UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:
INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la obtención del título de:
INGENIEROS DE SISTEMAS

TEMA:
ANÁLISIS Y DISEÑO DE UNA RED MÓVIL AD HOC BASADA EN EL PROTOCOLO DE ENRUTAMIENTO AODV USANDO LA HERRAMIENTA DE SIMULACIÓN NS-2 PARA EL SECTOR DE SAN MARTÍN.

AUTORES:
JAIME ANÍBAL ACOSTA RODRÍGUEZ
JHONATAN GUILLERMO QUELAL MAFLA

TUTOR:
JORGE ENRIQUE LÓPEZ LOGACHO

Quito, julio de 2016

Cesión de derechos de autor

Nosotros, Jaime Aníbal Acosta Rodríguez y Jhonatan Guillermo Quelal Mafla con cédula de identificación N° 1719766766 y 1717597346 respectivamente, manifestamos nuestra voluntad de ceder a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: ANÁLISIS Y DISEÑO DE UNA RED MÓVIL AD HOC BASADA EN EL PROTOCOLO DE ENRUTAMIENTO AODV USANDO LA HERRAMIENTA DE SIMULACIÓN NS-2 PARA EL SECTOR DE SAN MARTÍN, mismo que ha sido desarrollado para optar por el título de: Ingenieros de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, julio de 2016

Jaime Aníbal Acosta Rodríguez
CI: 1719766766

Jhonatan Guillermo Quelal Mafla
CI: 1717597346

Declaratoria de coautoría del docente tutor

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación ANÁLISIS Y DISEÑO DE UNA RED MÓVIL AD HOC BASADA EN EL PROTOCOLO DE ENRUTAMIENTO AODV USANDO LA HERRAMIENTA DE SIMULACIÓN NS-2 PARA EL SECTOR DE SAN MARTÍN realizado por Jaime Aníbal Acosta Rodríguez y Jhonatan Guillermo Quelal Mafla obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerado como trabajo final de titulación.

Quito, julio de 2016

Jorge Enrique López Logacho
CI: 1712082484

DEDICATORIA

A mis padres Etelvina y Guillermo que no dudaron ni por un segundo que culminaría este gran paso en mi vida, gracias por creer en mí, que pese a varios momentos donde todo parecía inalcanzable y complicado seguíamos adelante; a mis hermanas Doris y Andrea que siempre serán pilares fundamentales en mi familia, con su esfuerzo y dedicación incentivaron en mi mentalidad que nada es imposible; a mis sobrinos y cuñado, que a través de ellos puede ver que la vida no se basa en una sola persona sino en el apoyo de los que te rodean. Por último, gracias a familiares y amigos los que nos apoyaron con ese grano de arena que daba aliento para seguir adelante. A todos ellos mis más sinceros agradecimientos.

Jhonatan Quelal

El desarrollo del presente trabajo de titulación va dedicado de manera especial a mis padres Jaime Acosta y Mariana Rodríguez, quienes estuvieron siempre a mi lado brindándome su apoyo, dándome el ejemplo de superación y sacrificio para que pueda cumplir con mis metas, por fomentar en mí principios morales y deseos de superación en la vida; a mi hermana que siempre ha estado a mi lado brindándome palabras de aliento para cumplir con mis ideales.

Jaime Acosta

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

Este proyecto busca observar el comportamiento de la red ad hoc en un entorno real, debido a esto se muestran todos los procesos necesarios para la simulación de una red Ad Hoc mediante el protocolo AODV, en el cual se detalla conceptos que son necesarios para comprender su funcionamiento, la elección del sector de San Martín para la obtención de datos fue debido a la ausencia de una buena calidad de servicio en cuanto a Internet ofrecida por los diferentes proveedores, en este escenario se obtiene la información que participará como base de la solución de la red simulada, estos datos son número de población obtenido del último censo, el área del sector y curvas de nivel del terreno. Para efectuar la simulación se usa la herramienta Network Simulator 2 (NS-2), este es un programa dedicado para la simulación de redes ya que muestra resultados para que puedan ser analizados y así permitir la toma de decisiones, para dichos análisis se emplea el programa NS-2 Wireless Trace Analyzer, el cual ayuda a discernir los datos entregados por el simulador NS-2 y ayuda a obtener gráficas que permitan observar el comportamiento de la red.

ABSTRACT

This project seeks to observe the ad hoc network's behavior in a real environment, so it shows all the processes required for the simulation of an Ad Hoc network using AODV protocol. This project details concepts needed to understand the operation of the network. The choice of San Martín-Quito area data collection was due to the absence of a good quality Internet service offered by different suppliers, this information in the scenary involved as the basis of the simulated network solution. These data are: number population's number from the last census, the area of the sector and terrain contour lines. To make the simulation the Network Simulator 2 (NS-2) tool is used, this is a program dedicated to network simulation and shows results so they can be analyzed and allow decision-making, for such analysis is used the NS-2 Wireless Trace Analyzer program, which helps to discern the data delivered by the NS-2 simulator and it helps to get graphs that allow observe the network's behavior.

INTRODUCCIÓN

Mucho tiempo se creyó que las redes actuales regirían el mercado para dar el servicio de Internet, pero a medida que se desenvuelve los diferentes dispositivos electrónicos se crean nuevas oportunidades de establecer una red dependiendo de la demanda y del tipo de usuario al que va dirigido este servicio, lo que le interesa al proveedor es obtener clientes y al mismo tiempo prestar un gran servicio, economizando los costes que alberga implementar una red para esto motiva plantear movimientos innovadores que ayuden de igual forma tanto empresa como usuario.

Las redes Ad Hoc proveen características de funcionamiento poco explotadas que brindan muchos beneficios anteriormente establecidos, además uno de los principales es minimizar el tiempo de implementación y manejar un establecimiento de red estable mostrando así una perspectiva viable al obtener resultados. La red Ad Hoc llega a escalas altas para poder competir con rivales alternas.

En el presente proyecto se detallarán todos los procesos necesarios para la simulación de una red ad hoc en el sector de San Martín, ya que debido a la poca demanda de usuarios que tienen acceso a Internet en cierto sector no se considera viable la implementación de ciertos equipos utilizados comúnmente debido a los altos costos, es por ello que las operadoras no invierten si no se tiene la visión de rentabilidad del sector.

El desarrollo del proyecto se realizará con el fin de ofrecer acceso a Internet al sector de San Martín, Quito (ubicación geográfica exacta). El proyecto ayudará a la configuración para que las operadoras ya existentes puedan hacer uso de él y extender el alcance de su cobertura.

CAPÍTULO 1

SOPORTE TEÓRICO

1.1. Redes Ad Hoc

Las redes móviles Ad Hoc o también llamadas redes MANET (Mobile Ad Hoc Networks), donde Ad hoc proviene de una locución del latín que significa "apropiado", "improvisado", "adecuado" o especialmente "dispuesto", la cual proporciona flexibilidad aprovechando los principios de auto-organización en lo que se refiere a cambios de topologías inesperadas, es decir, topologías dinámicas y su rápida solución de nuevos caminos que encuentra para enviar la información desde un nodo origen hacia el nodo destino.

Estas redes son muy utilizadas actualmente ya que no utiliza una infraestructura existente, y su funcionamiento se basa en múltiples nodos móviles no centralizados, los mismos que tienen la misma prioridad y que da como resultado una red de rápido acceso. Cuando un nodo pierde el alcance de otro busca otros dispositivos que estén conectados a la misma red lo que es conocida como conexión de múltiples nodos.

Las redes MANET tienen muchas aplicaciones entre las cuales son:

- Operaciones de rescate en zonas remotas.

Las redes móviles Ad Hoc son muy utilizadas en zonas de difícil acceso a Internet, ya que si pasa algún problema y no se cuenta con una infraestructura existente en dicha zona se ve la necesidad de intercambiar la información para poder mantener comunicación con

otros. Esta representaría una solución para establecer una red de este tipo la cual reduciría el tiempo estimado en la solución y se enfocarían en el problema.

- Cobertura local en sitios de construcción.

En este caso también son útiles ya que la instalación de una red de este tipo permitiría que los trabajadores reciban instrucciones en un sitio determinado y así de esta forma lograr optimizar el tiempo.

- Acceso inalámbrico en zonas urbanas.

Debido a la falta de acceso a Internet en ciertos lugares ya sea que por la creación de una estructura en dicho lugar tenga un costo elevado se opta por la creación de una red Ad Hoc porque es muy viable para establecer la comunicación.

En la actualidad existen muchas aplicaciones que pueden hacer uso de esta red para diferentes proyectos con la finalidad de mejorar la calidad de vida tanto en educación, salud, redes de acceso público, entretenimiento entre otros.

1.2. Protocolos de encaminamiento

Para la selección de un protocolo en una red Ad Hoc se debe tener en cuenta cada una de las diferentes características que presenta cada uno. En lo cual posteriormente se explicará el motivo del uso el protocolo AODV en el presente proyecto.

Actualmente existen varios grupos de protocolos de encaminamiento que se pueden utilizar en una red Ad Hoc. A continuación, se explicarán los diferentes grupos existentes.

- Protocolos reactivos.
- Protocolos proactivos.

- Protocolos híbridos.

1.2.1. Protocolos reactivos.

Cuando un nodo necesita una ruta para enviar la información a su destino este efectúa una búsqueda del camino a seguir para encontrar el destinatario mediante broadcast, es decir, esta búsqueda no es permanente si no que se activa cuando se necesita transferir información de un nodo a otro. El problema que se puede presentar en este proceso es el tiempo que tiene que invertir el primer paquete para encontrar la ruta necesaria o cuando se realice una retransmisión de paquetes ya sea porque se presenta algún error en la ruta. Estos protocolos no realizan una constante actualización en las tablas de enrutamiento evitando un uso innecesario en la red por lo que es recomendable su implementación en entornos cambiantes.

1.2.2. Protocolos proactivos.

Este protocolo se maneja encontrando todas las rutas posibles hacia los diferentes nodos, es decir, siempre envía paquetes de control para que cada nodo tenga conocimiento de todos los caminos factibles, aunque no exista la necesidad de enviar información de un nodo a otro para saber si la topología ha cambiado, lo que genera una exigencia continua de la red. Este modo de ejecución es el más utilizado en el envío de información.

1.2.3. Protocolo híbrido.

Este tipo de protocolo combina los protocolos reactivos y proactivos. Se usa el protocolo proactivo con nodos que están cerca debido a que existe poca posibilidad de que estos nodos cambien de posición, en cambio, los protocolos reactivos se usan con nodos lejanos porque cambian su posición con más frecuencia. (Troyano, 2011)

Entre los protocolos más destacados en las redes Ad Hoc son AODV y OLSR. El presente proyecto hace uso de una red móvil y como se mencionó anteriormente los protocolos reactivos son más recomendables para entornos cambiantes se hará uso del protocolo AODV.

1.3. AODV

Ejemplo de funcionamiento AODV

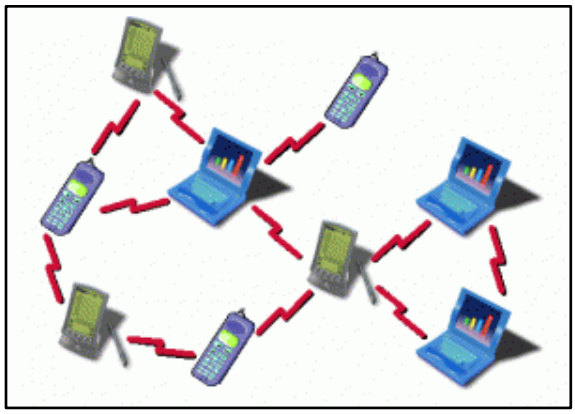

Figura 1. Ejemplo de funcionamiento AODV
Fuente: (Hasson & Jasim, 2012)

El protocolo AODV (Ad-Hoc on Demand Distance Vector) fue creado por un grupo de trabajo conformado por Charles E. Perkins, Centro de investigaciones Nokia, Elizabeth M. Belding-Royer de la Universidad de California en Santa Bárbara y Samir R. Das de la Universidad de Cincinnati aceptada en Julio del 2003.

AODV es un protocolo reactivo, lo que quiere decir es que busca los nodos requeridos para establecer una comunicación y así realizar el envío de la información solicitada sabiendo que el primer paquete genera retardos en la red hasta que descubra el camino a seguir. Una vez establecida la ruta, la información se almacena en cada uno de los nodos

que conforma dicha ruta de tal manera que si existe una pérdida de enlace en los nodos comprendidos se encargan de notificar al resto de nodos que se debe encontrar una nueva ruta.

El protocolo AODV utiliza el número de saltos que permite que el nodo origen seleccione la ruta que se considere más adecuada que va hacer utilizada para enviar la información entre nodos, además añade un número de secuencia a la información que se envía por cada ruta, dicho número de secuencia es creado por el nodo que recibe la información. Este protocolo también se caracteriza por eliminar rutas que no son actuales y para ello toma consideración del tiempo de vida que la ruta está vigente.

En el protocolo AODV es posible realizar una configuración para conectarse a routers que no usen el mismo protocolo, es decir, se puede conectar a redes externas para proveer comunicación hacia las redes internas.

La RFC-3561 manifiesta que en el protocolo AODV se definen tres tipos de mensajes que son Route Requests (RREQ) requerimiento de la ruta, Route Replies (RREPs) respuesta de una ruta y Route Errors (RERRs) error de una ruta. El protocolo que se usa para la transmisión de estos mensajes es dirigido por el protocolo de transporte UDP. Estos mensajes son dirigidos a los diferentes nodos y cada uno de ellos tomará las acciones respectivas.

A continuación, se detalla de mejor manera cada uno de los mensajes mencionados anteriormente.

1.3.1. Requerimiento de la ruta (RREQ).

El mensaje de requerimiento de ruta (RREQ) es enviado a todos los nodos que estén conectados directamente vía broadcast desde el nodo origen y se encargan de la petición de una ruta que viaja a través de toda la red buscando su destino. Cada camino que elige este mensaje es validado dependiendo si el mensaje llega al nodo destino o si encuentra un nodo intermedio que tenga una ruta hacia dicho nodo y esta pueda ser usada como una ruta viable para poder llegar a su destino sin necesidad de completar toda la ruta. Como se sabe puede haber más de un camino hacia un nodo destino, se selecciona una ruta principal y el resto de rutas se los guarda como opcionales en caso de error. (Perkins, Belding-Royer, & Das, 2003)

Este proceso comienza después de que el nodo haya buscado en su tabla si no tiene una ruta ya designada hacia un nodo destino.

Mensaje RREQ

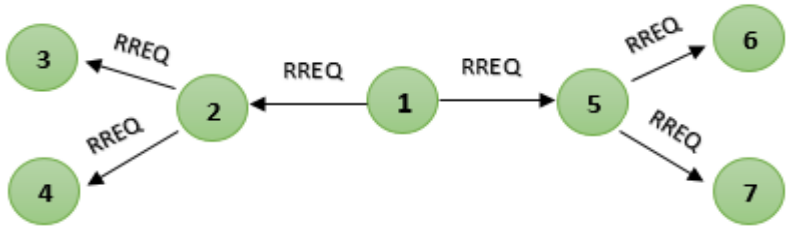

Figura 2. Mensaje RREQ
Elaborado por: Jaime Acosta y Jhonatan Quelal

Para establecer una ruta desde el nodo 1 hasta el nodo 6, el nodo 1 envía a sus nodos vecinos bajo broadcast un mensaje de requerimiento (RREQ) el cual es recibido por los nodos 2 y 5 como se muestra en la figura 2, estos nodos a su vez propagan el mensaje a

sus nodos vecinos que son los nodos 3, 4, 6 y 7 hasta encontrar el nodo destinatario. Todos los nodos que reciben los mensajes guardan en su tabla de enrutamiento el número de secuencia para identificar los caminos, la dirección IP del nodo origen y nodo destinatario para solventar problemas que pueden ocurrir o el correcto funcionamiento bidireccional de la información. Cada vez que pasa por un nodo el número de secuencia se aumenta en 1 ya que la métrica de este protocolo se basa en el menor número de saltos y es considerada como la ruta más óptima. Si el RREQ llega al nodo 2 y este nodo ya conoce la ruta con anterioridad, es decir, que el nodo 2 ya fue utilizado para enviar información al nodo 6, la petición ya es solventada y el RREQ cumpliría con su objetivo. Esto se da solamente si la ruta no ha sido borrada con anterioridad ya que el protocolo AODV tiene la característica de no mantener las rutas por más de un tiempo determinado.

1.3.1.1. Formato de mensaje.

Formato de mensaje RREQ

Figura 3. Formato de mensaje RREQ
Elaborado por: (Perkins, Belding-Royer, & Das, 2003)

A continuación, se detallarán cada uno de los campos que se observa en la figura 4:

Tabla 1. Detalle de campos del formato de mensaje RREQ

TIPO	Se refiere al tipo de mensaje que puede ser: RREQ=1, RREP=2 y RERR=3 en este caso se configurara con el número 1.
J	Bandera de unión: Campo está reservado para multicast.
R	Bandera de reparación: Campo reservado para multicast.
G	Bandera gratuita de RREP: Indica si el mensaje de respuesta debe ser unicast para la dirección IP destino del nodo especificado.
D	Bandera de único destino: Indica que el mensaje RREQ llego al destino.
U	Número de secuencia desconocido: Indica que el número de secuencia es desconocido.
RESERVADO	Envía un 0 que es ignorado por el receptor.
NÚMERO DE SALTO	Cuenta el número de saltos en cada nodo desde que se inicia el mensaje RREQ hasta que llega al destino.
ID RREQ	Número de secuencia que identifica a cada petición formada desde nodo origen junto con la dirección IP.
DIRECCIÓN IP DESTINO	Dirección IP destino a la cual se quiere tener acceso.
NÚMERO DE SECUENCIA DESTINO	El último número de secuencia recibido en el pasado por el origen, para cualquier ruta hacia el destino.

DIRECCIÓN IP ORIGEN	Dirección IP del nodo que origina la solicitud de la ruta.
NÚMERO DE SECUENCIA ORIGEN	El número de secuencia actual que se usara en la entrada de la tabla de rutas hacia el origen del mensaje RREQ.

Nota. Fuente: (Perkins, Belding-Royer, & Das, 2003)

1.3.2. Respuesta de una ruta (RREP).

Cuando el mensaje RREQ llega al nodo destinatario envía un mensaje al nodo origen, este mensaje es el RREP que es un mensaje respuesta de aceptación hacia la petición de conexión, este mensaje viaja en sentido contrario de como llego el mensaje RREQ hasta el nodo origen y así se pueda determinar cuál es la mejor ruta según su algoritmo.

Mensaje RREP

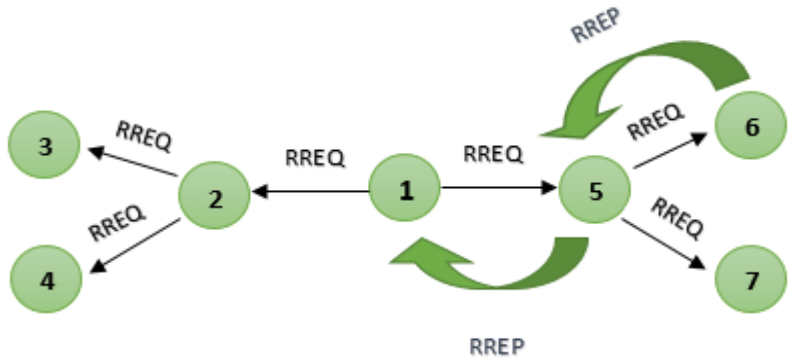

Figura 4. Mensaje RREP
Elaborado por: Jaime Acosta y Jhonatan Quelal

Cuando el mensaje RREQ ha llegado al destino que en este caso es el nodo 6 como se mostró anteriormente este nodo responde con el mensaje RREP manteniendo la misma ruta como llego el paquete RREP pero en sentido contrario como se muestra en la figura 5, este mensaje regresa con el total de saltos requeridos hasta el nodo destino, en caso de

que existiera otra ruta el nodo origen analizaría cada uno de los mensajes RREP recibidos para enviar la información por la ruta más destacada. Todos los RREP que fueron recibidos se almacenan como rutas opcionales para que en caso de tener problemas en la ruta que se seleccionó anteriormente se escogerá la siguiente ruta más destacada. Cuando el proceso de transmisión ha finalizado, es decir, que la información se envió correctamente esta ruta sigue válida por un tiempo establecido anteriormente, en caso de superar este tiempo toda la información almacenada en los nodos comprendidos es eliminada, y si se requiere enviar nueva información al mismo nodo se tiene que repetir el procedimiento.

1.3.2.1. Formato de mensaje.

Formato de mensaje RREP

Figura 5. Formato de mensaje RREP
Elaborado por: (Perkins, Belding-Royer, & Das, 2003)

A continuación, se detallarán cada uno de los campos que se observa en la figura 5:

Tabla 2. Detalle de campos del formato de mensaje RREP

TIPO	Se refiere al tipo de mensaje que puede ser: RREQ=1, RREP=2 y RERR=3 en este caso se configurara con el número 2.

R	Bandera de reparación: Campo reservado para multicast.
A	Acuse de recibo: Respuesta del mensaje del router destino hacia el nodo destino (RREP-ACK). Se pone como 'A' si la ruta hacia el destino fue completa.
TAMAÑO DEL PREFIJO	Si es diferente de 0, este prefijo de cinco bits especifica que existe una ruta alterna y dicha ruta se pondrá como opcional.
NÚMERO DE SALTO	Cuenta el número de saltos desde el nodo origen hacia el nodo destino.
DIRECCIÓN IP DESTINO	Dirección IP destino a la cual se quiere tener acceso.
NÚMERO DE SECUENCIA DESTINO	Es un número de secuencia de destino que está asociado a la ruta.
DIRECCIÓN IP ORIGEN	Dirección IP del nodo que origina la solicitud de la ruta.
TIEMPO DE VIDA	El tiempo en milisegundos, durante el cual el nodo que recibe el mensaje RREP considerará la ruta como válida.

