

# DESEMPEÑO DE LOS PROTOCOLOS DE ENRUTAMIENTO MESH

## PERFORMANCE MESH ROUTING PROTOCOLOS

### RESUMEN

En este artículo se presentan los resultados obtenidos de la evaluación del desempeño de los principales protocolos que se podrían utilizar en enrutamiento malla, considerando tasas de entrega de paquetes, retardos y carga de enrutamiento promedio. La investigación estableció que el algoritmo de enrutamiento de fuente dinámica es el que posee mejor rendimiento debido a su naturaleza híbrida.

**Palabras clave:** enrutamiento, NS-2, protocolo, red Ad-hoc, red malla.

### ABSTRACT

This article presents the results obtained of performance evaluation of the main protocols that could be used in mesh routing considering packet delivery rate, delay and routing load average. The investigation established that the algorithm Dynamic Source Routing is the one that has better performance due to its hybrid nature.

**Key words:** network Ad-hoc, network mesh, NS-2, protocol, routing.

### Danilo Alfonso López Sarmiento

Magister en Ciencias de la Información y las Comunicaciones  
Docente planta de la Universidad Distrital Francisco José de Caldas  
dalopezs@udistrital.edu.co  
Bogotá, Colombia

### Octavio José Salcedo Parra

Magister en Ciencias de la Información y las Comunicaciones  
Docente planta de la Universidad Distrital Francisco José de Caldas  
osalcedo@udistrital.edu.co  
Bogotá, Colombia

### Edwin Rivas Trujillo

Doctor en Ingeniería Eléctrica, Electrónica y Automática  
Docente planta de la Universidad Distrital Francisco José de Caldas  
erivas@udistrital.edu.co  
Bogotá, Colombia

**Tipo:** Artículo reporte de caso

**Fecha de Recepción:** Octubre 17 de 2013

**Fecha de Aceptación:** Diciembre 22 de 2013

## 1. INTRODUCCIÓN

Establece que una estructura ad-hoc es aquel tipo de red inalámbrica en la que cada nodo podría actuar como un router independiente [1], sin tener en cuenta si está conectado a otra red o no, ya que aunque no lo estuviese, en determinado momento podría llegar a estarlo. De otro lado las redes mesh son aquellas networks ad-hoc que poseen un backbone de nodos casi estacionarios o con poca movilidad como lo describe [2]. Dado la dinámica de este tipo de redes (topológicamente hablando) y a las características poco estables de los medios de transmisión inalámbricos, el enrutamiento mesh es un campo de investigación abierto a generar posibles mejoras a protocolos existentes, desarrollo de nuevos protocolos y determinación de evaluaciones de desempeño según se muestra en [3]. Específicamente el presente paper evalúa el comportamiento de los protocolos de routing proactivos (que son aquellos mediante los cuales se intenta obtener una representación fiel de la topología de la red en determinado momento mediante el envío periódico de información tal y como lo define [4], los protocolos reactivos (que por el contrario, tan solo intentan determinar una ruta cuando se ha solicitado el envío de información por parte de un nodo fuente a determinado nodo destino, según [4], y los protocolos híbridos que son una combinación de los dos paradigmas anteriormente mencionados.

El uso de procesos proactivos podrían llegar a ser útiles en escenarios como el de las redes mesh y ad-hoc, dada la dinámica que proporciona el movimiento de los nodos; sin embargo, la actualización de las posibles rutas en cada nodo está sujeta a un alto costo en términos de velocidad y throughput en la red, ya que se “envían” periódicamente actualizaciones con destino Broadcast, así no existan cambios en la topología de la red en razón a que las rutas almacenadas en las tablas de routing vencen cada determinado tiempo. Es por esta razón, que en muchas aplicaciones para las cuales se requiere manejo de altas velocidades de transmisión, se hace uso de protocolos reactivos. Sin embargo para redes ad-hoc ésta podría no ser una buena

opción si los nodos se mueven con velocidades elevadas en intervalos de pausa muy cortos.

