HERRAMIENTAS DE USO LIBRE PARA LA PARAMETRIZACIÓN DE ANCHO DE BANDA TENIENDO EN CUENTA CARACTERÍSTICAS DE FUNCIONALIDAD DEL CANAL DE UNA RED

JUAN DAVID BALLESTEROS CASTRO
JUAN CARLOS CARO CIPAMOCHA

UNIVERSIDAD SANTO TOMÁS DE AQUINO
Facultad de Ingeniería Electrónica
Tunja, Colombia
2015

HERRAMIENTAS DE USO LIBRE PARA LA PARAMETRIZACIÓN DE ANCHO DE BANDA TENIENDO EN CUENTA CARACTERÍSTICAS DE FUNCIONALIDAD DEL CANAL DE UNA RED

JUAN DAVID BALLESTEROS CASTRO
JUAN CARLOS CARO CIPAMOCHA

Proyecto de grado para optar por el título de Ingeniero Electrónico

Director
Msc. WILLIAM FABIÁN CHAPARRO BECERRA

UNIVERSIDAD SANTO TOMÁS
Facultad de Ingeniería Electrónica
Tunja, Colombia
2015

Exoneración de responsabilidades

Las ideas expresadas en esta tesis
Son responsabilidad exclusiva de los autores,
No es la opinión de la universidad Santo Tomas
O Facultad de Ingeniería Electrónica

Nota de aceptación

Observaciones
_____
_____
_____
_____
_____

_____
Firma Decano

_____
Firma primer Jurado

_____
Firma Segundo Jurado

_____
Firma Director

Dedicatoria

Juan David Ballesteros Castro

Principalmente a Dios porque es mi guía, señor y padre que me orienta en mi diario vivir a tomar las mejores decisiones posibles; también a mi núcleo familiar porque son un apoyo incondicional, productivo y con un bien único donde buscan que sea el mejor y me dan ese empuje que necesito para salir adelante.

Por último a todas aquellas personas que depositaron su confianza en mí desde un comienzo y no dejaron que desfalleciera en este camino tan arduo. Esas mismas que me han formado y han logrado un ser humano lleno de principios y valores, con ideales claros y metas propuestas para lograr ser una persona íntegra el día de mañana. Espero que se den cuenta no solo por el resultado de este proyecto, sino que se sientan orgullosos de mi formación como persona. Gracias.

Juan Carlos Caro

Agradecimientos

Principalmente a Dios por la sabiduría y camino por el que nos guió y ayudó a salir adelante, brindándonos fortaleza en los momentos de desesperación y angustia.

A nuestros padres quienes siempre nos incentivaron y fueron el motor que permitió seguir en este proceso de formación y aprendizaje para alcanzar cada propósito planteado en este camino.

A la Universidad Santo Tomás de Aquino, por el apoyo que nos brindó durante nuestro proceso de aprendizaje; de igual forma a la Facultad de Ingeniería Electrónica con su cuerpo de docentes por brindar herramientas y conocimientos que se están viendo reflejados en este proceso.

A los diferentes ingenieros por el tiempo empleado en el desarrollo de la aplicación del proyecto, y también por compartir sus habilidades y experiencias en las diversas ramas de la ingeniería.

Por último a todas aquellas personas que confiaron en nuestras aptitudes y actitudes, de nuestra responsabilidad y decoro con el que manejamos la investigación y de los conocimientos que poseemos.

Tabla de contenido

Lista de Tablas

Lista de Ecuaciones

Lista de Figuras

Lista de Anexos

Glosario

- Máximo ancho del canal: es la cantidad de información que puede ser relativamente transmitida sobre un canal de comunicación.
- Overhead: Para casos de transmisión de datos, el overhead hace referencia al ancho de banda consumido en procesos de control, necesarios para el correcto funcionamiento de una red algunos como datos de secuencia, flags o datos de cabecera. (Stallings, 1999)
- Latencia: hace referencia al tiempo que demora un paquete con respecto a otro en redes de comunicaciones.
- Jitter: En redes de comunicación de paquetes, se utiliza este término para referirse a la variación entre el retardo de unos paquetes con respecto a otros. (Stallings, 1999)
- Ancho de banda disponible: El ancho disponible (AB) es la capacidad no utilizada en un enlace de datos.
- Testbed (banco de pruebas): Sistema que se utiliza para realizar medidas, en el que algunos de los elementos son idénticos al sistema original, mientras que el resto son emulados o sustituidos por otros que imitan su comportamiento. (Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)
- Router (enrutador): Dispositivo que conmuta paquetes entre varias redes a las que se encuentra conectado.

RESUMEN

Título.
Herramientas de uso libre para la parametrización de ancho de banda teniendo en cuenta características de funcionalidad del canal de una red.

Autores.
BALLESTEROS CASTRO JUAN DAVID
CARO CIPAMOCHA JUAN CARLOS

Palabras Claves.
Ancho de Banda, Linux, Software libre, Traceband, Iperf, Ifstat, Overhead, Consumo, Jitter, comunicación, capacidad máxima, congestión, cross-trafic, Testbed.

Descripción.
La estimación de ancho de banda disponible es un parámetro fundamental para el funcionamiento adecuado de cualquier red de datos y es estudiado por investigadores de todo el mundo con el fin de contar con esta información para mejorar la operación de varias aplicaciones de red como cumplimiento de niveles de servicio, gestión de redes, ingeniería de tráfico en tiempo real, ataques de red y control de admisión.

En este documento se presenta un estudio de aplicaciones disponibles para la medición de ancho de banda de una red que funcionan bajo el sistema operativo Linux. Para el desarrollo se realizó una investigación de información en diferentes fuentes para evidenciar los antecedentes acerca del estudio de dicho parámetro; posterior a esto, se tuvo en cuenta información de documentos e investigaciones de autores encontradas en IEEE Xplore y libros que se fundamentan su contenido en el estudio de esta variable.

Seguido de esto, a partir de pruebas en una maqueta (Testbed) que recrea las características de una red, se realizó una comparación de cada herramienta con la finalidad de evidenciar su comportamiento, variables medidas, usabilidad, ventajas y deficiencias. Teniendo en cuenta este referente, se dio prelación a las aplicaciones que muestren de manera más adecuada la medición de ancho de banda se realizaron pruebas más exhaustivas para realizar comparaciones entre cada una de ellas.

Además de las variables mencionadas para realizar una medición adecuada, las aplicaciones más relevantes debieron cumplir con los siguientes parámetros (robustez, aplicabilidad, medición, entre otros); la cantidad de recursos necesarios para su funcionamiento desde el punto de vista de carga de hardware, la calidad de la medición y finalmente su comportamiento en diferentes situaciones de usabilidad de la red.

Finalmente se argumentó cual es la aplicación más robusta por su adaptación al entorno en la parametrización de ancho de banda basándose en los argumentos anteriormente mencionados, donde se caracterizó su comportamiento por medio de un análisis de datos y una interpretación de gráficas generadas a partir de Microsoft Excel® y Matlab® donde se evidencio la calidad de la medición y sus diferentes ventajas sobre los otros estimadores.

1. INTRODUCCIÓN

A partir de las redes de datos y el avance de la tecnología se ha logrado entablar diferentes tipos de servicios e interacción entre usuarios que desean transmitir y recibir diferentes contenidos (información) de carácter científico, cultural, económico, entre otros; (Ruiz, 2012) la calidad con la que se prestan dichos servicios representan la eficiencia con la que cuenta la red de datos; el despliegue desmesurado han hecho que la saturación sea un problema el cual ha tenido relevancia y por tal motivo se ha convertido en objeto de estudio; (Reggini, 2009) en este sentido existen aplicaciones que permiten realizar auditorías para evidenciar problemas y a su vez posibles soluciones a detrimentos presentados en las redes mencionadas. Con el fin de encontrar una aplicación que permita determinar de manera apropiada el ancho de banda utilizado en una red de datos y a través del mismo parametrizar la usabilidad de este importante parámetro de red. Con ello se busca garantizar un servicio robusto y de calidad. En concordancia con lo anterior, se busca realizar una investigación basada en un modelo descriptivo experimental a través del cual se modele un estudio y con ello la parametrización de referentes de uso y medición del ancho de banda disponible y utilizada sobre cualquier red de datos; el desempeño, la calidad en el servicio, y la robustez de red, principalmente. Por consiguiente, por medio de una selección previa de aplicaciones libres cuya función es el estudio de ancho de banda se buscó realizar un diagnóstico previo evidenciando características tales como latencia, overhead, tráfico del canal, ancho del mismo, entre otros; lo anterior pretende estructurar un documento que respalde el uso de una u otra aplicación bajo parámetros de funcionamiento variantes y consecuentemente la que mejor se integre a la red de datos definida.

Con el anterior referente, se realizaron pruebas específicas en una red de laboratorio (pruebas controladas con y sin cross-trafic) que permitieron caracterizar el comportamiento de una red LAN en diferentes situaciones de usabilidad. Ahora, utilizando un Testbed específico donde se caracterizó la mayoría de redes de datos (router switch y host) se obtuvieron resultados que permitieron realizar de manera objetiva la selección de una aplicación que se integra de manera robusta y eficiente a la medición de ancho de banda.

2. JUSTIFICACIÓN

El desarrollo del presente trabajo se realiza con la intención de encontrar herramientas robustas que permitan evidenciar problemas en una red de datos y presentar posibles soluciones las cuales potencialmente podrían ser implementadas generando impactos positivos en diferentes aspectos.

Como se ha mencionado anteriormente, el ancho de banda en una red es muy importante porque a partir del mismo se puede evidenciar que tan eficiente es la red para el intercambio de información. Por consiguiente, durante el desarrollo de esta investigación se evidenciaron diferentes características del ancho de banda por medio de herramientas especiales para la parametrización de este factor; resaltando que el bajo rendimiento de una red se ve representado en demoras de acceso y procesamiento de la información. (Escorsa Castells & Valls, 2009)

De igual forma, garantizando una mejor calidad de servicio, a partir del ancho de banda disponible en una red se pueden solucionar múltiples problemas; algunos por mencionar como operación de aplicaciones que demandan gran cantidad de este recurso, una mejor gestión en las redes de datos, y aún más buscando impactos desde el punto de vista de ataques de red y control de usuarios. (Oshiba & Nakajima, 2014)

3. PLANTEAMIENTO DEL PROBLEMA

El ancho de banda de una red es un aspecto fundamental para el envío y recepción de datos; sin embargo están expuestas a degeneraciones en la calidad del servicio, por este motivo existen diferentes herramientas que permiten el estudio y parametrización de variables implícitas que correlacionan el comportamiento del mismo. En la actualidad, y con la popularización de redes de datos sobre protocolo TCP/IP transitan diferentes tipos de información sobre aplicaciones que requieren en cierta medida mayor o menor ancho de banda, la prioridad se destaca en la obtención de una calidad de servicio cada vez mejor. (Jordan & Galperín, 2013)

Se tiene en cuenta que una baja calidad de transmisión de información en una red puede darse a partir de diferentes factores que puedan existir en el canal tales como congestión, alta latencia, procesos implícitos que ocupan parte del ancho de banda, entre otros. Por esto, cuando se intenta acceder a datos específicos, carga de archivos, tecnología VOIP; aparecen herramientas (algunas de uso libre) que permiten verificar, parametrizar y diagnosticar las características que existen en el ancho de banda y dar soluciones a los problemas del mismo. (Stallings, 1999)

Teniendo en cuenta dicho antecedente es importante obtener características que permitan evidenciar el comportamiento de cualquier red; para alcanzar dicho propósito se realizará una investigación sobre los historiales acerca del estudio de ancho de banda, reconociendo que aplicaciones se ajustan adecuadamente al estudio del parámetro específico y dando prioridad a las aplicaciones que brindan información más amplia y detallada. Finalmente exaltar que herramienta se adapta mejor al entorno y al estudio de ancho de banda dando prioridad al tipo de información entregada, variables parametrizadas, eficiencia y robustez de la misma.

4. OBJETIVOS

4.1. Objetivo General.

Estructurar el modelo de selección de una aplicación de uso libre que permita diagnosticar los efectos del ancho de banda en el desempeño de una red de datos definida.

4.2. Objetivos Específicos.

- Reconocer la información relevante encontrada en las diferentes fuentes consultadas, para corroborar la orientación y desarrollo de la investigación.
- Realizar un estudio de las aplicaciones que existen actualmente, prestando atención en el tipo de medición PRM o PGM, tener en cuenta este parámetro para evidenciar cual se ajusta de manera más apropiada a las condiciones de las redes a realizar la parametrización.
- Estudiar la funcionalidad de aplicaciones en la medición de ancho de banda resaltando su eficiencia en funciones específicas, teniendo en cuenta este antecedente, correlacionar las diferentes herramientas que operan en sistemas operativos de uso libre para la descripción del ancho de banda de una red.
- Exaltar la herramienta que se adapta mejor al estudio del ancho de banda priorizando en su tipo de medición, variables parametrizadas, eficiencia, robustez y operatividad.

5. MARCO TEÓRICO

El modo de transferencia asincrónica sirve para una amplia gama de tráfico multimedia. Mientras ATM proporciona mayor flexibilidad en el apoyo a varios tipos de tráfico, los problemas de control de tráfico se convierten en una incógnita difícil de resolver cuando se trata de lograr un uso eficiente de los recursos de red. Uno de esos problemas es la asignación de ancho de banda. (Oshiba & Nakajima, 2014)

Antes de mencionar la conceptualización necesaria para el estudio de ancho de banda; es importante tener en cuenta lineamientos dentro de la pila OSI (Open Systems Interconection) fundamentales para el funcionamiento de una red; dentro de los lineamientos que especifica OSI; hay cuatro objetivos fundamentales:

- Interconexión: definir las reglas que posibiliten la interconexión física y la transmisión de datos entre diferentes máquinas.
- Interoperabilidad: posibilita el trabajo interactivo entre máquinas, es decir, además de la transmisión de la información, la compresión y el proceso de la misma.
- Independencia de la instalación: el modelo puede ser implementado en cualquier arquitectura.
- Extremos abiertos: que la comunicación no se vea limitada a máquinas que trabajan con el mismo software.

El modelo OSI define los servicios y protocolos que posibilitan la comunicación dividiéndolos en siete niveles diferentes. En la figura número 1 se muestran las diferentes capas que OSI caracteriza. (Marin Moreno, Modelo OSI, 1997)

Figura 1. Estructura del modelo OSI. **(Marin Moreno, Modelo OSI, 1997)**

5.1. Explicación niveles pila OSI

5.1.1. Capa 7, Aplicación.

Esta es la capa que interactúa con el sistema operativo o aplicación cuando el usuario decide transferir archivos, leer mensajes, o realizar otras actividades de red. Por ello, en esta capa se incluyen tecnologías tales como http, DNS, SMTP, SSH, Telnet, etc.

5.1.2. Capa 6, Presentación.

Esta capa tiene la misión de coger los datos que han sido entregados por la capa de aplicación, y convertirlos en un formato estándar que otras capas puedan entender. En esta capa tenemos como ejemplo los formatos MP3, MPG, GIF, etc.

5.1.3. Capa 5, Sesión.

Esta capa establece, mantiene y termina las comunicaciones que se forman entre dispositivos. Se pueden poner como ejemplo, las sesiones SQL, RPC, NetBIOS, etc.

En el grupo de transporte tenemos:

5.1.4. Capa 4, Transporte.

Esta capa mantiene el control de flujo de datos, y provee de verificación de errores y recuperación de datos entre dispositivos. Control de flujo significa que la capa de transporte vigila si los datos vienen de más de una aplicación e integra cada uno de los datos de aplicación en un solo flujo dentro de la red física. Como ejemplos más claros tenemos TCP y UDP. (Marin Moreno, Modelo OSI, 1997)

5.1.5. Capa 3, Red.

Esta capa determina la forma en que serán mandados los datos al dispositivo receptor. Aquí se manejan los protocolos de enrutamiento y el manejo de direcciones IP. En esta capa hablamos de IP, IPX, X.25, etc.

5.1.6. Capa 2, Datos.

También llamada capa de enlaces de datos. En esta capa, el protocolo físico adecuado es asignado a los datos. Se asigna el tipo de red y la secuencia de paquetes utilizada. Los ejemplos más claros son Ethernet, ATM, Frame Relay, etc.

5.1.7. Capa 1, Física.

Este es el nivel de lo que llamamos hardware. Define las características físicas de la red, como las conexiones, niveles de voltaje, cableado, etc. Se incluye en esta capa la fibra óptica, el par trenzado, cable cruzados, etc.

Cada nivel se encarga de puntualizar la forma de funcionamiento para realizar una conexión exitosa; cada nivel está compuesto por entidades que son los elementos que realizan las funciones impuestas en ese nivel, las dos únicas capas del modelo con las que interactúa el usuario son la primera capa que como menciona

la pila se refiere a la capa física, junto con la última o más superior que se refiere a la capa de aplicación. (Hu & Steenkiste, 2003)

5.2. Jitter.

En general, cuando se habla de Jitter en el entorno de comunicaciones en redes de ordenadores, se hace referencia a la variación entre el retardo de unos paquetes con respecto a otros. Las redes de paquetes pueden retardar de manera diferente los distintos paquetes de un mismo flujo y eso produce problemas.

El Jitter representa diferentes tipos de problemas cuando se descarga un fichero; esta variable es casi despreciable ya que no importa si se recibe un paquete antes o después del otro; por el contrario, lo que realmente importa es tener un ancho de banda grande para que la transferencia de información se haga lo más rápida posible.

Se define Variación Instantánea del retardo del paquete, IPDV para un par de paquetes del mismo flujo como la diferencia entre el retardo en un sentido (OWD) de dos paquetes. De igual manera, el jitter tiene otros dos significados: el primero, la variación de una señal respecto a otra señal "reloj"; y segundo, la variación de una métrica respecto a una métrica de referencia. En la figura se puede apreciar de manera gráfica este comportamiento. (Stallings, 1999)

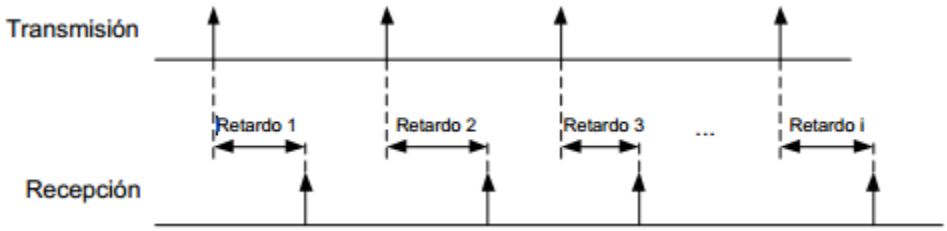

Figura 2. Diferentes retardos para paquetes del mismo flujo. **(Stallings, 1999)**

5.3. Overhead.

Para casos de transmisión de datos, el overhead hace referencia al desperdicio de ancho de banda que existe en una red; este es causado por información adicional en diferentes procesos como de control, de secuencia, entre otros; que deben transmitir no solo en datos sino que en paquetes de un medio de comunicación.

El overhead, afecta directamente a la cantidad de paquetes que se entregan en una unidad de tiempo; en un nodo de la red siendo medio físico o lógico. En redes se transmiten los mensajes subdivididos en paquetes. Para poder identificarlos es necesario contar con una cabecera para así volver a armar el paquete y que los datos sean identificados. (Stallings, 1999)

5.4. Ancho de banda.

En sistemas digitales, el ancho de banda digital es la cantidad de datos que pueden ser transportados por algún medio en un determinado período de tiempo (generalmente segundos). Por lo tanto, a mayor ancho de banda, mayor transferencia de datos por unidad de tiempo (mayor velocidad). (Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)

5.4.1. Ancho de Banda Disponible.

El ancho de banda Disponible (AB) es la capacidad no utilizada en el enlace.

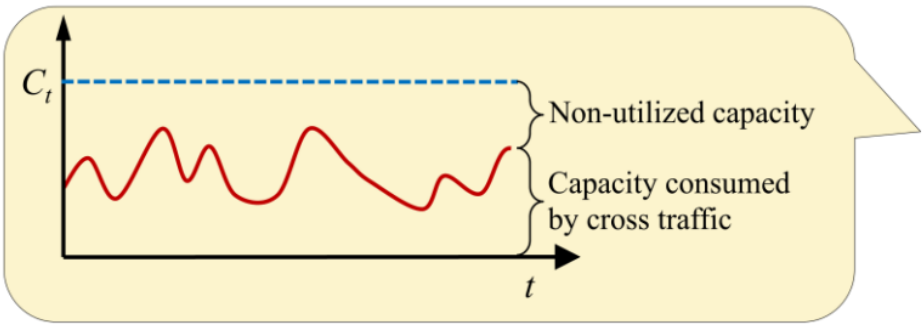

Figura 3. Explicación gráfica del ancho de banda consumido. **(Guerrero, Traceband: Available Bandidth Estimation based on a Hidden Markov Mode, 2009)**

El cual estará dado por la siguiente expresión:

$$AB_i = C_i - (Cross\ traffic)_i$$

Ecuación 1. Ancho de Banda consumido. **(Guerrero, Traceband: Available Bandidth Estimation based on a Hidden Markov Mode, 2009)**

El enlace con la tasa de transmisión mínima determina la capacidad del camino (Enlace estrecho – Narrow link), mientras que el enlace con el mínimo de capacidad no utilizada (enlace apretado – Tight link) limita la capacidad del ancho de banda disponible.

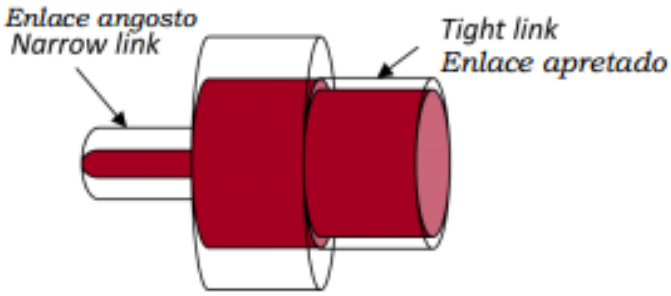

Figura 4. Enlace angosto y apretado. **(Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)**

Para AB el mínimo de todos los anchos de banda disponibles, es igual a AB en el enlace apretado, y se puede denotar por:

5.4.2. Ancho de banda disponible de extremo a extremo (End to End)

$$AB_{path} = \min(AB_i) = AB_{tight\ link}$$

Ecuación 2. Ancho de banda End to End. **(Guerrero, Traceband: Available Bandidth Estimation based on a Hidden Markov Mode, 2009)**

El mínimo de todas las capacidades de enlace no utilizadas en toda la ruta de conmutación se llama ancho de banda disponible de extremo a extremo. Esta es una métrica variable en el tiempo que está relacionada con la utilización individual de cada enlace en toda la ruta. Definiendo a $\boldsymbol{\tau}$**,** como la escala de tiempo promedio del ancho de banda disponible, la utilización promedio del enlace i para una muestra de tiempo $\boldsymbol{\tau}$, se obtiene por medio de la ecuación: (Allman & Paxson, 1999)

$$\bar{u}_i = \frac{1}{t}\int_t^{t+\tau} u_t(s)ds_t \;\; 0 \leq \bar{u}_i \leq 1$$

Ecuación 3. Tiempo promedio del ancho de banda. **(López, 2010)**

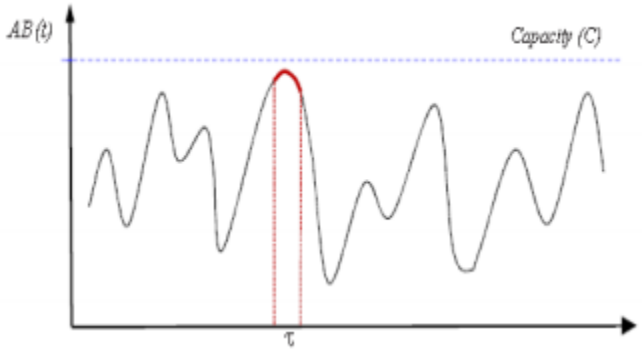

Figura 5. Ancho de banda disponible en un periodo promedio de la escala de tiempo. **(López, 2010)**

Para un enlace i con una capacidad Ci, el ancho de banda disponible del enlace en el intervalo:

$$(t, t+\tau)$$

Ecuación 4. Intervalo de tiempo para la capacidad del ancho de banda. **(López, 2010)**

Se puede definir como la capacidad no utilizada promedio durante el tiempo $\boldsymbol{\tau}$

$$\boldsymbol{\bar{A}_i = C_i[1 - \bar{u}_i]}$$

Ecuación 5. Capacidad no utilizada promedio. **(López, 2010)**

Para una ruta de extremo a extremo con H saltos, el ancho de banda disponible durante, lo da el enlace con la capacidad no utilizada mínima de todos los saltos.

$$\overline{A} = \boldsymbol{min}_{i=1\ldots H}[\overline{A}_i]$$

Ecuación 6. Capacidad no utilizada mínima. **(López, 2010)**

Tal como aparece en la ecuación 6, el enlace con la capacidad mínima se conoce como el narrow link (enlace angosto) y el enlace con el mínimo ancho de banda disponible se conoce como el tight link (enlace estrecho); el cual se considera el cuello de botella de la ruta y el enlace que determina el ancho de banda disponible de extremo a extremo. (Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)

5.5. Técnicas de estimación de ancho de banda disponible

Las técnicas que más se utilizan actualmente son el Probe Gap Model – PGM (modelo de separación de pruebas) y el Probe Rate Model- PRM (modelo de velocidad de prueba); ambas son enfoques válidos y actualmente son implementadas en la mayoría de las herramientas vigentes para realizar estimación de ancho de banda. (López, 2010)

5.5.1. Método PRM (Packet Rate Method)

Basado en los conceptos de congestión auto inducida; PRM se basa en la siguiente consideración, si se envía tráfico de prueba a una razón más baja que el ancho de banda disponible a través de un enlace, entonces la razón de llegada del tráfico de prueba debe coincidir con la razón a la cual fue enviado. Por el contrario, si la razón a la que se envía el tráfico de prueba es mayor que el ancho de banda disponible, se crea una cola en la red y el tráfico de prueba debe presentar retardo. Como resultado, la razón de los paquetes de prueba en el receptor debe ser menor que la de envío. De esta manera se puede medir el ancho de banda disponible, localizando el punto de retorno en el cual los paquetes de prueba enviados y los recibidos, comienzan a coincidir. (López, 2010)

5.5.1.1. Herramientas basadas en método PRM.

Las aplicaciones que utilizan este algoritmo para la medición de ancho de banda son PRT®, PathChirp® (Pathchirp, 2014) y TOPP® con las siguientes características:

- Un tren de paquete de prueba enviados a una velocidad menor que el ancho de banda disponible debería ser recibido a la misma velocidad enviada.
- Si la velocidad a la cual fue enviado excede el ancho de banda disponible; la velocidad de recepción es menor que la de transmisión y los paquetes de prueba van a tender a encolarse provocando retardo en ese sentido (OWD).

- El ancho de banda disponible puede ser estimado mediante la observación de la velocidad de trasmisión en la cual sucede la transición entre estos dos comportamientos.

5.5.2. Método PGM (Packet Gap Method).

Este modelo hace uso de la información del espacio de tiempo entre la llegada sucesiva de dos paquetes de prueba al receptor. Un par de paquetes de prueba es enviado con espacio determinado entre ellos $\Delta_{entrada}$ y llega al receptor con una distancia Δ_{salida}, asumiendo un único cuello de botella y que la cola no se encuentra vacía entre el envío del primer y el segundo paquete de prueba. Entonces, Δ_{salida} es el tiempo que le toma al cuello de botella y que la cola no se encuentra vacía entre el envío del primer y el segundo paquete de prueba; resumiendo así que Δ_{salida} es el tiempo que le toma al cuello de botella transmitir el segundo paquete de prueba en el par y el trafico cruzado que llega durante $\Delta_{entrada}$ como se muestra en la figura 6. (López, 2010)

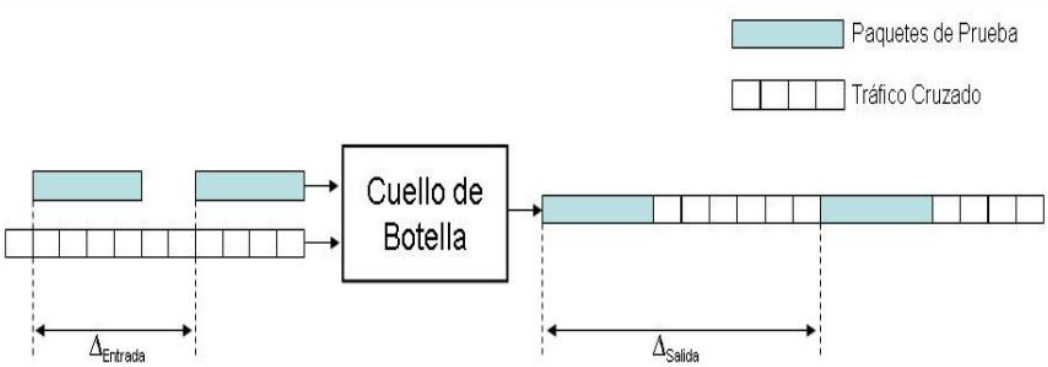

Figura 6. Tráfico cruzado a partir de la explicación del cuello de botella. **(Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)**

De esa forma, tenemos que el tiempo para transmitir el tráfico cruzado es:

$$\Delta_{salida} - \Delta_{entrada}$$

Ecuación 7. Referencia del tráfico cruzado de salida y entrada. **(Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)**

y tenemos que la razón de llegada del tráfico cruzado es Δ

$$\Delta_{salida} - \Delta_{entrada} * C, C = capacidad\ del\ cuello\ de\ botella$$

Ecuación 8. Llegada del tráfico cruzado con respecto a la capacidad del cuello de botella. **(Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, 2012)**

5.5.2.1. Herramientas basadas en método PGM

Las aplicaciones que utilizan este algoritmo para la medición de ancho de banda son SPRUCE®, Delphi® e IGI® con las siguientes características:

- Envían pares de paquetes de prueba de igual trabajo separados acorde al tiempo de transmisión de las pruebas sobre el cuello de botella del enlace. (López, 2010)
- Si no se inserta “cross-trafic” entre los paquetes de prueba se mantiene hasta el destino.
- El incremento del espacio entre paquetes de prueba es restado de la capacidad estimada para obtener el valor del ancho de banda disponible estimado.

A diferencia de PRM, el PGM asume que el enlace comprometido es el enlace más estrecho del camino y es susceptible a retardos por encolamiento en los enlaces no comprometidos. Ambos modelos usan trenes de paquetes para generar una medición y toman como ciertos los siguientes supuestos:

1. La cola en todos los routers es atendida siguiendo el modelo FIFO. (First in–First out).
2. El tráfico cruzado sigue un modelo fluido.
3. Las tazas de transmisión del tráfico cruzado, cambian lentamente y de forma constante para cada medición. (Vega Lebrun & Arvizu Gutierrez, 2011)

Adicionalmente, el modelo PGM asume un único cuello de botella que puede ser el narrow link o el tight link para cada enlace punto a punto. Estas consideraciones son necesarias para el análisis del modelo, pero las herramientas pueden trabajar, aún si algunas de ellas no se cumplen. (Comparasion of Public end-to-end Bandwidth Estimation tools on high speed links , 2005)

5.6. Herramientas de estimación de ancho de banda disponibles

En la actualidad existen diferentes herramientas de estimación de ancho de banda disponible; cada una posee diferentes formas de realizar la medición; algunas miden tráfico de red o realizan una sumatoria en la cantidad de datos enviados y recibidos. Iftop® por ejemplo, se limita a observar las direcciones IP que se resuelven a partir de los dominios de las páginas visitadas; mientras que Bmon®, permite observar el tráfico sobre todas las interfaces de red. (Capone & Martignon, 2010) En un capítulo posterior, se realizarán pruebas con cada herramienta para evidenciar su comportamiento y variables a medir.

En general estas son las aplicaciones más relevantes para la medición de ancho de banda en Linux: Abing®, Pathchirp®, Traceband®, Pathload®, Nload®, Iftop®, Bmon®, Slrum®, Nethogs®, Tcptrack®, Vnstat®, Trafshow®, Speedometer®, Ifstat®, Netload®, Netwatch®, Pkstat®, CBM®, Iperf®.

5.7. TestBed

Un banco de pruebas o Testbed es una plataforma para la experimentación de proyectos de gran desarrollo. Los bancos de pruebas brindan una forma de comprobación precisa y transparente de elementos computacionales y otras nuevas tecnologías. (Guerrero & Labrador, "On the Applicability of Available Bandwidth Estimation Techniques and Tools", 2010)

El término es utilizado en muchas disciplinas para describir un ambiente de desarrollo que está protegido de los riesgos de las pruebas en un ambiente de producción. Es un método para probar un módulo particular en forma aislada.

Ahora, el Testbed que se utilizará para realizar pruebas se muestra en la figura 7; y se puede afirmar que esté es un ambiente controlado con un canal de 10 Mbps de capacidad del enlace. Por consiguiente, se pretende generar un cross-trafic a partir de dos hosts anfitriones utilizando un tráfico sintético periódico. Por otro lado, la estimación se llevará a cabo en otros dos terminales y de esta manera controlar la variable de usabilidad de recursos en los hosts.

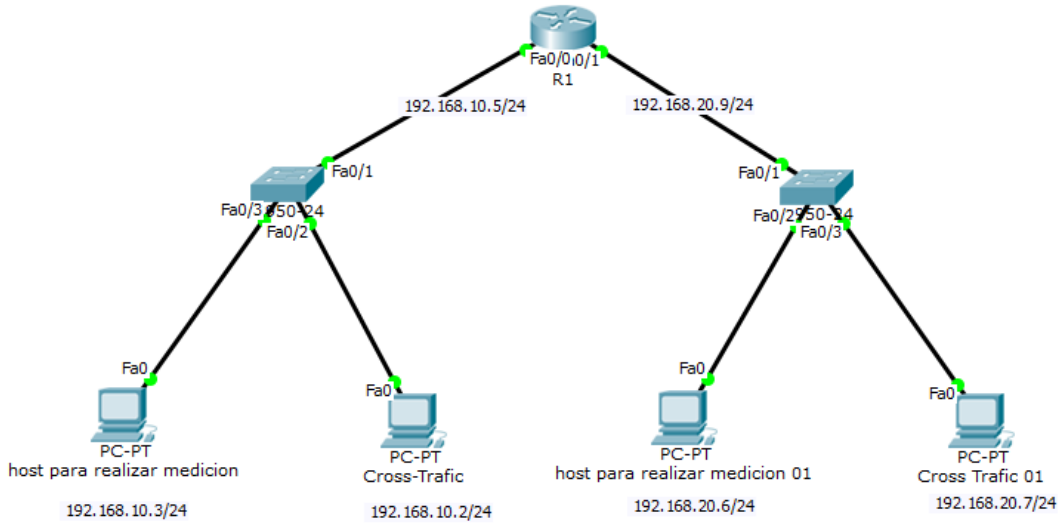

Figura 7. Topología Cisco® del Testbed a utilizar en Cisco® Packet-Tracer.

