TESIS DEFENDIDA POR

Roberto Adolfo Romero Hernández

Y aprobada por el siguiente comité:

Dr. José Antonio García Macías

Director del Comité

Dr. José Alberto Fernández Zepeda

Miembro del Comité

Dr. Roberto Conte Galván

Miembro del Comité

Dr. Oscar Iván Lepe Aldama

Miembro del Comité

Dr. Pedro Gilberto López Mariscal

Coordinador del programa en Ciencias de la Computación

Dr. Raúl Ramón Castro Escamilla

Director de Estudios de Posgrado

10 de Enero del 2006

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y DE EDUCACIÓN SUPERIOR DE ENSENADA

PROGRAMA DE POSGRADO

EN CIENCIAS DE LA COMPUTACIÓN

Descubrimiento de Servicios en Capa de Red para Redes Ad Hoc

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de

MAESTRO EN CIENCIAS

Presenta:

Roberto Adolfo Romero Hernández

Ensenada, Baja California a Enero del 2006.

RESUMEN de la tesis de **Roberto Adolfo Romero Hernández**, presentada como requisito parcial para obtener el grado de MAESTRO EN CIENCIAS en CIENCIAS DE LA COMPUTACIÓN. Ensenada, Baja California. Enero del 2006.

Descubrimiento de Servicios en Capa de Red para Redes Ad Hoc

Resumen aprobado por:

Dr. José Antonio García Macías

Director de Tesis

Actualmente con la proliferación de dispositivos móviles con capacidades de cómputo y comunicación inalámbrica, es posible formar redes Ad hoc dentro de diferentes entornos, así como compartir la información y los recursos con los cuenta cada dispositivo. A pesar del gran desarrollo en las redes ad hoc, aun existen grandes retos que deben ser abordados para que este tipo de redes funcionen adecuadamente. El descubrimiento de servicios en redes ad hoc, es una de las áreas que requieren especial atención, ya que la mayoría de las soluciones/propuestas de descubrimiento de servicios no han sido diseñadas especialmente para este tipo de entornos inalámbricos y móviles.

En este trabajo de tesis se propone una solución para el descubrimiento de servicios en redes ad hoc móviles (OLSRSD), la cual combina los procesos de descubrimiento de rutas y descubrimiento de servicios con la finalidad de minimizar el número de mensajes por búsqueda. Dicha propuesta está basada en el protocolo proactivo OLSR. Además, en este trabajo se realiza un estudio comparativo entre la solución AODVSD definida en Arias (2004), que realiza el descubrimiento de servicios de igual forma, pero utilizando el protocolo de enrutamiento AODV. Por otra parte, se realiza un estudio comparativo que busca determinar si el combinar el descubrimiento de rutas con el descubrimiento de servicios es una mejor opción que los protocolos de descubrimiento de servicios tradicionales, los cuales realizan ambas actividades de forma independiente.

Por medio de los estudios comparativos, se demuestra que el combinar el descubrimiento de rutas con el de servicio no es tan solo una opción viable para el descubrimiento de servicios, si no que minimiza el costo generado por generación de mensajes de búsqueda en comparación con protocolos tradicionales para el descubrimiento de servicios.

Palabras clave: Descubrimiento de Servicios, Capa de Red, Redes Ad Hoc.

ABSTRACT of the thesis presented by **Roberto Adolfo Romero Hernández**, as a partial requirement to obtain the MASTER OF SCIENCE degree in COMPUTER SCIENCES. Ensenada, Baja California. January 2006.

Service Discovery in Network Layer for Ad Hoc Networks

Abstract approved by:

Dr. José Antonio García Macías

Thesis director

Currently, with the proliferation of mobile devices with computing and wireless communication capabilities, it is possible to form Ad hoc networks within different enviroments, sharing the information and the resources of each device. In spite of all the recent developments in ad hoc networks, great challenges still exist that must be tackled in order for these networks to work suitably. Service discovery in ad hoc networks is one of the areas that requires special attention, since most of the solutions for service discovery have not been designed specifically for this type of wireless and mobile enviroments.

In this thesis work a solution for service discovery in mobile ad hoc networks is provided (OLSRSD); this solution combines the processes of route discovery and service discovery with the purpose of reducing the number of search messages.The OLSRSD proposal is based on the proactive OLSR protocol. In addition, in this work a comparative study with AODVSD (defined in Arias (2004)) is provided; AODV performs service discovery in a similar way, but using the reactive routing protocol AODV. On the other hand, a comparative study is made that aims at determining if combining route discovery with service discovery is a better option than traditional protocols for service discovery, which perform both activities independently.

The comparative studies demonstrate that combining route discovery with service discovery is a good option, since the cost generated by search messages is reduced when compared with traditional protocols for service discovery.

Keywords: Service Discovery, Network Layer,Ad Hoc Networks.

A mis Padres: Roberto Nelson y Ana Maria, agradeciendo con todo mi amor su ejemplo, esfuerzo, apoyo y confianza incondicional.

Agradecimientos

Al Dr. Antonio García Macías, por el apoyo y orientación desde inicios del posgrado y durante la realización de este trabajo de investigación.

Al Dr. Roberto Conte Galván, por todos sus consejos y por su interés por contribuir en mi desarrollo profesional, muchísimas gracias.

Al Dr. Alberto Fernández Zepeda, por estar al pendiente de mi integración como estudiante de tiempo completo cuando inicié el posgrado, así como por permitirme ver las cosas de manera más sencilla. Muchas gracias por su apoyo.

Al Dr. Oscar Lepe Aldama, por todo el apoyo que me brindó desinteresadamente para el desarrollo de este trabajo.

Al Dr. Luis Villaseñor, por orientarme desinteresadamente como resolver algunos problemas enfrentados.

A mis Abuelos, Antonio Adolfo y Rosalina por creer en mi y estar siempre al pendiente de mis estudios.

A mi Tía Ángela, por estar siempre al pendiente de mi bienestar.

A Adrián Olguín, por ayudarme a sobrellevar las desveladas de estudio, así como contribuir con ideas y darme ánimo para concluir este trabajo.

A Ana González, por ayudarme en sacar adelante la instalación del protocolo OLSR, así como por el intercambio de ideas.

A Milton Rodríguez y Adrián Burciaga, por ayudarme con los problemas enfrentados con el formato del documento de tesis.

A mis amigos Adrián Olguín, Brenda, Ana González, Pablo Soto, Oscar Morita, Ricardo Cook, Edgar Topete por su amistad y por estar al pendiente de la finalización de este trabajo.

Ensenada, México
10 de Enero del 2006.

Roberto Adolfo Romero Hernández

Tabla de Contenido

Tabla de Contenido (Continuación)

Tabla de Contenido (Continuación)

Capítulo Página

Lista de Figuras

Lista de Figuras (Continuación)

Figura Página

Lista de Tablas

Tabla Página

Capítulo I

Introducción

Las ciencias de la computación y la tecnología de telecomunicaciones han evolucionado a través del tiempo, en el cual han unido esfuerzos para revolucionar los métodos de comunicación utilizados por individuos, instituciones y empresas. Los continuos avances en las redes de cómputo han permitido pasar de una arquitectura cableada a una inalámbrica y móvil. Además, los aparatos y dispositivos electrónicos se han hecho cada vez más pequeños hasta llegar a ser portátiles. Algunos de estos dispositivos se llaman PDA (*Personal Digital Assistant*), los cuales cuentan con diferentes funciones, tales como telefonía celular, agendas, editores, organizadores personales, etc. Las características de estos dispositivos han permitido integrarlos a las redes de cómputo mediante enlaces inalámbricos, habilitando a los usuarios combinar el poder de cómputo, la telefonía y las redes. Dicha integración ha hecho posible tener acceso a información y a recursos que ofrecen servicios (impresoras, fax, proyectores, pizarrones electrónicos, etc.), así como a mantenerse en contacto con amigos y compañeros de trabajo (Navarro, 2002) .

Ante la convergencia de estas tecnologías surgieron las redes ad hoc, este nuevo paradigma de las redes de cómputo consiste de un conjunto de nodos móviles que se pueden desplazar dentro de la red libremente, por lo que la red puede tomar múltiples formas. En este tipo de redes cada uno de los nodos cuenta con un transmisor y un receptor para establecer comunicación. Además, cada uno de los nodos participa en el proceso de búsqueda y mantenimiento de rutas hacia otros nodos. Existen diferentes

escenarios en los cuales las redes ad hoc pueden ser utilizadas, algunos ejemplos son en entornos militares, operaciones de emergencia y rescate, misiones de exploración, conferencias, entre otros (Vidal, 2003);(Jun-Zhao, 2001).

I.1. Motivación y Justificación

A pesar del gran desarrollo que han tenido las redes ad hoc, aun existen grandes retos que se deben superar para que este tipo de redes funcionen adecuadamente. Entre ellos se encuentra la calidad de servicios, seguridad, enrutamiento y el descubrimiento de servicios. Aunque la mayoría del trabajo de investigación está centrado en el área de enrutamiento, las otras áreas no dejan de ser menos importantes (Vidal, 2003). En el presente trabajo de tesis se desea contribuir en el descubrimiento de servicios, que es el nombre con el cual se designa a la actividad que permite encontrar y hacer uso de los servicios disponibles en una red de cómputo (Chen, 2002).

Para ilustrar la utilizad del descubrimiento de servicios en las redes ad hoc considere el siguiente ejemplo: En una sala de conferencias varios asistentes y expositores cuentan con dispositivos móviles. Algunos de ellos brindan servicios tales como impresión y visualización de documentos, acceso a internet, almacenamiento de información, entre otros. En el momento que alguien quiera usar algún servicio, requiere emplear mecanismos/protocolos para poder localizar los servicios sin intervención de usuario, por lo que el solicitante no necesita saber la dirección del dicho servicio. En este tipo de situaciones es donde se requiere un mecanismo eficiente para el descubrimiento de servicios.

Actualmente existen distintos protocolos para el descubrimiento de servicios, pero en general, enfocados a redes con infraestructura estable y poco adaptados a redes

ad hoc, por tal razón el descubrimiento de servicios en redes ad hoc sigue siendo un problema de investigación abierto (Baker, 2002).

En su trabajo reciente, Arias (2004) propone una solución para el descubrimiento de servicios en redes ad hoc móviles, basada en la combinación de los procesos de descubrimiento de servicios y enrutamiento que permite optimizar el uso de los recursos de cómputo. Asimismo, basándose en la solución propuesta presenta una arquitectura para el descubrimiento de servicios que permite a los desarrolladores de aplicaciones, incorporar en éstas el descubrimiento de servicios por medio de una interfaz de programación de aplicaciones (*API*). Al analizar el trabajo antes mencionado, uno se puede percatar que se abre una ventana de oportunidades de investigación.

I.2. Planteamiento del problema

Las redes ad hoc son de gran importancia en situaciones donde no se cuenta con infraestructura de red predefinida o bien se requiere conectividad temporal. Sin embargo, las redes ad hoc se ven afectadas por la movilidad constante e impredecible de los nodos, las limitaciones de ancho de banda de los enlaces inalámbricos y por las características de los nodos en la red. Algunos de los problemas que se enfrentan son: limitaciones de energía, limitaciones de poder de cómputo, limitaciones de memoria, riesgos en seguridad por hacer uso de enlaces inalámbricos no seguros, atenuaciones en la señal de transmisión, interferencias, entre otras (Jun-Zhao, 2001).

Es necesario contar con mecanismos/protocolos específicamente diseñados para dar soporte a las redes ad hoc, los cuales permitan reducir las limitaciones impuestas por las características intrínsecas de éstas. Diferentes grupos de investigación se han enfocado a la creación de mecanismos de provisión de calidad de servicios, seguridad, conservación

de energía, enrutamiento, descubrimiento de servicios, etc; tomando en cuenta algunas de las limitaciones mencionadas anteriormente (Basagni, 2004);(Rendon, 2002).

Las redes ad hoc hacen uso de una arquitectura basada en 7 capas (capa física, capa de enlace, capa de red, capa de transporte, capa de sesión, capa de presentación y capa de aplicación) para definir la interconexión de dispositivos, donde cada una de las capas tiene asociado funciones bien especificas (Basagni, 2004). La figura 1 muestra las capas de dicho modelo y los retos de investigación de cada una.

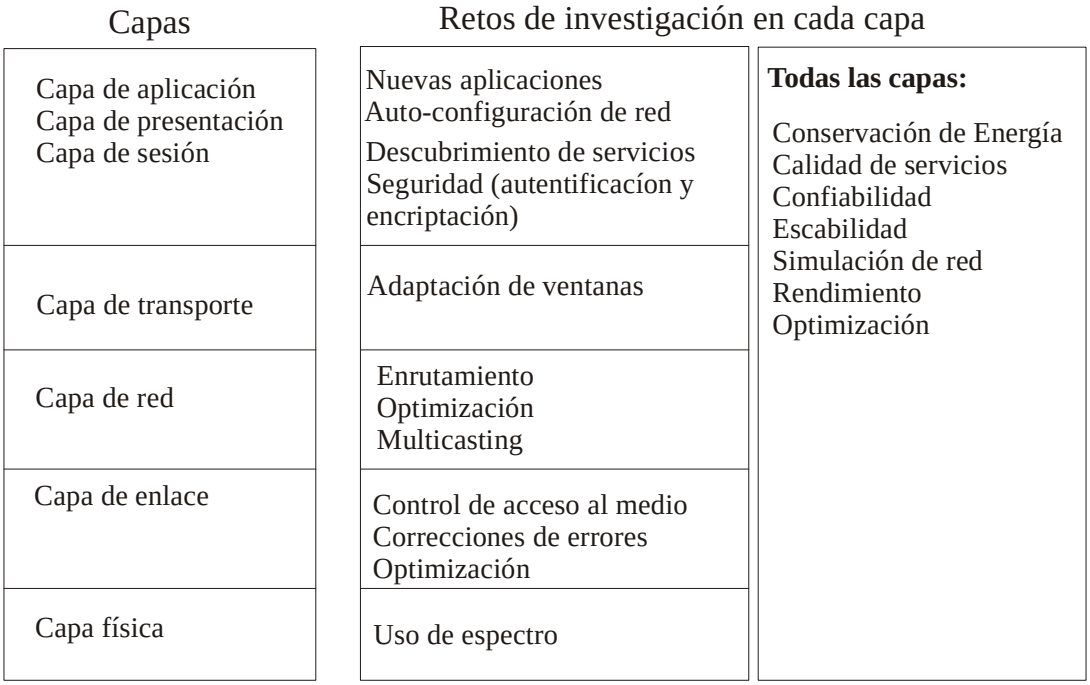

Figura 1: Capas del modelo de interconexión para redes ad hoc y retos de investigación

Hay que recordar que actualmente existen protocolos que dan soporte al descubrimiento de servicios, pero éstos se han diseñado en su mayoría enfocados a dar soporte a redes con infraestructura, y no a las redes ad hoc. Estos protocolos han sido diseñados siguiendo el enfoque tradicional para el descubrimiento de servicios basado en

la realización de esta actividad en capa de aplicación.

Por otra parte, existen protocolos que dan soporte al enrutamiento en redes ad hoc; estas soluciones se encuentran clasificadas en protocolos reactivos, proactivos e híbridos. Los protocolos reactivos determinan rutas sólo cuando se requiere, los protocolos proactivos mantienen las rutas entre par de nodos todo el tiempo, mientras que los híbridos es una combinación de ambos. Cabe mencionar que este tipo de protocolos pertenecen a capa de enrutamiento.

El descubrimiento de servicios y el enrutamiento son dos procesos independientes que requiere uno del otro para que un cliente localice y haga uso de los servicios en la red. Como se mencionó, es necesario hacer un mejor uso de los recursos de la red, de manera que se ahorre en energía, memoria y poder de cómputo. Con la propuesta mencionada en Arias (2004) se ataca el problema de descubrimiento de servicios en redes ad hoc móviles, ésta tiene como objetivo estudiar los efectos de combinar el descubrimiento de rutas y de servicios dentro de la capa de red. Dicha solución (AODVSD, Ad hoc On-Demand Distance Vector with Service Discovery) está basada en el protocolo reactivo AODV (Ad hoc On-Demand Vector) y ha generado resultados alentadores, superando el desempeño de las soluciones existentes, ya que logra obtener más rápido los servicios y disminuir el número de mensajes involucrados en el descubrimiento de servicios. Sin embargo en dicha propuesta no se investiga el desempeño obtenido utilizando un protocolo proactivo en lugar de uno reactivo. De obtener resultados similares, vendría a garantizar que realizar las actividades de descubrimiento rutas y de servicios de forma conjunta dentro de la capa de red es mejor.

I.3. Objetivos

Los objetivos de esta tesis son:

I.3.1. Objetivo general

El objetivo general de este trabajo de investigación es continuar con el estudio definido en Arias (2004), y proponer una arquitectura para el descubrimiento de servicios a nivel de capa de red para redes ad hoc utilizando un protocolo proactivo.

I.3.2. Objetivos Particulares

- Estudiar las características y limitaciones de las redes ad hoc.
- Analizar de los mecanismos existentes para el descubrimiento de servicios.
- Analizar el comportamiento del protocolo reactivo AODVSD (Ad hoc On-Demand Distance Vector with Service Discovery).
- Comparar el rendimiento de la red en el descubrimiento de servicios utilizando un protocolo proactivo (OLSR) en vez de uno reactivo (AODV).
- Analizar las ventajas y desventajas de combinar el descubrimiento de rutas y de servicios en las redes ad hoc.

I.4. Organización del Documento

El presente trabajo de tesis se encuentra organizado de la siguiente manera:

- **Capítulo 1:** Se presenta una introducción a la temática general de este trabajo, así como cada uno de los objetivos que se pretenden conseguir.

- **Capítulo 2:** Se presenta el marco teórico del nuevo paradigma de las redes de cómputo, las redes ad hoc. Se especifican conceptos, características, aplicaciones, problemáticas enfrentadas y áreas abiertas a investigación en las redes ad hoc.

- **Capítulo 3:** Se presenta el marco teórico sobre el descubrimiento de servicios, por lo que se especifican definiciones, criterios de diseño y se analizan las propuestas de descubrimiento de servicios de mayor relevancia en los últimos años, tanto en redes con y sin infraestructura estable.

- **Capítulo 4:** Se describe el funcionamiento detallado del protocolo de enrutamiento para redes ad hoc OLSR, además, se presenta una propuesta de descubrimiento de servicios que extiende la funcionalidad de dicho protocolo para que incluya el descubrimiento de servicios.

- **Capítulo 5:** Se estudia el rendimiento de la propuesta para el descubrimiento de servicios definida en el capítulo anterior. Se presenta un estudio comparativo entre la nueva propuesta y otras soluciones para descubrimiento de servicios. Se muestran experimentos realizados, resultados obtenidos y discusión de los mismos.

- **Capítulo 6:** Se presentan conclusiones, contribuciones y recomendaciones a futuro de este trabajo de tesis.

- **Apéndice A:** Se presenta información del simulador de redes NS2, el cual se utilizó para realizar cada uno de los experimentos del estudio de simulación descritos en el capítulo anterior. Se describe detalladamente cómo realizar la instalación de dicho simulador, así como de cada una de las soluciones de descubrimiento sujetas a estudio en este trabajo de tesis.

- **Apéndice B:** Se brinda un glosario y diccionario de términos utilizados en este trabajo de investigación.

Capítulo II

Redes Ad hoc Móviles

II.1. Introducción

En los últimos años, las redes de cómputo y las tecnologías de las telecomunicaciones han evolucionado significativamente, lo que ha propiciando el surgimiento de una gran variedad de dispositivos móviles, cada uno equipado con tecnología inalámbrica. Además, dichos avances han generado una nueva forma de comunicación, conocida como cómputo ubicuo. Básicamente, el cómputo ubicuo consiste en integrar capacidades de cómputo y de comunicaciones dentro de entornos cotidianos y poder aprovechar la información y recursos ofrecidos por los dispositivos distribuidos en el entorno. Cabe mencionar que mediante el uso de redes ad hoc es posible la creación de entornos de cómputo ubicuo, por lo que este capítulo está dedicado a las redes Ad hoc.

II.2. Redes Ad hoc móviles

Las redes ad hoc, también conocidas como MANET (*Mobile Ad Hoc Networks*), son un nuevo paradigma de las redes de cómputo. Estas redes consisten de una colección de dispositivos móviles y autónomos, donde los dispositivos se comunican entre sí inalámbricamente con algún otro nodo que se encuentra dentro del rango de transmisión. Estos dispositivos se pueden comunicar con otros dispositivos fuera del rango de transmisión mediante el uso de dispositivos intermedios (Toh, 2002). La figura 2

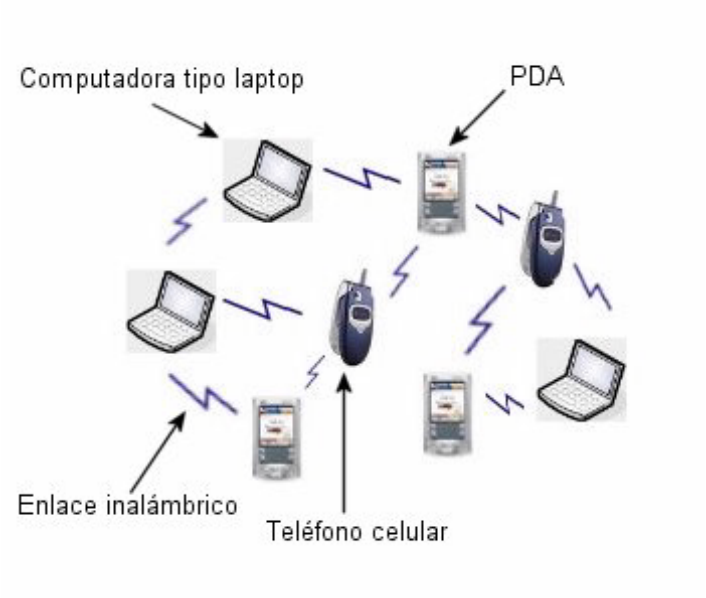

Figura 2: Red Ad hoc

muestra una red ad hoc heterogénea que consiste de tres tipos de dispositivos móviles (tres computadoras tipo laptop, dos teléfonos celulares y 3 PDA)

A diferencia de las redes de cómputo tradicionales (cableadas), las redes ad hoc no cuentan con una infraestructura de red fija, ya que los nodos o dispositivos pueden entrar y salir de la red dinámicamente. Además, cada uno de los dispositivos debe detectar la presencia de los demás dispositivos dentro de la red y comunicarse con ellos, de manera que éstos puedan compartir la información y servicios que cada uno ofrece.

