

# Análisis del Desempeño del Protocolo OLSR versus BMX en una Red Mallada Inalámbrica en Escenario Real

Eduardo Rodríguez, Claudia Deco, Luciana Burzacca,  
Mauro Pettinari, Cristina Bender

Departamento de Investigación Institucional  
Facultad de Química e Ingeniería (Frei), Universidad Católica Argentina (UCA)  
CP: 2000 - Rosario - SF - Argentina

{ejrodriguez, cdeco, lucianaburzacca, mauro\_pettinari, cbender}@uca.edu.ar

**Abstract.** *The aim of this paper is to analyze and compare the performance of OLSR and BMX protocols on a mesh network configured with OpenWrt firmware using different hardware equipment. OpenWrt is a perfectly good software that can be used in several different devices. Its configuration for using it with BMX and OLSR protocols is simple to perform and has no problem to work with both protocols. Empirical results of several tests are presented using the same scenario. This scenario is a real world network and not a laboratory network, with real tests and no simulations. To measure performance, we use Round-Trip Time - RTT, jitter, probability of error and testing of bandwidth as metrics. Both protocols were installed in a variety of equipment and both presented a satisfactory behaviour.*

**Resumen.** *El objetivo de este trabajo es analizar y comparar el comportamiento de los protocolos OLSR y BMX sobre una red mallada configurada con firmware OpenWrt utilizando distintos equipos de hardware. OpenWrt es un software perfectamente válido que puede ser utilizado en una gran variedad de dispositivos y su configuración para utilizarlo con protocolo OLSR y BMX es sencilla de realizar y no presenta problemas de funcionamiento con dicho protocolo. Se presentan los resultados empíricos de varias pruebas utilizando como escenario una red de mundo real y no de laboratorio, con pruebas reales y no simulaciones. Para medir el rendimiento se utilizaron como métricas el tiempo de ida y vuelta (Round-Trip Time - RTT), el Jitter, la probabilidad de error y el testeo de ancho de banda. Ambos protocolos pudieron ser instalados en una variedad de equipos y presentaron un comportamiento satisfactorio.*

## 1. Introducción

El objetivo de este trabajo es analizar y comparar el comportamiento de los protocolos OLSR [OLSR] y BMX [BMX6] sobre una red mallada configurada con firmware OpenWrt [OpenWRT] utilizando distintos equipos de hardware.

Las redes malladas inalámbricas (Wireless Mesh Networks) han tenido un gran éxito en la historia de las ciencias de la computación y de la ingeniería ([Acuña

Martínez et al., 2006], [Akyildiz et al., 2005]). Sus aplicaciones son numerosas en el dominio industrial, militar y comercial. Son en particular un dominio rápidamente creciente y esto trae muchos desafíos. Un desafío difícil e inmediato es el enrutamiento efectivo debido a la volatilidad típica de tráfico en topologías complejas. Muchos estudios han intentado resolver el problema de enrutamiento mediante métodos heurísticos, pero este enfoque no proporciona los límites de cuán bien se asignan los recursos. Generalmente, en este tipo de investigación se asume que el tráfico de demandas de la red es estático y conocido de antemano, por lo que, estos algoritmos tienden a mostrar un desempeño pobre. Comprender el impacto de la incertidumbre de la demanda en el ruteo y el diseño de algoritmos de enrutamiento para proporcionar robustez, es un problema de investigación aun relativamente incipiente.

Las redes Mesh abiertas son redes ad-hoc descentralizadas que no se basan en infraestructuras previas, como routers o puntos de acceso. En su lugar, cada nodo participa en el enrutamiento, siendo él mismo un router y enviando datos de otros, y de ese modo la determinación de las rutas se hace dinámicamente, basándose en la conectividad que va surgiendo. Para ello, necesitan de protocolos que viabilicen ese comportamiento.

Es de suma importancia el análisis de la performance de diferentes protocolos de comunicación que deben interactuar con diversos dispositivos que hacen al enlace de los nodos de la red a los fines de establecer la integración tecnológica disponible. No menos importante es la determinación de la relación costo/beneficio de una determinada implementación. El conocimiento en tiempo real de la configuración topológica de la red, mediante el uso de distintas herramientas de hardware y software, nos permite el monitoreo de su comportamiento y sus alcances. Todo ello posibilita optimizar la red para que brinde un mejor servicio. En general, la optimización se basa en lograr el mejor camino para enrutar los paquetes de datos, sin demoras o con una demora mínima en función de lograr un mejor aprovechamiento de los recursos utilizados.

En [Rodríguez et al., 2013] se presentó el desempeño de OLSR en una red mallada inalámbrica en un escenario real. En este trabajo, se analiza el comportamiento de dicho protocolo, el comportamiento del protocolo BMX y se los compara en un escenario real. La elección de estos protocolos se basó en el equipamiento con el que se contaba y por el tiempo que se disponía para la experimentación.

