UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA

UNIDAD ENSENADA

Soporte de Calidad de Servicio (QoS) para Redes Móviles Ad-Hoc (MANETs) con Enrutamiento Proactivo, Utilizando el Protocolo HOLSR

TESIS

Que para obtener el grado de MAESTRO EN INGENIERÍA presenta:

ELBA ABIGAIL MORALES VANEGAS

Aprobada por:

Dr. Luis Armando Villaseñor González

Director del Comité

M.C. Humberto Cervantes de Ávila

Miembro del Comité

Dr. Juan Iván Nieto Hipólito

Miembro del Comité

Ensenada, Baja California, Enero de 2007

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA ENSENADA
MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA

Soporte de Calidad de Servicio (QoS) para Redes Móviles Ad-Hoc (MANETs) con Enrutamiento Proactivo, Utilizando el Protocolo HOLSR

TESIS

que para obtener el grado de

MAESTRO EN INGENIERÍA

Presenta:

ELBA ABIGAIL MORALES VANEGAS

Ensenada, Baja California, México. Enero de 2007.

RESUMEN de la tesis de **Elba Abigail Morales Vanegas**, presentada para obtener el grado de **MAESTRO EN INGENIERÍA**. Ensenada, Baja California, México. Enero de 2007.

Soporte de Calidad de Servicio (QoS) para Redes Móviles Ad-Hoc (MANETs) con Enrutamiento Proactivo, Utilizando el Protocolo HOLSR

Resumen aprobado por:

Dr. Luis Armando Villaseñor González
Director de Tesis

Las redes móviles ad hoc, mejor conocidas como MANETs, se caracterizan por estar compuestas de diversos dispositivos que se comunican mediante enlaces inalámbricos. Las MANETs presentan diversas ventajas, comparado con las redes de infraestructura, debido a que cuentan con una gestión descentralizada lo cual permite que sus nodos se puedan comunicar entre si, sin la necesidad de un punto de acceso. De igual forma, al implementar una MANET se puede extender el área de cobertura de una red de datos inalámbrica, ya que es posible establecer sesiones de comunicación utilizando enlaces multi-salto.

Debido a la creciente demanda por parte de los usuarios que utilizan hoy en día aplicaciones que necesitan mayores prestaciones de red, como lo son las aplicaciones multimedia y en tiempo real, se vuelve necesaria la implementación de mecanismos con aprovisionamiento de calidad de servicios. En el caso de las MANETs, el uso de mecanismos para proporcionar calidad de servicios continúa siendo un tema de investigación de gran interés. Por otro lado, se han realizado diversas propuestas para reducir la sobrecarga debido a los mecanismos de enrutamiento que son utilizados en las MANETs; por ejemplo el protocolo HOLSR organiza a los nodos de la topología de red en agrupamientos o "clusters" con lo que se reduce la sobrecarga debido a los mensajes de control. Sin embargo la utilización de topologías de red basadas en agrupamientos o "clusters" introduce algunos inconvenientes como lo es la posibilidad de congestionamiento en los nodos denominados cabeza de cluster. Por lo anterior, es importante proponer mecanismos que ayuden a garantizar la calidad de servicios de aquellas sesiones de datos que requieren de mayores prestaciones en una MANET.

En este trabajo de investigación se propone la implementación del mecanismo DiffServ para garantizar las prestaciones en redes móviles ad hoc de tipo jerárquicas que utilizan el protocolo de enrutamiento proactivo HOLSR.

Palabras Clave: HOLSR, DiffServ, QoS, Topología Jerárquica

ABSTRACT of the thesis presented by **Elba Abigail Morales Vanegas**, to obtain the **MASTER OF ENGINEERING** degree. Ensenada, B. C. January 2007.

Quality of Service (QoS) Support for Mobile Ad-Hoc Networks (MANETs) with Proactive Routing using the HOLSR Protocol

Abstract approved by:

Luis Armando Villaseñor González, Ph.D.
Thesis Director

The mobile Ad-Hoc networks, better known as MANETs, are characterized by a collection of diverse devices that communicate by means of wireless connections. The MANETs offer multiple advantages, as opposed to the infrastructure networks, because they can provide a decentralized management which allows their nodes to communicate to each other without the requirement of an access point. Similarly, a MANET can be used to extend the coverage of a wireless data network, since it is possible to establish communication sessions using multihop connections.

Due to the increasing demands of the users which make use of network demanding applications, such as multimedia and real time applications, it is increasingly becoming important to implement mechanisms to provide quality of service. In the case of MANETs, the use of mechanisms to provide quality of service continues to be a research topic of great interest. On the other hand, multiple proposals have been made to reduce the overhead due to the routing mechanisms that are used in the MANETs; for example the HOLSR protocol organizes the network topology nodes in groups or "clusters" which result in a reduction of the overhead due to the control messages. However the use of network topologies based on groups or "clusters" introduces some disadvantages, such as, the possibility of traffic congestion in the cluster head nodes. As a result, it is important to propose mechanisms to guarantee the quality of service of data sessions in a MANET.

In this work we propose the implementation of the DiffServ mechanism to guarantee the quality of service in hierarchical mobile Ad-Hoc networks that make use of the proactive routing protocol HOLSR.

Keywords: HOLSR, DiffServ, QoS, Hierarchical Topology.

CONTENIDO

Página

LISTA DE FIGURAS

Figura **Descripción** **Página**

LISTA DE TABLAS

Tabla **Descripción** **Página**

Capítulo I Introducción

I.1 Antecedentes

En épocas pasadas las redes de telecomunicaciones soportaban únicamente un solo tipo de servicio, es decir existían diferentes redes para los diferentes servicios, por ejemplo, existía una red de servicios conmutados para prestar servicios de datos, otra para servicios de telefonía, etc. Gracias a la expansión de Internet los usuarios demandaron que los diferentes tipos de servicios existentes funcionaran sobre una misma red, así como los nuevos servicios que fueran apareciendo. De esta forma, las redes se ven en la necesidad de soportar diferentes tipos de tráfico sobre un mismo enlace así como de poder brindar un trato diferente para que cada tipo de servicio que se ofrece a los usuarios funcione de manera adecuada conforme a las características de cada tipo de tráfico [Caballero, 2002].

Actualmente las aplicaciones que se utilizan en la red requieren de mayor ancho de banda, por lo que las redes necesitan tener mejor desempeño y flexibilidad ya que estas deben de sostener las aplicaciones actuales y las futuras, esto nos lleva a implementar mecanismos que permitan brindar Calidad de Servicio también conocido como QoS (por sus siglas en inglés – *Quality of Service*).

Es importante brindar mecanismos que proporcionen Calidad de Servicio a las redes de comunicaciones, debido a que estas se encuentran inmersas en todo tipo de aplicaciones

de la vida diaria. Las redes de comunicaciones se dividen en dos tipos: Redes Cableadas y Redes Inalámbricas. Uno de los mejores aciertos que han tenido las redes de comunicaciones son las redes inalámbricas ya que han capturado la atención de todo tipo de usuarios debido a la posibilidad que tienen de poder desplazarse, acceder a información e incluso poder conectarse a Internet aunque el usuario se desplace de un lugar a otro, esto mientras permanezca en un área de cobertura, entre otras muchas ventajas que pueden tener las redes inalámbricas esta el hecho de que este tipo de redes son más económicas y fáciles de instalar al no requerir de una infraestructura cableada.

Actualmente se esta manejando un nuevo concepto para las redes de comunicaciones, las denominadas redes heterogéneas, en este tipo de redes los nodos que integran la red tiene diferentes capacidades de procesamiento y número de interfaces de radio; por otro lado con anterioridad se consideraban las redes como homogéneas en las cuales todos los nodos contaban con las mismas capacidades de radio y de procesamiento.

Como se mencionó anteriormente, las redes de comunicaciones se dividen en redes cableadas y redes inalámbricas, estas últimas es decir las redes inalámbricas se dividen en redes con infraestructura y redes sin infraestructura ó modo ad hoc. Las redes inalámbricas con infraestructura necesitan una estación base o un punto de acceso para su gestión sin embargo las redes inalámbricas sin infraestructura ó modo ad hoc, se componen por nodos móviles autónomos que se comunican entre si mediante enlaces inalámbricos y no necesitan de un punto de acceso o estación base para su administración ya que esta se realiza de forma descentralizada.

I.2 Planteamiento del Problema

Las redes inalámbricas han crecido de forma significativa estos últimos años, así como la necesidad de los usuarios por el uso de aplicaciones en tiempo real. Este tipo de aplicaciones son muy demandadas, sin embargo representan un problema para las redes de datos ya que estas necesitan ofrecer parámetros de Calidad de Servicio adecuados para que las aplicaciones en tiempo real tengan un desempeño aceptable. Contar con calidad de servicio en las redes ya no se considera como un servicio de valor añadido, ya debe ser parte de ellas, cualquier tipo de red debe contar con este servicio, para que la red pueda brindar un servicio de transmisión y recepción adecuado a los diferentes tipos de información que los usuarios lleguen a necesitar. Las redes basadas en el protocolo de IP (por sus siglas en inglés – *Internet Protocol*) proporcionan por defecto el nivel de servicio de entrega de datos denominado "Mejor esfuerzo" o BE (por sus siglas en inglés – *Best Effort*) [Caballero, 2002], esto es que no soporta niveles de Calidad de Servicio, ya que solo brindará mecanismos de envío de paquetes sin tomar en cuenta garantías de tiempo de entrega. En el servicio de mejor esfuerzo todo el tráfico es manejado de la misma manera sin importar la aplicación o el nodo que generó el tráfico, y el servicio es brindado en el momento que puede y como puede sin importar el tipo de aplicación que se este utilizando. Sin embargo, algunas aplicaciones como lo son las aplicaciones en tiempo real necesitan que ciertos parámetros como el retardo, la pérdida de paquetes, el Throughput entre otros no se vean afectados de manera significativa ya que estas aplicaciones son muy delicadas y pueden dejar de tener un funcionamiento aceptable si no se respetan ciertos parámetros para

su buen funcionamiento, este tipo de aplicaciones requieren algo más que mejor esfuerzo tal como un servicio diferenciado, el cual hace una diferenciación de tráfico para tratarlo dependiendo de las características que tenga cada tipo tráfico.

Todas las redes de comunicaciones deben implementar algún mecanismo que les permita tener Calidad de Servicio, es así el caso de las redes móviles ad hoc; se puede definir una red móvil ad hoc como una red autónoma de múltiples saltos, conformada de nodos móviles interconectados entre si mediante enlaces inalámbricos, sin la ayuda de estaciones base o puntos de acceso. Cualquier nodo en una red ad hoc desempeña el papel de enrutador y nodo al mismo tiempo y soporta conectividad con otros nodos que estén en el rango de cobertura. Los nodos son libres de moverse y de organizarse de manera arbitraria y dinámica. En escenarios actuales de redes móviles ad hoc se ve la existencia de redes móviles ad hoc de tipo heterogéneo organizadas de manera jerárquica y haciendo uso de nodos de gran capacidad, por ejemplo, los nodos de gran capacidad pueden ser aquellos nodos que tengan más de una interfaz inalámbrica, a estos nodos se les puede denominar como nodos "*cluster heads*". Se debe mencionar que al hacer uso de este tipo de redes se puede provocar que en los nodos de más de una interfaz se lleguen a crear congestionamientos, ya que en muchos de los casos la comunicación de unos nodos a otros debe hacerse mediante los nodos de interfaces múltiples. Por tal motivo se considera de gran interés para el área de redes móviles ad hoc heterogéneas la implementación de un mecanismo de calidad de servicio, que pueda ayudar a evitar que información importante se llegue a perder debido a los congestionamientos que se pueden llegar a provocar en los,

clusters heads, estos nodos son los encargados de la administración de los mensajes entre los niveles de la topología.

Este trabajo de tesis esta enfocado a proporcionar calidad de servicio a las redes móviles ad hoc de tipo heterogéneo, integrando el protocolo de enrutamiento proactivo de tipo heterogéneo HOLSR (por sus siglas en inglés - *Hierarchical Optimized Link State Protocol*) y el mecanismo para brindar calidad de servicio conocido como Servicios Diferenciados (DiffServ).

I.3 Objetivos

I.3.1 Objetivo General

El objetivo general de este trabajo de tesis es proporcionar calidad de servicio a las redes móviles ad hoc de tipo heterogéneo haciendo uso del mecanismo de Servicios Diferenciados (DiffServ) e integrando este mecanismo al protocolo de enrutamiento proactivo para redes heterogéneas HOLSR. El mecanismo propuesto para el soporte de QoS deberá tomar en cuenta los requerimientos de las aplicaciones (e.g. voz, video, audio), así como las características inherentes de las redes inalámbricas de tipo ad hoc.

El mecanismo de DiffServ proporcionará el medio necesario para brindar calidad de servicio a la red inalámbrica tipo ad hoc heterogéneas tomando en cuenta sus principales

características, el protocolo HOLSR proporcionará el mecanismo de enrutamiento jerárquico para las redes móviles ad hoc heterogéneas.

Es de vital importancia que la pérdida de paquetes sea lo menor posible o nula, para la transmisión correcta de información, así como que el caudal eficaz sea el correcto para que el envió y la recepción de información sea el adecuado para su uso, conforme a las necesidades que cada tipo de tráfico pueda llegar a presentar.

I.3.2 Objetivos Específicos

Para lograr el objetivo general se desprenden los objetivos específicos que se listan a continuación:

- El protocolo HOLSR y el mecanismo de DiffServ deben unificarse de manera correcta para poder proporcionar enrutamiento a las redes móviles ad hoc heterogéneas así como proporcionar parámetros de calidad de servicio.

- Los nodos inalámbricos en la red ad hoc, deben ser aptos de soportar las políticas del mecanismo de DiffServ para proporcionar calidad de servicio al tráfico en la red, para poder diferenciar y proporcionar prioridades según sea el tráfico que se este manejando.

Para poder cumplir con los objetivos antes expuestos se hará uso del programa de simulación de redes denominado NS-2 (por sus siglas en inglés - Network Simulator 2), para crear escenarios de simulación que proporcionen los resultados esperados.

I.4 Motivación

El uso comercial de las redes inalámbricas de tipo ad hoc, así como el gran uso de las aplicaciones multimedia y en tiempo real, hacen que el hecho de proporcionar calidad de servicio a las redes inalámbricas de tipo ad hoc se convierta en un área de gran interés para la investigación de este tipo de redes.

Sin embargo en la actualidad aún no se tienen muchos trabajos enfocados a proporcionar calidad de servicio a las redes móviles ad hoc, quedando como un área sin considerar todavía; el proporcionar calidad de servicio a las redes móviles ad hoc de tipo heterogéneo.

En el caso de este trabajo de tesis se hace uso de redes móviles ad hoc de tipo heterogéneo es decir, se cuenta con nodos con diferentes capacidades, se utiliza el protocolo de HOLSR como protocolo proactivo para proporcionar enrutamiento a este tipo de redes, además de trabajar con el mecanismo de servicios diferenciados (DiffServ) para proveer calidad de servicio a las redes móviles ad hoc. Siendo DiffServ uno de los principales mecanismos existentes para proporcionar calidad de servicio a las redes de

comunicaciones, resulta de gran interés el poder unificar el protocolo HOLSR y DiffServ para proporcionar calidad de servicio a las redes móviles ad hoc de tipo heterogéneo

I.5 Organización del trabajo

En el Capítulo II se describen las redes inalámbricas de infraestructura y redes inalámbricas sin infraestructura o redes móviles ad hoc, enfocándose principalmente al estudio de las redes ad hoc, sus principales características, escenarios de uso, tipo de enrutamiento y funcionamiento en general.

En el Capítulo III se describe y explica la funcionalidad del protocolo de enrutamiento OLSR y la forma en como proporciona enrutamiento a las redes ad hoc.

En el Capítulo IV se describe y explica la funcionalidad del protocolo de enrutamiento HOLSR y la forma en que proporciona enrutamiento a las redes ad hoc, así como la manera en que mejora el protocolo plano OLSR; el protocolo HOLSR es utilizado para la realización de este trabajo.

En el Capítulo V se detallan los tipos de mecanismos para proporcionar calidad de servicio, enfocándose principalmente en el mecanismo DiffServ, describiendo sus principales características y modo de operación para proveer calidad de servicio, ya que este mecanismo es el utilizado para proporcionar calidad de servicio en este trabajo de tesis.

El Capítulo VI se explica la herramienta de simulación utilizada para realizar este trabajo que es el simulador NS -2, así como las extensiones que se agregaron para poder implementar el protocolo de enrutamiento OLSR, que fue la base para poder trabajar con el protocolo HOLSR, y la utilización del modulo de DiffServ en NS-2.

El Capítulo VII se indican las modificaciones realizadas al simulador para implementar la extensión de OLSR como protocolo de enrutamiento, y de esta forma poder realizar las modificaciones necesarias para incorporar el protocolo HOLSR, así como las modificaciones hechas para que se implementará el mecanismo de DiffServ con redes móviles ad hoc. También se presentan los escenarios que se utilizaron para realizar las simulaciones y se describen los resultados obtenidos de tales simulaciones.

El Capítulo VIII se presenta las conclusiones obtenidas en este trabajo de tesis.

I.6 Resumen

En este capítulo se presentan los objetivos de este trabajo de tesis, así como el problema a resolver, la motivación que existió para realizar este trabajo, así como la organización de este trabajo de tesis.

El capítulo siguiente es referente a las redes inalámbricas con infraestructura y sin infraestructura o redes móviles ad hoc, enfocándonos principalmente al estudio de estas últimas.

Capítulo II Redes inalámbricas

II.1 Introducción

Hoy en día las redes inalámbricas tienen un papel muy importante, debido a que día a día muchas son las personas que cuentan con dispositivos portátiles ya que deja a un lado las molestas instalaciones de cable y en muchos casos no se necesitará la ayuda de una infraestructura central, sumado a esto un punto muy importante es que existen dispositivos móviles a costos muy económicos, y esto hace que el uso de las redes inalámbricas cada vez se este extendiendo más entre los usuarios.

La principal función de estas redes es proporcionar movilidad a sus usuarios al mismo tiempo que les provee conectividad y les proporciona acceso a las redes cableadas tradicionales.

Debido a la creciente proliferación de redes inalámbricas, se han desarrollado estándares para controlar y mejorar su funcionamiento, entre estos se encuentran HiperLAN, Bluetooth y el IEEE 802.11 (siendo este el más popular) los cuales se detallan a continuación.

II.1.1 Estándar IEEE 802.11

El IEEE (por sus siglas en inglés - *Institute of Electrical and Electronic Engineers*) estableció las especificaciones para el estándar 802.11, como un estándar para las Redes Inalámbricas de Área Local [IEEE, 1996]. La IEEE 802.x define el uso para los dos niveles más bajos de la arquitectura del modelo OSI (la capa física y la de enlace de datos). La primera versión del 802.11 tenia velocidades de 1 hasta 2Mbps, y trabajaba en la banda de frecuencia de los 2.4 GHz, sin embargo debido al creciente desarrollo de las redes inalámbricas estas velocidades eran insuficientes por lo que se creo el IEEE 802.11b, este nuevo estándar tiene velocidades hasta de 11 Mbps y también trabaja en la banda de frecuencia de los 2.4 GHz, el estándar de la IEEE 802.11a alcanzaba velocidades de hasta 54 Mbps pero estaba especificado para trabajar en la banda de frecuencia de 5Ghz, lo cual provocó que fuera incompatible con los productos del estándar 802.11b [Nobel, 2000], debido a eso se desarrolló otro estándar que fuera compatible con el IEEE 802.11b que fue el estándar 802.11g. Actualmente se están estudiando el desarrollo de nuevos estándares dentro de la familia IEEE 802.11 para proporcionar mejores y nuevas prestaciones a los usuarios de las redes inalámbricas.

El estándar 802.11 cuenta con dos diferentes tipos de redes inalámbricas: redes con infraestructura, y las redes sin infraestructura o modo Ad-Hoc, los cuales se explican más adelante en este capítulo.

II.1.2 Otros estándares

Además del estándar 802.11 existen otros estándares para redes inalámbricas WLANs [Oyoqui, 2003], como lo son: HiperLAN y Bluethooth más sin embargo el mercado actual lo ha dominado el estándar 802.11. No obstante han empezado a haber varios dispositivos Bluethooth, estos dispositivos están orientados hacia la creación de las redes WPAN (por sus siglas en inglés - *Wireless Personal Area Networks*).

II.1.2.1 Hiperlan

El Instituto Europeo de Estándares en Telecomunicaciones (por sus siglas en inglés - *European Telecomunications Standards Institute*) comenzó a desarrollar estándares para redes inalámbricas más rápidas, dentro del proyecto BRAN (por sus siglas en inglés - *Broadband Radio Access Networks*) donde crearon el estándar HiperLAN (por sus siglas en inglés - *High Performance Radio Local Area Network*). HiperLAN fue separado en diferentes tipos, esto es debido a que se deseaba tener diferentes clases de servicios y aplicaciones para las WLANs. Se divide en cuatro tipos:

HiperLAN 1: este es el primer estándar de la familia de HiperLAN. Este estándar opera en la banda de 5.2 GHz con un espectro de 100 MHz y ofrece comunicación uno a uno así como uno a muchos a través de broadcasting. Cuenta con una velocidad de hasta 19 Mbps. El canal ofrece auto configuración y flexibilidad de uso gracias al canal de acceso

distribuido EYNPMA (por sus siglas en inglés - *Elimination Yield No Pre Emptive Prioriti Multiple Access*). Para resolver el problema de colisiones HiperLAN utiliza CSMA/CA, el esquema reparte la capacidad de radio disponible entre los usuarios activos que intentan trasmitir datos en un intervalo de tiempo en común.

HiperLAN 2: opera en la banda de 5.2 GHz con un espectro de 100 MHz, y tiene velocidades de hasta 54 Mbps. Trabaja de forma centralizada, cada Terminal móvil se comunica con los puntos de acceso. Cada uno de los usuarios puede moverse por la red libremente, obtiene las mejores prestaciones de transmisión posibles, [Magnus *et al.*, 2000].

HiperAccess: este es el estándar HiperLAN 3 pero su nombre fue cambiado por HiperAccess. Proporciona acceso inalámbrico en exteriores. Dado que proporciona conexión inalámbrica externa se ha propuesto para aplicaciones estacionarias y semi estacionarias, llega a tener alcance de hasta 5 Km de cobertura entre los puntos de acceso inalámbricos y terminales inalámbricas. Trabaja en el rango de frecuencias de entre 11GHz y 42GHz.

HiperLink: este es el estándar HiperLAN 4 pero su nombre fue cambiado por HiperLink. El HiperLink proporciona servicios de interconexión que necesiten altas velocidades de datos. HiperLink proporciona interconexiones punto a punto a velocidades de hasta 155 Mbps en distancias de hasta 150 metros Trabaja en el rango de frecuencias de 17 GHz con un espectro de 200 MHz.

Los estándares de HiperLAN 2 e HiperLink fueron diseñados en un principio para soportar únicamente redes ATM. Más sin embargo actualmente HiperLAN2 soporta accesos a redes IP.

II.1.2.2 Bluetooth

Bluetooth es un estándar de tecnología económica, que se utiliza para proporcionar conexiones inalámbricas de corto alcance entre dispositivos tales como teléfonos celulares, impresoras, PDAs (por sus siglas en inglés - *Personal Digital Assistants*), teclados, computadoras de escritorio, computadoras portátiles, etc. Esta tecnología es utilizada principalmente para la transferencia de datos y voz entre los diferentes dispositivos y las computadoras personales. Bluetooth alcanza distancias de hasta 10 metros, actualmente se esta estudiando un aumento de potencia en el transmisor a 100 mW para que cuente con un alcance de hasta 100 metros.

La compañía Ericsson fue la que desarrollo bluetooth cuando empezó a estudiar diferentes opciones para comunicar los teléfonos celulares con otros dispositivos. Bluetooth SIG es el que dirige actualmente este estándar que incluye a varios miembros como Nokia, Motorota, Lucent Technologies, 3 Com, Intel, Toshiba, IBM y Microsoft. Bluetooth opera en la banda de 2.4 GH utilizando la tecnología de radio de espectro disperso, es una

tecnología de radio frecuencia (RF). Su banda de operación esta dividida en canales de 1 MHz, y a 1 mega símbolo por segundo se obtiene el ancho de banda máximo por canal.

Utiliza el esquema de modulación GFSK (por sus siglas en inglés - *Gaussian Frequency Shift Keying*), [Haartsen, 1998]. Utilizando GFSK, un 1 binario representa una desviación positiva de la portadora nominal de la frecuencia, mientras que un 0 representa una desviación negativa. Cuando se termina la transmisión de un paquete, ambos dispositivos re-sintonizan su radio transmisor a una frecuencia diferente de la que tenían en ese momento, pasando de un canal de radio de manera pseudoaleatoria conocida como espectro disperso con salto de frecuencia (FHSS).

Bluetooth fue diseñado para aplicaciones móviles de poca potencia, la potencia del radio transmisor debe ser minimizada. Tiene potencias cortas de conexión y en lo referente a ancho de banda solo puede soportar hasta 780 kbps, los cuales pueden ser utilizados parata transferir unidireccionalmente 721 kbps y 57.6 kbps en la dirección de retorno o hasta 432.6 kbps de manera simétrica en ambas direcciones.

II.2 Características de las redes inalámbricas

Las redes inalámbricas cuentan con las siguientes características [Corson y Macker, 1999]:

- ***Ancho de banda reducido***: los enlaces inalámbricos tienen capacidades más bajas que las redes cableadas, esto provoca que las redes inalámbrica tengan una baja calidad de servicio, mayor retardo y jitter (variación en el retardo), comparado con las redes cableadas tradicionales.

- ***Energía limitada***: algunos o todos los nodos pueden usar baterías u otros medios agotables para que les proporcionen la energía necesaria para su funcionamiento. Para estos nodos, los criterios más importantes del diseño del sistema para la optimización puede ser la conservación de energía.

- ***Seguridad física limitada***: las redes inalámbricas son más propensas a sufrir ataques físicos que las redes cableadas, ya que es más fácil poder acceder a su información dada su naturaleza.

- ***Topología dinámica***: los nodos se pueden mover arbitrariamente, la topología de la red es multi-salto, esto es que puede cambiar aleatoria y rápidamente, y puede tener enlaces bidireccionales y unidireccionales.

- ***Radio de transmisión limitado***: como utilizan antenas esto esta limitado al alcance de las mismas.

- ***Errores de transmisión y/o pérdida de paquetes***: esto es debido a la desconexión de los nodos o a la movilidad en la que se encuentran y sus condiciones pueden cambiar.

II.3 Tipos de redes inalámbricas

Se cuenta con dos diferentes tipos de redes inalámbricas: Redes con Infraestructura, y las Redes sin Infraestructura o modo Ad-Hoc.

II.3.1 Red con Infraestructura

Este tipo de red inalámbrica consta de una Estación Base (BS por sus siglas en inglés - *Base Station*) o mejor conocido como Punto de Acceso (AP por sus siglas en inglés - *Access Point*) mediante el cual los dispositivos móviles se comunican de manera inalámbrica, los AP brindan servicio a un cierto número de usuarios, estos usuarios tienen que estar contenidos en un área determinada, donde el AP tenga alcance para brindarles el servicio, estos usuarios se comunicarán con otros mediante el AP. Por lo tanto es indispensable que los usuarios que desean comunicarse se encuentren dentro del área donde el AP pueda brindarles servicio.

Varios AP pueden existir separados en área geográfica y los usuarios móviles pueden pasar de un área de cobertura que brinda un AP a otra área de cobertura brindada por otro AP para mantenerse conectados en diferentes áreas de cobertura, de esta manera los usuarios móviles podrán estar conectados, al pasar a otra área de cobertura. Es decir cada nodo móvil debe comunicarse con el AP dentro de su radio de acción. El nodo puede moverse libremente pero si sale fuera del rango de su enlace, debe conectar con otro para asegurarse de que la información llegue a su destino final.

Un ejemplo claro de este tipo de redes, es la red de telefonía móvil formada por numerosas estaciones y antenas dispersas por todas las ciudades. En la Figura 1, se muestra el ejemplo de una red inalámbrica con infraestructura.

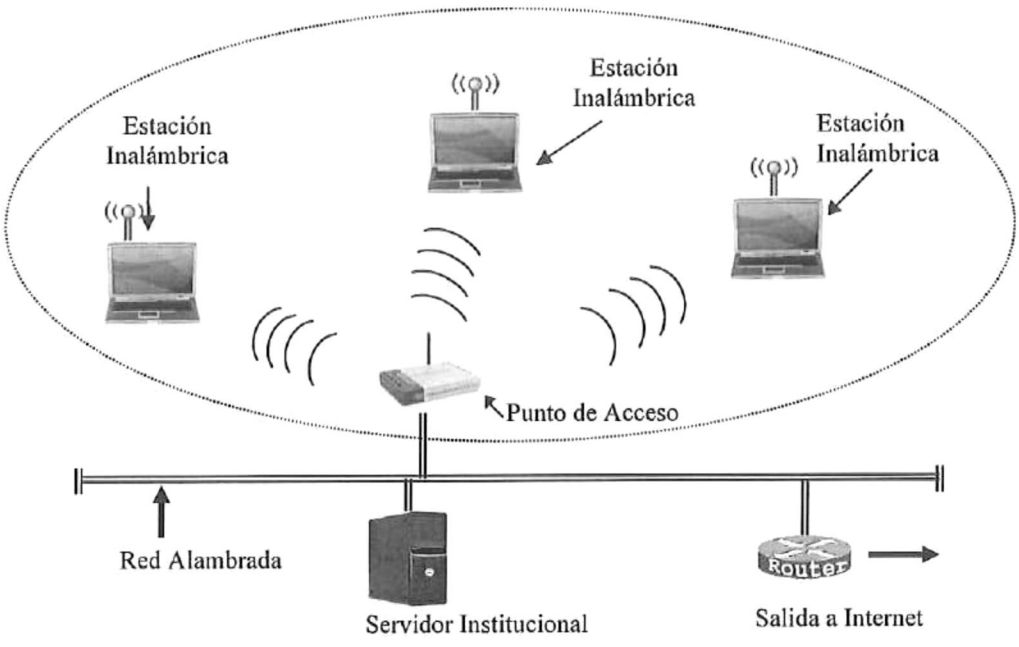


Figura 1.- Red inalámbrica con infraestructura.

II.3.2 Red sin Infraestructura o modo Ad-Hoc

Este tipo de redes son básicamente un conjunto de estaciones 802.11 que se comunican entre si de manera directa sin la necesidad de un punto de acceso. En este tipo de redes no existen estaciones base y por tal motivo no se necesitan de ningún permiso para poder comunicarse entre los nodos existentes en dicha red. En las redes Ad-Hoc los dispositivos interactúan unos con otros permitiendo que exista una comunicación directa entre los dispositivos es decir que tengan una comunicación punto a punto. Usando esta modalidad las comunicaciones se realizan todos con todos, sin que existan elementos centrales que canalicen esta comunicación. En este tipo de redes el crecimiento se da a medida que se van incorporando nuevos elementos a la red, dentro del mismo espacio de cobertura de los dispositivos de red. El funcionamiento más detallado de este tipo de redes se explica a continuación.

II.4 Redes móviles Ad-Hoc

Este tipo de red inalámbrica sin infraestructura es también conocida como Red Móvil Ad-Hoc o MANET (MANET por sus siglas en inglés *Mobile Ad-Hoc NETwork*), consiste en un conjunto de nodos móviles autónomos que se comunican entre si mediante enlaces inalámbricos [Corson y Macker,1999].

Los nodos que forman una MANET pueden estar conectados entre sí arbitrariamente y de manera dinámica, en este tipo de redes no existe una infraestructura de red fija y la administración se realiza de forma descentralizada, todos los nodos funcionan como enrutadores y se ven involucrados tanto en el descubrimiento como en el mantenimiento de rutas. En la Figura 2, se muestra el ejemplo de una red inalámbrica sin infraestructura.

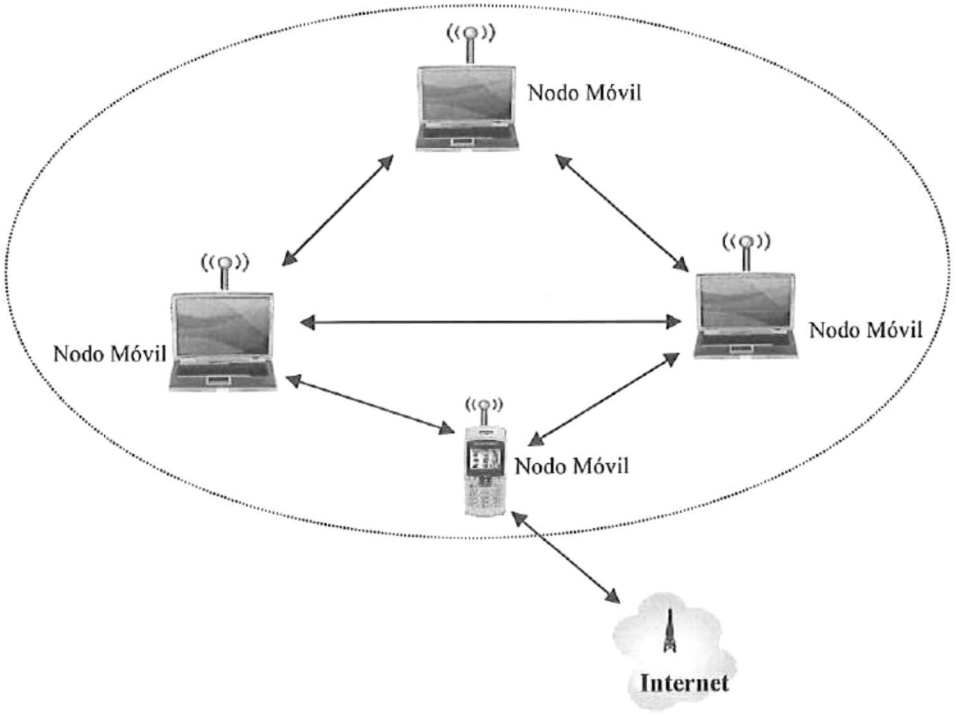


Figura 2.- Red inalámbrica Ad-Hoc.

Entonces podemos decir que una MANET es una estructura de red autónoma compuesta de varios nodos que pueden moverse libremente, es decir, una plataforma móvil donde todos los nodos actúan como enrutadores y nodos indistintamente [Corson y Macker,

1999]. Algunos ejemplos de uso de las redes *Ad-Hoc* son: operaciones de emergencia de búsqueda y rescate, convenciones y análisis de datos en terrenos catastróficos.

Las redes móviles Ad-Hoc se forman de manera dinámica, estas redes pueden surgir prácticamente de la nada y ser creadas de manera espontánea, todo esto sin la ayuda de algún administrador de red o algún encargado de la misma, para su configuración y administración. El tiempo de vida de los nodos participantes en una MANET es variante, dependiendo de la necesidad de cada usuario pero por lo regular el tiempo de los nodos es corto, estas redes por si solas agregan dispositivos conforme se van necesitando, esto es en medida que los usuarios se están adentrando al área de cobertura de la red, así como también elimina estos si se alejan de la misma, todo de manera dinámica, y sin aviso, entonces los nodos entran y salen de la red, lo que trae como consecuencia que el tamaño de la red sea variable en todo momento.

Debido a las características presentes en las MANET son una buena opción para su utilización en la actualidad, dado que se pueden colocar en cualquier parte y para cualquier uso.

