G. Pañeda ,David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load*. (11)

**Figura 27 Configuración peer to peer**



#### 4.3.3.1.5 E-MAIL

El tiempo entre envíos de los correos depende exclusivamente de los usuarios, por esto tomamos este modelo matemático, el cual está basado en el comportamiento de los usuarios en páginas web. El modelo POM (MODELO ORIENTADO A PÁGINA) el cual mide el tiempo entre paquetes ascendentes dentro de las páginas (ver Tabla 15) (10) fue el utilizado en OPNET para representar la variable *sent interarrival time* en la configuración de la aplicación e-mail (ver Figura 28).

**Tabla 15 Modelo matemático para sent interarrival time**

Parámetros	Lognormal
$\mu$	1.22
$\Sigma$	51.52

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

El tiempo en el que se reciben y envían correos fue tomado según el comportamiento de los usuarios. El modelo POM (MODELO ORIENTADO A PÁGINA) el cual mide el tiempo entre paquetes descendentes dentro de las páginas (ver Tabla 16) (10) fue el que se utilizó (ver Figura 28).

**Tabla 16 Modelo matemático para receive interarrival time**

Parámetros	Lognormal
$\mu$	0.64
$\Sigma$	10.88

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

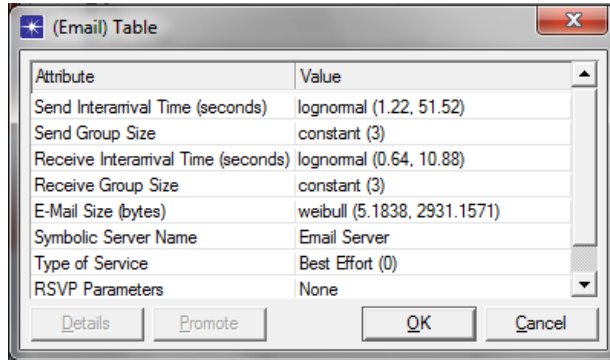
En (11) se realizó un análisis al tráfico de internet enfocado a *peer to peer*, pero a su vez se aplicaron modelos para otros tipos de tráficos como es el caso de los emails, donde se representa el tamaño en bits (ver Tabla 17), el cual se utilizó para representar el tamaño de los correos (ver Figura 28).

**Tabla 17 Modelo matemático para E-MAIL size**

Parámetros	Weibull
A	5.1838
B	2931.1571

Fuente: Roberto Garcíá, Victor Garcíá, Xabiel G. Pañeda ,David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load*. (11)

**Figura 28 Configuración email**



**4.3.3.1.6 FTP**

Los modelos matemáticos utilizados para representar el protocolo FTP fueron tomados de (ver Tabla 18, Tabla 19) (14), en donde se realizó un análisis para este protocolo con ayuda de la herramienta OPNET, por lo tanto fueron aplicados como se muestra en la Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN.*

**Figura 29.**

**Tabla 18 Modelo matemático para Download file size**

Parámetros	Constante
Valor	10000

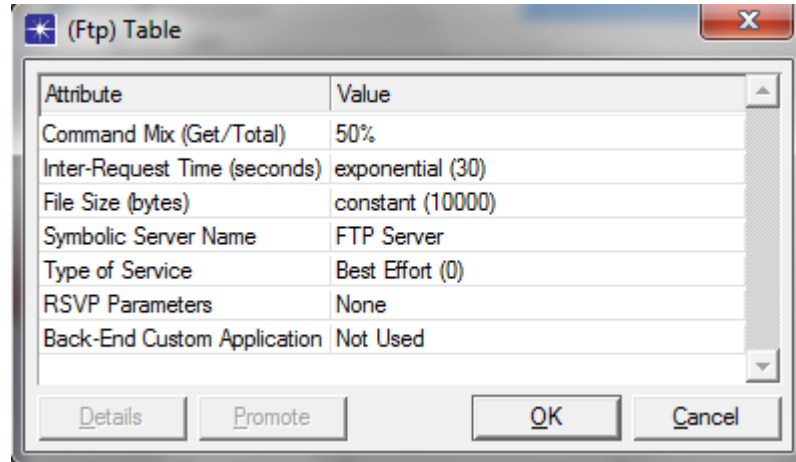
Fuente: Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN.*(14)

**Tabla 19 Modelo matemático para Inter-arrival time**

Parámetros	Exponencial
$\mu$	30

Fuente: Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN.*(14)

**Figura 29 Configuración ftp**



#### 4.3.3.1.7 HTTP

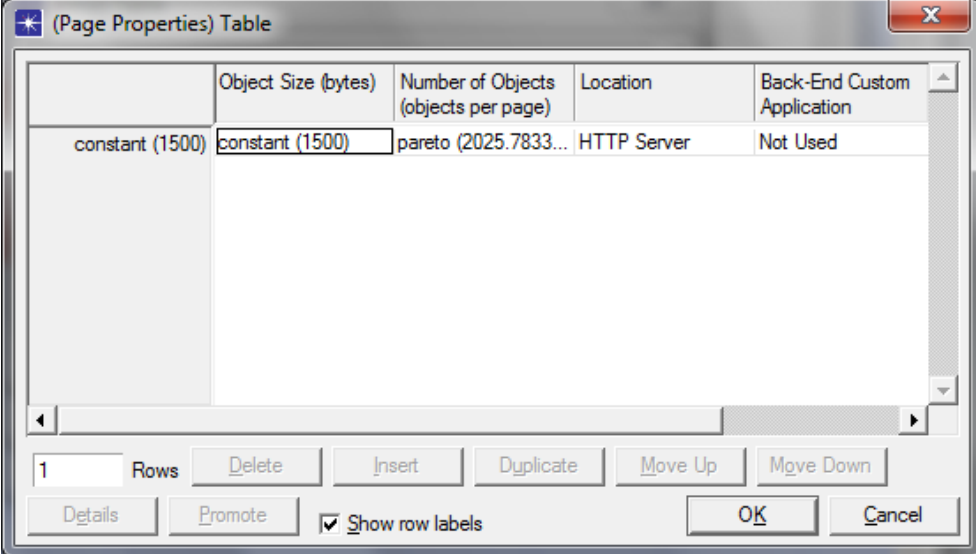
Se utilizó el mismo modelo de datos, para representar el tamaño de 1 objeto (ver Tabla 11). Se aplicó el modelo para http dado en (11) (ver Tabla 20), en este caso para representar el número de objetos por página en OPNET (ver Figura 30).

**Tabla 20 Numero de objetos en http**

Parámetros	Pareto
Loc	2025.7833
Shape	5.1430

Fuente: Roberto Garcíá, Víctor Garcíá, Xabiel G. Pañeda, David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load.* (11)

**Figura 30 Configuración http**



	Object Size (bytes)	Number of Objects (objects per page)	Location	Back-End Custom Application
constant (1500)	constant (1500)	pareto (2025.7833...	HTTP Server	Not Used

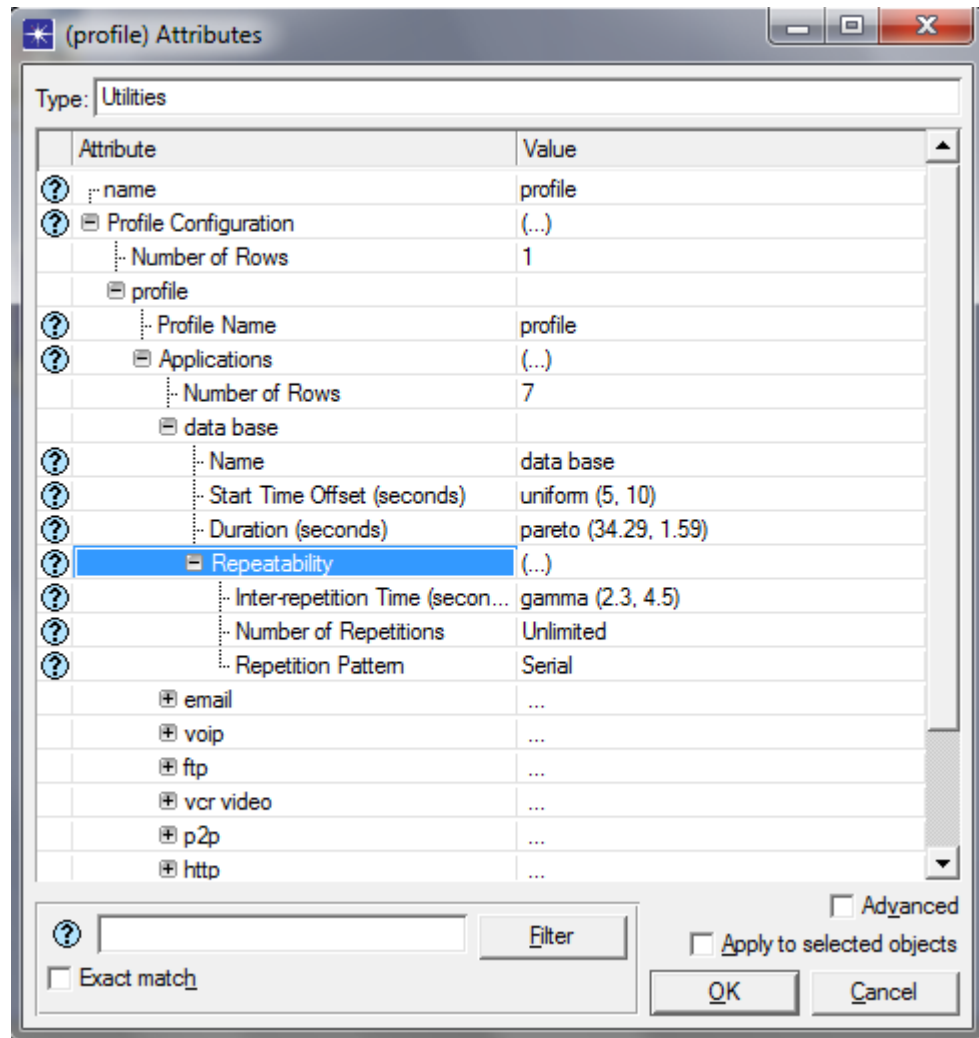
1 Rows Delete Insert Duplicate Move Up Move Down  
Details Promote  Show row labels OK Cancel

Los modelos matemáticos se seleccionaron en base a su compatibilidad con OPNET MODELER y fueron obtenidos de análisis en tráfico real a los cuales se les modeló su comportamiento. Además se buscó que fueran lo más recientes debido al rápido desarrollo de estos servicios.

#### **4.3.3.2 PROFILE DEFINITION**

Existen otros modelos matemáticos que también influyen en esta simulación como son el comportamiento de los usuarios. Estos se representan con la duración y el tiempo entre llegadas de nuevos usuarios. Todos estos modelos fueron definidos en esta aplicación (ver Figura 31).

**Figura 31 Profile attributes**



#### 4.3.3.2.1 Duración

Este modelo (ver Tabla 21) fue tomado de (10) en donde se analizó la duración de un usuario en una página y fue utilizado en OPNET para la configuración de profile definition de la variable duration para todas las aplicaciones como se ve en Figura 31.

**Tabla 21 Duración del usuario**

<b>Parámetros</b>	<b>Pareto</b>
$\mu$	34.29
A	1.14

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

#### **4.3.3.2.2 Llegada de nuevos usuarios**

Se aplicó un modelo matemático para representar el tiempo entre conexiones de los usuarios en un entorno empresarial (ver Tabla 22) (10). El cual se usó para representar la sección repearability, variable inter repetition time en profile definition para todas las aplicaciones en OPNET (ver Figura 31).

**Tabla 22 Modelo matemático para inter repetition time**

<b>Parámetros</b>	<b>Gamma</b>
A	2.3
B	4.5

Fuente: Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga (10).