5.7.1. Configuración de equipos desde Cisco® Packet-Tracer.

Con la tabla de direccionamiento se procede a asignar los puertos y así crear la topología.

Dispositivo	Dirección	Puerta de Enlace
R1	Fa0/0 192.168.10.5/24 Fa0/1 19.168.20.9/24	----------
S1	----------	192.168.10.5
S2	----------	192.168.20.9

PC1	192.168.10.2/24	192.168.10.5
PC2	192.168.10.3/24	192.168.10.5
PC3	192.168.20.6/24	192.168.20.9
PC4	192.168.20.7/24	192.168.20.9

Tabla 1. Direccionamiento de los equipos para el Testbed.

Para dar inicio es necesario que en el Router y los switch's se revise que no existan programas previamente almacenados; por lo que se debe introducir los siguientes comandos:

--erase startup-config
--reload

Lo que hacen estos dos comandos es borrar los programas que están almacenados en los dispositivos y vuelve a cargar el sistema operativo de las máquinas para poder trabajar en estas. Seguido de esto se procede a programar el router1841 teniendo en cuenta la asignación de nombre y sabiendo que se utilizaran sus dos FastEthernet.

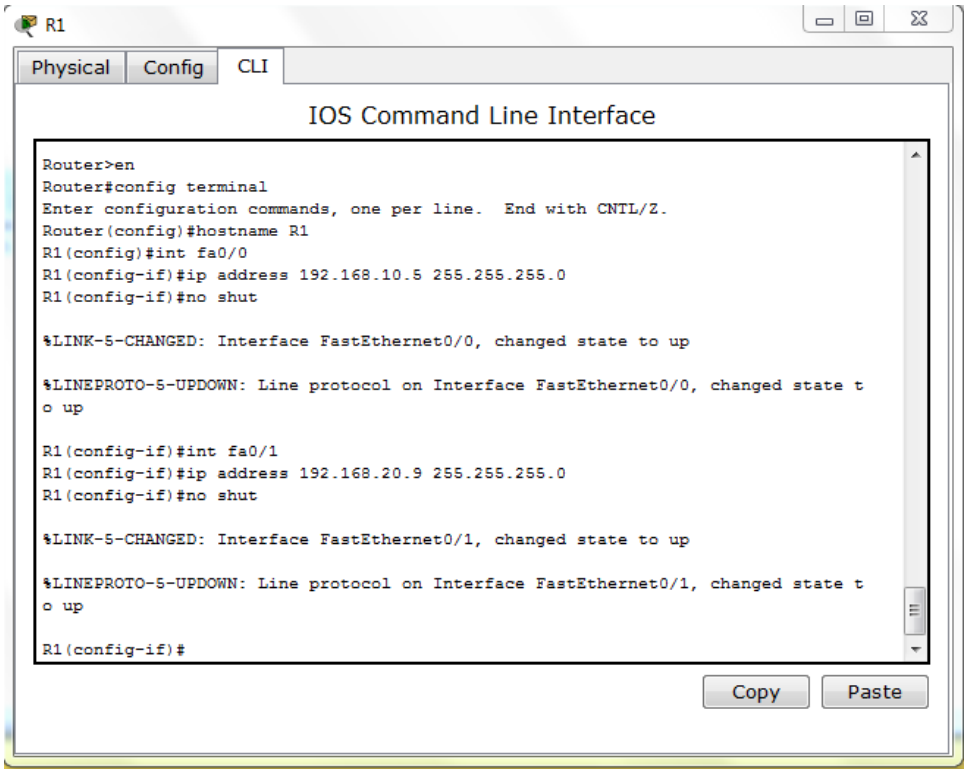

Figura 8. Interfaz del router1841 desde el software Cisco® Packet-Tracer.

En la figura 8 se observa la configuración del router1841 y el cambio de estado de las Ethernet. Posterior a esto se utiliza el comando show ip router para mirar que direcciones de red quedaron asignadas y en que puertos.

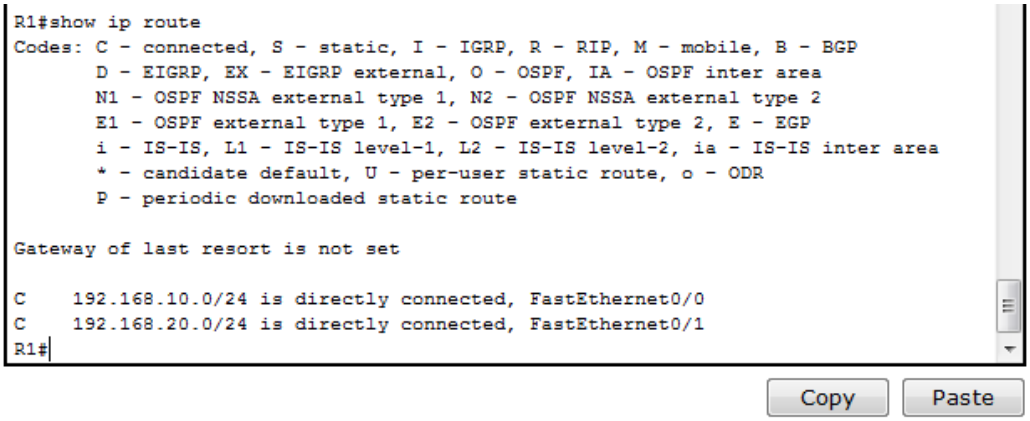

Figura 9. Interfaz del router1841 desde el software Cisco® Packet-Tracer.

Luego de que nuestro router1841 está configurado para operar con la topología propuesta se procede a trabajar sobre los switch2950 con el fin de asignar nombres y puertas de enlace en los mismos. A continuación en las figuras 10 y 11 se puede observar la interfaz de cada swicth2950 y la configuración de los mismos.

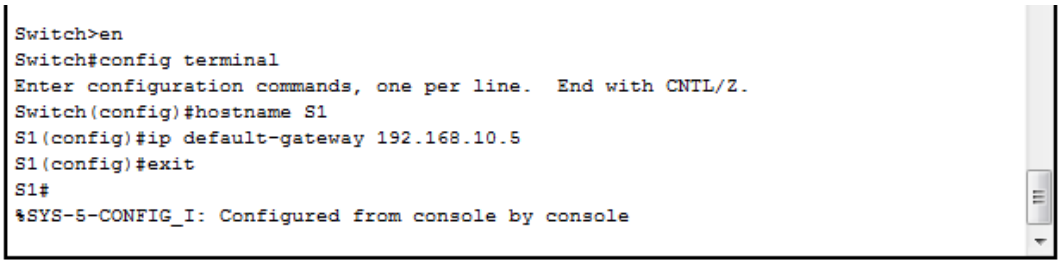

Figura 10. Interfaz del switch2950 desde el software Cisco® Packet-Tracer.

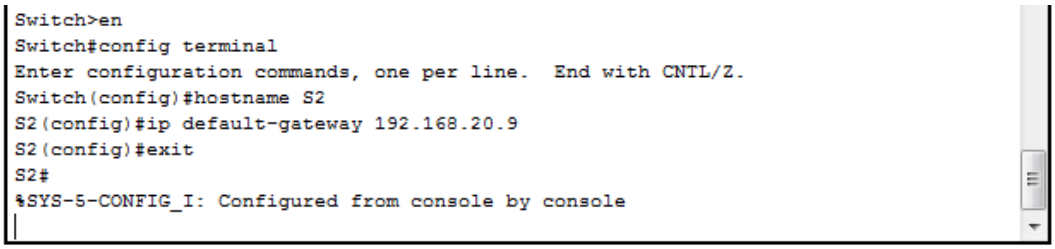

Figura 11. Interfaz del switch2950 desde el software Cisco® Packet-Tracer.

Como se observa en la figura 10 se asigna la puerta de enlace por defecto para el primer S1 según la tabla de direccionamiento y trabajando una conexión entre Fa0/1 para S1 y fa0/0 para R1. De igual forma, en la figura 11 para S2 se asigna su puerta por defecto y una conexión entre Fa0/1 para S2 y Fa0/1 para R1.

Seguido de esto, se procede a conectar los hosts a los S1 y S2 y asignar sus ip's, máscaras de red y Gateway.

Finalmente se procede a realizar un testeo con un ping entre los ordenadores para verificar que exista comunicación y así concluir que la topología no tiene problemas y se puede implementar de manera física.

Como se observa anteriormente el software Cisco Packet Tracer® permite recrear diferentes topologías que facilitan la simulación del Testbed y posterior a esto implementarlo de manera física.

5.7.2. Maquinas utilizadas en el Testbed

Los hosts utilizados para la implementación en el Testbed poseen un sistema operativo Linux (Software libre) Ubuntu en distribución 14.4. La versión del núcleo es 2.6.18.8 y dispone de un procesador inter core 2 duo a 2.40 Ghz, 2Mb de cache nivel 2, además de 4 Gb de memoria RAM y un disco duro Sata de 320 Gb.

En un principio se utilizaron máquinas virtuales del sistema operativo antes mencionado utilizando Virtual Box, pero al tener que soportar el sistema operativo nativo junto con la máquina virtual; los recursos de hardware del equipo estaban demasiado sobrecargados y eso ocasionaba retardos en las mediciones con las aplicaciones utilizadas. Dado que la finalidad es la de poder destacar una aplicación para la medición de ancho de banda teniendo en cuenta la usabilidad del sistema para su funcionamiento; fue necesario instalar en cada equipo la distribución de Linux Ubuntu y así garantizar un entorno adecuado para la correcta medición de las variables.

5.7.3. Router Cisco® 1841

El Router Cisco® 1841 es un sistema capa 3 robusto con múltiples configuraciones enfocado para pequeñas y medianas empresas. Cuenta con IOS de Cisco® para un rendimiento superior en cualquier tipo de entorno soporta VPN con un módulo adicional para aceleración. También, con un sistema de prevención de intrusos (IPS) y funciones de firewall básicas y avanzadas. Interfaces para una amplia gama de conectividad incluyendo soporte para puertos de switch. (Cisco 1841 spects, 2015)

Las características más destacables son:

- RAM 128 MB: (instalados) / 384 MB (máx.) – SDRAM.
- Memoria Flash: 32 MB (instalados) / 128 MB (máx.).
- Enlace de datos: Ethernet, Fast Ethernet.
- Red / Protocolo de transporte: IPSec.
- Protocolo de gestión remota: SNMP.
- Protocolo de señalización digital: ADSL sobre RDSI.
- 2 x RJ-45 10/100Base-TX 10/100Base-TX LAN.

- 1 x Consola Gestión.
- 1 x Auxiliar Gestión.
- 1 x USB.
- Número de Puertos : 2.
- Fast Ethernet Port: Sí.

Es importante mencionar que el Router Cisco® 1841 viene equipado con dos puertos Fast Ethernet y sus especificaciones señalan que la capacidad es de 100Mbps. Pero la capacidad de proceso del Router (Throughput) es de 75pps (Process Switching); entonces teóricamente se podría alcanzar los 100Mbsps entre ambas interfaces, pero el Router solo procesará hasta 38.4 Mbps (bajo condiciones ideales, Cisco® recomienda que se instalen capacidades hasta de un E1 en esta plataforma).

5.7.4. Switch Cisco® 2911.

El switch Cisco® 2911 ofrece características avanzadas como aceleración de cifrado integrada incluida en hardware, ranuras para procesamiento digital de señales DSP con capacidades para voz y video, firewall opcional, prevención de intrusiones, procesamiento de llamadas, diferentes opciones de conectividad cableada e inalámbrica dentro del sector entre ellas T1/E1, T3/E3 xDSL y GE en cobre y también soporte para fibra óptica. (Cisco 2911 spects, 2015)

También cuenta con sistema operativo IOS con tecnologías innovadoras que se ejecutan desde el terminal del sistema operativo, las características más representativas de este switch son las siguientes:

- 3 puertos 10/100/1000 Ethernet integradas (RJ-45 solamente).
- Ranura del módulo de servicio 1.
- 4 ranuras para tarjetas de interfaz WAN mejoradas de alta velocidad.
- 2 procesadores de señales a bordo.
- 1 ranura para módulo de servicio interno para los servicios de aplicación.
- Cifrado mediante VPN por medio de Hardware.
- Gestión de identidad mediante autenticación, autorización y contabilidad.
- Modulo DSP de voz de alta densidad- paquete.
- Soporte de correo de voz cisco Unity Express.

5.8. Mgen®.

Mgen® es una aplicación que permite la generación de tráfico sintético; tiene diferentes formas de configuración dependiendo de la necesidad y aplicabilidad. Dicho software funciona en cualquier red de datos que funcionen sobre IP; por consiguiente, su funcionamiento se basa en generar tráfico en tiempo real y realizar diferentes cargas en una topología específica. Con el código fuente en C (proporcionado por el autor) es posible realizar modificaciones para que emule diferentes comportamientos o ampliar sus capacidades, dentro de las cuales esta tráfico punto – punto, punto - multipunto. (Mgen Navymil, 2015)

MGEN® actualmente se encuentra en su versión 3.0 para sistema operativo LINUX y también disponible para Windows.

Para la presente investigación, este software tendrá la tarea de generar cross-trafic y de esta manera realizar mediciones de ancho de banda con diferente carga y usabilidad de la red, teniendo en cuenta la topología implementada en dispositivos Cisco®.

5.9. Herramientas de Simulación.

5.9.1. Excel®.

Excel® es un software que permite crear tablas, calcular y analizar datos. Este tipo de software se denomina software de hoja de cálculo. Excel® permite crear tablas que procesan de forma automática los totales de los valores numéricos que especifica; imprimir tablas con diseños y crear gráficos. Dicho software se utilizará para parametrizar variables entregadas por las aplicaciones de medición de ancho de banda y de esta forma mostrar el comportamiento gráfico y los valores en las celdas ser exportados al software Matlab®.

5.9.2. Matlab®.

Matlab® es un entorno software ampliamente utilizado en el ámbito de la investigación, ya que permite realizar con eficiencia operaciones matemáticas. También permite representar funciones, implementar algoritmos y crear interfaces de usuario. Además resulta adecuado para realizar simulaciones.

Este software será utilizado para la parametrización de variables que corresponden a las mediciones de las aplicaciones implementadas; esto con el fin de evidenciar su comportamiento con diferentes cargas de la red y tiempos de medición. Además de la generación de variables a partir de cálculos analíticos para su posterior visualización gráfica.

6. METODOLOGÍA

Para la obtención y de desarrollo de este proyecto fue necesario una investigación cualitativa experimental que permitió recoger y analizar datos cuantitativos sobre las variables a medir. El desarrollo de esta investigación se realizó por etapas de acuerdo a cada objetivo previamente planteado; al trabajar con distintas variables en condiciones controladas se describió la situación sometida y experimento provocado por el investigador. (Méndez, 2001)

Primero, se buscó un entorno con los diferentes software libres y a partir de estos se generaron respuestas que sirvieron para el estudio de la red. De igual manera, con las variables encontradas tales como velocidad, ancho del canal, overhead, latencia y Jitter; se pudo realizar un diagnóstico más fiable y de esta manera buscar una mejor eficiencia de la red.

Desde el punto de vista experimental, luego de haber realizado las diferentes simulaciones con los software libres y considerando las variables se pudo realizar un pronóstico sobre la red estudiada logrando una parametrización y aplicabilidad de la red.

Para cumplir los objetivos trazados, se realizó una serie de actividades en orden cronológico para un buen desarrollo y correlación de la investigación que se realizó. A continuación, se puede evidenciar la descripción de las actividades de manera sencilla.

En primer lugar, se realizó una investigación sobre los antecedentes acerca del estudio de ancho de banda de una red. Posterior a esto, se delimitó que tipos investigaciones eran pertinentes para la descripción, relación y cumplimiento de nuestro objetivo general.

Teniendo en cuenta lo anterior, se utilizó S.O Linux para el análisis de diferentes aplicaciones que funcionan sobre esta plataforma. Consiguiente a esto se realizó una búsqueda de herramientas cuya función era la parametrización del ancho de banda de una red. A partir de esto, se conoció por pruebas, la operación de cada una con la finalidad de evidenciar su resultado en la medición del ancho de banda de una red.

De acuerdo a resultados obtenidos, se efectuó una comparación entre las distintas aplicaciones para observar el comportamiento en la medición de ancho de banda; y así se resaltó que ventajas y deficiencias que poseían las diferentes herramientas en la medición de esta variable.

Finalmente, se priorizó el desempeño de cada una de las herramientas para así profundizar en la más adecuada para estimación de ancho de banda y como resultado final se aconsejó cuál se adaptó mejor para el estudio del ancho de banda a fin de que pueda ser utilizada en futuros estudios o necesidades en una aplicación específica.

7. RESULTADOS

El desarrollo de esta investigación está dividido en 3 capítulos. En un primer momento, se abordó una investigación de las herramientas encontradas que resultaron viables en la medición de ancho de banda; de igual forma, se realizaron pruebas para observar las medidas y la calidad de los datos mostrados.

Posterior a esto, en la segunda etapa, se profundizo más en nuevas pruebas con herramientas previamente seleccionadas con una característica de mayor eficiencia y robustez, que permitieron una medición más detallada del ancho de banda. Para lograr los requerimientos anteriores, se realizó una implementación de Testbed en sistemas Cisco® y se realizaron pruebas de laboratorio con cross-trafic (utilizando únicamente tráfico sintético periódico desde Mgen®); finalmente con los datos obtenidos, se procedió a su análisis y a través de operaciones numéricas se encontró parámetros tales como Overhead, Jitter, latencia, congestión de canal, capacidad, entre otras. Es de aclarar, que este proceso matemático se realizó desde software Excel® con el fin de lograr resultados numéricos y gráficos. Para concluir esta etapa, se procedió a importar desde el software Matlab® el libro de trabajo de Excel® (previamente elaborado); con el fin de evidenciar la robustez en la medición de las aplicaciones en entornos de diferente usabilidad de la red.

Finalmente en el capítulo 3, se destacaron las herramientas a partir del estudio de la parametrización de las variables entregadas por dichas aplicaciones y así se pudo realizar una comparación buscando ventajas, fiabilidad, eficiencia y robustez de los datos; con el fin de resaltar cuál o cuáles se ajustan más a una correcta medición del ancho de banda.

7.1. Capítulo 1. Investigación de aplicaciones para la medición de ancho de banda en S.O Linux.

En primer instante, se realizó una investigación de las diferentes aplicaciones que funcionan bajo el sistema Operativo Linux (dado que se propone trabajar con herramientas de uso libre) en sus distribuciones Ubuntu y Fedora con el fin de realizar pruebas y evidenciar su comportamiento en la medición de variables. Por consiguiente, las aplicaciones encontradas y testeadas fueron: Abing®, Pathchirp®, Traceband®, Pathload®, Nload®, Iftop®, Bmon®, Slrum®, Nethogs®, Tcptrack®, Vnstat®, Trafshow®, Speedometer®, Ifstat®, Netload®, Netwatch®, Pkstat®, CBM®, Iperf®. (Capone & Martignon, 2010), (López, 2010)

Teniendo en cuenta estas herramientas, se procede a indagar sobre cómo es el funcionamiento cada una y que variables entrega como resultado de la medición del ancho de banda. A continuación, se muestra el resultado de cada aplicación explicando en que forma trabaja la misma.

7.1.1. Nload.

Permite al usuario a partir de línea de comandos, evidenciar el tráfico entrante y saliente por separado. Es un software muy limitado y no permite muchas opciones. Para realizar pruebas se ejecuta la herramienta desde el terminal con comando sudo y se instala bien desde los archivos .tar que se encuentran en la página del autor o descargándola desde la página de Ubuntu o Fedora.

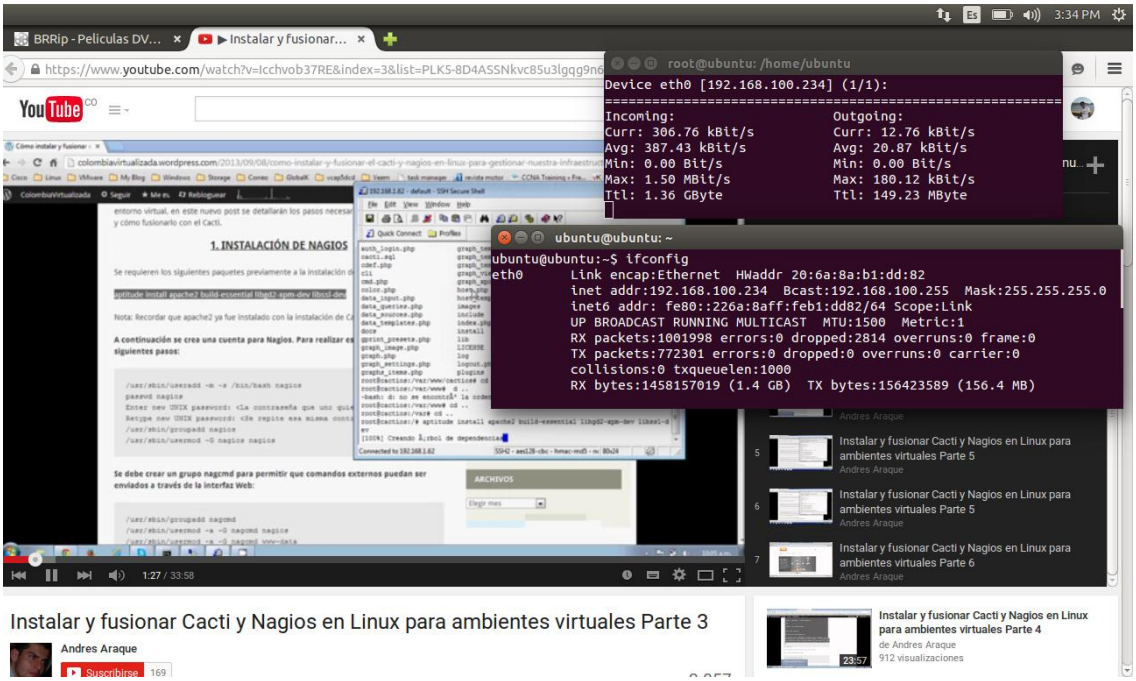

Figura 12. Ejecución del Nload® desde S.O. Linux.

Nload nos ofrece un entorno gráfico de líneas de comando tanto para la ejecución como para la visualización de las variables. Como se puede evidenciar en la figura 12 hay dos Terminal. El primero muestra la interfaz física del hosts y el segundo, los datos de entrada y salida tal como se explican en las tablas 1 y 2.

Incoming				
Curr.	Avg.	Min.	Max.	Ttl.
Velocidad de transmisión actual	Velocidad promedio total	Velocidad mínima de transmisión entrante	Velocidad máxima de transmisión entrante	Volumen total de datos entrantes

Tabla 2. Variables que ofrece Nload desde Incoming.

Outcoming				
Curr.	Avg.	Min.	Max.	Ttl.
Velocidad de transmisión actual	Velocidad promedio total	Velocidad mínima de transmisión saliente	Velocidad máxima de transmisión saliente	Volumen total de datos saliente

Tabla 3. Variables que ofrece Nload desde Outcoming.

Teniendo en cuenta la información de los dos Terminal, se evidencia el tráfico que existe en la red al reproducir un video la cantidad de datos aumenta de manera drástica.

7.1.2. Iftop.

La aplicación permite al usuario observar las direcciones ip que se resuelven a partir de los dominios de las páginas de internet.

Durante las pruebas que se realizaron se hizo el siguiente ejercicio: teniendo en cuenta un switch de 8 puertos conectado a internet, este se desconectó de la red wan y el comportamiento que tuvo la aplicación, es la de evidenciar los vecinos que tenía conectados al mismo switch. A continuación en la figura 13 se muestra el resultado de esta prueba.

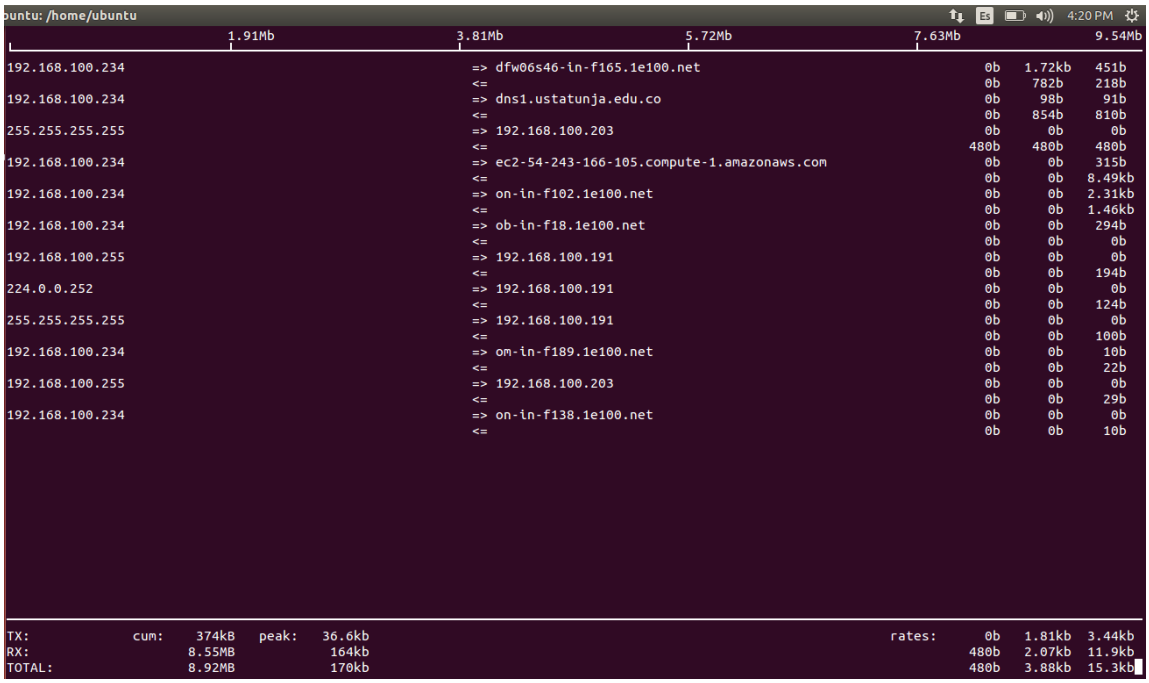

Figura 13. Ejecución del Iftop® desde S.O. Linux.

Se puede evidenciar en la figura 13, que a la izquierda se están las direcciones ip del equipo desde donde se envían las peticiones dns; seguido de esto, en el

centro las direcciones que corresponden a los dominios en su respectiva dirección ip.

7.1.3. Bmon.

Es una herramienta que permite observar el trafico sobre todas las interfaces de red en el sistema, tanto alámbricas como inalámbricas. Como se observa en la figura 14 se tiene en cuenta la información de tráfico de todas las interfaces del host; incluso en pruebas realizadas con un equipo que tiene soporte bluetooth, también se muestra el tráfico de la misma.

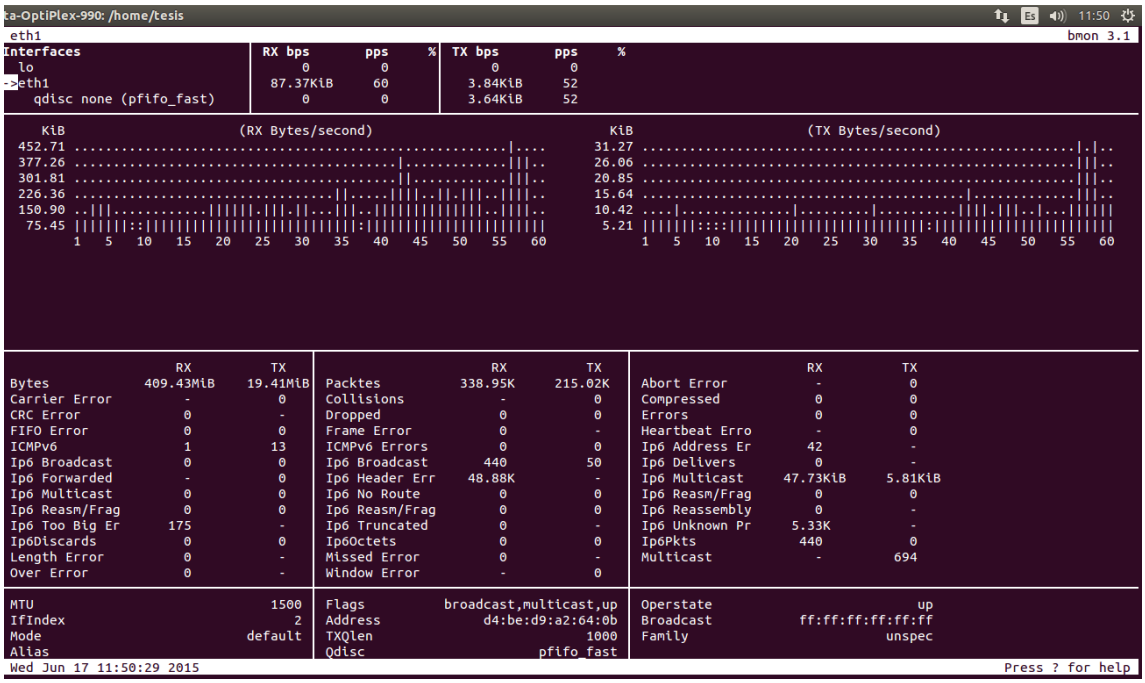

Figura 14. Ejecución del Bmon® desde S.O. Linux.

Al igual que Nload o Iptraf, Bmon entrega información del tráfico entrante y saliente. Algo relevante en esta aplicación, es el soporte que tiene para evidenciar trafico IPV6, la tasa de error en el envío y recepción de datos y finalmente una gráfica muy básica utilizando el terminal de la información de entrada y salida.

7.1.4. Slrum.

Es una aplicación muy básica; se limita a medir la cantidad de datos que se envían y reciben por una interfaz del equipo. Esta muestra la velocidad de envío y recepción de datos en tiempo real de la interfaz que sea partícipe del intercambio de datos. A continuación en la figura 15 se muestra el comportamiento de esta herramienta.

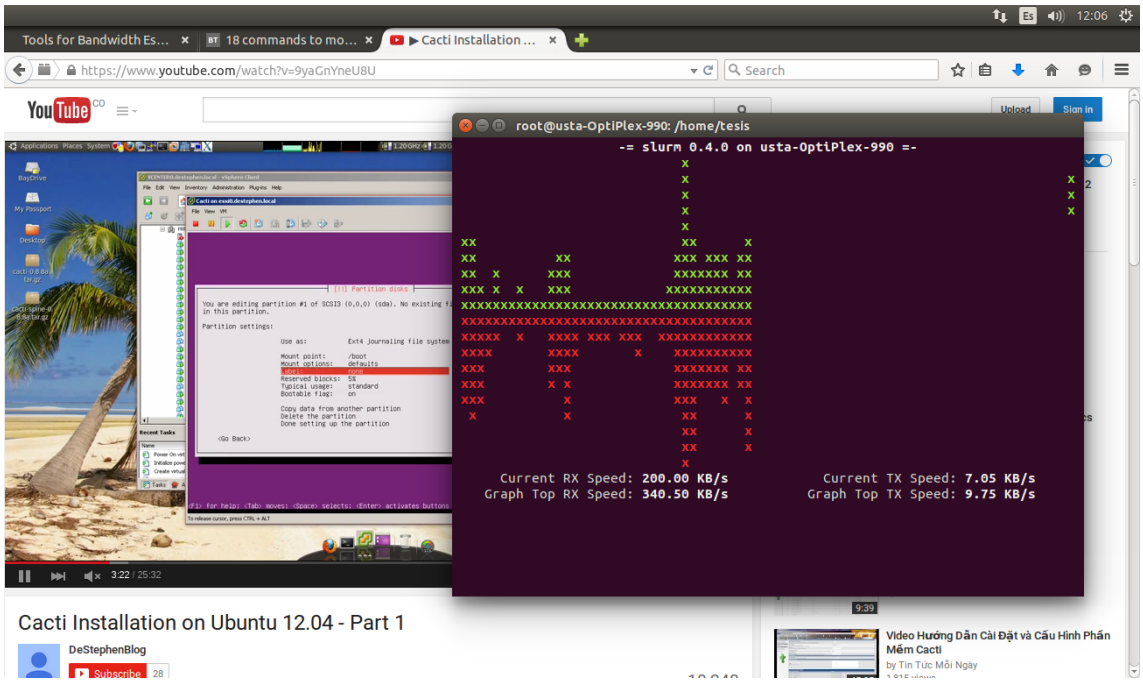

Figura 15. Ejecución del Slrum® desde S.O. Linux.

Como se observa en la figura anterior, Slrum permite al usuario visualizar la velocidad de datos de entrada y salida; seguido de una gráfica que muestra los picos de los valores adquiridos por la aplicación.

7.1.5. Nethogs.

Es un programa que se ejecuta desde el terminal, se puede descargar desde la página del autor o desde la App de Ubuntu. Actualmente se encuentra en su versión 0.8.0.

Permite ver el tráfico de información que existe desde la capa de aplicación del sistema operativo, de esta manera no solo se evidencia el tráfico desde las interfaces como hacen la gran mayoría de programas para la estimación de ancho de banda.

A continuación en la figura 16, se aprecia las aplicaciones que están demandando información Firefox, Python y el protocolo TCP. Esta herramienta permite observar la tasa de bits que están siendo enviados y la acumulación de los paquetes que se reciben desde cada interfaz que tiene el hosts.

