

Un soporte de comunicación grupal para agentes móviles

Guillermo Rigotti

UNICEN – Fac. de Ciencias Exactas-ISISTAN
Pje. Arroyo Seco, (7000) Tandil, Bs. As. Argentina
e-mail: grigotti@exa.unicen.edu.ar

Resumen

En los últimos años se ha producido un desarrollo significativo en el área de agentes móviles. Estos sistemas han demostrado su aptitud para un conjunto diverso de aplicaciones de gran uso en la Internet.

Si bien se han abordado con éxito diferentes aspectos de dichos sistemas, logrando soluciones concretas, se ha dejado de lado el aspecto de la comunicación, referido al transporte de la información a intercambiar entre los agentes móviles. Los mecanismos utilizados se han heredado principalmente de la teoría de sistemas distribuidos, no adaptándose completamente a las características de la comunicación entre agentes móviles. Es así que gran parte de los sistemas en uso cubren redes locales o intranets, o bien soportan sólo una cantidad limitada de agentes. Un problema aún abierto a la investigación lo constituye el hecho de encontrar un soporte de comunicación y localización adaptable a sistemas de agentes móviles que involucran gran cantidad de agentes dispersos en áreas extensas de la Internet. Se han realizado intentos de implementar este soporte utilizando transmisión multicast, sin embargo, cuando éste es implementado a nivel de red, presenta limitaciones en cuanto a su despliegue en la totalidad de la red y en cuanto al servicio ofrecido, en particular a los agentes móviles.

En este trabajo se propone una infraestructura de comunicación para agentes móviles, que está basada en un soporte de transmisión multicast, implementado en los servidores de agentes, que permite satisfacer las demandas de las aplicaciones a un costo razonable y con una performance aceptable

Palabras clave: Agentes móviles, Multicast.

Abstract

In the last years a significant development has taken place in the area of mobile agents. These systems have demonstrated their aptitude for a diverse group of applications of great use in the Internet. Although different aspects of this systems have been approached with success, achieving concrete solutions, it has been left aside the aspect of the communication, related to the transport of the information to be exchanged among the mobile agents. The used mechanisms have been inherited mainly of the theory of distributed systems, not adapting completely to the communications requirements posed by mobile agents. Due to this problem, this kind of applications are either restricted to be deployed in intranets or to support only a limited number of agents. A problem even open to the investigation is to develop a support for communication among and localization of mobile agents, that involves a great number of agents moving in vast areas of the Internet. Some approaches have been carried out aimed to implement this kind of support using the existent infrastructure for multicast transmission. However, because it is implemented at network level, it presents limitations in several aspects, mainly its deployment all over the Internet and the service offered to the mobile agents. In this work we propose a communication infrastructure for mobile agents that is based on a multicast support implemented in the agent servers- This support is able to satisfy the demands from the applications to a reasonable cost and with an acceptable performance

Key Words: Mobile agents, Multicast.

1. Introducción

Las aplicaciones de agentes móviles que involucran áreas extensas de la Internet y están a su vez constituidas por gran cantidad de agentes, presentan graves problemas de escalabilidad aún no resueltos, respecto a la localización de y a la comunicación entre agentes.

En [7] se han identificado los requerimientos más importantes que deben ser satisfechos por la infraestructura de comunicación en este tipo de aplicaciones. Estos requerimientos se enumeran a continuación, especificando la manera en que pueden ser satisfechos por una implementación del soporte multicast a nivel aplicación, cuya ejecución está a cargo de los servidores de agentes móviles.

1-Cobertura de amplias áreas en la Internet: la infraestructura de comunicación multicast debe cubrir el área de acción de los sistemas de agentes móviles. Esta es la totalidad de la Internet o gran parte de ella, dependiendo de cada aplicación y de la distribución de los servidores de agentes móviles. En la actualidad, el soporte multicast a nivel de red está restringido al área cubierta por el MBONE, requiriendo modificaciones en los routers en el caso de que se necesite extenderla. En nuestra propuesta, las funciones multicast se asignan a los servidores de agentes, lo cual permite desplegar rápidamente y con costos mínimos el soporte en aquellos lugares de la red de interés para la aplicación; sólo es necesario extender la funcionalidad de los servidores incorporando un módulo multicast. Este tipo de soporte multicast a nivel aplicación, cuando es utilizado en redes peer to peer, presenta el problema de la imprevisibilidad de los equipos de los usuarios [4], que pueden ser apagados sin previo aviso, dando lugar a costosos procedimientos de recuperación del árbol de distribución, con la consiguiente carga en la red y deterioro de la calidad del servicio. En nuestro caso, este soporte está implementado en los servidores de agentes, que pueden considerarse equipos tan seguros como los routers que implementan multicast a nivel de red.

2- Tipo de árbol de distribución adaptado al tráfico generado por las aplicaciones: el árbol de distribución multicast es el medio a través del cual los paquetes se distribuyen a los receptores, y está compuesto de nodos (raíz, intermedios y hojas) y vínculos de transmisión, por los cuales fluye la información dirigida al grupo. El tipo de árbol utilizado influye de manera significativa en la performance de las aplicaciones y en la carga generada en la red, y debe ser considerado teniendo en cuenta las características del tráfico producido por las aplicaciones. Han sido definidos varios tipos de árboles, cada uno de ellos con características que responden mejor a cierto tipo de tráfico. Para una clasificación de árboles de distribución, referirse a [6].

La implementación de multicast a nivel aplicación presenta la ventaja de poder construir el tipo de árbol de distribución más adecuado para el tráfico generado, en lugar de tener que utilizar el implementado a nivel de red, común a todos los usuarios y con objetivos propios del nivel de red. En el caso de agentes móviles, debe tenerse en cuenta, para seleccionar el tipo de árbol a utilizar, el tráfico producido por los agentes, la cantidad de posibles emisores, y la dinamicidad del grupo. De acuerdo con el uso esperado de la comunicación multicast en este caso (funciones de management y notificación de ciertos eventos relevantes por parte de los agentes al grupo), la comunicación será esporádica, de poco volumen¹, y originada en múltiples puntos (potencialmente cualquier agente puede enviar información). Esto da lugar a la necesidad de un árbol de distribución multicast bidireccional compartido con raíz en el servidor de origen de la aplicación. La alternativa de árboles no compartidos (uno por cada emisor), produciría una carga no admisible en relación al volumen de datos a enviar, debido a la cantidad de emisores y a los mecanismos de construcción de los árboles. Además, al trasladarse un agente de un nodo a otro, sería necesario construir un nuevo árbol de distribución con raíz en el nuevo servidor. Esto indica que debería utilizarse un árbol compartido, es decir, un único árbol que transporte información de todos los emisores. Esta alternativa reduce significativamente la carga, ya que se tendrá un único árbol, con raíz en el nodo desde el que se lanza la aplicación, y cuyas hojas irán cambiando a medida que los agentes se desplacen de servidor.