Nota. Fuente: (Perkins, Belding-Royer, & Das, 2003)

1.3.3. Error de una ruta (RERR).

Este mensaje de error (RERR) se encarga de notificar a todos los nodos que estén involucrados en esta ruta para que de esta forma la información relaciona sea eliminada.

Las causas por las que se envía un mensaje RERR son las siguientes:

- Cuando un enlace este caído.
- Cuando un RREQ no logro obtener un RREP, es decir, que el nodo no logro obtener acceso al destino final.

- Cuando el nodo destino dejo de funcionar y se convierte inalcanzable.

Mensaje RERR

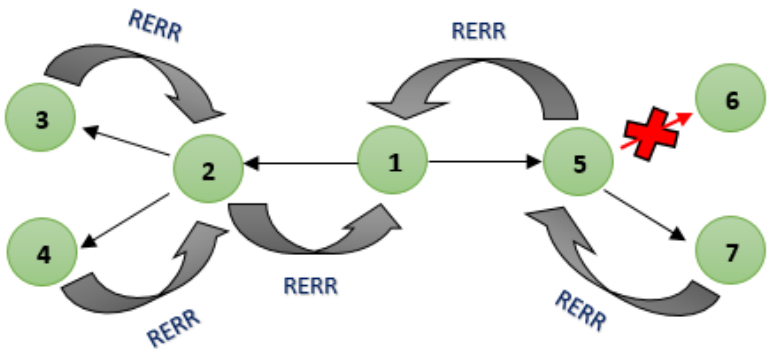

Figura 6. Mensaje RERR
Elaborado por: Jaime Acosta y Jhonatan Quelal

El mensaje de error (RERR) actúa informando a los nodos predecesores que hubo un problema en la ruta por causas anteriormente mencionadas. El nodo origen una vez que haya sido notificado con un mensaje RERR este se encarga de buscar otra ruta en la lista precursora, el nodo que envía este mensaje de error tiene más de un nodo conectado y todos estos nodos que hicieron uso de este enlace caído se los notifican vía broadcast para que puedan descartar esta ruta.

Este mensaje de error es enviado por los nodos 3, 4 y 7 en este caso, los cuales son encargados de notificar a los demás nodos como una ruta inalcanzable.

Todos los nodos que reciben el RREQ están a la espera de una respuesta, es decir, esperan un mensaje RREP y guarda un espacio en la tabla para que pueda ser llenado con dicho mensaje, en este caso como se muestra en la figura 6 son los nodos 2 y 5. Los nodos 3, 4 y 7 al no alcanzar la ruta se encargan de notificar a los nodos 2 y 5 con un mensaje de

error (RERR) que la ruta es inalcanzable con lo cual se procederá con la eliminación de la información innecesaria. Este mensaje de error solamente puede ser enviado una vez (por ruta) por los nodos con el fin de que no exista redundancia de mensajes. Entiéndase que el nodo 1, 5 y 7 comprenden un camino y el nodo 1, 5 y 6 comprenden otro camino es por eso que este enviará un mensaje de error del nodo 7 al nodo 5 y este enviará otro mensaje de error al nodo 1 y como cayó el enlace que se conecta al nodo 6 el nodo 5 generará otro mensaje de error para que en nodo 1 y 5 borren la ruta 1, 5 y 6.

1.3.3.1. Formato de mensaje.

Formato de mensaje RERR

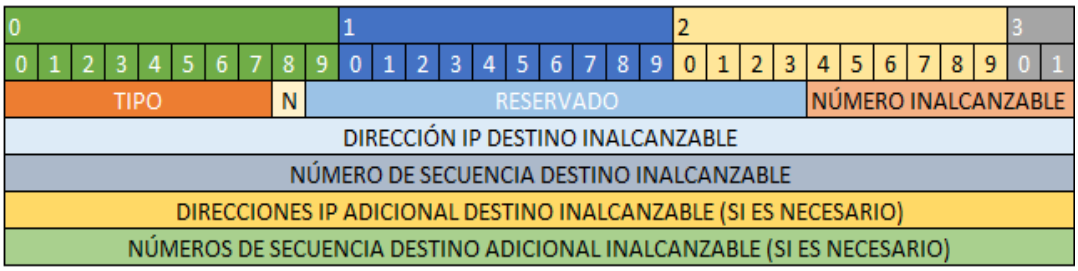

Figura 7. Formato de mensaje RERR
Elaborado por: (Perkins, Belding-Royer, & Das, 2003)

A continuación, se detallarán cada uno de los campos que se observa en la figura 7:

Tabla 3. Detalle de campos del formato de mensaje RERR

TIPO	Se refiere al tipo de mensaje que puede ser: RREQ=1, RREP=2 y RERR=3 en este caso se configurara con el número 3.
N	Bandera sin eliminar: Establece cuando un nodo ha realizado la reparación de un enlace local, los nodos de upstream no deberían borrar esa ruta.

RESERVADO	Envía un 0 que es ignorado por el receptor.
NÚMERO INALCANZABLE	Indica el número de destinos inalcanzables incluidos en el mensaje. Indica el número de destinaciones
DIRECCIÓN IP DESTINO INALCANZABLE	Destino que no se pudo alcanzar debido algún error en la ruta.
NÚMERO DE SECUENCIA DESTINO INALCANZABLE	Es un número de secuencia de destino que está asociado a la ruta.
DIRECCIONES IP ADICIONAL DESTINO INALCANZABLE (SI ES NECESARIO)	Dirección IP del nodo inalcanzable por rotura del enlace. Se pueden aumentar más campos si es necesario.
NÚMEROS DE SECUENCIA DESTINO ADICIONAL INALCANZABLE (SI ES NECESARIO)	Es un número de secuencia de destino que está asociado a la ruta. Se pueden aumentar más campos si es necesario.

Nota. Fuente: (Perkins, Belding-Royer, & Das, 2003)

1.4. Network Simulator 2 (NS2)

Network Simulator comenzó como una variante del simulador de red real en 1989 y ha evolucionado considerablemente en los últimos años. En 1995 el desarrollo NS fue apoyado por DARPA a través del proyecto VINT en LBL, Xerox PARC, UCB, y USC / ISI. Actualmente el desarrollo NS es el apoyo a través de DARPA con SAMAN y a través de NSF con CONSER, tanto en colaboración con otros investigadores. NS siempre ha incluido contribuciones substanciales de otros investigadores, incluyendo código inalámbrico de los proyectos UCB Daedelus y CMU monarca y Sun Microsystems. (Simulator, s.f.)

El propósito de este simulador es obtener datos del comportamiento tanto de redes cableadas y no cableadas, muchos de los programas de simulación de red existentes son utilizados para crear topologías, pero no brindan toda la capacidad que pueda ofrecer network simulator (NS) ya que en este simulador que trabaja bajo OTcl (Object oriented programming in Tool Comand Language) que se crea bajo los parámetros que trabaja Tcl, este crea un escenario más cercano a la realidad. En cuanto a lo que es utilizar C++ es variar o cambiar parámetros tanto de protocolos como envío de paquetes de mensajes, pérdidas de enlaces, es decir, manipular hasta tener la simulación de una red muy parecida a la realidad y por supuesto con datos más acertados, por ejemplo, los colapsos de red para hacer un análisis con la información obtenida. NS soporta fácilmente cualquier cantidad de flujo, procesa grandes velocidades y permite la edición de cualquier protocolo que se necesita utilizar. Con C++ también se puede configurar la cantidad de paquetes procesados por cada nodo, además se puede configurar los errores que existan con el fin de ver cómo actúa el protocolo en tales circunstancias.

Esquema de funcionamiento de NS-2

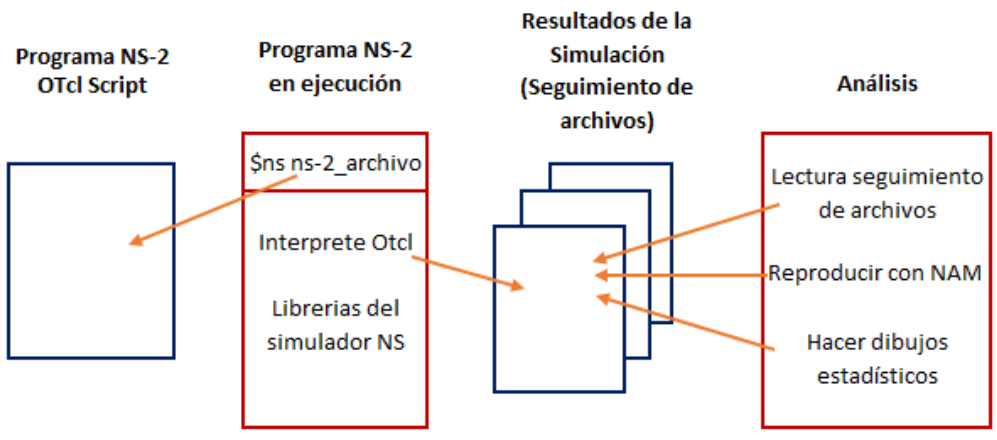

Figura 8. Esquema de funcionamiento de NS-2
Fuente: (Wang, NS-2 Tutorial, 2004)

1.4.1. C++.

Es un lenguaje de programación orientado a objetos creado por Bjarne Stroustrup muy utilizado ya que permite realizar programas de alta gama, su programación de bajo nivel es muy complicada, pero al momento de su ejecución es consistente. Es compatible con el lenguaje C.

NS-2 permite:

- La manipulación de bytes, el procesamiento de paquetes, el algoritmo de implementación.
- Una importante velocidad en tiempo de ejecución.
- El tiempo de vuelta es más lento (simulación de ejecución, encontrar errores, corregir errores, recompilar, volver a ejecutar). (Wang, 2004)

1.4.2. TCL.

Es un lenguaje de programación estructurada creado por John Ousterhout, este posee una sintaxis fácil y funcional, produce aplicaciones de alta calidad, las mismas reglas que poseen todos los tipos de lenguajes. Es multiplataforma, es decir, que puede ser utilizado en diferentes sistemas. Es un lenguaje de bajo nivel que puede ser utilizado e integrado en otras plataformas como C++ por la extensión que tiene (OTcl) puede ser usado por NS-2.

- Simulación de diferentes parámetros o configuraciones.
- Explorar rápidamente diferente número de escenarios.
- Tiempo de iteración (cambiar el modelo y repetición). (Wang, 2004)

NS utiliza el lenguaje de programación OTcl que es una versión extendida del TCL, pero con sus mismas funcionalidades.

Lenguajes de programación utilizados en NS-2

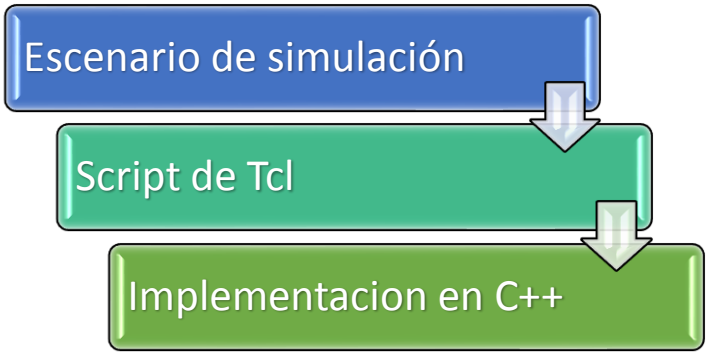

Figura 9. Lenguajes de programación utilizados en NS-2
Elaborado por: Jaime Acosta y Jhonatan Quelal

1.5. Requisitos para la solución del proyecto

Requisitos del proyecto

Figura 10. Requisitos del proyecto
Elaborado por: Jaime Acosta y Jhonatan Quelal

En esta parte se describen los requisitos necesarios para la solución del problema, para ello se realizará un estudio necesario de las diferentes situaciones que se presentan en la zona, es por eso la importancia de la selección del diseño óptimo que pueda satisfacer de mejor manera todos los inconvenientes presentados en el sector.

Para satisfacer los requerimientos del proyecto en desarrollo será necesaria la recolección de información relativa al sector de San Martín:

- Es necesario determinar el área del sector para ofrecer la total cobertura de la misma.
- Se deberá recopilar la información de la cantidad de población que exista en el sector para tener una perspectiva de cuantas personas podrían hacer uso de la red.
- Identificar la ubicación de los puntos de mayor elevación en el sector.
- Finalmente decidir la ubicación de los nodos de acceso que serán necesarios para complementar la topología.

1.6. Requisitos de la plataforma de simulación

Para cumplir de la mejor manera el diseño de simulación se hará uso de una máquina virtual (Ubuntu Linux) con la ayuda del programa VMware, en el mismo que se instalará la herramienta NS-2 (Network Simulator 2) con el que se va a realizar la simulación de red. Se va hacer uso de este software ya que posee una licencia de distribución libre y por su flexibilidad permite realizar modificaciones pertinentes por ser de código abierto, también se puede mencionar que la herramienta de simulación NS-2 presenta la característica multiplataforma, es decir, que puede ser usada en los diferentes Sistemas Operativos existentes sin verse limitados a usar alguno en específico.

Esta herramienta facilitará el diseño de la red Ad-Hoc ya que en el mencionado simulador se puede configurar el protocolo a usarse que es AODV (Ad-hoc On-demand Distance Vector routing), en este se podrá visualizar el comportamiento de los diferentes nodos, bajo esta simulación se realizará el análisis de datos que llevaran a realizar los cambios pertinentes para llegar al correcto funcionamiento de la red.

CAPÍTULO 2

RECOLECCIÓN DE DATOS

En este capítulo se detallará la información de la zona San Martín, ubicado en el sur de Quito, la información del número de población se obtiene en base a los resultados del último censo que fue realizado en el año 2010. El programa Fields Area Measure Free proveerá el área que se tendrá que cubrir en el sector; la visualización de alturas se realizará con la herramienta Google Earth y Global Mapper para poder validar la colocación de nodos.

2.1. PROGRAMAS A USARSE

2.1.1. Fields Area Measure Free.

Esta aplicación es muy útil para medir distancias, áreas y perímetros de un terreno, es por eso que se va a usar para la medición del área y determinar los límites de la zona San Martín.

2.1.2. Google Earth.

Google Earth es un programa de mucha ayuda ya que permite visualizar imágenes satelitales a nivel mundial, tomar medidas ya sea de altitud, longitud, elevación de cualquier lugar, en este caso se lo utilizará para obtener información de la altura que existe en la zona San Martín.

2.1.3. Global Mapper.

El programa Global Mapper puede ser usado con distintos propósitos, como diseños de curvas de nivel, apertura de mapas topográficos, diseño de mapas con perspectivas 3D, soporta diferentes formatos entre ellos cargar áreas de Google Earth y de esta forma apreciar diferentes latitudes y longitudes teniendo así una gran facilidad de mostrar elevaciones del terreno.

2.1.4. AutoCAD.

AutoCAD es una potente herramienta para recrear planos u objetos en graficas 2D o modelos 3D de manera fácil.

2.2. INFORMACIÓN

Los datos a recolectar son necesarios para poderlos incluir como parámetros de los cuales se va a basar la simulación.

2.2.1. Área del sector.

Para saber el área de cobertura que se va a cubrir en la zona de San Martín se debe obtener el total de la expansión de red, sin tomar en cuenta los diferentes niveles del terreno. La solución es la implementación de una antena omnidireccional (nodo principal) la cual va estar ubicado en un punto estratégico del sector el mismo que se va expandir con la utilización de cada usuario que posea el servicio, en caso de tener un espacio vacío y que no se logre una señal apropiada se implementará otra antena, cabe recalcar que no es necesario que las antenas que se van a instalar necesiten una línea de vista ya que cada antena es independiente de otra. La misma red Ad-Hoc se encarga se abrirse campo y

proporcionar señal a otros usuarios (nodos secundarios) pero se tomará a la otra antena como un respaldo de la red, con el fin de aumentar la cobertura de la red.

En la figura 28 se observa la cartografía descargada de la página del INEC, esta figura muestra la zona San Martín dividida en 10 subsectores delimitada por las otras zonas del sector en la parte Sur de Quito.

Cartografía de la zona San Martín

Figura 11. Cartografía de la zona San Martín
Elaborado por: http://www.ecuadorencifras.gob.ec/

Área de San Martín

Figura 12. Área de San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Como se muestra en la figura 12, la cual que fue obtenida del programa Fields Area Mesure Free, muestra la cobertura que tiene la zona de San Martín, lo que indica que tiene un área igual a 0,55 Km^2 lo que es igual a 550000 m^2.

2.2.2. Ubicación de los puntos de mayor elevación en la zona San Martín.

Para la ubicación del nodo principal y secundario se hará uso del programa Google Earth, Global Mapper 16 y AutoCAD 2014, los cuales permitirán establecer los puntos con mayor precisión tanto latitud como longitud con la finalidad de tener el lugar más apropiado para la emitir la señal de acceso a Internet, es decir, lograr la mayor cobertura posible en el sector.

Polígono de extracción en la zona San Martín

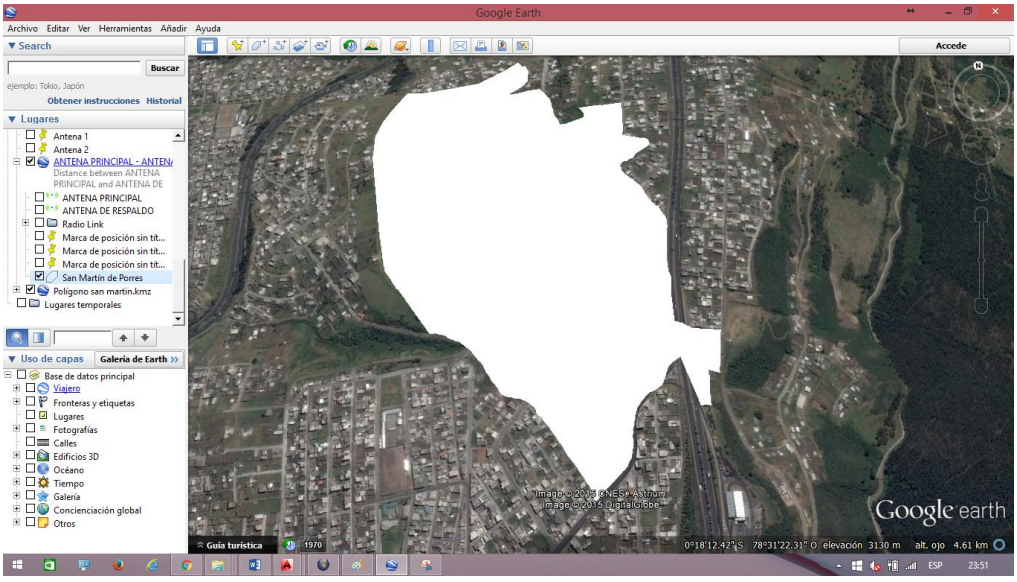

Figura 13. Polígono de extracción en la zona San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Con la ayuda del programa Google Earth se extrajo la zona San Martín (figura 30) para posteriormente ser cargada en el programa Global Mapper (figura 31), el cual es utilizado para generar las curvas de nivel con su respectiva altitud, longitud y elevación.

Global Mapper sector San Martín

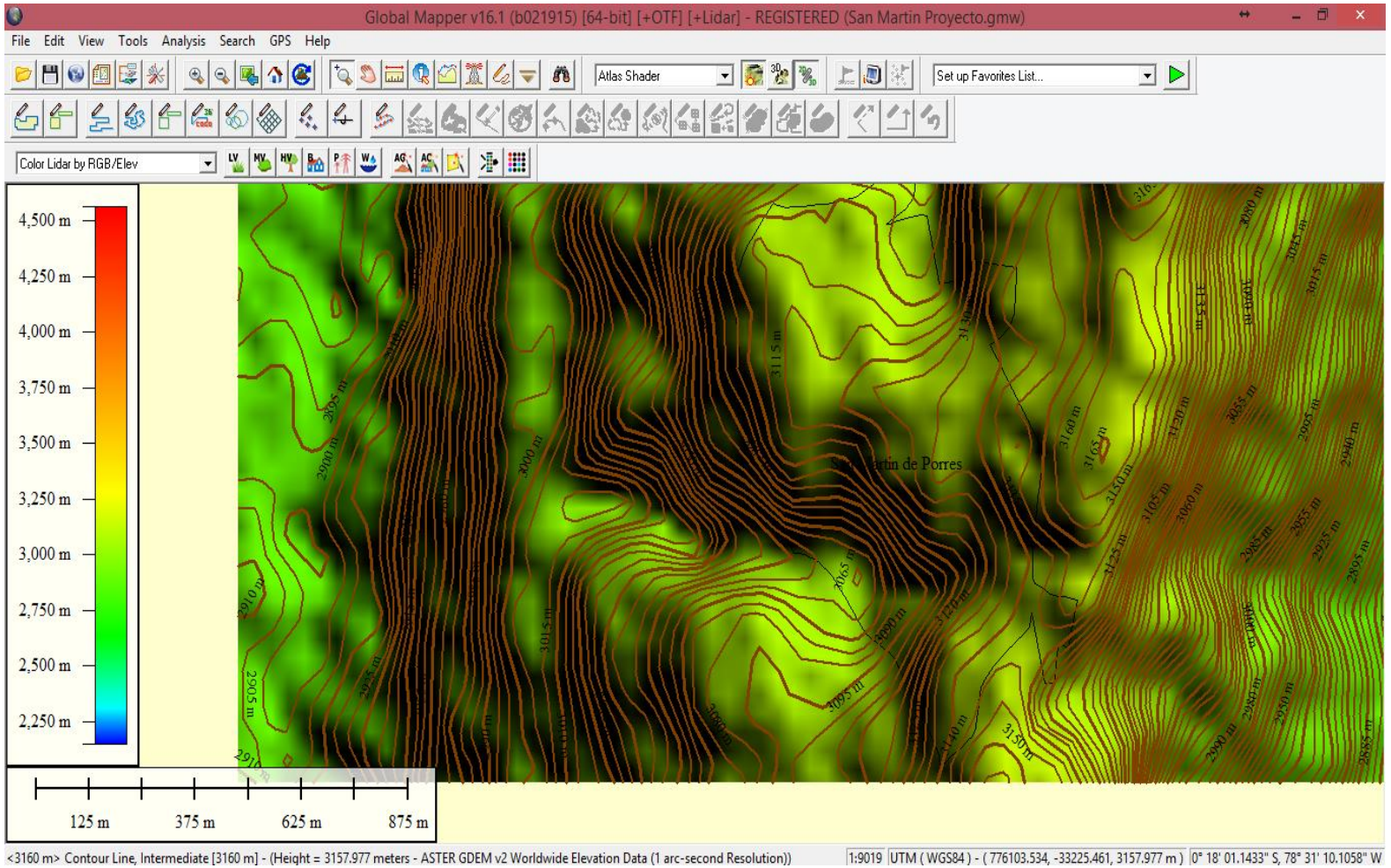

Figura 14. Global Mapper sector San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Global Mapper proporciona la opción de exportar el diseño a un formato que puede ser leído por AutoCAD donde se observan las distintas elevaciones que posee el terreno.