Es por esto que el objetivo del presente trabajo es la evaluación del desempeño de protocolos que están basados en paradigmas distintos; representando a los protocolos proactivos: DSDV, a los protocolos reactivos: AODV y a los protocolos híbridos: DSR.

## 2. METODOLOGÍA

La forma en que se exponen los resultados encontrados se divide en varias secciones. Específicamente se parte de las características principales de los procesos de routing AODV, DSR y DSDV, a fin de ofrecer una base teórica que soporte la investigación desarrollada, para posteriormente definir la topología de red utilizada incluyendo las características de la simulación más representativas; seguidamente se muestran los resultados obtenidos que permitirán generar un análisis a partir de los datos obtenidos. Finalmente se encuentra la sección de conclusiones.

## 3. PROTOCOLOS DE ENRUTAMIENTO

### 3.1. Ad-hoc DSDV

El protocolo Destinación ordenada del vector distancia) es esencialmente una modificación del algoritmo de encaminamiento Vector Distancia de Bellman-Ford empleado por su utilidad en redes fijas; como por ejemplo en RIP (protocolo de información de ruta). En este algoritmo, los nodos vecinos intercambian periódicamente (protocolo proactivo) sus tablas de enrutamiento completas con los vecinos para estimar la distancia a la que se encuentran los demás nodos no vecinos. Las modificaciones introducidas por DSDV proporcionan esencialmente la obtención de rutas sin bucles mediante la introducción de números de secuencia para la determinación de las rutas más actuales. Aunque DSDV sólo proporciona un camino para cada destino (no permite almacenar rutas secundarias o de soporte), siempre elige el camino más corto basándose en el número de saltos hacia este destino. DSDV utiliza dos tipos de

mensajes de actualización, uno más grande en tamaño (full-dump) y otro mucho más pequeño (incremental). Los mensajes incrementales pueden utilizarse para actualizaciones intermedias entre envíos periódicos (full-dump) de la tabla entera de enrutamiento.

### 3.2. Ad-hoc AODV

En el protocolo AODV (Ad-hoc Sobre Demanda de Vector Distancia) los nodos mantienen una tabla de enrutamiento para los destinos conocidos (empleando el algoritmo vector distancia). Inicialmente esta tabla estará formada por sus vecinos. Solamente se le añadirán destinos nuevos cuando sea necesario, es decir, cuando un nodo necesita comunicarse con otro que no está en su tabla inicia un proceso de descubrimiento de ruta (reactivo) hacia el destino concreto. Para ello se emiten mensajes de descubrimiento de ruta "Request" que se van propagando entre todos los nodos de modo similar al DSR. La diferencia es que aquí los nodos generan una tabla de encaminamiento inversa para que puedan regresar las respuestas "Reply" a las solicitudes de ruta al nodo que la originó. Se recomienda el uso de mensajes "Hello" entre vecinos para determinar la conectividad, aunque para reducir el volumen de estos mensajes, sólo debe permitirse su envío a los DCE (equipo de conmutación de datos) que estén transmitiendo datos. Tal y como aparece en [5] AODV permite dar soporte para transmisiones multicast además de permitir la utilización de técnicas de "búsqueda secuencial por anillos" y "reparación local del enlace".

### 3.3. Ad-hoc DSR

El protocolo DSR (Enrutamiento de Fuente Dinámico) se fundamenta en el encaminamiento desde el origen, es decir, los paquetes de datos incluyen una cabecera de información acerca de los nodos exactos que deben atravesar. No requiere ningún tipo de mensajes periódicos (reactivo), disminuyendo así la sobrecarga y desperdicio de ancho de banda a nivel de mensajes de control. Además ofrece la posibilidad de obtener, con la solicitud de una ruta, múltiples caminos posibles hacia el destino siempre