Por otra parte, la administración dentro de una red Ad hoc se realiza de forma distribuida, es decir cada uno de los nodos participa en la toma de decisiones, realizando las funciones propias de mantenimiento de red y tomando parte en los algoritmos de enrutamiento (Vidal, 2003).

II.3. Problemática en redes ad hoc

Las redes ad hoc presentan varias limitaciones de operación debido a las sus propias características. Las principales características de las redes ad hoc son las siguientes: (Vidal, 2003);(Jun-Zhao, 2001)

1. Cambio constante e impredecible en la distribución de los nodos (topología de red).
2. Participación de los nodos en las actividades de mantenimiento de la red y enrutamiento.
3. Poder de cómputo limitado de algunos o todos los nodos en el entorno.
4. Uso de enlaces inalámbricos en la comunicación.

A continuación se describen brevemente algunas las limitaciones existentes dadas las características anteriores

- Limitaciones de banda ancha: Actualmente los enlaces inalámbricos tienen menor capacidad de transmisión que las redes cableadas. Sin embargo, al igual que en las redes cableadas, en las redes ad hoc se deben atender los mismos servicios y demandas, por lo que el congestionamiento generado puede afectar el tiempo de transmisión.
- Limitaciones de energía por operación: Algunos o todos los nodos en una red ad hoc operan con una fuente de poder pequeña, por lo que la conservación de energía es crucial, ya que esto dicta cuándo la red es operacional o no.

- Limitaciones de seguridad: Las redes inalámbricas móviles son generalmente más sensibles a ataques de seguridad que las redes cableadas. Una de las razones principales de ataques se debe por mantener la señal de transmisión libre de uso a cualquier nodo que reciba cobertura. Dado lo anterior es necesario proteger el acceso a la red, prevenir el acceso y modificación indebido de la información, así como rechazar ataques de nodos maliciosos.

Cualquier propuesta enfocada a las redes ad hoc, debe de tomar en cuenta las limitaciones mencionadas anteriormente.

II.4. Tecnologías inalámbricas

Actualmente, existen varias soluciones disponibles para permitir la formación de redes de cómputo inalámbricas, las dos tecnologías más comunes son Bluetooth y IEEE 802.11. A continuación se describen brevemente:

II.4.1. Bluetooth

Bluetooth es una tecnología de comunicación inalámbrica de corto alcance y de bajo costo. Ésta fue diseñada para la interconexión de dispositivos móviles, y debido a que es independiente de una infraestructura fija, es un candidato para la construcción de redes ad hoc (Willekens, 2001).

Esta tecnología opera en la banda libre ISM (*Industrial, Scientific and Medicine*) de frecuencia de 2.4 GHz, la cual la comparte con otras aplicaciones que transmiten libremente energía radial, esto puede ocasionar interferencias en el enlace Bluetooth.

El radio de transmisión de esta tecnología es aproximadamente de 10 metros y cuenta con un canal de 1 MHz para transmitir paquetes, los cuales se transmiten cada 625 ms (Haartsen, 2000).

Básicamente, la tecnología Bluetooth elimina la necesidad de cables y conectores entre teléfonos, módems, PDAs, computadoras, impresoras, proyectores, pizarrones electrónicos, etc, lo que abre el camino para nuevos dispositivos y aplicaciones muy diferentes que pueden impactar varios campos de comunicación inalámbrica.

La comunicación entre dispositivos en una red que utiliza la tecnología Bluetooth (red Bluetooth) se lleva a cabo mediante una relación maestro-esclavo. Los roles de maestro y esclavo son dinámicos: el dispositivo que inicia la comunicación actúa como maestro y el otro como esclavo. El dispositivo maestro se puede conectar con 7 esclavos formando una red Piconet (red Bluetooth formada por 2 a 8 dispositivos, donde sólo existe un maestro y los demás son esclavos). A cada maestro se le asigna una frecuencia de secuencia, la cual la utilizan sus esclavos. Sólo se permiten las comunicaciones entre maestro y esclavo, el dispositivo maestro es el que controla la comunicación. Varias piconets se pueden localizar en una misma área, y se pueden interconectar de manera que cada dispositivo participe en más de una piconet (Melodia, 2003). La especificación de Bluetooth permite que cada nodo asuma múltiples roles, de manera que un nodo sea maestro en una piconet y en una o más piconets sea esclavo, o sea esclavo en múltiples piconets. La interconexión de piconets se les conoce como scatternet (Basagni, 2004).

II.4.2. IEEE 802.11

El estándar inicial de comunicación IEEE 802.11 fue publicado en 1997 en el *Institute of Electrical and Electronics Engineers* (IEEE), actualmente este estándar es

la tecnología más adecuada para dar soporte a la conectividad inalámbrica dentro de redes de área local (Bruno, 2001).

Para el desarrollo de esta tecnología, la IEEE asignó un grupo de trabajo, el cual se ha extendido en varios grupos, designados por las letras a,b,c, etc. Entre las especificaciones más populares se encuentran 802.11a, 802.11b y 802.11g. Cabe mencionar que el desarrollo de la familia de protocolos de 802.11 aún esta en crecimiento y desarrollo.

En particular la especificación 802.11b ha tenido mucho éxito (Broustis, 2004). Actualmente existen muchos productos en el mercado que utilizan este estándar. Por otra parte, muchas compañías están desarrollando productos basados en 802.11a. Debido a que las especificaciones de 802.11a y 802.11b fueron creados para que operaran en diferentes bandas de frecuencia, por lo que los productos generados a partir de ellas no son compatibles. Por otra parte, el grupo 802.11g, busca tener compatibilidad con productos de 802.11b.

La especificación 802.11a fue creada para que opere en la banda de frecuencia de los 5GHz a tasas de transmisión de hasta 54 Mbps. Mientras que 802.11b fue creada para que opere en la banda de frecuencia de los 2.4 GHz a tasas por arriba de los 11 Mbps. Por lo que corresponde a 802.11g aún está en desarrollo, y se espera que opere a tasas por arriba de 20 Mbps en la frecuencia de los 2.4 GHz.

La tecnología IEEE 802.11 se puede utilizar tanto para implementar redes inalámbricas con infraestructura como redes inalámbricas sin infraestructura. El modo de transmisión ad hoc de esta tecnología es con los nodos que se cuenta con comunicación directa, es decir a un solo salto, sin embargo, está emergiendo la comunicación con dispositivos fuera del rango de transmisión a través de dispositivos intermedios (comunicación multisaltos) (Bruno, 2001). La tecnología 802.11 utiliza el paradigma (CSMA/CA).

La introducción de estas tecnologías ha generado un interés renovado en contribuir en la investigación y desarrollo de las redes ad hoc (Chlamtac, 2003).

II.5. Aplicaciones en las redes ad hoc

Las redes ad hoc móviles desde sus orígenes se han utilizado en el área militar, recientemente con la introducción de nuevas tecnologías, como Bluetooth e IEEE 802.11 se está ayudando al desarrollo de aplicaciones fuera del dominio militar. Básicamente las redes ad hoc se pueden aplicar en cualquier lugar, ya sea que exista poca o ninguna infraestructura de comunicación o donde la infraestructura sea costosa o inconveniente utilizarla (Chlamtac, 2003). A continuación se describen brevemente algunos ejemplos del uso de las redes ad hoc en diversos sectores:

- Militar: Las redes ad hoc en la milicia ha venido a mejorar la forma de comunicación dentro de los campos de batalla, permitiendo tener una red de información entre soldados, tanques, aviones, vehículos y cuarteles generales.
- Operaciones de emergencia: Las redes ad hoc pueden ser de gran ayuda a policías y bomberos en situaciones de búsqueda o rescate, donde no existe infraestructura de red o se encuentra dañada, y se requiere la formación rápida de una red de comunicación.
- Conferencias y entornos educativos: El uso de las redes ad hoc en conferencias y ambientes educativos es de gran ayuda para compartir información entre los participantes de estos entornos. Por ejemplo se podrían transferir archivos relacionados con la presentación de una clase o conferencia o bien distribuir información acerca de itinerarios.

II.6. Investigaciones en el área de redes ad hoc

Dado el interés por las redes ad hoc, en 1997 se estableció dentro de la IETF (*Internet Engineering Task Force*) un nuevo grupo de trabajo llamado MANET (*Mobile Ad hoc Networking group*), cuyo principal objetivo es estimular la investigación en el área de las redes ad hoc (Baker, 2002). Las investigaciones dentro del grupo MANET se han centrado principalmente en temas relacionados con enrutamiento, sin embargo, existen otros aspectos no menos importantes que este grupo considera, tales como calidad de servicio, seguridad y descubrimientos de servicios (Chen, 2002).

Además, se han llevado acabo varias conferencias o talleres por parte de la IEEE y la ACM. Por ejemplo, MobiHoc (*Simposium on Mobile Ad Hoc Networking & Computing*) ha sido una de las conferencias más importantes de ACM SIGMOBILE (*Special Interest Group on Mobility of Systems, Users, Data and Computing*). Cabe mencionar, que actualmente las investigaciones en el área de redes ad hoc están recibiendo más atención por parte de la academia, industria y del gobierno. Desde que este tipo de redes presentan muchos tópicos complejos, ha habido muchos problemas abiertos para investigación, así como contribuciones significativas (Arias, 2004).

II.7. Áreas de investigación dentro de las redes ad hoc

Las áreas de calidad de servicio, seguridad y descubrimiento de servicios son los tres aspectos principales que se deben considerar en el diseño de una red ad hoc de elevada funcionalidad y disponibilidad (Vidal, 2003). Aunque estos tópicos son todavía sujetos de investigación en las redes tradicionales (cableadas), debe prestarse especial atención en las redes ad hoc con la finalidad de brindar una forma viable para dar

soporte de calidad de servicio, seguridad, enrutamiento y descubrimiento de servicios. A continuación se aborda brevemente cada una de estas áreas de investigación:

II.7.1. Calidad de servicio

En los últimos años el número de usuarios en la red Internet ha crecido exponencialmente y lo seguirá haciendo. Los usuarios requieren de nuevas y mejores aplicaciones que le permitan comunicarse más fácilmente. Las aplicaciones multimedia o aplicaciones con ciertos requisitos en cuanto a ancho de banda o retardo han surgido ante esta necesidad. Ahora bien hay que tener en cuenta que Internet ofrece sus servicios tan rápido como le es posible (*best effort*, mejor esfuerzo) y esto no garantiza que un paquete sea recibido rápidamente o, peor aún, que sea recibido. Las Aplicaciones sensibles al retardo requieren de un mejor servicio con el fin de que los paquetes sean recibidos lo más pronto posible (Romero, 2003).

Básicamente, la Calidad de Servicio (QoS, *Quality of Service*) es un nivel de servicio proporcionado a un cliente permitiéndole obtener el rendimiento requerido para una aplicación en particular.

Actualmente existen soluciones que han sido estudiadas por bastante tiempo, las cuales permiten brindar una buena calidad de servicio. Estas soluciones involucran técnicas de reservación de recursos, calendarizado y eliminación de paquetes y demás mecanismos.

Niveles de calidad de servicio

El nivel de calidad de servicio se refiere a la capacidad que tiene una red para entregar un servicio requerido con cierto control en el desempeño de los parámetros de

calidad de servicio (ancho de banda, retardo y pérdida de paquetes). Existen tres niveles de calidad de servicio que puede ofrecer una red, los cuales se describen a continuación:

Servicio de mejor esfuerzo Este tipo de servicio no ofrece ningún tipo de garantía en la entrega de paquetes, sólo lo hace tan pronto le sea posible. Actualmente este tipo de servicio es ofrecido por Internet y no realiza ningún tipo de reservación de recursos.

Servicios Diferenciados (DiffServ) Este tipo de servicio clasifica a los paquetes basados en su requerimiento de servicio permitiendo priorizar los diferentes tipos de tráfico. Este tipo de servicio no garantiza que los paquetes sean atendidos, solamente le da una probabilidad más alta de que sea atendido más rápidamente.

Servicio Garantizado (IntServ) Este tipo de servicio brinda reservación de recursos sobre la red para asegurar que un paquete llegue a su destino. Básicamente este servicio involucra asegurar que para cierto tráfico habrá un ancho de banda disponible, un límite de retardo máximo y garantizar que no habrá pérdidas de paquetes mediante herramientas como RSVP (*Resource Reservation Protocol,* Protocolo de reservación de recursos).

Calidad de servicio en redes ad hoc

En la redes ad hoc, el brindar una calidad de servicio diferente a la de mejor esfuerzo es un problema complejo, lo anterior hace que esta área sea un reto en el campo de las redes ad hoc. La habilidad de la red para brindar calidad de servicio depende de las

características intrínsecas de los elementos de la red (capacidad, topología dinámica, energía, etc.). Por lo que corresponde a los modelos de calidad de servicio de Internet "servicios diferenciados" y "servicios integrados" requieren cierta cantidad de ancho de banda disponible, taza de perdida de paquetes, etc. así como información de la topología. Lo anterior sugiere un modelo de calidad de servicios para redes ad hoc (Dimakis, 2003).

Modelo de calidad de servicio para redes ad hoc Un intento por desarrollar un modelo de calidad de servicio basado en los modelos DiffServ e IntServ fue FQMM (*Flexible QoS model for Manet*). Este modelo utiliza IntServ para aplicaciones con alta prioridad, y utiliza DiffServ para aplicaciones de baja prioridad. Otro modelo más realista para brindar calidad de servicio en las redes ad hoc está basado en un modelo adaptivo, de allí surge INSIGNIA, el cual es el primer protocolo de señalización específicamente diseñado para la reservación de recursos en ambientes ad hoc. Éste se basa en un mecanismo de señalización que incorpora información de control dentro de los paquetes de datos (cabecera IP) para indicar los recursos de ancho de banda mínimo y máximo (Vidal, 2003).

II.7.2. Seguridad

Al igual que en las redes tradicionales, en las redes ad hoc, los requisitos de seguridad son los mismos (disponibilidad, confidencialidad, integridad, autentificación, entre otros). Sin embargo, las características generales de una red ad hoc: topología dinámica, enlaces de ancho de banda limitado y capacidad variable, limitaciones de energía y seguridad limitada en la transmisión de datos, hacen del cumplimiento de los

requisitos anteriores un problema mucho más complejo de abordar y con ello nuevos retos.

Claramente, en las redes ad hoc, los nodos son especialmente más susceptibles a ataques que los nodos en las redes tradicionales. Sin embargo el nivel o grado de seguridad a aplicar depende en gran medida al ambiente en el cual los nodos se encuentran operando.

Algunas investigaciones en el área, especifican que los sistemas de detección de intrusiones, seguridad de enrutamiento y los sistemas de gestión de clave son los tres aspectos que cualquier política de seguridad en redes ad hoc debe cubrir. A continuación se habla brevemente de cada uno de estos aspectos (Zheng, 2005).

Sistemas de detección de intrusiones

Una intrusión se define como "cualquier acción que intente comprometer la integridad, confidencialidad o disponibilidad de la red". Las técnicas de protección de intrusiones, tales como el cifrado y la autenticación, son necesarias como primera línea de defensa en una red ad hoc. Sin embargo, la protección de intrusiones por sí sola no es suficiente ya que no existe seguridad perfecta en los sistemas de red, especialmente en las redes ad hoc. La detección de intrusiones permite establecer una segunda línea de defensa, y puede ser necesaria para detectar cuando un sistema está siendo atacado (Zheng, 2005).

Existe una arquitectura distribuida y cooperativa para la detección de intrusiones que utiliza un modelo de detección de anomalías. En el sistema propuesto, cada nodo de la red ejecuta un agente que monitoriza las actividades locales. Si el agente detecta una intrusión a partir de las trazas locales inicia un procedimiento de respuesta. Si detecta

una anomalía, pero no tiene una evidencia concluyente de que se esté produciendo un ataque, puede iniciar un proceso cooperativo con sus nodos vecinos, de modo que puedan determinar finalmente si la intrusión ha tenido o no lugar (Vidal, 2003).

Seguridad de enrutamiento

Los nodos en una MANET actúan como enrutadores, participando en el enrutamiento para descubrir y mantener rutas a otros nodos de la red. En general, el objetivo de un algoritmo de enrutamiento es establecer una ruta adecuada entre cada par de nodos. Si el resultado de este algoritmo es manipulado, el funcionamiento normal de la MANET puede verse seriamente afectado, actuando en contra del requisito de disponibilidad. Por este motivo la seguridad en el enrutamiento tiene un gran peso sobre la seguridad del sistema (Chlamtac, 2003).

Servicios de gestión de claves

Este servicio asiste a los nodos en el proceso de comunicación, permitiendo establecer relaciones de confianza entre las entidades que se comunican. Este servicio requiere del uso de claves criptográficas que serán utilizadas por las partes en comunicación. Con frecuencia, este servicio lo proporciona un tercero de confianza, en el que confían todos los nodos de la red (Vidal, 2003);(Zheng, 2005).

II.7.3. Enrutamiento

Es fundamental proporcionar un mecanismo que permita direccionar tráfico de un nodo origen a un nodo destino a través de red. Debido a la movilidad de los nodos en las redes ad hoc, los mecanismos de enrutamiento desarrollados para las redes cableadas se vuelven inoperables en este tipo de redes.

Desde los años 70s se han estado desarrollando protocolos para las redes ad hoc móviles, los cuales deben tener en cuenta las limitaciones de estas redes, tales como el alto consumo de energía, la limitación de ancho de banda, las tasas altas de errores, etc (Chlamtac, 2003);(Toh, 2002).

Actualmente, los protocolos de enrutamiento están clasificados en tres categorías: protocolos proactivos, protocolos reactivos y protocolos híbridos. A continuación se describen cada una de estas categorías.

Protocolos proactivos

Este tipo de protocolos manejan tablas de enrutamiento, cada uno de los nodos que pertenece a la red cuenta con una tabla que deberá de mantener actualizada. Cuando la topología cambia, los nodos mandan mensajes a través de la red para mantener actualizada y consistente la información de enrutamiento. La ventaja de este tipo de protocolos es que una vez que la ruta es formada su uso es eficiente, sin embargo, la necesidad de mantener actualizada las tablas genera tráfico extra al de los paquetes que se transmiten en red.

DSDV El protocolo DSDV (*Destination Sequenced Distance Vector*) es un protocolo proactivo, éste esta basado en la versión tradicional del algoritmo de enrutamiento vector-distancia. Este mecanismo previene los ciclos de enrutamiento en una red móvil de enrutadores, mediante el uso de números de secuencia para cada destino en la tabla de enrutamiento. La tabla de enrutamiento contiene información de destino, siguiente salto, distancia para alcanzar destino y el número de secuencia.

Debido a que es un protocolo proactivo, la tabla de enrutamiento se actualiza periódicamente, lo que puede generar tráfico de control. Para aliviar este problema, el

protocolo DSDV utiliza dos tipos de paquetes: paquetes *full dump* y paquetes *incremental*. En el paquete *full dump* toda la información de enrutamiento se envía, los cuales se transmiten en periodos de movimientos ocasionales. Mientras que pequeños paquetes *incremental* se utilizan para enviar sólo información que ha cambiado desde el último full dump (Toh, 2002).

Protocolos reactivos

Este tipo de protocolos, a diferencia de los proactivos, no se preocupa de detectar cambios en la red, por lo cual elimina la necesidad de mantener tablas de enrutamiento actualizadas. Cuando un nodo desea transmitir un paquete a otro nodo, se procede con anterioridad a determinar la ruta que deberá seguir el paquete, una vez que se descubre la ruta, el paquete se transmite. La principal ventaja de este tipo de protocolos es que sólo se propaga información por la red cuando es necesario, sin embargo se crea un sobrecosto grande cuando la ruta está siendo determinada.

AODV El protocolo AODV (*Ad Hoc On-Demand Distance Vector Routing*) es una mejora sobre el protocolo DSDV, ya que típicamente minimiza el número requerido de broadcast, debido a que se crean las rutas sobre demanda, es decir sólo cuando se requiere (Toh, 2002).

Cuando un nodo origen requiere una ruta hacia un nodo destino, éste envía un broadcast con un paquete de solicitud de ruta (RREQ) hacia los nodos vecinos. Cada uno de los vecinos revisa su tabla de enrutamiento para verificar si tiene ruta hacia el destino. Si el nodo no tiene la ruta, éste retransmite el broadcast a sus vecinos. Si el nodo receptor es el destino o tiene la ruta hacia el destino, se envía un mensaje de Respuesta (RREP) al nodo origen. Las entradas de enrutamiento para el nodo destino

se crean en cada nodo intermedio cuando el paquete RREP se envía de regreso al origen. Al llegar al nodo origen, éste encamina los datos a transmitir en base a la información brindada por las entradas (Chen, 2002).

DSR El proceso de descubrimiento de ruta de DSR (*Dynamic Source Routing*) es similar al de AODV. Este protocolo consiste de dos fases: descubrimiento y mantenimiento de rutas. Un nodo origen genera un paquete de solicitud de ruta sólo cuando ocupa una nueva ruta a un destino. Cuando se solicita una ruta, primero se verifica en cache si ya se cuenta con la ruta, de ser así se transmiten los datos, de lo contrario se transmite el paquete de solicitud de ruta conteniendo la dirección destino, un identificador y la dirección del nodo origen. Cada uno de los nodos revisa si tiene la ruta, si no la tiene añade su propia íainformación y la transmite a los siguientes vecinos. Un paquete de respuesta sólo se genera cuando se alcanzó la ruta o destino, y se envía de regreso al nodo origen en reversa conteniendo la ruta al destino. La ruta se añade al paquete de datos enviándolo al destino utilizando los nodos intermedios (Chen, 2002);(Toh, 2002).

Protocolos Híbridos

Este tipo de protocolos hacen uso de ambos enfoques mediante la adaptación en base a las condiciones en un momento dado.