En general, cualquier propuesta real aplicable a una MANET deberá tener en cuenta las restricciones impuestas por las características con las que cuenta estas redes. Las características son: topología dinámica, enlaces de ancho de banda limitado y capacidad variable, limitaciones de energía y capacidad de procesamiento en los nodos y seguridad física limitada [Corson y Macker, 1999].

Las MANET como se me mencionó anteriormente pueden desplegarse de forma completamente autónoma o combinarse con las redes locales inalámbricas para conectarse a Internet utilizando puntos de acceso inalámbricos. Los nodos móviles deben poder adaptase de forma dinámica ante los cambios continuos que se presentan en estas redes, como los cambios de posición de las estaciones, cambio de potencia de la señal, tráfico de red, así como la distribución de la carga. De lo anterior, el principal reto de las redes Ad-Hoc son los continuos cambios existentes en la topología de la red.

Existen en la actualidad estándares de comunicación inalámbrica y protocolos de control de acceso al medio (MAC), los cuales nos brindan los elementos necesarios para que se pueda establecer una comunicación inalámbrica entre dispositivos. Existen dos estándares que ofrecen el soporte necesario para poder tener una MANET, el Bluetooth y IEEE 802.11.

II.4.1 Características de las redes móviles Ad-Hoc

A continuación se enlistan las características particulares de una Red Móvil Ad-Hoc (MANET) [Macker y Corson, 1998]:

- ***Nodos Autónomos*:** Todos los nodos que conforman la red MANET son autónomos, estos nodos pueden realizar tareas de enrutamiento (enrutadores), y se

ven involucrados tanto en procesamiento de datos, así como de descubrimiento y mantenimiento de rutas.

- ***Topología Dinámica de la red***: Las redes MANET se forman de manera dinámica, estas redes por si solas agregan dispositivos en medida en que los usuarios se están adentrado a la red, así como también los elimina de la misma, todo esto de manera dinámica, por lo tanto las rutas de estas cambian sin previo aviso y por tanto su topología.

- ***Funcionamiento no centralizado***: El control y la administración de la red no se encuentran en algún nodo central, en este caso se encuentra distribuido en cada uno de sus nodos, cada nodo actúa de manera que se acople a las necesidades de la red.

- ***Nodos con capacidades limitadas***: Los nodos en este tipo de redes son dispositivos con limitaciones de energía y capacidad de procesamiento, debido a esto se debe hacer uso de algoritmos que permitan a estos nodos optimizar los recursos.

- ***Enrutamiento multi-salto***: En virtud de que cada nodo en una MANET funciona como un enrutador si es necesario, es decir cada nodo puede enrutar paquetes entre los nodos. Los paquetes que son mandados de un nodo fuente pueden alcanzar el nodo destino mediante múltiples saltos entre varios nodos intermedios.

En una MANET los nodos se utilizan a si mismos y a otros nodos como enrutadores, para que estos hagan llegar el paquete hasta su destino. Por ejemplo si un nodo desea enviar un paquete a otro nodo que se encuentre fuera de su alcance de transmisión (e.j. fuera de la cobertura de la señal de radio), los paquetes se van enrutando salto por salto por diferentes nodos intermedios hasta alcanzar su destino final.

Para poder implementar cualquier aplicación o protocolo para las redes MANET se debe tener en cuenta las características inherentes que tiene este tipo de redes para proporcionar un buen funcionamiento a las redes y tratar de mitigar los efectos perjudiciales que pudieran tener.

II.4.2 Escenarios de uso

La tecnología de redes móviles Ad-Hoc inicialmente estuvo enfocada y limitada únicamente a los entornos militares, ya que la mayor parte de las investigaciones sobre telecomunicaciones tiene su origen en el ejercito [Yang *et al.*, 2003].

Las redes móviles ad- hoc son muy útiles en entornos militares dado que permite que los soldados de una misma tropa se puedan comunicar entre ellos en terrenos desconocidos y donde no exista una infraestructura de comunicaciones o que en su defecto esta haya sido destruida previamente, es decir este tipo de redes en el campo militar permiten que los usuarios puedan estar comunicados sin contar con una infraestructura

establecida y les permite tener gran movilidad en el campo de batalla. Dado que en un campo de batalla las telecomunicaciones siempre deben estar presentes para tener un buen funcionamiento de coordinación con las tropas aliadas este tipo de redes vienen a ser una gran ayuda para los diferentes ejércitos militares.

Debido a los continuos avances que se están presentando en el área tecnológica se ha ampliado el uso de las redes móviles Ad-Hoc más allá del ámbito militar.

Las MANET tienen una gran gama de usos, son muy útiles cuando ha ocurrido algún desastre natural (terremotos, huracanes, etc.) que dejan incomunicados tanto a las personas civiles como a los cuerpos de rescate, ya que los sistemas de comunicaciones pueden resultar dañados, haciendo que las tareas de rescate se dificulten al no saber si todavía existen sobrevivientes o en que estado se encuentran los sobrevivientes, y bajo que condiciones esta el terreno. Es aquí donde las redes MANET tienen una labor importante, ya que al encontrarse sin infraestructura el área donde ha ocurrido la catástrofe, los cuerpos de rescate pueden seguir teniendo una comunicación interna con la ventaja de que con un dispositivo móvil pueden encontrar un enlace que les permita comunicarse al exterior del área y de esta forma pedir lo que necesiten.

Otra aplicación que pueden tener este tipo de tecnologías es el uso didáctico, ya que mediante una red Ad-Hoc se pueden compartir archivos en un aula escolar entre los diferentes alumnos sin la necesidad de que exista una infraestructura de red establecida,

haciendo con esto que se tenga un ahorro en las instituciones educativas, al no tener que contar con Puntos de Acceso.

Pero este tipo de redes no excluyen al ambiente doméstico, que sin duda alguna está creciendo día con día ya que los nuevos electrodomésticos se están equipando con capacidades especiales para poder interconectarse entre si de forma inalámbrica. Es decir que el refrigerador pueda mandar a la PDA la lista de alimentos que se han agotado para que puedan volver a ser surtidos, y todo esto de forma inalámbrica, o mandar las instrucciones de lavado a la lavadora mediante el refrigerador, entonces una forma en poder crear una interconexión entre los dispositivos es creando una MANET en el hogar.

Como se ha podido observar este tipo de redes cuentan con un espectro amplio de escenarios de uso, que va desde lo militar hasta lo doméstico.

II.4.3 Enrutamiento

Debido a la movilidad de sus nodos, los mecanismos utilizados para enrutar paquetes en las MANET impiden la utilización de protocolos de enrutamiento desarrollados para las redes clásicas (con infraestructura), dado que estos suponen que la topología de red es poco cambiante y los algoritmos clásicos no convergen lo suficientemente rápido. Los protocolos desarrollados para las redes clásicas no se adaptarían al dinamismo de una red MANET.

Como se ha explicado anteriormente los nodos en una MANET funcionan como enrutadores, es decir que los nodos deben elegir la trayectoria que los paquete deben seguir en función de algunos parámetros como lo es la sobrecarga; de igual forma los nodos en una MANET necesitan actualizar continuamente sus tablas de enrutamiento, para que así puedan conocer la ruta hacia los demás nodos que forman la red.

Existen varios esfuerzos de investigación que han buscando darle solución al enrutamiento en las redes Ad-Hoc proponiendo algunas técnicas para conseguir buenos mecanismos de enrutamiento, el cual este acorde a las características de las redes Ad-Hoc, es decir a la alta movilidad de los nodos, al poco ancho de banda y sus nodos con memoria reducida, el grupo de trabajo MANET dentro del IETF (por sus siglas en inglés -Internet Engineering Task Force) [Corson y Macker, 1999] ha realizado investigaciones referentes a los temas relacionados con el enrutamiento y el cual ha tenido buenos resultados.

Actualmente existen algunos protocolos que cuentan con RFC (por sus siglas en inglés - *Request for Comment*) ya que han superado a las otras propuestas existentes para este tipo de redes [IETF, 2006]: *Ad-Hoc On Demand Distance Vector* (AODV) se define en el RFC 3561 [Perkins y Belding-Royer, 2003], *Optimized Link State Routing Protocol* (OLSR) se define en el RFC 3626 [Clausen y Jacques, 2003], y *Topology Broadcast base don Reverse-Path Forwarding* (TBRPF) se define en el RFC 3684 [Ogier *et al.*, 2004].

II.4.3.1 Protocolos Proactivos (Table-Driven)

Estos protocolos están basados en tablas de enrutamiento, es decir que cada nodo siempre mantiene las rutas a todos los posibles destinos, de esta forma las rutas a todos los destinos se calculan antes de ser utilizados, de esta manera cuando se requiere hacer uso de alguna ruta siempre va a estar disponible, lo cual tiene como consecuencia que los retardos sean pequeños o nulos para la determinación de la ruta, debido a que ya esta establecida. Es decir las rutas generadas por estos protocolos van a estar actualizadas frecuentemente dado que mandan mensajes de actualización periódicos a través de la red para mantener las tablas siempre actualizadas, lo cual provoca que se tenga un alto consumo de ancho de banda y energía además de que mantiene rutas que quizás nunca se utilicen, más sin embargo se pueden seleccionar rutas de forma casi inmediata, lo cual hace que sean muy eficientes porque no provoca retardos para determinar una ruta. Los nodos presentes en una red van a tener conocimiento de las tablas de enrutamiento para poder enviar los paquetes hacia cualquier nodo en la red, de esta forma cuando un nodo desea mandar un paquete hacia un nodo destino, el nodo fuente conocería cual es la ruta para que el paquete viaje a su destino final.

Enseguida se enlistan algunos de los principales protocolos proactivos:

- *DSDV (Destination Sequence Distance Vector).*

- *OLSR (Optimized Link State Routing).*

- *TBRPF (Topology Dissemination Based on Reverse-Path Forwarding).*

- *FSR (Fisheye Routing Protocol).*

También existe una propuesta de un protocolo de enrutamiento proactivo que trabaja de forma jerárquica, es el protocolo HOLSR (por sus siglas en inglés - *Hierarchical Optimized Link State Routing*).

A continuación se describen de manera breve cada uno de los protocolos proactivos enlistados anteriormente:

II.4.3.1.1 DSDV

DSDV (por sus siglas en inglés - *Destination Sequence Distance Vector*) es un protocolo proactivo y esta basado en tablas de enrutamiento, lo cual quiere decir que cada nodo tiene una tabla con las direcciones hacia los demás nodos en las red, así como el número de saltos (métricas) que tendría que dar cada paquete para que llegue a su destino final, este protocolo esta basado en el algoritmo de vector distancia. El protocolo DSDV únicamente proporcionara una ruta para cada destino, sin embargo siempre será la ruta más corta, esto lo determina basándose en el número de saltos que hay que dar a cada destino.

El hecho de tener mensajes de actualización constantes (aproximadamente cada 15 segundos) [Parking *et al.*, 1994], para mantener las tablas de enrutamiento actualizadas, provoca que exista mucho tráfico; para minimizar esto se cuentan con dos tipos de paquetes: *full dump*, este paquete contiene toda la información existente referente al enrutamiento y en algunas ocasiones puede ser necesario que este paquete sea dividido en fracciones más pequeñas, el paquete *full dump* no es utilizado cuando los cambios en la red son pequeños. Cuando la información ha variado muy poco a partir del último *full dump* entonces se utiliza otro paquete que se llama *incremental*, cada nodo tiene una tabla donde guarda todos los datos recibidos por estos paquetes. Existen varias opciones en la red que se pueden llegar a utilizar para la elección de la ruta al destino, en primera instancia se tiene que se elige dependiendo del número de secuencia más actualizado, si no es posible elegir el destino por este medio se toma en cuenta el que contenga la métrica más corta. DSDV es un buen protocolo si la red donde esta implementado tiene una movilidad media, además de que los nodos que intervengan mantengan una buena comunicación.

II.4.3.1.2 OLSR

OLSR (por sus siglas en inglés - *Optimized Link State Routing*) es un protocolo proactivo es decir que esta basado en tablas de enrutamiento, en donde los nodos continuamente hacen difusión (broadccast) de mensajes para que se conozca el estado de sus enlaces, es decir que este protocolo esta basado en el intercambio periódico de

mensajes, y de esta forma mantiene actualizada la información de los nodos y de los enlaces. Los nodos utilizan la información recibida para determinar cuales son las trayectorias que puede alcanzar el solo o mediante la ayuda algún nodo en la red. Cada nodo **X** que conforme la red debe tener la capacidad de alcanzar a cualquier "vecino a 2 saltos", para esto el nodo **X** escogerá de entre sus vecinos a un nodo que sea capaz de proporcionarle esta conectividad, a estos nodos se les conoce como MPR's. El protocolo OLSR hace uso de dos tipos de mensajes para el descubrimiento de sus rutas, los mensajes HELLO y los menajes TC [Clausen y Jacquet, 2003]. Este protocolo se describe detalladamente en el capítulo siguiente.

II.4.3.1.3 TBRPF

TBRPF (por sus siglas en inglés - *Topology Dissemination Based on Reverse-Path Forwarding*) es un protocolo de enrutamiento proactivo, la difusión de la topología esta basada en el envió de la trayectoria de reversa [Ogier *et al.*, 2004]. TBRPF es un protocolo de estado del enlace, esta diseñado para redes móviles Ad-Hoc, y provee un enrutamiento salto por salto (hop by hop), a lo largo de las trayectorias más cortas al destino correspondiente. Los nodos que utilizan este protocolo, cada uno debe funcionar con TBRPF, de esta forma se calcula un árbol de la fuente (el cual proporciona todas las trayectorias a todos los nodos accesibles) basado en la información parcial de la topología almacenada en su tabla de la topología, usando una modificación del algoritmo de Dijkstra.

Para reducir al mínimo la sobrecarga generada, cada nodo divulga solamente "parte" de su árbol fuente a los vecinos. TBRPF utiliza una combinación de actualizaciones periódicas y diferenciadas para mantener a todos los vecinos informados sobre la divulgación de su árbol de la fuente. Cada nodo también tiene la opción para divulgar la información adicional de la topología (hasta la topología completa), para proporcionar robustez mejorada en redes altamente móviles. TBRPF realiza descubrimiento vecino usando los mensajes HELLO que divulgan solamente los cambios en el estado de los vecinos. Esto da lugar a los mensajes HELLO que son mucho más pequeños que los de otros protocolos de enrutamiento de estado del enlace tal como OSPF (por sus siglas en inglés - *Open Shortest Path First*) [Ogier *et al.*, 2004].

II.4.3.1.4 FSR

FSR (por sus siglas en inglés - *Fisheye Routing Protocol*) es un protocolo proactivo que funciona manteniendo las distancias precisas así como la información que indique cual es la mejor ruta de mejor calidad para el vecindario que este más cercano al nodo. A medida que la distancia del nodo aumente, entre más lejano esté el nodo de algún vecindario los detalles van a ir disminuyendo progresivamente. Para realizar esta tarea la red se divide en partes, específicamente en círculos con respecto a cada uno de sus nodos. FSR realiza un intercambio periódico de los mensajes para conocer el estado de los enlaces.

Los círculos son hechos en función al número de saltos que se tienen que hacer con respecto al nodo central. FSR proporciona dos mejoras para tener una disminución de sobrecarga al realizar un broadcast: La información es mandada únicamente a los vecinos de "un solo salto" y la información con respecto a los enlaces no será mucha, ya que se estará mandando información con mayor frecuencia, y así no se irá acumulando. Las actualizaciones son frecuentes para los nodos cercanos y los nodos lejanos tienen grandes latencias, dado que se tiene poco conocimiento de los nodos lejanos, esto es compensado en medida que el mensaje se mueve hacia el destino. Este protocolo es bueno ya que presenta gran escalabilidad para las redes grandes así como reducción de la latencia en comparación con el enrutamiento basado en demanda [Pei *et al.*, 2000].

II.4.3.1.5 HOLSR

Es un protocolo que esta basado en las especificaciones del algoritmo de OLSR.

HOLSR (por sus siglas en inglés - *Hierarchical Optimized Link State Routing*) es un protocolo proactivo esto es que esta basado en tablas de enrutamiento, en donde los nodos continuamente hacen difusión (broadccast) de mensajes para que se conozca el estado de sus enlaces, es decir que este protocolo esta basado en el intercambio periódico de mensajes, y de esta forma mantiene actualizada la información de los nodos y de los enlaces. Los nodos utilizan la información recibida para determinar cuales son las trayectorias que puede alcanzar el solo o mediante la ayuda algún nodo en la red. Cada

nodo tiene sus rutas establecidas, lo cual hace que no se pierda tiempo en la búsqueda de alguna ruta, ya que se están actualizando constantemente y los nodos siempre tienen sus rutas activas a todos los destinos posibles. Este protocolo utiliza los mismos mensajes para el descubrimiento de rutas que usa que usa el OLSR, utiliza el mensaje HELLO y el mensaje TC, sin embargo este protocolo introduce dos nuevos mensajes para la administración jerárquica del protocolo, usa el mensaje CIA y el mensaje HTC. El mensaje CIA es utilizado para la creación de clusters, este mensaje es enviado por primera vez por el cluster head a los nodos vecinos para que mediante el, los nodos vecinos decidan unirse a algún cluster, y el mensaje HTC es un mensaje similar al mensaje TC, sin embargo este mensaje es mandado únicamente por los cluster heads en su interfaz superior. HOLSR organiza a los nodos de manera dinámica en cluster y estos clusters a su vez son organizados en una arquitectura jerárquica y de esta forma reduce la cantidad de información de control de la topología que se difunden en la red. Otra ventaja importante, es la reducción en el costo del cálculo de la ruta, por ejemplo si un enlace de la red se rompe, solo los nodos que están dentro de ese cluster necesitarán volver a calcular sus tablas de enrutamiento, ya que los nodos que estén en otros clusters no serán afectados. Con la estructura jerárquica se optimiza el protocolo OLSR reduciendo los gastos de los mensajes y el tamaño de la tabla de enrutamiento de los nodos. En HOLSR se entrega un mayo número de paquetes que con el protocolo OLSR lo que da mayor ventaja al protocolo HOLSR, además este protocolo reduce la incidencia de paquetes de datos perdidos, y mejora la escalabidad del protocolo, permite un uso eficiente de nodos heterogéneos [Villaseñor-González *et al.*, 2005]. Este protocolo se explica a más detalle en el Capítulo III.

II.4.3.2 Protocolos Reactivos (On-demand Driven)

Los protocolos Reactivos funcionan bajo demanda, esto quiere decir que los protocolos solo calculan las rutas cuando son necesarias, o en otras palabras se calcula solo cuando van a ser utilizadas. Por ejemplo, cuando un nodo desea mandar información a otro, el nodo fuente es el encargado de buscar la ruta a utilizar mediante un procedimiento de descubrimiento de la ruta del nodo destino.

El hecho de buscar una ruta solo cuando es necesaria provoca que se tenga un bajo nivel de carga (*overhead*). Es decir las rutas solo son determinadas según la demanda lo cual es un punto bueno para su utilización en las redes móviles Ad-Hoc. Una de las desventajas con las que cuenta este tipo de protocolos es que tienen retardos importantes al intentar determinar una ruta, esto provoca que la latencia inicial pueda degradar el desempeño de aplicaciones interactivas, y no permite que la red reaccione rápido ante los cambios que se puedan llegar a presentar; una de las características de este tipo de protocolos es que hacen uso de algoritmos de inundación.

Como se ha visto, estos protocolos lo que intentan hacer es reducir la sobrecarga que se genera por los mensajes de control para el mantenimiento de las rutas. Este tipo de protocolos permite que cuando las rutas no se necesitan los nodos puedan entrar en estado de "inactividad", esto es algo muy bueno ya que permite que se tenga un ahorro de energía.

Enseguida se enlistan algunos de los principales protocolos reactivos:

- *AODV (Ad-Hoc on Demand Distance Vector).*
- *DSR (The Dynamic Source Routing).*
- *TORA (Temporally Ordered Routing Algorithm).*

A continuación se describen de manera breve cada uno de los protocolos reactivos enlistados anteriormente:

II.4.3.2.1 AODV

AODV (por sus siglas en inglés - *Ad-Hoc On Demand Distance Vector*) es un protocolo reactivo, por lo tanto es un protocolo en donde se descubren las rutas según se utilicen, es decir bajo demanda. Este protocolo esta basado en el algoritmo de vector distancia, además es de enrutamiento dinámico; el protocolo AODV esta diseñado para redes móviles Ad-Hoc. Permite a los nodos móviles de la red obtener rutas rápidamente para los destinos nuevos y no se requiere que los nodos mantengan las rutas activas a los destinos que no se estén utilizando, de esta forma AODV mantiene tablas de información de los nodos que estén interviniendo en la comunicación. Para poder descubrir una ruta en AODV el nodo fuente envía un mensaje RREQ (*Route Request*) a sus nodos vecinos, y

estos nodos vecinos van a difundir este mensaje de RREQ a otros nodos en la red, el proceso de difusión del mensaje RREQ continua hasta que el mensaje se recibe por el nodo destino o un nodo que puede brindar la ruta para llegar al destino; entonces el nodo que recibe el mensaje de RREQ va a responder con un mensaje RREP (*Route Reply*), este mensaje va a ser enviado al nodo que originó el mensaje de RREQ. AODV ofrece la adaptación rápida a las condiciones dinámicas del enlace. Utiliza números de serie del destino para asegurar que siempre este libre de ciclos, evitando los problemas asociados a protocolos clásicos del vector de la distancia (tales como "cuenta al infinito").

Los nodos que no estén participando en la comunicación no intercambian tablas ni deben por lo tanto mantener las rutas, es decir en AODV las rutas a los destinos permanecen en memoria mientras se utilicen; en el momento que dejan de ser útiles se inhabilitan. Cuando ocurre una fractura en algún enlace de la red, permite que los nodos móviles respondan a esto y cambien de topología de la red de una manera oportuna, en AODV en el momento en el que algún enlace se rompe el sistema notifica a los nodos de modo que estos puedan invalidar las rutas para que ya no usen el enlace perdido [Perkins y Belding-Royer, 2003].

II.4.3.2.2 DSR

DSR (por sus siglas en inglés - *Dynamic Source Routing*) es un protocolo de enrutamiento dinámico de la fuente, este protocolo fue diseñado específicamente para las

redes móviles Ad-Hoc, es un protocolo simple. Este protocolo trabaja con dos mecanismos: "descubrimiento de la ruta" y "mantenimiento de la ruta". Los dos mecanismos mencionados trabajan de manera conjunta para permitir que los nodos que integran la red descubran y mantengan las rutas a los destinos arbitrarios de la red. Cada nodo mantiene una tabla, en la cual se encuentran almacenadas las entradas que incluyen el destino y los nodos involucrados para llegar a tal, dichas tablas son actualizadas en medida que se aprenden más rutas, debido a que se almacenan tablas de los puntos de acceso de la red, hace que la sobrecarga (*overhead*) se disminuya, el exceso de sobrecarga es provocada cada vez que se descubren nuevas rutas, al momento de descubrir una ruta, se pueden tener más rutas a un destino. Este protocolo funciona por demanda, es decir si un nodo fuente desea mandar información a un nodo destino, el nodo fuente verifica si cuenta en su tabla con la ruta al nodo destino, si cuenta con ella la utiliza, si no la tiene hará una búsqueda hasta que encuentra una ruta válida. En DSR se puede permitir múltiples rutas a un destino además de permitir que los nodos fuente escojan y controlen rutas usadas para enrutar sus paquetes.

Este protocolo tiene gran eficiente en la recuperación rápida de las rutas, cuando la red cambia [Johnson *et al.*, 2004].

II.4.3.2.3 TORA

TORA (por sus siglas en inglés - *Temporally Ordered Routing Algorithm*) es un protocolo basado en el algoritmo *Link Reversal Routing*, su objetivo es minimizar las cargas en la red, basándose en mantener un grafo dirigido y que no contenga ciclos para cada destino. Utilizando este protocolo los nodos pueden tener múltiples rutas hacia los diferentes destinos, este protocolo detecta las particiones existentes en la red y elimina las rutas invalidas dentro de un tiempo finito, TORA no elige siempre el camino más óptimo que existe, pero cuenta con la ventaja de que es un algoritmo muy eficiente y no satura a la red con tráfico excesivo. Es un protocolo libre de ciclos que provee múltiples caminos al destino, se adapta fácilmente a los cambios de topología en la red, este protocolo minimiza la sobre carga de comunicación [*Park y Corson, 1998*].

II.5 Resumen

En este capítulo se presenta información referente a las redes inalámbricas, sus estándares, características más importantes así como el tipo de redes inalámbricas que existen.

Se analizó a mayor detalle las redes móviles Ad-Hoc, sus características principales, escenarios de uso, se hizo un enfoque más profundo a lo referente al enrutamiento, se analizó el tipo de protocolos de enrutamiento para las redes Ad-Hoc. Los protocolos de

enrutamiento se dividen en protocolos proactivos, que tienen sus rutas actualizadas en todo momento ya que las rutas son actualizadas de manera periódica aunque no se este haciendo uso de ellas y protocolos reactivos que únicamente tiene las rutas de las cuales se esta haciendo uso y las rutas son descubiertas únicamente cuando son solicitadas.

Tanto los protocolos proactivos, como los reactivos, tienen sus ventajas y sus desventajas, por lo tanto no se puede decir que un protocolo es mejor que otro ya que va a depender del escenario donde esté trabajado. Ningún protocolo es mejor en todos los escenarios que se presenten, todo esto va a depender de varios factores como la movilidad, la carga de la red, el diámetro de la red, entre otros.

En el siguiente capítulo se describe con mayor detalle el protocolo OLSR.

Capítulo III Protocolo OLSR

III.1 Introducción

El protocolo OLSR (por sus siglas en inglés - *Optimized Link State Protocol*) es un protocolo de enrutamiento para las redes móviles Ad-Hoc estudiado por el grupo de trabajo de las MANET dentro del IETF (por sus siglas en inglés - *Internet Engineering Task Force*). OLSR pertenece a un grupo de los protocolos para MANETs que están definidos con RFC (por sus siglas en inglés - *Request For Comments*), en este caso OLSR se define en el RFC 3626 [Clausen y Jacquet, 2003].

OLSR es un protocolo proactivo como se mencionó anteriormente, esta basado en una optimización del clásico protocolo de estado del enlace puro como el OSPF (por sus siglas en inglés - *Open Shortest Path First*) definido en el RFC 1583 [Moy, 1994]. Al ser OLSR un protocolo proactivo sus rutas se encuentran disponibles en todo momento, para este fin OLSR realiza intercambio de información referente a la topología con sus nodos vecinos. Para llevar a cabo la actualización y el descubrimiento de las rutas existentes en la topología, OLSR se auxilia de dos tipos de mensajes: los mensajes HELLO y los mensajes TC (por sus siglas en inglés - *Topology Control*).

OLSR reduce la cantidad de mensajes de control que se difunden en la red mediante el uso de los MPR (por sus siglas en inglés - *Multipoint Relay*), el cual se explica en la siguiente sección.

Utilizando los MPR los nodos móviles reducen la cantidad de tráfico de control enviada en la red Ad-Hoc, para que funcione de manera adecuada cada nodo debe seleccionar un conjunto de vecinos como MPR, haciendo uso de los MPRs los mensajes de control se reducen al mínimo ya que solo los MPRs van a ser los nodos encargados de retransmitir los mensajes de control de la topología a través de toda la red Ad-Hoc, de esta forma la sobrecarga en la red se reduce de manera significativa, comparado con un mecanismo puro de inundamiento de mensajes [Qayyum *et al.*, 2002], donde cada nodo retransmite mensajes de control a través de la red. Cada nodo MPR va a ser seleccionado por un conjunto de nodos vecinos que van a declarar a este nodo como un nodo MPR, y al conjunto de nodos vecinos que declaren a un nodo como MPR se les denomina selectores MPR.

El protocolo OLSR trabaja de forma distribuida, es decir no necesita que exista una administración centralizada además de contar con una información de control actualizada ya que de manera periódica se están enviando mensajes a través de la red, por los nodos MPR, los cuales son los encargados de realizar esta tarea.

III.2 Formato del paquete OLSR

La Figura 3 muestra el formato del paquete OLSR, el cual se describe en la siguiente sección, así como también cada uno de sus campos.

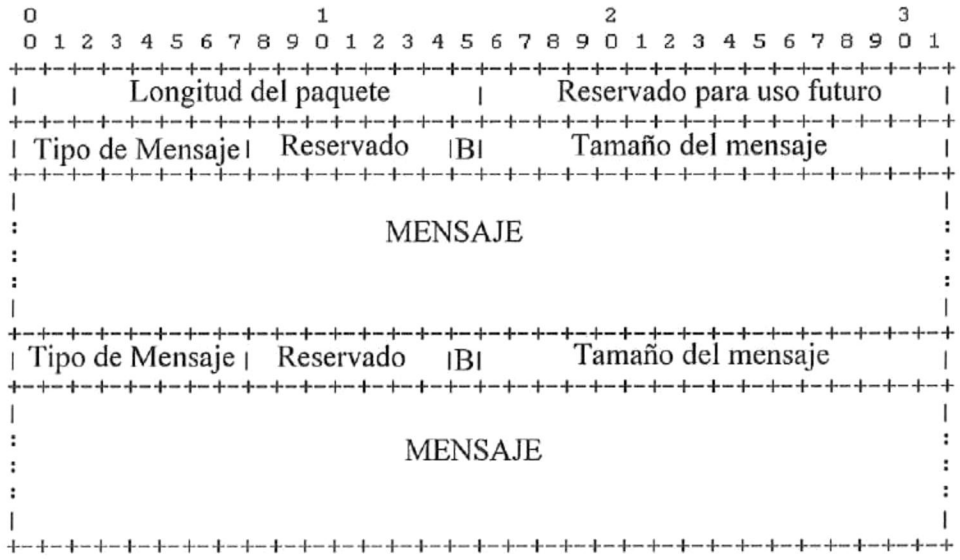


Figura 3.- Formato del paquete OLSR.

El encabezado del paquete OLSR cuenta con los campos:

- *Longitud del paquete [16 bits]:* este campo define la longitud del paquete (en bytes).

- *Reservado para uso futuro [16 bits]:* debe ser fijado a ' 0000000000000000' para conformidad del RFC 3626.

La información de la cabecera de este paquete es equivalente a la información de la cabecera obtenida de UDP.

El encabezado del mensaje OLSR cuenta con los campos siguientes:

- *Tipo de Mensaje [8 bits]:* este campo indica que tipo de mensaje debe ser encontrado en la parte de "MENSAJE". Los mensajes en el rango del 0-127 son mensajes reservados.

- *Reservado [7 bits]:* debe ser fijado a ' 0000000 ' para conformidad con el RFC 3626.

- *B [1 bit]:* Este campo indica a un nodo si el mensaje es para difusión en la red entera. Esto permite a un nodo retransmitir el mensaje correctamente, incluso si no reconoce el "tipo de mensaje".

- *Tamaño del mensaje [16 bits]:* en este campo se indica cual es el tamaño del mensaje en bytes y se mide desde el inicio del campo "Tipo de Mensaje" hasta el inicio del siguiente campo "Tipo de Mensaje", o si no hay más mensajes hasta el fin del paquete OLSR.

III.3 Descubrimiento de vecinos en OLSR

Para el descubrimiento de vecinos OLSR lo realiza mediante la utilización del mensaje HELLO. Los mensajes HELLO se utilizan para detectar a los vecinos y sus enlaces, a continuación se explican con detalle el funcionamiento del mensaje HELLO.

III.3.1 Mensaje HELLO

El protocolo OLSR esta basado en la detección de vecinos a un salto, es decir cada nodo debe conocer a los vecinos que tiene a un salto así como conocer el tipo de enlace que tiene con cada uno de ellos. Para que los nodos conozcan a sus vecinos utilizan la difusión del mensaje HELLO. Este tipo de mensajes son generados por todos y cada uno de los nodos que conformen la red. Gracias a los mensajes HELLO los nodos en OLSR conocen a sus vecinos, el mensaje HELLO es transmitido a todos los nodos a un salto, pero no son retransmitidos por los nodos que reciben este mensaje, la difusión del mensaje HELLO se puede apreciar en la Figura 4.

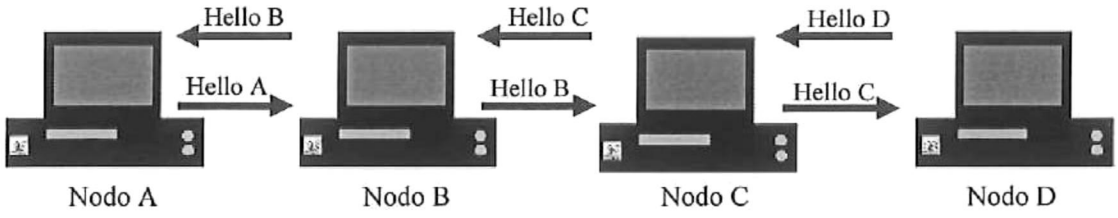


Figura 4.- Difusión del mensaje HELLO.

Los mensajes HELLO también contienen la información sobre el dirección IP de todos los vecinos a un salto del nodo que transmite dicho mensaje, es decir, la lista de vecinos divulgada por cada mensaje HELLO corresponderá a los vecinos a un salto de cada nodo móvil, así como el tipo de enlace, de los nodos; el tipo de enlace se puede declarar como: enlace simétrico, enlace asimétrico, enlace perdido o enlace MPR.

Enlace simétrico: este tipo de enlace indica que el enlace a un nodo se ha verificado como un enlace bidireccional, es decir que los datos se pueden transmitir en ambas direcciones.

Enlace asimétrico: a este enlace también se le conoce como "escuchado", esto es porque el nodo puede oír el mensaje de HELLO de un vecino, sin embargo el nodo que "escucho" el mensaje de HELLO no puede verificar que el vecino pueda recibir mensajes del nodo que "escucha".

Enlace MPR: este tipo de enlace es intrínsicamente un enlace simétrico, cuando se declara este enlace indica que el nodo ha sido seleccionado como un nodo MPR por el nodo emisor del mensaje HELLO.

Enlace Perdido: este tipo se indica cuando el enlace se ha perdido, es decir cuando la conexión entre los nodos ya no existe.

El mensaje HELLO tiene tres tareas fundamentales:

- Sensado de enlaces: esta tarea se encarga de verificar el tipo de enlace que se tiene entre los nodos de la red.

- Sensado de Vecinos: el sensado de vecinos tiene la tarea de descubrir cuales son los vecinos a un salto, así como los vecinos a dos saltos de cada nodo.

- Sensado de vecinos MPR: el sensado de vecinos MPR tiene la tarea de descubrir cuales son los vecinos que se han declarado como MPRs.

Cada mensaje HELLO tiene la siguiente información dentro de sus tablas:

- Una lista con la información sobre las direcciones de todos los vecinos a un salto en los cuales el enlace se ha declarado como simétrico.

- Una lista con la información sobre las direcciones de todos los vecinos a un salto en los cuales el enlace se ha declarado como asimétrico o "escuchado".

- Una lista con la información sobre las direcciones de todos los vecinos que han sido declarados como MPR.

Todos los nodos vecinos a un salto son citados por lo menos una vez dentro de un periodo determinado denominado HELLO_INTERVAL, en el mensaje de HELLO. Esto es debido a que la lista de vecinos de un mensaje HELLO puede ser parcial, (e.g. debido a las limitaciones del tamaño del mensaje, impuestas por la red), por eso la regla es que al menos una vez los nodos sean mencionados en dicho mensaje.

En la siguiente sección se define el formato del mensaje HELLO.

III.3.2 Formato del mensaje HELLO

La Figura 5 muestra los campos del mensaje HELLO.