y cuando se garanticen condiciones de factibilidad. Para poder realizar el encaminamiento en el origen, a cada paquete de datos se le inserta una cabecera DSR de opciones que se colocará entre la cabecera de la capa de transporte y el datagrama IP cuya finalidad es incluir la ruta que debe seguir el paquete nodo a nodo. Cada DCE mantiene una memoria caché de rutas en la que se van almacenando las rutas obtenidas a través de procesos de descubrimiento de rutas ya sean propios u obtenidos a través de detecciones de envíos en la red. En los procesos de descubrimiento de nuevos caminos a destinos se generan mensajes de solicitud, respuesta y error siendo estos mensajes "Route", "Request", "Reply" y "Error" respectivamente de acuerdo a lo estipulado en [6].

## 4. RESULTADOS Y DISCUSIÓN

A continuación se describen los procesos de simulación llevados a cabo, incluyendo en primera instancia una breve descripción del simulador utilizado NS-2.

### 4.1. Network simulator (NS-2)

[7] definen a NS-2 como un simulador de eventos discretos, cuya elaboración se inició en 1989 con el desarrollo de REAL Network Simulator. Probablemente una de las principales razones que explican su éxito es el hecho de que la distribución posee licencia GPL, condición que impulsa el desarrollo libre del mismo, además de permitir la implementación de cualquier estructura de red mediante el diseño e implementación del código, condición que es atractiva para el desarrollo de tesis de maestría y doctorales. Inicialmente, NS-2 fue ideado para redes fijas, sin embargo, se han desarrollado ampliaciones para el análisis de redes inalámbricas donde se incluyen las principales propuestas de redes Ad-hoc así como de redes WLAN (Redes de área local inalámbricas).

Específicamente su arquitectura se apoya en dos lenguajes de programación para su correcto funcionamiento. Por un lado, el usuario introduce las especificaciones del escenario que desea analizar a través del lenguaje OTcl

(Herramienta de lenguaje de comandos Orientado a Objetos), versión extendida de Tcl (Herramienta de lenguaje de comandos). Por otro lado, la implementación de los protocolos se encuentra en el software C++. Como resultado de la simulación, se pueden obtener datos matemáticos para un estudio posterior o bien trazas específicas para visualizarlas en las herramientas Nam y Xgraph del NS y poder generar comparaciones. Con el fin de poder modelar cada tecnología (DSDV, AODV y DSR) y poder evaluar sus ventajas y/o desventajas fue necesario la implementación de agentes en NS-2, que permitieran simular la topología física sobre la que se trabajó. A manera de soporte y referencia se incluye en las figura 1 y figura 2 parte del programa que se desarrolló para poder establecer claramente las características de DSDV y AODV únicamente.

```
#define cmu_dsdv_h_
#include "config.h"
#include "agent.h"
#include "ip.h"
#include "delay.h"
#include "queue.h"
#include "trace.h"

class DSDV_establecimiento;
class DSDVdanilo;

class DSDV_Agente : public Agente {
friend class DSDV_Helper;
friend class DSDVdanilo;
public:
DSDV_Agente();
virtual int command(int argc, const char * const * argv);
void lost_link(Paquete *p);
...
char *subred;
NodoMovil *Nodo;
char *direccion;
NsObject *puerto_dmx;
Evento *periodico_callback;
...
};

class DSDV_establecimiento : public Handler {
public:
DSDV_Helper(DSDV_Agente *a_) { a = a_; }
virtual void handle(Evento *e) { a->establecimiento(e); }
private:
DSDV_Agente *a;
};
```