ZRP Como se observó, los protocolos proactivos utilizan exceso de banda ancha para mantener la información de enrutamiento, mientras que los protocolos reactivos involucran retardos de solicitud de ruta. El protocolo ZRP (*Zone Routing Protocol*) combina ambos enfoques y utiliza las mejores propiedades de cada uno.

El protocolo ZRP está definido en tres documentos separados de la IETF denominados Internet Draft, los cuales son documentos en fase de evaluación para su estandarización. Estos documentos son: IARP (*The Intrazone Routing Protocol*), IERP (*IntErzone Routing Protocol*) y BRP (*Bordercast Resolution Protocol*). El ZRP reduce la cantidad de tráfico comparado con los protocolos reactivos o proactivos. Las trayectorias a nodos dentro de una zona están disponibles inmediatamente. El ZRP es capaz de identificar múltiples rutas al destino, lo que incrementa la confiabilidad y el rendimiento (Beijar, 2002) .

En la figura 3 se muestra un esquema con varios protocolos de enrutamiento categorizados

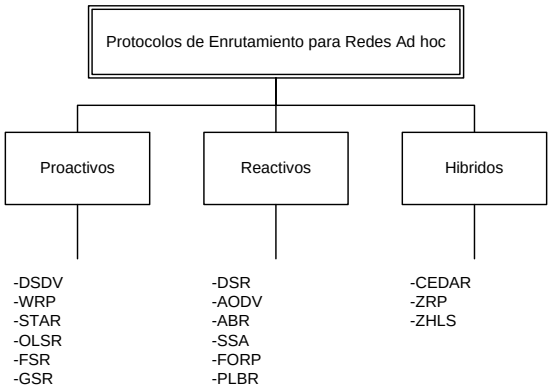

Figura 3: Clasificación de Protocolos de Enrutamiento para Redes Ad hoc

II.7.4. Descubrimiento de servicios

El descubrimiento de servicios es un componente muy importante y necesario de las redes ad hoc. El descubrimiento de servicios está definido como un problema de la

localización automática de los servicios existentes en una red. Cada uno de estos servicios se puede anunciar para que los usuarios puedan descubrirlos. Existen protocolos y esquemas disponibles para dar apoyo tanto al anuncio como al descubrimiento de servicios. Los dispositivos móviles inalámbricos pueden conectarse en cualquier lugar y anunciar y descubrir servicios. Los protocolos de descubrimiento y anuncio son una herramienta importante para ayudar a encontrar los servicios en la red desde cualquier lugar que se esté conectado.

II.8. Conclusiones

A lo largo de este capítulo se presenta un estudio del estado actual de las Redes Ad hoc Móviles, las cuales a diferencia de las redes tradicionales, presentan grandes retos que se deben abordar con la finalidad de que éstas funcionen adecuadamente. Este nuevo paradigma de las redes de cómputo se puede aplicador en diferentes entornos, desde el académico al militar, cualquier propuesta debe tomar en cuenta los problemas de comunicación inherentes a este tipo de redes. Actualmente, existe un gran interés en contribuir al desarrollo de las redes ad hoc, tanto por parte de la academia, industria y gobierno, por lo que muchos grupos de trabajo han surgido con la finalidad de estimular el desarrollo e investigación en las redes ad hoc. La calidad de servicio, enrutamiento, seguridad y descubrimiento de servicios son áreas abiertas de investigación, en este trabajo es de especial interés contribuir en el descubrimiento de servicios, por lo que en el próximo capítulo se profundiza en esta área de investigación.

Capítulo III

Descubrimiento de servicios

III.1. Introducción

El incremento en la popularidad de aplicaciones de red y el uso potencial de las redes ad hoc en la vida civil, ha generado especial interés en la comunidad científica por compartir los recursos existentes en este tipo de redes. La mayoría del trabajo relacionado con el descubrimiento de servicios está enfocado a redes con infraestructura estable. Sin embargo, las redes ad hoc presentan distintas características que hacen el descubrimiento de servicios un reto (Liu, 2003).

Este capítulo esta dedicado al descubrimiento de servicios, para ello se presentan definiciones, criterios de diseño y se analizan las propuestas de descubrimiento de servicios de mayor relevancia en los últimos años, tanto en redes con y sin infraestructura de red estable.

III.2. Definiciones

Con la finalidad de abordar el descubrimiento de servicios, a continuación se definen algunos conceptos básicos

- **Servicio:** Entidad que puede ser utilizada por una persona, programa de cómputo u otro servicio. Por ejemplo un archivo, un canal de comunicación a otro usuario, dispositivos de hardware, etc (Chen, 2002).

- **Descubrimiento y anuncio de servicios:** Éste se define como un problema de localización automática de los servicios existentes en una red (Liu, 2003).

- **Protocolo de descubrimiento de servicios:** Mecanismo que permite a un dispositivo localizar automáticamente los servicios que se ofrecen dentro de una red de cómputo y determinar si cumplen con los requerimientos de búsqueda. Dichos mecanismos incluyen técnicas para detectar cambios en la disponibilidad del servicio y para mantener una vista general de la red (Dabrowski, 2001).

III.3. Participantes en el descubrimiento de servicios

Los protocolos para descubrimiento de servicio consisten de por lo menos dos y en algunos casos de tres participantes (Marin-Perianu, 2005). A continuación se hace referencia a cada uno de ellos.

- **Cliente o usuario:** Es aquella entidad que está interesada en encontrar un servicio.

- **Servidor:** Es aquella entidad que ofrece el servicio.

- **Servidor de directorios:** En algunos casos, los protocolos de descubrimientos cuentan con un servidor de directorios, el cual es un repositorio de servicios que contiene la descripción de alguno o todos los servicios que se ofrecen en la red.

III.4. Criterios de Diseño

En los últimos años han surgido diversos mecanismos para el descubrimiento y anuncio de servicios. El diseño de estos mecanismos debe de tomar en cuenta las diferentes restricciones que impone el tipo de entorno de desarrollo. En Zhu (2002) se han identificado varios criterios para el diseño que deben considerarse al momento de diseñar o bien seleccionar un mecanismo de descubrimiento de servicios. A continuación se hace referencia a cada uno de estos criterios:

III.4.1. Diseño de Servicios

Los protocolos para descubrimiento de servicios, con la finalidad de lograr su objetivo, deben de brindar un medio para nombrar servicios, así como para especificar atributos. Actualmente, muchas de las soluciones de descubrimiento de servicios cuentan con sus propios esquemas de nombrado. Al momento de diseñar un protocolo para el descubrimiento de servicios es importante decidir si establecer un nuevo estándar o hacer uso de alguno de los ya propuestos, el optar por los ya existentes permitiría la coexistencia entre diferentes protocolos de descubrimiento.

Por otra parte, una vez que se ha encontrado un servicio, la solicitud para su uso es otro aspecto importante a considerar. Algunos de los protocolos para el descubrimiento de servicios brindan soporte para resolver este problema. Algunas de las tecnologías utilizadas son RMI (*Remote Method Invocation*), Java RMI, XML (*Extesible Markup Language*), SOAP (*Simple Object Access Protocol*) y HTTP (*HyperText Transfer Protocol*).

III.4.2. Diseño de directorios

Los directorios almacenan las descripciones de algunos o de todos los servicios que se ofrecen en la red. El diseño de estos directorios depende del ambiente de desarrollo del protocolo, por lo que al momento de diseñar un protocolo para el descubrimiento de servicios debe decidirse si utilizar un esquema de directorios centralizado o bien uno distribuido. En un esquema centralizado toda la información de los servicios se encuentra en un solo directorio, mientras en uno distribuido, la información se encuentra distribuida en varios directorios. Cabe mencionar que los directorios centralizados son susceptibles a cuellos de botella y a fallas en un solo punto, lo que genera que todo el sistema falle. Cuando se cuenta con escenarios de gran escala, hacer uso de un esquema distribuido es lo recomendable, ya que permite hacer búsquedas en cada directorio de manera más eficiente, aunque aumenten los costos de comunicación y de retardo.

III.4.3. Anuncio y búsqueda de servicios

El anuncio y la búsqueda de servicios son las piezas clave del descubrimiento de servicios. Los métodos *PULL* y *PUSH* se utilizan para localizar los servicios existentes en la red. Básicamente, en el método PULL los nodos anuncian la presencia de los servicios con los que cuentan, mientras en el método PUSH los nodos envían peticiones de servicios, a las cuales responderán los servidores de dichos servicios.

Los protocolos de descubrimiento de servicios hacen uso de técnicas de comunicación para el anuncio y búsqueda de servicios. A continuación se especifican las cuatro más comunes:

- **Unicast:** Esta técnica de comunicación se utiliza cuando se desea establecer comunicación directa entre un solo emisor y un solo receptor. Los protocolos de

descubrimiento hacen uso de ella cuando se conoce la dirección de un directorio o de un servicio, así como cuando se desea realizar un anuncio o petición directamente.

- **Anycast:** Esta técnica de comunicación se utiliza para establecer comunicación entre un solo emisor y otro receptor, el cual pertenece a un grupo. Los protocolos de descubrimiento de servicios hacen uso de esta técnica cuando varios servicios similares están agrupados y se desea obtener el más próximo.

- **Multicast:** Esta técnica de comunicación se utiliza para establece comunicación entre un solo emisor y varios receptores. El uso de multicast facilita el funcionamiento de los protocolos para el descubrimiento de servicios, principalmente en clientes móviles. Sus principales desventajas son la escasez de las direcciones multicast globales y el hecho que algunos enrutadores no permiten multicast aun cuando lo soporten. Adicionalmente el uso de multicast también introduce sobrecosto (overhead) en los canales de comunicación en comparación con el uso de unicast, ya que más nodos están involucrados en la comunicación.

- **Broadcast:** Esta técnica de comunicación se utiliza para establecer comunicación entre un emisor y todos los receptores que existan en la red. El utilizar broadcast para inundación de paquetes es un proceso costoso, ya que causa redundancia de paquetes, colisiones y desperdicio de ancho de banda (Guo, 2005).

III.4.4. Selección de Servicios

Debido a que en una red de cómputo pueden existir muchos servicios similares, es necesario localizar los servicios que el cliente requiere de manera precisa y eficaz.

Algunas de las técnicas utilizadas por los protocolos de descubrimiento para la selección de servicios se describen a continuación.

- **Selección por usuario vs Selección por protocolo:** La selección de servicios la puede realizar el protocolo de descubrimiento de servicios o bien por el cliente que solicitó el servicio. Dejar la responsabilidad al protocolo permite simplificar los programas clientes, lo que sólo requiere una participación mínima del usuario. Sin embargo, el usar este esquema puede no reflejar las necesidades reales del usuario. Por otra parte, el dejar la responsabilidad al cliente de la selección, permite obtener el servicio que se ajuste a sus necesidades, pero puede convertirse en un proceso tedioso para el cliente.

- **Selección por conciencia del contexto:** El contar con información de contexto puede ser útil para la selección de servicios. Cuando un usuario maneja en una carretera puede usar su teléfono celular para encontrar sus audífonos empleando una conexión Bluetooth o puede utilizarlo para acceder a su correo electrónico mediante una conexión de tercera generación. En estos dos escenarios, la selección de una de las conexiones puede basarse en la información del contexto. (Arias, 2004).

- **Selección por conciencia de ámbito:** El contar con información del ámbito puede ser útil y facilitar la selección de servicios. Los servicios se pueden categorizar en base a la ubicación geográfica, dominio y topología de red.

III.4.5. Seguridad y privacidad

Existen mucha investigación en el área de seguridad y privacidad, sin embargo, pocos son los protocolos de descubrimiento de servicios que integran elementos de

seguridad y privacidad en su funcionalidad.

- **Autentificación de usuarios y servicios:** El acceso a cada uno de los servicios se debe proteger de aquellos usuarios que tienen restringido su uso. Por ejemplo, algún servicio de almacenamiento sólo lo pueden acceder los usuarios que tienen autorización, inclusive los usuarios sólo podrán acceder algunos de los archivos almacenados. El manejar listas de control de acceso por cada uno de los servicios que se ofrecen se convierte en un problema, más para dispositivos con poco poder de cómputo.

- **Confidencialidad e integridad:** La comunicación entre componentes de descubrimiento de servicios deberá ser segura. Deben tomarse precauciones para no estar expuestos a ataques de usuarios que puedan hacer mal uso o cambiar la información de los servicios en el proceso de comunicación.

- **Disponibilidad y privacidad:** Los servicios y directorios están expuestos a ataques, por lo que es necesario contar con políticas de seguridad de manera que la privacidad de la información sea segura y que el uso del sistema sea fácil. El contar con un esquema de seguridad conlleva realizar trabajo adicional administrativo, lo que en un ambiente dinámico se vuelve una tarea abrumadora, afectando las ventajas obtenidas con el esquema de seguridad.

III.5. Protocolos

Actualmente existen distintos protocolos para el descubrimiento de servicios, pero en general enfocados a redes con infraestructura estable y poco adaptados a redes ad hoc. A continuación se describen brevemente algunos de estos protocolos.

III.5.1. SLP

El protocolo de localización de Servicios (SLP, *Service Location Protocol*) proporciona un marco de trabajo para el descubrimiento de servicios en redes IP (Veizades, 1997). Éste lo desarrolló el grupo de trabajo SvrLoc de la IETF; como todos los estándares desarrollados por esta comunidad internacional se describe a detalle en documentos llamados RFC (*Request For Comments*). Existen 2 versiones de dicho estándar, los cuales se definen en los RFC's 2165 y 2608 respectivamente.

Este protocolo consiste de 3 tipos de agentes: agentes de usuario, agentes de servicio y agentes de directorio (Guttman, 1999). Básicamente, SLP asigna un agente usuario a cada cliente para que actúe en su representación y realice el descubrimiento de servicios. El agente usuario interactúa con el agente de directorio, el cual contiene una lista de de los servicios que se están anunciando en la red por parte de los agentes de servicios.

SLP permite especificar las características del servicio que el cliente requiere. Después de realizar el descubrimiento, se obtiene la localización de todos los servicios que satisfagan la petición. Cabe mencionar que SLP no específica ningún mecanismo de cómo hacer uso de los servicios una vez que ha sido descubierto.

III.5.2. Universal Plug & Play

Universal Plug & Play (UPnP) es una arquitectura diseñada para permitir la interconexión de computadoras personales, dispositivos inalámbricos y electrodomésticos. UPnP es una tecnología definida por el consorcio UPnP Forum para extender el modelo de periféricos Plug & Play (Lee, 2002).

El UPnP consiste de un subconjunto de protocolos para el descubrimiento de servicios, interacción y notificación de eventos. El protocolo Simple Service Discovery Protocol (SSDP) se utiliza para descubrir servicios (Lee, 2002).

SSDP brinda un mecanismo que permite a cada uno de los dispositivos en la red localizar los servicios existentes, ya sea con poca o ninguna configuración previa. El descubrimiento de servicios se puede realizar de dos maneras. La primera consiste en que un cliente SSDP envía la petición de servicio utilizando multicast, de existir el servicio, el nodo solicitante recibe respuesta vía unicast. La segunda manera es mediante el anuncio periódico de presencia del servicio (Goland, 1999)

III.5.3. Jini

Jini es una tecnología definida por Sun Microsystems basada en Java. Ésta permite que los servicios existentes en la red se puedan anunciar, descubrir y solicitar. Cada uno de los servicios que se ofrecen los pueden utilizar tanto otros servicios como los usuarios (Chen, 2002).

El sistema Jini está compuesto por

- Una infraestructura
- Un modelo de programación.
- Servicios.

La infraestructura Jini es un conjunto de componentes que permiten la formación de federaciones Jini (servicios distribuidos en la red organizados en grupos).

Los componentes de la infraestructura son los siguientes (Sun, 1999):

- Un sistema de seguridad distribuido, integrado en RMI (*Java Remote Method Invocation*), el cual extiende el modelo de seguridad proporcionado por Java al contexto de los sistemas distribuidos.
- Un protocolo para el descubrimiento y unión, que permite a los servicios (sea hardware o software) descubrir, formar parte y anunciarse a otros miembros de la federación.
- El servicio de búsqueda (*lookup service*), el cual sirve como un repositorio de servicios. La información almacenada en el servicio de búsqueda son objetos que se pueden descargar como parte de una operación de búsqueda y actúan como representantes del servicio

El modelo de programación Jini es un conjunto de interfaces que permiten establecer la comunicación entre los servicios y la infraestructura. Las interfaces son:

- Interfaz leasing (arrendamiento): Define la forma de asignar y liberar recursos utilizando un modelo de renovación basado en periodos de tiempo.
- Interfaz de eventos y notificación: Permite la comunicación entre servicios basada en eventos.
- Interfaz de transacción: Permite la cooperación entre entidades de tal manera que los cambios realizados a un grupo ocurran automáticamente o bien que ninguno ocurra.

La infraestructura y el modelo de programación Jini permiten que los servicios se ofrezcan y se descubran dentro de la federación. Los servicios hacen uso de la infraestructura Jini para llamarse entre si, descubrirse entre si, y anunciar su presencia a otros servicios o usuarios (Sun, 1999).

Jini se basa en tres procesos denominados discovery, join y lookup. El primer proceso permite que un servicio se registre en un repositorio de servicios. El segundo permite utilizar un servicio una vez que se ha localizado. El tercer proceso permite localizar e invocar cada uno de los servicios registrados. (Sun, 1999)

III.5.4. BlueTooth SDP

El protocolo SDP (*Service Discovery Protocol*) está diseñado específicamente para permitir el descubrimiento de servicios en ambientes BlueTooth. El protocolo permite buscar un servicio en específico y buscar todos los servicios que estén actualmente disponibles (Gryazin, 2000). El SDP no define cómo acceder a los servicios, solamente realiza el descubrimiento, por lo que el método de acceso a estos servicios puede ser variado. Por otra parte, los servicios se representan por medio de un identificador de 128 bits denominado UUID (*Universally Unique Identifier*), además mediante éste también se pueden identificar los atributos asociados a los servicios.

III.5.5. Análisis de los protocolos anteriores

Cada una de las infraestructuras y arquitecturas de descubrimiento de servicios mencionadas anteriormente, no fueron desarrolladas especialmente para dar soporte al descubrimiento de servicios en redes ad hoc. A continuación se especifican los enfoques que sí dan soporte al descubrimiento en este tipo de redes.

III.5.6. Konark

Konark es un protocolo para descubrimiento y entrega de servicios específicamente diseñado para redes ad hoc multi salto. Esta tecnología está diseñada de manera que

cada dispositivo pueda actuar simultáneamente como servidor o cliente, por otra parte, Konark cuenta con un lenguaje de descripción de servicios basado en XML, el cual permite caracterizar a cada uno de los servicios a ofrecer. El diseño del protocolo asume que la red permite utilizar multicast, por lo que al realizar una petición, ésta llegará a todos los nodos en la red (Helal, 2003).

Cabe mencionar que cada dispositivo consta de una aplicación Konark que permite iniciar y controlar el anuncio de servicios, así como el uso de los mismos de manera interactiva.

III.5.7. GSD

GSD es un protocolo para el descubrimiento de servicios basado en grupos, específicamente diseñado para redes ad hoc. Esta tecnología hace uso de DAML (*DARPA Agent Markup Language*) para la descripción de cada uno de los servicios presentes en la red. Las peticiones de servicio se expresan en DAML y se comparan con las descripciones de servicio utilizando el modulo *service matching,* el cual determina si existe el servicio o no. Cada uno de los nodos contiene un repositorio con las descripciones de sus propios servicios, así como de los servicios remotos que ha registrado. Los nodos que contengan uno o más servicios, envían cada cierto tiempo una lista con los servicios que brindan. Dicho anuncio se transmite a todos los nodos que se encuentran en su rango de transmisión, a estos nodos se les dice que están a un salto. El nodo que realiza el anuncio también puede realizar el anuncio a múltiples saltos si se le especifica. GSD garantiza localizar aquellos servicios que se encuentren en la red. Además, la cantidad de servicios descubiertos por GSD es igual y en algunos casos mayor a los encontrados por protocolos basados en broadcast, pero con un número menor de mensajes. (Chakraborty, 2002)

III.5.8. PDP

El protocolo PDP (*Pervasive Discovery Protocol*) es un protocolo de ámbito local, totalmente distribuido en el que tanto las peticiones de servicio como las respuestas se envían por multicast, cada dispositivo almacena en una memoria local los anuncios recibidos, cuyo contenido comparte con los dispositivos que lo rodean. El PDP consigue minimizar el número de mensajes transmitidos por búsqueda, manteniendo tasas de descubrimiento de servicio altas, además permite que los dispositivos con mayor tiempo de disponibilidad transmitan un mayor número de respuestas, minimizando el consumo energético de los más limitados (Campo, 2005).

Algunas otras características de este protocolo son las siguientes (Campo, 2005)

- Todos los dispositivos que conozcan un servicio pueden responder a los mensajes de búsqueda, es decir aquellos que lo ofrecen y aquellos quien los tienen registrados en memoria
- Mezcla el método pull y push para beneficiarse de sus ventajas e intentando compensar sus inconvenientes.
- Cuando se solicita una búsqueda, el dispositivo transmite una petición a la red incluyendo los servicios que conoce, de manera que sólo responden los dispositivos que ofrecen servicios que no están incluidos en el propio mensaje de petición.
- Existen 2 versiones implementadas de PDP, una empleando el lenguaje de programación JSME, la cual se ha probado satisfactoriamente en una Pocket PC. La segunda implementación está desarrollada en el simulador de redes NS2.