En la Sección 2 se presentan algunos conceptos básicos sobre redes malladas y protocolos de ruteo. En la Sección 3 se describe el escenario, hardware y software utilizados. En la Sección 4 las pruebas realizadas. Finalmente, se presentan las conclusiones.

## **2. Redes Malladas Inalámbricas**

Una Red Mallada Inalámbrica (Mesh) es una red compuesta por nodos organizados en una topología de malla. Son redes en las cuales la información es pasada entre nodos en una forma de todos contra todos y en una jerarquía plana, en contraste a las redes centralizadas. Toda variación no prevista en el diseño, puede cambiar su topología y afectar a la distribución de carga de la red y al rendimiento general [Akyildiz et al., 2005].

Las ventajas que presenta frente a otras redes son: el bajo costo al utilizar enlaces inalámbricos, la facilidad de aumentar el área de cobertura incluyendo nuevos nodos ya que no es necesario cambiar infraestructuras como en el caso de las redes

cableadas, la robustez que presenta ante fallos al disponer de rutas alternativas y la capacidad de transmisión que permiten aplicaciones a los usuarios en tiempo real de voz, video y datos. Por tanto se puede incluir un nuevo nodo en cualquier momento y lugar. Como consecuencia el costo de este tipo de redes inalámbricas es mucho menor que en las redes cableadas, ya que no hay que invertir en materiales de cableado y en estudios enfocados a la unión más óptima de los nodos. En la realidad, la topografía raramente viene en forma de anillo, línea recta o estrella. En terrenos difíciles, sean remotos, rural o urbano, donde no todos los usuarios ven uno o algunos puntos centrales, lo más posible es que el usuario sólo vea a uno o más usuarios vecinos [Acuña Martínez y Roncallo Kelsey, 2006].

Una de las debilidades y limitaciones de las redes Mesh es la latencia (el retardo de propagación de los paquetes), que crece con el número de saltos. Los efectos del retardo son dependientes de la aplicación. Por ejemplo, los correos electrónicos no son afectados por grandes latencias, mientras que los servicios de voz son muy sensibles a los retardos. Otra debilidad es la disminución del rendimiento en todas las redes multisalto, esto es, a mayor número de saltos, se tiene menor rendimiento.

Con respecto al hardware, prácticamente cualquier nodo inalámbrico puede convertirse en un nodo Mesh simplemente mediante modificaciones de software.

## 2.1. Protocolos de Enrutamiento

La principal función de los protocolos de enrutamiento es seleccionar el camino entre el nodo fuente y destino de una manera rápida y fiable. Las redes malladas inalámbricas pueden utilizar los protocolos de enrutamiento de otras redes ya existentes, pero modificándolos para que funcionen correctamente con ellas. Si se elige esta opción, el protocolo de enrutamiento modificado debe asegurar las principales características que son el número de saltos, el rendimiento, la tolerancia a fallos, el equilibrado de carga, la escalabilidad y el soporte adaptativo.

Otra opción es diseñar un nuevo protocolo de enrutamiento para las redes malladas inalámbricas. Esta solución es más costosa ya que cuando se desarrolla un nuevo protocolo hay que probarlo, modificarlo y solucionar los fallos. Por tanto, el tiempo de realización es mayor que si nos centramos en un protocolo ya experimentado.

En este trabajo utilizamos los protocolos OLSR y BMX para el enrutamiento en la red mallada dado que son algunos de los más difundidos en este tipo de redes inalámbricas. A continuación se presenta una breve reseña.

**OLSR:** Optimized Link State Routing Protocol [OLSR] es un protocolo proactivo que se basa en el estado de los enlaces. Se utiliza la técnica MPR (Multipoint Relaying) que consiste en elegir un conjunto de nodos vecinos que cubran el acceso de nodos distantes a dos saltos o más. Se adapta bien en redes con un gran número de nodos y de alta movilidad. El formato del paquete es igual para todos los datos del protocolo, así es fácil la extensión del mismo. Para saber el estado de un enlace se envían mensajes de HELLO. Cada nodo tiene asociado a cada vecino el estado del enlace. Cuando un nodo detecta la aparición de un nuevo vecino se debe incluir una nueva entrada a la tabla de enrutamiento e incluir el estado del enlace. Además, si se detecta una variación en el estado de un enlace, se debe comprobar en la tabla de enrutamiento que el cambio ha sido reflejado. Si no se recibe información de un enlace durante un tiempo determinado se elimina de la tabla de enrutamiento dicho enlace y el vecino correspondiente. Para calcular las rutas, cada nodo contiene una tabla de

enrutamiento con el estado del enlace y el nodo. El estado de un enlace se mantiene gracias al intercambio de mensajes periódicos. La tabla de enrutamiento se actualiza si se detecta algún cambio en el campo de enlace, de vecino, de vecino de dos saltos o en la topología. En [Rodríguez et al., 2013], se presenta un análisis del desempeño de OLSR en una red mallada inalámbrica en un escenario real.