Estos modelos matemáticos fueron utilizados para todos los servicios con excepción de voip, debido a que se pudo observar que este presentaba un comportamiento distinto (18).

#### **4.3.3.2.3 Duración de una llamada VoIP**

La duración de un usuario ordinario en el momento de una llamada es irregular, a veces puede ser larga y otras veces es corta. Pero también es posible que un usuario no sea ordinario, como en el caso de las llamas usadas para publicidad o para ofrecer algún producto, el cual presentara una duración corta y muy rara vez una larga (18). En nuestro caso decidimos utilizar un usuario ordinario cuyo modelo matemático fue aplicado en la configuración de profile definition para la aplicación VoIP en la variable duration ver en la Tabla 23.

**Tabla 23 Modelo matemático para duración llamada voip**

Parámetros	Exponencial
$\mu$	90

Fuente: Roberto Garcíá, Victor Garcíá, Xabiel G. Pañeda ,David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load.* (11)

#### 4.3.3.2.4 Llegada de nuevas llamadas para voip

Para este caso al igual que en la duración, se tuvo en cuenta un usuario ordinario que presenta de manera irregular estas llegadas. Esto se representó con el modelo matemático para la llegada de nuevas llamadas en (18) el cual fue aplicado en la configuración de profile definition para la aplicación VoIP en la sección repearability, variable inter repetition time ver Tabla 24 (ver Figura 32).

**Tabla 24 Modelo matemático para llegada de nuevas llamadas en voip**

Parámetros	Exponencial
$\mu$	5310

Fuente: Hyung-Jong Kim a, Myuhng Joo Kim a, Yoonjeong Kim a, Hyun Cheol Jeong b. *DEVS-Based modeling of VoIP spam callers' behavior for SPIT level calculation.*(18)

**Figura 32 Profile voip attributes**

voip	
Name	voip
Start Time Offset (seconds)	uniform (5,10)
Duration (seconds)	exponential (90)
Repeatability	(...)
Inter-repetition Time (secon...	exponential (5310)
Number of Repetitions	Unlimited
Repetition Pattern	Serial

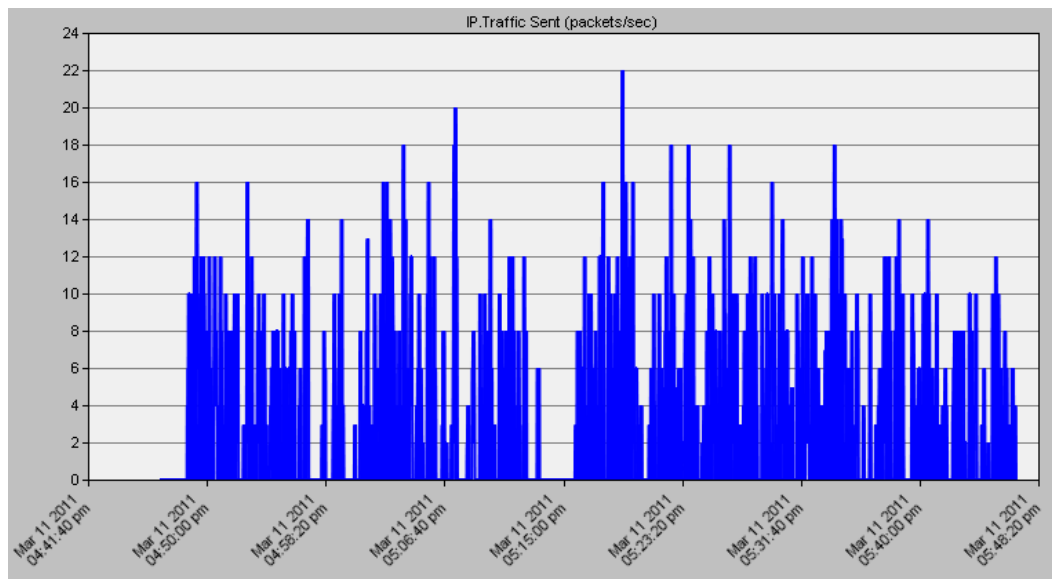
#### 4.3.4 Resultados prueba 3

En esta prueba se observaron los resultados para cada usuario con su respectivo servicio y el tráfico resultante de todos estos usuarios. Esta simulación fue hecha para un tiempo de una hora y se tomaron muestras cada segundo.

##### 4.3.4.1 RESULTADO SERVICIO VOIP

En la Figura 33 se observa que el tráfico comienza a generarse después de un tiempo de arranque, esto se debe a la configuración que trae OPNET MODELER, la cual está definida con un tiempo previo a la simulación en application definition, start time con una distribución uniforme con un parámetro de 100. Se calculó la media cuyo valor es 1,65 paquetes.

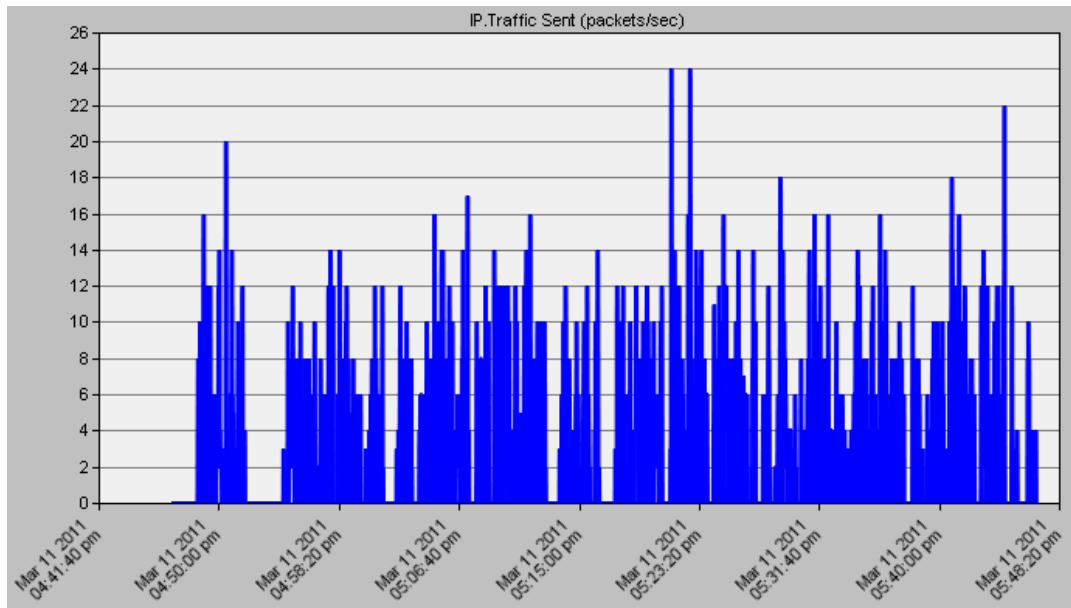
**Figura 33 Usuario voip**



#### **4.3.4.2 RESULTADO TRAFICO DE DATOS**

En la Figura 34 representa el tráfico de datos con un usuario. Se calculó la media cuyo valor es 1,72 paquetes.

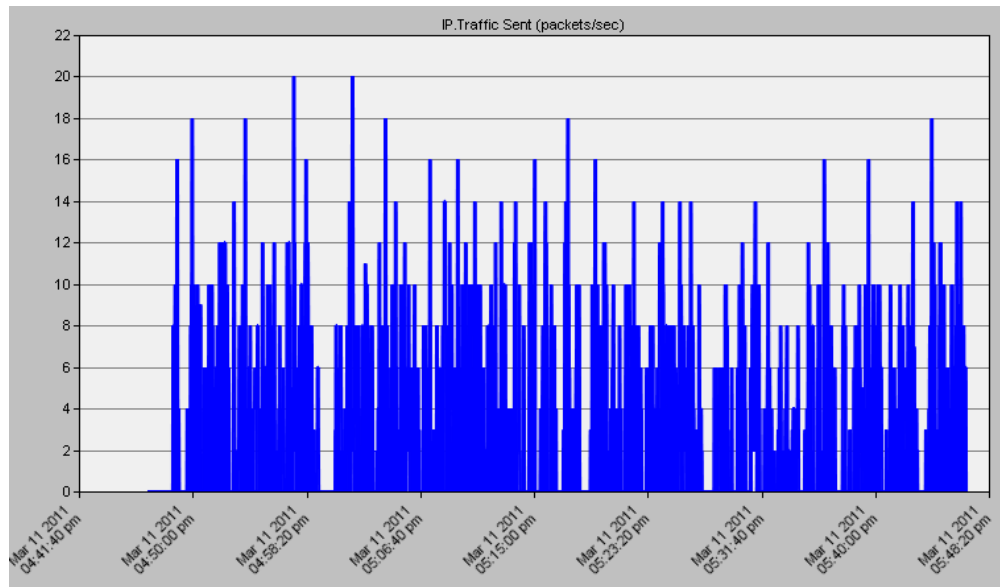
**Figura 34 Usuario database**



#### **4.3.4.3 RESULTADO TRÁFICO DE VIDEO**

En la Figura 35 representa el tráfico de video con un usuario. Se calculó la media cuyo valor es 1,71 paquetes.

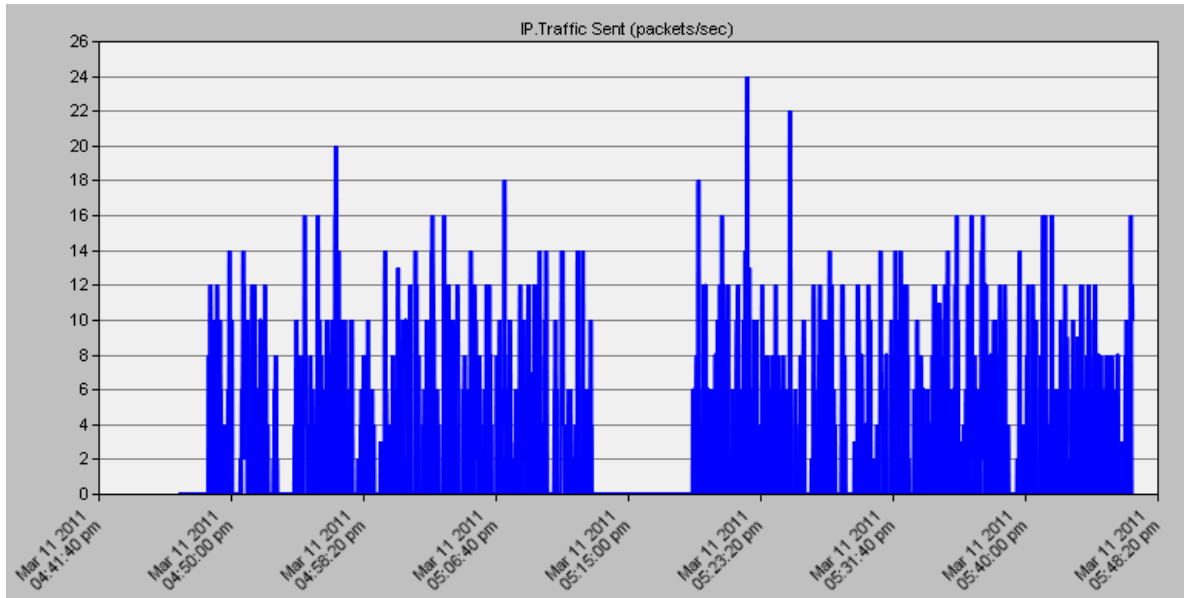
**Figura 35 Usuario video**



#### **4.3.4.4 RESULTADO PEER TO PEER**

En la Figura 36 representa el tráfico de peer to peer con un usuario. Se calculó la media cuyo valor es 1,79 paquetes. En esta simulación se puede observar de una manera notoria un tiempo sin actividad, ya que los usuarios presentan una actividad aleatoria en internet.