III.5.9. VSD

VSD (*Service Discovery based in Volunteers*) es una arquitectura para el descubrimiento de servicios en ambientes heterogéneos y dinámicos de cómputo ubicuo. En VSD, un conjunto de nodos denominados voluntarios brindan servicio de directorios para facilitar el descubrimiento a los demás nodos en la red. Cada uno de los nodos en la red, ya sea voluntario o cliente, puede solicitar servicios como ofrecer servicios al mismo tiempo. Cada cliente tiene asignados k voluntarios, además cada voluntario mantiene una lista de voluntarios y un directorio de servicios para su región. VSD se adapta fácilmente a cambios en la topología y se reajusta automáticamente a dichos cambios. Esta arquitectura supera en la mayoría de los escenarios al modelo pull. (Kim, 2005)

Otras características de VSD son las siguientes:

- Los nodos que actúan como voluntarios responden directamente a los solicitantes de servicio, por lo que voluntarios pueden tener una utilización de servicios mayor.
- Los clientes introducen un retardo para responder a solicitantes de servicio, ya que las peticiones de servicio se transfieren a través de nodos voluntarios.
- Los Voluntarios anuncian periódicamente su existencia a través de mensajes que incluyen id, dirección origen, número de saltos y tiempo de vida.
- Los servicios se registran mediante mensajes que contienen número de saltos al nodo voluntario y descripción del servicio.
- El VSD no hace uso de mensajes broadcast al realizar una petición, ya que el nodo que realiza la petición sólo consulta a los k nodos voluntarios.

- La reasignación de un nodo voluntario se aplaza lo más posible para evitar sobrecarga en la red.
- El VSD se ajusta a la mayoría de escenarios de cómputo ubicuo, excepto en aquellos donde todos los nodos se mueven rápidamente, en este tipo de escenarios es preferible el uso de esquemas tipo broadcast.

III.6. Conclusiones

En este capítulo se realizó un estudio del estado actual del descubrimiento de servicios, tanto para redes con y sin infraestructura de red estable. Se percató que al disenar un protocolo para el descubrimiento de servicios es necesario tomar en cuenta diferentes criterios de diseño, los cuales están sujetos al tipo de entorno de desarrollo del protocolo. Además, se conocieron las principales soluciones para el descubrimiento de servicios, sin embargo la mayoría de ellas están enfocadas a redes con infraestructura estable y poco adaptadas a redes ad hoc móviles. Al analizar las soluciones de descubrimiento de servicios existentes, se percató que aún existen problemas por resolver para que el uso de estos mecanismos sea viable en las redes ad hoc. En el próximo capítulo se presenta una propuesta para el descubrimiento de servicios en redes ad hoc móviles que combina el descubrimiento de rutas con el de servicios, con lo que se espera mejorar el desempeño obtenido.

Capítulo IV

Descubrimiento de Servicios con OLSRSD

IV.1. Introducción

En el capítulo anterior se presentaron algunas soluciones para el descubrimiento de servicios, tanto para redes con y sin infraestructura de red fija. Cada una de las soluciones realiza el proceso de obtención de rutas y de servicios en forma independiente. Por otra parte, en Arias (2004) se presenta una solución de descubrimiento de servicios denominada AODVSD, la cual tiene como objetivo estudiar los efectos de combinar el descubrimiento de rutas y de servicios dentro de la capa de red. Dicha solución está basada en el protocolo reactivo AODV y ha generado resultados alentadores. El el presente trabajo de tesis se continúa con el estudio de este nuevo paradigma de descubrimiento de servicios, por lo que en este capítulo se propone una solución para brindar soporte al descubrimiento de servicios basada en un protocolo de enrutamiento proactivo para redes ad hoc móviles, para lo cual se tomó como ejemplo al protocolo OLSR.

A continuación, se da a conocer el funcionamiento detallado del protocolo OLSR, así como de las extensiones que se realizaron al mismo para que integrara el descubrimiento de servicios.

IV.2. OLSR

OLSR (*Optimized Link State Routing Protocol*) es un protocolo de enrutamiento especialmente diseñado para redes inalámbricas móviles. Éste pertenece a los protocolos de tipo proactivo, por lo que los nodos existentes en la red intercambian periódicamente información sobre la topología, permitiendo a los nodos contar con la información suficiente para determinar rápidamente rutas.

El concepto clave de OLSR es el uso de MPRs (*multipoint relays*) para el proceso de inundación de paquetes. Esta técnica reduce substancialmente la sobrecarga de paquetes en comparación con el mecanismo tradicional de inundación, ya que sólo los nodos seleccionados como MPRs son los encargados de transmitir los mensajes a todos los nodos que sean simétricos o cuenten con enlace (Clausen, 2003).

A continuación se hace referencia a los mecanismos de inundación de paquetes mencionados anteriormente. La figura 4 muestra el mecanismo tradicional, donde cada nodo en la red al recibir la primera copia de un mensaje lo retransmite. La figura 5 muestra el mecanismo utilizado por OLSR, el cual reduce el conjunto de nodos que retransmitirán los mensajes, permitiendo realizar un menor número de retransmisiones.

IV.2.1. Mensajes del protocolo OLSR

El protocolo OLSR requiere de tres tipos de mensajes para su funcionamiento, estos son: mensajes HELLO, TC y MID (Clausen, 2003). A continuación se describe cada uno de estos mensajes.

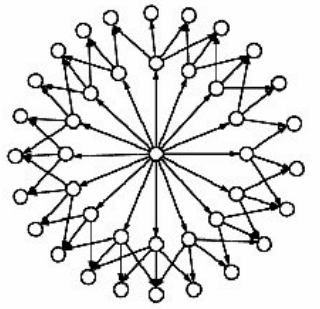

Figura 4: Red inalámbrica móvil utilizado inundación nodo a nodo

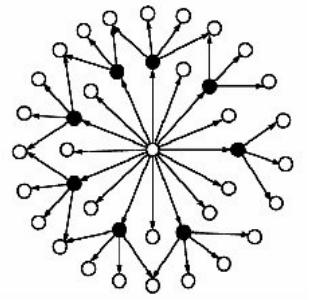

Figura 5: Red inalámbrica móvil utilizado MPRs (nodos negros)

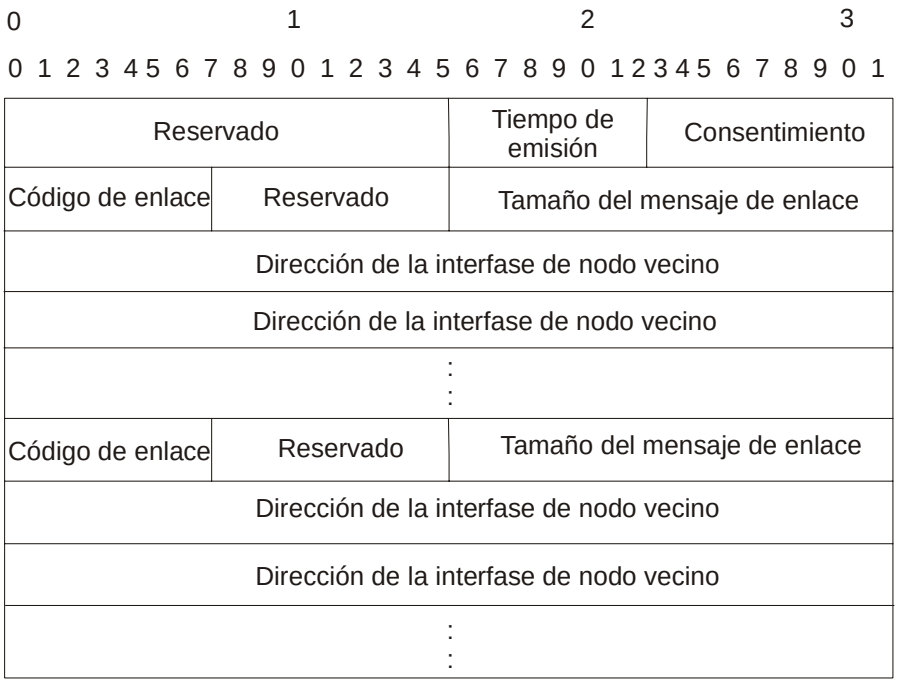

Figura 6: Formato de mensaje HELLO de OLSR

Mensajes HELLO

Este tipo de mensaje se utiliza para dar a conocer los enlaces que existen en la red y cuál es el estado (unidireccional, bi-direccional o roto) de dichos enlaces. Además, mediante el intercambio de mensajes HELLO se conocen los vecinos a un salto y a dos saltos, lo que permite seleccionar el conjunto de nodos MPRs, los cuales también se notifican mediante el uso de estos mensajes (Tonnesen, 2004).

El formato del mensaje HELLO se muestra en la figura 6. A continuación se describe brevemente cada uno de los campos

- **Reservado:** Este campo se debe especificar con el valor '0000000000000' con la finalidad de cumplir con la especificación del protocolo.
- **Intervalo de Emisión:** Este campo especifica el intervalo de emisión de los mensajes HELLO para una interfase en particular.
- **Consentimiento:** Este campo indica el consentimiento de un nodo para transportar tráfico a otros nodos. Por ejemplo, si dicho campo está especificado con la constante 0 nunca se debe seleccionar como MPR, si la constante es 7, siempre se debe seleccionar.
- **Código del enlace:** Este campo especifica información relacionada con el enlace de la interfase del emisor y la lista de interfases que se transmiten en este mensaje.
- **Tamaño del mensaje de enlace:** Indica el tamaño en bytes del mensaje de enlace, el cual se inicia desde el campo Link Code hasta el siguiente Link code, o bien, si ya no hay más enlaces por especificar, entonces hasta el final del mensaje.
- **Dirección de la interfase de un nodo vecino:** En este campo se especifica la dirección de la interfase de un nodo vecino.

Mensajes MID

El OLSR brinda soporte para el manejo de múltiples interfases en cada nodo, lo que permite a los dispositivos tener más de una interfase ejecutando OLSR. Mediante el uso de mensajes MID (*Multiple Interface Declaration*) se anuncia la presencia de múltiples interfases en un nodo (Clausen, 2003). El formato de este tipo de mensaje se muestra a continuación.

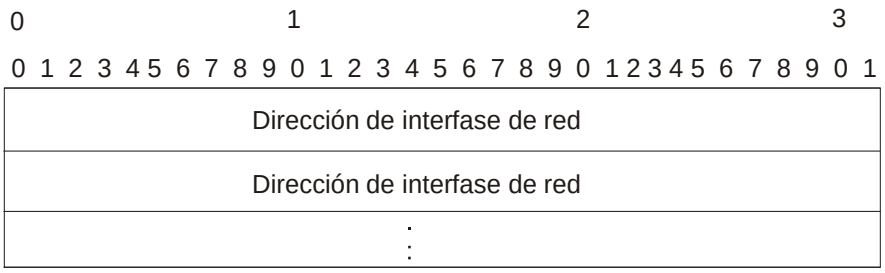

Figura 7: Formato de mensaje MID de OLSR

Como se observa este mensaje consta de uno o más campos, los cuales representan las direcciones de interfase de red con las que cuenta determinado nodo.

Mensajes TC

Mediante el empleo de los mensajes anteriores, un nodo conoce los enlaces que tiene con sus vecinos y el estado de los mismos, así como la información necesaria para aplicar el mecanismo de difusión utilizado por OLSR (Tonnesen, 2004). La información anterior permite contar con lo necesario para diseminar la información topológica de estado de enlace.

Los mensajes TC (*Topology Control*) se utilizan para diseminar información topológica de red a través del uso de MPRs. Cada uno de estos mensajes contiene una lista de vecinos correspondientes a determinado nodo, lo que permite a los demás nodos calcular su tabla de enrutamiento (Clausen, 2003).

El formato de los mensajes TC se muestra en la figura 8. A continuación se brinda la descripción de cada uno de los campos

- **ANSN:** Número de secuencia que permite saber que tan actualizada es la información.

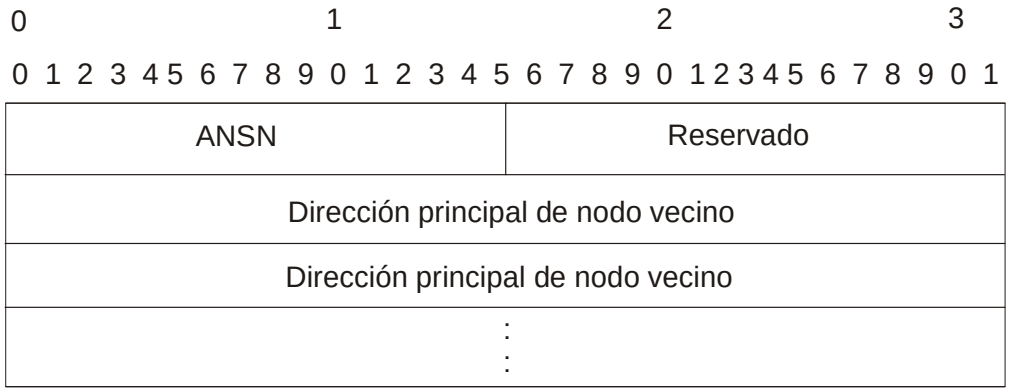

Figura 8: Formato de mensaje TC de OLSR

- **Reservado:** Este campo debe de ser especificado con el valor '0000000000000' con la finalidad de cumplir con la especificación del protocolo.
- **Dirección principal de nodo vecino**

IV.2.2. Paquete OLSR

Para la transmisión de todos los mensajes, OLSR define un formato de paquete básico que es entendido por todos los nodos que implementan el protocolo. Este paquete permite transportar varios mensajes a la vez, los cuales pueden ser de diferentes tipos (Tonnesen, 2004);(Clausen, 2003).

El formato del paquete OLSR se muestra en la figura 9. A continuación se describe brevemente cada uno de los campos.

- **Longitud del paquete:** Campo en el encabezado del paquete OLSR, el cual contiene la longitud en bytes del paquete OLSR.

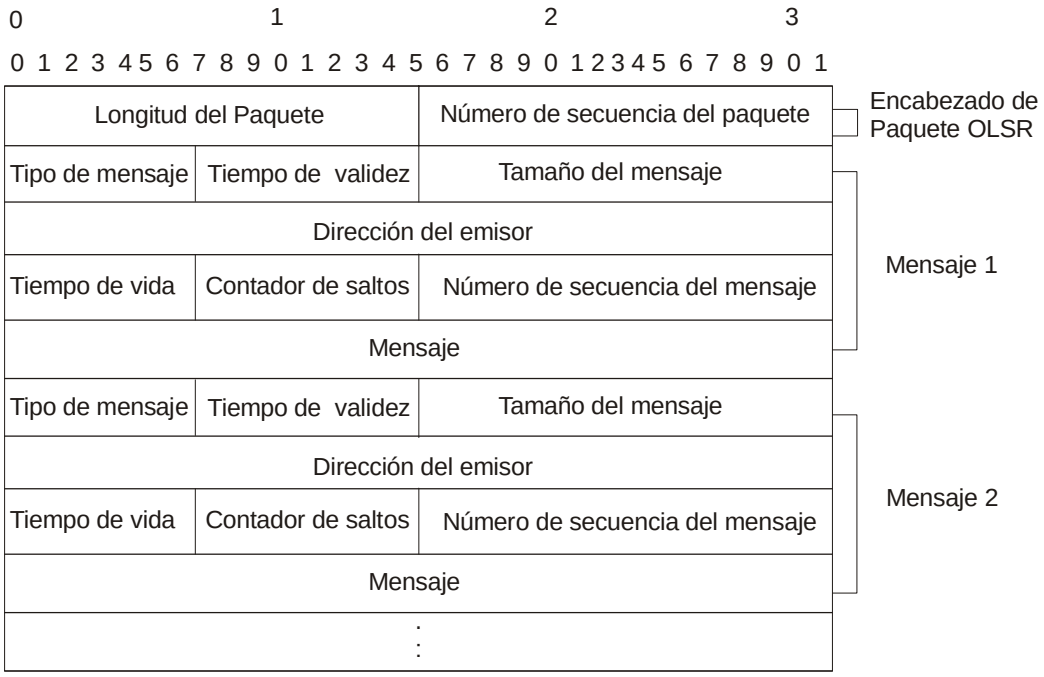

Figura 9: Formato de paquete OLSR

- **Número de secuencia del paquete:** Campo del encabezado del paquete OLSR que contiene el número de secuencia del paquete OLSR, el cual se incrementa en 1 cada vez que se transmite un nuevo mensaje.

- **Tipo de mensaje:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR, éste se utiliza para indicar el tipo de mensaje. OLSR utiliza valores del 0 al 127 para especificar mensajes ya definidos, así como posibles extensiones.

- **Tiempo de validez:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR, éste se utiliza para especificar cuánto tiempo la información transmitida en el mensaje deberá considerarse válida. El tiempo de validez está sujeto a actualizaciones.

- **Tamaño del mensaje:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR. En este campo se indica el tamaño en bytes del mensaje, dicha medida se determina desde el campo "Message Type" hasta el inicio del siguiente mensaje, o bien, hasta el final del paquete, si es que ya no existen más mensajes.

- **Dirección del emisor:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR. Este campo contiene la dirección principal del nodo que originalmente generó el mensaje, por lo que este campo no se modifica durante retransmisiones.

- **Tiempo de vida:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR. Este campo contiene el número máximo de saltos que un mensaje se retransmitirá. Antes de que un mensaje se retransmita, el tiempo de vida se debe decrementar en 1. En el momento que un mensaje tenga tiempo de vida de 0 o 1 dejara de ser retransmitido.

- **Contador de saltos:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR. Este campo indica el número de saltos que ha alcanzado un mensaje. Dicho valor se incrementa en 1 cada vez que el mensaje se retransmite.

- **Número de secuencia del mensaje:** Campo que pertenece a cada uno de los mensajes que se envían en el paquete OLSR. Este campo es un número único de identificación para cada mensaje, el cual lo asigna el nodo que lo genera. El número de secuencia se incrementa en 1 cada vez que un mensaje se origina por determinado nodo.

- **Mensaje**.

IV.2.3. Repositorios de información en OLSR

OLSR mantiene el estado de la red mediante el uso de varios repositorios de información, cada uno de los nodos obtiene y actualiza esta información a través del intercambio de mensajes. La mayoría de la información almacenada está sujeta a tiempos de vida, por lo que si a un mensaje se le acaba el tiempo de vida se remueve del repositorio.

Los repositorios que manejan cada uno de los nodos que implementan OLSR son los siguientes:

- **Repositorio de múltiples interfases:** Este repositorio contiene la información de los nodos que utilizan más de una interfaz de comunicación.
- **Repositorio de estado de enlace:** Este repositorio se mantiene para calcular el estado de enlace de los nodos vecinos.
- **Repositorio de vecinos:** Este repositorio contiene información de los nodos vecinos a un salto, este se actualiza dinámicamente basándose con información del repositorio de estado de enlace.
- **Repositorio de vecinos a dos saltos:** Este repositorio contiene a todos los nodos que se puede alcanzar por los nodos a un salto.
- **Repositorio de MPRs:** Todos los nodos seleccionados por el nodo local como MPR se almacena en este repositorio.
- **Repositorio de selectores :** En repositorio almacena todos los nodos vecinos que han seleccionado al nodo local como un MPR.

- **Repositorio de información topológica:** Este repositorio contiene información de todos los estados de enlace
- **Repositorio de duplicados:** Este repositorio contiene la información de los mensajes enviados y procesados por el nodo local.

IV.2.4. Procesamiento y retransmisión de mensajes

La especificación de OLSR señala que cuando un nodo recibe un paquete OLSR, éste deberá realizar las siguientes actividades por cada uno de los mensajes encapsulados en el paquete.

1. Si el paquete no contiene ningún mensaje, el paquete se debe descartar.
2. Si el tiempo de vida del mensaje es menor o igual a 0, o si el mensaje fue enviado por el nodo receptor, éste se debe borrar.
3. Condición de procesamiento
   a) Si el mensaje ha sido procesado anteriormente, no se debe procesar nuevamente
   b) De otra manera, si el nodo implementa el tipo de mensajes, entonces se debe procesar de acuerdo a las especificaciones para dicho mensaje.
4. Condición de retransmisión

a) Si el mensaje ha sido procesado anteriormente, no se debe procesar nuevamente.

b) De otra manera, si el nodo implementa el tipo de mensajes, entonces se debe procesar de acuerdo a las especificaciones para dicho mensaje.

c) De otra manera, el nodo lo debe procesar utilizando el algoritmo de omisión de retransmisión de mensajes.

El OLSR hace uso de un algoritmo por omisión dedicado a realizar la retransmisión de mensajes. Cabe mencionar que el OLSR permite utilizar un algoritmo propio u algún otro, sin embargo es necesario respetar que si llega un mensaje de tipo desconocido, éste se debe retransmitir. El algoritmo por omisión es el siguiente

1. Si el enlace al cual llegó el mensaje se considera no simétrico, el mensaje se descarta.
2. Si el TTL en el encabezado del paquete es 0, entonces el paquete se descarta.
3. Si el mensaje ya ha sido reenviado, entonces el paquete se descarta.
4. Si el emisor del último salto ha escogido este nodo como un MPR, entonces el mensaje se reenvía.
5. Si el mensaje será reenviado, el tiempo de vida se reduce en uno y el número de saltos del mensaje se incrementa de la misma manera en todas las interfaces. antes de realizar el broadcast del mensaje.

A continuación se brinda una visión general del protocolo OLSR. Se muestra la relación entre los repositorios de información y el procesamiento de mensajes, generación de mensajes y cálculo de ruta.

