

Búsqueda de Recursos en Redes Peer-to-Peer  
Totalmente Descentralizadas Basada en Redes  
Neuronales Artificiales



Leonardo C. Corbalán

Directores: Armando E. De Giusti y Laura C. Lanzarini

Tesis presentada para obtener el grado de Magister en Redes de Datos

Facultad de Informática - Universidad Nacional de La Plata

Mayo de 2014



# Índice

Índice .....	III
Lista de acrónimos .....	VI
<b>INTRODUCCIÓN.....</b>	<b>1</b>
Motivación .....	5
Objetivo y Aporte de esta Tesis .....	6
Publicaciones.....	7
Contenido de la Tesis .....	8
El Background.....	8
El Aporte .....	9
<b>CAPÍTULO 1 SISTEMAS PEER-TO-PEER (P2P) .....</b>	<b>11</b>
1.1 Introducción .....	11
1.2 Características .....	14
1.3 Visión histórica .....	16
1.3.1 El Modelo Peer-to-Peer en los Inicios de la Internet .....	17
1.3.2 La transición a un Modelo Asimétrico en la Internet .....	18
1.3.3 El Nuevo Auge por los Sistemas Peer-to-Peer .....	20
1.3.4 Hitos Históricos.....	21
1.3.4.1 Protocolo FastTrack y Redes Kazaa, Grokster e iMesh .....	21
1.3.4.2 Gnutella 0.4.....	22
1.3.4.3 eDonkey.....	23
1.3.4.4 Gnutella 0.6.....	24
1.3.4.5 Redes Estructuradas. Chord, Pastry, Tapestry, CAN .....	25
1.3.4.6 BitTorrent.....	26
1.4 Tipos de Redes P2P .....	27
1.4.1 Sistemas P2P Puros no Estructurados.....	27
1.4.2 Sistemas P2P Puros Estructurados.....	30
1.4.3 Sistemas P2P Híbridos.....	36
1.5 Aplicaciones .....	39
1.6 Resumen del Capítulo.....	44
<b>CAPÍTULO 2 REDES NEURONALES ARTIFICIALES (RNA) .....</b>	<b>49</b>
2.1 Introducción .....	49
2.2 La Neurona Biológica .....	53
2.2.1 Ventajas de la Comunicación Química .....	58
2.3 La Neurona Artificial .....	59
2.4 El <i>Perceptrón</i> . Primera Red Neuronal Artificial con Capacidad de Aprendizaje .....	61
2.4.1 El <i>Perceptrón</i> Simple .....	62
2.4.2 Aprendizaje del <i>Perceptrón</i> Simple.....	66
2.4.3 <i>Perceptrón</i> Multicapa.....	67
2.5 Backpropagation Network (BPN).....	70
2.5.1 Regla de Aprendizaje .....	73
2.5.2 Consideraciones Prácticas sobre la BPN.....	74

## IV

2.6	Self-Organizing Maps (SOM) .....	77
2.6.1	Arquitectura SOM.....	78
2.6.2	La Neurona Competitiva .....	81
2.6.2.1	Regla de Aprendizaje.....	82
2.6.3	Entrenamiento de la Red SOM.....	83
2.7	Resumen del Capítulo.....	85
<b>CAPÍTULO 3 BII-P2P. EL MÉTODO PROPUESTO.....</b>		<b>87</b>
3.1	Con Base en <i>Gnutella</i> .....	88
3.1.1	Búsqueda de Recursos en la Red <i>Gnutella</i> .....	91
3.1.2	Normas en la Propagación de los Mensajes.....	92
3.1.3	Limitaciones de <i>Gnutella</i> .....	93
3.2	Escenario de Aplicación .....	94
3.3	Los dos Aspectos Destacados en BII-P2P .....	98
3.4	Resumen del Capítulo.....	100
<b>CAPÍTULO 4 BÚSQUEDA INTELIGENTE EN BII-P2P .....</b>		<b>103</b>
4.1	El Rol de los Nodos en BII-P2P.....	105
4.1.1	Conceptos Generales .....	105
4.1.2	La Red Neuronal de los Nodos P2P.....	106
4.1.3	Arquitectura Neuronal.....	110
4.1.4	Funcionamiento de la Búsqueda Inteligente.....	112
4.1.4.1	Funcionamiento de la Red Neuronal.....	112
4.1.4.2	Propagación de la Solicitud de Búsqueda.....	114
4.2	El Aprendizaje .....	117
4.2.1	Extensión del Protocolo <i>Gnutella</i> .....	118
4.2.2	Información de Actualización. Mensaje <i>NeuralInf</i> .....	120
4.2.3	Políticas de Adquisición y Actualización de Conocimiento.....	123
4.2.3.1	Política de Difusión Simple .....	123
4.2.3.2	Difusión limitada en profundidad (DLP). .....	124
4.2.3.3	Difusión Limitada en Profundidad y Anchura (DLPA) .....	125
4.2.3.4	Alcance Real de la Información de Actualización en DLP y DLPA .....	125
4.2.3.5	Determinación de la Frecuencia de Actualización.....	127
4.2.4	Enfoque Híbrido .....	127
4.2.4.1	Learning Vector Quantization (LVQ).....	128
4.2.4.2	Implementación de Aprendizaje en Línea Basado en LVQ1.....	130
4.2.4.3	Difusión de Información de Actualización con Entrenamiento en Línea ....	132
4.3	Protocolo de Ingreso a la Red P2P.....	132
4.4	Resumen del Capítulo.....	133
<b>CAPÍTULO 5 EVALUACIÓN DE LA BÚSQUEDA INTELIGENTE BII-P2P .....</b>		<b>137</b>
5.1	Evaluación de la Capacidad de Búsqueda .....	137
5.1.1	La Experimentación .....	138
5.1.2	Construcción de los Casos de Prueba .....	141
5.1.3	Resultados de la Experimentación.....	142
5.1.3.1	Acerca del Análisis de los Datos .....	142
5.1.3.2	Rendimiento de la Búsqueda Inteligente sobre Red de 2.000 Nodos .....	144
5.1.3.3	Rendimiento de la Búsqueda Inteligente sobre Red de 5.000 Nodos .....	152
5.1.3.4	Rendimiento de la Búsqueda Inteligente sobre Red de 10.000 Nodos .....	158
5.1.3.5	Rendimiento de la Búsqueda Inteligente sobre Red de 20.000 Nodos .....	164

5.1.3.6	El Tamaño de la Red y su Impacto en el Rendimiento de la Búsqueda.....	170
5.1.4	Conclusiones sobre la Capacidad de Búsqueda .....	175
5.2	Evaluación de la Capacidad de Adaptación .....	176
5.2.1	Experimentación.....	176
5.2.2	Construcción de los Casos de Prueba.....	177
5.2.3	Resultados de la Experimentación.....	178
5.2.4	Conclusiones sobre la Capacidad de Adaptación.....	189
5.3	Conclusiones Generales sobre la Búsqueda Inteligente en BII-P2P.....	190
<b>CAPÍTULO 6 EXPLORACIÓN INCREMENTAL EN BII-P2P .....</b>		<b>193</b>
6.1	Motivación .....	194
6.2	Descripción de la Estrategia de Exploración Incremental .....	195
6.3	Implementación de la Exploración Incremental en BII-P2P .....	198
6.4	Evaluación de la Estrategia de Exploración Incremental.....	202
6.4.1	Experimentación.....	202
6.4.2	Construcción de los Casos de Prueba.....	204
6.4.3	Resultados de la Experimentación.....	204
6.5	Conclusiones sobre la Estrategia de Exploración Incremental en BII-P2P.....	220
<b>CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN.....</b>		<b>223</b>
<b>BIBLIOGRAFÍA .....</b>		<b>229</b>
<b>A.</b>	<b>APÉNDICE A .....</b>	<b>241</b>

## Lista de acrónimos

<b>ADSL</b>	Asymmetric Digital Subscriber Line (Línea de abonado digital asimétrica)
<b>ARPANET</b>	Advanced Research Projects Agency Network
<b>BFS:</b>	Breadth First Search (búsqueda primero en anchura)
<b>BII-P2P:</b>	Búsqueda Inteligente Incremental en sistemas Peer-to-Peer
<b>BONIC:</b>	Berkeley Open Infrastructure for Network Computing (infraestructura abierta de Berkeley para la computación en red)
<b>BPN:</b>	BackPropagation Network (red de propagación hacia atrás)
<b>CPN:</b>	ConterPropagation Network (red de contrapropagación)
<b>DHT:</b>	Distributed Hash Table (tablas de hash distribuidas)
<b>DLP:</b>	Difusión Limitada en Profundidad
<b>DLPA:</b>	Difusión Limitada en Profundidad y Anchura
<b>DNS</b>	Domain Name System (sistema de nombres de dominio)
<b>FTP:</b>	File Transfer Protocol (protocolo de transferencia de archivos)
<b>HTTP</b>	Hypertext Transfer Protocol (protocolo de transferencia de hipertexto)
<b>IP:</b>	Internet Protocol
<b>IRC</b>	Internet Relay Chat
<b>ISP:</b>	Internet Server Provider.
<b>LRU</b>	Least Recently Used (Menos usado recientemente)
<b>NAT:</b>	Network Address Translation (traducción de direcciones de red)
<b>NFS:</b>	National Science Foundation

<b>OSI</b>	Open System Interconnection (modelo de interconexión de sistemas abiertos)
<b>P2P:</b>	Peer-to-Peer (redes entre pares o iguales)
<b>P2PTV:</b>	Peer to Peer TeleVision.
<b>P2P-VoD:</b>	Peer to Peer Video on Demand
<b>RNA:</b>	Redes Neuronales Artificiales.
<b>RNE:</b>	Redes Neuronales Evolutivas
<b>SETI:</b>	Search for Extraterrestrial Intelligence (búsqueda de inteligencia extraterrestre)
<b>SOM:</b>	Self Organizing Maps (mapas auto-organizativos)
<b>TCP</b>	Transmission Control Protocol (protocolo de control de transmisión)
<b>TELNET:</b>	TELEcommunication NETwork
<b>TTL:</b>	Time To Live (tiempo de vida)
<b>USENET</b>	USErs NETwork (red de usuarios)
<b>VoIP:</b>	Voice over IP (voz sobre IP)





# Introducción

Las redes *Peer-to-Peer* (P2P) conforman sistemas distribuidos, frecuentemente de gran escala, donde computadoras y otros dispositivos se conectan entre sí en una relación simétrica como pares o iguales. Estos equipos así conectados intercambian información actuando como servidores y clientes al mismo tiempo. Pueden funcionar sin ninguna administración central, evitando las jerarquías y los roles diferenciados. Los pares integrantes gozan de un alto grado de autonomía.

Los sistemas P2P permiten conformar grandes comunidades que comparten ciclos de CPU —*SETI@Home*, *Docking@Home*, *MindModeling@Home*—, espacio de almacenamiento, —*Napster*, *FreeNet*, *Gnutella* 0.4, *BitTorrent*—, ancho de banda —*Skype*, *SopCast*, *PPLive*— o ambientes colaborativos —*Microsoft SharePoint Workspace*—. El explosivo crecimiento que han tenido estos sistemas en los últimos tiempos se debe en gran medida a su bajo costo y a la alta disponibilidad de recursos factibles de ser compartidos en Internet. A diferencia de otros sistemas distribuidos, las redes P2P promueven la participación de una gran cantidad de computadoras que se unen y dejan la red con mucha frecuencia por lo que constituyen sistemas altamente dinámicos.

Existen sistemas P2P muy populares, entre ellos *eDonkey* y *BitTorrent*, que han incorporado algún grado de centralización para implementar algún aspecto de la actividad que desarrollan. A estos sistemas se los denomina híbridos. En oposición, protocolos como *Gnutella* 0.4 respetan cabalmente la filosofía P2P y no existe en ellos ningún vestigio de control central o parcial de la red, ni

relaciones jerárquicas, ni tareas especializadas, ni roles distintivos. A estos sistemas se los denomina Puros.

El modelo híbrido se basa en la utilización de uno o más servidores centrales que mantienen el estado de las transacciones entre los pares. Estos servidores cuentan con el detalle de los recursos que posee cada nodo de la red. Cuando un usuario necesita encontrar algún recurso, envía la consulta a alguno de estos servidores que le responde con la identificación de los nodos que pueden satisfacerlo. Por lo tanto, es posible contar con un mecanismo de búsqueda de recursos relativamente sencillo en estos sistemas híbridos. Sin embargo este modelo presenta algunas debilidades. Una falla sobre un servidor central tiene graves consecuencias en el rendimiento de todo el sistema completo. Además los servidores frecuentemente se transforman en cuellos de botella y su utilización dificulta la escalabilidad.

En los sistemas P2P puros, computadoras individuales se comunican directamente con otras y comparten información y recursos sin la utilización de servidores centrales. No existe punto alguno de centralización. Cualquier nodo de la red puede iniciar una petición por un recurso y responder a otras. Los usuarios tienen un alto grado de autonomía y control sobre los servicios que utilizan. Características claves de estos sistemas incluyen: descentralización, auto-organización, dinamismo y tolerancia a fallas, que los hacen naturalmente escalables y soluciones atractivas para compartir información y otras clases de recursos.

Los sistemas P2P puros, a su vez se subdividen en sistemas estructurados y no estructurados. *Chord* es un ejemplo de un sistema estructurado y, al igual que otros de la misma categoría, nació en un entorno universitario y basa su funcionamiento en una tabla de *hash* distribuida entre los nodos de la red organizados en una estructura no jerárquica. Aunque los sistemas estructurados implementan mecanismos de búsqueda eficientes, mantener la estructura

funcionando correctamente presenta algunos inconvenientes por lo que estos sistemas no alcanzaron mucha popularidad. En oposición, *Gnutella* 0.4 es un sistema P2P puro no estructurado que no requiere tareas especiales de mantenimiento, sencillo y fácil de implementar, pero proveer un mecanismo de búsqueda eficiente representa un desafío mayor. El problema consiste en grafos, nodos, enlaces y recursos.

En líneas generales los sistemas descentralizados no estructurados funcionan de la siguiente forma: los nodos sólo conocen los recursos que ellos mismos poseen y a un conjunto de vecinos con los que se conectan en forma directa. Cualquier nodo puede iniciar una búsqueda de recursos difundiendo un mensaje de solicitud en la red. Dicho mensaje atraviesa enlaces en base a decisiones locales y siempre que la solicitud alcance un nodo con el recurso requerido, éste le responde. El desafío en estos sistemas es proveer un mecanismo de búsqueda que minimice la cantidad de mensajes generados sin degradar la capacidad de localización de los recursos.

Para la búsqueda de recursos *Gnutella* 0.4, la red más representativa del modelo puro no estructurado, implementa una especie de *broadcast* limitado basado en un algoritmo *Breadth-First Search* (BFS). El nodo que inicia una solicitud la envía a todos sus vecinos. A su vez, ellos la pasan a todos sus otros vecinos y la contestan en caso de poseer el recurso buscado. Los nodos ignoran solicitudes repetidas para evitar ciclos. Sin embargo esta estrategia no es escalable pues los paquetes de solicitudes crecen rápidamente conforme más nodos se unen a la red. Con el fin de que el número de mensajes en la red no la colapse, las solicitudes tienen un contador decreciente *Time-to-Live* (TTL) que limita el número de saltos —*hops* en inglés— y por lo tanto el alcance de cada petición en el sistema.

Varios métodos han sido propuestos y evaluados por diferentes investigadores intentando obtener mejores soluciones de búsquedas en sistemas como *Gnutella*

0.4 [1]. Se ha encontrado que su algoritmo BFS no es escalable y su rendimiento es bajo especialmente sobre una topología con distribución *power-law* —pocos nodos con muchos vecinos y muchos nodos con pocos vecinos—, los efectos de la inundación de paquetes de solicitudes son desastrosos: el número de mensajes aumenta drásticamente cuando TTL es incrementado.

Un método propuesto denominado *Expanding Ring*, donde el TTL es extendido gradualmente, puede mejorar la situación pero produce el envío de mensajes duplicados a nodos que la petición ya ha alcanzado disminuyendo su efectividad.

En la búsqueda *Random Walk*, múltiples agentes son impulsados a través de una selección de vecinos aleatoria, se incrementa el número de saltos y por lo tanto el retardo, pero se disminuye el tráfico total del sistema.

*Modified Random BFS* [2] se comporta como BFS de *Gnutella* 0.4, pero los nodos seleccionan solo un subconjunto aleatorio de vecinos para reenviar la solicitud. Esto reduce el tráfico pero ajustar el correcto tamaño del subconjunto puede ser difícil.

En otra variante llamada simplemente *Intelligent Search Mechanism*, los nodos mantienen la pista de las peticiones recientes respondidas por sus vecinos. Utilizar esta información para encaminar las próximas solicitudes en general mejora el rendimiento de la red.

Otras búsquedas basadas en heurísticas han sido propuestas [3]. Estas heurísticas incluyen el número de resultados retornados, el más corto tiempo promedio de satisfacción, el número promedio de saltos más chico, el retardo más corto, etc. Los resultados sugieren que para minimizar el tiempo de satisfacción la mejor estrategia es pasar la petición al vecino que ha tenido el más corto promedio en tiempo de satisfacción para las últimas 10 solicitudes.

Sin embargo todas estas estrategias presentan limitaciones para adaptarse a un ambiente dinámico con nodos que entran y salen de la red con mucha frecuencia. Existen otras propuestas similares basadas en heurísticas [4] y algunas utilizan Redes Neuronales Artificiales (RNA) para dirigir la consulta hacia los vecinos más prometedores. En particular *NeuroSearch* [5] utiliza Redes Neuronales Evolutivas (RNE) para implementar un algoritmo eficiente de búsqueda basado en información local. Los resultados indican que la optimización evolutiva puede mejorar el rendimiento del algoritmo BFS de *Gnutella* 0.4, sin embargo, debido al tiempo necesario para evolucionar la población de redes neuronales sólo es adecuado para entornos estáticos, impracticable en la mayoría de las aplicaciones P2P.

## Motivación

Queda claro que hallar un mecanismo eficiente de búsqueda de recursos en redes P2P puras es un tema abierto que aún despierta interés en la comunidad científica. Aunque las soluciones estructuradas que utilizan tablas de *hash* distribuidas han mostrado buenos resultados, la transitoriedad de los nodos tan característica en los sistemas P2P ocasiona un importante gasto general—*overhead*— asociado al mantenimiento de las tablas necesarias para soportar la estructura del sistema. [6]. Los sistemas estructurados superan a los sistemas no estructurados cuando el recurso buscado es escaso, pero si existen muchas instancias del mismo —como suele ser habitual— esta ventaja se desvanece rápidamente [7].

Las utilidades obtenidas de un sistema de búsqueda eficiente sobre redes P2P puras no estructuradas, resultan motivadoras para abordar el estudio de esta problemática con la intención de poder desarrollarlo. Es precisamente este desarrollo lo que constituye el objetivo principal de esta tesis.

## Objetivo y Aporte de esta Tesis

El objetivo principal establecido para la ejecución de este trabajo de tesis ha sido el estudio, diseño y desarrollo de un nuevo mecanismo eficiente para la búsqueda de recursos en redes P2P puras no estructuradas. La estrategia elegida ha sido dotar a cada nodo del sistema con la "inteligencia" necesaria para que, en función del recurso buscado, se dirijan las solicitudes sólo hacia los vecinos más apropiados, disminuyendo así la cantidad de paquetes necesarios sin perder la capacidad de localización de los recursos.

Esta inteligencia de los nodos proviene de una red neuronal competitiva de características similares a los mapas auto-organizativos (SOM) y redes LVQ implementada en cada uno de ellos. Cuando un nodo necesita propagar una solicitud de búsqueda, estimula con la información del recurso buscado a su red neuronal, la cual identifica el subconjunto de vecinos más apropiados a los cuales dirigir la consulta.

Uno de los desafíos surgidos tempranamente en el desarrollo del presente trabajo ha sido la necesidad de idear un mecanismo eficiente de aprendizaje para estas redes neuronales que permita mantener actualizado el conocimiento adquirido. La vigencia de este conocimiento se ve rápidamente comprometida ante los cambios tan característicos de los sistemas P2P. Nodos nuevos que se unen y otros que se retiran del sistema, o recursos que se ocupan o liberan en un sitio de la red constituyen el tipo de cambio más frecuente que el subsistema de búsqueda debe saber manejar.

El aporte de este trabajo se sustenta en la necesidad de encontrar variantes eficientes para compartir recursos en sistemas P2P descentralizados sin estructura. En los recientes años el interés por la tecnología P2P se ha incrementado especialmente debido a los sistemas para compartir archivos tales como *Kazaa*, *eDonkey*, *Gnutella* 0.4 etc. Sin embargo, las redes P2P no se limitan a esta

actividad, sino que pueden utilizarse para conseguir sistemas capaces de compartir recursos más complejos como por ejemplo documentos, comunicaciones multimedia, poder de cómputo, servicios, etc. Incluso existen trabajos de investigación que implementan *task scheduling* en entornos de computación *Grid* utilizando este tipo de redes.

Por todo ello, proveer un manejo eficiente de redes P2P puras no estructuradas constituye una necesidad al mismo tiempo que un desafío interesante que enfrentar.

La nueva estrategia de búsqueda concebida en el marco de este trabajo de tesis se presenta aquí con el nombre de Búsqueda Inteligente e Incremental en Sistemas *Peer-to-Peer* (BII-P2P)

## Publicaciones

Parte de los resultados obtenidos en el trabajo de investigación realizado en el contexto de la presente tesis fueron publicados en tres artículos, dos de ellos en congresos en el exterior [8] [9] [10].

- ***Resources NeuroSearch in Peer-to-Peer networks.*** Leonardo Corbalán, Laura Lanzarini, Armando De Giusti. ITI 2009 31st International Conference on Information Technology Interfaces, 2009. pp. 597-602. IEEE.
- ***Búsqueda Neuronal de Recursos con Exploración Incremental en Redes Peer-to-Peer.*** Leonardo Corbalán, Laura Lanzarini, Armando De Giusti. Jornadas Chilenas de Computación. JCC 2009. Santiago de Chile. Chile. 9 al 14 de noviembre de 2009. XIII Workshop on Distributed Systems and Parallelism (WSDP). pp 11-20.

- ***Búsqueda Neuronal de Recursos Adaptativa en Sistemas Peer-to-Peer (BNAP2P)***. Leonardo Corbalán, Laura Lanzarini, Armando De Giusti. XVI Congreso Argentino de Ciencias de la Computación CACIC 2010. Buenos Aires, Argentina. 18 al 22 de octubre de 2010. pp 191-200.

## Contenido de la Tesis

Además de esta introducción la tesis cuenta con seis capítulos, las conclusiones, un apéndice y la bibliografía. En los dos primeros capítulos se presenta el *background* necesario para que el documento sea aceptablemente autocontenido. Los capítulos 3, 4, 5 y 6 representan el aporte de este trabajo. En ellos se describen los fundamentos y características de la Búsqueda Inteligente Incremental en redes P2P (BII-P2P), el método propuesto en esta tesis para resolver eficientemente las búsquedas de recursos en redes P2P puras no estructuradas. También se describen los detalles de la experimentación, los resultados obtenidos y las conclusiones alcanzadas durante los procesos de evaluación llevados a cabo para medir el rendimiento de BII-P2P.

## El Background

En el capítulo 1 se exponen los aspectos más sobresalientes de las redes P2P: origen y evolución, características, clasificación y campo de aplicación. Se presentan las ventajas y desventajas de cada enfoque pretendiendo hacer obvias las motivaciones que dieron origen a la propuesta y realización de esta tesis.

El capítulo 2 está dedicado al conexionismo dentro de la Inteligencia Artificial, en particular a las Redes Neuronales Artificiales. A medida que se describen algunas de las arquitecturas más importantes se van introduciendo conceptos fundamentales relacionados con sus capacidades, limitaciones y tipos de aprendizaje. Los contenidos expuestos en este capítulo dan contexto a algunos



tópicos del capítulo 4 que ilustran los mecanismos subyacentes de BII-P2P.

## El Aporte

En el capítulo 3 se presentan los aspectos más generales de BII-P2P y su relación con la red *Gnutella* 0.4. Se analiza el escenario de aplicación de la propuesta y se delimitan los dos aspectos fundamentales de BII-P2P que dan nombre a la estrategia: i) búsqueda inteligente y ii) exploración incremental. Ambos aspectos se presentan y analizan en profundidad en capítulos separados.

En el capítulo 4 se expone el aspecto más significativo de BII-P2P: la búsqueda inteligente. Se presentan los principios de su funcionamiento, el rol desempeñado por las redes neuronales, el aprendizaje de los nodos y las políticas de actualización de conocimiento. En el capítulo 5 se detalla la experimentación realizada, se analizan los resultados obtenidos y se formulan conclusiones.

En el capítulo 6 se describe la estrategia de exploración incremental que completa la especificación de BII-P2P. Se expone su funcionamiento y detalles de implementación. Se presentan las pruebas, se analizan los datos y se exponen los resultados obtenidos.

Finalmente se presentan las conclusiones generales de la tesis y líneas de trabajo futuras.



# Capítulo 1

## Sistemas *Peer-to-Peer* (P2P)

### 1.1 Introducción

En un artículo titulado *What is p2p and what isn't*, Clay Shirky describe a los sistemas P2P como aplicaciones que aprovechan los recursos de almacenamientos, de ciclos de CPU, de contenido e incluso de la presencia humana, disponibles en la periferia de Internet. Debido a que el acceso a estos recursos descentralizados se realiza en entornos de conectividad inestable y direcciones IP impredecibles, los nodos P2P deben operar fuera de los sistemas de DNS y tienen una significativa o absoluta autonomía de los servidores centrales. [11] Esto, según el autor, es lo que hace a las P2P distintivas.

David Anderson, director del proyecto *SETI@Home* [12] denomina a este último y a otros proyectos P2P similares “cliente/servidor invertido” haciendo hincapié en que las computadoras en los bordes son las que proveen el poder de cómputo mientras que las centrales sólo están presentes para coordinarlas.

Estas y otras descripciones como: “Uso de dispositivos en la periferia de Internet” o “Distribución de recursos y servicios de computación por medio del intercambio directo entre sistemas”, son algunas formas que utiliza la comunidad P2P cuando hace referencia a la actividad que ella misma desarrolla [13].

Un sistema *Peer-to-Peer* (P2P), puede definirse como una red de computadoras o equipos, compuesta por pares o iguales que intercambian información actuando como servidores y clientes al mismo tiempo.

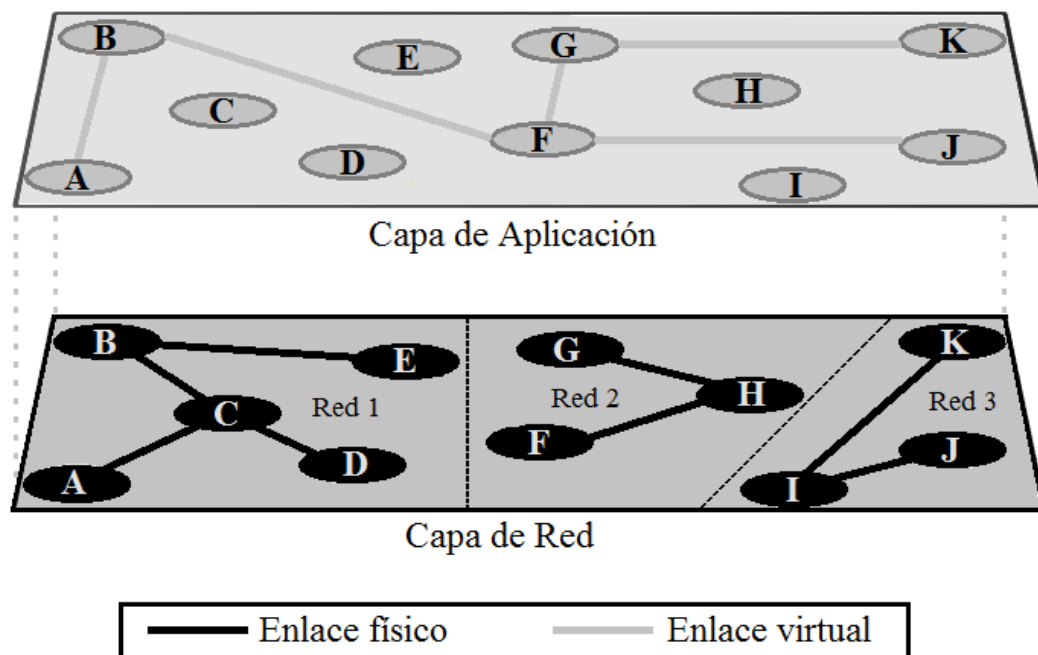
Un sistema P2P, también referido como red P2P, enlaza los recursos de todos los nodos permitiendo que estos recursos sean compartidos sin necesidad de servidores centrales. Los nodos, que también son llamados pares, poseen los mismos roles y responsabilidades e intercambian información o comparten recursos unos con otros. Aunque la utilización más popular de estos sistemas sea la de compartir archivos en la red, en principio pueden ser utilizados para compartir cualquier tipo de recurso entre los nodos integrantes.

Los sistemas P2P pueden funcionar sin ninguna administración central y constituyen sistemas distribuidos que se caracterizan generalmente por su alto grado de dinamismo. Los pares son autónomos en al menos tres sentidos: i) pueden unirse al sistema en cualquier momento, ii) pueden dejar el sistema sin previo aviso y iii) pueden tomar decisiones locales de encaminamiento atendiendo las particularidades de cada caso [14].

Los sistemas P2P constituyen redes virtuales con sus propios mecanismos de ruteo en las que nodos vecinos pueden hallarse a miles de kilómetros de distancia y pertenecer a organizaciones diferentes. En términos del modelo tradicional OSI la interacción P2P se encuentra en la capa de aplicación. Los sistemas P2P son entonces redes superpuestas —*overlay network* en inglés—.

Las redes superpuestas se denominan así por estar construidas sobre otras redes y por tener una arquitectura independiente de las mismas. Constituyen redes virtuales de nodos enlazados lógicamente. En la Figura 1-1 se visualiza un esquema general de una red superpuesta. El objetivo de estas redes es implementar servicios que no están disponibles en las redes físicas subyacentes. Un conjunto de aplicaciones, a veces una misma aplicación, corriendo en los

distintos nodos de la red proporciona a la misma la conectividad y los servicios relacionados con el mantenimiento del sistema.



**Figura 1-1 Red superpuesta.** En la capa de aplicación, los nodos A, B, F, G, J y K conforman una red superpuesta. En la capa de red, las redes 1, 2 y 3 se encuentran conectadas entre ellas directa o indirectamente a través de ruteadores.

En resumen, una red P2P es un sistema distribuido compuesto por participantes que comparten alguna porción de sus recursos como poder de procesamiento, espacio de almacenamiento o ancho de banda, dejándolos disponibles para otros participantes de la red sin necesidad de controles centralizados, servidores ni de integrantes permanentes de la red.

Los sistemas P2P establecen así una forma particular de relación dinámica, basada en el supuesto de la equipotencia de sus participantes, organizándose a

través de la libre cooperación autónoma de los pares en vista de la realización de una tarea común y del bien global [15].

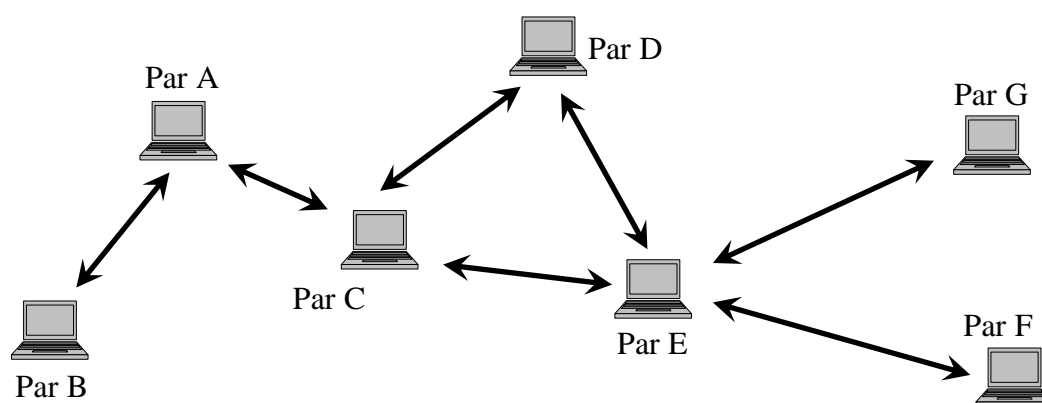


Figura 1-2 Arquitectura básica de una red P2P pura

Los nodos integrantes de la red P2P que se observa en la Figura 1-2 se conectan unos a otros con una relación simétrica, no existiendo jerarquías o roles diferenciados. Este modelo se corresponde a un sistema de interacción P2P puro.

Aunque los sistemas P2P han ganado visibilidad a partir de la aparición de *Napster* [16], —servicio de distribución de archivos de música en formato mp3 creado por Shawn Fanning en el año 1999— y de su pleito con las compañías de música, se han convertido en una técnica importante en diversas áreas tales como la computación distribuida y colaborativa tanto en Internet como en redes locales. Los sistemas P2P han recibido la atención tanto de la industria como de los ambientes académicos.

## 1.2 Características

Los sistemas P2P poseen varias características y potencialidades atractivas. En comparación con los sistemas convencionales centralizados basados en el

esquema cliente/servidor, las redes P2P pueden ofrecer una mejor escalabilidad [17] y tolerancia a fallos [18] [19]. La descentralización, el equilibrio de carga y el anonimato son características distintivas en estos sistemas mientras que la seguridad es un aspecto deseable pero difícil de conseguir.

**ESCALABILIDAD.** Los sistemas P2P se construyen promoviendo que los pares ingresen a la red ofreciendo recursos para compartir, ya sea ancho de banda, espacio de almacenamiento o poder de cómputo. Así, cuando un nuevo nodo ingresa al sistema, si bien crece la demanda global de recursos, también crece la disponibilidad total en toda la red. Esto no ocurre en los sistemas cliente/servidor donde la llegada de un nuevo cliente supone una disminución en la transferencia de datos para todos los usuarios. En este sentido, los sistemas P2P son generalmente más escalables que los sistemas centralizados.

**TOLERANCIA A FALLAS.** La naturaleza distribuida de los sistemas P2P permite construir sistemas robustos tolerante a fallas facilitando el replicado de datos entre muchos pares y, en los sistemas P2P puros, como el caso de *Gnutella*, haciendo posible que los pares encuentren la información buscada sin necesidad de recurrir a servidores centrales de indexación.

Una categoría de sistemas denominados híbridos combinan conexiones P2P con servidores centralizados para indexar el conjunto total de recursos compartidos en la red. Aunque de esta manera se resuelve el problema de la búsqueda de recursos en la red, se crean puntos de fallas sensibles, debilitando el sistema que depende íntegramente del buen funcionamiento de estos pocos servidores.

**DESCENTRALIZACIÓN.** La arquitectura P2P por definición es descentralizada llegando, en el caso de los sistemas puros, al grado que absolutamente todos los nodos son iguales, desempeñan el mismo rol y realizan las mismas funciones. Sin especializaciones de ningún tipo, todos los nodos se auto-organizan conformando enormes sistemas distribuidos que funcionan a partir

de la actividad que desarrollan sus integrantes comunicándose en sus entornos locales.

**EQUILIBRIO DE CARGA.** Debido a que todos los nodos desempeñan el mismo rol y realizan la misma tarea, las cargas se comparten distribuyéndose equilibradamente entre todos los integrantes de la red P2P.

**ANONIMATO.** La protección del anonimato es, según algunos la única forma de garantizar la libertad de expresión. Al menos así lo manifiestan los responsables de *FreeNet* [20], una red P2P creada con el firme propósito de guardar el anonimato de los usuarios que utilizan la red para publicar contenidos. La infraestructura distribuida de una red P2P brinda ciertas facilidades para implementar políticas que dificulten el rastreo de mensajes y garanticen el anonimato.

**SEGURIDAD.** Es una de las características deseables pero de las menos implementadas en los sistemas P2P. Los objetivos que una aplicación P2P segura debería conseguir son: proteger los recursos de la red, identificar y evitar espionaje de las comunicaciones, nodos maliciosos y contenido falso o adulterado entre otros.

### 1.3 Visión histórica

Hacia el año 2000, el modelo de comunicación asimétrica que dominó la red durante los años de enorme crecimiento de la Internet sufrió cambios significativos. A través de la aplicación llamada *Napster* para el intercambio de música y del masivo movimiento que rápidamente se conoció con el nombre *Peer-to-Peer*, millones de usuarios conectados a Internet comenzaron a cambiar la forma de utilizar los recursos que ofrecía la red de redes. Además de navegar por la Web e intercambiar correos electrónicos, las máquinas en el hogar y en las instituciones empezaron a conectarse unas a otras entre sí formando grupos



colaborativos y transformándose en motores de búsquedas, supercomputadoras virtuales y grandes sistemas de archivos [21].

Sin embargo, este acontecimiento es más una vuelta a un modelo un tanto relegado, más que un descubrimiento totalmente novedoso. Si se desea indagar sobre las cuestiones relativas a los orígenes de este paradigma de comunicación, al menos en la era de la Internet, es necesario remontarse a los comienzos de la propia red de redes.

### 1.3.1 El Modelo Peer-to-Peer en los Inicios de la Internet

La Internet original fue fundamentalmente diseñada como un sistema *Peer-to-Peer* hacia finales de los 60'. El objetivo de ARPANET fue compartir recursos de computación alrededor de Estados Unidos. Los primeros integrantes de esta red de redes fueron sitios de cómputo independientes que gozaban todos del mismo status como partes de la red. ARPANET no los conectó de una manera *maestro-esclavo* o *cliente-servidor* sino en una relación de iguales o pares.

La Internet de aquel entonces fue más abierta y libre que la de los días actuales. Los cortafuegos —*Firewalls* en inglés— no se conocieron hasta fines de los 80'. Por lo general, cualquier máquina en la red podía enviar y recibir paquetes desde cualquier otra y aunque las primeras aplicaciones de TELNET —*TELEcommunication NETwork*— y FTP —*File Transfer Protocol*— responden al modelo cliente/servidor, los patrones de uso en su conjunto eran simétricos. En general, cualquier nodo de la red podía hacer un TELNET o FTP a cualquier otro equipo de la misma.

En particular, tanto *Usenet* —*Users Network*— como DNS —*Domain Name System*— introdujeron tempranamente importantes componentes de comunicación *Peer-to-Peer*. *Usenet*, red global de grupos de noticias, implementó un modelo descentralizado de comunicación que de alguna forma representa el antecesor de redes más modernas tales como *Gnutella* y *Freenet*, ambas enroladas

en el paradigma de la comunicación simétrica. Fundamentalmente *Usenet* copia archivos entre computadora sin utilizar ningún control central. Como *Usenet* nació alrededor del año 1979 puede considerarse como un precursor de los sistemas *Peer-to-Peer* más modernos.

Por otro lado, el Sistema de Nombres de Dominio (DNS) es un ejemplo de un sistema que combina *Peer-to-Peer* con un modelo jerárquico para conformar un sistema que ha sido capaz de escalar sin problemas desde los pocos miles de equipos para los que fue diseñado originalmente en 1983 a cientos de millones en la actualidad en Internet. Hay varios elementos claves en el diseño del DNS que son replicados en muchos sistemas distribuidos más modernos. Uno de ellos consiste en que los equipos pueden operar como clientes y servidores al mismo tiempo propagando requerimientos cuando es necesario.

### 1.3.2 La transición a un Modelo Asimétrico en la Internet

El modelo de comunicación simétrica que había predominado en la red de redes por cerca de dos décadas se vio abruptamente transformado en el último lustro del siglo pasado, junto con el advenimiento de la llamada *Internet comercial*. A partir de 1994 la forma de Internet cambió radicalmente convirtiéndose en un medio de comunicación bullicioso donde millones de personas se interesaron por la red como un medio para intercambiar correos y ver páginas web. Este cambio de la Internet a un fenómeno cultural de masas ha impactado con fuerza en la arquitectura de la red, afectando la capacidad para crear comunicaciones *Peer-to-Peer*. Esto se debe a la ruptura de la cooperación en la Red, el creciente despliegue de servidores de seguridad y el crecimiento de los enlaces de red asimétricos como ADSL —*Asymmetric Digital Subscriber Line*— y cable módems.

El modelo de red cambió drásticamente con la llegada de millones de usuarios domésticos. La web y muchas otras aplicaciones que se consolidaron

durante la rápida comercialización de la Internet, se basan en un simple protocolo *cliente-servidor*. Aunque la Web originariamente fue concebida como un sistema que permite a todos los usuarios publicar sus puntos de vista en lugar de simplemente consumir los medios de comunicación, la explosión comercial de Internet de manera rápida cambió la mayoría del tráfico en el paradigma “*downstream*” ya utilizado por la televisión y los periódicos, donde la información se transmite desde unos pocos nodos hacia muchos otros —comunicación asimétrica—.

En los principios de la Internet, todos los equipos participantes eran iguales. La red era simétrica, si un equipo podía llegar a la red, todos los equipos de la red podían llegar hasta él. Cada equipo también podía ser cliente o servidor. Esta capacidad comenzó a debilitarse a mediados de los 90’ con la aparición de los servidores de seguridad —*Firewalls*—, las direcciones IP dinámicas y el protocolo *Network Address Translation* (NAT) que produjeron un serio obstáculo a las comunicaciones de tipo *Peer-to-Peer*. Estas tres tecnologías ayudan a hacer los sistemas escalables y seguros, resolviendo el problema de traer millones de equipos clientes a la Internet de forma rápida y manejable. Sin embargo, han debilitado la infraestructura de Internet en su conjunto, relegando a la mayoría de los equipos a una condición de segunda clase, la condición de sólo cliente.

El ancho de banda asimétrico provisto por las compañías que ofrecen el acceso a Internet, tendencia que se hizo fuerte a finales de la década de 1990, presupo otro obstáculo a las comunicaciones entre pares. Favoreciendo el rol del cliente se provee mayor ancho de banda para recibir datos que para enviarlos. La razón por la que los usuarios toleran esta diferencia es porque la mayoría son sólo clientes de la Web, no servidores.

En [21] se puede hallar un relato con más detalle sobre el papel desempeñado por el modelo *Peer-to-Peer* en la historia de la Internet hasta el año 2001.

### 1.3.3 El Nuevo Auge por los Sistemas Peer-to-Peer

La era moderna de las comunicaciones *Peer-to-Peer* comienza en un clima de agitación a mediados del año 1999 con la irrupción de *Napster* y su rápido conflicto con las compañías discográficas. *Napster* fue creada por un alumno universitario llamado Shawn Fanning que se inspiró en los esfuerzos que sus compañeros realizaban con el *chat* (IRC —*Internet Relay Chat*—) y la web para descargar música en formato mp3. Construyó un programa cliente para los usuarios y un programa servidor controlado por la compañía que fundó llamada *Napster Corporation*. Al acceder al sistema, el cliente de *Napster* subía al servidor la lista de archivos mp3 que compartía. De esta forma, la búsqueda de recursos en la red era realizada de forma centralizada por el servidor de *Napster* que, ante una consulta por parte de un cliente, devolvía la lista de los mejores 100 archivos coincidentes ordenados por una estimación de la distancia más cercana para su descarga. Luego el usuario decidía desde donde descargar el archivo abriendo una conexión directa con el usuario que lo compartía.

*Napster* fue un éxito de inmediato. En los primeros nueve meses luego de su lanzamiento en julio de 1999, consiguió cerca de 10 millones de usuarios, y luego de 18 meses ya contaba con 80 millones de usuarios registrados en el sistema [22]. Sin embargo su legalidad fue rápidamente cuestionada por la industria musical y en diciembre de 1999 la *Recording Industry Association of America* le inició una demanda debido a la que, tras año y medio de batalla legal, *Napster* fue obligada a cerrar sus servidores.

Sin embargo, pese al cierre de los servidores de *Napster*, se había iniciado un movimiento que ya no podría ser frenado, un nuevo y renovado entusiasmo por los sistemas basados en la comunicación entre pares. Rápidamente tanto la industria como la academia avizoraron la amplia gama de posibilidades que este modelo prometía. No se trata tan sólo de compartir archivos, sino de crear

ambientes colaborativos —*Microsoft SharePoint Workspace*<sup>1</sup>—, resguardar el anonimato y la confidencialidad al difundir información —*FreeNet* y redes anónimas—, y compartir otro tipo de recursos como ciclos de CPU —*Docking@Home*, *Einstein@home*, *Climateprediction.net*—, ancho de banda —*TPLive*, *Skype*, *SopCast*— etc.

### 1.3.4 Hitos Históricos

Aunque la evolución de la tecnología *Peer-to-Peer* no siga una trayectoria lineal hacia formas más eficientes, masivas u horizontales de compartir recursos en la Internet, a continuación se mencionan algunos de los sistemas más relevantes según la perspectiva histórica. El objetivo de este apartado es brindar una visión sobre el proceso de consolidación definitiva de los sistemas *Peer-to-Peer* en la Internet y de cómo distintos dominios de aplicación fueron alcanzados por esta tecnología.

#### 1.3.4.1 Protocolo FastTrack y Redes Kazaa, Grokster e iMesh

En 1999 se desarrolló el protocolo *FastTrack* con dos ventajas respecto de *Napster*: 1) el intercambio no estaba restringido a archivos mp3 y 2) se podía reanudar una descarga interrumpida localizando otro par con el mismo archivo.

Aunque no se conoce con certeza los detalles de este protocolo puesto que se desarrolló en el seno de una empresa privada, se sabe que está basado en dos tipos de nodos: nodos estándar y supernodos y que, a diferencia de *Napster*, al carecer de un control central la red es difícil de clausurar puesto que los pares trabajan con independencia de cualquier servidor central.

Utilizando este protocolo que fue vendido a distintas empresas se crearon las redes *Kazaa*, *Grokster* e *iMesh* [23] que terminaron siendo incompatibles entre sí.

---

<sup>1</sup> Anteriormente conocido como *Microsoft Office Groove*

Las redes *FastTrack* alcanzaron gran popularidad concentrando el 76% del tráfico P2P en la Internet durante los años 2003 y 2004. Aunque menos vulnerables que *Napster* también fueron demandadas por la industria en los Estados Unidos.

#### 1.3.4.2 Gnutella 0.4

*Gnutella* [24] nació como un pequeño proyecto en una compañía llamada Nullsoft a principios del año 2000. Su principal innovación fue la ausencia total de control central. Esta diferencia con *Napster* es fundamental y hace difícil o quizás imposible desactivarla tal como se hiciera con la red creada por Shawn Fanning cuando fue obligada a cerrar sus servidores.

*Gnutella* es una red P2P pura, totalmente descentralizada y no estructurada dónde sólo existe un único rol en todo el sistema: cliente y servidor al mismo tiempo, por ello a los nodos de esta red se los conoció con el nombre de *servents* —conjunción de las palabras *servers* y *clients*—.

Un *servent* se une a la red conectándose con otros *servents* conformando así una verdadera red de pares. La búsqueda de un recurso en esta red se hace por medio de una estrategia conocida como inundación —*flooding* en inglés—. Un nodo envía su solicitud a todos sus vecinos con los que comparte una conexión TCP/IP directa. Éstos a su vez, la replican a todos sus otros vecinos. Si un nodo posee el contenido solicitado, envía siguiendo la ruta inversa un mensaje de respuesta con sus datos de contacto. El intercambio final entre el nodo que inició la solicitud y aquel que posee el recurso buscado se realiza fuera de la red *Gnutella* utilizando una conexión directa por medio del protocolo HTTP.

La decisión de enviar los mensajes de búsqueda hacia delante y las respuestas hacia atrás por el mismo camino en la red fue una decisión deliberada para proteger el anonimato evitando colocar la dirección del remitente que originó la consulta. De esta forma, el espionaje corporativo se hace más difícil y también la

posibilidad de emprender acciones legales [25]. Claramente el conflicto de la industria discográfica con la compañía *Napster* influyó significativamente en la evolución de la tecnología P2P.

La búsqueda de recursos implementada por *Gnutella* no es escalable y cuando el número de equipos conectados creció, disminuyó el rendimiento global y la experiencia del usuario. Algunos equipos conectados a la red por medio de modem telefónico utilizaban todo su ancho de banda sólo en participar en el mecanismo de búsqueda. Para controlar el tráfico generado por los paquetes de búsqueda fue necesario limitar el alcance de las consultas con un campo *time to live* (TTL) con una consecuencia negativa: la búsqueda ya no alcanza todos los nodos de la red y por lo tanto no existe garantía de encontrar un recurso existente si es compartido por algún nodo situado más allá del TTL utilizado.

En el próximo capítulo se presentan más detalles de *Gnutella* 0.4 —Ver sección 3.1 “Con Base en *Gnutella*” en la página 88— ya que el presente trabajo toma como punto de partida esta versión de *Gnutella* para implementar un mecanismo de búsqueda inteligente, eficiente y escalable.

#### 1.3.4.3 eDonkey

*eDonkey* fue creado por Jed McCaleb y liberado el 16 de septiembre del 2000. En esta red, a diferencia de *Napster*, los clientes podían descargarse distintas partes de un archivo desde diferentes pares en forma simultánea. A esta técnica se la conoce con el nombre de *swarming*. Esto permite aumentar la velocidad de descarga que ahora no está sujeta al ancho de banda de un único nodo, sino que se combinan los anchos de bandas de varios nodos desde los que se está transfiriendo alguna parte del archivo. Una vez conseguidas todas las piezas, que pueden gestionarse como recursos independientes, en el cliente que inició la solicitud se ensamblan obteniendo así el archivo completo.

Utilizando *swarming* es posible acceder a contenidos aún cuando uno de los

proveedores del archivo hubiese apagado o desconectado su equipo. Además permite mejorar la velocidad de descarga puesto que en una conexión estándar entre dos usuarios es probable que el proveedor del archivo tenga una limitación en la velocidad de subida, ya sea debido a la asimetría de la tecnología ADSL o simplemente por las restricciones impuestas por su ISP —*Internet Server Provider*—. Mediante *swarming* la descarga puede combinar varios proveedores, cada uno a su velocidad máxima hasta alcanzar la velocidad máxima de descarga del cliente. Por lo tanto se hace más fácil descargar archivos de gran tamaño, comúnmente denominados “archivos pesados” y de allí proviene el nombre *donkey* que significa burro en inglés, haciendo referencia a la posibilidad de trasladar cargas pesadas.

Se pensó a la red como clientes y servidores al estilo de *Napster*, aunque se proponía utilizar una serie de servidores independientes en lugar de un único servidor central como *Napster*. Lo que no previó McCaleb fue que muchos operadores conectarían sus servidores entre sí, para crear una red de gran tamaño que iba a ser conocida como *eDonkey2000*, y que transformó la arquitectura de la red en algo similar a *Gnutella* o a *FastTrack* 0.6. [25].

#### 1.3.4.4 Gnutella 0.6

A pesar del interés por la comunidad de *Gnutella* de mantener una red completamente descentralizada, debido al problema del escalado deficiente hubo un corrimiento hacia una estructura más jerárquica.

La versión 0.6 del protocolo *Gnutella* [26] que se introdujo en 2002 distinguió entre dos tipos de pares o usuarios: los nodos hoja y los nodos *superpares*. Los nodos hojas sólo podían conectarse con uno o más *superpares*, mientras que los *superpares* podían tener conexiones tanto con los nodos hojas como con otros *superpares* quienes son los responsables de encaminar las consultas y mantener viva la red.



En esta versión del protocolo tampoco existe un control central, algo que sus creadores diseñaron meticulosamente con el fin de evitar acciones legales como había sucedido con *Napster*. Sin embargo, al existir una jerarquía de dos niveles, escala mejor que *Gnutella* 0.4 puesto que los *superpares* pueden concentrar gran número de nodos hojas. Así puede existir una cantidad mucho menor de *superpares* que de nodos hojas y los *superpares* son quienes se conectan entre sí como lo hacen los nodos en *Gnutella* 0.4.

#### 1.3.4.5 Redes Estructuradas. Chord, Pastry, Tapestry, CAN

En el año 2001, un enfoque totalmente distinto de soluciones al problema de la búsqueda de recursos en redes P2P totalmente descentralizadas se gestó en las universidades y otras instituciones como producto de la actividad de investigación realizada en esta área que ya había captado la atención del público en general.

El rasgo característico de redes como *Pastry* [27], *Tapestry* [28], *CAN* [29] y *Chord* [30] es que los archivos se almacenan en la red de forma predecible guardando la correspondencia entre éstos y los nodos que los contienen en una tabla de *hash* distribuida (DHT por sus siglas en inglés) entre todos los nodos de la red.

Aunque estos sistemas, a diferencia de *Gnutella*, garantizan que si el recurso existe en la red éste será efectivamente hallado, la complejidad de las mismas y de la gestión necesaria para manejar el comportamiento dinámico de los usuarios que entran o dejan la red muy frecuentemente hizo que este tipo de red no llegara a ser muy popular. Sin embargo algunas de sus técnicas fueron incorporadas más tarde por otras redes P2P como *Kademlia* [31], que alcanzó cierto éxito incorporándose a *eMule*, un cliente libre que también participa en la red *eDonkey* 2000 [32]. El cliente *eMule* fue creado en un principio como alternativa gratuita al programa *eDonkey* y en poco tiempo lo superó en funciones y popularidad. Los desarrolladores de *eMule* crearon la red *Kad* que utiliza una variante del protocolo

*Kademlia.*

#### **1.3.4.6 BitTorrent**

En sus comienzos, en el año 2001, *BitTorrent* [33] era un proyecto de código abierto no comercial que se popularizó rápidamente entre los usuarios de Linux pues se utilizó, y aún se utiliza para compartir y descargar las distribuciones de este sistema operativo. Se ha estimado que en la primera mitad del año 2004 el tráfico *BitTorrent* aumentó del 26% al 53% mientras que *FastTrack* descendió del 46% al 19%. En un estudio presentado en [34] sobre datos de tráfico en internet recogidos en julio de 2010 señala a *BitTorrent* como el protocolo de intercambio P2P con mayor transferencia durante los 14 días que duró el monitoreo. Para el año 2011 BitTorrent se convirtió en la red P2P más utilizada en el mundo constituyendo el 94% de todo el tráfico P2P generado y con una cantidad de usuarios que superó los 100 millones [35].

La idea subyacente en *BitTorrent* es la de compartir y publicar, más que la de descargar, por lo tanto queda fuera del protocolo todo aquello que no coincida con este principio. Incluso no proporciona ningún mecanismo de búsqueda que pueda atraer a los descargadores. Los usuarios deben contactar con un rastreador —*tracker*— que proporciona información sobre la ubicación de un archivo específico y sobre otros pares que lo están descargando también. La ubicación del rastreador y los contenidos que alberga se gestionan al margen del protocolo generalmente a partir de la web, descargando un archivo de extensión *.torrent* que utilizan los clientes de este protocolo para poder contactarse con uno o más *trackers*.

Una de las grandes ventajas de este modo de operación es el alto nivel de integridad debido al trabajo de sus componentes centralizados. Sin embargo, esto tiene un precio: la disponibilidad del sistema se ve obstaculizada por la naturaleza centralizada de estos componentes. La descentralización proporcionaría una

solución obvia pero disminuiría la integridad como sucede en otras redes P2P donde abundan los archivos falsos [36].

## 1.4 Tipos de Redes P2P

Los sistemas P2P se categorizan en puros e híbridos, y los puros se subdividen en estructurados y sin estructura. La figura siguiente muestra esta categorización

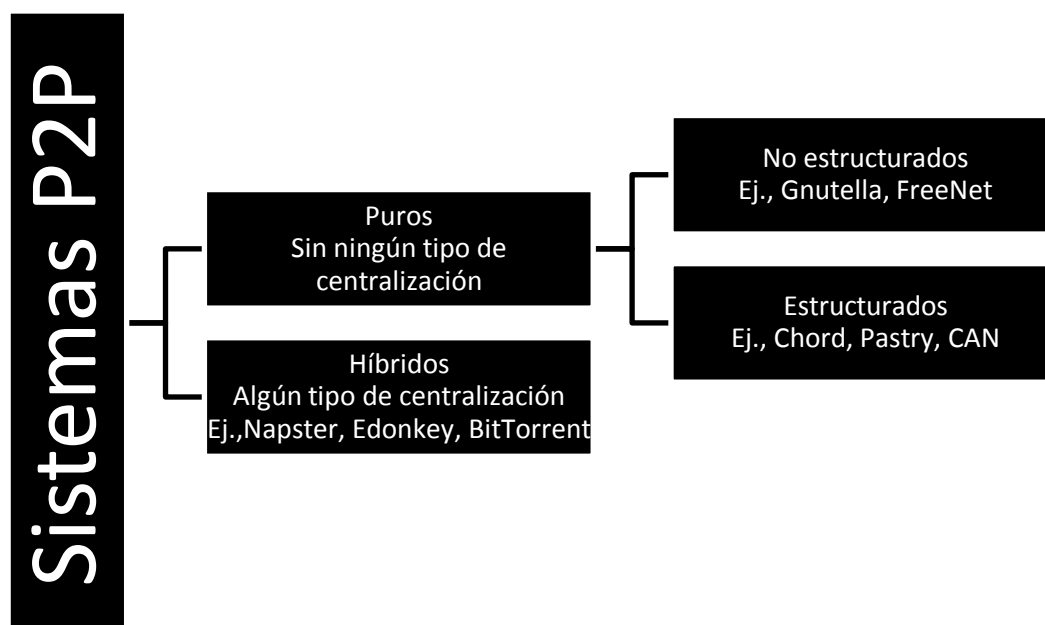


Figura 1-3 Clasificación de los sistemas P2P

### 1.4.1 Sistemas P2P Puros no Estructurados

Los sistemas P2P puros no estructurados conforman redes completamente descentralizadas donde no existen servidores centrales ni roles especializados. Todos los integrantes de la red poseen el mismo *status* y cumplen las mismas

funciones conectándose unos con otros en una relación simétrica como pares o iguales. Estos sistemas permiten la creación de entornos dinámicos donde los pares entran y salen de la red con mucha frecuencia debido a que la carga de trabajo que presuponen estas tareas no es considerable.