**BMX:** (BATMAN eXperimental), sucesor del protocolo BATMAN (Better Approach to Mobile Adhoc Networks), ([Neumann et al., 2012], [BMX6]) es un protocolo de enrutamiento relativamente reciente y que continúa bajo desarrollo con el objeto de reemplazar a OLSR. La principal innovación que introduce es, como su propio nombre indica, un diseño especialmente enfocado a las redes móviles inalámbricas. Resulta evidente que, cuando hablamos de dispositivos inalámbricos de bajo consumo y baja potencia computacional, como pueden ser los routers mesh en un despliegue urbano, por ejemplo, la utilización de OLSR se hace dificultosa dado que obliga a mantener información sobre la topología completa de una red en constante cambio y, por consiguiente, a encaminar los datos basándose en una tabla de enrutamiento altamente volátil y poco fiable. BMX pretende resolver este aspecto mediante la distribución del conocimiento sobre la topología de la red, de manera que cada nodo almacena y mantiene solamente la información relativa al siguiente salto para cada nodo destino. De este modo, los nodos no almacenan las rutas completas extremo a extremo y los cambios en la topología sólo afectan localmente a los nodos cercanos en lugar de globalmente. Es un protocolo de enrutamiento dinámico y proactivo para redes malladas ad-hoc que utiliza las tablas de enrutamiento para las decisiones de ruteo. Este protocolo no calcula rutas completas entre un nodo origen y el nodo destino sino que selecciona un nodo de salto para utilizarlo como gateway hacia el destino. Encuentra otros nodos y define el mejor vecino para llegar a ellos. Además hace un seguimiento de los nuevos nodos e informa a sus vecinos de su existencia. Es decir, cuando un nodo se incorpora a la red envía un paquete broadcast para avisar de su existencia y este mensaje se va distribuyendo por toda la red. El protocolo mantiene la información sobre la existencia de los nodos mientras sean accesibles. Como no hay necesidad de encontrar o calcular la ruta completa, la implementación es muy rápida y eficiente.

Los paquetes que transfiere BMX son muy pequeños. Cada nodo percibe y mantiene sólo la información sobre el próximo mejor salto hacia todos los otros nodos. De ahí que el conocimiento global acerca de los cambios en la topología local se convierte innecesario. El algoritmo está diseñado para tratar con redes que están basadas en enlaces no confiables y el código fuente es liberado bajo la licencia GPL.

## 2.2. Trabajos relacionados

[Baig Viñas R., 2012] enfoca el estudio del comportamiento de los protocolos Babel, BatManeXperimental versión 6 (BMX6) y Optimized Link State Routing Protocol (OLSR) con protocolo de enrutamiento de Internet versión 6 (IPv6), en términos de protocolo de los gastos generales y el tiempo de convergencia. En una primera etapa, la zona de guifi.net de Barcelona se ha caracterizado en términos de enlaces y nodos relevantes para el enrutamiento de núcleo, utilizando 66 nodos y 69 enlaces. Para este fin, la información estática de la base de datos guifi.net se ha combinado con mediciones obtenidas a través de una red real tomada como referencia. Como resultado de esta medición, se recogió un amplio conjunto de datos nuevos y desconocidos, incluyendo los enlaces en términos de disponibilidad, pérdida de paquetes y de retardo.

Como conclusión de su trabajo, plantea que el empleo de un protocolo de malla como Babel o BMX6 para proyectos parece una alternativa más prometedora a la del protocolo OLSR.

En [Daneels G., 2013] se ofrece una visión general elaborada de diferentes aspectos del protocolo de enrutamiento BMX6. Comprender el protocolo, especialmente el código fuente, resultó ser una tarea difícil dado que no hay mucha documentación disponible. Por lo tanto, el trabajo de Daneels aporta algunos de los conceptos y de los procesos internos del protocolo. El trabajo presenta cuatro diferentes escenarios de experimentación para ejecutar BMX6 y OLSR en función del tamaño de la topología. Debido a algunos problemas y la presión del tiempo, el autor ejecuta cada experimento sólo tres veces y, como consecuencia, no todos los resultados son estables. Aún así, los resultados fueron claros: BMX6 era mejor protocolo en los cuatro escenarios de experimentación que OLSR, a veces con algunas mejoras significativas.