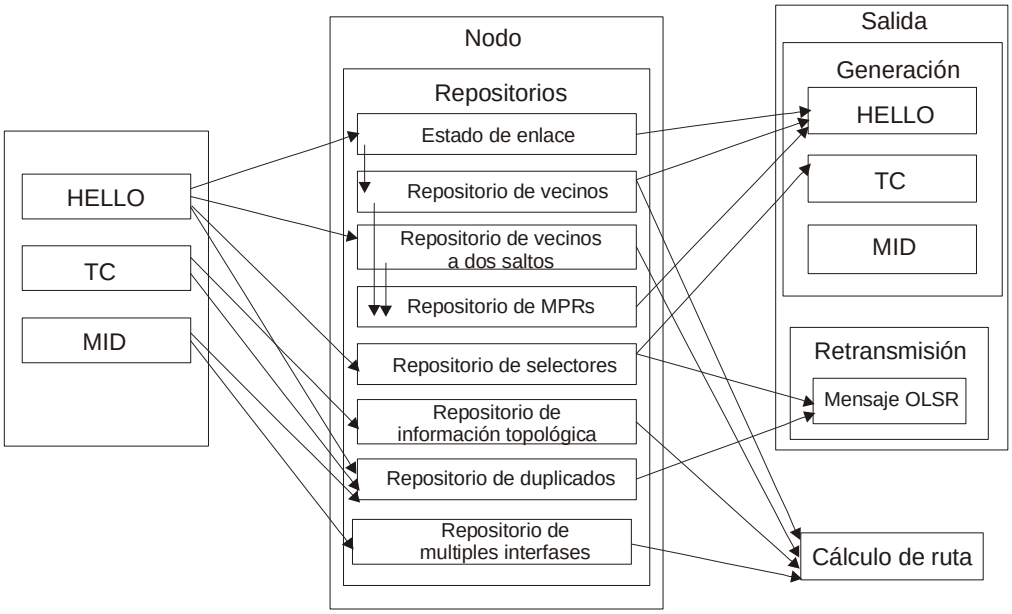

Figura 10: Visión general de OLSR

IV.3. Extensiones para el descubrimiento de servicios en OLSR

El diseño de este protocolo permite transmitir más de 1 mensaje a la vez dentro de un paquete OLSR. Además brinda la posibilidad de definir nuevos tipos de mensajes, de manera que se puede añadir funcionalidad al protocolo. Lo anterior se puede aprovechar para proporcionar a OLSR la capacidad de descubrir rutas y servicios simultáneamente. Cabe señalar que cualquier extensión que se le realice al protocolo no debe de romper compatibilidad con versiones anteriores.

Según indica en el RFC 3626 Clausen (2003) los tipos de mensajes manejados por OLSR y los posibles mensajes que extiendan funcionalidad están en el rango de 0 a 127. También se menciona que los mensajes HELLO, TC, MID y HDNA tienen asignados

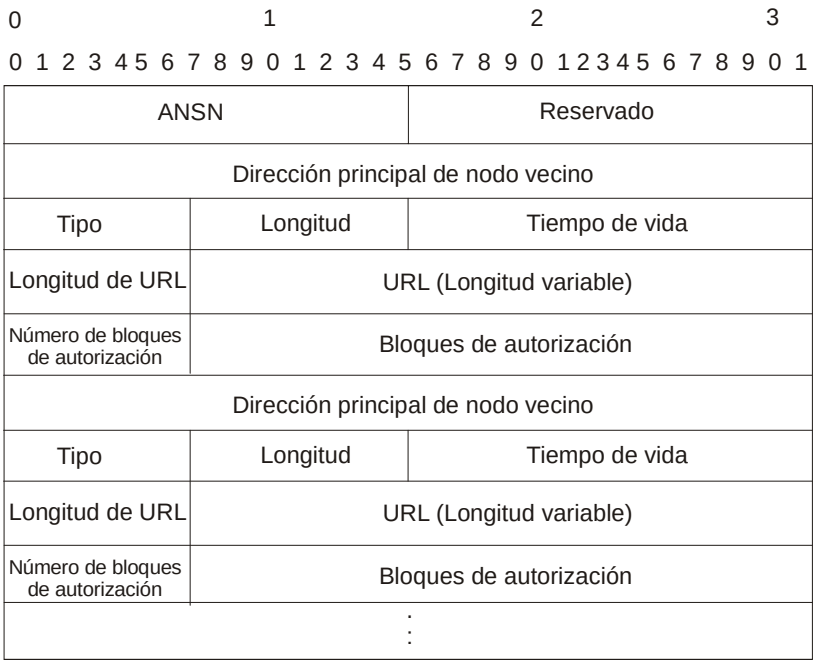

Figura 11: Formato de mensaje SVC de OLSR

el tipo de mensajes 1, 2, 3 y 4 respectivamente. Con la finalidad de difundir en la red cada uno de los servicios que brinda cada nodo, en el presente trabajo de tesis se realiza una extensión al mensaje TC (ver figura 8). Cabe señalar que el identificador asignado a esta extensión será por convención el 5.

La figura 11 muestra la extensión realizada en el presente trabajo al mensaje TC, la cual llamaremos SVC. La idea de estos mensajes es que al anunciar los nodos vecinos también se anuncien los servicios asociados a dichos nodos.

A continuación se describen brevemente cada uno de los campos del mensaje SVC

- **ANSN:** Número de secuencia que permite saber qué tan actualizada es la información.

- **Reservado:** Este campo se debe especificar con el valor '0000000000000' con la finalidad de cumplir con la especificación del protocolo.
- **Dirección principal de nodo vecino.**
- **Tipo:** Este campo indica el tipo de mensaje, éste se debe especificar con el valor 5, tal como se determino anteriormente.
- **Longitud:** Longitud en bytes de la extensión, ésta se determina desde el campo Type hasta antes del campo Advertised Neighbor Main Address, o bien, hasta que termine el mensaje, si es que no existe dicho campo.
- **Tiempo de vida:** Tiempo en milisegundos en el cual el nodo que recibe el anuncio considera el servicio como válido.
- **Longitud del URL:** Longitud de la definición del URL del servicio, los valores posibles son en el rango desde 1 a 255.
- **URL:** En este campo se especifica el URL del servicio asociado a determinado nodo.
- **Número de bloques de autorización.**
- **Bloque de autorización.**

IV.3.1. Procesamiento del mensaje SVC

El procesamiento de este tipo de mensajes se realiza de igual manera como se realiza el procesamiento del mensaje TC.

IV.3.2. Tabla de Servicios

Al igual que en Arias (2004), además de la extensión realizada al protocolo para que incluyera el anuncio de servicios, se propone que cada uno de los nodos disponga de una tabla de servicios, la cual estará compuesta por entradas que representan los servicios que el nodo brinda, así como de otros servicios que se proporcionan en la red.

Tabla I: Estructura de la Tabla de Servicios de OLSRSD

Id	Puerto	Protocolo	Dirección IP	Tiempo de vida	Lista de atributos

La tabla I muestra la estructura de una entrada en la tabla de servicios. A continuación se da la descripción de cada uno de los campos

- **Id:** Campo que permite identificar de forma única un servicio.
- **Puerto:** Número del puerto TCP/UDP utilizado por el servicio.
- **Protocolo:** Campo que permite especificar el protocolo utilizado por el servicio.
- **Dirección IP:** Campo que permite especificar la dirección IP del proveedor del servicio.
- **Tiempo de vida:** Tiempo en milisegundos que indica el tiempo de validez de un servicio. Una vez que el tiempo de vida de un servicio ha expirado, la entrada en la tabla de servicios se elimina.
- **Lista de atributos:** Campo que permite indicar cualidades ligadas a un servicio.

A continuación se brinda una visión general del protocolo OLSRSD. Se muestra la relación entre repositorios de información y procesamiento de mensajes, generación de mensajes y cálculo de ruta.

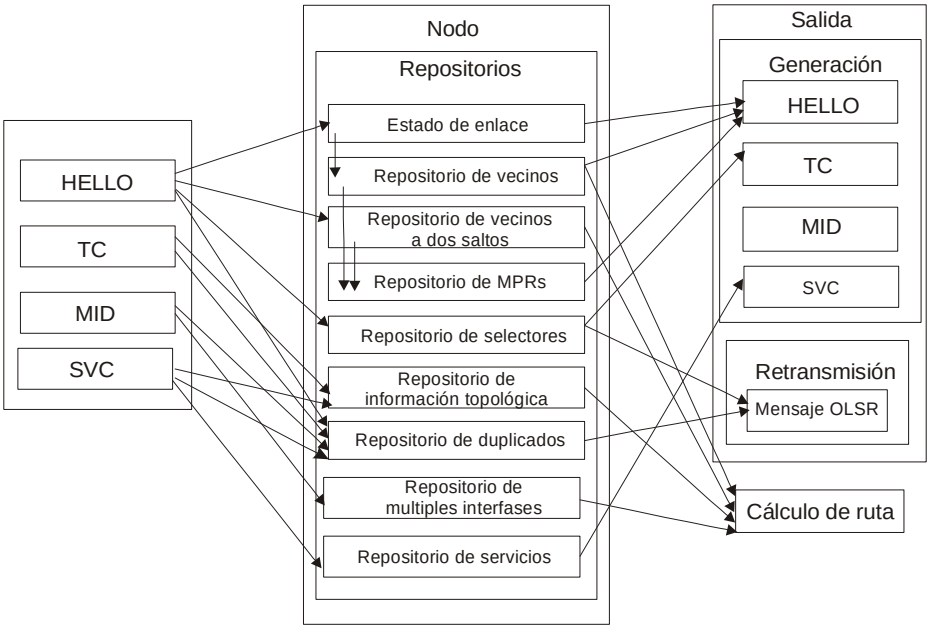

Figura 12: Visión general de OLSRSD

IV.4. Conclusiones

En este capítulo se propuso una solución para el descubrimiento de servicios en redes ad hoc móviles, ésta está basada en el protocolo de enrutamiento OLSR. Dicha solución extiende formato de los mensajes de dicho protocolo para permitir el descubrimiento de servicios al mismo tiempo que realiza el descubrimiento de rutas. En el próximo

capítulo se presenta un estudio de simulación que compara la propuesta (OLSRSD) con otra solución (AODVSD) para el descubrimiento de servicios, la cual realiza el descubrimiento de servicios de la misma manera. Además, se compara tanto OLSRSD y AODVSD con otros protocolos que realizan el descubrimiento de servicios y de rutas en forma independiente

Capítulo V

Estudio de Simulación

V.1. Introducción

En este trabajo, se ha propuesto una solución para el descubrimiento de servicios en redes ad hoc móviles (ver capítulo IV), la cual combina el descubrimiento de rutas con el descubrimiento de servicios. Dicha solución está basada en el protocolo de enrutamiento proactivo OLSR, y fue denominada OLSRSD (*Optimized Link State Routing Protocol with Service Discovery*). Esta propuesta tiene como objetivo estudiar los efectos de combinar ambas actividades dentro de la capa de red, en lugar de utilizar el enfoque tradicional que separa estas dos actividades dentro de las capas de red y de aplicación respectivamente.

A continuación se realiza un estudio de simulación que compara OLSRSD con protocolos de aplicación para descubrimiento de servicios en redes ad hoc. En Arias (2004) se presenta un estudio similar utilizando el protocolo reactivo AODV (AODVSD), por lo que en este trabajo se compara este paradigma entre capas utilizando tanto un protocolo reactivo como uno proactivo.

V.2. Modelo de Simulación

El simulador de redes NS-2 fue utilizado para analizar, implementar y comparar las diferentes soluciones para descubrimiento de servicios mencionadas anteriormente. Básicamente NS-2 es un simulador de redes enfocado a la investigación científica, el cual

brinda soporte para realizar simulaciones de redes cableadas e inalámbricas (terrestres y satelitales) utilizando diferentes protocolos y escenarios. Ver Apéndice 1

V.2.1. Parámetros de Simulación

Como se menciona anteriormente, en este trabajo se realizaron dos estudios, el primero enfocado a comparar OLSRSD con el protocolo AODVSD, mientras el segundo compara protocolos aplicativos para el descubrimiento de servicios contra protocolos de capa de red con soporte de descubrimiento de servicios (AODVSD y OLSRSD). Con la finalidad de capturar la mayor cantidad de efectos en las simulaciones, cada experimento se llevó a cabo bajo diferentes condiciones y factores. Para ambos estudios se utilizó IEEE 802.11 como protocolo MAC con función de coordinación distribuida (DFC), operando en un rango de transmisión de 100 metros y con tasa de transmisión de 11 Mbits/seg. El tiempo de ejecución para el primer estudio fue de 600 segundos de tiempo real, mientras para el segundo estudio fue de 700 segundos.

Para investigar mejor los efectos se utilizaron 3 diferentes configuraciones, dos de ellas empleadas para el primer estudio y una para el segundo. La primera configuración cuenta con 50 nodos definida en un área geográfica de 1000 m^2, la segunda configuración consta de 100 nodos definida en un área geográfica de 1500 m^2, mientras la tercera consta de 50 nodos definida en un área geográfica de 1000 m^2. En cada una de las configuraciones se utiliza *random waypoint* como modelo de movilidad, estableciendo un tiempo de espera entre cada movimiento de 2 segundos y una velocidad máxima de 2 m/s. Además, se generaron y utilizaron 5 escenarios de movilidad basados en este modelo. Por otra parte, se crearon 5 conexiones CBR mediante las cuales se introduce tráfico en la red, donde el tamaño de cada paquete CBR es de 512 bytes y son enviados cada 0.25 segundos. En el primer estudio se ofrecen 5 tipos de servicios, mientras en el segundo estudio se ofrecen 10 tipos, los cuales se generan con una distribución de

probabilidad uniforme. Cabe mencionar que las peticiones que se realizan se generan aleatoriamente siguiendo el mismo esquema, así como que cada uno de los parámetros utilizados son los valores comúnmente utilizados para evaluar redes ad hoc dentro de los grupos de investigación (Kim, 2005).

Los parámetros empleados en cada estudio se resumen en la tabla II.

Tabla II: Parámetros empleados por cada una de las comparativas

# Estudio	Estudio 1		Estudio 2
Parámetros	Valor 1	Valor 2	Valor 1
Número de nodos	50	100	50
Tipos de servicios	5		10
Tiempo de simulación	600 seg		700 seg
Área geográfica	1000 m^2	1500 m^2	1000 m^2
Modelo movilidad	Random waypoint		
Espera entre movimientos	2 seg		
Velocidad máxima	2 m/s		
Escenarios de movilidad	5		
Conexiones CBR	5		
Tamaño paquete CBR	512 bytes		
Transmisión de paquete CBR	0.25 seg		

V.2.2. Comparativas

- Peticiones de servicio VS paquetes de control: Cantidad de paquetes generados por el proceso de descubrimiento de rutas y de servicios, después de realizar determinada cantidad de peticiones de servicios.

- Peticiones de servicios VS servicios localizados: Cantidad de servicios encontrados por el proceso de descubrimiento de rutas y de servicios, después de realizar determinada cantidad de peticiones de servicios.

- Redundancia de servicios VS tiempo de adquisición: Tiempo transcurrido desde que se realizó la petición del servicio hasta que fue localizado, para esta comparativa se incrementa la cantidad de servicios de cada tipo que existe en la red.

- Redundancia de servicios VS servicios localizados: Cantidad de servicios encontrados en base a la cantidad de servicios del mismo tipo que existen en la red.

V.3. Experimentos y Resultados

A continuación se describen los experimentos que se llevaron a cabo. Además, se brindan los resultados obtenidos con sus respectivas gráficas, las cuales muestran cada una de las comparativas anteriores al ejecutar 5 simulaciones con diferente escenario de movilidad. Con la finalidad de comparar los resultados obtenidos, los experimentos realizados fueron llevados acabo utilizando los mismos escenarios de movilidad y de tráfico.

V.3.1. Estudio de OLSRSD VS AODVSD

Experimento 1: Peticiones de servicios VS paquetes de control

El objetivo de este experimento fue determinar el número de paquetes de control generados por los protocolos OLSRSD Y AODVSD al realizar 0, 50, 100 y 200 peticiones de servicio.

El experimento fue realizado utilizando el modelo de simulación descrito anteriormente tanto para una red de 50 nodos como para la de 100. Se llevaron a cabo 25 simulaciones (5 diferentes corridas por cada uno de los 5 tipos de peticiones) por cada configuración. A continuación se reportan los resultados obtenidos para cada una de las configuraciones:

Experimento 1 utilizando configuración de 50 nodos

En la primera parte de este experimento se utiliza una configuración que consiste de 50 nodos. Los parámetros de mayor relevancia de la configuración se muestran en la tabla III.

Tabla III: Parámetros empleados en experimento 1 para una red de 50 nodos

Parámetros	Valor
Número de nodos	50
Tipos de servicios	5
Área geográfica	1000 m^2
Tiempo de simulación	600 seg
Modelo de movilidad	Random waypoint
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5

La representación gráfica que realiza la compartiva peticiones de servicios VS paquetes de control para una red de 50 nodos se muestra en la figura 13

Análisis: Los resultados de la figura 13 permiten observar que OLSRSD genera un número mayor de paquetes de control que AODVSD, aunque los generados por este último van incrementando al aumentar el número de peticiones de servicio, a diferencia de OLSRSD donde el número de paquetes de control se mantiene constante. Además, se observa que cuando no se realizan peticiones de servicio, el protocolo OLSRSD genera paquetes de control, mientras AODVSD no lo hace, lo anterior era de esperarse dadas las características intrínsecas de ambos protocolos.

Experimento 1 utilizando configuración de 100 nodos

En la segunda parte del experimento se desea analizar los efectos al aumentar el número de nodos involucrados en la simulación, por lo que se aumentó la cantidad de

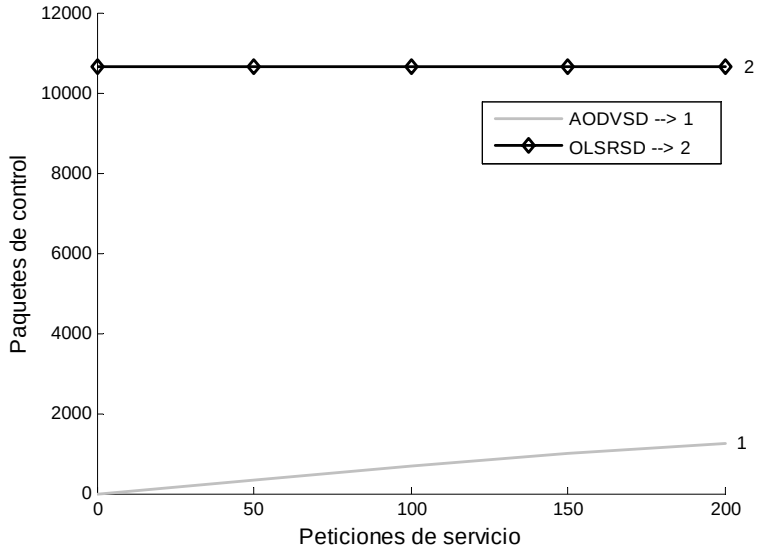

Figura 13: Comparativa peticiones de servicio VS paquetes de control para configuración de 50 nodos

nodos de 50 a 100. Los parámetros de la configuración para una red de 100 nodos se muestran en la tabla IV

La figura 14 muestra la representación gráfica de la compartiva peticiones de servicios VS paquetes de control para una red de 100 nodos.

Análisis: Los resultados obtenidos al simular una red de 100 nodos, indican un comportamiento similar a los obtenidos al simular una red de 50 nodos. OLSRSD genera un número de mensajes constante y superior que AODVSD, sin embargo, el número de mensajes que genera OLSRSD incrementa considerablemente de configuración a configuración.

Experimento 1 con configuración de 50 y 100 nodos (escala semilogarítmica)

Con la finalidad de analizar y comparar en detalle los resultados generados de ambos

Tabla IV: Parámetros empleados en experimento 1 para una red de 100 nodos

Parámetros	Valor
Número de nodos	100
Tipos de Servicios	5
Área geográfica	1500 m^2
Tiempo de simulación	600 seg
Modelo movilidad	Random waypoint
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5

protocolos, se procedió a representar el número de mensajes en escala semilogarítmica, lo cual nos permite comparar los resultados obtenidos por ambas configuraciones fácilmente. A continuación se muestran los resultados tanto para una configuración de 50 nodos como para una de 100 nodos en escala semilogarítmica

Análisis: Al analizar las figuras 15 y 16, se observa que la cantidad de paquetes de control incrementa de la primera a la segunda configuración. Como se puede observar la cantidad de paquetes generados por OLSRSD en todos los casos es mayor a los generados por AODVSD, pero se observa que al incrementar la cantidad de peticiones de servicios, los paquetes de control generados por AODVSD se van acercando a los generados por OLSRSD. Además, el número de mensajes generados por OLSRSD en una misma configuración no presenta cambio en crecimiento al aumentar las peticiones de servicio, a diferencia de AODVSD, el cual presenta un incremento acelerado desde la primera petición hasta la 50, donde posteriormente el incremento disminuye considerablemente tendiendo a estabilizarse.