Históricamente, la primera red P2P diseñada como un sistema P2P puro no estructurado fue *Gnutella* 0.4 [24] que se implementó con ausencia total de control central. Esta característica, a diferencia de *Napster* la convirtió en una red difícil o quizás imposible de desactivar. *Freenet* [20] es otra red P2P que se encuentra en esta categoría y que fuera diseñada con el objetivo de proporcionar libertad de expresión mediante una fuerte protección del anonimato. La estructura de una red P2P totalmente descentralizada sumado al cifrado de todas las comunicaciones entre los pares de *Freenet* hacen extremadamente difícil rastrear quién está solicitando la información y cuál es su contenido.

Cada par en una red P2P pura actúa como cliente y servidor al mismo tiempo en relación con otros pares vecinos con los que mantiene una conexión directa en la red *overlay*. El gráfico presentado en la Figura 1-2 se corresponde con este modelo de red. Los nodos sólo conocen a aquellos nodos con los cuales están conectados directamente. En principio todos los demás nodos de la red son desconocidos. La comunicación se realiza por medio de la difusión de mensajes a los vecinos directos. Los pares transportan el tráfico enviando y recibiendo peticiones de recursos y respuestas a dichas peticiones junto a algunos mensajes de control que facilitan el descubrimiento de otros nodos.

La búsqueda de recursos en este tipo de redes debe resolverse de manera totalmente descentralizada. Para ello *Gnutella* 0.4 implementa un algoritmo BFS —siglas extraídas de su nombre en inglés: *Breadth First Search*— para realizar difusiones de peticiones que inundan la red —*flooding* en inglés— llegando a un gran número de pares. Un nodo que desea localizar un determinado recurso envía una petición a todos sus nodos vecinos, quienes a su vez la reenvían a todos sus

otros vecinos alcanzando así nodos más alejados.

Una importante desventaja de este esquema de búsqueda es la generación de una gran cantidad de tráfico por la difusión de las peticiones. La solución habitual en este tipo de redes consiste en utilizar un campo TTL —*time to live*— en la cabecera de los mensajes que es decrementado conforme las peticiones atraviesan la red. Cuando el TTL de una petición llega a cero la misma se descarta. Esto evita el colapso de la red a costa de limitar la propagación de las solicitudes convirtiendo en inalcanzables a aquellos nodos que se encuentren a mayor distancia del radio impuesto por este valor.

Entre las ventajas más destacadas de esta categoría de sistemas P2P se encuentra la robustez y tolerancia a fallas, que se ve incrementada por la naturaleza distribuida, la total ausencia de control central y la no existencia de roles diferenciados. De esta forma, ningún nodo es imprescindible para el funcionamiento de la red que no tiene puntos de fallas sensibles.

Otra ventaja deseable en cualquier sistema distribuido y presente en las redes P2P puras no estructuradas es el equilibrio de carga entre los nodos. Al no existir tareas diferenciadas todos los pares se ven sometidos a iguales exigencias de trabajo.

La sencillez general del sistema es otra característica inherente al modelo P2P puro no estructurado. Esta simplicidad se debe sobre todo a la ausencia de control central que evita la necesidad de cualquier tipo de administración global. Esta simplicidad se extiende también al protocolo que los pares deben realizar para ingresar o salir de la red.

La facilidad para implementar mecanismos tendientes a mantener el anonimato como lo hace la red *FreeNet* también es una característica bien soportada por estos sistemas P2P y muy deseable en ciertas comunidades que sostienen que es imposible tener libertad de expresión sin la opción del anonimato

[20].

Aunque la escalabilidad suele ser una ventaja presente en las redes P2P debido al incremento en la cantidad de recursos compartidos que se produce con el ingreso de un nuevo integrante a la red, no lo es en el caso de los sistemas P2P puros sin estructura. Ello se debe al excesivo tráfico que genera el sistema de búsqueda de recursos en su esfuerzo por alcanzar la mayor cantidad de pares de la red. Por citar un ejemplo, *Gnutella* no escala bien hasta los niveles que hoy en día demandan las aplicaciones de *file-sharing* en Internet para las cuales este protocolo ha sido diseñado.

Para un tamaño de red considerable, con gran cantidad de pares, la búsqueda de un recurso escaso puede resultar infructuosa debido a que el nodo que lo posee puede encontrarse a una distancia mayor al valor establecido por el campo TTL respecto del nodo que inicia la petición. Un enfoque totalmente distinto para resolver estos problemas relacionados con el tráfico generado y la eficiencia de las búsquedas lo constituyen las redes P2P estructuradas que se presentan a continuación.

#### 1.4.2 Sistemas P2P Puros Estructurados

Las redes P2P puras estructuradas superan las dificultades relacionadas con la búsqueda de recursos, mencionadas en la sección anterior, manteniendo el índice de los recursos compartidos en tablas de *hash* distribuidas (DHT por sus siglas en inglés) haciendo que cada par se haga responsable de mantener actualizada una porción del índice total de los recursos compartidos. *Chord* [30], CAN, *Tapestry* [28], *Pastry* [27] y *Kademlia* [31] son ejemplos de esta categoría de sistemas P2P.

En estas redes se utilizan funciones de *hash* para asignar valores a cada recurso y siguen un protocolo para determinar cuál será el par responsable de mantener la información de la locación del recurso. De igual manera, utilizando la misma función de *hash*, cuando un par desea buscar un recurso puede determinar

cuál es el nodo de la red que posee la información de su localización.

Una DHT es una tabla de *hash* utilizada para guardar la localización de todos los recursos compartidos en la red. Por localización del recurso entiéndase la dirección del nodo que lo contiene. Esta tabla no reside físicamente en ningún punto particular de la red sino que se encuentra distribuida entre todos los nodos del sistema P2P. Los nodos se comportan como pares con idénticos roles en el sistema. Todos poseen la responsabilidad de mantener una porción de la tabla de *hash* y resolver las consultas de otros pares en relación a los datos contenidos en dicha porción.

Una tabla de *hash* guarda pares de la forma (*clave*, *valor*). Aquí la clave se corresponde con un recurso y el valor con su localización. Los recursos en el sistema se identifican a partir de su clave obtenida por medio de una función de *hash* que es única en todo el sistema. Sea  $h$  dicha función y  $r$  un recurso dado, la expresión  $h(r)$  denota la clave asociada al recurso  $r$ . Sea  $T$  la tabla de *hash* distribuida del sistema.  $T$  almacena pares de la forma  $(h(r), d(r))$  siendo  $d(r)$  la dirección del nodo que contiene al recurso  $r$ .

El espacio de las claves en una tabla de *hash* está asociado al conjunto de posiciones de la tabla, es decir que una clave representa un índice de la tabla. Si la tabla  $T$  contiene el par  $(h(r), d(r))$  es común referirse a ello expresando que  $T$  contiene el valor  $d(r)$  en la posición  $h(r)$ .

Cuando un nodo ingresa a la red se incorpora a una estructura en forma de anillo, asumiendo la tarea de mantener una porción de la tabla  $T$ , es decir un conjunto de valores para un determinado espacio de claves, de acuerdo a su ubicación relativa en la estructura. El anillo está formado por los nodos ordenados de acuerdo al número ID de cada uno.

El ID de cada nodo es un entero de  $k$  bits elegido de manera aleatoria. Para evitar probables colisiones  $k$  debe ser suficientemente grande, por lo general

$k=64$ . El valor de  $k$  también representa el tamaño en bits de las claves que genera la función *hash* del sistema. Debido a que el espacio de claves —índices de la tabla  $T$ — y los números ID de los nodos se definen en el mismo rango:  $0, 1, \dots, 2^k-1$  es posible particionar fácilmente la tabla  $T$  entre los nodos del anillo. Cada nodo se hace cargo de mantener el espacio de claves de la tabla mayores al ID de su nodo antecesor y menores o iguales a su propio ID.

Por simplicidad supóngase  $k=6$ . El espacio de claves de la tabla  $T$  es  $0, 1, \dots, 63$ . Los ID de los nodos también se definen en el rango  $0, 1, \dots, 63$ . Supóngase que en la red se encuentran participando los nodos 5, 23, 33, 40 y 50 —ver Figura 1-4—. El espacio de claves de la tabla  $T$  se distribuye entre los nodos de la siguiente forma: al nodo 5 se asignan las claves  $51, \dots, 63, 0, \dots, 5$ ; al nodo 23 las claves  $6, \dots, 23$ ; al nodo 33 las claves  $24, \dots, 33$ ; al nodo 40 las claves  $34, \dots, 40$  y al nodo 50 las claves  $41, \dots, 50$ .

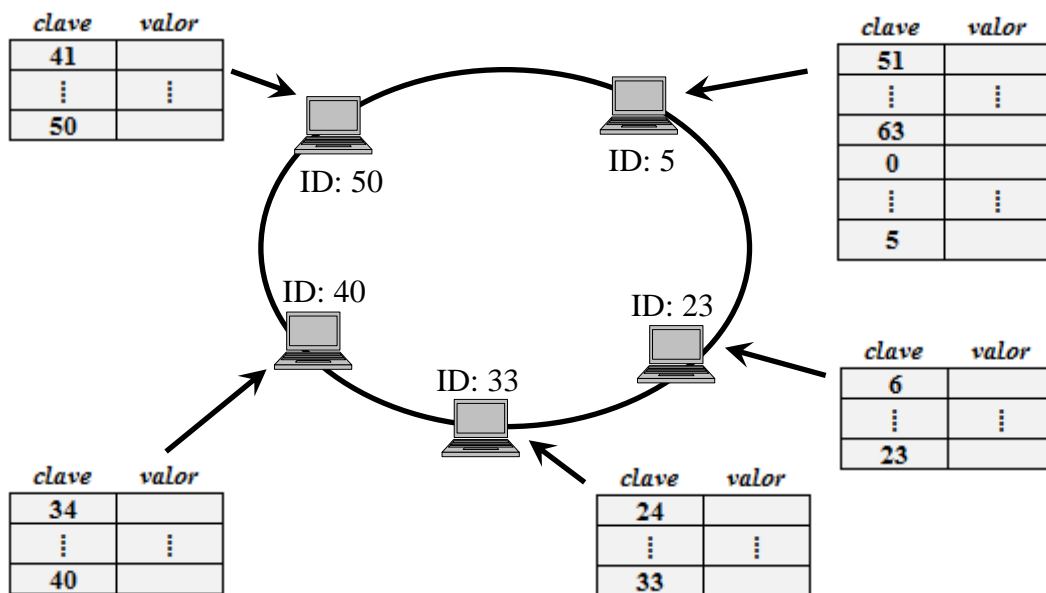


Figura 1-4 Estructura de anillo de una red P2P basada en DHT. Distribución de la tabla de hash entre los nodos integrantes para un espacio de  $2^6$  claves.



Supóngase que se desea guardar en la tabla  $T$  la información de localización para el recurso  $r$ . Se procede de la siguiente forma: *i*) se aplica la función de *hash* sobre  $r$  para obtener la clave  $h(r)$  del recurso; *ii*) se localiza el nodo responsable del espacio de claves al que pertenece  $h(r)$  y *iii*) se envía a dicho nodo el par  $(h(r), d(r))$  para su almacenamiento. Para la búsqueda de un recurso el procedimiento es similar, salvo que en el punto *iii*) se recibe  $d(r)$ , es decir la dirección del nodo que posee el recurso buscado  $r$ .

Supóngase un esquema simplificado en donde los integrantes del anillo mantienen una única referencia al nodo sucesor en la estructura. La localización del nodo que contiene información sobre la clave buscada puede realizarse fácilmente de manera colaborativa. El nodo que recibe una petición de búsqueda con una clave  $h(r)$  que no pertenece a su espacio de claves asignado, simplemente reenvía la petición al nodo siguiente en el anillo. Así, en algún momento la petición alcanzará al nodo indicado que responderá con el valor  $d(r)$  almacenado o bien informará que el recurso buscado no existe en el sistema.

En el esquema simplificado del párrafo anterior puede ocurrir que una petición de búsqueda deba visitar todos los nodos de la red. Esta situación se resuelve agregando más referencias a otros nodos de la red. Así los nodos mantienen una tabla con punteros a  $k$  sucesores — $k$  es el número de bits de los números ID—. Para abarcar gran parte del anillo la distancia a estos nodos sucesores se incrementa exponencialmente —el nodo cuyo ID es  $i$ , guarda las referencias a los nodos cuyos ID son  $(i+2^0) \bmod (2^k)$ ,  $(i+2^1) \bmod (2^k)$ ,  $(i+2^2) \bmod (2^k)$ , ...,  $(i+2^{k-1}) \bmod (2^k)$ —.

La Figura 1-5 muestra un ejemplo donde se visualiza cómo se forma la tabla de sucesores para el nodo 8 y cuántos saltos debe realizar una petición de búsqueda para la clave 54 hasta alcanzar el nodo 56 responsable de mantener la información para la clave buscada.

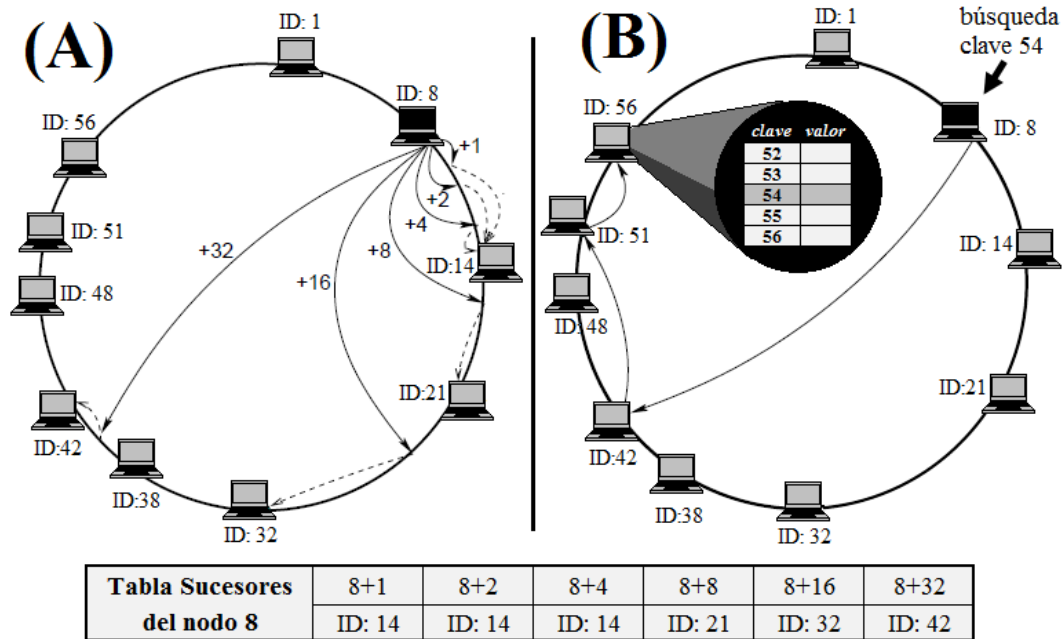


Figura 1-5 (a) Tabla de sucesores para el nodo 8. (b) Camino de petición para la clave 54 iniciada en el nodo 8

Cuando se busca un recurso  $r$ , hay que decidir cuál de los  $k$  nodos apuntados en la tabla se debe elegir para el siguiente salto. Para ello se selecciona aquel nodo cuyo ID tenga la menor distancia al valor  $h(r)$ . Puede probarse que la cantidad de saltos necesarios para resolver la búsqueda pertenece a  $O(\log N)$  siendo  $N$  la cantidad de nodos de la red.

Para mantener la estructura funcionando los nodos deben seguir un determinado protocolo cuando desean unirse o dejar la red. El primer paso para el ingreso al sistema es buscar el sucesor del nuevo nodo de acuerdo a su ID utilizando el mismo protocolo usado para dirigir las peticiones de búsqueda ya descrito. El nuevo nodo debe insertarse entre el sucesor hallado y el antecesor. Para ello recibe el espacio de claves asignado desde su sucesor —todas las claves que el sucesor posee menores o iguales al ID del nuevo nodo—. Para garantizar el

correcto funcionamiento no se cambian los punteros hasta que se hayan copiado todas las claves que le corresponden al nuevo nodo.

La salida voluntaria de un nodo del sistema también implica una carga de trabajo a realizar. El nodo que deja la red transfiere todas sus claves al nodo sucesor y notifica a su predecesor que deja la red para que este último reconstruya su tabla de sucesores. Sin embargo, en ocasiones un nodo se cae y el protocolo de salida nunca se lleva a cabo. En este caso el predecesor del nodo caído deberá iniciar un proceso de actualización para volver a establecer el anillo.

Si las tablas de sucesores de los nodos no están actualizadas el sistema puede seguir funcionando pero pierde eficiencia. Para mantener las tablas de sucesores actualizadas es necesario realizar tareas de mantenimiento. *Chord* utiliza un protocolo de estabilización que cada nodo corre periódicamente en segundo plano para actualizar su tabla de sucesores.

La ventaja de los sistemas P2P puros estructurados se hace evidente cuando la disponibilidad del recurso buscado es reducida, ya que aún en el caso de existir una sola instancia del mismo en algún rincón de la red, será encontrado rápidamente con escasa cantidad de tráfico generado. Sin embargo, para mantener ese alto grado de eficiencia, el sistema debe encontrarse estable y actualizado. Si por el contrario existen muchas instancias del recurso buscado, esta ventaja en relación a otros tipos de redes P2P mucho más simples y fáciles de mantener, como los sistemas puros no estructurados, se hace mucho menos notoria.

Por otro lado, la transitoriedad de los nodos, tan característica de las redes P2P actuales, que entran y salen de la red con mucha frecuencia en períodos cortos de tiempo, ocasionan una importante carga de trabajo asociada al mantenimiento de las tablas de sucesores y al espacio de claves asignado a cada nodo.

Otra limitación de las redes estructuradas basadas en DHT está en la

imposibilidad de realizar consultas complejas sobre los recursos indizados. Obsérvese que la tabla de *hash*  $T$  no almacena el valor  $r$  sino su clave  $h(r)$ . Una clave de *hash* no da por sí misma ningún tipo de información relevante, sólo se utiliza para comparar con otra clave de *hash*. Considérese el ejemplo de un sistema P2P utilizado para compartir archivos. Lo usual en este caso es que  $r$  sea igual al nombre del archivo. Sin embargo  $h(r)$  es un número de  $k$  bits a partir del cual no es posible volver a obtener  $r$ . Por lo tanto aún teniendo acceso al conjunto de claves almacenado no es posible conocer los nombres de los archivos que dieron origen a estas claves. Por lo tanto, estos sistemas sólo admiten consultas simples por el nombre completo del archivo, excluyendo otras más interesantes como podría ser la búsqueda de archivos cuyos nombres contengan una determinada cadena de caracteres.

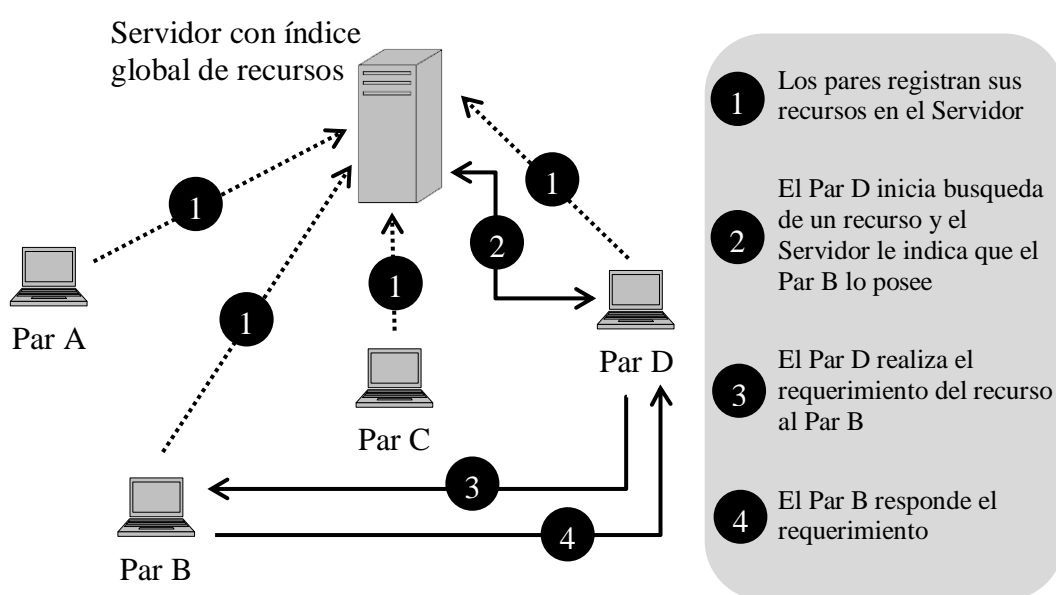
Dado los inconvenientes mencionados en los párrafos anteriores no es difícil entender por qué estos tipos de redes nacidas en entornos académicos nunca alcanzaron la popularidad que sí lograron otros sistemas más simples.

### 1.4.3 Sistemas P2P Híbridos

En esta clase de sistemas P2P existe algún tipo de control central que lleva el inventario del conjunto total de recursos compartidos en la red. Generalmente se trata de un servidor al que deben reportarse todos los nodos del sistema. Cuando un par ingresa a la red informa este servidor cuáles son los recursos que posee para compartir. Cualquier cambio relacionado con la disponibilidad de estos recursos también tiene que ser informado para garantizar que la base de datos en el servidor central se mantenga siempre actualizada.

El mecanismo de búsqueda implementado en los sistemas híbridos es sencillo y la misma se resuelve rápidamente. Si un nodo desea localizar algún recurso en el sistema simplemente envía un requerimiento al servidor central. El servidor, luego de buscar en su base de datos envía al solicitante la lista de los pares que

tienen disponibles el recurso buscado. Una vez que el nodo solicitante obtiene esta información, se comunica en forma directa con uno o más nodos de la lista recibida para gestionar la utilización del recurso. En caso de un sistema P2P de *file sharing* esta gestión simplemente implica la transferencia de un archivo.



**Figura 1-6 Sistema P2P híbrido basado en el uso de un índice global de recursos**

Se dice que estos sistemas P2P son híbridos porque, aunque utilicen un servidor central, éste sólo presta servicios para implementar el mecanismo de búsqueda de recursos en la red. Todos los otros nodos del sistema se conectan unos con otros como pares idénticos, de manera simétrica sin roles diferenciados.

Estos sistemas no son muy escalables debido a la limitación computacional, capacidad y ancho de banda relacionados con el servidor de búsqueda. Además presentan una pobre tolerancia a fallas por la falta de replicación de objetos, y el

desbalanceo de carga. En general su confiabilidad es baja pues poseen un simple punto de falla que puede ser susceptible de ataques malintencionados, o bajar su rendimiento debido a congestiones cerca del servidor.

La ventaja de estos sistemas es que la búsqueda de recursos se resuelve relativamente fácil con la utilización del servidor central que mantiene toda la información relevante sobre los recursos compartidos en la red.

A este tipo de redes P2P híbridas que utilizan un índice global para localizar recursos suele llamárselas también sistemas P2P centralizados porque han perdido la capacidad de trabajar en ausencia de algún servidor central. Las redes *eDonkey* [32] y *BitTorrent* [33] constituyen ejemplos de esta categoría.

Para mejorar la robustez y el rendimiento en los sistemas P2P híbridos, se suelen utilizar múltiples servidores centrales con datos replicados. Así se consigue que el funcionamiento de toda la red no dependa de un único punto de falla. Además, de esta manera se descongestiona el acceso a un único servidor repartiendo el tráfico entre varios mejorando el rendimiento del sistema al reducir los tiempos de respuesta en las consultas a los servidores.

Otro tipo de redes P2P híbridas distribuyen sus nodos en dos grupos: nodos hojas y supernodos. Los supernodos se conectan con otros supernodos conformando una red P2P pura sin estructura entre ellos. Los nodos hojas se conectan a los supernodos enviándoles la lista de los recursos que comparten. Los supernodos concentran la información relacionada con los recursos compartidos por todos sus nodos hojas. Esta división entre los nodos hojas y supernodos se hace con el fin de abordar los problemas de escala en las redes P2P puras disminuyendo el problema de la inundación de la red con paquetes de búsqueda. Un ejemplo de este tipo de red P2P es la red *Gnutella* 0.6 [26] que también suele denominarse *Gnutella 2*— para distinguirla de *Gnutella* 0.4 [24] que simplemente suele referenciarse como *Gnutella*.

## 1.5 Aplicaciones

Sin lugar a dudas las aplicaciones de *file-sharing* en Internet ocupan un lugar destacado entre las múltiples formas de aprovechar los beneficios ofrecidos por un sistema P2P. *Napster*, *FastTrack*, *Kazaa*, *Grokster*, *iMesh*, *Gnutella*, *eDonkey*, *Ares Galaxy* [37], *Kademlia* y *BitTorrent* son algunas de las redes que se han utilizado o todavía se utilizan para el intercambio de archivos en Internet.

La utilización de la tecnología P2P en aplicaciones de *file sharing* sigue resultando de interés para la comunidad científica que continúa realizando publicaciones en el área. Sólo por citar algunos ejemplos: en [38] se reportan los resultados de un estudio de monitoreo sobre la red *Gnutella* informando topología y estado de recursos compartidos; en [39] se presenta un algoritmo eficiente para la replicación de archivos en una red P2P dedicada al *file sharing*; en [40] se presenta un modelo de red P2P para compartir archivos multimedia en el que se pueden realizar controles discriminando pares autorizados y no autorizados; en [41] se propone un mecanismo que tiene en cuenta a los dispositivos móviles que participan en un sistema P2P de *file sharing* priorizando aquellos con baja carga de batería mejorando significativamente la posibilidad de que el cliente complete la descarga del archivo antes de agotar la batería.

Compartir espacio de almacenamiento ha sido por mucho tiempo la utilización más representativa de las redes P2P. Sin embargo de acuerdo al último reporte publicado por la empresa canadiense Sandvine [42], “*Global Internet Phenomena 2h 2013*”, en la segunda mitad del año 2013 las aplicaciones P2P de *file sharing* representan menos del 10% del total diario de tráfico generado en América del Norte, mientras que 5 años atrás representaban más del 31%.

Pero las redes P2P no son patrimonio exclusivo de las aplicaciones de *file sharing*. Lejos de ello, existen muchos escenarios capaces de aprovechar la infraestructura ofrecida por estos sistemas distribuidos para compartir otra clase

de recursos. Ejemplos de estos escenarios lo constituyen aplicaciones como *PPLive* [43], *PPStream* [44], *SopCast* [45] [46] y *Tvants* [47] utilizadas para la transmisión de señales de televisión en directo. Para hacer referencia a aplicaciones diseñadas para redistribuir flujos —*streaming* en inglés— de video, típicamente cadenas de televisión, por todo el mundo sobre una red P2P se utiliza el término P2PTV —*Peer to Peer TeleVision*—.

Por medio de las aplicaciones P2PTV se comparte ancho de banda. Cada nodo de la red que recibe un flujo de video, ayuda a su distribución reenviando la señal a otros nodos de la red. Claramente este modelo tiene grandes ventajas respecto a la difusión tradicional en la que un único servidor envía la señal replicándola a cada uno de los clientes que la consumen. P2PTV permite que una señal de video sea transmitida a un mayor número de usuarios de manera fluida, y sin la necesidad de disponer de gran cantidad de ancho de banda.

Gran cantidad de las publicaciones recientes en el área de las redes P2P están referidas a temas concernientes a P2PTV. Algunas proponen mejoras de rendimiento general del sistema P2PTV [48] otras se centran en la transmisión de video P2P para dispositivos móviles [49], y muchas otras se ocupan de sistemas P2P-VoD —*Peer to Peer Video on Demand*— [50] [51] [52] [53].

Estrechamente relacionadas con las aplicaciones anteriores se encuentran aquellas que utilizan las redes P2P para transmitir señales de voz por Internet (VoIP por sus siglas en inglés *Voice over IP*). Aplicaciones como *F-Talk* [54] y P2P VoIP utilizan esta tecnología. *F-Talk*, que aún se encuentra en versión Beta, cifra la comunicación para garantizar privacidad. Sin embargo, el producto más conocido en este conjunto de aplicaciones es *Skype* [55], diseñado en 2003 por los creadores de *Kazaa*, que además de transmitir señales de voz se utiliza para distribuir *streaming* de video.

Un estudio experimental sobre el sistema VoIP de *Skype* puede consultarse en



[56]. En [57] se presenta una solución para la rápida recuperación del sistema VoIP sobre P2P en caso de falla. Un nuevo método para identificar tráfico P2P VoIP se expone en [58]. En [59] se presenta un prototipo de un sistema P2P de conferencia basado en SIP —*Session Initiation Protocol*— también aplicable a video. SIP [60] es un protocolo de señalización que permite iniciar, mantener y terminar una sesión multimedia. Actualmente existen varios trabajos que intentan mezclar técnicas de P2P con SIP.

Otro importante campo de acción para estas tecnologías lo constituyen las aplicaciones P2P para dispositivos móviles que algunos autores referencian como *Mobile P2P Networks*. Debido a la rápida proliferación de los *smart phones* aplicaciones P2P que en sus orígenes fueron creadas para correr en computadoras personales —*messengers, file sharing, streaming*— hoy en día están adaptándose a esta nueva plataforma. *Skype* y *Blue Talk* [61] son dos buenos ejemplos de estos desarrollos.

Una aplicación cada vez mayor de la tecnología P2P aprovecha la capacidad de procesamiento ociosa de las PC de escritorio [62]. De esta manera, es posible conseguir el poder de cómputo de costosas supercomputadoras simplemente utilizando las capacidades de procesamiento de muchas pequeñas computadoras de escritorio distribuidas en Internet [63]. Universidades y centros de investigaciones pueden realizar tareas que demandan grandes cantidades de recursos computacionales utilizando el poder de cómputo que existe en las computadoras hogareñas conectadas a Internet. El proyecto *SETI@Home* —SETI se corresponde con las siglas en inglés para Búsqueda de Inteligencia Extraterrestre— es un ejemplo de este tipo de aplicación.

La meta de SETI es detectar vida inteligente fuera de la Tierra. Utiliza radio telescopios para escuchar señales de radio de banda estrecha provenientes del espacio. No se tiene conocimiento de que estas señales ocurran de manera natural, por lo que su detección puede proporcionar evidencias de tecnología

extraterrestre. Las señales de radio telescopio obtenidas consisten principalmente de ruido de fuentes celestes, y otras generadas por el hombre —estaciones de televisión, radares, satélites artificiales etc.—. Actualmente SETI analizan los datos de forma digital por lo que requiere el mayor poder de cómputo posible. Más potencia de cálculo permite realizar mejores búsquedas para cubrir mayores rangos de frecuencias con más sensibilidad.

Anteriormente SETI utilizó computadoras especializadas localizadas en el mismo radio telescopio para realizar el grueso del análisis de datos. Sin embargo, con la idea de convertir radio SETI en un súper ordenador virtual compuesto de un gran número de ordenadores conectados a través de Internet en 1999 se lanzó el proyecto *SETI@home*, un experimento científico que utiliza computadoras conectadas a Internet para la búsqueda de inteligencia extraterrestre que ejecutan un programa libre que descarga y analiza datos obtenidos por radio telescopios cuando la computadora se encuentra ociosa. [12]

El proyecto *SETI@Home* está alojado en el sitio web de la Universidad de California, Berkeley. Actualmente es uno más de los proyectos agrupados en la Infraestructura Abierta de Berkeley para la Computación en Red (BOINC por sus siglas en inglés *Berkeley Open Infrastructure for Network Computing*).

BOINC [64] es la plataforma de computación voluntaria más popular disponible actualmente. Fue desarrollado en la Universidad de California en Berkeley y es financiado por la *National Science Foundation* (NSF). Aunque originalmente se desarrolló para el proyecto *SETI@home*, en la actualidad se utiliza para más de 40 proyectos en diversos campos como física, medicina nuclear, climatología, etc. La Tabla 1-1 muestra algunos de estos proyectos accesibles desde el sitio web de BOINC.

De acuerdo a lo expuesto previamente en este capítulo —ver sección 1.4 “Tipos de Redes P2P” en la página 27—, estos proyectos no parecen pertenecer a

ninguna de las categorías de red P2P vistas. Sin embargo, la literatura especializada, incluida una gran cantidad de publicaciones, los incluye como ejemplos de sistemas P2P [65] [66] [67] [68] [69]. El argumento esgrimido para considerar a los proyectos BOINC como sistemas P2P se basa en el hecho que la computación está basada en pares aunque la transferencia de los datos esté implementada por medio de una comunicación cliente/servidor [70].

**Tabla 1-1 Algunos proyectos BOINC**

<b>Nombre</b>	<b>Área</b>
Climateprediction.net	Ciencias de la Tierra – Estudio del clima
Docking@Home	Biología – Estudio de proteínas
Einstein@home	Astrofísica
Ibercivis	Múltiples aplicaciones
LHC@home	Física
MindModeling@Home	Ciencias Cognocitivas
NFS@home	Matemáticas – factorización
Primaboinka	Matemáticas – computación y juegos
Quake Catcher Network	Sismología – detección distribuida

Muchos autores señalan a los proyectos BOINC como pertenecientes a un grupo de aplicaciones conocido como P2P *grid*. La computación *grid*, que implica la ejecución de una tarea distribuida en gran cantidad de equipos bajo algún control centralizado o distribuido, es apta para implementarse sobre una arquitectura P2P aprovechando la capacidad ociosa de los equipos de usuario final conectados a Internet [71].

Otro campo de acción para los sistemas P2P todavía no explotado pero que puede transformarse en una tendencia futura es la combinación de sus técnicas con la computación en la nube —P2P *cloud*—. Así se argumenta en [72]. Xu *et al* proponen la construcción de un gran espacio en la nube confiable, distribuido y

eficiente basado en tecnologías P2P. Otros autores también han escrito sobre el tema [73] [74] [75].

Finalmente, el comercio electrónico o *E-commerce* también puede enriquecerse sobre una infraestructura P2P eliminando los sistemas centralizados y promoviendo C2C —*Consumer-to-Consumer*— además de B2B —*Business-to-Business*— y B2C —*Business-to-Consumer*—. Para hacer posible los sistemas P2P *E-commerce* es necesario contemplar varias consideraciones, como por ejemplo la seguridad en las comunicaciones, la transparencia en las transacciones y estandarizar los procesos de ventas [68].

## 1.6 Resumen del Capítulo

Las redes P2P constituyen sistemas distribuidos compuestos por pares o iguales que comparten espacio de almacenamiento, ancho de banda, poder de cómputo u otro tipo de recursos, con un modelo de comunicación simétrico sin roles diferenciados ni control central. Se destacan por ser escalables, tolerante a fallas y favorecer el equilibrio de carga, la descentralización y el anonimato.

Aunque en los comienzos de la Internet el modelo de comunicación simétrica estaba presente, a mediados de los 90' con el advenimiento de la Internet comercial cambiaron los patrones de comportamiento de los usuarios hacia una modalidad asimétrica con algunos nodos ofreciendo servicios y la mayoría consumiéndolos. Aparecieron los *firewalls*, las direcciones IP dinámicas, el protocolo NAT y el ancho de banda asimétrico ofrecido por los ISP que entorpecieron las comunicaciones simétricas.

La era moderna de las comunicaciones P2P comienza en el año 1999 con el éxito de *Napster*. El *file sharing* se convirtió en el área propicia para el desarrollo de la tecnología P2P. El protocolo *FastTrack* que dio origen a las redes *Kazaa*, *Grokster* e *iMesh* alcanzó gran popularidad.

*Gnutella* nació a principios del año 2000 y fue la primera red P2P pura sin ningún tipo de centralización. La red *eDonkey* aparece en septiembre de 2000 y rápidamente alcanzó gran popularidad. Sin embargo la actividad de *file sharing* sobre redes P2P en la actualidad está liderada por *BitTorrent*, un protocolo que apareció en julio de 2001. Ese mismo año, producto de la investigación científica en el área surgieron soluciones estructuradas basadas en tablas de *hash* distribuidas —*Chord*, *Pastry*, *Tapestry* y *CAN*— que debido a su complejidad no llegaron a ser muy populares. Sin embargo, algunas de sus técnicas fueron incorporadas más tarde por otras redes P2P como *Kademlia*, que alcanzó cierto éxito incorporándose en el cliente de *eMule* de la red *eDonkey* 2000.

*Gnutella* 0.4 y *FreeNet* pertenecen a la categoría de **sistemas P2P puros no estructurados** en la que no existe ningún tipo de centralización. Todos los pares se comunican en forma simétrica, no hay roles diferenciados ni tareas especiales, todos cumplen exactamente las mismas funciones. Son sencillos de implementar y favorecen la robustez, el equilibrio de carga y el dinamismo al permitir la entrada y salida frecuente de los pares con mínima carga operativa. Sin embargo, la búsqueda de recursos descentralizada conlleva algunas dificultades. *Gnutella* 0.4 la implementa inundando la red con mensajes de petición. El nodo que busca un recurso envía una solicitud a todos sus vecinos que a su vez la replican a todos sus otros vecinos. Esta técnica no es escalable. Para controlar el tráfico generado por las solicitudes fue necesario limitar su alcance con un campo *time to live* (TTL) que degrada la capacidad de búsqueda.

*Chord*, *Pastry*, *Tapestry*, *CAN* y *Kademlia* son **sistemas P2P puros estructurados**. En esta categoría no existe ningún tipo de centralización, todos los pares cumplen la misma tarea incorporándose a una estructura no jerárquica usualmente en forma de anillo. La búsqueda de recursos se resuelve eficientemente accediendo a un índice global almacenado en una tabla de *hash* distribuida entre todos los nodos del anillo. Cada nodo se hace cargo de mantener y contestar las peticiones sobre una partición del espacio de claves de la tabla. La

cantidad de mensajes necesarios para resolver la búsqueda pertenece a  $O(\log N)$  siendo  $N$  la cantidad de nodos de la red. Sin embargo, la implementación de estos sistemas no es sencilla. Mantener la estructura funcionando correctamente es costoso y exige una carga operativa considerable para permitir la entrada y salida de los nodos a la red. Además es necesario realizar periódicamente tareas de mantenimiento para volver a establecer el anillo ante la posible caída de algún nodo.

*Napster*, *eDonkey*, *BitTorrent*, *Gnutella 0.6* y *Ares Galaxy* son ejemplos de **sistemas P2P híbridos**. En ellos existe algún modo de centralización para facilitar la búsqueda de recursos. El esquema más simple es concentrar en un servidor central el índice global de todos los recursos compartidos en la red, conformado con los datos suministrados por cada uno de los pares. La búsqueda se resuelve en forma sencilla por medio de una consulta al servidor. Luego el intercambio entre los pares se realiza a través de una comunicación simétrica. Estos sistemas no son muy escalables pues el servidor se transforma en un cuello de botella. La solución obvia utilizada por la mayoría de estas redes fue multiplicar la cantidad de servidores centrales y replicar los datos. *BitTorrent* no utiliza servidores que manejen información de toda la red pero se vale de gran cantidad de *trackers* que controlan las transacciones sobre subconjuntos, posiblemente reducidos, de recursos. Otro esquema de centralización más débil es el ofrecido por *Gnutella 0.6* y *Ares Galaxy* que utilizan nodos de mayor jerarquía llamados superpares para coordinar a los nodos hojas.

Una de las aplicaciones más destacadas de la tecnología P2P ha sido la de intercambio de archivos en Internet. Se destacan las soluciones híbridas como las más exitosas en esta actividad aunque no debe excluirse a *Gnutella* sobre todo en sus comienzos. La utilización de la tecnología P2P en aplicaciones de *file sharing* sigue resultando de interés para la comunidad científica que continúa realizando publicaciones en el área.

Otro escenario donde se aprovecha la infraestructura ofrecida por las redes P2P lo constituye la transmisión de flujo de video —*PPLive*, *PPStream*, *SopCast*, *Tvants*— y de señales de voz —*F-Talk*, *Skype*, *Blue Talk*— que permite compartir ancho de banda entre los pares. En particular *Skype* y *Blue Talk* también corren en dispositivos móviles. Gracias a la proliferación de los *smart phones* ha emergido un nuevo y prometedor campo de acción que algunos autores referencian como *Mobile P2P Networks*.

Una aplicación cada vez mayor de la tecnología P2P aprovecha la capacidad de procesamiento ociosa de las PC de escritorio para compartir ciclos de CPU. Ejemplo de ello lo constituyen los proyectos *SETI@home*, *Docking@Home*, *Climateprediction.net* y muchos otros agrupados en la plataforma de computación voluntaria BOINC de la Universidad de California, Berkeley.

Finalmente, según lo señalado por distintos autores, la computación *grid*, la computación *cloud* y el *E-commerce* son tres áreas de posible aplicación para las tecnologías P2P aún no exploradas suficientemente pero con buenas perspectivas.





# Capítulo 2

## Redes Neuronales Artificiales (RNA)

### 2.1 Introducción

La automatización de ciertas tareas que los seres humanos realizan cotidianamente y sin esfuerzo, como por ejemplo la identificación de un individuo a partir de sus rasgos faciales, constituyen verdaderos desafíos para los sistemas de software convencionales. Las tramas complejas, formadas por numerosos elementos visuales que individualmente dicen poco acerca de la trama global y, sin embargo, colectivamente, representan objetos fácilmente identificables para una persona, son los tipos de tramas que más complican a los algoritmos de reconocimiento de patrones.

Por otro lado, algunas funciones cerebrales superiores de los seres humanos como la percepción<sup>2</sup> y el lenguaje sólo han sido imitadas artificialmente de manera incompleta o deficiente.

---

<sup>2</sup> La percepción está siempre asociada a un proceso cognoscitivo. Existe una diferencia sutil, y muy importante, entre las funciones de "sentir" y "percibir". Mientras que la primera es una operación simple a nivel de los sentidos, la segunda es una función compleja a nivel cerebral. El "sentir" es innato, pero el "percibir" debe ser aprendido. Estudios sobre personas ciegas de nacimiento que recuperaron la visión en la edad adulta evidencian la necesidad de aprender a percibir imágenes visuales. También se ha encontrado que no existen conexiones innatas entre la identificación auditiva del sonido producido por un objeto y su identificación visual. Todas estas asociaciones de ideas deben establecerse a través de la experiencia.

Estas dificultades y limitaciones han conducido a investigadores a centrar su atención en el desarrollo de sistemas de tratamiento de la información que permitan solucionar problemas cotidianos tal como lo hace el cerebro humano. Existen muchas características de este órgano biológico deseables para cualquier sistema artificial de procesamiento. El cerebro es robusto y tolerante a fallas, diariamente mueren neuronas sin afectar su desempeño; es capaz de interpretar información imprecisa, con ruido o inconsistente suministrada por los sentidos a muy alta velocidad; es flexible ajustándose a nuevos ambientes aprendiendo, sin instrucciones explícitas de ninguna clase; a crear las representaciones internas que hacen posibles sus habilidades.

Basados en la eficiencia de los procesos llevados a cabo por el cerebro, e inspirados en su funcionamiento, varios investigadores han desarrollado la teoría de las Redes Neuronales Artificiales (RNA), las cuales emulan las redes neuronales biológicas. Estos sistemas no requieren que la tarea a ejecutar se programe, sino que generalizan y aprenden de la experiencia. Algunas de las características más destacadas de las RNA son las siguientes:

- Las RNA implementan algoritmos de aprendizaje. Extraen sus propias reglas para resolver los problemas a partir de ejemplos de entrenamiento y modifican su comportamiento en función de la experiencia.
- Las RNA son capaces de realizar abstracciones y generalizar. Una vez entrenadas pueden operar sobre datos distintos a los usados durante la etapa de aprendizaje. Las respuestas obtenidas dependerán del parecido de estos datos con los ejemplos de entrenamiento.
- Las RNA son tolerantes al ruido en la información. Debido a su capacidad para abstraer las características esenciales de los datos aprendidos son capaces de procesar correctamente información incompleta o distorsionada.
- Las RNA son tolerantes a fallas. El conocimiento adquirido por una red neuronal artificial no se representa explícitamente, como ocurre en los

sistemas de Inteligencia Artificial simbólica<sup>3</sup>, sino que se almacena de manera distribuida en las conexiones de su arquitectura. Así una red neuronal artificial es capaz de seguir funcionando adecuadamente a pesar de que algunos de sus nodos o circuitos de comunicación dejen de estar operativos.

- Las RNA son fácilmente paralelizables. Sus topologías formadas por nodos donde se procesan datos y enlaces por los que circula información las hacen especialmente adecuadas para el procesamiento paralelo.

Cuando la investigación sobre redes neuronales parecía prometedora, en 1969 Minsky y Papert publicaron un trabajo [76] que significó el abandono masivo de las prácticas investigativas en el área, sobre todo en los Estados Unidos, ya que en Europa su influencia fue menor. Sus críticas estaban centradas en la limitación del *Perceptrón* —primera red neuronal con capacidad de aprendizaje— para resolver una operación lógica tan básica como la disyunción exclusiva o XOR. Las instituciones dejaron de financiar los trabajos que giraban alrededor de las RNA, ya que parecía un camino sin salida.

Sin embargo, a principios de los 80, esta disciplina tomó un nuevo y mayor impulso. Apareció el exitoso algoritmo de propagación hacia atrás—en realidad fue redescubierto— [77] y que da nombre a la red *Backpropagation* (BPN), y un conjunto de nuevas arquitecturas y algoritmos de entrenamiento entre los que se destacan los mapas autorganizados de Kohonen o *Self Organizing Maps* (SOM) [78] [79] [80] y la red de contrapropagación o *Counterpropagation* (CPN) desarrollada por Robert Hecht-Nielsen a mediados de los 80 [81] [82] [83].

La implementación de soluciones basadas en RNA presupone habitualmente una fase o proceso de aprendizaje, también llamado de entrenamiento, seguido

---

<sup>3</sup> El estudio de las RNA forma parte de un enfoque más amplio denominado conexionismo y es además uno de los tópicos más importantes del enfoque subsimbólico de la Inteligencia Artificial. Este último se opone al enfoque simbólico conocido generalmente como Inteligencia Artificial Clásica o simplemente Inteligencia Artificial.

opcionalmente de una fase de test, y finalmente de aplicación.

El entrenamiento es el proceso por el cual los parámetros libres de una red neuronal son ajustados a medida que ésta se expone a la estimulación externa. El tipo de aprendizaje es determinado por la forma que tienen lugar dichos cambios.

Uno de los modos de aprendizaje más difundido consiste en la presentación de patrones de entrada junto a la salida deseada para cada patrón. A este tipo de aprendizaje se lo denomina *aprendizaje supervisado* y consiste en la adecuación interna de la red permitiendo reducir paulatinamente la diferencia entre la salida obtenida y la deseada hasta alcanzar un nivel de error global aceptable para todo el conjunto de patrones de entrenamiento utilizado. El aprendizaje supervisado es utilizado por la red BPN y debido a ello es uno de los más frecuentes entre las aplicaciones tecnológicas basadas en redes neuronales.

Una variante de este tipo de aprendizaje es el llamado *aprendizaje supervisado por refuerzo*. En éste no es necesario el conocimiento exacto de la salida deseada de la red. Consiste en poner en funcionamiento a la red y evaluar el éxito o fracaso de la salida, se produce entonces una señal llamada de refuerzo que mide el buen funcionamiento del sistema. Esta señal puede ser simplemente una indicación de que el dispositivo funcionó correctamente o no lo hizo. Si una acción tomada por el sistema de aprendizaje es seguida por un estado satisfactorio, entonces la adecuación interna de la red tenderá a reforzar esa acción particular ante la presencia del mismo estímulo. En otro caso, la tendencia del sistema será disminuir la respuesta en esa dirección. En el presente trabajo, la red neuronal implementada con base en la arquitectura denominada *Learning Vector Quantization (LVQ)* —ver sección 4.2.4.2 “Implementación de Aprendizaje en Línea Basado en LVQ1” en la página 130— utiliza este tipo de aprendizaje.

Existe otro tipo de aprendizaje llamado *aprendizaje auto-organizado* por el cuál se extraen las características de los ejemplos de aprendizaje sin supervisión

alguna, agrupándolos en clases de patrones similares. No se sabe *a priori* que salida corresponderá a cada grupo o categoría de patrones de entrada. Este tipo de aprendizaje se utiliza para implementar sistemas de clasificación o *clustering* y mapeo de características de los datos de entrada. Un ejemplo típico de red que utiliza este modo de aprendizaje, lo constituye la red SOM de Kohonen.

Los contenidos de este capítulo incluyen una breve descripción de las características relevantes de las neuronas biológicas y de su versión artificial. Asimismo se presentan tres de las arquitecturas más relevantes de la disciplina o bien por sus implicaciones históricas como es el caso del *Perceptrón* o por el éxito obtenido en la resolución de problemas concretos, tal es el caso de las redes BPN y SOM.

La importancia de la red BPN y de la red SOM en el contexto de las RNA es significativa. Gran parte de los desarrollos en el área están basados en alguna de estas dos redes o en derivaciones de las mismas. Además, dado que ambas utilizan distintos tipos de aprendizaje, la elección de estas dos arquitecturas en los contenidos de este capítulo, permite cubrir varios de los aspectos más importantes de las RNA.

## 2.2 La Neurona Biológica

El investigador español Ramón y Cajal (1/5/1852 – 17/10/1934) quien recibió en 1906 el premio Nobel en medicina por su trabajo sobre la morfología y conectividad de las células nerviosas, demostró que el tejido nervioso está compuesto por células individuales, separadas una de otra, es decir que no se continúan como se creía en esa época. Este postulado, que fue plenamente confirmado con el uso del microscopio electrónico a mediados del siglo XX, es esencial para comprender la forma en que se comunican las células nerviosas.

Las neuronas, constituyen la unidad neurológica del sistema nervioso. Son

células altamente especializadas en la comunicación y de una morfología muy particular. En ellas se distingue una zona más voluminosa llamada cuerpo o soma —donde se encuentra el núcleo celular y la maquinaria química para fabricar proteínas y otras moléculas— con forma de pera, esférica o ligeramente alargada. Múltiples ramificaciones parten del soma en la que se distingue una particularmente larga llamada axón y muy delgada en comparación con el cuerpo de la neurona. La función del axón es conducir el impulso nervioso hacia la periferia de la célula.

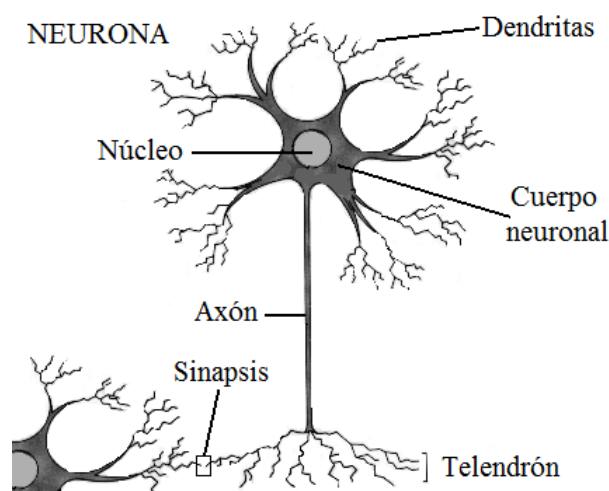


Figura 2-1 Esquema general de la forma y estructura de las neuronas

Además del axón, que arranca desde un solo sitio del soma pero que puede después ramificarse profusamente —telodendrón—, existen en las neuronas muchas otras prolongaciones que parten también del soma, generalmente del lado opuesto al del origen del axón. A diferencia de éste, son casi siempre muy numerosas desde su origen mismo, se ramifican casi inmediatamente y reciben el nombre de dendritas. Los estímulos —información de entrada— ingresan a la célula a través de las dendritas, provocando en ocasiones un impulso nervioso que se propaga por

el axón y es comunicado a otras neuronas. Esta comunicación tiene lugar en las sinapsis —espacio microscópico entre las neuronas— por medio del pasaje de neurotransmisores.

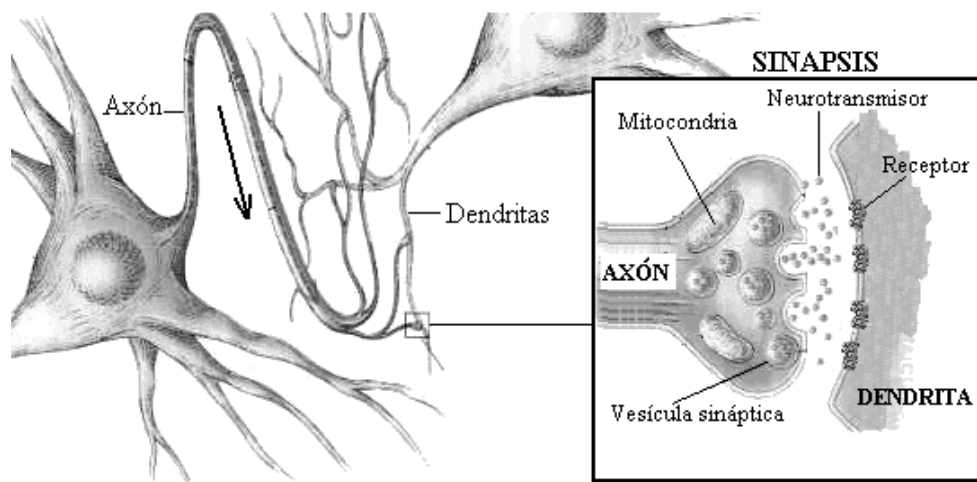
Las neuronas son células que se han especializado en la comunicación abandonando prácticamente toda otra función celular. La información que comunican permite al sistema nervioso interactuar con el medio externo e interno y llevar a cabo todo los procesos mentales como razonar, memorizar, evocar, imaginar, sentir etc.

El axón, bajo ciertas circunstancias, genera en su sitio de origen una corriente eléctrica que es conducida a lo largo del mismo hasta el sitio donde termina y establece comunicación ya sea con un músculo, una glándula endocrina u otras neuronas. Pero como postulara Ramón y Cajal, las neuronas no se continúan unas con otras, no comparten la membrana celular, sino por el contrario están perfectamente delimitadas y separadas de cualquier otra célula. Así las terminaciones del axón y las dendritas se encuentran separadas físicamente y se comunican por medio del espacio sináptico.

Las fotografías de microscopía electrónica indican que si bien existen algunas sinapsis de apenas 2 nanómetros de espesor —2 millonésimas de milímetro— en donde la conducción eléctrica es factible y de hecho así sucede, la gran mayoría de estos espacios de comunicación son de unos 20 nanómetros, lo suficientemente grandes como para que la corriente eléctrica no pueda atravesarlo.

Esto hace necesario algún mecanismo para transformar la información eléctrica en algún otro tipo capaz de trasponer los espacios sinápticos. Ese mecanismo consiste en que, como efecto directo de la llegada del impulso eléctrico hasta la terminal del axón en la neurona presináptica, ésta libera una sustancia química a través de su membrana, la cual atraviesa el espacio de 20 millonésimas de milímetro que la separa de la siguiente neurona y así llega a ella.

Se requieren moléculas receptoras en la membrana de la segunda neurona —neurona postsináptica— para reconocer el mensajero químico y que la comunicación se lleve a cabo. Estas moléculas mensajeras reciben el nombre de neurotransmisores porque se forman en las neuronas.



**Figura 2-2 Sinapsis química.** En las terminaciones del axón se encuentran mitocondrias —para el aporte de energía— y pequeñas vesículas que contienen moléculas de neurotransmisores

El reconocimiento del neurotransmisor mediante el receptor correspondiente en la membrana de la neurona postsináptica puede generar en esta última una nueva corriente eléctrica que se propaga por su axón hasta todas las terminales del mismo. Sin embargo, en muchas ocasiones la neurona postsináptica ignora las señales recibidas y permanece inactiva.

Mientras que el efecto de algunos neurotransmisores es excitatorio, existen otros que inhiben a la neurona postsináptica disminuyendo la probabilidad de que esta última dispare señales eléctricas por su axón. Este mecanismo se explica de la siguiente manera:



La proteína que actúa como receptor del neurotransmisor en la membrana de las neuronas postsinápticas, al reconocer y aceptar a un determinado neurotransmisor sufre un cambio en su estructura tridimensional que produce una modificación en otras proteínas. Estas últimas son canales para sodio ( $\text{Na}^+$ ) o para cloro ( $\text{Cl}^-$ ), por lo que al modificarse por efecto de la combinación transmisor-receptor dejan pasar estos átomos que existen en el medio externo a las neuronas.

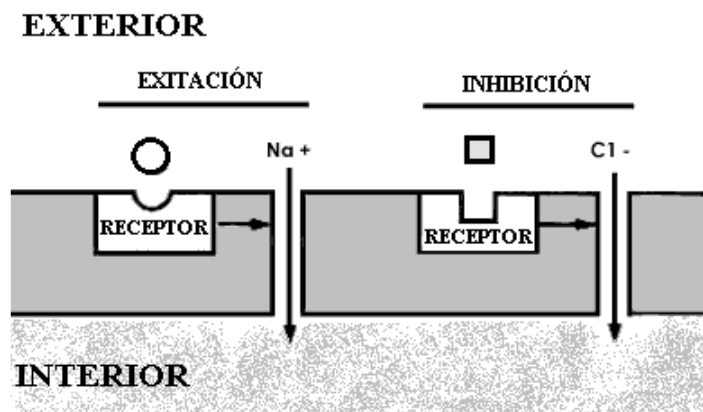


Figura 2-3 Excitación e inhibición de las neuronas.

Debido a que, los átomos de sodio tienen carga positiva y los de cloro negativa, al entrar, cambian las propiedades eléctricas de la membrana: cuando entra el sodio hay excitación de la neurona y cuando entra el cloro hay inhibición.