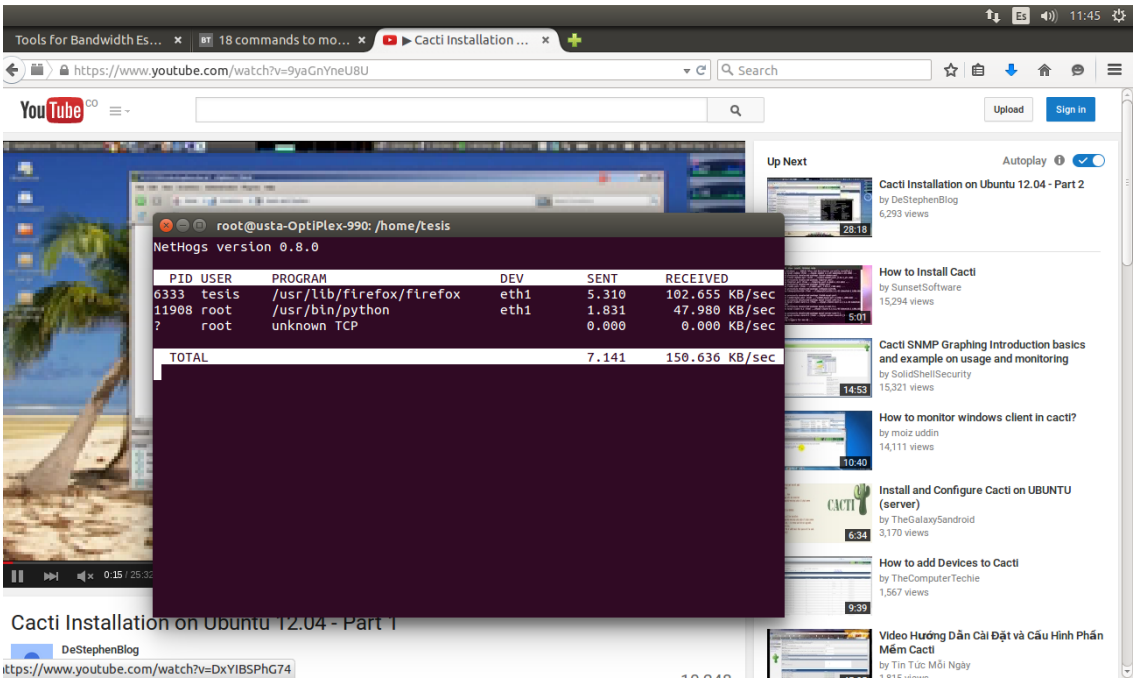

Figura 16. Ejecución del Nethogs® desde S.O. Linux.

7.1.6. Tcptrack.

Dicha herramienta nos muestra información sobre las direcciones IP presentes en el intercambio de datos, como se muestra en la figura 17 está la IP cliente con dirección 192.168.100.201 y las IP destino que son diferentes teniendo en cuenta a cual servidor se encuentra conectado en la red WAN

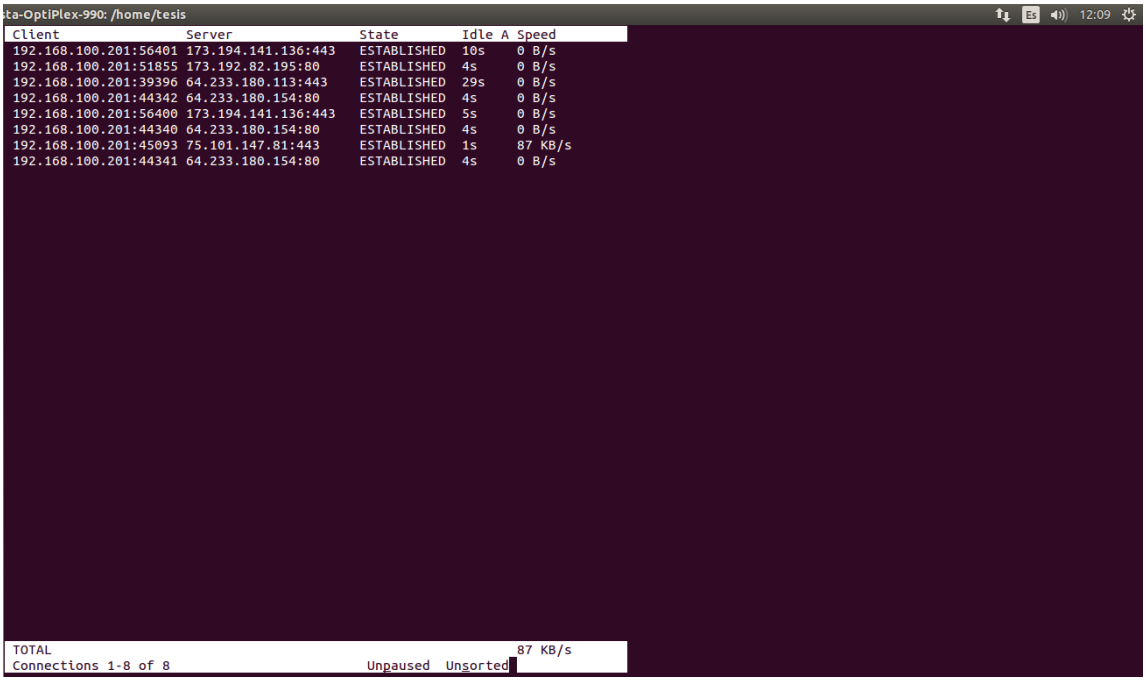

Figura 17. Ejecución del Tcptrack® desde S.O. Linux.

Se puede evidenciar en la figura 17 el comportamiento de esta aplicación; al lado izquierdo encontramos las ip del cliente con su respectivo puerto de salida; seguido se observa en la pestaña de servidor, la ip a donde van lo datos y en la pestaña de velocidad, los KBPS con que se realiza la conexión.

7.1.7. Vnstat.

Es una herramienta que se puede instalar solamente desde la página del autor; no existe soporte de la misma en las tiendas de Ubuntu y Fedora.

Se ejecuta a través del terminal y brinda la siguiente información: la cantidad de datos transmitidos en determinado tiempo. Para mejor entendimiento, se muestran dos pruebas; la primera con una duración de 1.05 minutos y otra con duración de 2.8 minutos.

La diferencia más relevante frente a las otras aplicaciones es la capacidad de medir la cantidad de información en el tiempo que dura la ejecución; de igual forma, brinda información de los datos enviados y recibidos por la interface a la cual se le está realizando el monitoreo, en este caso es la interfaz eth0 tal y como se muestra en la figura 18.

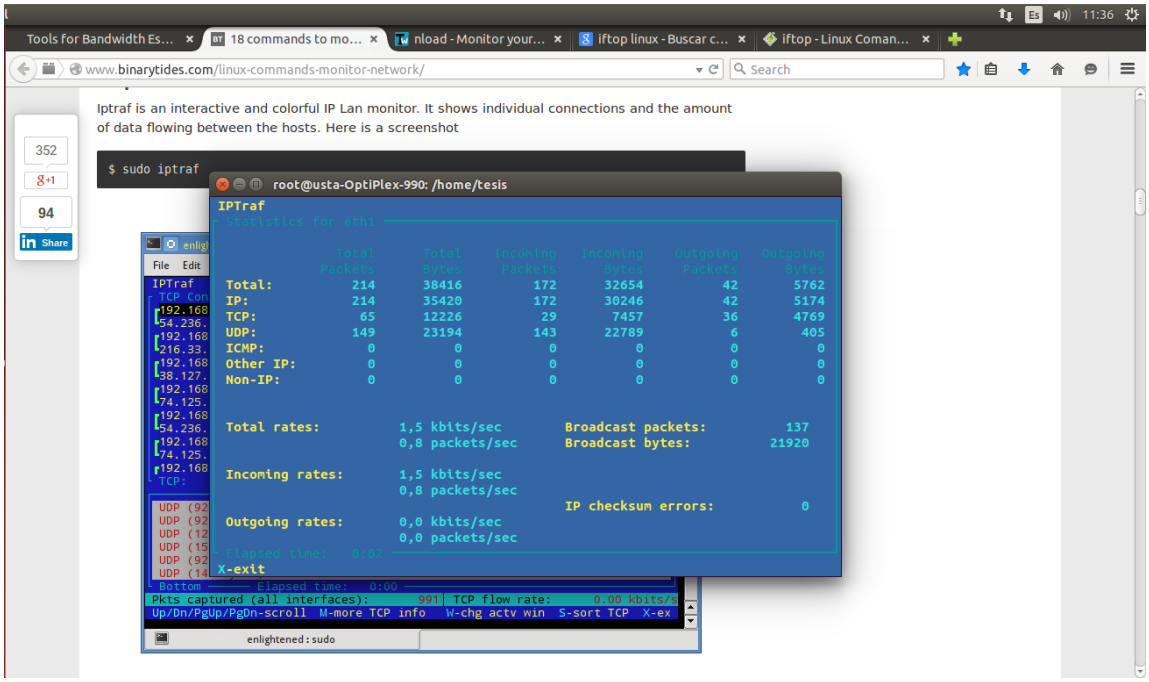

Figura 18. Ejecución del Vnstat® desde S.O. Linux.

7.1.8. Trafshow.

Es una aplicación de uso libre que se puede descargar desde la página del autor, no tiene soporte en ninguna tienda de Ubuntu o Fedora.

Permite realizar un monitoreo del tráfico en una interfaz determinada. Esta interfaz se especifica por comandos antes de la ejecución de la aplicación. A continuación en la figura 19 se muestra el comportamiento de esta aplicación.

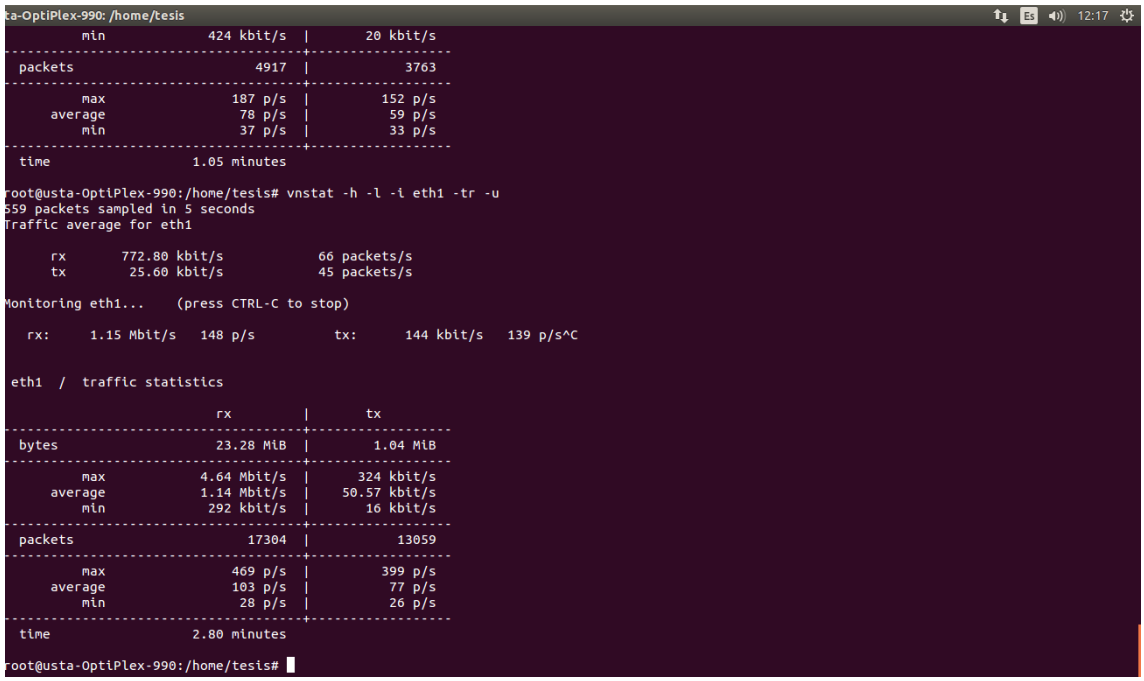

Figura 19. Ejecución del Trafshow® desde S.O. Linux.

Se puede evidenciar en la figura 19 el funcionamiento de esta herramienta; en la parte superior se muestra el tráfico TCP UDP e ICMP. Es de resaltar, que una adición que posee esta herramienta es la capacidad de medir los paquetes de Broadcast.

7.1.9. Speedometer.

Es una herramienta de estimación de ancho de banda que permite mostrar el tráfico de datos (Tx y Rx) de un mismo host. Esto haciendo tráfico sintético entre cada uno de los dos puntos. Existen aplicaciones similares en internet como test okla. A continuación, en la figura 20 se muestra el comportamiento de speedometer.

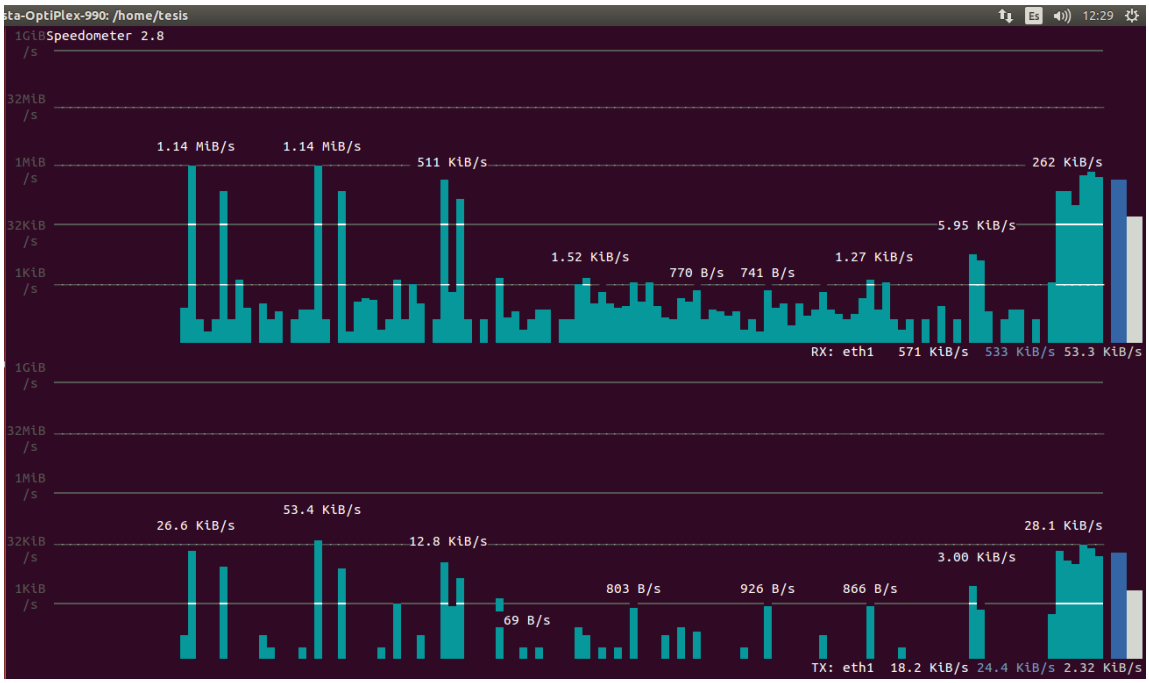

Figura 20. Ejecución del Speedometer® desde S.O. Linux.

Se puede observar en la figura 20, el análisis de tráfico de datos de eth1 tanto de envío como de recepción, teniendo en cuenta que se tienen muestras de intervalos de 1 segundo y denota el valor de bits/s para cada una. Es de resaltar que cada valor que entrega corresponde a un average del valor máximo y mínimo del momento.

7.1.10. Netload.

Es una herramienta muy sencilla dado que analiza la ip del servidor y acumula los bits de transmisión y recepción del mismo en intervalos de 1 segundo. Para cada muestra permite clasificar que datos son erróneos. Es de aclarar que por defecto la herramienta traza un step de 1 segundo para la nueva toma de datos, dicho valor no se puede modificar.

A continuación en la figura 21 se evidencia el funcionamiento de la herramienta con una navegación paralela en Mozilla Firefox®; utilizando la eth1 y reproduciendo videos desde la red con el fin de encontrar como se congestiona el canal y que transferencia de bits existe.

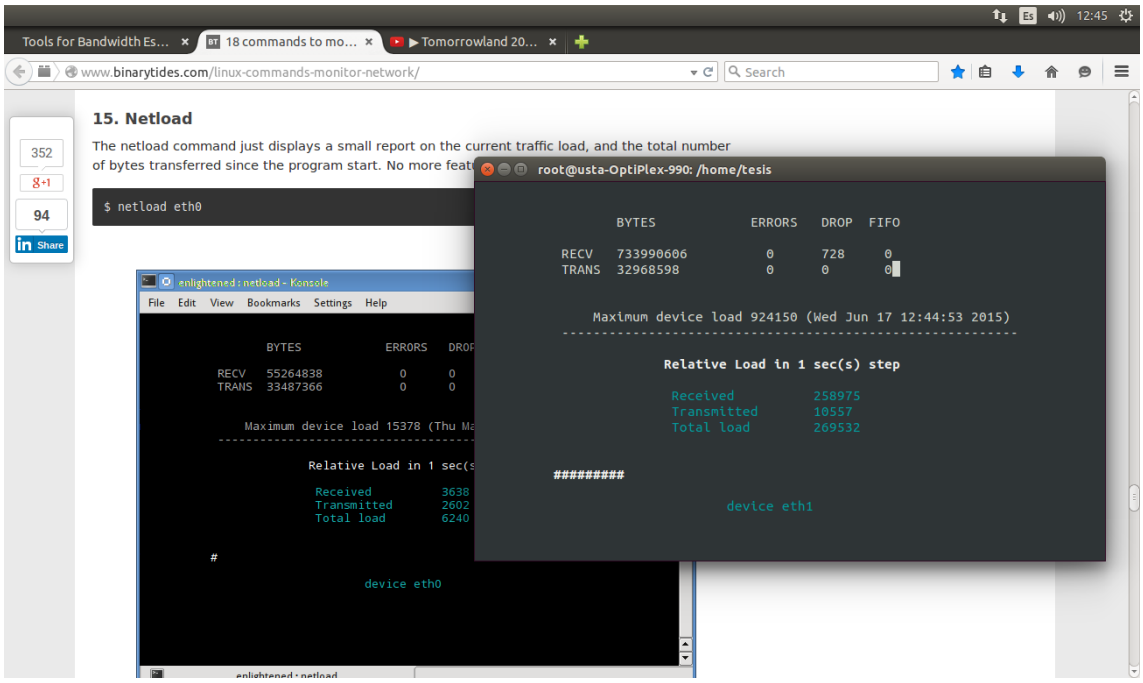

Figura 21. Ejecución del Netload® desde S.O. Linux.

Se observa en la figura 21, la tasa de bits tanto de transmisión como de recepción seguido de una suma de los mismos y un monitoreo de paquetes erróneos, fifo y drop.

7.1.11. Netwatch.

Es una herramienta que permite observar el tráfico de datos en una red a partir de un servidor local y uno remoto donde muestra capacidad del canal, trafico, identificación de direcciones en el momento del tráfico, entre otras.

A continuación en la figura 22 se evidencia el funcionamiento de la herramienta con una navegación paralela en Mozilla Firefox®; utilizando la eth1 y reproduciendo videos desde la red con el fin de encontrar como se congestiona el canal y el proceso de direccionamiento que existe en la comunicación.

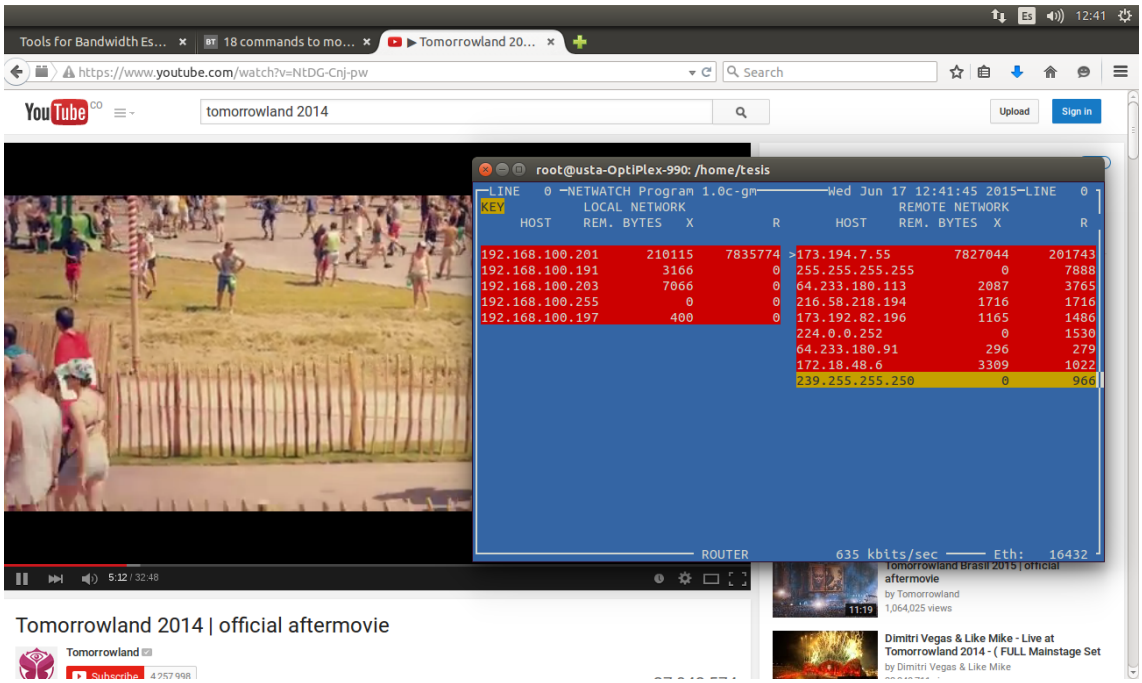

Figura 22. Ejecución del Netwatch® desde S.O. Linux.

En la figura 22 se observa cómo se genera tráfico congestionando el canal y así se evidencian las direcciones de origen y llegada seguido de una acumulación de los bits recibidos.

7.1.12. Pktstat.

Herramienta encargada para la estimación del ancho de banda donde muestra un average del canal en el instante de tiempo. De igual forma, realiza un seguimiento de las ips a donde se dirige la comunicación entre el local y su web server.

A continuación en la figura 23, se observa en la a carga de cada protocolo y a donde apunta la ip de inicio; de igual forma mostrando en intervalos de 0,5 segundos y los bits por segundo de cada muestra.

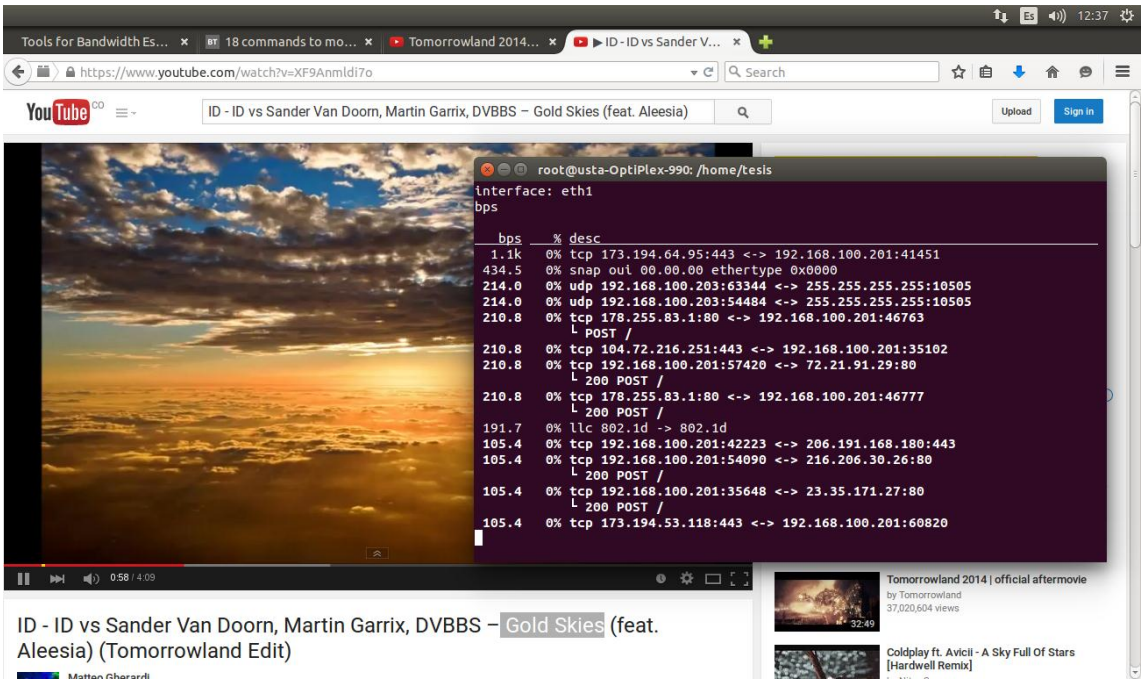

Figura 23. Ejecución del Pktstat® desde S.O. Linux.

7.1.13. CBM.

Es un herramienta interactiva para el usuario dado que recrea la interfaz con colores; sin embargo, solo muestra la transferencia de bits y luego la suma de los mismos. De igual manera, muestra la interfaz que se está trabajando con su respectiva ip.

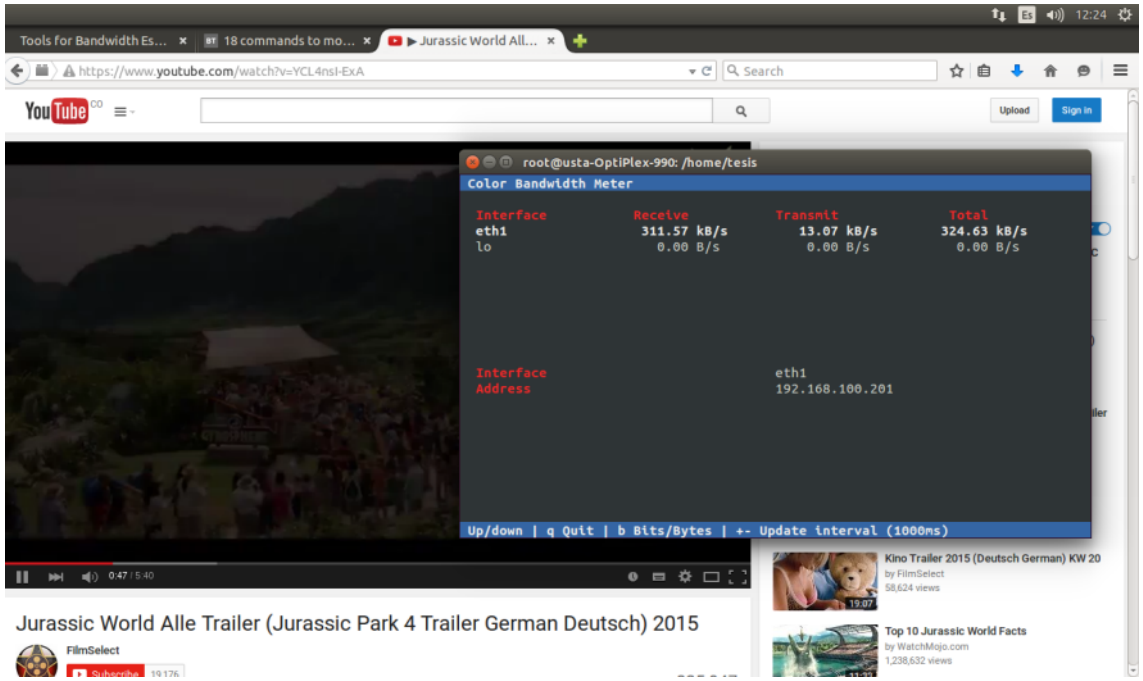

Figura 24. Ejecución del CBM® desde S.O. Linux.

Se observa en la figura 24, la interfaz eth1 y los bits tanto de entrada como de salida, seguidos de una suma de los mismos para así asumir cuanto canal está siendo utilizado; es de resaltar que la toma las está realizando cada 0,5 segundos.

7.2. Capítulo 2. Pruebas en Testbed de Traceband®, Iperf® e Ifstat® con resultados en Excel® y Matlab®.

Como se menciona al comienzo de la etapa de resultados, en este capítulo ya se trabajará con herramientas previamente seleccionadas luego de realizar la primera fase de indagación e investigación de múltiples aplicaciones que permitan la medición del ancho de banda. Por consiguiente, las herramientas a trabajar son Traceband®, Iperf® e Ifstat®.

Lo anterior se debe a que estos estimadores cumplieron dos referentes importantes: el primero, una investigación de aplicaciones de uso libre que permitieran evidenciar ancho de banda disponible (desarrollo del capítulo 1); y segundo factor a tener, las investigaciones llevadas a cabo por personas que intentan caracterizar el ancho de banda en una red y las aplicaciones que se utilizaron para tal fin.

Teniendo en cuenta los factores mencionados se procede a trabajar más detalladamente con estas tres herramientas con el fin de evidenciar su operabilidad en contextos diferentes (en términos de canal libre y generando cross-trafic) y analizando los datos entregados por las mismas para una caracterización más sólida del ancho de banda disponible buscando así resultados como Jitter, Overhead, Latencia y ancho de banda.

Posterior a esto, se realizará una seria de pruebas en el Testbed como se mencionó en la sección 5.7. Testbed; utilizando cuatro hosts con el fin de generar cross-trafic en dos de estos y en los restantes analizar el comportamiento de cada herramienta. Es de resaltar que las pruebas se trabajaron en 1, 5 y 15 minutos (60, 300 y 900 segundos correspondientemente) con el canal libre y posterior a esto generando tráfico desde Mgen®.

Finalmente, al concluir cada prueba se almacenan los datos de cada estimador en un .txt para luego ser estudiados, procesados en el software Excel® y aún más siendo finalmente parametrizados en el entorno de Matlab®.

Para un mejor entendimiento de este capítulo, en primer lugar se realizaron las pruebas en los tiempos anteriormente mencionados sin generar tráfico; con el fin de analizar el comportamiento cada herramienta. Seguido de esto, se generará tráfico sintético periódico desde Mgen® (remitirse a la sección 5.8. Mgen® para conocer cómo trabaja esta herramienta) en los mismos intervalos de tiempo que se realizan las pruebas sin tráfico; esta vez para evidenciar cómo se comporta el canal. A continuación se muestra el procedimiento de ejecución para generar

cross-trafic desde Mgen® aclarando que solamente se utilizará de forma periódica este tráfico.

Lo primero que se debe tener en cuenta es los comandos para la ejecución de Mgen®, aclarando que se necesitan de dos hosts.

En primer lugar, en el hosts de envío y desde terminal de Ubuntu se ejecuta el siguiente comando:

#mgen event "ON 1 UDP DST <ipaddress>/<port> <tipo de tráfico> [<# de paquetes> <cantidad de bits>]" INTERFACE eth0
(Salcedo Morillo, 2011)

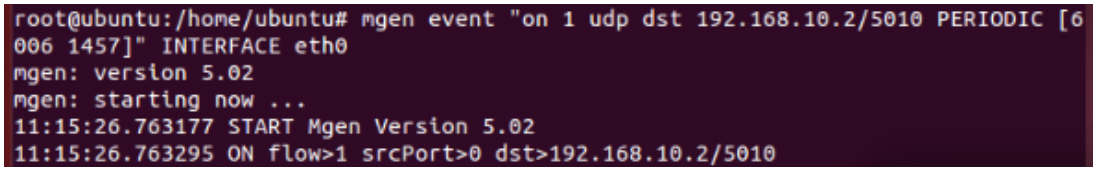

Figura 25. Ejecución de Mgen® desde hosts de envío con tráfico periódico.

Por otro lado, en el otro hosts se ejecuta el comando de recepción de tráfico que es el siguiente:

#mgen event “listen UDP <port> outputlog.drc”
(Salcedo Morillo, 2011)

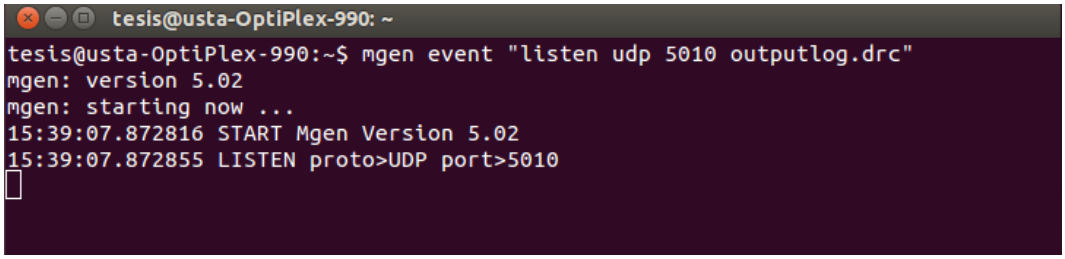

Figura 26. Enlace de Mgen® con el otro hosts evidenciando el puerto 5010.

Con lo anterior, se genera el tráfico sintético periódico dando un volumen a los paquetes que se enviaran en esta herramienta. A continuación, se evidencia una gráfica que muestra el comportamiento en un intervalo de tiempo de 300 segundos y denotando un promedio de 7.1 Mbps.

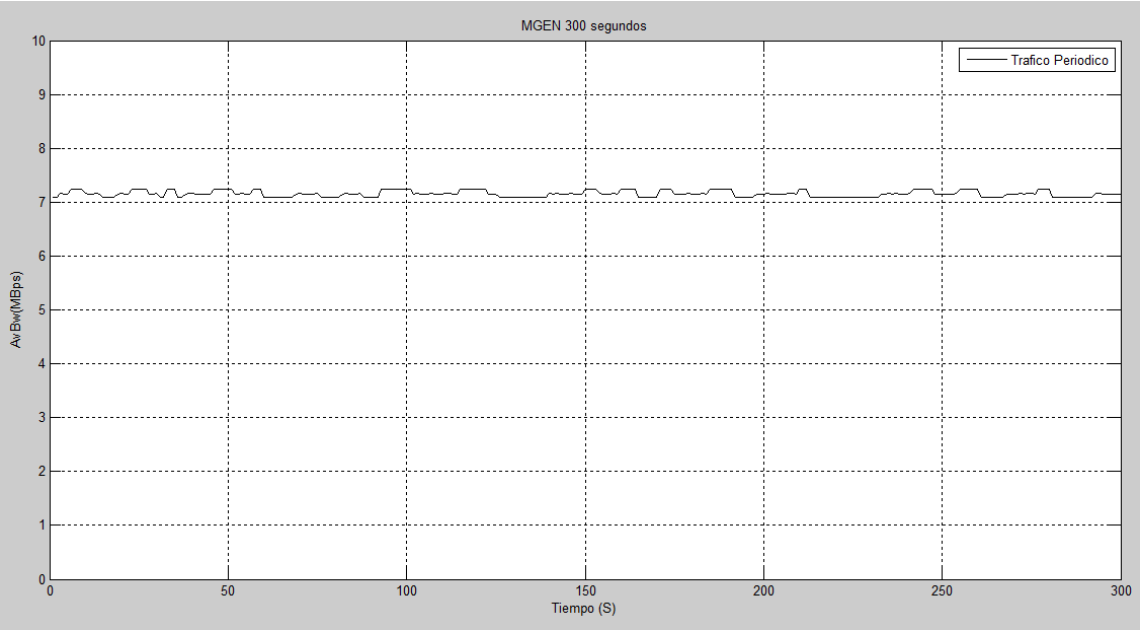

Figura 27. Grafica del tráfico inducido durante 300 segundos.