Curvas de nivel San Martín

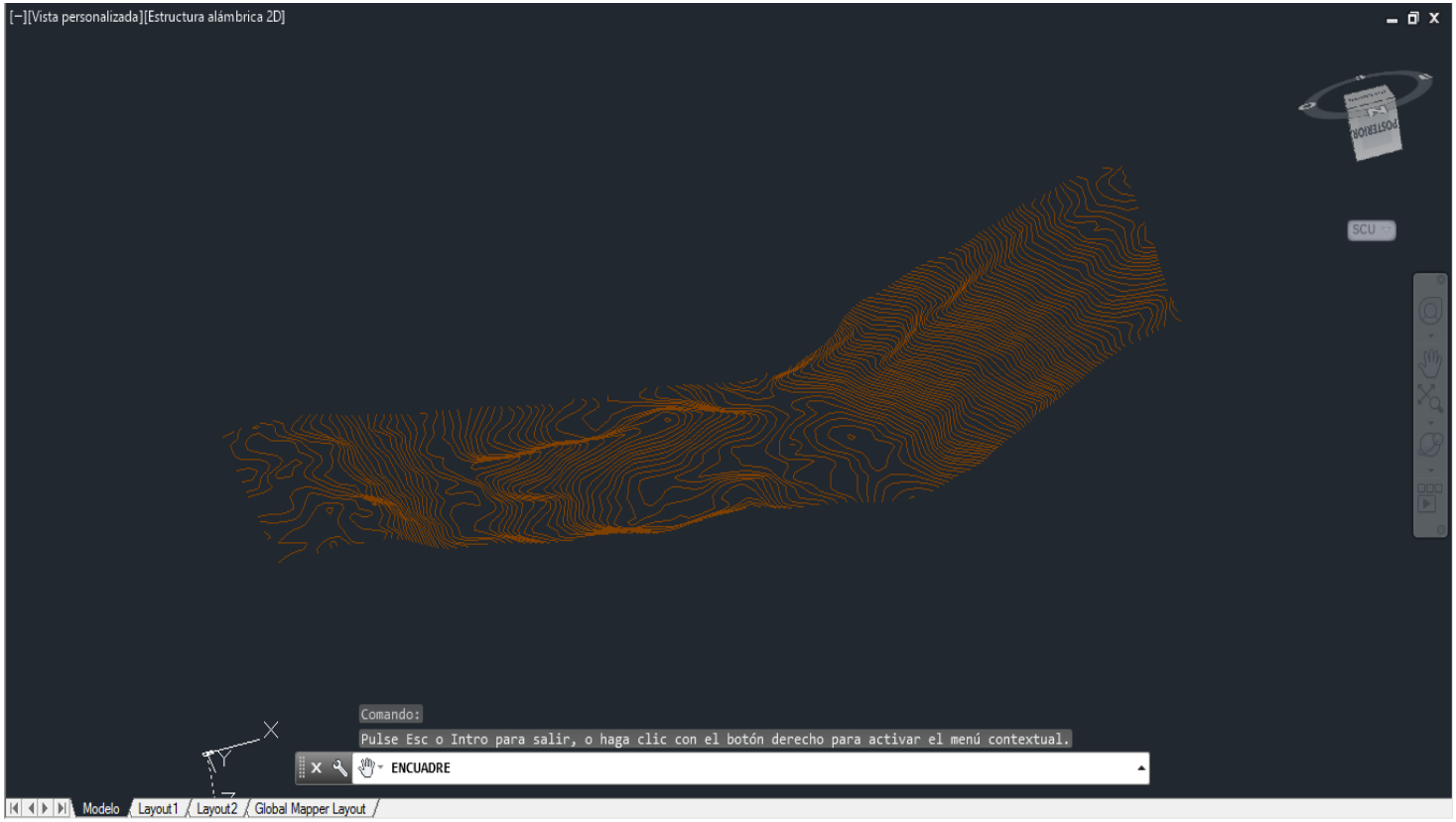

Figura 15. Curvas de nivel San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Elevaciones del terreno

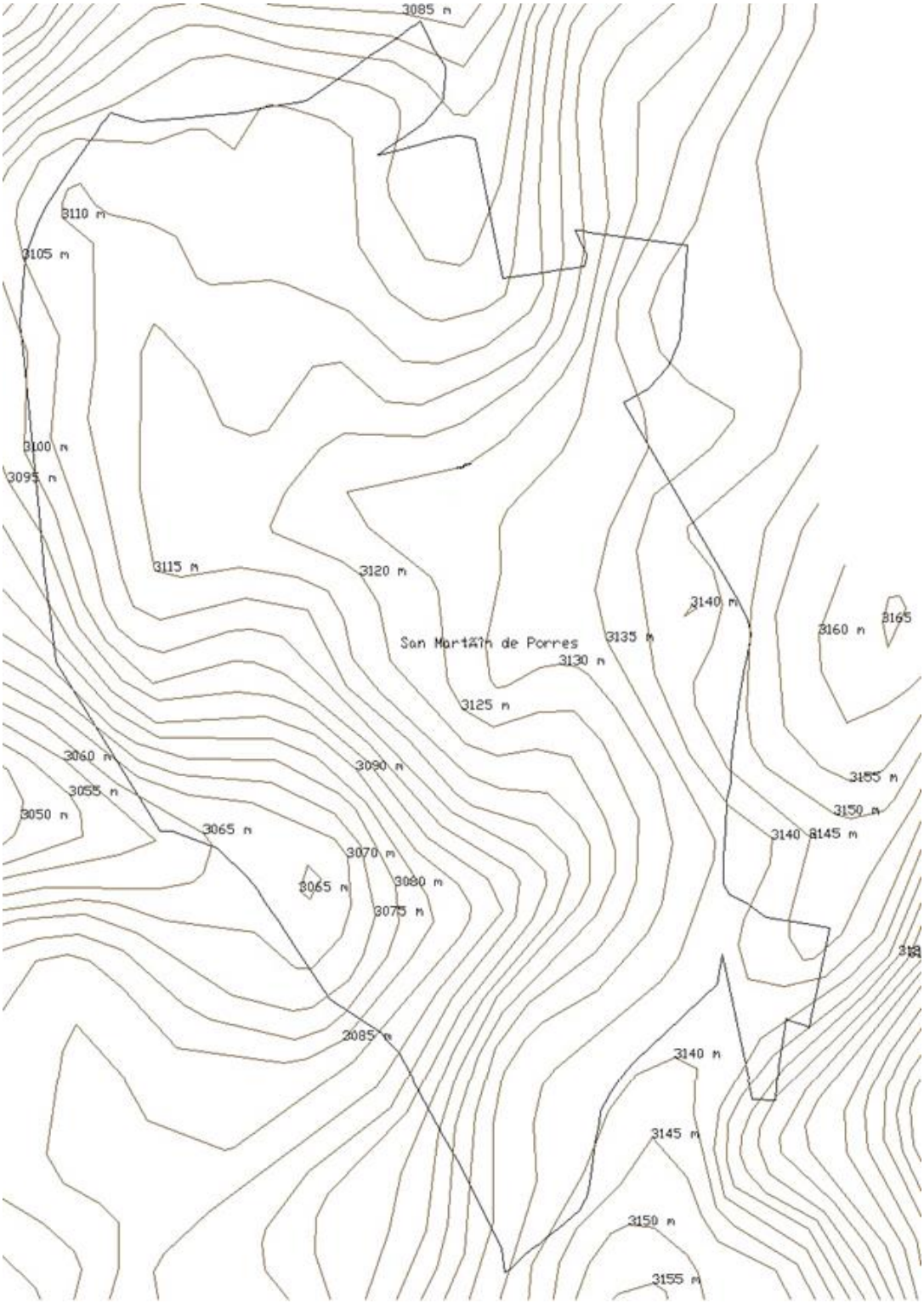

Figura 16. Elevaciones del terreno
Elaborado por: Jaime Acosta y Jhonatan Quelal

AODV presenta la facilidad que la red crezca de manera que si un usuario está conectado la señal se extiende hacia otros usuarios y estos usuarios que reciben la señal emiten hacia otros usuarios y así sucesivamente, con esto se logra que la red se vaya expandiendo conforme se vaya utilizando, es por eso que se colocará de ser necesario dos antenas de respaldo, esto no significa que la red decaiga si no que ayuda a disminuir los problemas, en el caso de que un nodo presente una señal baja ya sea porque está alejado del nodo principal este se conectará automáticamente al nodo de respaldo más cercano. Estas antenas de respaldo también ayudarán a disminuir latencias debido a que los usuarios tendrán más posibilidades de conectarse a una antena más cercana. Estas antenas ayudan a que la red tenga una alta disponibilidad.

Para la colocación de las antenas se tienen que tomar en cuenta los puntos más altos en el sector, pero como se observan los puntos más altos se encuentran ubicados en el límite de la zona, por lo cual la ubicación de las antenas en esa posición sería inadecuada debido a que la antena a usarse es omnidireccional y cubriría una zona que no corresponde a San Martín, es por eso que se tomó los puntos de mayor elevación en el centro del sector. La curva de nivel con mayor elevación en el centro del sector es 3125m a nivel del suelo, una vez localizado el punto más alto se pudo observar edificaciones (ver figura 16).

2.2.2.1. Antena principal.

En la figura 35 se aprecia el primer punto con mayor elevación en la zona, el cual tiene las siguientes coordenadas:

Latitud 0°18’02.47” S.

Longitud 78°31’25.90” O.

Elevación 3140 m.

Coordenadas y elevación del primer punto más alto en el centro de la zona San Martín

Figura 17. Coordenadas y elevación del primer punto más alto en el centro de la zona San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Lugar del primer punto más alto

Figura 18. Lugar del primer punto más alto
Elaborado por: Jaime Acosta y Jhonatan Quelal

En la figura 36 se puede observar el lugar exacto de las coordenadas del primer punto con mayor elevación en el centro de la Zona.

2.2.2.2. Antena de respaldo.

Coordenadas y elevación del segundo punto más alto en la zona de San Martín

Figura 19. Coordenadas y elevación del segundo punto más alto en la zona de San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

En la figura 37 se aprecia el segundo punto con mayor elevación referenciado en el centro de la Zona, el cual tiene las siguientes coordenadas:

Latitud 0°17’58.79” S.

Longitud 78°31’34.64” O.

Elevación 3126 m.

Lugar del segundo punto más alto

Figura 20. Lugar del segundo punto más alto
Elaborado por: Jaime Acosta y Jhonatan Quelal

En la figura 38 se puede observar la localización exacta de las coordenadas del segundo punto con mayor elevación referenciados con el centro de la zona.

Ubicación de las antenas en los puntos más altos de San Martín

Figura 21. Ubicación de las antenas en los puntos más altos de San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

2.2.3. Población.

El Instituto Nacional de Estadística y Censos (INEC) provee toda la información en cuanto a población y vivienda, en base al último censo realizado.

Página principal del INEC

Figura 22. Página principal del INEC
Fuente: (INEC, 2016)

Una vez en la página dirigirse a la pestaña en el índice de árbol con el nombre Población y Demografía. Posteriormente observar en información técnica existe el hipervínculo Base de datos que dirigirá a la siguiente página, en la cual se selecciona en la parte de inicio el último censo realizado y se procede a dar clic en sector, en esta página se opta por la base de datos SPSS de la provincia de pichincha como se muestra en la figura 22.

Descarga de la base de datos SPSS de la provincia de Pichincha

Figura 23. Descarga de la base de datos SPSS de la provincia de Pichincha
Fuente: (INEC, 2016)

El uso del programa SPSS fue utilizado para filtrar la información de la zona 406 que corresponde al sector San Martín de Porres (Z-406), esta zona comprende 10 subsectores según la distribución realizado por el municipio metropolitano de Quito. Para sacar la población del sector se filtra la edad desde los 5 años en adelante, ya que desde esa edad son considerados como personas activas en el uso del Internet.

Plano censal de la parroquia Quitumbe

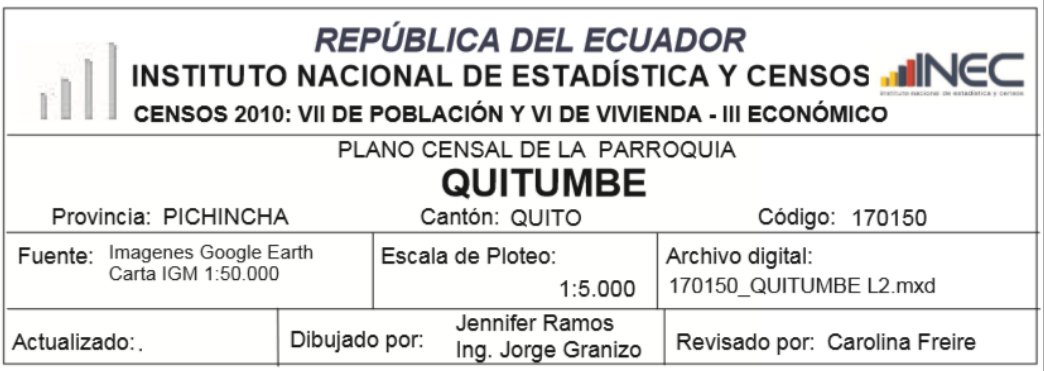

Figura 24. Plano censal de la parroquia Quitumbe
Fuente: (INEC, 2016)

Tabla 4. Población de cada subsector de la zona San Martín

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido 1	287	10,7	10,7	10,7
2	313	11,6	11,6	22,3
3	281	10,4	10,4	32,8
4	285	10,6	10,6	43,3
5	212	7,9	7,9	51,2
6	191	7,1	7,1	58,3
7	261	9,7	9,7	68,0
8	279	10,4	10,4	78,4
9	323	12,0	12,0	90,4
10	258	9,6	9,6	100,0
Total	2690	100,0	100,0	

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

En la tabla 4 se pueden observar los 10 subsectores que comprende a la zona San Martín con su respectivo número de habitantes, por subsector y el total de habitantes.

Diagrama de barras de los subsectores que corresponden a la zona de San Martín

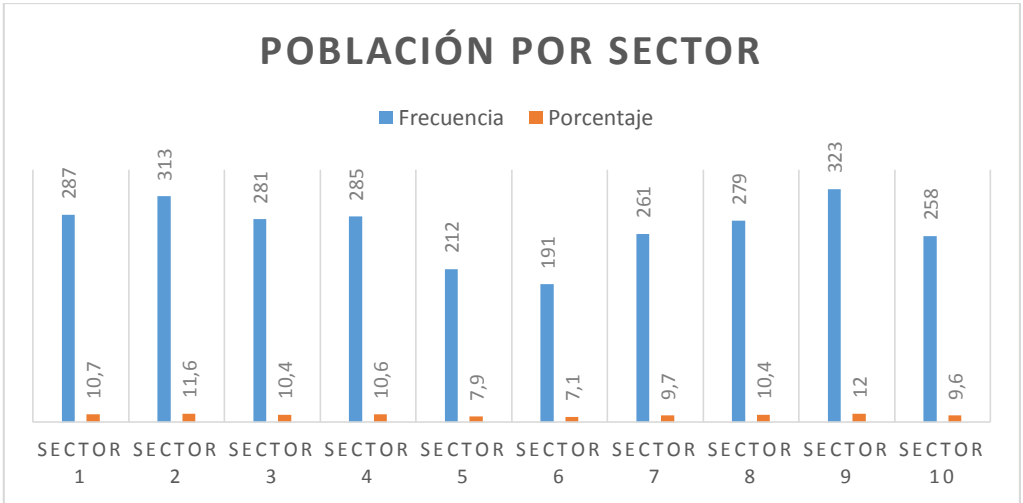

Figura 25. Diagrama de barras de los subsectores que corresponden a la zona de San Martín
Elaborado por: Jaime Acosta y Jhonatan Quelal

Este proceso de búsqueda de información se realizó debido a la necesidad de saber cuántas personas pueden llegar a hacer uso de la red, y así se estime la cantidad de usuarios, se tomará en cuenta que:

> "En Pichincha se usa 1 553 212 teléfonos celulares lo que representa el 74,2 % de la población y 1 096 016 computadores portátiles lo que representa el 52,3%". (INEC, 2010)

Según los datos anteriores se deduce que el 52,3% de la población cuenta con 2 dispositivos, mientras que 21,9% de la población hace al menos uso de un dispositivo. En base a estos resultados se toma en cuenta de que existe al menos un dispositivo por cada persona. Lo que implica que si existe 2690 personas en la zona San Martín existirá por lo menos 2690 dispositivos que acceden al Internet.

2.2.4. Muestra de la población.

La muestra estadística sirve para tomar el porcentaje mínimo del total de la población que se desea analizar, la fórmula de la muestra incluye la credibilidad e impacto que se tomará en cuenta sobre la solución que se plantea en el proyecto, matemáticamente si la muestra es satisfecha con lo propuesto da por concluido con cierto porcentaje de error que afectará de igual manera al total de la población

Para sacar la muestra se emplea la siguiente fórmula:

$$n = \frac{k^2 * p * q * N}{(e^2 * (N - 1)) + k^2 * p * q}$$

Donde:

- **n**: Número de la muestra.
- **N**: Total de la población.

- **p**: “Es la proporción de individuos que poseen en la población la característica de estudio. Este dato es generalmente desconocido y se suele suponer que p=q=0.5 que es la opción más segura.” (Technologies, 2013)
- **q**: Cantidad de población que no cumple las características a ser analizadas, caso del proyecto usuarios que se no se unirán a la red Ad Hoc, formula q = (1 – p).
- **k**: Nivel de confiabilidad, este dato ya es previamente establecido,

Tabla 5. Nivel de confiabilidad del muestreo

K	**1,15**	**1,28**	**1,44**	**1,65**	**1,96**	**2**	**2,58**
Nivel de confianza	75%	80%	85%	90%	95%	95,5%	99%

Nota. Fuente: (Feedback Networks Technologies, 2013)

- **e**: Porcentaje de error que tomará la muestra al momento de ser analizada.

Datos

- **K** tendrá el valor 1,65 dando un 90% de confiabilidad de la muestra tomada.
- **p** tendrá el valor de 0,5 que es el valor por defecto.
- **q** tendrá el valor de 0,5 de la solución 1 – p.
- **N** tendrá el valor de 2690 que es el total de la población.
- **e** tendrá el valor de 6 que es el porcentaje de error con el cual se trabajará.

$$n = \frac{1{,}65^2 * 0{,}5 * 0{,}5 * 2690}{\left(6^2 * (2690 - 1)\right) + 1{,}65^2 * 0{,}5 * 0{,}5}$$

n = 177

177 será el total de la muestra a ser analizada para la simulación.

Esto quiere decir que si el proyecto satisface a 177 usuarios también lo hará para 2690.

2.3. Resultado de encuestas

Según los resultados de las encuestas realizadas en el Sector San Martín y en base a la muestra (2.2.4) se obtuvieron los siguientes resultados.

Servicio de Internet

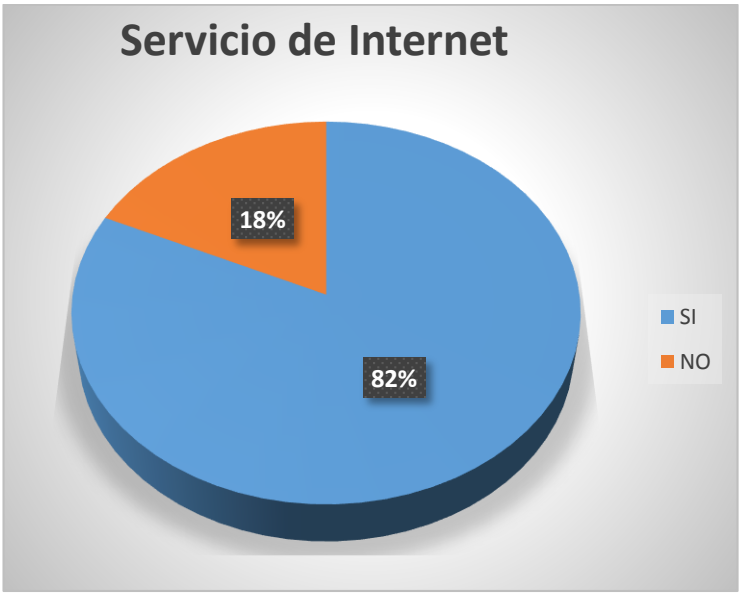

Figura 26. Servicio de Internet
Elaborado por: Jaime Acosta y Jhonatan Quelal

Como se puede observar en la figura 26 la empresa proveedora de Internet cubre el 82% que en este caso solo es la empresa pública CNT, por otra parte, el 18% no posee un servicio de Internet, lo que da como resultado que no existen contrataciones por motivo de costos, por falta de cobertura, o por cancelación del contrato por la baja calidad de servicio.