Debido a que los paquetes de control generados por AODVSD aumentan al incrementar las peticiones de servicio, así como éstos se van acercando a los generados por OLSRSD, surge la pregunta que si al incrementar las peticiones de servicio más allá de

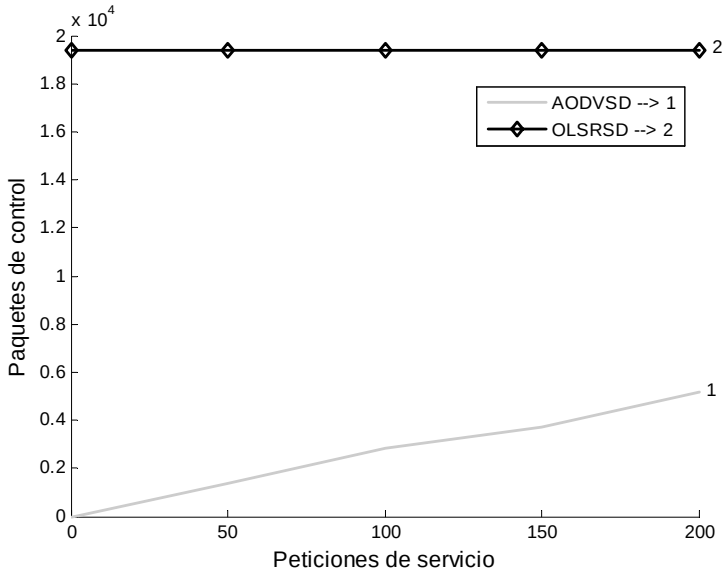

Figura 14: Comparativa peticiones de servicio VS paquetes de control para configuración de 100 nodos

200 se podría obtener un desempeño similar o mayor al obtenido por OLSRSD. De las figuras 15 y 16, se puede observar que para que se obtenga un desempeño similar o mayor, la cantidad de paquetes generados por AODVSD tendría que cambiar de orden de magnitud(de 10^3 a 10^4), es decir incrementar la cantidad de mensajes en un factor de 10. Con la finalidad de determinar si en algún momento AODVSD obtiene una cantidad de mensajes de control similar o mayor al de OLSRSD, se procedió a incrementar la cantidad de peticiones realizadas. Las figuras 17 y 18 muestran los resultados obtenidos al incrementar el número de peticiones realizadas a AODVSD

Las figuras 17 y 18 permiten observar que es posible que AODVSD genere una cantidad de paquetes de control similar o superior a OLSRSD, siempre y cuando se incrementen las peticiones de servicio considerablemente. Los resultados obtenidos utilizando la configuración de 50 nodos muestran que con 4000 peticiones AODVSD se obtiene practicamente la misma cantidad de paquetes de control que OLSRSD, en base

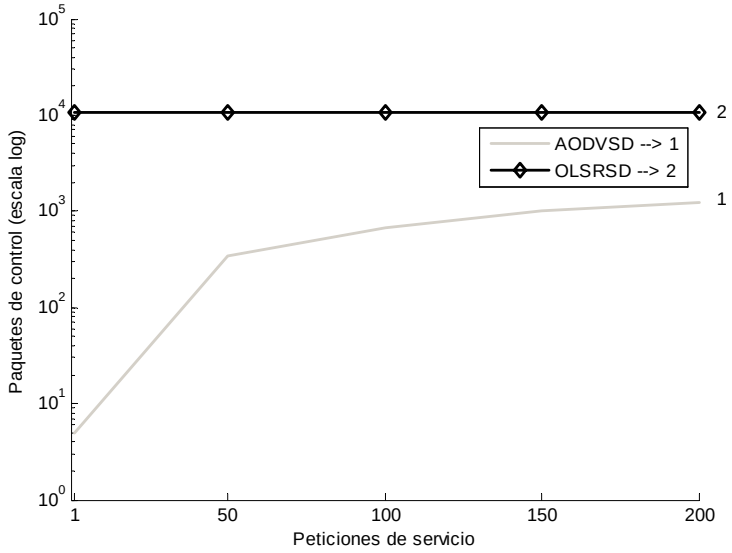

Figura 15: Comparativa peticiones de servicio VS paquetes de control para configuración de 100 nodos (escala semilogarítmica)

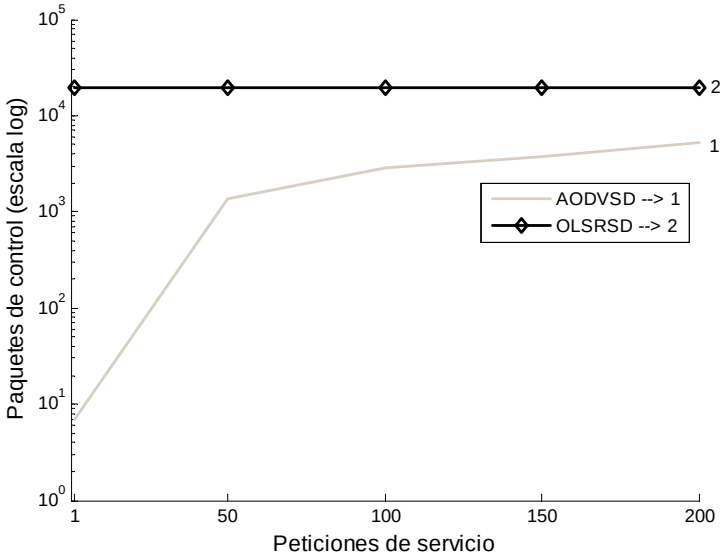

Figura 16: Comparativa peticiones de servicio VS paquetes de control para configuración de 100 nodos (escala semilogarítmica)

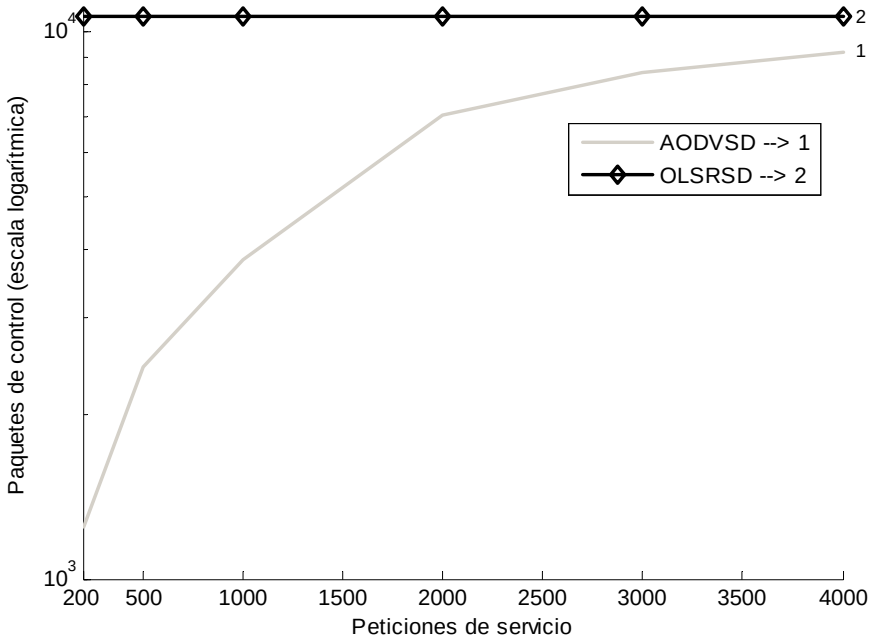

Figura 17: Comparativa peticiones de servicio Vs paquetes de control para configuración de 50 nodos (escala semilogarítmica) al incrementar peticiones

a lo anterior, se esperara que al incrementar un poco más las peticiones en dicho experimento, se generaría una cantidad similar o superior de mensajes. Por otra parte, los resultados obtenidos utilizando la configuración de 100 nodos muestran que con 1000 peticiones se alcanzan el número de mensajes de control generados por OLSRSD.

En este experimento se concluye que AODVSD permite obtener un mejor desempeño que OLSRSD, siempre y cuando el número de peticiones de servicios sea relativamente pequeño. Hay que tener en cuenta que si se realizan una cantidad muy grande de peticiones, la cantidad de mensajes generados por AODVSD podría ser mucho más alto que el obtenido por OLSRSD, por lo que el número de peticiones a realizar es un parámetro para decidir que protocolo se ajusta más a las necesidades de determinada aplicación. Sin embargo, hasta este momento no se sabe cual es el impacto en el número

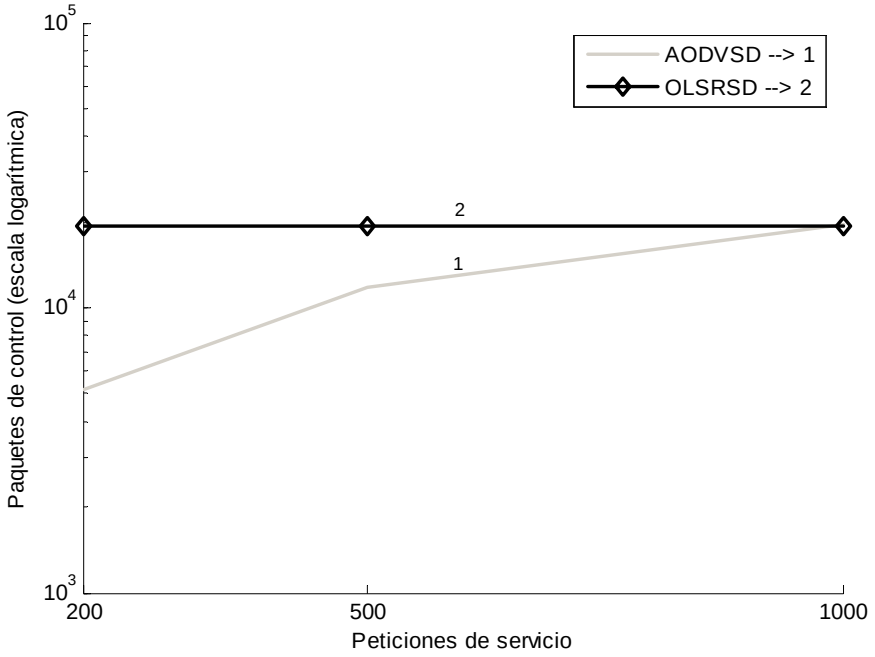

Figura 18: Sobrecosto por paquetes de control para configuración de 100 nodos (escala semilogarítmica) al incrementar peticiones

de servicios localizados, el siguiente experimento esta enfocado a determinarlo.

Experimento 2: Peticiones de servicios VS servicios localizados

El objetivo de este experimento fue determinar el número de servicios localizados por los protocolos OLSRSD y AODVSD. Para este experimento se realizaron 50, 100, 150 y 200 peticiones mientras se incrementa la redundancia de servicios (cantidad de servicios disponibles de cada tipo), es decir, si la redundancia es de 1, entonces sólo existe un proveedor para cada tipo de servicio disponible.

Los parámetros utilizados para este experimento se muestran en la tabla V:

Las figuras 19, 20, 21 y 22 representan los resultados para una redundancia de servicios 1, 2, 3 y 4 respectivamente, después de 50, 100, 150 y 200 peticiones

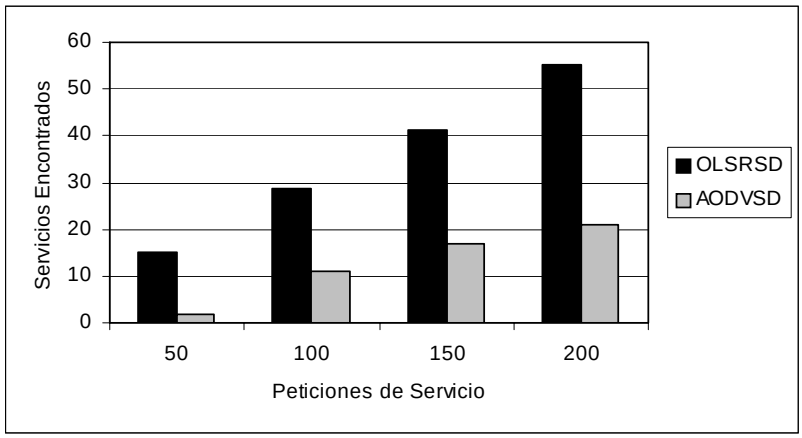

Figura 19: Comparativa de peticiones de servicios VS servicios localizados (Redundancia 1)

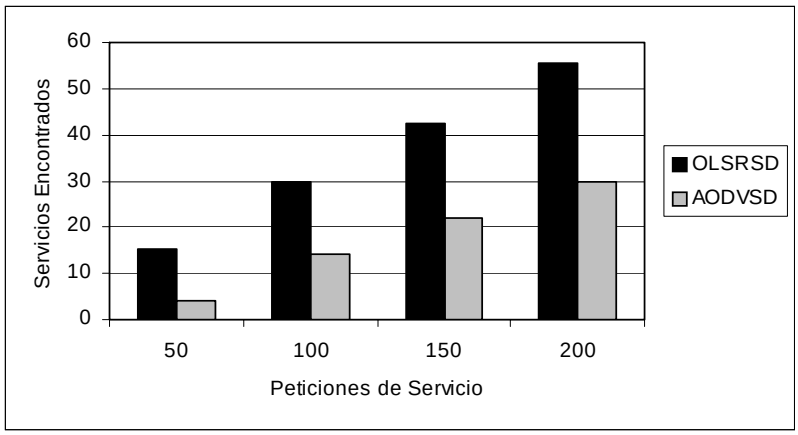

Figura 20: Comparativa de peticiones de servicios VS servicios localizados (Redundancia 2)

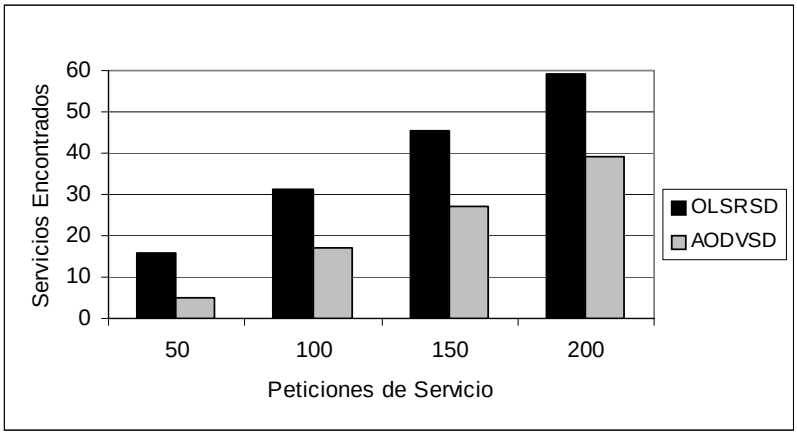

Figura 21: Comparativa de peticiones de servicios VS servicios localizados (Redundancia 3)

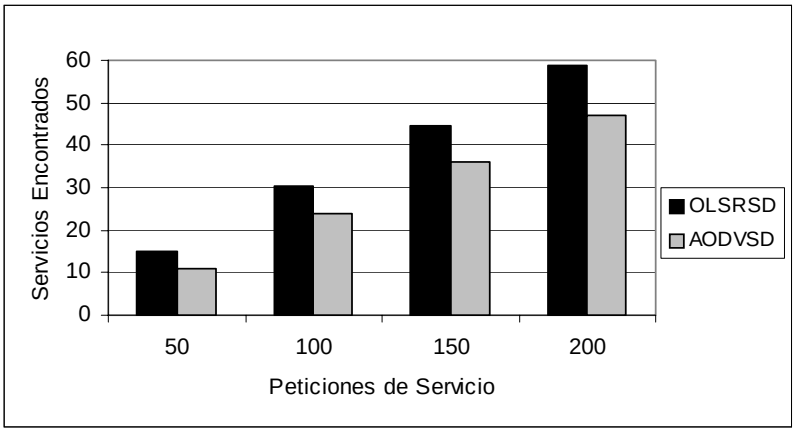

Figura 22: Comparativa de peticiones de servicios VS servicios localizados (Redundancia 4)

Tabla V: Parámetros empleados en experimento 2 para una red de 50 nodos

Parámetros	Valor
Número de nodos	50
Tipos de servicios	5
Área geográfica	1000 m^2
Tiempo de simulación	600 seg
Modelo movilidad	Random waypoint
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5
Conexiones CBR	5
Tamaño paquete CBR	512 bytes
Transmisión de paquetes CBR	0.25 seg

Análisis: El experimento permite observar que el número de servicios localizados por OLSRSD es superior a los obtenidos por AODVSD. Además, se puede visualizar que al incrementar la redundancia de servicios, el número de servicios localizados por AODVSD incrementa, a diferencia a los encontrados por OLSRSD, los cuales se mantienen prácticamente igual.

Al igual que en el experimento 1, con la finalidad de observar los resultados con otra perspectiva, se realizó la representación gráfica utilizando escala semilogarítmica con dirección en el eje Y.

La figura 23 muestra el desempeño obtenido por servicios localizados utilizando AODVSD, ésta permite observar que a medida que se incrementa la redundancia de servicios utilizando AODVSD, la cantidad de servicios encontrados se incrementa. Es decir, al incrementar la redundancia se localizan más servicios. Por otra parte, se visualiza que la forma de las curvas de crecimiento son relativamente similares.

Análisis: En la figura 24 se visualiza el desempeño obtenido por servicios localizados utilizando OLSRSD. Como se observa, el incrementar la redundancia de servicios

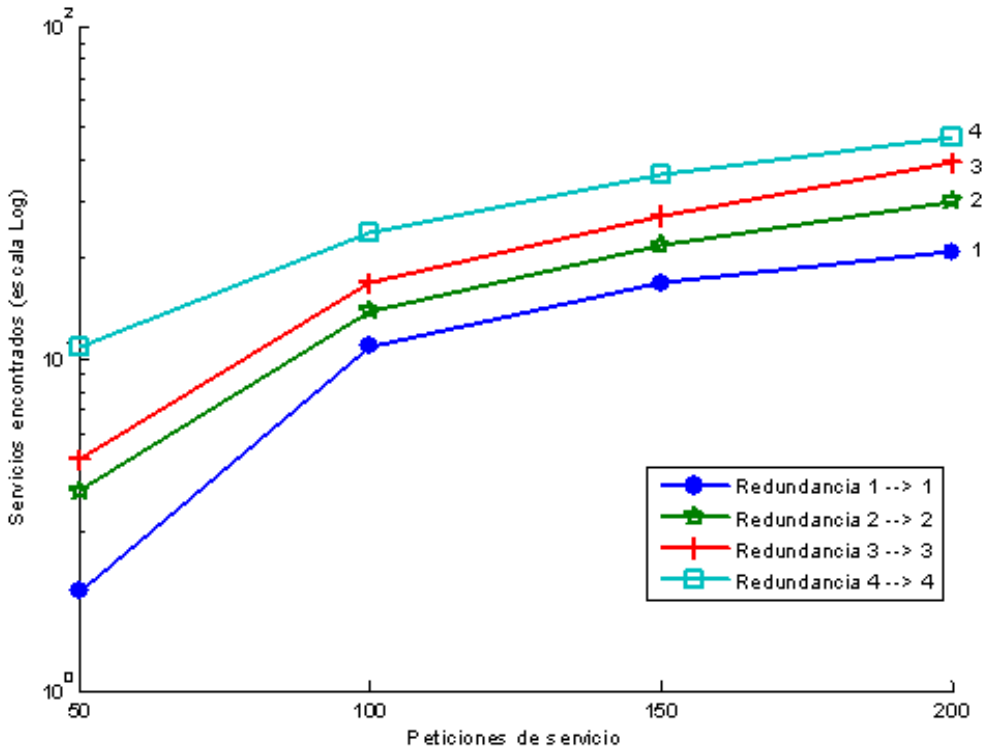

Figura 23: Comparativa de peticiones de servicio por servicios encontrados con AODVSD (escala semilog)

con este protocolo produce prácticamente el mismo comportamiento, tanto en el incremento de crecimiento como en el orden de magnitud, ya que las curvas que se producen convergen a los mismos valores.

Para finalizar el análisis de este experimento, a continuación se brinda una gráfica conjunta de las dos gráficas anteriores

Análisis: En la figura 25, se observa que el incremento de crecimiento y de orden de magnitud generado por AODVSD con redundancia 4, converge al obtenido por el protocolo OLSRSD con redundancia 1,2,3,4. Dicho comportamiento permite concluir que el contar con redundancia 4, es más que suficiente para casi igualar la cantidad

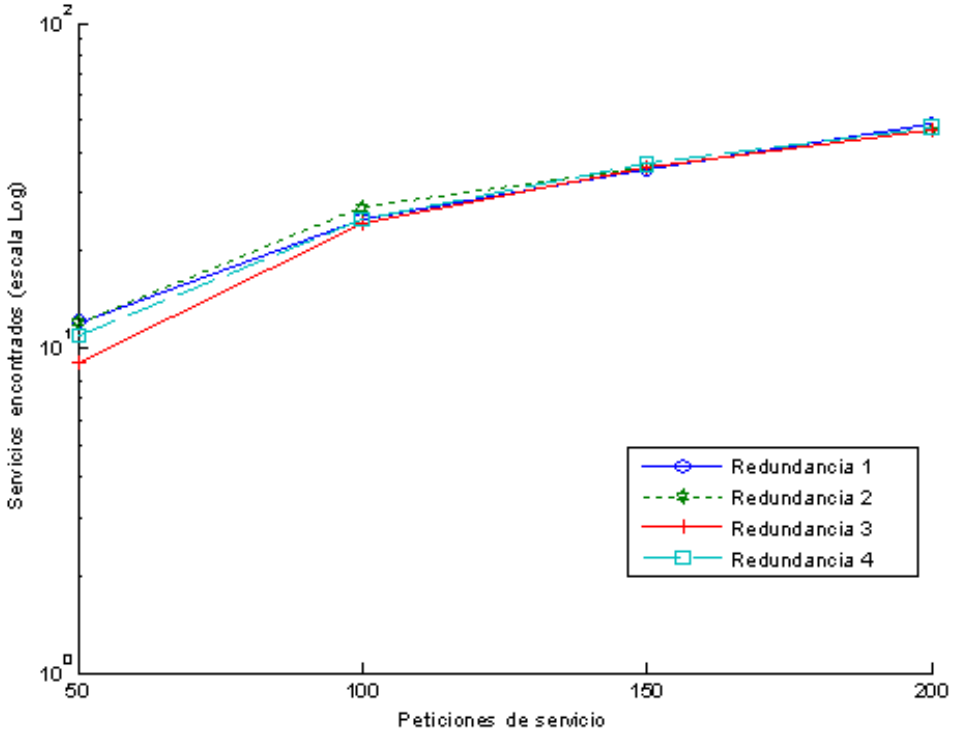

Figura 24: Comparativa de peticiones de servicios VS servicios encontrados con OLSRSD (escala semilog)

de servicios localizados por OLSRSD. Además, se concluye que entre más se incrementa las redundancia en AODVSD se logran descubrir mayor cantidad de servicios. Los resultados obtenidos indican que si se incrementa más la redundancia se lograría descubrir mayor cantidad de servicios con AODVSD que con OLSRSD. El siguiente experimento esta enfocado a determinar lo anterior.