Teniendo en cuenta como se genera el tráfico y se procede a trabajar con cada estimador en el siguiente orden: Traceband®, Iperf® e Ifstat®.

7.2.1. Traceband®.

Para lograr evaluar los efectos del Cross-trafic sobre el ancho de banda disponible y demás variables; primero se seleccionaron 3 escenarios; primero sin tráfico con tiempo de 1 10 y 15 minutos, análogamente se realiza el mismo proceso pero teniendo en cuenta tráfico generado con Mgen®, dichos tiempos sugeridos por el director del proyecto con el fin de analizar el comportamiento en cada situación. El proceso de la información se logró haciendo el envío de paquetes entre los dos equipos con la aplicación instalada en las cuales se ejecuta el Traceband_rcv y Traceband_snd respectivamente, la cantidad de datos es diferente en cada muestra. (Guerrero & Labrador, Traceband: A fast, low overhead and accurate tool for available bandwidth estimation and monitoring, 2010)

A continuación se muestra el código que se debe escribir en el terminal para la ejecución de la herramienta:

#./traceband_snd –s 192.168.20.6 –C 10M –t 6000
#./traceband_rcv output.log
(Chaparro, Guerrero, & Fraile, 2013)

Donde el primer comando hace referencia al hosts de envío y el segundo comando se ubica en el hosts de recepción. Teniendo en cuenta estos dos comandos. A continuación en la figura 28 se muestra el comportamiento de la

herramienta desde el terminal ubicados en el hosts de envío teniendo en cuenta que cada estimación la realiza en aproximadamente 0.5 segundos.

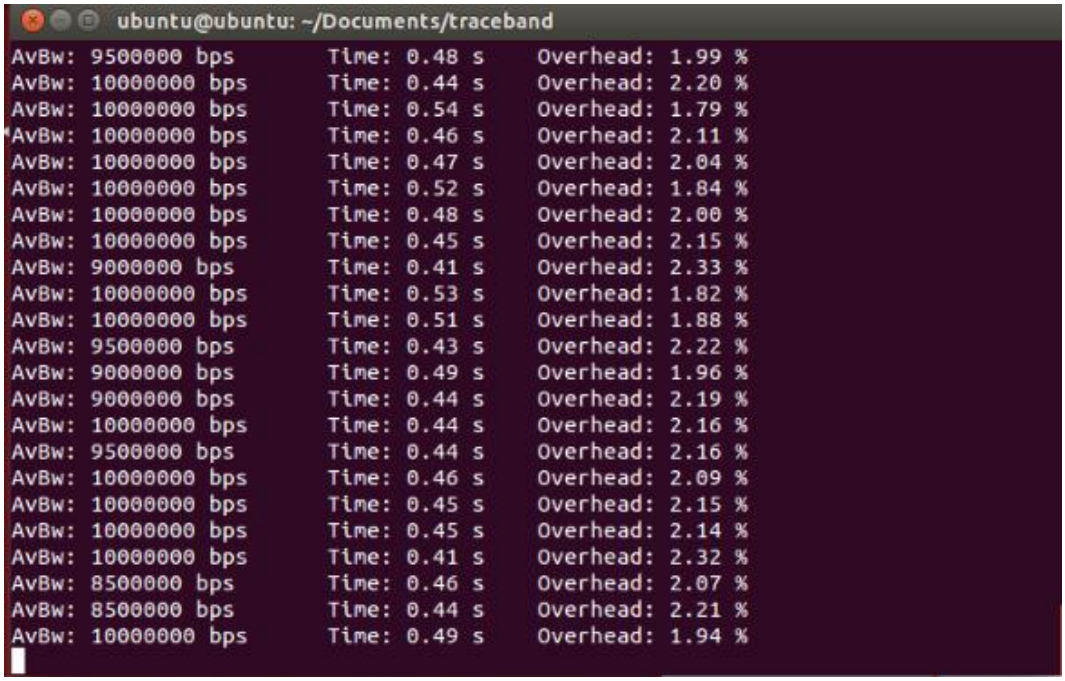

Figura 28. Traceband® ejecutándose desde el hosts de envío sin tráfico.

7.2.1.1. Traceband®, 60 segundos sin tráfico.

Para la medición, desde el host de envío se asigna un tiempo de muestreo de 60 segundos donde se limita el canal a 10 Megas y se evidencia que sin nada de tráfico se podría asumir que está a un 95% libre el canal. Luego de esto se procede a almacenar estos datos en un .txt para ser parametrizados en dos softwares como lo son Excel® y Matlab®.

TRACEBAND® 60 SEGUNDOS SIN TRÁFICO

AvBw (bps)	Tiempo (seg)	Overhead (%)	Intervalo (seg)
10000000	0.66	2.1	0.66
10000000	0.45	1.92	1.11
10000000	0.47	2.02	1.58
9500000	0.46	1.79	2.04
10000000	0.5	2.09	2.54
9500000	0.48	2.03	3.02
10000000	0.53	2.13	3.55
10000000	0.39	1.95	3.94
9000000	0.47	1.98	4.41
9500000	0.46	2.13	4.87

10000000	0.46	1.73	5.33
10000000	0.5	2.05	5.83
10000000	0.39	1.95	6.22
10000000	0.41	2.15	6.63
10000000	0.5	2.02	7.13
9500000	0.43	1.79	7.56
9500000	0.49	2.09	8.05
9000000	0.51	2.15	8.56
10000000	0.46	2.02	9.02
10000000	0.47	1.79	9.49
9500000	0.45	2.05	9.94

Tabla 4. Resultados Traceband® desde Excel® durante 60 seg. sin tráfico.

Se observa en la tabla 4, que los datos que fueron adquiridos desde el terminal del Linux se guardan en un.txt y posterior a esto se llevan a Excel® donde se empieza la parametrización de los mismos; en el Anexo 1 se observa el total de muestras durante el tiempo estimado. Es de resaltar, que a partir de los datos que entregó la herramienta Traceband® se busca encontrar la capacidad máxima del canal, el ABw con y sin cross-trafic, el jitter.

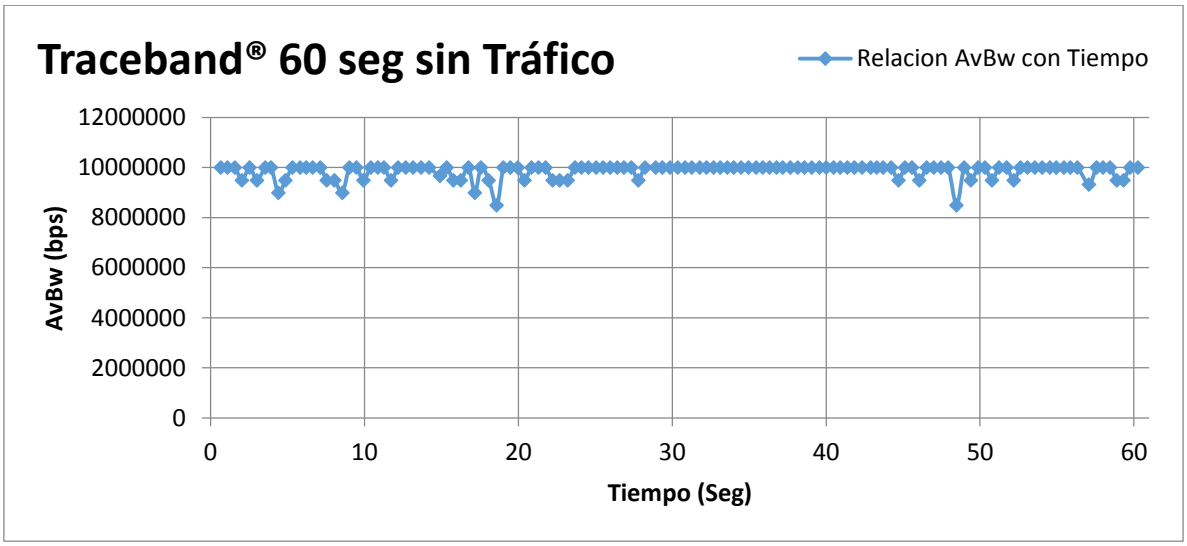

Figura 29. Traceband® desde Excel® durante 60 seg. sin tráfico.

En la figura 29 se evidencia el ABw disponible cuando no existe nada de tráfico en el Testbed; teniendo en cuenta que se limitó el canal a 10 Megas. Es de resaltar que el ABw promedio es de 9,8 MBps, con un promedio de Overhead de 2% y un total de 128 muestras. Luego de tener los datos en Excel® se procede a importar este archivo a Matlab® y a partir de un breve programa se realiza la respectiva visualización de los datos desde este entorno.

```
%//////////////////TRACEBAND® 60 SEGUNDOS SIN TRÁFICO//////////////////////

fileName = ('1Min_ST.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,5);         %tiempo de muestra 60 segundos
y = a(:,1);         %ancho de banda disponible
w = a(:,3);         %porcentaje de overhead

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA/////////

subplot(2,1,1);                 %posicion de la grafica en el marco general
z = plot(x,y,'g')                   %grafica ancho de banda con respecto a tiempo
axis([0 61 0 10200000])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw(bps)')
title('Traceband® 60 segundos sin tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA DEL % DE OVERHEAD/////////

subplot(2,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica porcentaje de overhead con respecto a tiempo
axis([0 61 1.5 2.5])            %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('Porcentaje de overhead')
title('Traceband® 60 segundos sin tráfico')
grid on
legend('Overhead vs. Time')
```

Figura 30. m-file en Matlab® para Traceband® durante 60 seg. sin tráfico.

Como se observa anteriormente, el m-file creado en Matlab® para lograr una visualización de los datos que brinda la herramienta en la estimación de ancho de banda. Posterior a esto, se muestra el resultado a partir de gráficas del ancho de banda y overhead.

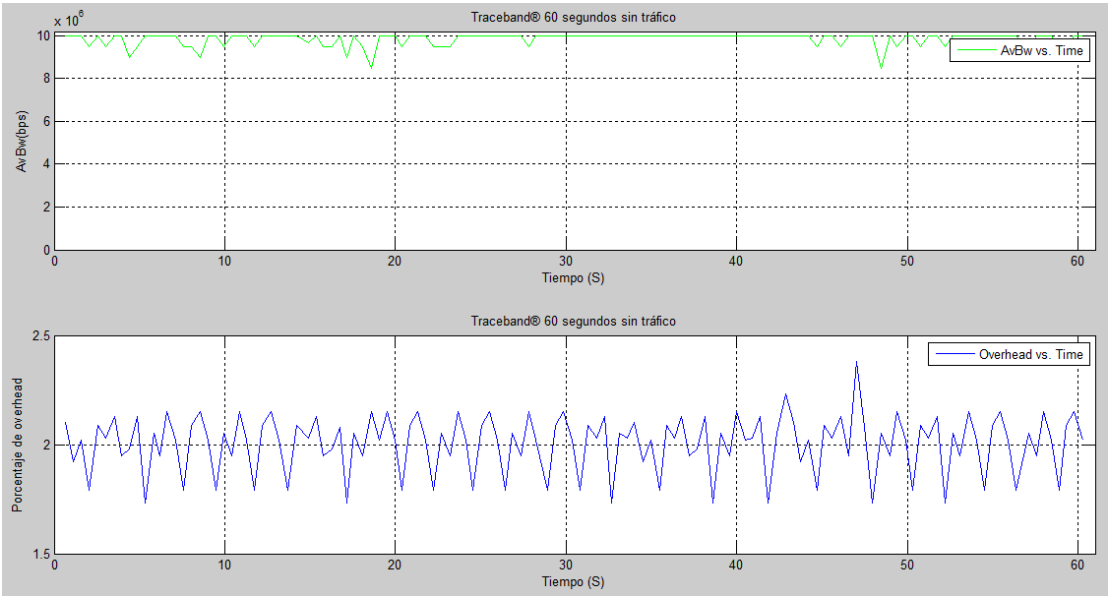

Figura 31. Traceband® desde Matlab® durante 60 seg. sin tráfico.

En la figura 31 se evidencia en la parte superior el ancho de banda disponible; resaltando que el canal está limitado a 10 Megas y se observa que prácticamente es válido que sin tráfico la línea verde permanece muy aproximada a los 10 dado que existen procesos de cache y memoria que ocupan parte pequeña del canal. En la parte inferior se muestra el porcentaje de overhead con una la línea azul. Finalmente, es bueno aclarar que al no generar este tráfico, por consiguiente el jitter y la latencia son de procesos de memoria; es más con el fin de analizar el comportamiento de la herramienta con respecto al tiempo.

7.2.1.2. Traceband®, 300 segundos sin tráfico.

De igual forma que en la prueba anterior de 60 segundos sin tráfico, se almacenaron los datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 2 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

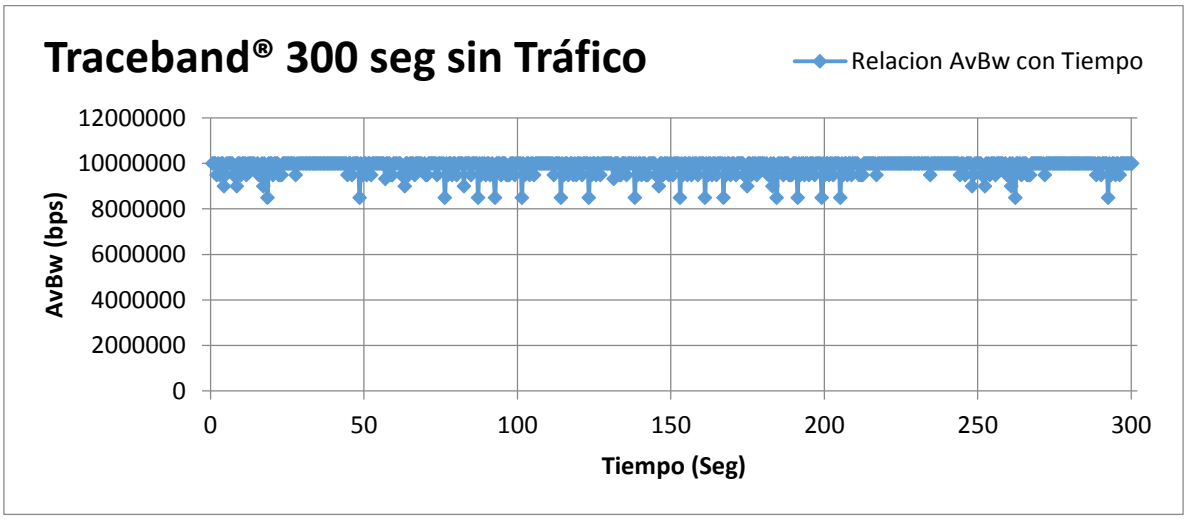

Figura 32. Traceband® desde Excel® durante 300 seg. sin tráfico.

Se observa anteriormente en la figura 32 el ABw durante los 5 minutos de prueba, es decir, 300 segundos y se concluye que el canal también se aproxima a los 10 Megas de su capacidad total y dado que existen diferentes procesos de caché en se tiene un ABw promedio de 9,8 MBps, con un overhead promedio de 2% y un total de muestras de 636. Posterior a esto, se muestra el comportamiento de estos datos en el entorno de Matlab®.

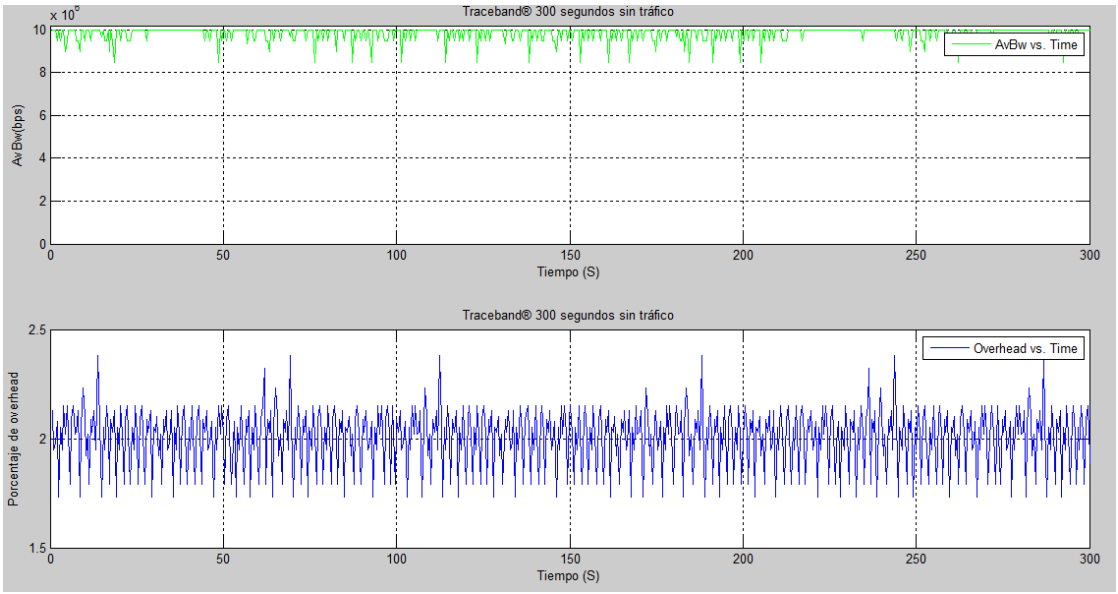

Figura 33. Traceband® desde Matlab® durante 300 seg. sin tráfico.

Se puede evidenciar en el entorno de Matlab®, en la parte superior el ABw disponible en color verde y en la parte inferior el porcentaje de overhead de cada muestra (de color azul). En el Anexo 3 se puede observar el m-file que se utilizó para esta prueba.

7.2.1.3. Traceband®, 900 segundos sin tráfico.

Finalmente, dado que anteriormente se resaltó los tiempos de pruebas de la herramienta, se procede a analizarla durante 15 minutos o su equivalente que son 900 segundos; entonces, los datos se almacenaron en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 4 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®. En la figura 34 se aprecia el ABw desde el software Excel®.

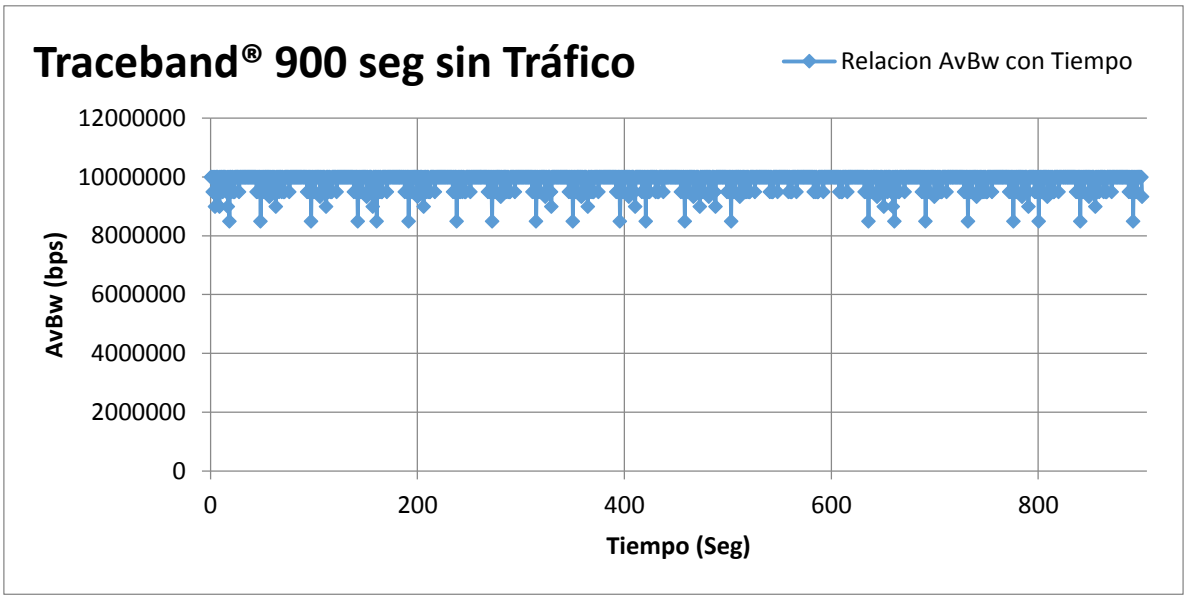

Figura 34. Traceband® desde Excel® durante 900 seg. sin tráfico.

Como se observa anteriormente, se puede considerar el ABw disponible durante el tiempo estimado de prueba (900 segundos), y se concluye que el ancho está casi en su total de disponibilidad que son las 10 Megas a las que está el canal. Es de resaltar que el overhead promedio se encuentra en 2% con un ABW promedio de 9,8 MBps y 1910 muestras.

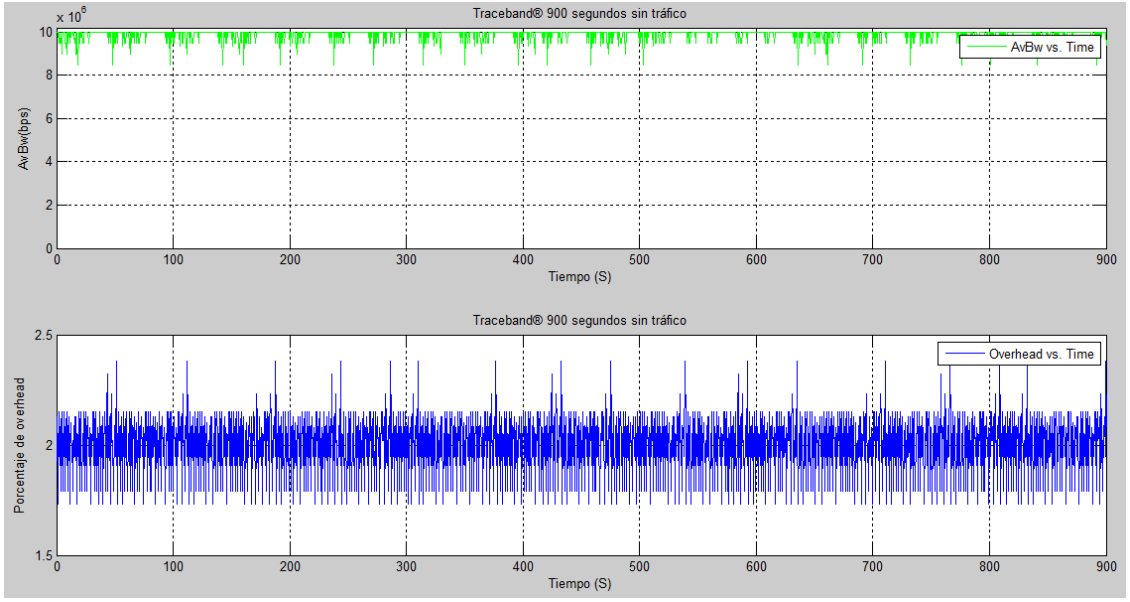

Figura 35. Traceband® desde Matlab® durante 900 seg. sin tráfico.

En la figura 35, se evidenciar en el entorno de Matlab® la respuesta de los datos que se importaron desde el software Excel® y en el Anexo 5 se puede observar más detalladamente el m-file de esta prueba. Por consiguiente en la parte superior

el ABw disponible en color verde y en la parte inferior el porcentaje de overhead durante el tiempo transcurrido de la prueba (de color azul); que se hace referencia a procesos de memoria propios del sistema.

Como se mencionó anteriormente en el apartado de los 60 segundos sin tráfico con Traceband®, el análisis sin cross-trafic en el Testbed es para analizar el comportamiento de la herramienta; ahora, se procede a realizar las pruebas en los mismos instantes de tiempo pero esta vez induciendo tráfico en el canal.

7.2.1.4. Traceband®, 60 segundos con tráfico.

Para la medición, es necesario desde el Testbed dos hosts sean los encargados de estar registrando los datos de la herramienta estimadora, para este caso Traceband®, y por otro lado se utilizan dos hosts cuya función será la de generar el tráfico, para este caso sintético generado desde una herramienta llamada Mgen®. Es de resaltar que Mgen® permite diferentes clases de tráfico pero en este caso se trabajará con tráfico periódico en los diferentes tiempos de muestreo predefinidos en la parte inicial de este apartado. Posterior, se procede a tomar lectura de los datos y estos ser almacenados en un .txt y ser parametrizados desde Excel® y Matlab®.

A continuación en la tabla 5 se muestra algunos datos ya importados en el entorno de Excel®; aunque en el Anexo 6 se puede observar todos los datos en el intervalo de tiempo que se analizaron.

TRACEBAND® 1 MINUTO CON TRAFICO

AvBw (bps)	Tiempo (seg)	Overhead (%)	Intervalo (seg)
4675325	0.46	2.03	0.46
4800000	0.53	2.1	0.99
7000000	0.47	1.92	1.46
7800000	0.43	2.02	1.89
9800000	0.54	1.79	2.43
8200000	0.76	2.09	3.19
6200000	0.64	2.03	3.83
7400000	0.45	2.13	4.28
7600000	0.55	1.95	4.83
7600000	0.51	1.98	5.34
9800000	0.45	2.08	5.79
7800000	0.76	1.73	6.55
8600000	0.58	2.05	7.13
7400000	0.56	1.95	7.69
8000000	0.49	2.15	8.18
7800000	0.54	2.02	8.72
8400000	0.55	1.79	9.27
8400000	0.46	2.09	9.73

7400000	0.53	2.03	10.26
8400000	0.47	2.13	10.73
8000000	0.43	1.73	11.16

Tabla 5. Resultados Traceband® desde Excel® durante 60 seg. con tráfico.

Se muestra en la tabla 5, los intervalos de 0.5 segundos donde el estimador realiza la toma de dato y seguido de esto, Traceband® muestra el ancho de banda que queda disponible junto con el overhead. Posterior a esto, se procede a graficar desde Excel® el ABw que hay disponible con respecto a la capacidad máxima del canal que son 10 Megas.

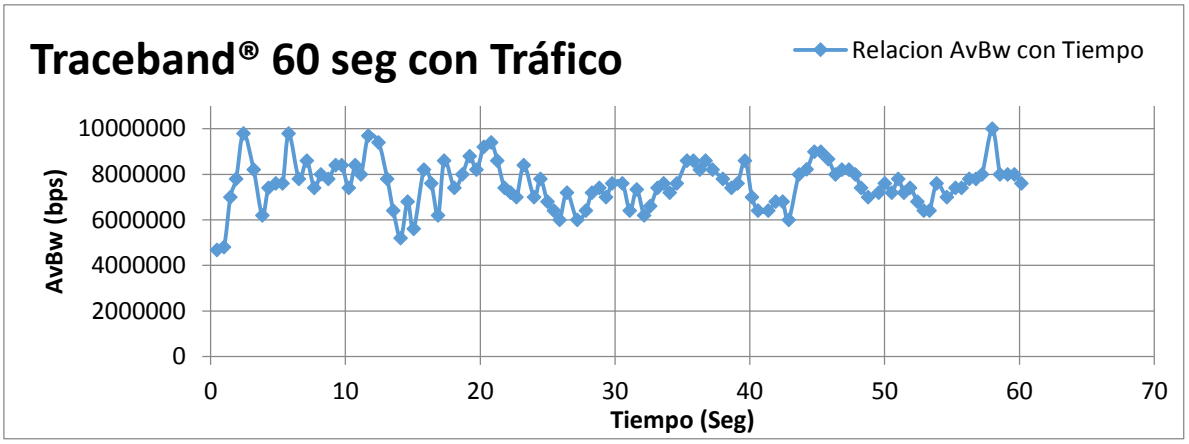

Figura 36. Traceband® desde Excel® durante 60 seg. con tráfico.

Se puede observar en la figura 36, como se reduce el Ancho de Banda en un promedio de 7,5 MBps; por lo que se puede decir que si la capacidad máxima del canal es de 10 Megas y se están utilizando aproximadamente 2,5 Megas del canal ocupados por procesos de caché y el tráfico sintético. Es de resaltar que el overhead promedio esta en aproximadamente 2.01%.

Ahora, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw. A continuación en la figura 37 se muestra el m-file creado para generar las diferentes gráficas del estudio del estimador Traceband® trabajado con tráfico sintético desde Mgen®.

```
%//////////////////TRACEBAND® 60 SEGUNDOS CON TRÁFICO//////////////////////

fileName = ('1Min_CT.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,5);         %tiempo de muestra 60 segundos
y = a(:,1);         %ancho de banda disponible
w = a(:,3);         %porcentaje de overhead

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA/////////

subplot(2,1,1);                 %posicion de la grafica en el marco general
z = plot(x,y,'g')                   %grafica ancho de banda con respecto a tiempo
axis([0 61 0 10200000])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw(bps)')
title('Traceband® 60 segundos con tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA DEL % DE OVERHEAD/////////

subplot(2,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica porcentaje de overhead con respecto a tiempo
axis([0 61 1.5 2.5])            %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('Porcentaje de overhead')
title('Traceband® 60 segundos con tráfico')
grid on
legend('Overhead vs. Time')
```

Figura 37. M-file en Matlab® para Traceband® durante 60 seg. con tráfico.

Se puede observar el m-file donde se importa el archivo de Excel® que posee los datos adquiridos del .txt de la herramienta estimadora (Traceband®); y luego de esto se genera a partir de código las gráficas de ABw y Overhead.

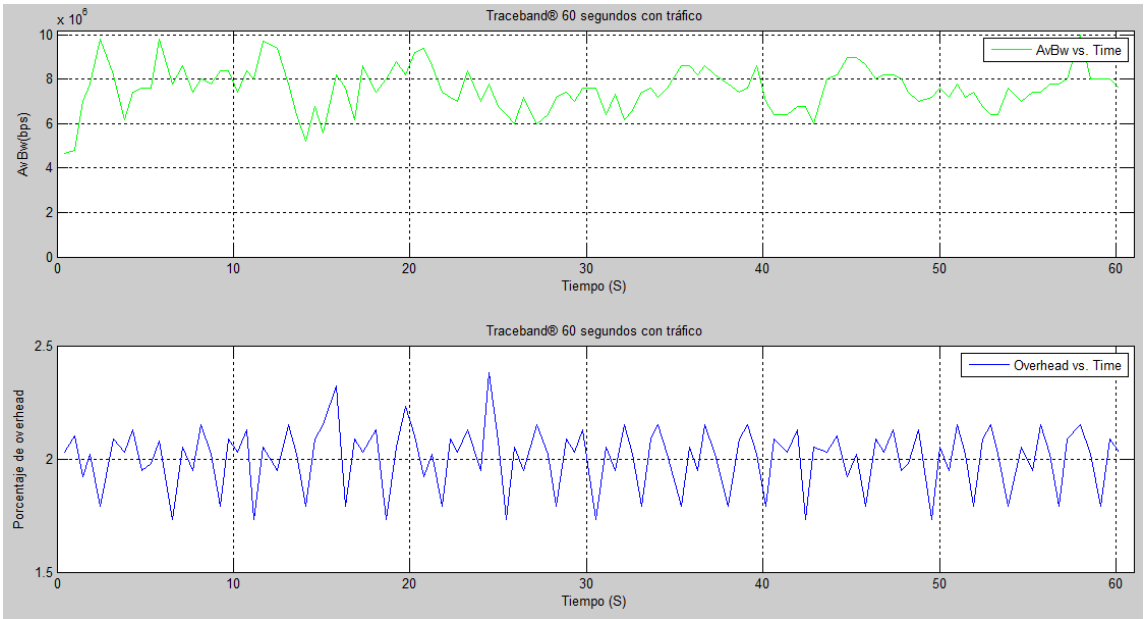

Figura 38. Traceband® desde Matlab® durante 60 seg. con tráfico.

En la figura 38, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente; y en la parte inferior y resaltado de un color azul el porcentaje de overhead de cada estimación. Es de aclarar, que estas dos imágenes se trabajaron en un periodo de 60 segundos a intervalos de 0,5 segundos cada estimación.

7.2.1.5. Traceband®, 300 segundos con tráfico.

De igual forma que en la prueba anterior de 60 segundos con tráfico, se almacenaron los datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 7 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 300 segundos como periodo de muestra y en el Anexo 8 se puede observar el m-file creado para esta prueba.

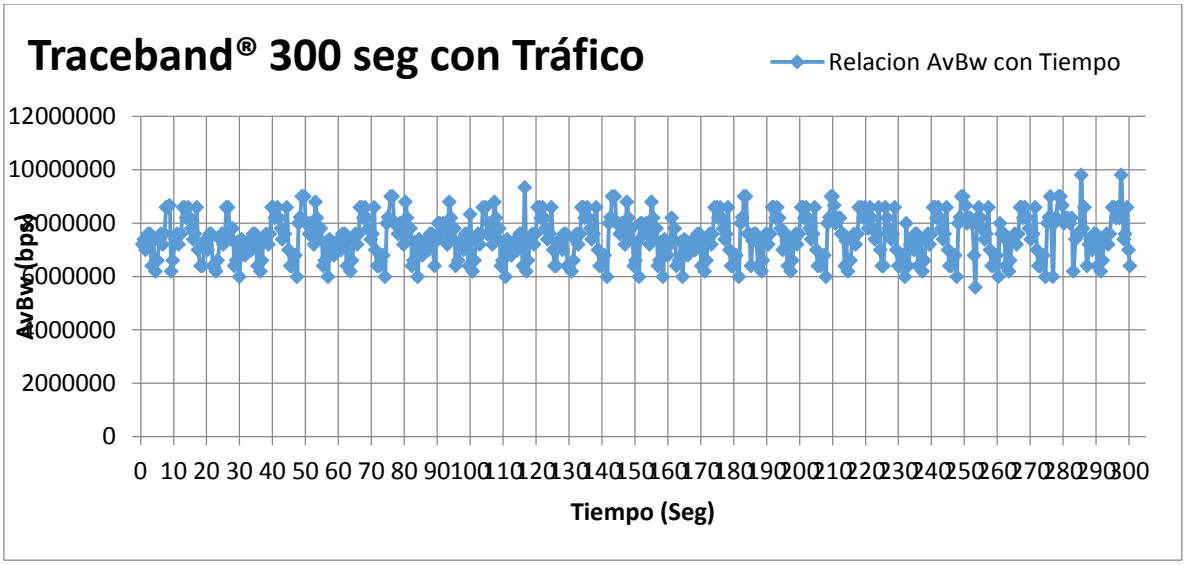

Figura 39. Traceband® desde Excel® durante 300 seg. con tráfico.