Figura 1. Generación del agente DSDV

```
Agente/rtProto/AODV set spuerto_ 0
Agente/rtProto/AODV set dpuerto_ 0
-----
proc crea-agente-routing { node id } {
global ns_ ragente_ tracefd opt
.
set danilo_($id) [new sopt(danilo) $id]
set danilo_($id)
$node adicionar $ragente 255
.
$ragente if-cola [$node set ifq_($id)].
$ns_ at 0.0 $id "$ragente_($id) ini cia".
.
}

proc crea-nodo-movil { id } {
global ns_ chan prop topo tracefd opt no
global chan prop traza topo opt
.
set node_($id) [new nodomovil]
.
set node_($id)
$node aleatorio 0
$node lopez $lo
.
if { $opt(pos) == "Box" } {
set espacio 200
set maxrow 3
set col [expr ($id - 1) % $maxrow]
set row [expr ($id - 1) / $maxrow]
$node set X_ [expr $col * $espacio]
$node set Y_ [expr $row * $espacio]
$node set Z_ 0.0
$node set speed 0.0
.
}
```

Figura 2. Generación del agente DSDV

## 4.2. Escenario de la simulación

La red está compuesta por cuatro dispositivos, que se comunican utilizando el protocolo de transporte TCP. La capacidad de los enlaces es de 4 Mb, con retardos de máximo 10 ms y políticas de procesamiento tipo CBQ. Los router cero (0) y tres (3) están comunicados a través de un enlace guiado, mientras que los demás nodos están interconectados de forma inalámbrica como se observa en la figura 3. La simulación tiene un tiempo de duración de 150 segundos.

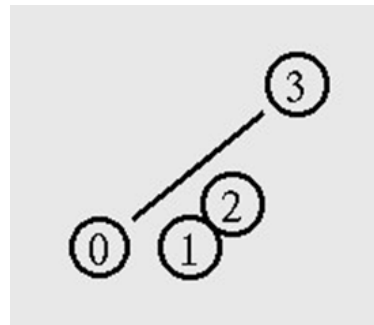


Figura 3. Conexión de nodos

Una vez arranca la simulación el programa desarrollado en TCL permite que los nodos empiecen a desplazarse a partir de los 10 segundos. El nodo cero (0) y uno (1) empiezan a intercambiar información y a enviar mensajes de “saludo” y/o “solicitud” que van a depender del protocolo de enrutamiento usado, tal y como se muestra en la figura 4. De allí también se concluye que inicialmente que la comunicación entre éste par de routers es directa, pero luego de que la distancia incrementa lo que obliga a usar caminos alternos para continuar con la transmisión (figura 5).

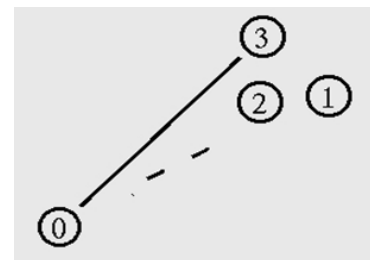
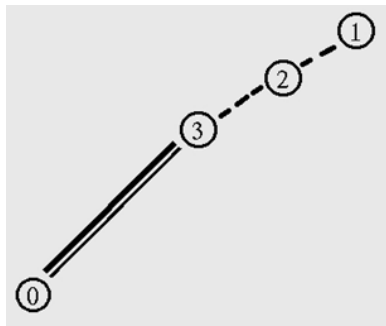


Figura 4. Desplazamiento de nodos (la comunicación entre el nodo “0” y “1” es directa)



**Figura 5.** Desplazamiento de nodos (la comunicación entre el nodo "0" y "1" se realiza a través de "3" y "2")

Con el fin de poder generar gráficos estadísticos y poder analizar el comportamiento generado por cada técnica, se partió de las ecuaciones (1), (2) y (3).

$$\text{Carga promedio de routing} = \frac{\text{Cantidad total de paquetes de enrutamiento}}{\text{Tiempo de simulación}}$$