3-Adaptabilidad a grupos altamente dinámicos: Los grupos generados por este tipo de aplicaciones son altamente dinámicos, dependiendo de la movilidad de los agentes. La migración de un agente, da lugar generalmente a una solicitud de integración del grupo (join) por parte del nuevo servidor y un abandono del grupo (leave) por parte del anterior. La gran dinamicidad de los agentes móviles plantea dos problemas a considerar. Por un lado, la carga introducida en la red por los procesos de pertenencia y abandono del grupo debe ser mínima. Por otra parte, estos procesos deben llevarse a cabo en un tiempo adecuado al de permanencia del agente en el nuevo servidor para que no se pierda información. Este último aspecto ha sido tratado en [5], donde la movilidad es considerada una fuente adicional de pérdida de información. En nuestra propuesta se plantea un esquema de construcción del árbol de distribución basado en el recorrido del agente en lugar de la topología de la red. Este proceso, si bien menos eficiente en cuanto a la conformación del árbol de distribución, evita la pérdida de paquetes y disminuye las demoras en el proceso de pertenencia². Como alternativa a este método propuesto, presentamos una heurística

¹ Precisamente el objetivo de los agentes móviles es evitar el intercambio de grandes volúmenes de información, trasladándose el proceso al lugar donde residen los datos.

² Considerando el tiempo total necesario para que un agente que ha migrado a un nuevo servidor comience a recibir información grupal en modo confiable.

simple que mejora sustancialmente las características del árbol de distribución obtenido a través de la explotación de un conocimiento parcial de la topología de la red en el proceso de construcción de dicho árbol.

4- Confiabilidad. Los usos más comunes de la transmisión grupal en sistemas de agentes móviles, son el envío de mensajes de management con el objeto de que la aplicación pueda controlar a los agentes, el envío de información entre los agentes para coordinar su operación, por ejemplo, un agente indica que ha cumplido con su objetivo, y mensajes para la localización de un agente en particular, a fin de iniciar una conexión con él³. En general, esta clase de comunicación requiere transmisión confiable. El problema de ofrecer transmisión multicast confiable no está aún resuelto. Cuando se utiliza soporte multicast a nivel de red, el agregado de confiabilidad se realiza agregando un nivel adicional, situado sobre el de red, denominado multicast confiable. Este campo está sujeto a investigación, no habiendo resultados satisfactorios que permitan pensar en implementaciones a nivel global, y menos en tipos de tráfico como el que nos ocupa. Dos de los protocolos más destacados son Selective Reliable Multicast (SRM) y Tree Based Reliable Multicast Protocol (TRAM), que han sido diseñados para casos particulares de tipos de comunicación, con diferentes características a las necesarias en el caso de agentes móviles.. Aunque existiera un soporte adecuado en el nivel de multicast confiable, el agregado de este nuevo nivel produciría tiempos de respuesta no compatibles con los requerimientos de estas aplicaciones y un consumo de recursos inadmisibles. En nuestra propuesta se propone la implementación del árbol de distribución utilizando conexiones TCP entre los nodos (conexiones entre padres e hijos), lo que permite ofrecer confiabilidad de manera simple, integrando la construcción del árbol de distribución con el soporte para confiabilidad..El uso de cadenas de conexiones TCP débilmente relacionadas se propone en [2], pero considerando la adaptación de las mismas a múltiples receptores heterogéneos; en nuestro caso, este problema no es relevante debido a la baja tasa de transmisión generada por la aplicación. Como mejora, en nuestro caso estas conexiones TCP pueden ser utilizadas para soportar canales pertenecientes a varios árboles de distribución y adicionalmente para intercambio de información de control entre los servidores.

5- Seguridad: La transmisión multicast a nivel de red presenta graves problemas de seguridad. Cualquier equipo puede escuchar las transmisiones, y lo que es aún más grave, generar paquetes para el grupo, produciendo como mínimo la saturación de los recursos asignados⁴.La incorporación de un sistema de comunicación grupal no debe representar una disminución en la seguridad. En nuestra propuesta, contrariamente a lo que ocurre en un sistema multicast a nivel de red, el control de acceso al árbol multicast está manejado por los servidores de agentes, que de acuerdo a la aplicación, pueden implementar las medidas de seguridad adecuadas en cada caso.

6- Cobro por uso de recursos: esta característica es de importancia ya que el mantenimiento de un nodo del árbol multicast implica consumo de recursos por parte de los servidores. La dificultad en cargar a los usuarios por el uso de facilidades multicast a nivel de red ha sido un factor que ha demorado su ofrecimiento por parte de los ISPs. En nuestro caso, el cargo por uso de facilidades multicast se realiza con los mismos mecanismos (si los hay) que los utilizados para cargar a la aplicación por el uso de otros recursos propios de la aplicación en el servidor correspondiente.

7-Flexibilidad en el procesamiento de paquetes: En algunas aplicaciones se requiere un tratamiento especial de los paquetes por parte de los routers, ya sea para mejorar el uso de los recursos o para adaptarse a los requerimientos de la aplicación. Esto no puede ser satisfecho con un soporte multicast a nivel de red, ya que debería agregarse en los routers funcionalidad especial de acuerdo a cada caso. En nuestra propuesta, los servidores de agentes pueden someter a los paquetes a los procesos que resulten necesarios, según los requerimientos de cada aplicación. Un caso típico lo constituye la sumaria de asentimientos en soportes de multicast confiable con asentimientos explícitos.

En resumen, las características más importantes de nuestra propuesta son las siguientes:

- Implementación del soporte de comunicación multicast en los hosts, particularmente en los servidores de agentes móviles involucrados en la aplicación.
- Construcción de un árbol de distribución multicast compartido y bidireccional, con raíz en el servidor en el cual se ha iniciado la aplicación.
- Asignación de direcciones multicast únicas para cada aplicación, basadas en la dirección IP del servidor que actúa como raíz del árbol.

³ La localización de un agente es resuelta con un tipo de comunicación multicast. Un agente emite un requerimiento de localización multicast que solo será respondido (unicast) a la dirección del solicitante por el agente que está siendo buscado.