La investigación presentada en [DeCristofaro M. A. et al., 2014], demostró que, dado el estado actual de la tecnología de hoy en día, se pueden implementar redes de malla simples. Aunque existen muchos protocolos, ninguno parece estar significativamente más avanzado en su desarrollo que otros. A los efectos de este estudio, BATMAN-Adv y HWMP fueron escogidos debido a que eran más fáciles de configurar y a que representaban protocolos muy diferentes. La distancia y el rendimiento podrían haber mejorado significativamente por los cambios de la capa física, incluyendo la direccionalidad y la potencia de transmisión, que se suman al tamaño, el peso, la complejidad y el consumo de energía. Sin embargo, ambos protocolos tenían defectos significativos. Una técnica que podría ser evaluada para reducir la pérdida de ancho de banda por salto es cambiar Carrier Sense Multiple Access (CSMA) por Grant to Send (GTS). En conclusión, plantean que la creación de redes de malla inalámbrica puede ser adaptada para soportar muchas aplicaciones espaciales no críticas. Asimismo, apuntan a crear aplicaciones de código abierto más robustas que se puedan utilizar para seguir avanzando en la exploración espacial humana, y mejorar el rendimiento de la red.

[Sajjad y Asad, 2011] comparan el rendimiento de los protocolos AODV, OLSR, DSR y DSDV, en el marco de dos escenarios. En el primer escenario se plantea baja movilidad y tráfico bajo y en el segundo escenario se trata gran movilidad y alto tráfico. Al observar los resultados se encuentran en OLSR y DSDV retardo y jitter bajos, lo cual se requiere principalmente en aplicaciones de voz sobre IP (VoIP) y aplicaciones en tiempo real, tales como juegos de múltiples jugadores en entorno móvil ad hoc. A menor valor de la carga normalizada de enrutamiento, menor será el desperdicio de ancho de banda que se utiliza para el intercambio de mensajes de enrutamiento entre nodos y mayor será el ancho de banda disponible para la transferencia de datos entre los nodos. Las aplicaciones como voz y videoconferencia necesitan más ancho de banda, por lo que en este caso se puede utilizar DSDV. Las aplicaciones como la telefonía de vídeo, juegos en red, etc., requieren un alto rendimiento, por lo que en este caso AODV se puede utilizar en condiciones de poca movilidad y poco tráfico y DSDV se puede usar en condiciones de gran movilidad y alto tráfico. En operaciones de emergencias y militares hay una alta movilidad de los usuarios y de los nodos de red. Para este caso, los autores han observado que a medida que aumenta la movilidad hay una mejora en el rendimiento de OLSR, DSR y DSDV. Así que estos tres protocolos se pueden utilizar en aplicaciones militares de emergencia.

En el trabajo de [Zakaria A. et al., 2013], el análisis de rendimiento de la red de malla inalámbrica se ha considerado en un contexto de entorno rural. El requisito principal es que los nodos sean bastantes estáticos para lograr un rendimiento satisfactorio en términos de alto rendimiento y baja demora. La simulación por ordenador se ha realizado para cuantificar el rendimiento de los diversos protocolos de enrutamiento teniendo en cuenta una serie de parámetros de red, tales como las cargas de tráfico, tamaños de red y el número de fuente. En su trabajo observó que estos parámetros tienen diferentes impactos en métodos de enrutamiento proactivos, reactivos e híbridos. Los resultados simulados indican que HWMP tiene un aparente beneficio frente a WMN, ya que alcanzó el rendimiento promedio más alto en el extremo más bajo de retraso extremo en comparación con AODV y OLSR.

En nuestro trabajo se plantea, a diferencia de los presentados, un escenario que corresponde a una red de mundo real (no de laboratorio) con pruebas reales y no simulaciones. Además, se utilizó hardware variado para realizar una red mallada con dispositivos disponibles en mercado (como los TP-LINK y Ubiquiti). Dados los resultados de nuestra evaluación, BMX6 y OLSR presentaron un comportamiento satisfactorio en términos de rendimiento de convergencia y una mejor escalabilidad en términos de gastos generales para grandes redes con cientos de nodos.

### 3. Escenario y Tecnologías Utilizadas

Se montó una red experimental distribuida en tres edificios del Campus Rosario de la Universidad a los efectos de tener un campo de pruebas más parecido a la realidad de las redes malladas. En la Figura 1 se muestra la distribución del equipamiento en el campus universitario.

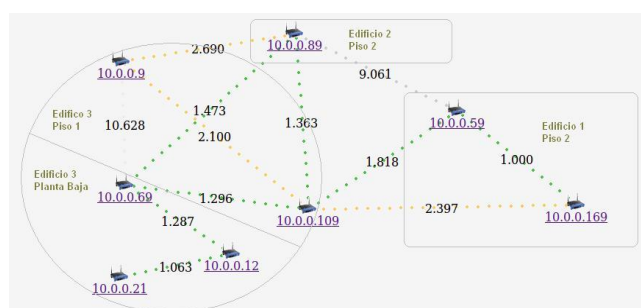


Figura 1. Escenario

En dicha figura se consignan las direcciones IP de los nodos de la red mallada y los números sobre los enlaces corresponden a los valores de la métrica ETX (Expected Transmission Count). El valor de esta métrica es una medida de la cantidad de retransmisiones que se deben hacer para alcanzar un determinado nodo satisfactoriamente. Al armar la red y definir la ubicación de los nodos se tuvo en cuenta el valor de la misma, tratando de que en el camino principal y en al menos un camino secundario este valor no fuera muy superior a 2.