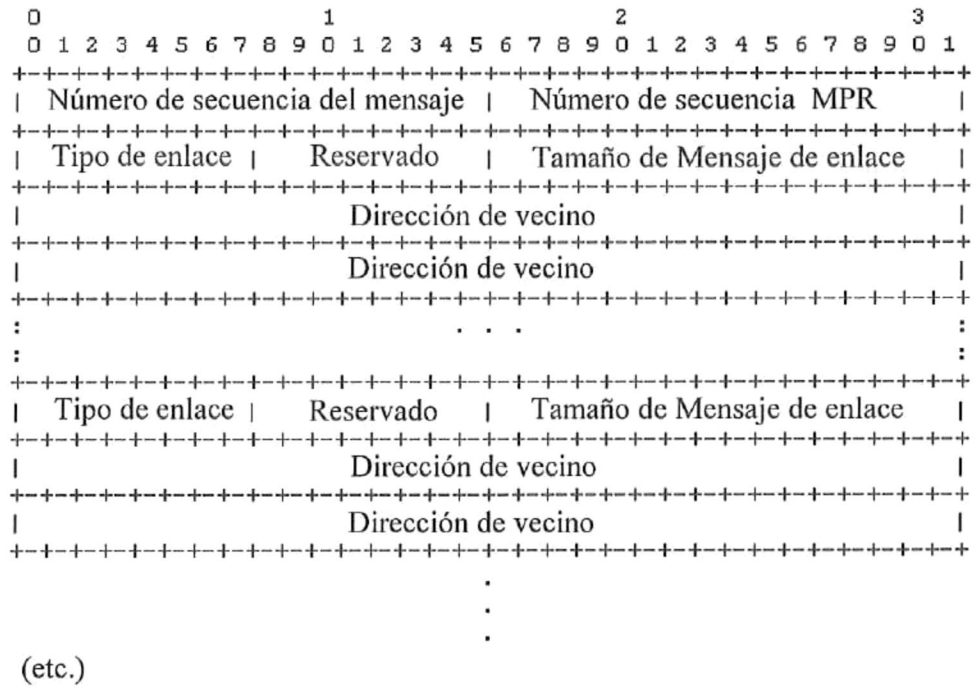


Figura 5.- Formato del mensaje HELLO.

Número de Secuencia del Mensaje [16 bits]: Mientras el mensaje de HELLO es generado, el nodo asignará un número de identificación único a este mensaje. Este número se pone en el campo número de secuencia del mensaje. El número de secuencia será diferente para todos los mensajes originados por este nodo.

Número de Secuencia MPR [16 bits]: este campo indica el número de secuencia correspondiente al sistema de MPR más reciente, calculado por el nodo remitente.

Tipo de Enlace [8 bits]: este campo especifica el tipo de enlace que tiene el nodo emisor con vecinos en su lista. OLSR requiere como mínimo los siguientes tres tipos de enlace:

- ***ASYM_LINK***: indica que los enlaces entre el nodo emisor y sus vecinos son asimétricos, no puede tener una comunicación bidireccional con ellos, solo los puede "escuchar".

- ***SYM_LINK***: indica que los enlaces entre el nodo emisor y sus vecinos son de tipo simétrico, es decir tiene una comunicación bidireccional con ellos.

- ***MPR_LINK***: indica que los nodos en la lista han sido seleccionados por el nodo emisor como MPR. (Esto implica que los enlaces entre en nodo emisor y los nodos en la lista son simétricos, es decir bidireccionales).

Es posible proporcionar información adicional especificando los enlaces adicionales como por ejemplo:

- ***LOST_LINK***: indica que el enlace entre el nodo emisor y sus vecinos se ha perdido.

Acerca de otro tipo de enlaces que no se mencionan aquí, el nodo también los procesará.

Reservado [8 bits]: este campo esta reservado para uso futuro, y se debe fijar en '00000000'.

Tamaño del mensaje [8 bits]: este campo define el tamaño del mensaje de enlace, el cual es medido desde el campo "tipo de enlace" hasta el siguiente campo "tipo de enlace", o si no existe otro campo "tipo de enlace" en el mensaje HELLO, entonces es medido hasta el final del mensaje HELLO.

Dirección de Vecino [32 bits]: este campo define la lista de vecinos que ya se han etiquetado con un "Código enlace" específico.

III.3.3 Proceso del mensaje HELLO

Los mensajes HELLO permiten que cada nodo pueda adquirir información sobre sus vecinos que se encuentran hasta una distancia de dos saltos. Un nodo mantiene una tabla de vecinos en la cual registra la información (obtenida de los mensajes HELLO) sobre sus vecinos a un salto, el estado de sus enlaces con estos vecinos, así como una lista de vecinos a dos saltos y que vecinos les dan acceso para poder llegar a los vecinos a dos saltos. La información se registra en una tabla, con los campos siguientes. En la Tabla 1 se puede observar como es registrada esta información.

Tabla 1.- Campos de la tabla de los mensajes HELLO.

1.	N_addr	N_status	N_2hop_list	N_time
2.	N_addr	N_status	N_2hop_list	N_time
3.	,,	,,	,,	,,

(etc.)

La información registrada en la tabla consiste de N_addr, N_status, N_2hop_list, y N_time. Esto quiere decir que el nodo con la dirección N_addr es un vecino a un salto de este nodo, el estado del enlace entre ellos es N_status, y que este vecino proporciona el acceso a el vecino a dos saltos N_2hop_list. El estado del enlace, N_status puede ser ASYM_LINK, SYM_LINK o MPR_LINK. Un estado del acoplamiento de MPR_LINK implica que el acoplamiento con el nodo vecino N_addr es simétrico y el nodo N_addr es seleccionado como MPR por este nodo. Cuando el N_time, expire debe ser eliminado el registro de la tabla de vecinos.

El nodo también contiene un número de secuencia N_MPR_seq. Esto especifica que el nodo ha seleccionado su MPR más reciente y este se encuentra establecido con el número de serie N_MPR_seq. Cada vez que un nodo selecciona o pone al día su sistema de MPRs, N_MPR_seq se incrementa en uno.

III.4 Mecanismos de descubrimiento de la topología en OLSR

OLSR para el descubrimiento de la topología utiliza el mensaje TC. Los mensajes TC brindarán información acerca de la topología de toda la red. Estos mensajes son generados por nodos especializados para llevar a cabo esta tarea, pero la información generada en este mensaje es utilizada por todos los nodos de la red. A continuación se explica el mensaje TC.

III.4.1 Mensaje TC

OLSR utiliza un mensaje denominado mensaje TC como se mencionó anteriormente, el cual tiene la función de construir una base de datos con información sobre la topología, la cual es necesaria para el cálculo de la tabla de enrutamiento de paquetes en la red. Este tipo de mensajes no es difundido por todos los nodos de la red, existen nodos especializados para llevar a cabo esta tarea, a estos nodos se les conoce como nodos MPR.

Un mensaje TC es enviado por un nodo MPR a los demás nodos en la red para dar a conocer la lista de vecinos que han seleccionado a este nodo como MPR, a los nodos que seleccionan a un nodo como MPR, se les conoce como nodos selectores de MPR.

La lista de direcciones puede ser parcial en cada mensaje TC, sin embargo dentro de cierto tiempo TC_INTERVAL, la información debe estar completa, es decir al momento de recibir todos los mensajes TC se hace un análisis de dichos mensajes TC, y se describirá todo el sistema de nodos MPR, para tener la lista completa de los nodos selectores del nodo MPR.

Cuando un nodo tiene un conjunto de nodos selectores de MPR vacío, significa que ningún nodo lo ha seleccionado como MPR, en este caso no se debe generar mensajes TC, únicamente son los nodos MPR, los encargados de generar y difundir los mensajes TC.

Como se ejemplifica en la Figura 6:

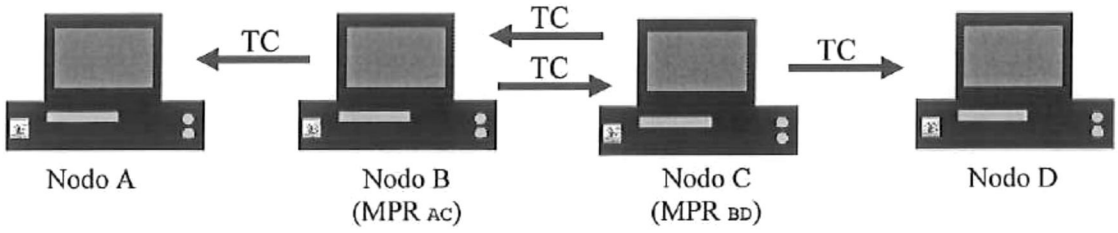


Figura 6.- Difusión del mensaje TC.

III.4.2 Formato del mensaje TC

La Figura 7 muestra los campos del mensaje TC, los cuales se describen a continuación.

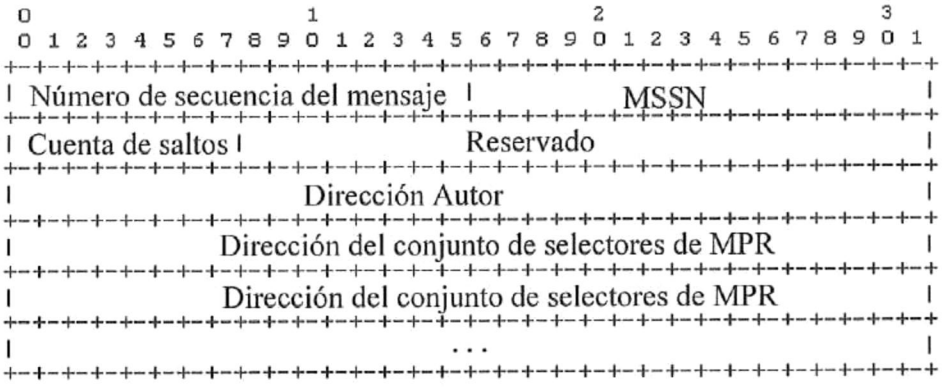


Figura 7.- Formato del Mensaje TC.

Número de secuencia del mensaje [16 bits]: Mientras que se genera el mensaje TC, el nodo emisor, asignará un número de identificación único a este mensaje, y pondrá este número en el campo de número de secuencia del mensaje. Este número de secuencia será diferente para todos los mensajes originados por este nodo, y se usará para reconocer la recepción duplicada del mensaje.

Número de secuencia del selector de MPR (MSSN) [16 bits]: Un número de secuencia se asocia a un sistema de selector de MPR. Cada vez que un nodo detecta un cambio en su sistema de MPRs, este incrementa dicho número de secuencia. Este número

se envía en el campo de MSSN del mensaje TC para no perder de vista la información más reciente. Cuando un nodo recibe un mensaje TC, puede decidir en base de este número de secuencia, si la información recibida sobre los nodos selectores de MPR es más reciente que la que ya tiene.

Cuenta de saltos [8 bits]: Este campo tiene el número máximo de saltos que un mensaje TC puede dar. Cada vez que el mensaje TC se retransmite, este campo es decrementado por 1. Cuando este campo llega a ser cero, el mensaje TC no es retransmitido y es descartado.

Reservado [24 bits]: este campo es reservado para uso futuro, es fijado a '000000000000000000000000'.

Dirección de Autor [32 bits]: este campo contiene la dirección IP del nodo que generó el mensaje TC. Este campo no se debe confundir con la dirección fuente del encabezado UDP, la dirección de la fuente de encabezado UDP cambia cada vez que el paquete OLSR es retransmitido por un nodo intermedio. Sin embargo la dirección del autor no cambia nunca en las retransmisiones.

Dirección del conjunto de selectores de MPR [32 bits]: este campo contiene la o las direcciones de los nodos que conforman al conjunto de nodos selectores del nodo MPR. Todas las direcciones de los nodos selectores de MPR deben ser notificadas en el mensaje TC. Si el mensaje TC alcanza el tamaño máximo permitido por la red y todavía no se han

enviado todas las direcciones del sistema entero de selectores MPR, se seguirán generando mensajes TC, hasta que se hayan terminado de mandar todas.

III.4.3 Proceso del mensaje TC

En el protocolo OLSR, los mensajes TC son difundidos y retransmitidos por los MPRs en toda la red. En este proceso, un nodo puede recibir el mismo mensaje TC más de una vez. Para evitar que se vuelva a procesar el mismo mensaje TC que fue recibido anteriormente, cada nodo mantiene una tabla duplicada. Esta tabla contiene la información de los mensajes TC más recientes. Así que cuando llega un mensaje TC, el nodo verifica en su tabla duplicada para revisar si el mensaje que se acaba de recibir ya se ha recibido antes o es en todo caso es un mensaje TC nuevo. Si encuentra que ya ha recibido ese mismo mensaje, lo desecha. Si no, registra una nueva entrada con el mensajes en la tabla duplicada, entonces el mensaje TC nuevo es registrado y procesado.

Cada nodo que forme parte de la red, mantiene una tabla de la topología, en la cual registra información sobre la topología de la red que ha obtenido en los mensajes TC. De acuerdo con esta información, se calcula la tabla de enrutamiento. El formato de la información de la topología se puede observar en la Tabla 2:

Tabla 2.- Campos de la Tabla del mensaje TC.

1.	T_dest	T_last	T_seq	T_time
2.	T_dest	T_last	T_seq	T_time
3.	„	„	„	„

(etc.)

Cada entrada en la tabla tiene T_dest, T_last, T_seq, y T_time, que especifica que el nodo T_dest ha seleccionado el nodo T_last como MPR y que el nodo T_last ha anunciado la información de su selector de MPR con el número de serie T_seq. Por lo tanto, el nodo T_dest pude ser alcanzado mediante el nodo T_last. Cada entrada en la tabla de topología tiene un tiempo asociado T_time, cuando T_time ya no es válido deber ser removida.

Entonces se tiene que:

- T_dest es la dirección del nodo selector de MPR.
- T_last es la dirección del nodo que origino el Mensaje TC.
- T_seq es el número de serie (MSSN) del mensaje TC recibido.

T_time es el tiempo asociado a los registros antes de ser removidos TOP_HOLD_TIME.

III.5 Multipuntos de retransmisión (MPR)

La idea principal del uso de los MPR es la de reducir la sobrecarga de los mensajes de control, reduciendo la información redundante. Cada nodo N en la red selecciona un conjunto de nodos a un salto y con los cuales tiene un enlace simétrico, estos nodos hacen una retransmisión de los mensajes de control. Al conjunto de nodos a un salto seleccionados por un nodo N para tener acceso a los nodos que estén a dos saltos se les llama conjunto de Multipuntos de Retransmisión (MPR) de un nodo. Los vecinos del nodo N que no son parte del conjunto de MPR, reciben y procesan los mensajes de difusión pero no los retransmiten.

Cada nodo selecciona a sus MPRs del conjunto de vecinos a un salto con los cuales se tenga un enlace simétrico. Este conjunto de nodos son seleccionados de tal manera que cubran (en términos de conectividad) a todos los nodos simétricos a dos saltos. El conjunto de nodos MPR del nodo N, es denotado como MPR(N), es un subconjunto de nodos simétricos a un salto del nodo N que satisface las siguientes condiciones: cada nodo en el vecindario a dos saltos de N debe tener un enlace simétrico hacia MPR(N). Cuando más pequeño es el sistema de MPR, es menor la sobrecarga de tráfico de control resultante del protocolo de enrutamiento, esto quiere decir que la sobrecarga de tráfico de control es proporcional al tamaño del conjunto de nodos MPR, entre más pequeño sea el conjunto de nodos MPR es más óptimo el protocolo.

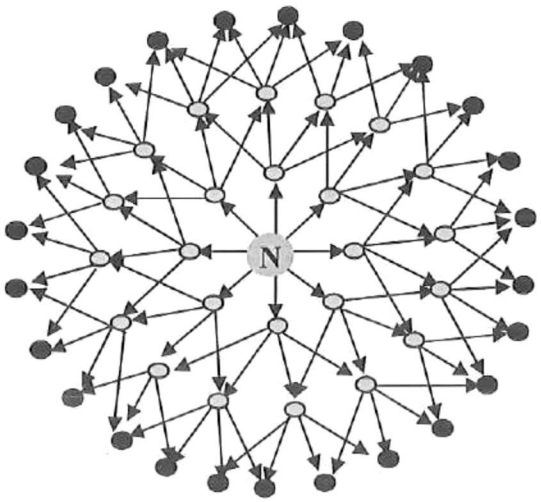
Cada nodo mantiene información sobre su sistema de vecinos. Los nodos conocen a su sistema de vecinos gracias a los mensajes HELLO que reciben de sus vecinos a un salto.

La inundación de mensajes de control a través de la red permite que las rutas de acceso a los nodos sean actuales y confiables, ya que de forma constante se están enviando este tipo de mensajes para tener las rutas actualizada, lo cual hace que la red sumamente confiable, sin embargo estos mensajes consumen gran cantidad de ancho de banda, cuestión que no es muy conveniente para las redes actuales, pero gracias al uso de MPRs se produce una menor cantidad de tráfico de control lo cual proporciona muy buenos resultados. Uno de los principales beneficios con el uso de los MPRs es la reducción de la sobrecarga producida por la información de control de topología, está información es reducida cuando el número de nodos que la reenvían están limitados a unos cuantos es decir a los nodos MPR, y no es reenviada por todos los nodos de la red.

- Los multipuntos de retransmisión del nodo X son sus vecinos a un salto y con enlace simétrico que proporcionan conectividad con los vecinos a dos saltos de distancia del nodo X.

- Cada nodo transmite su lista de vecinos en mensajes periódicos HELLO, de forma que todos los nodos puedan saber cuales son sus vecinos a un salto y a dos saltos, con los cuales se tenga un enlace simétrico, para de esta forma poder seleccionar a sus multipuntos de retransmisión.

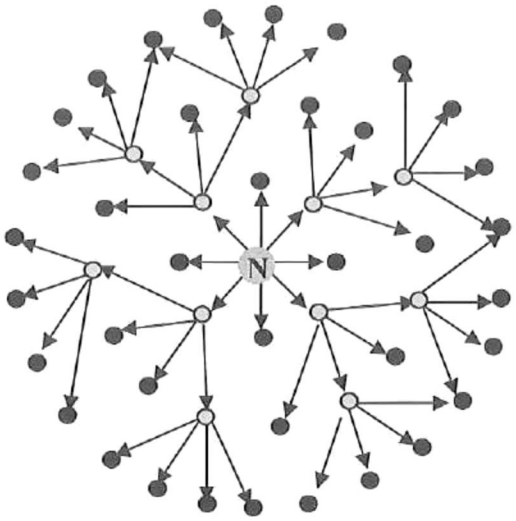
En el ejemplo de la Figura 8 y la Figura 9 se desea hacer una difusión de un mensaje a 3 saltos de distancia, entonces cuando no se utilizan los nodos MPRs se tendrían 24 retransmisiones para que el mensaje pueda ser difundido desde el nodo N hasta algún nodo que este a 3 saltos de distancia del nodo N, en cambio si se hace uso de nodos MPRs únicamente se necesitarían 11 retransmisiones para difundir el mensaje a 3 saltos de distancia [Qayyum et al., 2002].



Son necesarias 24 retransmisiones para difundir el mensaje, hasta tres saltos de distancia, desde el nodo que origina el paquete, nodo N.

○ Nodo que retransmite

Figura 8.- Difusión de un mensaje a 3 saltos sin uso de MPR.



Son necesarias únicamente 11 retransmisiones para difundir el mensaje, hasta tres saltos de distancia, desde el nodo que origina el paquete, nodo N.

○ Nodo que retransmite

Figura 9.- Difusión de un mensaje a 3 saltos utilizando los nodos MPR.

OLSR hace uso de la selección de los multipuntos de retransmisión (MPRs), y cálcula las rutas a sus destinos utilizando estos nodos. Es decir los nodos MPR se seleccionan como nodos intermedios con los cuales un nodo fuente puede llegar a su destino, esto ocurre cuando el destino no este a un salto del nodo fuente. Para que este tipo de mecanismo funcione cada nodo debe enviar periódicamente mensajes de control, para describir la información de sus vecinos, cuando se trate de un nodo MPR este debe mandar información de sus vecinos acerca de que nodos lo han seleccionado como un nodo MPR.

Los MPRs se seleccionan entre los vecinos a un salto de un nodo X, con los cuales tenga un enlace simétrico y bidireccional. Por lo tanto, la selección de una ruta a través de los multipuntos de retransmisión evita automáticamente problemas asociados a la

transferencia de paquetes de datos en los enlaces unidireccionales, tal como el problema de conseguir un reconocimiento para los paquetes de datos en cada salto.

III.5.1 Selección de los multipuntos de retransmisión

El sistema de MPR se debe calcular de tal forma que un nodo X pueda alcanzar a todos sus vecinos a dos saltos a través de los nodos MPR, cuando se mandan mensajes de difusión a través de la red todos los nodos reciben esa información sin embargo solo es retransmitida por los nodos MPR. Es esencial que todos los vecinos a dos saltos puedan ser alcanzados con los nodos seleccionados como MPR, no existe una cantidad mínima o máxima de nodos MPR, eso dependerá de las necesidades de la red, entre más MPR existan mayor será el número de mensajes de control que se difundan en la red y por consecuencia es menor eficiencia del algoritmo de enrutamiento.

El mecanismo que se describe específica una propuesta heurística para la selección de nodos MPR. La terminología siguiente será utilizada en describir este algoritmo:

X: Nodo que hará la selección de nodos MPR.

N: Conjunto de vecinos a un salto de un nodo (X) con quienes existe un enlace simétrico.

N2: Conjunto de vecinos a dos saltos de un nodo (X). Este conjunto no contiene a ningún vecino a un salto.

Para la selección de nodos MPR se tiene una propuesta heurística y se define a continuación:

1. Comienza con un sistema vacío de MPR.

2. Selecciona como MPRs a aquellos nodos en N que proporcionan una única trayectoria de comunicación con algunos nodos en N2, y se agregan estos nodos de N al sistema de MPRs.

3. Mientras existan nodos en N2 que no son cubiertos por el sistema de MPRs:

    3.1 Para cada nodo en N que no este en el sistema de MPRs, cálcula el número de nodos en N2 que todavía no son cubiertos por los nodos MPRs y que son accesibles a través de este vecino de un salto.

    3.2 Selecciona como MPR al nodo en N que alcance el número máximo de nodos no cubiertos en N2.

La Figura 10 ejemplifica el procedimiento anterior.

Cuando un nodo ya ha seleccionado a sus nodos MPRs entre sus nodos vecinos a un salto con los cuales tiene un enlace simétrico, el estado de los enlaces que tiene el nodo selector de MPRs con sus MPRs, debe ser cambiado de ser un enlace simétrico (SYM_LINK) a un enlace MPR (MPR_LINK) en la tabla de vecinos. El valor de MPR_Seq_Num en la tabla de vecinos es incrementado en uno.

Se recalcula el sistema de MPR cuando:

- Un cambio en la vecindad es detectado, es decir, un enlace simétrico con un vecino falla, o un nuevo nodo con enlace simétrico se ha agregado, o

- Un cambio es detectado en la vecindad a dos saltos, tal que un enlace simétrico se ha detectado o se ha perdido entre un vecino a un salto y un vecino a dos saltos.

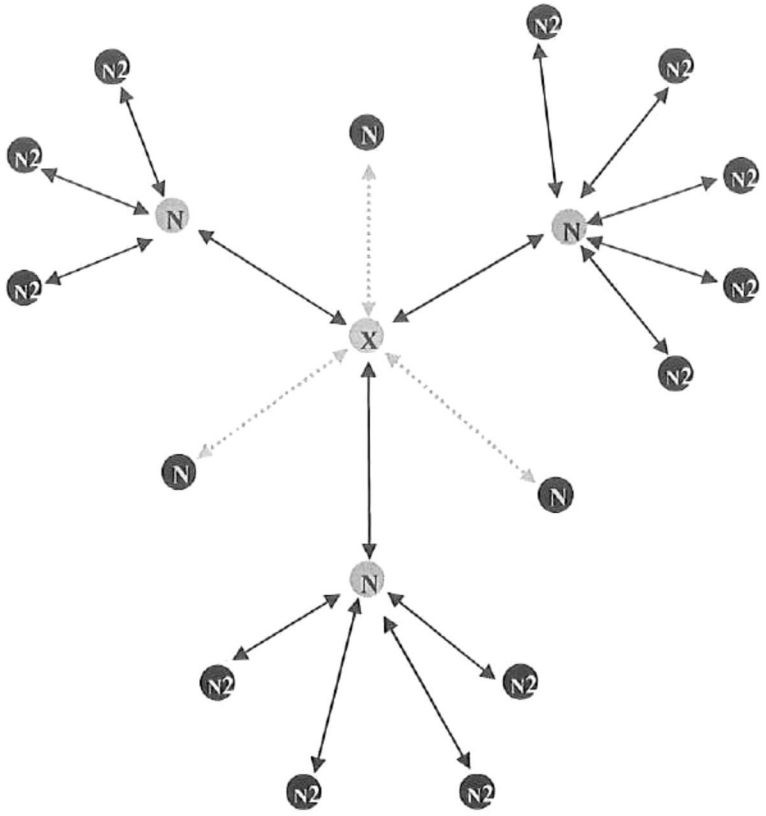


Figura 10.- Selección de los MPR.

La técnica de retransmisión de los MPRs trabaja de manera distribuida. Cada nodo calcula su propio sistema de MPRs, que es totalmente independiente de la selección que hagan otros nodos en cuanto a sus MPRs. Los nodos reaccionan cuando los nodos de su vecindario cambian y modifica por consiguiente su sistema de MPR para continuar cubriendo a sus vecinos a dos saltos de distancia [Qayyum *et al.*, 2002].

III.6 Resumen

En este capítulo se presentó la descripción del protocolo de enrutamiento OLSR, su funcionamiento y características principales.

Se explicó cuales son los mensajes que OLSR utiliza para su funcionamiento, estos mensajes son: mensaje HELLO y mensaje TC. Estos dos tipos de mensajes son fundamentales para el desempeño de este protocolo. El mensaje de HELLO es enviado cada 0.5 segundos y hará referencia a los vecinos a un salto, es decir hace un sensado de vecinos con lo cual le permite conocer cuales son los nodos que tiene como vecinos, así como el tipo de enlace que tiene con cada uno de ellos, cada nodo que recibe el mensaje de HELLO puede construir una tabla con la información sobre su sistema actual de vecinos a 1 salto, así como también, el nodo puede crear una tabla con información de sus vecinos hasta 2 saltos. Gracias al mensaje TC que es enviado cada 2 segundos se pueden actualizar las tablas de enrutamiento en la red, ya que cuando el mensaje TC es enviado este contiene la tabla de todos los nodos que lo han seleccionado como un nodo MPR al nodo que manda el mensaje TC.

Bajo el protocolo OLSR, los nodos móviles reducen la cantidad de tráfico de control, enviada por la red. Esta reducción es posible ya que la información de tráfico de control únicamente es mandada por los nodos designados como MPRs, los nodos MPRs son los encargados de la retransmisión y generación de los mensajes TC. En este capítulo

se explicó que es un nodo MPR y como es seleccionado por los nodos selectores de MPR.

En el capítulo siguiente se habla sobre el protocolo HOLSR.

Capítulo IV Protocolo HOLSR

IV.1 Introducción

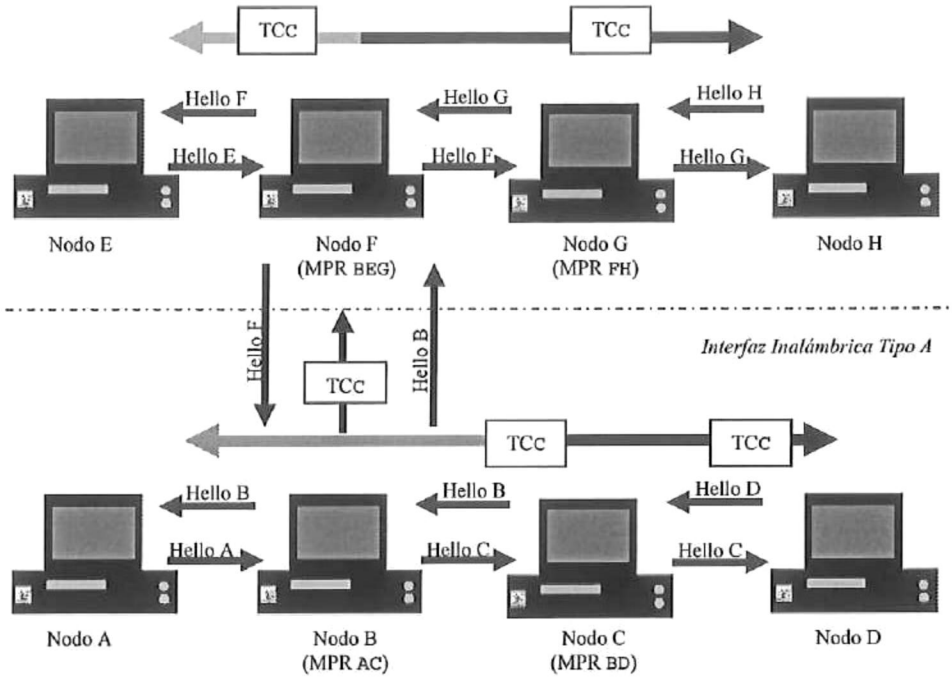
La mayor parte de los protocolos de enrutamiento existentes para las redes móviles Ad-Hoc son para redes homogéneas, esto es, que todos los nodos tengan las mismas capacidades en términos de procesamiento y en términos del número de interfaces de radio. Pero actualmente existen diferentes topologías de redes heterogéneas, es por eso que es necesario que se cuente con protocolos que tengan la capacidad de poder brindar soporte a este tipo de redes. Otro aspecto muy importante es que los protocolos para las redes heterogéneas proporcionan mayor soporte de escalabilidad a las redes móviles Ad-Hoc actuales [Villaseñor-González *et al.*, 2005].

A medida que el número de nodos aumentan, las demandas de la red llegan a ser mayores, si se cuenta con un protocolo que brinde una buena escalabilidad será capaz de poder ajustar y mantener el funcionamiento de la red aún cuando el número de nodos pueda aumentar. Cuando se tiene un protocolo de enrutamiento plano, el funcionamiento de una red se tiende a degradar a medida que el número de nodos móviles aumenta, además cuando se emplea un protocolo como OLSR y existen nodos que tienen más de una interfaz de red, los mensajes TC y los mensajes HELLO van a aumentar dependiendo del número

de interfaces de red de cada nodo, ya que estos mensajes van a ser enviados por todas sus interfaces sin ninguna consideración [Villaseñor-González *et al.*, 2005].

IV.2 OLSR

El protocolo OLSR tiene la capacidad de poder trabajar con nodos que tengan más de una interfaz, pero el problema es que como OLSR es un protocolo plano por naturaleza, los mensajes HELLO y los mensajes TC son enviados a través de todas las interfaces de red de los nodos, provocando así que la red se sobrecargue de mensajes HELLO y de mensajes TC, teniendo de esta forma un desempeño inapropiado del protocolo de red. Como se puede ver en laFigura 11, el nodo B tiene dos interfaces de red una de ellas la usa para comunicarse con los nodos que están en la interfaz inalámbrica tipo A y la otra interfaz de red la usa para comunicarse con los nodos que están en la interfaz inalámbrica tipo B, cuando el nodo C manda un mensaje TC este es enviado a los demás nodos de la red y es retransmitido por el nodo B en todos sus interfaces de red, el nodo F que esta en la interfaz inalámbrica B recibe el mensaje TCc del nodo B, entonces el mensaje TC inunda a la red de mensajes, es decir que el mensaje TCc será conocido por los nodos E, F ,G y H, a pesar de que estos nodos se encuentran en una interfaz inalámbrica diferente del nodo C, que fue el que generó el mensaje TCc. Entonces el protocolo OLSR plano da lugar a que se tenga una gran cantidad de mensajes de control propagados en toda la red, por lo que el OLSR no es un protocolo óptimo para las redes heterogéneas [Villaseñor-González *et al.*, 2005].



HELLO A : Mensaje de HELLO generado por el Nodo A.
TCC: Mensaje TC generado por el Nodo C.
MPR AC : Nodo seleccionado como MPR por los nodos A y C.

Figura 11.- Red Ad-Hoc Heterogénea con OLSR plano.

IV.3 OLSR jerárquico (HOLSR)

El protocolo HOLSR (por sus siglas en inglés - *Hierarchical Optimized Link State Protocol*) ayuda a reducir los mensajes de control en las redes móviles Ad-Hoc heterogéneas. El protocolo HOLSR es un protocolo jerárquico para redes móviles Ad-Hoc, basado en las especificaciones del algoritmo OLSR, una de las características principales del protocolo HOLSR es la capacidad de reducir la cantidad de información de control de topología, debido a que organiza a los nodos en clusters y estos clusters son organizados en diferentes niveles jerárquicos. HOLSR logra una reducción en la difusión

de mensajes de control al controlar los mensajes que se intercambian en los diferentes niveles jerárquicos de la red además en HOLSR se pueden emplear nodos heterogéneos y el protocolo puede hacer un uso eficiente de nodos de gran capacidad. Otra ventaja significativa es la reducción en el cálculo de las rutas de enrutamiento, ya que si se llega a romper un enlace en una parte de la red, únicamente los nodos que se encuentren en ese cluster necesitarán recalcular las tablas de enrutamiento, y no se tendrán que recalcular todas las rutas de la red completa.

Con una estructura jerárquica en HOLSR se logra una reducción de los mensajes empleados para el control de la topología dado que ahora los mensajes de control de los nodos se restringen a un área específica, y con esto el tamaño de la tabla de enrutamiento de los nodos también es reducida. La arquitectura de red jerárquica consiste en múltiples niveles lógicos jerárquicos distintos en la topología [Villaseñor-González et al., 2005]. En la Figura 12 se puede apreciar una red con arquitectura jerárquica y con elementos de red heterogéneos.

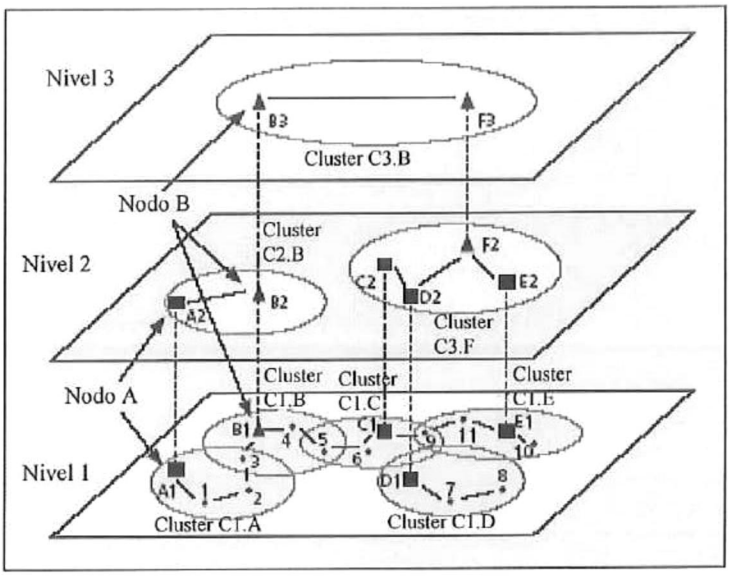


Figura 12.- Red jerárquica con elementos heterogéneos.