⁴ Si bien es posible autenticar emisores a nivel aplicación, con lo cual un receptor no aceptará paquetes provenientes de emisores no autorizados, los paquetes dirigidos al grupo, provenientes de equipos mal intencionados de todas maneras son introducidos en la red, provocando el consumo de recursos y el posible “denial of service”.

- Uso de protocolo TCP para mantener los vínculos entre nodos padres e hijos que conforman el árbol de distribución

2. Operación

A continuación se describe la operación de los servidores relacionada con la construcción y el mantenimiento del árbol de distribución, la asignación de direcciones multicast, y la transferencia de datos multicast entre los agentes.

2.1. Selección de la raíz del árbol de distribución y asignación de direcciones multicast

El proceso de construcción del árbol de distribución comienza cuando es iniciada una aplicación, activando el primero de los agentes móviles en el servidor seleccionado. Este servidor se convierte en la raíz del árbol de distribución, adquiriendo responsabilidad en la asignación de direcciones multicast a la aplicación y respecto al mantenimiento de dicho árbol.

Las direcciones multicast definidas para nuestro caso, no son las utilizadas en el soporte multicast standard a nivel de red (direcciones clase D en IP versión 4), sino que se componen de una parte inicial, que consiste en la dirección IP del servidor raíz, más una identificación única local a dicho servidor. De esta manera, las direcciones multicast asignadas por diferentes servidores difieren siempre entre sí, evitando el uso de los costosos protocolos de asignación global de direcciones multicast a nivel de red destinados a evitar colisión de direcciones. En nuestro caso, bastará con implementar un método local a cada servidor, que permita elegir sufijos locales diferentes para cada dirección multicast generada.

2.2 Generación periódica de información por parte del nodo raíz

El servidor origen de la aplicación, además de constituirse en raíz del correspondiente árbol de distribución, adquiere responsabilidad en el mantenimiento del mismo: es el encargado de emitir, a intervalos regulares, mensajes heartbeat (multicast) que se difundirán por el árbol de distribución. Un heartbeat, cuando es originado por el servidor raíz, sólo contiene su identificación y el número de nodos hijos que actualmente tiene. Cada servidor que lo reciba, antes de difundirlo a sus hijos, agregará en él su dirección, su distancia al servidor anterior⁵ y su número actual de servidores hijos. De esta manera, cualquier servidor conocerá su camino hacia la raíz del árbol. Esta información es utilizada 1-para casos de falla de nodos (lo que obliga a reparar el árbol), o 2-para determinar el servidor que conviene seleccionar como padre, en caso de que un nuevo servidor desee integrar el árbol de distribución⁶, o 3-para mejorar el árbol de distribución a través de la reconexión de nodos a servidores más adecuados.

2.3. Modificaciones en el árbol de distribución

Las modificaciones que pueden producirse en el árbol de distribución consideradas en este trabajo⁷ son 1-altas de nuevos nodos: cuando un agente móvil arriba a un servidor que no pertenece al árbol de distribución, 2-bajas de nodos ya pertenecientes al árbol: por partida del último agente o desconexión del último nodo hijo, y 3-cambios en la composición de dicho árbol con el objeto de mejorar su performance: cuando el nodo adquiera información adicional respecto a la topología del árbol, que le permita mejorar su conexión al mismo (por ejemplo arribo de un nuevo agente a un nodo ya conectado). Las operaciones a realizar difieren de las llevadas a cabo en multicast a nivel de red, debido al tipo de nodos que integran el árbol de distribución (hosts en lugar de routers), a los requerimientos de la aplicación, y a la información topológica utilizada.

Las altas de nuevos nodos, en casos de uso tradicional de facilidades multicast, se producen debido a que un proceso en un host de la red local solicita agregarse al grupo multicast para recibir la información correspondiente. Como consecuencia, el host notifica a su router local utilizando IGMP, y éste inicia el procedimiento de conexión al árbol de distribución. En este caso, la localización de un nodo ya perteneciente al árbol, a quien conectarse, se realiza en base a información topológica (generalmente derivada de las tablas de ruteo unicast), que corresponde a la métrica utilizada como objetivo a mejorar en la construcción del árbol⁸. Esto implica que no será necesario un ajuste del árbol (que este nuevo nodo cambie de padre, si es notificado de uno mejor) excepto en caso de fallas en nodos que hagan que se requiera una reconstrucción de la totalidad o parte del árbol.

⁵ Para determinar la distancia desde el servidor anterior hasta el que recibe el paquete, no es necesario ningún procedimiento especial, ya que se utiliza el contador de saltos provisto por IP (TTL).

⁶ En el caso del método de construcción del árbol que utiliza el conocimiento de la topología de la red.

⁷ Otras modificaciones a considerar en la continuación de este trabajo son las relativas a recuperación de la conectividad ante fallas en nodos o vínculos de transmisión.

⁸ Un nodo que requiere agregarse al árbol de distribución buscará el próximo nodo en dirección a la raíz, según su tabla de ruteo. Así recursivamente hasta que encuentra un nodo ya perteneciente al árbol de distribución, al cual se agrega como hijo

En el caso de agentes móviles, la incorporación de un nuevo servidor al árbol de distribución se produce como consecuencia de que un agente se traslada al servidor, y solicita recibir información dirigida al grupo. Nuestra solución propone una alternativa diferente respecto a la información en la cual basarse para construir el árbol de distribución: en lugar de utilizar información topológica como se describió, se utiliza información transportada por el agente, consistente en la lista de servidores que componen la rama del árbol desde la raíz hasta el último nodo visitado por el agente, derivada del último heartbeat recibido por dicho servidor. Esto implica que no se dispone de información topológica que permita construir un árbol eficiente (se sacrifica esta característica para obtener un agregado inmediato al árbol con transmisión confiable, prescindiendo de la existencia de un soporte multicast de red generalizado), y por lo tanto, se deberá mejorar el árbol durante la operación normal, basándose en nuevos agentes que soliciten residir en el nodo o en procedimientos periódicos adicionales.⁹

La desconexión de un servidor del árbol de distribución, se producirá cuando éste no tenga agentes locales ni servidores hijos para el grupo. Estas condiciones se pueden producir como consecuencia de la migración o destrucción de agentes locales y de modificaciones en el árbol de distribución debidas a bajas o cambio de padre por parte de otros servidores dependientes de este servidor. Estas condiciones son equivalentes a la no existencia de procesos interesados en recibir información del grupo en la red local y a la desconexión de los nodos dependientes del router en multicast a nivel de red.