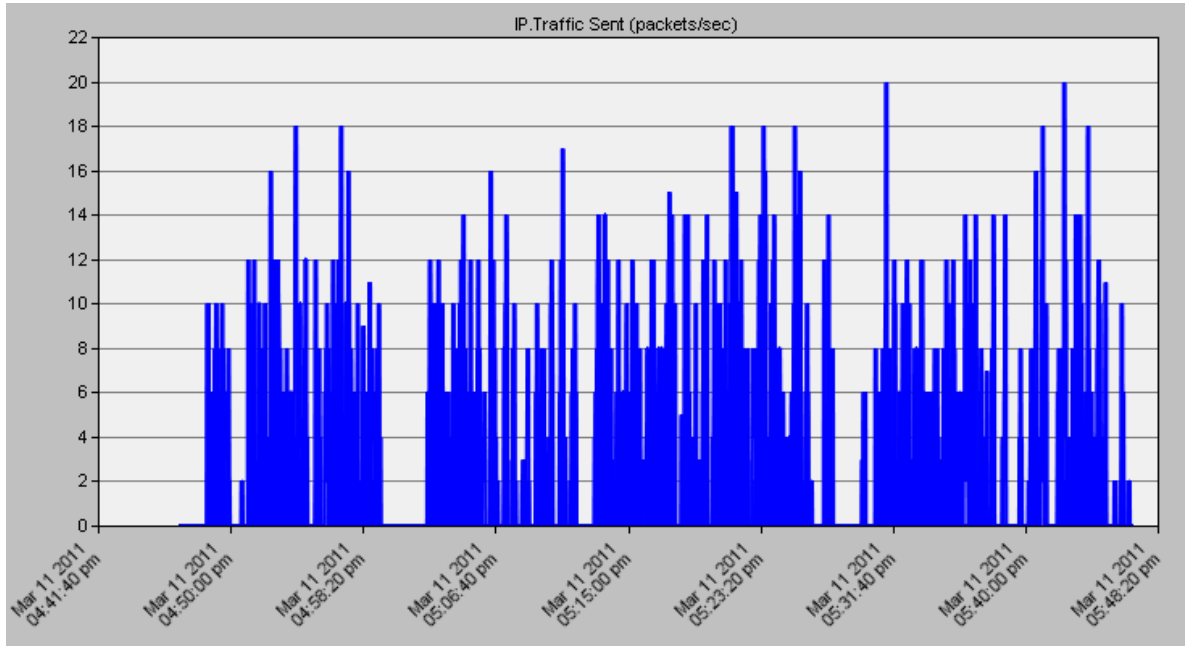
**Figura 36 Usuario peer to peer**



#### **4.3.4.5 RESULTADO TRÁFICO DE E-MAIL**

En la Figura 37 representa el tráfico email con un usuario. Se calculó la media cuyo valor es 1,69 paquetes.

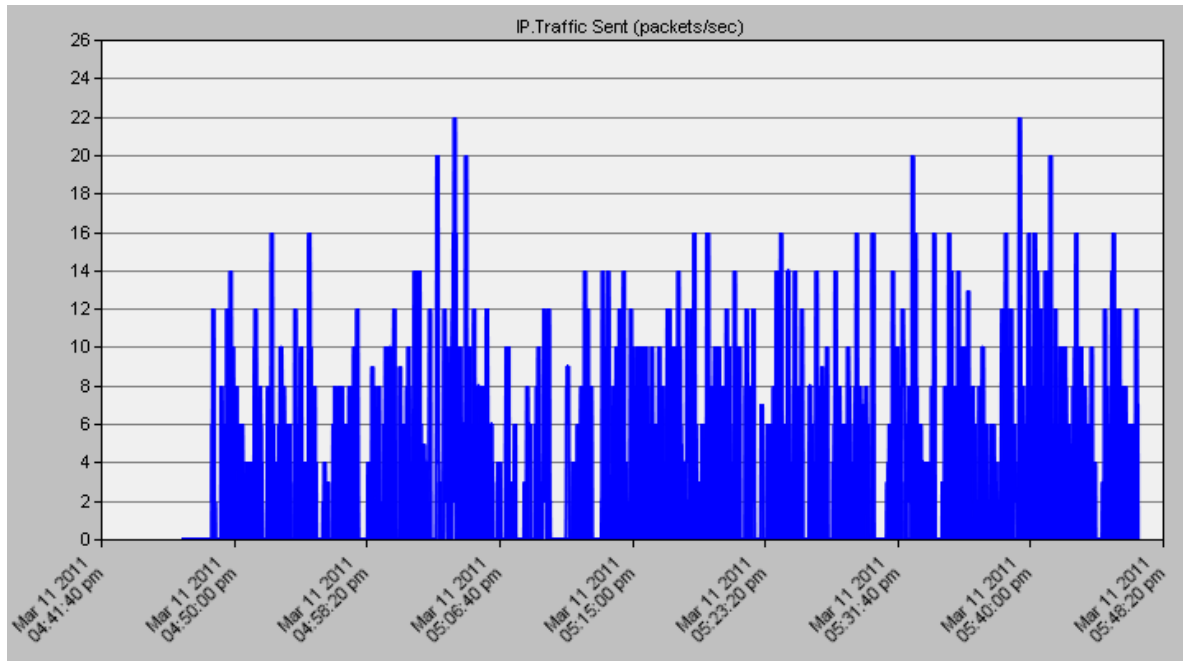
**Figura 37 Usuario email**



#### **4.3.4.6 RESULTADO FTP**

En la Figura 38 representa el tráfico FTP con un usuario. Se calculó la media cuyo valor es 1,81 paquetes.

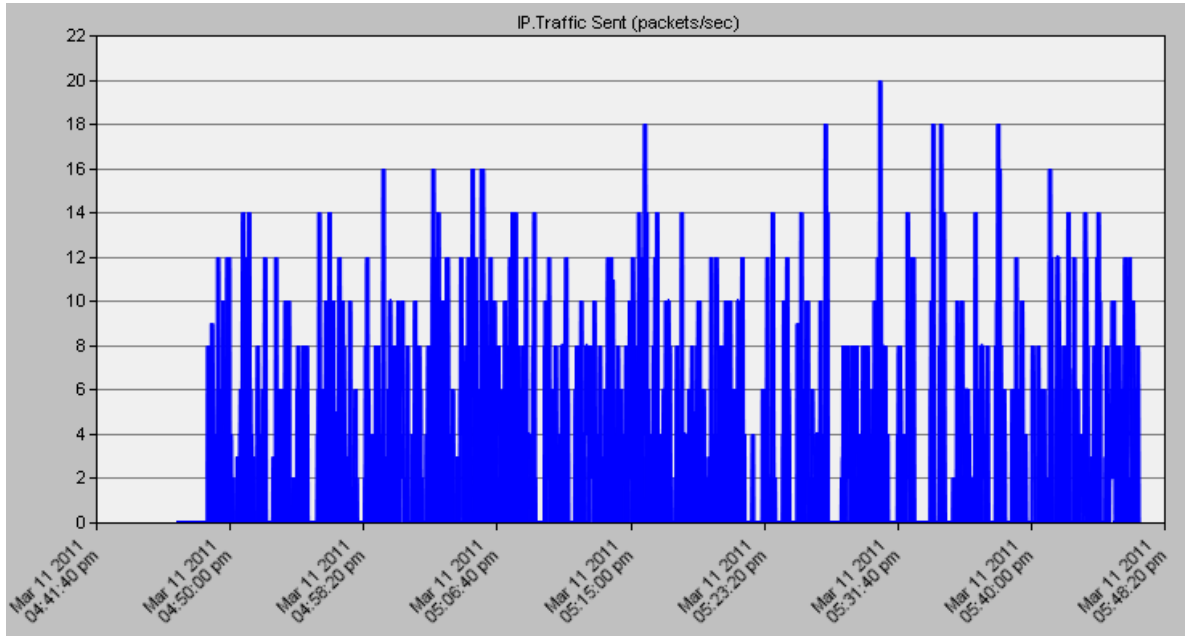
**Figura 38 Usuario FTP**



#### **4.3.4.7 RESULTADO HTTP**

En la Figura 39 representa el tráfico HTTP con un usuario. Se calculó la media cuyo valor es 1,67 paquetes.

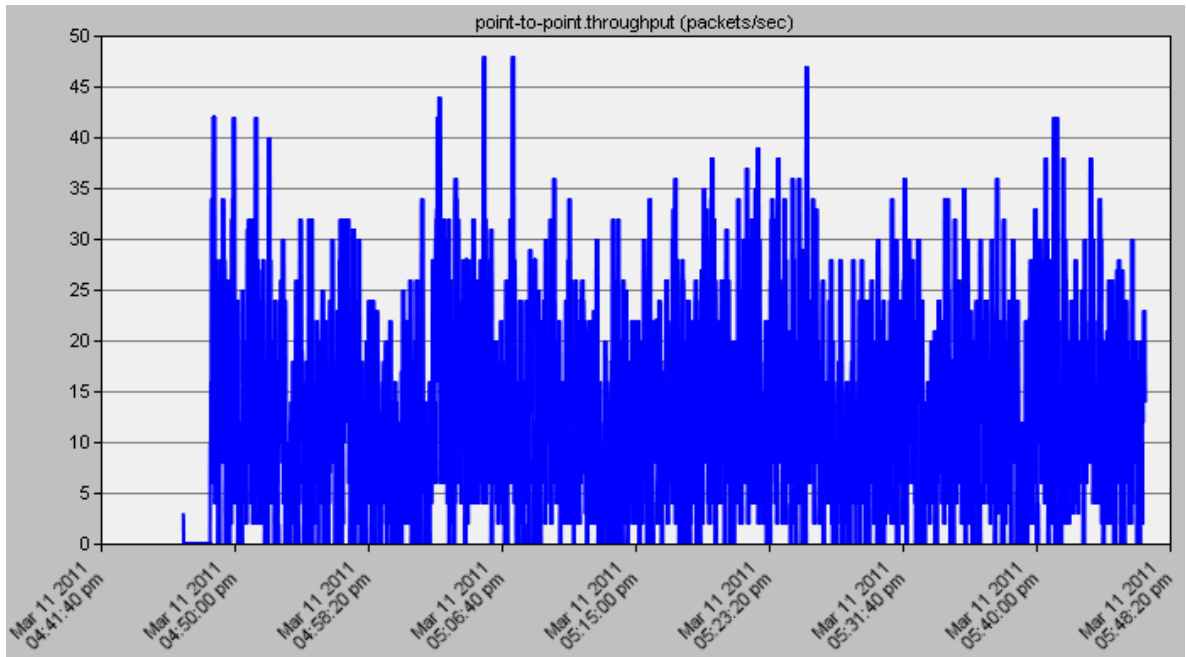
**Figura 39 Usuario HTTP**



#### **4.3.4.8 RESULTADO TOTAL DE LA PRUEBA 3**

En la Figura 40 se calculó la media cuyo valor es 12,05 paquetes. Esta simulación es el resultado de siete usuarios cada uno con una aplicación diferente.

**Figura 40 Total prueba 3**



## 4.4 SIMULACIÓN FINAL

### 4.4.1 Objetivo

- Realizar una simulación de una red con 300 usuarios, los cuales utilizan diferentes servicios de internet aleatoriamente y tienen sesiones de duración variable.