Al momento de montar la red mallada, se realizó un análisis del campo electromagnético en la frecuencia 2.4 GHz. Para esto se utilizó un analizador de frecuencia de Ubiquiti AirView2 ext. Se detectó que el canal 6 no estaba siendo utilizado por la red inalámbrica de infraestructura. A raíz de esto se eligió esta frecuencia para la red mallada pero de todas maneras se debió “vigilar” constantemente

el espectro dado que las redes coexistentes de infraestructura operan en el modo de cambio automático de canal, con lo que la ocupación de un determinado canal puede variar con el tiempo.

En el montaje de esta red se utilizaron equipos de las marcas Ubiquiti (Nanostation 2, Nanostation Loco M2) y TP-Link (TLWR841ND) como se muestra en la Tabla 1. Se eligieron por la gran popularidad y su bajo precio. Cabe aclarar que también se hicieron pruebas con Linksys WRT54GL pero el desempeño de los mismos no fue satisfactorio con OpenWRT Backfire y se observaron problemas en las pruebas con BMX. Tampoco se obtuvo buen desempeño con los TP-Link TLWR741ND y TLWR743ND.

**Tabla 1. Hardware utilizado**

Nro	Nombre	Hardware	SO	Firmware
1	nodo59	Ubiquiti loco M2	OpenWrt	Back fire (10.03.1, r29592)
2	nodo69	Ubiquiti loco M2	OpenWrt	Back fire (10.03.1, r29592)
3	nodo89	Ubiquiti Nanostation2	OpenWrt	Back fire (10.03.1, r29592)
4	nodo09	Ubiquiti Nanostation2	Openwrt	Back fire (10.03.1, r29592)
5	nodo109	Ubiquiti loco M2	OpenWrt	Back fire (10.03.1, r29592)
6	nodo169	TP-LINK TL841ND	OpenWrt	ATTITUDE ADJUSTMENT (12.09, r33312)
7	nodo21	Ubiquiti Nanostation2	OpenWrt	Back fire (10.03.1, r29592)
8	nodo12	TP-LINK TL841ND	OpenWrt	ATTITUDE ADJUSTMENT (12.09, r36088)

Se utilizó como sistema operativo OpenWrt [OpenWRT] que es una distribución de Linux usada para dispositivos embebidos tales como routers personales. El soporte fue limitado originalmente al modelo Linksys WRT54G, pero desde su rápida expansión se ha incluido soporte para otros fabricantes y dispositivos. OpenWrt utiliza principalmente una interfaz de línea de comando, pero también dispone de una interfaz web en constante mejora. El soporte técnico es provisto como en la mayoría de los proyectos de Software Libre, a través de foros y su canal IRC. El desarrollo de OpenWrt fue impulsado inicialmente gracias a la licencia GPL, que obligaba a todos aquellos fabricantes que modificaban y mejoraban el código, a liberar éste y contribuir cada vez más al proyecto en general.

Como se puede ver en la tabla de dispositivos se utilizaron dos versiones de OpenWrt: la BackFire 10.03.1 para los equipos Ubiquiti y la ATTITUDE ADJUSTMENT 12.09 para los equipos TP-Link que eran de las recientes, las más estables al momento de las pruebas, solamente con el agregado del protocolo OLSR (versión 0.6.3-3) o BMX (versión 0.3 r2012011001) según el caso. Cabe aclarar que las versiones de los protocolos son las disponibles para instalación en los repositorios para OpenWRT dado que nuestra propuesta es trabajar con lo que existe para cualquier usuario con conocimientos mínimos pueda configurar estos dispositivos.

#### **4. Pruebas Realizadas**

Utilizando el escenario descrito, se realizaron pruebas para medir la efectividad de los protocolos. En la ejecución de estas pruebas se utilizó el camino formado por los nodos 21, 12, 69, 109 y 59. Las métricas de rendimiento utilizadas son: el tiempo de ida y vuelta (Round-Trip Time - RTT), Jitter, la probabilidad de error y el testeo de ancho de banda. La elección de un camino de cuatro saltos se debe a que no pudimos lograr en el escenario de pruebas, caminos estables de cinco saltos con los equipamientos

disponibles y la disposición que podíamos hacer de los mismos.