En ambos casos, multicast a nivel de red y nuestra propuesta, el nodo (servidor) envía un mensaje solicitando desconexión (leave) a su padre.

2.4. Transmisión de información multicast

Como fue mencionado, el volumen de datos (en nuestro caso, multicast) a transferir en aplicaciones de agentes móviles es bajo, y fundamentalmente derivado de operaciones de management y notificaciones de un agente al grupo, pudiendo originarse en cualquiera de los agentes. Estas características determinan que el árbol de distribución deba ser bidireccional y compartido. Cuando un agente desea emitir un dato multicast, lo solicita a su servidor actual, quien lo difundirá por todas sus interfaces multicast (independientemente de si el interlocutor en la interfaz es padre o hijo). Para realizar esta difusión, utilizará conexiones TCP que ha establecido en el momento de agregarse al árbol (en su rol de hijo) o posteriormente cuando ha agregado a un servidor hijo (en su rol de padre). De esta manera, se tienen las bases para implementar un multicast confiable de manera simple. Además de difundirlo a los servidores adyacentes, el servidor entregará el mensaje a todos los agentes locales de la aplicación, salvo al originador del mensaje. Un servidor que recibe un paquete multicast originado en otro servidor, lo difundirá de manera similar.

Si bien la demora producida por el proceso de agregado al árbol de distribución (joining) por parte de un servidor es mínima, debe tenerse en cuenta que un agente puede perder información si es recibido un paquete durante el lapso de tiempo comprendido entre su partida de un servidor, y el momento en que el nuevo servidor concrete su pertenencia al árbol de distribución y comience efectivamente a recibir información multicast. Para evitar esta pérdida de datos, se prevé que el servidor del cual parte un agente almacene temporalmente la información dirigida a él (en particular la información multicast), para entregarla posteriormente al nuevo servidor que aloje al agente.

3. Una heurística simple para construir y mejorar el árbol de distribución

Si bien la alternativa descrita para agregarse al árbol de distribución se produce en tiempo mínimo, ya que se utiliza como nodo padre el servidor del cual proviene el agente y el nexo se establece utilizando la conexión TCP creada para el transporte del agente, el árbol de distribución producido resulta sumamente ineficiente (como puede apreciarse en los resultados obtenidos en la simulación), debido a que la relación padre-hijo entre los servidores se establece teniendo en cuenta sólo el recorrido de los agentes y no la topología de la red. A continuación se presenta una heurística simple utilizada para la construcción y eventual modificación del árbol de distribución, que conserva la característica de un agregado en un tiempo aceptable¹⁰ mejorando el árbol de distribución obtenido.

Esta heurística ha sido probada a través de simulación (ver sección 4) produciendo resultados satisfactorios. En la actualidad se está trabajando para ampliar su aplicación y mejorarla.

Cuando un agente arriba a un servidor que aún no forma parte del árbol de distribución, producirá el joining al grupo correspondiente por parte de este nuevo servidor. Como consecuencia de no utilizar un soporte multicast a nivel de red, el nuevo servidor debe basarse sólo en la información transportada por el agente para conectarse al

⁹ En la actualidad, la manera de mejorar el árbol de distribución se basa en la llegada de nuevos agentes. Se esta mejorando la heurística descrita a continuación para lograr, a costo razonable, que los nodos puedan mejorar su conexión al árbol en base a información provista por el nodo raíz a intervalos regulares, incluida en los mensajes heartbeat. Esta alternativa deberá ser evaluada en base al consumo de recursos que significa esta mejora y al tiempo que permanezcan los agentes en los servidores.

¹⁰ Debe remarcarse que este agregado al árbol de distribución incluye el equivalente al agregado a nivel de red y el establecimiento de las bases para un multicast confiable.

árbol. Esta información, en su versión más simple, consiste en una referencia al servidor inmediatamente anterior, del cual provino el agente. El nuevo servidor solicitará a éste último su agregado como hijo (join). Si bien esta alternativa produce un agregado rápido, sin necesidad de la invocación a otros mecanismos (por ejemplo a los mecanismos de multicast a nivel de red), es altamente ineficiente, ya que produce árboles de distribución con excesiva superposición de nodos y vínculos de transmisión, dependiendo la calidad de los mismos exclusivamente del recorrido que realicen los agentes.

El procedimiento que se describe a continuación tiene por objeto mejorar la calidad del árbol y acercarla a la obtenida con un protocolo multicast a nivel de red a costo razonable. Según los resultados de las simulaciones produce árboles aceptables mejorando sustancialmente a la alternativa anterior.

a-Dados dos nodos, a y b, definimos distancia de red entre ellos - $Dr(a,b)$ - como la cantidad mínima de nodos que es necesario recorrer para llegar de uno al otro. Notar que se supone redes simétricas, por lo tanto $Dr(a,b) = Dr(b,a)$.

b-Dados dos nodos a y b pertenecientes a un árbol de distribución, definimos la distancia multicast entre ellos - $Dm(a, b)$ - como la cantidad de nodos que debe visitarse, siguiendo el recorrido indicado por el árbol de distribución, para llegar desde a hasta b. De la misma manera se supone $Dm(a,b) = Dm(b,a)$.

c-Si el camino desde un nodo j de un árbol de distribución hasta otro nodo k del árbol está compuesto por los nodos intermedios x, y, z,

$$Dm(j, k) = Dr(j,x) + Dr(x,y) + Dr(y,z) + Dr(z,k) \quad \text{y en particular, al ser z padre de k,}$$

$$Dm(j,k) = Dm(j,z) + Dr(z,k)$$

d-En general, para cualquier par de nodos a y b del árbol, se cumplirá:

$Dm(a,b) \geq Dr(a,b)$, cumpliéndose la igualdad en el caso en que a y b tengan una relación padre-hijo, debido al método de construcción del árbol propuesto¹¹.

Suponiendo que el camino producido por el método de construcción del árbol entre el nodo raíz n_0 y un nodo miembro n_i está constituido por los nodos $n_0, n_1, \dots, n_{i-1}, n_i$, los heartbeats generados periódicamente por el nodo raíz, proporcionarán a todo nodo n_k , $0 < k \leq i$, la siguiente información:

-La distancia de red entre cada par de nodos padre-hijo, $Dr(n_r, n_{r+1})$ para todo r tal que $0 \leq r < k$

-El número de nodos hijos de cada nodo anterior, H_{n_r} , para todo r tal que $0 \leq r < k$

Por otra parte, suponemos que cualquier nodo (en particular n_i) puede determinar la distancia entre él y cualquier otro nodo de la red (en particular los integrantes de la rama del árbol) a través de mecanismos de nivel de red (por ejemplo ICMP echo requests y replys). Por lo tanto se conoce $Dr(n_i, n_k), \forall k, 0 < k < i$.