En la Figura 12 los nodos están organizados en múltiples niveles lógicos. Los nodos de baja capacidad, señalados por los círculos son equipados de solamente una interfaz de red, estos nodos participan en el nivel 1 de la topología, los nodos representados por los cuadrados son equipados de hasta dos interfaces de red, estos nodos pueden participar en el nivel 1 y también puede retransmitir mensajes en el nivel 2 de la topología usando una banda de frecuencia diferente a la que está usando para comunicarse con el nivel 1, la interfaz que usa para comunicarse con el nivel 2 puede proporcionar una tasa de transmisión más larga que la usada para comunicarse con los nodos del nivel 1. Los nodos que están representados por triángulos son nodos de gran capacidad que pueden estar equipados de hasta tres interfaces de red, esos nodos son capaces de comunicarse con nodos de los 3 niveles. Los nodos del nivel 2 o 3 pueden tener menos de 2 o 3 interfaces inalámbricas respectivamente, sin embargo los nodos que no sean heterogéneos de los

niveles 2 y 3, se deben equipar de al menos una interfaz de red para que se puedan comunicar con los nodos del nivel 2 y 3 respectivamente; por ejemplo en la figura Z2 el nodo F está equipado solamente de 2 interfaces a pesar de que esta en el nivel 3.

IV.4 Creación de clusters y Clusters Heads en HOLSR

El principio fundamental del protocolo HOLSR es que los nodos están organizados en clusters y estos clusters a su vez están organizados en una topología de niveles jerárquicos. Cada nivel que compone a la topología de red, puede estar compuesto de uno o más clusters, y cada uno de estos clusters va a contener un nodo que va a estar equipado con más de una interfaz de red, es decir va a tener múltiples interfaces, a este nodo se le designará como cluster head.

Para que exista una reducción en la cantidad de tráfico de control que se genera en el protocolo HOLSR es necesario que los nodos estén organizados en clusters. Organizando los nodos en clusters la difusión de los mensajes de control de la topología (TC) se limitan a una región específica. Un cluster es entonces una región que esta compuesta de nodos móviles que se encuentran en el mismo nivel jerárquico, estos nodos móviles seleccionan un cluster head, y el nodo que es seleccionado como cluster head es aquel que pueda proporcionar comunicación a un nivel lógico superior. En HOLSR los mensajes TC son transmitidos entre los nodos dentro de un cluster; mientras que para que los niveles superiores tengan la información de los miembros los cluster en los niveles inferiores, los

cluster head utilizan un mensaje jerárquico de control de la topología denominado HTC (por sus siglas en inglés - *Hierarchical Topology Control*).

El cluster head transmitirá periódicamente mensajes jerárquicos de control de la topología (HTC) para informar los cambios relacionados con los miembros del cluster. De esta manera los nodos en los niveles superiores de la topología identificarán al autor del mensaje HTC; es decir al cluster head, como el siguiente salto para alcanzar cualquiera de los miembros de ese cluster.

IV.4.1 Configuración del Cluster

En la topología jerárquica los clusters del nivel 1 están compuestos principalmente de equipo de comunicaciones que tiene una cobertura de radio limitada, al igual que en todos los niveles de la topología los nodos del nivel 1 se agruparán en clusters, en cada cluster se seleccionará un cluster head, y el cluster head brindará la capacidad de enrutamiento de paquetes a los niveles más altos de la topología y viceversa.

En la topología de red inalámbrica existe la posibilidad de que los nodos móviles detecten que existen dos o más cluster heads dentro de alcance, en estas circunstancias el nodo móvil tomará la decisión de seleccionar un cluster head. La decisión para seleccionar un cluster head va a estar basada en la evolución de una métrica de desempeño. En el caso de HOLSR la decisión de seleccionar al cluster head se basa en el número de saltos que

existen entre el nodo móvil y el cluster head; de esta forma los nodos móviles seleccionarán al cluster head más cercano.

El proceso de selección de cluster head es un aspecto muy importante dentro del protocolo HOLSR. Para estos fines se emplea el mensaje CIA (por sus siglas en inglés - *Cluster ID Announcement*) este mensaje es utilizado por todos los nodos de la red, el mensaje CIA (Aviso de Identificación de Cluster) contiene información muy relevante para la selección del Cluster Head ya que proporciona información sobre la dirección IP además de un campo que contiene la cuenta de saltos que se tienen para llegar a el Cluster Head. Los nodos móviles que implementan la funcionalidad de Cluster Head, son aquellos nodos que participan en los diferentes niveles de la topología y proporcionan enrutamiento entre los diferentes niveles lógicos, en nuestro caso los nodos que cuentan con interfaces múltiples son configurados para trabajar como Cluster Head. Los Cluster Head son configurados durante el proceso de arranque de HOLSR

Los mensajes CIA son transmitidos por los Cluster Head con una cuenta de saltos igual a cero, el nodo Cluster Head se anunciará como tal transmitiendo el mensaje CIA con su dirección IP mientras que el campo de número de saltos en el mensaje de CIA se inicializa a cero. Tales mensajes CIA son transmitidos junto con los mensajes HELLO para reducir el número de transmisiones de paquete, de esta forma los nodos vecinos pueden identificar el cluster head y unirse al cluster. Es decir los nodos móviles que reciban el mensaje de CIA tomarán la decisión de unirse al Cluser Head del cual recibieron el mensaje de CIA, y luego estos nodos retransmitirán el mensaje de CIA e informarán a

sus vecinos cual es el Cluster Head que han seleccionado y por lo tanto a que cluster pertenecen conforme los nodos van retransmitiendo el mensaje CIA, el campo de número de saltos se incrementa en uno, de esta forma los nodos conocen la distancia al cluster head. Una vez que se haya seleccionado un cluster head, el nodo comenzará a generar mensajes CIA junto con los mensajes de HELLO para informar a sus vecinos a que cluster pertenecen, e invitará a los nodos más lejanos a que se unan al cluster. La cuenta de saltos indicada en los mensajes CIA es incrementada en uno cada que un nodo acepta el mensaje CIA para unirse al Cluster Head, entonces el nodo móvil comenzará a generar mensajes de CIA con un valor de cuenta de salto igual a la cuenta registrada más uno. El uso del mensaje CIA permitirá que otros nodos móviles en HOLSR identifiquen un cluster head posible en una región dada de igual forma, los nodos pueden identificar la distancia al cluster head candidato basado en la información proporcionada en el mensaje CIA. Los nodos pueden recibir dos a más mensajes CIA, si el nodo esta en una región traslapada, sin embargo el nodo se unirá al cluster que este más cercano al nodo móvil basado en el número de saltos que tenga que dar para poder llegar al Cluser Head. El mensaje CIA debe ser el primer mensaje encapsulado en el paquete HOLSR para informar a los nodos de recepción si necesitan procesar el resto del paquete HOLSR.

Los nodos vecinos procesarán el paquete de HOLSR generado por otros nodos si cualquiera de los siguientes casos es verdadero:

1. El nodo móvil que recibe el paquete de HOLSR no ha definido un Cluster Head. Por lo tanto procesará la información del paquete de HOLSR y utilizará la información de CIA para seleccionar el Cluster Head que más le convenga.

2. El nodo móvil que recibe el paquete de HOLSR es también un miembro del mismo cluster, es decir, el nodo móvil aceptará el paquete de HOLSR si el mensaje CIA es igual al que tiene actualmente, esto es que sea del mismo Cluster Head seleccionado.

3. El nodo móvil ya tiene un cluster head registrado, sin embargo, mediante el nuevo mensaje CIA que se recibe se detecta un nuevo cluster head que esta más cercano que el cluster head registrado, en este caso el nodo se asociará al nuevo cluster head.

IV.4.1.1 Cluster traslapados

En las redes móviles Ad-Hoc, es posible concebir una situación en la cual dos o más cluster se traslapen. En este caso, los nodos móviles que están situados en las regiones traslapadas recibirán dos o más mensajes CIA. Sin embargo el nodo solo podrá seleccionar un cluster head; para la selección del cluster head, el nodo móvil se basará en el número de saltos que tendrá que dar al cluster head candidato. De esta manera, el nodo móvil seleccionará como su cluster head, al cluster head que tenga la cuenta de saltos más baja según lo divulgado en el mensaje CIA.

IV.5 Funcionamiento del protocolo HOLSR

El protocolo de HOLSR se basa en el mismo formato del paquete definido para el protocolo de OLSR. El protocolo de HOLSR y el protocolo de OLSR diferencian en algunos de los algoritmos, que se utilizan para procesar sus mensajes de administración del protocolo. El protocolo HOLSR introduce mensajes adicionales para permitir la administración de la arquitectura jerárquica.

HOLSR esta diseñado para hacer un uso más eficiente de los nodos móviles que tengan una o más interfaces (múltiples interfaces). Los nodos móviles que están equipados con múltiples interfaces son capaces de comunicarse entre los diferentes niveles de la topología. En el caso del protocolo de OLSR se especifica que un nodo debe transmitir y retransmitir todos los mensajes de control en todas sus interfaces, sin embargo el protocolo HOLSR restringirá la transmisión y la retransmisión de los mensajes de control.

IV.5.1 Formato del paquete HOLSR

El formato del paquete HOLSR se deriva del formato del paquete definido para el protocolo OLSR. La diferencia principal en el formato del paquete y la descripción de sus campos entre HOLSR y OLSR es: la introducción del campo "dirección fuente", este campo se requiere en HOLSR para identificar la dirección IP del nodo HOLSR que transmite o retransmite mensajes de HOLSR. Debe ser observado que la "dirección fuente"

no es igual a "dirección autor". La "dirección autor" corresponde a la dirección IP del nodo móvil que creó o generó el mensaje HOLSR, por otra parte la "dirección fuente" es la dirección IP de la interfaz del nodo que transmitió o retransmito el mensaje HOLSR. En la Figura 13 se ilustra el formato del paquete HOLSR.

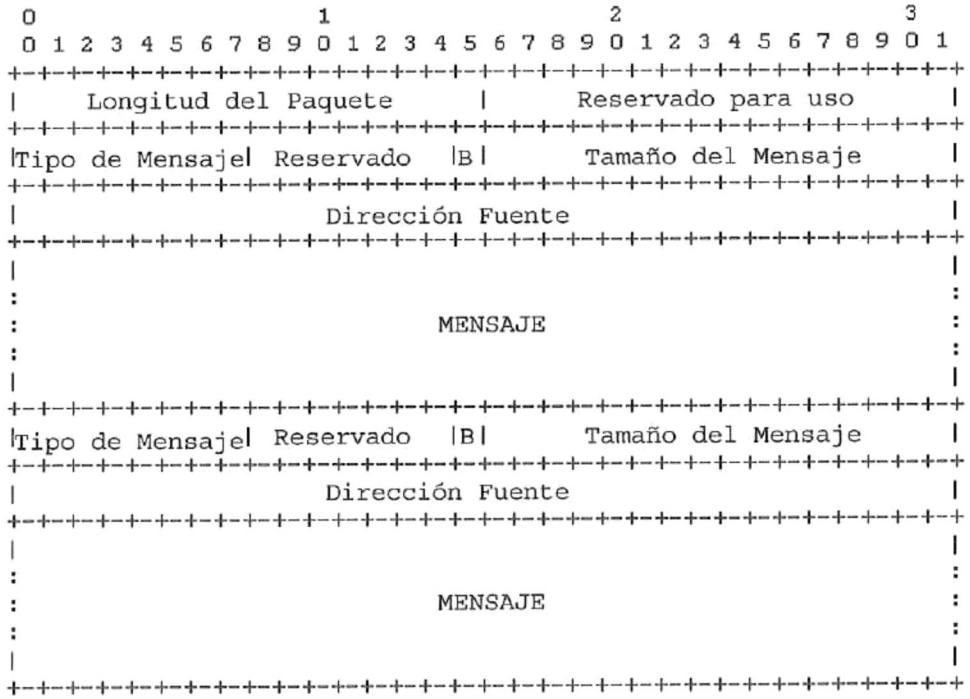


Figura 13.- Formato del paquete HOLSR.

IV.6 Descubrimiento de vecinos en HOLSR

El protocolo HOLSR utiliza la transmisión de los mensajes HELLO para detectar a sus vecinos al igual que lo hace el OLSR. Una de las características principales del protocolo HOLSR es la creación de clusters para reducir la cantidad de información de

control que se genere en la red. Por lo tanto el mecanismo de descubrimiento de vecinos en HOLSR se utiliza para controlar el descubrimiento de vecinos dentro del cluster.

En HOLSR como se mencionó anteriormente utiliza los mensajes CIA, cada cluster head transmitirá los mensajes CIA en los cuales se identifican como cluster head de dicho cluster. Así como que los mensajes CIA incluyen un campo que indica la cuenta de saltos al cluster head, al momento de que un cluster head manda el mensaje de CIA el campo de la cuenta de saltos se fija a cero. Los mensajes CIA son transmitidos junto con los mensajes de HELLO. De esta forma todos los nodos móviles vecinos pueden identificar el cluster head y unirse al cluster más cercano. Una vez que un nodo móvil se une a un cluster, el nodo comenzará a transmitir mensajes de CIA junto con los mensajes de HELLO, es así como el cluster head puede ser descubierto por nodos móviles distantes multi-hop. Debe ser observado que cada vez que un nodo móvil descubre el cluster head de algún cluster mediante los mensajes CIA, registrará una entrada con la información sobre la dirección IP del cluster head y el valor de cuenta de saltos al cluster head; entonces el nodo móvil una vez que se una a un cluster comenzará a generar mensajes CIA con un valor de cuenta de saltos igual a la cuenta de saltos registrada en el mensaje CIA recibido más uno, para que los nodos más lejanos puedan conocer que cluster head les queda más cerca, y de esta forma poder unirse a él y formar parte de dicho cluster.

Además de servir para el descubrimiento de los Cluster Heads, los mensajes CIA son utilizados por los nodos móviles para identificar si los mensajes HELLO son generados por un nodo móvil en el mismo cluster o en un cluster diferente y los mensajes HELLO son

descartados o aceptados por consiguiente. Los mensajes HELLO contienen la información sobre la dirección IP de todos los vecinos a 1 salto y de vecinos hasta 2 saltos de distancia, además del estado de su enlace. El estado de su enlace se puede declarar como "simétrico", "asimétrico", "MPR" o "perdido". Las explicaciones de cada uno de estos estados de enlace es igual a las hechas en OLSR vistas en el capítulo anterior.

Respecto al mecanismo de descubrimiento de vecinos en HOLSR, a diferencia de OLSR, los nodos de HOLSR divulgan un subconjunto de vecinos escuchados dependiendo del nivel de la topología en que se localizan. Es decir, los vecinos HOLSR que estén localizados en un nivel inferior de la topología no serán reportados en los mensajes HELLO transmitidos por los Cluster Head a los nodos que se encuentren en los niveles superiores de la topología. De igual forma, los vecinos que están localizados en los niveles superiores de la topología no serán divulgados en los mensajes HELLO transmitidos por los Cluster Head, a los nodos de los niveles inferiores. Es así como la información de la topología proporcionada por los mensajes HELLO esta aislada entre los diferentes niveles topológicos. El tipo de mensaje que se utiliza para transportar información de la topología entre los diferentes niveles de la red, son los mensajes jerárquicos TC, es decir los mensajes HTC.

IV.6.1 Formato del mensaje CIA

El protocolo HOLSR restringe el inundamiento de mensajes de HOLSR a una región específica, es decir en un cluster. Para que esto se pueda realizar es necesario que

cada nodo móvil este asociado a un cluster head, de esta forma el cluster head se convertirá en el enrutador por omisión para cada nodo móvil en el cluster. Y como se mencionó anteriormente cuando un nodo móvil se asocie a un cluster comenzará a generar un mensaje CIA junto con los mensajes de HELLO, además el mensaje CIA debe ser el primer mensaje encapsulado en el paquete de HOLSR.

El formato del mensaje CIA se ilustra a en la Figura 14.

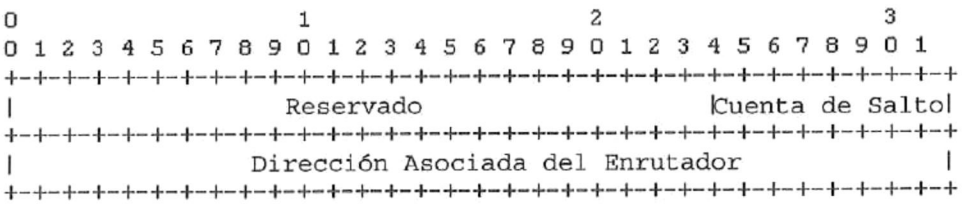


Figura 14.- Formato del mensaje CIA.

El formato del mensaje CIA cuenta con 3 campos que se describen a continuación:

- *Reservado [24 bits]:* este campo esta reservado para uso futuro.

- *Cuenta de Salto [8 bits]:* este campo es utilizado para indicar la distancia al cluster head. Cada nodo móvil aumentará en 1 el valor en el campo "cuenta de salto" del mensaje de CIA. El cluster head transmite el mensaje CIA con el campo "cuenta de salto" igual a cero, por defecto.

- *Dirección Asociada del enrutador [32 bits]:* este campo indica cual es la dirección que tiene el cluster head.

Un nodo móvil que use el protocolo HOLSR, realizará las siguientes tareas al momento de recibir un mensaje CIA:

- Si el nodo que recibe el Mensaje CIA no esta asociado a ningún cluster head, entonces va a actualizar sus tablas de enrutamiento de asociación y actualizar las tuplas.

RAssoc_addr = Dirección Asociada del Enrutador.

RAssoc_dist = Cuenta de Salto.

RAssoc_time = Tiempo Actual + RAssoc_HOLD_TIME.

Donde RAssoc_addr corresponde a la dirección IP del nodo que se ha seleccionado como Cluster Head, RAssoc_dist es el número de saltos al Cluster Head y RAssoc_time específica el tiempo en el cual expira el registro.

- Si el nodo de recepción ya está asociado a un cluster head y en el mensaje CIA que se recibe el campo RAssoc_addr == Dirección

Asociada del Enrutador, entonces se deben de considerar los siguientes casos:

1. Si (RAssoc_dist == Cuenta de Salto) entonces:

   RAssoc_time = Tiempo Actual + RAssoc_HOLD_TIME.

2. Si (RAssoc_dist < Cuenta de Salto) y (RAssoc_time a expirado) entonces :

   RAssoc_dist = Cuenta de Salto.
   RAssoc_ time = Tiempo Actual + RAssoc_HOLD_TIME

3. Si (RAssoc_dist > Cuenta de Salto) entonces no se toma ninguna otra acción.

Finalmente el nodo procesará el resto del paquete de HOLSR.

- Si el nodo de recepción ya está asociado con un cluster head diferente, RAssoc_addr ≠ Dirección Asociada del Enrutador, entonces el nodo comprobará la cuenta de saltos en el mensaje CIA:

- Si "Cuenta de Salto" >= RAssoc_dist el desechara el resto del Paquete de HOLSR.

- Si "Cuenta de Salto" < RAssoc_dist entonces el nodo actualiza su información de enrutamiento.

RAssoc_addr = Dirección Asociada del Enrutador.

RAssoc_dist = Cuenta de Salto.

RAssoc_time = Tiempo Actual + RAssoc_HOLD_TIME.

IV.6.2 Mensaje HELLO y el descubrimiento de vecinos

Para el descubrimiento de vecinos HOLSR hace uso del mensaje HELLO al igual que OLSR como se menciono anteriormente, la diferencia principal es que los nodos de HOLSR incluirán un índice de interfaz como parte del mecanismo de descubrimiento de vecinos. El campo de índice de interfaz permitirá que el nodo móvil cree su sistema de vecinos que serán divulgados mediante el mensaje HELLO transmitidos en cada una de las interfaces que tenga el nodo. Es decir que los nodos de HOLSR que estén equipados de interfaces múltiples generarán diversos mensajes de HELLO en cada una de las interfaces inalámbricas. Cada mensaje HELLO hará referencia a los vecinos a un salto que son accesibles por cada una de las interfaces, de manera individual. De esta forma el cluster head distinguirá, mediante los mensajes HELLO, a vecinos que son accesibles en los

diversos clusters y así evitará mezclar la información de vecinos que pertenecen a otros clusters.

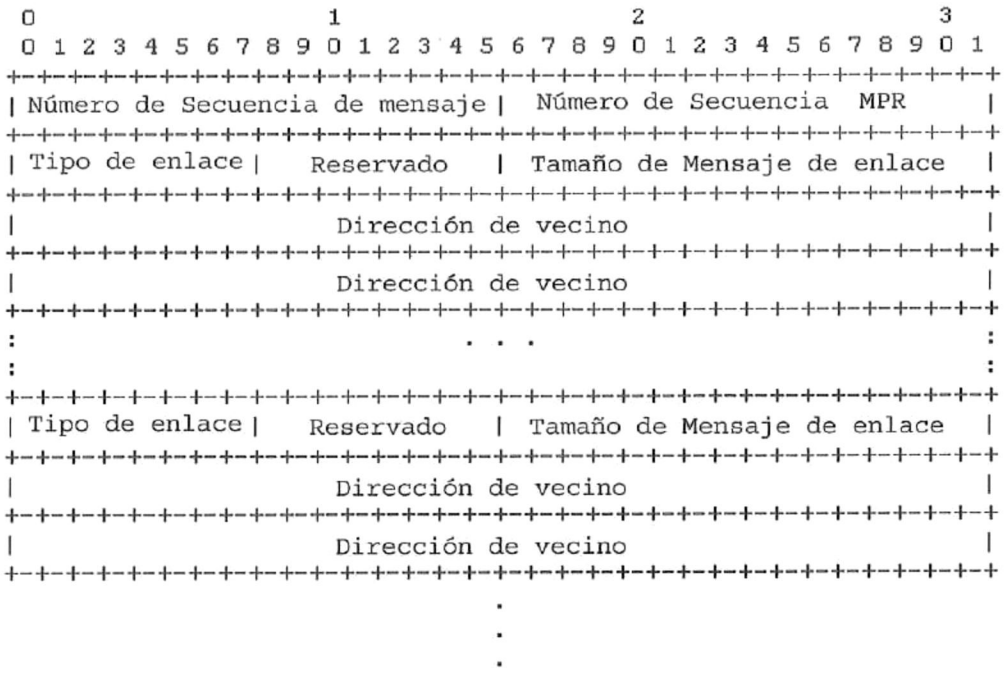
Utilizando el descubrimiento de vecinos es posible crear el sistema de vecino a un salto y a dos saltos, el sistema MPR y el sistema de selección de MPR, según lo descrito en OLSR.

IV.6.2.1 Generación del mensaje HELLO

La generación del mensaje HELLO es similar al que está descrito para el protocolo OLSR. La diferencia principal es que el mecanismo de detección de vecinos contiene un campo adicional para indicar el índice de interfaz de red, por la cual mandara el mensaje HELLO. De esta forma el nodo móvil generará diversos mensajes de HELLO para cada una de las interfaces inalámbricas del nodo. La lista de vecinos divulgada en cada uno de los mensajes de HELLO corresponderá al vecino a un salto de cada interfaz individual.

IV.6.2.2 Formato del mensaje HELLO

El formato del mensaje HELLO en el protocolo de HOLSR es igual que en OLSR el cual se ilustra en la Figura 15.



(etc.)

Figura 15.- Formato del mensaje HELLO.

IV.7 Mecanismos de enrutamiento

La arquitectura de red jerárquica es caracterizada por los diferentes niveles de topología en la red, como se ha descrito anteriormente. La comunicación entre los niveles de la topología se realiza solamente con los nodos HOLSR equipados de múltiples interfaces, que proporcionan la funcionalidad de un enrutador dentro de un cluster, es decir que los cluster head son los que funcionan como enrutadores en los diferentes niveles de la topología. El mecanismo de descubrimiento de un cluster head de HOLSR es mediante la

transmisión de los mensajes CIA, como se ha descrito previamente. El mecanismo usado para la selección del MPR es igual a la que se utiliza en OLSR.

Todos los nodos cluster head se equipan de interfaces múltiples y se deben configurar para funcionar como un enrutador en HOLSR.

IV.8 Mecanismo de descubrimiento de la topología HOLSR

HOLSR hace uso de dos tipos de mensajes para el descubrimiento de la topología: los mensajes TC y los mensajes HTC. Por una parte los mensajes TC proporcionan información de control de la topología dentro del cluster y transportan la información con respecto a los nodos móviles que se pueden alcanzar vía el autor del mensaje TC. Los mensajes TC son generados periódicamente dentro del cluster por los MPRs. Como se ha mencionado existen otro tipo de mensajes para el descubrimiento de la topología, que son los mensajes HTC, este tipo de mensajes proporcionan información de control de la topología para permitir el descubrimiento de los nodos móviles en los diferentes niveles jerárquicos que conforman la topología, los mensajes HTC son generados periódicamente por los cluster head y se difunden a los niveles superiores de la topología.

IV.8.1 Mensajes TC (Topology Control)

Los mecanismos utilizados para el descubrimiento de la topología en HOLSR están basados en los mismos principios que el OLSR. Los mensajes de TC son generados y enviados periódicamente por los MPRs dentro de los clusters, de esta forma los nodos móviles en un cluster aprenden la topología que tiene dicho cluster. En HOLSR existe una diferencia muy marcada con OLSR en cuanto a la difusión de los mensajes TC, dado que en HOLSR los mensajes solo están inundados dentro del cluster. Es decir, que los mensajes TC no fluyen de un cluster a otro, únicamente son difundidos dentro del cluster, y en OLSR los mensajes son difundidos en toda la red.

IV.8.1.1 Formato del mensaje TC

El formato del mensaje TC usado por el protocolo HOLSR es similar al que es usado por el protocolo OLSR descrito en el capítulo anterior.

El campo "dirección del autor" corresponde a la dirección IP de la interfaz del autor del mensaje TC. Por lo tanto, en caso de que el nodo móvil tenga interfaces múltiples, la "dirección del autor" es diferente para cada una de las interfaces, así que los mensajes TC tendrán una "dirección del autor" diferente por cada interfaz. La descripción de cada uno de los campos del formato del mensaje de TC de HOLSR es igual que en el protocolo OLSR. La Figura 16 ilustra el formato del mensaje TC.

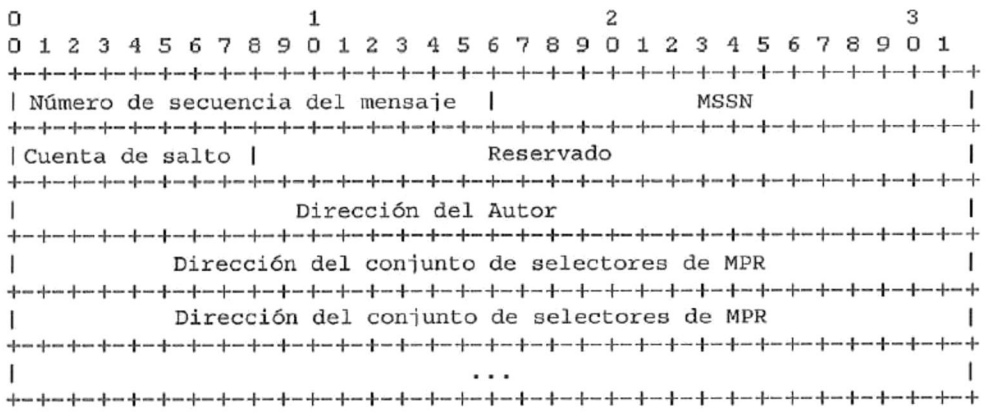


Figura 16.- Formato del Mensaje TC.

IV.8.1.2 Base de información de la topología

La base de datos de información de la topología es usada por HOLSR y esta compuesta por los campos denominados T_interf, T_dest, T_last, T_seq, T_time. T_interf es utilizada para especificar la interfaz que se debe utilizar para alcanzar T_dest. T_dest es la dirección IP de la interfaz de un nodo que esta a 1 salto de distancia del nodo T_last. T_seq es el número de serie, y T_time específica el tiempo en el cual el registro expira y debe ser quitado.

IV.8.1.3 Conjunto de anuncio de vecino

Cuando un mensaje TC es enviado por un nodo en la red para declarar su conjunto de enlaces, llamado conjunto de anuncio de vecino, que debe incluir al menos los últimos enlaces a todos los nodos de su sistema selector de MPR, es decir los vecinos que han

seleccionado al nodo remitente como un nodo MPR. Cuando un nodo tiene interfaces múltiples en este caso los mensajes TC son generados y enviados por cada una de las interfaces de red que tenga el nodo, esto es que el mensaje TC es conocido por otros nodos en los distintos niveles de la topología.

El número de secuencia asociado al conjunto de vecinos (MSSN) debe ser incrementado cuando los enlaces se quitan del conjunto de anuncio de vecino, el número de MSSN debe ser incrementado cuando los enlaces se agregan al sistema de anuncio de vecino.

IV.8.1.4 Generación del mensaje TC

Para generar el mensaje TC el protocolo OLSR específica que para construir la tabla de información de la topología de la red, se necesita que cada nodo seleccionado como MPR difunda los mensajes TC. Los mensajes TC son inundados a todos los nodos en la red, y se aprovechan de los MPRs para realizar la difusión de estos mensajes. El protocolo HOLSR hace uso de un esquema similar para la generación y la transmisión de los mensajes TC, sin embargo los mensajes TC no inundan la red entera como lo hace en OLSR, este mensaje solamente es inundado en regiones específicas, es decir en los clusters.

Los nodos de HOLSR que estén equipados de una sola interfaz inalámbrica generarán mensajes de TC según las mismas reglas usadas por el protocolo OLSR. Los

nodos de HOLSR que estén equipados de interfaces múltiples generarán mensajes TC según la regla siguiente: El nodo de HOLSR generará un mensaje diverso e individual de TC que se transmitirá por cada una de las interfaces de red a los diversos niveles de la topología. Cada mensaje de TC hará referencia al sistema de los vecinos que pueden ser alcanzados por cada interfaz individual y que han seleccionado a este nodo como MPR. De esta manera, la información de control de la topología se aísla de los clusters vecinos y que pertenecen a niveles topológicos diferentes. El mensaje TC es inundado en cada cluster por los MPRs que pertenecen al cluster. En cada nivel jerárquico, los mensajes de TC se generan independientemente.

IV.8.2 Mensajes HTC (Hierarchical Topology Control)

Mediante la transmisión del mensaje Jerárquico de Control de la Topología (HTC) un cluster head puede informar a los nodos más altos de la topología sobre los miembros de sus cluster más bajos. Existen tres tipos básicos de mensajes HTC, los mensajes HTC completos, los de actualización HTC y los mensajes de petición HTC.

Los mensajes HTC completos son transmitidos son transmitidos periódicamente por los Cluster Heads para proporcionar información con respecto a los nodos que son miembros de un cluster específico incluyendo los nodos de cualquier cluster de nivel inferior de él. De esta forma, el cluster head se anuncia para ser el siguiente salto a cualquiera de los nodos móviles enlistados en el mensaje de HTC. Los mensajes de

actualización HTC proporcionan información con respecto a los cambios en los miembros del clúster, esté mensaje se utiliza para informar si algunos nodos móviles se han unido al cluster o han dejado de pertenecer a él. Dado que los mensajes HTC llevan un campo que indica el número de secuencia, es posible determinar si ha ocurrido alguna perdida del mensaje HTC (por ejemplo, si un mensaje HTC esperado no llega) en caso de que haya existido una perdida existe en mensaje de petición HTC para la retransmisión de un mensaje HTC completo.

Los mensajes de HTC se inundan a través del cluster mientras que son retransmitidos por los nodos MPR.

IV.8.2.1 Formato del mensaje HTC

El formato del mensaje HTC, es similar al formato del mensaje TC. Sin embargo cuenta con algunas diferencias, el mensaje HTC incorpora un nuevo campo denominado "Tipo de HTC". Este campo se utiliza para indicar que el mensaje de HTC está proporcionando una descripción del sistema completo de los miembros del cluster, divulgado por el cluster head que originó el mensaje HTC. El campo "Tipo de HTC" se puede utilizar alternativamente para indicar que el mensaje contiene la información de actualización.

El campo de "Dirección del nodo" que es parte del mensaje HTC corresponde al conjunto de selectores de MPR, así como la lista de los nodos móviles que son miembros del cluster (en un nivel más bajo de la topología) que pueden ser alcanzados vía este cluster head (es decir, el nodo móvil que esta generando el mensaje de HTC).

El formato del mensaje HTC se ilustra en la Figura 17.

Figura 17.- Formato del mensaje HTC.

Uno de los propósitos del mensaje HTC es que el cluster head anuncie a los nodos de otros clusters en los niveles superiores de la topología jerárquica el conjunto de nodos móviles con el que cuenta su cluster, es decir que mediante dicho cluster head los nodos del cluster pueden ser alcanzados. De esta forma la funcionalidad del mensaje HTC es similar a la del mensaje TC. Desde el punto de vista del nodo móvil que recibe el mensaje de HTC, el mensaje de HTC es solamente otro mensaje de control de la topología, ya que

proporciona información sobre todos los nodos móviles que se pueden alcanzar vía el autor del mensaje HTC.

IV.8.2.2 Generación del mensaje HTC

A partir del mensaje HTC se puede crear la base de datos de los nodos vecinos y de información de la topología. La lista de nodos que se divulga en el mensaje HTC está formada por la lista de los selectores de MPR (esta información es obtenida de la base de datos de los nodos vecinos). Además en el mensaje HTC se incluye también una lista de los miembros del cluster que son accesibles vía el nodo móvil que esta generando el mensaje de HTC, es decir el Cluster Head. El cluster head va a mandar el mensaje HTC a los nodos de los niveles superiores y en este mensaje incluye la lista de los nodos móviles que están en un nivel inferior, este mensaje incluye un campo adicional que indica el índice de interfaz, para identificar la interfaz que se utiliza para alcanzar cualquier destino. El procesamiento de los mensajes HTC es equivalente al procesamiento de los mensajes TC.

De acuerdo a la estructura de niveles lógicos utilizados por HOLSR los nodos de los niveles superiores mantienen un conocimiento más amplio sobre la topología de la red. Los nodos en los niveles más altos de la arquitectura jerárquica poseen el conocimiento completo de todos los nodos de la red, por lo tanto el tamaño de sus tablas de enrutamiento es mucho mayor.

IV.9 Resumen

En este capítulo se trato acerca del protocolo de enrutamiento HOLSR, su funcionamiento y características principales.

Se explicó que tipos de mensajes usa HOLSR para su funcionamiento, estos mensajes son: CIA, HELLO, TC, HTC. Los cuatro tipos de mensajes utilizados por HOLSR son fundamentales para su funcionamiento y buen desempeño del protocolo. El mensaje TC y el mensaje HELLO funcionan de la misma forma que en OLSR, explicado en el capítulo anterior. El mensaje de CIA es un mensaje fundamental para la creación de clusters en HOLSR; el mensaje de CIA es enviado junto con el mensaje de HELLO, para dar a conocer quien es el cluster head de algún cluster, este mensaje es enviado por primera vez por el cluster head con un campo que contiene la dirección IP del cluster head, así como un campo que indica la cuenta de saltos que hay para poder llegar a él, haciendo uso de este mensaje se invita a los nodos vecinos a unirse a un cluster, los nodos no pueden pertenecer a dos clusters a la vez estos elegirán unirse al cluster head más cercano basándose en la cuenta de saltos que hay para poder llegar al cluster head seleccionado y así pertenecer a un cluster.

El mensaje HTC es enviado a los nodos de los niveles superiores para que estos conozcan los nodos que están en la topología inferior, es decir que los nodos que estén en los niveles superiores tendrán el conocimiento de los nodos de los niveles inferiores. Los

mensajes HTC son enviados por los cluster heads por su interfaz de índice superior. La selección de los nodos MPRs en HOLSR es igual a la utilizada por OLSR.

En HOLSR se utilizan clusters y clusters head para su administración, así como una topología jerárquica, en cada nivel de la topología pueden existir uno o más cluster, y en cada cluster va a haber un cluster head, el cluster head es el nodo que va a tener más de una interfaz inalámbrica, esto es que se puede comunicar con los nodos de su cluster y con nodos de un cluster de nivel superior.