RTT: es el tiempo que le lleva a un paquete alcanzar un nodo remoto y regresar. Está relacionado con la latencia de la conexión. Cuanto más bajo es el RTT, mejor es la conexión.

Jitter: es la variación en la latencia de paquetes recibidos de un nodo remoto. Cuanto más bajo es, mejor conexión. Es importante cuando se utilizan aplicaciones de voz sobre IP.

Probabilidad de error: los errores en una red causan que los paquetes se pierdan, se corrompan, se dupliquen o queden fuera de servicio. Cuando ocurre un error es importante saber la probabilidad con la que suceden y el tiempo entre ellos. Lo ideal es no tener errores, pero una tasa baja es aceptable.

Ancho de banda: es la tasa de transmisión de un enlace o sistema de transporte de datos y se puede definir como la capacidad de un enlace o sistema para transmitir datos. Se expresa en bit por segundo.

Nuestra herramienta principal de testeo fue Iperf para todas las métricas a excepción de RTT, que se midió con ping. Iperf es una herramienta que se utiliza para hacer pruebas en redes informáticas. El funcionamiento habitual es crear flujos de datos TCP y UDP y medir el rendimiento de la red. Típicamente la salida de Iperf contiene un informe con marcas de tiempo con la cantidad de datos transmitidos y el rendimiento medido.

Se utilizó iperf con los siguientes parámetros:

```
iperf -c 10.0.0.69 -u -b 20M -t 60 -i 5
```

donde

-u indica protocolo udp

-b especifica el ancho de banda máximo

-t tiempo en segundos que dura la transmisión de paquetes

-i intervalo en segundos para el envío de reportes de ancho de banda

Iperf arroja al final del tiempo un promedio de todos los valores obtenidos cada i segundos.

Para medir el valor de RTT se utilizó la herramienta ping. Ping es el acrónimo de Packet Internet Groper, que significa “Buscador o rastreador de paquetes en redes”. Es un utilitario que analiza el estado de la comunicación entre un host local y uno o varios remotos por medio del envío de paquetes. Se utiliza para diagnosticar el estado, velocidad y calidad de una red determinada. En nuestras pruebas se utilizó el siguiente comando:

```
ping 10.0.0.59 -s 1016 -c 50
```

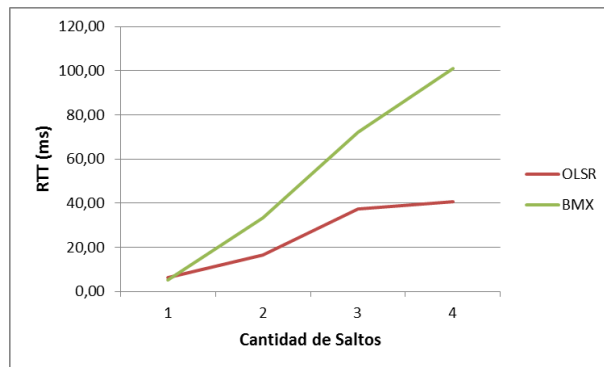
donde 1016 indica el tamaño del paquete transmitido (en bytes) y 50 la cantidad de ping enviados, al final el comando arroja los valores mínimos y máximos obtenidos y el promedio.

Respecto del horario de las pruebas las mismas fueron realizadas a la mitad de la mañana o a la mitad de la tarde por ser estos los horarios donde la utilización de la demás redes inalámbricas existentes en el campus es moderada evitando así una excesiva influencia de las mismas sobre nuestra red mallada.

La Figura 2 muestra los resultados de RTT. Los resultados muestran un patrón donde RTT se incrementa con el número de saltos. Lo cual es lógico porque se tiene un camino mayor por recorrer. Por otro lado se puede observar un mayor valor para BMX que para OLSR que en nuestra opinión se debe a que BMX no tiene el camino

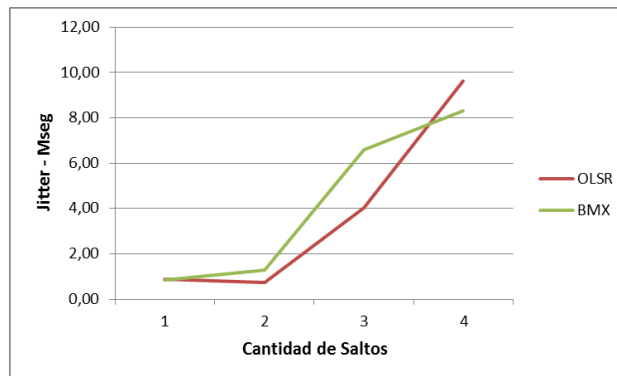


determinado a priori y debe definirlo a medida que va avanzando en la ruta.