Experimento 3: Redundancia de servicios VS servicios localizados

Este experimento es continuación del experimento anterior, y tiene como objetivo determinar si AODVSD descubre mayor cantidad de servicios al incrementar la

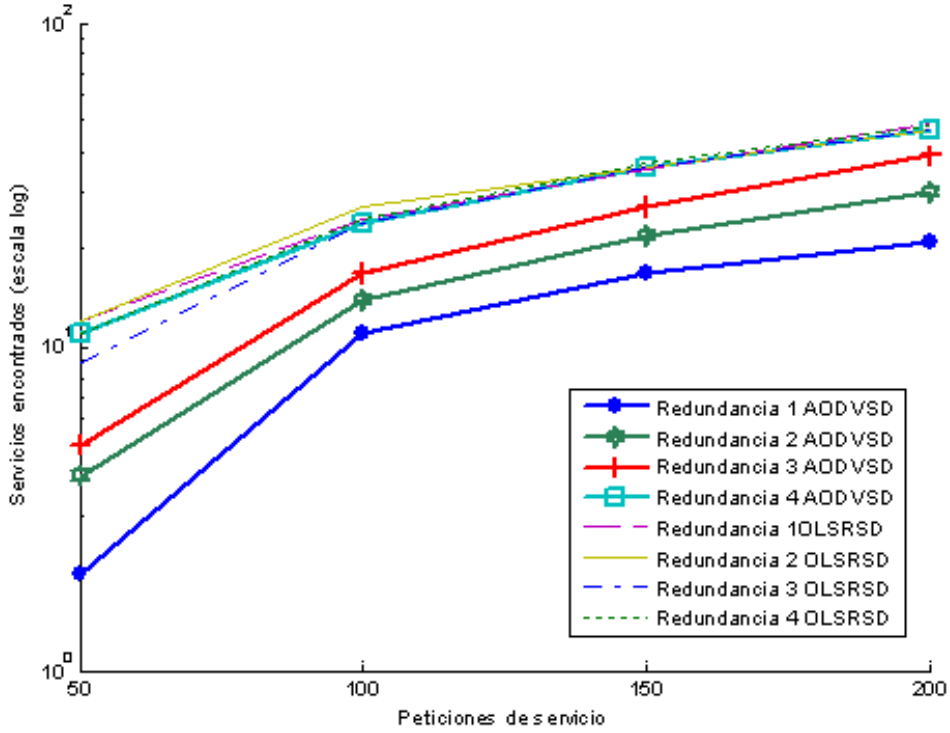

Figura 25: Sobrecosto de AODVSD vs OLSRSD por servicios encontrados (escala semilog)

redundancia más allá de cuatro, así como ver cuál es el efecto que presenta OLSRSD.

Los parámetros utilizados para realizar este experimento se muestran en la tabla VI

Con la finalidad de realizar este experimento, se realizó el estudio hasta tener redundancia 10. La representación gráfica de los resultados obtenidos por AODVSD se muestra en la figura 26

Análisis: La figura 26, indica que entre más peticiones y mayor redundancia de servicios se tenga, mayor cantidad de servicios serán localizados. Como se puede observar, el tener redundancia 10 iguala y en un caso supera a la obtenida con redundancia

Tabla VI: Parámetros empleados en experimento 3 para una red de 50 nodos

Parámetros	Valor
Número de nodos	50
Tipos de servicios	5
Área geográfica	1000 m^2
Tiempo de simulación	600 seg
Modelo movilidad	Random waypoint
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5

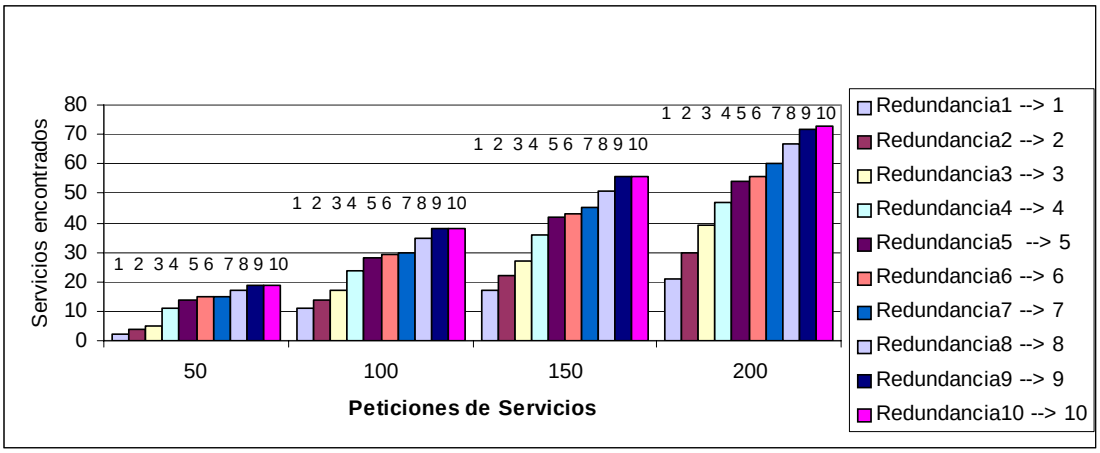

Figura 26: Comparativa redundancia de servicios VS servicios localizados (AODVSD)

9, por lo cual el tener redundancia 10 genera el mejor desempeño de AODVSD, lo anterior para los experimentos llevados a cabo.

La representación gráfica de los resultados obtenidos utilizando OLSRSD se muestra en la figura 27

Análisis: La figura 27 indica que al incrementar la redundancia de servicios utilizando el protocolo OLSRSD, no necesariamente se localizan mayor cantidad de servicios. Como se observa, el comportamiento obtenido entre cada nivel de redundancia no es

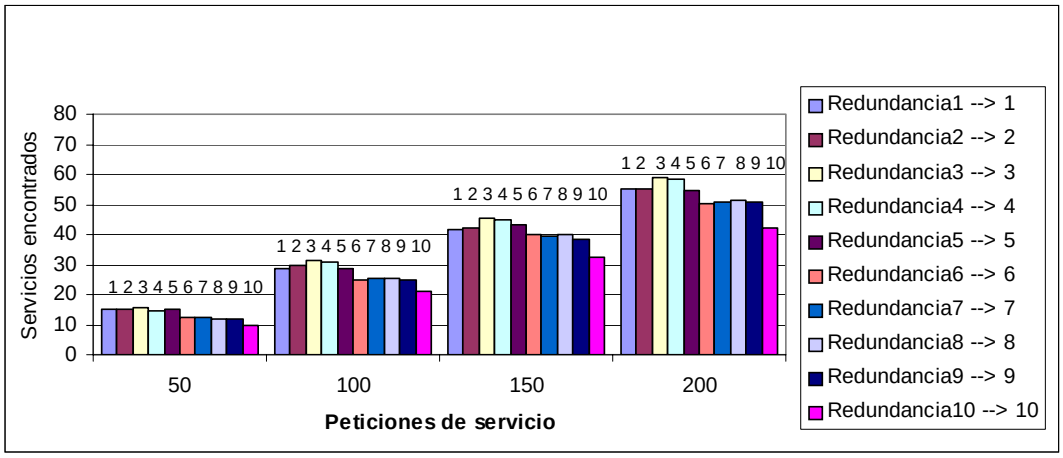

Figura 27: Comparativa redundancia de servicios VS servicios localizados (OLSRSD)

muy estable, ya que como pueden aumentar los servicios encontrados, éstos pueden disminuir. Los resultados anteriores llamaron especial atención, ya que se esperaría que al incrementar la redundancia se localizaran mayor o igual cantidad de servicios entre los niveles, lo cual no se refleja en los resultados obtenidos. Con la finalidad de determinar el comportamiento de estos resultados, se analizó la implementación del protocolo OLSRSD. Posteriormente de realizar el análisis, se determinó que los parámetros siguientes, y los valores asignados por omisión a estos, afectan directamente los resultados obtenidos relativos a la eficiencia de búsqueda.

- INFLTIME: Cantidad de tiempo que un proveedor brinda el servicio (255 segundos).
- SBINDLTIME: Cantidad de tiempo que el servicio se mantiene registrado en las tablas de servicios de los nodos que no son proveedores, en los cuales se registró el servicio por la inundación de paquetes (10 segundos).

- SDISEXTIME: Cantidad de tiempo de búsqueda del servicio en la tabla de servicios del nodo solicitante (5 segundos).

Dado lo anterior, se realizaron varios experimentos, en los cuales se modificaron cada uno de los parámetros anteriores. A continuación se resumen los resultados obtenidos:

El primer parámetro modificado fue el tiempo de vida en que un proveedor ofrece el servicio (INFLTIME), se estableció que el servicio sea ofrecido durante toda la simulación (600 segundos), ya que este inicialmente era de 255 segundos, por lo que el servicio se dejaba de ofrecer durante la simulación. Cabe mencionar que los demás parámetros no fueron modificados hasta este punto. Los resultados obtenidos al aumentar el tiempo en que el servicio es ofrecido por un proveedor reflejan mayor cantidad de servicios localizados que el experimento anterior, sin embargo se obtiene un comportamiento prácticamente similar en cuanto el aumento y disminución de servicios localizados entre los niveles. La representación gráfica de dichos resultados se muestra en la figura 28

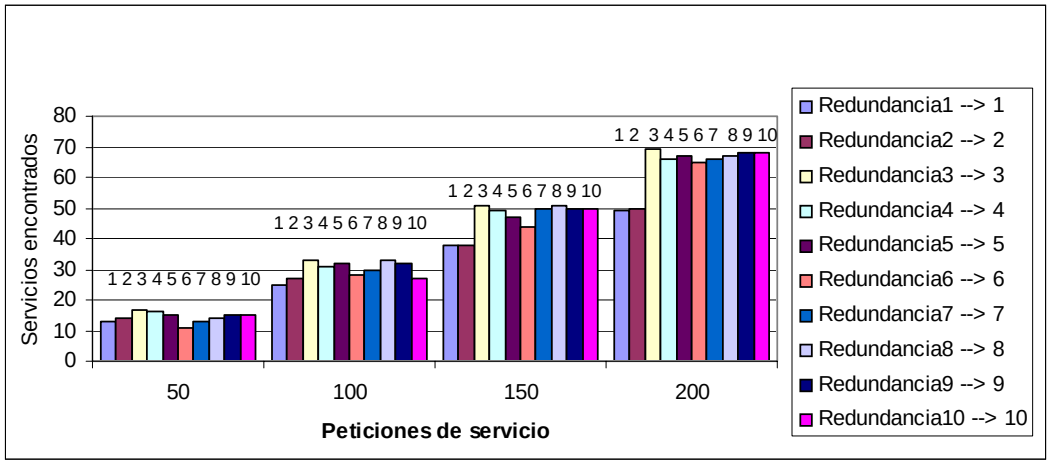

Figura 28: Comparativa redundancia de servicios VS servicios localizados (OLSRSD) 600 seg

Debido a los resultados anteriores se procedió a modificar el tiempo de búsqueda en las tablas de servicios de los nodos locales y el tiempo de registro de los servicios en dichos nodos (SDISEXTIME y SBINDLTIME respectivamente). Los efectos experimentados al de incrementar el valor de los parámetros anteriores, hacen que al ir aumentando la redundancia de servicios se localicen mayor o igual cantidad de servicios, éstos se muestran en la figura 29.

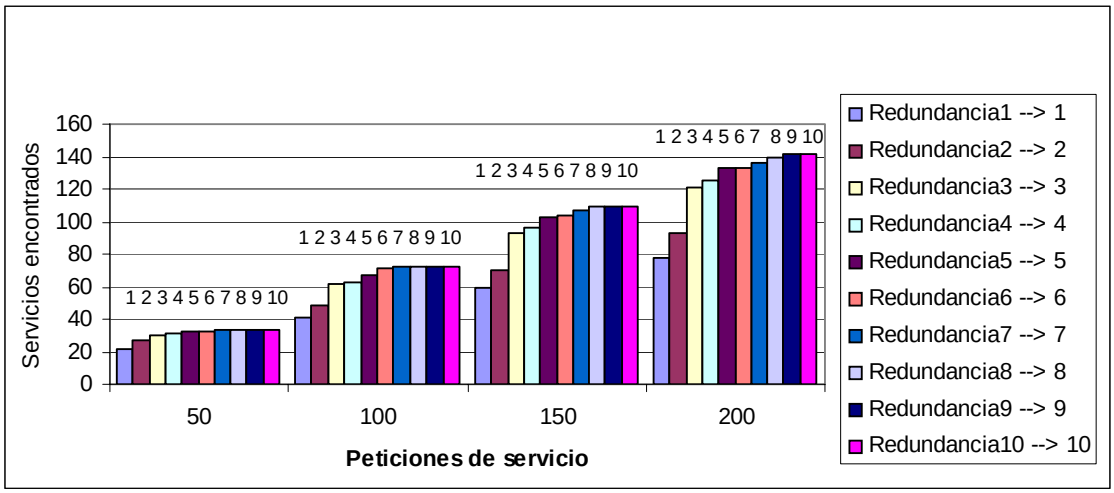

Figura 29: Comparativa redundancia de servicios VS servicios localizados (OLSRSD) tiempos de vida modificados.

El análisis realizado al protocolo OLSRSD y los resultados de la figura 27 permiten determinar que el comportamiento experimentado se debe a los tiempos de vida mencionados anteriormente, así como al congestionamiento que se introduce en la red al incrementar la cantidad de servicios de un mismo tipo. Debido a que OLSRSD está enfocado a redes ad hoc móviles, no es conveniente el incrementar los parámetros SDISEXTIME y SBINDLTIME debido a la movilidad constante e impredecible de los nodos, a diferencia del parámetro INFLTIME el cual si es conveniente que se ajuste

al tiempo de la simulación. Cabe mencionar que al realizar el ajuste mencionado anteriormente también se obtuvo que OLSRSD obtiene la mayor cantidad de servicios cuando se utiliza redundancia 3.

En este experimento se concluye que AODVSD puede localizar mayor cantidad de servicios que OLSRSD cuando se tiene redundancia mayor a 4. Además se sabe que con redundancia 3 empleando OLSRSD se obtiene la mayor cantidad de servicios.

Experimento 4: Redundancia de servicios VS tiempo de adquisición

El objetivo de este experimento fue determinar el tiempo de adquisición de servicios generado por los protocolos OLSRSD y AODVSD. Se realizaron 50, 100, 150 y 200 peticiones de servicios, utilizando 4 niveles de redundancia.

Los parámetros utilizados en este experimento se muestran en la tabla VII.

Tabla VII: Parámetros empleados en experimento 4 para una red de 50 nodos

Parámetros	Valor
Número de nodos	50
Tipos de servicios	5
Área geográfica	1000 m^2
Tiempo de simulación	600 seg
Modelo movilidad	Random waypoint
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5

A continuación se muestra el tiempo de adquisición promedio obtenido

Análisis: Como se observa en la figura 30, el tiempo de adquisición generado por AODVSD disminuye al incrementar el nivel de redundancia, mientras el protocolo OLSRSD muestra que el tiempo de adquisición es nulo para cualquier nivel de redundancia. Lo anterior se debe a que al solicitar un servicio en OLSRSD, dicho servicio

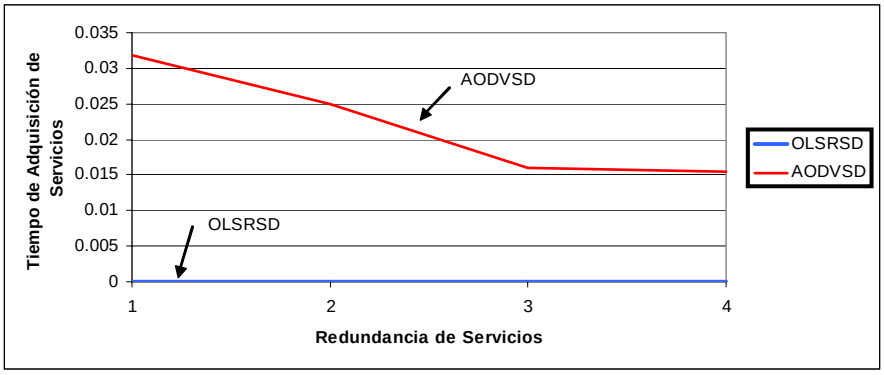

Figura 30: Comparativa de Redundancia de servicios VS Tiempo de adquisición

se busca en la tabla de servicios local, si este existe en dicha tabla, entonces se ha descubierto, de lo contrario no existe.

V.3.2. Comparativa con el método PULL y VSD

Este experimento tiene como objetivo comparar la cantidad de paquetes de control generados por las soluciones para el descubrimiento de servicios de nivel capa de red OLSRSD y AODVSD en comparación con el obtenido por métodos de nivel aplicativo. Existen diferentes propuestas a nivel aplicativo para el descubrimiento de servicios, cada una está enfocada a atacar ciertos objetivos y diseñada para arquitecturas en específico. Algunas de las propuestas existentes utilizan un esquema PULL, un esquema PUSH, o bien un esquema híbrido para atacar el descubrimiento de servicios. El realizar un estudio comparativo con todas las propuestas existentes es algo difícil, por lo cual es recomendable comparar mediante los esquemas mencionados anteriormente. En un esquema PULL, las peticiones se realizan sobre demanda, por lo que es menos costoso en cuanto a generación de mensajes que el método PUSH, el cual anuncia periódicamente servicios. Para llevar acabo este experimento, se hará uso de un estudio definido en

Kim (2005), en el cual se compara el método PULL con una propuesta definida en el mismo trabajo llamada VSD, la cual logra obtener mejor desempeño que el método PULL.

La siguiente figura muestra resultados obtenidos en el trabajo Kim (2005)

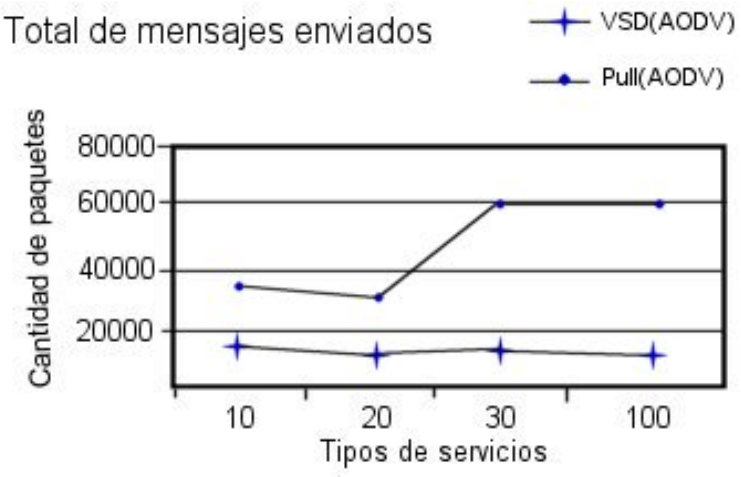

Figura 31: Sobrecosto por paquetes de control en capa de aplicación

Análisis: La figura 31 representa el número de mensajes generados por la solución VSD y por el método PULL, ambos utilizando AODV y DSDV como protocolos de capa de red. En el experimento se realizaron 600 peticiones de servicio durante 700 segundos de tiempo real, contando con 10 , 20, 40 y 100 tipos de servicios, la cantidad de servicios de cada tipo está dada por la cantidad de nodos en la red entre la cantidad de tipos de servicios existentes. Cabe mencionar que cada uno de los servicios se asigna a cada nodo utilizando asignación round robin.

Con la finalidad de realizar la comparativa entre capas, es necesario utilizar el ambiente y parámetros de simulación utilizados el trabajo mencionado, por lo que en este experimento varía un poco el ambiente y parámetros de simulación al de los

experimentos previos que en este documento se han presentado. Cabe mencionar que para este estudio se simulará la red teniendo solamente 10 tipos de servicios, por lo que en la red sólo existirán 5 proveedores de un mismo tipo de servicio. Por restricciones de equipo, simulaciones con mayor cantidad de servicios no pudieron ser realizadas, pero dadas las características y comportamiento de los protocolos a comparar, se espera obtener resultados similares.

Tabla VIII: Parámetros empleados en experimento 5

Parámetros	Valor
Número de nodos	50
Tipos de Servicios	10
Proveedores de un tipo de servicio	5
Tiempo de simulación	700 seg
Área geográfica	1000 m^2
Modelo movilidad	Random waypoint
Peticiones	600
Espera entre movimientos	2 seg
Velocidad máxima	2 m/s
Escenarios de movilidad	5

Se procedió a determinar el número de mensajes de control generados por los protocolos AODVSD y OLSRSD bajo los parámetros anteriores. La figura 32 muestra los resultados obtenidos

Análisis: En las figuras 31 y 32 se observa que el método PULL utilizando AODV y DSDV y contando con 10 tipos de servicios, genera cerca de 40000 mensajes control, a diferencia de AODVSD que genera cerca de los 3000 paquetes de control. Además, se observa que OLSRSD genera cerca de 10000 mensajes de control. Por los experimentos previos se sabe que OLSR genera muchos más mensajes de control que AODV, por lo que de lo anterior concluimos que tanto AODVSD como OLSRSD son mejores

Figura 32: Paquetes de control generados por los protocolos AODVSD y OLSRSD (600 peticiones)

alternativas para el descubrimiento de servicios que los métodos que utilizan esquemas PULL en lo que se refiere a cantidad de mensajes de control generados. Por otra parte, se observa que aunque la propuesta VSD es mejor que el método PULL; AODVSD y OLSRSD superan el desempeño en lo que se refiere a mensajes de control generados.

V.4. Conclusiones

En este capítulo, se realizó un estudio de simulación para evaluar el desempeño que genera aprovechar el descubrimiento de rutas para descubrir servicios, en lugar de realizar estas dos actividades por separado. Se comparan las soluciones AODVSD y OLSRSD entre si, con la finalidad de ver las ventajas y desventajas de utilizar un protocolo de tipo reactivo o uno proactivo como base para soportar descubrimiento de servicios. Además, se compara AODVSD y OLSRSD con soluciones que separan ambas

actividades, en específico con el protocolo VSD y el esquema pull, este último utilizado por varios protocolos tradicionales para descubrimiento de servicios. Los resultados obtenidos indican que el combinar el descubrimiento de rutas con el de servicios mejora el desempeño obtenido, ya que se logra obtener un menor número de mensajes de control, por lo que representan una mejor alternativa para el descubrimiento de servicios. Por otra parte, OLSRSD genera más mensajes de control que AODVSD cuando el número de peticiones es relativamente chico, logrando obtener una mayor cantidad de servicios hasta redundancia de servicios 4 en menor tiempo. Sin embargo, AODVSD logra superar los servicios descubiertos al incrementar la redundancia más allá de 4.

Capítulo VI

Conclusiones

En este capítulo se presentan las conclusiones de este trabajo de investigación, además se presentan las aportaciones y trabajo a futuro

VI.1. Conclusiones

- La cantidad de paquetes de control generados por AODVSD dependen del número de peticiones realizadas, es decir, la cantidad de paquetes incrementa al aumentar el número de peticiones, lo anterior se debe a que las peticiones de servicio se realizan sobredemanda.
- La cantidad de paquetes de control generados por OLSRSD son independientes al número de peticiones realizadas, es decir, la cantidad de mensajes siempre será constante durante una simulación sin importar si se realizan peticiones o no, lo anterior se debe a que sin importar las peticiones, periódicamente se anuncian los servicios existentes en la red.
- Si no se realizan peticiones de servicio, AODVSD no genera paquetes de control, mientras OLSRSD si los genera.
- El estudio de simulación realizado a los protocolos AODVSD y OLSRSD indica que OLSRSD puede obtener más rápido los servicios entre par de nodos.