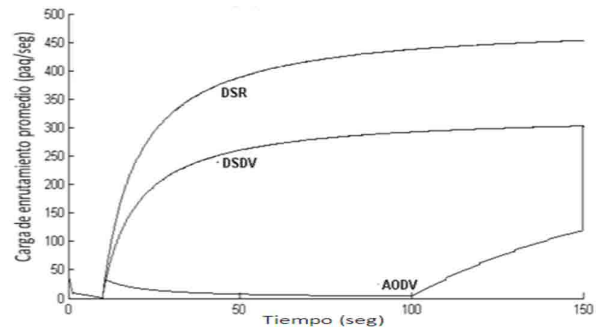
$$\text{Retardo promedio} = \sum_k \frac{\text{Tiempo envío paquete} - \text{Tiempo recepción paquete}}{\text{Total de paquetes transmitidos}} \quad (1)$$

En donde k representa el k-ésimo paquete enviado.

$$\text{Tasa entrega paquetes} = \frac{\text{Total de paquetes recibidos exitosamente}}{\text{Total de paquetes enviados}} \quad (2)$$

La figura 6, muestra la relación de la carga promedio de enrutamiento (paq/seg) vs tiempo (seg) de simulación para cada caso. Lo que se busca con ello es poder determinar el nivel de carga exigido (paquetes/seg) para el descubrimiento de dispositivos vecinos y así establecer cual consume mayor ancho de banda en el mantenimiento de la red. Se encuentra que DSR es el que mayor tasa de envío de mensajes (los cuales no corresponden a datos de usuario) de enrutamiento genera. Esta curva creciente comienza a formarse desde los 10 segundos, que es el tiempo dado en la simulación para que la red empiece a transmitir información de usuario, y su comportamiento es dado ya que DSR a pesar de poseer características híbridas, envía mensajes de actualización innecesariamente aunque ya exista una ruta establecida que no permitirá que los dos nodos que se están comu-

nicando se pierden de vista.



**Figura 6.** Carga promedio de enrutamiento

De otro lado el protocolo DSDV también presenta un comportamiento exponencial creciente en el envío de mensajes de routing. Ello implica que aunque exige menor ancho de banda que el utilizado por DSR la demanda de recursos sigue siendo relativamente alta. Tal conducta se aprecia por las características proactivas del encaminamiento de fuente dinámica que envía y recibe periódicamente actualizaciones y solicitudes para poder tener actualizadas las tablas y así proporcionar una rápida respuesta ante las solicitudes de ruta; no obstante esto redundante en una lenta convergencia incrementándose la probabilidad de que aparezcan loops en la medida que la red se haga escalable. Además se observa que antes de los 10 segundos DSDV muestra actividad ya que constantemente está buscando sus vecinos para adaptarse a cambios que puedan aparecer en la topología.

No obstante el protocolo AODV tiene el distintivo de ser "reactivo bajo demanda", lo que se traduce en que solo actualiza su base de datos de envío cuando aparecen eventos inesperados en la red (rutas más óptimas a destinos, congestión en nodos o enlaces, etc.). Esta es la razón de la curva exponencial negativa observada en la figura 6, entre los 10 y 100 segundos en la que se muestra actividad mientras se determina la forma más óptima de alcanzar nodos contiguos y remotos a partir de las métricas utilizadas (confiabilidad, unidad de transferencia máxima, ancho de banda, entre otros), así que para este caso en particular, en los primeros instantes de tiempo se producen mensajes que van disminuyendo hasta llegar a los 100

segundos, momento en el que se produce un cambio abrupto producto de la caída del enlace. La figura 7, relaciona el retardo promedio de paquetes vs tiempo. Es de notar que los tres tienen un comportamiento similar de tal forma que entre los 10 y 11 segundos (que es cuando inicia la comunicación), el retardo es máximo (hasta valores arriba de 2,5 ms), luego de este lapso la variable comienza a disminuir de manera considerable para los tres protocolos, teniendo un período de estabilización entre los 11 y 12 segundos que evidencia que DSDV y AODV oscilan entre valores que van desde 0,5 y 1,0 ms respectivamente.