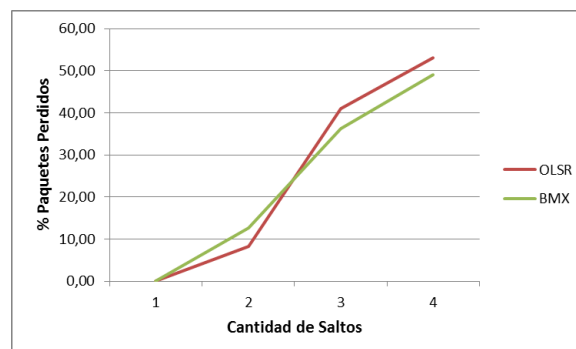
**Figura 2. Evaluación de Round-trip Time**

La Figura 3 muestra los resultados de la variación del retardo (jitter) obtenidos.



**Figura 3. Evaluación de Jitter**

A medida que aumenta el número de saltos aumenta el valor de la variación del retardo y el aumento se torna significativo para 3 y 4 saltos aunque su valor no es muy alto en función de los valores de retardo para el caso de BMX pero sí para OLSR. Consideramos que los valores altos obtenidos se deben a que nos estamos manejando en un escenario real en el que las condiciones de interferencia de otros dispositivos y de obstáculos pueden variar significativamente de un instante a otro.

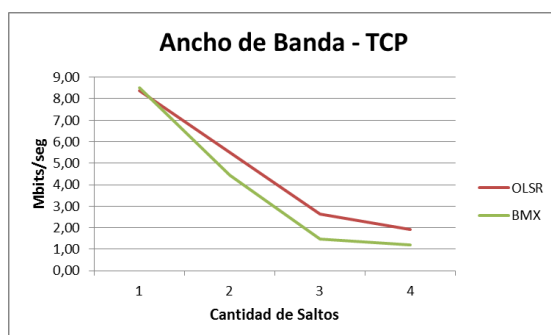


**Figura 4. Evaluación de la Probabilidad de Error**

La Figura 4 muestra los resultados de Probabilidad de Error. Se observa que la

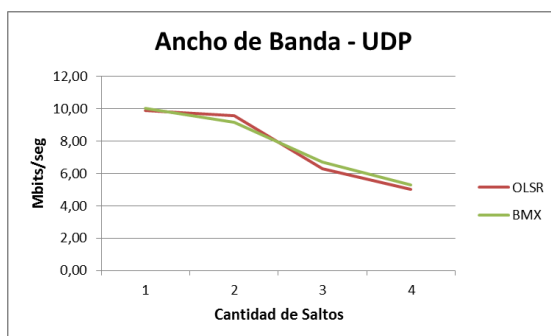
pérdida de paquetes crece con el número de saltos y se vuelve significativa a partir de 3 saltos. Aquí remarcamos nuevamente la influencia de tratarse de un escenario real.

La Figura 5 muestra los resultados obtenidos de las pruebas con TCP. En ambos casos el comportamiento es similar. El ancho de banda sufre un decrecimiento a medida que se incrementa el número de saltos, esto se debe a que los dispositivos inalámbricos no transmiten y reciben al mismo tiempo, realizando sólo una de estas acciones en un determinado instante de tiempo, lo que hace que el ancho de banda se divida por dos en cada salto. Las diferencias entre ambos protocolos las atribuimos al hecho que BMX necesita determinar el camino a medida que avanza, lo que hace que disminuya el ancho de banda, cosa que no ocurre con OLSR, que tiene los caminos predeterminados.



**Figura 5. Evaluación del ancho de banda con TCP**

La Figura 6 muestra los resultados obtenidos de las pruebas con UDP. En ambos casos el comportamiento es similar. El ancho de banda sufre un decrecimiento a medida que se incrementa el número de saltos.



**Figura 6. Evaluación del ancho de banda con UDP**

Las diferencias que se observan entre TCP y UDP se deben a las características propias del protocolo. TCP implementa más controles que UDP por lo que sus retransmisiones disminuyen el ancho de banda.

Además, utilizando el mismo escenario, se realizaron pruebas para medir el tiempo que tarda el protocolo en entrar en funcionamiento cuando se incorpora un nuevo nodo a la red.

Para medir el tiempo de arranque, se apagó un nodo y luego se volvió a arrancar registrando la hora de encendido. Este procedimiento se realizó ejecutando un script para evaluar cuantos segundos demoraba en re arrancar. Se tomó la hora de la ejecución del primer ping inalcanzable y luego la hora del primer 0% paquetes perdidos.

El promedio obtenido con el Protocolo OLSR fue de 30,63 segundos y el

promedio obtenido con el Protocolo BMX fue de 10,93 segundos. Dichos resultados son compatibles con las características de los protocolos dado que OLSR construye en todos los nodos un mapa de toda la red lo cual le insume más tiempo que a BMX que determina el camino a medida que va avanzando en la red.