Se puede observar en la figura 39, que el ABw se reduce aproximadamente igual que en la prueba de 60 segundos; por los que se concluye que el canal también se reduce a un promedio de 7,4 MBps; por lo que se puede decir que si la capacidad máxima del canal es de 10 Megas y se están utilizando aproximadamente 2,6 Megas del canal ocupados por procesos de caché y el tráfico sintético. Es de resaltar que el overhead promedio está en aproximadamente 2%. Posterior a esto, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw.

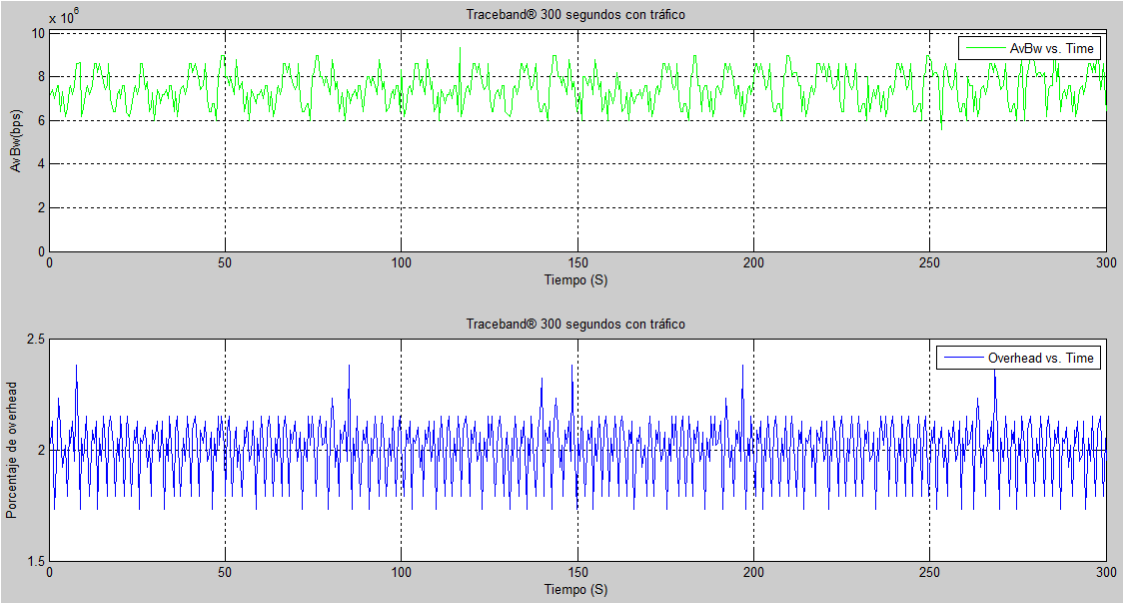

Figura 40. Traceband® desde Matlab® durante 300 seg. con tráfico.

En la figura 40, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente y en la parte inferior y resaltado de un color azul el porcentaje de overhead de cada estimación. Es de aclarar, que estas dos imágenes se trabajaron en un periodo de 300 segundos a intervalos de 0,5 segundos cada estimación.

7.2.1.6. Traceband®, 900 segundos con tráfico.

Finalmente, la última prueba que se realiza es de 15 minutos (900 segundos) y se realiza el mismo procedimiento que con las pruebas anteriores con tráfico; que son almacenar datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 9 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 900 segundos como periodo de muestra y en el Anexo 10 se puede observar el m-file creado para esta prueba.

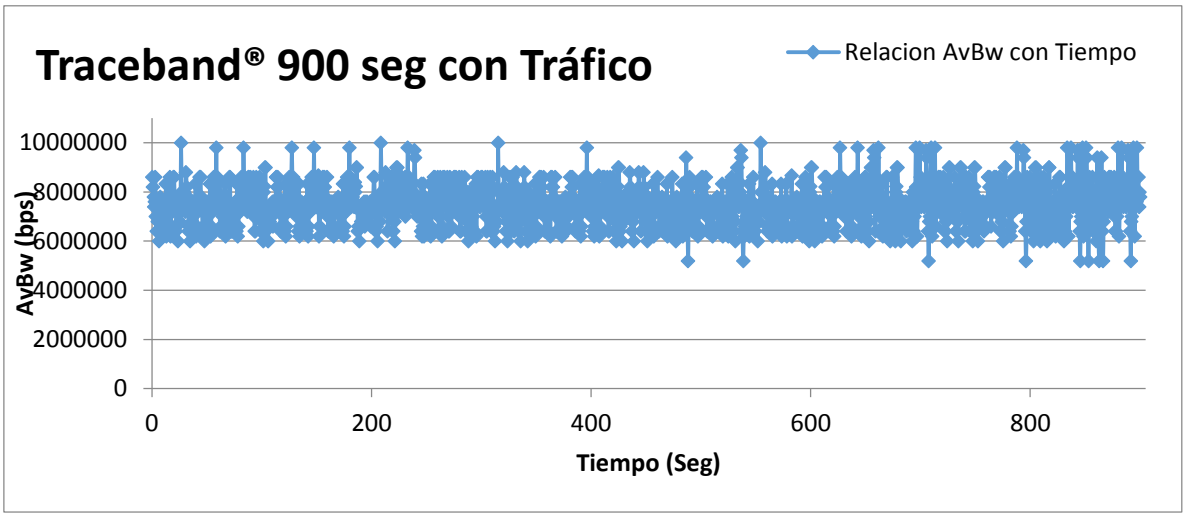

Figura 41. Traceband® desde Excel® durante 900 seg. con tráfico.

Se puede observar en la figura 41, que el ABw se reduce aproximadamente igual que en la prueba de 60 segundos; promedio de 7,4 MBps; por lo que se puede decir que si la capacidad máxima del canal es de 10 Megas y se están utilizando aproximadamente 2,6 Megas del canal ocupados por procesos de caché y el tráfico sintético. Es de resaltar que el overhead promedio esta en aproximadamente 2%. Posterior a esto, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw.

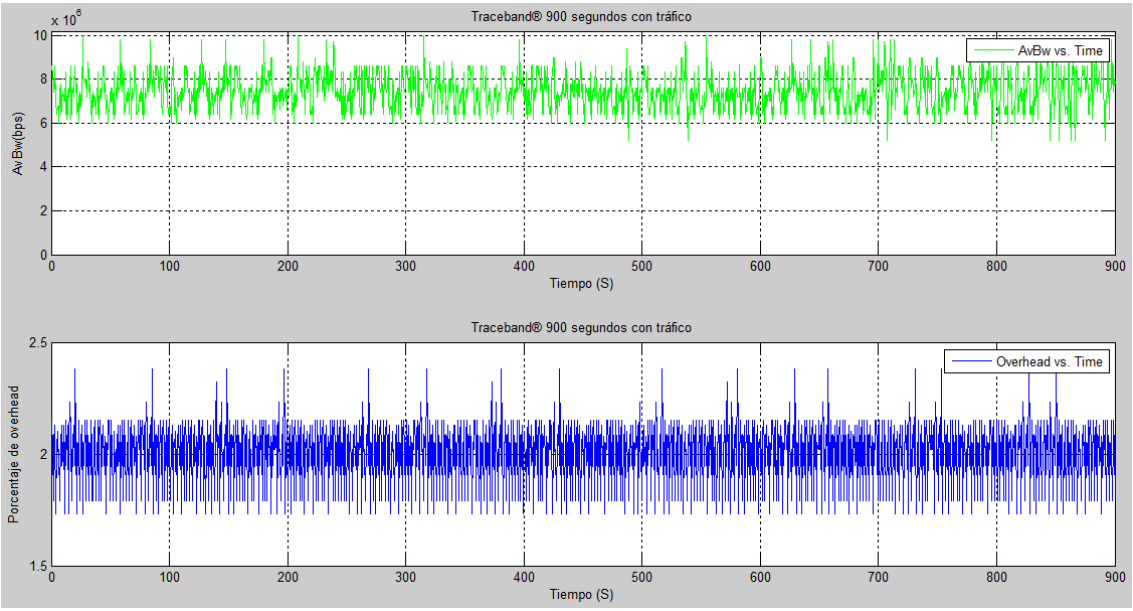

Figura 42. Traceband® desde Matlab® durante 900 seg. con tráfico.

En la figura 42, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente y en la parte inferior y resaltado de un color azul el porcentaje de

overhead de cada muestra en el intervalo de tiempo. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 900 segundos a intervalos de 0,5 segundos cada estimación.

7.2.2. Iperf®.

Dado que esta herramienta brinda una gran fiabilidad en los diferentes datos entregados y posterior a esto ser parametrizados en los mismos intervalos de tiempo que se utilizó en la herramienta anterior (1, 5, 15 minutos); es de resaltar que para esta aplicación se utilizó el Testbed y se evidenció con y sin tráfico.

Para lograr la ejecución de esta herramienta es necesario utilizar dos hosts ubicados a los extremos del Testbed y teniendo en cuenta que uno actuará como sender y el otro como revsender. Posterior a esto, se habilita la herramienta con permisos de ejecución desde comandos, con el fin de que se active la herramienta con sus librerías y se pueda ejecutar las dos aplicaciones de envío y recepción.

Ahora, el código de ejecución del Iperf® es el siguiente:

#iperf –c 192.168.20.6 –p 1 –i 1 –P 5001 –f k –t 60
#iperf –s 192.168.10.2
(Steenkiste, 2013)

Donde el primer comando hace referencia al hosts de envío y el Segundo comando se ubica en el hosts de recepción. Teniendo en cuenta estos dos comandos, se realizan las respectivas pruebas sin y con tráfico aclarando que la toma de cada muestra es en intervalos de un segundo. A continuación se mostrará los resultados obtenidos en un orden de tiempo desde 60 segundos hasta 900 segundos, primero sin tráfico y luego se repetirá el proceso con congestión en el canal. A continuación en la figura 43 se muestra el comportamiento de la herramienta desde el terminal ubicados en el hosts de envío teniendo en cuenta que cada estimación la realiza en intervalos de 1 segundo.

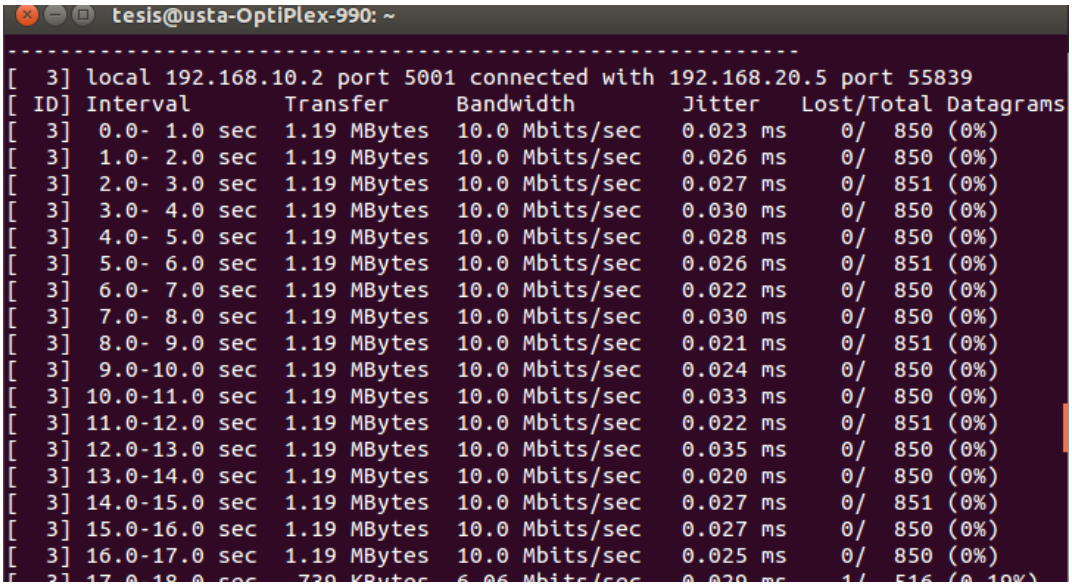

Figura 43. Iperf® ejecutándose desde el hosts de envío sin tráfico.

7.2.2.1. Iperf®, 60 segundos sin tráfico.

Para la medición, desde el host de envío se asigna un tiempo de muestreo de 60 segundos donde se limita el canal a 10 Megas y se evidencia que sin nada de tráfico se podría asumir que está a un 95% libre el canal. Luego de esto se procede a almacenar estos datos en un .txt para ser parametrizados en dos softwares como lo son Excel® y Matlab®.

IPERF® 60 Segundos sin Tráfico

Intervalo	Transferencia (KBtes)	AvBw (MB/seg)
1	0	10
2	0	9.2
3	0	9.1
4	0	9.2
5	0	9.5
6	0	9.1
7	0	9.1
8	0	9.2
9	0	9.2
10	0	9.5
11	0	9
12	0	9.4
13	0	9.1
14	0	9.4
15	0	9
16	0	9.5

17	0	9
18	0	9.1
19	0	9.1
20	0	9.6

Tabla 6. Resultados Iperf® desde Excel® durante 60 seg. sin tráfico.

Se observa en la tabla 6, que los datos que fueron adquiridos desde el terminal del Linux se guardan en un.txt y posterior a esto se llevan a Excel® donde se empieza la parametrización de los mismos; en el Anexo 11 se observa el total de muestras durante el tiempo estimado. Es de resaltar, que a partir de los datos que entregó la herramienta Iperf® se busca encontrar la capacidad máxima del canal, el ABw con y sin cross-trafic, el jitter.

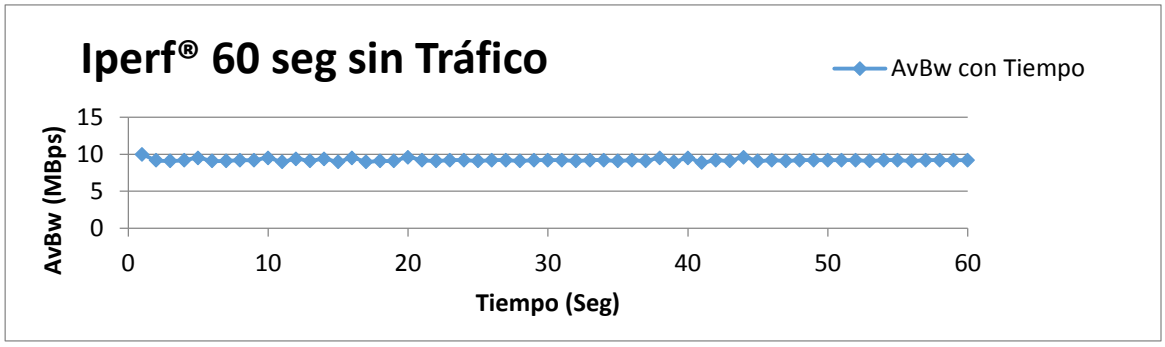

Figura 44. Iperf® desde Excel® durante 60 seg. sin tráfico.

En la figura 44 se evidencia el ABw disponible cuando no existe nada de tráfico en el Testbed; teniendo en cuenta que se limitó el canal a 10 Megas. Luego de tener los datos en Excel® se procede a importar este archivo a Matlab® y a partir de un breve programa se realiza la respectiva visualización de los datos desde este entorno.

```
%//////////////////IPERF® 60 SEGUNDOS SIN TRAFICO//////////////////////

fileName = ('60_ST.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,1);         %tiempo de muestra 60 segundos
y = a(:,3);         %ancho de banda disponible
w = a(:,2);         %transferencia de bits

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA////////

subplot(2,1,1);                 %posicion de la grafica en el marco general
z = plot(x,y,'g')                   %grafica ancho de banda con respecto a tiempo
axis([0 60 0 10.2])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw (MBps)')
title('Iperf® 60 segundos sin tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA DE LA TRANSFERENCIA////////

subplot(2,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica porcentaje de overhead con respecto a tiempo
axis([0 60 -2 10.2])            %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('Transferencia (Kbps)')
title('Iperf® 60 segundos sin tráfico')
grid on
legend('Tráfico vs. Time')
```

Figura 45. M-file desde Matlab® para Iperf® durante 60 seg. sin tráfico.

Como se observa anteriormente, el m-file creado en Matlab® para lograr una visualización de los datos que brinda la herramienta en la estimación de ancho de banda. Posterior a esto, se muestra el resultado a partir de gráficas del ancho de banda y la tasa de bits de transferencia.

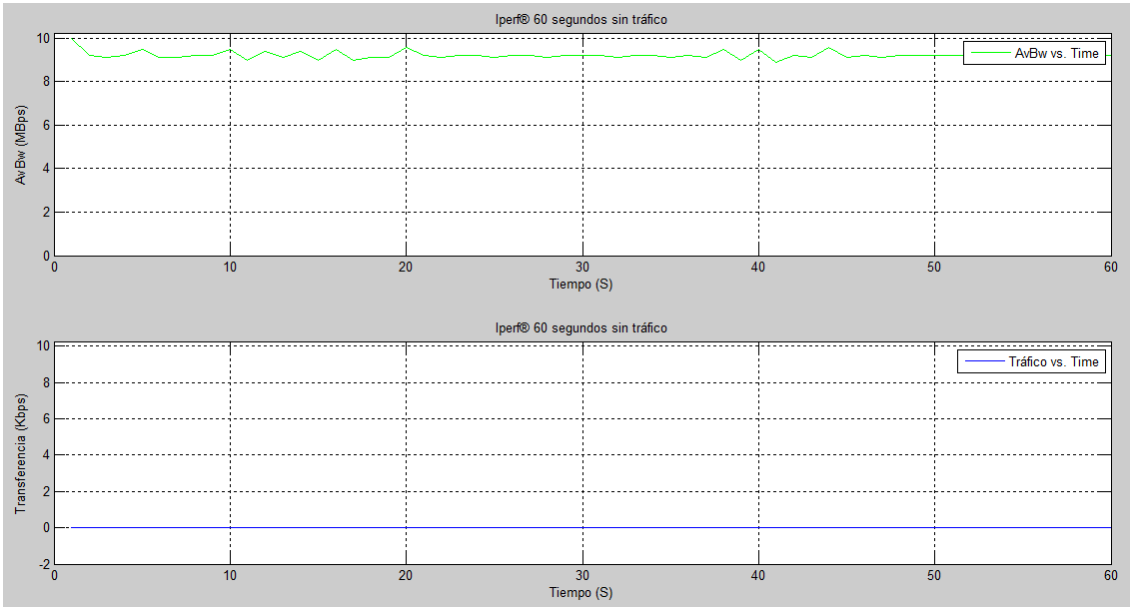

Figura 46. Iperf® desde Matlab® durante 60 seg. sin tráfico.

En la figura 46 se evidencia en la parte superior el ancho de banda disponible; resaltando que el canal está limitado a 10 Megas y se observa que prácticamente es válido que sin tráfico la línea verde permanece muy aproximada a los 10 dado que existen procesos de cache y memoria que ocupan parte pequeña del canal. En la parte inferior se muestra el tráfico o para más claridad los bits de transferencia entre los hosts y dado que la prueba se realizó sin generar cross-trafic; entonces la línea azul permanece en 0 bits de transferencia. Finalmente, es bueno aclarar que al no generar este tráfico, por consiguiente no existe un jitter ni latencia en esta etapa; es más con el fin de analizar el comportamiento de la herramienta con respecto al tiempo.

7.2.2.2. Iperf®, 300 segundos sin tráfico.

De igual forma que en la prueba anterior de 60 segundos sin tráfico, se almacenaron los datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 12 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

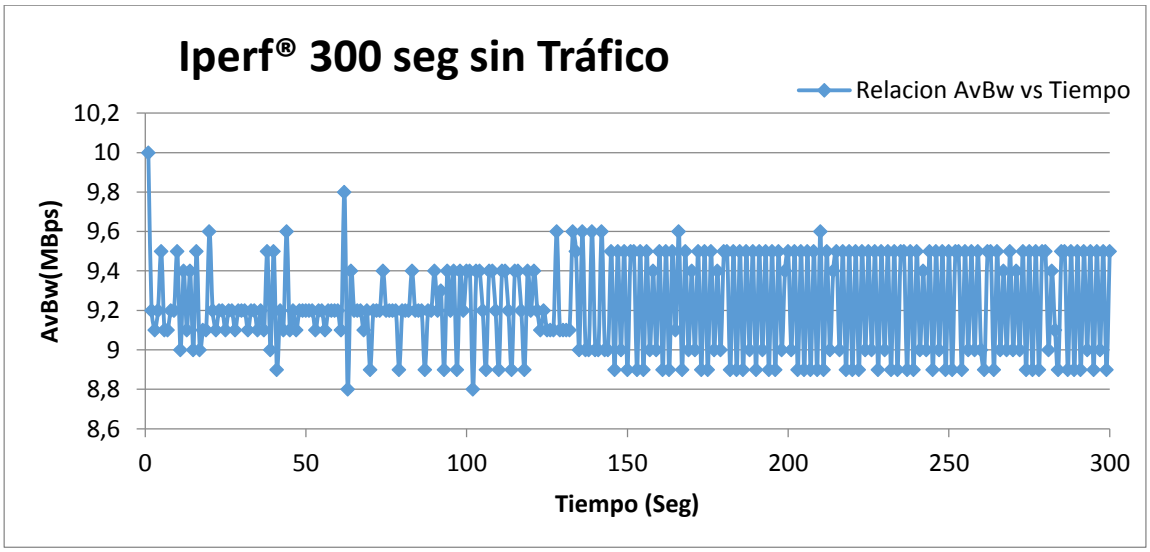

Figura 47. Iperf® desde Excel® durante 300 seg. sin tráfico.

Se observa anteriormente en la figura 47 el ABw durante los 5 minutos de prueba, es decir, 300 segundos y se concluye que el canal también se aproxima a los 10 Megas de su capacidad total y dado que existen diferentes procesos de caché en momentos aparece el ancho de banda en un mínimo de 8,8 de disponibilidad. Posterior a esto, se muestra el comportamiento de estos datos en el entorno de Matlab®.

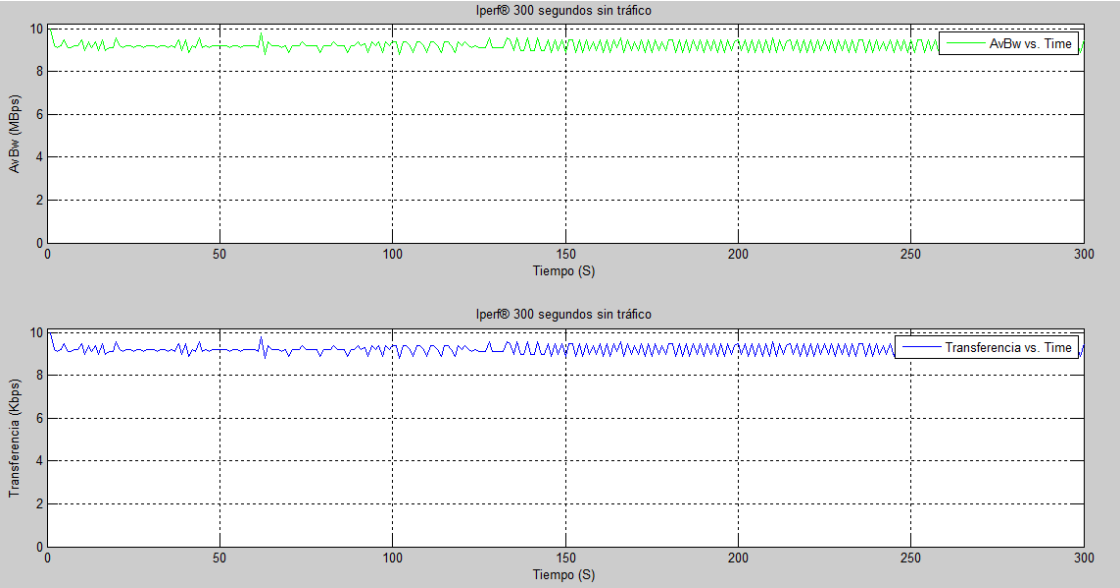

Figura 48. Iperf® desde Matlab® durante 300 seg. sin tráfico.

Se puede evidenciar en el entorno de Matlab®, en la parte superior el ABw disponible en color verde y en la parte inferior una tasa despreciable de transferencia en Kbps (de color azul); que se hace referencia a procesos de memoria propios del sistema. En el Anexo 13 se puede observar el m-file que se utilizó para esta prueba.

7.2.2.3. Iperf®, 900 segundos sin tráfico.

Finalmente, dado que anteriormente se resaltó los tiempos de pruebas de la herramienta, se procede a analizarla durante 15 minutos o su equivalente que son 900 segundos; entonces, los datos se almacenaron en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 14 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®. En la figura 49 se aprecia el ABw desde el software Excel®.

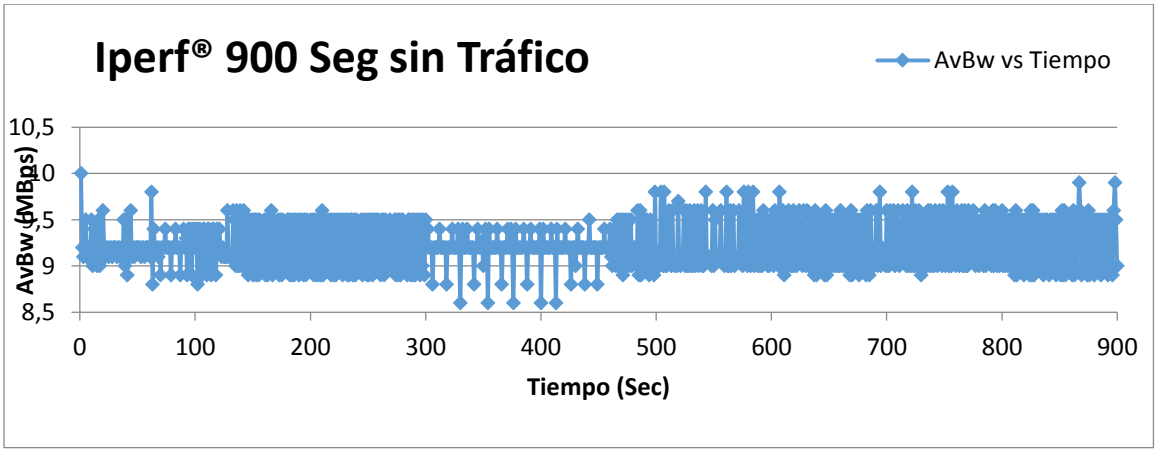

Figura 49. Iperf® desde Excel® durante 900 seg. sin tráfico.

Como se observa anteriormente, se puede considerar el ABw disponible durante el tiempo estimado de prueba (900 segundos), y se concluye que el ancho está casi en su total de disponibilidad que son las 10 Megas a las que está el canal.

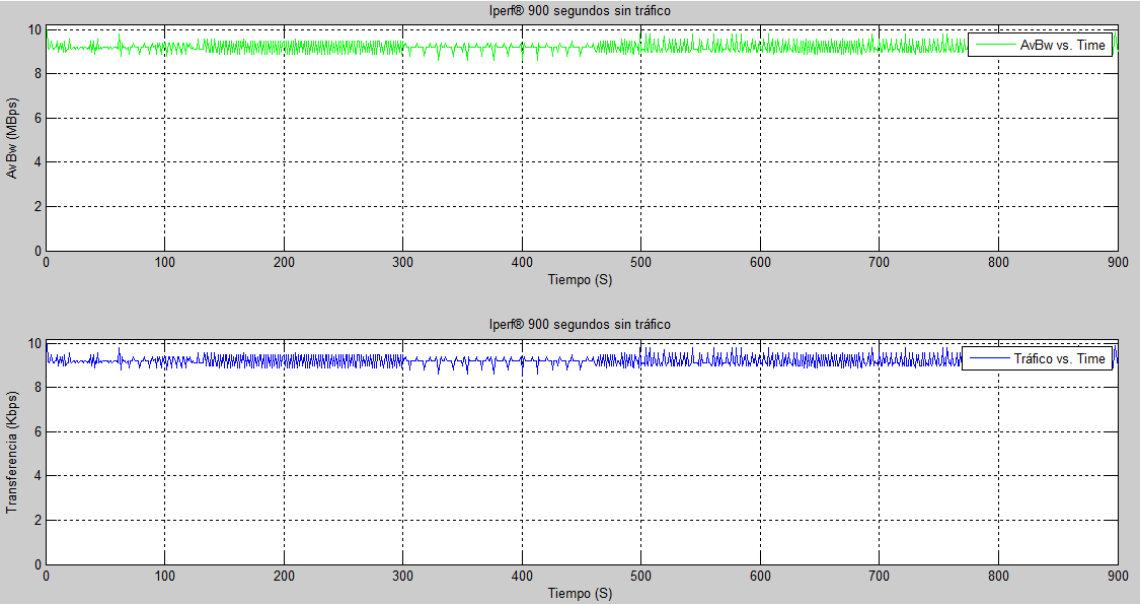

Figura 50. Iperf® desde Matlab® durante 900 seg. sin tráfico.

En la figura 50, se evidenciar en el entorno de Matlab® la respuesta de los datos que se importaron desde el software Excel® y en el Anexo 15 se puede observar más detalladamente el m-file de esta prueba. Por consiguiente en la parte superior el ABw disponible en color verde y en la parte inferior una tasa despreciable de transferencia en Kbps (de color azul); que se hace referencia a procesos de memoria propios del sistema.

Como se mencionó anteriormente en el apartado de los 60 segundos sin tráfico con Iperf®, el análisis sin cross-trafic en el Testbed es para analizar el comportamiento de la herramienta; ahora, se procede a realizar las pruebas en los mismos instantes de tiempo pero esta vez induciendo tráfico en el canal.

7.2.2.4. Iperf®, 60 segundos con tráfico.

Para la medición, es necesario desde el Testbed dos hosts sean los encargados de estar registrando los datos de la herramienta estimadora, para este caso Iperf®, y por otro lado se utilizan dos hosts cuya función será la de generar el tráfico, para este caso sintético generado desde una herramienta llamada Mgen®. Es de resaltar que Mgen® permite diferentes clases de tráfico pero en este caso se trabajará con tráfico periódico en los diferentes tiempos de muestreo predefinidos en la parte inicial de este apartado. Posterior, se procede a tomar lectura de los datos y estos ser almacenados en un .txt y ser parametrizados desde Excel® y Matlab®.

A continuación en la tabla 7 se muestra algunos datos ya importados en el entorno de Excel®; aunque en el Anexo 16 se puede observar todos los datos en el intervalo de tiempo que se analizaron.

IPERF® 60 Segundos con Tráfico

Intervalo	Transferencia (MBytes)	AvBw (Mbytes/seg)	Jitter (ms)
1	2.124	8	0.023
2	1.324	7.2	0.026
3	1.224	7.1	0.027
4	1.324	7.2	0.03
5	1.624	7.5	0.028
6	1.224	7.1	0.026
7	1.224	7.1	0.022
8	1.324	7.2	0.03
9	1.324	7.2	0.021
10	1.624	7.5	0.024
11	1.124	7	0.033
12	1.524	7.4	0.022
13	1.224	7.1	0.035
14	1.524	7.4	0.02
15	1.124	7	0.027
16	1.624	7.5	0.027
17	1.124	7	0.025
18	1.224	7.1	0.029
19	1.224	7.1	0.029
20	1.724	7.6	0.029

Tabla 7. Resultados Iperf® desde Excel® durante 60 seg. con tráfico.

Se muestra en la tabla 7, los intervalos de un segundo donde el estimador realiza la toma de dato y seguido de esto, Iperf® muestra la cantidad de bits transferidos, junto al ancho de banda que queda disponible y el jitter o promedio de paquetes en el instante de tiempo. Posterior a esto, se procede a graficar desde Excel® el ABw que hay disponible con respecto a la capacidad máxima del canal que son 10 Megas.

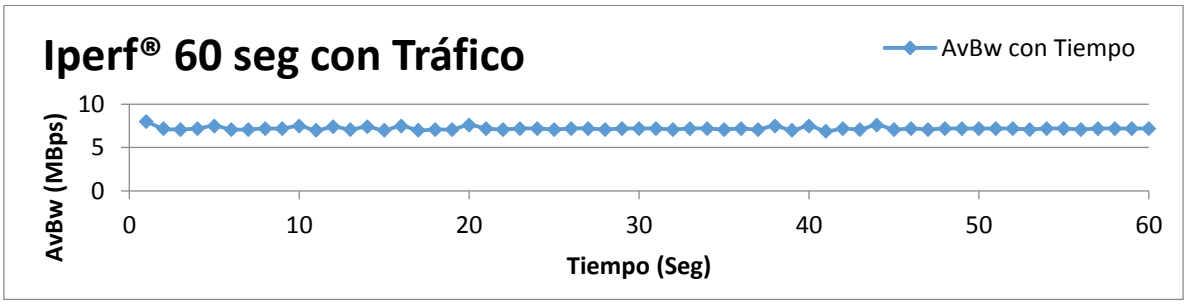

Figura 51. Iperf® desde Excel® durante 60 seg. con tráfico.

Se puede observar en la figura 51, como se reduce el Ancho de Banda en aproximadamente un 75% con respecto a cuándo no existía tráfico por lo que se puede decir que si la capacidad máxima del canal es de 10 Megas y existen un promedio de 7,2 Megas disponibles el canal estaría ocupando entre procesos de caché y el tráfico sintético unos 2,8 Megas aproximados.