Grado de satisfacción

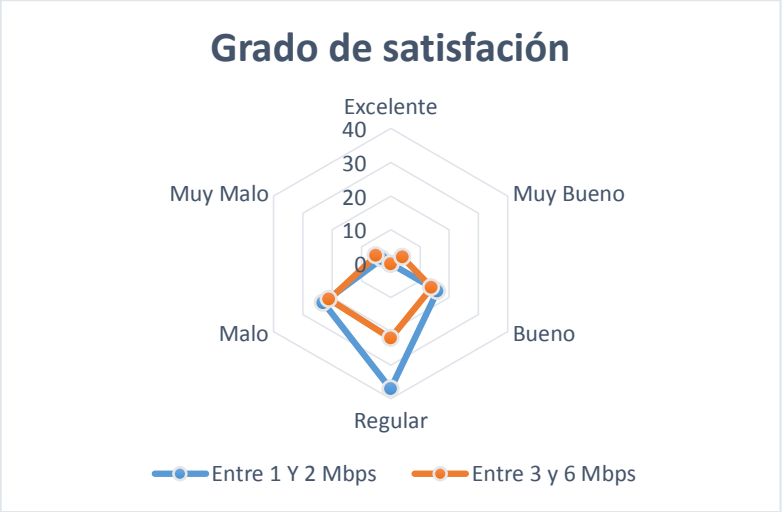

Figura 27. Grado de satisfacción
Elaborado por: Jaime Acosta y Jhonatan Quelal

En la figura 27 se muestra que el 82% de clientes manifiestan que el servicio es regular tanto para los que poseen una velocidad de bajada entre 1 - 2 Mbps y 3 – 6 Mbps. En base a estos resultados se demuestra que el proyecto podría lograr un mejor grado de satisfacción, ya que una empresa que brinde un servicio de Internet debe ofrecer un grado de satisfacción muy bueno a los clientes.

Ver Anexo 2. Encuestas realizadas en el sector de San Martín.

2.4. Cálculo el ancho de banda

Según las encuestas realizadas el ancho de banda promedio contratado es 5 Mbps en el sector San Martín a esta velocidad se la conoce como velocidad nominal ya que es la que el cliente puede observar, pero la verdadera velocidad que se entrega es la velocidad efectiva, es decir, la carga de datos real que consume cada usuario, pero no existe una fórmula para calcular la velocidad efectiva a partir de la velocidad nominal es por eso que depende de la decisión del administrador, para

que cada persona que utiliza el servicio se tendrá que calcular el ancho de banda de la siguiente manera:

Fórmula:

$$AB = G * C$$

Donde:

AB = Ancho de Banda

G = Ancho de banda utilizado por el consumidor para garantizar el servicio. Según las encuestas realizadas la empresa proveedora de Internet que tiene la mayor cobertura en el sector de San Martín tiene un medio de compartición de 8:1, es por eso que en este proyecto para mejorar el rendimiento se toma en cuenta una compartición de 4:1, es decir, 5000 Kbps / 4 esto es igual a 1250 Kbps.

C = Concurrencia de personas que utilicen el Internet (N = cantidad de usuarios), debido a que no todos los usuarios se conectan a la red se toma el 30% en las 24 horas haciendo un uso constante de la red más el 10% de conectividad que hagan uso parcial de la red.

Datos:

N = 2690

G = 1250 Kbps

C = 0.3(N) → 0.3 * (2690) → 807

Desarrollo:

$$AB = 1250\ Kbps * 807$$

$$AB = 1008750\ Kbps$$

$$AB = 1008\ Mbps$$

El resultado obtenido es un estimado cuando el total de usuarios se encuentren utilizando el servicio.

CAPÍTULO 3

SIMULACIÓN

En este capítulo se desarrollará la simulación de eventos discretos para la red ad-hoc con la configuración del protocolo AODV, para obtener resultados del comportamiento de la red en la zona San Martín, para esto se usará la herramienta NS-2.

3.1. NS2

Esta herramienta permitirá simular una red ad-hoc bajo eventos discretos, es decir, realizar un análisis bajo periodos de tiempo.

Para la simulación se necesita crear un script con extensión Tcl, ejemplo <<simulacion.tcl>> y este archivo puede ser compilado en el simulador NS-2, el archivo <<tcl>> contendrá un conjunto de instrucciones donde se crearán diferentes instancias tanto de cantidad de nodos, tipo de protocolo que va a ser empleado, modelo de la antena, tipo de canal entre otros, a estas variables se asignarán datos que fueron obtenidos en la recolección de información del Capítulo 2.

En la ejecución de NS-2 se genera un archivo con extensión .tr o archivo trace, en el cual se almacenan todos los procesos que se ejecutan durante la simulación mostrando así el tiempo de vida, nodo inicio y nodo de llegada del paquete, toda la ejecución se presenta en el NAM (Network Animator) que es una ventana que muestra la animación de lo descrito en el archivo Tcl para su mayor entendimiento, además muestra las trazas de simulación a nivel de paquetes y los tipos de enlaces en cada uno de los nodos.

3.2. OTcl (Object Oriented Tool Command Language)

Este lenguaje utiliza comandos orientados a objetos y es utilizado por NS-2, debido a la simplicidad que da a los usuarios para realizar las simulaciones de redes con sus respectivos protocolos, tanto así que al culminar la simulación se obtienen resultados muy parecidos a la realidad, por lo que ayuda a la toma de decisiones. Los protocolos de NS-2 o librerías están desarrolladas con C++ lo que permite una mejor compatibilidad con otros lenguajes.

El motivo de la utilización de estos dos lenguajes es que en OTcl tiene un rápido acoplamiento con los métodos de C++ ya que con una simple sentencia en OTcl llama a las clases necesarias de C++ para que puedan ser ejecutadas en una sola orden en este caso si el cambio de configuración se genera en OTcl manipula todos los objetos necesarios de C++ para realizar dicha modificación. En cambio, en C++ se puede moldear nuevas formas de funcionamiento como proceso de cada paquete, comportamiento del nodo origen y nodo destino, es decir, la manipulación directa de un protocolo para tener una mejor eficiencia del mismo.

Los parámetros que acepta OTcl para poder crear las redes son:

Tabla 6. Parámetros de OTcl

OPCIÓN	DESCRIPCIÓN	VALORES DISPONIBLES	POR DEFECTO
GENERAL			
addressType	Tipo de Dirección	flat, hierarchical	flat
MPLS	Protocolo de Comunicación	ON, OFF	OFF
Ifqlen	Número máximos de paquetes en cola	50 paquetes, máximo 64	
Nn	Número de nodos móviles	<cantidad de nodos>	
X	Dimensión en eje X	<valor en metros>	
Y	Dimensión en eje Y	<valor en metros>	

OPCIÓN	DESCRIPCIÓN	VALORES DISPONIBLES	POR DEFECTO
Stop	Tiempo de duración de la simulación en segundos	<valor en milisegundos>	
INALÁMBRICA Y SATÉLITE			
wiredRouting	Ruta cableada	ON, OFF	OFF
llType	Tipo de capa de enlace	LL, LL/Sat	""
macType	Tipo de MAC	Mac/802_11, Mac/Csma/Ca, Mac/Sat, Mac/Sat/UnslottedAloha, Mac/Tdma	
ifqType	Tipo de cola	Queue/DropTail, Queue/DropTail/PriQueue	""
phyType	Tipo de interfaz de red	Phy/WirelessPhy, Phy/Sat	""
INALÁMBRICA			
adhocRouting	Protocolo de enrutamiento en este caso es AODV	DIFFUSION/RATE, DIFFUSION/PROB, DSDV, DSR, FLOODING, OMNIMCAST, AODV, TORA, M-DART, PUMA	""
propType	Modelo de propagación de radio	Propagation/TwoRayGround, Propagation/Shadowing	""
propInstance	Instancia de Propagación	Propagation/TwoRayGround, Propagation/Shadowing	""
antType	Modelo de antena	Antenna/OmniAntenna	""
Cannel	Canal	Channel/WirelessChannel, Channel/Sat	""
topoInstance	Instancia de Topología	<topology file>	""
mobileIP	IP móvil	ON, OFF	OFF
energyModel	Modelo de Energía	EnergyModel	""
initialEnergy	Energía Inicial	<valor en Julios>	""
rxPower	Potencia de Recepción	<valor en Vatios>	""
txPower	Potencia de Transmisión	<valor en Vatios>	""
idlePower	Energía en reposo	<valor en Vatios>	""
agentTrace	Traza de agente	ON, OFF	OFF
routerTrace	Traza de ruta	ON, OFF	OFF
macTrace	Traza de MAC	ON, OFF	OFF
movementTrace	Traza de movimiento	ON, OFF	OFF

OPCIÓN	DESCRIPCIÓN	VALORES DISPONIBLES	POR DEFECTO
errProc	Error de Protocolo	UniformErrorProc	""
toraDebug	Paquete de Información de TORA	ON, OFF	OFF
SATÉLITE			
satNodeType	Tipo de nodo satelital	polar, geo, terminal, geo-repeater	""
downlinkBW	Ancho de Banda de bajada	<bandwidth value, e.g. "2Mb">	""

Nota. Fuente: (VINT, 2000)

3.3. Programación

Para realizar el script de simulación, el código base fue tomado del manual de NS2, el manual cuenta con todos los distintos escenarios para realizar simulaciones de los diferentes protocolos existentes y así poder conformar una red con las configuraciones necesarias y dependiendo de los cambios que se necesita ejercer en el ejemplo. A continuación, se detallarán los parámetros a usarse para el desarrollo del proyecto.

3.3.1. Velocidad de la red

En la primera parte del código se configura el rango de velocidad que tendrá la red que fue calculada en la sección 2.4.

<< Mac/802_11 set dataRate_ 1008Mb >>

3.3.2. Variables globales

Estas variables globales son declaradas al inicio del script con el fin de que puedan ser usadas en cualquier parte del código facilitando la programación.

Variables globales

```
set val(chan)           Channel/WirelessChannel
set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(ifqlen)         64
set val(nn)             92
set val(rp)             AODV
set val(x)              742
set val(y)              742
set val(stop)           120
```

Figura 28. Variables globales
Elaborado por: Jaime Acosta y Jhonatan Quelal

El comando <<set>> es usado para la definición de las instancias.

El comando <<val( )>> es un arreglo de valores definidos en OTcl usadas para definir constantes.

- **CHAN**: Tipo de canal que va ser usado en la red el cual en este caso será inalámbrico.

  <<Channel/WirelessChannel>>

- **PROP**: Modelo de radio propagación. En este caso va ser usado el modelo de dos rayos de reflexión en el suelo, para tener una simulación más real, este modelo toma en consideración la tierra contra las ondas de radio que emite cada dispositivo.

  <<Propagation/TwoRayGround>>

- **NETIF**: Tipo de interfaz de red que utilizará el dispositivo, en este caso se hará uso de una interfaz física de red.

  <<network interface type>>

- **MAC:** Cabecera de enlace de los dispositivos. En este caso se utilizará la norma IEEE 802.11.

  <<Mac/802_11>>

- **IFQ**: Tipo de interfaz de cola. En este caso se usará el que da prioridad a los paquetes de enrutamiento del protocolo, es decir, si existen paquetes de información y al mismo tiempo paquetes de conectividad en la cola, este da prioridad a los paquetes de conectividad del protocolo. El DropTail define el orden FIFO (First In First Out).

  <<Queue/DropTail/PriQueue>>

- **LL**: Tipo de capa de enlace. En este caso se utilizará LL para la resolución de direcciones IP de origen, destino y siguiente salto.

<<LL>>

- **ANT**: El modelo de antena. En este caso se hará uso de una antena omnidireccional, que emite la señal en 360°.

<<Antenna/OmniAntenna>>

- **IFQLEN**: La cantidad en cola que se podrá albergar en cada dispositivo. En este caso el valor por defecto es de 50 paquetes.

<<50>>

- **NN:** Cantidad de nodos. En el caso de NS2 no abastece más de 100 nodos, pero en el resultado obtenido de la muestra son 177 personas, por lo que se procedió a sacar una nueva muestra partiendo como dato inicial el valor obtenido en la muestra, dando como resultado 92 nodos, ya que por regla estadística si los resultados obtenidos en la muestra son satisfactorios de la misma manera será para el total de la población.

<<92>>

- **RP:** Se define el protocolo que se hará uso.

<<AODV>>

En los siguientes literales se define el área que es 0.55 Km^2 (Ver sección 2.2.1.).

- **X:** En esta variable se define la dimensión de la topología en el eje x. En este caso es de 742 metros.

<<742>>

- **Y:** En esta variable se define la dimensión de la topología en el eje y. En este caso es de 742 metros.

<<742>>

- **STOP:** Tiempo de la simulación. En este caso 120 segundos.

<<120>>

3.3.3. Inicialización de la simulación, seguimiento de traza y archivo de animación.

Inicialización de procesos, traza y animación

```
set ns          [new Simulator]
set trazaAODV       [open t_AODV.tr w]
set archivoNAM      [open a_NAM.nam w]


$ns trace-all $trazaAODV
$ns namtrace-all-wireless $archivoNAM $val(x) $val(y)
```

Figura 29. Inicialización de procesos, traza y animación
Elaborado por: Jaime Acosta y Jhonatan Quelal

- **NS**: Variable utilizada para la inicialización del simulador.
- **trazaAODV**: Creación de un archivo t_AODV.tr en el que se va a grabar (write w) todos los caminos que los paquetes sigan y lo que suceda durante la iteración en el proceso.
- **archivoNAM**: Creación de un archivo a_NAM.nam en el que se ve a grabar (write w) los movimientos de cada nodo y la transmisión de los paquetes entre dispositivos, el cual va a ser utilizada para ser compilado y así poder ver la simulación.
- **Trace-all**: Esta función es ejecutada para poder dar seguimiento a los paquetes y así poder grabar todo lo que suceda en la red en el archivo correspondiente.

- **Namtrace-all-wireless:** Esta función es utilizada para indicar que la simulación es de un modelo inalámbrico, además indica el tamaño de la topología tanto para << x >> y << y >>.

3.3.4. Configuración de la topología.

Topología

```
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
```

Figura 30. Topología
Elaborado por: Jaime Acosta y Jhonatan Quelal

- **Topo:** Utilizada para crear el área en el cual se va a crear la topología.
- **Load_flatgrid:** Esta permite inicializar una topografía tipo cuadricula el cual cada cuadro tendrá un valor por defecto de 1, es decir, que las variables de <<x>> y <<y>>, se establece como un plano cartesiano.
- **Create-god:** Este comando permite almacenar la información de las posiciones de los nodos móviles en una tabla para que así la ejecución evitar buscar por cada nodo la acción que deba seguir.

3.3.5. Configuración de los nodos.

Configuración de nodos

```
$ns node-config -adhocRouting $val(rp) \
     -llType $val(ll) \
     -macType $val(mac) \
     -ifqType $val(ifq) \
     -ifqLen $val(ifqlen) \
     -antType $val(ant) \
     -propType $val(prop) \
     -phyType $val(netif) \
     -channelType $val(chan) \
     -topoInstance $topo \
     -agentTrace ON \
     -routerTrace ON \
     -macTrace OFF \
     -movementTrace ON
```

Figura 31. Configuración de nodos
Elaborado por: Jaime Acosta y Jhonatan Quelal

En la configuración del nodo se toma las variables globales (Ver sección 3.3.2.) para poder configurar a cada uno de los 92 nodos.

- **Adhoc routing**: Configuración del protocolo que va a ser usado en este caso AODV.
- **AgentTrace**: Indicador de la información de los agentes que permiten establecer conectividad entre los nodos.
- **routerTrace**: Información de enrutamiento del protocolo.
- **macTrace**: Información acerca de la capa MAC.
- **moventTrace**: Información del movimiento que se establece en los nodos.

3.3.6. Configuración de movimiento y posición de los nodos.

Configuración de movimiento y posición de los nodos

```
        for {set i 0} {$i < $val(nn) } { incr i } {

            set node_($i) [$ns node]
            $node_($i) set X_ [ expr round(rand()*100)+round(rand()*500) ]
            $node_($i) set Y_ [ expr round(rand()*200)+round(rand()*500) ]
            $node_($i) set Z_ 0.0

    }



        for {set i 0} {$i < $val(nn) } { incr i } {
            $ns at [ expr round(rand()*60)+round(rand()*60) ] "$node_($i) setdest
    [ expr round(rand()*71)+round(rand()*300) ] [ expr round(rand()*300)+round(rand
    ()*71) ] [ expr round(rand()*5)+round(rand()*2) ]"

        }
```

Figura 32. Configuración de movimiento y posición de los nodos
Elaborado por: Jaime Acosta y Jhonatan Quelal

Las sentencias <<for>> están establecidas para que recorran los 92 nodos que están representados por la variable <<$val (nn)>>. En el primer <<for>> establece la posición inicial de cada nodo tomando en cuenta los valores determinados tanto para las variables <<x>> y <<y>> (los valores que adoptarán <<x>> y <<y>> no podrán exceder del valor del área de San Martín).

El programa está diseñado para que en cada ejecución los nodos se ubiquen de manera aleatoria en cualquier parte del área.

En el segundo << for >> está diseñado de igual manera que el anterior caso que recorra los 92 nodos pero aquí se designa los movimientos que tendrán cada uno de ellos, en la primera expresión << expr round(rand()*60)+round(rand()*60) >> se define un método random para el periodo de tiempo, lo que dará como resultado que cada uno de los nodos se moverán en diferentes tiempos.

En la expresión << "$node_($i) setdest [expr round(rand()*71)+round(rand()*300)] [expr round(rand()*300)+round(rand()*71)] >> hace uso del método << setdest >> que ayuda a la movilidad de los nodos, dirigiendo a cada uno de los nodos desde una posición origen hacia una posición destino, como se puede observar en la expresión, aquí también se hace uso del método random para que las posiciones destino sean aleatorias.

En la última expresión << [expr round(rand()*5)+round(rand()*2)]" >> se define de manera aleatoria la velocidad que alcanzara cada nodo.

En conclusión, la expresión se define de la siguiente manera:

$ns at [**tiempo en el que se va a mover el nodo**] "$node_ ($i) setdest [Destino en la posición X**] [**Destino en la posición Y**] [**Velocidad del nodo en Km/h**]"**

Cabe recalcar que el área que se tiene en la simulación es muy amplia para los 92 nodos, lo que implica que va a existir distancias muy alejadas de un nodo a otro, esto provocará que el protocolo se someta a buscar rutas alternas para poder llegar a su destino, lo que conlleva a que la red se encuentra simulada en las peores situaciones, en lo que se refiere velocidad del individuo, topología en la cual interfiere el ruido y factores ambientales como la reflexión.

En las siguientes configuraciones se hará uso del agente Newreno, ya que modifica la acción de recibimiento del ACK lo que permite que, si un paquete se pierde, tome acción creando así una rápida recuperación del paquete tratándolo como prioridad, muchas veces utiliza el total de la longitud de la ventana. (Henderson, 2011)

Como se conoce la red puede transmitir dos tipos de tráfico unos orientados a conexión y otros no orientados a conexión, es decir, que la red puede transmitir cualquier tipo de información como es http, https, pop3, ftp, smtp, voz, datos, video entre otros. En las siguientes configuraciones de tráfico se mostrará de manera general el flujo de datos ya que, dando una carga constante a la red, esta se enfrenta a mantener paquetes en frecuente movimiento tanto de envío como de recepción de información. Si se enfrenta a la red a una carga en la cual distintos nodos estén conectados y enviando en todo el lapso de ejecución se demostrará que la red puede solventar cualquier problema en cuanto a densidad de información.

Los paquetes de ruteo que el protocolo AODV necesita para la conectividad se transmiten bajo UDP, es por eso que si el nodo origen puede conectarse al destino mediante el protocolo AODV se deduce que la red puede satisfacer las necesidades del tráfico no orientado a conexión. Si el protocolo UDP no fuera factible inmediatamente la comunicación entre dispositivos fallaría.

Como resultado se ha optado por simular la red mediante el protocolo TCP con el fin de ver el comportamiento de estos paquetes.

3.3.7. Configuración tráfico exponencial.

El tráfico exponencial se basa en el modelo de distribución exponencial, la cual estudia la fiabilidad de intervalos de tiempo antes de que ocurra algún fallo inesperado, es decir, estudia la tasa de fallo mientras transcurre un tiempo definido. Entonces el tráfico exponencial genera cada cierto intervalo de tiempo una tasa

constante de paquetes y en la otra parte genera un decaimiento o fallo para así poder observar como la red actúa en estas situaciones.

Configuración tráfico exponencial

```
set tcp0 [new Agent/TCP/Newreno]
        $tcp0 set class_ 2
        set sink0 [new Agent/TCPSink]
        $ns attach-agent $node_(0) $tcp0
        $ns attach-agent $node_(1) $sink0
        $ns connect $tcp0 $sink0

set exp0 [new Application/Traffic/Exponential]
        $exp0 attach-agent $tcp0
        $exp0 set packetSize_ 210
        $exp0 set burst_time_ 500ms
        $exp0 set idle_time_ 500ms
        $exp0 set rate_ 5mb


$ns at 0.1 "$exp0 start"
```

Figura 33. Configuración de tráfico exponencial
Elaborado por: Jaime Acosta y Jhonatan Quelal

Para realizar la conexión de un nodo origen a un nodo destino se hace uso de una variable tcp0 el cual contiene el agente TCP. Este agente se agrega en este caso al nodo 0, el cual va a enviar los paquetes es decir, se lo designa como nodo origen.

En la variable sink0 contiene el agente TCPSink el cual permite designar el nodo destino.

En la sentencia connect se realiza la conexión entre la variable $tcp0 y $sink0, es decir, el nodo origen con el nodo destino.

Para realizar la configuración del tráfico exponencial se establece la variable exp0 la cual contiene el tipo de tráfico a utilizarse y al mismo tiempo realiza una conexión con la variable tcp0, es decir, que el tráfico que se generará entre el nodo 0 y el nodo 1 será exponencial.

3.3.8. Configuración tráfico pareto.

La distribución de Pareto se basa en enviar ciertas cantidades de información con respecto al tiempo, es decir, si por ejemplo en una iteración de tiempo se envía un 100% de información, en la siguiente podría enviar un 20% y así sucesivamente.