Bajo el protocolo HOLSR se mejora el funcionamiento del protocolo OLSR cuando se utilicen nodos heterogéneos, como los son los nodos que cuentan con más de una interfaz de red. Utilizando HOLSR se reducen los gastos indirectos ya que trabaja con clusters y niveles jerárquicos, así como también en HOLSR se mejora la escalabilidad del protocolo, es decir que se puede trabajar con redes grandes sin que esto tenga repercusiones en el funcionamiento del protocolo.

Dentro de la arquitectura jerárquica de HOLSR, los cluster head potencialmente se pueden congestionar debido a que la comunicación inter-cluster se realiza a través de ellos. Por esta razón resulta de gran importancia el estudio de mecanismos de Calidad de Servicio (QoS) para su implementación en HOLSR.

En el capítulo siguiente se habla sobre la calidad de servicio.

Capítulo V Calidad de servicio y DiffServ

V.1 Introducción

Para poder evitar perdida de información que es ocasionada debido a los congestionamientos que se forman en los Cluster Heads usados en la arquitectura jerárquica de HOLSR, es muy importante que se cuenten con mecanismos que proporcionen Calidad de Servicio, ya que estos brindarán una administración de recursos correcta y de esta forma se tratará de evitar las perdidas de información.

Calidad de Servicio o QoS (por sus siglas en inglés - *Quality of Service*) en una red se refiere a "una garantía de la red para satisfacer un desempeño predeterminado para un conjunto de servicios al usuario" basándose en parámetros cuantitativos tales como: caudal eficaz (*Throughput),* retardo extremo a extremo *(Latencia),* variación en el retardo *(Jitter)*, pérdida de paquetes. En el diseño de redes, la administración de recursos es extremadamente importante para ofrecer a los usuarios finales y a sus aplicaciones las condiciones adecuadas para una transmisión apropiada de sus flujos de tráfico. Debido a que los usuarios de la red quieren experimentar nuevas aplicaciones que requieren mayor ancho de banda e imponen límites en la red es necesario que puedan existir mecanismos que proporcionen las garantías necesarias para la correcta administración de los recursos en la red esto para que el tráfico pueda fluir de manera correcta.

La llegada de nuevas aplicaciones multimedia, o aplicaciones de gestión con ciertos requisitos en cuanto a ancho de banda o retardo, han conseguido que desde hace algunos años la provisión de cierto nivel de calidad de servicio en la red sea un objetivo de vital importancia. Sin embargo este problema lleva siendo estudiado largo tiempo, y han surgido diferentes iniciativas para resolverlo. Algunas de las propuestas para proporcionar calidad de servicio se encuentran dentro de dos grupos de trabajo de la IETF: Intserv y Diffserv.

V.1.1 ¿Qué es calidad de servicio?

Calidad de servicio se puede definir como el proceso de la entrega de datos de una forma confiable, incluyendo aspectos importantes como lo es la perdida de paquetes y retardos, estos deben se ser mínimos o casi nulos, principalmente para aplicaciones en tiempo real, ya que los retardos y la perdida de paquetes afecta en gran medida este tipo de aplicaciones. La calidad de servicio es una garantía que la red debe de ofrecer para asegurar un servicio adecuado en la transmisión de datos. Las garantías de calidad de servicio ofrecen a los usuarios que los datos sean transmitidos adecuadamente en un periodo de tiempo establecido.

V.1.2 Parámetros de calidad de servicio

Para poder brindar cierto nivel de QoS, existen algunos parámetros a medir, los cuales se listan a continuación:

V.1.2.1 Paquetes perdidos

La pérdida de paquetes se puede dar por diferentes motivos, por ejemplo cuando se congestionan los canales de comunicación, la red se satura y empieza a ocasionar perdida de paquetes. La perdida de paquetes de datos es cuando estos no llegan a su destino final. El porcentaje de paquetes perdidos o desechados se puede obtener en base a las siguientes expresiones.

$$P_{PP} = \frac{P_P}{P_T} \qquad (1)$$

Donde:

P_{pp} = Porcentaje de paquetes perdidos.

P_P = Paquetes perdidos.

P_T = Paquetes transmitidos.

$$P_P = P_T - P_R \qquad (2)$$

Donde:

P_P = Paquetes perdidos.

P_T = Paquetes transmitidos.

P_R = Paquetes recibidos.

V.1.2.2 Retardo

A este también se le conoce como retardo extremo a extremo o latencia, se refiere al tiempo total que transcurre desde que un paquete de datos es transmitido de un nodo fuente hasta que es recibido por el nodo destino, este parámetro se mide en unidades de tiempo. En las aplicaciones en tiempo real como lo son el video y la voz es necesario que exista un nivel mínimo de retardo para poder obtener una buena calidad de la aplicación.

V.1.2.3 Variación en el retardo o Jitter

Es la variación en los tiempos de llegada de los datos a un destino final. Se basa en la diferencia del retardo que pertenece a paquetes similares, los cuales siguen una misma trayectoria dentro de la red, y se va a ver afectado por factores como el almacenamiento de los datos en una cola, y dependerá del tamaño de la misma.

V.1.2.4 Caudal eficaz o Throughput

Se refiere al tráfico total que es recibido con éxito por el nodo destino en un tiempo determinado, se puede obtener en base a la siguiente expresión.

$$C_E = \frac{D_R}{T_T} \qquad (3)$$

Donde:

C_E = Caudal eficaz.

D_R = Datos recibidos.

T_T = Tiempo total.

V.2 Enfoques previos de QoS

V.2.1 Servicios Integrados (IntServ)

IntServ proporciona calidad de servicio extremo - extremo a través de la red, esta basado en una reservación de recursos a lo largo de toda la trayectoria por cada flujo de trafico, Intserv se define en el RFC 1633 [Braden *et al.*, 1994]. Las reservaciones de recursos se realizan antes de colocar el tráfico en la red, para reservar estos recursos se verifica los requerimientos que las aplicaciones demanden.

El protocolo de Reservación de Recursos (RSVP) se define en el RFC 2205 [Braden *et al.*, 1997], ha sido propuesto por la IETF para el uso de las reservaciones de recurso para IntServ. RSVP es un componente clave en la arquitectura de Servicios Integrados (IntServ), aquí se define la forma de petición así como el intercambio de información entre los elementos de la red, tal como su funcionamiento, de esta forma se puede realizar un control de calidad de servicio. El receptor es el encargado de solicitar a la red la calidad de servicio que necesita para mandar determinado flujo en ese momento. El RSVP utiliza el multicast IP para distribuir los datos, cada enrutador de la red pasa la solicitud RSVP al siguiente enrutador que hay en el camino. Los enrutadores pueden o no hacer la reserva de los recursos necesarios para satisfacer el QoS, esto va a depender de la disponibilidad que tengan.

IntServ tiene como meta reservar recursos en la red, reservar la ruta entre el nodo fuente y el nodo destino, la reserva de los recursos se realiza en los enrutadores intermedios situados a lo largo de toda la ruta de datos, por ejemplo puede ser que el nodo fuente solicite que se reserven 2Mbps y un retardo de 100ms, los enrutadores intermedios tendrán que hacer esas reservaciones de QoS, a lo largo de toda la trayectoria por donde viajará el paquete, los recursos que se reserven pueden ser ancho de banda, retardo, etc. Para lograr la reservación de los recursos cada enrutador de la red debe mantener una tabla con el estado de reserva dado por la ruta a garantizar.

El mecanismo de QoS IntServ no es utilizado para redes que manejan muchos flujos, esto es debido a que los requerimientos de procesamiento y almacenamiento se

incrementan de manera lineal, con el número de reservaciones, lo que provoca que los enrutadores se puedan sobrecargar. IntServ básicamente utiliza el protocolo de RSVP para establecer y controlar la reservación de recursos; en RSVP existen dos clases de mensajes fundamentalmente, que son: *Resv* y *Path*. Cuando una aplicación requiere transmitir información a un destino primero envía un mensaje de *Path,* de esta manera indica que se va a establecer una sesión de RSVP; el mensaje de *Path* va a ser transmitido a lo largo de la misma trayectoria que va a tomar el flujo de información, cuando el destino recibe este mensaje envía un mensaje de *Resv* que va dirigido hacia la dirección fuente (transmisor) que envió el mensaje de *Path*. El mensaje *Resv* se transmite por la misma ruta que se utilizó para la transmisión del mensaje de *Path* pero de manera inversa. El mensaje *Resv* específica el tipo de reservación que se va a realizar a lo largo de todo el camino y establece la reservación de recursos.

Mediante el uso de RSVP, el transmisor puede definir el tipo de tráfico que va a mandar para que se conozcan cuales van a ser los límites con los que va a trabajar determinado flujo de información, en cuanto al ancho de banda, retardo entro otros. La información de las especificaciones del tráfico va a estar contenido en el mensaje *Path*. Cuando una solicitud no pueda ser atendida por falta de recursos, por otro lado el enrutador RSVP transmite un mensaje de error hacia el receptor, por otro lado si el último enrutador recibe el mensaje de *Resv* y este es aceptado se procede a transmitir un mensaje de confirmación al receptor.

Se pude decir entonces que IntServ permite que las aplicaciones que se van a utilizar puedan hacer una solicitud de calidad de servicio que requieren a la red. Su tarea es mantener y establecer reservaciones de recursos a lo largo de la trayectoria. Una de las desventajas del uso de IntServ, es que debido a que utiliza dos clases de mensajes *Resv* y *Path*, provoca que exista una gran cantidad de sobre carga (*overhead*), debido a las reservaciones que se necesitan hacer. Además, el mecanismo IntServ necesita una ruta dedicada, lo que provoca que el uso del ancho de banda no sea usado de manera eficiente.

V.2.2 Servicios Diferenciados (DiffServ)

Al igual que IntServ el protocolo DiffServ fue propuesto por la IETF después de IntServ. DiffServ esta definido en el RFC 2475 [Blake, S. et al. 1998], para proveer calidad de servicio distinguiendo diferentes clases de tráfico o servicios. El modelo esta orientado hacia un servicio extremo a extremo a través de un dominio único.

Servicios diferenciados consiste en un método para marcar o etiquetar paquetes dependiendo del tipo de tráfico que se este manejando; marcando o etiquetando los paquetes es como los enrutadores van a modificar su comportamiento de envío. Cuando llega un paquete a un enrutador intermedio este enrutador va a revisar cual es la etiqueta con la que viene marcado el paquete. El etiquetado del paquete se realiza previamente por los enrutadores de frontera, y así se específica el nivel de QoS que se le va aplicar al

paquete. Todo el tráfico que este etiquetado con el mismo tipo de etiqueta va a ser tratado igual por los enrutadores, aquí no se especifica un nivel de señalización como en IntServ.

En DiffServ la complejidad queda en los extremos, haciendo que los nodos intermedios tengan un manejo rápido y eficiente de los paquetes que requieren Calidad de Servicio; toda la información del flujo, el estado y el comportamiento deseado se mantiene en los extremos, como se ha mencionado DiffServ proporciona servicio basado en identificar el tipo de tráfico.

El protocolo de DiffServ, se explica más adelante con mayor detalle.

V.3 QOS en redes Ad-Hoc

Es necesario tener en cuenta las principales características que presentan las redes Ad-Hoc a la hora de proveer QoS [Wu y Harms, 2001]. Principalmente su topología dinámica, que modifica los nodos vecinos constantemente, así como el estado de sus enlaces, modificando de esta forma el ancho de banda disponible, así como el retardo presente en los enlaces.

Como se menciona en el capítulo II, los protocolos de enrutamiento en redes Ad-Hoc se pueden dividir principalmente en dos grupos: proactivos y reactivos. Los protocolos proactivos son aquellos que mantienen una ruta hacia todos los nodos, aunque en ese

momento no se utilicen, tienen la ventaja de que las rutas van a estar disponibles en todo momento y los retardos debido al descubrimiento de rutas nuevas va a ser casi nulos. El caso de los protocolos reactivos intenta optimizar el uso de ancho de banda descubriendo la ruta hacia un destino sólo en el caso en que sea necesario enviar un paquete.

Un protocolo de enrutamiento con capacidades de QoS debería intentar establecer una ruta que cumpla con determinados requisitos como ancho de banda, retardo extremo a extremo, variación en el retardo, etc.

Para proporcionar calidad de servicio en las redes móviles Ad-Hoc, existen algunos métodos especialmente diseñados para este tipo de redes, estos métodos no han sido estandarizados por la IETF, a continuación se mencionan y se explican brevemente cada uno de ellos.

V.3.1 *CEDAR*

Es un protocolo de enrutamiento, pero con soporte de QoS. Este protocolo conoce con anticipación los requisitos de QoS que se van a utilizar, es decir mediante la ruta precalculada CEDAR puede ver si esta satisface los requerimientos de QoS. Este protocolo establece dinámicamente un núcleo de la red, propaga los estados de los enlaces estables y con gran ancho de banda a los nodos del núcleo [Sivakumar *et al.*, 1999].

CEDAR tienen tres procesos dominantes:

- *Extracción del Núcleo:* En primer lugar se escoge un sistema de nodos de la red, de forma periódica, los cuales formarán parte del núcleo. Los nodos del núcleo son escogidos de forma dinámica y distribuida, estos nodos elegidos mantendrán la topología local de los nodos en su dominio, es decir los nodos del núcleo serán un sistema mínimo dominante de la red Ad-Hoc. Cada nodo del núcleo mantiene información de la topología local de los nodos móviles que están bajo su dominio, es así como realizará el cálculo de las rutas. Los nodos móviles deben formar parte del núcleo o ser vecinos de algún nodo en el núcleo.

- *Propagación del estado de los enlaces*: su objetivo principal es que cada nodo que integre al núcleo conozca el estado y la topología de cada uno de los enlaces locales, así como los enlaces que estén más lejanos pero que tengan estabilidad y cuenten con un gran ancho de banda. CEDAR propaga toda la información disponible de ancho de banda de los enlaces estables a todos los nodos del núcleo. La idea es que la información de los enlaces estables así como de los que tengan gran ancho de banda los puedan conocer los nodos lejanos en la red, mientras que sigue habiendo información sobre los enlaces dinámicos o de bajo ancho de banda dentro del área local.

- *Cálculo de la ruta.*- cuando una fuente desea enviar tráfico a un destino, previamente manda un mensaje indicando (origen, destino, ancho de banda solicitado). CEDAR trata de encontrar de manera iterativa una ruta parcial del núcleo fuente al núcleo destino que satisfaga el ancho de banda solicitado. La información de ancho de banda, origen y destino es propagado por el núcleo, haciendo uso del broadcast, hasta que alcanza el destino, los nodos intermedios deben comprobar la disponibilidad de ancho de banda a lo largo de la trayectoria es decir en cada salto intermedio.

Cuando un nodo necesita establecer una conexión a otro nodo, entra en contacto con el nodo núcleo ya que los nodos núcleos son los encargados de obtener y mantener las rutas; el núcleo esta formado por el conjunto mínimo dominante de nodos en la red Ad-Hoc. Para encontrar el conjunto mínimo dominante de nodos, se hace uso de un algoritmo robusto pero simple que únicamente hace cálculos locales.

En resumen CEDAR, es un protocolo que propone la elección en la red de un núcleo que será el responsable de todos los cálculos de las rutas, contiene nodos que se eligen dinámicamente para formar el núcleo de la red, de modo que cada uno de ellos mantenga la topología local de los nodos que pertenecen a su dominio, los nodos del núcleo propagan información sobre la disponibilidad de ancho de banda en los enlaces estables de la red, y guardan la información de los enlaces dinámicos y de bajo ancho de banda.

V.3.2 *INSIGNIA*

Este protocolo diseñado especialmente para redes móviles Ad-Hoc puede ser caracterizado como un protocolo "*RSVP dentro de banda*", la meta principal en el diseño de INSIGNIA [Mohapatra *et al.*, 2003] fue proporcionar QoS; el objetivo principal de INSIGNIA es el soporte de servicios adaptables, es decir, aplicaciones con capacidad de adaptarse a cambios en el ancho de banda disponible, utiliza el campo de opciones de la cabecera IP para indicar los requisitos de recursos ancho de banda mínimo y máximo que necesita.

INSIGNIA esta diseñado para proporcionar QoS a los servicios de forma adaptable. En este contexto, los servicios se adaptan de tal forma que proporcionan un aseguramiento mínimo de ancho de banda a los flujos en tiempo real como los datos de la voz, el vídeo, etc. INSIGNIA permite que los datos sean entregados cuando los recursos estén disponibles es decir cuando exista un ancho de banda máximo [Seoung-Bum *et al.*, 1999]. Los flujos requieren control de admisión, reservación de recursos y el mantenimiento de los recursos a lo largo de todos los enrutadores intermedios entre el nodo fuente y el destino proporcionando calidad de servicio de principio a fin. Algunos servicios pueden ser incapaces de adaptarse a los cambios de ancho de banda que se puedan presentar, mientras otros se pueden adaptar de manera adecuada.

INSIGNIA permite proporcionar QoS a los datos en tiempo real, de tal forma que las fuentes puedan especificar sus necesidades de ancho de banda máximos y mínimos. La asignación de recursos, la restauración y la adaptación de la sesión entre los anfitriones móviles es un papel central para poder proporcionar QoS. INSIGNIA establece y mantiene las reservaciones de flujos continuos. INSIGNIA puede restaurar el estado del flujo (es decir, una reservación) cuando la topología cambia, dentro del intervalo de algunos paquetes consecutivos de IP. INSIGNIA puede proporcionar los niveles adaptables asegurados de QoS a los usos en tiempo real, basados en el QoS por petición y a la disponibilidad de recursos en la MANET. Para proporcionar QoS adaptable utiliza:

- Reservación Rápida.
- Restauración Rápida.
- Divulgación de QoS.
- Un mecanismo de regeneración.
- Adaptación según las condiciones de la red.

El control de la admisión, es el responsable de asignar el ancho de banda a los flujos basándose en el ancho de banda máximo y mínimo solicitado. Una vez que se hayan

asignado los recursos son restaurados periódicamente, con las reservaciones nuevas no se afectan las reservaciones existentes.

V.3.3 *SWAN*

Utiliza algoritmos distribuidos de control, para entregar la diferenciación del servicio en la red móvil Ad-Hoc.

El hecho de que exista movilidad entre los nodos hace que la reservación de recursos sea más difícil de asignar para un nodo fuente así como mantener la conexión.

SWAN (por sus siglas en inglés - *Stateless Wireless Ad-Hoc Networks*) esta basado en los resultados a pedido, en el que un nodo fuente toma la decisión de admitir un flujo nuevo en tiempo real o no, una vez que una sesión es admitida como sesión en tiempo real sus paquetes son marcados como DCHA (para servicios en tiempo real), si los paquetes no se marcan son tratados como paquetes de mejor esfuerzo [Ahn *et al.*, 2002].

Midiendo el retardo de la MAC, SWAN configura automáticamente el control del tráfico y midiendo el índice de flujos en tiempo real que pasa a través de sus vecinos, evalúa la cantidad de ancho de banda que todavía este disponible para nuevas conexiones en tiempo real, de esta forma es como puede configurar el control de la admisión.

El tráfico en tiempo real se puede ayudar mucho con la diferenciación del servicio [Ahn *et al.*, 2002], ya que da prioridad al tráfico en tiempo real y eso permite que este tipo de tráfico tenga una mejor calidad.

SWAN simplifica la arquitectura entera y proporciona una solución que aunque no puede garantizar las necesidades de QoS de cada uno de los flujos de una sesión completa, proporciona una diferenciación entre el tráfico de tiempo real y el de mejor esfuerzo, dando una prioridad al tráfico en tiempo real.

V.3.4 *FQMM*

Es un modelo para proporcionar calidad de servicio (QoS) y fue diseñado especialmente para redes móviles Ad-Hoc, este modelo combina los mecanismos de IntServ y DiffServ, es decir propone un esquema híbrido para proporcionar calidad de servicio, combina la granularidad del flujo de IntServ y la granularidad por clase de DiffServ) [Guimañaes *et al.*, 2004]. En la actualidad FQMM (por sus siglas en ingles - *Flexible QoS Model for MANETs*) es utilizado para redes móviles Ad-Hoc de tamaño pequeño, con menos de 50 nodos, además de ser utilizado para topologías planas, no se utiliza para topologías de tipo jerárquico. FQMM es utilizado para determinar y para asignar los recursos en varios nodos móviles, es un esquema híbrido que cuenta con un mecanismo que se adapta al tráfico, para mantener la diferenciación constante entre el tipo de tráfico. FQMM proporciona Calidad de Servicio por clase como en DiffServ para el

tráfico menos importante, el tráfico es clasificado en el nodo fuente y el aprovisionamiento de QoS se hace en cada nodo a lo largo de la trayectoria. Para el tráfico que necesita tener mayor aprovisionamiento de Calidad de Servicio lo hace por flujos como en IntServ. Este modelo para proporcionar Calidad de Servicio presenta muchos problemas de escalabilidad, sin embargo se continúa intentando explorar lo mejor de ambos mecanismos.

A continuación se profundiza en el modelo de servicios diferenciados (DiffServ) el cual este trabajo de tesis propone como modelo de QoS para redes Ad-Hoc.

V.4 Servicios diferenciados

DiffServ ó Servicios Diferenciados (en el Apéndice A se encuentra la terminología utilizada en esta sección) ofrece diversas clasificaciones de servicios de tráfico para proporcionar calidad de servicio a los usuarios, clasificando el tráfico que entra en la red en diferentes clases, está arquitectura es orientada a la no reservación de recursos por lo tanto no necesita tener estados de señalización por cada flujo de datos, en los nodos que conforman la red. DiffServ esta conformado básicamente por los nodos extremos o de frontera (denominados nodos edge) y los nodos intermedios o de núcleo (denominados nodos core).

En DiffServ el tráfico que se genera es clasificado y acondicionado en los nodos extremos de la red, de esta forma es asignado a diferentes comportamientos de flujo, cada

comportamiento de flujo corresponde a un código DiffServ. La clasificación de los paquetes es realizada en base al acuerdo de nivel de servicio, SLA (por sus siglas en inglés - *Service Level Agreement*) que es establecida previamente entre los usuarios de la red y el Proveedor de Servicio de Internet, ISP (por sus siglas en inglés - *Internet Service Provider*). Los nodos extremos tiene la tarea de realizar el control de admisión, en donde el tráfico que no este establecido en el SLA es descartado, retardado o remarcado con una prioridad menor, para que de esta forma se pueda lograr el SLA acordado. Los nodos en el núcleo utilizan la información de marcación de los paquetes que es realizada por los nodos extremos para determinar la prioridad de reenvió de paquetes así como su descarte, es decir estos nodos brindarán en el interior de la red el tipo de servicio con la que los paquetes vienen marcados.

En la arquitectura de DiffServ [Blake *et al.*, 1998], cada paquete es clasificado y marcado para recibir un trato específico en cada uno de los saltos que los paquetes hagan a lo largo de su trayectoria. La clasificación, el marcado, las políticas y las operaciones de acondicionamiento necesitan sólo ser implementadas en los extremos de la red. Esta arquitectura logra escalabilidad al implementar complejas funciones de clasificación y condicionamiento sólo en los nodos de extremo de la red, y aplicando comportamientos por salto a los agregados del tráfico que han sido apropiadamente marcados usando el campo DS en los encabezados de IPv4 o IPv6. DiffServ clasifica a los paquetes dentro de diferentes clases, cada paquete pertenece a una clase específica.

Los comportamientos por salto PHB (por sus siglas en inglés - *Per Hop Behavior*) se definen para permitir los medios razonables de asignar recursos de almacenamiento intermedio y de ancho de banda en cada nodo. El uso del flujo o el estado de la expedición no necesita ser mantenido en el núcleo de la red.

V.4.1 *DIFFSERV en el encabezado IP*

El estándar de DiffServ utiliza un código DS conocido como DSCP (por las siglas en inglés - *DiffServ Code Point*). Cada paquete IP lleva un byte, para especificar el tipo o la clase de servicio que se esta manejando, en IPv4 este byte es llamado Tipo de Servicio o TOS (ver Figura 18). En la nueva versión de IP, la versión IPv6, hay un byte equivalente al TOS llamado octeto de Clase de Tráfico (ver Figura 19).

Figura 18.- Formato del Encabezado de IPv4.

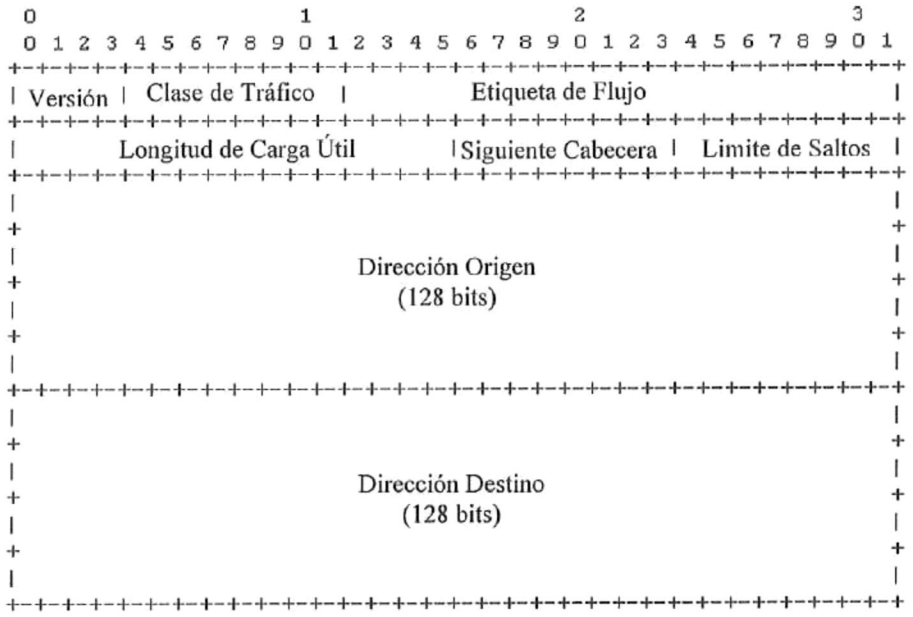


Figura 19.- Formato del Encabezado de IPv6.

En el octeto del campo DS esta definido en el RFC 2474 [Nichols *et al.*, 1998], los primeros seis bits del campo son utilizados por el DSCP, este *Code Point* de DiffServ es usado para indicar la forma en como cada enrutador debe tratar a los paquetes, es decir es la selección de PHB adecuado al tipo de tráfico. Los dos bits restantes, CU (por sus siglas en inglés - *Currently Unused*) actualmente se encuentran reservados y sin uso. El octeto se puede apreciar en la Figura 20.

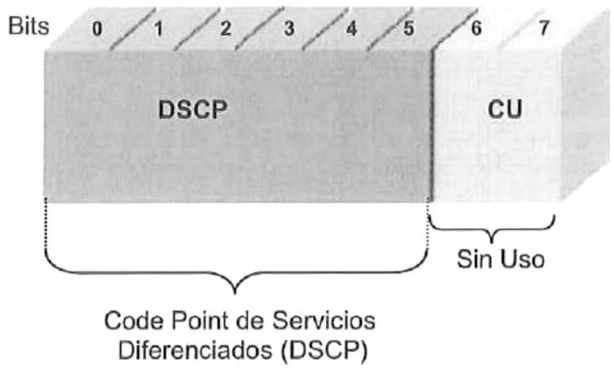


Figura 20.- Estructura del campo DS.

Como se mencionó anteriormente de los 8 bits de este campo solamente se utilizan los primeros 6 bits. Hay (2^6) diferentes valores binarios es decir, 64 posibles combinaciones para brindar un nivel de calidad de servicio adecuado, dependiendo del tipo de tráfico que se utilice.

Las posibles formas de proporcionar a los paquetes un nivel de calidad de servicio están implementadas en cada uno de los enrutadores que forman parte del dominio de DiffServ, utilizando un comportamiento por salto PHB.

Diffserv realiza la clasificación compleja y el acondicionamiento de tráfico (TCA- por sus siglas en inglés - *Traffic Conditioner Agreement*) en los enrutadores extremos. Los enrutadores que están en el núcleo realizan una reclasificación de los paquetes únicamente en caso de ser requerida y manda el tráfico al siguiente enrutador basándose en el código que contenga que fue establecido previamente por el enrutador extremos.

V.4.1.1 Requerimientos para DiffServ

Continuamente Internet esta teniendo un crecimiento significativo en cuanto al número de nodos, la gran variedad de aplicaciones y la capacidad de la infraestructura de la red. Debido a esto la arquitectura de servicios diferenciados debe tener la capacidad de poder adaptarse a los continuos cambios de crecimiento de la red, para poder brindar servicios diferenciados de manera confiable, independientemente del tamaño de la red, del número de aplicaciones que se manejen, etc., los siguientes requerimientos fueron identificados y ubicados en esta arquitectura:

- Debe acomodar una amplia variedad de servicios y proveer políticas, extendiendo una red punta a punta o una red particular.

- Debe permitir el desacoplamiento del servicio de la aplicación particular en uso.

- Debe trabajar con aplicaciones existentes sin la necesidad de cambios en interfaces de aplicaciones programables.

- Debe desacoplar las funciones de condicionamiento de tráfico y aprovisionamiento de servicios de comportamientos de envío implementados en los nodos del núcleo de la red.

- No debe depender de la aplicación de señalización salto a salto.

- Debe requerir solo una pequeña cantidad de comportamientos de envío, cuya complejidad de implementación no domine el costo de un dispositivo de red o debe utilizar solo estado de clasificación de agregados dentro del núcleo de la red.

- Debe permitir interoperabilidad razonable con nodos de red que no implementen DS.

- Debe permitir implementaciones de clasificación de paquetes simples en nodos del núcleo de la red.

V.4.2 Modelo Arquitectónico de Servicios Diferenciados

La arquitectura de DiffServ está basada en un modelo simple en donde el tráfico que entra a la red es clasificado y acondicionado en los extremos de esta, asignado a los paquetes diferentes comportamientos. Cada comportamiento es identificado por un Code Point DS ó DSCP, en el núcleo de la red el procesamiento de los paquetes es más simple, los paquetes son enviados según su comportamiento por salto asociado al DSCP.

V.4.2.1 Dominio de Servicios Diferenciados (DS)

Un dominio DiffServ es un conjunto de nodos DS que operan con una política de aprovisionamiento de servicios comunes y con un conjunto de grupos PHB implementados en cada nodo. Esta formado por nodos DS de frontera que clasifican y condicionan el tráfico entrante para asegurarse que los paquetes que transitan el dominio estén apropiadamente marcados para seleccionar un PHB de los grupos PHB que son soportados dentro del dominio. Los nodos del núcleo seleccionan el comportamiento de envío basándose en su código DSCP, y lo hacen asociando éste valor a unos de los PHB soportados. La inclusión de nodos que no soportan DS dentro de un dominio DS puede resultar en un desempeño impredecible y puede impedir la habilidad de satisfacer el acuerdo del nivel de servicio (SLA). Un dominio DS consiste en general de una o más redes bajo la misma administración.

V.4.2.1.1 Nodos DS de frontera e interiores

Los nodos DS de frontera interconectan el dominio DS con otros dominios que pueden o no soportar DS. Los nodos interiores son los que se encuentran en el interior del dominio DS y se encargan de interconectarse con otros nodos interiores o de frontera dentro del mismo dominio DS.

Ambos tipos de nodos deben ser capaces de aplicar los flujos de tráfico dependiendo del PHB apropiado a los paquetes basándose en el código DSCP, sino puede resultar en un comportamiento impredecible. Los nodos DS de frontera puede que sean requeridos para realizar funciones de condicionamiento de tráfico como se define en un acuerdo de condicionamiento de tráfico (TCA) dentro del mismo dominio DS y el dominio contiguo al cual conectan. Los nodos DS interiores deben poder realizar re-marcado de códigos. Si un nodo no actúa como nodo de frontera, entonces el nodo DS topológicamente más cercano a este nodo, actúa como el nodo DS de frontera para el tráfico de ese nodo.

V.4.2.2 Nodos de Ingreso y Egreso

Los nodos de frontera actúan como nodos DS de ingreso y nodos DS de egreso para el tráfico de distintas direcciones. Un nodo DS de ingreso es el responsable de asegurarse de que el tráfico que entra en el dominio DS sea conforme con el TCA que se estableció entre él y el otro dominio con los cuales el nodo del ingreso está conectado. Un nodo de egreso del DS puede realizar funciones de condicionamiento del tráfico.

V.4.2.3 Región de Servicios Diferenciados

Una región de Servicios Diferenciados (Región DS) es un conjunto de uno o más dominios DS contiguos. La Región DS es capaz de soportar servicios diferenciados a lo largo de toda la trayectoria.

Los dominios DS en una región DS que puede soportar distintos grupos PHB internamente y diferentes códigos PHB. Más sin embargo, para permitir servicios que se expanden a través de los dominios, los dominios DS contiguos deben cada uno establecer un SLA entre ellos que define cierta TCA, para especificar cómo el transito del tráfico de un dominio DS a otro es condicionado en la frontera entre los dos dominios DS.

V.4.2.4 Clasificación y condicionamiento de tráfico

Los Servicios Diferenciados se extienden a través de la frontera del un dominio DS estableciendo un SLA entre una red *upstream* (transferencia de datos desde nuestro nodo a un nodo remoto) y un dominio DS *downstream* (transferencia de datos desde un nodo remoto al local).

El SLA puede especificar la clasificación del tráfico y las reglas de re-marcado, así como perfiles de tráfico y acciones a los flujos de tráfico que están dentro o fuera del perfil. El TCA entre dominios deriva del SLA.

La política de clasificación de paquetes identifica el subconjunto de tráfico que puede llegar a recibir un servicio diferenciado, al ser condicionado y/o mapeado a uno o más agregadores de tráfico (BA- por sus siglas en inglés - *Behavior Agregate*).

El acondicionamiento de tráfico realiza la medición, conformación, política y/o remarcado para asegurarse que el tráfico entrante al dominio DS es conforme a las reglas especificadas en el TCA, en acuerdo con los dominios de aprovisionamiento de servicio.

V.4.2.4.1 Clasificadores

El clasificador de paquetes selecciona a los paquetes de tráfico de acuerdo al contenido del encabezado del paquete. Existen dos tipos de clasificadores. El *clasificador* Behavior Aggregate clasifica los paquetes basándose en el código DSCP solamente. El *clasificador MF* (por sus siglas en inglés - *Multi-Field*) clasifica los paquetes mediante la selección de una combinación de valores de uno o más campos del encabezado, como dirección fuente, dirección destino, campo DS, protocolo ID, número de puerto fuente y destino, entre otra información.

Los clasificadores se utilizan para dirigir los paquetes marcados a un elemento acondicionador de tráfico para su procesamiento posterior. Los clasificadores se deben configurar por un procedimiento administrativo en acuerdo a un apropiado TCA. El clasificador debe autentificar la información que usa para clasificar el paquete.

V.4.2.4.2 Perfiles de tráfico

Los perfiles de tráfico especifican las propiedades temporales de un flujo de tráfico seleccionado por el clasificador. Esto proporciona las reglas para determinar si un paquete está dentro o fuera del perfil. Por ejemplo, un perfil basado en "*token bucket*" puede parecer así:

$$Codepoint = X, \text{ used token-bucket } r,\ b$$

El perfil anterior indica que todos los paquetes marcados con un código DS X deben ser medidos con medidor "*token bucket*" con tasa r y de tamaño b. En este ejemplo los paquetes fuera del perfil son los que llegan cuando hay insuficientes fichas (tokens) disponibles en la cubeta (bucket).