Ahora, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw. A continuación en la figura 52 se muestra el m-file creado para generar las diferentes gráficas del estudio del estimador Iperf® trabajado con tráfico sintético desde Mgen®.

```
%/////////////////IPERF® 60 SEGUNDOS CON TRAFICO//////////////////////

fileName = ('60_CT.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,1);         %tiempo de muestra 60 segundos
y = a(:,3);         %ancho de banda disponible
w = a(:,2);         %tráfico de bits
v = a(:,5);         %jitter durante el periodo de muestra

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA////////

subplot(3,1,1);                 %posicion de la grafica en el marco general
z = plot(x,y,'g')                   %grafica ancho de banda con respecto a tiempo
axis([0 60 0 10.2])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw (Mbps)')
title('Iperf® 60 segundos con tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA DE LA TRANSFERENCIA////////

subplot(3,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica porcentaje de overhead con respecto a tiempo
axis([0 60 0 2.5])            %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('Transf.(Mbps)')
title('Iperf® 60 segundos con tráfico')
grid on
legend('Tráfico vs. Time')
```

FiveMin_CTIperf.m | FiveteenMin_CTIperf.m | OneMin_CTIperf.m

Figura 52. m-file de Matlab® para Iperf® durante 60 seg. con tráfico.

Se puede observar el m-file donde se importa el archivo de Excel® que posee los datos adquiridos del .txt de la herramienta estimadora (Iperf®); y luego de esto se genera a partir de código las gráficas de ABw, Tráfico y Jitter.

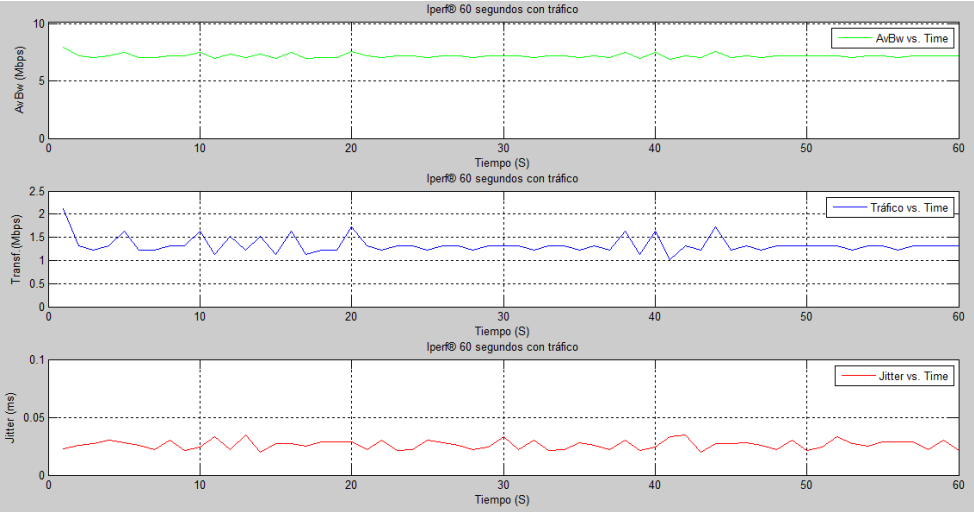

Figura 53. Iperf® desde Matlab® durante 60 seg. con tráfico.

En la figura 53, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente; seguido de esta en la mitad y de color azul los MBytes que están siendo transferidos; y en la parte inferior y resaltado de un color rojo el promedio o Jitter de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 60 segundos a intervalos de 1 segundo cada estimación.

7.2.2.5. Iperf®, 300 segundos con tráfico.

De igual forma que en la prueba anterior de 60 segundos con tráfico, se almacenaron los datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 17 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 300 segundos como periodo de muestra y en el Anexo 18 se puede observar el m-file creado para esta prueba.

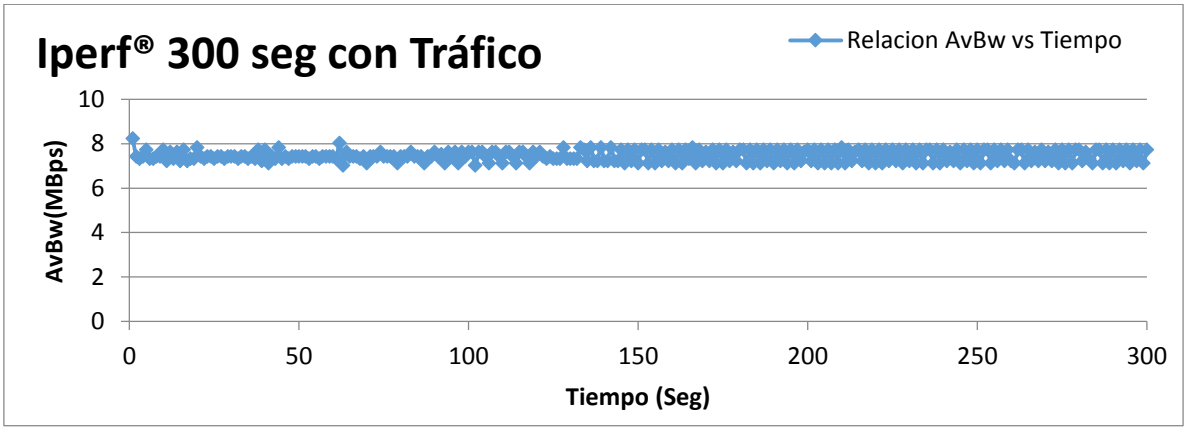

Figura 54. Iperf® desde Excel® durante 300 seg. con tráfico.

Se puede observar en la figura 54, que el ABw se reduce aproximadamente igual que en la prueba de 60 segundos; por los que se concluye que el canal también se reduce en aproximadamente un 75% y que el promedio de ABw es de 7,4 Megas. Posterior a esto, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw.

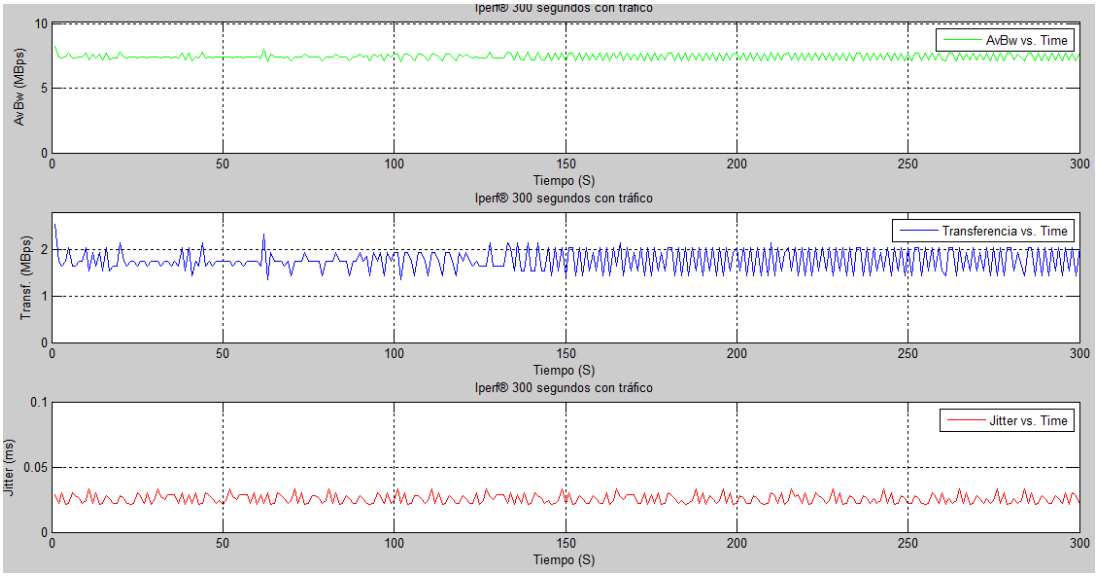

Figura 55. Iperf® desde Matlab® durante 300 seg. con tráfico.

En la figura 55, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente; seguido de esta en la mitad y de color azul los MBytes que están siendo transferidos; y en la parte inferior y resaltado de un color rojo el promedio o Jitter de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 300 segundos a intervalos de 1 segundo cada estimación.

7.2.2.6. Iperf®, 900 segundos con tráfico.

Finalmente, la última prueba que se realiza es de 15 minutos (900 segundos) y se realiza el mismo procedimiento que con las pruebas anteriores con tráfico; que son almacenar datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 19 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 900 segundos como periodo de muestra y en el Anexo 20 se puede observar el m-file creado para esta prueba.

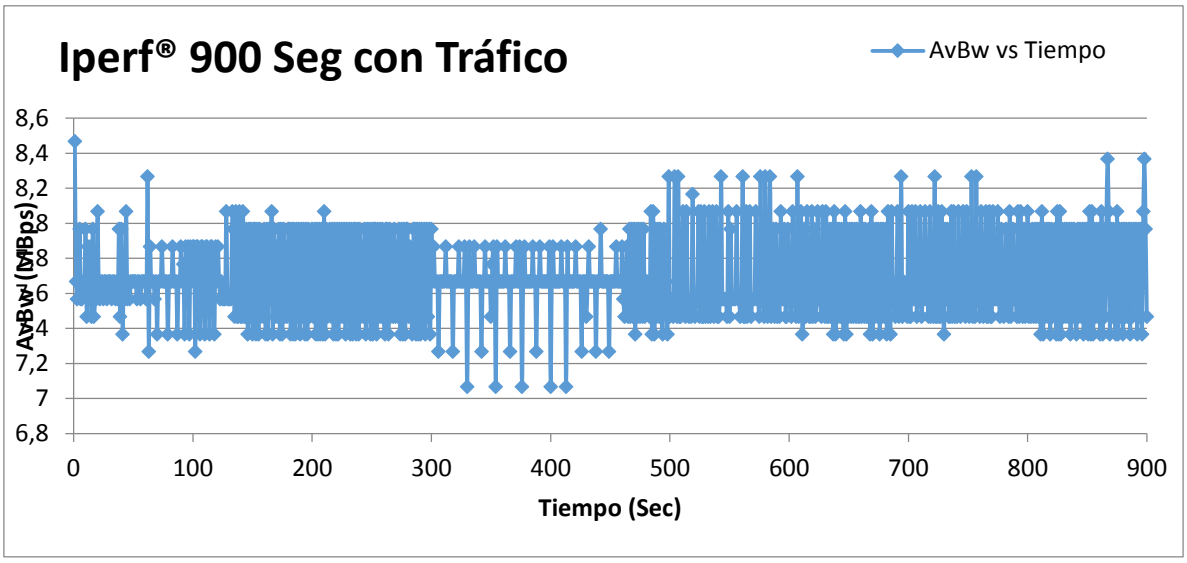

Figura 56. Iperf® desde Excel® durante 900 seg. con tráfico.

Se puede observar en la figura 56, que el ABw se reduce aproximadamente igual que en la prueba de 60 segundos; por los que se concluye que el canal también se reduce en aproximadamente un 75% y que el promedio de ABw es de 7,6 Megas. Posterior a esto, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw.

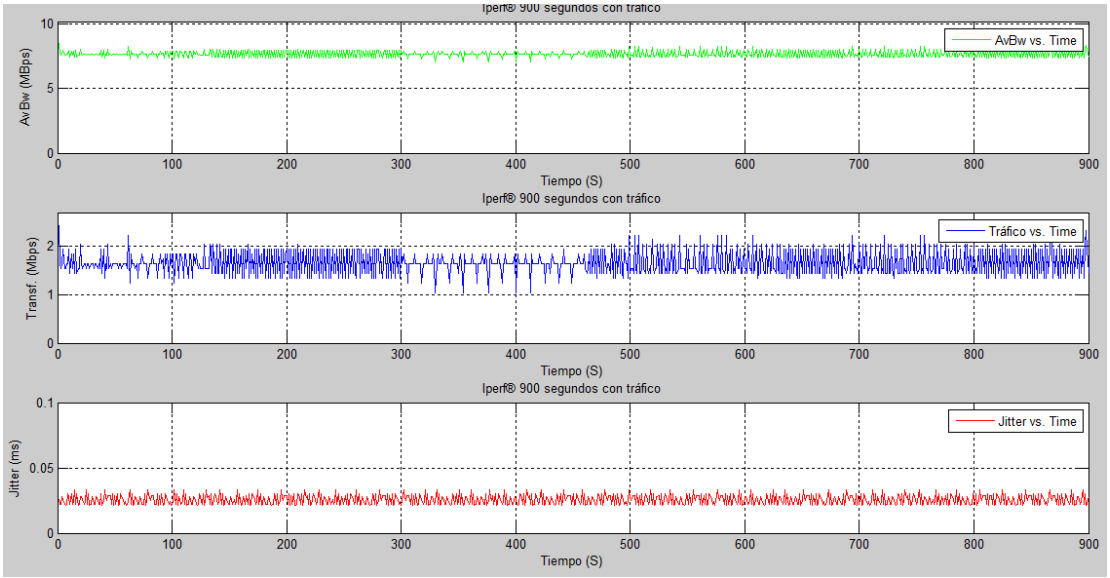

Figura 57. Iperf® desde Matlab® durante 900 seg. con tráfico.

En la figura 57, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente; seguido de esta en la mitad y de color azul los MBytes que están siendo transferidos; y en la parte inferior y resaltado de un color rojo el promedio o Jitter

de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 900 segundos a intervalos de 1 segundo cada estimación.

7.2.3. Ifstat®.

Ifstat® permite al usuario evidenciar el momento en que hay tráfico sea de salida como de entrada en la interfaz física de la maquina (bien puede ser eth o wlan).En el momento que no existe tráfico, automáticamente el estimador muestra en 0 (cero) los bits de entrada como de salida; mientras que al momento de evidenciar tráfico muestra los valores de los mismos. Posterior a esto, se procede a realizar las mismas pruebas que con los estimadores anteriores en las mismas condiciones del Testbed con y sin cross-trafic utilizando los mismos intervalos de tiempo (1, 5, 15 minutos).

De igual forma que los otros estimadores mencionados anteriormente, Ifstat® necesita de dos hosts ubicados a los extremos del Testbed y teniendo en cuenta que uno actuará como sender y el otro como revsender. Posterior a esto, se habilita la herramienta con permisos de ejecución desde comandos para el envío y recepción.

Ahora, el código de ejecución del Ifstat® es el siguiente:

#sudo ifstat –s 192.168.20.6 –t 60
#sudo ifstat –r 192.168.10.2
(Noregistra, 2013)

Donde el primer comando hace referencia al hosts de envío y el segundo comando se ubica en el hosts de recepción. Teniendo en cuenta estos dos comandos, se realizan las respectivas pruebas sin y con tráfico aclarando que la toma de cada muestra es en intervalos de un segundo. A continuación se mostrará los resultados obtenidos en un orden de tiempo desde 60 segundos hasta 900 segundos, primero sin tráfico y luego se repetirá el proceso con congestión en el canal. A continuación en la figura 58 se muestra el comportamiento de la herramienta desde el terminal ubicados en el hosts de envío teniendo en cuenta que cada estimación la realiza en intervalos de 1 segundo.

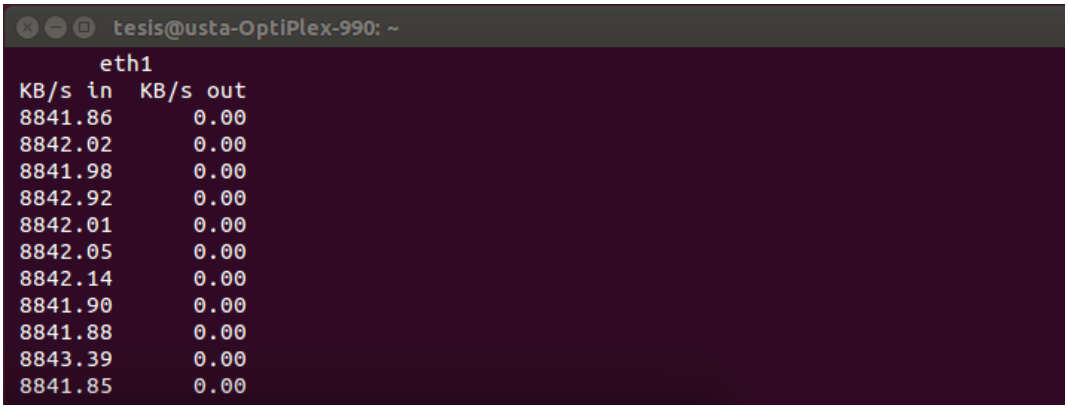

Figura 58. Ifstat® ejecutándose desde el hosts de envío sin tráfico.

7.2.3.1. Ifstat®, 60 segundos sin tráfico.

Para la medición, desde el host de envío se asigna un tiempo de muestreo de 60 segundos teniendo en cuenta que el canal es de 10 Megas y se evidencia que sin nada de tráfico el canal permanece libre. Luego de esto se procede a almacenar estos datos en un .txt para ser parametrizados en dos softwares como lo son Excel® y Matlab®.

IFSTAT® 60 SEGUNDOS SIN TRÁFICO

Host Rx		Host Tx	
Tráfico de Entrada (KBps)	Tiempo (S)	Tráfico de Salida (KBps)	Abw (MBps)
0	1	0	10
0	2	0	10
0	3	0	10
0	4	0	10
0	5	0	10
0	6	0	10
0	7	0	10
0	8	0	10
0	9	0	10
0	10	0	10
0	11	0	10
0	12	0	10
0	13	0	10
0	14	0	10
0	15	0	10
0	16	0	10
0	17	0	10
0	18	0	10
0	19	0	10
0	20	0	10

Tabla 8. Resultados Ifstat® desde Excel® durante 60 seg. sin tráfico.

Se observa en la tabla 8, que los datos que fueron adquiridos desde el terminal del Linux se guardan en un.txt y posterior a esto se llevan a Excel® donde se empieza la parametrización de los mismos; en el Anexo 21 se observa el total de muestras durante el tiempo estimado. De igual forma que con los estimadores anteriores, las pruebas de Ifstat® sin tráfico es para analizar el comportamiento de la herramienta en tiempos altos. Ahora, con los datos en el entorno de Excel® se procede a graficar el ABw.

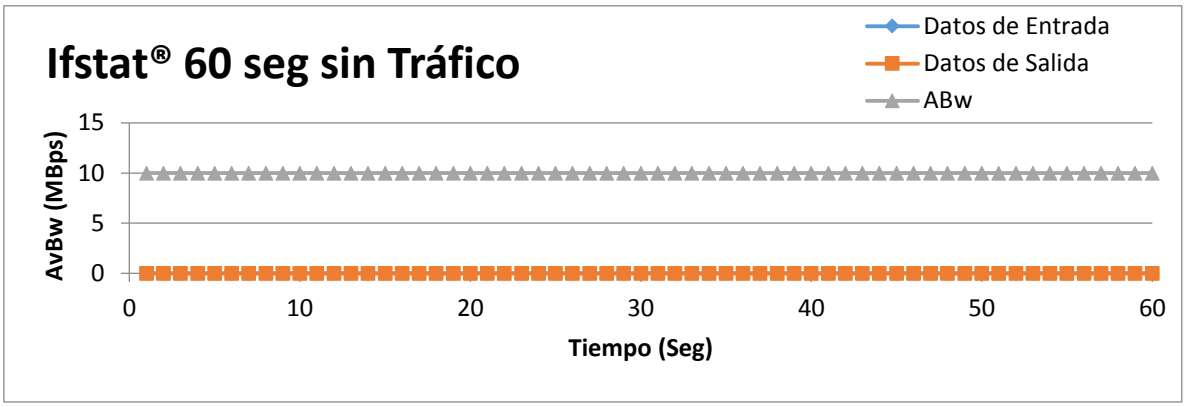

Figura 59. Ifstat® desde Excel® durante 60 seg. sin tráfico.

En la figura 59 se evidencia el ABw disponible cuando no existe nada de tráfico en el Testbed; teniendo en cuenta que se limitó el canal a 10 Megas. De igual manera, se muestra que el tráfico entre los hosts donde está la aplicación Ifstat® no muestra valores de bits, por esta razón se puede decir que el canal está libre en un 100%. Ahora, se procede a importar el archivo de Excel® a Matlab® y a partir de un breve programa se realiza la respectiva visualización de los datos desde este entorno.

```
%//////////////////IFSTAT® 60 SEGUNDOS SIN TRÁFICO///////////////////////

fileName = ('60_SecondsST.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,3);         %tiempo de muestra 60 segundos
y = a(:,1);         %ancho de banda disponible rx
w = a(:,5);         %ancho de banda disponible tx
v = a(:,7);         %ancho de banda disponible
%//////////////////////GRÁFICA ENCARGADA DE ANCHO DE BANDA////////////////

subplot(2,1,1);             %posicion de la grafica en el marco general
z = plot(x,v,'g')           %grafica ancho de banda con respecto a tiempo
axis([0 60 0 10.2])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw(MBps)')
title('Ifstat® 60 segundos sin tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA TRÁFICO DE TX Y RX/////////

subplot(2,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica de Rx con respecto a tiempo
hold on
i = plot(x,y,'r.')              %grafica de Tx con respecto a tiempo
legend([z,i],'Rx','Tx')
axis([0 60 -2 10])              %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('Tráfico (MBps)')
title('Tráfico de Rx y Tx')
grid on
```
Figura 60. M-file desde Matlab® para Ifstat® durante 60 seg. sin tráfico.

En la figura 60 se evidencia el m-file creado en Matlab® para lograr una visualización de los datos que brinda Ifstat®. Posterior a esto, se muestra el resultado a partir de gráficas del ancho de banda y la tasa de bits de transferencia.

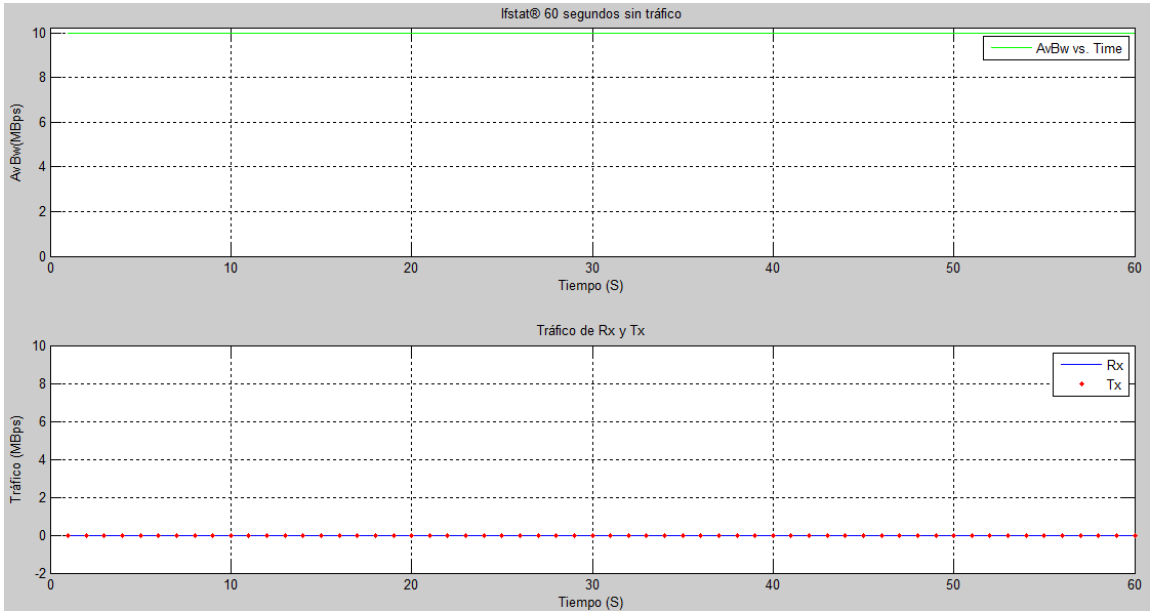

Figura 61. Ifstat® desde Matlab® durante 60 seg. sin tráfico.

Se puede observar en la parte superior en la figura 61, el resultado del ABw y el tráfico de bits de Tx y RX. Ahora, sin tráfico en la primera gráfica la línea verde permanece en los 10 Megas del canal aunque en teoría este resultado como tal no podría permanecer todo el tiempo ahí dado que existen procesos de caché y memoria que ocupan parte pequeña del canal. En la parte inferior se muestra el tráfico o para más claridad los bits de transferencia entre Tx y Rx, pero como la prueba está realizada en un entorno controlado y sin generar tráfico se puede evidenciar que permanece en 0 (cero) la línea azul y la punteada roja. Finalmente, es bueno aclarar que al no generar este tráfico, por consiguiente no existe un promedio de paquetes para esta prueba y esto mismo se espera en los resultados de 300 y 900 segundos.

7.2.3.2. Ifstat®, 300 segundos sin tráfico.

Dado que el proceso es el mismo que el ítem anterior, los datos que se adquieren desde el terminal se parametrizan en los software Excel® y Matlab®. En el Anexo 22 se muestra los datos que brindó la herramienta estimadora desde el software Excel® y desde ese entorno se parametrizan y son exportados al software Matlab®.

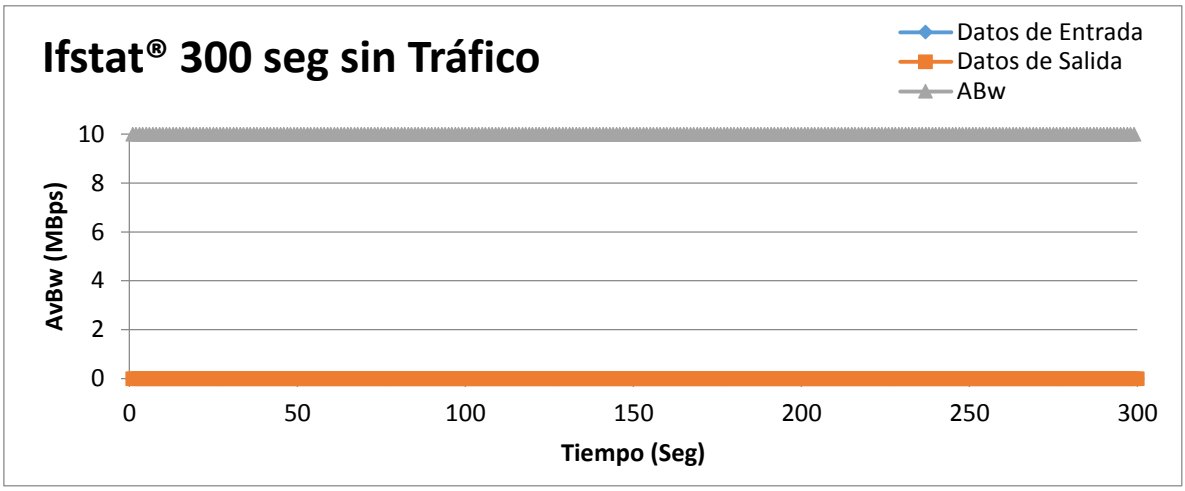

Figura 62. Ifstat® desde Excel® durante 300 seg. sin tráfico.

Se observa anteriormente en la figura 62 el ABw durante los 5 minutos de prueba, es decir, 300 segundos y se concluye que el canal está en un 100% de disponibilidad pero como se menciona anteriormente no es cierto del todo dado que existen diferentes procesos de caché. Posterior a esto, se muestra el comportamiento de estos datos en el entorno de Matlab®.

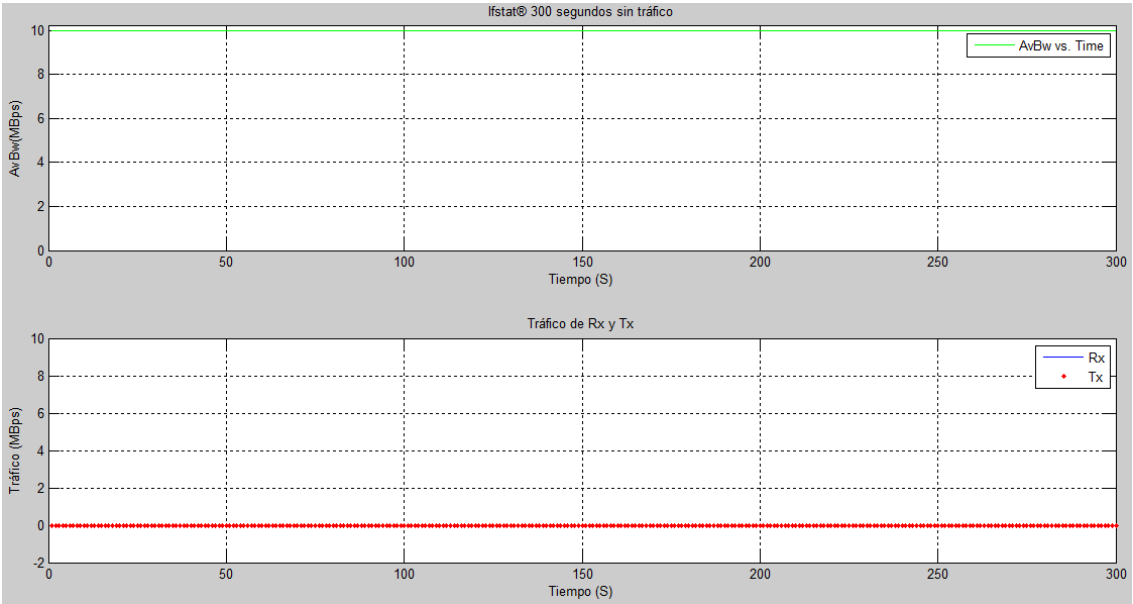

Figura 63. Ifstat® desde Matlab® durante 300 seg. sin tráfico.

Se puede evidenciar en el entorno de Matlab®, en la parte superior el ABw disponible en color verde y en la parte inferior la tasa de transferencia en Kbps (de color azul). En el Anexo 23 se puede observar el m-file que se utilizó para esta prueba.

7.2.3.3. Ifstat®, 900 segundos sin tráfico.

Finalmente, dado que anteriormente se resaltó los tiempos de pruebas de la herramienta, se procede a analizarla durante 15 minutos o su equivalente que son 900 segundos; luego de esto se parametrizan los datos obtenidos en el software Excel® y Matlab®. En el Anexo 24 se muestra los datos que brindó la herramienta estimadora desde el software Excel® y desde ese entorno se parametrizan y son exportados al software Matlab®. En la figura 64 se aprecia el ABw desde el software Excel®.

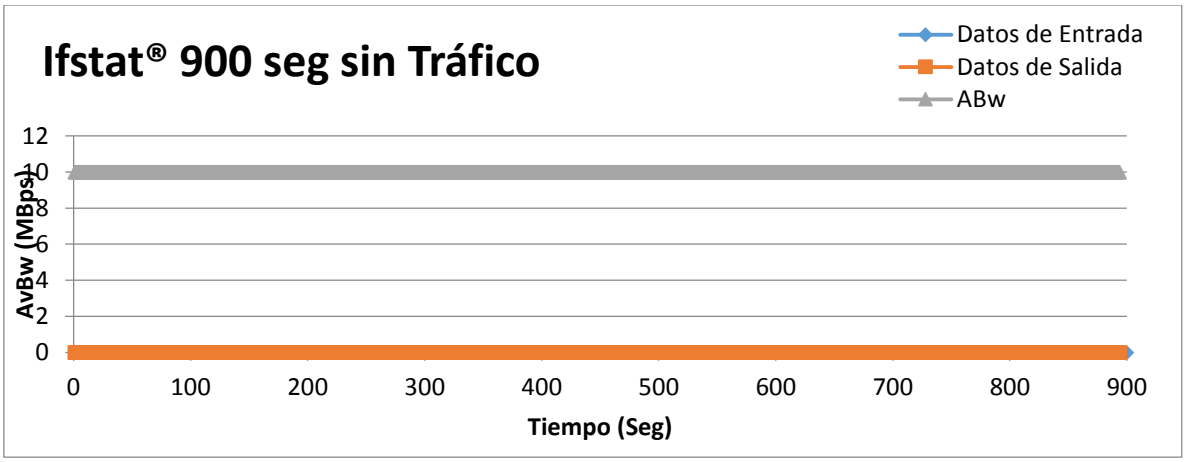

Figura 64. Ifstat® desde Excel® durante 900 seg. sin tráfico.

Como se observa anteriormente, se puede considerar el ABw disponible durante el tiempo estimado de prueba (900 segundos), y se concluye que el ancho está en su total de disponibilidad que son las 10 Megas a las que está el canal.

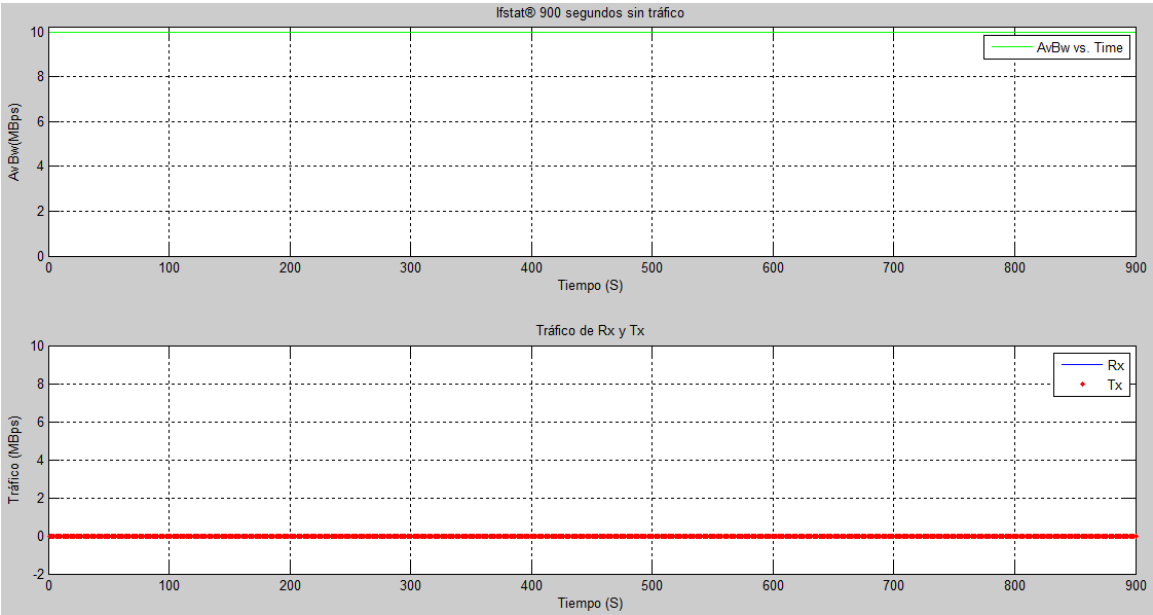

Figura 65. Ifstat® desde Matlab® durante 900 seg. sin tráfico.