## 5. Conclusiones

En este trabajo se presentó un estudio comparativo sobre el rendimiento de los protocolos OLSR y BMX. Se presentaron los resultados empíricos de varias pruebas utilizando el mismo escenario. El escenario que se presenta es una red de mundo real (no de laboratorio) con pruebas reales y no simulaciones. Cuando se trata con este tipo de entornos, los experimentos son cada vez más difíciles de repetir en forma exacta al anterior.

Por lo observado se confirma que el rendimiento de la red decrece con el número de saltos. Su valor, que en nuestro caso es bajo para cuatro saltos, depende mucho de la ubicación y la conectividad entre dispositivos, como así también de la actividad radioeléctrica circundante y de los obstáculos. No se observan diferencias significativas entre OLSR y BMX para el caso del rendimiento.

Si analizamos el retardo además de ser significativo en valores sobre todo para cuatro saltos, hecho que atribuimos al escenario real de pruebas, hay diferencia entre los dos protocolos y lo adjudicamos, como ya comentamos, a que BMX no tiene sus rutas determinadas a priori y debe efectuar procesamiento extra a medida que avanza por el camino.

A la luz de los resultados hay algunos descubrimientos interesantes. OpenWrt es un software perfectamente válido que puede ser utilizado en una gran variedad de dispositivos y su configuración para utilizarlo con protocolo OLSR es sencilla de realizar y no presenta problemas de funcionamiento con dicho protocolo.

Ex-profeso se utilizó hardware variado para demostrar que se puede realizar una red mallada con los dispositivos disponibles en el mercado (como los TP-LINK y Ubiquiti). De todas maneras en el relevamiento de redes existentes realizado y en el grado de desarrollo de los firmwares, se pudo observar que los equipos más utilizados son las distintas versiones de Nanostation de la marca Ubiquiti.

Cabe aclarar que por tratarse de una red montada sobre un escenario real y no de laboratorio hemos tenido que escoger adecuadamente los horarios de realización de las pruebas dado que las otras redes inalámbricas instaladas en el edificio y la circulación de personas tienen una marcada influencia en el funcionamiento de la red mallada.

## 6. Referencias

- Acuña Martínez, D., Roncallo Kelsey, R. (2006) Redes inalámbricas enmalladas metropolitanas. p. 46--91.
- Akyildiz, I., Wang, X., Wang, W. (2005) "Wireless mesh networks: a survey". In: Computer Networks. Vol. 47. No.4 p. 455--487.
- Baig Viñas R. (2012) Evaluation of Dynamic Routing Protocols on Realistic Wireless Topologies. Master thesis. Universidad Autónoma de Barcelona.
- BMX6. Disponible en: <http://bmx6.net/projects/bmx6> Consultado el 01/10/2013.
- Daneels G. (2013) Analysis of the BMX6 routing protocol. Master thesis. University of

Antwerp.

- DeCristofaro M. A., Lansdowne C. A., and Schlesinger A. M. (2014) Heterogeneous Wireless Mesh Network Technology Evaluation for Space Proximity and Surface Applications. DOI, 10:6–2014.
- Neumann, A, López E, Navarro L. (2012) “An evaluation of BMX6 for Community Wireless Networks”. International Workshop on Community Networks and Bottom-up-Broadband (CNBuB 2012).
- OLSR. Disponible en: <http://www.olsr.org/> Consultado el 01/05/2014.
- OpenWRT. Disponible en: <https://openwrt.org/> Consultado el 08/03/2014.
- Rodríguez E., Deco C., Burzacca L., Petinari M. (2013) “Estudio del desempeño de OLSR en una red mallada inalámbrica en un escenario real”. En Anales del Workshop Arquitectura, Redes y Sistemas Operativos (WARSO), XIX Congreso Argentino de Ciencias de Mar del Plata. Argentina, Octubre 2013. p. 1066-1074.
- Rodríguez E., Deco C., Burzacca L., Petinari M. (2012) “Protocolos de enrutamiento para redes malladas inalámbricas: Un estudio comparativo”. En Anales del VII Workshop Arquitectura, Redes y Sistemas Operativos (WARSO), XVIII Congreso Argentino de Ciencias de la Computación. Universidad Nacional del Sur Bahía Blanca. Argentina, Octubre 2012. ISBN 978-987-1648-34-4 pp 1128-1137.
- Sajjad Ali, Asad Ali. (2011) Performance Analysis of AODV, DSR and OLSR in MANET. LAP Lambert Academic Publishing (17 July 2011).
- Zakaria A., Mohamad H., Ramli N., and Ismail M. (2013) “Performance evaluation of routing protocols in wireless mesh network”. In Proc. 2013 15th International Conference on Advanced Communication Technology (ICACT), Korea, pp.1111-1115.