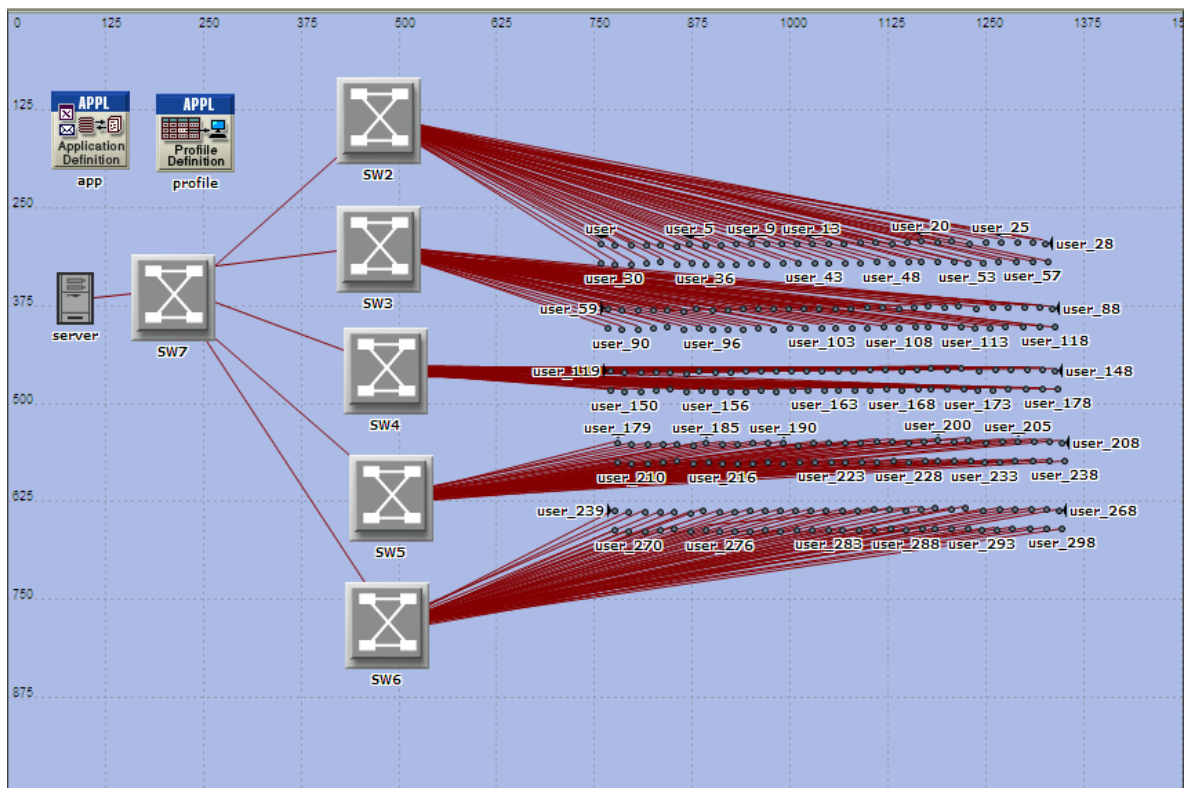
### 4.4.2 Topología final

Como se ve en la Figura 41, se implementó una topología tipo estrella en 5 switches, cada uno con 60 usuarios para así generar un total de 300 usuarios, estos switches están unidos a otro switch (sw7), el cual se encuentra conectado con el servidor (ver Figura 41). Debido al gran número de usuarios se utilizó un servidor ideal (ver Figura 10) capaz de soportar este tráfico, también se utilizaron cables de par trenzado de 100mb para realizar los enlaces (ver Figura 8) entre

usuarios- switch, 1gbps entre switch-switch y 10gbps entre switch-server. Los switch al igual que el servidor tienen un comportamiento ideal. El switch se configuró para que no tuviese calidad de servicio con el fin de que se realizara un multiplexado normal.

El tráfico fue observado en la unión entre el servidor y el switch número 7 (SW7), debido a que en esta unión circula todo el tráfico de los 300 usuarios.

**Figura 41 Esquema de la simulación.**



#### 4.4.3 Configuración de aplicaciones

La configuración que se utilizó fue la misma que se realizó en la prueba 3. A diferencia que en esta, se cargaron todas las aplicaciones en cada usuario (database, email, http, voip, ftp, video y peer to peer) y para el servidor.

#### 4.4.3.1 APPLICATION DEFINITION

En la Tabla 25 se pueden observar los diferentes modelos matemáticos investigados y su adecuación en OPNET según la aplicación.

**Tabla 25 Resumen de las distribuciones para application definition**

Aplicación	Opnet	Distribución	Parámetros
voip	Silence length	Exponencial	$\mu=0,65$
voip	talk spurt length	Exponencial	$\mu=0,352$
datos	transaction interarrival time	Lognormal	$\mu=1.22$ $\sigma=51.52$
datos	transaction size	Constante	1500
video	frame size	Weibull	$\alpha=1.144$ $\beta=11.193$
peer to peer	transaction interarrival time	Exponencial	$\mu=30$
peer to peer	transaction size	Lognormal	$\mu=82.762$ $\sigma=0.355$
email	sent interarrival time	Lognormal	$\mu=1,22$ $\sigma=51,52$
email	recive interarrival time	Lognormal	$\mu=0,64$ $\sigma=10,88$
email	recive interarrival time	Weibull	$\alpha=5,1838$ $\beta=2931.1571$
ftp	Download file size	constante	10000
ftp	transaction interarrival time	Exponencial	$\mu=30$
http	object size	constante	1500
http	number of objects	Pareto	Loc=2025,833 shape=51,430

#### 4.4.3.2 PROFILE DEFINITION

Los modelos matemáticos para esta configuración son los que podemos ver en la Tabla 26, los cuales fueron analizados con anterioridad en la prueba 3.

**Tabla 26 Resumen de las distribuciones para profile definition**

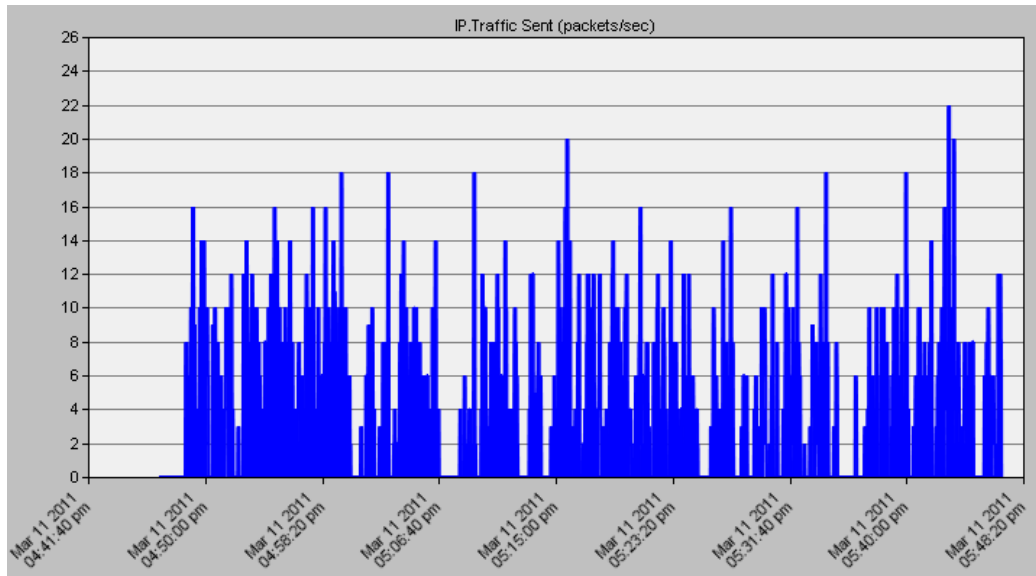
<b>Aplicación</b>	<b>Opnet</b>	<b>Distribución</b>	<b>Parámetros</b>
todas	Duración	pareto	$\mu=34.29$ $\alpha=1.59$
todas	inter repetition time	gamma	$\alpha=2,3$ $\beta=4.5$
voip	Duración	exponencial	$\mu=90$
voip	inter repetition time	exponencial	$\mu=5310$

#### 4.4.4 Resultados

Una vez finalizado el tiempo de simulación, que para este caso fue de 1 hora (3600 muestras tomadas 1 cada segundo), se procede a observar las gráficas del tráfico total obtenido, en donde se observa la tasa de bits/seg (ver Figura 44), paquetes/seg (ver Figura 43) y el queuing delay en segundos (ver Figura 45) transmitido por los 300 usuarios los cuales están usando los servicios (data base, email, http, voip, ftp, video y p2p) aleatoriamente, esto último acerca el comportamiento de la red , al de una red real.

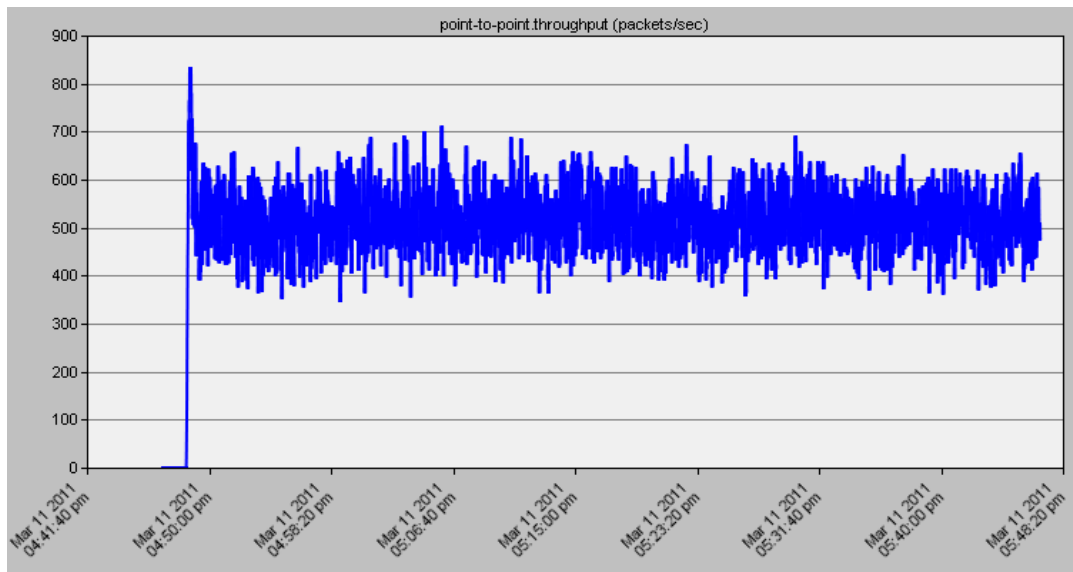
En la Figura 42 se observa el comportamiento del tráfico de internet para un usuario con todas aplicaciones usadas de manera aleatoria durante 1 hora. Nótese que el envío de tráfico inicia un tiempo después de iniciada la simulación. Esto se debe a que así se configuró la simulación en OPNET.

**Figura 42 Resultado Final para un usuario en paquetes/segundos**



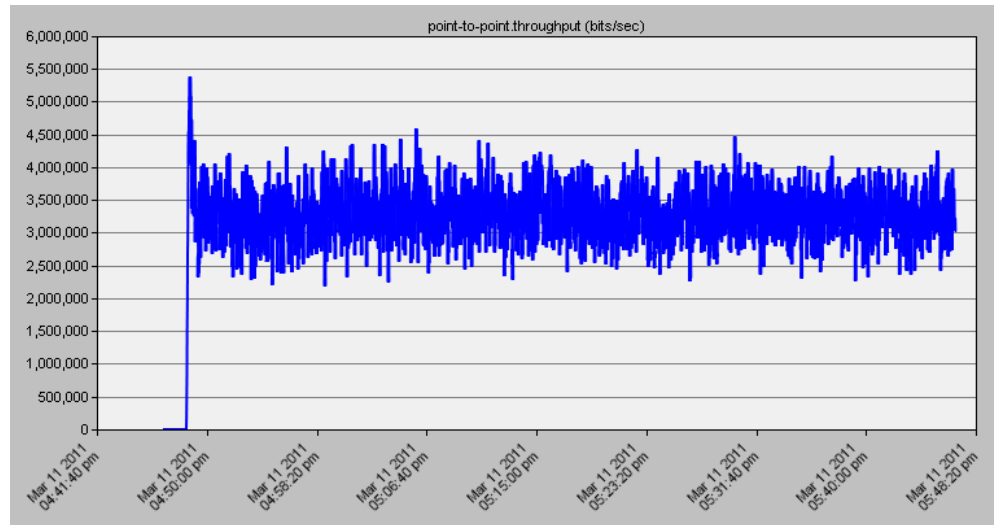
En la Figura 43 se muestra el tráfico de paquetes/seg de 300 usuarios con todas las aplicaciones. Se puede observar su aleatoriedad en el número de paquetes enviados. Además se puede ver el incremento en la media, debido a la gran cantidad de usuarios conectados a la red.