El soma de la neurona integra a cada instante toda la información inhibitoria y excitatoria recibida desde miles de terminales nerviosas provenientes de otras neuronas para obtener un resultado final que se traduce en un impulso eléctrico en caso de existir suficiente carga excitatoria.

Una vez generado el impulso eléctrico —llamado *potencial de acción*— se transmite a lo largo del axón como resultado de una serie de despolarizaciones

sucesivas que tienen lugar en los nódulos de Ranvier. Estos nódulos interrumpen periódicamente la vaina de mielina que recubre el axón. Cuando uno de los nódulos se despolariza se desencadena la despolarización del siguiente, así el potencial de acción viaja “a los saltos” de un nódulo a otro de manera mucho más veloz que en el caso de un axón sin recubrimiento de mielina. Una vez que el impulso eléctrico ha pasado por un cierto punto, ese punto no puede volver a ser excitado durante un lapso aproximado de 1 milisegundo, que es el tiempo que tarda en volver a su potencial de reposo.

El poder de cómputo del cerebro no reside en la velocidad de procesamiento de cada neurona que trabaja con un reloj interno de 1 KHz. sino en el alto grado de paralelismo y de interconexión. En el cerebro de un ser humano promedio existen alrededor de  $10^{11}$  neuronas, cada una conectada por medio de unas 10.000 a 100.000 sinapsis.

### 2.2.1 Ventajas de la Comunicación Química

La primera gran ventaja de la comunicación química es la posibilidad de inhibición. Si la comunicación fuese eléctrica, está claro que no podría haber inhibición tan sólo excitación o ausencia de estímulo. Solamente mediante el procedimiento de los neurotransmisores que al acoplarse con su receptor abren canales para átomos con carga negativa, es posible que las neuronas se inhiban.

Otra ventaja es la naturaleza unidireccional de la comunicación sólo de la terminal axónica a la neurona siguiente, pero no a la inversa. Con una comunicación eléctrica la corriente podría pasar de una neurona a otra y viceversa, es decir, la comunicación entre dos neuronas sería bidireccional, lo cual podría tener una serie de consecuencias indeseables en términos del control de la información a lo largo de determinados circuitos.

Además de estas ventajas, la existencia de la comunicación química posibilita la capacidad de regular o controlar la transmisión de información entre las

neuronas que no podría realizarse con una comunicación eléctrica. Por ejemplo, la neurona presináptica puede regular la velocidad de síntesis y de destrucción de los neurotransmisores como también controlar la liberación del mismo desde la terminal axónica, con lo cual se puede gobernar la comunicación. Pero también la neurona receptora tiene capacidad de control haciéndose más sensible o insensible a un neurotransmisor determinado.

Por lo tanto, la comunicación química permite la modificación de la comunicación interneuronal haciendo que la transmisión de información sea flexible, variable a lo largo del tiempo. Los científicos suponen que la capacidad de aprender del cerebro está íntimamente ligada a esta posibilidad de controlar la transmisión de información entre las neuronas que participan en intrincados circuitos de comunicación. Esta facultad está directamente relacionada con la plasticidad neuronal, la memoria y el aprendizaje. Precisamente esta capacidad es la base fundamental para la mayoría de los algoritmos de aprendizaje de las redes neuronales artificiales en donde las arquitecturas fijas “aprenden” cambiando la importancia o peso de cada conexión —sinapsis— regulando así la transmisión de información a través de la red.

## 2.3 La Neurona Artificial

A partir de las observaciones biológicas mencionadas en el apartado anterior, Warren McCulloch y Walter Pitts en 1943 [84] presentaron el primer modelo de una neurona artificial.

La neurona artificial posee un número finito  $n$  de conexiones de entrada que se corresponden con las dendritas en el modelo biológico, por las cuales ingresa cada una de las  $n$  componentes de un vector de entrada  $\mathbf{x}$ . Toda conexión de entrada tiene asociada una magnitud llamada peso o intensidad —eficiencia sináptica— definida por las componentes del vector de pesos  $\mathbf{w}$ . Estos pesos  $w_i$  pueden ser positivos o negativos, es decir excitatorios o inhibitorios

respectivamente.

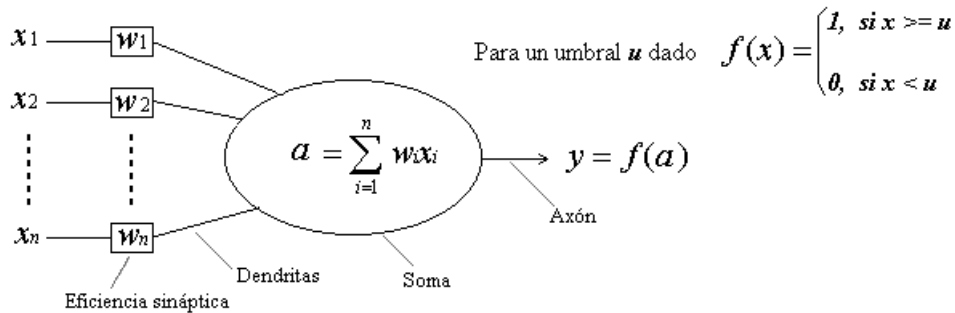


Figura 2-4 Modelo de neurona artificial de McCulloch y Pitts

La neurona acumula todas las señales de entradas multiplicadas por sus pesos de conexión obteniendo el valor de activación — $a$  en la ilustración anterior— al que se aplica una función llamada de *transferencia*<sup>4</sup>, en este caso la función umbral para obtener la salida de la neurona.

Al igual que en una neurona biológica, existen muchas entradas pero una única salida, la cual podrá no obstante, conectarse con muchas otras neuronas postsinápticas. En este modelo, una neurona es un elemento binario cuyo estado se manifiesta por la salida que pertenece al conjunto binario  $\{0,1\}$ . Así, los autores han utilizado la lógica proposicional para describir la acción de ciertas redes formadas por la conexión de estas neuronas artificiales.

<sup>4</sup> Desafortunadamente no existe una terminología unificada respecto al nombre de esta función. Suele denominarse también función de activación. Freeman y Skapura en [83] hace la siguiente distinción: a la sumatoria de la entrada ponderada por los pesos sinápticos la denomina *entrada neta*, una vez calculada se transforma en el valor de activación aplicando alguna función  $a(t) = F(a(t-1), neta(t))$  para denotar que la activación es una función explícita de la entrada neta y puede también depender del valor anterior de activación. Sin embargo en la mayoría de los casos la activación y la entrada neta son idénticas y los términos suelen emplearse de manera intercambiable. Una vez calculada la activación se obtiene el valor de salida aplicándole la *función de transferencia*.

El modelo McCulloch-Pitts intenta emular el procesamiento neuronal, pero no se incluye proceso de aprendizaje alguno. Sin embargo su investigación ha sido significativa pues ha inspirado a un conjunto de investigadores del área que lo sucedieron.

## 2.4 El *Perceptrón*. Primera Red Neuronal Artificial con Capacidad de Aprendizaje

Aunque la teoría de McCulloch-Pitts ha resultado no ser un modelo preciso de la actividad cerebral, ha servido de inspiración para los investigadores de las ciencias de la computación en la actualidad [85].

El primer científico en desarrollar una aplicación práctica para la neurona de McCulloch y Pitts, aplicando las ideas de aprendizaje de Hebb, fue el psicólogo Frank Rosenblatt que diseñó la red tipo *perceptrón* [86]. La teoría de Donald Hebb procedente de un libro escrito en 1949 llamado “*Organization of Behaviour*” [87] explicaba que el aprendizaje biológico sólo sería posible si existiese una simultaneidad temporal del funcionamiento de dos neuronas conectadas entre sí, una presináptica y otra postsináptica. Ese funcionamiento simultáneo, permite la conformación de circuitos neuronales y sería el factor indispensable para la formación del recuerdo.

Rosenblatt creía que la conectividad existente en las redes biológicas tiene un elevado porcentaje de aleatoriedad, por lo que se oponía al análisis de McCulloch-Pitts en el cual se empleaba la lógica simbólica para analizar estructuras bastante idealizadas. Rosenblatt suponía que la herramienta de análisis más apropiada era la teoría de probabilidades.

El primer modelo de *Perceptrón* fue desarrollado imitando el funcionamiento del ojo humano, el *fotoperceptrón* como se le llamó era un dispositivo que respondía a señales ópticas, y era capaz de clasificar patrones correctamente, en lo

que Rosenblatt denominaba un entorno diferenciado, en el cual cada clase estaba formada por patrones similares.

El *Perceptrón* es un dispositivo de aprendizaje que debe entrenarse hasta adquirir la capacidad de distinguir patrones. Trabaja con patrones de entrada binarios, y su funcionamiento se realiza en dos etapas: Una primera etapa de entrenamiento en la que se presentan las entradas y las salidas deseadas para que la red aprenda la salida que debe dar para cada entrada, y una segunda etapa de operación, en la que el *Perceptrón* será capaz de clasificar correctamente los patrones de entradas del conjunto de entrenamiento e idealmente cualquier otro nuevo patrón nunca antes presentado a la red.

En esencia, el entrenamiento del *Perceptrón* implica un proceso supervisado mediante el cual los pesos de las conexiones se incrementan o se decrementan dependiendo de si las mismas contribuyen o no a las respuestas correctas del *Perceptrón* para una entrada dada.

El *Perceptrón* es una red neuronal que consiste en una o más neuronas —que son también referidas individualmente como *perceptrones*—. Un *Perceptrón* simple tiene capacidad para distinguir sólo 2 clases de patrones de entre un conjunto de entrada. A continuación se presenta un ejemplo de un *Perceptrón* simple que realiza una clasificación de vectores de dos dimensiones.

#### 2.4.1 El *Perceptrón* Simple

El *Perceptrón* de la Figura 2-5 realiza una agrupación de vectores en dos clases  $A=\{(0,0),(0,1),(1,0)\}$  y  $B=\{(1,1)\}$ . Para la clase  $A$  el *Perceptrón* produce la salida 1 y para la clase  $B$  la salida 0. Obsérvese que otra forma de expresar el comportamiento de este dispositivo es indicando que calcula la operación lógica AND.

Si en lugar de utilizar entradas binarias se estimula al *Perceptrón* de la

Figura 2-5 con vectores del plano del plano  $R^2$ , este dispositivo aún seguirá agrupándolos en dos clases distintas.

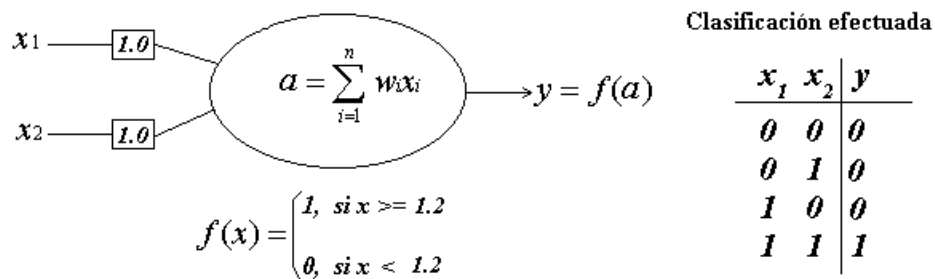


Figura 2-5 Ejemplo de clasificación efectuada por un Perceptrón simple. A la derecha se observa una tabla con el valor de salida del Perceptrón para cada entrada binaria.

Para visualizar con claridad las posibilidades y limitaciones del *Perceptrón* simple es posible realizar una interpretación geométrica de su salida. Para ello se puede re-escribir la función  $f(a)$  de la siguiente manera:

$$f(a) = \begin{cases} 1 & \text{si } w_1 x_1 + w_2 x_2 \geq u \\ 0 & \text{si } w_1 x_1 + w_2 x_2 < u \end{cases} \quad (2-1)$$

En el plano  $x_1 x_2$  la ecuación  $w_1 x_1 + w_2 x_2 = u$  representa la ecuación de una recta y en el ejemplo presentado, para  $u=1,2$ ;  $w_1=w_2=1$  la recta queda definida por la ecuación  $x_1+x_2=1,2$ .

La recta divide al plano en dos regiones distintas determinando así en función de la salida del *Perceptrón* dos clases posibles de vectores. Así el *Perceptrón* de la Figura 2-5 es capaz de clasificar cualquier punto perteneciente al plano  $R^2$ .

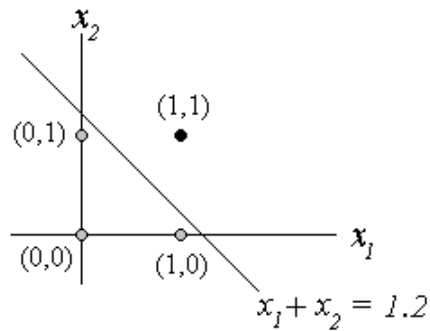


Figura 2-6 Interpretación geométrica de la función de clasificación que realiza el Perceptrón definido en la Figura 2-5. La salida de la neurona será 1 si como entrada recibe cualquier punto “por encima” o perteneciente a la recta graficada, y 0 en caso contrario.

La interpretación geométrica pone en evidencia una gran limitación del *Perceptrón* simple: su incapacidad para clasificar grupos de vectores que no sean linealmente separables. Un problema es linealmente separable si en el espacio  $n$ -dimensional de patrones de entrada se puede encontrar un hiperplano —de dimensionalidad  $n-1$ — que separe a los patrones de una clase respecto de la otra.

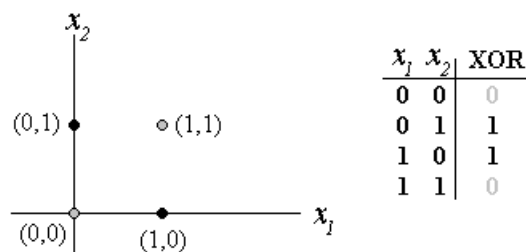


Figura 2-7 XOR Lógico. Esta función no puede implementarse por un Perceptrón ya que los puntos en el plano no son linealmente separables. Observe que no hay forma de trazar una recta que agrupe los puntos rojos de un lado y los azules del otro.



Como se observa en la Figura 2-7, un problema muy sencillo que no es linealmente separable y que por lo tanto un *Perceptrón* simple no puede resolver es la implementación de la función lógica XOR.

Esta limitación del *Perceptrón* simple, como se verá más adelante en este capítulo, se supera con la utilización de varios *perceptrones* conectados adecuadamente en una red *Perceptrón* multicapa.

Una variación en el modelo presentado evita tener que considerar el valor adecuado para el umbral  $u$ . Consiste en incluir al umbral como el peso de una nueva conexión llamada tendencia. Por lo tanto, al igual que cualquier otro peso sináptico, su valor será determinado adecuadamente por el proceso de aprendizaje que se expone más adelante.

La ecuación  $\sum_{i=1}^n w_i x_i = u$  se reescribe como  $\sum_{i=1}^n w_i x_i - u = 0$  y haciendo  $w_0 = -u$ ,  $x_0 = 1$  resulta  $\sum_{i=0}^n w_i x_i = 0$

De esta manera se puede reescribir la activación del *Perceptrón* ahora con  $(n+1)$  entradas, por el agregado del término de tendencia  $x_0$  cuyo valor es siempre 1. También se reescribe la función de transferencia  $f(x)$  ya que el umbral ahora siempre es cero.

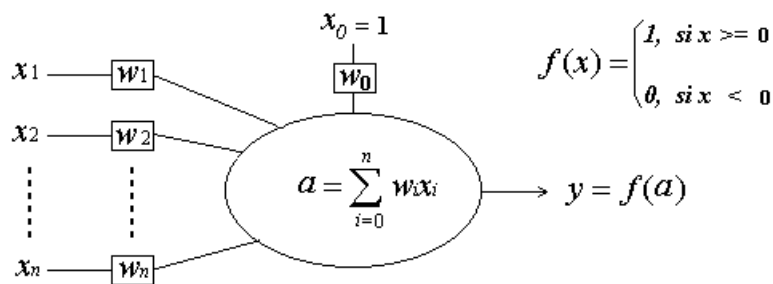


Figura 2-8 Perceptrón simple. Observe que se ha incluido una nueva entrada  $x_0$  llamada término de tendencia.

La ventaja de la inclusión del término de tendencia está dada por la determinación automática del umbral —indirectamente por  $w_0$ — a través del algoritmo de aprendizaje que se expone a continuación.

## 2.4.2 Aprendizaje del Perceptrón Simple

El aprendizaje del *Perceptrón* simple es de tipo supervisado por lo que se debe elegir para el entrenamiento un conjunto de patrones  $\mathbf{x}_i$  para los cuales se conoce la salida deseada  $d_i$  del dispositivo, es decir la clase a la que pertenece  $\mathbf{x}_i$ . Debido a que el *Perceptrón* simple es capaz de distinguir sólo entre dos clases distintas, los valores posibles para  $d_i$  quedan restringidos al conjunto binario  $\{0,1\}$ . El *Perceptrón* a través del proceso iterativo de entrenamiento aprenderá la correspondencia  $(\mathbf{x}_i, d_i)$ .

En el proceso de entrenamiento el *Perceptrón* se expone repetidamente a una serie de ejemplos, ajustando el vector de pesos  $\mathbf{w}$  en cada caso de forma que al final de este proceso se obtengan las salidas esperadas clasificando correctamente todos los patrones del conjunto de entrenamiento. Idealmente el *Perceptrón* entrenado será capaz de generalizar el conocimiento adquirido clasificando adecuadamente patrones con los que no ha sido entrenado.

El entrenamiento del *Perceptrón* simple se resume en el siguiente algoritmo:

Inicializar los pesos  $w_i$  con valores aleatorios

**Repetir**

**Para** cada par de entrenamiento  $(\mathbf{x}_j, d_j)$  hacer

$y \leftarrow$  salida del Perceptrón para  $\mathbf{x}_j$

        si  $y = d_j$  «no hacer nada. Clasificación correcta de  $\mathbf{x}_j$ » (1)

        si  $(y = 0)$  and  $(d_j=1)$  « hacer  $\mathbf{w} \leftarrow \mathbf{w} + \mathbf{x}_j$  » (2)

        si  $(y = 1)$  and  $(d_j=0)$  « hacer  $\mathbf{w} \leftarrow \mathbf{w} - \mathbf{x}_j$  » (3)

**Hasta** que todos los patrones hayan sido correctamente clasificados

Obsérvese que las adecuaciones de los pesos sinápticos en (1), (2) y (3) pueden reescribirse como  $\mathbf{w} \leftarrow \mathbf{w} + (d_j - y)\mathbf{x}_j$  que representa una expresión

unificada para el ajuste de pesos del *Perceptrón*. Esta expresión es completada con la incorporación del parámetro  $\mu$  llamado velocidad de aprendizaje, con  $0 < \mu \leq 1$ , que impide cambios bruscos en el vector  $\mathbf{w}$  evitando una posible oscilación infinita del sistema. Generalmente el valor elegido para  $\mu$  es un número pequeño, cercano a 0,1.

Asumiendo que en cada instante  $t$  se realiza un ajuste de  $\mathbf{w}$ , la regla de aprendizaje del *Perceptrón* expresada en función de  $t$  es la siguiente:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \mu(d_j - y) \mathbf{x}_j \quad (2-2)$$

Con la adición del parámetro  $\mu$  puede demostrarse un importante resultado conocido con el nombre de teorema de convergencia del *Perceptrón*. Este teorema afirma que, si las clases a distinguir son linealmente separables, luego de un número finito de pasos, el algoritmo de entrenamiento presentado anteriormente termina con la obtención de los pesos  $w_i$  adecuados que configuran la respuesta correcta del *Perceptrón* para el conjunto completo de patrones de entrenamiento.

### 2.4.3 *Perceptrón* Multicapa

La conocida limitación del *Perceptrón* simple para implementar la función lógica XOR se supera componiendo varios perceptrones en una red multicapa. Básicamente la idea consiste en utilizar dos perceptrones en una capa intermedia para implementar las funciones OR y NAND —ambas implican regiones linealmente separables—. A su vez, la salida de estos perceptrones conforma la entrada de un tercer *Perceptrón* en una capa de salida que implementa la función AND que también implica una clasificación linealmente separable. La Figura 2-9 ejemplifica esta solución.

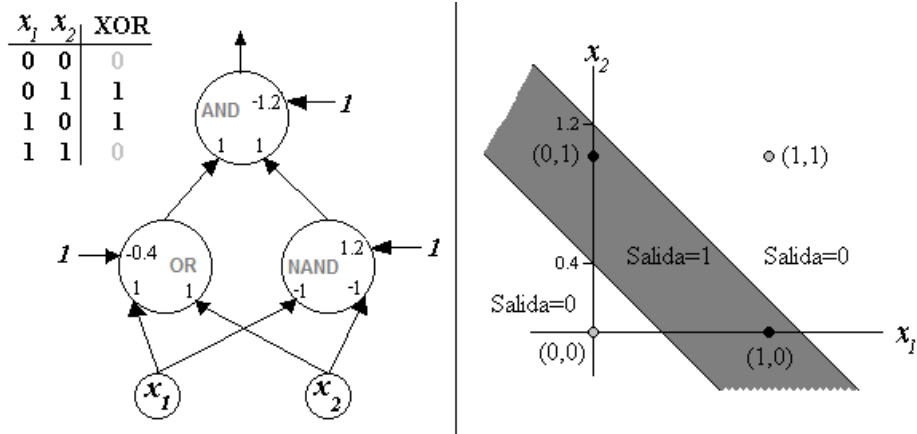


Figura 2-9 Este Perceptrón multicapa resuelve correctamente el problema XOR. Los números que figuran dentro de los nodos en el gráfico de la izquierda representan los pesos de cada una de las conexiones. La capa oculta establece dos rectas que separan el plano en tres regiones. Las dos regiones que contienen los puntos  $(0,0)$  y  $(1,1)$  se asocian a una salida nula de la red. La región central se asocia a la salida de la red con valor igual a 1.

Un *Perceptrón* multicapa es una red con alimentación hacia delante —*feedforward*—, compuesta de varias capas de neuronas entre la entrada y la salida de la misma, esta red permite establecer regiones de decisión mucho más complejas que las de dos semiplanos, como lo hace el *Perceptrón* simple.

El *Perceptrón* simple sólo puede establecer dos regiones separadas por una frontera lineal en el espacio de entrada de los patrones. Un *Perceptrón* con dos capas, puede formar cualquier región convexa en este espacio. Las regiones convexas se forman mediante la intersección de las regiones que definen cada neurona de la capa oculta, cada uno de estos elementos se comporta como un *Perceptrón* simple, activándose su salida para los patrones de un lado del hiperplano. Si la neurona del nivel de salida implementa la función lógica AND, la región de decisión resulta ser la intersección de todos los semiplanos formados en el nivel anterior. Esta región de decisión será una región convexa con un

número de lados a lo sumo igual al número de neuronas de la capa oculta.

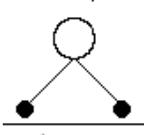
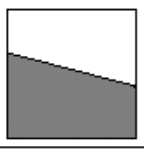
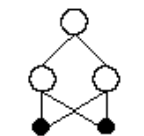
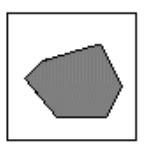
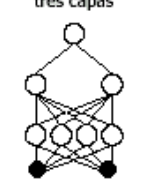

<i>Estructura</i>	<i>Tipos de regiones de decisión</i>	<i>Formas de las regiones de decisión</i>
<p>una capa</p> 	<p>medio plano limitado por un hiperplano</p>	
<p>dos capas</p> 	<p>regiones convexas o cerradas</p>	
<p>tres capas</p> 	<p>arbitraria (limitada por el número de nodos)</p>	

Figura 2-10 Imagen editada a partir de gráfico extraído de [88]

A partir de este análisis surge el interrogante respecto a los criterios de selección para las neuronas de las capas ocultas de una red multicapa, este número en general debe ser lo suficientemente grande como para que se forme una región compleja que pueda resolver el problema, sin embargo no debe ser muy grande pues podría producirse un sobreajuste a los patrones de entrenamiento mermando la capacidad de generalizar de la red.

El *Perceptrón* multicapa no se transformó en una red exitosa sino hasta la aparición de un algoritmo de aprendizaje adecuado llamado *Backpropagation*. Actualmente este es el algoritmo que más se emplea en la obtención de los pesos de la red multicapa de perceptrones y emplea una técnica de búsqueda del gradiente que minimiza el error esperado entre la salida actual y la deseada. Para

ello es necesario reemplazar la función escalón del *Perceptrón* por otra que sea derivable.

## 2.5 Backpropagation Network (BPN)

El primer algoritmo de entrenamiento para redes multicapa fue desarrollado por Paul Werbos en 1974 [77], este se desarrolló en un contexto general, para cualquier tipo de redes, siendo las redes neuronales una aplicación particular, razón por la cual el algoritmo no fue aceptado dentro de la comunidad de desarrolladores de redes neuronales.

Fue solo hasta mediados de los años 80 cuando el algoritmo *Backpropagation* o algoritmo de propagación inversa fue redescubierto al mismo tiempo por varios investigadores. El algoritmo se popularizó cuando fue incluido en el libro "*Parallel Distributed Processing*" por el psicólogo David Rumelhart [89]. La publicación de este libro trajo consigo un nuevo auge en las investigaciones con redes neuronales y posicionó al algoritmo *Backpropagation* entre los más utilizados, aún en nuestros días.

En la literatura especializada el término *Backpropagation* se utiliza para referirse tanto al algoritmo de aprendizaje como a la red de tipo *Perceptrón* multicapa entrenada con este algoritmo.

La red *Backpropagation* es una red formada por capas, con propagación hacia delante, completamente interconectada. No existen conexiones de retroalimentación, ni conexiones que salten una capa para ir directamente a otra superior. La figura siguiente muestra el esquema de una red *Backpropagation* con una capa oculta, este corresponde a la arquitectura más utilizada, sin embargo el algoritmo de aprendizaje puede generalizarse para más de una capa oculta.

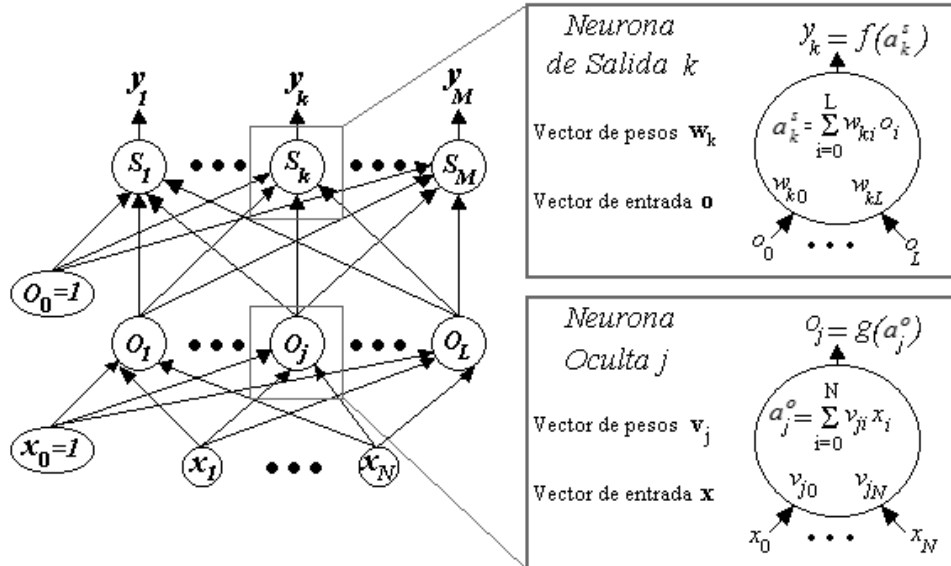


Figura 2-11 Arquitectura de una red Backpropagation de dos capas. Los pesos y unidades de tendencia son opcionales. Las funciones de transferencia  $g$  en la capa oculta y  $f$  en la capa de salida deben ser derivables para poder aplicar la regla de aprendizaje.

La función de transferencia de cada neurona tanto en la capa de salida como en la capa oculta debe ser una función derivable, por lo tanto queda excluida la función escalón que se utilizó en el *Perceptrón*.

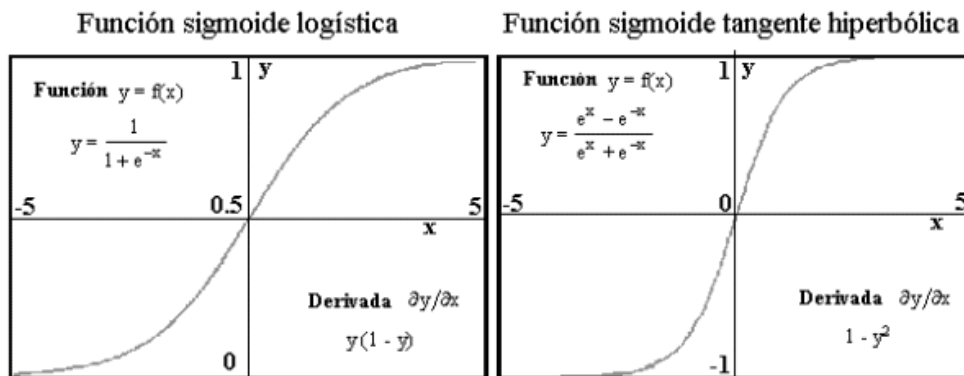


Figura 2-12 Funciones sigmoides.

Las funciones más utilizadas son la función Identidad y la sigmoide logística. La elección de esta función depende de la forma en que se pretende representar los datos de salida. Por ejemplo, si se desea que las unidades de salida sean binarias, se utiliza una función sigmoide que puede verse como una aproximación a la función escalón pero derivable en todo su dominio.

El aprendizaje *Backpropagation* es de tipo supervisado empleando un ciclo propagación–adaptación de dos fases. Una vez estimulada la red con un patrón de entrenamiento, la señal se propaga desde la primera capa a través de las capas superiores hasta generar la salida correspondiente. La señal de salida se compara con la salida deseada y se calcula una señal de error que se propaga hacia atrás, actualizando los pesos de las conexiones de todas las neuronas basándose en su contribución relativa a la salida de la red. A medida que este proceso se repite, la señal de error va disminuyendo alcanzando eventualmente el estado que permita clasificar correctamente todos los patrones de entrenamiento.

La importancia de este proceso consiste en que, a medida que se entrena la red, las neuronas de las capas intermedias se organizan a sí mismas de tal modo que las distintas neuronas aprenden a reconocer distintas características del espacio total de entrada.

Varias investigaciones han demostrado que, durante el proceso de entrenamiento, la red BPN tiende a desarrollar relaciones internas entre neuronas con el fin de organizar los datos de entrenamiento en clases. Esta tendencia se puede extrapolar, para llegar a la hipótesis consistente en que todas las unidades de la capa oculta de una BPN son asociadas de alguna manera a características específicas del patrón de entrada como consecuencia del entrenamiento. Esta misma representación interna se puede aplicar a entradas que la red no haya visto antes, y la red clasificará estas entradas según las características que compartan con los ejemplos de entrenamiento. Este último concepto es muy importante, se trata de la capacidad de abstracción para poder generalizar a través de ejemplos.



### 2.5.1 Regla de Aprendizaje

El algoritmo de entrenamiento consiste en la iteración del bloque presentación de una entrada, cálculo de la salida, determinación del error y corrección de los pesos de las conexiones. Conforme avanza el entrenamiento, el error va reduciéndose hasta que, al alcanzar un nivel aceptable, el proceso se detiene.

Sea la red BPN que se visualiza en la Figura 2-11, la regla de aprendizaje ajusta los pesos sinápticos de toda la red con el objetivo de minimizar la función de error  $E = \sum_{k=1}^M (d_k - y_k)^2$  calculada para cada ejemplo de entrada. Se emplea una versión iterativa del método de mínimos cuadrados denominada técnica del descenso más pronunciado. La deducción de la regla de aprendizaje, que puede consultarse en [85], utiliza el gradiente de la función  $E$ , expresada como una función de los pesos neuronales, para determinar la magnitud, dirección y sentido de la adecuación de los pesos. Las neuronas de la capa de salida actualizan sus pesos de la siguiente forma:

$$w_{ki}(t + 1) = w_{ki}(t) + \mu(d_k - y_k)f'(a_k^s)o_i \quad (2-3)$$

El factor  $\mu$  se denomina parámetro de velocidad de aprendizaje, siempre es positivo y suele ser menor que 1. Si la función de transferencia es la Identidad, la actualización de pesos será  $w_{ki}(t + 1) = w_{ki}(t) + \mu(d_k - y_k)o_i$ , en cambio si se utiliza la función sigmoide logística, la ecuación de actualización es la siguiente:

$$w_{ki}(t + 1) = w_{ki}(t) + \mu(d_k - y_k)y_k(1 - y_k)o_i$$

Los pesos de las neuronas de la capa oculta se actualizan de la manera siguiente:

$$v_{ji}(t+1) = v_{ji}(t) + \mu g'(a_j^o) x_i \sum_{k=1}^M (d_k - y_k) f'(a_k^s) w_{kj}(t) \quad (2-4)$$

Obsérvese que todas las actualizaciones de pesos de la capa oculta dependen de todos los términos de error  $(d_k - y_k)$  de la capa de salida. De aquí surge la noción de que los errores conocidos de la capa de salida se propagan hacia atrás, hacia la capa oculta, para determinar los cambios de pesos adecuados en esa capa.

### 2.5.2 Consideraciones Prácticas sobre la BPN

Frecuentemente para el entrenamiento sólo se necesita un subconjunto pequeño de los datos que se disponen. Por lo general el resto se utiliza para probar la red una vez concluido la etapa de aprendizaje.

Durante la etapa de aprendizaje de la red, los pesos son modificados de forma iterativa. Sin embargo, cuando el número de pesos es excesivo en relación al problema, el modelo se ajusta demasiado a las particularidades irrelevantes presentes en los patrones de entrenamiento, perdiendo su habilidad para generalizar a casos nuevos —fenómeno del sobreajuste—. Obsérvese que se puede disminuir el número de pesos en una arquitectura *Backpropagation* utilizando menos unidades ocultas. También se logra disminuyendo la cantidad de neuronas de entrada y salida, pero éstas son dependientes del problema, y sólo puede conseguirse cambiando la codificación de los parámetros de entrada y de la salida.

Para evitar el problema del sobreajuste, algunos autores sugieren crear tres conjuntos con los datos de entrenamiento, uno de aprendizaje, otro de validación y finalmente uno de test. Durante el aprendizaje la red va modificando los pesos en

función de los datos de entrenamiento y de forma alternada se la alimenta con los datos de validación. Con el grupo de validación se puede averiguar cuál es el número de pesos óptimo —y así evitar el problema del sobreajuste—, en función de la arquitectura que ha tenido la mejor ejecución con los datos de validación.

En otro sentido, Freeman y Skapura [85] aconsejan no entrenar por completo a la red con vectores de una clase, pasando después a otra clase; la red se olvidará del entrenamiento original, una estrategia de selección aleatoria del orden en que se presentarán los datos de entrada, o el intercalado entre distintas clases pueden ser dos alternativas viables.

Otra decisión importante para implementar una red *Backpropagation* se refiere a la cantidad de capas ocultas y cuántos nodos incluir en cada una de ellas. La cantidad de nodos de la capa de salida generalmente está fijada por el problema particular a resolver y depende si se desean valores analógicos o binarios en las unidades de salida.

Utilizar dos capas ocultas a menudo hace que la red aprenda más deprisa, pero habitualmente con una sola alcanza. Determinar el número más conveniente de neuronas que hay que utilizar en la capa oculta puede resultar dificultoso. El objetivo debe ser utilizar el menor número posible para ahorrar carga de procesamiento y al mismo tiempo evitar el sobreajuste ya referido. Para redes de un tamaño considerable —cientos o miles de entradas—, el tamaño de la capa oculta sólo necesita ser una fracción relativamente pequeña del que tiene la capa de entrada. Si la red no llega a converger para llegar a una solución, es posible que se necesiten más nodos ocultos. Si converge, se puede probar con un número inferior de nodos ocultos y determinar un tamaño final basándose en el rendimiento global del sistema.

Además si se examinan periódicamente los valores de los pesos de las neuronas de la capa oculta durante el entrenamiento, es posible detectar algunos

pesos que cambian muy poco respecto de sus valores iniciales. Estos nodos pueden no estar participando del aprendizaje indicando que probablemente existan unidades ocultas superfluas que pueden eliminarse de la red.

Los pesos iniciales deberían ser pequeños y aleatorios, por ejemplo entre  $-0,5$  y  $+0,5$ . Más problemático suele ser la elección del parámetro de velocidad de aprendizaje  $\mu$ , ya que tiene un efecto significativo en el rendimiento de la red. Normalmente  $\mu$  debe ser un número pequeño —del orden de  $0,05$  a  $0,25$ — para asegurar que la red llegue a estabilizarse en una solución. Suele ser posible incrementar el valor de  $\mu$  a medida que progresa el aprendizaje. Aumentar  $\mu$  a medida que disminuye el error de la red suele acelerar la convergencia incrementando el tamaño del paso conforme el error alcanza un valor mínimo, pero la red puede rebotar, alejándose demasiado del valor mínimo verdadero, si  $\mu$  llegara a ser demasiado grande.

Otra forma de aumentar la velocidad de convergencia consiste en utilizar una técnica llamada momento. Cuando se calcula el valor del cambio de peso, se añade una fracción del cambio anterior. Este término adicional tiende a mantener los cambios de peso en la misma dirección. La ecuación de cambio de pesos de la capa de salida se reescribe de la siguiente manera:

$$w_{ki}(t + 1) = w_{ki}(t) + \mu(d_k - y_k)f'(a_k^s)o_i + \alpha\Delta_{ki}(t - 1) \quad (2-5)$$

En la ecuación anterior,  $\alpha$  es el parámetro de momento, y permite filtrar las oscilaciones en la superficie del error provocadas por la tasa de aprendizaje  $\mu$  al acercarse al mínimo, y acelera considerablemente la convergencia de los pesos, ya que si en el momento  $t$  el incremento de un peso era positivo y en  $t + 1$  también, entonces el descenso por la superficie de error en  $t + 1$  será mayor. Sin embargo,

si en  $t$  el incremento era positivo y en  $t + 1$  es negativo, el paso que se da en  $t + 1$  es más pequeño, lo cual es adecuado, ya que eso significa que se ha pasado por un mínimo y los pasos deben ser menores para poder alcanzarlo. El factor momento suele tomar un valor próximo a 1

Debe tenerse en cuenta que para aprovechar la capacidad de las RNA de aprender relaciones complejas o no lineales entre variables, es imprescindible la utilización de funciones no lineales al menos en las neuronas de la capa oculta. Por lo general se utiliza la función sigmoide –logística o tangente hiperbólica– como función de transferencia en las neuronas de la capa oculta.

Por su parte, la elección de la función de transferencia en las neuronas de la capa de salida dependerá del tipo de tarea a realizar. En tareas de clasificación, las neuronas normalmente toman la función de activación sigmoide. Así, cuando se presenta un patrón que pertenece a una categoría particular, los valores de salida tienden a dar como valor 1 para la neurona de salida que representa la categoría de pertenencia del patrón, y 0 ó -1 para las otras neuronas de salida. En cambio, en tareas de predicción o aproximación de una función, generalmente las neuronas toman la función de activación lineal.

## 2.6 Self-Organizing Maps (SOM)

La red neuronal *Self-Organizing Maps* (Mapas auto-organizativos) fue desarrollada por Teuvo Kohonen durante el período 1979-1982 [78]. Históricamente, la red SOM fue una de las redes neuronales más importantes surgidas antes del gran impulso que tomara la disciplina a mediados de la década de 1980.

A diferencia del *Perceptrón* y la BPN, El tipo de aprendizaje que implementa SOM se denomina aprendizaje sin supervisión o auto-organizado, y se lleva a cabo únicamente mediante la aplicación de datos, sin ninguna información sobre

la salida deseada de la red.

Existe evidencia que en el cerebro las neuronas asociadas a distintas capacidades motoras y sensitivas se organizan en distintas zonas conformando un mapeo bidimensional sobre la corteza cerebral. Aunque en gran medida esta organización puede estar predeterminada genéticamente, es probable que parte de ella se origine mediante aprendizaje gracias a cierta capacidad del cerebro para formar mapas topológicos con la información recibida por el exterior. La red SOM está basada en estas ideas y constituye un sistema con capacidad de formar mapas de características de manera similar a como ocurre en el cerebro. La idea de Kohonen era demostrar que un estímulo externo por sí solo, sin ningún tipo de información extra como sucede en los casos de aprendizaje supervisado, era suficiente para forzar la formación de estos mapas.

La red SOM es capaz de establecer un mapeo  $f: A \subset \mathbf{R}^n \rightarrow \mathbf{R}^m$  respetando la topología, es decir que las unidades que se encuentren físicamente próximas entre sí van a responder a clases de vectores de entrada que, análogamente, se encuentren cercanos unos con otros. Así, los vectores de entrada de muchas dimensiones son proyectados sobre un mapa generalmente bidimensional, manteniendo el orden natural de los vectores de entrada.

La red SOM pertenece al grupo denominado redes competitivas, y aunque históricamente no fue la primera de este grupo, el trabajo de Kohonen en la teoría de la competición ha sido tan importante que hoy en día las neuronas competitivas suelen denominarse unidades de Kohonen.

### 2.6.1 Arquitectura SOM

La arquitectura clásica de una red SOM está formada por dos capas: *i*) la capa de entrada o capa sensorial que, al igual que en la BPN, sólo se utiliza como medio para el ingreso de los datos, y *ii*) la capa de salida formada por un conjunto de neuronas competitivas dispuestas en forma de arreglo bidimensional con el que

se consigue el mapeo topológico de los datos de entrada.

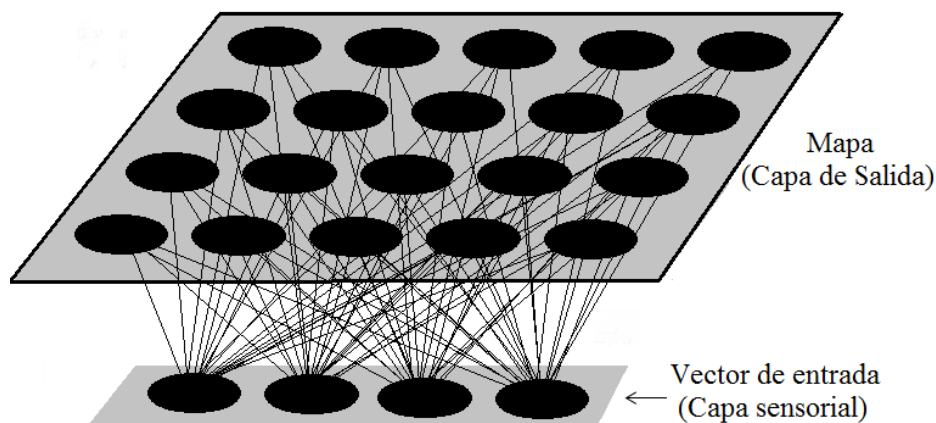
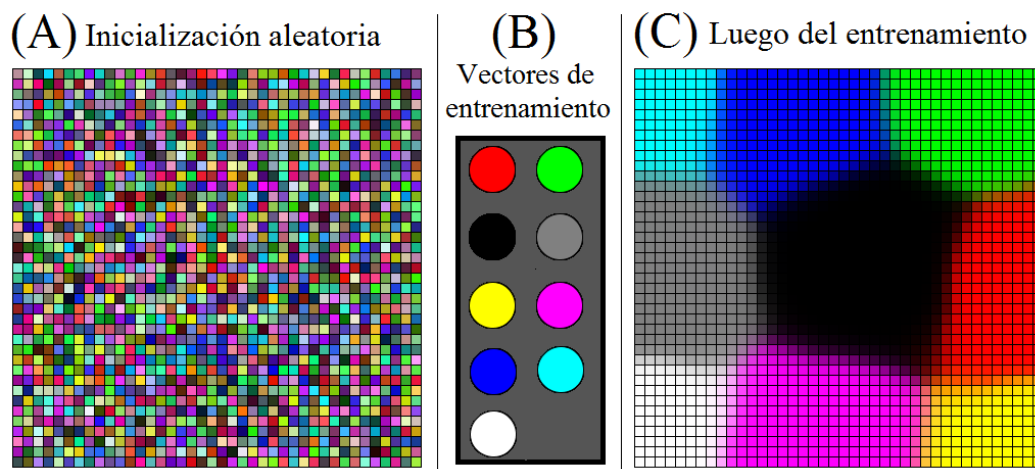


Figura 2-13 Arquitectura de la red SOM

En la capa de salida las neuronas compiten unas con otras por erigirse como la mejor representante del patrón de entrada presentado a la red. Al igual que en otras redes competitivas, el objetivo primario de la red es establecer a que clase pertenece el dato ingresado. La neurona que gane la competición estará representando a la clase que mejor se ajusta a las características del vector de entrada.

Sin embargo, el trabajo realizado por una red SOM es más que una simple clasificación de patrones. Debido a que la red responde ante datos con características comunes activando neuronas situadas en zonas próximas en la capa de salida, las clases identificadas por un SOM suelen quedar establecidas por regiones en la capa de salida que incluyen a un grupo de neuronas vecinas. Esto también permite apreciar zonas en los límites entre distintas clases que se activarán ante la presencia de patrones con características comunes a más de una clase.

Un ejemplo comúnmente utilizado para mostrar el funcionamiento de una red SOM es el mapeo en dos dimensiones del espacio tridimensional de colores representados por sus componentes RGB. La Figura 2-14 muestra un SOM que se entrena para reconocer 9 clases de colores diferentes.



**Figura 2-14 SOM que mapea colores RGB a un espacio bidimensional. Imagen confeccionada a partir de aplicación SOM de [90]**

La Figura 2-14(A) muestra el SOM inicializado aleatoriamente. Cada celda del mapa representa una neurona de la capa competitiva que se ha dibujado con el color que representa su correspondiente vector de pesos. Luego del entrenamiento con el conjunto de patrones que se muestran en la Figura 2-14(B) el SOM se estabiliza estableciendo el mapeo que se observa en la Figura 2-14(C). Observe que la gran variedad de colores existentes en el SOM sin entrenar se reduce drásticamente luego del entrenamiento lo que permite realizar la clasificación de cualquier entrada en las 9 clases diferentes pretendidas. No obstante observe la existencia de colores intermedios que aparecen en los límites de las regiones establecidas en el mapa. De esta forma el SOM, además de clasificar puede utilizarse para obtener grados de pertenencia a una clase y distancias a otras según



la ubicación relativa de la neurona que se active dentro de la región establecida para cada clase.

### 2.6.2 La Neurona Competitiva

En la Figura 2-15 se muestra un esquema del funcionamiento de una neurona competitiva. Este tipo de neurona computa una función  $f$  de similitud entre el vector de entradas  $\mathbf{x}$  y el vector de pesos  $\mathbf{w}$ . El valor de  $f(\mathbf{x}, \mathbf{w})$  será más alto cuanto más similares sean los vectores  $\mathbf{x}$  y  $\mathbf{w}$  entre sí. En la mayoría de los casos  $f$  es una función inversamente proporcional a la distancia euclídea entre los vector  $\mathbf{x}$  y  $\mathbf{w}$  pero no se descartan otras más específicas en distintos dominios de aplicación.

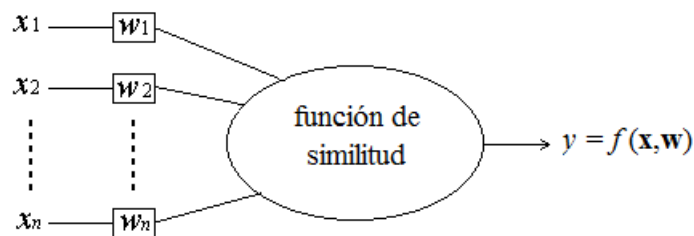


Figura 2-15 Neurona competitiva.

La utilización de computadoras para implementar redes neuronales artificiales suele imponer algún procesamiento de escalado de los datos de entrada para evitar el desbordamiento durante los cálculos de sumas y productos que predominan en la mayoría de las simulaciones de redes. Tratándose de redes competitivas es usual que los vectores de entrada  $\mathbf{x}$  se presenten a la red normalizados de tal forma que  $\|\mathbf{x}\| = 1$ . En este caso, los vectores de peso  $\mathbf{w}$  también se normalizan para que su longitud sea 1 y la función  $f$  de similitud suele calcularse como el producto

escalar de los vectores de entrada y de pesos, es decir  $f(\mathbf{x}, \mathbf{w}) = \mathbf{x} \cdot \mathbf{w} = \sum_{i=1}^n x_i w_i$ . Obsérvese que en este caso el producto escalar de los vectores  $\mathbf{x}$  y  $\mathbf{w}$  está calculando el coseno del ángulo comprendido entre ellos. Ello se debe a que  $\mathbf{x} \cdot \mathbf{w} = \|\mathbf{x}\| \cdot \|\mathbf{w}\| \cdot \cos(\theta) = \cos(\theta)$ . Así  $f$  determina lo próximo que están los vectores  $\mathbf{x}$  y  $\mathbf{w}$  entre sí, según lo mide el ángulo  $\theta$  entre ellos. El valor máximo de  $f$  se producirá cuando los vectores de entrada y de peso estén perfectamente alineados ( $\theta=0$ ).

### 2.6.2.1 Regla de Aprendizaje

El ajuste de los pesos sinápticos durante la etapa de entrenamiento de una red neuronal competitiva sólo se realiza sobre la neurona que ganó la competición en relación a una entrada de datos  $\mathbf{x}$ . En el caso de la red SOM además se ajustan los pesos de las neuronas ubicadas en un entorno cercano alrededor de la ganadora.

La neurona que gana la competición cuando la red es estimulada con el vector de entrada  $\mathbf{x}$ , será la que posee el vector de pesos  $\mathbf{w}$  más cercano a  $\mathbf{x}$ . La regla de aprendizaje aplicada sobre esta neurona pretende reforzar la ventaja en relación al vector  $\mathbf{x}$  acercando aún más el vector  $\mathbf{w}$  a la entrada  $\mathbf{x}$ .

De esta forma la regla de aprendizaje para una neurona competitiva queda definida de la siguiente forma:

$$\mathbf{w}(t + 1) = \mathbf{w}(t) + \alpha(\mathbf{x} - \mathbf{w}(t)) \quad (2-6)$$

En la Figura 2-16 se visualiza una interpretación geométrica de esta regla de aprendizaje. En Figura 2-16(A) se muestra un caso con vectores  $\mathbf{x}$  y  $\mathbf{w}$  sin normalización. Obsérvese que efectivamente, como resultado de la aplicación de la regla de aprendizaje, el vector  $\mathbf{w}$  se acerca al vector de entrada  $\mathbf{x}$ , es decir que

$w(t+1)$  está más próximo a  $x$  que  $w(t)$ . En la Figura 2-16(B) se muestra un caso con vectores  $x$  y  $w$  normalizados — $\|x\|=\|w\|=1$ —. También en este caso la aplicación de la regla de aprendizaje provoca el acercamiento de  $w$  a  $x$ . Sin embargo se observa claramente que una vez ajustado  $\|w(t+1)\| \neq 1$ , por lo que se hace necesario volver a normalizar el vector de pesos en cada ajuste.

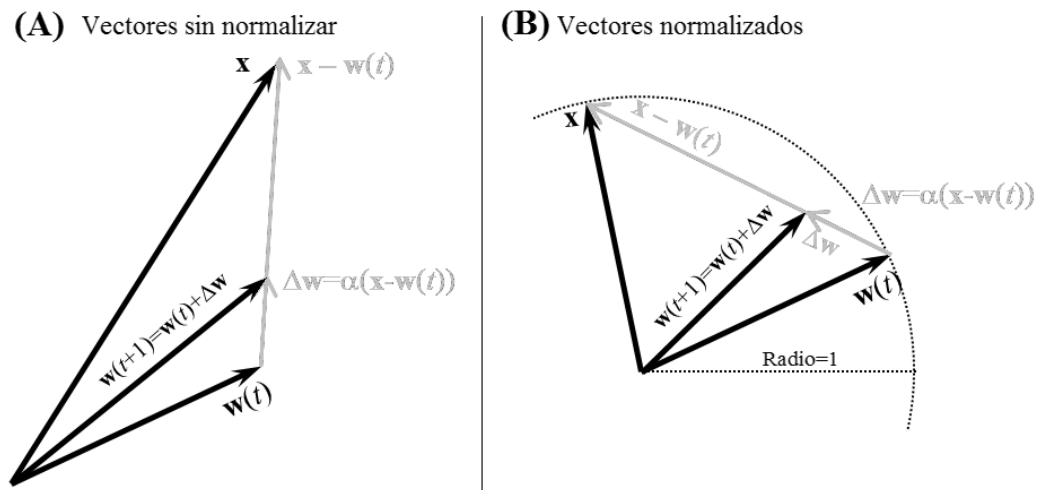


Figura 2-16 Regla de aprendizaje de la neurona competitiva. Para ambos casos —vectores normalizados y no normalizados— se utiliza la misma regla de aprendizaje. Sin embargo, para el caso (B), luego de efectuado el ajuste sobre el vector  $w$  es necesario volver a normalizarlo para garantizar que  $\|w(t+1)\|=1$

### 2.6.3 Entrenamiento de la Red SOM

En lugar de actualizar únicamente los pesos de la unidad ganadora, se define una vecindad física en torno a ésta, y todas las unidades que residan en dicha vecindad participan en el proceso de actualización de pesos. A medida que progresa el aprendizaje, el tamaño de la vecindad va disminuyendo hasta que abarca únicamente a la neurona ganadora. Si  $c$  es la neurona ganadora cuando se ingresa el vector  $x$  a la red, y  $N_c$  es la lista de índices de unidades que forman la

vecindad, entonces las ecuaciones de actualización de pesos son:

$$\mathbf{w}_i(t+1) = \begin{cases} \mathbf{w}_i(t) + \alpha(t)(\mathbf{x} - \mathbf{w}_i(t)) & i \in N_c(t) \\ 0 & \text{en caso contrario} \end{cases} \quad (2-7)$$

El factor  $\alpha(t)$  y la vecindad  $N_c(t)$  se escriben como función del tiempo para expresar que se modifican a medida que avanza el proceso de aprendizaje. Cada vector de peso que participa en la actualización se acerca ligeramente hacia el vector de entrada  $\mathbf{x}$ . Una vez que el entrenamiento haya progresado lo suficiente, el vector de peso de cada neurona irá convergiendo a un valor que sea representativo de las coordenadas de puntos que se encuentren cerca de la ubicación física de la unidad.

Respecto de la conformación de vecindades existen distintas alternativas. Kohonen ilustra a modo de ejemplo alguna de ellas en la siguiente figura.

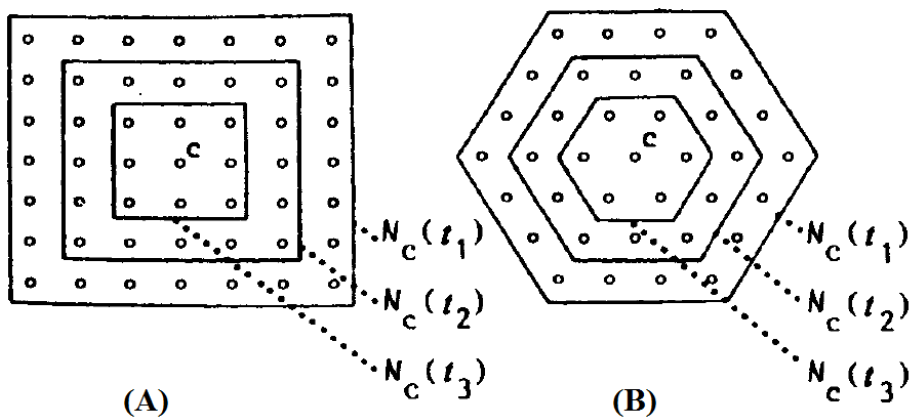


Figura 2-17 Dos ejemplos de vecindades topológicas ( $t_1 < t_2 < t_3$ ). Imagen presentada en [79]

## 2.7 Resumen del Capítulo

Inspirados en la eficiencia de los procesos llevados a cabo por el cerebro, varios investigadores han desarrollado en los últimos 70 años, la teoría de las Redes Neuronales Artificiales (RNA). Al igual que las redes neuronales biológicas, las RNA aprenden de la experiencia, son tolerantes a fallas, pueden procesar información incompleta o distorsionada y son capaces de generalizar el conocimiento adquirido desenvolviéndose correctamente en situaciones nunca antes experimentadas.

El primer modelo de neurona artificial fue presentado por Warren McCulloch y Walter Pitts en 1943. Pero fue el *Perceptrón* diseñado por Frank Rosenblatt en 1957 la primera red neuronal con capacidad de aprendizaje. Luego de un proceso de entrenamiento supervisado con un conjunto de patrones de ejemplos para los que se conoce la clase a la que pertenecen, el *Perceptrón* aprende a clasificar patrones correctamente, incluso aquellos nunca antes presentados a la red. El entrenamiento consiste en la adecuación del vector de pesos  $\mathbf{w}$  del *Perceptrón* que permite reducir paulatinamente la diferencia entre la salida obtenida y la deseada hasta obtener el comportamiento requerido.

Un *Perceptrón* simple tiene capacidad para distinguir sólo 2 clases de patrones linealmente separables. Para ello al resultado del producto escalar entre el vector de entrada  $\mathbf{x}$  y el vector de pesos  $\mathbf{w}$  le aplica una función escalón, que clasifica a  $\mathbf{x}$  según  $\mathbf{x} \cdot \mathbf{w}$  sobrepase un determinado umbral.

La red *Perceptrón* multicapa permite realizar clasificaciones mucho más complejas que un *Perceptrón* simple. Sin embargo, esta arquitectura no se popularizó sino hasta la aparición de un algoritmo de aprendizaje adecuado que dio origen a la red Backpropagation (BPN). El entrenamiento de la red BPN emplea una técnica basada en el gradiente para minimizar el error entre la salida de la red y la salida deseada utilizando el descenso más pronunciado por la

superficie del error. Para ello es necesario reemplazar la función escalón del *Perceptrón* por otra que sea derivable. Las funciones más utilizadas en las neuronas de la red BPN son la identidad y la sigmoide logística.