Configuración tráfico pareto

```
set tcp11 [new Agent/TCP/Newreno]
        $tcp11 set class_ 2
        set sink11 [new Agent/TCPSink]
        $ns attach-agent $node_(20) $tcp11
        $ns attach-agent $node_(21) $sink11
        $ns connect $tcp11 $sink11

set p11 [new Application/Traffic/Pareto]
        $p11 attach-agent $tcp11
        $p11 set packetSize_ 210
        $p11 set burst_time_ 500ms
        $p11 set idle_time_ 500ms
        $p11 set rate_ 5mb
        $p11 set shape_ 1.5

$ns at 0.1 "$p11 start"
```

Figura 34. Configuración tráfico pareto
Elaborado por: Jaime Acosta y Jhonatan Quelal

3.3.9. Configuración tráfico CBR.

En el tráfico CBR (Constant Bit Rate) se basa en enviar una cantidad constante de información por determinados intervalos de tiempo. La función random_ 1 permite establecer estos intervalos de tiempo aleatoriamente.

Configuración tráfico cbr

```
set tcp31 [new Agent/TCP/Newreno]
        $tcp31 set class_ 31
        set sink31 [new Agent/TCPSink]
        $ns attach-agent $node_(60) $tcp31
        $ns attach-agent $node_(61) $sink31
        $ns connect $tcp31 $sink31

set e31 [new Application/Traffic/CBR]
        $e31 attach-agent $tcp31
        $e31 set packetSize_ 210
        $e31 set rate_ 5mb
        $e31 set random_ 1
$ns at 0.1 "$e31 start"
```

Figura 35. Configuración de tráfico CBR
Elaborado por: Jaime Acosta y Jhonatan Quelal

3.3.10. Configuración FTP.

La implementación FTP simulará la transferencia masiva de datos a diferencia de los demás transmitirá paquetes que simularan la descarga de un paquete del nodo origen como contenedor y nodo destino como creador de petición. La cantidad de paquetes se generará en la clase Application/FTP que tomará en cuenta las características de la simulación y luego decidirá la cantidad de información a enviar.

Configuración tráfico cbr

```
set tcp21 [new Agent/TCP]
$ns attach-agent $node_(40) $tcp21
set sink21 [new Agent/TCPSink]
$ns attach-agent $node_(41) $sink21
$ns connect $tcp21 $sink21

set ftp21 [new Application/FTP]
$ftp21 attach-agent $tcp21
$ftp21 set type_ FTP

$ns at 0.1 "$ftp21 start"
```

Figura 36. Configuración de tráfico FTP
Elaborado por: Jaime Acosta y Jhonatan Quelal

3.3.11. Definir el tamaño de los nodos NAM.

Tamaño de nodos

```
for {set i 0} {$i < $val(nn)} { incr i } {
$ns initial_node_pos $node_($i) 30
}
```

Figura 37. Tamaño de nodos
Elaborado por: Jaime Acosta y Jhonatan Quelal

La sentencia <<for>> que se puede observar en la figura 37 realiza un bucle del total de los nodos que en este caso son 92, con el fin de asignar el tamaño de cada nodo a través del método <<initial_node_pos>>.

3.3.12. Resetea los nodos cuando finaliza la simulación.

Reseteo de nodos

```
for {set i 0} {$i < $val(nn) } { incr i } {
    $ns at $val(stop) "$node_($i) reset";
}
```

Figura 38. Reseteo de nodos
Elaborado por: Jaime Acosta y Jhonatan Quelal

En esta sentencia <<for>> se recorre todos los nodos existentes que en este caso son 92, para resetear todos los dispositivos, borrando toda la información que existe y dando como finalizada la simulación.

3.3.13. Finalizando el NAM y la simulación.

Finalización de la simulación

```
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "stop"
$ns at 120 "puts \"end simulation\" ; $ns halt"
```

Figura 39. Finalización de la simulación
Elaborado por: Jaime Acosta y Jhonatan Quelal

Al igual que los nodos deben tener un reinicio para borrado de información de igual manera todos los procesos necesitan un método de cierre.

<<$ns at $val(stop) "$ns nam-end-wireless $val(stop)">> Sentencia utilizada para la finalización de la animación.

<< $ns at $val(stop) "stop" >> Sentencia utilizada para llamar el procedimiento <<proc stop>>.

<< $ns at 120 "puts \"end simulation\"; $ns halt" >> Sentencia utilizada para la finalización en cuanto a programación.

Finalización de los archivos

```
proc stop {} {
    global ns trazaAODV archivoNAM
    $ns flush-trace
    close $trazaAODV
    close $archivoNAM
}

$ns run
```

Figura 40. Finalización de los archivos
Elaborado por: Jaime Acosta y Jhonatan Quelal

El procedimiento << proc stop >> es utilizado para el cierre de los archivos .tr y .nam en este caso son los archivos $trazaAODV y $archivoNAM respectivamente. *Ver Anexo 3. Código fuente de la simulación adjunto en el disco magnético.*

3.4. Gráfica de animación de la red

El network Animator (NAM) sirve para poder realizar la lectura del archivo a_NAM.nam, el cual contiene información acerca del movimiento, trazas o rutas que permiten la conectividad de los nodos emisores como receptores, la transmisión de información que se haya establecido mediante el protocolo a través de las rutas. Mostrando así la información de paquetes enviados, recibidos, perdidos o desechados.

Gráfica de posicionamiento de nodos

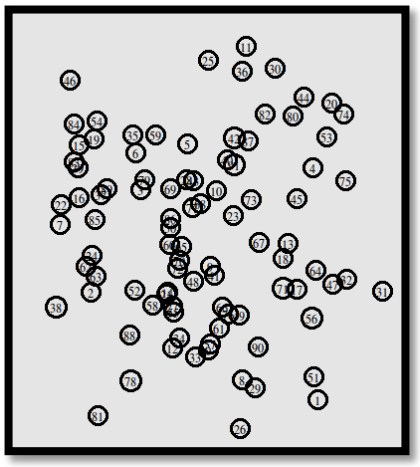

Figura 41. Gráfica de posicionamiento de nodos
Elaborado por: Jaime Acosta y Jhonatan Quelal

El programa NAM ayuda a visualizar la animación en diferentes velocidades para una mejor observación en el lapso determinado. Se puede observar la animación a la velocidad de 1 µs hasta 794.3 ms.

Gráfica de animación de la red

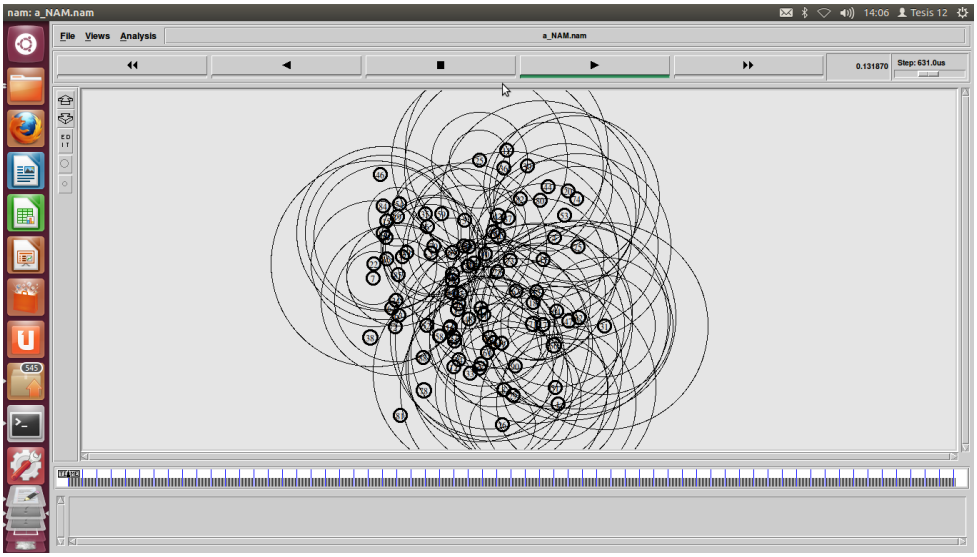

Figura 42. Gráfica de animación de la red
Elaborado por: Jaime Acosta y Jhonatan Quelal

CAPÍTULO 4

ANÁLISIS DE RESULTADOS

En este capítulo se hará uso de los programas NS-2 Wireless Trace Analyzer para determinar el comportamiento de la red simulada, también se definirán los términos usados por NS-2 los cuales son mostrados en los resultados, no se analizará a detalle el archivo t_AODV.tr ya que el programa NS-2 Wireless Trace Analyser versión 0.2.72 realiza la lectura de todo el archivo y muestra directamente los resultados.

4.1. NS-2 Wireless Trace Analyser

NS2 no cuenta con una interfaz para interpretar los datos que se generan, es por eso que se hace uso de un programa que permita la lectura de estos datos. Este programa utiliza los archivos .tr y .tcl los mismos que son generados del programa de simulación NS-2.

Al tener cargado los archivos en el programa analizador de trazas toma automáticamente la estructura de la red del archivo con extensión .tcl, es decir, cantidad de nodos, área en el cual se efectuó el análisis, posiciones iniciales de los nodos, tipos de tráfico que van a ser usados y tiempo de duración de la simulación.

Con respecto al archivo t_AODV.tr al momento de ser cargado en el programa analizador toma los datos de los dispositivos durante el tiempo de ejecución, es decir, carga flujos, movimientos, conexiones y transmisión de información entre los nodos.

NS2 Visual Trace Analyzer

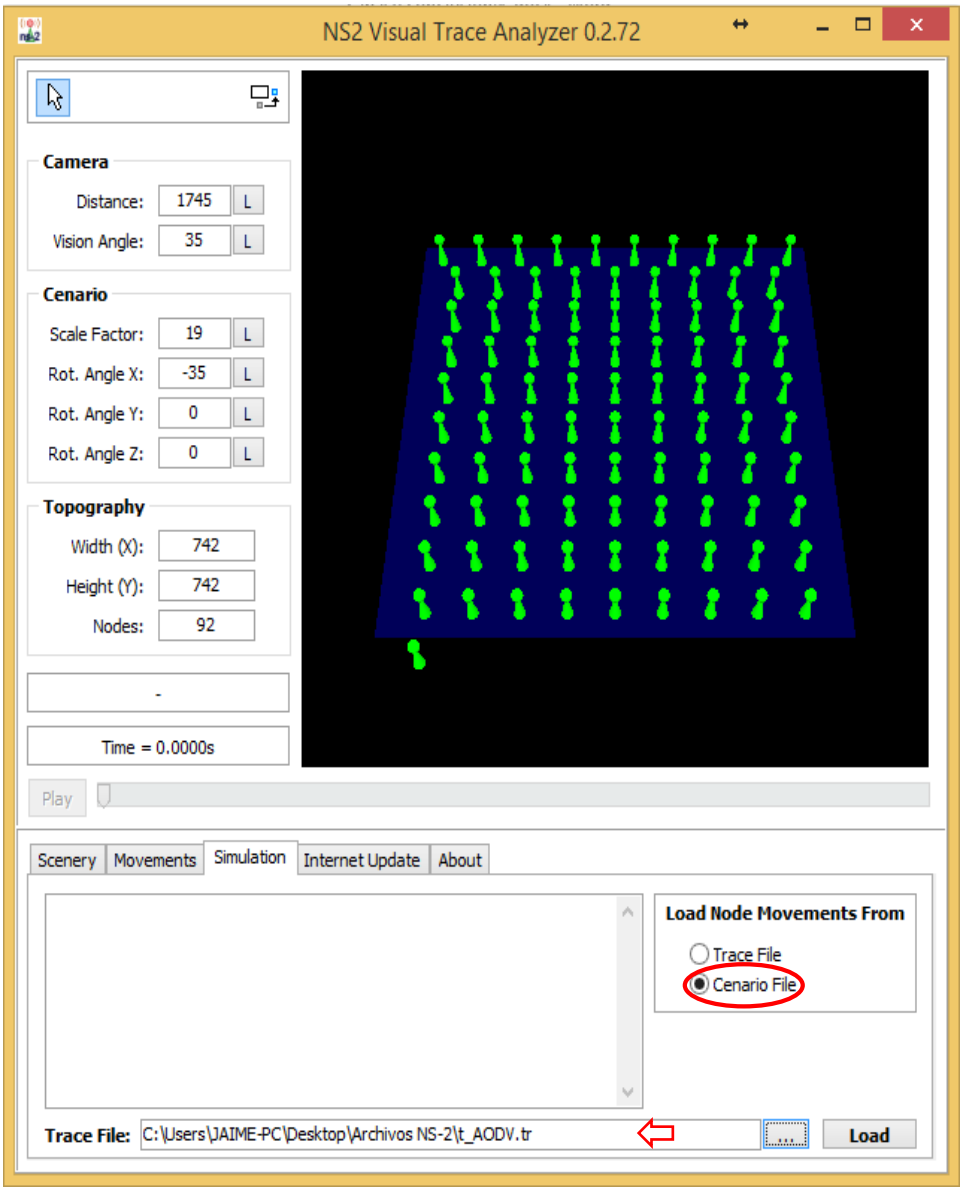

Figura 43. NS2 Visual Trace Analyzer
Elaborado por: Jaime Acosta y Jhonatan Quelal

El programa se encarga de analizar diferentes parámetros que se establecen durante el recorrido del programa en este caso se explicarán los diferentes estados que se atribuyen en el archivo t_AODV.tr

Cabecera AODV

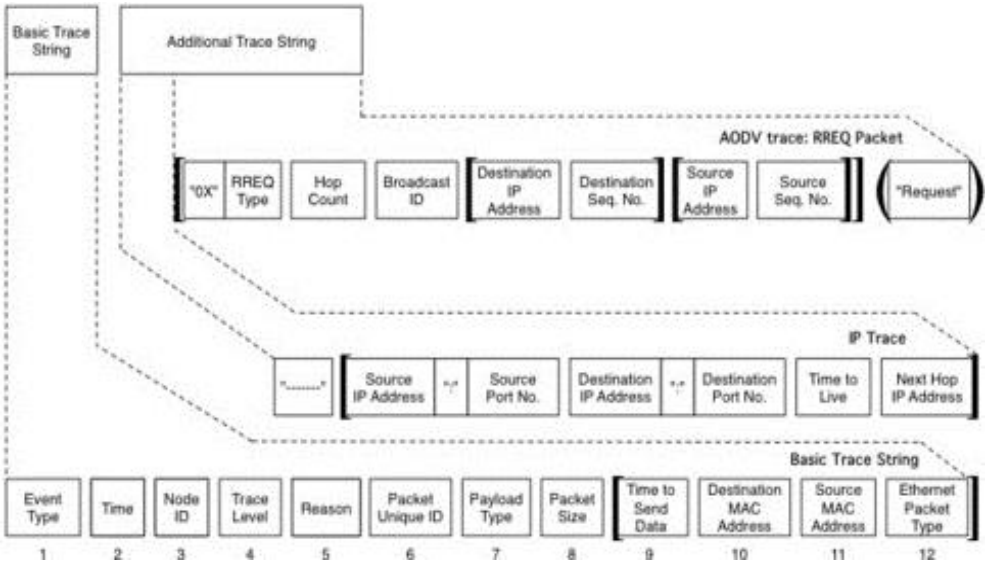

Figura 44. Cabecera AODV
Fuente: (Issariyakul, 2009)

Tabla 7. Detalle de los campos de la cabecera AODV

TIPO DE ACCIÓN	
S:	Send (Enviado)
R:	Receive (Recibido)
D:	Drop (Eliminado)
F:	Forward (Renvío)
M:	Movement (Movimiento)
TIEMPO	
t:	Tiempo
NODO ORIGEN	
Número del identificador del nodo origen	
NIVEL DE TRAZA	
AGT	Agente de enlace
RTR	Routing
IFQ	Cola de prioridad de interfaz
RAZÓN	
END	Disminución final de la simulación
NRTE	Ningún router disponible
TTL	Alcance máximo de tiempo de vida
CBK	Descarte de devolución de petición
ARP	Descarte de resolución de direcciones
IDENTIFICADOR DE PAQUETE	
Número único que va a identificar al paquete que se está enviando.	

TIPO DE CARGA	
TCP	Flujo TCP
AODV	Flujo AODV
EXP	Flujo EXPONENCIAL
|PARETO	Flujo PARETO
TAMAÑO DE PAQUETE	
Número que representa el tamaño del paquete	
TIEMPO ESTIMADO	
Número hexadecimal que representa el tiempo estimado que el paquete demorara en llegar a su destino	
DIRECCIÓN MAC DE DESTINO	
Número hexadecimal que representa el identificador MAC del nodo de envío.	
DIRECCIÓN MAC DE ORIGEN	
Número hexadecimal que representa la dirección MAC de origen.	
TIPO DE PAQUETE ETHERNET	
Número que representa la dirección IP del paquete que está corriendo sobre la red ethernet.	
DIRECCIÓN IP ORIGEN	
Número que representa la dirección IP origen del paquete.	
NÚMERO DE PUERTO ORIGEN	
Número que representa el puerto origen que se va a utilizar.	
DIRECCIÓN IP DESTINO	
Número que representa la dirección IP destino del paquete.	
NÚMERO DE PUERTO DESTINO	
Número que representa el puerto destino que se va a utilizar.	
TIEMPO DE VIDA	
Número de saltos que puede realizar el paquete.	
DIRECCIÓN IP DEL SIGUIENTE SALTO	
Número que identifica al nodo el cual es el siguiente salto.	
"Ox" – TIPO DE RREQ	
Número que representa la información de traza del protocolo AODV.	
NÚMERO DE SALTOS	
Conteo del número de saltos.	
IDENTIFICADOR DE BROADCAST	
Número identificador de broadcast a usarse.	
DIRECCIÓN IP DESTINO	
Número que representa la dirección IP destino del paquete.	
NÚMERO DE SECUENCIA DEL DESTINO	
Número de secuencia del destino.	
DIRECCIÓN IP ORIGEN	
Número que representa el puerto origen que se va a utilizar.	
NÚMERO DE SECUENCIA ORIGEN	
Número de secuencia del origen.	
(REQUEST)	
Identificación de confirmación del paquete.	

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

NS-2 crea sus propios parámetros como el establecimiento de conexión, envío de información, búsqueda de nuevas rutas, datos de nodo origen y destino, requerimientos y respuestas, esto no quiere decir que NS-2 no funcione de la misma manera que el protocolo AODV, simplemente crea una cabecera con la cual pueda trabajar para efectuar datos reales para el proceso de información, ya que, si realizara el proceso completo de creación de cabecera, segmentación de paquetes, configuración de IP por nodo, codificación y encriptación al momento de ejecutar el TCL tendría que durar un tiempo considerable para mostrar el total de la función de la red, aun durando este tiempo no podría mostrar los datos necesarios en una situación real. NS-2 no se enfoca en la función sino en la acción de los paquetes utilizando solamente lo necesario, que es programado en los métodos de C++ para poder presentar una información de que sucedió durante el envío, transmisión y recepción de la información.

Las variables que se utilizan en NS-2 y que se muestran a continuación fueron obtenidos del programa de simulación NS-2 específicamente del archivo t_AODV.tr el cual se explican los diferentes procesos que se llevan a cabo durante la ejecución. Para una mayor comprensión y análisis de resultados se deberán entender los campos explicados anteriormente ya que estos rigen el proceso de transmisión que realiza cada uno de los paquetes, es decir, el comportamiento de la información a través de los enlaces por cada intervalo de tiempo.

EJEMPLOS

Tabla 8. Ejemplo 1 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14		15	16		17	18	19	20	21	22	23
s	0.100000000	_40_	AGT	---	0	tcp	40	[0	0	0	0]	-------	[40	:	0	41	:	0	32	0]	[0	0]	0	0

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
1.	S	Indica que el paquete va a ser enviado.
2.	0.100000000	Tiempo en el que comienza el envío.
3.	40	Indica el nodo origen en este caso el nodo <<40>>.
4.	AGT	Genera un proceso de enganche para que el nodo pueda enviar la información a su destino.
6.	0	Identificador del paquete en este caso es <<0>> por que recién iniciando la conexión.
7.	TCP	Tipo de información que se va a enviar en este caso es TCP, es decir, orientada a conexión.
8.	40	Representa el tamaño del paquete que se va a enviar en este caso es 40.
9.	0	En este caso el tiempo estimado de demora es <<0>> ya que para el envío de la información no supera el 1 segundo, es decir, la demora son milisegundos, (De base 10 a base 16 no se puede expresar cantidades flotantes).
10.	0	Este <<0>> representa la dirección MAC destino.
11.	0	Este <<0>> representa la dirección MAC origen.
12.	0	Este <<0>> representa el tipo de MAC.
14.	40	El número <<40>> representa la IP del nodo origen.
15.	0	El número <<0>> representa el puerto origen.
16.	41	El número<<0>> representa la IP del nodo destino.
17.	0	El número <<0>> representa el puerto destino.
18.	32	Representa el campo TTL en la cabecera IP.

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 9. Ejemplo 2 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
r	0.100000000	_40_	RTR	---	0	tcp	40	[0	0	0	0]	-------	[40	: 0	41	: 0	32	0]	[0	0]	0	0

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
1.	R	Indica que el paquete va a ser recibido.
4.	RTR	Mensaje de protocolo.