En base a la información anterior, un nodo n_i puede obtener las distancias multicast entre la raíz y sus predecesores n_k ,

$$Dm(n_0, n_k) = \sum_{z=0}^{k-1} Dr(n_z, n_{z+1})$$

De esta manera, un nodo k que ha recibido un agente y debe conectarse al árbol de distribución, seleccionará el padre más adecuado contando con la información proporcionada por el agente, que a su vez la ha obtenido del último servidor visitado, actualizada al último heartbeat recibido. El nodo padre n_p será seleccionado, según nuestra heurística, en base a tres aspectos :

1- la distancia multicast Dm_{n_p} que lo separara del nodo raíz del árbol en caso de utilizar como padre a n_p ,

$Dm_{n_p}(n_0, n_k) = Dm(n_0, n_p) + Dr(n_k, n_p)$. El hecho de hacer mínima esta distancia, tiende a que el nodo se conecte al árbol de distribución en un punto que no produzca demasiada carga debido a involucrar superposición de vínculos a nivel de red. Sin embargo, teniendo en cuenta solo la minimización de este parámetro, obtendríamos un árbol compuesto de tantas ramas nodo_raíz-nodo_miembro como miembros del árbol haya, ya que los nodos tenderían a conectarse directamente al nodo raíz. Como consecuencia, tendríamos sólo vínculos punto a punto¹² con la consiguiente sobrecarga del servidor que actúa como nodo raíz y sus adyacencias.

¹¹ La igualdad puede cumplirse en otros casos especiales, dependiendo de la ubicación de los nodos en la red.

¹² Sólo se limitaría la conexión al nodo raíz en función del peso asignado a la cantidad de hijos de un servidor.

2- la distancia de red entre todos los posibles padres y el nodo en cuestión. Hacer mínima esta distancia tiende a lograr árboles de distribución en los que los nodos adyacentes se encuentren cercanos entre sí, lo cual mejora la carga global. Teniendo en cuenta sólo la minimización de esta distancia, se obtendría un comportamiento similar al que tienen los protocolos multicast a nivel de red con pertenencia explícita, pero en este caso trabajando con un conjunto restringido de posibles nodos padre (solo los servidores de la rama a través de la cual arribó el agente).

3- la cantidad de nodos hijos que ya tiene un posible candidato a ser padre. Este factor se tuvo en cuenta para evitar una carga excesiva en los servidores, y así obtener un árbol con un número limitado de hijos por servidor. Si bien aquí se tuvo en cuenta sólo la cantidad de hijos, podría reemplazarse este valor por otros relacionados con el costo económico de establecer una bifurcación en un servidor, con motivos de seguridad, etc. En definitiva, este valor puede ser determinado por cada aplicación.

Según la heurística desarrollada, para un nodo Z , perteneciente al árbol de distribución con raíz en R , el costo C_{p_i} para cada potencial padre p_i , se calcula de la siguiente manera:

$$C_{p_i} = a * Dm_{p_i}(R, Z) + (1 - a) * Dr(p_i, Z) + b * H_{p_i},$$

donde Dm_{p_i} es la distancia multicast entre la raíz R y el nodo Z , si éste utiliza como padre a p_i , y H_{p_i} es el número de hijos del nodo p_i , y $(0 \leq a \leq 1)$ y b son valores arbitrarios a seleccionar; a determina la importancia relativa dada a los puntos 1 y 2, mientras que b representa el costo arbitrario asociado a cada nodo según el número de hijos que tenga en caso de ser seleccionado como padre.

Para determinar el nodo al cual se solicitará el join, se establece un orden considerando todos aquellos nodos de los cuales se dispone de información, es decir, los integrantes de la rama por la cual arribó el agente. El nodo intentará un join con el posible padre de costo mínimo, y en caso de no recibir respuesta debido a fallas, a que dicho nodo ya no pertenece al árbol de distribución ya que la información está actualizada al último heartbeat, o debido a que tiene colmada su capacidad, intentará con los demás en orden ascendente. Para minimizar el tiempo de joining en caso de respuestas negativas, el nodo podría enviar múltiples joins simultáneos, con el consiguiente aumento de carga en la red.

Además de utilizar esta heurística para que un nodo no conectado al árbol seleccione un padre adecuado, se propone su uso en otros casos, por ejemplo cuando un servidor ya integrante del árbol recibe un agente, puede recalcular los costos al conjunto ampliado de posibles padres, los propios más los pertenecientes a la rama del árbol a través de la cual arribó el agente.. En caso de detectar un nodo en mejores condiciones que el actual padre, se producirá el correspondiente cambio, siendo esto transparente a los servidores descendientes de los nodos involucrados.

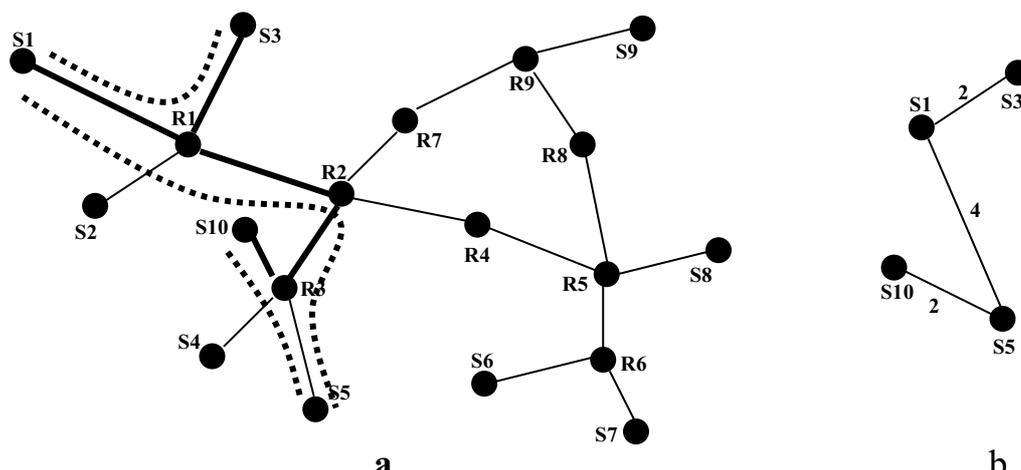


Fig.1 a-Árboles de distribución obtenidos por un protocolo multicast a nivel de red (líneas gruesas de router a router) y utilizando la heurística descrita (líneas punteadas de servidor a servidor). Las líneas punteadas se han dibujado de manera tal que reflejen los routers involucrados en el tráfico de paquetes. b- Árbol de distribución a nivel servidores (entre paréntesis se indica la distancia en nodos).