V.4.2.4.3 Acondicionadores de tráfico

El acondicionador de tráfico puede contener los siguientes elementos: medidor, marcador, conformador y descartador. Un flujo de tráfico es seleccionado por un clasificador, que dirige los paquetes a un acondicionador de tráfico. El medidor es usado para medir flujo de tráfico en base a un perfil de tráfico. El estado del medidor respecto a un paquete en particular puede ser usado para afectar el marcado, descarte, o acción de conformación.

Cuando los paquetes salen del acondicionador de tráfico de un nodo DS frontera, el código DS de cada paquete debe asignarse a un valor apropiado.

En la Figura 21, se muestra un diagrama a bloques del clasificador y acondicionador de tráfico.

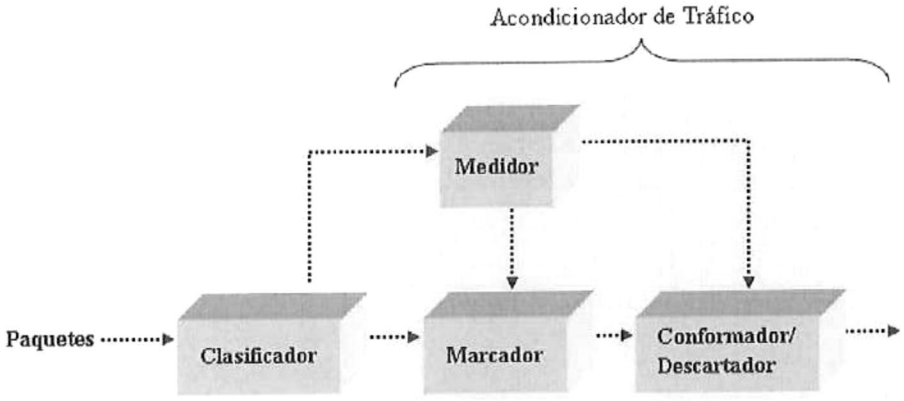


Figura 21.- Diagrama a bloques del clasificador y acondicionador de tráfico.

Medidor: miden las propiedades temporales del flujo de paquetes seleccionado por el clasificador en base a un perfil de tráfico especificado en el TCA. Los medidores pasan información de estado a otras funciones de acondicionamiento para determinar cierta acción para cada paquete tanto dentro como fuera de perfil.

Marcador: marca los paquetes asignando al campo DS un DSCP particular, basado en un agregador de tráfico (BA) DS. Puede el marcador ser configurado para etiquetar

todos los paquetes que son dirigidos con un código particular DS o puede estar configurado para marcar los paquetes con un conjunto de códigos para seleccionar un PHB dentro de un grupo PHB, de acuerdo a las mediciones realizadas por el medidor Cuando el marcador cambia el código DSCP en un paquete, se dice que se ha "remarcado" el paquete.

Conformador: Este bloque es encargado de retardar uno o todos los paquetes de un flujo de tráfico con el fin de que el flujo cumpla con el perfil de tráfico estipulado por el SLA. El conformador normalmente tiene un buffer de tamaño finito y los paquetes pueden ser descartados si no existe suficiente espacio en el buffer para soportar a los paquetes retrasados.

Descartador: Aquí es donde se descartan algunos o todos los paquetes de un flujo de tráfico para evitar el congestionamiento de manera que el flujo cumpla con los requisitos de tráfico estipulado. Un descartador puede ser implementado como un caso especial de un conformador si se pone el tamaño del buffer del conformador igual a 0 (o muy pocos) paquetes, a este proceso se le conoce como "política".

V.4.2.5 Ubicación de los acondicionadores de tráfico o clasificadores MF

Los acondicionadores de tráfico están usualmente localizados dentro de los nodos de DS de frontera de ingreso y egreso, pero pueden ser localizados en los nodos interiores del dominio DS.

Dentro del dominio de la fuente: se define como el dominio que contiene el nodo que origina el tráfico que recibe un servicio particular. El tráfico originado del dominio fuente a través de una frontera puede ser marcado por las fuentes de tráfico directamente o por medio de nodos intermedios antes de que dejen el domino origen. Esto se conoce como "pre-marcado".

Hay ciertas ventajas en el hecho de marcar paquetes cerca de la fuente/origen de tráfico.

Primero, una fuente de tráfico puede más fácilmente tomar en cuenta las preferencias de las aplicaciones en el momento de decidir qué paquetes deben recibir mejor tratamiento de envío. Además, la clasificación de paquetes es más simple antes que el tráfico ha sido agregado con paquetes de otras fuentes, ya que el número de reglas de clasificación que deben ser aplicadas dentro de un nodo único es reducido.

El nodo frontera del dominio fuente debe también monitorear la conformidad con el TCA, así como aplicar políticas, conformado, o pre-marcado de paquetes según se necesite.

En las fronteras de un dominio DS: El SLA entre dominios debe especificar cuáles nodos tiene la responsabilidad de mapear los flujos de tráfico a los BA DS y condicionar esos agregados conforme con el apropiado TCA. Sin embargo, un nodo de ingreso debe asumir que el tráfico entrante puede no estar conforme el TCA y debe estar preparado para reforzar el TCA de acuerdo a la política local.

Cuando los paquetes son pre-marcados y condicionados en un dominio de flujo superior, una clasificación y condicionamiento potencialmente menores necesitan ser soportados en el dominio DS de flujo inferior.

Si un nodo de ingreso esta conectado a un dominio superior sin capacidad DS, el nodo de ingreso DS debe poder cumplir con todas las funciones de condicionamiento de tráfico necesarias en el tráfico entrante.

En dominios sin capacidad DS: Las fuentes de tráfico o los nodos intermedios en un dominio sin capacidad de DS pueden emplear acondicionadores de tráfico para premarcar los paquetes de tráfico antes de que ingrese al dominio DS en sentido inferior. De esta manera las políticas locales para la clasificación y la marcación pueden ser canceladas.

En los nodos intermedios de un dominio DS: En la arquitectura básica de DiffServ se asume que las funciones complejas de acondicionamiento y clasificación de tráfico estén situadas únicamente en los nodos que están ubicados en las fronteras tanto de ingreso como de egreso, sin embargo estas funciones no están imposibilitadas en el núcleo de la red.

V.4.2.6 Comportamiento Por Salto (PHB)

Un comportamiento por salto PHB, es el tratamiento que se le proporciona al tráfico para su posterior transmisión, el PHB es administrado por cada uno de los enrutadores que conforman el dominio DS, se define como un comportamiento PHB particular, al conjunto de paquetes que tienen las mismas características para la transmisión, a este conjunto de paquetes se le conoce como agregador de tráfico (BA), este tráfico será administrado de la misma manera en un enlace determinado en el dominio DS.

Los PHBs están específicados de acuerdo a ciertas prioridades en cuanto al uso de recursos en la red como lo son: retardo, pérdidas, jitter, etc.

Cuando se implementa un PHB en un enrutador, se ve reflejado en los mecanismos para la administración de colas y en la calendarización de paquetes. Para la asignación de un PHB se hace mediante la asignación de un código DSCP.

Tipos de PHBs:

Envío Expedito EF *(por sus siglas en inglés - Expediated Forwarding)*: es un tratamiento que se da a la transmisión de tráfico, en donde la tasa de envío debe ser la que se estableció en el SLA, en este tipo de envío el tráfico debe ser recibido a la tasa de transmisión que fue acordada independientemente si existe o no otro tipo de tráfico dirigido al mismo enrutador al cual se le esta mandando el tráfico EF.

EF es un tipo de comportamiento PHB que puede manejar tráfico como aplicaciones multimedia, video conferencias, voz sobre IP, etc. Aplicaciones que presenten ciertas exigencias como es que tenga bajo retardo, baja variación en el retardo y pérdida de paquetes mínimas, es por eso que EF es un comportamiento PHB adecuando para aplicaciones muy susceptibles.

Envío Asegurado AF (por sus siglas en inglés - *Assured Forwarding):* este tipo de comportamiento PHB asigna porcentajes de los recursos de la red como lo es el ancho de banda y el tamaño de la cola, para diferentes tipos de tráfico. En el AF los enrutadores de frontera revisan el tráfico y lo marcan, dentro o fuera de un perfil de tráfico determinado, entonces cuando se presenta un congestionamiento en la red, el tráfico marcado como fuera del perfil es descartado.

V.5 Resumen

En este capítulo se revisó lo referente a la calidad de servicio en las redes Ad-Hoc y los tipos de mecanismos de calidad de servicio existen para las redes Ad-Hoc.

En particular se analizó el mecanismo de DiffServ para proporcionar calidad de servicio, se analizó su funcionamiento, los mecanismos que utiliza para proveer a la red de calidad de servicio. En este trabajo de tesis se propone el uso de DiffServ para una MANET que implementa el protocolo HOLSR.

Se analizó la arquitectura de DiffServ, los requerimientos que son necesarios para que DiffServ pueda funcionar haciendo un marcado de los paquetes.

DiffServ realiza un marcado o etiquetado de los paquetes entrantes en la red donde se esta brindado este servicio, cuando un paquete llega a un enrutador intermedio este va a revisar con que tipo de marcación viene el paquete (por ejemplo, si se trata de tráfico que necesite mayor prioridad como el tráfico en tiempo real) es decir mediante la etiqueta con la que venga el paquete se conocerá cual es el nivel de QoS que se le necesita dar a dicho paquete. El paquete que llega a los enrutadores intermedios fue etiquetado previamente por los enrutadores de los extremos o de frontera, es decir la complejidad de esta arquitectura va a quedar en los nodos extremos, haciendo que los nodos intermedios tengan un manejo más rápido y sencillo.

Utilizando Diffserv se evita crear información de estado a lo largo de todo el camino del flujo de tráfico como en el caso de IntServ.

El capítulo siguiente trata acerca del simulador NS-2 (Network Simulator 2).

Capítulo VI Implementación en el simulador NS-2

VI.1 Introducción

El NS-2 (por sus siglas en inglés – *Network Simulator*) es un simulador orientado a objetos, escrito en C++, con un interprete de OTcl como interfaz. El simulador soporta una jerarquía de clases en C++ (también llamada la jerarquía compilada), y una jerarquía de clases similar a la del intérprete de OTcl (también llamada la jerarquía interpretada). Las dos jerarquías están fuertemente relacionadas entre ellas; desde la perspectiva del usuario, hay una correspondencia uno-a-uno entre una clase en la jerarquía interpretada y la correspondiente en la jerarquía compilada. La raíz de esta jerarquía es la clase TclObject.

Los usuarios crea nuevos objetos del simulador a través del interprete; estos objetos son instanciados en el interprete, y están fuertemente reflejados por un objeto correspondiente en la jerarquía compilada. La jerarquía de clase interpretada se establece automáticamente a través de los métodos definidos en la clase TclClass. Los objetos instanciados por el usuario se reflejan a través de los métodos definidos en la clase TclObject [ISI, 2006a].

El NS-2 esta basado en dos lenguajes porque el simulador tiene dos tipos diferentes de tareas que debe cumplir. Por un lado, las simulaciones detallan el protocolo que se va a utilizar y requieren un lenguaje de programación de sistemas que pueda manipular

eficientemente los bytes, las cabeceras de paquetes, e implementar los algoritmos que se ejecutaran sobre un conjunto grande de datos. Por otra parte, en la red pueden variar parámetros o configuraciones, o rápidamente cambiar el escenario. En este caso, el tiempo de la iteración (cambiar el modelo y volver a correr) es más importante. Puesto que la configuración se corre una vez (al principio de la simulación), el tiempo de la corrida es una de las tareas menos importantes.

El NS-2 resuelve ambas necesidades con dos lenguajes diferentes, C++ orientado a objetos y el interprete OTcl. C++ funciona de manera rápida en la ejecución de los diferentes escenarios dado que reduce el tiempo de ejecución de los mismos, y tiene mayor eficiencia en la ejecución de los escenarios de simulación, pero sin embargo es muy lento cuando se necesitan hacer cambios en los diferentes escenarios. OTcl como se mencionó anteriormente es un interprete, debido a esto funciona mucho más lento al momento de correr los escenarios, es decir es menos eficiente en comparación con C++, sin embargo se pueden hacer cambios muy rápidamente en los diferentes escenarios de simulación, haciéndolo de esta forma ideal para la configuración de la simulación.

En NS-2 cuando se crea un archivo de simulaciones es decir un *script* realizado en tcl, las instrucciones del código se van analizando una a una conforme se va ejecutándose, es decir va llamando los procedimientos para formar la topología y los parámetros que estén relacionados con la simulación. Los eventos que ocurran en la simulación son controlados por los calendarizadores.

El *script* realizado en tcl puede tener como salida un archivo de texto que generalmente tiene extensión *.tr*, adicionalmente se le puede indicar que genere un archivo para ver de forma gráfica los resultados a través del animador, NAM (por sus siglas en inglés - *Network ANimator*) o generar archivos adicionales para visualizar en el xgrap (sirve para realizar gráficas a partir de los datos de la simulación).

VI.2 Modelo inalámbrico en NS-2

Las simulaciones con modelos inalámbricos está conformada principalmente por los nodos móviles, estos nodos móviles cuentan con características adicionales para poder soportar simulaciones de redes móviles Ad-Hoc. Los nodos móviles tienen la capacidad de poder moverse dentro de una topología determinada, pudiéndose comunicar con los nodos con los cuales tenga un alcance de radio, estos nodos tienen la capacidad de recibir y transmitir señales sobre canales inalámbricos para comunicarse con los nodos en su topología.

Las características que presentan los nodos móviles como lo es el hecho de moverse constantemente, hace que se tengan actualizaciones periódicas debido al cambio de posición de los nodos, este tipo de situaciones se ponen en ejecución en C++. Mientras que los cambios como son: canal a utilizar, el MAC, el demulticanalizador (dmux), etc., esto se pone en ejecución en OTcl.

Los cuatro protocolos de enrutamiento Ad-Hoc que tiene NS-2 son: DSDV (por sus siglas en inglés - *Destination Sequence Distance Vector*), DSR (por sus siglas en inglés - *Dynamic Source Routing*), TORA (por sus siglas en inglés - *Temporally Ordered Routing Algorithm*), AODV (por sus siglas en inglés - *Ad-Hoc On Demand Distance Vector*).

Las API's (por sus siglas en inglés- *Application Programming Interface*) se emplean para configurar los nodos móviles, mediante las API's se pueden configurar el protocolo de enrutamiento Ad-Hoc que se va a utilizar, modelo de propagación, modelo de canal, si se tiene o no enrutamiento cableado, etc.

Para la creación de un nodo móvil es necesario tener diferentes procedimientos tal como: la creación de agentes de enrutamiento especiales para una red Ad-Hoc, crear la pila de red que consiste en una capa de enlaces, tener una interfaz de cola, capa MAC (control de acceso al medio), interfaz de red con antena, además de utilizar algún modelo de propagación definido; todos estos componentes de interconectan y se conectan al canal. El esquema de un nodo móvil en NS-2 se muestra en la Figura 22.

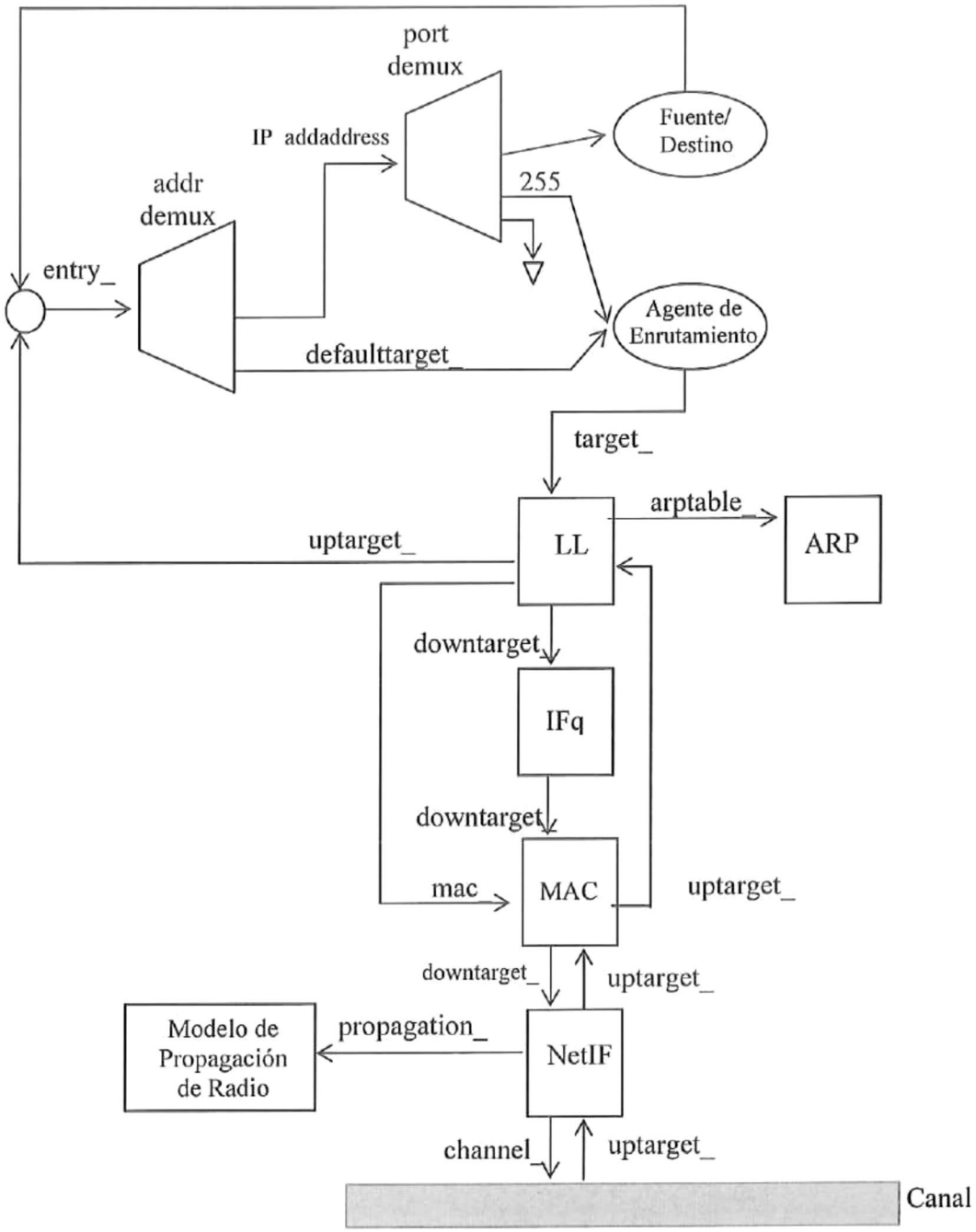


Figura 22.- Esquema de un nodo inalámbrico en NS-2.

VI.2.1 Componentes de un nodo móvil

La pila de red de un nodo móvil consiste de una capa de enlace (LL), un módulo ARP conectado con LL, una cola (IFq) de prioridad de interfaz, una capa de control de acceso al medio (MAC), una interfaz de red (net IF), todos estos conectado al canal. Estos componentes de red se crean y se enlazan con OTcl. Los componentes se describen a continuación de manera breve.

VI.2.1.1 Capa de Enlace

La capa de enlace para el nodo móvil, tiene un módulo ARP (por sus siglas en inglés - *Address Resolution Protocol*) conectado, el cuál resuelve la conversión de las direcciones IP a direcciones de hardware (MAC). Normalmente los paquetes que salen del nodo (entrantes al canal), los paquetes son enviados por la capa de enlace (LL) por el agente de enrutamiento a la capa MAC. Para todos los paquetes que entran al nodo (fuera del canal), la capa MAC manda los paquetes a la capa de enlace (LL), de ahí se van al punto de entrada (*node_entry_point*).

VI.2.1.2 ARP

El módulo de Protocolo de Resolución de Direcciones recibe las solicitudes de la capa de enlace. Si el ARP tiene la dirección de hardware del destino, lo escribe en el

encabezado MAC del paquete. Si no la tiene difunde una solicitud ARP y guarda el paquete de forma temporal. Para cada dirección de hardware destino que no se conozca, existe un buffer para cada uno de los paquetes que no se conozca dicha dirección hardware. Una vez que se conoce la dirección de hardware del siguiente salto de un paquete, el paquete se agrega en la interfaz de cola.

VI.2.1.3 Interfaz de Cola

La clase PriQueue es implementada como una cola de prioridad, que da prioridad a los paquetes correspondientes a algún protocolo de enrutamiento, insertándolos en el encabezado de la cola.

VI.2.1.4 Capa Mac

El protocolo Mac de la función de la coordinación de IEEE 802.11 DCF (por sus siglas en inglés - *Distributed Coordination Function*), usa un patrón RTS / CTS / DATA / ACK para todos sus paquetes unicast y simplemente envía datos para todos los paquetes de broadcast. La implementación usa tanto el sentido físico como virtual del portador.

VI.2.1.5 Interfaz de Red

La capa de red sirve como una interfaz de hardware que es usada por el nodo móvil para tener acceso al canal. La interfaz inalámbrica se pone como Phy / WirelessPhy. Esta interfaz conforme a las colisiones y al modelo de radio de propagación recibe los paquetes transmitidos por otras interfaces de los nodos al canal. La interfaz etiqueta cada paquete transmitido con los metadatos relacionados con la interfaz que transmite, como lo es la longitud de onda, la energía de transmisión etc. Estos metadatos se ponen en la cabecera del paquete y son utilizados por el modelo de propagación en la recepción, para determinarse si el paquete tiene la energía mínima para ser recibido y/o de ser capturado y/o de ser detectado por el nodo receptor. El modelo se aproxima a la interfaz de radio de DSSS (por sus siglas en inglés- *Direct Sequence Spread Spectrum*).

VI.2.1.6 Modelo de propagación de radio

Utiliza un modelo de atenuación de espacio Friss ($1/r^2$) para distancias cercanas y una aproximación de rayos ($1/r^4$) para distancias lejanas.

VI.2.1.7 Antena

Una antena omnidireccional que tiene una ganancia unitaria es utilizada por los nodos móviles.

VI.3 Enrutamiento en NS-2

En general, cada implementación de enrutamiento en NS-2 consiste de tres bloques:

- El agente de enrutamiento, intercambia paquetes de enrutamiento con sus vecinos.

- Lógica de la ruta, utiliza la información recopilada por los agentes de enrutamiento (o la base de datos de la topología global en el caso de enrutamiento estático) para realizar el calculo de la ruta actual.

- Los clasificadores en el nodo. Utilizan la tabla de enrutamiento calculada para realizar la tarea del envío del paquete.

Cuando se desea implementar un protocolo diferente de enrutamiento, no va a ser necesario implementar los tres bloques. Esto es que cuando se pone en ejecución un protocolo de enrutamiento, simplemente se implementa el agente de enrutamiento para el

intercambio de información, y una lógica de ruta se hace a través del algoritmo de Dijkstra en la base de datos resultante de la topología. Al momento de implementar un nuevo protocolo de enrutamiento que necesita de más de un bloque de función, especialmente cuando contiene su propio clasificador, es deseable tener otro objeto, que se denomina *módulo de enrutamiento*, quien es el que maneja los bloques y es la interfaz con el nodo para organizar sus clasificadores.

VI.4 Agentes de enrutamiento para las redes móviles Ad-Hoc en NS-2

NS-2 cuenta con diferentes tipos de agentes de enrutamientos implementados para las redes móviles Ad-Hoc. Los cuatro diferentes tipos de protocolos de enrutamiento implementados actualmente son: DSDV, DSR, AODV y TORA.

Para poder trabajar con algún protocolo de enrutamiento que no este implementado actualmente en NS-2 es necesario que se agregue el agente de enrutamiento correspondiente a el protocolo del cual se quiera hacer uso. En caso particular de este trabajo de tesis, fue necesario agregar el agente de enrutamiento de OLSR, como se puede apreciar en la Figura 23.

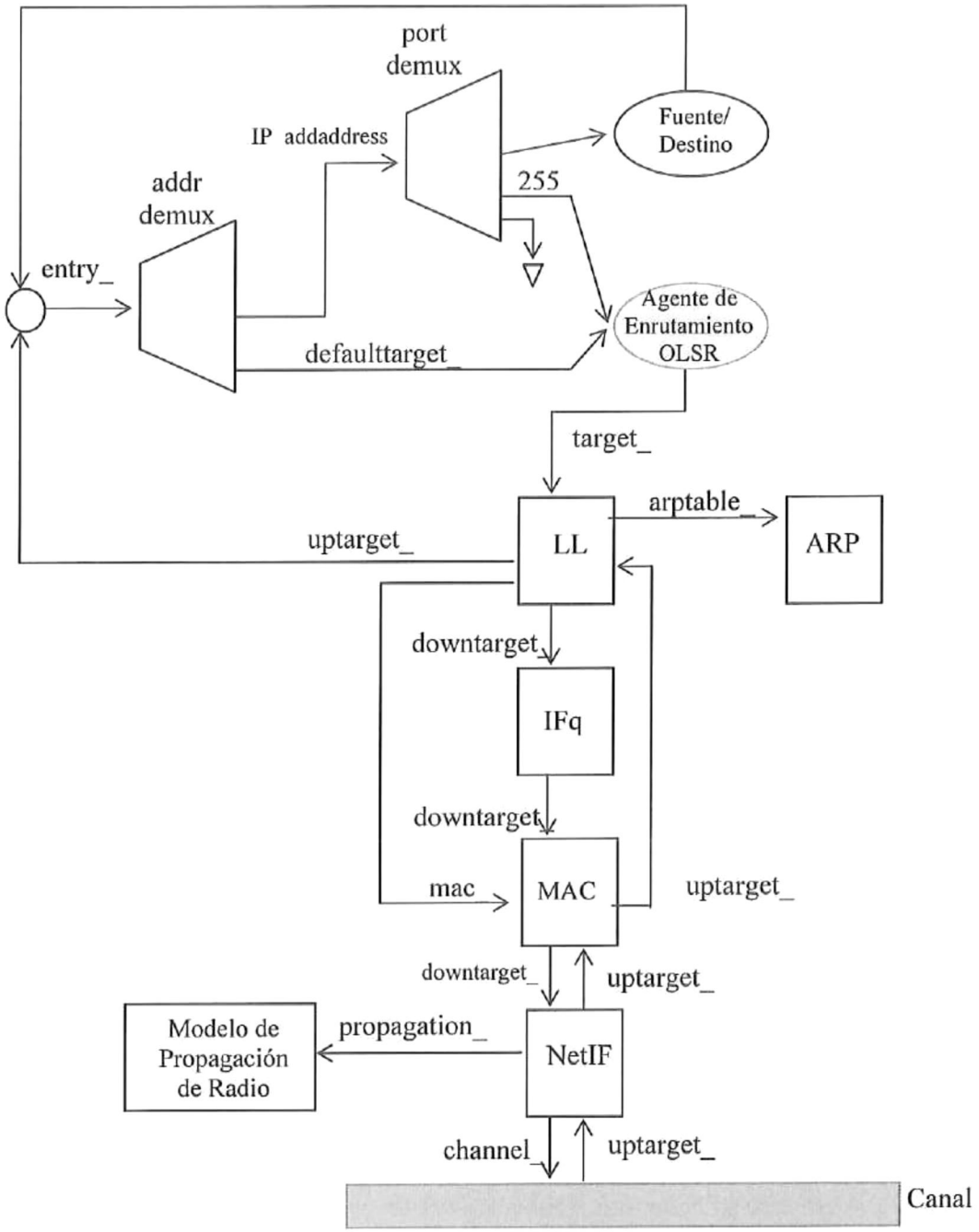


Figura 23.- Esquema de un nodo inalámbrico en NS-2 con el agente de enrutamiento OLSR.

VI.5 El protocolo HOLSR en NS-2

Debido a que en esta tesis se hace uso del protocolo HOLSR, se necesito hacer modificaciones al protocolo OLSR, ya que el protocolo HOLSR no es un agente como tal, es decir que no se puede agregar el agente de enrutamiento para el protocolo HOLSR, ya que este protocolo no esta implementado en el NS-2.

HOLSR fue desarrollado para OPNET en 2003. Sin embargo este protocolo no se encontraba implementado en NS-2, por este motivo fue necesario modificar la estructura del código del protocolo OLSR.

Dado a que el protocolo HOLSR trabaja con nodos de más de una interfaz, y la implementación del NS-2 que se utilizó no maneja interfaces múltiples fue necesario hacer uso de una extensión del NS-2 denominada TENS (por sus siglas en inglés - *The Enhanced Network Simulator*), esta extensión del NS-2 fue desarrollado por el Departamento de Ciencias de la Computación e Ingeniería del Instituto de Tecnología de la India. El TENS intenta solucionar las deficiencias del NS-2 en el modelado del protocolo IEEE 802.11 de la capa MAC, el cual es altamente simplificado en el NS original. Incorpora características adicionales como el soporte de interfaces múltiples, un protocolo de enrutamiento estático para escenarios inalámbricos, y también ofrece la inclusión de antenas direccionales simples [ISI, 2006b].

VI.5.1 Mejoras y modificaciones al NS-2

Aquí se describen brevemente las características adicionales que se le han agregado al NS-2. El interés principal del TENS fue el mejorar el protocolo 802.11 en la capa MAC. A continuación se enlistan las modificaciones realizadas:

1) Soporte adicional para interferencia de canales. La versión disponible de NS tiene soporte para un solo canal. Se agregó el soporte para 11 canales, según la especificación de 2.4 GHz DSSS PHY en el protocolo IEEE 802.11, y también se han agregado características para la taza de error de bit aleatorio (random bit error rate), y errores provocados por la interferencia del canal. La frecuencia usada para IEEE 802.11 es de 2.4000 Ghz a 2.4835 Ghz.

2) Se agregó el soporte para antena direccional, además de la omni-antena presente en la versión disponible de NS. Actualmente se está proveyendo soporte para algunos patrones de radiación de antena direccional típicos, usando un archivo de entrada. También se proporciona soporte para antena direccional limitante, en el cual se puede limitar el ancho angular para el esparcimiento de la señal especificando el grado de esparcimiento.

3) Soporte para interfaces múltiples en un nodo móvil (máximo 10), ya que la actual implementación del NS solamente maneja una interface por nodo.

4) Soporte para las transiciones adaptivas de la tasa de datos (según lo especificado en la especificación 802.11b). Las tasas soportadas son 11 Mbps, 5.5 Mbps, 2 Mbps, 1 Mbps.

5) Soporte para la región gris en NS (anteriormente había una transición simple de 2 estados de la señal para completar cero, sin ninguna clase de decaimiento)

6) Variaciones temporales al azar en fuerza de la señal (para modelar un panorama de la vida real).

7) Protocolo estático de enrutamiento para implementación inalámbrica (llamado "*wlstatic*") el cual también tiene soporte para propagación miltisalto (*multihop*).

VI.5.2 Interfaces Múltiples en el NS-2

NS-2 soporta únicamente una sola interface de red para los nodos móviles como se mencionó anteriormente, en el TENS los escenarios de simulación puede soportar nodos con interfaces múltiples haciendo uso de diferentes canales. Las modificaciones fueron realizadas en una función de agregar interfaz dentro del nodo móvil. Para agregar interfaces múltiples se agregó una variable en el *strip* de tcl, la variable denominada *numifs*, para denotar el número de interfaces que el nodo tenga. En la Figura 24 se ejemplifica tres nodos

A, B, C, el nodo B tiene dos interfaces una en el canal 1 y otra en el canal 11. El nodo A tiene una interface en el canal 1 y el nodo C tiene una interface de red en el canal 11. Cuando los nodos A y C desean comunicarse debe ser mediante el nodo B la razón es porque el canal en donde están trabajando es diferente y no pueden tener una comunicación directa.

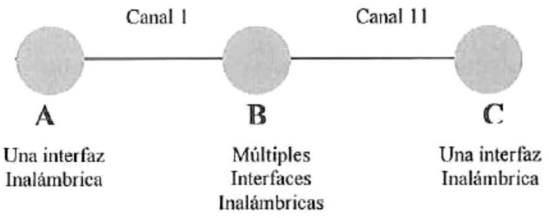


Figura 24.- El escenario consiste de 3 nodos. El nodo B tiene 2 interfaces inalámbricas.

VI.5.3 Validación Adyacente de Interferencia

EL DSSS PHY (la capa física para 802.11b) especifica 11 canales en el espectro de 2.4 Ghz para la comunicación, cada canal cuenta con 22 MHz de ancho de banda. Sin embargo, estos canales se traslapan y los canales adyacentes tienen un traslape de 5 MHz. Solamente los canales con una diferencia de canal de 5 o más son totalmente independientes de los demás, por ejemplo, los canales 1, 6 y 11. Cuando los canales están cerca el rendimiento es más bajo que lo normal. En la Figura 25 se muestra un ejemplo de los canales de comunicación en el 8002.11.

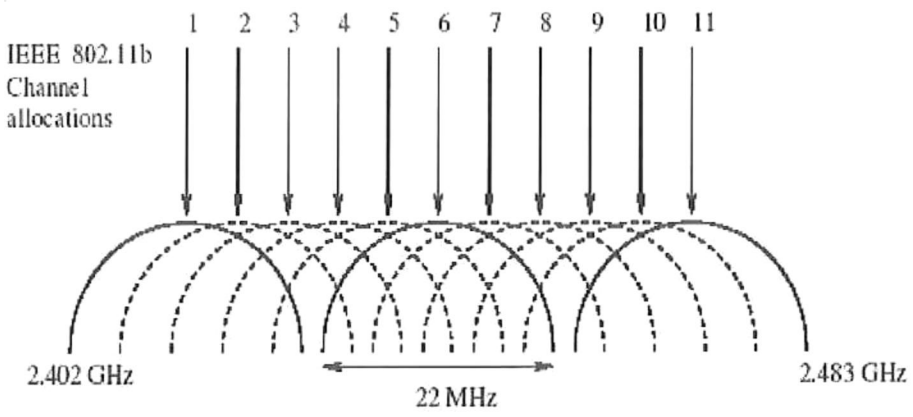


Figura 25.- Canales en IEEE 802.11b DSS PHY.

Al TENS se le agregó el agente de enrutamiento OLSR ya que no lo trae implementado, posteriormente de haber agregado el agente de enrutamiento OLSR se procedió a realizar las modificaciones necesarias para que los nodos pudieran trabajar con el protocolo OLSR y con más de una interfaz inalámbrica.

Como se explica en el Capítulo IV, acerca del funcionamiento de HOLSR se menciona que trabaja con cluster y cluster heads, y para esto se agregó el mensaje de CIA en la estructura del protocolo OLSR, mediante el mensaje CIA los nodos van a poder seleccionar a su cluster head y por tanto van a conocer a que cluster van a pertenecer, ya que estos mensajes van a tener la dirección IP y el número de saltos que hay para poder llegar al cluster head. Los nodos cluster head en el caso de este trabajo de tesis van a ser los nodos que van a tener más de una interfaz de red.

Se agregó otro tipo de mensaje que es el mensaje HTC que funciona de una forma similar al mensaje TC que es difundido únicamente por los nodos MPR, el mensaje HTC fue agregado a la estructura del protocolo OLSR ya que es un mensaje jerárquico, y no esta en la estructura plana del protocolo OLSR, el mensaje HTC es mandado por los nodos denominados cluster head por su interfaz superior.

Haciendo las modificaciones necesarias para tener un protocolo que pueda trabajar con nodos heterogéneos de manera jerárquica, que permitiera la creación de cluster, y que se pudiera utilizar nodos con interfaces múltiples como cluster heads, se pasa de trabajar con el protocolo OLSR plano a un protocolo OLSR jerárquico, el HOLSR. La pila de red de un nodo móvil para TENS va a variar de la que se usa normalmente para los nodos en el NS-2, ya que en el TENS los nodos cuentan con más de una interfaz inalámbrica y por lo tanto ahora va a depender del número de interfaces con el que cuente el nodo, el nodo puede tener una ó más de una capa de enlace (LL), una ó más de un módulo ARP conectado con las LLs, una ó más de una cola (IFq) de prioridad de interfaz, una ó más de una capa de control de acceso al medio (MAC), y una ó más de una interfaz de red (net IF) todo esto conectado al canal, estos componentes al igual que cuando solo cuenta una interfaz de red se crean y se enlazan con OTcl.