En la figura 65, se evidencia en el entorno de Matlab® la respuesta de los datos que se importaron desde el software Excel® y en el Anexo 25 se puede observar más detalladamente el m-file de esta prueba. Por consiguiente en la parte superior el ABw disponible en color verde y en la parte inferior una tasa de transferencia en Kbps (de color azul).

Como se mencionó anteriormente en el apartado de los 60 segundos sin tráfico con Ifstat®, el análisis sin cross-trafic en el Testbed es para analizar el comportamiento de la herramienta; ahora, se procede a realizar las pruebas en los mismos instantes de tiempo pero esta vez induciendo tráfico en el canal.

7.2.3.4. Ifstat®, 60 segundos con tráfico.

Como se mencionó anteriormente, se requiere en el Testbed dos hosts que sean partícipes de estar registrando los datos de la herramienta estimadora, para este caso Ifstat®, y por otro lado dos hosts cuya función será la de generar el tráfico, para este caso sintético generado desde una herramienta llamada Mgen®. Es de resaltar que igual que los dos estimadores anteriores (Traceband® e Iperf®) trabajaron con tráfico periódico en los diferentes tiempos de muestreo; Ifstat® también fue sometido a las mismas prácticas. Posterior, se procede a tomar lectura de los datos y estos ser almacenados en un .txt y ser parametrizados desde Excel® y Matlab®.

A continuación en la tabla 9 se muestra algunos datos ya importados en el entorno de Excel®; aunque en el Anexo 26 se puede observar todos los datos en el intervalo de tiempo que se analizaron.

IFSTAT® 60 SEGUNDOS CON TRÁFICO

Host Rx		Host Tx		
Tráfico de Entrada (KBps)	Tiempo (S)	Tráfico de Salida (KBps)	Abw(KBps)	Jitter (ms)
7243.23	1	7242.33	2756.77	0.066
7241.88	2	7241.86	2758.12	0.012
7241.86	3	7241.91	2758.14	0.031
7241.64	4	7241.82	2758.36	0.041
7243.36	5	7241.75	2756.64	0.034
7241.77	6	7243.5	2758.23	0.033
7241.94	7	7241.93	2758.06	0.022
7243.12	8	7242.75	2756.88	0.055
7242.33	9	7242.02	2757.67	0.027
7241.86	10	7241.96	2758.14	0.078
7241.91	11	7241.98	2758.09	0.057
7241.82	12	7242.71	2758.18	0.075
7241.75	13	7242.27	2758.25	0.049

7243.5	14	7241.81	2756.5	0.039
7241.93	15	7241.81	2758.07	0.049
7242.75	16	7243.43	2757.25	0.012
7242.02	17	7241.01	2757.98	0.036
7241.96	18	7242.26	2758.04	0.027
7241.98	19	7241.23	2758.02	0.078
7242.71	20	7243.44	2757.29	0.057

Tabla 9. Resultados Ifstat® desde Excel® durante 60 seg. con tráfico.

Se muestra en la tabla 9, los intervalos de un segundo donde el estimador realiza la toma de dato y seguido de esto, Ifstat® muestra la cantidad de bits transferidos tanto de salida como de entrada, junto al jitter o promedio de paquetes en el instante de tiempo. Posterior a esto, se procede a graficar desde Excel® el ABw que hay disponible con respecto a la capacidad máxima del canal que son 10 Megas.

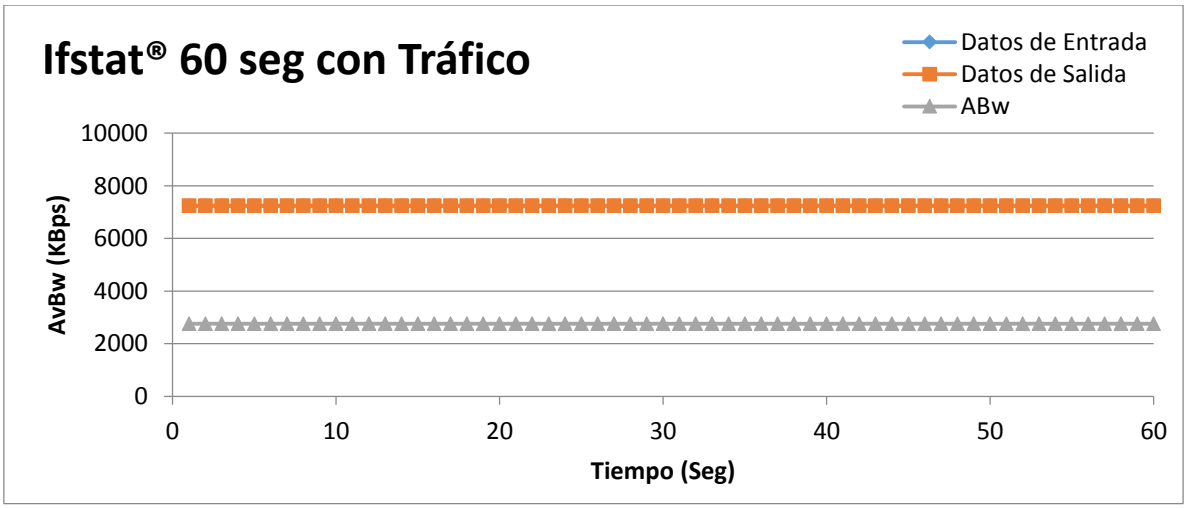

Figura 66. Ifstat® desde Excel® durante 60 seg. con tráfico.

Se puede observar en la figura 66, como se reduce el Ancho de Banda en aproximadamente un 30% con respecto a cuándo no existía tráfico por lo que se puede decir que si la capacidad máxima del canal es de 10 Megas y existen un promedio de 7,2 Megas ocupadas en el canal y unas 2,8 Megas disponibles aproximadamente.

Ahora, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw. A continuación en la figura 67 se muestra el m-file creado para generar las diferentes gráficas del estudio del estimador Ifstat® trabajado con tráfico sintético desde Mgen®.

```
%//////////////////IFSTAT® 60 SEGUNDOS CON TRÁFICO//////////////////////

fileName = ('60_SecondsCT.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

x = a(:,3);         %tiempo de muestra 60 segundos
y = a(:,1);         %ancho de banda Rx
w = a(:,5);         %ancho de banda Tx
v = a(:,7);         %ABw durante el periodo de muestra
s = a(:,9);         %jitter durante el periodo de muestra

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA/////////

subplot(3,1,1);                 %posicion de la grafica en el marco general
z = plot(x,v,'g')               %grafica ABw con respecto a tiempo
axis([0 60 0 4000])         %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw(KBps)')
title('Ifstat® 60 segundos con tráfico')
grid on
legend('AvBw vs. Time')

%//////////////////////GRAFICA ENCARGADA DE LA TRANSF. DE Tx Y Rx/////////

subplot(3,1,2);                 %posicion de la grafica en el marco general
z = plot(x,w,'b')               %grafica ancho de banda de Tx con tiempo
hold on
i = plot(x,y,'r')               %grafica ancho de banda de Rx con tiempo
axis([0 60 6800 6870])          %asignacion de min y max en el plano a graficar
xlabel('Tiempo (S)')
ylabel('AvBw (KBps)')
title('Ifstat® 60 segundos Transf. Tx y Rx')
grid on
```

Figura 67. M-file desde Matlab® para Ifstat® durante 60 seg. con tráfico.

Se puede observar el m-file donde se importa el archivo de Excel® que posee los datos adquiridos del .txt de la herramienta estimadora (Ifstat®); y luego de esto se genera a partir de código las gráficas de ABw, Tráfico y Jitter.

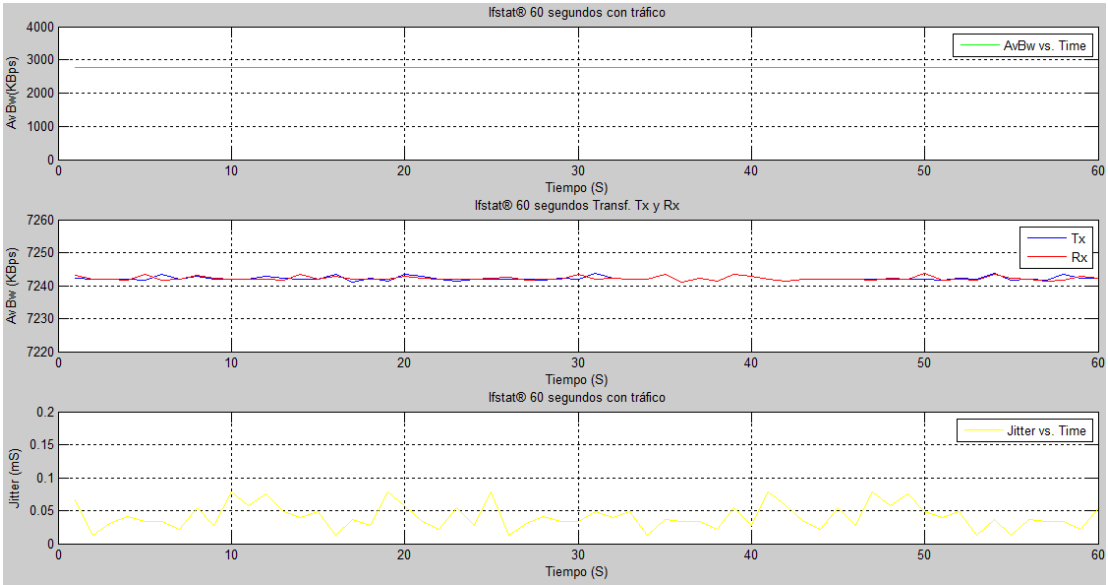

Figura 68. Ifstat® desde Matlab® durante 60 seg. con tráfico.

En la figura 68, se puede evidenciar en la parte superior el ABw de color verde teniendo como resultado que de 0 a 2800 KBps es el canal libre y ya lo restante a completar 10000 KBps hace referencia a la congestión del canal entre tráfico sintético y procesos de memoria. Seguido de esto, en la mitad y de color azul y rojo los KBytes que están siendo transferidos; y en la parte inferior y resaltado de un color amarillo el promedio o Jitter de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 60 segundos a intervalos de 1 segundo cada estimación.

7.2.3.5. Ifstat®, 300 segundos con tráfico.

De igual forma que en la prueba anterior, se almacenaron los datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 27 se muestra los datos que brindó la herramienta estimadora desde el software Excel® y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 300 segundos como periodo de muestra y en el Anexo 28 se puede observar el m-file creado para esta prueba.

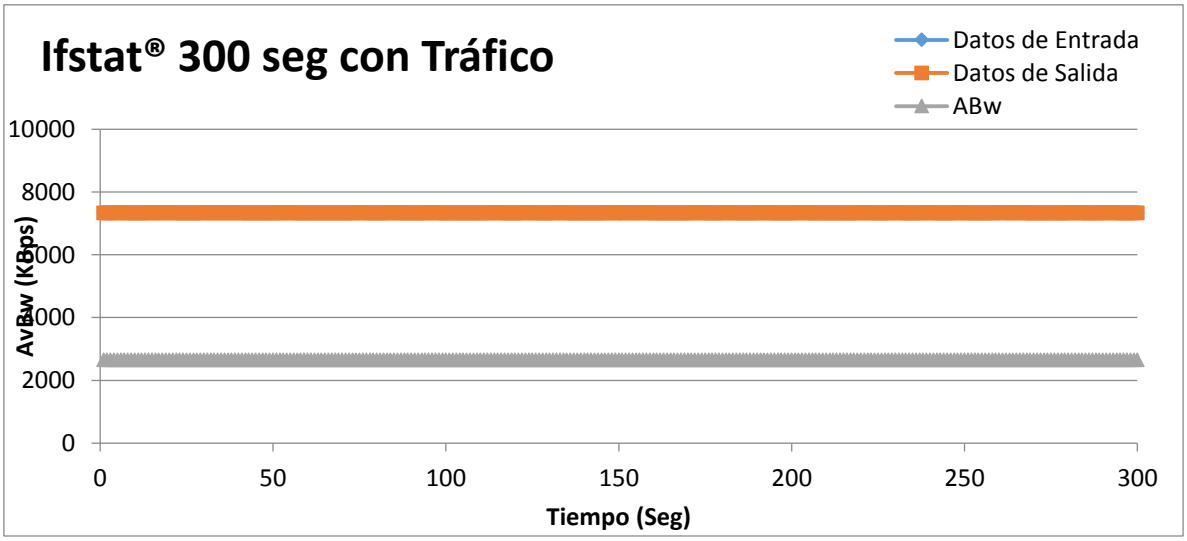

Figura 69. Ifstat® desde Excel® durante 300 seg. con tráfico.

Se puede observar en la figura 69, que el ABw se reduce bastante; teniendo en cuenta que de 0 a el límite de la línea verde es el ancho de banda disponible y de 0 al límite de la línea roja es el ancho que está siendo ocupado por el tráfico periódico de Mgen® y procesos de memoria. Posterior a esto, se procede a exportar los datos de Excel® al entorno de Matlab®.

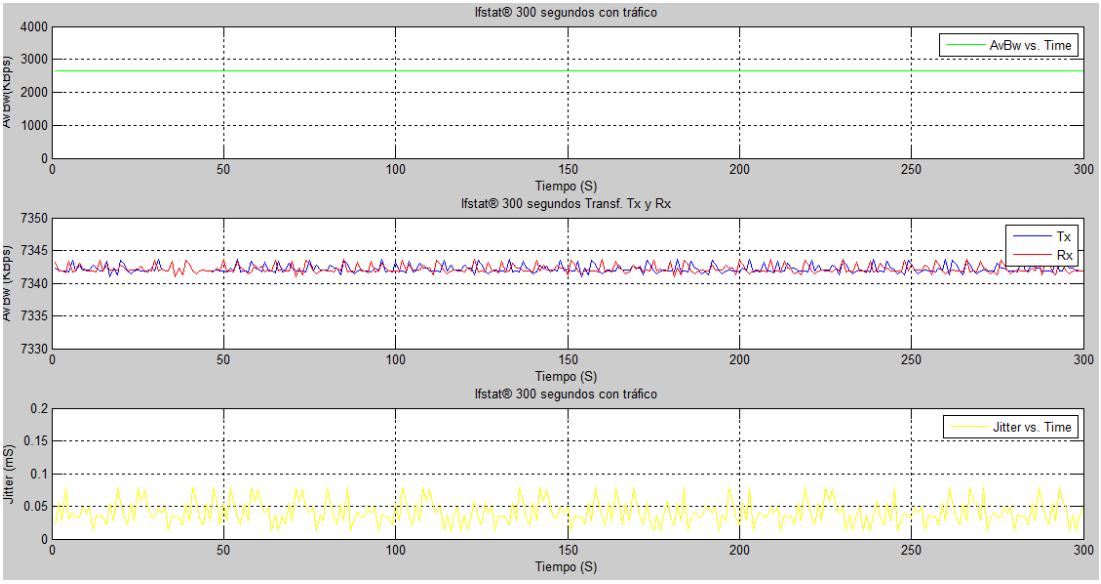

Figura 70. Ifstat® desde Matlab® durante 300 seg. con tráfico.

En la figura 70, se puede evidenciar en la parte superior el ABw de color verde de 0 al valor de la línea es el ABw disponible; seguido de esta en la mitad y de color azul y rojo los KBytes que están siendo transferidos; y en la parte inferior y resaltado de un color amarillo el promedio o Jitter de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 300 segundos a intervalos de 1 segundo cada estimación.

7.2.3.6. Ifstat®, 900 segundos con tráfico.

Finalmente, la última prueba que se realiza es de 15 minutos (900 segundos) y se realiza el mismo procedimiento que con las pruebas anteriores con tráfico; que son almacenar datos en un .txt y posterior a esto se parametrizan en los software Excel® y Matlab®. En el Anexo 29 se muestra los datos que brindó la herramienta estimadora desde el software Excel y desde ese entorno se parametrizan y son exportados al software Matlab®.

A continuación, se muestra la gráfica en Excel® del ABw disponible en los 900 segundos como periodo de muestra y en el Anexo 30 se puede observar el m-file creado para esta prueba.

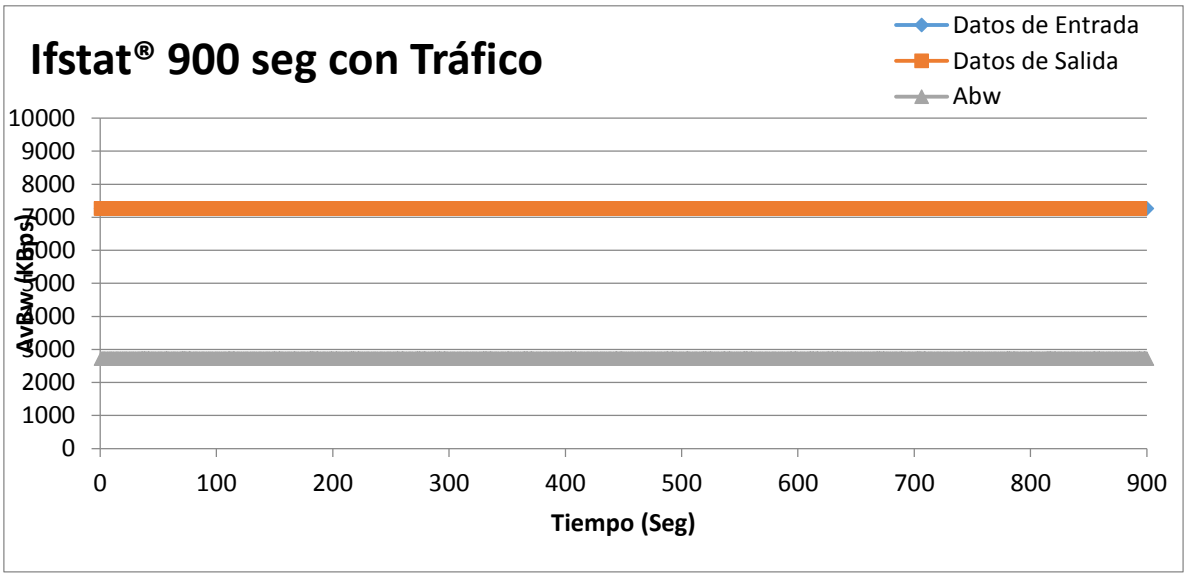

Figura 71. Ifstat® desde Excel® durante 900 seg. con tráfico.

Se puede observar en la figura 71, que el ABw se reduce aproximadamente un 70% con una tasa promedio de transferencia de 7,2 MBps por los que se concluye que el canal queda con una disponibilidad aproximadamente de un 30%. Posterior a esto, se procede a exportar los datos al entorno de Matlab® donde se graficarán otros parámetros a parte del ABw.

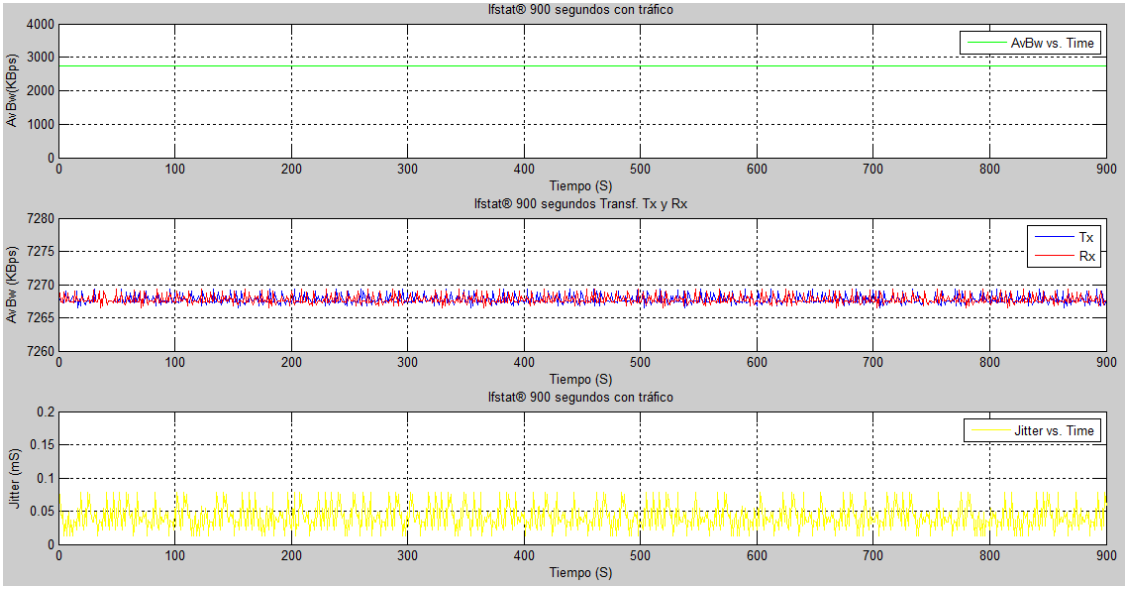

Figura 72. Ifstat® desde Matlab® durante 900 seg. con tráfico.

En la figura 72, se puede evidenciar en la parte superior el ABw de color verde y como se mencionaba con anterioridad se ve una reducción del canal bastante evidente; seguido de esta en la mitad y de color azul y rojo los KBytes que están siendo transferidos; y en la parte inferior y resaltado de un color amarillo el

promedio o Jitter de los paquetes transmitidos. Es de aclarar, que estas tres imágenes se trabajaron en un periodo de 900 segundos a intervalos de 1 segundo cada estimación.

7.3. Capítulo 3. Proceso para destacar la herramienta más adecuada para la medición de ancho de banda.

En el desarrollo del estudio de las tres herramientas estimadoras (Traceband®, Iperf® e Ifstat®) analizando la información obtenida y teniendo en cuenta las variables y calidad de los datos medidos (jitter, máximo ancho de canal, estimación de ancho de banda y latencia) en diferentes contextos de usabilidad de la red (desarrollo Capítulo 2 de la sección 7); se procede a realizar una comparación en el comportamiento de cada aplicación con el fin de justificar y dar una conclusión de cuál es la herramienta que más se ajusta para la medición del ancho de banda en los diferentes parámetros que componen este fenómeno (ABw).

La comparación de dichas herramientas se hizo a partir del análisis de las gráficas obtenidas del software Matlab®; el cual brindará información detallada de cada variable a fin de obtener valores precisos que permitan evidenciar la robustez de cada aplicación.

Finalmente, a partir de los resultados obtenidos se recrearán en diferentes entornos (Matlab®), las tres herramientas estimadoras a fin de encontrar una tendencia que permite evidenciar información de las variables en las mediciones de diferentes intervalos de tiempo con ausencia y presencia del Cross-Trafic.

7.3.1. Comportamiento en la medición de variables con ausencia de Cross-trafic.

En primer lugar, se estudiará los resultados de las herramientas del Capítulo dos aclarando una ausencia del cross-trafic; es decir, como se menciona en ese mismo capítulo, estudiar el comportamiento de las herramientas en el tiempo para verificar su fiabilidad y robustez. De igual forma, se trabajará en orden ascendente desde la primera prueba de 60 segundos, seguida de 300 segundos y finalizando en los 900 segundos.

7.3.1.1. Estimadores sin Cross-trafic 60 segundos.

Para dar inicio a este análisis en los 60 segundos sin cross-trafic, es necesario crear un libro en el software Excel® para importar los datos de cada herramienta del ancho de banda y posterior a esto a exportar esos datos al entorno de Matlab® a fin de lograr un estudio de los mismos. A continuación en la tabla 10 se evidencian algunos datos del ancho de banda de las herramientas aclarando que todas los valores estarán en MBps y que existirán dos tiempos de muestreo dado que las estimaciones de Traceband® se realizan en aproximadamente 0,5 segundos mientras que Iperf® e Ifstat® en 1 segundo.

60 Segundos sin Tráfico (MBps)

Traceband	Iperf	Ifstat	Tiempo(Tr)	Tiempo
10	10	10	0.66	1
10	9.2	10	1.11	2
10	9.1	10	1.58	3
9.5	9.2	10	2.04	4
10	9.5	10	2.54	5
9.5	9.1	10	3.02	6
10	9.1	10	3.55	7
10	9.2	10	3.94	8
9	9.2	10	4.41	9
9.5	9.5	10	4.87	10
10	9	10	5.33	11
10	9.4	10	5.83	12
10	9.1	10	6.22	13
10	9.4	10	6.63	14
10	9	10	7.13	15
9.5	9.5	10	7.56	16
9.5	9	10	8.05	17
9	9.1	10	8.56	18
10	9.1	10	9.02	19
10	9.6	10	9.49	20

Tabla 10. Resultados del ABw de los tres estimadores desde Excel® durante 60 seg. sin tráfico.

Posterior a esto, se generar en el entorno de Matlab® un programa que permitirá el análisis de las tres herramientas en el mismo tiempo. A continuación se puede observar el programa que se realizó.

```
%//////////////////60 SEGUNDOS SIN TRÁFICO//////////////////////

fileName = ('60S_ST.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

tr = a(:,1);          %datos traceband
ip = a(:,3);          %datos iperf
is = a(:,5);          %datos ifstat
tt = a(:,7);          %tiempo traceband
ti = a(:,9);          %tiempo iperf e ifstat

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA////////

b = plot(tt,tr,'g--')
hold on
c = plot(ti,ip,'b')
hold on
d = plot(ti,is,'r.')
hold on
axis([0 60 8 10.5])
xlabel('Tiempo (S)')
ylabel('AvBw(MBps)')
title('60 segundos sin tráfico')
grid on
legend([b,c,d],'Traceband','Iperf','Ifstat')
```

Figura 73. M-file desde Matlab® para los estimadores durante 60 seg. sin tráfico.

Finalmente, en la figura 74 se muestra el comportamiento en la medición de ABw de Traceband®, Iperf® e Ifstat® teniendo en cuenta ausencia de tráfico en un tiempo de 60 segundos.

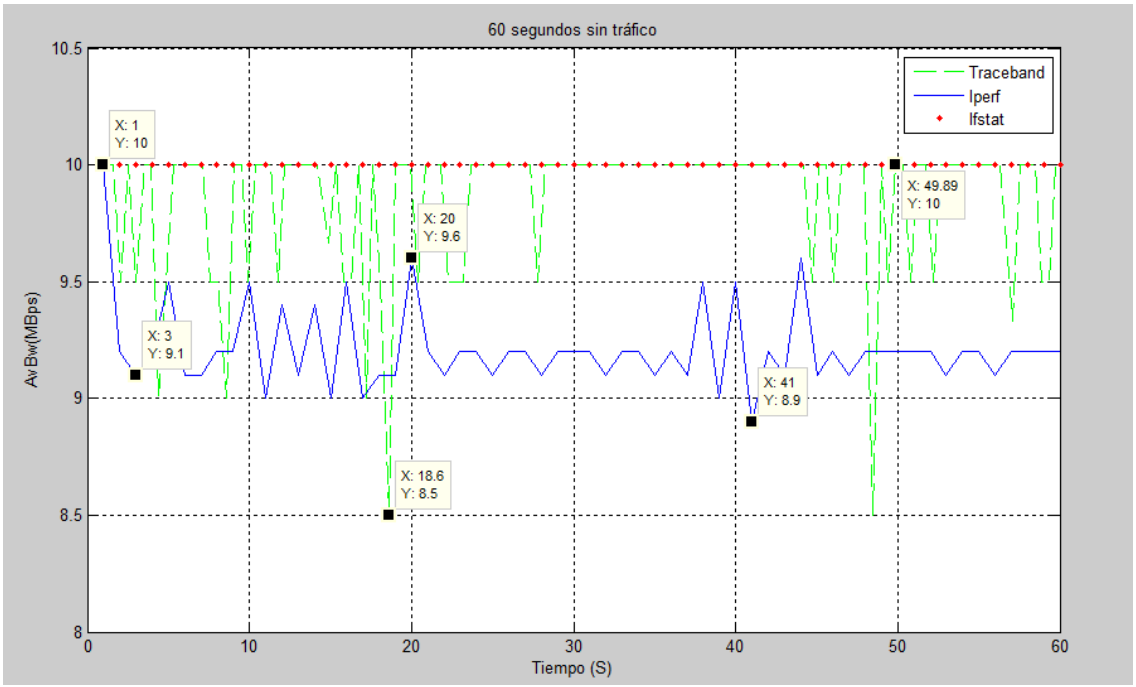

Figura 74. Estimadores desde Matlab® durante 60 seg. sin tráfico.

La información que Traceband® entrega nos refleja un canal descongestionado con una tendencia a los 10Mbps, un mínimo de 8.5 Mbps y un promedio de 9.5 Mbps. Por otro lado Iperf® nos muestra información diferente, en el momento de iniciar la prueba el ancho de banda disponible es de 10Mbps, casi de manera inmediata por procesos de Overhead, el canal se ve reducido a 9.1 Mbps, luego regresa a 9.5 y cambia entre 8.9 y 9.6 durante el resto de la prueba en ocasiones trata de estabilizarse, pero los procesos de cache que se llevan en el canal no lo permiten, la información no presenta ningún tipo de periodicidad. Finalmente Ifstat® entrega información de un canal libre durante toda la medición y un ABw constante de 10Mbps, de esta primera prueba se puede concluir que dicho estimador entrega información real pero poco precisa a diferencia de Traceband® e Iperf®.

7.3.1.2. Estimadores sin Cross-trafic 300 segundos.

De igual manera que el ítem anterior, se procede a trabajar los datos en el software Excel y luego ser exportados al entorno de Matlab a fin de evidenciar y analizar el comportamiento de los estimadores. En los anexos 31 y 32 se puede evidenciar los datos en Excel® y el m-file en Matlab®.

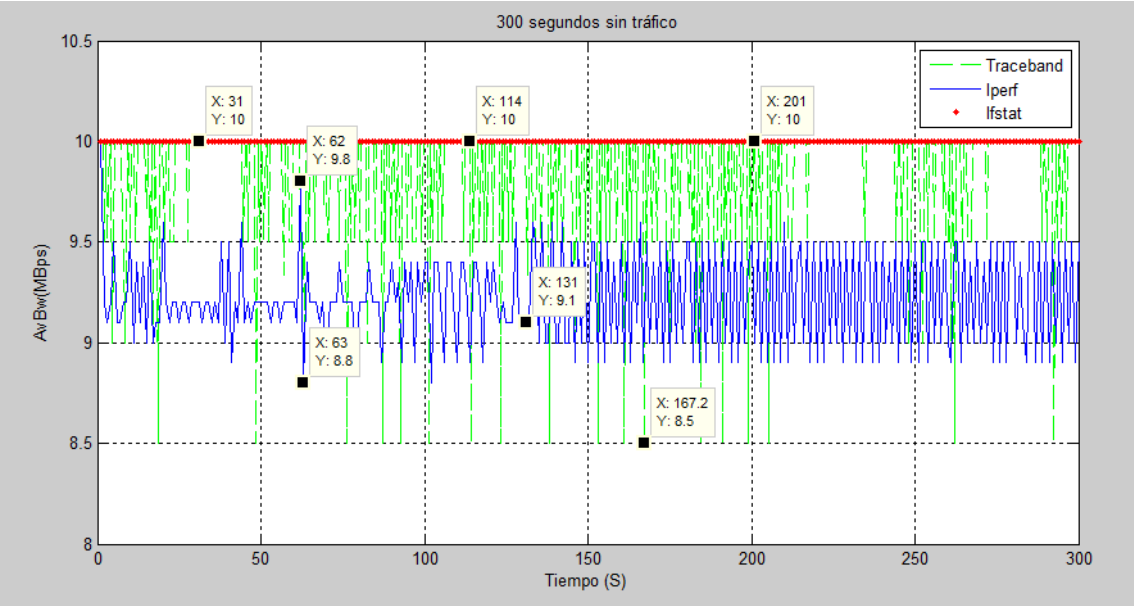

Figura 75. Estimadores desde Matlab® durante 300 seg. sin tráfico.

En la figura 75 se muestra el comportamiento de los estimadores con tiempo de 300 segundos. Traceband®, el estimador con línea verde entrega información de un canal descongestionado, con un ancho de banda máximo de 10Mbps y un mínimo de 8.5. Por otro lado, Iperf® con línea azul entrega un comportamiento similar al mostrado con tiempo de 60 segundos, el estimador al inicio de la prueba muestra un ABw de 10Mbps valor mínimo de 8.8 y un promedio de 9.1Mbps se destaca de los otros estimadores porque solo al inicio del test están disponibles las 10 Mbps; este comportamiento refleja una medición más coherente. Finalmente Ifstat® entrega información más imprecisa ya que la sensibilidad de

dicho estimador no es capaz de percibir la fluctuación de los datos y siempre entrega un ABw de 10Mbps.

7.3.1.3. Estimadores sin Cross-trafic 900 segundos.

Finalmente se realiza la prueba para 900 segundos. En los anexos 33 y 34 se puede evidenciar los datos en el software Excel® y el m-file del entorno de Matlab®.

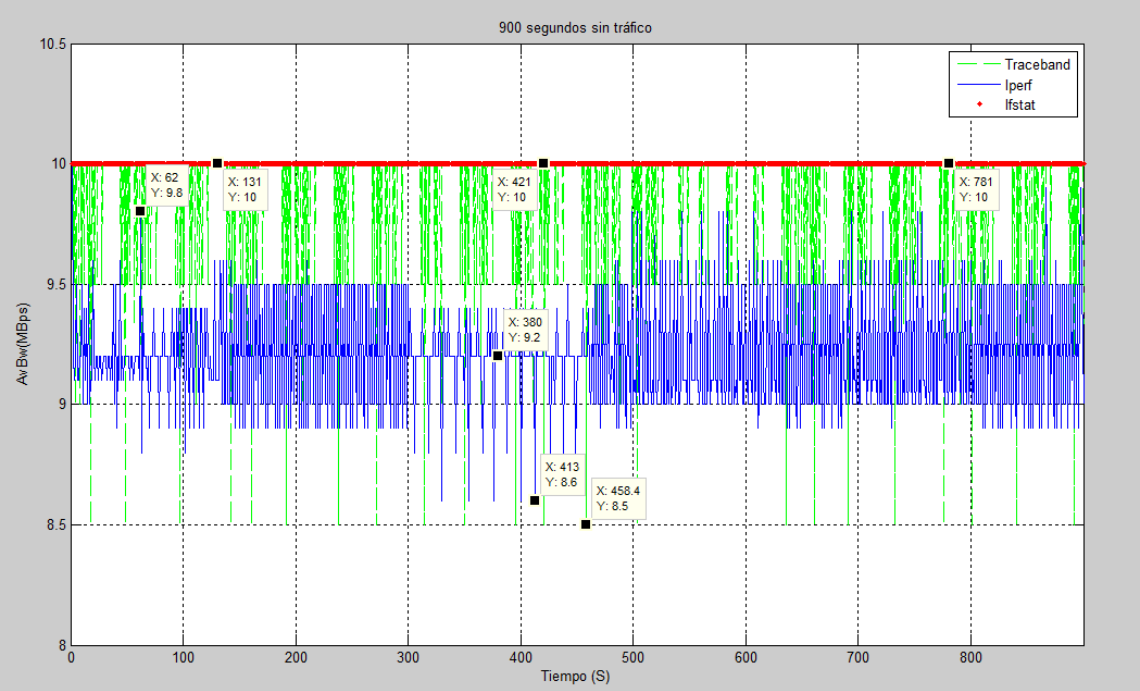

Figura 76. Estimadores desde Matlab® durante 900 seg. sin tráfico.