Este mismo chequeo de padres podrá hacerse en el caso de detectar un cambio en la información recibida a través de los heartbeats. Debe tenerse en cuenta que no existe el riesgo de producir ciclos en el árbol, debido a que un nodo que selecciona un padre, conoce la secuencia de nodos entre él y el raíz. Otro caso de recálculo de costos

puede darse si se contemplan procedimientos para difundir información respecto a la conformación del árbol a los nodos (además de los heartbeats, que proporcionan información respecto a la rama a la cual pertenece cada servidor).

En la figura 1 se muestra el árbol de distribución construido con la heurística descrita, comparándolo con un árbol construido por un protocolo multicast a nivel de red, suponiendo el mismo orden de solicitudes de joining por parte de los nodos.

3.1 Ejemplo de funcionamiento

A continuación se describe el comportamiento de los servidores de agentes al recibir un agente y agregarse, en consecuencia, al árbol de distribución. La topología de la red y la ubicación de los servidores corresponde a la figura 1. Los parámetros utilizados en la función son $a = 0,75$ y $b = 4$.

1-El agente **a** es activado en el servidor **S1**, como consecuencia éste se convierte en raíz del árbol de distribución
 2-Agente **a** se traslada a **S5**, lleva la información (S1,0,0) que hace referencia al único servidor conocido por el agente; en este caso la distancia de red al servidor anterior carece de sentido (se le da valor cero) y el número de hijos del servidor, es cero (no tiene servidores hijos).

3-Servidor **S5** realiza el join con **S1**, único servidor conocido. El camino a nivel de red es S5,R3,R2,R1,S1

4-Agente **a** se clona en **S5**, produciendo agente **b**. Agente **a** migra a **S3**, y **b** migra a **S9**

5-**S3** recibe de agente **a** la información (S1,0,1/S5,4,0), indicando el camino desde la raíz recorrido por el agente (**S1** y **S4**), la distancia de red entre **S4** y **S1** (4), y el número de hijos de ambos servidores. De la aplicación de la función definida por la heurística se obtiene:

$$C(S1) = 0,75 * 2 + 0,25 * 2 + 4 * 1 = 6$$

$$C(S5) = 0,75 * 8 + 0,25 * 4 + 4 * 0 = 7$$

Por lo tanto decide join con **S1**, el servidor de menor costo

6- **S9** recibe de agente **b** la información (S1,0,1/S5,4,0), aplica la función y obtiene:

$$C(S1) = 0,75 * 5 + 0,25 * 5 + 4 * 1 = 9$$

$$C(S5) = 0,75 * 9 + 0,25 * 5 + 4 * 0 = 8$$

Por lo tanto decide join con **S5**

7-Luego de los correspondientes joins, suponemos que **S1** emite su heartbeat: (S1,0,2), que indica identificación del servidor, distancia al servidor anterior en el árbol de distribución y número de hijos. Es recibido por **S3** y **S5**

S5 lo actualiza con su información: (S1,0,2/S5,4,1) y lo envía a **S9**

8- Agente **b** en **S9** migra a **S10**, éste recibe la información (S1,0,2/S5,4,1/S9,5,0). Calcula costos y obtiene:

$$C(S1) = 0,75 * 0 + 0,25 * 4 + 4 * 2 = 9$$

$$C(S5) = 0,75 * 4 + 0,25 * 2 + 4 * 1 = 7,50$$

$$C(S9) = 0,75 * 9 + 0,25 * 5 + 4 * 0 = 8$$

S10 decide por lo tanto conectarse vía **S5**, y como consecuencia **S9** se desconecta del árbol.

9- El agente **b** en **S10** se clona produciendo un agente **c** que decide migrar a **S3**

10- **S1** envía un nuevo heartbeat, (S1,0,2).

11-**S5** lo actualiza con su información y lo envía a **S10**, (S1,0,2/S5,4,1)

12-**S3** recibe al agente **c**, con la información proveniente de **S10**: (S1,0,2/S5,4,1/S10,2,0).

Además de esta información, el nodo **S3** dispone de la suya: (S1,0,2) correspondiente al último heartbeat generado por el nodo raíz **S1** (en este caso no agrega ningún nodo a los ya conocidos). Aunque **S3** ya pertenece al árbol de distribución, chequeará nuevamente los costos por si logra reducir el actual:

$$C(S1) = 0,75 * 0 + 0,25 * 2 + 4 * 2 = 8,5 - 4 = 4,5 \text{ (el propio)}$$

$$C(S1) = 0,75 * 0 + 0,25 * 2 + 4 * 2 = 8,5 - 4 = 4,5 \text{ (recibido de S10)}$$

$$C(S5) = 0,75 * 4 + 0,25 * 4 + 4 * 1 = 8$$

$$C(S10) = 0,75 * 6 + 0,25 * 4 + 4 * 0 = 5,5$$

Debe tenerse en cuenta que cuando no se trata del nodo raíz, la evaluación de datos para un posible nodo padre, recibidos de diversas fuentes puede dar lugar a obtener distintos costos, ya que las distancias multicast al nodo raíz pueden diferir. Al estar **S3** conectado directamente como hijo a **S1**, esto debe reflejarse en la función de costos, restando uno al número de hijos anunciado por **S1**. De esta forma, **S3** decide no cambiar de padre. Notar que en las mismas circunstancias pero con **S3** no miembro del árbol, se hubiera decidido por el join a **S10**.

4. Simulación

Con el objeto de estimar la performance de nuestra propuesta, se realizaron simulaciones para las siguientes alternativas: 1-para un protocolo a nivel de red que construye árboles bidireccionales alternativos, Core Based Trees [1], tomado como referencia; 2-para la versión más simple de soporte multicast implementado a nivel servidores de agentes, en la cual un servidor se conecta al árbol de distribución utilizando el último servidor visitado por el agente que solicita pertenencia al grupo, y 3-para multicast implementado a nivel servidores de agentes utilizando la heurística descrita en la sección 3.