Las modificaciones que se efectuaron en el agente OLSR son:

- Se modificó la estructura de OLSR, para que los nodos tuvieran la posibilidad de tener más de una interfaz inalámbrica.

- Se agregó el mensaje CIA, este mensaje es incluido en la estructura de envió y recepción del mensaje HELLO, el mensaje CIA es enviado por primeara vez por el nodo denominado cluster head lo envía junto con el mensaje de HELLO a sus vecinos a un salto, con el fin de que sus nodos vecinos se unan a él para de esta forma crear el cluster, este mensaje es enviado cada 0.5 segundos, ya que es mandado junto con el mensaje Hello. Este mensaje es difundido por todos los nodos después de haber recibido el mensaje por cluster head, los nodos actualizaran sus tablas para saber a que cluster pertenecen y cuantos saltos tienen que dar para poder llegar al nodo cluster head, mediante este mensaje los nodos podrán hacer la elección del cluster más conveniente, es decir se unirán al cluster head más cercano, en el formato del mensaje CIA está formado por dos campos: dirección IP del cluster head, y número de saltos al cluster head.

- Se agregó el mensaje de HTC, este mensaje fue agregado con la misma estructura que el mensaje TC, ya que es un mensaje TC de tipo jerárquico, sin embargo la diferencia principal es que este mensaje únicamente es enviado por los nodos cluster head, en su interfaz superior, para dar a conocer a los nodos del nivel superior del estado de los nodos en el nivel inferior.

Se modificaron los scripts TCL para que se pudieran utilizar nodos de más de una interfaz.

VI.6 Servicios diferenciados en NS-2

Para hacer uso de DiffServ en NS-2 no es necesario escribir un módulo de DiffServ propio para NS-2, ya que existen módulos ya programados, es decir no se necesita agregar ningún tipo de agente para hacer uso de este servicio, ya que estos módulos están integrados en NS-2, se utilizó el módulo desarrollado por Nortel Networks [Pieda *et al.*, 2000], el cual ya esta incluido en el NS-2.

VI.6.1 Diffserv en NS-2

Para la implementación de la arquitectura DiffServ en NS se cuenta con cuatro módulos: el primero para la funcionalidad básica de enrutadores con DiffServ (dsRED), el siguiente para la definición de los enrutadores de núcleo (enrutador core), otro para los enrutadores de frontera (enrutador edge) y el último para la administración de la política (policy).

VI.6.2 Módulo dsRED

dsRED es el módulo base para la implementación de Diffserv. Su propósito es implementar toda la funcionalidad y declarar todos los parámetros comunes a los enrutadores de frontera y de núcleo. En la jerarquía de NS-2, extiende la clase cola (Queue)

dsREDQueue forma una estructura de colas consistente en cuatro colas físicas, cada una de las cuales contiene tres colas RED virtuales, que corresponden a los niveles de precedencia. Cada cola física corresponde a un tipo de tráfico y cada combinación de cola y precedencia se asocia a un Code Point.

Los paquetes se alinean en una determinada cola y número de precedencia de acuerdo a su marca de Code Point. Su tratamiento se realiza a los parámetros correspondientes a esa cola y número de precedencia. De esta manera el Code Point especifica el nivel de servicio.

La elección de cuatro colas físicas se realiza según la recomendación del RFC 2597 [Heinanen *et al.*, 1999] y corresponde a las clases de servicio definidas para AF (por sus siglas en inglés - *Assured Forwarding*). No todas las colas físicas y virtuales requieren ser usadas. El usuario puede configurar una instancia dsRED con menos colas físicas o virtuales, sin embargo el valor no puede ser excedido sin modificar el código y recompilando NS.

VI.6.3 Tabla de PHB (Per Hop Behavior)

Una instancia de *dsREDQueue* contiene una estructura de datos conocida como la Tabla PHB.

Los dispositivos de frontera manejan el marcado de paquetes según code points y los dispositivos de núcleo simplemente responden a los code points existentes. Sin embargo, ambos dispositivos necesitan determinar como mapear un code point a una cola determinada y un nivel de precedencia.

La tabla PHB maneja este mapeo a través de un arreglo de tres campos que debe ser asignado en la programación del simulador.

- Code point.
- Clase (Cola física).
- Precedence (Cola virtual).

VI.6.4 Módulo de frontera

El módulo de frontera implementa el enrutador de frontera Diffserv. Define la clase *edgeQueue* que modela el enrutador respectivo. La clase *edgeQueue* deriva de la clase *dsREDQueue* y es responsable por mantener y procesar múltiples colas físicas y virtuales de acuerdo a sus parámetros, además de marcar los paquetes con los code point y aplicar la política sobre los agregados de tráfico.

VI.6.5 Módulo de política

El módulo o clase de política es utilizado por la clase *edgeQueue* para sostener la funcionalidad de la política en Diffserv. La clase de política administra la creación, manipulación y aplicación de las políticas del enrutador de frontera. Una política determina el tratamiento que un agregado de tráfico recibirá en el dispositivo de frontera. Los dispositivos de frontera usan información y política para determinar con que code point deben marcar los paquetes.

Una política se establece entre un nodo de origen y uno de destino. Todos los flujos con el mismo par origen destino son tratados como un agregado de tráfico único. Cada política define un tipo de aplicador de política (policer), una tasa de traspaso objetivo y otros parámetros dependientes de la política específica. Como mínimo, una política define dos valores de Code Point y la elección depende de la comparación entre la tasa objetivo del agregado y la tasa de envío actual (bits/seg).

Cada agregado de tráfico tiene asociado un aplicador de política, un medidor y un code point inicial. El tipo de medidor especifica el método para medir las variables de estado requeridas por el aplicador de política, por ejemplo el medidor TSW tagger mide la tasa de tráfico promedio a través de una ventana de tiempo específica.

Tras la llegada de los paquetes al dispositivo de frontera, son examinados para determinar a que agregado de tráfico pertenecen. Se invoca un medidor específico para ese agregado para actualizar todas sus variables de estado. Se utiliza ya sea el code point inicial u otro degradado respecto del primero y el paquete es puesto en la cola correspondientemente.

VI.6.6 Tabla de políticas

La clase políticas (class Policy), es una tabla de políticas para almacenar las políticas de cada agregado de tráfico.

Para la aplicación de políticas, NS-2 ha implementado seis tipos distintos:

1) ***Time Sliding Window with 2 Color Marking (TSW2CMPolicer)***: utiliza la tasa de información comprometida (CIR por sus siglas en inglés - *Committed Information Rate*) y dos niveles de precedencia para descartar. La precedencia mas baja es usada probabilistícamente cuando el CIR es excedido.

2) ***Time Sliding Window with 3 Color Marking (TSW3CMPolicer)***: utiliza la tasa de información comprometida (CIR), y la tasa pico de información (PIR por sus siglas en inglés – *Peak Information Rate*) y tres niveles de

precedencia para descartar. La precedencia de descarte media es usada probabilísticamente cuando el CIR es excedido y la precedencia más baja se usa probabilísticamente en conjunto cuando el PIR es excedido.

3) ***Token Bucket (tokenBucketPolicer):*** utiliza CIR, tamaño de ráfaga comprometido (CBS por sus siglas en inglés – *Committed Burst Size*) y dos niveles de precedencia para descarte. Un paquete que va llegando es marcado con baja precedencia si y solo si es mayor (en bits) que el token bucket.

4) ***Single Rate Three Color Marker (srTCMPolicer):*** utiliza CIR, CBS y EBS para elegir entre tres precedencias de descarte.

5) ***Two Rate Three Color Marker (trTCMPolicer):*** utiliza CIR, CBS, PIR y un tamaño de ráfaga pico (PBS por sus siglas en inglés – *Peak Burst Size*) para elegir entre tres precedencias de descarte.

6) ***NullPolicer:*** no baja precedencias a ningún paquete.

Todas las políticas son almacenadas en la tabla de políticas *PolicyClassifier*. Esta tabla es un arreglo que incluye campos para los nodos de origen y destino, tipo de aplicador de política, tipo de medidor y code point inicial, y varios estados de información, los cuales se muestran a continuación:

- Las tasas de envío de CIR y PIR son especificadas en bits por segundo.

- CIR: Committed Information Rate.

- PIR: Peak Information Rate.

- Los buckets CBS, EBS y PBS son especificados en bytes.

- CBS: Committed Burst Size.

- EBS: Excess Burst Size.

- PBS: Peak Burst Size.

- C bucket: tamaño actual del committed bucket (algo así como el bucket comprometido).

- E bucket: tamaño actual del bucket excedido.

- P bucket: tamaño actual del peak bucket.

Tiempo de llegada del último paquete.

VI.6.7 Módulo de núcleo

Esta clase emula los enrutadores de núcleo (core) en la arquitectura Diffserv. De esta manera se conecta con los enrutadores de frontera y reenvía los paquetes de acuerdo a la marca definida por el enrutador de frontera. Paquetes con code point de baja prioridad, son eliminadas a tasas considerablemente más altas que los paquetes marcados con code point de alta prioridad.

Primero que nada, no se debe perder de vista lo que esta en el apartado que se llama *Tabla de políticas* ya que tanto las 6 diferentes políticas como el definidor de términos nos van a ayudar a saber como configurar la simulación.

Dependiendo del tipo de política que queramos implementarle a la simulación se tendrán diferentes tipos de parámetros:

Tabla 3.- Políticas de DiffServ en NS-2.

Null	Initial code point				
TSW2CM	Initial code point	CIR			
TSW3CM	Initial code point	CIR	PIR		
TokenBucket	Initial code point	CIR	CBS		
srTC	Initial code point	CIR	CBS	EBS	
trTCM	Initial code point	CIR	CBS	PIR	PBS

En este trabajo de tesis se necesita que se pueda aplicar DiffServ con el protocolo proactivo HOLSR. El uso de diffserv en NS-2 no está implementado para redes móviles Ad-Hoc, para poder trabajar con diffserv en este tipo de redes, se tiene que realizar unos cambios a la estructura general del módulo de diffserv ya que en el caso de las redes Ad-Hoc como todos los nodos trabajan como enrutadores y nodos a la vez, es decir que no van a existir nodos intermedios (core), si no los nodos serán declarados como enrutadores extremos (edge), se tiene que implementar un tipo de administración de nodos diferente.

- Todos los nodos que van a participar en la red son configurados como enrutadores extremo.

Las políticas son establecidas entre un nodo fuente y un nodo destino, sin embargo todos los nodos de la red deben agregar estas políticas de calidad de servicio que son establecidas entre el nodo fuente y destino.

VI.7 Resumen

En este capítulo se analizó lo relacionado con el simulador NS-2 en las redes móviles Ad-Hoc, así como la extensión del NS-2 denominada TENS que sirvió como parte fundamental para la realización de este trabajo de tesis, ya que gracias al uso del TENS se pudo implementar nodos con más de una interfaz inalámbrica, en el NS-2 original no cuenta con la opción de utilizar nodos con más de una interfaz inalámbrica.

Se analizó la forma de integrar el agente de enrutamiento OLSR al TENS, ya que esta extensión no trae implementado el protocolo OLSR. Fue necesario modificar la estructura del OLSR a fin que pudiera trabajar de manera jerárquica y con nodos heterogéneos, es decir ya que se implemento el protocolo OLSR en el TENS, el OLSR se modificó para ser un protocolo de enrutamiento que pudiera trabajar con nodos de más de una interfaz ya que el TENS tiene implementada la opción de trabajar con nodos de más de una interfaz inalámbrica. Contando con nodos de más de una interfaz inalámbrica trabajando bajo el protocolo de enrutamiento OLSR se procede a tener una gestión de red de tipo jerárquico, para así tener implementado el protocolo proactivo de enrutamiento jerárquico denominado HOLSR, que es con el protocolo con el que se trabajo en esta tesis.

Se estudió el mecanismo de Calidad de Servicio DiffServ en el simulador NS-2, y se estudió el manejo de políticas que usa DiffServ en el NS-2. Además se modificaron algunos módulos de DiffServ para que funcionaran con redes móviles Ad-Hoc, ya que estaban implementados únicamente para redes cableadas. Se hizo una unificación entre el protocolo de enrutamiento HOLSR y el mecanismo de DiffServ para de esta forma realizar las simulaciones que permitan tener una evaluación del desempeño de calidad de servicio que proporciona HOLSR y DiffServ.

En el capítulo siguiente se analizan las simulaciones y los escenarios utilizados para realizar estas mismas. Además se muestran los resultados obtenidos.

Capítulo VII Simulación y resultados

VII.1 Introducción

Es necesario utilizar programas de simulación para tener una perspectiva del comportamiento de las redes, en este trabajo de tesis el programa de simulación sirvió para analizar el comportamiento de las redes móviles ad hoc utilizando el mecanismo de DiffServ para proporcionar calidad de servicio y al protocolo HOLSR como protocolo de enrutamiento; es importante que antes de realizar evaluaciones en escenario físicos se utilicen programas de simulación, ya que estos nos permiten visualizar el comportamiento que pueden llegar a tener las redes dependiendo de las diferentes características que se le incorporen así como los escenarios de uso que se manejen.

VII.2 Calidad de servicio con redes Ad-Hoc

Recientemente debido al auge que han tenido las aplicaciones multimedia y a la demanda de aplicaciones en tiempo real, así como el uso potencial de las redes MANET, es imprescindible contar con calidad de servicio en este tipo de redes, para mejorar el funcionamiento de las MANETs y permitir que la información crítica fluya incluso bajo condiciones de alta demanda de los recursos disponibles en la red.

Utilizar un protocolo de enrutamiento proactivo, como lo es el HOLSR, es más conveniente cuando se desea proporcionar calidad de servicio a las redes móviles ad hoc ya que usualmente se reduce el tiempo para encontrar nuevas rutas y también se evita la perdida de recursos en la red al procurar descubrir rutas nuevas cada vez que los nodos necesiten mandar información a un nodo destino, ya que como es un protocolo proactivo tiene en sus tablas de enrutamiento las rutas a todos los nodos de la red, ya que las actualiza constantemente.

El modelo de DiffServ se utiliza como un mecanismo conveniente para proporcionar calidad de servicio a las redes incluyendo a las redes MANETs. En una MANET, existen funcionalidades básicas que el mecanismo de calidad de servicio tiene que realizar en cada nodo: el marcado y el desmarcado de los paquetes así como la aplicación de políticas de calidad de servicio; cada uno de los nodos en una MANET es un enrutador, el marcado de los paquetes es hecho por los nodos que deseen mandar información con calidad de servicio y desmarcado por los nodos al que vaya dirigida dicha información; los nodos necesitan estar bajo el mecanismo de calidad de servicio DiffServ para poder establecer políticas de calidad de servicio, cada uno de los nodos marcará los paquetes que genera conforme las políticas establecidas, y el nodo destino desmarcará los paquetes que fueron dirigidos a él, sin embargo todos los nodos intermedios a lo largo de la trayectoria van a tener conocimiento de los requerimientos de calidad de servicio que se establecieron entre el nodo fuente y el nodo destino. Cada paquete que fue marcado será tratado según las políticas de calidad de servicio fijadas para dicho paquete. Debido a lo

anterior es necesario realizar la integración del protocolo de enrutamiento proactivo HOLSR y el mecanismo de calidad de servicio DiffServ.

En una red MANET heterogénea de tipo jerárquica que esta formada por cluster y con nodos heterogéneos como cluster heads es necesario implementar un mecanismo de calidad de servicio ya que en los nodos cluster heads se crearán congestionamientos de tráfico dada la comunicación inter-cluster, lo que provoca que exista perdida de información. El hecho de utilizar modelos de red jerárquicos y aplicar en ellos protocolos que puedan proporcionar mecanismos de enrutamiento adecuados representa una gran ventaja si se compara con los modelos de red planos, si aunado a esto se implementan mecanismos de calidad de servicio se incrementan las prestaciones para los usuarios que utilizan este tipo de redes.

Para realizar el soporte de calidad de servicio en redes MANET heterogéneas es necesario que todos los nodos tengan la capacidad de poder implementar políticas de calidad de servicio a sus paquetes.

VII.3 Protocolo HOLSR

El nodo cluster head manda un mensaje de CIA a los nodos que se encuentran a un salto de el. El nodo cluster head es el único nodo que puede generar los mensajes de CIA, el algoritmo para el envió del mensaje CIA decidirá si se trata de un nodo cluster head, de

ser así este nodo generará los mensajes de CIA y los transmitirá a sus nodos vecinos junto con el mensaje de HELLO, es decir el mensaje de CIA es incluido dentro del mensaje de HELLO, como se describe en el Capítulo IV.

El protocolo HOLSR manda mensajes de control de la topología TC por nodos especializados para realizar este tipo de tarea que son los nodos MPR, envía mensajes jerárquicos de control de la topología HTC mediante los nodos cluster heads en su interfaz de índice superior es decir los mensajes HTC se envían hacia los niveles jerárquicos superiores; estos dos tipos de mensajes son enviados cada dos segundos.

VII.4 Escenario de simulación

Para poder evaluar el desempeño de la integración entre los protocolos HOLSR y DiffServ se realizaron dos clases de escenarios, el primero era cuando se contaba con políticas de calidad de servicio y el segundo cuando no existían dichas políticas de calidad de servicio, se realizaron simulaciones de escenarios bajo igualdad de condiciones. Los escenarios consistían de dos cluster heads, es decir había dos nodos con interfaces múltiples los cuales se comunicaban entre si por una interfaz inalámbrica, y dichos cluster heads eran parte de dos clusters diferentes, en cada uno de los clusters coexistían 29 nodos los cuales se comunicaban con su cluster head correspondiente. En total había 60 nodos de los cuales 2 eran nodos cluster head, cada uno tenía dos interfaces inalámbricas, el primer cluster head tenía una interfaz que trabajaba en el cana 1 y otra que trabajaba en el canal 6, el segundo

nodo cluster head tenía dos interfaces inalámbricas también una que trabajaba en el canal 11 y otra que trabajaba en el canal 6. Los nodos del primer cluster trabajan con el canal de comunicaciones 1 mientras que los nodos del segundo cluster trabajaban en el canal 11, los nodos cluster head usaban el canal 6 para comunicarse entre ellos. Todos los nodos transmiten con la misma potencia de 0.5 w con un rango de radio de cobertura de 200 metros.

Los mensajes de HELLO es enviado cada 0.5 segundos al igual que el mensaje de CIA, el mensaje de TC es enviado cada 2 segundos al igual que el mensaje HTC, los mensajes respecto al protocolo de enrutamiento HOLSR eran independientes del mecanismo de calidad de servicio que se aplicó, así que para ambos escenarios los mensajes de gestión del protocolo HOLSR se enviaban con el mismo intervalo de tiempo.

El tipo de tráfico que se utilizó fue tráfico de tipo CBR (por sus siglas en inglés - *Constant Bit Rate*), se modelaron 7 fuentes de tráfico que generaban paquetes CBR, una fuente generaba paquetes de tamaño de 1000 bytes a intervalos de 0.125 segundos, es decir transmitía a 64 kbps, y 6 fuentes generaban paquetes de 1000 bytes a intervalos de 0.01333 segundos esto es que transmitían a 600 kbps. Las simulaciones tenían un tiempo de duración de 30 segundos.

Como se mencionó anteriormente se trabajo con escenarios que no implementarán el mecanismo de calidad de servicio DiffServ y con escenarios que si lo implementaban, para poder apreciar las diferencias que existen cuando se cuenta con calidad de servicio y

cuando no se cuenta con ella. En los escenarios que se implementa calidad de servicio se establece que para la fuente de tráfico que genera paquetes CBR con una tasa de 64 kbps se aplica una política Token Buket para manejo de tráfico de VoIP, mientras que para las otras 6 fuentes de tráfico se les asigna la política de mejor esfuerzo, de esta forma va a dar prioridad al tráfico con una política de tráfico VoIP sobre el tráfico que generen las otras 6 fuentes. Para los escenarios donde no se implemente DiffServ no es necesario establecer ningún tipo de política, ya que todo el tráfico será tratado de la misma forma; es decir como tráfico de mejor esfuerzo.

El escenario de simulación consiste de dos cluster head cada uno con dos interfaces inalámbricas, una de las interfaces inalámbricas con las que cuenta cada cluster head es usada para mantener comunicación entre los dos cluster heads y la otra interfaz es utilizada para comunicarse con los nodos que se encuentran en sus respectivos cluster. Cada cluster está formado por 29 nodos, y cada uno de estos nodos cuenta con una interfaz inalámbrica para comunicarse con los miembros de su respectivo cluster, los nodos del cluster 1 no se pueden comunicar directamente con los nodos del cluster 2, esto lo harán haciendo uso de los cluster heads, ya que nos nodos del cluster 1 trabajan en el canal 1 y los nodos del cluster 2 trabajan en el canal 11. El escenario de simulación se muestra en la Figura 26.

El acomodo de los 29 nodos en cada uno de los clusters es hecho de manera aleatoria; se mandaba el tráfico CBR de los nodos de un cluster al otro cluster. La comunicación inter-clusters es realizada por los nodos cluster head, estos nodos funcionan como nodos intermedios para mantener la comunicación inter-clusters, debido a esto se

crean congestionamientos en los nodos cluster head, lo que provoca que se tengan pérdidas de información al saturar los canales de comunicación de los cluster head. Mediante el uso de mecanismos de calidad de servicio como DiffServ, se observa que existe una mejoría ya que permite primero el paso de la información con mayor prioridad y posteriormente la información menos relevante.

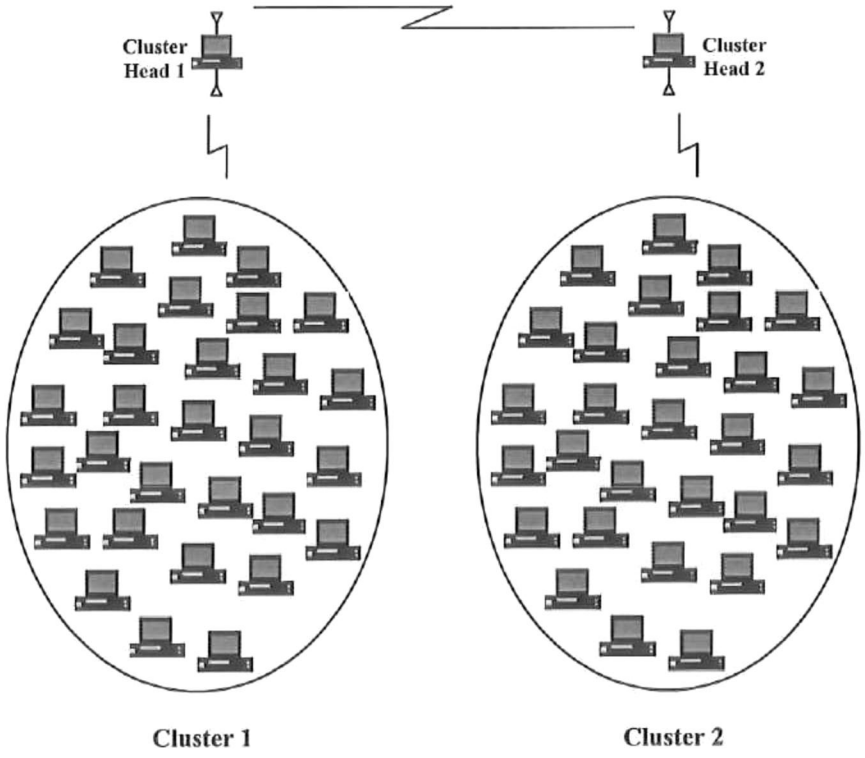


Figura 26.- Escenario de simulación.

Los nodos en el escenario de simulación contaban con movilidad, se configuró el escenario para que se dividiera en dos partes; es decir que los nodos del cluster 1 tenían movilidad en la primera parte del escenario de simulación y los nodos del cluster 2

únicamente se movían en la segunda parte del escenario, esto es para que los nodos de un cluster no se mezclaran con el otro cluster ya que debido a que se encontraban trabajando en canales diferentes provocaría que se encontraran con mucha facilidad destinos inalcanzables; en las simulaciones realizadas se vario la velocidad promedio de los nodos con 3 diferentes velocidades, 5 m/s, 10 m/s y 20 m/s, en los dos tipos de escenario el que contaba con DiffServ y el que no contaba con DiffServ, se implementaron los 3 tipos de velocidades promedio.

Para utilizar interfaces múltiples en los nodos inalámbricos se utilizó la variable agregada en el TENS, la variable de *numif* en los *scrips* de tcl, a esta variable se le asigna el número de interfaces que va a tener el nodo en el escenario de simulación.

A continuación se ejemplifica como se utilizaba la variable de *numif*, cuando se trataba de un nodo de una interfaz, la variable *numif* esta a 1 y esta utilizando el canal 1 en el ejemplo de a continuación:

$ns_ node-config -numif 1

[$node_(0) set netif_(0)] set channel_number_ 1

En el siguiente ejemplo se muestra como se utilizaba la variable de *numif*, cuando se trataba de un nodo de dos interfaces de red, la variable *numif* esta a 2 y se utilizan dos canales, el canal 1 y el canal 6:

$ns_ node-config -numif 2

[$node_(1) set netif_(0)] set channel_number_ 1

[$node_(1) set netif_(0)] set Pt_ $Pt_tx #potencia

[$node_(1) set netif_(1)] set channel_number_ 6

[$node_(1) set netif_(1)] set Pt_ $Pt_tx #potencia

Haciendo uso del TENS se pueden utilizar 11 canales diferentes es decir de 11 frecuencias distintas como se explica en el capítulo VI, para este trabajo de tesis se utilizaron 3 canales diferentes, el canal 1, canal 6 y el canal 11.

VII.5 Resultados de la simulación

El utilizar mecanismos de calidad de servicio en las redes debe tener como objetivo brindar una mejoría en el desempeño de las redes, es por eso que en este trabajo de tesis se presentan las mejorías en cuanto el número de paquetes recibidos, número de paquetes perdidos y throughput como parámetros para medir el desempeño de una red móvil ad hoc

con enrutamiento proactivo usando el protocolo de enrutamiento jerárquico HOLSR y cuando se aplica el mecanismo DiffServ.

Para obtener los resultados de las simulaciones primeramente se creo una fuente de tráfico de VoIP es decir una fuente a 64 kbps; en el escenario con calidad de servicio la fuente de tráfico contaba con la política de Token Buket para VoIP, en otro escenario sin calidad de servicio se creo la misma fuente de tráfico pero sin ningún criterio de calidad de servicio, para obtener los resultados se realizaron 100 simulaciones para cada una de los dos clases de escenarios es decir el escenario que contaba calidad de servicio y el que no contaba con calidad de servicio, las 100 simulaciones realizadas para sacar el promedio se hicieron de la siguiente forma: 100 simulaciones para el escenario con calidad de servicio y con una velocidad promedio de los nodos de 5 m/s, 100 simulaciones para el escenario con calidad de servicio y con una velocidad promedio de los nodos de 10 m/s y 100 simulaciones para el escenario con calidad de servicio y con una velocidad promedio de los nodos de 20 m/s, se obtuvo un promedio de los escenarios para cada una de las velocidades promedio de los nodos, lo mismo se realizó para los escenarios que no contaban con políticas de calidad de servicio, se manejaron las mismas condiciones para los dos tipos de escenarios; el promedio obtenido por las simulaciones fue utilizado para obtener los parámetros del throughput, paquetes recibidos y paquetes perdidos, en cada uno de los escenarios para evaluar el impacto que tiene el tráfico en la red sobre la fuente que requiere QoS se fueron incrementado sesiones de tráfico CBR a 600 kbps de una en una hasta establecer 6 sesiones de tráfico CBR a 600 kbps y analizar como iba modificándose el desempeño de la red a medida que se aumentaban las sesiones CBR. Las 6 sesiones de

tráfico que se incrementaron contaban con la política de de Mejor Esfuerzo, solamente existía una sesión de tráfico que contaba con una política de Token Buket es decir la sesión de VoIP como se mencionó anteriormente; eso en cuanto al escenario que contaba con calidad de servicio, en el otro escenario ninguna de las sesiones de tráfico CBR que se incrementaron incluyendo la sesión de VoIP tenía implementada políticas de calidad de servicio.

Los resultados obtenidos en la Figura 27, la Figura 28 y la Figura 29 son tomados del desempeño que tiene el tráfico de VoIP, ya que este tráfico es el de mayor importancia y prioridad para efectos de las simulaciones realizadas para este trabajo; por lo tanto las métricas deben ser tomadas en base al desempeño que tenga el tráfico de VoIP, se maneja una velocidad promedio de los nodos de 5 m/s, 10 m/s y de 20 m/s. Como se ha venido mencionado a lo largo del capitulo se implementaron dos clases de escenarios uno que contaba con la política de DiffServ y uno que no contaba con dicha política y para ambos escenarios se manejaron las velocidades de los nodos antes mencionadas, en el escenario que contaba con DiffServ el tráfico de VoIP era el tráfico de mayor prioridad, y en el que no tenia dicha política todo el tráfico tenía la misma prioridad.

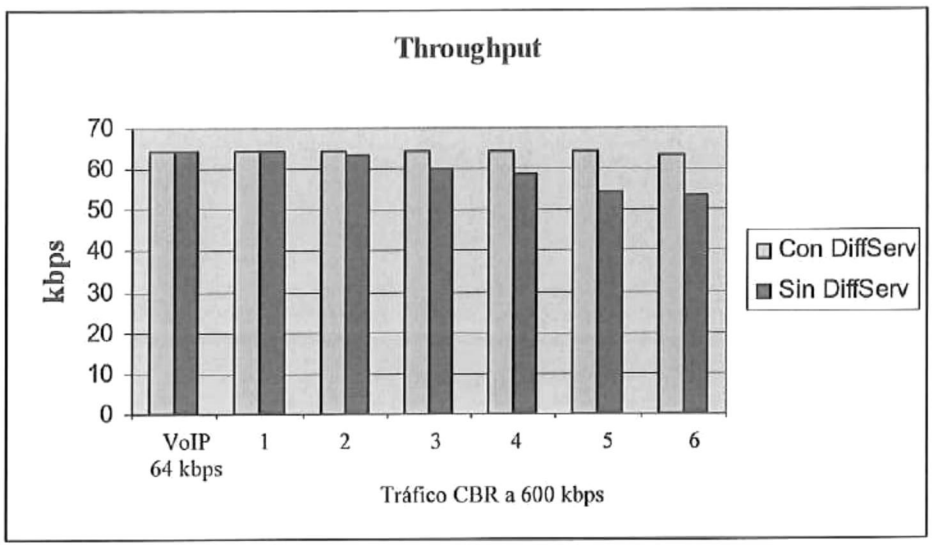


Figura 27.- Gráfica del Throughput, con un movimiento de nodos de 5 m/s.

En la Figura 27 se analiza la gráfica que muestra el parámetro de throughput aquí los nodos tienen una velocidad promedio de 5 m/s, se observa que cuando se tiene solamente la sesión de VoIP se puede observar que el desempeño de la red es el mismo cuando se tiene DiffServ que cuando no se tiene, sin embargo conforme se van aumentando sesiones de CBR una a una hasta llegar a 6 sesiones de CBR a 600 kbps, los canales de comunicación se van congestionando en el cluster head y el desempeño de la red se degrada, se puede observar que cuando no se cuenta con DiffServ la degradación de la red es significativa, sin embargo se puede ver que cuando se tiene implementado DiffServ el desempeño de la red no llega a degradarse.

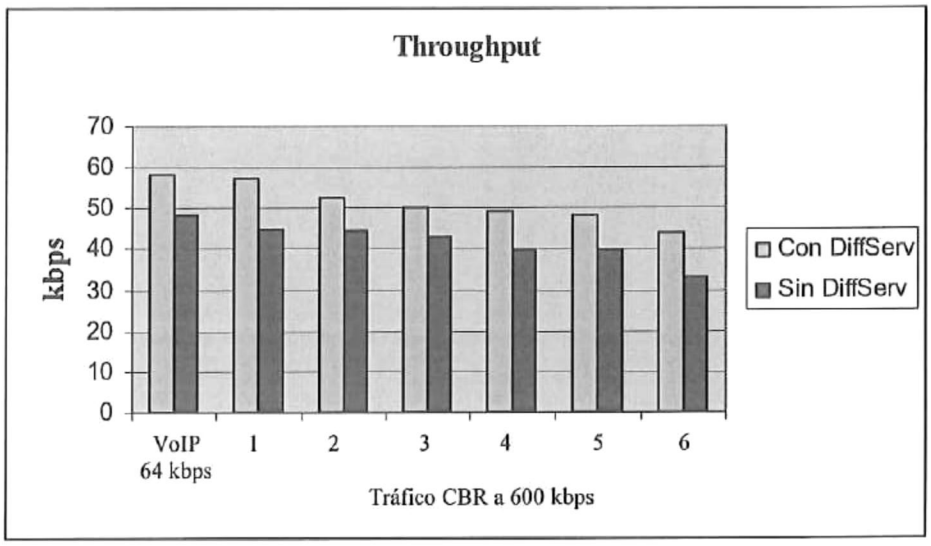


Figura 28.- Gráfica del Throughput, con un movimiento de nodos de 10 m/s.

En la Figura 28 se analiza el parámetro throughput al igual que en la gráfica anterior, sin embargo en este escenario se incremento la velocidad promedio de los nodos a 10 m/s, se puede ver que el desempeño de la red se vio afectado debido al movimiento, esto es debido a que los nodos se mueven con mayor rapidez en el escenario entonces algunos destinos van a ser inalcanzables en algún tiempo a lo largo de la simulación; aunque el desempeño de la red debido a la movilidad de los nodos haya disminuido se puede observar en la figura P23 que cuando se cuenta con DiffServ el desempeño es mejor a cuando no se cuenta con este mecanismo de calidad de servicio. Desde la primera conexión de VoIP se ve que el desempeño es mejor cuando se tiene DiffServ, y más aun cuando se van incrementando las sesiones CBR, la red sin DiffServ se ve afectada de manera significativa.

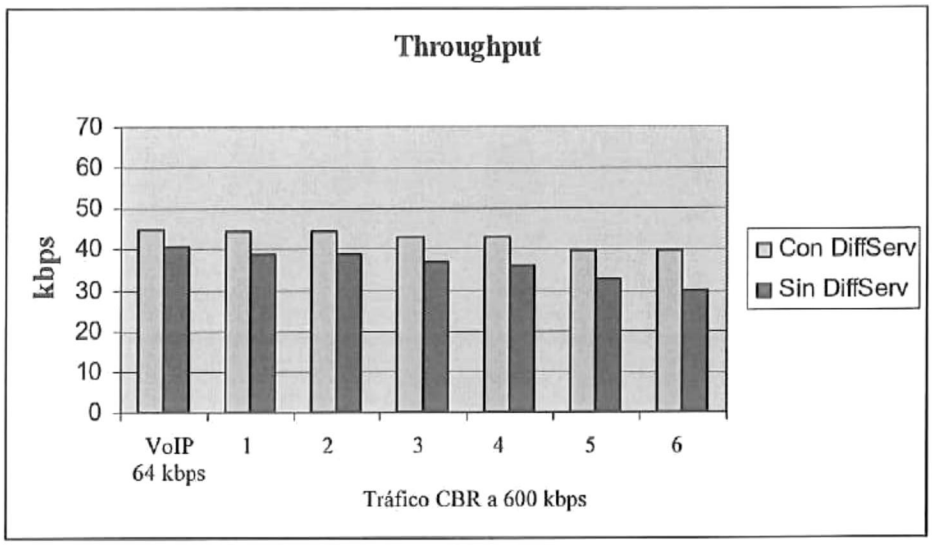


Figura 29.- Gráfica del Throughput, con un movimiento de nodos de 20 m/s.