**Figura 43 Resultado Final en paquetes/segundos**

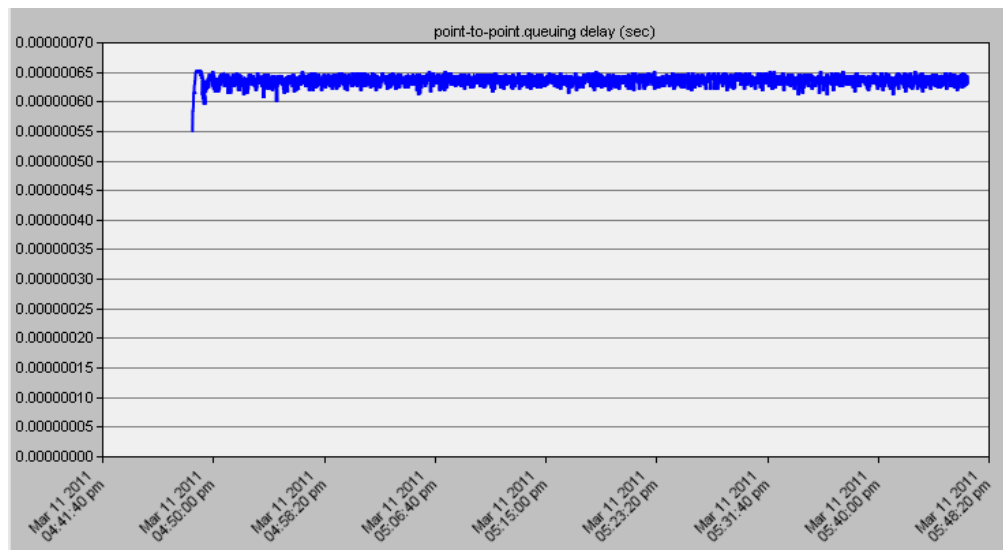


En la Figura 44 se muestra el tráfico de bits/seg de 300 usuarios con todas las aplicaciones. Se puede observar la aleatoriedad en el número de bits enviados. Además se puede ver el incremento en el ancho de banda, debido a la gran cantidad de usuarios conectados a la red.

**Figura 44 Resultado Final en bits/segundos**



**Figura 45 Resultado Final queue delay en segundos**

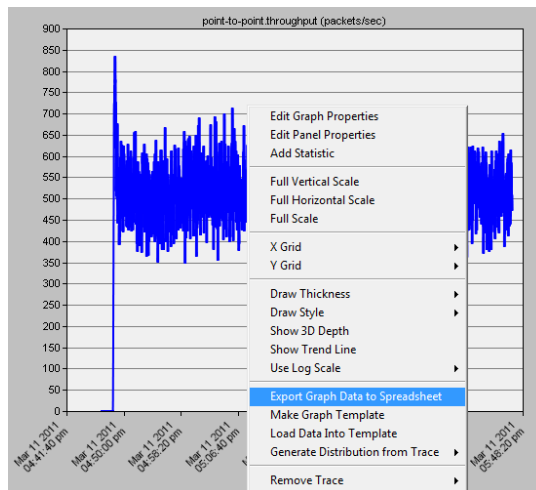


## 5 ANÁLISIS ESTADÍSTICO DE LOS RESULTADOS

### 5.1 Estimación del parámetro de hurst

Los datos de las simulaciones fueron exportados (ver Figura 46) a Excel para ser analizados, uno de esos análisis es el parámetro de hurst el cual nos indica si el trafico obtenido es autosimilar. Se utilizaron 2 métodos para probar la autosimilaridad de los resultados: el método RS y el método de varianza agregada. Estos fueron aplicados en los resultados obtenidos con 300 usuarios con todas las aplicaciones. Se aplicó a los resultados en paquetes/segundo, bits/segundo y queue delay.

Figura 46 Exportar datos a excel



#### 5.1.1 Metodo RS (19)

Para series de tiempo  $X = \{x_i, i \geq 1\}$ , con sumatoria parcial  $Y_n = \sum_{i=1}^n x_i$  y varianza  $s^2_n := \frac{1}{n} \sum_{i=1}^n x_i^2 - \frac{1}{n} y(n)^2$  la estadística R / S, o el ajuste de Hurst, está dado por:

### Ecuación 16 Ajuste de Hurst método RS

$$\frac{R}{S} n = \frac{1}{S n} \max_{0 \leq i \leq n} Y t - \frac{t}{n} Y n - \min_{0 \leq i \leq n} Y t - \frac{t}{n} Y n$$

Fuente: Murad S. Taqqu, Vadim Teverovsky, Walter Willinger. *Estimators for long-range dependence: an empirical study*. (19)

Para fractional Gaussian noise o fractional ARIMA

### Ecuación 17 fractional gaussian

$$E R/S(n) \sim C_H n^H$$

Fuente: Murad S. Taqqu, Vadim Teverovsky, Walter Willinger. *Estimators for long-range dependence: an empirical study*. (19)

Como  $n \rightarrow \infty$ . Donde  $C_H$  es otro positivo, constante finito, no dependiente de  $n$ . Para determinar  $H$  mediante la estadística de  $R / S$ , se procede de la siguiente manera. Para series de tiempo de longitud  $N$ , subdividir las series en  $K$  bloques, cada uno de tamaño  $N / K$ . Entonces, para cada intervalo  $n$ , calcular  $R(k_i, n) / S(k_i, n)$ , Comenzando por los puntos  $k_i = \frac{iN}{k} + 1, i = 1, 2, \dots$ , de tal manera que  $k_i + n \leq N$ . Para valores de  $n$  más pequeños que  $N / K$ . Se tiene  $k$  diferentes estimaciones de  $R(n) / S(n)$ . Para valores de  $n$  acercándose a  $N$ , se tienen menos valores, tan sólo 1 cuando  $n \geq N - N/K$ .

Seleccionando valores de  $n$  logarítmicamente espaciados. Graficando  $\log[R(k_i, n) / S(k_i, n)]$  versus  $\log n$  y obtener, para cada  $n$ , varios puntos de la gráfica. Esta grafica es llamada el "Pox plot" para para la estadística  $R/S$ . El parámetro  $H$  se puede estimar mediante la instalación de una línea a los puntos en el "Pox plot". Ya que cualquier dependencia de corto alcance en la serie, suele dar lugar a una zona transitoria en el extremo inferior de la gráfica, elija un punto de corte, y no utilice el extremo inferior de la gráfica para la estimación de  $H$ . Por lo general, el extremo más alto de la gráfica tampoco se usa, porque hay pocos puntos hacer estimaciones fiables. Los valores de  $n$  que se encuentran entre el corte inferior y superior de los puntos se utilizan para calcular  $H$ .

## Programa para calcular el coeficiente de hurst con el método RS

```
function H = RS(sequence, isplot)
%
% 'RS' estimate the hurst parameter of a given sequence with R/S method.
%
% Inputs:
%   sequence: the input sequence for estimate
%   isplot: whether display the plot. without a plot if isplot equal to
0
% Outputs:
%   H: the estimated hurst coeffeient of the input sequence

% Author: Chu Chen
% Version 1.0, 03/10/2008
% chen-chu@163.com
%

if nargin == 1
    isplot = 0;
end

N = length(sequence);
dlarge = floor(N/5);
dsmall = max(10, log10(N)^2);
D = floor(logspace(log10(dsmall), log10(dlarge), 50));
D = unique(D);
n = length(D);
x = zeros(1, n);
y = zeros(1, n);

R = cell(1, n);
S = cell(1, n);
for i = 1:n
    d = D(i);
    m = floor(N/d);
    R{i} = zeros(1, m);
    S{i} = zeros(1, m);
    matrix_sequence = reshape(sequence(1:d*m), d, m);

    Z1 = cumsum(matrix_sequence);
    Z2 = cumsum(repmat(mean(matrix_sequence), d, 1));
    R{i} = (max(Z1-Z2) - min(Z1-Z2));
    S{i} = std(matrix_sequence);

    if min(R{i}) == 0 || min(S{i}) == 0
```

```

        continue;
    end

    x(i) = log10(d);
    y(i) = mean(log10(R{i}./S{i}));
end

% fit a line with middle part of sequence
index = x~=0;
x = x(index);
y = y(index);
n2 = length(x);
cut_min = ceil(3*n2/10);
cut_max = floor(9*n2/10);

X = x(cut_min:cut_max);
Y = y(cut_min:cut_max);
p1 = polyfit(X,Y,1);
Yfit = polyval(p1,X);
H = (Yfit(end)-Yfit(1))/(X(end)-X(1));

if isplot ~= 0
    figure,hold on;
    bound = ceil(log10(N));
    axis([0 bound 0 0.75*bound]);

    temp = (1:n).*index;
    index = temp(index);
    for i = 1:n2
        plot(x(i),log10(R{index(i)}./S{index(i)}),'b. ');
    end

    x = linspace(0,bound,10);
    y1 = 0.5*x;
    y2 = x;
    h1 = plot(x,y1,'b--','LineWidth',2);
    h2 = plot(x,y2,'b-.','LineWidth',2);
    plot(X,Yfit,'r-','LineWidth',3);
    legend([h1,h2], 'slope 1/2', 'slope 1', 4)
    xlabel('log10(blocks of size m)'),ylabel('log10(R/S)'),title('R/S
Method');
end

```

Fuente: <http://www.mathworks.com/matlabcentral/fileexchange/19148-hurst-parameter-estimate> (20)

### 5.1.1.1 Resultado para paquetes/segundos

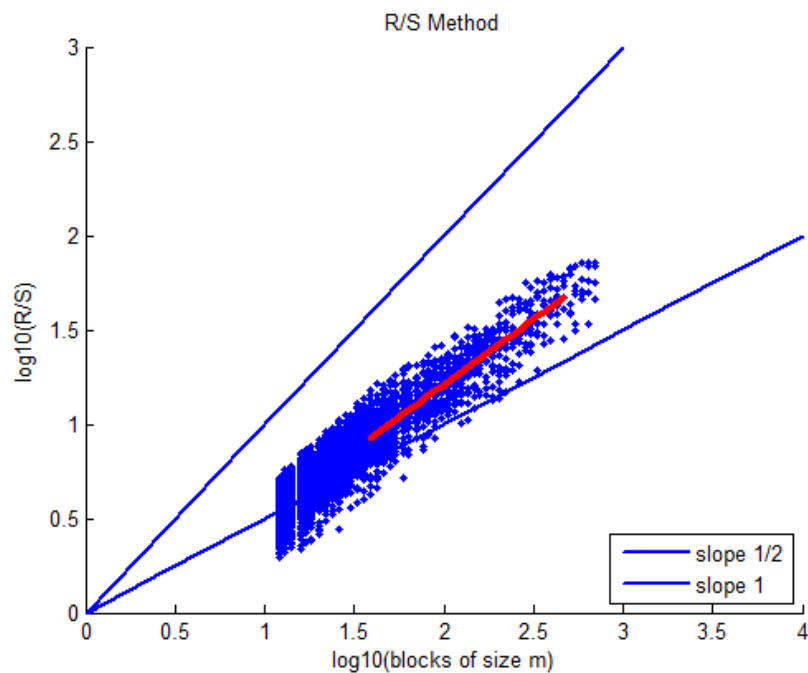
```
>> H = RS(paquetes,1)
```

```
H =
```

```
0.6895
```

Como se observa en la Figura 47 los datos de paquetes/segundos se encuentran dentro del rango de confianza para decir que son autosimilares con un coeficiente de hurst de 0.6895. Como el parámetro de Hurst es mayor que 0,5 se puede concluir que hay autosimilaridad.