Las simulaciones fueron realizadas utilizando el simulador de redes Network Simulator (Ns) [3]. Se desarrolló el código (simplificado) para soporte multicast en los routers en el caso de CBT y el necesario en los servidores de agentes para las dos alternativas restantes. La topología utilizada consistió de 256 nodos interconectados por 266 vínculos de transmisión y se generó a partir de una pequeña porción real de la Internet.

Cada corrida de la simulación consistió en ejecutar una aplicación inicializada en un servidor elegido al azar, compuesta al comienzo por un único agente. La movilidad y frecuencia de clonado de los agentes se generó al azar, no correspondiendo con ningún patrón observado para algún tipo de aplicaciones en particular, análisis que se realizará en el futuro. De esta manera, cada agente perteneciente a la aplicación permanece un tiempo en un servidor, pudiendo luego generar clones y seleccionar al azar un nuevo servidor para migrar. Estos clones se comportan de la misma manera que el agente original.

Para no generar una cantidad de agentes excesiva para la red definida, su máximo fue restringido a 10, luego de alcanzarse esta cantidad, los agentes pueden migrar, pero no clonarse. La cantidad máxima de migraciones se configuró en 80, dado que la visita de varios agentes a un nodo no se correspondería con la realidad en redes de área extensa.

Las corridas consistieron en la generación y migración de agentes hasta alcanzar los números máximos mencionados. Luego, se enviaron datos desde uno de los agentes al resto, con el objeto de evaluar la performance del árbol de distribución obtenido en cada caso. Los resultados obtenidos, para las tres alternativas, se muestran en las figuras 2, 3 y 4.

Los parámetros evaluados fueron los siguientes:

1-Relativos a la calidad del árbol de distribución obtenido:

a-Carga en número de paquetes producidos en la red como consecuencia del envío de un paquete de datos multicast dirigido al grupo por parte de un agente. b-Tiempo promedio de las demoras registradas por los agente en recibir dicho paquete. Estos valores nos dan una idea de la medida en que el árbol de distribución se ajusta a la topología de la red. Como es obvio, el protocolo multicast a nivel de red produce los mejores resultados, tanto en tiempos de envío como en carga en la red, ya que explota la topología. En los casos de multicast implementado a nivel aplicación, la diferencia entre la solución más simple, que sólo tiene en cuenta la secuencia de migración de los agentes y la que evalúa padres alternativos, que explota un conocimiento parcial de la topología de la red y está limitada respecto a los nodos que pueden integrar el árbol de distribución (sólo servidores de agentes), es significativa, mejorándola en ambos aspectos aproximadamente en un factor de 3.

2-Relativos a los mecanismos de pertenencia al árbol de distribución.

Se evaluó: a-la carga producida por las solicitudes de pertenencia al árbol (joins), su confirmación y b-los mensajes de abandono del árbol (leaves) y la demora de los nodos para lograr su pertenencia al árbol de distribución. En ambos casos se observó que la solución a nivel de red mejora a las soluciones a nivel aplicación. Esto se debe a que el soporte multicast a nivel de red sólo provee conectividad entre los miembros del grupo, sin ofrecer ningún tipo de soporte adicional para transmisión confiable, como lo hacen las soluciones a nivel aplicación. Entre éstas, el agregado de miembros al árbol acorde a la secuencia de migración de los agentes resulta mejor, debido a que no es necesario buscar a ni establecer una nueva conexión TCP con el servidor padre, que es aquél del cual provino el agente. Sin embargo, esta aparente mejora debe confrontarse con la baja calidad del árbol de distribución obtenido.

3-Relativos a tiempos de migración

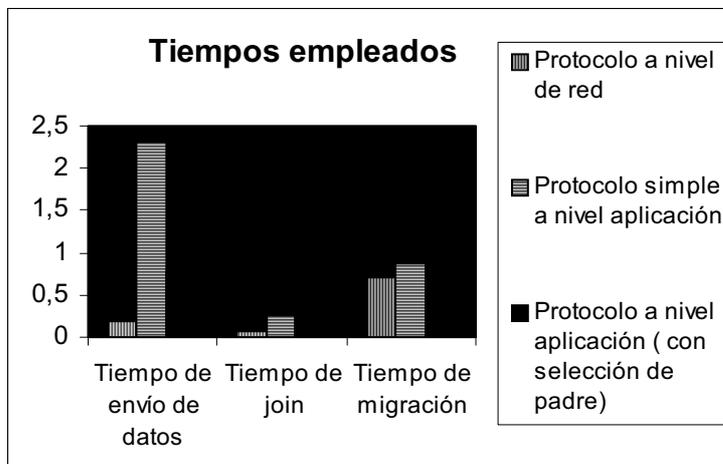
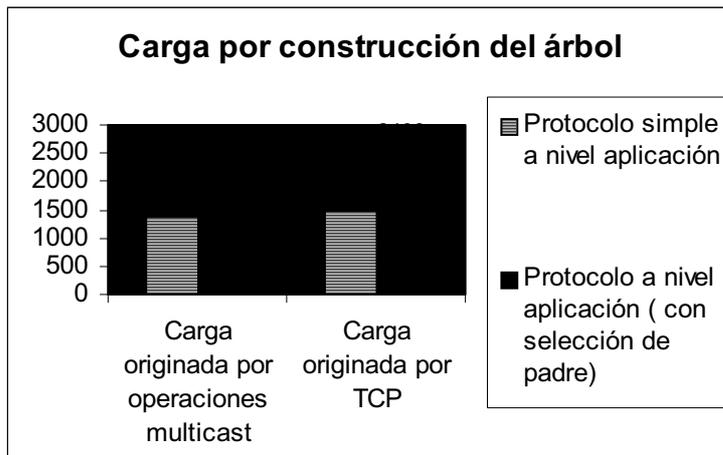
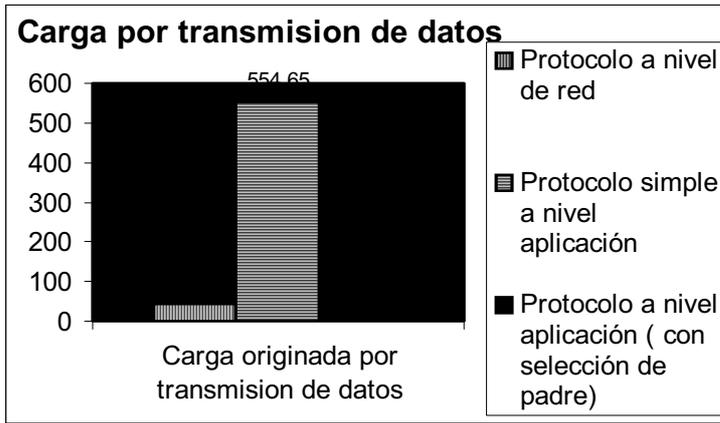
El tiempo de migración de un agente, es tomado desde que el agente solicita migrar a su servidor actual, hasta que es activado en el nuevo servidor, estando éste en condiciones de recibir información multicast. Por lo tanto, incluye el tiempo de transferencia del agente, y el posible tiempo de incorporación al árbol de distribución por parte del nuevo servidor. Cabe aclarar que también en este caso, aunque el tiempo obtenido por el protocolo multicast a nivel de red es menor, no incluye soporte alguno para posibilitar transmisión confiable de datos. La incorporación de un mecanismo de este tipo aumentaría considerablemente los tiempos obtenidos. En el caso de la solución basada en el recorrido de los agentes, el tiempo es menor debido a que no es necesario localizar y conectarse a un nuevo servidor de agentes, ya que el padre del nuevo servidor es el servidor anterior.