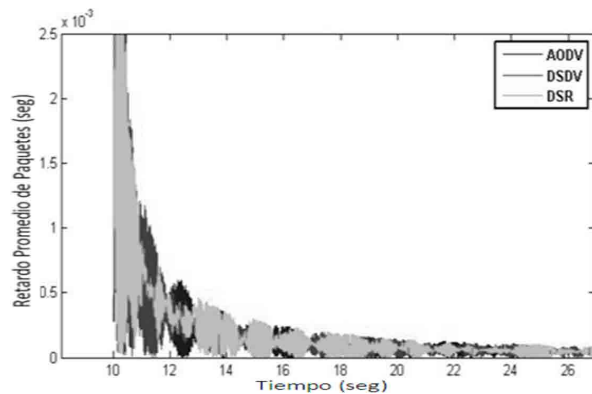


Figura 7. Retardo promedio

Es importante resaltar que el retardo del protocolo DSR es el que más predomina, esto en gran parte gracias a que la carga de routing es alta. Este comportamiento casi oscilatorio no es deseable porque producen características altamente no lineales en la topología. El retardo del protocolo AODV si bien es mayor en un comienzo y tarda más tiempo en estabilizarse que las demás tecnologías es bastante aceptable teniendo en cuenta su propiedad reactiva. Tal evento encuentra explicación en la topología de la red utilizada para la simulación (sin nodos interferentes); muy probablemente el retardo para AODV sería mucho mayor a los demás si se implementará una topología de red más compleja. El retardo en DSDV es relativamente alto por el envío de mensajes de reconocimiento sin ser necesario, desperdiciando así recursos que se podrían utilizar para una entrega más

eficiente al cliente. Otro factor importante es el relacionado con la tasa de entrega de paquetes (figura 8).

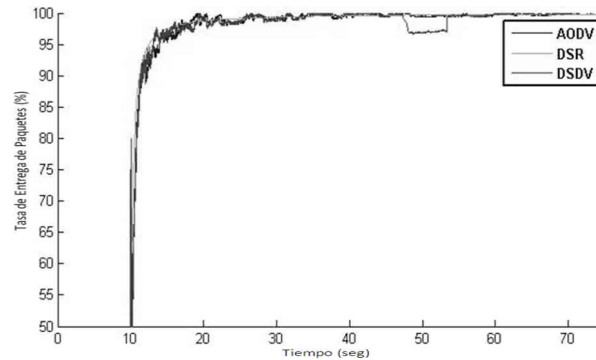


Figura 8. Tasa de entrega de paquetes

Allí se observa la eficiencia en el transporte de los datos entre el nodo "0" y nodo "1". Puntualmente se concluye que el protocolo DSR es el que mejor desempeño tiene, ya que alcanza el 100% de eficiencia de manera rápida y sin variaciones, en razón a que se establece la ruta a seguir para la transmisión desde el inicio y en la cabecera de los mensajes incluye el camino a seguir, de otro lado tiene múltiples rutas secundarias que entran en funcionamiento cuando la principal deja de funcionar; mientras que el protocolo que más inconvenientes presenta es el DSDV por la cantidad de variaciones que posee, lo que redundaría en una tasa de entrega no eficiente, presentando deficiencias cuando los nodos directos pierden la conexión y es necesario establecerse otro enlace para comunicarse. Por último con AODV se presenta un mejor desempeño y es más eficiente, sin embargo se demora un tiempo en estabilizarse y alcanzar el 100%, ya que debe establecer una nueva ruta cuando los nodos dejan de comunicarse directamente.

Finalmente la figura 9, referencia el throughput encontrado en cada caso donde claramente se deduce que DSR posee un mejor comportamiento en el tiempo, lo que indica que se puede obtener un mayor aprovechamiento del ancho de banda disponible en el canal de comunicaciones para el transporte de los datos.