**Figura 47** Parámetro de hurst método RS para paquetes/segundos



### 5.1.1.2 Resultado para bits/segundos

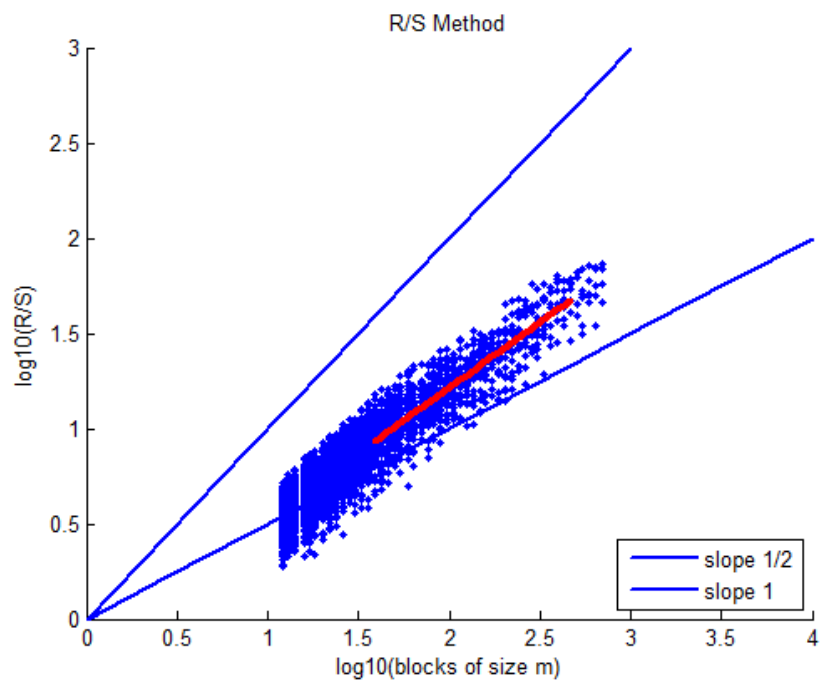
```
>> H = RS(bits,1)
```

```
H =
```

```
0.6872
```

Tanto paquetes/segundos como bits/segundos se encuentran relacionados por esto en la Figura 48 se observa que los resultados de bits/segundos también se encuentran en el rango de confianza para decir que son autosimilares con un coeficiente de hurst de 0.6872.

**Figura 48** Parámetro de hurst método RS para bits/segundos



### 5.1.1.3 Resultado para queue delay

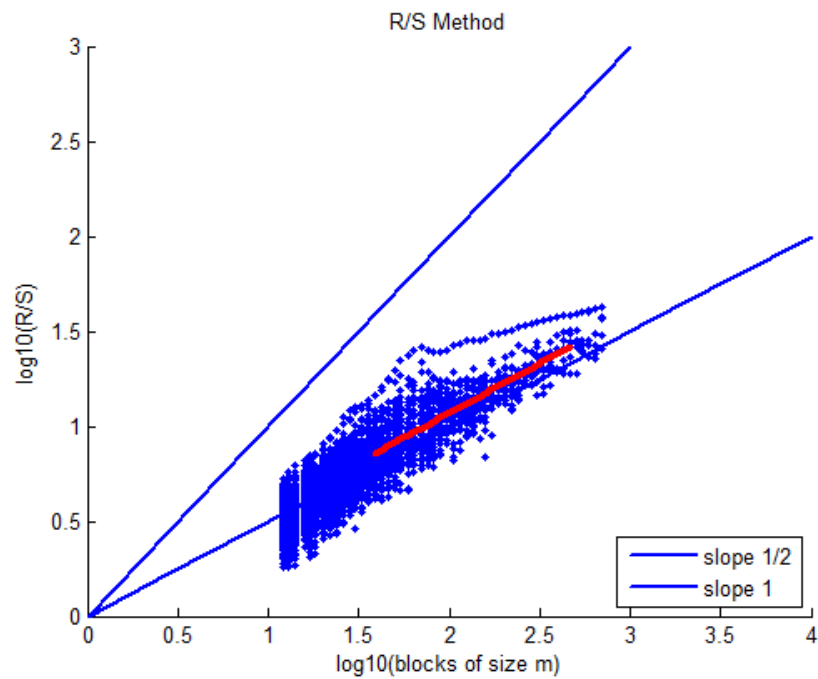
```
>> H = RS(queue,1)
```

```
H =
```

```
0.5205
```

En la Figura 49 se observa que los resultados de queue delay que se encuentran en el límite dentro del rango, con un coeficiente de hurst de 0.5205.

**Figura 49** Parámetro de hurst método RS para queue dealy



### 5.1.2 Método de varianza agregada (19)

Consiste en dividir las series de tiempo originales  $X = \{x_i, i \geq 1\}$  en bloques de tamaño  $m$  y media dentro de cada bloque, esto se consideran las series agregadas

#### Ecuación 18 Series agregadas

$$X^{(m)}_k = \frac{1}{m} \sum_{i=k-1}^{km} X_i \quad k = 1, 2, \dots$$

Fuente: Murad S. Taqqu, Vadim Teverovsky, Walter Willinger. *Estimators for long-range dependence: an empirical study*. (19)

Para valores sucesivos de  $m$ . El índice  $k$ , etiqueta el bloque. Después tomar la muestra de variación de  $x^{(m)}_k$ ,  $k = 1, 2, \dots$  dentro de cada bloque. Esta muestra de variación es un estimador de  $VarX^{(m)}$ . Ya que por el fractional gaussian noise y el fractional ARIMA,  $VarX^{(m)} \sim \sigma_0^2 m^\beta$  como  $m \rightarrow \infty$  donde  $\beta = 2H - 2 < 0$

Podemos obtener una estimación de  $\beta$  o  $H$ , procediendo de la siguiente manera.

Para un  $m$  dado, divide los datos  $X_1, \dots, X_n$ , en  $N/m$  bloques de tamaño  $m$ , calcular  $x^{(m)}_k$ ,  $k = 1, 2, \dots, N/m$  y la varianza de la muestra.

#### Ecuación 19 Varianza de la muestra

$$\overline{VarX^{(m)}} = \frac{1}{N/m} \sum_{k=1}^{N/m} x^{(m)}_k{}^2 - \left( \frac{1}{N/m} \sum_{k=1}^{N/m} x^{(m)}_k \right)^2$$

Fuente: Murad S. Taqqu, Vadim Teverovsky, Walter Willinger. *Estimators for long-range dependence: an empirical study*. (19)

Repita este procedimiento para diferentes valores de  $m$  y grafique el logaritmo de la varianza de la muestra frente al log de  $m$ . Seleccione los valores de  $m$ ,  $m_i, i \geq 1$ , que son equidistantes en escala logarítmica, de modo que  $m_{i+1}/m_i = C$ , donde  $C$  es una constante que depende de la longitud de la serie y el número deseado de puntos.

Desde que ya que  $\overline{Var}X^{(m)}$  es una estimación de  $Var X^{(m)}$ , los puntos resultantes deben formar una línea recta con pendiente  $\beta = 2H - 2, -1 \leq \beta < 0$ . En la práctica, la pendiente se calcula mediante la instalación de una línea a los puntos obtenidos a partir de la trama. Se supone que tanto M y N son grandes, y que  $m \ll N$ , de modo que la longitud de cada bloque y el número de bloques es grande. Si X (es de corto alcance o) no dependencia, la pendiente obtenida debe ser igual a -1.

### Programa para calcular el coeficiente de hurst con el método de varianza agregada

```
function H = aggvar(sequence, isplot)
%
% 'aggvar' estimate the hurst parameter of a given sequence with
% aggregate variance method.
%
% Inputs:
%   sequence: the input sequence for estimate
%   isplot: whether display the plot. without a plot if isplot equal to
%   0
% Outputs:
%   H: the estimated hurst coeffeient of the input sequence
% Author: Chu Chen
% Version 1.0, 03/10/2008
% chen-chu@163.com
%

if nargin == 1
    isplot = 0;
end

N = length(sequence);
mlarge = floor(N/5);
M = [floor(logspace(0, log10(mlarge), 50))];
M = unique(M(M>1));
n = length(M);
cut_min = ceil(n/10);
cut_max = floor(6*n/10);

V = zeros(1, n);
for i = 1:n
    m = M(i);
    k = floor(N/m);
```

```

    matrix_sequence = reshape(sequence(1:m*k),m,k);
    V(i) = var(sum(matrix_sequence,1)/m);
end

x = log10(M);
y = log10(V);
y1 = -x+y(1)+x(1);
X = x(cut_min:cut_max);
Y = y(cut_min:cut_max);
p1 = polyfit(X,Y,1);
Yfit = polyval(p1,X);
yfit = polyval(p1,x);
beta = -(Yfit(end)-Yfit(1))/(X(end)-X(1));
H = 1-beta/2;

if isplot ~= 0
    figure,hold on;
    plot(x,y,'b*');
    h = plot(x,y1);
    plot(X,Yfit,'r-','LineWidth',2);
    plot(x(1:cut_min),yfit(1:cut_min),'r:','LineWidth',2);
    plot(x(cut_max:end),yfit(cut_max:end),'r:','LineWidth',2);
    xlabel('log10(Aggregate Level)'), ylabel('log10(Variance)'), title('Time
Variance Method');
end

```

Fuente: <http://www.mathworks.com/matlabcentral/fileexchange/19148-hurst-parameter-estimate> (20)

### 5.1.2.1 Resultado para paquetes/segundos

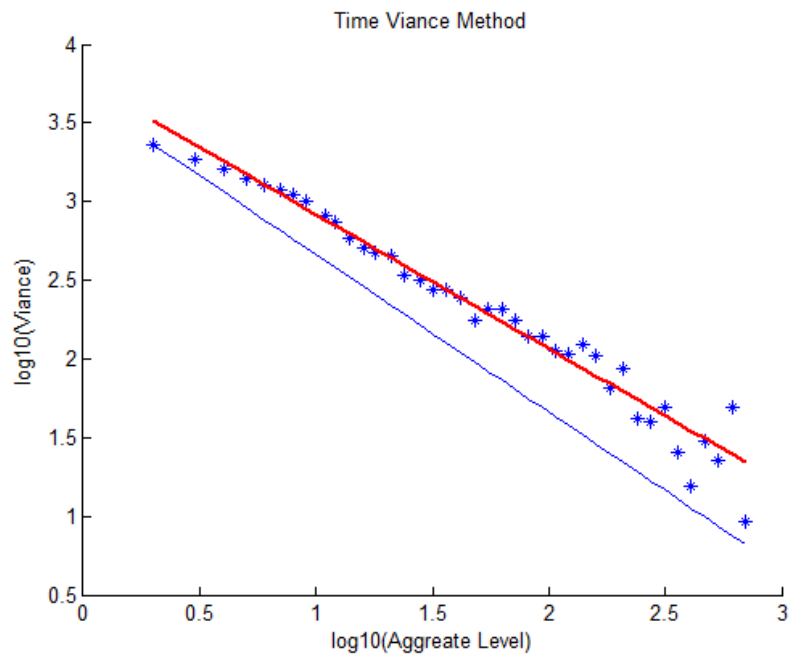
```
>> H = aggvar(paquetes,1)
```

```
H =
```

```
0.5739
```

En la Figura 50 se observa que los datos (paquetes/segundos) tienden a una línea recta por esto podemos decir que el resultado es autosimilar con un coeficiente de hurst de 0.5739.