La red neuronal *Self-Organizing Maps* (SOM) fue desarrollada por Teuvo Kohonen durante el período 1979-1982 y fue una de las redes neuronales más importantes surgidas antes del gran impulso que tomara la disciplina a mediados de la década de 1980. El tipo de aprendizaje que implementa SOM se denomina aprendizaje auto-organizado y se lleva a cabo mediante la aplicación de datos, sin ninguna información sobre la salida deseada de la red.

La red SOM es capaz de proyectar vectores de entrada  $n$ -dimensionales en un mapa topológico de neuronas competitivas de tal forma que las unidades físicamente próximas entre sí respondan a clases de vectores de entrada que análogamente, se encuentren cercanos unos con otros.

El trabajo de Kohonen en la teoría de las RNA ha sido muy importante. En la sección 4.2.4.2 “Implementación de Aprendizaje en Línea Basado en LVQ1” —pág. 130— de esta tesis se propone una variante a un algoritmo de entrenamiento introducido por este autor denominado *Learning Vector Quantization*. Aquí se ha utilizado para contribuir con la actualización del conocimiento que los nodos de la red P2P deben poseer respecto de sus vecinos para garantizar el buen funcionamiento del sistema de búsqueda desarrollado.

## Capítulo 3

### BII-P2P. El método Propuesto

La Búsqueda Inteligente Incremental en redes P2P (BII-P2P), diseñada y desarrollada en el contexto de esta tesis, constituye una implementación de un sistema de búsqueda sobre una red P2P pura, totalmente descentralizada, sin pares especializados y sin estructura como es el caso de la red *Gnutella* 0.4.

En lo sucesivo, al igual que en la mayoría de la literatura existente, cuando en este trabajo se utilice el término *Gnutella* sin más especificaciones, se estará haciendo referencia particularmente a la versión *Gnutella* 0.4, la primera implementación de un sistema P2P puro no estructurado y el más representativo de su clase —ver sección 1.3.4 “Hitos Históricos” en la página 21—. Es decir que por *Gnutella* ha de entenderse la mención a la red P2P totalmente descentralizada, sin estructura, sin ningún tipo de jerarquía ni roles diferenciados en ninguno de sus pares. En caso de ser necesario la referencia a otra versión de *Gnutella* siempre irá acompañada por el número correspondiente como es el caso de *Gnutella* 0.6 que, al jerarquizar algunos nodos, se aparta del modelo de red P2P mencionado anteriormente y pierde relevancia en el contexto de la presente tesis.

En resumen, BII-P2P es una estrategia para la búsqueda de recursos en una red P2P pura, sin estructura, completamente descentralizada, sin ningún tipo de jerarquías ni roles diferenciados, basada en la solución introducida por *Gnutella*.

Sin embargo BII-P2P es mucho más eficiente que *Gnutella* pues resuelve satisfactoriamente las limitaciones más importantes de este último —ver sección 3.1.3 “Limitaciones de *Gnutella*” en la página 93—. Para lograrlo, BII-P2P utiliza una componente de “inteligencia” en todos los pares de la red. De esta forma la búsqueda ciega de recursos que implementa *Gnutella* se convierte en una búsqueda inteligente impulsada por los nodos que toman decisiones locales basadas en el conocimiento adquirido sobre la distribución de los recursos compartidos en la red P2P.

BII-P2P no utiliza tablas de *hash* distribuidas —*Distributed Hash Tables* (DHTs)— por lo tanto no es un sistema estructurado. Las redes P2P que utilizan DHTs para la búsqueda de recursos han mostrado buenos resultados. Sin embargo, la transitoriedad de los nodos tan característica en estos sistemas, causa pocos problemas a las redes desestructuradas como *Gnutella*, pero ocasiona un importante gasto general —*overhead*— en los sistemas que utilizan DHTs [6]. Las DHTs superan a los sistemas tipo *Gnutella* cuando el recurso buscado es escaso, pero si existen muchas instancias del mismo —como suele ser habitual— esta ventaja se desvanece ya que *Gnutella*, con su sencillo protocolo, puede hallarlo fácilmente [7].

Debido a que en lo sucesivo se utilizan conceptos y términos inherentes al protocolo *Gnutella* a continuación se expone brevemente lo más relevante de esta red P2P y su relación con BII-P2P.

### 3.1 Con Base en *Gnutella*

*Gnutella* es un protocolo abierto y descentralizado. Conceptualmente es un sistema distribuido para almacenamiento y búsqueda de información. Sin embargo ha sido utilizado principalmente para compartir archivos. Al igual que otros sistemas P2P *Gnutella* construye una red virtual a nivel de aplicación con sus propios mecanismos para encaminar información entre los nodos.



Fue diseñado para operar en entornos dinámicos donde los nodos puedan unirse y dejar la red con mucha frecuencia sin que ello implique una carga de trabajo considerable. Debido a que se trata de una red P2P donde los pares se comportan como clientes y servidores al mismo tiempo, los nodos que componen una red *Gnutella* suelen denominarse *servents* —palabra que deriva de la conjunción de los términos en inglés *server* y *client*— pues realizan tareas normalmente asociadas a ambos servidores y clientes [24].

La topología de esta red y su sencillo algoritmo para la búsqueda de recursos tienen una influencia significativa en la aplicación distribuida que se implemente sobre ella brindando propiedades tales como rendimiento y confiabilidad.

El enfoque totalmente descentralizado de *Gnutella* para la búsqueda de recursos está basado en la difusión de peticiones —*broadcast*— que inundan la red —*flooding*— llegando a un gran número de pares. Para ello, los *servents* se ajustan a un protocolo que implementa un algoritmo distribuido de búsqueda en anchura o algoritmo BFS —siglas extraídas de su nombre en inglés: *Breadth First Search*—. Aquellos nodos que poseen el recurso buscado responderán a la solicitud.

Una importante desventaja de este esquema de búsqueda es la generación de una gran cantidad de tráfico por la difusión de las peticiones. Para evitar esta situación, *Gnutella* 0.6 introduce una jerarquía de nodos llamados superpares, que almacenan el contenido disponible en los pares conectados, junto con sus direcciones IP. La misión principal de estos superpares es reducir la cantidad de saltos en el proceso de búsqueda reduciendo así el tráfico sobre la red [6]. En el protocolo *Gnutella* 0.6 aparecen otros problemas a resolver como ser los efectos y recuperación de la caída de un superpar. En ese sentido es menos tolerante a fallas, más complicado y costoso de mantener que *Gnutella* 0.4

BII-P2P presentada en esta tesis está basada en *Gnutella* 0.4 manteniendo una

red P2P pura sin estructura, sin ninguna jerarquía ni roles diferenciados. Así se evitan las complicaciones mencionadas para el caso de *Gnutella* 0.6 y se simplifica el mantenimiento y administración del sistema. BII-P2P plantea un enfoque totalmente distinto para superar las limitaciones de *Gnutella* 0.4. En lugar de jerarquizar la red haciéndola menos tolerante a fallas y mucho más complicada de administrar se ha optado por agregar una componente de inteligencia a los pares de la red con la que se pretende hacer un uso más eficiente del ancho de banda disponible en el sistema.

Por lo expresado anteriormente es la versión 0.4 del protocolo *Gnutella* la que resulta de interés en el presente trabajo y por ello se describe brevemente a continuación.

El protocolo *Gnutella* define como se comunican los nodos de la red. Consiste en un conjunto de tipos de mensajes entre los nodos y de reglas que gobiernan su intercambio. Un nodo se conecta a la red estableciendo conexión con otros nodos de la red.

Es un protocolo muy sencillo, no requiere ningún tipo de administración global. Los nodos se unen o dejan la red libremente interactuando localmente con un conjunto de pares a los que se encuentran conectados directamente y que constituyen su vecindad. *Gnutella* posee cinco tipos de mensajes: *Ping* y *Pong* — para descubrir nuevos nodos en la red— *Query* y *QueryHit* —implementan el mecanismo de búsqueda en la red— y *Push* que permite proveer datos a los nodos detrás de un cortafuegos.

Un nodo que desea unirse a la red debe conocer al menos la dirección de otro nodo que forme parte de la misma. El mecanismo para conocer los nodos iniciales no forma parte del protocolo. En general, los nodos iniciales son provistos por nodos Host Server Cache, que se conocen de ante mano.

En la red *Gnutella* 0.4 todos los nodos se conectan en forma de grafo donde

cada nodo es igual a todos los demás no existiendo ningún tipo de jerarquía. Para conectarse crea una conexión TCP con un nodo de la red, y se envía una cadena de solicitud de conexión:

```
GNUTELLA CONNECT /<versión del protocolo>\n\n.
```

Si el otro nodo acepta la conexión debe responder:

```
GNUTELLA OK\n\n.
```

Cualquier otra respuesta se asume como no aceptación.

Los nodos sólo conocen a aquellos nodos con los cuales están conectados directamente. En principio todos los demás nodos de la red son desconocidos. La comunicación se realiza por medio de la difusión de mensajes a todos los vecinos directos. Un *servent* envía un mensaje a sus nodos vecinos y éstos a su vez lo reenvían a sus otros vecinos. De esta forma los mensajes se propagan inundando la red y alcanzando un gran número de nodos.

Todos los mensajes están formados por una cabecera seguida generalmente por una sección de datos. Entre otros campos, en la cabecera se encuentran el ID que se utiliza como identificador de una transacción, y dos campos TTL y HOPS que permiten controlar el tiempo de vida del mensaje en el sistema.

Para explorar la red en busca de nuevos nodos con los cuales conectarse, se emplean los mensajes *Ping*. Aquellos nodos que lo reciben pueden responder con un *Pong* indicando la dirección de algún nodo activo en la red. Un nodo puede enviar más de un mensaje *Pong* como respuesta a un único *Ping*, dado que informa a cerca de otros nodos que tenga almacenado en su memoria cache.

### 3.1.1 Búsqueda de Recursos en la Red *Gnutella*

Para la búsqueda de un recurso, un *servent* realiza un *broadcast* enviando un

mensaje *Query* de petición a todos sus vecinos con los que mantiene una conexión TCP directa. Estos *servents* comprueban si poseen el recurso solicitado y, en caso afirmativo envían al emisor un mensaje de respuesta *QueryHit*. Además retransmiten la petición a todos sus otros vecinos haciendo que el mensaje *Query* se propague hacia nodos más alejados. Los resultados positivos de la petición, mensajes *QueryHit*, se van enviando al *servent* emisor recorriendo el camino inverso realizado por los mensajes *Query* —ver Figura 3-1—.

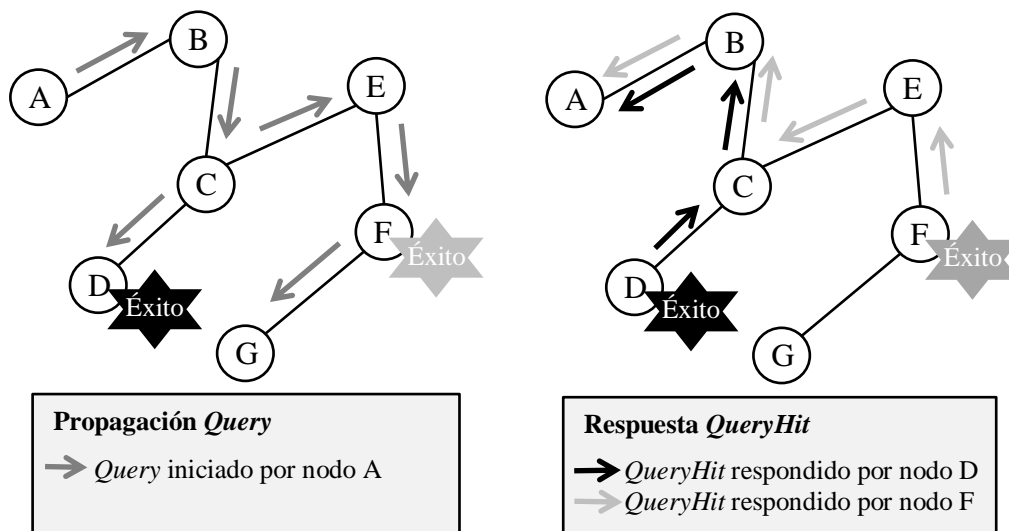


Figura 3-1 Propagación de mensajes *Query* y *QueryHit* en *Gnutella*

### 3.1.2 Normas en la Propagación de los Mensajes

Si un *servent* recibe el mismo mensaje más de una vez simplemente lo descarta —los mensajes son individualizados por el ID de su cabecera—. La eliminación de los mensajes repetidos evita los ciclos innecesarios en las transmisiones causados por los ciclos en la topología de *Gnutella*.

Un *servent* que recibe un nuevo mensaje *Query* lo propaga a todos sus vecinos excepto a aquel del cual lo ha recibido. Los mensajes *QueryHit* son

propagados por la red recorriendo la ruta inversa realizada por los mensajes *Query*. Esto asegura que sólo aquellos *servent* que propagaron el mensaje *Query* verán el mensaje *QueryHit* en respuesta. Esta modalidad fue deliberadamente establecida para proteger el anonimato en la red evitando identificar la dirección del remitente que originó la consulta [25] —un repaso de la sección 1.3 “Visión histórica” que comienza en la página 16 puede ayudar a comprender las motivaciones de los desarrolladores de *Gnutella* que nació en un clima de agitación en relación a las tecnologías P2P—.

Los nodos incrementan en 1 el campo HOPS y decrementan en 1 el campo TTL de los mensajes antes de propagarlos a sus vecinos. Sin embargo, si una vez decrementado, el valor del TTL llega a cero el mensaje es descartado. Esto permite establecer el alcance de cada mensaje evitando que la red colapse debido al crecimiento exponencial de mensajes *Query* conforme aumenta el número de *hops* permitidos.

### 3.1.3 Limitaciones de *Gnutella*

El último párrafo del apartado anterior evidencia claramente el mayor problema del protocolo implementado en *Gnutella*: su escalabilidad deficiente. Por citar un ejemplo, *Gnutella* no escala bien hasta los niveles que hoy en día demandan las aplicaciones de *file-sharing* en la Internet. Es precisamente este tipo de aplicación para el cual fue desarrollado el protocolo.

La utilización del campo TTL en la cabecera de los mensajes evita el colapso de la red a costa de limitar la propagación de las solicitudes. Esto convierte en inalcanzables a aquellos nodos que se encuentren a mayor distancia del radio impuesto por este valor.

La estrategia BII-P2P propuesta en esta tesis resuelve el problema de escalabilidad de *Gnutella*. Las búsquedas en BII-P2P, si bien utilizan como base el algoritmo BFS de *Gnutella*, resultan más eficientes al reemplazar la difusión

ciega de los mensajes *Query* a todos los vecinos por una selectiva en la que, en función del recurso buscado, sólo se propaga la solicitud a un subconjunto reducido de vecinos. En base a mecanismos de aprendizaje y acción que se detallan en el capítulo siguiente, con alta probabilidad este subconjunto selecto de vecinos resultará el más adecuado, garantizando el éxito de la búsqueda y minimizando las distancias que los mensajes *Query* deban recorrer hasta hallar el recurso buscado.

Para lograrlo, BII-P2P se vale de una componente inteligente en los nodos de la red que les permite aprender cuáles son las clases de recursos que existen distribuidos en la red P2P y que pueden ser encontrados fácilmente a través de cada uno de sus vecinos. Esta inteligencia ha sido implementada en cada uno de los nodos por medio de sendas redes neuronales para las que se ha desarrollado un algoritmo de entrenamiento especialmente diseñado para este problema.

## 3.2 Escenario de Aplicación

Antes de proceder con la exposición de los detalles técnicos de la estrategia BII-P2P —en los dos próximos capítulos—, es necesario realizar algunas consideraciones. Con ellas se pretende establecer con precisión el escenario donde la solución presentada, ha mostrado los mejores resultados, alcanzando mayor porcentaje de hallazgos y generando menor cantidad de tráfico sobre la red.

Los enunciados expuestos a continuación están relacionados con la naturaleza de la distribución de los recursos compartidos en una red P2P. Sin embargo, como se verá inmediatamente, los supuestos asumidos son de carácter general y contemplan prácticamente cualquier aplicación de uso común sobre los sistemas P2P.

Considérese a una red P2P como un gran conjunto de recursos distribuidos entre los nodos que la componen. Sin pérdida de generalidad es válido asumir que

existen diferentes clases de estos recursos con diversos grados de similitud repartidos en el sistema con alguna disposición determinada. También es acertado suponer múltiples instancias de un mismo recurso replicadas en todo el sistema aunque con diversos grados de concentración en distintas regiones de la red P2P.

Las suposiciones del párrafo anterior es todo cuanto se necesita para que BII-P2P resulte una opción ventajosa debido a su habilidad para aprender y explotar las particularidades de la distribución de recursos en el sistema. Obsérvese que se trata de una restricción sumamente débil ya que no se impone ninguna distribución en particular sino, por el contrario, asume cualquiera excepto el caso en el que los recursos se encuentren distribuidos de manera uniforme sobre la red. En este último caso los recursos compartidos no formarían ningún patrón que BII-P2P pudiese aprender para utilizar convenientemente.

Prácticamente en toda implementación real de un sistema P2P los recursos compartidos no se reparten uniformemente entre los nodos sino que, por el contrario, surgen espontáneamente patrones de distribución debido a las características propias de la red P2P, las aplicaciones y el perfil de los usuarios que las utilizan. Los nodos de un sistema real pueden ser heterogéneos desde muchos puntos de vista: diferentes hardware con distintas capacidades, diferentes sistemas operativos, diferentes servicios, distintos procesos en ejecución, distinta información almacenada, distintos usuarios, etc. Esta heterogeneidad y las particularidades de la topología de la red debilitan la uniformidad haciendo que los nodos respondan de distinta manera ante una solicitud de un determinado recurso.

A modo de ejemplo considérese la situación de una red P2P cuyos recursos compartidos son textos digitalizados. Si el número de documentos compartidos en todo el sistema es grande, seguramente existirán distintos tipos de documentos siendo posible, en base a similitudes y diferencias, practicar numerosas clasificaciones de los mismos. Eventualmente existirán muchas copias de algunos

documentos —instancias del mismo recurso— que se replican en distintos nodos de la red. También es muy probable que los nodos no posean una colección aleatoria uniformemente distribuida de documentos de cualquier tipo sino más bien que respondan a un determinado perfil relacionado con el gusto o necesidad de los usuarios de ese nodo. El conocimiento de la disposición particular con que se mapean las clases de documentos sobre los distintos nodos de la red permite el diseño de un sistema de búsqueda eficiente que utilice esta información convenientemente. Esta es la idea principal que se ha desarrollado en BII-P2P.

Las consideraciones presentadas en el ejemplo anterior no constituyen un caso especial sino que se aplican en general a cualquier otro tipo de sistema P2P en el que se comparte algún tipo de recurso de hardware o software.

BII-P2P implementa un mecanismo para adquirir conocimiento sobre cómo se distribuyen las clases de recursos en la red P2P y otro para conducir, en base a este conocimiento, las búsquedas sobre la red en forma más eficiente. Así es posible, determinar a qué clase pertenece el recurso buscado y dirigir las solicitudes sólo hacia las regiones de la red más convenientes para esa clase de recursos en particular. Además esto se lleva a cabo con un algoritmo totalmente descentralizado, por el accionar coordinado de todos los nodos de la red quienes llevan a cabo su rol participando del sistema de búsqueda por medio de acciones en base a decisiones locales.

Si la red P2P presentara una distribución aleatoria uniforme de sus recursos compartidos todas las regiones de la red resultarían igualmente aceptables para hallar el recurso buscado. En este caso la habilidad de BII-P2P para explotar el conocimiento sobre cómo se distribuyen las clases de recursos en la red, perdería relevancia.

Para comprender la idea subyacente en BII-P2P y la forma en que es posible obtener ventajas de una determinada distribución de recursos, es necesario prestar



atención al escenario que ofrece una red completamente descentralizada sin estructura como es el caso de *Gnutella*. Las conexiones directas de los nodos con un conjunto de varios vecinos originan múltiples caminos por los cuales dos pares de la red, distantes entre sí, pueden comunicarse.

Estas múltiples vías de comunicación entre pares son las que se utilizan para el envío de los mensajes *Query* que propagan una solicitud de búsqueda determinada. Si  $n_i$  es un nodo que inicia una solicitud de búsqueda por el recurso  $r$  y  $n_j$  es un nodo poseedor de  $r$ , de todas las posibles trayectorias existentes entre  $n_i$  y  $n_j$  la más corta es la más atractiva porque con ella se genera menor cantidad de tráfico sobre la red. Pero además, existen otras muchas trayectorias indeseables que llevan la solicitud de  $n_i$  hacia regiones infructuosas de la red donde no se encuentra el recurso  $r$  alejándose de las zonas más prometedoras.

Así, el camino que un mensaje *Query* debe recorrer desde el nodo que inició la búsqueda hasta hallar el recurso buscado puede ser más corto y fructífero —menor cantidad de saltos y mayor cantidad de instancias del recurso halladas—, si los nodos involucrados en el trayecto propagan la solicitud hacia un subconjunto específico de vecinos. Por lo tanto, cada vez que un nodo debe propagar una solicitud de búsqueda, dependiendo del recurso buscado y en relación a la probabilidad de éxito, algunos vecinos pueden presentarse como más prometedores que otros para conducir esa búsqueda en particular.

Desde el punto de vista del nodo que propaga una solicitud de búsqueda, sus vecinos constituyen la vía de acceso a distintas regiones —posiblemente solapadas— de la red P2P. Por lo tanto cada nodo puede percibir a la red P2P como un gran repositorio de recursos dividido en tantas partes como vecinos posea. Estas partes o regiones de la red presentan sus propias características y responden a distintos perfiles en cuanto a cantidad, disponibilidad y clases de recursos que se comparten a través de ellas.

Justamente la idea subyacente en BII-P2P es dotar a los nodos de la red con el conocimiento sobre qué clases de recursos pueden hallarse con facilidad a partir de cada uno de sus vecinos. Esto resulta de suma utilidad para implementar un mecanismo eficiente de búsqueda que dirija los mensajes *Query* sólo por las rutas más prometedoras haciendo un mejor aprovechamiento del ancho de banda disponible.

BII-P2P instrumenta los mecanismos necesarios para que los nodos de la red utilicen el conocimiento referido anteriormente a su favor. Para ello se ha diseñado un sistema de aprendizaje que permite a los nodos de la red conocer qué clases de recursos pueden ser accedidos fácilmente a través de cada uno de sus vecinos. Sin embargo, es necesario remarcar que este conocimiento puede quedar desactualizado rápidamente. El alto grado de dinamismo que distingue a las redes P2P hace que cualquier caracterización realizada sobre ellas sólo sea válida por períodos de tiempos relativamente pequeños.

Por lo tanto, BII-P2P no sólo debe conocer la forma en que los recursos se distribuyen entre los nodos sino también ha de poseer los mecanismos apropiados para mantener este conocimiento actualizado o de lo contrario perdería eficiencia rápidamente. En la sección 4.2.3 “Políticas de Adquisición y Actualización de Conocimiento” en la página 123 se discute este aspecto de BII-P2P.

### 3.3 Los dos Aspectos Destacados en BII-P2P

En BII-P2P se destacan dos aspectos importantes: *i)* búsqueda inteligente y *ii)* exploración incremental. Entre ellos, la implementación de la búsqueda inteligente constituye el aporte más importante y representativo de BII-P2P, potenciado luego con el aditamento de la estrategia de exploración incremental de la red.

El término “búsqueda inteligente”, que forma parte del nombre de la

propuesta presentada en este trabajo de tesis, se ha elegido en contraposición al tipo de búsqueda ciega que lleva a cabo el algoritmo BFS de *Gnutella*, pero sobre todo debido a la utilización intensa que el sistema de búsqueda hace de las redes neuronales artificiales, uno de los objetos de estudio más importante de la rama subsimbólica de la disciplina Inteligencia Artificial.

La implementación de redes neuronales en cada nodo de la red P2P transforma a la búsqueda que lleva a cabo BII-P2P en un proceso basado en la adquisición de conocimiento y la toma de decisiones, características propias de los sistemas inteligentes.

En los Capítulos 4 y 5 se desarrollan tanto los conceptos más generales como los detalles de la implementación, pruebas y resultados de la búsqueda inteligente en BII-P2P por lo que han de considerarse capítulos centrales de esta tesis.

Los resultados experimentales evidencian que la búsqueda inteligente utiliza más eficientemente el ancho de banda disponible y encuentra mayor cantidad de instancias del recurso buscado que el algoritmo BFS de *Gnutella* —ver Capítulo 5 “Evaluación de la Búsqueda Inteligente” en la página 137—. Sin embargo, combinándola con una adecuada estrategia de exploración incremental, los resultados pueden ser aún mucho mejores.

La exploración incremental consiste en permitir que los nodos comiencen con una búsqueda inicial mínima, economizando recursos de ancho de banda, sobre una región acotada de la red. El nodo iniciador de la búsqueda, de acuerdo a sus necesidades y a las respuestas obtenidas puede emitir solicitudes de ampliación para explorar nuevas regiones de la red en sucesivos incrementos hasta considerar sus necesidades satisfechas, o bien desistir.

Los conceptos generales y detalles sobre la implementación, las pruebas y los resultados de la búsqueda incremental en BII-P2P se exponen en el Capítulo 6, por lo que también éste último ha de considerarse uno de los capítulos centrales de

la presente tesis.

### 3.4 Resumen del Capítulo

La Búsqueda Inteligente Incremental en redes P2P (BII-P2P) extiende el algoritmo BFS de *Gnutella* para la búsqueda de recursos, dotando a los nodos con la inteligencia necesaria para conducir las peticiones de búsqueda de manera selectiva y eficiente.

*Gnutella* es un protocolo muy sencillo. Posee sólo cinco mensajes distintos de los cuales sólo dos se utilizan para implementar el mecanismo de búsqueda: *Query* y *QueryHit*. Para la búsqueda de un recurso, un nodo *Gnutella* envía una petición por medio de un mensaje *Query* a todos sus vecinos. Éstos contestan con un mensaje *QueryHit* en caso de poseer el recurso buscado, pero además retransmiten el *Query* a todos sus otros vecinos. Los mensajes *QueryHit* llegan al nodo que originó la petición siguiendo el camino inverso realizado por los mensajes *Query*.

*Gnutella* no es escalable. Para evitar el colapso de la red debido al crecimiento exponencial de mensajes durante la propagación de una petición, los nodos decrementan en 1 el campo TTL antes de su retransmisión, que realizan sólo en caso que el valor del TTL no haya llegado a cero. Quedan entonces fuera del alcance de la búsqueda los nodos más allá del radio impuesto por el TTL.

La búsqueda en BII-P2P resulta mucho más eficiente que la estrategia BFS de *Gnutella* al reemplazar la difusión ciega de peticiones a todos los vecinos por una selectiva de acuerdo al recurso buscado y al perfil aprendido de cada uno de ellos. Con alta probabilidad este subconjunto selecto de vecinos resultará el más adecuado, garantizando el éxito de la búsqueda y minimizando el tráfico generado por la propagación de los mensajes *Query*. BII-P2P excluye de la búsqueda a los nodos menos apropiados para responderla y no a los más alejados como ocurre en

el caso del algoritmo BFS de *Gnutella*.

BII-P2P implementa un mecanismo para adquirir conocimiento sobre cómo se distribuyen las clases de recursos en la red P2P y otro para conducir, en base a este conocimiento, las búsquedas sobre la red en forma más eficiente. Además esto se lleva a cabo con un algoritmo totalmente descentralizado, por el accionar coordinado de todos los nodos de la red que participan del sistema de búsqueda por medio de acciones en base a decisiones locales.

BII-P2P también implementa políticas de actualización de conocimiento para hacer frente al alto grado de dinamismo que distingue a las redes P2P y que hace que cualquier caracterización aprendida sobre ellas sólo sea válida por períodos de tiempos relativamente pequeños.

BII-P2P presenta dos aspectos importantes. Los párrafos anteriores están relacionados con uno de ellos: la **búsqueda inteligente**. Este aspecto constituye el aporte más importante y representativo de BII-P2P. En los capítulos 4 y 5 se desarrollan los conceptos generales, detalles de implementación, pruebas y resultados de la búsqueda inteligente en BII-P2P por lo que constituyen los capítulos centrales de esta tesis.

El otro aspecto destacado de BII-P2P, es la estrategia de **exploración incremental** que potencia a la búsqueda inteligente. La exploración incremental permite a los nodos realizar búsquedas iniciales mínimas sobre una región acotada de la red para luego, en función de sus necesidades y respuestas obtenidas ampliar la región explorada en sucesivos incrementos. En el Capítulo 6 se presentan los conceptos generales y detalles sobre la implementación, pruebas y resultados de la búsqueda incremental en BII-P2P. Por lo tanto también este último capítulo constituye uno de los más significativos de esta tesis.



## Capítulo 4

# Búsqueda Inteligente en BII-P2P

La búsqueda inteligente implementada en BII-P2P se concentra en la forma de encaminar adecuadamente las solicitudes —mensajes *Query*— difundidas por los nodos que participan en una búsqueda concreta. Ellos deciden a qué vecinos propagar el mensaje y a cuáles no hacerlo. Se busca que las decisiones locales tomadas por los nodos sirvan para minimizar el tráfico generado por el sistema y aumentar la probabilidad de hallar el recurso buscado.

Estas decisiones se toman en los nodos en base al conocimiento de los perfiles aprendidos de sus vecinos y al recurso en particular buscado. Para ello, los nodos implementan mecanismos para adquirir, ajustar y mantener la habilidad de dirigir la búsqueda sólo hacia las regiones más apropiadas de la red P2P, evitando el uso innecesario de ancho de banda y mejorando en consecuencia la eficiencia del sistema de búsqueda.

De esta forma BII-P2P promueve la utilización de los caminos más prometedores, aquellos que garanticen la probabilidad más alta de conducir con éxito la búsqueda de ese recurso específico y se descartan aquellos evaluados como demasiado largos o infructuosos.

Es preciso remarcar que este conocimiento acerca de las rutas más adecuadas para la búsqueda de cada recurso en particular no se encuentra localizado en

ningún punto específico de la red, sino que por el contrario permanece distribuido entre todos los nodos que la componen. El accionar individual de cada uno de ellos en base a decisiones locales tiene efectos globales sobre el sistema promoviendo el descubrimiento de estas rutas apropiadas.

Los nodos deben adquirir y mantener actualizado el conocimiento acerca de su entorno cercano y más distante sólo a través de la interacción directa con un reducido grupo de nodos —sus vecinos—. Otra forma utilizada para la adquisición de conocimiento es el análisis de las solicitudes de búsqueda y respuestas positivas en las que el nodo participa o bien como actor principal o bien como punto intermedio en la ruta de estos mensajes. Así los nodos poseen la capacidad de aprender adquiriendo información sustancial para implementar una búsqueda de recursos eficiente. Este conocimiento es utilizado por el nodo para inferir probabilidades de éxito y tomar decisiones en cada caso particular de búsqueda. Dependiendo entonces de las características particulares del recurso buscado, el nodo realizará una valoración sobre cuáles de sus vecinos son los más apropiados para encaminar la búsqueda de ese recurso específico.

La valoración antes mencionada se realiza por medio de una red neuronal artificial implantada en cada nodo del sistema P2P. Esta red neuronal es la responsable de asistir al nodo en sus decisiones de propagación de mensajes *Query*. Al mismo tiempo implementa un mecanismo de adquisición continua de conocimiento. Esta capacidad de aprendizaje se obtiene a través de un mecanismo de intercambio de información específicamente desarrollado en esta propuesta que suplanta los algoritmos tradicionales de entrenamiento. Estos conceptos de Inteligencia Artificial constituyen los cimientos de BII-P2P y convalidan la expresión “Búsqueda Inteligente” en el nombre de la estrategia presentada.

Concretamente, la red neuronal implementada en cada nodo del sistema P2P, le aporta al mismo el conocimiento sobre qué clases de recursos pueden hallarse más rápidamente a partir de cada uno de sus vecinos. Esta información es



utilizada para propagar los mensajes selectivamente, sólo por los caminos más adecuados haciendo un uso más eficiente del ancho de banda disponible.

Mientras que el algoritmo BFS de *Gnutella* alcanza sólo a los nodos más cercanos —aquellos que se encuentran dentro del radio definido por el TTL—, BII-P2P llega a nodos más alejados considerados como posibles propietarios del recurso. Sin embargo, esto no implica un aumento en la cantidad de tráfico generado en la red pues BII-P2P lo compensa excluyendo de la búsqueda a aquellos nodos con menores chances de satisfacerla, aún en el caso de estar muy cerca del nodo que la originó.

## 4.1 El Rol de los Nodos en BII-P2P

En ausencia de cualquier tipo de control central, la implementación de un algoritmo con efectos globales en un sistema P2P completamente descentralizado debe llevarse a cabo por el accionar individual y coordinado de cada uno de los nodos que lo conforman. Esta es precisamente la forma en que se ha diseñado BII-P2P que por medio de la toma de decisiones locales y la interacción entre nodos vecinos, implementa un mecanismo eficiente de búsqueda global de recursos basado en la difusión selectiva de mensajes.

### 4.1.1 Conceptos Generales

El algoritmo definido en BII-P2P está basado en la búsqueda BFS que realiza *Gnutella*. No obstante, a diferencia de este último que utiliza un *broadcast* de alcance limitado por un campo TTL en la cabecera de sus mensajes —ver secciones 3.1.1 “Búsqueda de Recursos en la Red *Gnutella*” y 3.1.2 “Normas en la Propagación de los Mensajes” a partir de la página 91—, en BII-P2P los nodos conducen la búsqueda sólo a través de un subconjunto de vecinos —los más apropiados— y no hacia todos ellos como es el caso de *Gnutella*. De esta manera, la menor cantidad de mensajes *Query* utilizada en BII-P2P disminuye los

requerimientos de ancho de banda permitiendo entonces valores de TTL más grandes alcanzando así a nodos más alejados.

A diferencia de “*Modified Random BFS*” [2], en BII-P2P la selección de los vecinos a los cuales propagar una solicitud de búsqueda en particular se realiza concienzudamente y de manera específica para cada recurso buscado por medio de la utilización de redes neuronales artificiales que representan uno de los aspectos más relevantes y distintivos del sistema de búsqueda propuesto.

El hallazgo de un camino conveniente para conducir una búsqueda concreta estará supeditado a una sucesión de “buenas decisiones” tomadas por todos los nodos involucrados en la trayectoria del mensaje de solicitud —aquel que originó el mensaje y todos los nodos intermedios—. Por lo tanto, el rendimiento del sistema de búsqueda es directamente proporcional a la habilidad de los nodos para inferir un subconjunto, preferentemente reducido, de vecinos apropiados hacia los cuales propagar la solicitud del recurso.

Como ya se anticipara en párrafos previos, las decisiones locales que direccionan los mensajes *Query* en un nodo de la red P2P se toman de acuerdo a la red neuronal implementada en el mismo. Esta red neuronal evalúa las características propias del recurso buscado en relación al perfil conocido de cada uno de los vecinos del nodo.

A continuación se discuten los detalles de su implementación, arquitectura, funcionamiento y modo particular de aprendizaje que dan forma al sistema de búsqueda inteligente e impulsan la mejora del rendimiento general del sistema.

#### 4.1.2 La Red Neuronal de los Nodos P2P

La red neuronal implementada en cada nodo se encuentra inspirada en las arquitecturas compuestas por unidades competitivas —ver la sección 2.6.2 “La Neurona Competitiva” en la página 81—. La habilidad de este tipo de

arquitecturas para realizar agrupamientos o clasificaciones ha sido ampliamente comprobada.

El trabajo que deben realizar los nodos de la red P2P para llevar a cabo la búsqueda BII-P2P es similar a una tarea de clasificación. Cada nodo realiza su propia clasificación. Los objetos a clasificar son los recursos que se comparten en el sistema. La cantidad de clases en las que se agrupan dichos recursos es igual a la cantidad de vecinos directos que posee el nodo. Así se establece una correspondencia uno a uno entre clases de recursos y nodos vecinos.

Si un recurso  $r$  es clasificado por la red neuronal de un nodo como perteneciente a la clase  $j$ , esto indica que la región de la red P2P más apropiada para hallar  $r$  es la que se encuentra accesible a través del vecino  $j$ . Es decir que, o bien el vecino  $j$  es poseedor de  $r$  o bien a partir de él se encuentran los caminos con mayor probabilidad de éxito. Estos caminos son aquellos con mayor cantidad de instancias de  $r$  y con menor número de nodos intermedios —menos tráfico generado en la red y respuestas positivas más rápidas—.

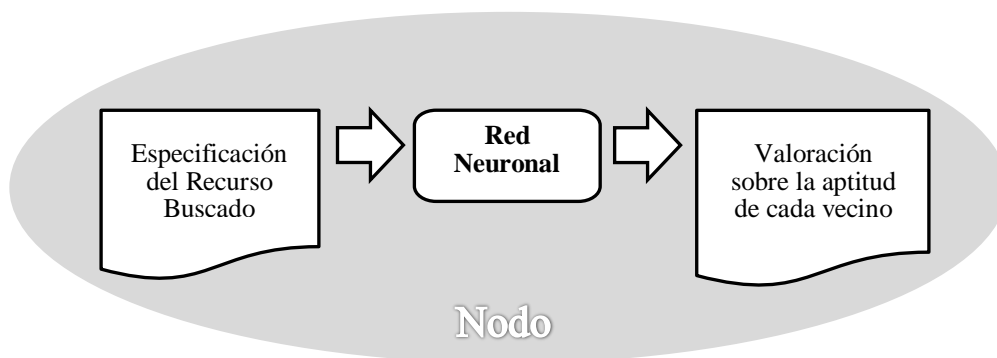
Sin embargo, para poder identificar más de un camino por donde direccionar la búsqueda de un recurso  $r$ , en lugar de una clasificación categórica de  $r$ , el nodo de la red necesita establecer una valoración sobre la pertenencia de  $r$  a cada una de las clases disponibles. De esta forma se eligen para propagar la solicitud de búsqueda a aquellos vecinos correspondientes a las clases para las cuales la valoración sobre la pertenencia de  $r$  ha sido más alta.

Por lo tanto, el rol desempeñado por los nodos en la búsqueda BII-P2P se aparta de un típico sistema clasificador mostrando algunas diferencias, consecuentemente la red neuronal implementada en cada nodo también se diferencia de la mayoría de las arquitecturas competitivas.

En BII-P2P la salida de la red neuronal de cada nodo consiste en un vector con los valores de activación de las neuronas de la capa de salida —cada neurona

está asociada unívocamente a un nodo vecino—. Todos estos valores resultan relevantes, en contraposición a lo que sucede habitualmente con las arquitecturas competitivas en donde sólo interesa la identificación de la neurona ganadora.

La salida de la red neuronal establece un ordenamiento entre los vecinos que permite conducir la búsqueda sólo a través de los más prometedores —aquellos que ocupan las primeras posiciones en este ordenamiento—. Más adelante se exponen los detalles sobre el funcionamiento y aprendizaje de esta red neuronal facilitando la comprensión de cómo se genera, de dónde surge y de qué manera se mantiene esta capacidad del nodo para estimar la probabilidad de hallar el recurso a través de cada uno de sus vecinos —ver sección 4.2 “El Aprendizaje” en la página 117—.



**Figura 4-1 Participación de la red neuronal en el sistema de búsqueda**

La Figura 4-1 muestra esquemáticamente cuál es el rol de la red neuronal dentro del nodo. La red neuronal es estimulada con las especificaciones que caracterizan el recurso buscado. La evaluación de la misma produce la salida correspondiente. Esta salida es interpretada por el nodo como un orden de mérito, una valoración sobre qué tan conveniente resulta propagar la búsqueda hacia los demás nodos vecinos.

La red neuronal trabaja sobre una representación vectorial de los recursos

compartidos. Este tipo de representación es comúnmente utilizada con redes neuronales y en algoritmos de *clustering*. Si  $\mathbf{v}$  es la representación vectorial del recurso  $r$ , entonces se dice que  $\mathbf{v}$  es el vector característico de  $r$ . La forma de construir los vectores característicos depende de las clases de recursos que se comparten en la red y la aplicación distribuida que corre sobre la misma.

Existen muchas formas posibles de construir los vectores característicos de los recursos. Sin embargo estas consideraciones son ajenas al sistema de búsqueda en sí mismo y en el presente trabajo no se asume ninguna representación en particular. Alcanza con observar que independientemente de la naturaleza de los recursos compartidos en la red P2P, la implementación de BII-P2P requiere la obtención de un conjunto de características cuantificables que sirvan para representar cada recurso en el sistema. La cantidad de características utilizadas determina la dimensión del vector y los valores cuantificados conforman el vector característico del recurso. La única suposición que se hará al respecto es que todos los nodos de la red P2P obtienen los vectores característicos de la misma forma y que todos los vectores tendrán la misma dimensión independientemente del recurso que caractericen.

Sólo para ilustrar el concepto considérese un sistema con vectores característicos binarios donde cada posición indica la existencia o no de una determinada característica en el recurso. Supóngase que el sistema P2P se utiliza para compartir artículos científicos. Cada elemento del vector podría representar la existencia o no de una determinada palabra clave en el artículo. Para ello los vectores deben ser suficientemente grandes reservando una posición para cada una de las posibles palabras claves existentes en la colección completa de artículos distribuidos en la red. Otra posibilidad es que cada posición del vector se asocie a una familia entera de términos —sinónimos, por ejemplo— implementando así un sistema de búsqueda semántica capaz de encontrar artículos relacionados por conceptos más que por términos específicos.

BII-P2P utiliza la red neuronal de la siguiente manera: El nodo que inicia la búsqueda de un recurso  $r$  estimula su red neuronal con la representación vectorial  $\mathbf{v}$  del recurso  $r$ . Como resultado, la red neuronal devuelve un vector de salida  $\mathbf{o}$ . La dimensión de  $\mathbf{o}$  se corresponde con el número de vecinos que posee el nodo y sus elementos representa la valoración —un número entre 0 y 1— sobre la conveniencia de propagar la solicitud del recurso por el vecino correspondiente.

Una vez obtenido este orden de preferencia, se determina la cantidad de nodos vecinos a los cuales propagar la búsqueda. Esta cantidad se fija en función del tamaño de la vecindad y de la tasa de selección  $\gamma$ , un parámetro conocido previamente por todos los nodos y que representa la proporción de vecinos a elegir. El valor de  $\gamma$  impacta fuertemente sobre el rendimiento del sistema de búsqueda y debe ser elegido con precaución. En la sección 5.1.3 “Resultados de la Experimentación” en la página 142 se presenta una gran cantidad de casos de prueba que arrojan algunos indicios sobre los posibles valores adecuados para  $\gamma$  y su relación con el valor del parámetro TTL con que los nodos inician las búsquedas.

### 4.1.3 Arquitectura Neuronal

La Figura 4-2 muestra la arquitectura elegida para construir la red neuronal que toma las decisiones en cada uno de los nodos. Como se observa, es de tipo *feedforward* con dos capas totalmente conectadas. Es decir que cada una de las neuronas de la capa de entrada —etiquetadas en el gráfico con  $v_1, v_2, \dots, v_m$ — se conecta con cada una de las neuronas de salida —etiquetadas en el gráfico con  $o_1, o_2, \dots, o_n$ —. Como se ha visto en el Capítulo 2 una conexión neuronal es una vía de comunicación unidireccional por la que se propaga el valor de activación de la neurona presináptica hacia la neurona postsináptica. Como es habitual en cualquier arquitectura neuronal, las conexiones existentes poseen un valor llamado peso de conexión —etiquetados en el gráfico con  $w_{ij}$  para  $1 \leq i \leq n$  y  $1 \leq j \leq m$ —, que habitualmente se asocia a la importancia de dicha conexión en

la red neuronal.

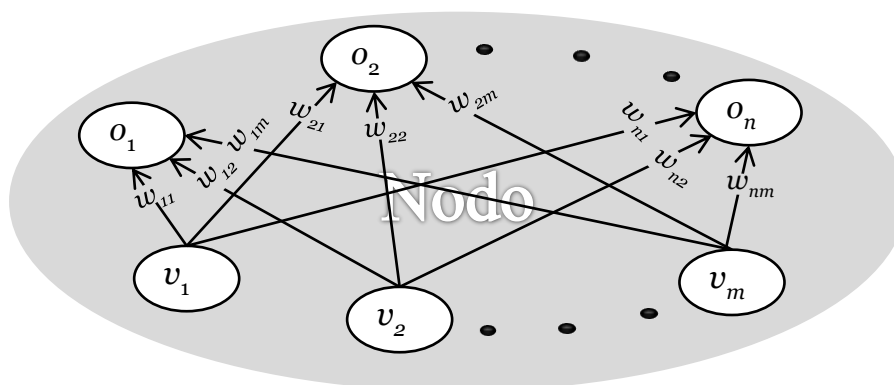


Figura 4-2 Arquitectura neuronal

Obsérvese también que la red neuronal exhibida en la Figura 4-2 ha de estar implantada en un nodo que posee exactamente  $n$  vecinos —tamaño de la capa de salida— y que forma parte de un sistema P2P donde los recursos compartidos se representan por medio de vectores característicos en un espacio  $m$ -dimensional —cantidad de neuronas de entrada—.

La capa de entrada, como es usual en el paradigma, se utiliza sólo para ingresar y distribuir los datos —vector de entrada— a la red neuronal. Estas neuronas no hacen procesamiento alguno sino que su valor de activación es simplemente el elemento del vector de entrada correspondiente a la posición que dicha neurona ocupa en la capa de entrada.

Es posible obtener una descripción matemática de la red neuronal considerando únicamente los pesos de sus conexiones. Obsérvese que el conjunto de todas las conexiones entrantes a la  $i$ -ésima neurona de la capa de salida puede representarse con un vector de peso  $\mathbf{w}_i$  con tantos elementos como conexiones de entrada tenga la neurona. En la red neuronal de la Figura 4-2 cada vector  $\mathbf{w}_i$  tiene exactamente  $m$  elementos, denotando que  $m$  es también el tamaño de los vectores

característicos de los recursos en el sistema.

Por lo tanto, si  $n$  es el tamaño del vector de salida —cantidad de vecinos del nodo— y  $m$  es el tamaño de los vectores de entrada —vectores característicos de los recursos— es posible describir completamente la red neuronal por medio de una matriz de pesos  $\mathbf{W}$  de tamaño  $n \times m$  donde la fila  $i$  está definida por el vector de pesos de la  $i$ -ésima neurona y representa el conocimiento que el nodo posee a cerca de los recursos que pueden hallarse a través del nodo vecino  $i$ .

$$\mathbf{W} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1m} \\ w_{21} & w_{22} & \dots & w_{2m} \\ \vdots & \vdots & & \vdots \\ w_{n1} & w_{n2} & \dots & w_{nm} \end{bmatrix} \quad (4-1)$$

#### 4.1.4 Funcionamiento de la Búsqueda Inteligente

En líneas generales ya ha sido expuesto el rol de los nodos P2P y de sus redes neuronales en el sistema de búsqueda inteligente. En los sucesivos párrafos se exponen los detalles que completan la descripción del algoritmo.

##### 4.1.4.1 Funcionamiento de la Red Neuronal

Sean  $nodo_k$  y  $rn_k$  el nodo  $k$  del sistema P2P y su respectiva red neuronal. Nuevamente considérese la situación en la que  $nodo_k$  posee exactamente  $n$  vecinos conectados en forma directa a él. Cuando  $nodo_k$  inicia la búsqueda de un recurso  $r$ , estimula  $rn_k$  con la representación vectorial  $\mathbf{v} = (v_1, v_2, \dots, v_m)$  de  $r$ . Como resultado de esta evaluación la red neuronal produce la salida  $\mathbf{o} = (o_1, o_2, \dots, o_n)$ . Cada elemento  $o_i$  representa la valoración sobre la aptitud del  $i$ -ésimo nodo vecino para conducir la búsqueda de  $r$ . Obsérvese además que  $o_i$  es el valor de activación de la  $i$ -ésima neurona de la capa competitiva de  $rn_k$ . Por lo



tanto, cada neurona de esta capa se especializa en el conocimiento acerca de un único vecino de  $nodo_k$ .

Sea  $\mathbf{w}_i = (w_{i1}, w_{i2}, \dots, w_{im})$  el vector de peso de la  $i$ -ésima neurona de la capa de salida de  $rn_k$ . El valor de  $m$  está determinado por la dimensión que tienen los vectores característicos de los recursos compartidos. Este es un parámetro fijo conocido por todos los nodos del sistema P2P. Como consecuencia del proceso de aprendizaje que se expone más adelante —ver sección 4.2 “El Aprendizaje” en la página 117—, desde el punto de vista de  $nodo_k$   $\mathbf{w}_i$  constituye un representante promedio de la clase de recursos fácilmente accesibles a partir de su vecino  $i$ . Por lo tanto, se ha definido la función de activación de la  $i$ -ésima neurona de la capa de salida de  $rn_k$  de la siguiente manera:

$$d(\mathbf{v}, \mathbf{w}_i) = \sqrt{(\mathbf{v}_1 - \mathbf{w}_{i1})^2 + \dots + (\mathbf{v}_m - \mathbf{w}_{im})^2} \quad (4-2)$$

$$o_i = \frac{1}{1 + d(\mathbf{v}, \mathbf{w}_i)} \quad 1 \leq i \leq n \quad (4-3)$$

Obsérvese que  $d(\mathbf{v}, \mathbf{w}_i)$  calcula la distancia euclídea entre los vectores  $\mathbf{v}$  y  $\mathbf{w}_i$ . Por lo tanto el valor de  $o_i$  será mayor cuanto más cercanos sean entre sí  $\mathbf{v}$  y  $\mathbf{w}_i$ . Nótese que  $o_i \in (0,1]$ , y si  $o_i = 1$  significa que  $\mathbf{v} = \mathbf{w}_i$ . En el otro extremo, observe que  $o_i \rightarrow 0$  cuando  $d(\mathbf{v}, \mathbf{w}_i) \rightarrow \infty$ , es decir que cuanto más lejanos se encuentren los vectores  $\mathbf{v}$  y  $\mathbf{w}_i$  entre sí, más bajo será el valor de  $o_i$ .

Considérese la situación en que el vecino  $i$  resulta ser el más apropiado para que  $nodo_k$  dirija la búsqueda de  $r$  en el sistema P2P. Si  $rn_k$  ha alcanzado un buen nivel de conocimiento acerca de sus vecinos, una vez evaluada con la entrada  $\mathbf{v}$ , el vector característico de  $r$ , debe ocurrir que  $o_i > o_j$  para todo  $j \neq i$ . Para ello el

vector  $\mathbf{w}_i$  debe ser más cercano a  $\mathbf{v}$  que cualquier otro vector de peso  $\mathbf{w}_j$ . Por lo tanto, a través del proceso de aprendizaje  $m_k$  deberá haber establecido los vectores de peso  $\mathbf{w}_i$  cerca del vector promedio —centroide— del cúmulo de vectores que representa la clase asociada al vecino  $i$ , es decir la clase de recursos fácilmente accesibles a través del vecino  $i$ .

Otra forma posible de calcular la activación de las neuronas de la capa de salida de la red neuronal es por medio del producto escalar de los vectores  $\mathbf{v}$  y  $\mathbf{w}_i$  previamente normalizados. Este cálculo también representa una medida de distancia entre los vectores  $\mathbf{v}$  y  $\mathbf{w}_i$  siendo mayor cuanto más cercanos son los vectores entre sí —ver la sección 2.6.2 “La Neurona Competitiva” en la página 81—. Sin embargo ha mostrado rendimientos inferiores con relación a la activación definida en la ecuación (4-3) en la experimentación realizada en este trabajo y ha sido descartada para la implementación de la búsqueda inteligente.

#### 4.1.4.2 Propagación de la Solicitud de Búsqueda

Una vez recibido un mensaje *Query* desde algún vecino  $j$ ,  $nodo_k$  verifica si posee el recurso buscado, en caso afirmativo contesta con un *QueryHit* al vecino  $j$  desde el cual recibió el mensaje *Query*. Luego, al igual que en *Gnutella*, decrementa y constata si el valor de TTL es igual a cero. Si es este el caso se descarta la búsqueda. En caso contrario inicia las acciones necesarias para la propagación de la búsqueda hacia otros vecinos.

Una vez evaluada  $m_k$  se establece un orden de preferencia a partir del vector de salida  $\mathbf{o}$  para conducir la búsqueda de  $r$  hacia un subconjunto de vecinos. Si el vecino  $j$  desde el que se recibió la solicitud de búsqueda forma parte de este subconjunto elegido es excluido evitando un mensaje innecesario.

BII-P2P no hace suposición alguna respecto del tamaño de vecindad de los nodos de la red P2P y sólo asume que en todos los casos la misma se compone de al menos un nodo vecino. Por lo tanto, la cantidad de nodos a los cuales propagar

la búsqueda se determina en función del tamaño de la vecindad en cada caso. La tasa de selección que se denota con el escalar  $\gamma \in (0,1]$  representa la proporción de vecinos elegidos para propagar la solicitud y constituye un parámetro importante de la búsqueda BII-P2P.

Sea  $NV_k = \{nodo_{k1}, nodo_{k2}, \dots, nodo_{kn}\}$  el conjunto de vecinos de  $nodo_k$ , la cantidad de vecinos a los cuales  $nodo_k$  propaga la búsqueda se determina por  $\lceil \gamma |NV_k| \rceil = \lceil \gamma n \rceil$ , es decir, la función techo ( $\lceil x \rceil = \min\{k \in \mathbb{Z} | x \leq k\}$ ), aplicada a la multiplicación de  $\gamma$  por la cantidad de vecinos de  $nodo_k$ . Por lo tanto se asegura que en todos los casos la cardinalidad del subconjunto de vecinos elegidos será mayor o igual a uno. Claramente, si  $nodo_k$  posee un único vecino desde el cual recibió la solicitud de búsqueda, una vez realizada la verificación acerca de la posesión del recurso buscado,  $nodo_k$  descarta la búsqueda.

El parámetro  $\gamma$  influye directamente en el rendimiento del sistema de búsqueda. Si  $\gamma = 1$  el algoritmo de búsqueda se comporta como el BFS de *Gnutella* pues es elegido el conjunto entero de vecinos para propagar la búsqueda. Haciendo  $\gamma < 1$  se generan menos mensajes *Query* y se ahorra tráfico sobre la red. Si las decisiones locales llevadas a cabo por los nodos son acertadas, se logrará un buen promedio de hallazgos con menor cantidad de mensajes.

En la Figura 4-3 puede verse un ejemplo en el que  $nodo_k$  recibe y a su vez propaga una solicitud de búsqueda. Dicha solicitud es recibida desde el nodo vecino 5 (Figura 4-3 I). En el mensaje *Query* recibido se especifica el vector característico  $\mathbf{v}$  del recurso buscado. La evaluación de  $m_k$  para el vector  $\mathbf{v}$  produce la valoración  $\mathbf{o}$  que se muestra en la Figura 4-3 II. Se establece el ordenamiento de preferencia de la siguiente manera: [4; 2; 5; 1; 6; 3]. Si  $\gamma = 0,4$  se determina la cantidad de vecinos a los que propagar la solicitud como  $\lceil 0,4 \times 6 \rceil = 3$ . Se preseleccionan los vecinos {4; 2; 5} y se excluye al vecino 5 por ser desde quién se recibió la solicitud. Así finalmente el subconjunto de vecinos elegidos para propagar el mensaje *Query* recibido queda determinado por {4; 2}. (Figura 4-3 III)

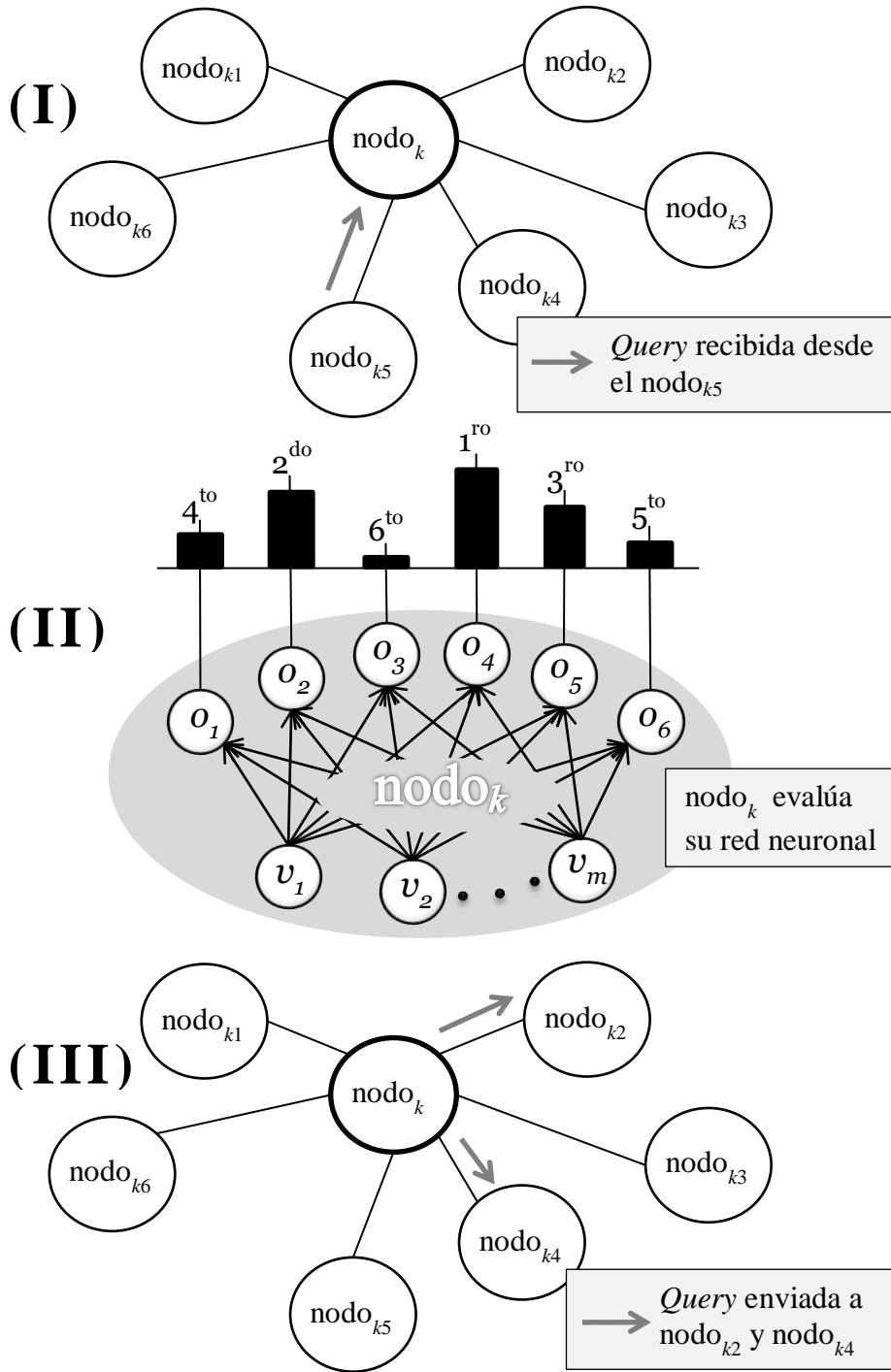


Figura 4-3 Participación de la red neuronal del nodo en la toma de decisión para encaminar un mensaje *Query* recibido

La selección de un subconjunto de vecinos adecuados permite la utilización de valores de TTL más grandes que en el algoritmo BFS de *Gnutella* posibilitando alcanzar nodos más lejanos poseedores del recurso, pero manteniendo el tráfico generado en niveles aceptables. La búsqueda se hace entonces más profunda pero menos ancha. La elección del valor de TTL con el que los nodos iniciarán las solicitudes y el parámetro  $\gamma$  —profundidad y ancho de la búsqueda respectivamente— resultan significativos en el rendimiento del sistema. En la sección 5.1.3 “Resultados de la Experimentación” en la página 142 se presenta una gran cantidad de casos estudiados con distintas asignaciones de valores para estos parámetros.