La figura 76 muestra el comportamiento en la estimación de Abw de cada estimador con 900 segundos (15 minutos). La línea roja representa al comportamiento evidenciado por Ifstat®, al igual que en las anteriores mediciones el ancho de banda disponible permanece constante en 10Mbps, esto entrega una referencia en el comportamiento de la herramienta apuntando a su capacidad de precisión en las mediciones, la cual es casi nula. Con esta prueba Traceband® nos muestra un comportamiento de la estimación donde el ABw disponible oscila entre 8.5 y 10 Mbps. Finalmente y en color azul está el resultado de Iperf® como se menciona en los resultados anteriores dicho estimador entrega información que muestra con más detalle el comportamiento de la red, al iniciar la prueba se muestra 10Mbps, luego el ABw oscila entre 8.6 y 9.8.

De este apartado se puede evidenciar las diferentes formas en la interpretación de los datos de cada uno de los estimadores; es importante mencionar que el comportamiento y calidad de los datos puede diferir dependiendo de la manera como se realice la medición puntualmente si se realiza con o sin cross-trafic.

A continuación, se realizarán las pruebas teniendo en cuenta tráfico cruzado, y se finalizará dando un dictamen de la aplicación que interpreta de manera más adecuada la información.

7.3.2. Comportamiento en la medición de variables con Cross-trafic.

Para este apartado se estudiará los resultados de las herramientas del Capítulo dos en presencia de cross-trafic; es decir, como se menciona en ese mismo capítulo, estudiar el comportamiento de las herramientas en el tiempo añadiendo una congestión al canal con tráfico sintético periódico desde Mgen. De igual forma, se trabajará en orden ascendente desde la primera prueba de 60 segundos, seguida de 300 segundos y finalizando en los 900 segundos.

7.3.2.1. Estimadores con Cross-trafic 60 segundos.

Para dar inicio a este análisis en los 60 segundos con cross-trafic, es necesario crear un libro en el software Excel® para importar los datos de cada herramienta del ancho de banda y posterior a esto a exportar esos datos al entorno de Matlab® a fin de lograr un estudio de los mismos. A continuación en la tabla 11 se evidencian algunos datos del ancho de banda de las herramientas aclarando que todas los valores estarán en MBps y que existirán dos tiempos de muestreo dado que las estimaciones de Traceband® se realizan en aproximadamente 0,5 segundos mientras que Iperf® e Ifstat® en 1 segundo.

60 Segundos con Tráfico (MBps)

Traceband	Iperf	Ifstat	Tiempo(Tr)	Tiempo
4.6	8	7.24323	0.46	1
4.8	7.2	7.24188	0.99	2
7	7.1	7.24186	1.46	3
7.8	7.2	7.24164	1.89	4
9.8	7.5	7.24336	2.43	5
8.2	7.1	7.24177	3.19	6
6.2	7.1	7.24194	3.83	7
7.4	7.2	7.24312	4.28	8
7.6	7.2	7.24233	4.83	9
7.6	7.5	7.24186	5.34	10
9.8	7	7.24191	5.79	11
7.8	7.4	7.24182	6.55	12
8.6	7.1	7.24175	7.13	13
7.4	7.4	7.2435	7.69	14
8	7	7.24193	8.18	15
7.8	7.5	7.24275	8.72	16
8.4	7	7.24202	9.27	17
8.4	7.1	7.24196	9.73	18
7.4	7.1	7.24198	10.26	19
8.4	7.6	7.24271	10.73	20

Tabla 11. Resultados del ABw de los tres estimadores desde Excel® durante 60 seg. con tráfico.

Posterior a esto, se generar en el entono de Matlab® un programa que permitirá el análisis de las tres herramientas en el mismo tiempo. A continuación se puede observar el programa que se realizó.

```
%//////////////////60 SEGUNDOS CON TRÁFICO//////////////////////

fileName = ('60S_CT.xlsx');    %Nombre del archivo en excel

a = xlsread(fileName);          %Se importa a matlab el archivo de excel

tr = a(:,1);          %datos traceband
ip = a(:,3);          %datos iperf
is = a(:,5);          %datos ifstat
tt = a(:,7);          %tiempo traceband
ti = a(:,9);          %tiempo iperf e ifstat

%//////////////////////GRAFICA ENCARGADA DEL ANCHO DE BANDA/////////

b = plot(tt,tr,'g--')
hold on
c = plot(ti,ip,'b')
hold on
d = plot(ti,is,'r.')
hold on
axis([0 60 4.5 10])
xlabel('Tiempo (S)')
ylabel('AvBw(MBps)')
title('60 segundos con tráfico')
grid on
legend([b,c,d],'Traceband','Iperf','Ifstat')
```

Figura 77. M-file desde Matlab® para los estimadores durante 60 seg. con tráfico.

Finalmente, La figura 78 muestra el comportamiento de los estimadores en la medición del ABw con cross-trafic generados a partir de Mgen. Traceband® de línea verde muestra un comportamiento oscilante, en el momento de iniciar la prueba refleja el ABw disponible en 4.6 Mbps, de manera casi inmediata sube hasta 9.8 y un promedio de 7.5Mbps. Por otro lado Iperf® tiene un comportamiento más estable de la medición, al inicio la variable es de 8Mbps, el ABw oscila entre 6.9 y 7.6 Mbps coherente teniendo en cuenta que Mgen está generando trafico periódico y la medición debería arrojar un ancho de banda orientado a ser constante teniendo en cuenta el tipo de tráfico. Finalmente, Ifstat® entrega información imprecisa pero no equivocada, interpretando la información como un ancho de banda constante de 7.2.

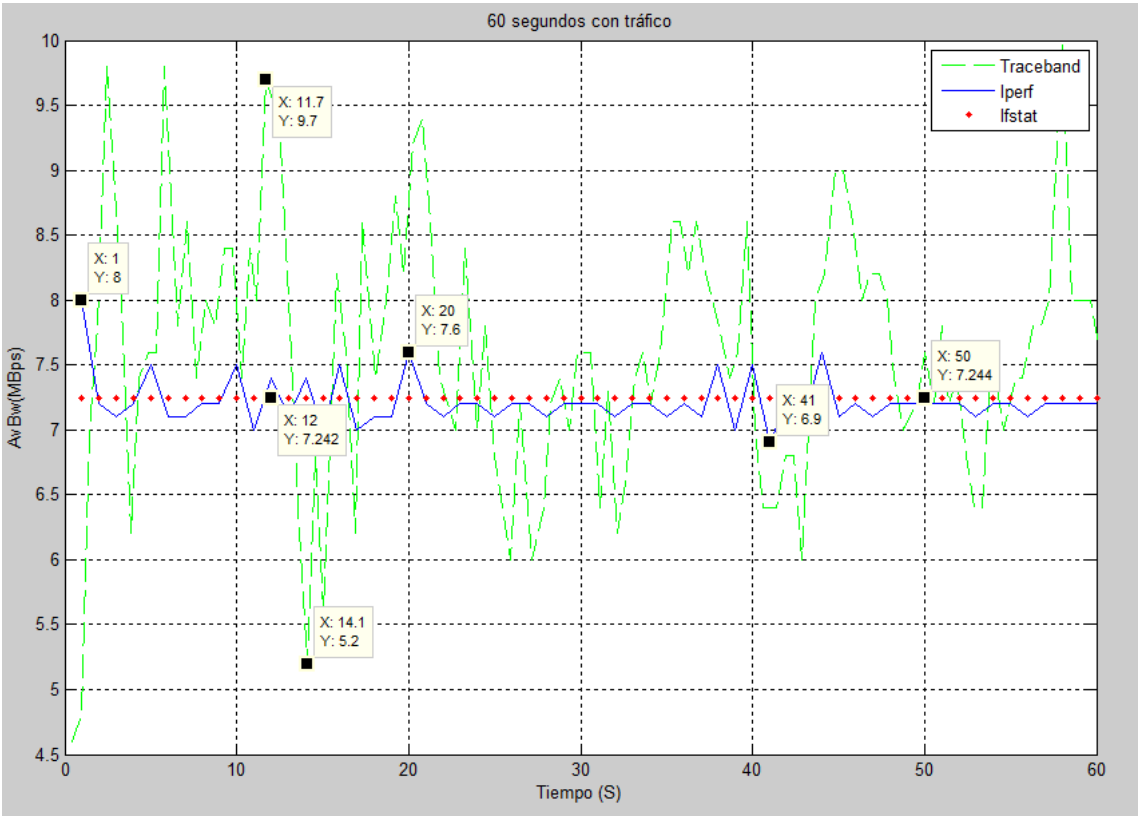

Figura 78. Estimadores desde Matlab® durante 60 seg. con tráfico.

Esta información nos revela el comportamiento de los estimadores cuando existe tráfico cruzado, con esta prueba podemos ver que Traceband® tiene un comportamiento muy oscilante, por el contrario Ifstat®, es demasiado impreciso en la medición pues el comportamiento es totalmente lineal durante toda la prueba y no se podría dar una conclusión acertada sobre el comportamiento de la red. Finalmente, Iperf® muestra una medición que se asemeja más a las condiciones de la red.

7.3.2.2. Estimadores con Cross-trafic 300 segundos

De igual manera que el ítem anterior, se procede a trabajar los datos en el software Excel y luego ser exportados al entorno de Matlab a fin de evidenciar y analizar el comportamiento de los estimadores. En los anexos 35 y 36 se puede evidenciar los datos en Excel® y el m-file en Matlab®.

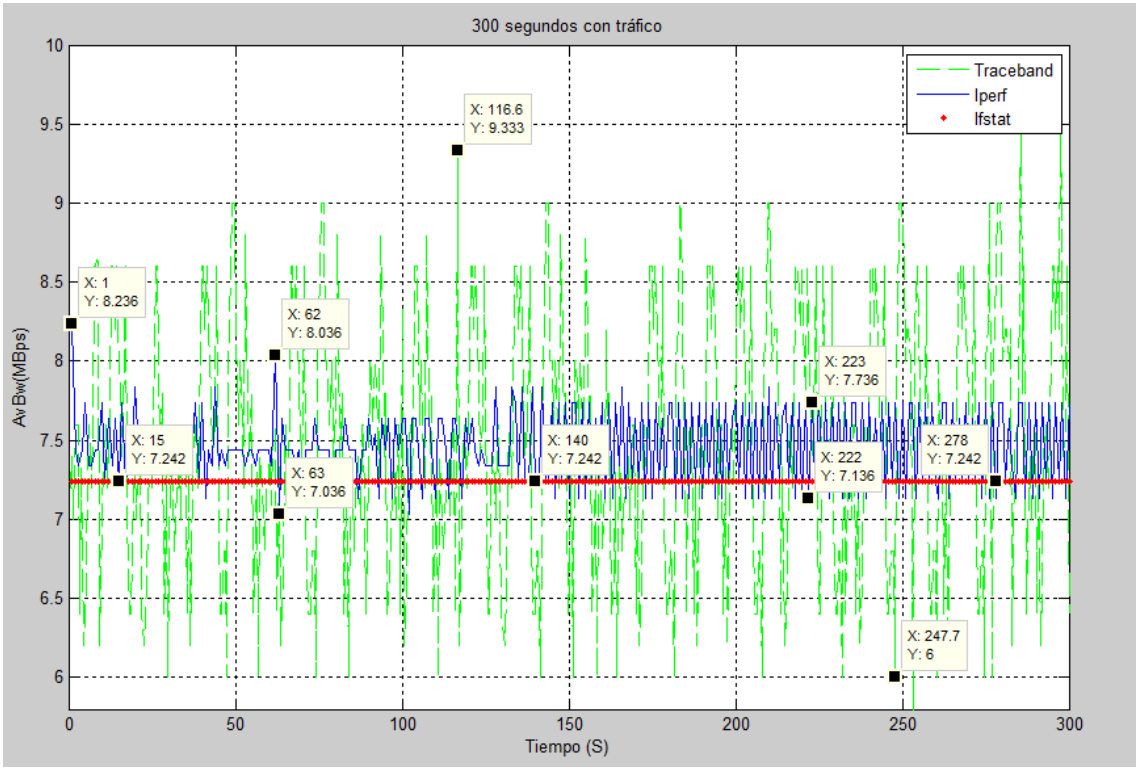

Figura 79. Estimadores desde Matlab® durante 300 seg. con tráfico.

La figura 79 muestra el comportamiento de los estimadores en la interpretación del ABw con tiempo de 300 segundos. En color verde Traceband® nos muestra un comportamiento oscilante muy similar a la prueba echa con 60 segundos, dicha oscilación se encuentra entre 6 y 9.3 Mbps. Ifstat® entrega una constante de 7.2Mbps de ABw disponible. Por otro lado Iperf® al inicio de la prueba muestra un ABw de 8.2 un mínimo de 7Mbps y un valor que oscila en 7.1 y 7.7 Mbps durante el resto de la prueba. Finalmente, al igual que la prueba anterior Iperf® muestra un comportamiento más acorde a la usabilidad del canal y la evidencia del mismo en la medición del ABw.

7.3.2.3. Estimadores con Cross-trafic 900 segundos.

Finalmente se realiza la prueba para 900 segundos. En los anexos 37 y 38 se puede evidenciar los datos en el software Excel® y el m-file del entorno de Matlab®.

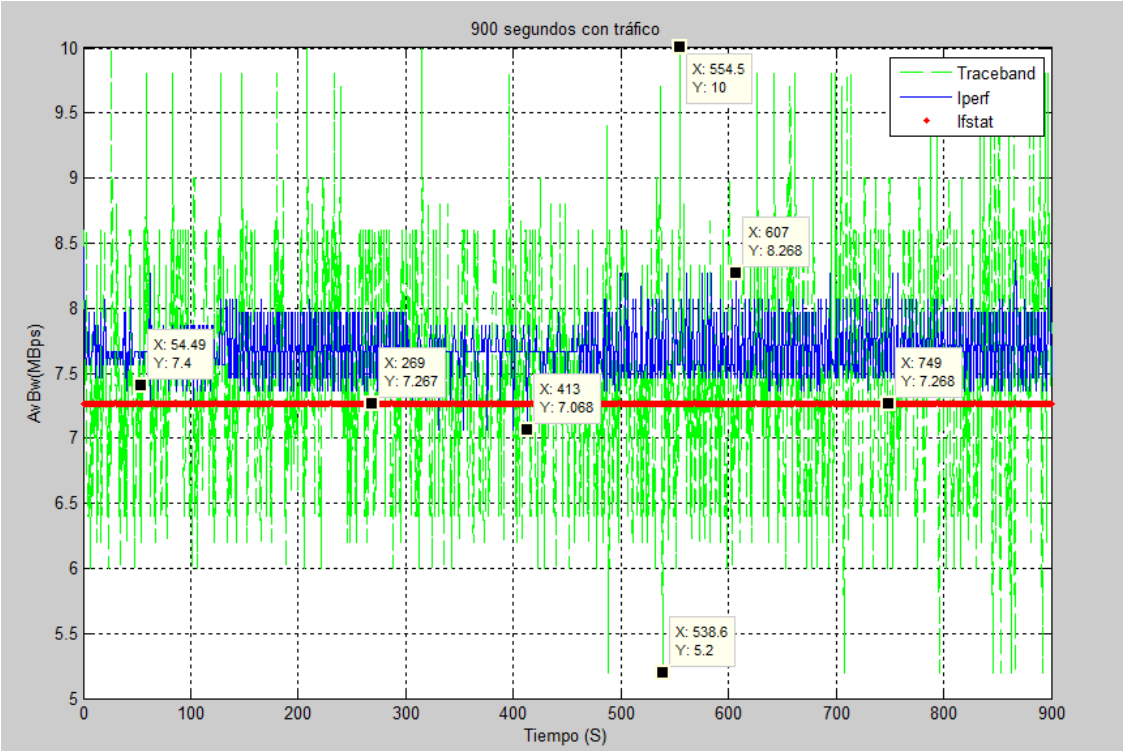

Figura 80. Estimadores desde Matlab® durante 900 seg. con tráfico.

En la figura 80 se muestra la medición de cada estimador con un muestreo de datos en 900 segundos. Se puede evidenciar un comportamiento muy similar al de las anteriores pruebas y es importante mencionar que al haber un muestreo mayor, existe una mejor precisión de medición por cada aplicación. Iperf® muestra mayor estabilidad a la hora de realizar mediciones con un canal en presencia de cross-trafic. El estimador Traceband®, entrega información con demasiada oscilación e Ifstat® entrega una constate en toda la prueba por lo que la conjeturas acerca del ABw disponible en la red pueden llegar ser carentes de precisión.

Es evidente la robustez de Iperf® en las mediciones echas en cada prueba, pues su información entrega de manera muy específica el estado de la red teniendo en cuenta que el tipo de tráfico que está transitando es periódico y por tal motivo se debería evidenciar un comportamiento similar en la medición. Sin embargo, los procesos de Overhead de cada estimador pueden variar de manera significativa en cada aplicación; caso puntual de Traceband®, cuya información tiende a oscilaciones rápidas durante toda prueba, permitiendo así resaltar errores que tal vez otras herramientas con menor velocidad podrían detectar y evidenciando la velocidad de este estimador durante un periodo de muestreo. Por otro lado y caso contrario Ifstat®, entrega información muy limitada sobre el ancho de banda disponible este estimador entrega una constate durante toda la prueba; aclarando que esta información no es errónea ya que el ancho de banda disponible es muy cercano a lo que muestran otros estimadores, pero la calidad de la datos no permite mostrar el comportamiento real de la red.

Teniendo en cuenta estos argumentos, Iperf® y Traceband® se consideran los estimadores que mejor se comportan tanto en mediciones con cross-trafic como ausencia del mismo; a partir de los estudios previos y diferentes análisis en el desarrollo de esta investigación; resaltando así que permiten al usuario muestreos significativos, tamaño del canal, versatilidad en las estimaciones, y variables a destacar como overhead, jitter; una ventaja significativa con respecto a otros estimadores.

8. OTROS APORTES DERIVADOS DEL TRABAJO

Durante el proceso de ejecución de este trabajo de grado, se encontraron diferentes aportes que pueden ser considerados como herramienta o complemento de futuras investigaciones. Lo anterior, en función de mediciones de ancho de banda disponible o pruebas equivalentes a la temática tratada en esta investigación.

Una característica importante a destacar fue la investigación llevada a cabo de las aplicaciones disponibles en sistema operativo Linux; las mismas que permiten al operario parametrizar el ancho de banda y encontrar sus diferentes características en un momento en una red. Para esto, se dispuso de un cronograma en donde se estableció un tiempo a fin de realizar una búsqueda detallada y posterior un análisis y estudio de diferentes aplicaciones. Seguido a esto, se logró un aproximado de 20 aplicaciones que experimentan el estudio de ancho de banda utilizando diferentes modelos para el análisis de esta fenómeno (PGM o PRM) resaltando algunas como: Nload, Iftop, Bmon, Slrum, Nethogs, Tcptrack, Vnstat, entre otras (para mayor información de las aplicaciones remitirse a la sección 7.1. de Resultados). Es de resaltar, de después de varios análisis no se logró encontrar un herramienta estimadora que muestre todos los parámetros que posee el canal de una red (latencia, jitter, overhead, troughtput, máximo canal, entre otros), razón por la cual se necesita de varias herramientas o de cálculos matemáticos para encontrar las diferentes variables de la red.

Para futuros investigadores que desean indagar y/o experimentar el ancho de banda de una red este trabajo de investigación brinda una cantidad de herramientas y a su vez un Testbed diseñado para el control de tráfico en el momento de la ejecución de las mismas. Seguido de esto, con las configuraciones de Router, Switch y Host (Sección 5.7) garantizando mediciones en un ambiente controlado permitiendo la adquisición adecuada de las variables implícitas y explicitas de ancho de banda.

Finalmente, al momento del testeo de las aplicaciones se trabajó con tiempos de muestreo que permitieron resultados contundentes y seguido del mismo ser utilizados como referente para realizar mediciones y análisis de comportamiento de las herramientas (robustez al momento de lapsos de tiempo altos).

9. CONCLUSIONES

En la indagación y estudio de las diferentes herramientas estimadoras de ancho de banda se observó en particular que cada una posee un plus o característica en alguna medición de variable del ABw; aunque es de resaltar que en otros aspectos carezca de factores para la medición de otras variables este patrón (ABw). Lo anterior, permitió concluir que se necesitan de diferentes estimadores para conocer con exactitud el comportamiento del canal de una red dado que en la investigación, encontrar una herramienta totalmente ideal es un poco complejo.

Es de resaltar que para el desarrollo de la investigación, se diseñó una topología para la ejecución de las diferentes pruebas (Testbed), basado en un esquema similar a una red LAN, donde se involucran equipos existentes (router, switch, hosts) respetando la jerarquía de una red. Teniendo en cuenta este parámetro, se ha dado la facilidad de emular un ambiente ideal para el desarrollo y análisis del ABw; dado que en una red ordinaria implicaría errores por la calidad de los enlaces, funcionamiento y tráfico existente en una topología con características similares.

Para el desarrollo de las pruebas de los tres estimadores (Traceband, Iperf e Ifstat); fue necesario buscar una forma de congestión del canal porque en el instante que no existía tráfico, se buscó el comportamiento de las herramientas en diferentes intervalos de tiempo; sin embargo, por otro lado se necesitaba evidenciar, estudiar y analizar que sucedía al momento de generar tráfico; razón por la cual se encontró la forma de inducirlo a partir de Mgen como tráfico sintético.

De igual forma, para introducir tráfico cruzado en el ambiente controlado, se encontró como resultado una independencia entre las diferentes redes; es decir, si una subred esta congestionada no tiene por qué afectar el funcionamiento de las demás subredes. Lo anterior, resaltando que esto involucra recursos por parte de los equipos necesarios del enlace (switch y router). De esta manera, al realizar medición de ancho de banda, se observó un decremento dependiendo de la cantidad de tráfico y carga de los equipos adscritos a la red; de modo que el ambiente que se diseñó garantiza unas condiciones muy similares a una red real con la misma topología.

En el momento de la adquisición de datos de las diferentes pruebas, se puede afirmar que en comparación con Ifstat; Iperf y Traceband son las herramientas más rápidas. Iperf, con un promedio de medición ajustable por el usuario; aclarando que esto se debe a que la herramienta Ifstat no ofrece un valor estimado del tiempo tomado para realizar la medición y su valor por defecto siempre es 1 segundo. Traceband, brinda un mayor monitoreo dado que realiza cada estimación en aproximadamente 0.5 segundos y como entrega en paralelo un porcentaje de overhead ausencia en la mayoría de los estimadores. Teniendo en cuenta dicho antecedente, las mediciones por Iperf y Traceband son más precisas ya que la tasa de muestreo es ajustable.

En el momento de realizar una comparación de los tres estimadores (Traceband, Iperf e Ifstat), se difiere que son herramientas muy completas para la toma de estimación de ancho de banda; sobresaliendo Traceband en el porcentaje de overhead de cada estimación; Ifstat en la entrega de datos con puntos decimales al momento de existir tráfico; y finalmente Iperf con su facilidad de ajustarse a diferentes parámetros de medida tanto en tiempo, capacidad y lectura de datos.

También, se encontró que la aplicación más sensible a cambios minúsculos en el ancho de banda disponible es Iperf, y la menos acertada en la medición sin tráfico es Ifstat pues cuando el canal está ausente de tráfico su valor es constante durante toda la medición. Resaltando, que la herramienta que presenta menor precisión en las mediciones cuando se aplica tráfico cruzado es Ifstat; pues entrega un valor constante durante toda la medición, lo cual no puede facilitar una información contundente sobre el estado real de la red; mientras que Traceband e Iperf son aplicaciones más robustas que entrega en detalle el comportamiento del ABw.

Es de resaltar que el cross-trafic inducido a partir de Mgen en la medición del ancho de banda es fundamental; dado que esta herramienta permite al usuario generar el tráfico sintético en cantidad y volumen de paquetes y datos. Sin embargo, se asignó únicamente tráfico sintético periódico, con el fin de garantizar carga en la red de datos y poder realizar mediciones constantes en intervalos grandes de tiempo. Sin embargo, Mgen es una aplicación que permite asignar el tipo de tráfico bien sea periódico (hace referencia a ser constante), burst (un tráfico de ráfagas en instantes de tiempos) o poisson (que se asemeja al trafico encontrado en una red ordinaria).

Finalmente en todo este proceso de investigación, los principales autores de este trabajo consideran que existe gran variedad de herramientas para el estudio y análisis del ancho de banda disponible en una red; desde el hecho de observar valores como capacidad o transferencia (Ifstat); otras con más variables como transferencia y porcentaje utilizado en máquina (Traceband); y herramientas un poco más robustas y flexibles mostrando capacidad, tiempos, jitter, transferencia, entre otros (Iperf); dando como resultado final que depende del estudio que se realice se utilice el estimador o estimadores adecuados.

10. RECOMENDACIONES

Existen otros recursos importantes además del ancho de banda que se pueden tener en cuenta para la congestión en una red de datos; como lo es tiempo de retransmisiones, la capacidad de los buffers en cada dispositivo o el manejo de capacidad de procesamiento de dichos elementos. Dando como resultado que el hardware involucrado también juega un papel fundamental en el rendimiento de una red. Para un caso más específico, en esta investigación se utilizó equipos Cisco®, populares por su robustez y capacidad de procesamiento. De igual forma, para el estudio adecuado del ABw fue necesario contar con elementos de hardware que garantizaran condiciones óptimas; como parte fundamental en que las mediciones fueran correctas. Cabe resaltar, que si las mediciones realizadas en esta investigación se hubiesen recreado en otros ambientes y con otros equipos, los resultados se verían claramente distintos y tal vez no tan fiables que con estos sistemas.

Al momento de la saturación con datos en el canal, se buscó diferentes medios o estrategias para lograr un resultado significativo. En primer lugar, con el intercambio de ficheros de gran peso en el orden de los GB; sin embargo se encontró que no era viable, pues la velocidad de transmisión no era constante por el trabajo del disco duro al momento de accesar a la información y paralelo a esto, las mediciones hechas por los estimadores no eran confiables al no tener certeza de la carga existente en la red. Seguido de esto, se intentó realizar cross-trafic a partir de la aplicación VLC, el cual permite observar un video desde un equipo remoto; por tal razón, se realizó la implementación utilizando un filme en 3d para garantizar un flujo masivo de datos; pero se evidenció que por el tipo de códec y eficiencia del mismo el ancho de banda disponible no se redujo de manera considerable; y los estimadores no reflejaban un cambio significativo en ancho de banda disponible; por tal motivo se concluyó que es importante tener una aplicación que garantice un ambiente controlado del tráfico en la red; que para este caso fue Mgen.

Por otro lado, en un principio se optó por utilizar máquinas virtuales en los equipos; sin embargo, al momento de realizar las pruebas la carga del procesador de cada computador estaba sobre el 70% de usabilidad, por lo que creó escepticismo en la calidad y veracidad de los datos entregados por los estimadores. Por esta razón, se decidió instalar Linux en su distribución Ubuntu desde la raíz de los equipos con el fin de generar comparaciones en la usabilidad del procesador y la medición de los estimadores; dando como respuesta resultados contundentes y fiables. El hecho de que exista un sistema operativo nativo y sobre este, crear una máquina virtual da como resultado que los recursos de hardware estén saturados y las mediciones de los estimadores sean muy diferentes; por tal motivo hay que generar condiciones apropiadas para garantizar la fiabilidad de las mediciones. De este apartado, se puede recomendar el uso de computadores con requisitos mínimos de hardware como: procesador core 2 duo o superior, mínimo de 2 GB de RAM y un disco duro de al menos 20GB junto con una tarjeta de red con velocidad de 10/100 Mbps.

BIBLIOGRAFÍA

Allman, M., & Paxson, V. (1999). “On Estimating End-to-End Network Parameters,”. *ACM SIGCOMM*.

Capone, A., & Martignon, L. (2010). Bandwidth Estimation Schemes for TCP over Netwoks. *IEEE Xplore*.

Chaparro, F., Guerrero, C., & Fraile, F. (2013). Available Bandwidth Estimation for High Quality Television Content. *IEEE Xplore*.

Cisco 1841 spects. (2015). Obtenido de http://www.cisco.com/c/en/us/products/routers/1841-integrated-services-routerisr/index.html

Cisco 2911 spects. (2015). Obtenido de http://www.cisco.com/c/en/us/products/routers/2911-integrated-services-router-isr/index.html

Comparasion of Public end-to-end Bandwidth Estimation tools on high speed links . (2005).

Escorsa Castells, P., & Valls, J. (2009). *Tecnología e innovación en la empresa.*

Guerrero, C. (2009). Traceband: Available Bandidth Estimation based on a Hidden Markov Mode.

Guerrero, C., & Labrador, M. (2010). Traceband: A fast, low overhead and accurate tool for available bandwidth estimation and monitoring. *IEEE Xplore*.

Guerrero, C., & Labrador, M. A. (2010). "On the Applicability of Available Bandwidth Estimation Techniques and Tools”. *IEEE Xplore*.

Hu, N., & Steenkiste, P. (2003). *Evaluation and characterization of available bandwidth probing techniques.*

Jordan, V., & Galperín, H. (2013). *Acelerando la revolución digital.*

López, Y. (2010). *Evaluacion de técnicas de medición de ancho de banda disponibles en una red de computadoras.*

Marin Carrasquilla, Ulloque Rodríguez, & Guerrero, C. (2012). *Evaluación de técnicas de medición de ancho de banda disponible.* Bucaramanga.

Marin Moreno, W. (1997). *Modelo Osi.*

Marin Moreno, W. (1997). *Modelo OSI.*

Méndez, C. (2001). *Metodología, diseño y desarrollo del proceso de investigación.*

Mgen Navymil. (2015). Obtenido de http://www.nrl.navy.mil/itd/ncs/products/mgen

Noregistra. (2013).

Oshiba, T., & Nakajima, K. (2014). Quick End-to-End Available Bandwidth Estimation for QoS of Real-Time Multimedia Communication. *IEEE Xplore*.

Pathchirp. (2014). *Measuring Available Bandwidth.*

Reggini, H. (2009). *La tecnología y la gente en tiempos de internet.*

Ruiz, F. (2012). *Sociedad de la formación y la educación.*

Salcedo Morillo, D. D. (2011). *Implementación y evaluación de un algoritmo de clustering en un estimador de ancho de banda disponible.* Bucaramanga.

Stallings, W. (1999). *Comunicaciones y Redes de Computadoras, 7ma Edición.*

Steenkiste. (2013). *Study of packet level UDP performance of NAT44, NAT 64 and IPV6 using Iperf.*

Vega Lebrun, C., & Arvizu Gutierrez, D. (2011). Algoritmos para Encriptación de Datos.

ANEXOS

Anexo 1**.** Libro de Excel® para Traceband® en 60 seg. sin tráfico.

Anexo 2**.** Libro de Excel® para Traceband® en 300 seg. sin tráfico.

Anexo 3**.** m-file en Matlab® para Traceband® en 300 seg. sin tráfico.

Anexo 4**.** Libro de Excel® para Traceband® en 900 seg. sin tráfico.

Anexo 5**.** m-file en Matlab® para Traceband® en 900 seg. sin tráfico.

Anexo 6**.** Libro de Excel® para Traceband® en 60 seg. con tráfico.

Anexo 7**.** Libro de Excel® para Traceband® en 300 seg. con tráfico.

Anexo 8**.** m-file en Matlab® para Traceband® en 300 seg. con tráfico.

Anexo 9**.** Libro de Excel® para Traceband® en 900 seg. con tráfico.

Anexo 10**.** m-file en Matlab® para Traceband® en 900 seg. con tráfico.

Anexo 11**.** Libro de Excel® para Iperf® en 60 seg. sin tráfico.

Anexo 12**.** Libro de Excel® para Iperf® en 300 seg. sin tráfico.

Anexo 13**.** m-file en Matlab® para Iperf® en 300 seg. sin tráfico.

Anexo 14**.** Libro de Excel® para Iperf® en 900 seg. sin tráfico.

Anexo 15**.** m-file en Matlab® para Iperf® en 900 seg. sin tráfico.

Anexo 16**.** Libro de Excel® para Iperf® en 60 seg. con tráfico.

Anexo 17**.** Libro de Excel® para Iperf® en 300 seg. con tráfico.

Anexo 18**.** m-file en Matlab® para Iperf® en 300 seg. con tráfico.

Anexo 19**.** Libro de Excel® para Iperf® en 900 seg. con tráfico.

Anexo 20**.** m-file en Matlab® para Iperf® en 900 seg. con tráfico.

Anexo 21**.** Libro de Excel® para Ifstat® en 60 seg. sin tráfico.

Anexo 22**.** Libro de Excel® para Ifstat® en 300 seg. sin tráfico.

Anexo 23**.** m-file en Matlab® para Ifstat® en 300 seg. sin tráfico.

Anexo 24**.** Libro de Excel® para Ifstat® en 900 seg. sin tráfico.

Anexo 25**.** m-file en Matlab® para Ifstat® en 900 seg. sin tráfico.

Anexo 26**.** Libro de Excel® para Ifstat® en 60 seg. con tráfico.

Anexo 27**.** Libro de Excel® para Ifstat® en 300 seg. con tráfico.