**Figura 50** Parámetro de hurst método varianza agregada para paquetes/segundos



### 5.1.2.2 Resultado para bits/segundos

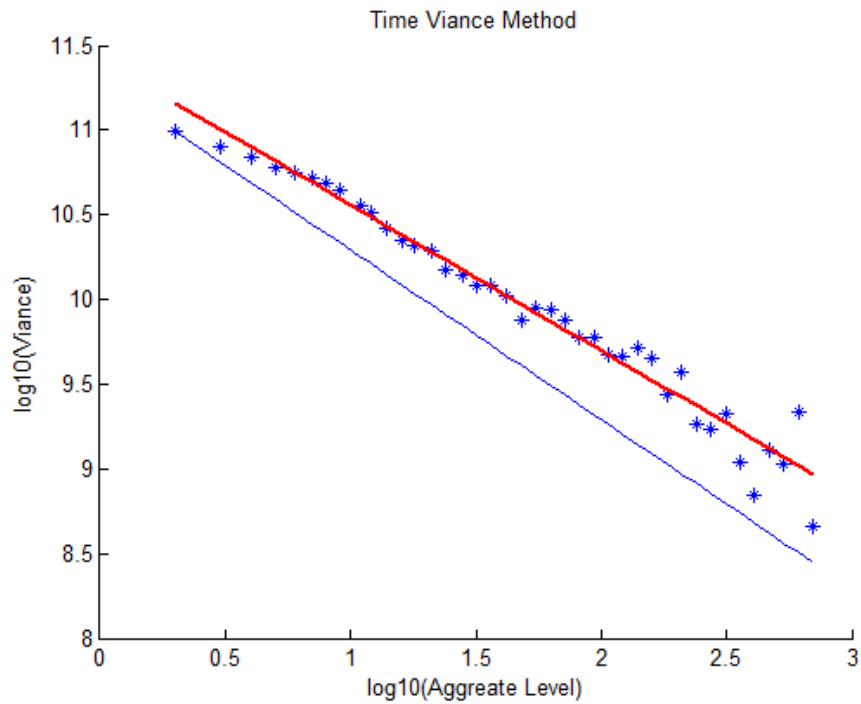
```
>> H = aggvar(bits,1)
```

```
H =
```

```
0.5698
```

En la Figura 51 debido a la relación directa que existe entre paquetes/segundos y bits/segundos, se observa que los datos tienden a una línea recta en ambos casos por esto podemos decir que el resultado es autosimilar en bits/segundos, con un coeficiente de hurst de 0.5698.

**Figura 51** Parámetro de hurst método varianza agregada para bits/segundos



### **5.1.2.3 Resultado para queue delay**

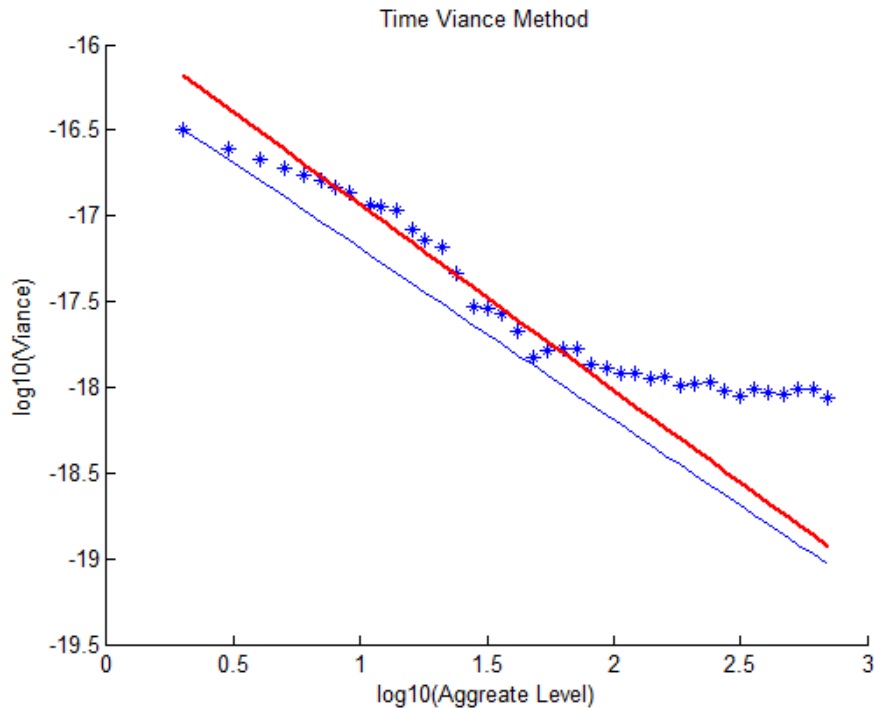
```
>> H = aggvar(queue,1)
```

```
H =
```

```
0.4602
```

En la Figura 52 se observa que los datos de queue delay se encuentran dispersos con respecto a la recta lo que nos dice que estos resultados no son autosimilares, además presentan un coeficiente de hurst menor de 0.5 como es 0.4602.

**Figura 52** Parámetro de hurst método varianza agregada para queue delay



## 5.2 APLICACIÓN DE DISTRIBUCIONES A LOS RESULTADOS

Se utilizó el software easyfit para aplicar diferentes distribuciones a los resultados obtenidos y aplicar 3 tipos de prueba de bondad como son Kolmogorov Smirnov, Anderson Darling y Chi-cuadrado. Para tener una mayor confiabilidad del software se aplicaron 2 distribuciones por nosotros y fueron comparadas con el software easyfit. Para verificar si el programa Easyfit estaba realizando bien la prueba de bondad de ajuste, se procedió a hacer una prueba en Excel con algunas distribuciones.

### 5.2.1 Distribución Gamma

Se caculo la varianza y la media de los datos en Excel usando los comandos promedio y var.

### 5.2.1.1 Usando Excel en datos paquetes/segundos

**Tabla 27 media y varianza para paquetes/segundos**

<b>Varianza</b>	3346,69729
<b>Media</b>	513,980544

Remplazando la media y la varianza en Ecuación 4 Media y varianza de la distribución gamma, resolvemos el sistema de ecuaciones para obtener alfa y beta.

**Tabla 28 Parámetros distribución gamma obtenidos manualmente**

<b>Beta</b>	6,51133069
<b>Alfa</b>	78,9363294

### 5.2.1.2 Usando Easyfit en datos paquetes/segundos

En Easyfit se introdujeron los datos de esta simulación (paquetes/segundos) y se corrió el programa según las distribuciones que deseamos aplicar, en este caso gamma, cuyos resultados son los que se observan en la Tabla 29 y la

Tabla 30.

**Tabla 29 media y varianza obtenidos en easyfit**

<b>varianza</b>	3346,7
<b>Media</b>	513,98

**Tabla 30 Parámetros distribución gamma obtenidos con easyfit**

<b>Beta</b>	6,5113
<b>Alfa</b>	78,936

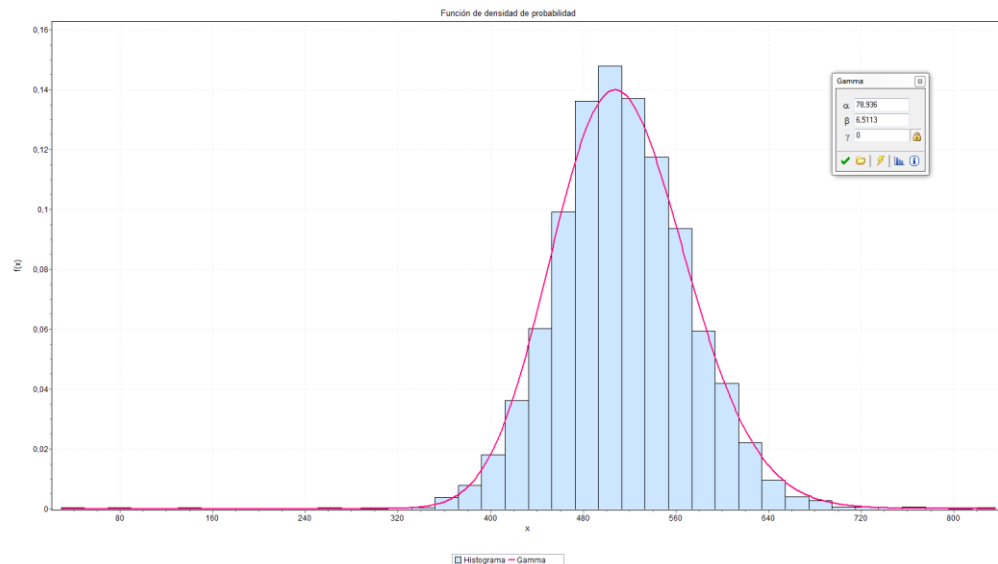
Se puede observar que los resultados de la media y la varianza obtenidos en Excel y easyfit son iguales como se ve en las tablas Tabla 27 y Tabla 29. Al igual

con el alfa y el beta de la distribución gamma, se obtienen los mismos resultados resolviendo el sistema de ecuaciones ver Tabla 28 y

Tabla 30.

En la Figura 53 se puede hacer un análisis visual del comportamiento de la distribución gamma, para los datos obtenidos en bits/segundos, los cuales son representados en el histograma. Se puede observar que el comportamiento de esta distribución se ajusta bastante al comportamiento de los resultados en la simulación.

**Figura 53 Histograma y distribución gamma con easyfit para paquetes/segundos**



## 5.2.2 Distribución lognormal

Se calculó la varianza y la media de los datos en Excel usando los comandos promedio y var.

### 5.2.2.1 Usando Excel en datos bits/segundos

**Tabla 31 media y varianza para bits/segundos**

<b>Varianza</b>	1,4122E+11
<b>Media</b>	3271328,78

Remplazando la media y la varianza en la Ecuación 12 Media y la varianza de la distribución logarítmica normal, resolvemos el sistema de ecuaciones para obtener  $\sigma$  y  $\mu$ .

**Tabla 32 Parámetros distribución lognormal obtenidos manualmente**

<b><math>\Sigma</math></b>	0.13
<b><math>\mu</math></b>	14.99

### 5.2.2.2 Usando Easyfit en datos bits/segundos

En easyfit se introdujeron los datos de esta simulación (bits/segundos) y se corrió el programa según las distribuciones que deseamos aplicar, en este caso lognormal, cuyos resultados son los que se observan en la Tabla 33 y la Tabla 34.

**Tabla 33 media y varianza obtenidos en easyfit**

<b>varianza</b>	1,41E+11
<b>Media</b>	3271328,8

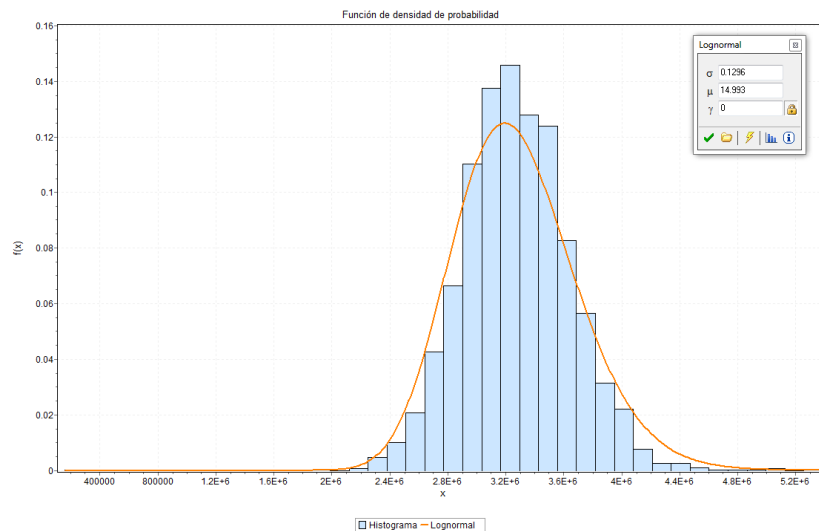
**Tabla 34 Parámetros distribución lognormal obtenidos con easyfit**

<b><math>\Sigma</math></b>	0.1296
<b><math>\mu</math></b>	14.993

Se puede observar que los resultados de la media y la varianza obtenidos en Excel y easyfit son iguales como se ve en las tablas Tabla 31 y Tabla 33. Al igual con el  $\sigma$  y  $\mu$  de la distribución lognormal, se obtienen los mismos resultados resolviendo el sistema de ecuaciones ver Tabla 32 y Tabla 34.