- El incrementar la redundancia de servicios en AODVSD aumenta el número de servicios encontrados para los tamaños de red empleados. Con redundancia 10 se descubren la mayor cantidad de servicios. Se presume que para otros tamaños el comportamiento sea el mismo.

- El incrementar la redundancia de servicios utilizando OLSRSD no mejora necesariamente el rendimiento para localizar mayor cantidad de servicios. Con redundancia 3 se descubren la mayor cantidad de servicios utilizando este protocolo.

- Se determinó que el tiempo en que un servicio se anuncia, se mantiene registrado y se busca, son parámetros que afectan directamente los resultados obtenidos relativos a la eficiencia de búsqueda.

- OLSRSD permite localizar mayor cantidad de servicios que AODVSD hasta un nivel de redundancia 4 para los tamaños de red empleados y los valores asignados por omisión a los tiempos de vida de los servicios en las tablas de servicios, así como el tiempo de busqueda en las tablas.

- AODVSD logra localizar mayor cantidad de servicios que OLSRSD después de un nivel de redundancia 4 para los tamaño de red empleados.

- Si se realiza un número muy grande de peticiones al protocolo AODVSD, éste podría generar mayor cantidad de mensajes de control que OLSRSD, por lo que el uso de AODVSD es ideal en casos donde el número de peticiones no es muy grande, aunque hay que tener en cuenta los efectos en la cantidad de servicios localizados para decidir que protocolo es el que se ajusta más a las necesidades.

- El uso de OLSRSD es ideal en casos donde el número de peticiones es muy grande, aunque también hay que tener en cuenta los efectos en la cantidad de servicios localizados.

- Tanto los protocolos reactivos como proactivos son una alternativa para combinar el descubrimiento de rutas con el de servicios.
- Los protocolos AODVSD y OLSRSD logran mejorar el desempeño de protocolos aplicativos que utilizan el esquema PULL para descubrir servicios. Además, logran mejorar el desempeño de una nueva propuesta llamada VSD.

VI.2. Aportaciones

- En este trabajo de investigación se brindó una solución al descubrimiento de servicios para redes ad hoc (OLSRSD), la cual aprovecha el proceso de descubrimiento de rutas para descubrir servicios. Dicha solución está basada en el protocolo de enrutamiento proactivo OLSR y logra minimizar la cantidad de paquetes involucrados en el proceso de descubrimiento, lo que permite hacer un mejor uso de los recursos en la red en comparación a soluciones tradicionales para el descubrimiento de servicios, las cuales realizan el descubrimiento de servicios en capa de aplicación. El combinar ambas actividades permite ahorrar en energía y en espacio de almacenamiento, así como minimizar el posible congestionamiento en la red.
- Se llevó a cabo un estudio de simulación con la finalidad de estudiar los efectos de combinar el descubrimiento de rutas con el descubrimiento de servicios, utilizando tanto un protocolo proactivo (OLSRSD) como uno reactivo (AODVSD), lo que permitió ver las ventajas y desventajas de utilizar protocolos de cierto tipo.
- Con los estudios realizados se brinda un nuevo paradigma para el descubrimiento de servicios en redes ad hoc móviles — el combinar el descubrimiento de rutas con el descubrimiento de servicios —.

VI.3. Trabajo Futuro

El presente trabajo de investigación abre las siguientes líneas de investigación

- La solución propuesta en este trabajo de tesis y la solución propuesta en Arias (2004), logran descubrir los servicios existentes en la red, sin embargo no se definen algún mecanismo para poder hacer uso de los servicios una vez que han sido descubiertos, por lo que es necesario brindar a estos protocolos un mecanismo que resuelva lo anterior.

- En el estudio de simulación realizado para comparar la solución propuesta en este trabajo con el esquema PULL y VSD, sólo se simulan 10 tipos de servicios en la red, el estudio podría extenderse a 20, 40 y 100 tipos de servicios.

- Actualmente diversos grupos de trabajos están realizando contribuciones al descubrimiento de servicios, con la finalidad de garantizar que diversas soluciones puedan operar conjuntamente es necesario definir una representación para el nombrado de servicios, de manera que cualquier solución pueda localizar los servicios existentes en la red.

- Actualmente están surgiendo soluciones a nivel aplicativo para el descubrimiento de servicios que utilizan multicast en la comunicación, por lo que se propone realizar un estudio de simulación utilizando protocolos de enrutamiento de tipo multicast, así como comparar este nuevo enfoque con el realizado en este trabajo de tesis.

Bibliografía

Arias, D. 2004. "Descubrimiento de servicios en redes ad hoc móviles". Tesis de Maestría, CICESE, Ensenada, México. 121 pp.

Baker, F. 2002. "An outsider's view of manet draft-baker-manet-review". Internet draft. URL. Consultado en: http://w3.antd.nist.gov/wctg/manet/draft-baker-manet-review-01.txt, Noviembre, 2004. Cisco Systems. Network Working Group.

Basagni, S. Bruno R y Petriolli C. 2004. "Scatternet formation in bluetooth networks". En: Basagni, Conti M, Giordano S y Stojmenovic I. "Mobile Ad Hoc Networking". IEEE Press, EUA. 117-119 p.

Beijar, N. 2002. "Zone routing protocol (zrp)". Networking Laboratory, Helsinki University of Technology. Helsinki, Finlandia. Consultado en: http://www.netlab.hut.fi/opetus/s38030/k02/Papers/08-Nicklas.pdf, Diciembre, 2004.

Broustis, I. Paterakis, M. 2004. "On the feasibility of integrated mpeg teleconference and data transmission, over ieee 802.11 wlans". NETWORKING 2004, Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications: Third International IFIP-TC6 Networking Conference, Atenas, Grecia, Mayo 9-14, 2004. Proceedings. Springer-Verlag Heidelberg. Chania, Grecia. 638-649 p.

Bruno, R. Conti, M. y Gregori E. 2001. "Wireless access to internet via ieee 802.11: An optimal control strategy for minimizing the energy consumption". En: Proceedings of the Thyrrhenian International Workshop on Digital Communications: Evolutionary Trends of the Internet IWDC 2001, Taormina, Italia. Springer-Verlag Heidelberg. Septiembre 17-20, 2001. Pisa, Italia. 120-138 p.

Campo, C. Perea, J. Marín, A. y García, C. 2005. "Pdp and gsdl: A new service discovery middleware to support spontaneous interactions in pervasive systems". In IEEE Middleware Support for Pervasive Computing (PerWare 2005) at the 3rd IEEE Conference on Pervasive Computing (PerCom 2005). IEEE Press. Hawaii, Marzo 8 - 12, 2005. 178-182 p.

Chakraborty, D. Joshi, A. 2002. "Gsd: A novel group-based service discovery protocol for manets". 4th IEEE Conference on Mobile and Wireless Communications Networks (MWCN 2002). Septiembre 9-11, 2002. Estocolmo. Suecia.

Chen, L. 2002. "Service advertisement and discovery in mobile ad hoc networks". En: Workshop on Ad Hoc Communications and Collaboration 104 in Ubiquitous Computing Environments, Noviembre 16-20, 2002. Lehigh University, Belen.

Chlamtac, I. Conti, M. Liu, J. 2003. "Mobile ad hoc networking: imperatives and challenges". Elsevier Ad Hoc Networks Journal. School of Engineering, University of Texas at Dallas, EUA. 1(1):13-64 p.

Clausen, T. y Jacquet, P. 2003. "Optimized link state routing protocol (olsr)". IETF. RFC 3626. Consultado en: http://www.rfc-editor.org/rfc/rfc3626.txt, Marzo, 2005.

Dabrowski, C. y Mills, K. 2001. "Analyzing properties and behavior of service discovery protocols using an architecture-based approach". En: Proceedings of Working Conference on Complex and Dynamic Systems Architectures. Diciembre, 2001. Brisbania, Australia.

Dimakis, A. Linhai, H. Musacchio, J. Hoi-Sheung, W. Tung, T. y Walrand, J. 2003. "Adaptive quality of service for a mobile ad hoc network". The Fifth IEEE Conference on Mobile and Wireless Communications Networks, Octubre, 2003. Singapur.

Goland, Y. Cai, T. Leach, P. Gu, Y. y Albright, S. 1999. "Simple service discovery protocol". IETF. Consultado en: http://www.ietf.org/internetdrafts/draft-cai-ssdp-v1-03.txt, Enero, 2005.

Gryazin, E. 2000. "Service discovery in bluetooth". Artículo, CiteSeer, Scientific Literature Digital Library. Group for Robotics and Virtual Reality,Department of Computer Science.Helsinki University of Technology, Finlandia. Consultado en: http://citeseer.nj.nec.com/392311.html, Enero,2005.

Guo, H. Ingelrest, F. Simplot-Ryl, D. y Stojmenovic, I. 2005. "Performance evaluation of broadcasting protocols for ad hoc and sensor networks". En: Proc. 4th IFIP Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET 2005). Junio, 2005. Ile de Porquerolles, Francia.

Guttman, E. Perkins, C. Veizades, J. y Day, M. 1999. "Service location protocol version 2". IETF. RFC 2608. Consultado en: http://www.rfc-editor.org/rfc/rfc2165.txt, Enero, 2005.

Haartsen, J. 2000. "Short-range connectivity with bluetooth". Interactive Distributed Multimedia Systems and Telecommunication Services: 7th International Workshop, Springer-Verlag Heidelberg. Octubre, 2000. Enschede, Países Bajos.

Helal, S. Desai, N. Verma, V. y Lee, C. 2003. "Konark – a service discovery and delivery protocol for ad-hoc networks". En: Proceedings of the Third IEEE Conference on Wireless Communication Networks (WCNC). Marzo, 2003. Nuevo Orleans, EUA.

Jun-Zhao, S. 2001. "Mobile ad hoc networking: An essential technology". En: Proceedings of IEEE ICII 2001. Octuber - Noviembre, 2001. Beijing, China. 3-4 p.

Kim, M. Kumar, M. y Shirazi, B. 2005. "Service discovery using volunteer nodes in overlapped clusters for pervasive computing environments". IEEE International Conference on Pervasive Services 2005 (ICPS'05). Julio, 2005. Santorini, Grecia.

Lee, C. y Helal, S. 2002. "Protocols for service discovery in dynamic and mobile networks". International Journal of Computer Research. 11:1-12 p.

Liu, J. Sohraby, K. Zhang, Q. Li, B. y Zhu, W. 2003. "Resource discovery in mobile ad hoc networks source". En: Mohammad, I. "The handbook of ad hoc wireless networks". CRC Press, EUA. 26:1-11 pp.

Marin-Perianu, R. Hartel, H. y Scholten, J. 2005. "A classification of service discovery protocols". Reporte técnico , Centre for Telematics and Information Technology, Univ. of Twente, Países Bajos. 22 pp.

Melodia, T. y Cuomo, F. 2003. "Locally optimal scatternet topologies for bluetooth ad hoc networks". En: Battiti, R. LO CIGNO, R. y CONTI M. "Wireless On-Demand Networks Systems". Lecture Notes in Computer Science. Springer. Roma, Italia. 2928:116-129 p.

Navarro, C. 2002. "Arquitectura de cómputo ubicuo para interactuar con personas, recursos y dispositivos". Tesis de Maestría, CICESE, Ensenada, México. 105 pp.

Rendon, A. 2002. "Ad hoc networks - design and performance issues". Tesis de Maestría, Helsinki University of Technology, Finlandia. 121pp.

Romero, R. 2003. "Transmisión de voz sobre redes de paquetes y parámetros de calidad de servicio (qos)". Tesis. UABC. Ensenada. México.

Sun, M. 1999. "Jini technology architectural overview". Consultado en: http://www.sun.com/jini/whitepapers/architecture.html, Enero, 2005. 35-38 p.

Toh, C. 2002. "Ad hoc mobile wireless networks: Protocols and systems". Prentice Hall, Nueva Jersey, EAU. 480 pp.

Tonnesen, A. 2004. "Implementing and extending the optimized link state routing protocol". Tesis de Maestría. University of Oslo. Noruega. 11-23 pp.

UC, B. 2002. "The ns manual". Consultado en: http://www.isi.edu/nsnam/ns/nsdoc.pdf, Diciembre, 2004.

Veizades, J. Guttman, E. y Perkins, C. 1997. "Service location protocol". IETF. RFC 2165. Consultado en: http://www.rfc-editor.org/rfc/rfc2165.txt, Enero, 2005.

Vidal, I. García, C, Soto, I. Moreno, J. 2003. "Servicios de valor añadido en redes móviles ad hoc". Jornadas Telecom I+D 2003. Departamento de Ingeniería Telemática. Noviembre, 2003., Universidad Carlos III de Madrid, España. 1-18 p.

Willekens, J. 2001. "Ad hoc routing in bluetooth". 6th International Conference, PROMS 2001, Springer-Verlag Heidelberg. Octubre 17 - 19, 2001. Enschede, Países Bajos.

Zheng, Y. 2005. "Security in ad hoc networks". Consultado en: http://citeseer.ist.psu.edu/536945.html, Noviembre, 2004. Networking Laboratory, Helsinki University of Technology.

Zhu, F. Mutka, M. Ni, L. 2002. "Classification of service discovery in pervasive computing environments". Reporte Técnico. MSU-CSE-02-24, Michigan State University. EUA. 1-17 pp.

Apéndice A

Introducción

En este apéndice se brinda información en términos generales del simulador de redes NS (Network Simulator), el cual fue utilizado para poder realizar el estudio comparativo entre soluciones de descubrimiento de servicios. Además, se describen los pasos necesarios para la instalación de dicho simulador, así como de las soluciones que fueron sujetas de estudio.

A.1. NS

NS es un simulador extremadamente popular en el área científica para el estudio de redes, el cual ha sido utilizado como una herramienta para experimentar nuevas ideas, protocolos y algoritmos (UC, 2002).

El simulador de redes NS incorpora una amplia gama de protocolos (Ethernet 802.11, ARP, DSDV, AODV, TORA, IP, TCP, UDP, FTP, TELNET, entre otros) para poder caracterizar diferentes tipos de arquitecturas, ya sean cableadas o inalámbricas (terrestres y satelitales).

NS inició como una variante del simulador de redes REAL en 1989 y ha evolucionado substancialmente durante los últimos años. En 1995 el desarrollo de NS estuvo a cargo por DARPA a través del proyecto VINT de LBL, Xerox PARC, UCB, y USC/ISI. Actualmente, el desarrollo de NS esta bajo responsabilidad de DARPA y NSF, aunque

NS cuenta con contribuciones sustanciales de otros investigadores, incluyendo código para redes inalámbricas de los proyectos CMU Monach y UCB Daedelus y también de Sun Microsystems (UC, 2002).

El simulador de redes NS, actualmente en la versión 2 (NS2), es de dominio público, por lo que puede ser descargado y utilizado sin ningún costo. Además, es de código abierto, es decir, el programa puede ser modificado libremente, lo cual permite la integración de nuevas características, así como resolver comportamientos inadecuados.

NS es un simulador orientado a objetos escrito en C++ y OTcl (Object Tool Command Language). Cabe mencionar que mediante el uso de los dos lenguajes anteriores, se puede ampliar la funcionalidad del simulador, pero dependiendo lo que se requiera hacer, será más apropiado usar uno o el otro. Utilizando C++ se puede manipular el procesamiento de paquetes, modificar tanto encabezados como el comportamiento de módulos, adaptar protocolos a necesidades, entre otros. Mientras OTcl, es utilizado para controlar simulaciones mediante la escritura de scripts de simulación, además se pueden modificar objetos C++ sin necesidad de crear nuevos.

La figura 33 muestra una vista general de NS.

Como se observa en la figura anterior, el simulador NS toma como entrada un script escrito en OTcl, el cual establece cada una de las características de la red que se desea simular (nodos, enlaces, protocolos, etc). Dicho script es interpretado y ejecutado por el interprete OTcl y por las bibliotecas NS (Calendarizador de eventos, bibliotecas de componentes de red y de módulos para el establecimiento de la red).

Al finalizar la simulación, se puede ver lo sucedido en la simulación mediante el análisis de los archivos de salida .tr y/o .nam. El archivo con extensión .tr consta

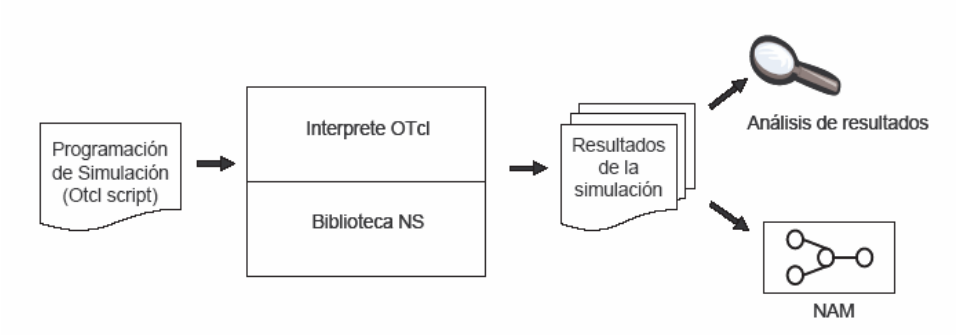

Figura 33: Vista General de Ns

de líneas de texto que describen cada uno de los paquetes que se generaron en la simulación, por otra parte el archivo con extensión .nam puede ser utilizado por la herramienta NAM (Network Animator) para visualizar gráficamente lo sucedido. Cabe mencionar que para que se genere uno o ambos archivos es necesario especificarlo en el script.

A.1.1. Instalación de NS2

Existen diferentes versiones disponibles de NS2 para varias plataformas (Linux, Windows y Mac), aunque en este trabajo se utilizó la versión 2.26 para una arquitectura tipo Linux, por que se recomienda utilizar el archivo ns-allinone-2.26.tar.gz, el cual puede ser obtenido desde el sitio oficial de NS2 en http://www.isi.edu/nsnam/ns/.

Después de descargar el archivo mencionado anteriormente, se deberá seguir las siguientes instrucciones

1. Descomprimir el archivo en ruta deseada

   a) tar -xzvf ns-allinone-2.26.tar.gz

2. Instalar simulador

   a) cd ns-allinone-2.26

   b) ./install

3. Establecer variables de ambiente, tal como muestra la salida del paso 2

4. Validar protocolos con los que cuenta NS2

   a) cd ns-2.26

   b) ./validate

A.1.2. Agregar AODVSD a NS2

Para agregar AODV con capacidades de descubrimiento de servicios (AODVSD), es necesario reemplazar los archivos existentes en el directorio aodv que se encuentra en la raíz ns (ej: /root/ns-allinone-2.26/ns-2.26) por los contenidos en el directorio AODVSD existente en el disco de tesis. Una vez realizado lo anterior, se procede a ejecutar los siguientes comandos dentro del directorio de la raíz ns

1. make clean

2. make

Con lo anterior ya se puede comenzar a utilizar el protocolo AODVSD.

A.1.3. Agregar OLSRSD a NS2

Para que el simulador NS2 pueda hacer uso del protocolo OLSRSD, primero es necesario instalar OLSR y posteriormente agregar las extensiones de descubrimiento de servicios. A continuación se brindan las instrucciones necesarias

1. Copiar el archivo de instalación de OLSR del disco de tesis en la raíz ns2.
2. Descomprimir el archivo de instalación de OLSR.
   a) tar -xzvf nrlolsr.tar.gz
3. Acceder y seguir las instrucciones del archivo readme contenido en el directorio nrlolsr que se acaba de crear al ejecutar el paso anterior.
4. Asegurarse de compilar el simulador ejecutando los comandos
   a) Make clean
   b) Make install

Hasta este momento se cuenta con OLSR instalado y listo para usarse, ahora es necesario agregar las extensiones de descubrimiento de servicio, para lo que es necesario remplazar los archivos existentes en el directorio nrlolsr de la raíz ns2 por los contenidos en el directorio del mismo nombre, pero del disco de tesis. Posteriormente, volver a ejecutar el paso 4. Con lo anterior ya se puede comenzar a utilizar el protocolo OLSRSD.

A.1.4. Creación de Escenarios de Movilidad

El simulador de redes Ns2 brinda una herramienta (setdest) para generar escenarios de movilidad, es decir, permite generar las posiciones de los nodos, su velocidad de movimiento y dirección a seguir. Esta herramienta utiliza el modelo de movilidad random waypoint.

A continuación se muestra la sintaxis de la herramienta setdest

setdest [-n num_de_nodos] [-p tiempopausa] [-s velocidadmáxima] [-t tiemposimulación] [-x maxx] [-y maxy] > [nombrearchivosalida]

Si se deseara crear un escenario de movimientos llamado escenario1 de 200 nodos con pausa de 2 s con máxima, velocidad de 2 m/s en un área de 1000m x 1000m durante 600 se deberá ejecutar

setdest -n 200 -p 2.0 -s 2.0 -t 600 -x 1000 -y 1000 > escenario1

Cabe mencionar que dicha herramienta se encuentra en ns/ns-allinone-2.26/ns-2.26/indep-utils/cmu-scen-gen/setdest/

Apéndice B

Acrónimos

AODV: Ad hoc On Demand Distance Vector.

AODVSD: AODV Service Discovery.

API: Application Programming Interface.

DAML: DARPA Agent Markup Language.

HTTP: Hyper Text Transport Protocol.

MANET: Mobile Ad Hoc Network.

MID: Multiple Interface Declaration

MPR: Multipoint Relay.

NAM: Network Animator.

NS: Network Simulator.

NS2: Network Simulator Version 2.

OLSR: Optimized Link State Routing.

OLSRSD: OLSR Service Discovery.

OTCL: Object Tool Command Language.

PDA: Personal Digital Assistant.

PDP: Pervasive Discovery Protocol.

QoS: Quality of Service.

RFC: Request For Comments.

RMI: Remote Method Invocation.

SDP: Service Discovery Protocol.

SLP: Service Location Protocol.

SOAP: Simple Object Access Protocol.

SSDP: Simple Service Discovery Protocol

TC: Topology Control