Al igual que en *Gnutella* los mensajes *QueryHit* se propagan siguiendo el camino inverso de los mensajes *Query*. Esto permite a los nodos que propagaron la solicitud conocer si la decisión realizada para conducir la búsqueda fue exitosa o no lo ha sido. El enfoque híbrido de aprendizaje implementado —ver la sección 4.2.4 “Enfoque Híbrido” En la página 127— toma ventaja de esta situación.

## 4.2 El Aprendizaje

Para comprender cómo a partir de una solicitud de búsqueda específica, trabajando de la manera descrita en los párrafos anteriores, las redes neuronales pueden inferir correctamente las mejores alternativas para encaminar los mensajes *Query*, es necesario conocer cómo se lleva a cabo el proceso de aprendizaje y el ajuste de los vectores de pesos de la capa de salida.

Tradicionalmente el aprendizaje de una red neuronal competitiva se realiza por medio de un proceso de entrenamiento. Éste consiste en una fase iterativa de sucesivos ajustes de pesos neuronales hasta que se consigue la correcta clasificación del conjunto de vectores de entrenamiento. Para que el aprendizaje sea efectivo, es necesario que este conjunto constituya una muestra representativa del universo total de vectores que se desean clasificar —en el Capítulo 2 se han

descripto varias formas de entrenamiento, en particular en la sección 2.6.3 se expone el entrenamiento de la red competitiva SOM—.

Una vez concluido el entrenamiento, cada neurona de la capa competitiva queda asociada a una clase de vectores. Esta asociación surge por la proximidad del vector de pesos de la neurona al vector promedio —centroide— de la clase correspondiente a la que se encuentra asociada. Este hecho, bajo ciertas circunstancias, puede aprovecharse para acelerar el proceso de entrenamiento. En el caso de conocerse todos los vectores que conforman una clase, simplemente se calcula el centroide de la misma y con él se establece el vector de pesos de la neurona asociada a la clase evitando el largo proceso iterativo descrito precedentemente.

Considérense nuevamente al  $k$ -ésimo nodo del sistema P2P  $nodo_k$  y a su respectiva red neuronal  $rn_k$ . Desde el punto de vista de  $nodo_k$ , el conocimiento de los vectores que conforman cada una de las clases reconocibles por  $rn_k$  es relativamente fácil de acceder. Tal conocimiento está presente en cada uno de sus vecinos directos. Para poder aprovechar esta situación se ha extendido el protocolo *Gnutella* con un nuevo tipo de mensaje llamado *NeuralInf* para que los nodos de la red puedan intercambiar esta información con sus nodos vecinos. Así es posible la adecuación rápida de los pesos y el acortamiento del proceso de aprendizaje de cada una de las redes neuronales que funcionan en cada uno de los nodos del sistema P2P.

#### 4.2.1 Extensión del Protocolo *Gnutella*

Como se ha visto anteriormente —ver sección 3.1 “Con Base en *Gnutella*” en la página 88— el protocolo *Gnutella* 0.4 utiliza un conjunto de 5 mensajes de los cuáles *Query* y *QueryHit* están involucrados directamente en el mecanismo de búsqueda que esta red implementa. Ya se ha mencionado también que BII-P2P está basado en *Gnutella* 0.4 que se destaca por ser simple, fácil de implementar y

fácil de administrar prescindiendo de la necesidad de cualquier tipo de gestión global. BII-P2P utiliza todos los mensajes de *Gnutella*.

Sin embargo para soportar adecuadamente la búsqueda inteligente es necesario extender el protocolo de *Gnutella* con un nuevo mensaje: el mensaje *NeuralInf*. A diferencia de la red *Gnutella*, la búsqueda inteligente aquí presentada necesita que los nodos de la red P2P adquieran información sobre las clases de recursos accesibles a través de sus vecinos. Esta información queda almacenada en los pesos de las redes neuronales en cada uno de los nodos de la red P2P.

Si bien los pesos de las neuronas pueden establecerse por medio de un entrenamiento tradicional como el que se utiliza con las redes LVQ —ver más adelante sección 4.2.4.1 “Learning Vector Quantization (LVQ)” en la página 128— BII-P2P implementa como estrategia principal de aprendizaje una nueva forma basada en la comunicación de información entre nodos vecinos. Esta comunicación facilita y hace aceptablemente rápido el proceso de adquisición del conocimiento requerido por BII-P2P.

En resumen, se han definido dos mecanismos distintos de aprendizaje para las redes neuronales en BII-P2P. El primero y principal utiliza el nuevo tipo de mensaje denominado *NeuralInf* para comunicar información de actualización permitiendo una rápida adecuación de los pesos neuronales en el nodo que la recibe. El segundo es opcional y está basado en un entrenamiento tradicional como el que se utiliza con las redes LVQ que se lleva a cabo por medio del monitoreo constante de los mensajes *Query* y *QueryHit* despachados por el nodo. Este mecanismo secundario de aprendizaje no genera tráfico sobre la red sin embargo, al converger lentamente, sólo se utiliza para realizar ajustes pequeños entre dos fases de aprendizaje vía difusión de mensajes.

La utilización conjunta de ambos mecanismos de aprendizaje puede reducir la sobrecarga en el tráfico permitiendo espaciar en el tiempo las difusiones de

mensajes *NeurlaInf*. Sin embargo, aunque se observan indicios sobre esta conveniencia, aún los resultados no son concluyentes al respecto —ver secciones 5.2.3 “Resultados de la Experimentación” en la página 178 y 5.2.4 “Conclusiones sobre la Capacidad de Adaptación” en la página 189—.

#### 4.2.2 Información de Actualización. Mensaje *NeuralInf*

En el mensaje *NeuralInf* los nodos comunican información a todos sus vecinos para que estos últimos puedan actualizar los pesos de sus redes neuronales. Sean  $nodo_i$  y  $nodo_j$  dos nodos vecinos de la red P2P. Cuando  $nodo_i$  recibe de  $nodo_j$  un mensaje *NeuralInf* actualiza el peso de la neurona de la capa de salida correspondiente a  $nodo_j$  con la información contenida en este mensaje. Esta información no es más que el vector centroide de la clase de recursos que  $nodo_i$  puede hallar con facilidad a través de  $nodo_j$ , ya sea directamente porque  $nodo_j$  posee el recurso buscado o indirectamente porque puede ser alcanzado por medio de los otros vecinos de  $nodo_j$  distintos de  $nodo_i$ .

Sea  $nodo_k$  un nodo cualquiera de la red P2P, con un conjunto de  $n$  vecinos  $NV_k = \{nodo_{k1}, nodo_{k2}, \dots, nodo_{kn}\}$  con los que se conecta directamente. Sea  $V$  el conjunto de vectores característicos de los recursos compartidos por  $nodo_k$ ,  $V = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p\}$ . Sea  $\mathbf{c}$  el vector centroide del conjunto  $V$ , es decir el promedio de todos los vectores que caracterizan los recursos compartidos por  $nodo_k$ .

$$\mathbf{c} = \frac{1}{p} \sum_{i=1}^p \mathbf{v}_i \quad (4-4)$$

Esta información resulta particularmente valiosa para los vecinos de  $nodo_k$  puesto que con ella pueden identificar cuál es la clase de recursos que  $nodo_k$  posee



para compartir. Obsérvese que  $nodo_k$  representa para cualquiera de sus vecinos una posible alternativa para encaminar una búsqueda que debe ser evaluada en comparación con las otras opciones ofrecidas por sus otros nodos vecinos.

En principio  $nodo_k$ , podría comunicar el vector centroide  $\mathbf{c}$  a todos sus vecinos en un mensaje *NeuralInf* para que establezcan el peso de la neurona asociada a  $nodo_k$  en sus respectivas redes neuronales. Sin embargo, si los vecinos utilizaran este vector  $\mathbf{c}$  como representante de la clase de recursos accesibles a través del  $nodo_k$ , las redes neuronales no podrían conducir acertadamente los mensajes *Query*, salvo que el recurso buscado se hallase en un vecino conectado al nodo en forma directa. Es necesario entonces transmitir información de nodos más alejados en los mensajes *NeuralInf*.

Por lo tanto la información que  $nodo_k$  suministra en un mensaje *NeuralInf* a su  $i$ -ésimo vecino  $nodo_{ki}$  debe también integrar la información que  $nodo_k$  posee sobre sus otros vecinos distintos de  $nodo_{ki}$ . Entonces un nodo calcula el vector centroide de actualización en forma individualizada para cada uno de sus vecinos. Para ello utiliza la información previa recibida desde todos los otros vecinos en mensajes *NeuralInf*.

Sea  $\sigma$  un parámetro escalar positivo tal que  $0 \leq \sigma < 1$  y que representa la relevancia asignada a la información proveniente de los vecinos. Sea  $\mathbf{c}$  el centroide de  $nodo_k$  calculado como se ha definido en la ecuación (4-4) y sean  $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n$  los vectores centroides de actualización recibidos en previos mensajes *NeuralInf* desde cada uno de los  $n$  vecinos del nodo, es decir  $\mathbf{c}_i$  es el vector centroide recibido por  $nodo_k$  desde  $nodo_{ki}$ . El cálculo del vector centroide de actualización  $\mathbf{c}^i$  que se enviará en un mensaje *NeuralInf* desde  $nodo_k$  a  $nodo_{ki}$  se calcula de la siguiente manera:

$$\mathbf{c}^i = (1 - \sigma)\mathbf{c} + \sigma \frac{1}{n-1} \sum_{\substack{j=1 \\ j \neq i}}^n \mathbf{c}_j \quad (4-5)$$

De esta forma un mensaje *NeuralInf* enviado por un nodo de la red P2P integra información sobre todos los recursos compartidos por él mismo más todos los recursos compartidos por otros nodos alcanzables a partir de él.

Así, con cada envío de un mensaje *NeuralInf*, los rastros de la información generada por numerosos nodos son impulsados un paso más hacia regiones cada vez más lejanas. Sin embargo, la intensidad de estos rastros disminuye con la distancia, conforme los mensajes atraviesan la red.

Aunque un mensaje *NeuralInf* integra mucha información, su tamaño se mantiene en un rango aceptable. En todos los casos, el vector  $\mathbf{c}^i$  calculado en la ecuación (4-5), tiene la misma cantidad de elementos que los vectores característicos de los recursos compartidos en el sistema. Por lo tanto el tamaño de los mensajes *NeuralInf* no supera al tamaño de los mensajes *Query* utilizados en las solicitudes de búsqueda.

Aunque la propagación de mensajes *NeuralInf* provoca tráfico sobre la red P2P los beneficios que conlleva son muy grandes, permitiendo la rápida adquisición de conocimiento requerida por BII-P2P. No obstante, para maximizar el rendimiento de la estrategia propuesta es necesario mantener lo más bajo posible el tráfico ocasionado por el envío de mensajes *NeuralInf*.

Cuando un nodo evalúa que es necesario enviar información de actualización comienza una difusión de mensajes *NeuralInf* que se lleva a cabo de forma similar a la propagación de mensajes *Query* en la red *Gnutella*. Los vectores  $\mathbf{c}^i$ ,

calculados para cada uno de sus vecinos según la ecuación (4-5), son empaquetados y enviados a su vecindad en mensajes *NeuralInf*. A su vez, cada vecino que recibe el mensaje, actualiza su red neuronal y dependiendo de la política de actualización adoptada en el sistema, puede repetir el mismo proceso propagando información a sus otros vecinos. Se utiliza un campo TTL en la cabecera de los mensajes *NeuralInf* para mantener el tráfico generado bajo control. Además, los mensajes son individualizados por un campo ID que establece el nodo que inicia la difusión. Los nodos que reciben varios mensajes *NeuralInf* provenientes de distintos vecinos con el mismo ID, sólo lo propagan, en caso de ser necesario, una sola vez con el arribo del primer mensaje. Sin embargo en todos los casos se actualizan los pesos de la neurona correspondiente al vecino desde dónde se recibió la información.

Con el objetivo de mantener reducido el tráfico generado por los mensajes *NeuralInf* es posible adoptar varias políticas que gobiernen la propagación de la información de actualización, tanto en lo referido al alcance como a la frecuencia de la difusión. En la sección siguiente se discuten algunas de estas políticas.

### 4.2.3 Políticas de Adquisición y Actualización de Conocimiento

Se han evaluado distintas estrategias para propagar la información de adquisición y actualización. El objetivo es generar el menor tráfico posible manteniendo aceptablemente actualizados los pesos de las redes neuronales en cada nodo.

#### 4.2.3.1 Política de Difusión Simple

La estrategia más simple, aunque ineficiente, consiste en que cada nodo, ante la evidencia de una variación en el ambiente —la baja de un nodo vecino, un cambio interno en el conjunto de recursos compartidos, etc.— reaccione inmediatamente actualizando su red neuronal e iniciando la difusión de mensajes *NeuralInf* hacia todos sus nodos vecinos. Por su parte, los nodos que reciban un

mensaje *NeuralInf*, actualizan sus redes neuronales y repiten el proceso difundiendo mensajes de actualización a todos sus otros vecinos. De esta forma la información del cambio alcanzará todos los nodos de la red P2P manteniendo sus redes neuronales actualizadas. Este enfoque no usa TTL.

Claramente esta estrategia no es escalable, los requerimientos de ancho de banda son excesivos. Aunque el conocimiento de todos los nodos de la red P2P se mantiene completamente actualizado, el esfuerzo por conservar el estado de BII-P2P en óptimas condiciones adolece de las mismas limitaciones que el algoritmo de búsqueda de la red *Gnutella* que se pretende mejorar. Por lo tanto el costo es inaceptable a menos que la frecuencia de propagación de información de actualización por parte de los nodos sea lo suficientemente baja. En La sección 5.2.3 —página 178— se presentan los resultados experimentales obtenidos y se analiza en detalle esta situación. Sin embargo, independientemente de su factibilidad, los resultados de la política Simple interesan a los efectos comparativos.

#### **4.2.3.2 Difusión limitada en profundidad (DLP).**

Resulta obligado limitar la propagación de los mensajes *NeuralInf* para evitar que inunden la red P2P afectando seriamente el rendimiento general de la misma. Alcanza con utilizar el campo TTL ya mencionado en los mensajes *NeuralInf* para poder implementar la difusión de mensajes de actualización con tiempo de vida. Este enfoque ha mostrado muy buenos resultados. Ello se debe a que la significación de los datos comunicados por un nodo en los mensajes *NeuralInf* va disminuyendo conforme se aleja del emisor al atravesar la red —ver ecuación (4-5)— y por lo tanto resulta innecesario propagarlos hasta nodos muy distantes. Se utilizará la notación  $DLP_i$  para hacer referencia a la política DLP que utiliza el parámetro TTL igual a  $i$ .

### 4.2.3.3 Difusión Limitada en Profundidad y Anchura (DLPA)

Además de utilizar el valor de TTL para controlar el tiempo de vida de los mensajes *NeuralInf*, en este enfoque los nodos sólo propagan la información de actualización a un subconjunto de vecinos utilizando el criterio de selección conocido con el nombre de “el menos recientemente usado” —LRU, por sus siglas en inglés—. El rendimiento de esta estrategia depende fuertemente del grado de acoplamiento de los nodos en la red. Se utilizará la notación  $DLPA_{(i, j)}$  para hacer referencia a la política DLPA que utiliza parámetros TTL igual a  $i$  y porcentaje de selección de vecinos igual a  $j$ .

### 4.2.3.4 Alcance Real de la Información de Actualización en DLP y DLPA

Debe quedar claro que, aunque la política elegida para la adquisición y actualización de conocimiento sea  $DLP_i$  o  $DLPA_{(i, j)}$ , el radio de influencia de una difusión no siempre está acotado por la distancia establecida por el parámetro  $i$  que representa el campo TTL de los mensajes *NeuralInf*.

La situación descrita en el párrafo anterior se debe a que, según la forma en que se suceden las difusiones iniciadas por distintos nodos, la información de actualización transmitida en los mensajes *NeuralInf* suele ser impulsada más allá del límite impuesto por el valor de  $i$ .

La Figura 4-4 muestra el modo en que fluye la información de actualización difundida por los nodos más allá del límite establecido por la política  $DLP_2$  utilizada en el ejemplo. En el momento (I) el nodo A envía un mensaje *NeuralInf* a su vecino B con el campo  $TTL=2$ . A su vez, el nodo B, luego de actualizar su red neuronal con la información recibida, envía un mensaje *NeuralInf* con  $TTL=1$  al nodo C. Así la información comunicada por A llega, de manera atenuada —ver ecuación (4-5)—, hasta el nodo C. El nodo C, que está a distancia 2 de A, actualiza su red neuronal como única acción terminando así la difusión iniciada por A. Sin embargo luego, en el momento (II) una difusión de actualización

iniciada por el nodo D induce al nodo C a propagar un mensaje *NeuralInf* haciendo que el nodo E, que está a distancia 3 del nodo A, reciba también parte de la información comunicada por A en la primera difusión. Si una vez alcanzada esta situación, el nodo C inicia otra difusión provocaría que el nodo E impulsara aún más lejos, aunque de forma cada vez más atenuada, la información proveniente desde el nodo A.

Por lo tanto, debe destacarse entonces que el límite impuesto por el parámetro  $i$  en las políticas DLP y DLPA establece una cota al tráfico generado por cada difusión de actualización pero no representa un límite infranqueable para la información transmitida en los mensajes *NeuralInf*.

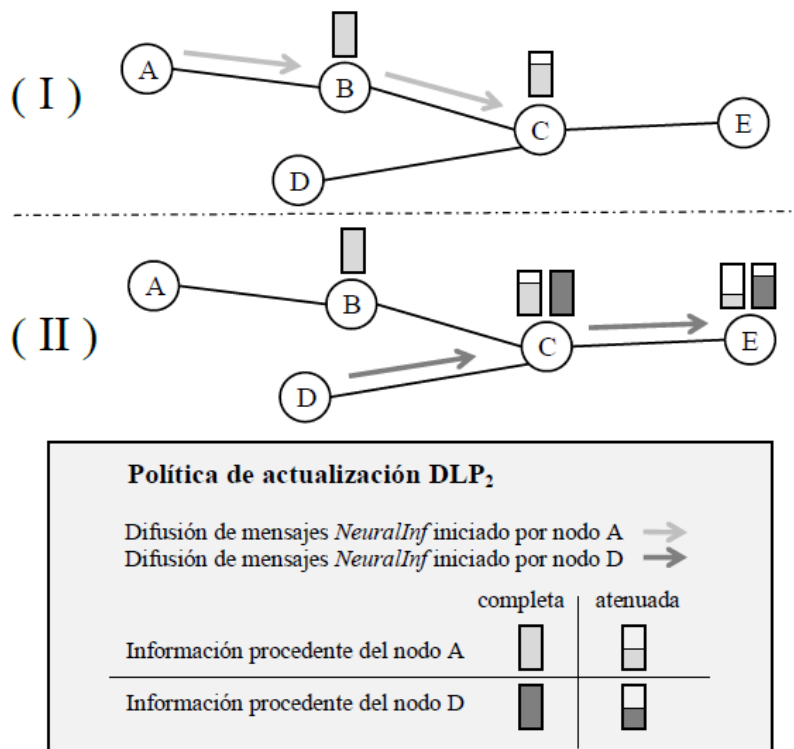


Figura 4-4 Difusión de mensajes *NeuralInf* utilizando política  $DLP_2$ . Primero el nodo A —en (I)— y luego el nodo D —en (II)— difunden información de actualización. Se observa en (II) cómo la información proveniente del nodo A es impulsada hasta el nodo E.

#### 4.2.3.5 Determinación de la Frecuencia de Actualización

Con independencia de la política adoptada para la difusión de mensajes *NeuralInf* debe regularse la forma en que se producen los desencadenantes de tales difusiones. Para ello, se utiliza el parámetro umbral  $u$  con el que se ajusta el grado de sensibilidad al cambio del sistema. Cuando un nodo experimenta un cambio de estado determina si debe propagar información de actualización o si, por el contrario, los efectos del cambio no merecen la difusión de mensajes *NeuralInf* a sus vecinos directos.

De esta forma  $nodo_k$  sólo propagará un mensaje *NeuralInf* a su  $i$ -ésimo vecino  $nodo_{ki}$  si el cambio detectado por  $nodo_k$  —producido por la caída de algún nodo vecino, el arribo de un mensaje *NeuralInf* o simplemente por un cambio interno en el conjunto de recursos compartidos por él— provoca que el nuevo vector centroide de actualización  $\mathbf{c}^i$  diste del último anterior enviado a  $nodo_{ki}$  en un valor mayor al umbral  $u$ . Utilizando distintos valores para  $u$  se puede ajustar la sensibilidad para que pequeños cambios con bajo impacto en el rendimiento de la búsqueda inteligente no generen tráfico innecesario en la red P2P.

Sin embargo, determinar el valor óptimo para el umbral  $u$  no es trivial y depende fuertemente de la topología de la red, y del conjunto de recursos compartidos por los vecinos cercanos.

#### 4.2.4 Enfoque Híbrido

Este enfoque propone espaciar en el tiempo la difusión de mensajes *NeuralInf* para disminuir los requerimientos de ancho de banda de BII-P2P. Compensa el desajuste provocado por los cambios un adecuado algoritmo de aprendizaje en línea que no produce tráfico alguno sobre la red P2P. La adquisición de conocimiento necesaria se consigue a partir del monitoreo de mensajes *Query* enviados y respuestas *QueryHit* recibidas que efectúan los nodos conforme participan en decisiones de encaminamiento de solicitudes de búsquedas en la red.

Para implementar este aprendizaje en línea se ha utilizado una adaptación de un tipo de entrenamiento supervisado para redes neuronales competitivas llamado *Learning Vector Quantization* (LVQ). Esto es factible porque el protocolo de propagación de las solicitudes de búsqueda establece el envío de los mensajes *QueryHit* siguiendo la ruta inversa de los mensajes *Query*. Por lo tanto, todos los nodos que participan en el encaminamiento de una búsqueda conocen por cuáles de los caminos de propagación elegidos se ha tenido éxito y por cuáles la búsqueda ha resultado infructuosa. Este conocimiento permite aplicar un entrenamiento de tipo supervisado como el que se describe a continuación.

#### 4.2.4.1 Learning Vector Quantization (LVQ)

LVQ describe una clase de algoritmos de aprendizaje supervisado aplicado a redes neuronales competitivas. Este tipo de algoritmos necesita conocer si cada una de las respuestas de la red neuronal ha resultado acertada o fallida. En particular en este trabajo se ha utilizado una adaptación de LVQ1 que se describe brevemente en esta sección. El lector interesado puede encontrar una completa descripción de la familia de algoritmos LVQ en [79].

Como es habitual, el entrenamiento LVQ1 consiste en una fase iterativa de estimulación, evaluación de la red y adecuación de pesos neuronales hasta la correcta clasificación del conjunto de vectores de entrenamiento. En cada ajuste de pesos se aplica la regla de aprendizaje que se describe a continuación.

Sea  $\mathbf{v}$  el vector de entrada con el que se estimula a la red competitiva en un instante de tiempo  $t$ , sea  $n_i$  la neurona que gana la competición —valor de activación más alto— ante el estímulo  $\mathbf{v}$  y sea  $\mathbf{w}_i$  su respectivo vector de pesos. LVQ1 sólo adapta el vector de pesos de la neurona ganadora utilizando la siguiente ecuación:



$$\mathbf{w}_i(t+1) = \begin{cases} \mathbf{w}_i(t) + \alpha(t)[\mathbf{v} - \mathbf{w}_i(t)], & \text{si } \mathbf{v} \text{ y } \mathbf{w}_i \text{ pertenecen} \\ & \text{a la misma clase} \\ \mathbf{w}_i(t) - \alpha(t)[\mathbf{v} - \mathbf{w}_i(t)], & \text{si } \mathbf{v} \text{ y } \mathbf{w}_i \text{ pertenecen} \\ & \text{a distintas clases} \end{cases} \quad (4-6)$$

Aquí  $0 < \alpha(t) < 1$  y  $\alpha(t)$  representa la velocidad de aprendizaje. Se recomienda comenzar con un valor pequeño para  $\alpha(t)$  que suele hacerse decrecer linealmente hasta cero, a medida que la red va convergiendo hacia la clasificación correcta.

Obsérvese que el entrenamiento consiste en reforzar los aciertos de la red neuronal, acercando el vector  $\mathbf{w}_i$  al vector de entrada  $\mathbf{v}$  siempre que la respuesta haya sido correcta. De esta forma, en lo sucesivo la neurona  $n_i$  se activará con más fuerza ante la presencia del estímulo  $\mathbf{v}$ .

Por el contrario, si la clasificación realizada por la red neuronal resulta equivocada,  $\mathbf{w}_i$  es alejado del vector  $\mathbf{v}$  debilitando así la activación de  $n_i$ . De esta manera se tiende a que en el futuro sea otra neurona la que gane la competición frente al estímulo  $\mathbf{v}$ .

Debe recordarse que LVQ1 implementa un tipo de aprendizaje supervisado, por lo que sólo puede ser aplicado si se conocen de antemano a qué clases pertenecen los vectores de entrenamiento utilizados. Como ocurre con otros tipos de arquitecturas neuronales su utilidad no se debe a la correcta clasificación de los vectores de entrenamiento sino a la facultad de adquirir una buena capacidad de generalización. Esta capacidad le permitirá a la red clasificar correctamente cualquier otro vector del universo total existente aunque nunca antes haya sido estimulada con él.

#### 4.2.4.2 Implementación de Aprendizaje en Línea Basado en LVQ1

Ya se ha mencionado que el mecanismo de propagación de los mensajes *Query* y las respuestas *QueryHit* introducido por *Gnutella* y adoptado por BII-P2P, permite que todos los nodos de la red P2P involucrados en una búsqueda conozcan si la propagación de la misma ha sido exitosa o improductiva para cada uno de los vecinos elegidos —ver Figura 3-1 en la página 92—. La disponibilidad de esta información permite aplicar de forma continua un entrenamiento especial basado en LVQ1.

LVQ1 no puede aplicarse directamente sobre las redes neuronales utilizadas en BII-P2P. Ello se debe a que estas últimas no realizan la tarea de un sistema típico de clasificación sino que se diferencian fundamentalmente en:

- i) Las redes neuronales utilizadas en BII-P2P no clasifican a los recursos como miembros de una determinada clase sino que producen una valoración sobre la pertenencia a varias clases simultáneamente. Por lo tanto no existe el concepto de única neurona ganadora de la competición con el que trabaja LVQ1.
- ii) En BII-P2P tampoco se tiene una única caracterización de éxito para rotular el resultado de una decisión de encaminamiento. Mientras que claramente el fracaso está asociado a la ausencia de respuestas, existen diversos grados de éxito determinado por la cantidad de mensajes *QueryHit* devueltos por los nodos hacia los que se dirigió la búsqueda.

En atención a las diferencias expuestas precedentemente, el aprendizaje en línea se ha implementado de la siguiente manera:

Dada una solicitud de búsqueda y su consecuente propagación, una vez vencido el tiempo de espera a partir de la difusión de los mensajes *Query* se procede a la actualización de los pesos de todas las neuronas involucradas, es decir de aquellas asociadas a los nodos vecinos por donde la búsqueda fue

propagada. El tiempo de espera para la recepción de los mensajes *QueryHit* es un parámetro de BII-P2P que puede ajustarse localmente en cada nodo de la red en función de su experiencia previa acumulada.

Sea  $nodo_k$  el nodo que realiza la actualización de pesos luego de vencido el tiempo de espera ya mencionado. Sea  $\mathbf{w}_i$  el vector de pesos de la neurona  $n_i$  de la capa competitiva, que se corresponde con su vecino  $nodo_{ki}$ . Sea  $h_i$  el número de mensajes *QueryHit* devueltos para la búsqueda desde el vecino  $nodo_{ki}$ . El vector  $\mathbf{w}_i$  se actualiza de la siguiente manera:

$$\mathbf{w}_i(t+1) = \begin{cases} \mathbf{w}_i(t) + h_i \alpha [\mathbf{v} - \mathbf{w}_i(t)], & \text{si } h_i \geq 1 \\ \mathbf{w}_i(t) - \alpha [\mathbf{v} - \mathbf{w}_i(t)], & \text{si } h_i = 0 \end{cases} \quad (4-7)$$

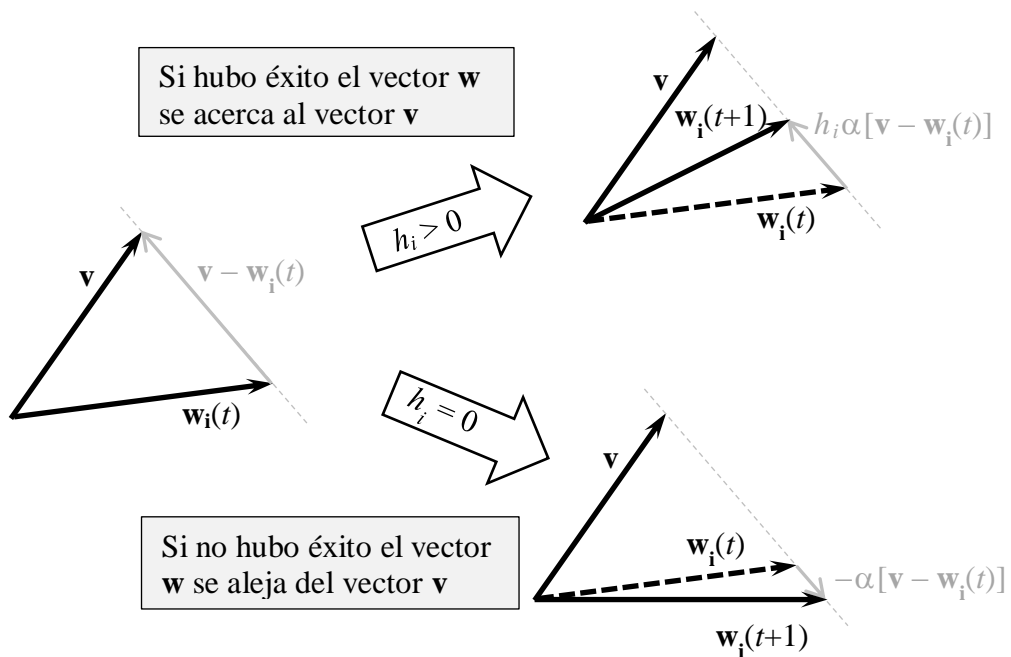


Figura 4-5 Entrenamiento LVQ. El vector de pesos de la  $i$ -ésima neurona se acerca o aleja del vector de entrada según se reciban o no mensajes *QueryHit* desde el  $i$ -ésimo vecino.

La ecuación (4-7) establece que en caso de acierto, la adecuación del peso será proporcional al número de respuestas *QueryHit* devueltos por el nodo vecino *nodo<sub>ki</sub>*. Obsérvese que se ha utilizado una velocidad de aprendizaje constante — $\alpha$  no depende del tiempo—. Ello se debe a que, asumiendo una tasa alta de cambios en la red P2P, no se espera que el entrenamiento en línea vaya convergiendo a una situación estable. Sólo se utiliza para atenuar el impacto negativo de los cambios de estado producidos en los nodos y así poder retrasar la propagación de mensajes *NeuralInf* con el correspondiente ahorro de ancho de banda.

#### **4.2.4.3 Difusión de Información de Actualización con Entrenamiento en Línea**

El entrenamiento de las redes neuronales por medio del algoritmo LVQ1 expuesto precedentemente converge de manera lenta. La naturaleza generalmente cambiante de los sistemas P2P hace aconsejable la propagación de mensajes *NeuralInf* para actualizar rápidamente los pesos de las neuronas en los nodos de la red.

Sin embargo, aunque el rendimiento del entrenamiento neuronal es inferior en cuanto a la velocidad de adecuación, aventaja a la difusión de información de actualización en otro aspecto, al no generar tráfico alguno sobre la red. Por lo tanto, utilizar ambos en forma complementaria permite reducir el tráfico manteniendo niveles aceptables de rendimiento.

Así es posible espaciar la propagación de mensajes *NeuralInf* realizando entrenamiento neuronal sostenido en todas las intervenciones de búsqueda en la que participen los nodos.

### **4.3 Protocolo de Ingreso a la Red P2P**

Al protocolo basado en mensajes *Ping* y *Pong* de *Gnutella* para el descubrimiento de nodos y conexión con los mismos, se ha añadido un protocolo

especial de ingreso al sistema P2P que tiene como objetivo el ajuste de las redes neuronales en la vecindad del nuevo nodo. Este protocolo también instrumenta la adquisición por parte del nuevo integrante de la información necesaria para participar eficientemente del sistema de búsqueda BII-P2P.

Cuando un nodo  $nodo_i$  desea ser aceptado como vecino de otro nodo  $nodo_j$ , le envía un mensaje de petición de aceptación. Si  $nodo_j$  acepta, envía a  $nodo_i$  un mensaje *NeuralInf* especial —TTL=1 para que no sea propagado—, con el vector centroide de actualización calculado para  $nodo_i$  —ver ecuación (4-5)—. Una vez que  $nodo_i$  haya sido aceptado por un número mínimo requerido de vecinos —dependiente de las necesidades del nodo—,  $nodo_i$  lanza una difusión ordinaria de mensajes *NeuralInf* quedando concluido el protocolo de ingreso requerido.

En principio BII-P2P no hace ninguna suposición sobre el tamaño de la vecindad de los nodos. Tal discusión es ajena a la incumbencia del sistema de búsqueda y se reserva como una decisión propia de la implementación particular de la red P2P que se quiera realizar. Sin embargo para que un nodo pueda propagar una búsqueda que no haya iniciado el mismo, debe estar conectado con al menos dos vecinos distintos y por tal razón este es el número mínimo aconsejado.

## 4.4 Resumen del Capítulo

La búsqueda inteligente BII-P2P basa su funcionamiento en la habilidad de los nodos para dirigir los mensajes *Query* sólo hacia las regiones más prometedoras de la red P2P. Así llega a nodos más alejados que el algoritmo BFS de *Gnutella*, sin aumento de tráfico, compensado por la exclusión de los caminos desfavorables. Encontrar un camino conveniente para conducir una búsqueda concreta depende de una sucesión de “buenas decisiones” tomadas por todos los nodos involucrados en la trayectoria del mensaje de solicitud —aquel que originó el mensaje y todos los nodos intermedios—. Para ello, cada nodo utiliza su propia

red neuronal para determinar las clases de recursos que pueden hallarse más rápidamente a partir de cada uno de sus vecinos.

La red neuronal en cada nodo trabaja clasificando recursos, por medio de su representación vectorial, en tantas clases como vecinos posea el nodo. Si un recurso  $r$  es clasificado por la red neuronal como perteneciente a la clase  $j$ , esto indica que el vecino  $j$  posee al recurso  $r$  o a partir de él se encuentran los caminos con mayor cantidad de instancias de  $r$  y con menor número de nodos intermedios —menos tráfico y respuestas positivas más rápidas—. Sin embargo, en lugar de una clasificación categórica de  $r$ , la red neuronal establece una valoración sobre la pertenencia de  $r$  a cada una de las clases disponibles. Este resultado se utiliza para conducir la búsqueda hacia el subconjunto de vecinos más apropiado, aquellos con los grados de pertenencia más altos.

La red neuronal implementada en cada nodo es de tipo *feedforward* con dos capas totalmente conectadas. La capa de entrada se utiliza para ingresar y distribuir los datos —vector de entrada— a la capa de salida. Las neuronas de esta última capa calculan su activación por medio de una función inversamente proporcional a la distancia euclídea entre el vector de entrada y su propio vector de pesos. Así cuanto más cercano estén estos vectores entre sí mayor será la activación de la neurona. Por medio de un proceso de aprendizaje exitoso, cada neurona de la capa de salida establece su vector de pesos cerca del centroide del cúmulo de vectores que se corresponde con la clase a la que identifica. Por lo tanto, la neurona con mayor activación indica la clase a la que pertenece el vector de entrada. Además, BII-P2P utiliza este valor como una medida de pertenencia del vector de entrada a la clase asociada con la neurona.

Para propagar una búsqueda, el nodo estimula su red neuronal con la representación vectorial del recurso buscado, obtiene la valoración para cada vecino y encamina los mensajes *Query* hacia el subconjunto más favorable. Si en este subconjunto se encuentra el nodo desde el que se recibió la solicitud, es

eliminado previamente para evitar ciclos. La tasa de selección que se denota con el escalar  $\gamma \in (0,1]$  representa la proporción de vecinos elegidos para propagar la solicitud y constituye un parámetro importante de la búsqueda BII-P2P. El otro parámetro de interés lo constituye el valor de TTL que establece, al igual que en *Gnutella*, la distancia máxima que los mensajes *Query* pueden alcanzar antes de ser descartados. Utilizando un valor  $\gamma$  adecuado, BII-P2P puede generar menor cantidad de tráfico que el algoritmo BFS de *Gnutella* aún con un valor de TTL más alto que le permite mejorar el rendimiento de la búsqueda y encontrar mayor cantidad de instancias del recurso buscado.

Al igual que en *Gnutella* los mensajes *QueryHit* se propagan siguiendo el camino inverso de los mensajes *Query*. Por lo tanto, los nodos que propagan una solicitud conocen si la decisión realizada para conducir la búsqueda fue exitosa o no lo ha sido. Esta información se utiliza para implementar un mecanismo secundario de aprendizaje en línea basado en el entrenamiento LVQ para redes neuronales.

El entrenamiento principal por el que los nodos adquieren el conocimiento necesario se implementa por medio de la comunicación entre vecinos directos que difunden mensajes *NeuralInf*. Así es posible establecer los vectores de pesos de las neuronas de salida en las cercanías del centroide de la clase de recursos alcanzables a partir de los vecinos asociados a cada una de las neuronas de salida.

El mensaje *NeuralInf* que envía *nodo<sub>k</sub>* a su *i*-ésimo vecino *nodo<sub>ki</sub>* integra en un único vector  $\mathbf{c}^i$  el centroide de los recursos compartidos por *nodo<sub>k</sub>* más todos los recibidos en difusiones anteriores por sus otros vecinos —*nodo<sub>kj</sub>* para  $j \neq i$ , ver ecuación (4-5)—. De esta forma, el vector de pesos de la neurona de salida asociada *nodo<sub>k</sub>* en la red neuronal de *nodo<sub>kj</sub>* es establecido con el valor  $\mathbf{c}^i$  recibido en el mensaje *NeuralInf*.

Se han definido distintas políticas que gobiernan la propagación de la

información de actualización, tanto en lo referido al alcance como a la frecuencia de la difusión. Las políticas  $DLP_i$  —Difusión Limitada en Profundidad con parámetro  $TTL=i$ — limitan la propagación de los mensajes *NeuralInf* a los nodos que se encuentran a lo sumo a distancia  $i$  del nodo que comenzó la difusión de actualización. Las políticas DLPA —Difusión Limitada en Profundidad y Anchura con parámetro  $TTL=i$  y porcentaje de selección de vecinos =  $j$ — también imponen restricciones sobre la cantidad de vecinos a los cuales comunicar la información de actualización. Finalmente, la política Simple que establece la propagación de los mensajes *NeuralInf* hasta alcanzar toda la red P2P es impracticable por el costo asociado al tráfico generado, sin embargo se ha definido por razones teóricas permitiendo evaluar el rendimiento de BII-P2P bajo condiciones óptimas de actualización.



## Capítulo 5

# Evaluación de la Búsqueda Inteligente BII-P2P

Se ha evaluado el rendimiento de la búsqueda inteligente considerando dos aspectos muy importantes. El primero y fundamental es la capacidad de búsqueda propiamente dicha, es decir la habilidad para hallar los recursos buscados manteniendo el tráfico generado sobre la red en niveles aceptables o mínimos. El segundo aspecto analizado ha sido la capacidad de adaptación de BII-P2P a los cambios del entorno tan frecuentes en los sistemas P2P. Esta capacidad de adaptación está relacionada con la habilidad para mantener actualizado el conocimiento que los nodos tienen respecto de sus vecinos.

### 5.1 Evaluación de la Capacidad de Búsqueda

Para evaluar la capacidad de búsqueda de BII-P2P se ha recurrido a la simulación. El objetivo de la evaluación ha sido medir el rendimiento de la estrategia propuesta en un sistema P2P genérico. Por lo tanto no se ha realizado ninguna suposición sobre la naturaleza de los recursos compartidos en este sistema. Ello es factible debido a que BII-P2P trabaja sobre la representación vectorial de los recursos y no sobre los recursos mismos lo que permite, en esta etapa de realización de pruebas, un alto grado de independencia respecto de la naturaleza de los recursos y de la aplicación que corre sobre la red P2P.

### 5.1.1 La Experimentación

Para llevar a cabo la experimentación se han simulado distintas configuraciones o descripciones instantáneas de sistemas P2P en un momento determinado, sobre las que se han realizado 40 ensayos de búsqueda independientes, manteniendo inalterable estas configuraciones iniciales durante la ejecución de todos los ensayos y registrando los resultados de los mismos para su correspondiente análisis estadístico. Por configuración, o descripción instantánea de un sistema P2P, debe entenderse una determinada topología de red, con sus nodos y conexiones particulares junto a una determinada distribución de recursos compartidos sobre los nodos de la red.

Para simular estas descripciones instantáneas de un sistema P2P genérico se ha procedido de la siguiente manera:

Se han generado de manera aleatoria 10.000 vectores característicos agrupados en 100 clases distintas que representan el universo total de recursos compartidos en toda la red P2P. La representación elegida para los vectores característicos ha sido la codificación binaria. Existen diversos escenarios en los que esta representación resulta sumamente adecuada. A modo de ejemplo considérese una aplicación para compartir documentos de texto sobre una red P2P en la que se desea realizar búsquedas por palabras claves. En esta implementación cada posición del vector característico estaría representando la existencia o no de una determinada palabra clave en el texto compartido. Por lo tanto, la longitud del vector debe ser lo suficientemente grande como para reservar una posición para cada una de las posibles palabras claves existentes en la colección completa de documentos.

Además del caso de referencia comparativa representado por el algoritmo BFS de *Gnutella* se han definido 255 casos de prueba distintos para evaluar la estrategia BII-P2P bajo distintas parametrizaciones —ver sección 5.1.2 “Construcción de los Casos de Prueba” en pág. 141—. Cada uno de estos casos de

prueba fueron evaluados sobre cuatro topologías distintas de redes P2P creadas de manera aleatoria. Estas topologías se generaron construyendo grafos de nodos conectados aleatoriamente. Se utilizaron grafos de 2.000, 5.000, 10.000 y 20.000 nodos, eligiendo para cada nodo el tamaño de su vecindad también de manera aleatoria. Para la elección de la vecindad se han utilizado dos parámetros  $V_{min}$  y  $V_{max}$  que corresponden a la cantidad mínima y máxima de vecinos admitidos por cada nodo. En todas las pruebas realizadas se ha elegido  $V_{min} = 4$  y  $V_{max} = 12$ .

La asignación de los recursos que comparten los nodos también ha sido elaborada de manera aleatoria. Sin embargo, no se ha descuidado la conformación de patrones en la distribución de los mismos que simulan distintos perfiles de recursos compartidos en cada nodo. Para lograrlo se ha elegido, también en forma aleatoria, una distribución característica de recursos en cada uno de ellos. Así a cada nodo se le asignó una cierta cantidad de clases de recursos y para cada clase una cierta cantidad de recursos.

Asignar un recurso a un nodo implica simplemente tomar del universo una copia del vector característico seleccionado y agregarla a la colección local de recursos compartidos del nodo. Nótese que los nodos reciben una copia del vector característico elegido y nunca el original que se conserva en el universo permitiendo que el mismo recurso sea nuevamente elegido en asignaciones para otros nodos de la red. Así se simula la existencia de varias instancias de un mismo recurso en el sistema P2P.

Justamente una característica importante que se ha mantenido en todas las simulaciones llevadas a cabo es que la cantidad total de recursos asignada a toda la red ha superado ampliamente la cantidad disponible en el universo de recursos. Tal situación ha sido deliberadamente provocada para crear la existencia de varias instancias de un mismo recurso repartidas en la red P2P. Precisamente una de las métricas utilizadas para valorar el desempeño de los distintos casos examinados en relación a la búsqueda de un recurso en particular ha sido la cantidad de

instancias del mismo halladas del total existente en toda la red P2P.

La significación asignada a la información proveniente de cada vecino, parámetro  $\sigma$  en la ecuación (4-5), es sumamente importante en la construcción de los mensajes *NeuralInf* e impacta directamente en el rendimiento de la estrategia BII-P2P. Una etapa previa de experimentación ha permitido fijar este valor en 0,85. Por lo tanto, todas las pruebas que se detallan en el apartado siguiente han sido llevadas a cabo con  $\sigma = 0,85$ .

Para independizar la medición de la capacidad de búsqueda que se desea evaluar respecto de la capacidad de adaptación cuyo análisis se presenta más adelante —ver sección 5.2 “Evaluación de la Capacidad de Adaptación” en la página 176—, en la conformación de las descripciones instantáneas de la red P2P se ha utilizado siempre la misma metodología para adquirir conocimiento que consistió en garantizar que cada nodo haya propagado sus mensajes *NeuralInf* al menos 5 veces antes de comenzar con la simulación de las búsquedas y la recolección de datos experimentales.

Aún con un valor alto de  $\sigma$ , que favorece la propagación de la información más allá de la primera línea de vecinos, la realización de una gran cantidad de pruebas preliminares ha mostrado que asegurar un mayor número de transmisiones previas de mensajes *NeuralInf* a la realización de las pruebas, aunque transmite información atenuada de nodos lejanos, no produce mejores resultados durante la ejecución de las mismas. Esto también sugiere que el valor del campo TTL en los mensajes *NeuralInf* de los protocolos de actualización de conocimiento puede mantenerse relativamente pequeño, pues asegurar la propagación a nodos muy alejados no reviste mejoras significativas. Más adelante, durante la evaluación de las políticas de adquisición de conocimiento se ofrecen más precisiones al respecto.

### 5.1.2 Construcción de los Casos de Prueba

Como ya se mencionara en párrafos anteriores, se han realizado pruebas sobre cuatro tamaños de redes distintas: 2.000, 5.000, 10.000 y 20.000 nodos, generadas de manera aleatoria en la forma expuesta precedentemente. Por cada una de estas topologías se definieron 256 casos de pruebas. Uno de estos casos se ha utilizado como referencia comparativa y se corresponde con el algoritmo BFS de *Gnutella* para  $TTL=4$ . Los otros 255 casos de prueba constituyen distintas parametrizaciones de BII-P2P surgidas de la combinación de 15 valores distintos de la tasa de selección  $\gamma$  (0,2; 0,25; 0,3; 0,35; 0,4; 0,45; 0,5; 0,55; 0,6; 0,65; 0,7; 0,75; 0,8; 0,85 y 0,9) con 17 valores de TTL (4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19 y 20). De esta forma, la experimentación para evaluar la capacidad de búsqueda de BII-P2P se ha llevado a cabo definiendo un total de 1024 casos de pruebas distintos.

Para medir el rendimiento de la estrategia BII-P2P y del algoritmo BFS de *Gnutella* se han considerado dos métricas muy importantes:

- i) *Mensajes Query generados*: Se ha medido la cantidad de mensajes *Query* que producen las búsquedas en cada uno de los casos de prueba realizados. Esta magnitud impacta directamente sobre el tráfico generado en la red P2P y por lo tanto es deseable mantenerla en los niveles más bajos posibles.
- ii) *Instancias halladas*: Para cuantificar la capacidad de búsqueda se ha considerado la cantidad de instancias halladas del recurso buscado. Sin embargo, debido a que el total de instancias de cada recurso existente en la red P2P no es igual para todos los recursos compartidos—considérese el proceso aleatorio utilizado en la distribución de los recursos para la simulación descrito en la sección 5.1.1 “La Experimentación”—, en lugar de registrar el valor absoluto de instancias del recurso encontradas, se ha procedido a calcular el

porcentaje de dicho valor en relación a la cantidad total existente en toda la red P2P.

Debido a que los resultados obtenidos a partir de la realización de un único ensayo de búsqueda no resultan estadísticamente significativos, se ha procedido a la recolección de datos por medio de la ejecución de 40 ensayos de búsqueda independientes para cada uno de los casos de prueba evaluados. Todas estas búsquedas fueron definidas seleccionando en forma aleatoria el recurso buscado y el nodo iniciador de la solicitud.

En el apartado siguiente se describen los resultados de la experimentación para cada uno de los casos de prueba evaluados.

### 5.1.3 Resultados de la Experimentación

En esta sección se presentan y analizan los resultados más relevantes alcanzados durante la experimentación. El conjunto extenso de los datos obtenidos puede consultarse en las tablas A-1, A-2, A-3, A-4, A-5, A-6, A-7 y A-8 a partir de la página 243 dentro de la sección titulada “Apéndice A”.

#### 5.1.3.1 Acerca del Análisis de los Datos

A partir de los 40 ensayos de búsqueda independientes realizados por cada uno de los casos de prueba definidos, se han obtenido las muestras:  $Q_1, Q_2, \dots, Q_{40}$  y  $H_1, H_2, \dots, H_{40}$ , siendo  $Q_i$  la cantidad de mensajes *Query* generada durante el  $i$ -ésimo ensayo de búsqueda y  $H_i$  el porcentaje de instancias halladas del recurso buscado en el  $i$ -ésimo ensayo de búsqueda. Para caracterizar las muestras, se han calculado los estadísticos  $\bar{Q}$ ,  $\bar{H}$ ,  $S_Q$  y  $S_H$  que se corresponden con los respectivos promedios y desviaciones estándares muestrales.

**Tabla 5-1 Estadísticos empleados en el análisis de los datos**

<b>Dada la muestra <math>X_1, X_2, \dots, X_n</math></b>	
<b>Media o promedio muestral</b>	$\bar{X} = \left(\frac{1}{n}\right) \sum_{i=1}^n X_i$
<b>Desviación estándar muestral</b>	$S_X = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}$

Se pretende comparar los valores de los promedios muestrales  $\bar{Q}$  y  $\bar{H}$  alcanzados por la búsqueda inteligente en todas sus versiones —distintos valores de TTL y  $\gamma$ — con respecto a los obtenidos por el algoritmo BFS de *Gnutella*. Las tablas de resultados presentadas en esta sección y en el “Apéndice A” muestran claras diferencias entre estas estrategias. Sin embargo es necesario establecer si tales diferencias son estadísticamente significativas. Esta cuestión se resuelve por medio de un test de hipótesis sobre diferencia de medias. Debido a que las muestras obtenidas por la experimentación llevada a cabo presentan una distribución cercana a la distribución Normal —se aplicó la prueba para normalidad de Kolmogorov-Smirnov—, es posible utilizar el test *t-Student* para diferencias de medias a partir de muestras independientes.

Básicamente el test *t-Student* para diferencia de medias postula como hipótesis nula que ambas medias son iguales y como hipótesis alternativa que las medias son distintas. Como resultado de aplicar el test *t-Student* sobre ambas muestras se obtiene el p-valor que representa el nivel de significancia  $\alpha$  más pequeño que conduce al rechazo de la hipótesis nula. El nivel  $\alpha$  de significancia de un test se define como la probabilidad de cometer error de tipo I, es decir de rechazar la hipótesis nula cuando ésta es verdadera. Consecuentemente, cuanto más pequeño sea el p-valor calculado, más evidencia habrá en contra de la

hipótesis nula a favor de la alternativa, es decir a favor de la aceptación que las medias son distintas.

El p-valor cuantifica lo verosímil que resulta obtener muestras como las actuales si la hipótesis nula es verdadera. Informar este valor permite decidir qué hipótesis aceptar bajo distintos niveles  $\alpha$  de significancia. En el presente trabajo se ha establecido  $\alpha = 0,01$  y por lo tanto se considera que si  $p\text{-valor} \leq 0,01$  existe suficiente evidencia en contra de la hipótesis nula concluyendo que la diferencia en los promedios muestrales observada es estadísticamente significativa.

Por lo tanto, en las tablas presentadas en esta sección se visualizan los p-valores obtenidos por la aplicación del test *t-Student* sobre las muestras de la búsqueda inteligente y el algoritmo BFS de *Gnutella*. Así es posible apreciar el nivel de significancia de las diferencias observadas. Para facilitar el análisis de los datos, en la presente sección también se utilizan diagramas de cajas que permiten visualizar fácilmente aspectos sobresalientes de las muestras brindando información sobre la distribución de los datos. —cuartiles 1, 2 y 3, valores máximo y mínimo de la muestra—.

A continuación se analizan por separado los datos obtenidos durante la experimentación realizada para cada uno de los tamaños de red considerados.

### **5.1.3.2 Rendimiento de la Búsqueda Inteligente sobre Red de 2.000 Nodos**

El conjunto de datos completo obtenido durante la experimentación realizada sobre la red de 2.000 nodos puede consultarse en la Tabla A-1 que, por razones de visualización se encuentra en la sección “Apéndice A” a partir de la pág. 243. Sin embargo, para facilitar la lectura de los resultados, en la presente sección se resumen los más relevantes.



**Tabla 5-2 Casos evaluados sobre la red de 2.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en ambas métricas consideradas. Las diferencias observadas para  $\bar{Q}$  y  $\bar{H}$  con respecto a BFS de *Gnutella* resultan estadísticamente significativa —p-valor inferior a 0,01—**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
<i>Gnutella</i> BFS (TTL=4)	3913,4	1204,5		81,79	15,24	
BII-P2P (TTL=6; $\gamma$ =0,55)	2041,65	492,63	< 0.0001	88,72	10,05	0,0097
BII-P2P (TTL=7; $\gamma$ =0,50)	2679,6	348	< 0.0001	94,36	6,35	< 0.0001
BII-P2P (TTL=7; $\gamma$ =0,55)	3085,63	344,85	0,0001	95,26	5,2	< 0.0001
BII-P2P (TTL=8; $\gamma$ =0,45)	1633	334,32	< 0.0001	90,26	8,84	0,0013
BII-P2P (TTL=8; $\gamma$ =0,50)	3222,58	198,41	0,0009	95,9	6,08	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,45)	1943,23	226,73	< 0.0001	92,56	6,88	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,40)	1419,03	173,84	< 0.0001	89,36	7,81	0,002
BII-P2P (TTL=10; $\gamma$ =0,45)	2124,95	135,57	< 0.0001	93,59	6,35	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,40)	1531,63	116,22	< 0.0001	90,64	6,56	0,0004
BII-P2P (TTL=11; $\gamma$ =0,45)	2217,48	92,83	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	1597,13	82,17	< 0.0001	91,03	6,46	0,0002
BII-P2P (TTL=12; $\gamma$ =0,45)	2265,35	74,52	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=13; $\gamma$ =0,40)	1634,68	64,6	< 0.0001	91,41	6,32	0,0001
BII-P2P (TTL=13; $\gamma$ =0,45)	2288,45	69,88	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,40)	1654,18	56,97	< 0.0001	91,67	5,97	0,0001
BII-P2P (TTL=14; $\gamma$ =0,45)	2300,93	68,29	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,40)	1663,88	53,06	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,45)	2307,33	67,76	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,40)	1669,33	52,3	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,45)	2309,78	67,67	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,40)	1672,75	50,93	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,45)	2311,15	67,14	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,40)	1674,43	50,76	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,40)	1674,93	50,66	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,40)	1675,1	50,39	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001

En la primera fila de la Tabla 5-2 se presentan los resultados experimentales obtenidos sobre la red de 2.000 nodos de la búsqueda BFS de *Gnutella* con TTL=4 que se utiliza como referente comparativo. Las filas siguientes muestran los valores alcanzados por la estrategia de búsqueda inteligente BII-P2P para distintas parametrizaciones ventajosas de TTL y  $\gamma$ . Claramente todos los casos BII-P2P presentados exhiben un mejor rendimiento que la metodología BFS de

*Gnutella* obteniendo un mayor porcentaje de hallazgos con un número menor de mensajes *Query* generados. El p-valor del test *t-Student*, que en todos los casos es muy inferior a 0.01, permite concluir que las diferencias en las medias observadas son estadísticamente significativas.