En la Figura 54 se puede hacer un análisis visual del comportamiento de la distribución lognormal, para los datos obtenidos en bits/segundos, los cuales son representados en el histograma. En este caso se observa que esta distribución no se ajusta a los resultados.

**Figura 54 Histograma y distribución lognormal con easyfit para bits/segundos**



### 5.2.3 Resultados de las pruebas de bondad de ajuste

Una vez verificados los resultados de las pruebas de bondad de ajuste en el easyfit, se procedió a hacer análisis de las pruebas de bondad de ajuste para diferentes distribuciones de probabilidad con diferentes variables de los resultados obtenidos en las simulaciones.

Un aspecto importante para determinar si una distribución se aproxima es la confianza (que se representa mediante el carácter alfa). Se escogió un alfa de 0.05 para las pruebas, debido a que entre mayor sea el alfa menor será el

intervalo de confianza de los resultados obtenidos con la distribución aplicada con respecto a los resultados de la simulación.

En la Tabla 35 se puede observar las distribuciones que aprobaron 2 o más de las 3 pruebas de bondad usadas con un alfa de 0.05, para los datos de paquetes/segundos con 300 usuarios.

**Tabla 35 Pruebas para Paquetes/segundos alfa= 0.05**

<b>Distribucion</b>	<b>Kolmogorov-smirnov</b>	<b>Anderson-Darling</b>	<b>Chi-cuadrado</b>
Gamma	✓	✓	✓
Gen. Logistic	✓	✓	
Log-Logistic (3P)	✓	✓	
Nakagami	✓	✓	✓

**Tabla 36 Parámetros de las distribuciones para paquetes/segundos**

<b>Distribución</b>	<b>parámetros</b>
Gamma	$\alpha = 78.936$ $\beta = 6.5113$
Gen. Logistic	$k = 0.02113$ $\sigma = 31.842$ $\mu = 512.87$
Log-Logistic (3P)	$\alpha = 111.87$ $\beta = 3587.2$ $\gamma = -3074.0$
Nakagami	$M = 20.146$ $\Omega = 2.6752E+5$

En la Tabla 37 se puede observar las distribuciones que aprobaron 2 de las 3 pruebas de bondad usadas con un alfa de 0.05, para los datos de bits/segundos con 300 usuarios. Se puede observar que en ningún caso la prueba de bondad chi cuadrada fue aprobada para este alfa.

**Tabla 37 Pruebas para bits/segundos alfa= 0.05**

Distribucion	Kolmogorov-smirnov	Anderson-Darling
Gamma	✓	✓
Log-Logistic (3P)	✓	✓
Nakagami	✓	✓

**Tabla 38 Parámetros de las distribuciones para bits/segundos**

Distribución	Parámetros
Gamma	$\alpha = 75.782$ $\beta = 43168.0$
Log-Logistic (3P)	$\alpha = 110.21$ $\beta = 2.2956E+7$ $\gamma = -1.9690E+7$
Nakagami	$M = 19.338$ $\Omega = 1.0843E+13$

En la Tabla 39 se puede observar las distribuciones que aprobaron 1 de las 3 pruebas de bondad usadas con un alfa de 0.05, para los datos de queue delay con 300 usuarios. Se puede observar que en ningún caso la prueba de bondad chi cuadrada y Anderson-darling fueron aprobadas para este alfa.

**Tabla 39 Pruebas para queue delay alfa= 0.05**

Distribucion	Kolmogorov-smirnov
Beta	✓
Gen. Extreme Value	✓
Gen. Logistic	✓
Weibull (3P)	✓

**Tabla 40 Parámetros de las distribuciones para queue delay**

<b>Distribución</b>	<b>Parámetros</b>
Beta	$\alpha 1 = 2.3644E+7$ $\alpha 2 = 6.7726$ $a = -0.06758$ $b = 6.5560E-7$
Gen. Extreme Value	$k = -0.50153$ $\sigma = 7.6601E-9$ $\mu = 6.3456E-7$
Gen. Logistic	$k = -0.11473$ $\sigma = 3.8894E-9$ $\mu = 6.3704E-7$
Weibull (3P)	$\alpha = 25.819$ $\beta = 1.5829E-7$ $\gamma = 4.8124E-7$

En la Tabla 36, Tabla 38 y Tabla 40 se tienen los parámetros de las distribuciones que aprobaron las pruebas de bondad de ajuste, para cada uno de los resultados paquetes/segundos, bits/segundos y queue delay.

## CONCLUSIONES Y TRABAJOS FUTUROS

Se desarrolló una simulación del tráfico en una red IP, donde se utilizaron aplicaciones como son VoIP, video, FTP, email, datos, http y peer to peer, todo con el fin de ser utilizada en futuros proyectos que busquen producir mejores equipos o protocolos con los que se pueda mejorar el servicio dado en una red, para que esta pueda manejar una mayor velocidad con un número elevado de usuarios y una mejor calidad.

Se investigaron varios artículos donde se encontraron diferentes modelos matemáticos, realizados con el fin de analizar el comportamiento de las aplicaciones que fueron usadas en la simulación, se utilizó el resultado de aquellos modelos matemáticos para representar cada una de las aplicaciones en OPNET MODELER.

Se realizaron diferentes pruebas, con el fin de realizar un análisis independiente de cada una de las aplicaciones y sus respectivas configuraciones, para que estas brindaran el resultado esperado en este proyecto.

Se calculó el coeficiente de hurst en los resultados obtenidos (paquetes/segundos con un valor de 0.6895, bits/segundos con un valor de 0.6872 y queue delay con un valor de 0.5205 con el método RS y paquetes/segundos con un valor de 0.5739, bits/segundos con un valor de 0.5698 y queue delay con un valor de 0.4602 con el método varianza agregada), demostrando autosimilaridad en las variables paquetes/segundos, bits/segundos y queue delay con el método RS.

Se realizó un análisis estadístico donde se obtuvieron posibles distribuciones que pueden llegar a representar los resultados obtenidos, según las pruebas de bondad de ajuste realizadas (Kolmogorov Smirnov, Anderson Darling y Chi-cuadrado). Las distribuciones que más se ajustaron al comportamiento del tráfico en paquetes/segundos son la Gamma, Gen. Logistic, Log-Logistic (3P) y Nakagami. Para el caso de bits/segundos las distribuciones que más se ajustaron fueron la Gamma, Log-Logistic (3P) y Nakagami. Y en queue delay las distribuciones que más se ajustan son la Beta, Gen. Extreme Value, Gen. Logistic y Weibull (3P).

Dentro de los trabajos futuros y que están fuera de los objetivos planteados inicialmente para el desarrollo de este trabajo de grado, se proponen:

- Renovar el comportamiento de tráfico usado y agregar nuevas aplicaciones que puedan aparecer con el tiempo debido al constante desarrollo de las redes IP.

## BIBLIOGRAFÍA

1. José María Barceló Ordinas, Jordi Íñigo Griera, Ramon Martí Escalé, Enric Peig Olivé y Xavier Perramon Tornil. *Redes de computadores Primera edición*. Barcelona : Eureka Media, SL, Marzo 2004.
2. E. Casilari, A. Reyes Lecuona, A. Díaz Estrella y F. Sandoval. *MODELADO DE TRÁFICO TELEMÁTICO*. Málaga : s.n., 1995.
3. Walpole, R.E., Myers, R.H. y Myers, S.L.&Ye,K. *Probabilidad y estadística para ingenieros*. sexta edición. México : PearsonPrentice-Hall, 2007.
4. MONTGOMERY, Douglas C. . *Probabilidad y estadística aplicadas a la ingeniería*. Segunda. s.l. : Editorial Limusa, 2007.
5. Monroy, Marco Aurelio Alzate. *Introducción al Tráfico Autosimilar en Redes de Comunicaciones*.
6. Carlos Antonio Acevedo Solano, Fabian Armando Arango Fontecha. *Verificación del modelo matemático de un rounter de servicios*. Bucaramanga : s.n., 2008.
7. [http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html). [En línea] fecha de consulta Abril de 2011
8. [http://www.opnet.com/solutions/network\\_rd/high\\_fidelity\\_modeling.html](http://www.opnet.com/solutions/network_rd/high_fidelity_modeling.html). [En línea] fecha de consulta Abril de 2011
9. mathwave. [http://www.mathwave.com/es/products/easyfit\\_features.html](http://www.mathwave.com/es/products/easyfit_features.html). [En línea] fecha de consulta Abril de 2011
10. Lecuona, Arcadio Reyes. *Modelado de tráfico de clientes www*. [Universidad de malaga] Malaga : s.n., 2001.
11. Roberto Garcíá, Victor Garcíá, Xabiel G. Pañeda ,David Melendi, Jesús Pérez. *Analysis and modelling of a broadband fiber access network with high peer-to-peer traffic load*. [Computer Science Department, University of Oviedo] septiembre de 2005.
12. Kuan-Ta Chen, Polly Huang, Chin-Laung Lei. *Game traffic analysis: An MMORPG perspective*. [Department of Electrical Engineering, National Taiwan University, Taiwan, ROC] diciembre de 2005.

13. Floyd, Vern Paxson and Sally. *Wide-Area Traffic: The Failure of Poisson Modeling*. [Lawrence Berkeley Laboratory and EECS Division, University of California] julio de 1995.
14. Xi Li, Su Li, Carmelita Görg, and Andreas Timm-Giel. *Traffic Modeling and Characterization for UTRAN*.
15. Marcelo Eduardo Pellenz, Richard Demo Souza. *A conceptually simple framework for simulating hierarchical MPEG video traffic*. [International Journal of Electronics and Communications (AEÜ),journal homepage: [www.elsevier.de/aeue](http://www.elsevier.de/aeue)] 2010.
16. Mojgan Soraya, Masood Zamani,Abdolreza Abhari. *MODELING OF MULTIMEDIA FILES ON THE WEB 2.0*. [Ryerson University,Toronto,Canadá]
17. Trang Dinh Dang, Balázs Sonkoly, Sándor Molnár. *Fractal Analysis and Modeling of VoIP Traffic*. [High Speed Networks Lab, Dept. of Telecommunications & Media Informatics] 2002.
18. Hyung-Jong Kim a, Myuhng Joo Kim a, Yoonjeong Kim a, Hyun Cheol Jeong b. *DEVS-Based modeling of VoIP spam callers' behavior for SPIT level calculation*. [Department of Information Security, Seoul Women's University, 623 Hwarangro, Nowongu, Seoul 139-774, Republic of Korea] marzo de 2008.
19. Murad S. Taqqu, Vadim Teverovsky, Walter Willinger. *Estimators for long-range dependence: an empitical study*. [Boston University, Bellcore]
20. *Matlab*. [En línea]  
<http://www.mathworks.com/matlabcentral/fileexchange/19148-hurst-parameter-estimate>. fecha de consulta Abril de 2011