4-Carga a nivel TCP

Este parámetro fue evaluado sólo en los casos de multicast implementado a nivel aplicación. Se evaluó la cantidad de paquetes introducida en la red como consecuencia de las conexiones TCP necesarias para lograr la transferencia de los agentes y la construcción del árbol de distribución, implementado como sucesiones de conexiones TCP compartidas.

Nuevamente, en el caso del árbol construido de acuerdo al itinerario de los agentes, se obtiene una mejora debido a que se comparte la conexión TCP entre los servidores involucrados parar el envío del agente y para las operaciones de pertenencia al árbol y posterior envío de datos.

Según los resultados obtenidos, puede concluirse que si bien una solución multicast a nivel de red presenta mejor performance en todos los aspectos, no tiene en cuenta el ofrecer un soporte de transmisión confiable, necesario para el trafico a intercambiar entre los agentes móviles. La incorporación de un nivel de multicast confiable, daría lugar a tiempos y cargas que superarían a los obtenidos por las soluciones de multicast a nivel



Figs. 2. 3 v 4. Resultados obtenidos de las simulaciones

aplicación¹³. Otro inconveniente lo constituye la no existencia de tal soporte para grupos altamente dinámicos que se extienden en áreas extensas de la Internet. Por otro lado, el uso de soluciones a nivel de red no permite desplegar los servicios requeridos en aquellas zonas de la red sin soporte multicast.

Si bien la solución más simple con multicast a nivel aplicación aparenta ser más liviana en cuanto a la carga y más rápida respecto a los tiempos registrados para los procesos de join y migración, se considera que estas diferencias se ven ampliamente superadas por la baja calidad del árbol de distribución obtenido, lo que se refleja en las cargas y tiempos relativos al envío de los datos.

5. Conclusiones y trabajo en curso

La solución propuesta al problema de comunicación grupal y localización en sistemas de múltiples agentes móviles operando sobre áreas extensas, de acuerdo a los resultados obtenidos, puede ser considerada una alternativa válida al uso de multicast a nivel de red. Las ventajas de mayor peso de nuestra solución sobre un soporte multicast tradicional, radican en la posibilidad de desplegar el sistema sin necesidad de una infraestructura multicast en la red, y la sencillez de implementación. Otra característica a destacar es que nuestra solución, al utilizar los servidores como nodos del árbol de distribución, y vincularlos a través de conexiones TCP, allana el camino para una implementación de transmisión grupal confiable, característica que con el esquema tradicional requiere la implementación de un nivel adicional, multicast confiable, el cual aún no ha sido definido satisfactoriamente para el tipo de características de comunicación que presentan los sistemas de agentes móviles.

De acuerdo a los resultados obtenidos en las simulaciones realizadas, considerando tiempos y carga en la red, esta solución resulta factible.

En la actualidad se está trabajando en la mejora de la heurística propuesta, mediante el ajuste de los parámetros definidos y con el agregado de métodos adicionales que permitan aplicarla de manera regular (sin necesidad de la visita de nuevos agentes) para mejorar el árbol de distribución. Debe ser evaluado cuidadosamente el costo de esta alternativa, confrontándolo con el beneficio obtenido de mejorar el árbol, ya que éste puede verse disminuido considerablemente debido a la movilidad de los agentes que puede ocasionar que un servidor permanezca en el grupo multicast por un período breve de tiempo.

Otro aspecto en el que se comenzará a trabajar es la implementación de distintas alternativas de transmisión confiable a ofrecer sobre el soporte definido (asentimientos implícitos, explícitos, etc.) previo relevamiento de las necesidades de comunicación grupal confiable de este tipo de aplicaciones,

Paralelamente se implementará el soporte propuesto en Java con el objeto de comprobar su comportamiento en un escenario real (aunque restringido en el número de servidores) a través del desarrollo de una aplicación simple.

Referencias

- [1], "Core based trees (CBT)" T. Ballardie, P. Francis & J. Crowcroft, SIGCOMM Symposium on Communications Architectures and Protocols , pp. 85-95, ACM, September 1993..
- [2] "Reliable Overlay Multicast with Loosely Coupled TCP Connections –ROMA", Gu-In –Kwon, John W. Byers, Proc. of IEEE INFOCOM '04, Hong Kong, March 2004.
- [3] K. Fall (ed), "Ns Notes and Documentation", VINT Project, UC Berkeley, LBL, USC/ISI, Xerox PARC, March 2000.
- [4], "Transcience of Peers and Streaming Media", Mayank Bava, Hrishikesh Deshpande, Hector Garcia-Molina, ACM SIGCOMM Computer Communication, Volume 33 , Issue 1, January 2003 Pp: 107 - 112 [Murphy,2002]
- [5]"Reliable Communications for Highly Mobile Agents", Amy Murphy, Gian Pietro Picco, Journal of Autonomous Agents and Multi-Agent Systems, vol. 5, no. 1, pp. 81-100, March 2002.
- [6] "Intra- and Inter- domain multicast routing protocols: A survey and taxonomy", M. Ramalho, IEEE Communications Surveys and Tutorials, vol.3, no.1, pp.2-25, Jan.-Mar. 2000
- [7] "Infraestructura de comunicación para sistemas de agentes móviles". Rigotti, G, Proceedings AST 2003, 32 JAIIO (Jornadas Argentinas de Informática e Investigación Operativa), Buenos Aires, Argentina, September 2003.

¹³ El overhead producido por el nivel de multicast confiable es estimativo, ya que su evaluación requeriría el diseño e implementación de un protocolo.