Los diagramas de caja que se muestran en las figuras 5-1 y 5-2 que se presentan más adelante en las páginas 147 y 148 respectivamente, permiten una mejor visualización de los datos descriptos en la Tabla 5-2.

Un diagrama de caja consiste en una caja rectangular dispuesta en forma horizontal —también puede ser en sentido vertical—, donde la base muestra el recorrido entre los cuartiles desde el primero hasta el tercero. Este rectángulo está dividido por un segmento vertical que indica donde se posiciona la mediana —segundo cuartil— haciendo visible su relación con los cuartiles primero y tercero. La caja se ubica a escala sobre un segmento que tiene como extremos los valores mínimo y máximo de la variable. Por lo tanto, los diagramas de caja son gráficas sumamente útiles que permiten una rápida comparación visual sobre la dispersión de los datos en cada una de las muestras representadas.

La Figura 5-1 presentada en la página 147, muestra el diagrama de cajas para los datos obtenidos sobre la cantidad de mensajes *Query* generados por los distintos métodos evaluados. Claramente todos los casos BII-P2P representan una mejor opción que el algoritmo BFS de *Gnutella* observando que los resultados obtenidos se concentran con mucha menor dispersión alrededor de valores más pequeños. Esta situación se hace más evidente para los métodos BII-P2P con TTL mayor a 9.

La Figura 5-2 presentada en la página 148, muestra el diagrama de cajas respecto del porcentaje de instancias halladas del recurso buscado por los distintos métodos evaluados. Al igual que en el gráfico de la Figura 5-1, también aquí se observa que todos los casos BII-P2P representan una mejor opción que el

algoritmo BFS de *Gnutella* con menor dispersión de datos agrupados alrededor de valores más altos.

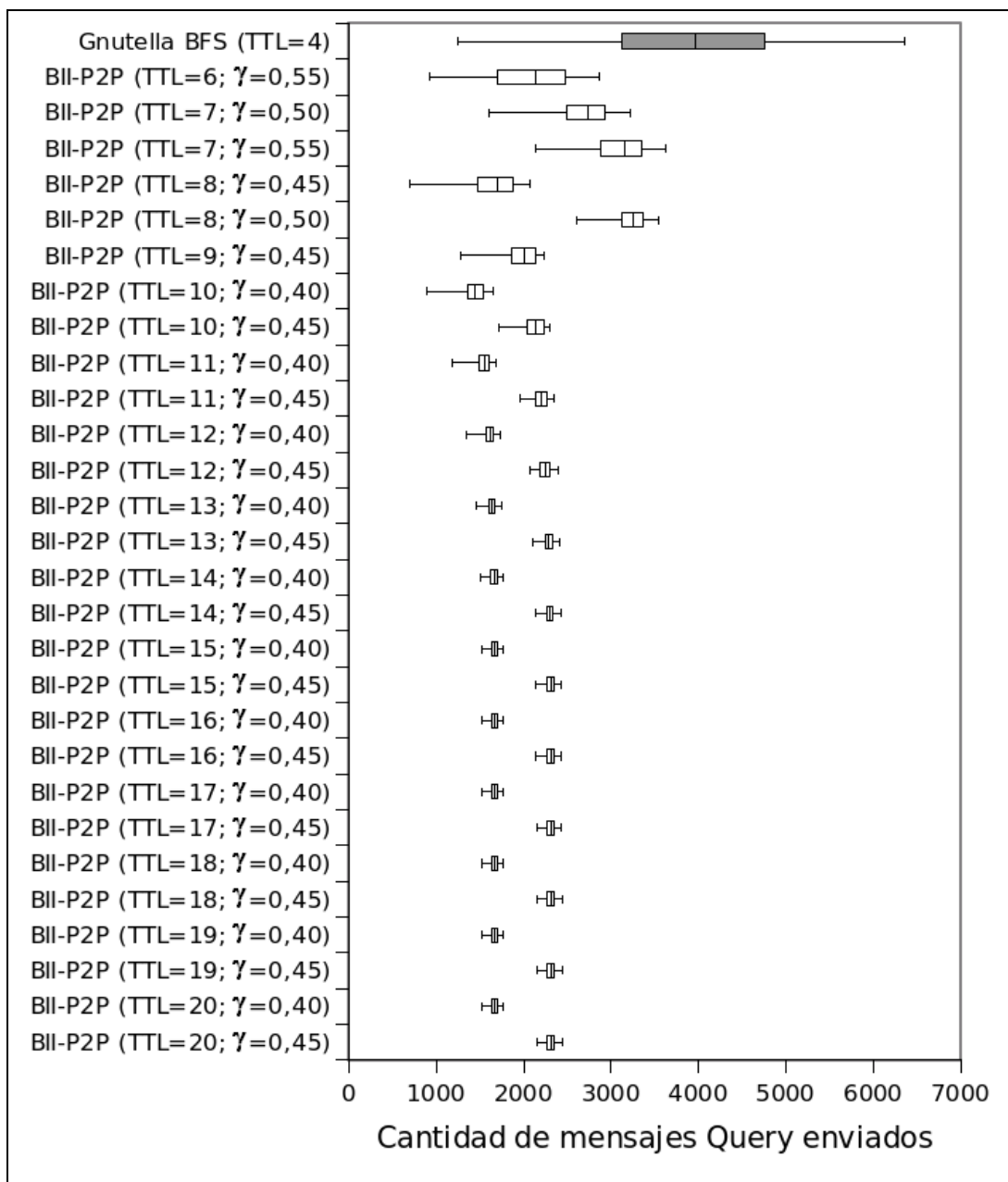


Figura 5-1 Diagrama de cajas. Cantidad de mensajes *Query* generados sobre red de 2.000 nodos

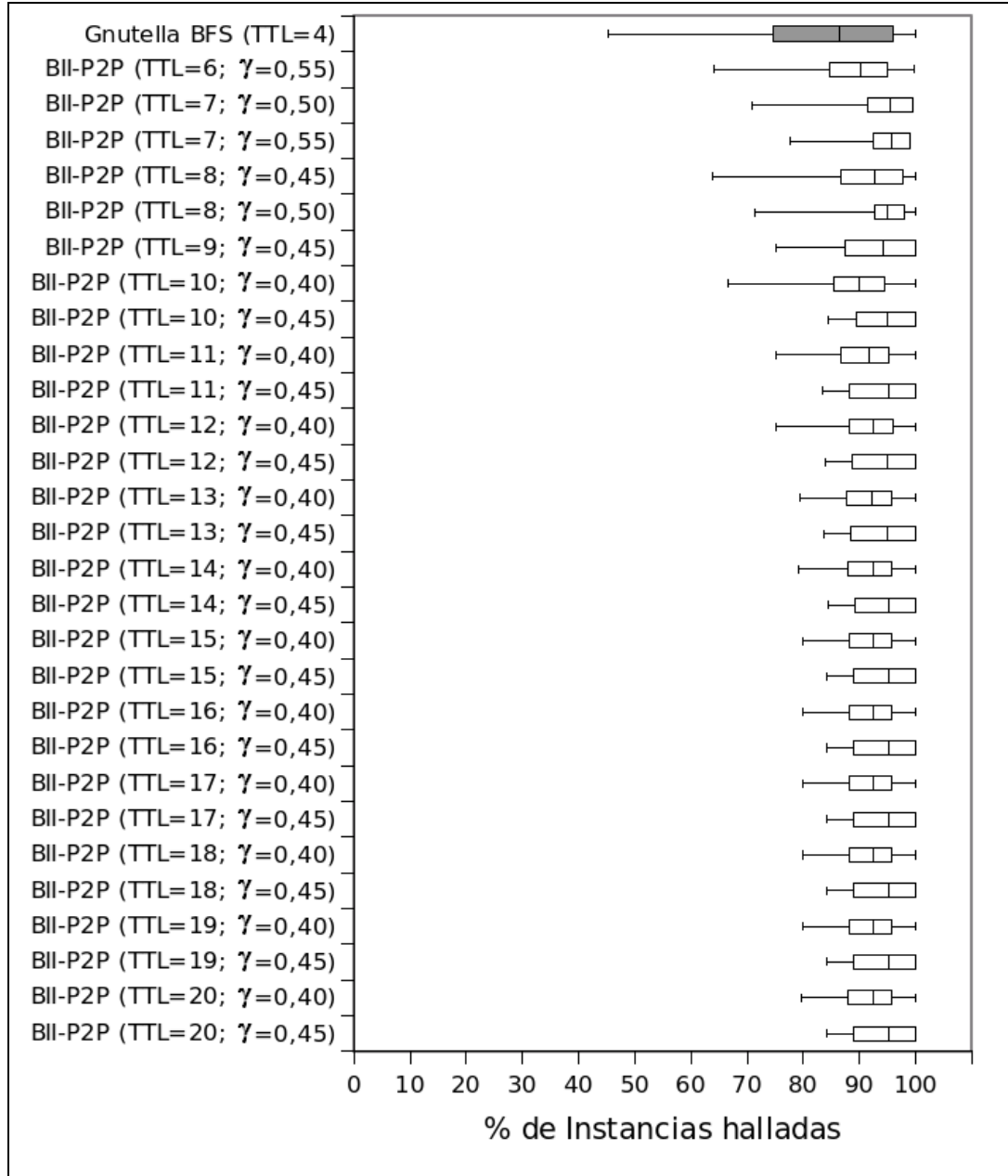


Figura 5-2 Diagrama de cajas. % de Instancias halladas del recurso buscado sobre red de 2.000 nodos.

Los diagramas de caja presentados permiten visualizar fácilmente cuán convenientes resultan algunas parametrizaciones de BII-P2P respecto del algoritmo BFS de *Gnutella*. Además, comparando ambos diagramas surge la

relación que existe entre la cantidad de mensajes *Query* generados y el porcentaje de instancias del recurso halladas. Esta relación se muestra más claramente en la Figura 5-3 en la que se han graficado sobre el plano  $\mathbb{R}^2$  los puntos  $(\bar{H}, \bar{Q})$  para todos los casos presentados en la Tabla 5-2.

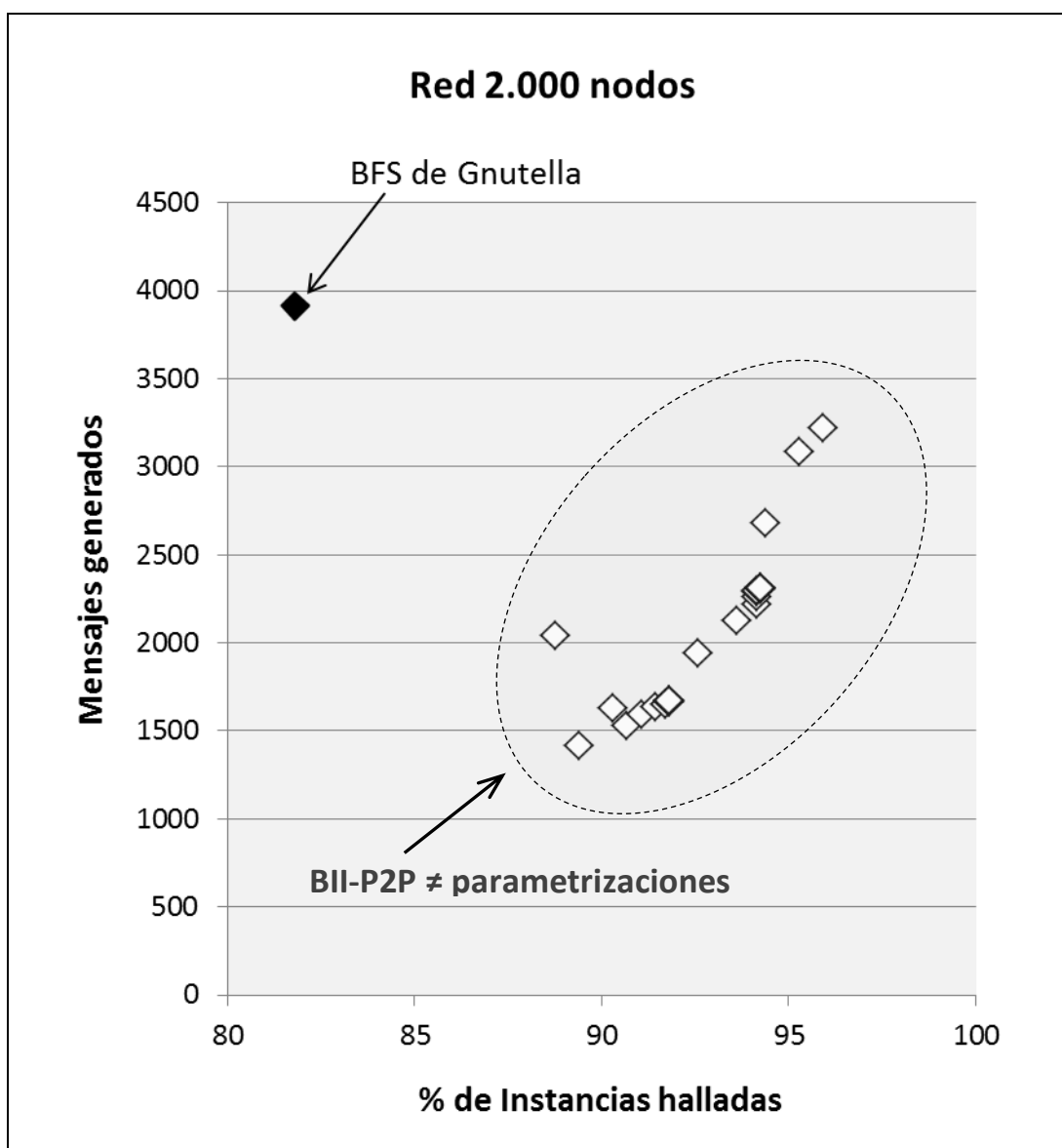


Figura 5-3 Relación entre mensajes *Query* generados y porcentaje de instancias halladas sobre red de 2.000 nodos

Claramente BII-P2P es preferible a BFS de Gnutella para todos los casos expuestos en la Tabla 5-2. Para puntualizar algunos ejemplos considérese el rendimiento alcanzado por BII-P2P con  $TTL=10$  y  $\gamma=0,40$  que utiliza 1.419,03 mensajes *Query* y encuentra el 89,36 % de las instancias existentes del recurso buscado —compárese con BFS que emplea 3.913,4 mensajes de *Query* y encuentra el 81,79% de las instancias—. Otro caso interesante se observa para BII-P2P con  $TTL=7$  y  $\gamma=0,55$  dónde se consigue el 95,26% de hallazgos de instancias con 3.222,58 mensajes *Query* generados. Ambos ejemplos se visualizan en la Figura 5-4.

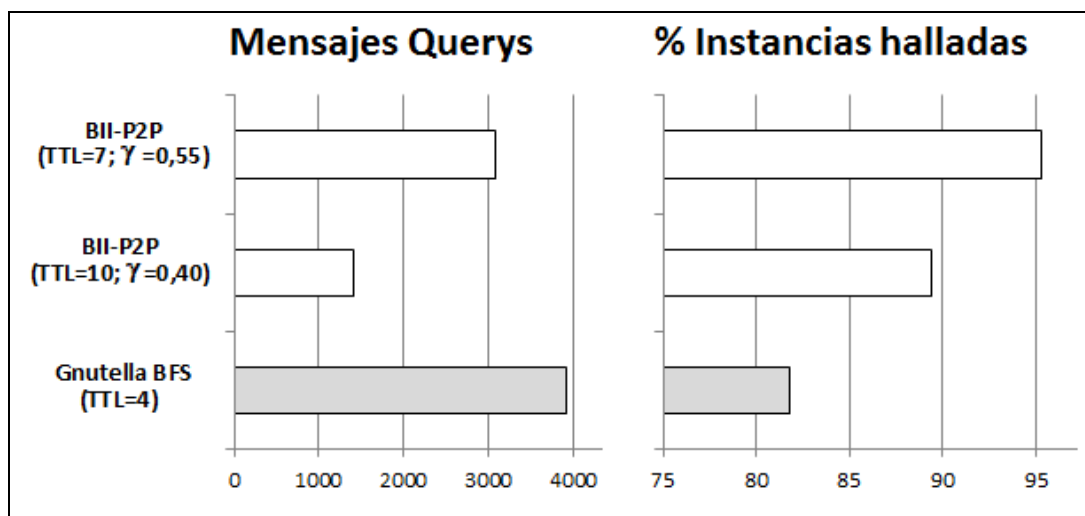


Figura 5-4 Comparativa de rendimiento sobre la red de 2.000 nodos para los casos de ejemplos puntualizados en la presente sección

Además de los casos expuestos, existen otros donde, si bien la media calculada resulta mejor en ambos casos —menor cantidad de mensajes *Query* y mayor porcentaje de instancias halladas—, los p-valores obtenidos por la prueba *t-Student* indican que en sólo una de las métricas consideradas, la diferencia entre las medias debe considerarse estadísticamente significativa. Se puede afirmar entonces que estos casos que se exponen en la tabla Tabla 5-3 Superan en

rendimiento al algoritmo BFS de Gnutella o bien generando menor cantidad de mensajes *Query*, o bien encontrando un mayor porcentaje de instancias del recurso manteniendo la otra métrica en niveles similares al alcanzado por BFS.

**Tabla 5-3 Casos evaluados sobre la red de 2.000 nodos en los que BII-P2P supera al algoritmo BFS de Gnutella en sólo una de las métricas consideradas y lo iguala en la restante. En todos los casos BII-P2P el p-valor es inferior a 0,01 sólo para una de las métricas observadas.**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
<i>Gnutella</i> BFS (TTL=4)	3913,4	1204,5		81,79	15,24	
BII-P2P (TTL=5; $\gamma$ =0,65)	1982,93	653,19	< 0.0001	82,69	12,65	0,5899
BII-P2P (TTL=5; $\gamma$ =0,70)	2913,03	871,61	< 0.0001	88,72	10,83	0,0146
BII-P2P (TTL=6; $\gamma$ =0,50)	1721,38	434,05	< 0.0001	86,92	11,23	0,0504
BII-P2P (TTL=6; $\gamma$ =0,60)	3409,2	596,41	0,0211	94,36	5,71	< 0.0001
BII-P2P (TTL=7; $\gamma$ =0,45)	1189,48	365,85	< 0.0001	83,08	14,49	0,4672
BII-P2P (TTL=8; $\gamma$ =0,55)	3620,65	185,69	0,1364	96,54	4,5	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,40)	1237,78	235,76	< 0.0001	85,77	10,31	0,0708
BII-P2P (TTL=9; $\gamma$ =0,50)	3468,63	119,13	0,0253	96,41	4,68	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,55)	3852,15	109,04	0,7504	96,67	4,19	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,50)	3576,08	90,71	0,0851	96,41	4,68	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,50)	3621,05	78,55	0,1336	96,41	4,68	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	820,95	142,54	< 0.0001	82,82	11,85	0,4263
BII-P2P (TTL=12; $\gamma$ =0,50)	3638,6	73,49	0,1577	96,41	4,68	< 0.0001
BII-P2P (TTL=13; $\gamma$ =0,35)	871,78	126,61	< 0.0001	84,23	10,2	0,2052
BII-P2P (TTL=13; $\gamma$ =0,50)	3646,95	72,11	0,1704	96,41	4,68	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,35)	908,88	109,09	< 0.0001	84,49	9,44	0,1716
BII-P2P (TTL=14; $\gamma$ =0,50)	3650,2	71,5	0,1755	96,41	4,68	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,35)	932,88	95,66	< 0.0001	85,26	8,46	0,0992
BII-P2P (TTL=15; $\gamma$ =0,50)	3651,9	71,62	0,1783	96,41	4,68	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,35)	949,15	85,26	< 0.0001	85,51	8,23	0,0807
BII-P2P (TTL=16; $\gamma$ =0,50)	3652,58	71,52	0,1794	96,41	4,68	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,35)	959,6	77,89	< 0.0001	85,51	8,23	0,0807
BII-P2P (TTL=17; $\gamma$ =0,50)	3652,93	71,55	0,1799	96,41	4,68	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,35)	965,28	72,4	< 0.0001	85,64	7,94	0,0725
BII-P2P (TTL=18; $\gamma$ =0,50)	3653,05	71,46	0,1801	96,41	4,68	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,35)	969,08	68,68	< 0.0001	85,77	8,04	0,0657
BII-P2P (TTL=19; $\gamma$ =0,50)	3653,2	71,37	0,1804	96,41	4,68	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,35)	972,08	67,03	< 0.0001	85,9	7,78	0,0555
BII-P2P (TTL=20; $\gamma$ =0,50)	3653,3	71,31	0,1805	96,41	4,68	< 0.0001

A modo de ejemplo considérese el rendimiento alcanzado por BII-P2P con

TTL=12 y  $\gamma=0,35$  que si bien exhibe una pequeña diferencia a su favor respecto del porcentaje de instancias halladas —82,82% de BII-P2P contra 81,79% de BFS— esta diferencia no puede considerarse estadísticamente significativa —p-valor = 0,4263 > 0,01—; sin embargo supera notablemente a BFS al utilizar en promedio sólo 820,95 mensajes *Query* contra los 3.913,4 de BFS.

### 5.1.3.3 Rendimiento de la Búsqueda Inteligente sobre Red de 5.000 Nodos

El conjunto de datos completo obtenido durante la experimentación realizada sobre la red de 5.000 nodos puede consultarse en la Tabla A-2 que debido a su extensión se encuentra en la sección “Apéndice A” a partir de la pág. 247. Sin embargo, para facilitar la lectura de los resultados, en la presente sección se resumen los más relevantes.

En la primera fila de la Tabla 5-4 se presentan los resultados experimentales obtenidos sobre la red de 5.000 nodos por la búsqueda BFS de *Gnutella* con TTL=4. Las filas siguientes muestran los valores alcanzados por la estrategia de búsqueda inteligente BII-P2P para distintas parametrizaciones ventajosas de TTL y  $\gamma$ . Claramente todos los casos BII-P2P presentados exhiben un mejor rendimiento que la metodología BFS de *Gnutella* obteniendo un mayor porcentaje de hallazgos con un número menor de mensajes *Query* generados. El p-valor del test *t-Student*, que en todos los casos es muy inferior a 0.01, permite concluir que las diferencias en las medias observadas son estadísticamente significativas.

**Tabla 5-4 Casos evaluados sobre la red de 5.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en ambas métricas consideradas. Las diferencias observadas para  $\bar{Q}$  y  $\bar{H}$  con respecto a BFS de *Gnutella* resultan estadísticamente significativa —p-valor inferior a 0,01—**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
<i>Gnutella</i> BFS (TTL=4)	4284,25	1451,5		56,78	14,96	
BII-P2P (TTL=6; $\gamma=0,50$ )	2319,43	729,52	< 0.0001	69,59	16,25	0,0005

Continúa la tabla...



Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
BII-P2P (TTL=6; $\gamma$ =0,55)	2882,05	949,5	< 0.0001	73,73	16,58	< 0.0001
BII-P2P (TTL=8; $\gamma$ =0,40)	1572,93	553,95	< 0.0001	68,31	18,14	0,0019
BII-P2P (TTL=8; $\gamma$ =0,45)	2906,58	910,94	< 0.0001	80,85	16,19	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,40)	2281,55	641,65	< 0.0001	79,11	15,74	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,40)	2878,23	634,11	< 0.0001	84,74	13,72	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,35)	1400,73	433,93	< 0.0001	72,3	18,07	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,40)	3302,13	576,86	0,0002	87,66	10,36	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	1662,43	451,38	< 0.0001	76,91	16,4	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	3587,13	498,91	0,006	89,5	6,72	< 0.0001
BII-P2P (TTL=13; $\gamma$ =0,35)	1863,4	441,87	< 0.0001	79,62	14,56	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,35)	2014,53	412,67	< 0.0001	81,21	11,98	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,35)	2126,6	364,55	< 0.0001	82,8	8,04	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,30)	952,28	239,26	< 0.0001	67,54	12,81	0,0009
BII-P2P (TTL=16; $\gamma$ =0,35)	2208,2	300,78	< 0.0001	83,67	6,46	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,30)	1028,33	218,64	< 0.0001	70,05	10,97	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,35)	2265,05	235,28	< 0.0001	84,69	5,13	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,30)	1092,28	193,8	< 0.0001	71,63	9,86	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,35)	2306,28	176,44	< 0.0001	84,95	5,13	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,30)	1143,18	168,64	< 0.0001	73,27	8,26	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,35)	2334,78	132,98	< 0.0001	85,36	5,18	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,30)	1183,4	146,22	< 0.0001	74,45	7,47	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,35)	2352,68	103,81	< 0.0001	85,36	5,18	< 0.0001

Los diagramas de caja que se presentan más adelante —ver Figura 5-5 y Figura 5-6—, permiten una mejor visualización de los datos descritos en la Tabla 5-4 y posibilitan una rápida comparación visual sobre la dispersión de las muestras obtenidas.

La Figura 5-5 muestra el diagrama de cajas para los datos obtenidos sobre la cantidad de mensajes *Query* generados por los distintos métodos evaluados. Claramente todos los casos BII-P2P representan una mejor opción que el algoritmo BFS de *Gnutella* observando que los datos se concentran con mucha menor dispersión alrededor de valores más pequeños. Esta situación se hace más evidente para los métodos BII-P2P con TTL mayor a 10.

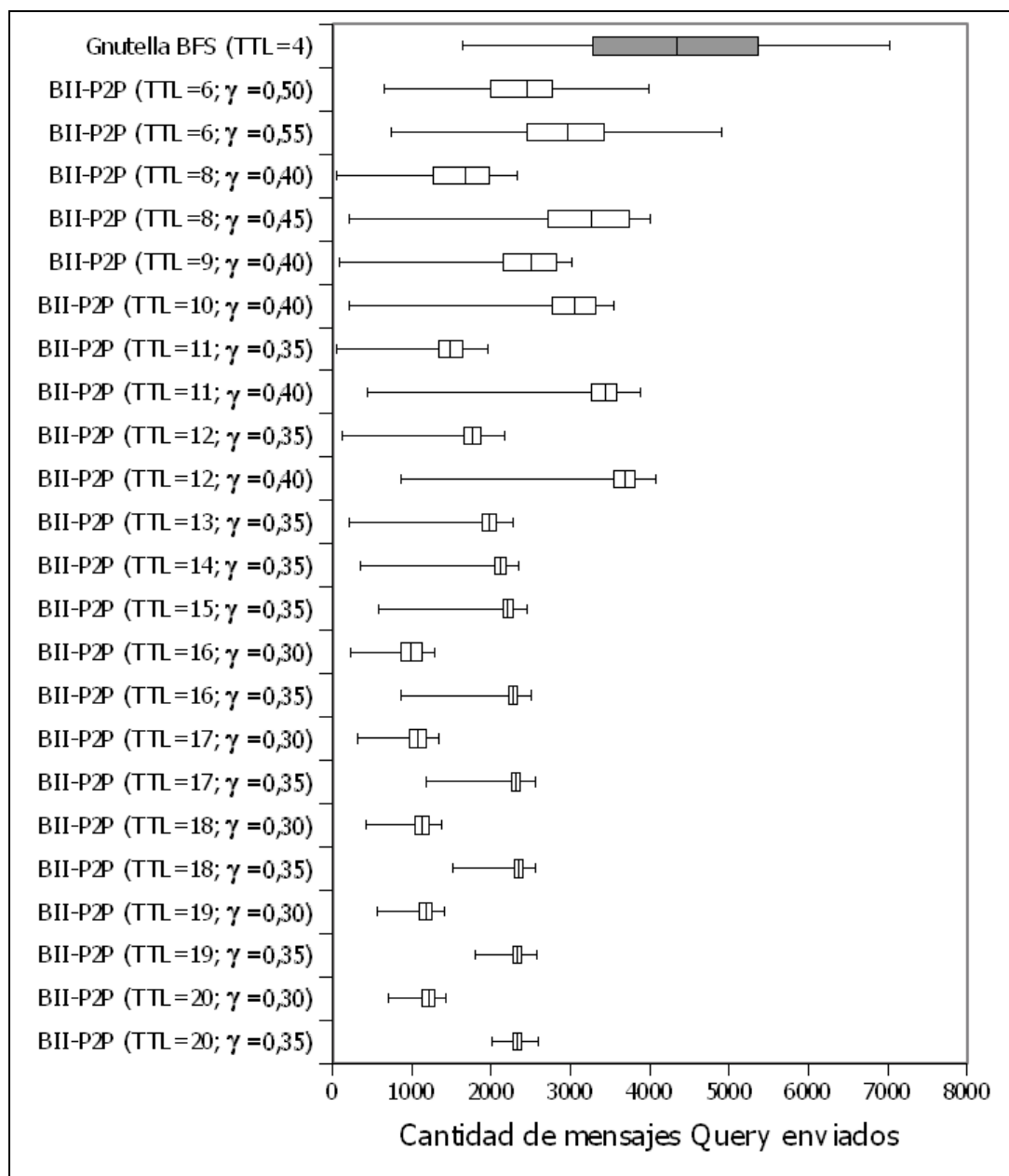


Figura 5-5 Diagrama de cajas. Cantidad de mensajes *Query* generados sobre red de 5.000 nodos

La Figura 5-6 muestra el diagrama de cajas para el porcentaje de instancias halladas del recurso buscado por los distintos métodos evaluados. Se observa que todos los casos BII-P2P superan el rendimiento del algoritmo BFS de *Gnutella*

con menor dispersión de datos agrupados alrededor de valores más altos. Sin embargo se observan algunos valores extremos de mala calidad cuya ocurrencia va disminuyendo a medida que aumenta el TTL. Obsérvese que para TTL > 16 y  $\gamma = 0.35$  los resultados son muy ventajosos.

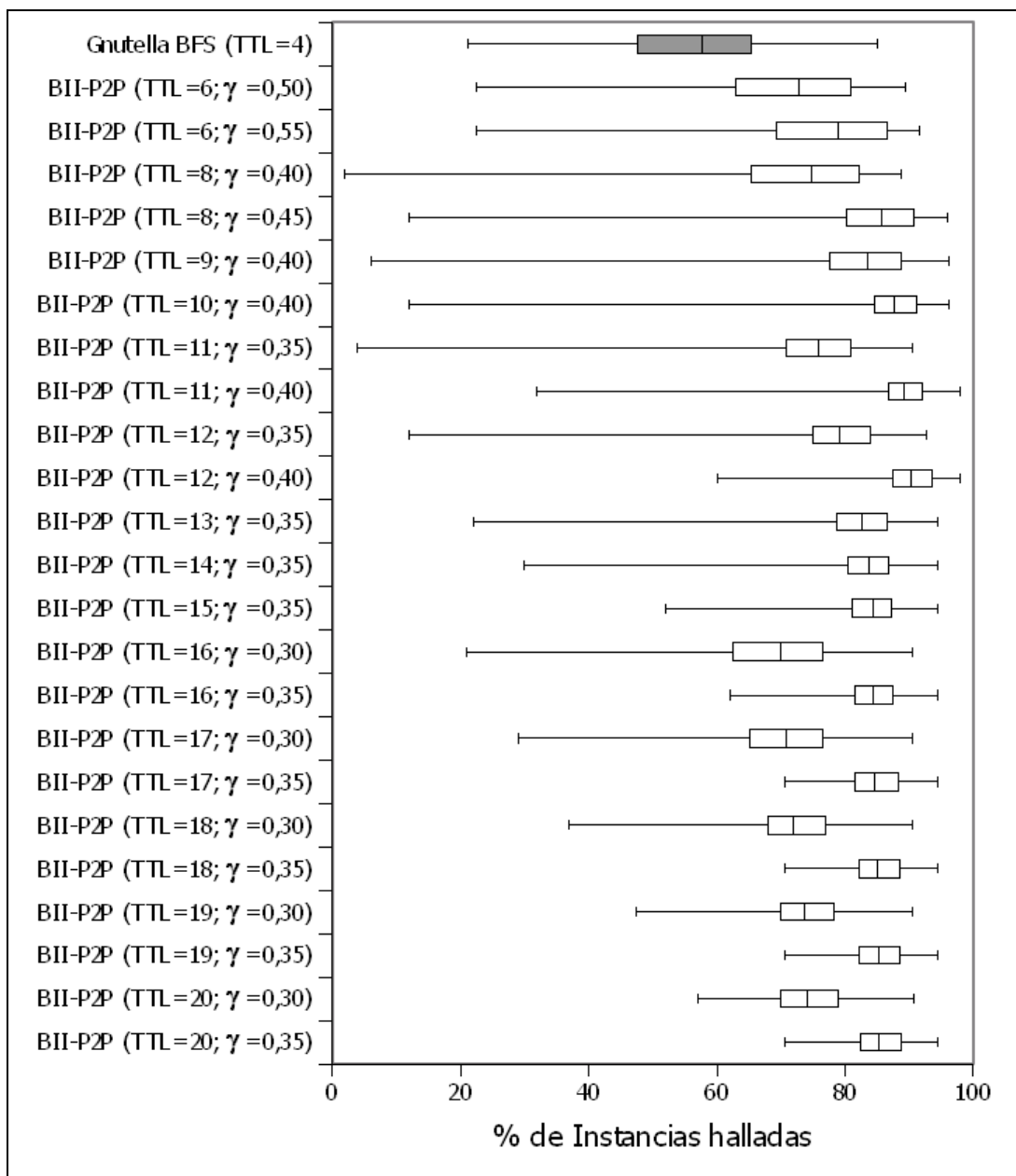


Figura 5-6 Diagrama de cajas. % de Instancias halladas del recurso buscado sobre red de 5.000 nodos.

La relación existente entre la cantidad de mensajes *Query* generados y el porcentaje de instancias halladas por las distintas estrategias sobre la red de 5.000 nodos se muestra claramente en la Figura 5-7 en la que se han graficado sobre el plano  $\mathbb{R}^2$  los puntos  $(\bar{H}, \bar{Q})$  para todos los casos presentados en la Tabla 5-4.

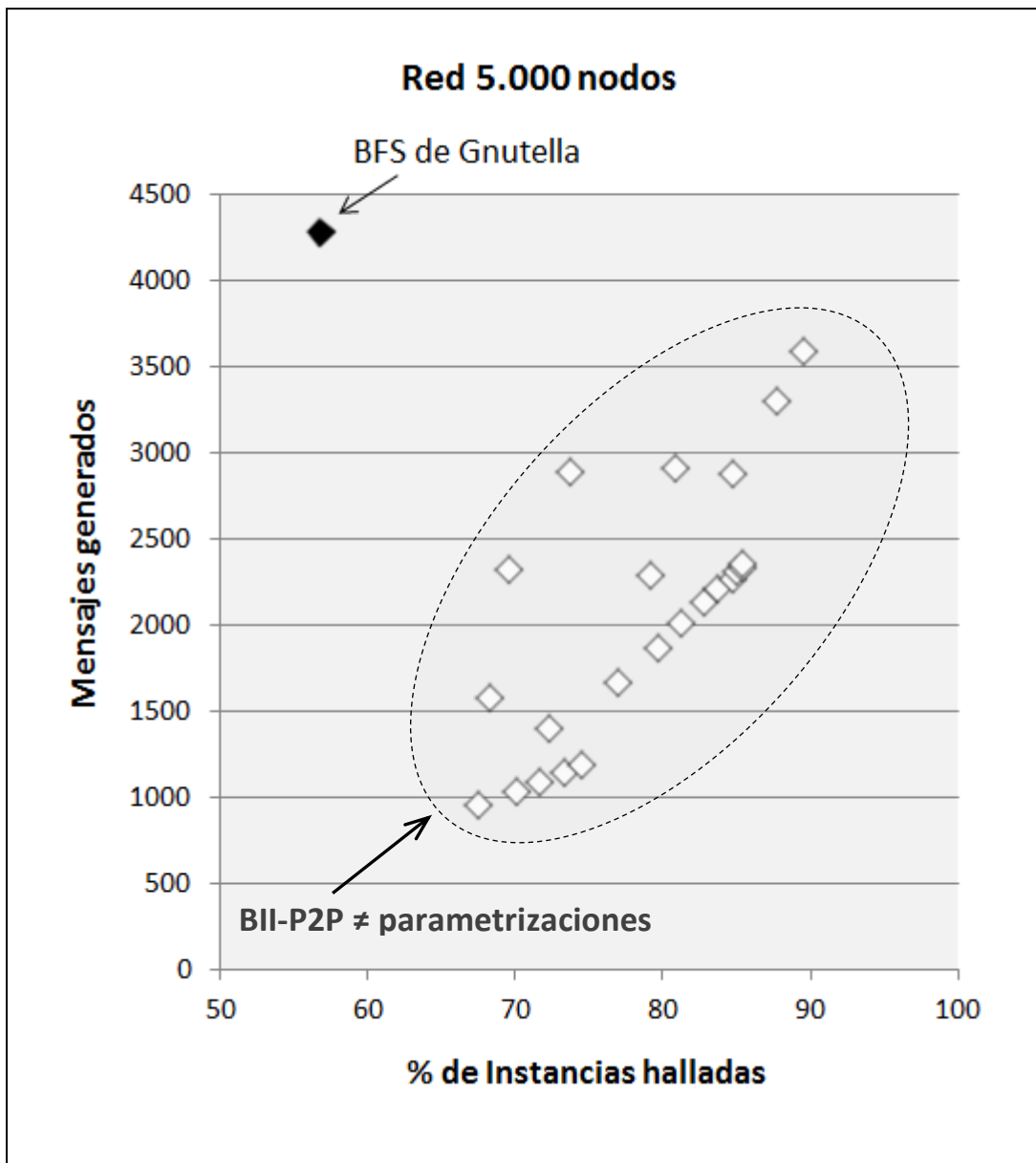


Figura 5-7 Relación entre mensajes *Query* generados y porcentaje de instancias halladas sobre red de 5.000 nodos

Claramente BII-P2P es preferible a BFS de Gnutella para todos los casos expuestos en la Tabla 5-4. Para puntualizar algunos ejemplos considérese el rendimiento alcanzado por BII-P2P con  $TTL=16$  y  $\gamma=0,30$  que utiliza 952,28 mensajes *Query* y encuentra el 67,54 % de las instancias existentes del recurso buscado —compárese con BFS que emplea 4.284,25 mensajes de *Query* y encuentra el 56,78% de las instancias—. Otro caso interesante se observa para BII-P2P con  $TTL=12$  y  $\gamma=0,40$  dónde se consigue el 89,5% de hallazgos de instancias con 3.587,13 mensajes *Query* generados. Ambos ejemplos se visualizan en la Figura 5-8.

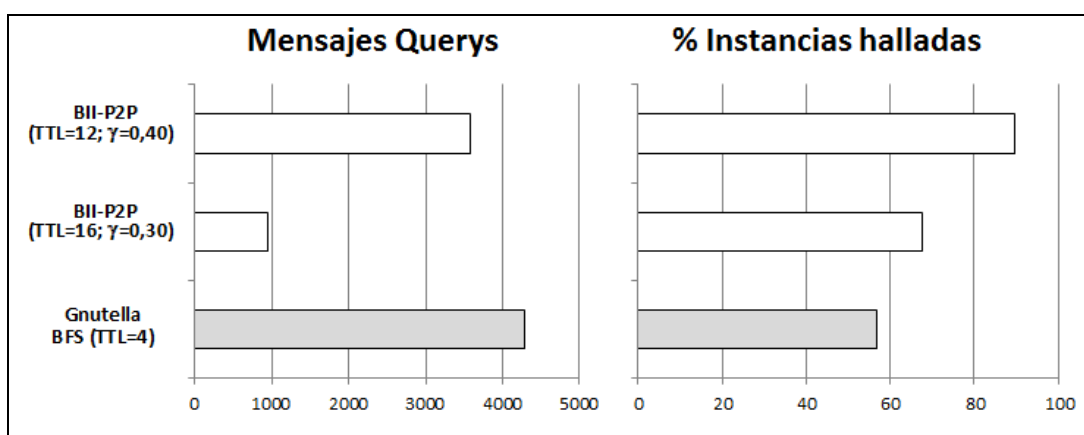


Figura 5-8 Comparativa de rendimiento sobre la red de 5.000 nodos para los casos de ejemplos puntualizados en la presente sección

La Tabla 5-5 que se presenta a continuación expone los resultados de otros casos de prueba donde, si bien en principio se observa que BII-P2P supera a BFS con menor cantidad de mensajes *Query* y mayor porcentaje de instancias halladas, los p-valores de la prueba *t-Student* indican que las diferencias en las medias observada resultan estadísticamente significativa en sólo una de ambas métricas consideradas. Se puede afirmar entonces que en todos estos casos BII-P2P supera al algoritmo BFS de Gnutella o bien generando menor cantidad de mensajes

*Query* con el mismo porcentaje de hallazgos; o bien mejorando el porcentaje de instancias halladas con la misma cantidad de tráfico generado sobre la red.

**Tabla 5-5 Casos evaluados sobre la red de 5.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en sólo una de las métricas consideradas y lo iguala en la restante. En todos los casos BII-P2P el p-valor es inferior a 0,01 en una sola de ambas métricas observadas.**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
BII-P2P (TTL=5; $\gamma$ =0,65)	2343,6	915,83	< 0.0001	59,19	17,62	0,5101
BII-P2P (TTL=5; $\gamma$ =0,70)	3595,4	1339,41	0,0304	69,48	18,14	0,001
BII-P2P (TTL=7; $\gamma$ =0,45)	1724,45	676,61	< 0.0001	66,31	20,88	0,0194
BII-P2P (TTL=9; $\gamma$ =0,45)	3944,28	911,95	0,2142	87,97	11,42	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,35)	1095,18	387,11	< 0.0001	65,13	19,22	0,0292
BII-P2P (TTL=13; $\gamma$ =0,40)	3765,63	391,53	0,0344	90,32	4,87	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,30)	766,73	262,64	< 0.0001	60,16	16,09	0,3308
BII-P2P (TTL=14; $\gamma$ =0,40)	3873,35	287,29	0,0863	90,83	4,44	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,30)	866,68	254,93	< 0.0001	64,26	14,09	0,025
BII-P2P (TTL=15; $\gamma$ =0,40)	3934,83	204,94	0,1394	91,19	4,13	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,40)	3972,23	151,43	0,1839	91,24	4,18	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,40)	3992,4	131,06	0,2127	91,24	4,18	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,40)	4003	121,18	0,2292	91,3	4,24	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,40)	4008,35	118,35	0,238	91,3	4,24	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,40)	4011,38	116,8	0,243	91,3	4,24	< 0.0001

A modo de ejemplo considérese el rendimiento alcanzado por BII-P2P con TTL=14 y  $\gamma$ =0,30 que si bien exhibe una pequeña diferencia a su favor respecto del porcentaje de instancias halladas —60,16% de BII-P2P contra 56,78% de BFS— esta diferencia no puede considerarse estadísticamente significativa —p-valor = 0,3308 > 0,01—; sin embargo supera notablemente a BFS al utilizar en promedio sólo 766,73 mensajes *Query* contra los 4.284,25 de BFS.

#### 5.1.3.4 Rendimiento de la Búsqueda Inteligente sobre Red de 10.000 Nodos

El conjunto de datos completo obtenido durante la experimentación realizada sobre la red de 10.000 nodos puede consultarse en la Tabla A-3 en la sección

“Apéndice A” a partir de la pág. 251. Sin embargo, para facilitar la lectura de los resultados en la presente sección se resumen los más relevantes.

En la primera fila de la Tabla 5-6 se presentan los resultados experimentales alcanzados por la búsqueda BFS de *Gnutella* con TTL=4 sobre la red de 10.000 nodos. Las filas siguientes muestran los valores arrojados por la estrategia de búsqueda inteligente BII-P2P para distintas parametrizaciones ventajosas de TTL y  $\gamma$ . Claramente todos los casos BII-P2P presentados muestran un mejor rendimiento que la metodología BFS de *Gnutella* obteniendo un mayor porcentaje de hallazgos con un número menor de mensajes *Query* generados. El p-valor del test *t-Student*, que en todos los casos es muy inferior a 0.01, permite concluir que las diferencias en las medias observadas son estadísticamente significativas.

**Tabla 5-6 Casos evaluados sobre la red de 10.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en ambas métricas consideradas. Las diferencias observadas para  $\bar{Q}$  y  $\bar{H}$  con respecto a BFS de *Gnutella* resultan estadísticamente significativa —p-valor inferior a 0,01—**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
<i>Gnutella</i> BFS (TTL=4)	4525,53	1297,2		36,14	9,26	
BII-P2P (TTL=6; $\gamma$ =0,50)	2606,33	932,32	< 0.0001	52,31	13,67	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,55)	3232,43	1219,37	< 0.0001	56,37	14,63	< 0.0001
BII-P2P (TTL=7; $\gamma$ =0,45)	2100,53	798,79	< 0.0001	52,38	13,96	< 0.0001
BII-P2P (TTL=8; $\gamma$ =0,40)	1967,95	765,71	< 0.0001	55,51	13,95	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,40)	3304,08	983,18	< 0.0001	72,64	8,77	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,35)	1431,15	620,37	< 0.0001	51,55	15,47	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,35)	2020,35	729,88	< 0.0001	62,65	14	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	2618	742,43	< 0.0001	70,95	10,89	< 0.0001
BII-P2P (TTL=13; $\gamma$ =0,30)	990,1	447,66	< 0.0001	47,84	16,01	0,0002
BII-P2P (TTL=13; $\gamma$ =0,35)	3159,85	666,55	< 0.0001	76,77	8,12	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,30)	1204,2	488,48	< 0.0001	53,49	15,97	< 0.0001
BII-P2P (TTL=14; $\gamma$ =0,35)	3608,78	541,54	0,0001	80,61	6,15	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,30)	1413,23	510,09	< 0.0001	58,66	15,76	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,30)	1605,65	513,95	< 0.0001	62,57	15,13	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,30)	1779,88	504,39	< 0.0001	65,62	14,45	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,30)	1928,98	488,35	< 0.0001	68,27	13,99	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,30)	2055,28	470,43	< 0.0001	70,19	13,63	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,30)	2157,75	453,89	< 0.0001	71,88	13,21	< 0.0001

Los diagramas de caja que se presentan a continuación —ver Figura 5-9 y Figura 5-10— permiten una mejor visualización de los datos descritos en la Tabla 5-6.

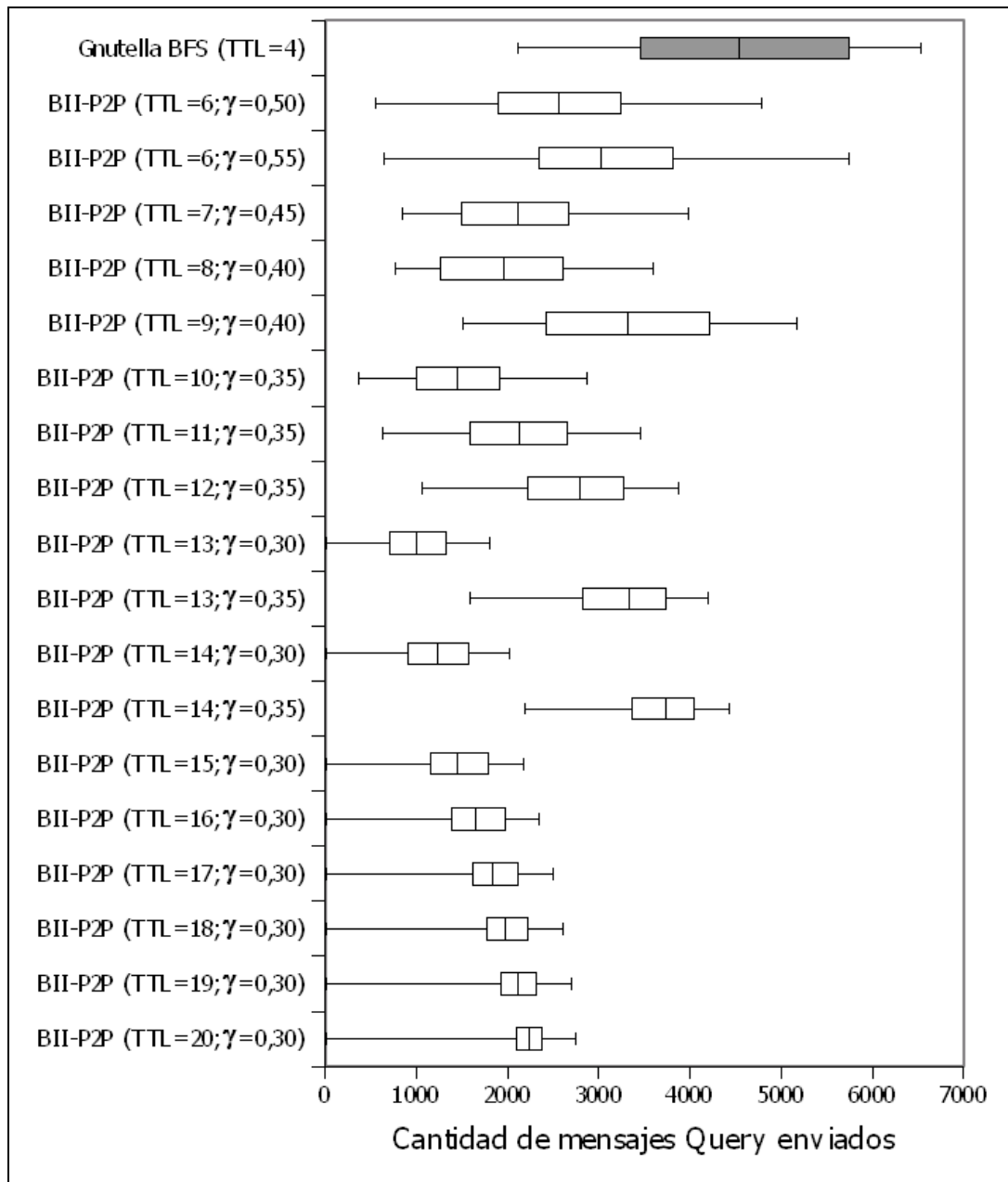


Figura 5-9 Diagrama de cajas. Cantidad de mensajes *Query* generados sobre red de 10.000 nodos



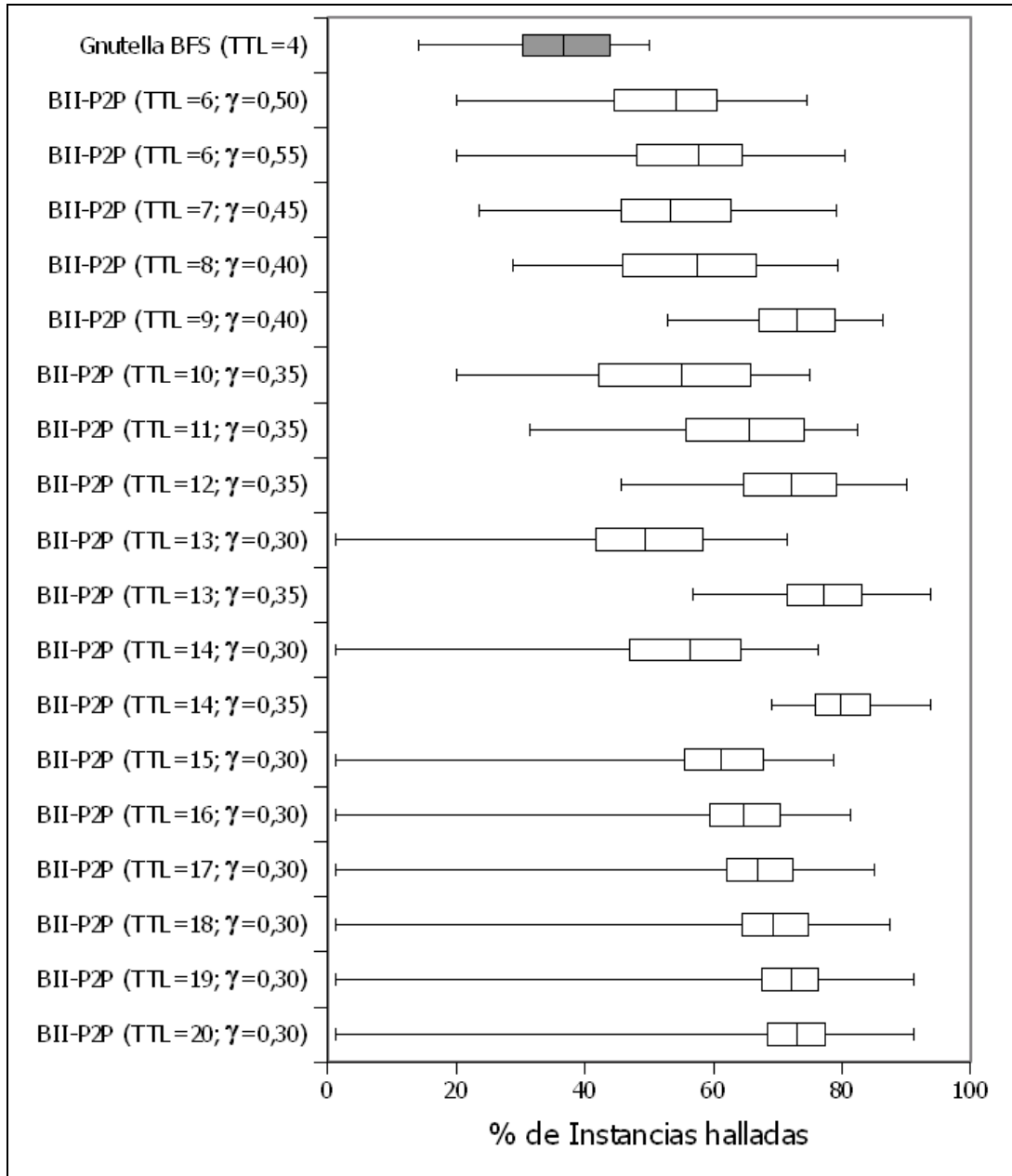


Figura 5-10 Diagrama de cajas. % de Instancias halladas del recurso buscado sobre red de 10.000 nodos.

De los gráficos anteriores se deduce que todos los casos BII-P2P presentados superan al algoritmo BFS de *Gnutella* con los datos agrupados alrededor de valores más convenientes y con menor dispersión para la métrica relacionada con

el tráfico generado —ver Figura 5-9—. Sin embargo también se visualizan algunos valores extremos de mala calidad que aparecen para  $TTL > 12$ .

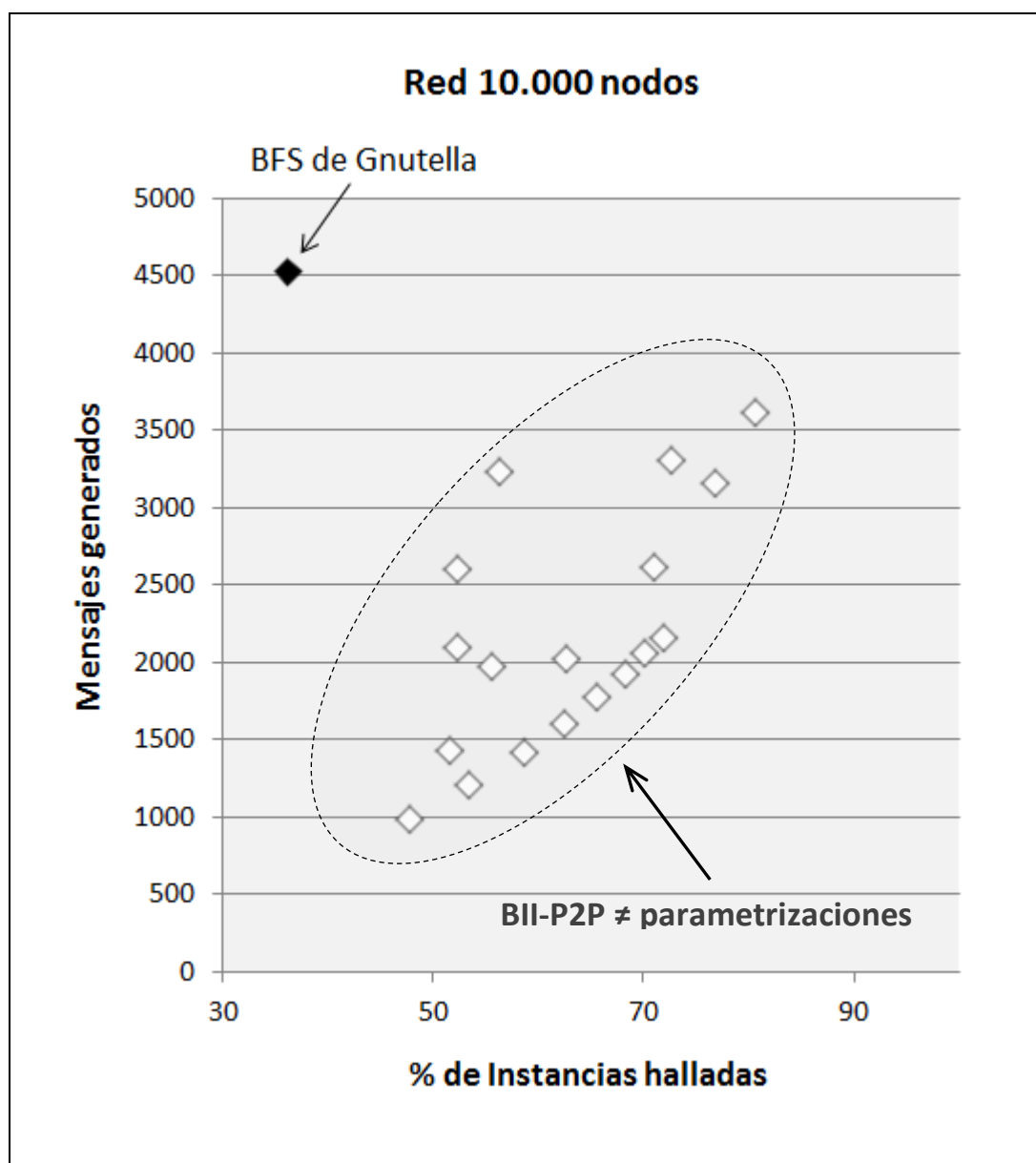


Figura 5-11 Relación entre mensajes *Query* generados y porcentaje de instancias halladas sobre red de 10.000 nodos

La Figura 5-11 muestra la relación existente entre la cantidad de mensajes *Query* generados y el porcentaje de instancias halladas por las distintas estrategias sobre la red de 10.000 nodos.