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 10. Ejemplo 3 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	26
s	0.100000000	_40_	RTR	---	0	AODV	48	[0	0	0	0]	-----	[40	: 255	-1	: 255	30	0]	[0x2	1	1	[41	0]	[40	4]]	(REQUEST)

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
16.	-1	La dirección IP destino es <<-1>>.
20.	0x2	Identificador de mensaje de requerimiento <<RREQ>>.
21.	1	El contador de saltos es <<1>>.
22.	1	El ID de difusión es <<1>>.
23.	41	Dirección IP destino es <<41>>.
24.	0	El número de secuencia es <<0>>.
25.	40	Dirección IP origen es <<40>>.
26.	4	El número de secuencia es <<4>>.
27.	(REQUEST)	Esta cadena de caracteres <<(REQUEST)>> confirma que es un paquete RREQ

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 11. Ejemplo 4 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
r	0.100948095	_21_	RTR	---	0	AODV	48	[0	ffffffff	3a	800]	------	[58	: 255	-1	: 255	30	0]	[0x2	1	1	[59	0]	[58	4]]	(REQUEST)

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
7.	AODV	Tipo de información que se va a enviar en este caso es AODV, información de protocolo para establecimiento de conexión.
10.	ffffffff	Este << ffffffff >> representa la dirección MAC destino.
11.	3a	Este <<3a >> representa la dirección MAC origen

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 12. Ejemplo 5 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
s	0.103290295	_45_	RTR	---	0	AODV	44	[0	0	0	0]	-------	[45	: 255	44	: 255	30	44]	[0x4	1	[45	4]	10.000000]	(REPLY)

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
19.	0x4	Identificador de Mensaje de respuesta <<REPLY>>.
20.	1	El ID de difusión es <<1>>.
21.	45	Indica el nodo origen en este caso el nodo <<45>>.
22.	4	El número de secuencia es <<4>>.
24.	(REPLY)	Esta cadena de caracteres <<(REPLY)>> confirma que es un paquete RREP

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 13. Ejemplo 6 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
R	7.372789163	_64_	AGT	---	1395	ack	40	[13a	40	11	800]	-------	[65 : 0	64 : 0	28	64]	[68	0]	3	0		

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
7.	ack	Mensaje de acuse de recibo
20.	68	Numero de secuencia
21.	0	Numero de ack
22.	3	Numero de retransmisión de este paquete
23.	0	Cantidad de paquetes del mismo tipo que van delante del paquete procesado

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 14. Ejemplo 7 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
D	9.986419683	_41_	RTR	NRTE	0	AODV	44	[13a	29	49	800]	-------	[73 : 255	72 : 255	29	41]	[0x4	1	[73	6]	10.000000]	(REPLY)		

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
1.	D	Indica que este paquete fue desechado
4.	NRTE	Indica que no existe una ruta disponible

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 15. Ejemplo 8 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14		15	16		17	18	19	20	21	22	23
f	9.984338320	_17_	RTR	---	2577	tcp	1040	[13a	11	40	800]	-------	[64	:	0	65	:	0	29	49]	[113	0]	1	0

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
1.	f	Indica que el paquete fue trasmitido

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 16. Ejemplo 9 resultado de la traza

1	2	3	4	5	6	7	8	9	10	11	12	13	14		15	16		17	18	19	20	21	22	23
D	120.000000000	_76_	IFQ	END	48495	tcp	1040	[0	4d	4c	800]	-------	[76	:	0	77	:	0	30	77]	[839	0]	0	0

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
4.	IFQ	Indica que el espacio de memoria donde se encola los paquetes está lleno.
5.	END	Indica que al final de simulación descartara todos los paquetes que estén en la ruta, ya que están al final de la simulación.

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 17. Ejemplo 10 resultado de la traza

1	2	3	4	5	6	7	8	9
M	14.00000	14	(211.00,	233.00,	0.00),	(331.00,	216.00),	2.00

NÚMERO DE CAMPO	PARÁMETRO	DESCRIPCIÓN
1.	M	Indica el movimiento del nodo.
2.	14.00000	Indica el tiempo en el que el nodo se moverá.
3.	14	Número de nodo.
4.	211.00	Indica la posición origen en el eje x.
5.	233.00	Indica la posición origen en el eje y.
6.	0.00	Indica la posición origen en el eje z.
7.	331.00	Indica la posición destino en el eje x.
8.	216.00	Indica la posición destino en el eje y.
9.	2.00	Indica la velocidad del nodo

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

4.2. Conceptos utilizados para el análisis de la información

Se tienen que establecer conceptos importantes para el análisis que se va a efectuar porque muchas veces no son comprendidos en su totalidad a pesar de ser considerados conceptos básicos.

4.2.1. Ancho de banda.

El concepto de ancho de banda está relacionado con la cantidad de datos que se pueden transmitir en un lapso determinado de tiempo, en si es la velocidad que se puede alcanzar y generalmente es ofrecida por los proveedores de Internet hacia sus clientes, se expresa cantidad de datos/tiempo por ejemplo Gb/s o Gbps, Mbps, Kbps etc.

El ancho de banda maneja dos tipos de velocidades simétricas y asimétricas. Las velocidades simétricas son las que manejan una misma velocidad tanto de subida como de bajada, este tipo de velocidades son contratadas por la mayoría empresas por que necesitan una velocidad adecuada para que exista una buena comunicación entre sus diferentes sucursales, debido a que la carga de archivos es constante es necesaria tener una velocidad que cumpla con estos requerimientos, mientras tanto la velocidad asimétrica tiene una velocidad de subida menor que la velocidad de bajada, por eso es usada por la mayoría de usuarios debido a que no existe prioridad en la carga de archivos sino se hace un uso constante en la descarga de archivos.

4.2.2. Rendimiento (Throughput).

También conocido como Throughput, es la capacidad de un enlace de transportar información útil. Representa la cantidad de información útil que puede transmitirse por unidad de tiempo. (Vesga & Granados, 2012)

El Throughput se ve afectado debido a ciertos factores los cuales limitan a todas las redes debido a que son usuales y siempre son tomadas en cuenta, además afectan a que una red no se desempeñe al 100%, estos factores son:

El medio físico es uno de los más confiables a comparación del medio inalámbrico, pero al enfocarse al medio inalámbrico se tiene factores que afectan directamente como es la distancia, clima, cantidad de nodos o dispositivos, seguridad entre otros.

En el medio inalámbrico se necesita mayor cantidad de paquetes para establecer una conexión lo que influye directamente en el rendimiento de la red, ya que necesita más confirmaciones por el hecho de enviar un paquete y esperar una respuesta, además cuando intervienen los protocolos cada uno trata de manera diferente a los paquetes tanto para el envío, retransmisión y descarte de paquetes. Todo este conjunto de factores debe ser tomados en cuenta para determinar qué tan óptimo puede llegar a ser la velocidad entrante y saliente del dispositivo con respecto a estas amenazas.

4.2.3. Delay.

También conocida como latencia o retraso y se define como la suma de los retardos dentro de una red, la latencia cero no existe sino siempre habrá un tiempo mínimo en la propagación de información de un dispositivo a otro, en transmisiones UDP la latencia se hace más perceptible, a diferencia que TCP la latencia se notará si es demasiado alta. El retraso influye el tamaño del paquete, la cantidad de información que quiere ser enviada y como esta es procesada en cada dispositivo.

En todos los tipos de tráfico admiten cierta tolerancia a lo que es la demora como, por ejemplo:

Juegos de ordenador - la experiencia demuestra que un retraso de 100 milisegundos es todavía aceptable, no obstante, por encima de ese límite, especialmente en el caso de los juegos de acción rápida, tales retrasos pueden dificultar el juego.

Video - ver películas en línea, como por ejemplo mediante el servicio de YouTube requiere un nivel de la velocidad de transmisión relativamente alto. Latencia de 300-500 milisegundos podría resultar en las interrupciones de la película, la pérdida de sincronización entre imagen y sonido, o en algunos casos causar la distorsión de imagen.

Teletrabajo, especialmente con el escritorio gráfico - un retraso de un segundo en muchos casos causa que la conexión falle y entonces hay que establecerla de nuevo, los valores que oscilan alrededor de 300 milisegundos tienen un impacto notable en la reducción de la comodidad de trabajo.

Aparte de los retrasos causados por mal funcionamiento de los enlaces de transmisión en algunas situaciones se puede encontrar retrasos causados por el software de sistema - como controladores de dispositivos inacabados, módulos de soporte de algunas funciones etc. (SPEED-TEST, s.f.)

4.2.4. Jitter.

Se define como una variación en el retardo de los paquetes recibidos. En el lado emisor, los paquetes se envían en un flujo continuo con los paquetes espaciados uniformemente separados. Debido a la congestión de la red, gestión de colas inadecuada o errores de configuración, este flujo constante puede convertirse en bultos, o el retardo entre cada paquete puede variar en lugar de constante restante. (CISCO, 2006)

El jitter o fluctuación puede variar, pero para tener una mejor calidad de servicio (QoS) no debe sobrepasar los valores de tolerancia que fueron indicados en la latencia.

El análisis se basará en el escenario o archivo Simulacion AODV.tcl que fue mostrado en el capítulo 3, como se puede apreciar en las siguientes figuras el tiempo establecido para el análisis es de 0 a 120 segundos.

Se tiene que tomar en cuenta que los nodos a los 0 segundos están activos y configurados listos para su funcionamiento, a los 0.1 segundos los nodos comienzan a efectuar el envío y recepción de información.

A continuación, se explicará las figuras obtenidas de la simulación de cada tipo de tráfico.

4.3. Gráfica del tráfico exponencial

Figura de tráfico exponencial

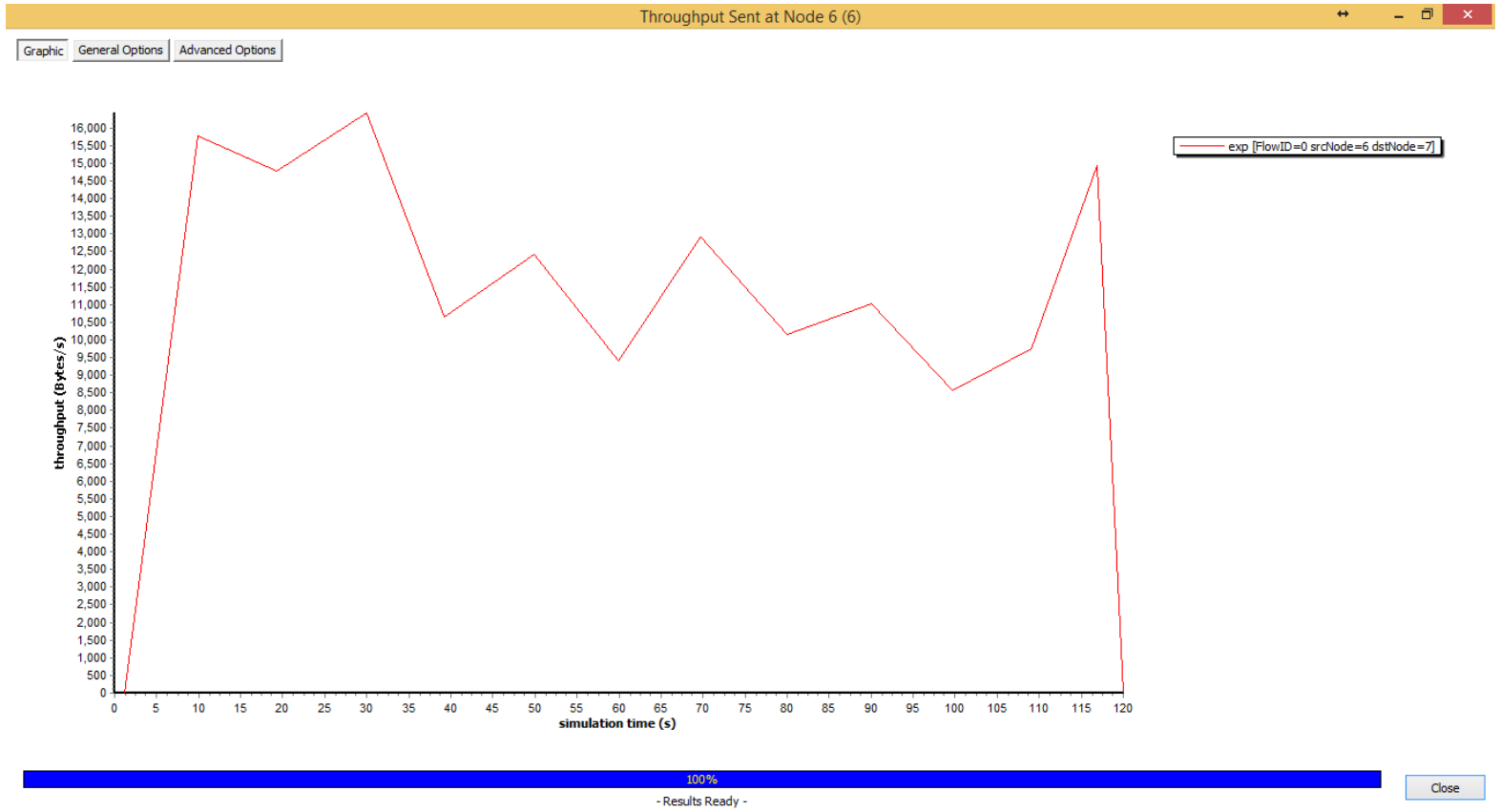

Figura 45. Gráfica del tráfico exponencial
Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 18. Datos del tráfico exponencial

TIEMPO	BYTES
0,00	0,00
1,15	0,00
9,85	15788,10
19,29	14793,35
29,91	16428,10
39,27	10663,76
49,94	12409,93
59,96	9411,87
69,77	12916,99
79,95	10157,89
89,99	11033,67
99,61	8563,38
109,09	9751,48
116,92	14941,26
120,00	0,00

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

En el eje <<y>> se establece el rendimiento medido por bytes/s, el máximo número de rendimiento es 16428,10 Bytes y en el eje <<x>> se establece el tiempo, que en este caso y en todas las figuras siguientes es de 120 segundos.

Como se explicó anteriormente el envío de información se efectuará a los 1,15 segundos, pero como se observa en la figura el valor de inicio de envío es mayor, debido a que existen procesos dentro de los nodos los cuales pueden ser: la identificación del protocolo, nodo origen y nodo destino, búsqueda de rutas, mensaje de petición entre otros (Observar la tabla 8). Es decir, al tiempo 1,14 segundos da inicio al proceso para realizar el envío del primer paquete. (Los primeros paquetes enviados son los de petición o request).

La curva creciente representa que los datos (bytes) se están transmitiendo por la red y al tiempo 9,85 segundos se observa que el rendimiento de la red llega a un máximo de 15788,10 bytes.

Al enfocarse en los puntos críticos de la figura de mayor y menor rendimiento que serían en los tiempos 29,91 y 99,61 respectivamente (no se toma en cuenta el inicio y el final del proceso ya que estos datos son nulos como se explicó anteriormente).

Tiempo 29,91

En este tiempo se observa que los bytes que han sido enviados son 16428,10, los incrementos que surgen a través de la ejecución se deben al proceso de tráfico que se está utilizando que en este caso es el exponencial (ver capítulo 3 sección 3.3.7), ya que durante un lapso de 500ms o medio segundo el nodo 6 exige una mayor cantidad de envío de paquetes debido al proceso de tráfico utilizado, en el total del alza también interviene una mejora en la calidad. En este segundo se representa que el nodo 6 no está siendo utilizado para encontrar una nueva ruta ya que el protocolo AODV los nodos son constantemente utilizados, ya que otros nodos generan broadcast de peticiones que pasan por nodos adyacentes para llegar a su destino, es decir, que el nodo 6 y nodo 7 están exclusivamente utilizados para enviar la información entre sí. (Observar figura 46)

Tiempo 99,61

En este tiempo se observa una caída en el rendimiento de la red en la cual se envían solamente 8563,38 bytes. Esta caída no representa un riesgo de pérdida de conexión ya que los datos se están transfiriendo en menor escala a pesar de que los dos nodos se encuentran en movimiento, la caída simplemente se debe a una baja de señal, las cuales acoplan para seguir con su conexión. El tráfico exponencial no simplemente cumple con la función de incrementar la cantidad de datos sino también de disminuir la cantidad de información que se envía.

Como se conoce el protocolo TCP realiza una transmisión segura de la información, mediante el envío de ACK (acuses de recibo), al momento que se envía información desde el nodo origen, el nodo destino responde con una ACK por el mismo canal del cual se recibió los datos para ver si la información pudo llegar a su destino, en el caso de que no llegue la información se reenvía nuevamente, este proceso de respuesta a la petición se ve reflejado en la figura como decrementos, el rendimiento va en descenso hasta que el canal quede libre para el envío.

En la figura de rendimiento no se muestran paquetes perdidos solo la influencia que ejercen los paquetes que son transportados por la red, pero también muestra si en la red existen tiempos en los que no haya conexión.

Como se observa en la figura 45 no existe ningún tiempo de inactividad o que el funcionamiento de la red detenga su proceso de transmisión garantizando el acceso al servicio.

Rutas de los nodos del tráfico exponencial

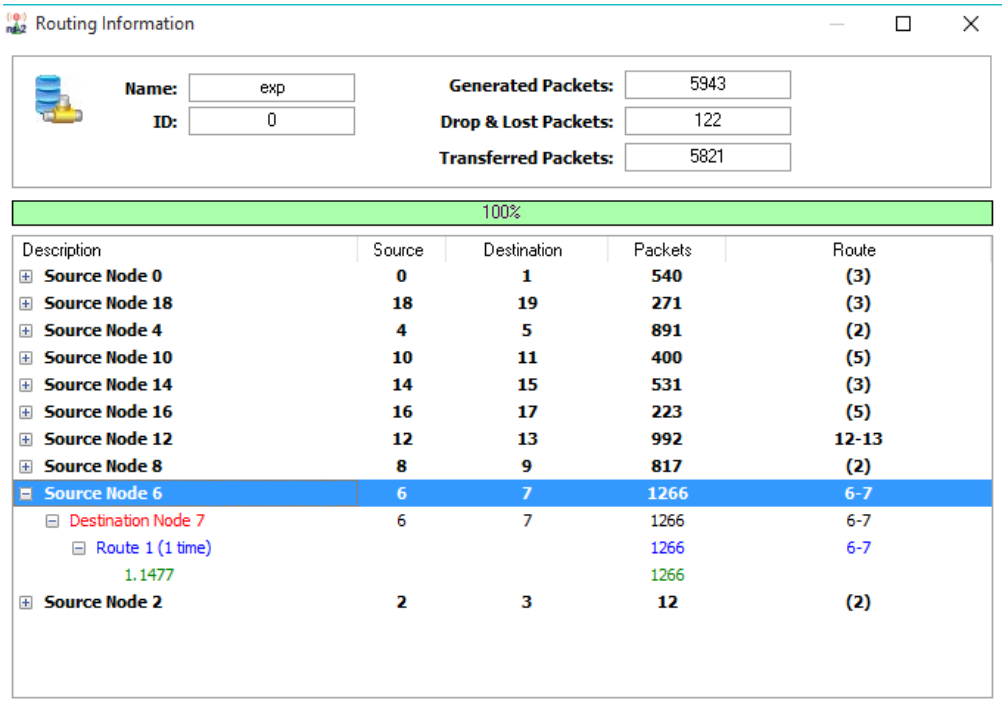

Figura 46. Rutas de los nodos del tráfico exponencial
Elaborado por: Jaime Acosta y Jhonatan Quelal

4.4. Gráfica del tráfico pareto

Gráfica del tráfico pareto

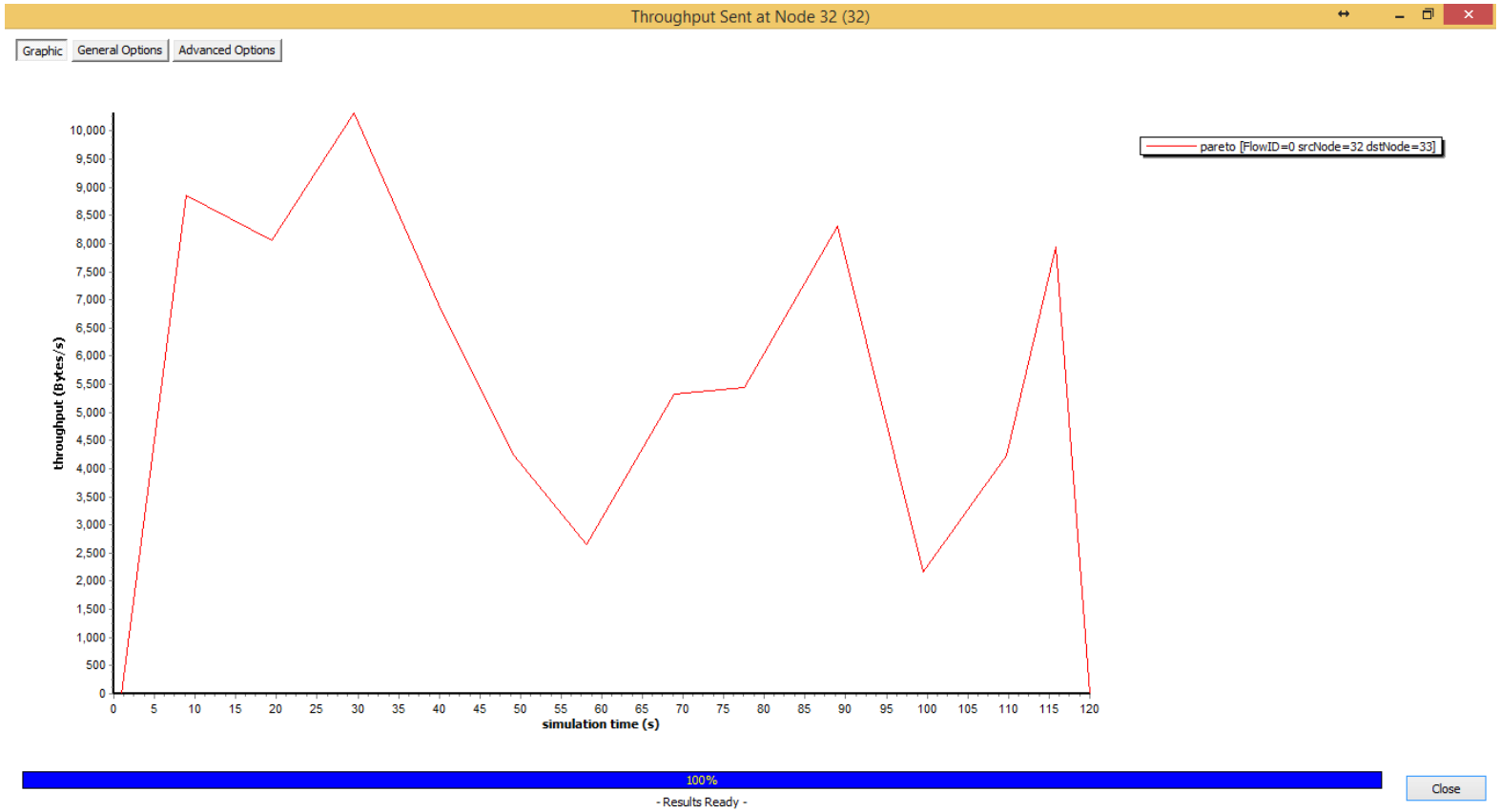

Figura 47. Gráfica del tráfico pareto
Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 19. Datos del tráfico pareto

TIEMPO	BYTES
0,00	0,00
0,94	0,00
8,93	8861,92
19,46	8058,43
29,48	10319,87
39,96	6928,18
49,17	4236,17
58,10	2661,77
68,85	5317,95
77,51	5441,38
88,95	8311,70
99,44	2156,65
109,76	4229,86
115,77	7938,25
120,00	0,00

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

En el eje <<y>> se establece el rendimiento medido por bytes/s, el número máximo de rendimiento es 10319,87 Bytes y en el eje <<x>> se establece el tiempo, que es de 120 segundos.