Finalmente la Figura 29 es la última gráfica en donde se analizó el parámetro de throughput, en estos escenarios se incrementó la velocidad promedio de los nodos a 20 m/s, se observa que la métrica de throughput se degrada aún más dado que el movimiento de los nodos se incremento, sin embargo el desempeño de la misma siempre fue mejor cuando se aplicó DiffServ, a diferencia del escenario donde no se contaba con DiffServ.

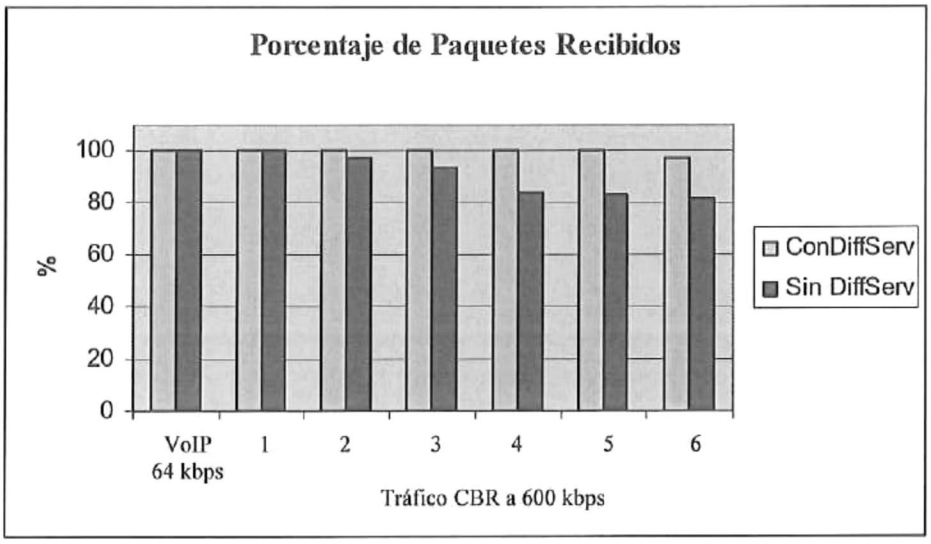


Figura 30.- Gráfica del Paquetes Recibidos, con un movimiento de nodos de 5 m/s.

En la Figura 30, se ejemplifica una gráfica donde se puede ver el porcentaje de paquetes que se recibieron, los nodos contaban con una velocidad promedio de 5 m/s, como se puede observar; cuando únicamente se contaba con una sesión de tráfico de VoIP se nota un comportamiento igual en ambos escenarios, esto es, cuando se tiene DiffServ y cuando no se tiene implementado el DiffServ. Sin embargo conforme se van aumentando el número de sesiones CBR (a 600 kbps) se va produciendo un congestionamiento en el cluster head y por lo tanto el número de paquetes que se reciben es menor. Como se puede observar en la figura anterior en el escenario donde esta aplicado el mecanismo de DiffServ el tráfico de VoIP no se degrada dado que este mecanismo le da prioridad de salida a este tráfico, en cambio en el escenario donde no esta aplicado este mecanismo el tráfico se degrada dado que no da prioridad al tráfico de VoIP, todo lo trata de la misma forma, es decir como tráfico de mejor esfuerzo.

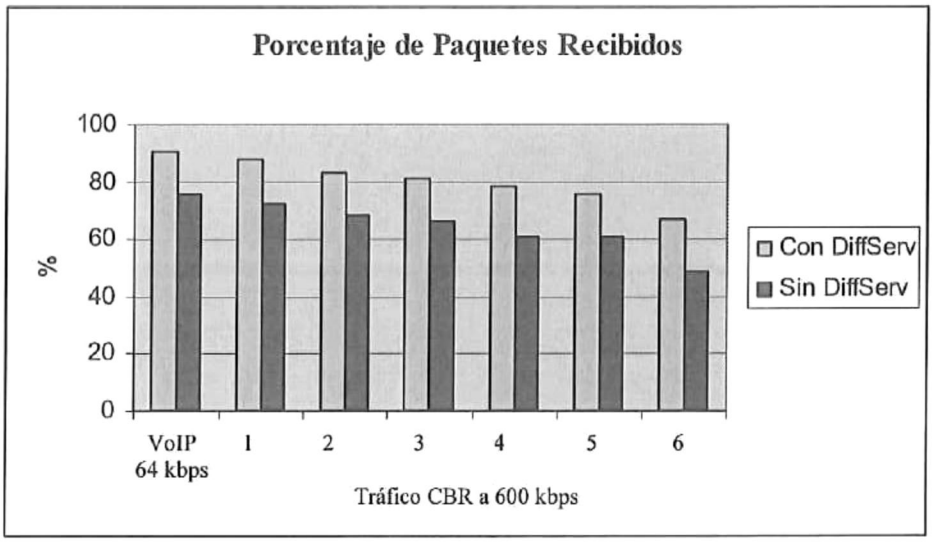


Figura 31.- Gráfica de Paquetes Recibidos, con un movimiento de nodos de 10 m/s.

En la Figura 31, muestra una gráfica del porcentaje de paquetes que se recibieron, cuando los nodos contaban con una velocidad promedio de 10 m/s, se puede ver que el tráfico sin el mecanismo de DiffServ se degrada en mayor medida que el que tiene DiffServ, los paquetes que se llegan a recibir sin DiffServ son menos que los que cuentan con este mecanismo. Y a medida que se van agregando sesiones de CBR (a 600 kbps cada una) el número de paquetes recibidos llega a ser menor.

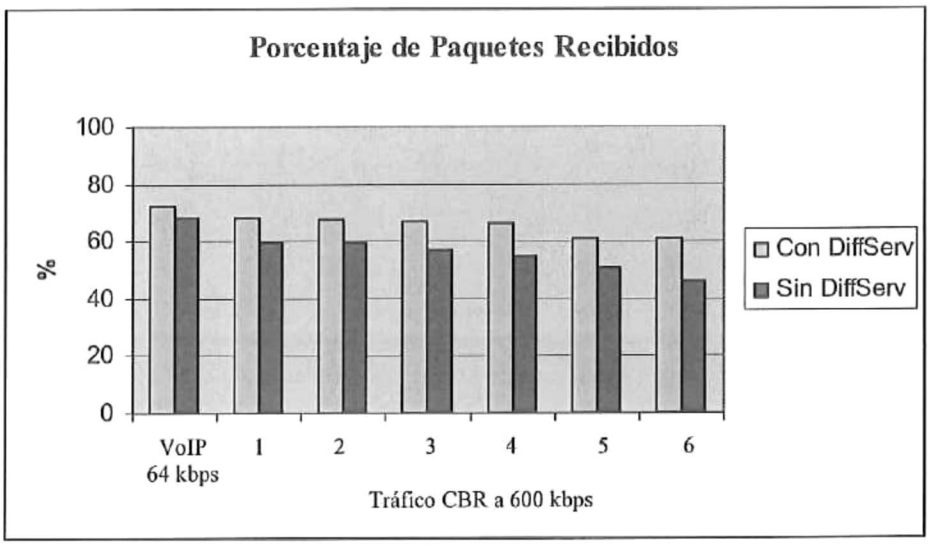


Figura 32.- Gráfica de Paquetes Recibidos, con un movimiento de nodos de 20 m/s.

En la Figura 32 se muestra la gráfica del porcentaje de paquetes que se recibieron, se puede observar que la recepción de los paquetes es menor que en las dos gráficas anteriores, esto es debido a que la velocidad promedio de movimiento de los nodos aumento a 20 m/s, más sin embargo en la red que se tiene implementado DiffServ se tiene un mejor desempeño al igual que en los casos anteriores, cuando la velocidad promedio de los nodos era menor.

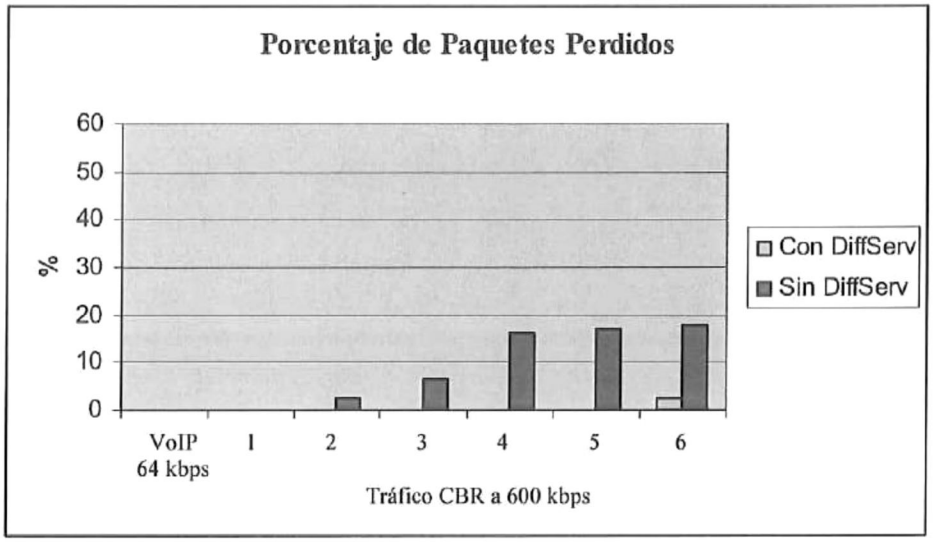


Figura 33.- Gráfica de Paquetes Perdidos, con un movimiento de nodos de 5 m/s.

En la Figura 33, muestra una gráfica del porcentaje de paquetes que se perdieron en la simulación, cuando los nodos tenían una velocidad promedio de 5 m/s, como se puede observar cuando únicamente se contaba con una sesión de tráfico de VoIP, se nota un comportamiento igual en el escenario cuando se tiene DiffServ y cuando no se tiene, sin embargo conforme se van aumentando una a una las sesiones de CBR a 300 kbps cada una se va produciendo congestionamiento en los nodos cluster head, lo que provoca que la red se degrade; cuando no se cuenta con DiffServ el número de paquetes perdidos es mayor. En el escenario que tiene implementado DiffServ debido a que le da prioridad de salida al tráfico de VoIP este no se degrada.

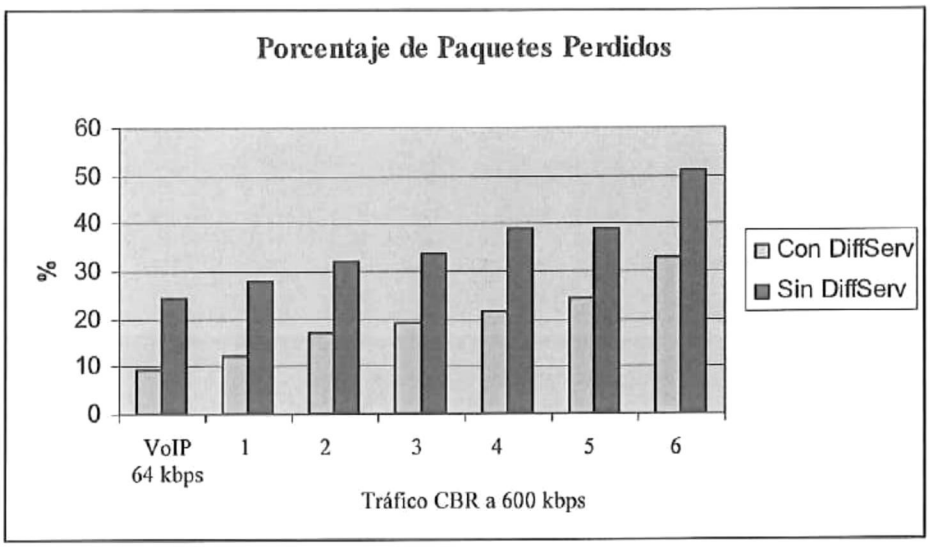


Figura 34.- Gráfica de Paquetes Perdidos, con un movimiento de nodos de 10 m/s.

En la Figura 34 se muestra el porcentaje de paquetes perdidos en una red con una velocidad promedio de los nodos de 10 m/s, se puede apreciar que existe un mayor porcentaje de paquetes perdidos que cuando se contaba con una velocidad promedio de 5 m/s, esto es dado que el movimiento de los nodos es más rápido en algún momento de la simulación van a existir destinos inalcanzables, sin embargo siempre se ve una pérdida de información mayor cuando no se aplica DiffServ.

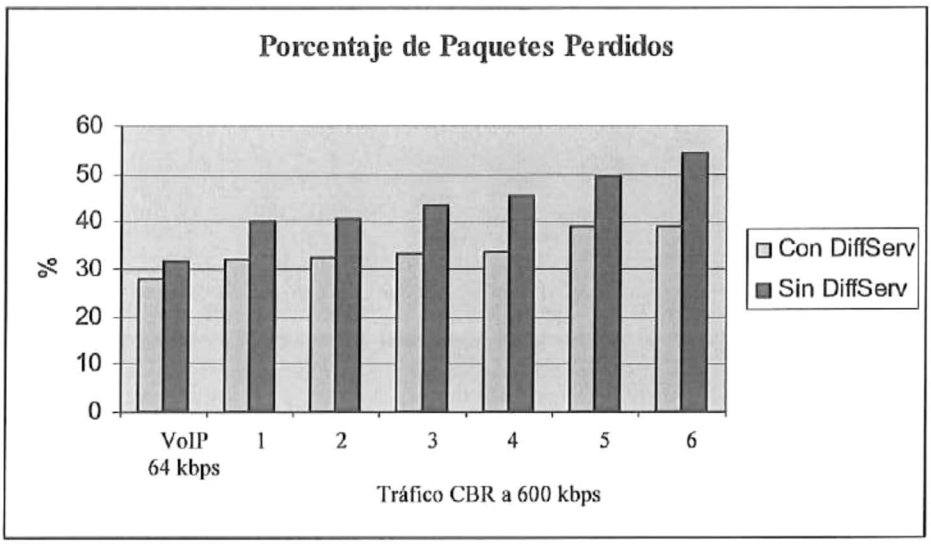


Figura 35.- Gráfica de Paquetes Perdidos, con un movimiento de nodos de 20 m/s.

En la Figura 35 es una gráfica que muestra el porcentaje de paquetes perdidos con una velocidad promedio de los nodos de 20 m/s, aquí se ve una degradación mucho mayor a la que existe en las dos gráficas anteriores la Figura 33 y la Figura 34, esto es debido a que en está gráfica la velocidad promedio de los nodos es mayor, debido a esto en algún momento de la simulación existirá un mayor número de destinos inalcanzables que cuando se contaba con una velocidad promedio de los nodos de 5 m/s o de 10 m/s, lo que provoca que el porcentaje de paquetes perdidos se incremente, sin embargo se tiene una situación similar a los resultados anteriores, ya que el número de paquetes perdidos se reduce cuando se tiene implementado DiffServ.

VII.6 Resumen

En este capítulo se analizó la implementación de un mecanismo de calidad de servicio, planteando la posibilidad de usar el mecanismo de DiffServ para proporcionar QoS a las redes móviles ad hoc de tipo jerárquico, utilizando el protocolo HOLSR para proporcionar enrutamiento. Se presentaron los resultados como throughput, perdida de paquetes y paquetes recibidos.

Se realizaron simulaciones en los cuales se establecían sesiones CBR entre los nodos, simulando una sesión CBR a 64kbps para simular tráfico de VoIP, y tráfico de fondo utilizando sesiones CBR a 600 kbps para otro tipo de tráfico. Se trabaja con dos tipos de escenarios los que contaban con políticas de QoS y los escenarios que no contaban con políticas de QoS. Para los escenarios con QoS se establecieron políticas de Token Buket para la sesión de VoIP, el tráfico de VoIP era el único tráfico que contaba con políticas de QoS, las otras sesiones CBR no contaban con QoS, únicamente era tráfico de tipo Mejor Esfuerzo; los otros escenarios de simulación no manejaban ningún tipo de políticas de QoS todo el tráfico era tratado como Mejor Esfuerzo. Las evaluaciones de desempeño se realizaron en base al Throughput, los paquetes perdidos y los paquetes recibidos.

La información obtenida en base a las simulaciones realizadas muestran que el hecho de aplicar mecanismos de calidad de servicio mejor el funcionamiento de las redes móviles ad hoc, ya que reduce la perdida de paquetes y por tanto los paquetes que ser reciben llega a ser mayor que si no se implementara el mecanismo de DiffServ, el caudal

eficaz también es mucho mejor cuando tiene aplicado el mecanismo de DiffServ; a pesar de los cambios en la red debido al movimiento de los nodos el desempeño de la red mejora de manera sustancial al tener implementado el mecanismo de DiffServ.

Capítulo VIII Conclusiones, aportaciones y trabajo futuro

VIII.1 Conclusiones

Actualmente existe un gran número de aplicaciones que se transportan sobre las redes de comunicaciones las cuales día a día demandan mayores prestaciones de recursos para su funcionamiento correcto. Por tal motivo es importante implementar mecanismos que puedan ofrecer una adecuada administración de los recursos con los que cuentan las redes.

En este trabajo de tesis se llevó a cabo la implementación del mecanismo de DiffServ el cual ayuda a mantener parámetros como son perdida de paquetes y Throughput en los rangos permitidos para el buen funcionamiento de las aplicaciones, se demostró que DiffServ ofrece mejor servicio que el mecanismo de Mejor Esfuerzo. Haciendo uso de DiffServ es necesario hacer una correcta diferenciación de servicio para poder brindar al tráfico el trato adecuado, de acuerdo a las características que necesite cada tipo de tráfico.

DiffServ es un mecanismo desarrollado para proporcionar calidad de servicio a las redes de comunicaciones, este mecanismo esta basado en hacer una diferenciación de servicio a los distintos tipos de tráfico y de esta forma proporcionar un trato diferente a cada tipo de tráfico, dando prioridad al tráfico en tiempo real o aplicaciones que necesiten

de mayores prestaciones de servicio como ancho de banda, menor retardo entre paquetes, entre otros, y dejando como menor prioridad a aquellos paquetes que necesiten menos prestaciones de servicio; de esta forma se garantiza que el tráfico de mayor importancia sea atendido conforme a sus necesidades. Este mecanismo fue propuesto en esta tesis para proporcionar calidad de servicio a las redes móviles ad hoc.

Las redes móviles ad hoc son un conjunto de nodos móviles que se comunican entre sí a través de enlaces inalámbricos, este tipo de redes no necesita de una infraestructura de red fija y su gestión es de forma descentralizada. En este tipo de redes todos los nodos funcionan como enrutadores y participan de forma directa en la toma de decisiones ya que no cuentan con una infraestructura central que administre sus funciones. Las redes MANETs cuentan con sus propios protocolos de enrutamiento especialmente diseñados para ellas que se adaptan a sus características principales como: topología dinámica, energía limitada, ancho de banda variable entre otras. Actualmente existen varias propuestas para proporcionar enrutamiento a las MANETs una de ellas son los protocolos proactivos, este tipo de protocolos siempre tienen actualizadas sus tablas de enrutamiento, otros son los protocolos reactivos estos protocolos solo tienen las rutas que les son solicitadas, no mantiene todas las rutas a todos los destinos como lo hacen los protocolos proactivos, entre los protocolos de enrutamiento proactivos se propone el protocolo HOLSR, que es un protocolo de enrutamiento proactivo que implementa una topología de tipo jerárquica. Para que una red tenga un buen desempeño, no es suficiente contar con un protocolo de enrutamiento adecuado, es necesario que además se tengan implementados mecanismos que les proporcionen calidad de servicio.

En este trabajo de tesis se utilizaron redes inalámbricas heterogéneas y se utilizó una administración de nodos de forma jerárquica utilizando cluster y nodos cluster heads para ayudar en la gestión, utilizando como protocolo de enrutamiento al protocolo HOLSR.

El protocolo de enrutamiento HOLSR es un protocolo de enrutamiento proactivo diseñado especialmente para proporcionar enrutamiento a las redes móviles ad hoc de tipo jerárquico con nodos heterogéneos, en el caso de este trabajo los nodos móviles heterogéneos son aquellos nodos caracterizados por contar con múltiples interfaces de red. Este tipo de protocolo es una alternativa para aquellas redes que necesiten contar con un enrutamiento jerárquico de tipo proactivo.

El objetivo principal de esta tesis es proporcionar calidad de servicio a las redes móviles ad hoc de tipo jerárquico utilizando el protocolo de enrutamiento proactivo HOLSR, para proporcionar calidad de servicio se utilizó el mecanismo de DiffServ. En las evaluaciones que se realizaron se utilizaron escenarios que tenían implementado el mecanismo de DiffServ y de escenarios que no tenían implementado el mecanismo de DiffServ, en ambos escenarios se utilizó el protocolo de enrutamiento HOLSR.

Para poder implementar el mecanismo de calidad de servicio se llevaron a cabo una serie de modificaciones de los módulos de DiffServ ya que en una red tipo ad-hoc todos los nodos funcionarán como enrutadores extremos y no habrá enrutadores intermedios como se describe en la arquitectura de DiffServ en el Capítulo V, y las políticas serán conocidas por todos los nodos que conforman la red Ad-Hoc.

En las simulaciones realizadas en la red móvil Ad-Hoc cuando se le implementó el mecanismo de DiffServ se pudo apreciar que los resultados de desempeño en la red son mucho mejores comparados con las simulaciones de las redes móviles Ad-Hoc carentes del mecanismo DiffServ. Cuando una red es saturada se producen congestionamientos de las colas y estas se desbordan provocando que exista perdida de paquetes, esto ocurre en mayor medida si se esta utilizando el mecanismo de Mejor Esfuerzo a diferencia de DiffServ que mantiene una administración adecuada de los recursos para tratar de evitar el congestionamiento de la red. Se evaluaron diferentes escenarios en los cuales se variaba la velocidad de los nodos, las variaciones de velocidad de los nodos se implementaron para ambos casos, y se pudo observar que cuando se aplicó DiffServ el desempeño de la red siempre fue mejor a pesar de las variaciones de velocidad que se aplicaron en los diferentes escenarios.

VIII.2 Aportaciones

Debido a que no existía un protocolo de enrutamiento proactivo de tipo jerárquico para las redes móviles Ad-Hoc en NS-2 se tuvieron que hacer una modificaciones a la extensión del NS-2 denominada TENS; esta extensión tenía la funcionalidad de que sus nodos tuvieran la capacidad de trabajar con múltiples interfaces, el TENS al igual que el NS-2 no tiene integrado el agente de enrutamiento OLSR es por eso que al TENS se le agregó la extensión del agente de enrutamiento OLSR, se modificó el protocolo de

enrutamiento proactivo OLSR para que sus nodos trabajaran con interfaces múltiples, además de hacer que el protocolo tuviera una administración de tipo jerárquica para NS-2.

Se implementó el mecanismo de DiffServ a una red MANET y se obtuvieron los resultados al utilizar este mecanismo en redes de tipo jerárquicas, además de obtener una comparación entre escenarios donde este mecanismo no es utilizado y en escenarios donde si se utiliza; los parámetros que se utilizaron para medir el desempeño de la red fueron: Throughput, paquetes recibidos y paquetes perdidos.

VIII.3 Trabajo futuro

Se propone como trabajo futuro evaluar otro tipo de parámetros a fin de obtener resultados con mayor detalle acerca de la aplicación calidad de servicio en las redes Ad-Hoc utilizando el mecanismo de DiffServ, los parámetros que se proponen a evaluar son el retardo y el *jitter*. Es importante evaluar el parámetro del retardo en las redes de comunicaciones cuando se aplican mecanismos de calidad de servicio ya que sin duda alguna el retardo juega un papel importante en muchas aplicaciones principalmente en aplicaciones multimedia o en tiempo real, es por eso que se considera importante para evaluar en un trabajo futuro estos dos parámetros.

Además de lo anterior también se propone implementar otra clase de aplicaciones multimedia en las redes móviles Ad-Hoc, aplicaciones tales como video conferencias,

video bajo demanda, etc., todo esto con el fin de evaluar a mayor detalle el mecanismo de DiffServ implementado en las redes móviles Ad-Hoc de tipo jerárquico.

Por otro lado, también se propone definir una ruta alterna de enrutamiento sobre el protocolo HOLSR que brinde el mejor camino basado en la sobre carga de información que contengan los enlaces.

Referencias

Ahn, Gahng-Seop.; Cambell, Andrew T.; Veres, Adras.; Sun, Li-Hsiang. 2002. **"Supporting Service Differentiation for Real-Time and Best-Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN) "**. IEEE Transactions On Mobile Computing, Vol. 1, No. 3.

Blake, S.; Black, D.; Carlson, M.; Davies, E.; Wang, z.; Weiss, W. 1998. **"An Architecture for Differentiated Services"**. Request for Comments: 2475. Disponible en http://www.ietf.org/rfc/rfc2475.txt.

Braden, R. Ed.; Zhang, L.; Berson, S.; Herzog, S.; Jamin, S. 1997. **"Resource ReSerVation Protocol"**. Request for Comments 2205. Disponible en http://www.ietf.org/rfc/rfc2205.txt.

Braden, R.; Clark, D.; Shenker, S. 1994. **"Integrated Services in the Internet Architecture".** Request for Comments: 1633. Disponible en http://www.ietf.org/rfc/rfc1633.txt.

Caballero Cárdenas, X. 2002. **"Análisis y Modelado de Mecanismos de Servicios Diferenciados (DiffServ) para implementación de redes con Calidad de Servicio (QoS)".** Tesis de Maestría. CICESE, Ensenada, Baja California, México.

Clausen, T y Jacquet, P. 2003. **Optimized Link State Routing Protocol**. Request for Comments: 3626. Disponible en http://www.ietf.org/rfc/rfc3626.txt.

Corson S. y Macker J., 1999. **"Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations"**, Request for Comments: 2501. Disponible en http://www.ietf.org/rfc/rfc2501.txt, January 1999.

Guimañaes, Rafael.; Morillo , Julián, ; Cerdá, Llorenc.; Barceló, José-M.; and García, Jorge. 2004. **"Quality of Service for Mobile Ad-Hoc Networks: an Overview"**. Technical University of Catalonia.

Haartsen, J. 1998. **"The Bluetooth radio system".** IEEE Personal Communiccations, 7(1):28-36 p.

Heinanen, J; Finland, T; Baker, F; Weiss, W, y Wroclawski, J. 1999. **Assured Forwarding PHB Group.** Request for Comments:2597. Disponible en http://www.ietf.org/rfc/rfc2597.

IEEE 1996. **"IEEE Standards for Informatict Technology –Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Networks –Specific Requirements"**- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

IETF, 2006, **"Mobile Ad-Hoc Networks (manet) working group"**, Disponible en http://www.ietf.org/html.charters/manet-charter.html.

ISI, 2006a. **The *ns* Manual (formerly *ns* Notes and Documentation).** Disponible en http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf.

ISI, 2006b. **The Enhanced Network Simulator.** Disponible en http://www.cse.iitk.ac.in/users/braman/tens/.

Johnson, David B.; Maltz, David A. y Hu, Yih-Chun. 2004. **The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks.** INTERNET-DRAFT. Disponible en http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt.

Macker Joseph P. y Corson M. Scott. **Mobile Ad Hoc Networking and the IETF, Information Technology Division**, Naval Research Laboratory, Washington, DC, USA. Institute for Systems Research, University of Maryland, College Park, MD, USA. 1998.

Magnus Frodigh, Per Johansson y Peter Larsson. 2000. **"Formación de redes inalámbricas ad hoc- El arte de la formación de redes sin red"**. Ericsson Review No.4, 2000.

Mohapatra, Prasant.; Li, Jian . y Gui, Chao. 2003. **"QoS in mobile ad hoc networks".** Department of Computer Science. University of California.

Moy, John. 1994, **Open Shortest Path First**. Request for Comments: 1583. Disponible en http://www.ietf.org/rfc/rfc1583.txt.

Nichols, K; Blake, S; Blaker, F; Black, D. 1998. **"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers".** Request for Comments: 2474. Disponible en http://www.ietf.org/rfc/rfc2474.txt.

Nobel, C. 2000. **"Making 8002.11 standards work together".** eWeek, Julio 19, 2000.

Ogier. R, Templin. F, Lewis. M. 2004. **Topology Dissemination Based on Reverse Path Forwarding.** Request for Comments: 3684. Disponible en http://www.ietf.org/rfc/rfc3684.txt.

Oyoqui Sanchez, J. 2003. **"Transferencia del contexto en redes locales inalámbricas".** Tesis de Maestría. CICESE, Ensenada Baja California, México.

Park, V. y Corson, S. 1998. ***Temporally-Ordered Routing Algorithms* (TORA)** Version 1 Functional Specification. IETF Internet Draft, draft-ietf-manet-tora-spec-01.txt.

Pei, Guangyu; Gerla, Mario; y Chen, Tsu-We. 2000. **Fisheye State Routing in Mobile Ad Hoc Networks.** CDCS Workshop on Wireless Networks and Mobile Computing.

Perking, Charles E. y Bhagwa Pravin. **"High Destination Sequenced Distancce Vector (DSDV) for Mobile Computers"**, In Proceedings of the SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, Agust 1994.

Perkins, C. y Belding-Royer, E. 2003. **Ad Hoc On Demand Distance Vector Routing.** Request for Comments: 3561. Disponible en http://www.ietf.org/rfc/rfc3561.txt.

Pieda, Peter, Ethridge, J, Baines, M. and Shallwani, F. 2000. **A Network Simulator Differentiated Services Implementation.** Open IP, Nortel Networks.

Qayyum, A ; Viennot, L y Laouiti, A. 2002. **"Multipoint Relaying for Flooding Broadcast Messages in MobileWireless Networks"**, Proceedings of the 35th Hawaii International Conference on System Sciences. Hawaii, USA.

Seoung-Bum, L; Gahng-Seop, A; Xiaowei, Z; y Andrew, T. 1999. **"INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks"**. Journal of Parallel and Distributed Computing 60, 374_406 (2000).

Sivakumar, R; Sinha, P; Bharghavan, V. 1999. **"CEDAR: A Core-Extraction Distributed Ad Hoc Routing Algorithm"**. IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 17, NO. 8, AUGUST 1999.

Villasenor-Gonzalez, L; Ge, Y; Lamont, L. 2005. **"HOLSR: A Hierarchical Proactive Routing Mechanism for Mobile Ad Hoc Networks"** IEEE Communications Magazine, July 2005.

Wu, K. and Harms, J. (2001). **"QoS Support in Mobile Ad Hoc Networks."** Crossing Boundaries – an interdisciplinary journal 1(1).

Yang, L.; Conner, S.; Guo, X. ; Hazra, M. y Zhu, J. 2003. **Common Wireless Ad Hoc Networks Usage Scenarios.** Internet Draft \draft-irtf-yang-ans-scenarios-00". Work in progress.

Apéndice A Terminología de DiffServ

Behavior Aggregate (BA, también llamado a veces "agregado de tráfico", TA): es una colección de paquetes con el mismo DSCP (por las siglas en inglés - *DiffServ Code Point*) atravesando un enlace en una dirección.

BA classifier: es un clasificador que selecciona paquetes basado solo en el contenido del campo de DS.

Boundary link (Enlace de frontera): es un enlace que conecta los nodos de borde de dos dominios.

Classifier (Clasificador): una entidad que selecciona los paquetes basado en el contenido de la cabecera del paquete de acuerdo a las reglas definidas.

DS behavior agregate: una colección de paquetes con el mismo código DS, cruzando un enlace en una dirección particular.

DS boundary node: un nodo DS que conecta un dominio DS con cualquier nodo en otro dominio DS o en un dominio que no acepta DS.

DS-capable: capaz de implementar servicios diferenciados según lo descrito en esta arquitectura; utilizado generalmente en referencia a un dominio que consiste en nodos DS-compliant.

Código DS: un valor específico de DSCP del campo DS, usado para seleccionar un PHB.

DS-compliant: capaz de soportar funciones y comportamientos de servicios diferenciados.

DS domain (Dominio DS): es un dominio capaz de tener DS; un conjunto contiguo de nodos que operan con un conjunto común de políticas de aprovisionamiento de servicios y definiciones PHB.

DS egress node (Nodo de egreso DS): un nodo DS de frontera que tiene como función manejar tráfico a medida que éste deja el dominio DS.

DS ingress node (Nodo de ingreso DS): un nodo DS de frontera que tiene como función manejar tráfico a medida que éste entra al domino DS.

DS interior node (Nodo interior DS): un nodo DS que no es un nodo DS extremo o de frontera.

DS field (Campo DS): es el octeto TOS de la cabecera de IPv4 o el octeto de la clase de tráfico de IPv6. Los bits del campo DSCP contienen el DS codepoint, mientras que los bits restantes no están en actualmente en uso.

DS node (Nodo DS): Nodo que está en el dominio de DiffServ.

DS region (Región DS):Un sistema de dominios DS contiguos que pueden ofrecer servicios diferenciados sobre las trayectorias a través de esos dominios DS.

Dropper: Un dispositivo que realiza el descarte.

Dropping: es el proceso de descartar paquetes basándose en reglas específicas; políticas.

Legacy node : Un nodo que implementa IPv-4.

Marker: un dispositivo que realiza el marcado.

Marking (marcado): es el proceso de configuración del DS codepoint en un paquete, basándose en reglas definidas; pre-marcado y re-marcado.

Mechanism (Mecanismo): algoritmo específico de operación, que es implementado en un nodo para realizar un conjunto de uno o más PHB.

Metering (mediciones): es el proceso de medir las propiedades temporales de una corriente de tráfico seleccionada por un clasificador (classifier).

Microflow (microflujo): es un conjunto de datos, enviados unidireccionalmente entre dos aplicaciones, únicamente identificado por una quíntupla: protocolo de transporte, IP origen, IP destino, puerto origen y puerto destino.

MF Classifier: a Multi-Fiel Clasificador es seleccionado en base a el contenido de los paquetes de algunos números arbitrarios de los campos de la cabecera; típicamente algunas combinaciones de la dirección fuente, dirección destino, campo DS, protocolo ID, puerto fuente y puerto destino

Per-Hop-Behavior (PHB): define el tratamiento en cada nodo. Es una descripción del comportamiento de reenvío observado exteriormente; puede ser implementado por distintos mecanismos.

PHB group: es un conjunto de uno o más PHBs que puede tener un significado específico e implementarse de manera simultanea.

Policing: el proceso de descarte de paquetes dentro de un flujo de tráfico en concordancia con el estado de un correspondiente medidor (meter) cumpliendo un determinado perfil.

Acuerdo del Nivel de Servicio (SLA): un contrato de servicio entre un cliente y un proveedor de servicio que especifica el servicio de envío que un cliente debe recibir.

Shaper : dispositivo que realiza el retardo de los paquetes.

Shaping (conformador): el proceso de retardar paquetes dentro de un flujo de tráfico, haciendo que conforme cierto perfil de tráfico ya definido.

Traffic Conditioner (acondicionador de tráfico): una entidad que realiza las funciones de condicionamiento del tráfico y que puede contener medidores, marcadores, descartadores (droppers) y conformadores. Están típicamente dispuestos en nodos de borde solamente.

Traffic Conditioning Agreement (TCA): un acuerdo especificando reglas de clasificación y perfiles de tráfico correspondientes, y mediciones, marcado, descarte y/o reglas de conformación que son aplicables a los arroyos de tráfico seleccionados por el clasificador.