Claramente BII-P2P es preferible a BFS de Gnutella para todos los casos expuestos en la Tabla 5-6. Para puntualizar algunos ejemplos considérese el rendimiento alcanzado por BII-P2P con  $TTL=13$  y  $\gamma=0,30$  que utiliza 990,1 mensajes *Query* y encuentra el 47,84 % de las instancias existentes del recurso buscado —compárese con BFS que emplea 4.525,53 mensajes de *Query* y encuentra el 36,14% de las instancias—. Otro caso interesante se observa para BII-P2P con  $TTL=14$  y  $\gamma=0,35$  dónde se consigue el 80,61% de hallazgos de instancias con 3.608,78 mensajes *Query* generados. Ambos ejemplos se visualizan en la figura siguiente.

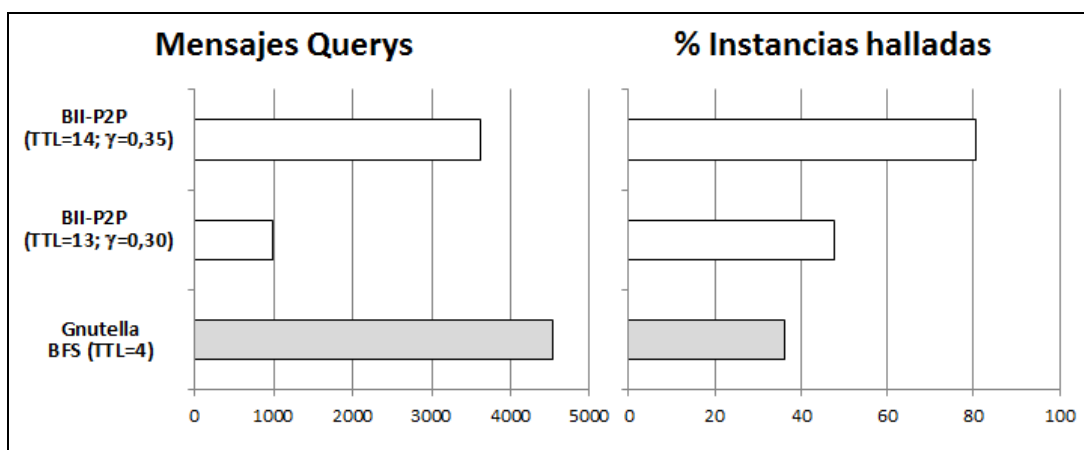


Figura 5-12 Comparativa de rendimiento sobre la red de 10.000 nodos para los casos de ejemplos puntualizados en la presente sección

La Tabla 5-7 que se presenta a continuación expone los resultados de otros casos de prueba donde BII-P2P supera a BFS en sólo una de las métricas consideradas y lo iguala en la restante. En todos estos casos las medias alcanzadas

por BII-P2P resultan mejores que las obtenidas por BFS, sin embargo las pruebas de significancia estadística —test *t-Student*— resultan positivas sólo para una de las métricas evaluadas.

**Tabla 5-7 Casos evaluados sobre la red de 10.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en sólo una de las métricas consideradas y lo iguala en la restante. En todos los casos BII-P2P el p-valor es inferior a 0,01 sólo para una de las métricas observadas.**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
BII-P2P (TTL=5; $\gamma$ =0,65)	2487,18	865,11	< 0.0001	40,05	11,66	0,1231
BII-P2P (TTL=5; $\gamma$ =0,70)	3817,75	1305,74	0,0173	49	12,27	< 0.0001
BII-P2P (TTL=8; $\gamma$ =0,45)	4170,58	1213,42	0,2101	74,93	9,39	< 0.0001
BII-P2P (TTL=9; $\gamma$ =0,35)	927,88	459,57	< 0.0001	39,42	15,21	0,2461
BII-P2P (TTL=12; $\gamma$ =0,30)	777,93	387,28	< 0.0001	40,78	15,95	0,1239
BII-P2P (TTL=15; $\gamma$ =0,35)	3953,23	411,06	0,0107	82,6	5,42	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,35)	4197,63	297,65	0,1265	83,81	4,94	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,35)	4360,13	218,42	0,431	84,41	4,69	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,35)	4463,65	173,08	0,7665	84,84	4,74	< 0.0001

A modo de ejemplo considérese el rendimiento alcanzado por BII-P2P con TTL=12 y  $\gamma$ =0,30 que si bien exhibe una pequeña diferencia a su favor respecto del porcentaje de instancias halladas —40,78% de BII-P2P contra 36,14% de BFS— esta diferencia no puede considerarse estadísticamente significativa —p-valor = 0,1239 > 0,01—; sin embargo supera notablemente a BFS al utilizar en promedio sólo 777,93 mensajes *Query* contra los 4.525,53 de BFS.

### 5.1.3.5 Rendimiento de la Búsqueda Inteligente sobre Red de 20.000 Nodos

En la presente sección se resumen los resultados más relevantes obtenidos durante la experimentación realizada sobre la red de 20.000 nodos. El conjunto completo de los datos recolectados puede consultarse en la Tabla A-4 en la sección “Apéndice A” a partir de la pág. 255.

En la Tabla 5-8 se exponen los resultados alcanzados por la búsqueda BFS de *Gnutella* con TTL=4 y por los mejores casos de búsqueda inteligente BII-P2P que superan a BFS respecto de las dos métricas consideradas —mayor porcentaje de hallazgos y menor cantidad de mensajes *Query* generados—. El p-valor del test *t-Student*, que en todos los casos es muy inferior a 0.01, permite concluir que las diferencias en las medias observadas son estadísticamente significativas.

**Tabla 5-8 Casos evaluados sobre la red de 20.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en ambas métricas consideradas. Las diferencias observadas para  $\bar{Q}$  y  $\bar{H}$  con respecto a BFS de *Gnutella* resultan estadísticamente significativa —p-valor inferior a 0,01—**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
<i>Gnutella</i> BFS (TTL=4)	4791,05	1753,6	< 0.0001	21,26	8,44	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,50)	2840,38	1028,41	< 0.0001	32,92	11,15	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,55)	3691,4	1529,69	0,0038	38	13,41	< 0.0001
BII-P2P (TTL=7; $\gamma$ =0,45)	2387,53	1110,73	< 0.0001	35,4	13,89	< 0.0001
BII-P2P (TTL=8; $\gamma$ =0,40)	2284,98	1137,95	< 0.0001	37,22	14,71	< 0.0001
BII-P2P (TTL=10; $\gamma$ =0,35)	1716,28	922,65	< 0.0001	35,38	15,62	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,35)	2654,45	1246,21	< 0.0001	48,13	17,1	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	3788,35	1479,45	0,0072	59,99	15,27	< 0.0001
BII-P2P (TTL=13; $\gamma$ =0,30)	1159,05	587,19	< 0.0001	32,37	14,28	0,0001
BII-P2P (TTL=14; $\gamma$ =0,30)	1525,53	704,86	< 0.0001	39,81	15,52	< 0.0001
BII-P2P (TTL=15; $\gamma$ =0,30)	1938,13	799,67	< 0.0001	46,27	15,35	< 0.0001
BII-P2P (TTL=16; $\gamma$ =0,30)	2375,65	854,64	< 0.0001	52,83	14,48	< 0.0001
BII-P2P (TTL=17; $\gamma$ =0,30)	2817,33	859,92	< 0.0001	57,67	13,1	< 0.0001
BII-P2P (TTL=18; $\gamma$ =0,30)	3239,4	816,17	< 0.0001	61,97	10,89	< 0.0001
BII-P2P (TTL=19; $\gamma$ =0,30)	3623,45	731,89	0,0003	65,59	9,09	< 0.0001
BII-P2P (TTL=20; $\gamma$ =0,30)	3954,7	630,23	0,0066	68,19	7,9	< 0.0001

Los diagramas de caja que se presentan a continuación —ver Figura 5-13 y Figura 5-14— permiten una visualización más completa sobre los datos que dieron origen a la tabla anterior. De su observación directa se concluye que todos los casos BII-P2P presentados superan ampliamente al algoritmo BFS de *Gnutella* con los datos agrupados alrededor de valores más convenientes y, en el caso de la Figura 5-13, casi siempre con menor dispersión que BFS

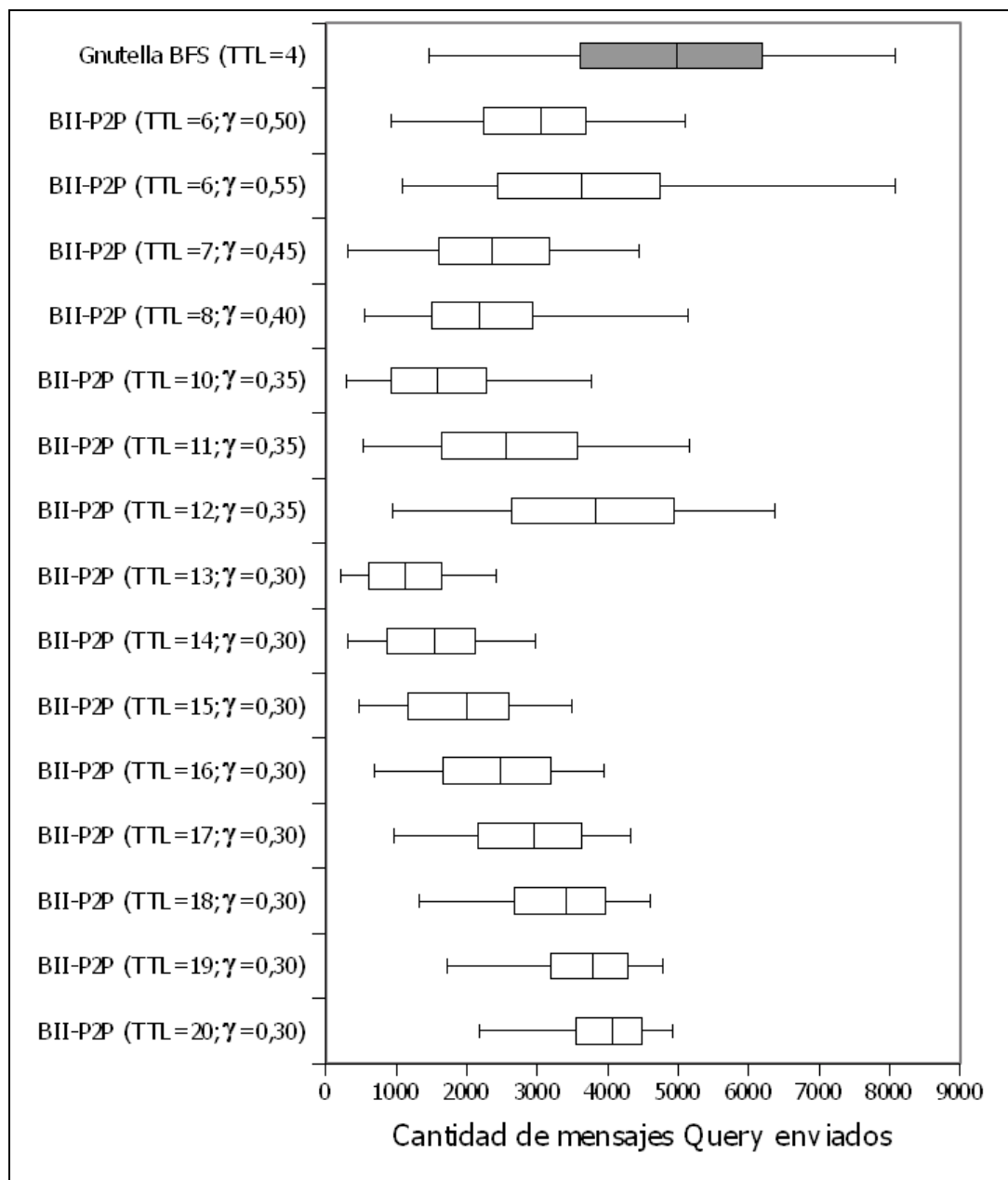


Figura 5-13 Diagrama de cajas. Cantidad de mensajes *Query* generados sobre red de 20.000 nodos

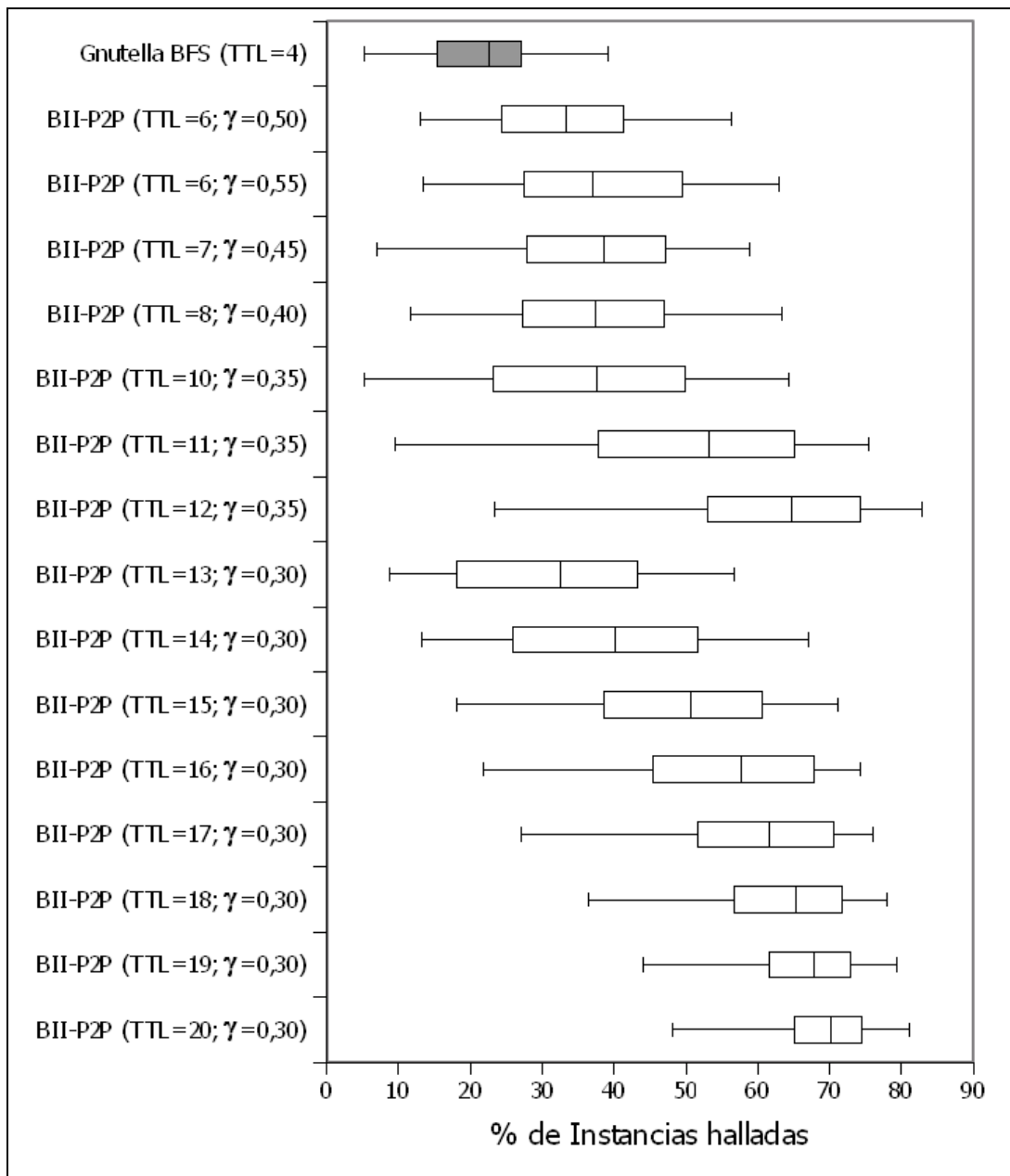


Figura 5-14 Diagrama de cajas. % de Instancias halladas del recurso buscado sobre red de 20.000 nodos.

La Figura 5-15 muestra la relación existente entre la cantidad de mensajes *Query* generados y el porcentaje de instancias halladas por las distintas estrategias

sobre la red de 20.000 nodos.

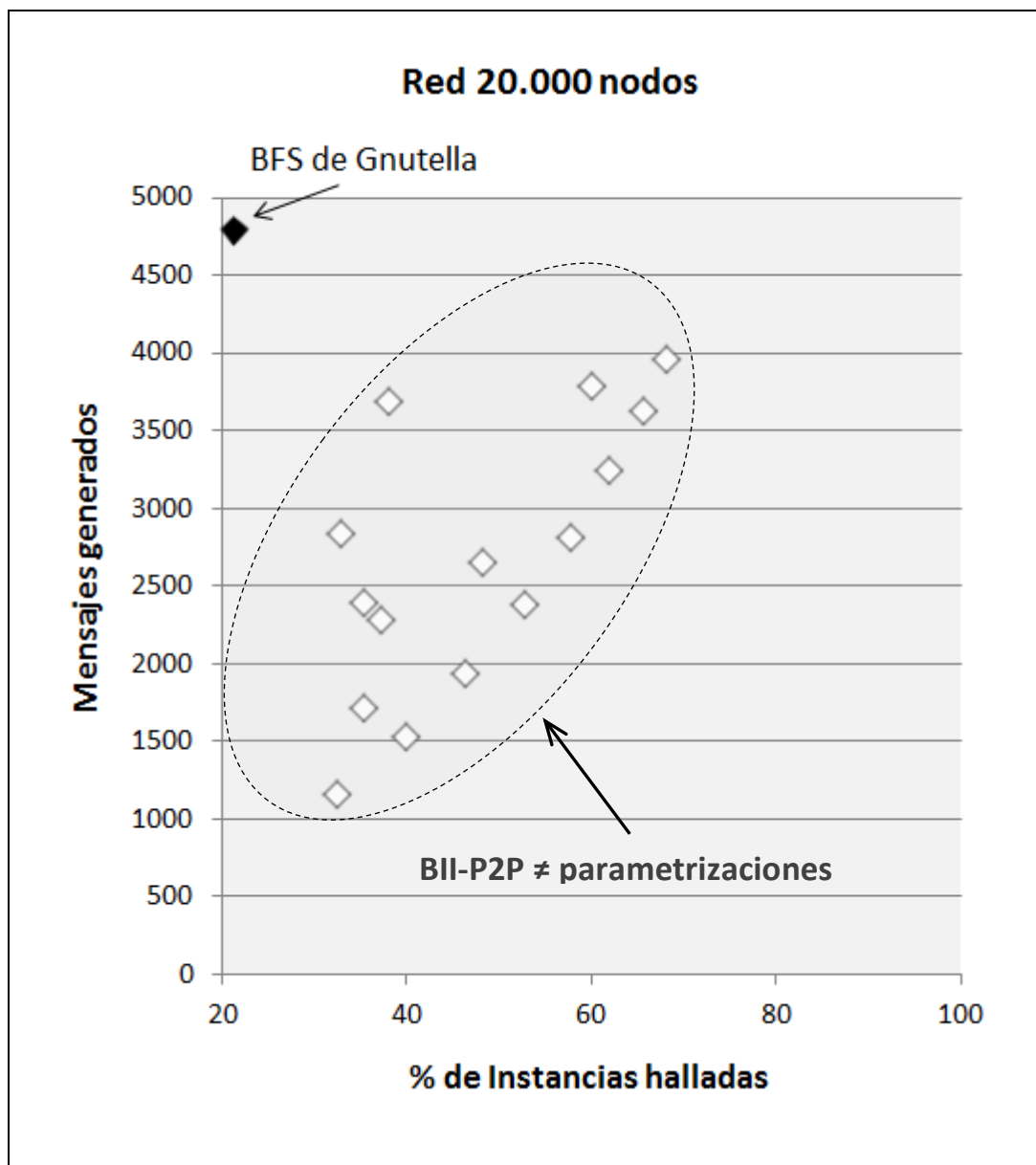


Figura 5-15 Relación entre mensajes *Query* generados y porcentaje de instancias halladas sobre red de 20.000 nodos

Claramente BII-P2P es preferible a BFS de Gnutella para todos los casos



expuestos en la Tabla 5-8. Para puntualizar algunos ejemplos considérese el rendimiento alcanzado por BII-P2P con  $TTL=13$  y  $\gamma=0,30$  que utiliza 1.159,05 mensajes *Query* y encuentra el 32,37 % de las instancias del recurso buscado —compárese con BFS que emplea 4.791,05 mensajes de *Query* y encuentra el 21,26 % de las instancias—. Otro caso interesante se observa para BII-P2P con  $TTL=20$  y  $\gamma=0,30$  consiguiendo el 68,19 % de hallazgos con 3.954,7 mensajes *Query* generados. Ambos ejemplos se visualizan en la figura siguiente.

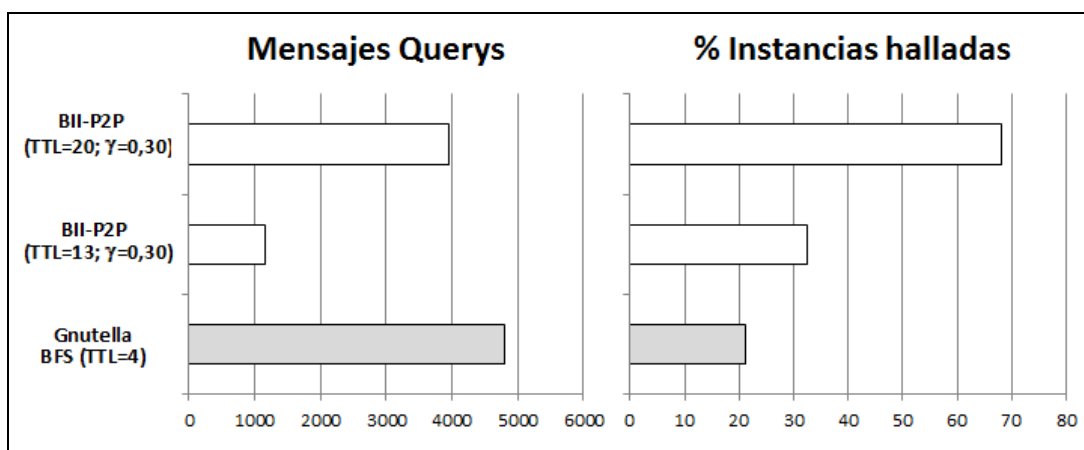


Figura 5-16 Comparativa de rendimiento sobre la red de 20.000 nodos para los casos de ejemplos puntualizados en la presente sección

**Tabla 5-9 Casos evaluados sobre la red de 20.000 nodos en los que BII-P2P supera al algoritmo BFS de *Gnutella* en sólo una de las métricas consideradas y lo iguala en la restante. En todos los casos BII-P2P el p-valor es inferior a 0,01 sólo para una de las métricas observadas.**

Método de búsqueda	Querys generadas			% Instancias halladas		
	$\bar{Q}$	$S_Q$	p-valor	$\bar{H}$	$S_H$	p-valor
BII-P2P (TTL=5; $\gamma=0,65$ )	2657,43	1130,3	< 0.0001	23,82	9,11	0,219
BII-P2P (TTL=5; $\gamma=0,70$ )	4072,93	1741,84	0,0699	30,41	11,7	0,0002
BII-P2P (TTL=9; $\gamma=0,35$ )	1032,15	610,3	< 0.0001	23,97	12,18	0,2846
BII-P2P (TTL=9; $\gamma=0,40$ )	4297,08	1831,94	0,2217	56,17	16,2	< 0.0001
BII-P2P (TTL=12; $\gamma=0,30$ )	851,63	464,87	< 0.0001	25,32	12,46	0,1172

La tabla anterior expone los resultados de otros casos de prueba donde BII-P2P supera a BFS en sólo una de las métricas consideradas y lo iguala en la restante. En todos estos casos las medias alcanzadas por BII-P2P resultan mejores que las obtenidas por BFS, sin embargo las pruebas de significancia estadística —test *t-Student*— resultan positivas sólo para una de las métricas evaluadas.

A modo de ejemplo considérese el rendimiento alcanzado por BII-P2P con  $TTL=12$  y  $\gamma=0,30$  que si bien exhibe una pequeña diferencia a su favor respecto del porcentaje de instancias halladas —25,32% de BII-P2P contra 21,26% de BFS— esta diferencia no puede considerarse estadísticamente significativa — $p\text{-valor} = 0,1172 > 0,01$ —; sin embargo supera notablemente a BFS al utilizar en promedio sólo 851,63 mensajes *Query* contra los 4.791,05 de BFS.

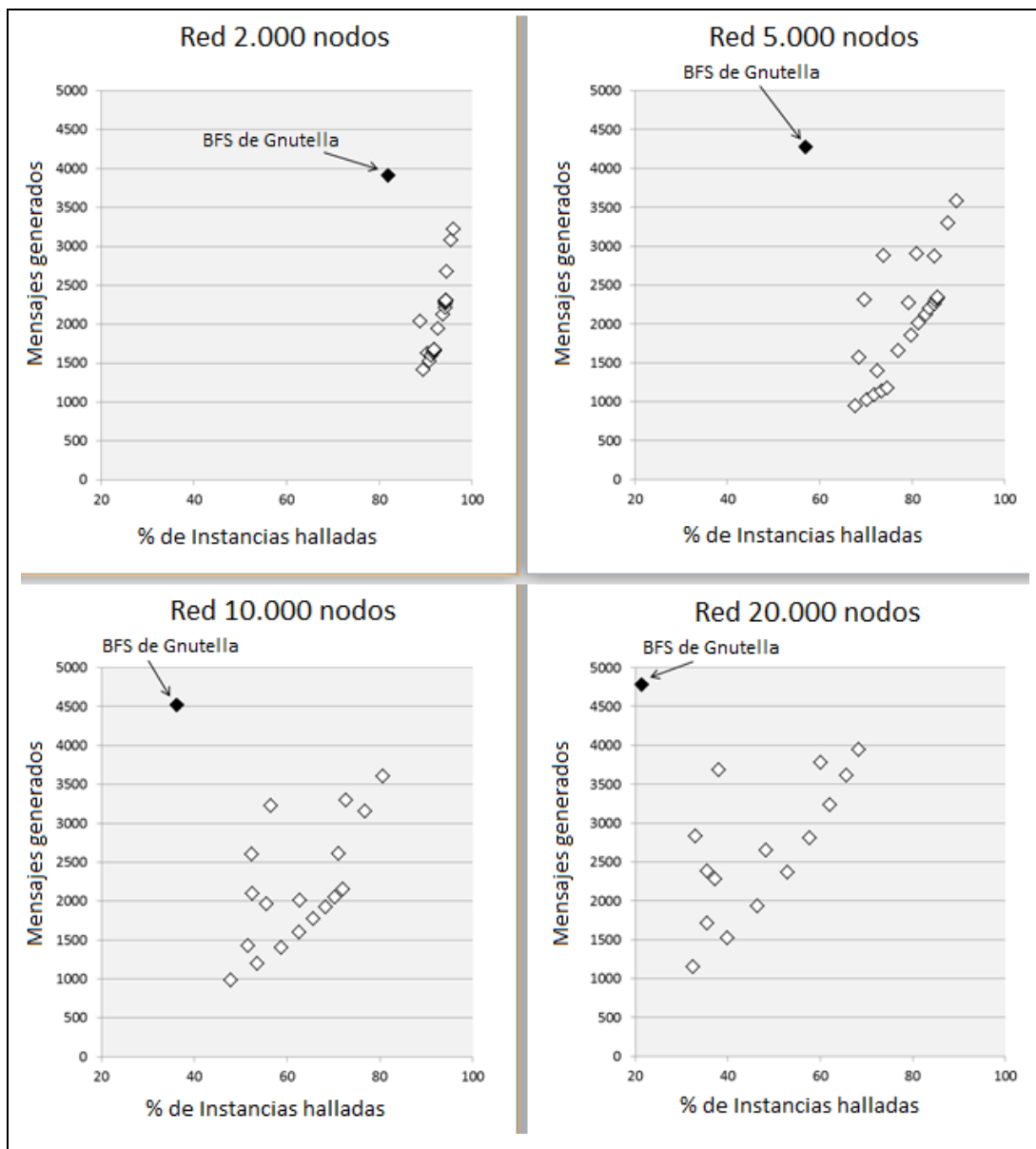
#### 5.1.3.6 El Tamaño de la Red y su Impacto en el Rendimiento de la Búsqueda

En la Figura 5-17 se presentan juntos a igual escala los gráficos de rendimiento sobre los cuatro tamaños de redes estudiadas para los casos BII-P2P más favorables. La visualización comparativa permite detectar si el desempeño de las metodologías de búsqueda se ve afectado de alguna forma por el tamaño de la red en las que fueron evaluadas.

Se aprecia con claridad que, a excepción de la topología de 2.000 nodos, el tamaño de la red no influye significativamente sobre el tráfico generado para ninguna de las estrategias graficadas. Se observa también que el nivel de mejora de rendimiento de BII-P2P con relación a BFS de *Gnutella*, representado en el gráfico por las distancias entre los puntos BII-P2P y el punto BFS, es muy similar en las topologías de 5.000, 10.000 y 20.000 nodos. Sin embargo esta diferencia a favor de BII-P2P disminuye para la red de 2.000 nodos en la que BFS obtiene un buen resultado siendo el único caso en el que el algoritmo de *Gnutella* genera un tráfico inferior a 4.000 mensajes de *Query* por búsqueda.

Por otro lado, el tamaño de la red afecta por igual la capacidad de búsqueda

del algoritmo BFS de *Gnutella* y de todos los casos BII-P2P probados. El porcentaje de instancias halladas por recurso decrece para todas las metodologías conforme se consideran las topologías con mayor cantidad de nodos.



**Figura 5-17** Visualización conjunta para los cuatro tamaños de red evaluados de la relación existente entre mensajes generados y porcentaje de instancias halladas para los casos BII-P2P más favorables

La Figura 5-17, en la que se dibujan sólo los casos BII-P2P dónde la búsqueda inteligente supera a BFS en ambas métricas consideradas, muestra claramente que, conforme aumenta el tamaño de las redes sobre las cuales se realizaron los experimentos, disminuye la cantidad de casos BII-P2P doblemente favorables. Se ha detectado que para redes más grande BII-P2P se hace más sensible a los valores de los parámetros TTL y  $\gamma$ . Experimentos adicionales han mostrado que, para un mismo valor de TTL, si BII-P2P(TTL= $ttl$ ;  $\gamma=\gamma_1$ ) y BII-P2P(TTL= $ttl$ ;  $\gamma=\gamma_2$ ), con  $\gamma_1 < \gamma_2$ , resultan dos casos favorables superando en ambas métricas consideradas a BFS, entonces para los valores intermedios  $\gamma_i$ , con  $\gamma_1 < \gamma_i < \gamma_2$ , los casos BII-P2P(TTL= $ttl$ ;  $\gamma=\gamma_i$ ) también resultan casos favorables. Por lo tanto es posible encontrar una gran cantidad de casos favorables aún en las redes más grandes —recuérdese que  $\gamma$  es un número real—, sin embargo hallarlos es más dificultoso pues el intervalo favorable para  $\gamma$  disminuye a medida que crece el tamaño de la red.

A modo de ejemplo considérese el rendimiento de BII-P2P para TTL=12. En el caso de la red con 2.000 nodos, la experimentación realizada permite fijar el intervalo favorable para  $\gamma$  en [0,35;0,50] —longitud del intervalo igual a 0,15—; mientras que para la red de 20.000 nodos el intervalo favorable queda establecido en [0,30;0,35] —longitud del intervalo igual a 0,05—.

Este comportamiento se aprecia mejor observando las tablas A-5, A-6, A-7 y A-8, para los tamaños de redes 2.000, 5.000, 10.000 y 20.000 nodos respectivamente, que se presentan en la sección “Apéndice A” a partir de la página 259. En estas tablas se visualizan los datos experimentales recogidos durante las pruebas realizadas ya presentados en las tablas anteriores en la misma sección pero dispuestos de manera diferente. Cada fila se corresponde con un valor de TTL y cada columna con un valor de  $\gamma$ . De esta forma, cada celda ( $i,j$ ) —intersección de la fila  $i$  con la columna  $j$ — representa el caso de prueba para BII-P2P(TTL= $i$ ;  $\gamma=j$ ). En cada una de estas celdas se describen los valores

obtenidos para ambas métricas consideradas es decir, la cantidad de mensajes *Query* generados y el porcentaje de instancias hallado del total existente del recurso buscado.

Además, las celdas de las tablas A-5, A-6, A-7 y A-8 correspondientes a los casos donde BII-P2P supera a BFS en ambas métricas consideradas, han sido sombreadas para visualizar su distribución sobre las tablas. Así se delimitan las siluetas de los mejores casos BII-P2P que también se pueden apreciar en la Figura 5-18.

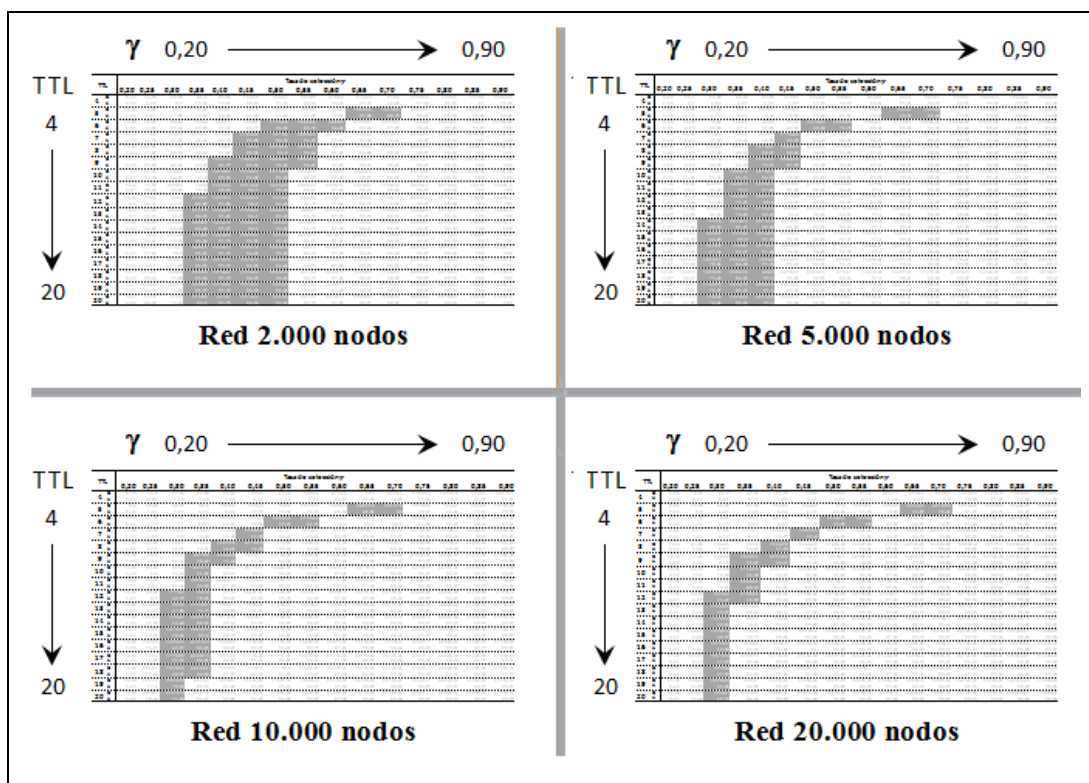


Figura 5-18 Miniatura de las tablas A-5, A-6, A-7 y A-8 en las que se aprecia la distribución de los mejores casos BII-P2P en forma de silueta sombreada sobre cada una de las tablas.

En la Figura 5-18 se observa claramente cómo las siluetas de los mejores

casos BII-P2P se hacen más delgadas a medida que aumenta el tamaño de la red, ocupando menor cantidad de columnas —valores para el parámetro  $\gamma$ —, especialmente en las filas correspondientes a los valores de TTL más altos. Esto indica que para las redes más grandes la búsqueda inteligente BII-P2P se hace más sensible al parámetro  $\gamma$ , cuyo rango óptimo de valores se estrecha requiriendo mayor precisión para su correcta elección. Por lo tanto,  $\gamma$  se convierte en una variable de ajuste importante para el rendimiento del sistema de búsqueda BII-P2P que, por tratarse de una variable de tipo real permite un ajuste más fino que el ofrecido por el parámetro TTL el cuál admite sólo valores naturales.

De lo anterior se desprende que la determinación *a priori* de los mejores valores para los parámetros TTL y  $\gamma$  exige conocer ciertas características de la topología de la red sobre la que se utilizará BII-P2P. Esto representa una dificultad ya que generalmente esta información no resulta fácilmente accesible en los sistemas P2P totalmente descentralizados dónde BII-P2P es aplicable.

Sólo a modo ilustrativo considérese una posible solución a la problemática expuesta precedentemente basada en la utilización de parámetros TTL y  $\gamma$  locales, es decir definidos individualmente y manipulados en cada nodo de la red. En esta solución, cada nodo que inicia una solicitud de búsqueda determina, de acuerdo a su experiencia pasada, el valor más adecuado para los parámetros TTL y  $\gamma$  para esa búsqueda en particular. Los demás nodos involucrados en la búsqueda deben respetar el valor de  $\gamma$  elegido por el nodo iniciador. Obsérvese que el mensaje *Query* ha de poseer un campo adicional para mantener el valor de  $\gamma$  indicado. Como primera aproximación, considérese la facultad de un nodo para iniciar una búsqueda intentando minimizar el tráfico sobre la red, proponiendo una configuración de valores pequeños para los parámetros TTL y  $\gamma$ . Sólo en caso de no obtener los resultados esperados, el nodo ha de repetir la búsqueda con valores más grandes aumentando así la probabilidad de hallazgos.

La estrategia esbozada anteriormente describe una posible solución que no ha

sido implementada en este trabajo. Se expone aquí con la intención de ilustrar cómo la búsqueda BII-P2P abre un nuevo abanico de posibilidades que merecen ser exploradas. Para resolver la misma problemática, en esta tesis se utilizó un enfoque algo más elaborado que se presenta en el Capítulo 6 “Exploración Incremental en BII-P2P” en la página 193.

La capacidad de realizar búsquedas incrementales con la que se ha dotado a BII-P2P complementa a la búsqueda inteligente permitiendo que, bajo ciertas circunstancias, la incertidumbre sobre las características topológicas de la red no impidan la implementación de un sistema de búsqueda inteligente con excelentes resultados.

#### 5.1.4 Conclusiones sobre la Capacidad de Búsqueda

En las secciones anteriores se han presentado y comentado los datos recogidos en las experimentaciones realizadas para evaluar la capacidad de búsqueda de la componente inteligente de BII-P2P —la componente incremental se introduce en el Capítulo 6—. De acuerdo con los resultados obtenidos se concluye que BII-P2P supera significativamente al algoritmo BFS de *Gnutella*, consiguiendo mayor porcentaje de hallazgos y reduciendo considerablemente el tráfico generado sobre la red.

La determinación *a priori* de los valores de los parámetros TTL y  $\gamma$  más convenientes para maximizar el rendimiento del sistema de búsqueda puede ser dificultosa. Estos valores están ligados a la topología de la red —cantidad de nodos y grado de conexión—. Sin embargo en la sección 5.1.3 —página 142— se han presentado y analizado una gran cantidad de combinaciones de valores para estos parámetros, todas ellas indiscutiblemente ventajosas para los cuatro tamaños distintos de redes considerados.

Una vez establecido que la búsqueda inteligente que implementa BII-P2P, parametrizada adecuadamente, posee un alto grado de rendimiento, el desafío

consiste en encontrar reglas o sugerencias que permitan definir los valores más convenientes para TTL y  $\gamma$ . Para superar esta dificultad es necesario desarrollar mecanismos de ajustes dinámicos que permitan al sistema autorregularse modificando estos u otros parámetros en función del rendimiento percibido y de las necesidades del momento. La exploración incremental que se aborda en el Capítulo 6 puede considerarse uno de estos mecanismos que trabaja sobre el parámetro  $\gamma$  para alcanzar el éxito de la búsqueda con la menor cantidad posible de mensajes *Query* generados sobre la red.

## 5.2 Evaluación de la Capacidad de Adaptación

En las siguientes secciones se presentan la experimentación realizada, el análisis de los datos recolectados y las conclusiones sobre la capacidad de adaptación de BII-P2P. Esta capacidad está relacionada con la habilidad de los nodos para adquirir conocimiento de su entorno. Se ha recurrido aquí también a la simulación para medir el rendimiento de BII-P2P en relación a las distintas políticas de adquisición y actualización de conocimiento propuestas.

### 5.2.1 Experimentación

Para analizar la capacidad de adaptación de BII-P2P es necesario medir el rendimiento de la búsqueda inteligente bajo las políticas de adquisición de conocimiento Simple, DLP y DLPA definidas en la sección 4.2.3 —ver “Políticas de Adquisición y Actualización de Conocimiento” en la página 123—.

Para cada caso de prueba, la experimentación consistió en la realización de 40 ensayos de búsqueda independientes a partir de una configuración o descripción instantánea de un sistema P2P obtenida por medio de un proceso de dos etapas:

- i) *Generación aleatoria de topología de red y distribución de recursos compartidos.* La forma de generar la topología, definir los perfiles de usuarios y distribuir los recursos compartidos es la misma descrita en



la sección 5.1.1 utilizada para evaluar la capacidad de búsqueda de BII-P2P. De igual manera, se estableció el valor del parámetro  $\sigma$  de la ecuación (4-5) en 0,85. Luego de unos ensayos preliminares que demostraron que el tamaño de la red no afecta significativamente los resultados, se decidió utilizar la misma topología para todos los casos fijándose el tamaño de la misma en 3.000 nodos

- ii) *Adquisición de conocimiento conforme la política a ser evaluada.* Medir con precisión el impacto del parámetro  $i$  en las políticas  $DLP_i$  o  $DLPA_{(i, j)}$  es dificultoso debido a que el alcance de la información transportada en los mensajes *NeuralInf* tiende a superar el valor de este parámetro —ver sección 4.2.3.4 “Alcance Real de la Información de Actualización en DLP y DLPA” en la página 125—. Además, este alcance puede variar en distintas regiones de la red P2P dependiendo de las condiciones particulares determinadas por qué nodos y con qué frecuencia propagan información de actualización. Para medir la magnitud del impacto del parámetro  $i$  en el rendimiento de la búsqueda inteligente se utilizó la estrategia del peor caso impidiendo que la información propagada por los nodos sobrepase el límite  $i$  establecido en todas las regiones de la red. Ello se consiguió simplemente haciendo que todos los nodos propaguen a la vez sus mensajes *NeuralInf* y actualicen sus respectivas redes neuronales antes del próximo ciclo de propagación. Estos ciclos de propagación se realizaron exactamente  $i$  veces para finalizar la etapa de adquisición de conocimiento previa a la experimentación. Así se obtuvieron las descripciones instantáneas del sistema P2P sobre las que se ejecutaron los ensayos de búsqueda para cada caso de prueba definido.

### 5.2.2 Construcción de los Casos de Prueba

Para medir el rendimiento de la búsqueda BII-P2P afectada por las distintas

políticas de adquisición de conocimiento, además del caso de referencia comparativa —algoritmo BFS de *Gnutella*—, se han definido 1848 casos de pruebas distintos. Se consideraron 14 políticas de adaptación distintas: Simple, DLP<sub>1</sub>, DLP<sub>2</sub>, DLP<sub>3</sub>, DLP<sub>4</sub>, DLP<sub>5</sub>, DLP<sub>6</sub>, DLP<sub>7</sub>, DLP<sub>8</sub>, DLPA<sub>(1;75)</sub>, DLPA<sub>(2;75)</sub>, DLPA<sub>(3;75)</sub>, DLPA<sub>(4;75)</sub> y DLPA<sub>(5;75)</sub>. Por cada una de las 14 políticas consideradas se definieron 132 casos de prueba constituidos por distintas parametrizaciones de la búsqueda inteligente BII-P2P surgidas de la combinación de 11 valores distintos de la tasa de selección  $\gamma$  (0,3; 0,35; 0,4; 0,45; 0,5; 0,55; 0,6; 0,65; 0,7; 0,75 y 0,8) con 12 valores de TTL (4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14 y 15).

Además, para cada una de las políticas consideradas se ha registrado la carga de tráfico extra generada por la difusión de mensajes *NeuralInf* permitiendo una valoración más precisa de la utilidad de cada una de ellas.

### 5.2.3 Resultados de la Experimentación

Los promedios de mensajes *Query* enviados y porcentaje de instancias de recursos hallados, calculados entre los 40 ensayos de búsqueda independientes realizados para cada caso de prueba, constituyen una buena caracterización del rendimiento de BII-P2P que permite visualizar cómo influye la política de adquisición y actualización de conocimiento elegida sobre la capacidad de búsqueda del sistema.

Los datos observados en los 132 casos de prueba definidos para cada una de las 14 políticas consideradas se recopilaron en las tablas A-9, A-10, ..., A-22 que, por razones de visualización, se presentan a partir de la página 263 dentro de la sección titulada “Apéndice A”. En todas estas tablas se han sombreado las celdas que corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para la métrica correspondiente a la columna que ocupan.

Para una lectura comparativa, la Tabla 5-10 resume los datos correspondientes a las 14 políticas consideradas para el caso de la búsqueda

BII-P2P con parámetros  $TTL=10$  y  $\gamma=0,40$ . Para estos valores de  $TTL$  y  $\gamma$ , la búsqueda inteligente supera al algoritmo BFS de *Gnutella* en ambas métricas registradas, cualquiera sea la política de adaptación elegida. Existen otros casos donde se repite esta situación que pueden hallarse fácilmente a través de la lectura de las tablas presentadas en la sección “Apéndice A”.

Además de los datos referentes al rendimiento de la búsqueda BII-P2P que lo conforman la cantidad promedio de mensajes *Query* generados y el porcentaje promedio de instancias halladas por búsqueda, la Tabla 5-10 se completó con información referente a la carga de tráfico extra generada por la política de adaptación elegida. Esta carga se cuantifica por el promedio de mensajes *NeuralInf* que se generan cada vez que un nodo propaga información de actualización.

**Tabla 5-10 Políticas de actualización y su impacto sobre el tráfico de red. Datos correspondientes al caso de prueba BII-P2P( $TTL=10$ ;  $\gamma=0,40$ )**

Política de Adaptación		Rendimiento de la Búsqueda	
Política	<i>NeuralInf</i> enviados	<i>Query</i> enviados	Instancias halladas
DLPA <sub>(1;75)</sub>	6,23	3141,65	88,21 %
DLP <sub>1</sub>	8,75	2077,30	79,86 %
DLPA <sub>(2;75)</sub>	42,08	2794,93	87,95 %
DLP <sub>2</sub>	77,45	1931,33	82,87 %
DLPA <sub>(3;75)</sub>	240,70	2692,53	89,24 %
DLP <sub>3</sub>	618,65	1971,68	85,71 %
DLPA <sub>(4;75)</sub>	1262,75	2687,10	88,90 %
DLP <sub>4</sub>	4372,65	1975,70	87,09 %
DLPA <sub>(5;75)</sub>	5118,15	2674,48	89,67 %
DLP <sub>5</sub>	16708,08	1975,95	86,83 %
DLP <sub>6</sub>	21227,73	1977,98	87,18 %
DLP <sub>7</sub>	21273,00	1974,85	86,83 %
DLP <sub>8</sub>	21273,00	1973,78	86,75 %
SIMPLE	21273,00	1974,00	86,75 %

Referencia comparativa: Algoritmo BFS de *Gnutella* para  $TTL=4$   
*Query* enviados=4.372,65 Instancias halladas=72,80%

Obsérvese que en todos los casos mostrados en la Tabla 5-10 BII-P2P supera al algoritmo BFS de *Gnutella* hallando un porcentaje mayor de instancias del recurso buscado. Sin embargo, en relación al tráfico generado es necesario considerar cuidadosamente la sobrecarga que produce la política de adaptación.

Según se observa en la Tabla 5-10, la política Simple es desaconsejable debido a la gran cantidad de tráfico que genera la difusión de los mensajes *NeuralInf*: 21.273 mensajes. Este número representa la cantidad de enlaces total existente en la topología de la red P2P generada aleatoriamente. Obsérvese que la misma situación ocurre con las políticas DLP<sub>7</sub> y DLP<sub>8</sub> que, debido al tamaño de la red P2P se comportan como la política Simple transmitiendo los mensajes *NeuralInf* por absolutamente todos los enlaces de la red cada vez que un nodo inicia una difusión de actualización.

Para proponer el límite considerado aceptable para la sobrecarga de tráfico generada por las difusiones de actualización es necesario conocer la frecuencia con que los nodos inician solicitudes de búsqueda y propagan información de actualización —ver sección 4.2.3.5 “Determinación de la Frecuencia de Actualización” en pág. 127—. Esto es difícil determinarlo *a priori*, dependerá de los patrones de uso de la red P2P.

También es necesario considerar el rendimiento de la búsqueda BFS de *Gnutella* que se pretende superar. De esta forma es posible elegir la política de actualización de conocimiento más adecuada que garantice el mayor porcentaje de hallazgos conservando siempre la ventaja requerida respecto del tráfico generado por la búsqueda BFS.

A modo ilustrativo considérese el siguiente ejemplo: Supóngase que los nodos realizan una propagación de actualización cada cuatro solicitudes de búsqueda iniciadas por ellos mismos, el tráfico total generado por búsqueda se calcula como  $(4Q+N)/4$ , donde  $Q$  es el número de mensajes *Query* generado por

búsqueda y  $N$  el número de mensajes *NeuralInf* generado por propagación de actualización. La Tabla 5-11 muestra los resultados para este hipotético caso calculados en base a los datos obtenidos por la experimentación realizada para la estrategia BII-P2P(TTL=10; $\gamma=0,40$ ).

**Tabla 5-11 Políticas de actualización y su impacto sobre el tráfico de red. Tráfico total por búsqueda calculado para el caso en que los nodos propaguen una difusión de actualización cada 4 solicitudes de búsqueda. Datos correspondientes al caso de prueba BII-P2P(TTL=10; $\gamma=0,40$ )**

Política	$N$	$Q$	Tráfico Total ( $4Q+N$ )/4	Instancias Halladas
DLPA <sub>(1;75)</sub>	6,23	3141,65	3143,21	88,21 %
DLP <sub>1</sub>	8,75	2077,30	2079,49	79,86 %
DLPA <sub>(2;75)</sub>	42,08	2794,93	2805,44	87,95 %
DLP <sub>2</sub>	77,45	1931,33	1950,69	82,87 %
DLPA <sub>(3;75)</sub>	240,70	2692,53	2752,70	89,24 %
DLP <sub>3</sub>	618,65	1971,68	2126,34	85,71 %
DLPA <sub>(4;75)</sub>	1262,75	2687,10	3002,79	88,90 %
DLP <sub>4</sub>	4372,65	1975,70	3068,86	87,09 %
DLPA <sub>(5;75)</sub>	5118,15	2674,48	3954,01	89,67 %
DLP <sub>5</sub>	16708,08	1975,95	6152,97	86,83 %
DLP <sub>6</sub>	21227,73	1977,98	7284,91	87,18 %
DLP <sub>7</sub>	21273,00	1974,85	7293,10	86,83 %
DLP <sub>8</sub>	21273,00	1973,78	7292,03	86,75 %
SIMPLE	21273,00	1974,00	7292,25	86,75 %

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4  
*Query* enviados=4.372,65 Instancias halladas=72,80%

Observando la columna “Tráfico Total” de la Tabla 5-11 se pone en evidencia que, bajo la hipótesis mencionada respecto de la frecuencia de actualización del sistema, las políticas de actualización Simple, DLP<sub>5</sub>, DLP<sub>6</sub>, DLP<sub>7</sub> y DLP<sub>8</sub>, no son elegibles si se pretende mejorar el tráfico que genera el algoritmo BFS de *Gnutella* con TTL=4 —4.372,65 mensajes *Query* por búsqueda—. Por el contrario, todas las otras políticas contempladas en la tabla representan casos favorables.

La Tabla 5-12 muestra el cálculo de tráfico total generado por búsqueda para distintas frecuencias de actualización de conocimiento. Obsérvese que si la frecuencia de actualización del sistema es demasiado alta, por ejemplo para el caso en que los nodos comiencen una propagación de actualización por cada solicitud de búsqueda iniciada por ellos mismos, la política  $DPL_4$  con un total de 6.348,35 mensajes, se convierte en una política no favorable. Si por el contrario la propagación de mensajes de actualización es menos frecuente, por ejemplo una cada 9 solicitudes de búsqueda, todas las políticas resultan favorables, incluso la política Simple que genera un promedio total de mensajes igual a 4.337,67.

**Tabla 5-12 Políticas de actualización y su impacto sobre el tráfico de red. Tráfico total por búsqueda para distintas frecuencias de actualización de conocimiento. Datos correspondientes al caso BII-P2P(TTL=10; $\gamma=0,40$ )**

Política	Frecuencia de actualización: 1 propagación de actualización cada $q$ solicitudes de búsqueda				
	$q=1$	$q=2$	$q=4$	$q=7$	$q=9$
$DLPA_{(1;75)}$	3147,88	3144,77	3143,21	3142,54	3142,34
$DLP_1$	2086,05	2081,68	2079,49	2078,55	2078,27
$DLPA_{(2;75)}$	2837,01	2815,97	2805,45	2800,94	2799,61
$DLP_2$	2008,78	1970,06	1950,69	1942,39	1939,94
$DLPA_{(3;75)}$	2933,23	2812,88	2752,71	2726,92	2719,27
$DLP_3$	2590,33	2281,01	2126,34	2060,06	2040,42
$DLPA_{(4;75)}$	3949,85	3318,48	3002,79	2867,49	2827,41
$DLP_4$	6348,35	4162,03	3068,86	2600,36	2461,55
$DLPA_{(5;75)}$	7792,63	5233,56	3954,02	3405,64	3243,16
$DLP_5$	18684,03	10329,99	6152,97	4362,82	3832,40
$DLP_6$	23205,71	12591,85	7284,91	5010,51	4336,62
$DLP_7$	23247,85	12611,35	7293,10	5013,85	4338,52
$DLP_8$	23246,78	12610,28	7292,03	5012,78	4337,45
SIMPLE	23247,00	12610,50	7292,25	5013,00	4337,67

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4  
 Query enviados=4.372,65 Instancias halladas=72,80%

Independientemente de cuál sea la frecuencia real de actualización del sistema, está claro que las políticas de adquisición y actualización de

conocimiento más convenientes son  $DLP_i$  y  $DLPA_{(i;j)}$  con un valor bajo del parámetro  $i$ . Sin embargo, la disminución del valor de  $i$  puede degradar la capacidad de búsqueda del sistema. Se pretende identificar el valor más ventajoso para el parámetro  $i$ . Para ello es necesario un análisis minucioso del impacto de las políticas de adaptación sobre el tráfico generado y la capacidad de búsqueda simultáneamente.

Se ha definido la métrica  $E$  para medir la eficiencia de los mensajes *Query*,  $E = H/Q$ , siendo  $H$  el porcentaje promedio de instancias hallado por búsqueda y  $Q$  la cantidad promedio de mensajes *Query* generado por búsqueda. Dividiendo la cantidad de hallazgos por la cantidad de mensajes utilizados se consigue una medida de eficiencia de los mensajes *Query*, al cuantificar cuál es el aporte de cada uno de ellos al hallazgo total de la búsqueda. La Tabla 5-13 muestra cómo afectan las políticas de adaptación a la eficiencia de los mensajes *Query* en el caso revisado —BII-P2P con parámetros TTL=10 y  $\gamma=0,40$ —.

**Tabla 5-13 Eficiencia de los mensajes *Query* para la búsqueda BII-P2P (TTL=10;  $\gamma=0,40$ ).  $E_{BFS}=0,0166502$  es la eficiencia alcanzada por BFS.  $E/E_{BFS}$  mide la eficiencia de los mensajes *Query* relativa a BFS para las distintas políticas de adaptación definidas**

Política	E	$E/E_{BFS}$	Política	E	$E/E_{BFS}$
BFS	0,0166502	1,0000000	DPL <sub>3</sub>	0,0434728	2,6109488
DPLA <sub>(1;75)</sub>	0,0280776	1,6863217	DPL <sub>4</sub>	0,0440812	2,6474872
DPLA <sub>(2;75)</sub>	0,0314684	1,8899708	DPL <sub>5</sub>	0,0439450	2,6393049
DPLA <sub>(3;75)</sub>	0,0331446	1,9906429	DPL <sub>6</sub>	0,0440740	2,6470552
DPLA <sub>(4;75)</sub>	0,0330834	1,9869679	DPL <sub>7</sub>	0,0439694	2,6407750
DPLA <sub>(5;75)</sub>	0,0335292	2,0137406	DPL <sub>8</sub>	0,0439498	2,6395947
DPL <sub>1</sub>	0,0384452	2,3089956	SIMPLE	0,0439448	2,6392938
DPL <sub>2</sub>	0,0429106	2,5771831			

En la Tabla 5-13 se ha incluido también la eficiencia de los mensajes *Query* para el caso BFS de *Gnutella*. Este valor, denotado por  $E_{BFS}$  ha sido utilizado para

calcular la eficiencia relativa a BFS para cada una de las políticas evaluadas y que se visualizan en la columna encabezada por:  $E/E_{BFS}$ . En la Figura 5-19 se grafican estos valores para una mejor visualización.