En el tráfico pareto el envío de información se efectuará a los 0.94 segundos.

El tráfico pareto es usado en NS2 para generar diferente cantidad de información en cada iteración de tiempo como se explica en el capítulo 3 sección 3.3.8, la dinámica utilizada por este tipo de tráfico es para asemejarse a la transmisión de datos real que existe en las redes; ya que una red no es constante en cuanto a datos recibidos o enviados y varían dependiendo a las exigencias del usuario, es por ello que el uso resulta importante para ver cómo actúa la red en base a estos cambios.

En la figura se puede observar que la red varia constantemente, la parte ascendente del inicio indica que entra en funcionamiento el envío o petición para el establecimiento de conexión, el envío de conexión llega a la cantidad de 8861,92 bytes al segundo 8,93. La cantidad máxima enviada al segundo 29,48 es de

10319,87 bytes, lo que implica que la función del tráfico utilizado entró en vigencia en cuanto a la cantidad máxima efectiva que puede ser enviada.

Los descensos que se provocan se deben en parte, a que en la red se generan cambios de cantidad de información transmitida debido al funcionamiento del tráfico pareto, pero como se puede observar la red solventa este tipo de situación y no existe un deterioro total de la red por lo que continua con su proceso. Por otra parte, también toma en cuenta que los envíos de los paquetes utilizan trafico TCP, lo cual creará una recesión en los datos debido a los acuses de recibo o respuestas que genera el de recepción, como se comentó anteriormente las respuestas hacia el nodo origen son enviadas por el mismo canal por el cual se recibió la información.

Como se puede observar en la figura 47 no se encuentra ninguna caída que conlleve a una pérdida total o parcial de datos, dando como resultado un funcionamiento estable que se enfrenta a diferentes cantidades de información enviados en diferentes lapsos de tiempo.

En la figura 47 se puede observar que se envía 2156,65 bytes al tiempo 99,44 segundos, en este tiempo el rendimiento de la red baja debido a que existe un cambio de ruta y como a través de esto se generaron paquetes de broadcast en búsqueda de la nueva trayectoria, el rendimiento de la red decae ya que se genera una nueva conexión, la primera ruta se genera a partir de la conexión entre los nodos 32-29-33, en este punto toma en cuenta el mensaje de respuesta de la petición y la toma como una ruta óptima debido a la rapidez de respuesta que obtiene mediante dicha ruta, la conexión permanece activa durante la transmisión de 423 paquetes, posteriormente se genera otra ruta 32-8-33, este cambio es provocado por la movimiento que existe entre los nodos ya que al alejarse pierde la conectividad y el

nodo origen opta por encontrar un nuevo camino, es importante mencionar que esto ocurre en milisegundos, el proceso pasa desapercibido por el usuario, pero continúa con el envío satisfactorio de la información. Ver figura 48.

Rutas de los nodos del tráfico pareto

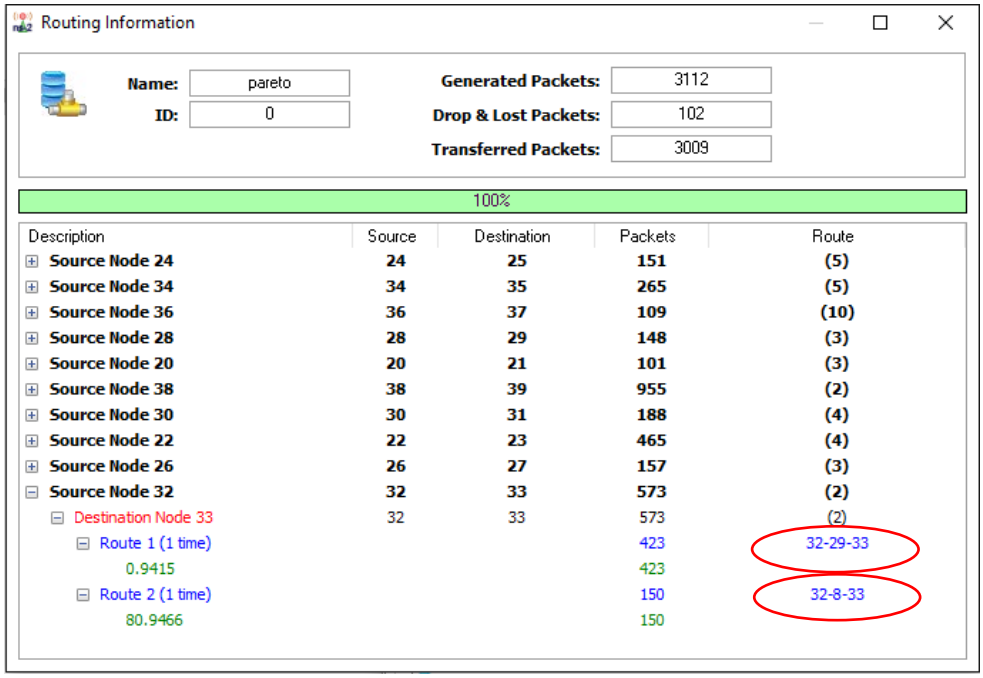

Figura 48. Rutas de los nodos del tráfico Pareto
Elaborado por: Jaime Acosta y Jhonatan Quelal

4.5. Gráfica del tráfico ftp

Gráfica del tráfico ftp

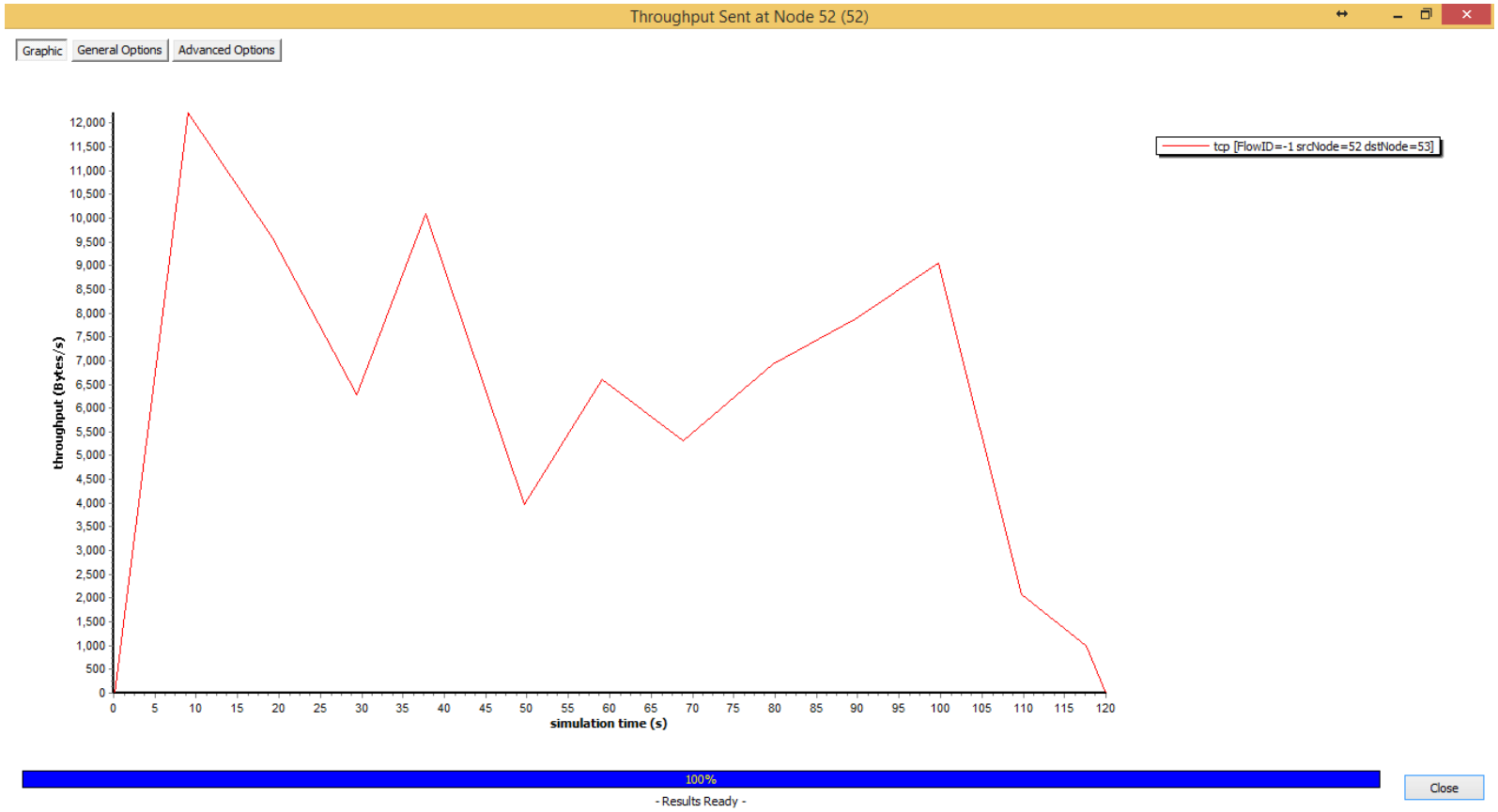

Figura 49. Gráfica del tráfico ftp
Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 20. Datos del tráfico FTP

TIEMPO	BYTES
0,00	0,00
0,10	0,00
8,96	12215,56
19,24	9589,75
29,35	6277,45
37,79	10098,89
49,64	3968,67
59,04	6602,08
68,87	5318,25
79,76	6928,17
89,67	7865,95
99,78	9050,83
109,81	2080,28
117,64	987,21
120,00	0,00

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

En el eje <<y>> se establece el rendimiento medido por bytes/s en este caso el máximo número de rendimiento es 12215,56 Bytes y en el eje <<x>> se establece el tiempo de 120 segundos.

En el tráfico ftp el envío de información se efectuará a los 0,1 segundos.

El tráfico ftp es usado en NS2 para verificar que la red pueda controlar la información basándose en la simulación de descarga, es decir, el nodo destino querrá obtener datos del nodo origen. El proceso que conlleva NS2 para la verificación que el nodo origen esté disponible es la creación de puerto, para el envío y recepción de peticiones. Los nodos que quieran acceder a esta información generaran un paquete de petición que será receptado y resuelto por el nodo origen para la transmisión de datos.

FTP utiliza el protocolo de transmisión segura de datos que es TCP, en el caso del simulador utilizado no tiene importancia que la información sea un mail, una imagen, documento etc. Es indiferente ya que al verse reflejado en la red se

representan en 0 y 1, lo que toma en cuenta el simulador es la cantidad de estos bytes enviados, es decir, la cantidad que datos que se va a transmitir.

Como se puede observar en la figura 49 en el tiempo 8,96 segundos la curva creciente llega un máximo de 12215,56 bytes que puede transmitir, debido a que se estableció una ruta inicial que permitió un buen rendimiento, además los nodos de la ruta se encontraban en el alcance apropiado dando como ventaja una señal adecuada para que la información se pueda propagar, a partir de este punto se efectúan siete cambios de ruta, lo que ocasiona que el rendimiento baje, pero la red regula el establecimiento de conexión para que no existan puntos críticos, si se observa la figura 41 en el capítulo 3 se visualiza que los nodos 52 y 53 en este caso están alejados, por lo que es necesario el uso de nodos intermedios para formar la ruta.

La red al depender de estos nodos intermedios para la transmisión de información se encuentra en una posición en la cual si uno de estos nodos intermedios o extremos se mueve tendrá que establecer otra ruta.

Entre todos los nodos existentes y caminos posibles el protocolo AODV toma una ruta por la que obtenga una rápida respuesta, cuando los nodos están lo más juntos posible existen una menor posibilidad de pérdida de la trayectoria, pero en este caso como se observa en las figuras anteriores existe una mayor distancia entre los nodos lo que da como resultado que la red tenga descensos o un bajo rendimiento, a pesar de que los nodos estén en una posición poco apropiada, la información nunca deja de ser enviada y el rendimiento nunca llega a 0.

Cabe indicar que los descensos que se menciona no solo implica la distancia en la que se encuentra un nodo a otro, sino que también interviene el proceso que efectúa

el tráfico FTP ya que en su funcionamiento implica que cuando un nodo receptor genera una petición de descarga hacia un nodo origen entra en funcionamiento la clase Application/FTP la cual genera randomicamente la cantidad de descargas a través de una petición del nodo destino.

Es por ello que NS2 trata de simular dos o más descargas lo que esto es muy común en las redes reales para ver cuál es la influencia en una red simulada, es por eso que en las redes actuales entre más descargas se realice el rendimiento de la red decae.

Rutas de los nodos del tráfico FTP

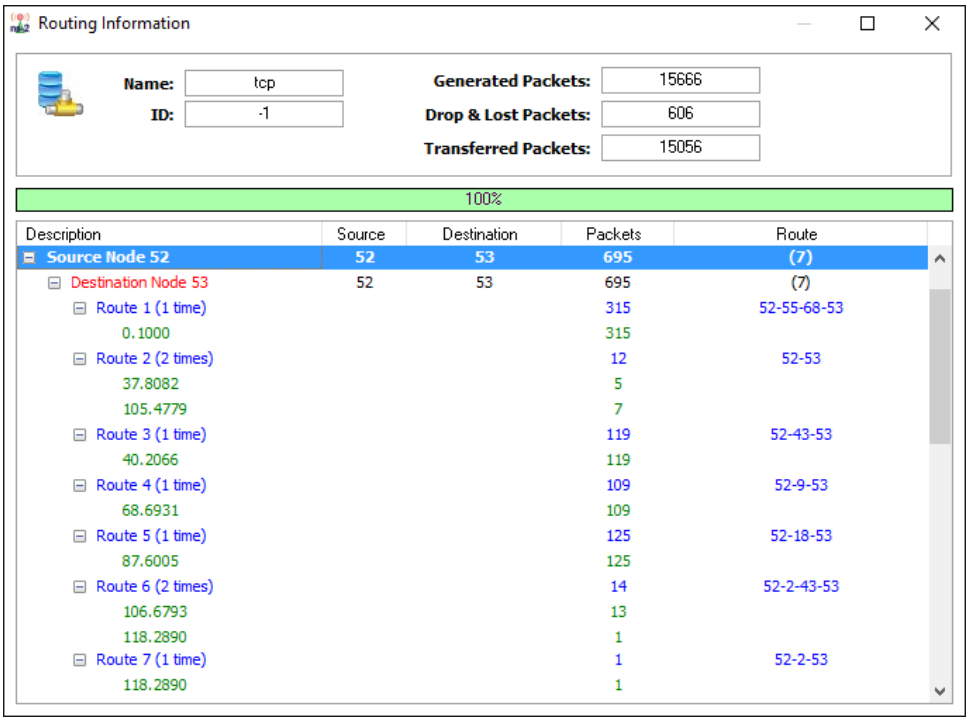

Figura 50. Rutas de los nodos del tráfico FTP
Elaborado por: Jaime Acosta y Jhonatan Quelal

4.6. Gráfica del tráfico CBR

Gráfica del tráfico CBR

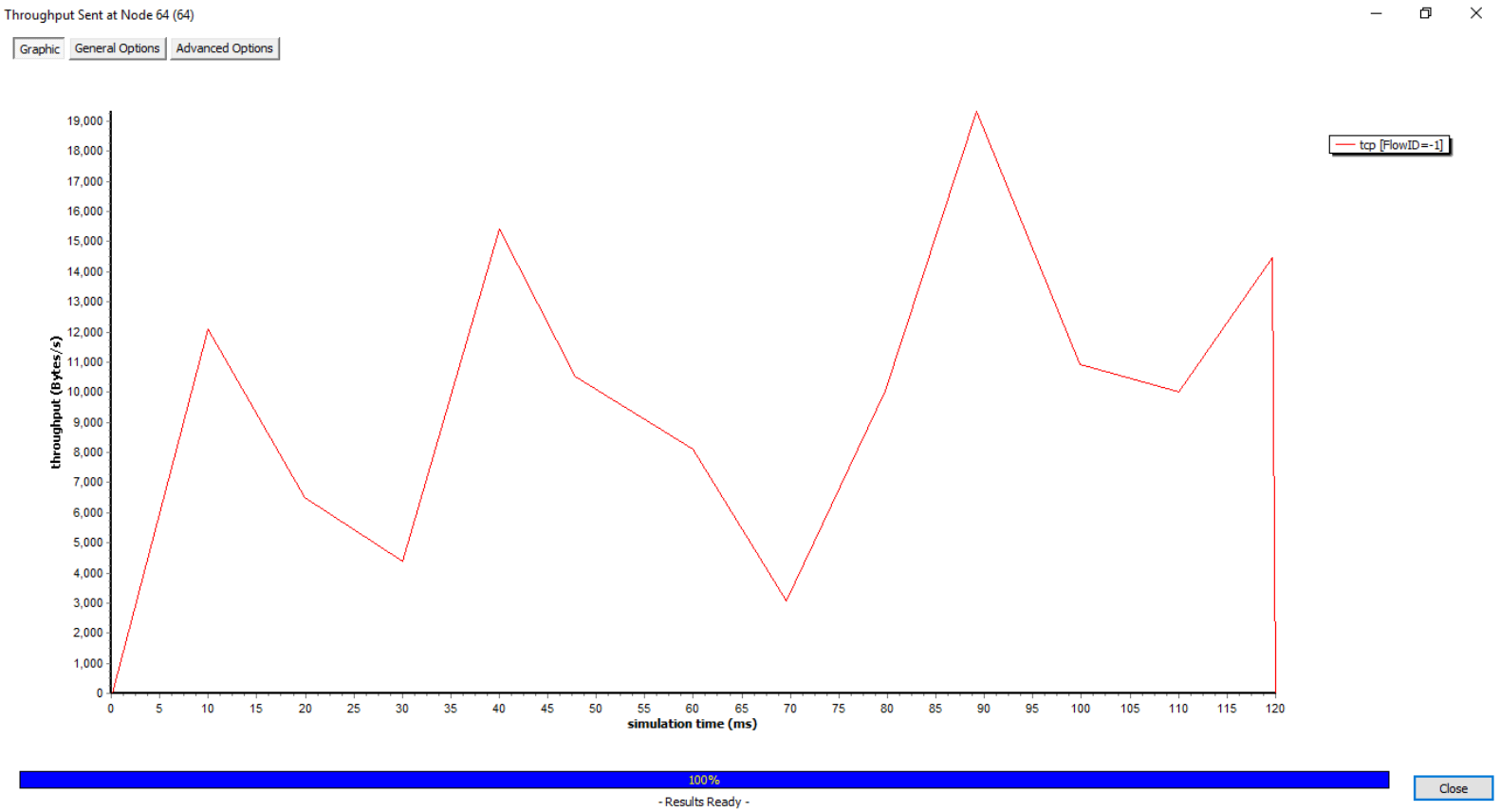

Figura 51. Gráfica del tráfico CBR
Elaborado por: Jaime Acosta y Jhonatan Quelal

Tabla 21. Datos del tráfico CBR

PAQUETES	TIEMPO
0,00	0,00
0,10	0,00
9,99	12093,35
19,93	6498,05
30,00	4371,91
39,99	15420,73
47,85	10506,81
59,92	8119,46
69,55	3084,13
79,76	10040,59
89,22	19327,28
99,83	10918,34
109,97	10026,17
119,68	14462,48
120,00	0,00

Nota. Elaborado por: Jaime Acosta y Jhonatan Quelal

En el eje <<y>> se establece el rendimiento medido por bytes/s, en este caso el máximo número de rendimiento es 19327,28 Bytes y en el eje <<x>> se establece el tiempo de 120 segundos.

En el tráfico CBR el envío de información se efectuará a los 0,1 segundos.

El tráfico CBR (Constant Bit Rate) tasa de bits constantes, es usado en NS2 con el fin de que la red simulada se enfrente a un constante flujo de datos durante intervalos de tiempo, estos lapsos de tiempo son determinados por el proceso <<random>>.

Como se puede observar en la figura 51 en el tiempo 89,22 segundos la curva creciente llega un máximo de 19327,28 bytes que puede transmitir, se puede visualizar que el rendimiento de la red aumenta cada cierto periodo de tiempo y al parecer los datos no son constantes, pero dentro de cada uno de estas crecientes trata de alcanzar la cantidad de datos que son requeridos para transmitir, el descenso de estas curvas se debe a los cuatro cambios de rutas que ocurrieron durante el envío

(ver figura 52), la figura no demuestra ningún momento de inactividad lo que permite deducir que la red efectuó satisfactoriamente el proceso de transición y retransmisión de datos, hay que comprender que las gráficas mostradas también bajan el rendimiento si no existe datos a enviar, pero también se está enfrentado procesos internos de procesamiento de respuestas o peticiones y qué hacer con la información que tiene cada nodo, es decir, el rendimiento también depende de la velocidad de procesamiento de cada dispositivo y que tan rápido pueda despachar los paquetes encolados.