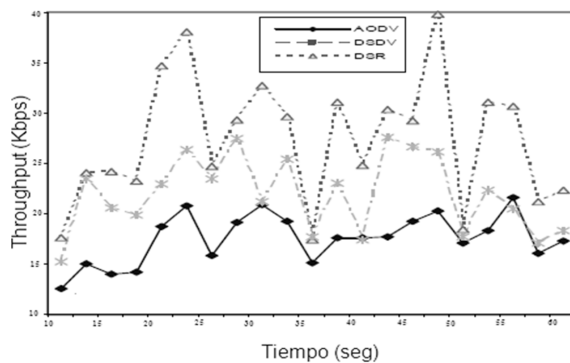


Figura 9. Rendimiento de la topología física con cada tipo de routing

## 5. CONCLUSIONES

A partir de las simulaciones y el análisis de resultados se establece que el protocolo más eficiente en relación a la tasa de entrega de paquetes es DSR como consecuencia a su naturaleza híbrida que permite establecer varias rutas a la vez permitiendo una reacción más eficiente ante fallos o saturación, además de ser robusto frente al hecho de que los nodos sean móviles.

DSR se soporta en un mecanismo de routing desde el origen, lo que le permite a los paquetes adicionar una cabecera donde se establecen los nodos exactos por donde deben circular

para llegar a su destino. No requiere de ningún mensaje periódico reduciendo así la sobrecarga de control dentro de la red.

En AODV los nodos intermedios almacenan en cache una base de datos de encaminamiento para destinos conocidos. Inicialmente esta tabla la componen solo las redes directamente conectadas y solamente se irán añadiendo nuevas entradas a destinos cuando sea necesario establecer comunicación con destinos inexistentes a través del llamado de un proceso de descubrimiento que va propagándose de nodo en nodo hasta el receptor en concreto similar al ocurrido en DSR.

Desde la perspectiva de la cantidad de difusiones AODV es más óptimo que DSDV, debido a que el proceso utilizado por "Ad-hoc sobre demanda de Vector Distancia" está basado en un mecanismo por demanda contrario a lo que sucede por la tecnología "Destinación Ordenada del vector de distancia"

DSR y AODV, son ambos procesos reactivos, esto implica que inician los procesos de routing bajo demanda; no obstante "Ad-hoc sobre demanda de Vector Distancia" usa routing que depende del número de secuencias en los nodos destino.

## Referencias Bibliográficas

- [1] [1] C. K. Toh; Ad Hoc mobile wireless Networks, Prentice Hall Publishers , 27-28, Michigan, USA, 2002.
- [2] [2] I. Akyildiz, W. Xudong, W. Weilin; wireless mesh networks: a survey. [En línea], consultado en Octubre 20 de 2013, disponible en: <http://www.sciencedirect.com>.
- [3] [3] I. Akyildiz, I. Xudong; Wireless mesh networks, John Wiley & Sons Ltd, 194-197, Chichester, UK, 2010.
- [4] [4] M. Elias, P. Campista, I. Moraes, M. Enrique, O. Costa, C. Passos, D. Albuquerque, G. Saade; Routing Metrics and Protocols for Wireless Mesh Networks, IEEE Network, 137-142, Boston, USA, 2008.
- [5] [5] C. Perkins, E. Belding, S. Royer; RFC 3561: Ad hoc On-Demand Distance Vector (AODV) Routing. [En línea], consultado en Junio 15 de 2013, disponible en: <http://www.faqs.org/rfcs/rfc3561.html>.
- [6] [6] D. Jonson, D. Maltz; DSR: the dynamic source routing protocol for multihop wireless Ad hoc networks. [En línea], consultado en Septiembre 18 de 2013, disponible en: <http://www.monarch.cs.rice.edu/monarch-papers/dsr-chapter00.pdf>.
- [7] [7] K. Fall, V. Varadhan; ns notes and documentation, The VINT Project. UC Berkeley. [En línea], Consultado en Octubre 1 de 2013, disponible en: <http://www.isi.edu/nsnam/ns/>.