Rutas de los nodos del tráfico CBR

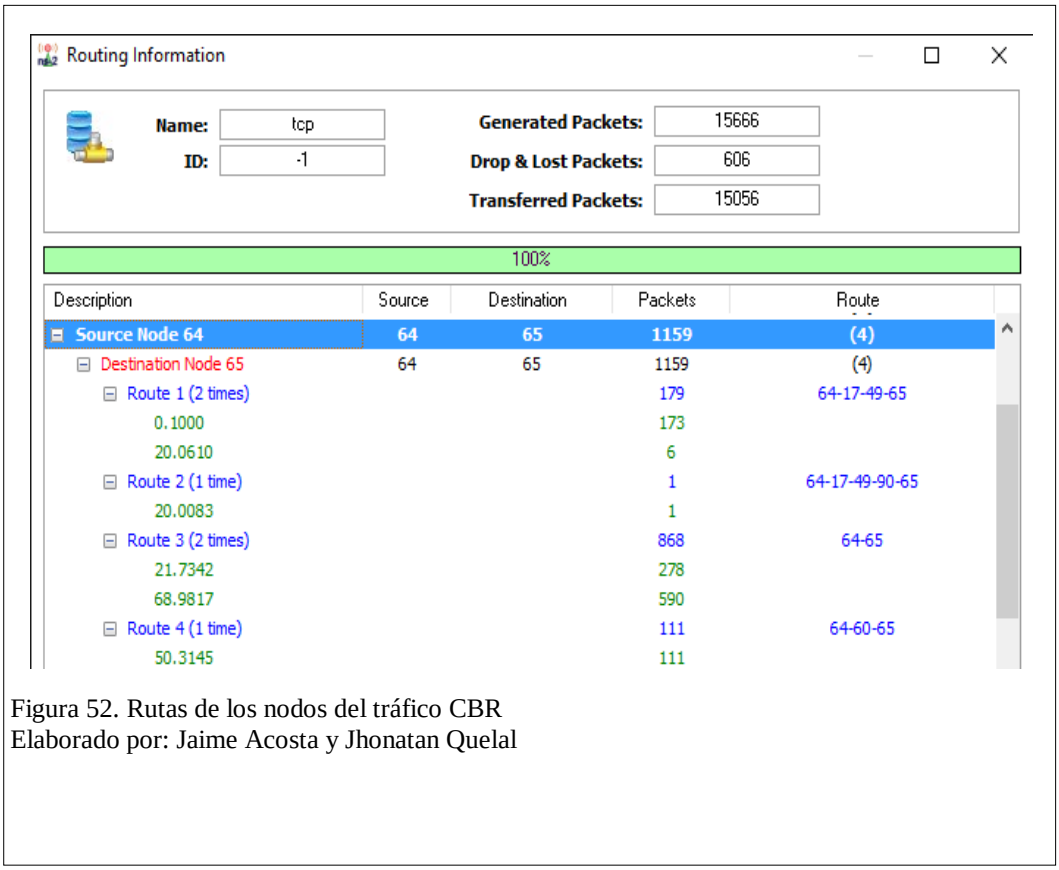

Figura 52. Rutas de los nodos del tráfico CBR
Elaborado por: Jaime Acosta y Jhonatan Quelal

4.7. Gráfica del tráfico delay (retardo)

Gráfica del tráfico delay (retardo)

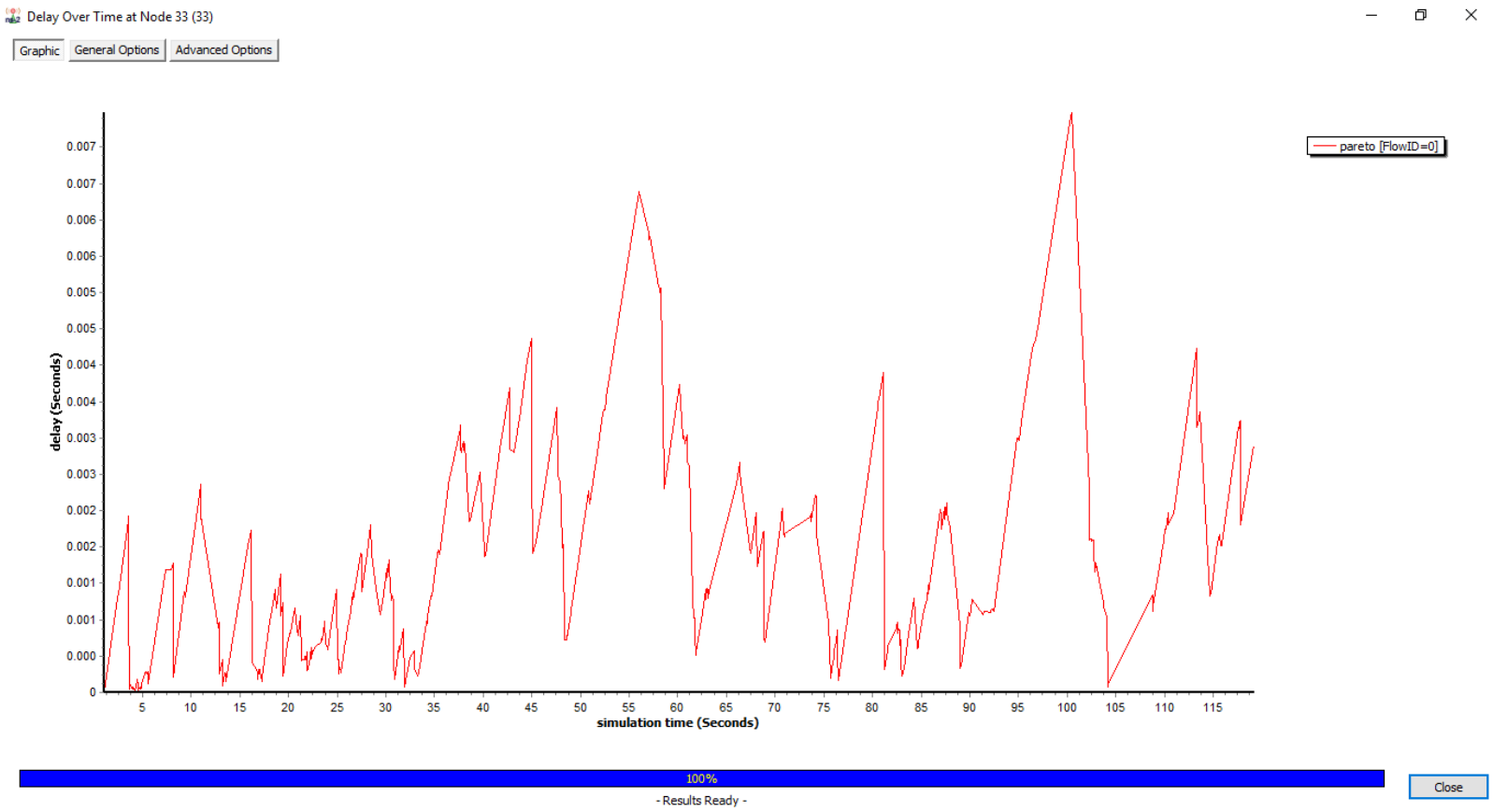

Figura 53. Gráfica del tráfico delay (retardo)
Elaborado por: Jaime Acosta y Jhonatan Quelal

En el eje <<y>> se establece el retraso de los paquetes en segundos en este caso el máximo número de retraso es 0,007975 segundos y en el eje <<x>> se establece el tiempo 120 segundos.

Los retrasos (ver figura 53) registrados a lo largo de la ejecución son inferiores a lo establecido, como se puede observar en la figura 53 el máximo retraso registrado fue en el tiempo 100,46175 segundos, con una cantidad de 7,975 milisegundos por paquete, lo que demuestra que la red simulada es estable y la tramision de paquete se efectúa de manera exitosa, a pesar de no usar una infraestructura intermedia y hacer uso de los dispositivos del usuario la señal obtenida funciona de manera adecuada, brinda una buena calidad de servicio, a pesar de que esta atravesando diferentes rutas la informacion no se se pierde y busca la manera de llegar a su destino, atraviesa los ditintos caminos de tal manera que no afecte en lo mas mínimo al funcionamiento, con el fin de realizar una entrega de datos a una velocidad satisfactoria para al usuario.

La red en la simulación funciona de manera interna, pero si se establece una conexión con servidores del exterior dependerá de la conexión que llegue al nodo de acceso, la red funcionara de igual manera ya que el punto de acceso es simplemente otro nodo visible para los nodos o dispositivos del usuario.

La figura 53 es el retraso que existe en los nodos 32 – 33, las gráficas en todos los nodos fueron analizadas pero presentaron las mismas características pero con una variación de ±3 milisegundos. Lo que significa que la red solventa cualquier tipo de tráfico con un retraso mínimo.

4.8. Gráfica del tráfico jitter

Gráfica del tráfico jitter

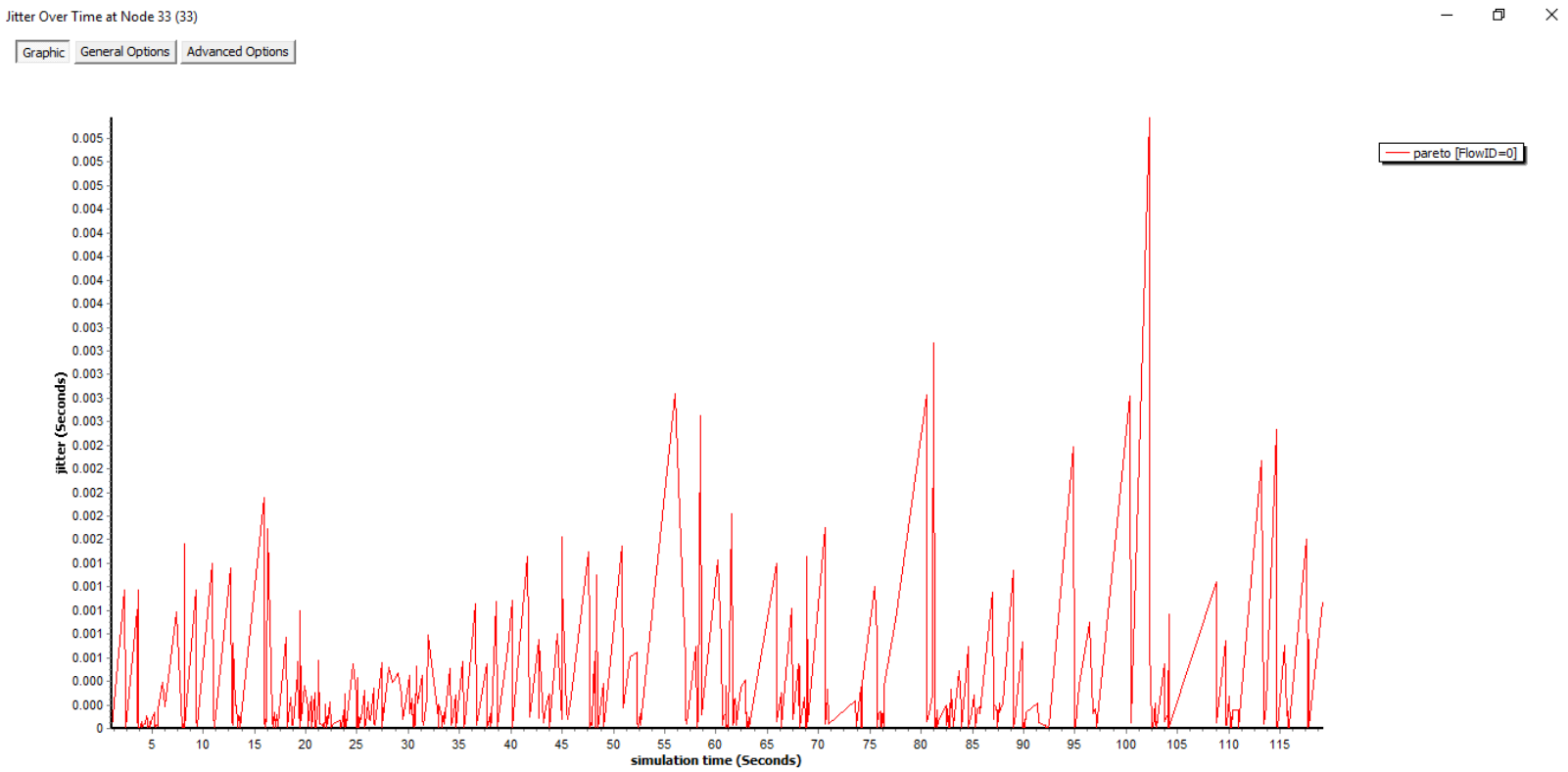

Figura 54. Gráfica del tráfico jitter
Elaborado por: Jaime Acosta y Jhonatan Quelal

Como se muestra en la figura 54 la variación del tiempo en la llegada de los paquetes viene dada con un máximo de 0,005176 segundos y un mínimo de 0,000001 segundos, descartando el 0 como valor a tomarse en cuenta el jitter se lo considera mucho más perjudicial en la red que el retraso debido a que se da a notar mucho más por el usuario pero al hablar de 5 milisegundos es imperceptible para el usuario final, este valor es tolerable al momento de realizar un envío o en la transmisión ya que no sobrepasa los estándares para realizar una buena calidad de servicio demostrando así que la simulación ofrece resultados viables para su implementación. En las gráficas del jitter también fueron analizadas, pero de igual manera los valores tienden a cambiar de ±4 milisegundos, aun así, tras efectuar una nueva búsqueda de una nueva ruta no implica mayor dificultad que la de una red comúnmente usadas como por ejemplo ADSL, satelital, HFC, entre otras, que usan protocolos de enrutamiento como OSPF, RIP, EIGRP, etc.

La red simulada permite acoplar todas estas interferencias para poder ser analizadas en la ejecución, alcanzando medidas que son tolerables dando por hecho que la red funcionará si se coloca en circunstancias extremas ya que en la simulación solo se ha usado 100 nodos y un solo punto de acceso para una área extensa, y el resultado fue satisfactorio ya que con esta mínima cantidad de dispositivos se logró cubrir la área de san Martín, si en los mejores casos se colocara más nodos, el área junto con el protocolo y la red full mesh cubriría los puntos de difícil acceso, dándonos una gran ventaja de no solamente poseer el Internet en el hogar sino también los usuarios podrán acceder a la red en cualquier punto que se encuentren siempre y cuando estén en la zona de San Martín.

CONCLUSIONES

- El proyecto de simulación fue basado en una red ad hoc que cumple con la exigencia mínima en cuanto a materiales para el establecimiento de la red, lo que sería beneficioso a diferencia de las redes usadas actualmente que necesitan de un módem que es entregado por el proveedor a cada uno de sus clientes, en el caso de la red Ad hoc no necesitan estos módems, ya que los propios dispositivos de los usuarios como smartphones, tablets, laptops entre otros, se vuelvan conductores de información y al mismo tiempo un punto de acceso, a mayor cantidad de dispositivos conectados en la red, el alcance de cobertura aumenta, para comprobar que la red funcione adecuadamente con 2690 usuarios, es decir, el número actual de habitantes que tiene el sector de San Martín, y basando la simulación en la muestra estadística del 90% de confiabilidad y con ±6% de error en el total de la población y dando resultados estables, esto conlleva a que la red contará con una alta disponibilidad y además funcionará adecuadamente con el total de usuarios.

- Las redes ad hoc al ser utilizadas en entornos amplios y de alta concurrencia no tendrán problemas en solventar todas las peticiones de conexión, reconstrucción de nuevos caminos, reenvíos necesarios, alertas, actuando de manera rápida y de forma invisible para el usuario, logrando cubrir zonas de difícil acceso al tener la opción de conectarse con distintos dispositivos intensifica la señal fortaleciendo la estructura de la red ya que dichos dispositivos se entrelazan y ayudan a identificar un camino o ruta para entregar o adquirir información desde un nodo origen hacia un nodo destino.

- Según el análisis realizado en la red al tener un mínimo de nodos que permiten la conectividad y no hacer uso de puntos fijos de acceso se podrá utilizar del servicio desde cualquier ubicación del sector, es decir, si un dispositivo quiere conectarse a la red desde la calle y no se encuentra ubicado cerca de su hogar lo podrá hacer sin problemas siempre y cuando existan nodos intermedios para su conectividad.

- En las gráficas obtenidas de la simulación se muestran pérdidas, caídas o bajas en la transmisión de información, esto no conlleva a que la red deje de funcionar, sino que todo tipo de red se encuentra sometida a diferentes circunstancias, por ejemplo, si un nodo origen hacia su destino está conectado a través de un nodo intermedio, y este se apaga o se aleja genera una acción de reconexión lo que da como resultado una caída parcial en la transmisión de la información.

- Según el análisis realizado al archivo t_AODV.tr en el que se encuentra especificado todo el proceso durante la ejecución obtenido a través de la simulación en el programa NS-2, en este archivo no se detectó ninguna acción de tipo <<***C***>> (tipo colisión) o <<***loop***>> (tipo bucle) lo que significa que la red está libre de este tipo de fallos gracias al protocolo AODV.

- Las antenas fueron colocadas en base a los niveles de elevación para que puedan tener un mayor campo de visión sobre la zona San Martín independientemente de estas localizaciones llamados puntos de acceso hacia el Internet podrán situarse de forma que se crea conveniente dependiendo de los dispositivos a los cuales se quiera

cubrir, lo que implica que la red no se basa en los puntos de acceso sino la cantidad de dispositivos móviles que vayan hacer uso de la red para efectuar su funcionamiento.

RECOMENDACIONES

- Para que la transmisión de información se desenvuelva de una mejor manera no es suficiente con aumentar el ancho de banda, también se podría aumentar el buffer y la clasificación de tráfico en cada dispositivo, lo que ayudará a que el envío de información mejore.

- El proyecto fue analizado en base a resultados ofrecidos por el programa Analyser NS2, se recomienda el estudio de dicho programa ya que revela procesos intermedios y acciones que realizan los nodos y la cantidad de información al propagarse por el medio de transmisión.

- El proyecto está enfocado a una red completa ad hoc ya que para establecerse no hace uso de una infraestructura y puede ser implementada sin ningún altercado, pero también sería recomendable vincularla con otras redes con el fin de mejorar los puntos de difícil acceso, es decir, hacer uso de una infraestructura preexistente y expandir la calidad de servicio.

- Se recomienda realizar la simulación en el tiempo moderado (mínimo 60 segundos) ya que con esto sería suficiente para establecer resultados de análisis, si se simula en un tiempo mayor a 540 segundos la información se vuelve redundante ya que se sometería a los mismos términos o procedimientos de la simulación en una forma repetitiva, dando por hecho que el programa NS2 es efectivo a lo hora de entregar resultados.

- Se recomienda tratar de ubicar las antenas de acceso en las localizaciones propuestas en el proyecto con el fin de mejorar el acceso al servicio obtentiendo una mejor visualización de los nodos hacia la antena, teniendo en cuenta que la altura de las antenas de acceso pueden variar dependiendo de los nuevos cambios en el sector, en cuanto a edificaciones.

REFERENCIAS

CISCO. (2006). *Understanding Jitter in Packet Voice Networks* . Retrieved from http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/18902-jitter-packet-voice.html

Feedback Networks Technologies. (2013). Retrieved from Feedbacknetworks: http://www.feedbacknetworks.com/cas/experiencia/sol-preguntar-calcular.html

Hasson, S. T., & Jasim, S. (2012). *computerscijournal.* Retrieved from http://www.computerscijournal.org/vol7no1/simulation-study-to-observe-the-effects-of-increasing-each-of-the-network-size-and-the-network-area-size-on-manets-routing-protocols/

Henderson, T. (2011, 11 05). *Newreno TCP*. Retrieved from http://www.isi.edu/nsnam/ns/doc/node399.html

INEC. (2010). *Fascículo Provincial de Pichincha.* Retrieved from http://www.ecuadorencifras.gob.ec/wp-content/descargas/Manu-lateral/Resultados-provinciales/pichincha.pdf

INEC. (2016). *INEC.* Retrieved from http://www.ecuadorencifras.gob.ec/

INEC. (2016). *INEC.* Retrieved from http://www.ecuadorencifras.gob.ec/base-de-datos-censo-2010/

Issariyakul, T. (2009). *ns2ultimate.* Retrieved from http://ns2ultimate.tumblr.com/post/20835608051/post-processing-ns2-result-using-ns2-trace

Perkins, C. E., Belding-Royer, E. M., & Das, S. R. (2003, julio). *RFC 3561*. Retrieved from https://www.ietf.org/rfc/rfc3561.txt

Simulator, N. (n.d.). *Network Simulator*. Retrieved from http://www.isi.edu/nsnam/ns/

SPEED-TEST. (n.d.). *Speed-test.es*. Retrieved from Speed-test.es: http://www.speed-test.es/qu_es_la_latencia

Technologies, F. N. (2013). *Calcular la muestra correcta*. Retrieved from http://www.feedbacknetworks.com/cas/experiencia/sol-preguntar-calcular.html

Technologies, F. N. (2013). *Clacular la muestra correcta.* Retrieved from http://www.feedbacknetworks.com/cas/experiencia/sol-preguntar-calcular.html

Troyano, A. B. (2011, Junio). *Protocolos de encaminamiento en redes inalámbricas mesh: un estudio teórico y experimental.* Retrieved from http://openaccess.uoc.edu/webapps/o2/bitstream/10609/8164/1/abatistet_TFM_0611.pdf

Vesga, J., & Granados, G. (2012, Diciembre). *Análisis sobre el comportamiento del Throughput en redes LAN bajo tecnología Power Line Communications* . Retrieved from http://revistas.ustabuca.edu.co/index.php/ITECKNE/article/view/79/71

VINT. (2000, 01 18). *Running Wireless Simulations in ns*. Retrieved from http://www.isi.edu/nsnam/ns/tutorial/nsscript5.html

Wang, J. (2004). *NS-2 Tutorial.* Retrieved from http://www.cs.virginia.edu/~cs757/slidespdf/cs757-ns2-tutorial1.pdf

Wang, J. (2004). *NS-2 Tutorial* . Retrieved from http://www.cs.virginia.edu/~cs757/slidespdf/cs757-ns2-tutorial1.pdf