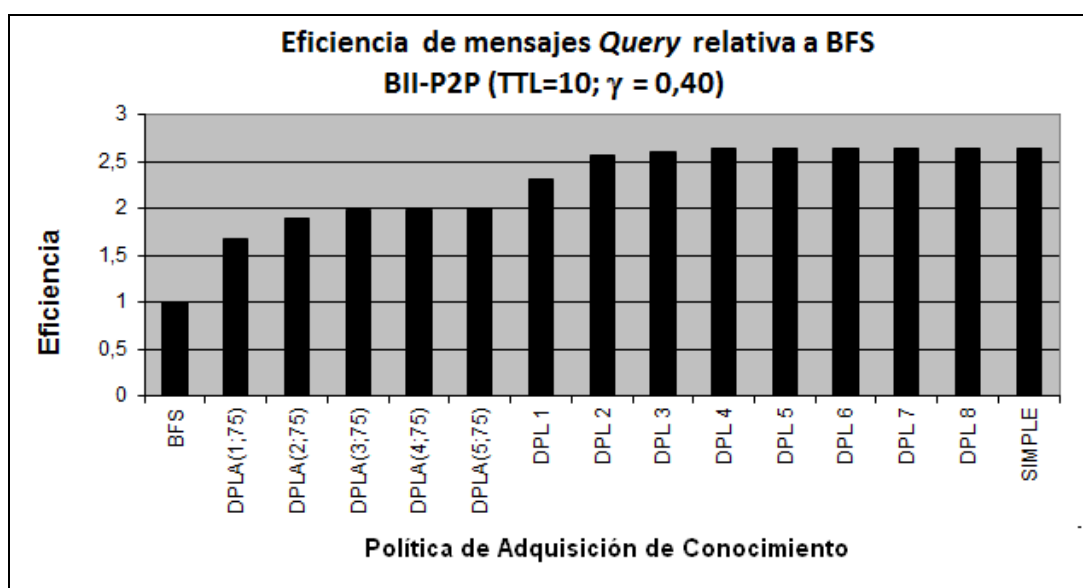


Figura 5-19 Eficiencia de los mensajes *Query* relativa a BFS en la búsqueda BII-P2P(TTL=10;  $\gamma=0,40$ ) según las distintas políticas de adaptación definidas.

La Figura 5-19 muestra claramente que las políticas DLP superan a las DLPA analizadas aprovechando mejor cada mensaje *Query* que aporta una mayor contribución al resultado final de la búsqueda. Pero también evidencia que conforme se aumenta el parámetro  $i$  de las políticas  $DLP_i$  la ganancia en eficiencia se hace cada vez menos significativa, tanto que  $DLP_3$  presenta valores muy similares a  $DLP_8$  y a la política Simple. Este resultado es altamente positivo porque establece que la propagación de los mensajes *NeuralInf* puede limitarse a los nodos más cercanos —distancia 2 o 3 a lo sumo— manteniendo sin embargo un rendimiento muy alto de la búsqueda inteligente.



Tanto la Tabla 5-13 como la Figura 5-19 se refieren al caso de la búsqueda inteligente BII-P2P con parámetros  $TTL=10$  y  $\gamma=0,40$  analizado para cada una de las políticas propuestas. Sin embargo, en las tablas A-9, A-10, ..., A-22 —presentadas a partir de la página 263 en la sección “Apéndice A”— se exponen muchos otros casos con los que se podría repetir el mismo análisis. En lugar de ello, por cada una de las políticas de actualización consideradas se ha obtenido el valor promedio de la eficiencia de los mensajes *Query* calculado entre los 132 valores conseguidos a partir de los distintos casos de prueba definidos. La Tabla 5-14 y la Figura 5-20 resumen estos resultados.

**Tabla 5-14 Eficiencia de los mensajes *Query* en la búsqueda inteligente BII-P2P promediada entre los 132 casos de prueba propuestos según las distintas políticas de adaptación definidas.**

<b>Política</b>	<b>E</b>	<b>E/E<sub>BFS</sub></b>	<b>Política</b>	<b>E</b>	<b>E/E<sub>BFS</sub></b>
BFS	0,0166502	1,0000000	DPL <sub>3</sub>	0,0457970	2,7505350
DPLA <sub>(1;75)</sub>	0,0351796	2,1128655	DPL <sub>4</sub>	0,0462903	2,7801665
DPLA <sub>(2;75)</sub>	0,0374046	2,2464955	DPL <sub>5</sub>	0,0460387	2,7650530
DPLA <sub>(3;75)</sub>	0,0389502	2,3393211	DPL <sub>6</sub>	0,0462724	2,7790891
DPLA <sub>(4;75)</sub>	0,0390217	2,3436181	DPL <sub>7</sub>	0,0460856	2,7678696
DPLA <sub>(5;75)</sub>	0,0394618	2,3700512	DPL <sub>8</sub>	0,0461140	2,7695755
DPL <sub>1</sub>	0,0418104	2,5111061	SIMPLE	0,0461018	2,7688450
DPL <sub>2</sub>	0,0454250	2,7281930			

Si se analizan los datos presentados en la Tabla 5-14 y graficados en la Figura 5-20 respecto a la eficiencia de los mensajes *Query* junto a los expuestos en la Tabla 5-11 y Tabla 5-12 respecto a la sobrecarga en el tráfico relacionada con las políticas de actualización, se concluye que DLP<sub>2</sub> y DLP<sub>3</sub> constituyen las políticas más convenientes. Sólo en caso que la frecuencia de actualización de conocimiento sea muy alta es aconsejable la utilización de DLP<sub>1</sub> que genera mínima sobrecarga por difusiones de mensajes *NeuralInf*. De lo contrario y para la mayoría de los casos DLP<sub>2</sub> aparece como la mejor opción y alternativamente, si la frecuencia de actualización es más baja, DLP<sub>3</sub> también puede alcanzar

excelentes resultados.

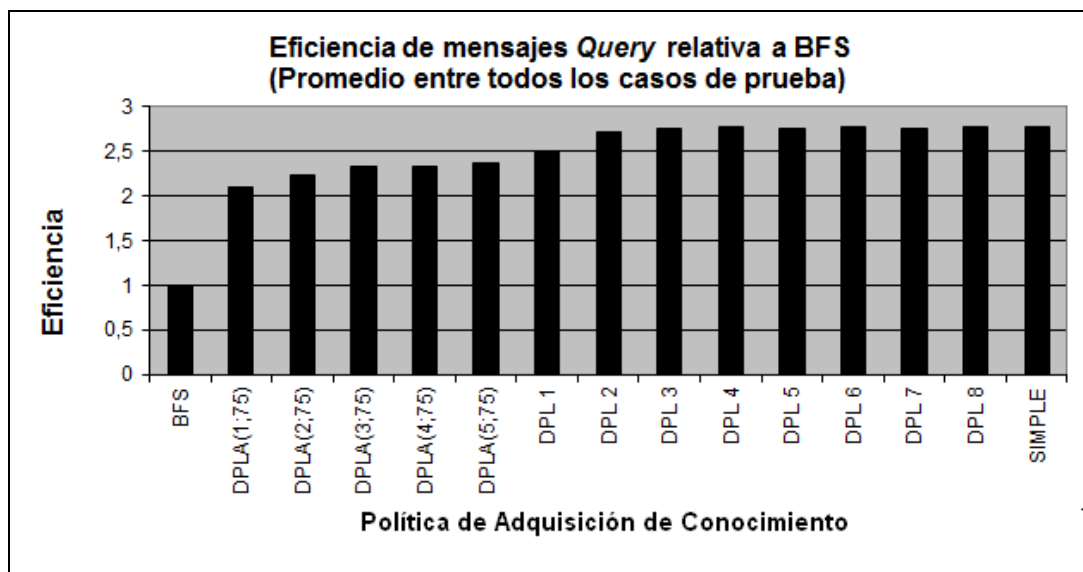


Figura 5-20 Eficiencia de los mensajes *Query* en la búsqueda inteligente BII-P2P promediada entre los 132 casos de prueba propuestos según las distintas políticas de adaptación definidas

Debe considerarse además que la situación descrita en la sección 4.2.3.4 “Alcance Real de la Información de Actualización en DLP y DLPA” —pág. 125— no ha sido considerada en la experimentación y, sin embargo puede mejorar la eficiencia de los mensajes *Query* para las políticas  $DLP_i$  cuando el valor de  $i$  es pequeño —para valores más altos de este parámetro la eficiencia permanece en un límite que no puede ser superado—. Este dato refuerza la idea sobre la conveniencia de utilizar una política  $DLP_i$  con  $i$  pequeño y debe tomarse como la conclusión más importante surgida de la evaluación sobre la capacidad de adaptación de BII-P2P.

Finalmente, para medir los efectos del enfoque híbrido de adquisición de conocimiento presentado en la sección 4.2.4 en la página 127, se midió la utilidad del aprendizaje en línea basado en LVQ1 procediendo de la siguiente manera:

Se partió de una configuración o descripción instantánea de un sistema P2P obtenido por medio del proceso de dos etapas descrito en la sección 5.2.1 “Experimentación” —pág. 176— construyendo una topología de 3.000 nodos que se sometió a una etapa de adquisición de conocimiento utilizando la política Simple para garantizar la mejor adaptación posible. Sea  $C_I$  esta configuración inicial.

Luego se midió el rendimiento de la búsqueda BII-P2P para distintas parametrizaciones  $(TTL, \gamma)$  sobre la configuración  $C_I$  de la red P2P por medio de la ejecución de 40 ensayos de búsqueda registrando los promedios de las métricas habituales —porcentaje de hallazgos y mensajes *Query* generados—. A partir de entonces se definieron dos escenarios distintos:

- i)* Partiendo de la configuración  $C_I$  se simularon 1.000 operaciones de búsqueda provocando un cambio en el conjunto de recursos compartidos cada dos búsquedas completadas y sin realizar ningún tipo de adaptación. Sea  $C_A$  la configuración de la red P2P así conseguida.
- ii)* Partiendo de la configuración  $C_I$  se simularon 1.000 operaciones de búsqueda con aprendizaje en línea basado en LVQ1 — $\alpha=0.05$ — provocando un cambio en el conjunto de recursos compartidos cada 2 operaciones de búsqueda. Sea  $C_B$  la configuración de la red P2P así conseguida.

En ambos casos el cambio en el conjunto de recursos compartido consistió en el movimiento de un recurso desde un nodo a otro nodo de la red P2P. Luego se midió la forma en que estos cambios degradaron el rendimiento del sistema de búsqueda sobre ambas configuraciones  $C_A$  y  $C_B$  realizando nuevamente 40 ensayos de búsqueda —sin ningún tipo de adaptación entre ellos— para obtener los promedios de las métricas habituales.

Los resultados de la experimentación se muestran en la Tabla 5-15. En todos los casos de prueba definidos para distintas parametrizaciones (TTL;  $\gamma$ ) de BII-P2P el rendimiento de la búsqueda a partir de la configuración  $C_B$  es ligeramente superior al conseguido a partir de la configuración  $C_A$ . Sin embargo, las diferencias observadas no son estadísticamente significativas —p-valor de la prueba *t-Student* mayor a 0,01— por lo que aún no puede concluirse que el aprendizaje en línea basado en LVQ1 atenúe efectivamente la pérdida de rendimiento que los cambios en el sistema producen sobre la búsqueda inteligente.

**Tabla 5-15 Comparativa de BII-P2P con y sin aprendizaje LVQ1.  $C_I$ =Configuración inicial.  $C_A$ =Configuración luego de simular cambios sin aprendizaje en línea.  $C_B$ =Configuración luego de simular cambios utilizando aprendizaje en línea basado en LVQ1**

BII-P2P (TTL; $\gamma$ )	Configuración	Mensajes <i>Query</i> enviados	Porcentaje de recursos hallados
(5; 0,50)	$C_I$	811,58	52,67 %
	$C_A$	815,11	49,45 %
	$C_B$	816,68	50,05 %
(7; 0,55)	$C_I$	4133,33	93,72 %
	$C_A$	4177,50	90,45 %
	$C_B$	4174,08	90,66 %
(10; 0,40)	$C_I$	1974,00	86,75 %
	$C_A$	1976,22	84,01 %
	$C_B$	1976,01	84,70 %
(13; 0,35)	$C_I$	1232,13	81,58 %
	$C_A$	1266,65	80,04 %
	$C_B$	1269,91	80,13 %

Una característica importante del aprendizaje en línea basado en LVQ1 es que no genera tráfico sobre la red P2P. Por ello, conseguir resultados significativos con este tipo de aprendizaje sigue siendo una meta de interés. Los resultados expuestos en la Tabla 5-15 pueden considerarse preliminares y un indicio de su factibilidad. Por lo tanto se deja planteada como una línea futura de investigación

la profundización de esta estrategia de adquisición de conocimiento.

#### 5.2.4 Conclusiones sobre la Capacidad de Adaptación

La capacidad de adaptación del sistema de búsqueda BII-P2P está relacionada con la habilidad de los nodos para adquirir conocimiento desde su entorno y así mantener bien ajustadas las redes neuronales que participan en las decisiones de encaminamiento de los mensajes *Query*. Esta habilidad a su vez se ve influenciada por la política de adquisición y actualización de conocimiento utilizada en el sistema que regula la forma en que se propagan los mensajes *NeuralInf* con información de actualización.

Las políticas de adaptación Simple,  $DLP_i$  —Difusión Limitada en Profundidad con parámetro TTL igual a  $i$ — y  $DLPA_{(i,j)}$  —Difusión Limitada en Profundidad y Anchura con parámetros TTL igual a  $i$  y porcentaje de selección de vecinos igual a  $j$ — han sido minuciosamente examinadas por medio de un extenso conjunto de casos experimentales.

Los resultados arrojados por la experimentación son concluyentes a favor de la utilización de la política  $DLP_i$  con  $i < 4$ . Incluso  $DLP_1$ , con impacto ínfimo sobre el tráfico de la red, garantiza rendimientos de búsqueda muy superiores al algoritmo BFS de *Gnutella*.  $DLP_2$  mejora el rendimiento de  $DLP_1$  y, en menor medida  $DLP_3$  mejora el rendimiento de  $DLP_2$ . Así, conforme crece el valor del parámetro  $i$  la ganancia en eficiencia se va haciendo más pequeña hasta que finalmente se estanca en un límite que no puede ser superado y que coincide con el rendimiento máximo de la búsqueda inteligente que, sin embargo, no se diferencia mucho del rendimiento alcanzado por  $DLP_3$ .

De los resultados observados también se concluye que si la frecuencia con la que los nodos inician difusiones de mensajes *NeuralInf* es alta, la política más conveniente será  $DLP_1$ . Por el contrario, si esta frecuencia es baja, es posible utilizar  $DLP_2$  o  $DLP_3$  sin impactar de manera considerable sobre el tráfico de la

red. Recuérdese que la frecuencia con que se desencadenan las difusiones de actualización está supeditada al dinamismo propio del sistema P2P y al parámetro umbral  $u$  de BII-P2P acordado previamente —ver sección 4.2.3.5 “Determinación de la Frecuencia de Actualización” en pág. 127—.

Por último, se concluye que el enfoque híbrido que combina difusiones de mensajes *NeuralInf* con entrenamiento en línea basado en LVQ1 puede aplicarse en el caso que el ahorro de tráfico sobre la red sea muy prioritario, utilizando un valor para el umbral  $u$  muy alto que provoque una frecuencia muy baja de difusiones de mensajes de actualización. Sin embargo los resultados de este enfoque de aprendizaje aún no son concluyentes y su profundización se plantea como una posible línea futura de investigación.

### 5.3 Conclusiones Generales sobre la Búsqueda Inteligente en BII-P2P

La búsqueda inteligente implementada en BII-P2P constituye una innovación que resuelve con eficiencia el problema de la búsqueda de recursos en redes P2P puras no estructuradas. Basando su funcionamiento en la utilización de redes neuronales artificiales implantadas en todos los nodos del sistema P2P, la búsqueda inteligente mejora considerablemente el rendimiento del algoritmo BFS de *Gnutella*. La comprobada habilidad de las redes neuronales para predecir los caminos más adecuados por dónde encaminar las solicitudes de búsqueda, es la responsable del alto porcentaje de decisiones acertadas que conducen a resultados exitosos y a una mejor utilización del ancho de banda de la red.

Los parámetros TTL y  $\gamma$  de la estrategia BII-P2P impactan significativamente sobre el rendimiento de la búsqueda inteligente. Determinar *a priori* los valores más adecuados para estos parámetros puede resultar difícil, sin embargo se han presentado una gran cantidad de combinaciones de valores posibles, todas ellas indiscutiblemente ventajosas. Una regla empírica con la que muy

frecuentemente se mejora el rendimiento del algoritmo BFS de *Gnutella* en ambas métricas consideradas —más instancias del recurso halladas y menos tráfico generado sobre la red—, consiste en utilizar un valor de TTL igual al doble del usado por *Gnutella* y un valor intermedio bajo para  $\gamma$ , entre 0,3 y 0,4. No obstante, esta situación puede ser distinta dependiendo de la topología de la red P2P por lo que se aconseja especial cuidado en la elección del valor para el parámetro  $\gamma$ , especialmente en una red con muchos nodos dónde el impacto de este valor es más grande.

Se observa que, bajo las parametrizaciones adecuadas, el tamaño de la red P2P no influye significativamente en los resultados de las pruebas realizadas conservándose la ventaja en el rendimiento de BII-P2P respecto a BFS en todas las topologías de redes probadas.

Para que la búsqueda inteligente BII-P2P conserve en el tiempo su alto rendimiento, es necesario mantener actualizado el conocimiento adquirido por las redes neuronales de los nodos. El mecanismo de aprendizaje definido, basado en la difusión de información de actualización por parte de los nodos, permite la rápida adaptación del sistema. La eficiencia de este mecanismo se ve afectada por la política de actualización que reglamenta la forma en que se propagan los mensajes *NeuralInf*.

Las pruebas realizadas demostraron que no es necesario propagar los mensajes de actualización hacia nodos alejados. La política  $DLP_2$ , que limita el alcance de los mensajes *NeuralInf* por medio del campo  $TTL=2$ , es la aconsejable en la mayoría de los casos.

Si la necesidad de restringir el tráfico sobre la red es grande, la política de difusión  $DLP_1$  es la indicada. Esta política establece que la propagación de información de actualización se limita al conjunto de vecinos directos —mensajes *NeuralInf* con  $TTL=1$ —.  $DLP_1$  ha mostrado ser sumamente eficiente generando

sobrecarga de tráfico prácticamente despreciable. Aunque los resultados todavía no son concluyentes,  $DLP_1$  puede complementarse con una adaptación en línea del algoritmo LVQ1 de aprendizaje para redes neuronales con la expectativa de atenuar los efectos negativos de los cambios en el ambiente y disminuir las difusiones de mensajes *NeuralInf*. Se propone como línea futura de investigación el desarrollo de las modificaciones necesarias para hacer que este entrenamiento híbrido resulte definitivamente ventajoso.



## Capítulo 6

# Exploración Incremental en BII-P2P

La búsqueda inteligente incremental para redes P2P, expuesta en esta tesis, se conforma con dos elementos fundamentales que se reflejan en el nombre de la estrategia. La componente más importante de BII-P2P es la *búsqueda inteligente* que representa la esencia del método propuesto, razón por la cual los capítulos 3, 4 y 5 han sido dedicados por completo a su presentación, descripción y análisis de rendimiento. Extendiendo a la primera, la *exploración incremental* constituye la segunda componente importante de BII-P2P y es el tema que se desarrolla en este capítulo. Por medio de ella es posible mejorar algunos resultados ejerciendo mayor control sobre la forma en que se difunden los mensajes de solicitud en la red P2P.

La exploración incremental constituye una extensión a la búsqueda inteligente que permite, en muchos casos, economizar el envío de mensajes evitando que el sistema continúe esforzándose por hallar mayor cantidad de instancias en el caso que la búsqueda ya se considere satisfactoria.

La idea subyacente en la exploración incremental es maximizar la eficiencia de los mensajes *Query*, permitiendo que los nodos inicien solicitudes de búsqueda sobre una región acotada de la red P2P y, sólo en caso de no alcanzar resultados satisfactorios, vayan ampliando en etapas sucesivas la región explorada. Estos

incrementos sobre el área de exploración deben realizarse además, evitando el solapamiento de las regiones visitadas en cada expansión para que los nodos no vuelvan a ser consultados por la misma búsqueda en distintas etapas incrementales.

## 6.1 Motivación

Los resultados experimentales obtenidos muestran que la búsqueda inteligente propuesta en esta tesis, bajo las parametrizaciones adecuadas, incrementa considerablemente el aprovechamiento del ancho de banda disponible al mejorar la eficiencia de los mensajes *Query*, conducidos sólo hacia los nodos más prometedores. De esta forma, la búsqueda inteligente genera menos tráfico que el algoritmo BFS de *Gnutella* y al mismo tiempo encuentra mayor cantidad de instancias del recurso buscado.

Sin embargo, determinar la mejor elección de los valores para los parámetros TTL y  $\gamma$  que aseguren siempre la ventaja en el rendimiento de BII-P2P respecto a BFS en cualquier topología de red P2P, puede ser dificultoso. Los valores más adecuados para estos parámetros dependen en gran medida de aspectos propios de la red que los nodos desconocen, como lo son el tamaño de la misma, la cantidad de conexiones existentes y forma particular en que se establecen los circuitos de comunicación entre los pares de la red. Los datos obtenidos experimentalmente han mostrado además que cuanto más grande es la red P2P más sensible se hace el sistema a las variaciones de estos parámetros.

Por lo tanto, encontrar alguna forma de resolver las dificultades asociadas a la correcta parametrización de BII-P2P resulta una motivación importante. La exploración incremental representa en parte una solución a este problema que se vale de una estrategia de tanteo para determinar los límites de la región de búsqueda a explorar. Aunque se utilizan valores fijos para los parámetros TTL y  $\gamma$  los efectos de las etapas incrementales se asemejan a un ajuste dinámico del

parámetro  $\gamma$  para cada búsqueda en particular.

Otra motivación para el desarrollo de la estrategia de exploración incremental recae en la necesidad de resolver situaciones donde no existe interés por hallar la mayor cantidad de instancias de un recurso sino por asegurar que todas las búsquedas sean exitosas. Por búsqueda exitosa se entiende aquella que alcanza al menos un nodo que posee el recurso buscado. En estas situaciones, si el recurso buscado es muy abundante, la búsqueda inteligente tal cual fue presentada en el capítulo 5, encontrará más instancias de las requeridas. Esta situación pone en evidencia que es posible reducir el tráfico generado limitando el alcance de la búsqueda y aún así seguir satisfaciendo los requerimientos de la misma. Si por el contrario, el recurso buscado es muy escaso, puede ocurrir que la búsqueda inteligente no encuentre ninguna de las instancias existentes. En tal caso sería conveniente ampliar el alcance de la misma para incrementar así la probabilidad de éxito. Estas ideas están presentes en la exploración incremental que se describe a continuación.

## 6.2 Descripción de la Estrategia de Exploración Incremental

Las motivaciones mencionadas en la sección anterior dieron origen a la estrategia de exploración incremental que trabaja de la siguiente manera: Los nodos comienzan siempre el proceso de búsqueda de un recurso realizando una exploración mínima, sobre una región acotada de la red. Sin embargo, luego de transcurrido un tiempo de espera previamente definido y ante la ausencia o escasez de resultados positivos, el nodo iniciador de la solicitud puede impulsar la búsqueda hacia nuevas regiones, ampliando el área de exploración y alcanzando mayor cantidad de nodos. A esta segunda instancia de búsqueda sobre el mismo recurso se la denomina primera etapa incremental. Si los resultados siguen siendo estériles el nodo puede proseguir lanzando sucesivas etapas incrementales hasta satisfacer su requerimiento, agotar las regiones incrementales disponibles, o bien,

desistir de la búsqueda.

Para implementar esta estrategia de exploración incremental, es necesario definir la forma en que se establecen las regiones de la red P2P sobre las que actúa la búsqueda en cada una de sus etapas. Pueden concebirse distintas modalidades de resolver esta cuestión pero rápidamente se visualizan dos posibles candidatas, una basada en la profundidad y la otra basada en el ancho de la búsqueda.

Considérese la siguiente estrategia para definir las regiones de exploración basada en profundidad. El nodo que inicia una búsqueda utiliza un valor pequeño en el campo TTL de los mensajes *Query* que, siguiendo los lineamientos de la búsqueda inteligente, propaga sólo a través de los vecinos más prometedores. Agotado el tiempo de espera, en ausencia o escasez de respuestas positivas —mensajes *QueryHit*— el nodo pasa a la etapa incremental repitiendo la búsqueda pero utilizando un valor de TTL más grande. Los nuevos mensajes *Query*, sometidos a las decisiones locales de ruteo que realizan los nodos, se propagan por las mismas rutas establecidas en la etapa anterior, pero sobrepasando el límite de la región previa explorada. Si los resultados siguen siendo negativos se irán sucediendo las etapas incrementales aumentando paulatinamente el valor de TTL para realizar búsquedas cada vez más profundas llegando a nodos más alejados.

La modalidad presentada en el párrafo anterior establece la propagación de la primera búsqueda sobre la región más cercana al nodo solicitante. Aquellos pares que se encuentren a mayor distancia sólo podrán ser alcanzados en alguna de las etapas incrementales subsiguientes. Claramente esta estrategia es ineficiente. En la segunda etapa de la búsqueda —primera etapa incremental—, se deben visitar todos los nodos ya consultados durante la primera para poder alcanzar aquellos que estén más allá del límite establecido por el TTL original. Lo mismo ocurre para todas las etapas incrementales subsiguientes. Generar nuevamente todo el tráfico de la etapa anterior para acceder a los nodos más alejados la convierte en una modalidad poco atractiva. Por lo tanto, se descarta la definición de las

regiones incrementales basada en el parámetro TTL.

Por el contrario, la definición de las regiones a explorar en cada etapa incremental basada en el ancho de la propagación resulta mucho más atractiva. BII-P2P utiliza esta modalidad que le permite, conforme se suceden las etapas incrementales, ir ampliando el alcance de la búsqueda haciendo cada vez más ancha la región explorada de la red. Esta modalidad resulta mucho más atractiva que la expuesta en los párrafos anteriores porque descubre nuevas rutas de propagación sin necesidad de volver a recorrer todos los caminos transitados en etapas previas.

La exploración incremental en BII-P2P utiliza un valor alto para el parámetro TTL y uno muy bajo para el parámetro  $\gamma$ . Así se produce una búsqueda inicial estrecha y profunda en la que los nodos afectados propagan la solicitud a un subconjunto pequeño de vecinos. Agotado el tiempo de espera, si el resultado no es satisfactorio el nodo iniciador de la búsqueda pasa a la etapa incremental repitiendo la misma solicitud, etiquetándola como primer incremento y enviándola a un número reducido de vecinos distintos de aquellos que recibieron la solicitud original, siguiendo el orden de mérito determinado por su red neuronal. Los nuevos mensajes *Query* entonces se propagan evitando la consulta repetida a nodos visitados en etapas anteriores, sin embargo, esta situación no puede evitarse por completo debido a la posible presencia de ciclos en la topología de la red. En caso de ser necesario, se irán sucediendo las etapas incrementales ensanchando cada vez más el espacio de búsqueda y haciendo más grande la región explorada de la red P2P.

Los valores de los parámetros TTL y  $\gamma$  se consideran parámetros globales del sistema que todos los nodos conocen. Utilizar otro esquema posible, como por ejemplo la utilización de valores, localmente definidos por los nodos para estos parámetros se deja planteado como trabajo futuro de investigación.

### 6.3 Implementación de la Exploración Incremental en BII-P2P

Para implementar la exploración incremental se ha redefinido el contenido de los mensajes *Query*, agregando un campo INC con el objeto de identificar la etapa incremental en la que está participando el mensaje.

Cuando un nodo inicia la búsqueda de un recurso, coloca el valor cero en el campo INC de los mensajes *Query* que envía al subconjunto elegido de vecinos. Este valor indica que la búsqueda se encuentra en su etapa regular. El valor del campo INC establecido por el nodo iniciador, al igual que el del campo ID de la búsqueda, permanece inalterado a lo largo de la propagación de la misma. Los nodos intermedios que participan en la propagación de la búsqueda replican los campos INC e ID en sus propios mensajes de *Query* antes de difundirlos.

Agotado el tiempo de espera el nodo iniciador de la búsqueda puede considerar insuficientes las respuestas obtenidas y comenzar la primera etapa incremental. Para ello establece en uno el campo INC de los mensajes *Query*, conserva el mismo ID de la búsqueda original e inicia la nueva propagación de mensajes. Agotado por segunda vez el tiempo de espera se tiene por concluida la primera etapa incremental. Si los resultados siguen siendo insatisfactorios de acuerdo con las expectativas del nodo, éste puede impulsar la búsqueda hacia la próxima etapa, incrementando en uno el campo INC de los nuevos mensajes de *Query* y comenzando una nueva difusión. En todas las etapas incrementales, los mensajes *Query* son enviados hacia el subconjunto de los mejores vecinos del nodo aún no elegidos en etapas previas de la búsqueda.

El nodo que recibe un mensaje *Query* trabaja de la siguiente manera. Verifica el ID e INC de la búsqueda. Si ya ha procesado una solicitud con iguales ID e INC, el *Query* se descarta para evitar ciclos. Si este no es el caso, y además se trata de una nueva búsqueda en su etapa regular —identificable por INC=0—

procederá de la forma habitual respondiendo con un *QueryHit* en caso de poseer el recurso y según sea necesario, si  $TTL > 1$ , propagando el *Query* sólo a los vecinos más prometedores de acuerdo al ordenamiento establecido por su red neuronal. La cantidad de vecinos elegida ha de ser reducida ya que el valor de  $\gamma$  utilizado es pequeño. Luego el nodo guarda en una base de datos local los campos ID e INC de la búsqueda junto con la lista de vecinos bloqueados para futuras etapas incrementales. Esta lista está formada por todos los nodos a quienes propagó el *Query* y aquel del cual se recibió la solicitud.

La utilidad de la información guardada es temporal y puede eliminarse luego de un tiempo prudencial. Esta información se utiliza con dos finalidades: *i*) evitar ciclos en la propagación de la búsqueda descartando las solicitudes repetidas, es decir aquellas con igual ID e INC, y *ii*) en caso de recibir un *Query* con el mismo ID de una búsqueda ya procesada pero con un INC distinto, evitar la propagación del mensaje hacia los vecinos bloqueados para esa búsqueda en particular, es decir a aquellos a los que ya se les envió la solicitud en alguna etapa anterior.

Si el nodo recibe un *Query* de una búsqueda que ya ha procesado en alguna etapa anterior, es decir que el ID coincide con alguno guardado en su base de datos local, pero el campo INC es distinto a todos los registros que posee la base para la búsqueda ID, no responde con *QueryHit* en caso de poseer el recurso solicitado pues ya lo hizo en alguna etapa anterior. En esta situación, el nodo actúa de la siguiente manera: *i*) obtiene la lista ordenada de vecinos determinada por su red neuronal para el recurso buscado, *ii*) excluye de esta lista a los nodos bloqueados según la información guardada en su base de datos local, *iii*) propaga el mensaje *Query* a los mejores vecinos elegidos de esta lista filtrada y *iv*) actualiza su base de datos incluyendo en la lista de vecinos bloqueados para la búsqueda ID a los nodos a quienes envió la consulta y aquel desde el cual la recibió.

La cantidad de vecinos a los que un nodo propaga una búsqueda es igual en

todas las etapas de la misma y se calcula por medio de la expresión  $\max(1, \lfloor \gamma n \rfloor)$  donde  $n$  es la cantidad de vecinos del nodo,  $\max$  es la función máximo y  $\lfloor \cdot \rfloor$  es la función piso que se expresa como  $\lfloor x \rfloor = \max\{k \in \mathbb{Z} | k \leq x\}$ . La excepción se produce cuando la cantidad de vecinos habilitados para una búsqueda en particular es menor a  $\max(1, \lfloor \gamma n \rfloor)$ , en ese caso la propagación se realiza a todos los vecinos habilitados. Por ejemplo si  $nodo_k$  posee 22 vecinos y  $\gamma=0,1$ ; la cantidad de nodos a los que  $nodo_k$  propaga las búsquedas en cualquiera de sus etapas regular o incrementales es  $\max(1, \lfloor 0,1 \times 22 \rfloor) = 2$ . Si debido a la presencia de ciclos en la topología de la red,  $nodo_k$  recibe varias solicitudes para una misma búsqueda en distintas etapas incrementales,  $nodo_k$  propagará todas las veces estas solicitudes a los 2 mejores vecinos elegidos de una lista cada vez más reducida de nodos habilitados.

En la Figura 6-1 se observa un esquema de la propagación de mensajes *Query* para la búsqueda de un recurso determinado  $r$ . Las flechas oscuras representan el envío de mensajes *Query* en la búsqueda inicial, y las más claras los mensajes propagados durante la primera etapa incremental. El valor que aparece sobre algunos de los enlaces representa la posición del vecino correspondiente en el ordenamiento establecido por la red neuronal del nodo emisor en función del recurso  $r$  y el perfil aprendido de cada uno de sus vecinos.

Siguiendo el ejemplo de la Figura 6-1, el nodo A inicia la búsqueda del recurso  $r$  alcanzando los nodos C, E y H. Debido a un valor pequeño de  $\gamma$  los nodos seleccionan sólo a un vecino al cual propagar la solicitud. Préstese atención a la situación del nodo E. Cuando éste recibe el mensaje *Query* desde el nodo C, verifica que se trata de una nueva búsqueda en su etapa regular, establece el ordenamiento de sus vecinos evaluando su red neuronal con entrada  $r$ , excluye de este ordenamiento al nodo C desde el que se recibió la consulta, propaga el mensaje hacia el nodo H que representa la opción mejor valorada y guarda los campos ID e INC junto a la lista [H,C] de nodos bloqueados.



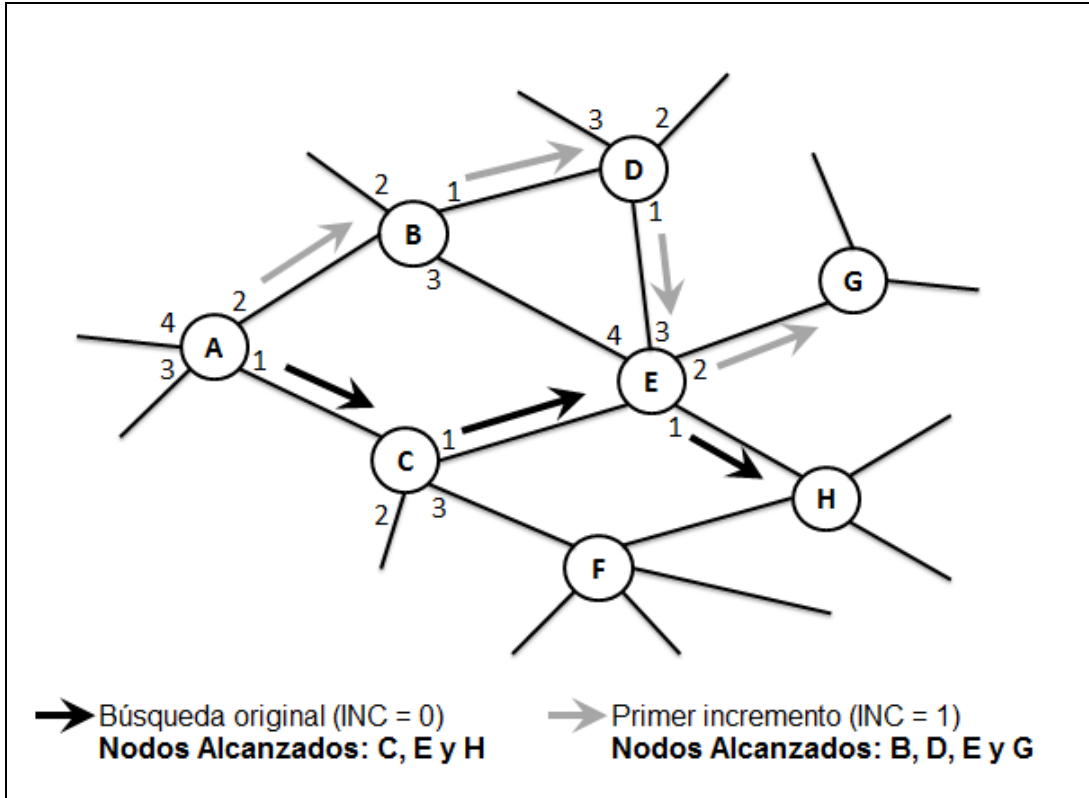


Figura 6-1 Búsqueda incremental. Propagación de mensajes *Query*

Agotado el tiempo de espera sin resultados positivos, el nodo A comienza la primera etapa incremental de búsqueda del recurso  $r$  alcanzando los nodos B, D, E y G. El nodo E es el único alcanzado dos veces por la misma búsqueda en distintas etapas y procede de la siguiente forma: una vez recibido el *Query* desde el nodo D, verifica accediendo a su base de datos local que se trata de una nueva etapa de una búsqueda previa procesada, recupera la lista de nodos bloqueados, establece el ordenamiento de vecinos excluyendo de éste a todos los nodos bloqueados, propaga el *Query* al nodo G y agrega a D y G a la lista de nodos bloqueados asociada al ID de la búsqueda.

Así puede verse que el nodo E elige en el primer intento al nodo H, que recibe la mejor valoración de parte de su red neuronal, pero en el segundo elige al nodo G, pues el H ya forma parte de los nodos bloqueados para esa búsqueda en

particular. Sin embargo el nodo B, aunque recibe el primer incremento de la búsqueda, al no haber recibido previamente la búsqueda original, utiliza su vecino mejor valorado por su red neuronal, el nodo D, para propagar la búsqueda.

Para maximizar la utilidad de esta estrategia el valor del  $\gamma$  debe ser pequeño. Así se posibilita que un nodo halle el recurso buscado generando muy poco tráfico. Pero en caso de no tener éxito en el primer intento, cuenta con la opción de seguir explorando la red sin que esto signifique revisar todos los nodos que ya habían sido alcanzados en etapas anteriores de la búsqueda original.

## 6.4 Evaluación de la Estrategia de Exploración Incremental

El objetivo propuesto para medir el rendimiento de la estrategia de exploración incremental, es hallar al menos una instancia del recurso buscado por cada una de las búsquedas iniciadas. La búsqueda inteligente de BII-P2P sin estrategia incremental obtiene muy buenos resultados maximizando la cantidad promedio de hallazgos por búsqueda, pero algunas de ellas, sobre todo si el recurso buscado es escaso, pueden resultar infructuosas a menos que se utilicen valores de TTL y  $\gamma$  más grandes con el consecuente aumento de tráfico sobre la red.

Por lo tanto, si se pretende garantizar el éxito en todas las búsquedas, hallando al menos una instancia del recurso buscado, la utilización de la búsqueda inteligente combinada con la estrategia incremental presentada en este capítulo resulta mucho más conveniente. Así lo evidencian los datos obtenidos durante la experimentación que se detalla a continuación.

### 6.4.1 Experimentación

La experimentación diseñada para evaluar el rendimiento de la estrategia incremental se ha basado en la simulación sobre cuatro configuraciones o

descripciones instantáneas distintas de una red P2P de 5.000 nodos. Cada configuración se construyó con distintos niveles de disponibilidad de recursos compartidos. Sobre estos cuatro escenarios se definieron distintos casos de pruebas variando el valor para el parámetro  $\gamma$  y la cantidad máxima permitida de etapas incrementales. Sobre estos casos de prueba se realizaron 40 ensayos de búsqueda independientes prestando atención a la cantidad de tráfico generado y al número de ensayos exitosos, es decir aquellos en los que se hallaron al menos una instancia del recurso buscado.

Para construir de manera aleatoria la topología de red y la distribución de recursos compartidos se prosiguió de igual forma a la descrita en el capítulo anterior, empleada para evaluar el rendimiento de la búsqueda inteligente. Cada uno de los 5.000 nodos que componen la red fueron conectados con una cantidad de al menos 4 y a lo sumo 12 vecinos elegidos de manera aleatoria. La asignación de los recursos compartidos en la red se realizó determinando de manera aleatoria el perfil de cada nodo, asignándole una cierta cantidad de clases distintas de recursos y para cada clase una cierta cantidad de recursos a compartir. Todos los recursos asignados fueron elegidos a partir de un mismo universo de recursos construido para tal fin. Permitiendo que cada elemento del universo sea elegido más de una vez, se promovió la existencia de múltiples instancias del mismo recurso en la configuración de la red.

Los cuatro escenarios distintos evaluados se construyeron variando la disponibilidad de los recursos compartidos asignados a cada nodo. Siguiendo el proceso de construcción de los escenarios de prueba, asignando distintas cantidades de recursos compartidos a cada nodo, han quedado definidos los siguientes niveles de disponibilidad: Escenario I) *Disponibilidad de recursos baja*, asignando 5 recursos compartidos a cada nodo, se obtuvieron en promedio 2,5 instancias por recurso en la red; Escenario II) *Disponibilidad de recursos media-baja*, asignando 50 recursos a cada nodo, se obtuvieron en promedio 25

instancias por recurso en la red; Escenario III) *Disponibilidad de recursos media*, asignando 100 recursos a cada nodo, se obtuvieron en promedio 50 instancias por recurso en la red y Escenario IV) *Disponibilidad de recursos alta*, asignando 200 recursos a cada nodo, se obtuvieron en promedio 100 instancias por recurso en la red.

### 6.4.2 Construcción de los Casos de Prueba

Para cada uno de los cuatro escenarios construidos se definieron 79 casos de prueba surgidos de la combinación de distintos valores para  $\gamma$  e *Inc*, dónde *Inc* representa el número máximo de etapas incrementales permitidas. Obsérvese que *Inc*=0 se corresponde con la búsqueda inteligente sin exploración incremental. Debe tenerse en cuenta que las etapas incrementales se disparan sólo si no se alcanza el éxito esperado en la etapa anterior, por lo tanto pueden existir búsquedas que se resuelvan tempranamente sin llegar a la etapa incremental correspondiente al valor de *Inc*.

En todos los casos de prueba definidos se ha utilizado TTL=8. Los distintos valores de  $\gamma$  utilizados para definir los casos de prueba fueron: 0,100; 0,125; 0,150; 0,175; 0,200; 0,225; 0,250; 0,375; 0,300; 0,325; 0,350; 0,375 y 0,400. Por cada valor de  $\gamma$  se definieron  $m+1$  casos de prueba eligiendo los valores para *Inc* = 0, 1, ...  $m$ . El valor de  $m$  está relacionado con el valor de  $\gamma$  porque a medida que aumenta  $\gamma$ , la cantidad de etapas incrementales posibles disminuye pues se agotan más rápidamente los vecinos habilitados de los nodos. El valor de  $m$  es el número natural que cumple con la siguiente condición:  $(m - 1)\gamma < 1$  AND  $m\gamma \geq 1$ .

### 6.4.3 Resultados de la Experimentación

Para cada caso de prueba definido se realizaron 40 búsquedas independientes determinadas aleatoriamente —recurso buscado y nodo iniciador—. Para cada búsqueda se registró la cantidad de mensajes *Query* generados, la cantidad de

etapas incrementales ejecutadas y si la búsqueda resultó exitosa de acuerdo al objetivo planteado —el hallazgo de al menos una instancia del recurso buscado—. De esta forma se pudo cuantificar el rendimiento de cada caso de prueba —cantidad promedio de mensajes *Query* generados por búsqueda— y la efectividad —cantidad de búsquedas exitosas de las 40 realizadas—. En el presente capítulo se presenta un resumen de estos resultados pero el conjunto completo de los datos recolectados puede consultarse en las tablas A-23, A-24, A-25 y A-26 a partir de la página 277 en la sección “Apéndice A”, que muestran los resultados obtenidos sobre los escenarios I, II, III y IV respectivamente. En cada una de estas tablas se ha recuadrado el caso no incremental con mayor efectividad y mejor rendimiento, es decir aquel con menor cantidad de tráfico elegido entre aquellos que alcanzaron 40 búsquedas exitosas. Este caso se utiliza como referente comparativo para calcular la significancia estadística sobre la diferencia de medias observadas respecto de los demás casos.

Las tablas 6-1, 6-2, 6-3 y 6-4 presentadas a continuación resumen los resultados más relevantes obtenidos durante la experimentación realizadas sobre los escenarios I, II, III y IV respectivamente. Se muestra el rendimiento, expresado en cantidad de tráfico generado, de la estrategia incremental y no incremental para los casos que maximizan la probabilidad de éxito de acuerdo al objetivo planteado —el hallazgo de al menos una instancia del recurso buscado—. Por lo tanto, las tablas 6-1, 6-2, 6-3 y 6-4 muestran sólo casos de prueba que han resultado 100% efectivos habiendo sido exitosas todas las búsquedas realizadas. En la primera fila de estas tablas se presenta el caso de referencia comparativa correspondiente al mejor rendimiento alcanzado por la búsqueda inteligente BII-P2P sin exploración incremental. En el resto de las filas se muestran todos los casos con exploración incremental que superan el rendimiento del caso de referencia con una media de mensajes *Query* significativamente menor. Para determinar el nivel de significancia estadística se ha utilizado un test *t-Student* para diferencia de medias, asumiendo que la evidencia es significativa cuando

p-valor < 0.001. Recuérdese que *Inc* representa el número máximo de etapas incrementales permitidas y que *Inc=0* se corresponde con la búsqueda inteligente sin exploración incremental.

**Tabla 6-1 Resultados obtenidos sobre escenario I con disponibilidad BAJA de recursos compartidos. La primera fila corresponde al mejor caso no incremental y las siguientes a los casos incrementales que mejoran significativamente el rendimiento del caso no incremental.**

$\gamma$	Inc	Éxitos	Querys		
			Media	Desv. Est.	p-valor
0,400	0	40 de 40	4376,975	1097,114	
0,300	2	40 de 40	931,875	448,010	2,55E-24
0,300	4	40 de 40	1020,550	438,162	9,60E-24
0,325	4	40 de 40	932,950	443,864	2,77E-24
0,350	1	40 de 40	2129,900	1049,141	2,18E-14
0,350	3	40 de 40	2119,225	953,810	3,36E-15
0,375	1	40 de 40	3472,675	1195,347	7,15E-04
0,375	3	40 de 40	3332,225	1025,327	3,40E-05

**Tabla 6-2 Resultados obtenidos sobre escenario II con disponibilidad MEDIA-BAJA de recursos compartidos. La primera fila corresponde al mejor caso no incremental y las siguientes a los casos incrementales que mejoran significativamente el rendimiento del caso no incremental.**

$\gamma$	Inc	Éxitos	Querys		
			Media	Desv. Est.	p-valor
0,275	0	40 de 40	376,900	212,920	
0,200	3	40 de 40	63,725	32,242	1,69E-11
0,200	4	40 de 40	63,275	40,506	1,54E-11
0,200	5	40 de 40	66,550	46,382	1,99E-11
0,225	1	40 de 40	99,700	45,346	4,31E-10
0,225	2	40 de 40	99,175	52,844	4,08E-10
0,225	3	40 de 40	104,575	46,979	6,82E-10
0,225	4	40 de 40	102,600	53,792	5,66E-10
0,225	5	40 de 40	104,125	45,464	6,54E-10

**Tabla 6-3 Resultados obtenidos sobre escenario III con disponibilidad MEDIA de recursos compartidos. La primera fila corresponde al mejor caso no incremental y las siguientes a los casos incrementales que mejoran significativamente el rendimiento del caso no incremental.**

$\gamma$	Inc	Éxitos	Querys		
			Media	Desv. Est.	p-valor
0,250	0	40 de 40	259,725	138,332	
0,100	4	40 de 40	12,925	6,919	7,28E-14
0,100	5	40 de 40	14,100	9,535	8,19E-14
0,100	6	40 de 40	12,400	7,459	6,80E-14
0,100	7	40 de 40	14,000	7,190	8,29E-14
0,100	8	40 de 40	13,800	7,244	8,08E-14
0,100	10	40 de 40	13,200	7,144	7,52E-14
0,125	5	40 de 40	14,400	10,119	8,43E-14
0,125	7	40 de 40	13,600	9,790	7,67E-14
0,150	3	40 de 40	14,575	8,277	8,80E-14
0,150	6	40 de 40	12,150	5,977	6,68E-14
0,150	7	40 de 40	12,900	9,607	7,06E-14
0,175	5	40 de 40	23,500	12,496	2,54E-13
0,200	1	40 de 40	59,425	32,926	2,35E-11
0,200	2	40 de 40	64,450	29,138	4,92E-11
0,200	3	40 de 40	60,000	38,699	2,45E-11
0,200	4	40 de 40	68,400	46,411	8,39E-11
0,200	5	40 de 40	61,650	34,550	3,18E-11
0,225	1	40 de 40	93,875	47,472	3,98E-09
0,225	4	40 de 40	100,100	42,797	9,61E-09
0,225	5	40 de 40	102,175	54,777	1,65E-08

**Tabla 6-4 Resultados obtenidos sobre escenario IV con disponibilidad ALTA de recursos compartidos. La primera fila corresponde al mejor caso no incremental y las siguientes a los casos incrementales que mejoran significativamente el rendimiento del caso no incremental**

$\gamma$	Inc	Éxitos	Querys		
			Media	Desv. Est.	p-valor
0,225	0	40 de 40	82,350	43,459	
0,100	3	40 de 40	10,400	5,495	5,92E-13
0,100	4	40 de 40	10,200	5,120	5,55E-13
0,100	5	40 de 40	10,800	7,370	6,30E-13
0,100	6	40 de 40	12,200	7,894	1,10E-12

Continúa la tabla...

$\gamma$	Inc	Éxitos	Querys		
			Media	Desv. Est.	p-valor
0,100	7	40 de 40	11,375	6,163	8,56E-13
0,100	8	40 de 40	11,400	5,088	9,10E-13
0,100	9	40 de 40	9,200	3,413	3,97E-13
0,100	10	40 de 40	10,800	5,599	6,94E-13
0,125	3	40 de 40	12,375	6,993	1,25E-12
0,125	4	40 de 40	11,750	7,469	9,33E-13
0,125	5	40 de 40	9,800	5,580	4,60E-13
0,125	7	40 de 40	10,800	4,268	7,36E-13
0,125	8	40 de 40	11,800	9,411	8,52E-13
0,150	2	40 de 40	9,325	3,633	4,14E-13
0,150	3	40 de 40	9,400	3,078	4,35E-13
0,150	4	40 de 40	9,925	4,015	5,21E-13
0,150	5	40 de 40	9,800	4,244	4,90E-13
0,150	6	40 de 40	9,775	4,627	4,77E-13
0,175	3	40 de 40	18,325	10,584	1,42E-11
0,175	4	40 de 40	18,875	11,225	1,79E-11
0,175	5	40 de 40	18,650	10,736	1,64E-11
0,175	6	40 de 40	16,400	8,732	6,46E-12

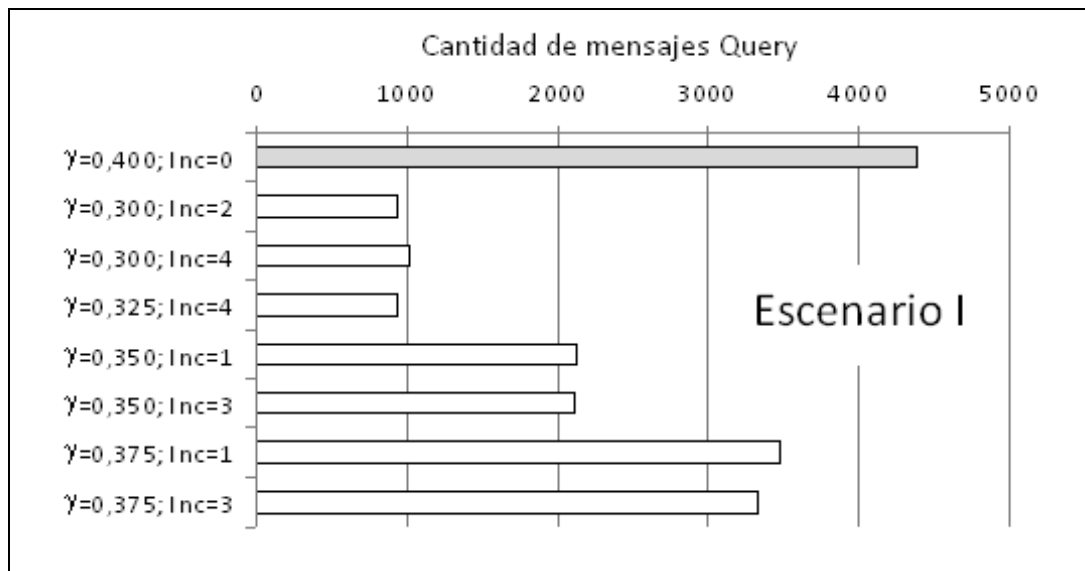


Figura 6-2 Tráfico generado por estrategia incremental y no incremental sobre escenario I con disponibilidad BAJA de recursos compartidos



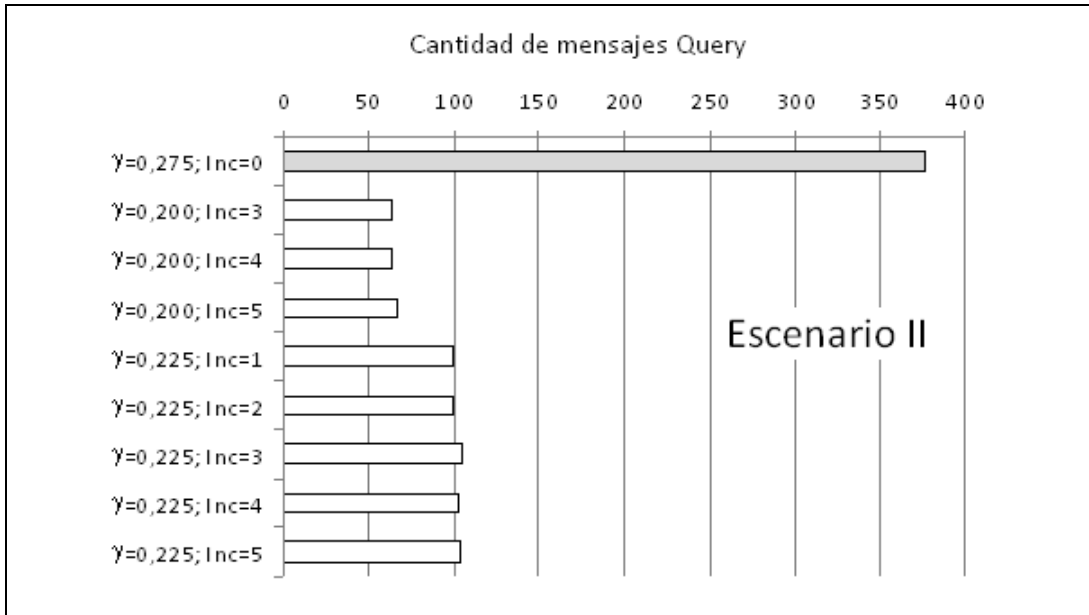


Figura 6-3 Tráfico generado por estrategia incremental y no incremental sobre escenario II con disponibilidad MEDIA-BAJA de recursos compartidos

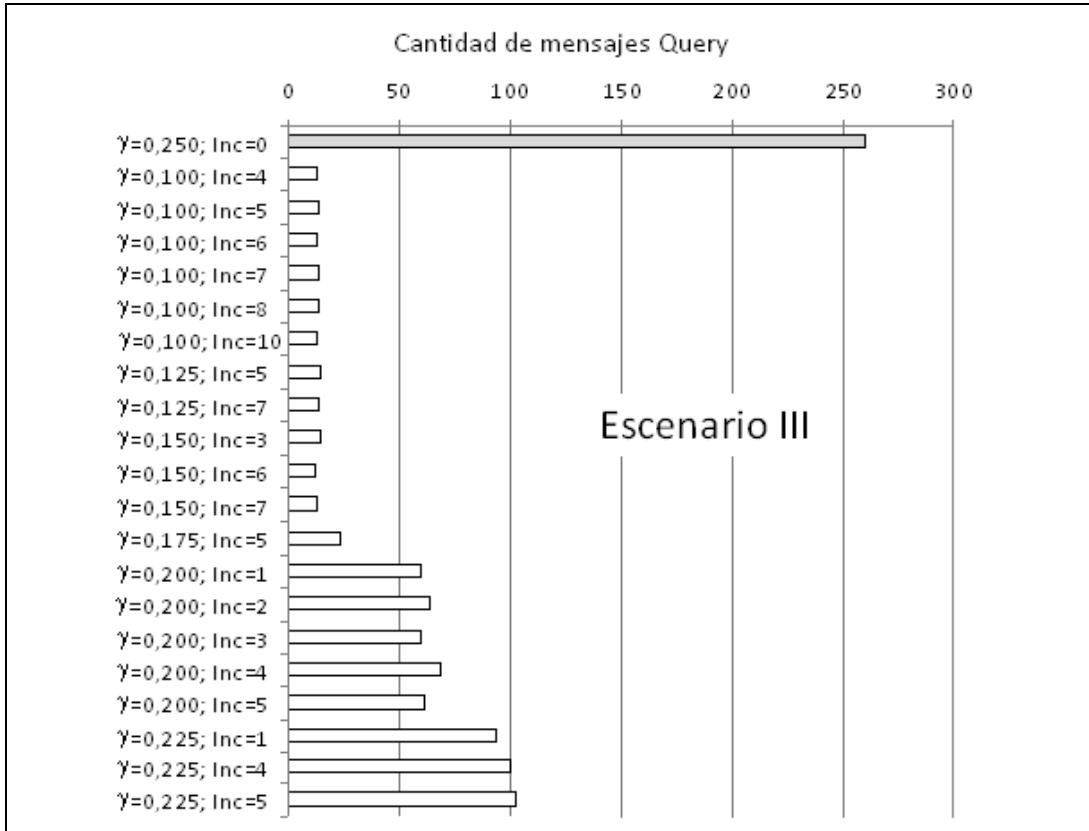
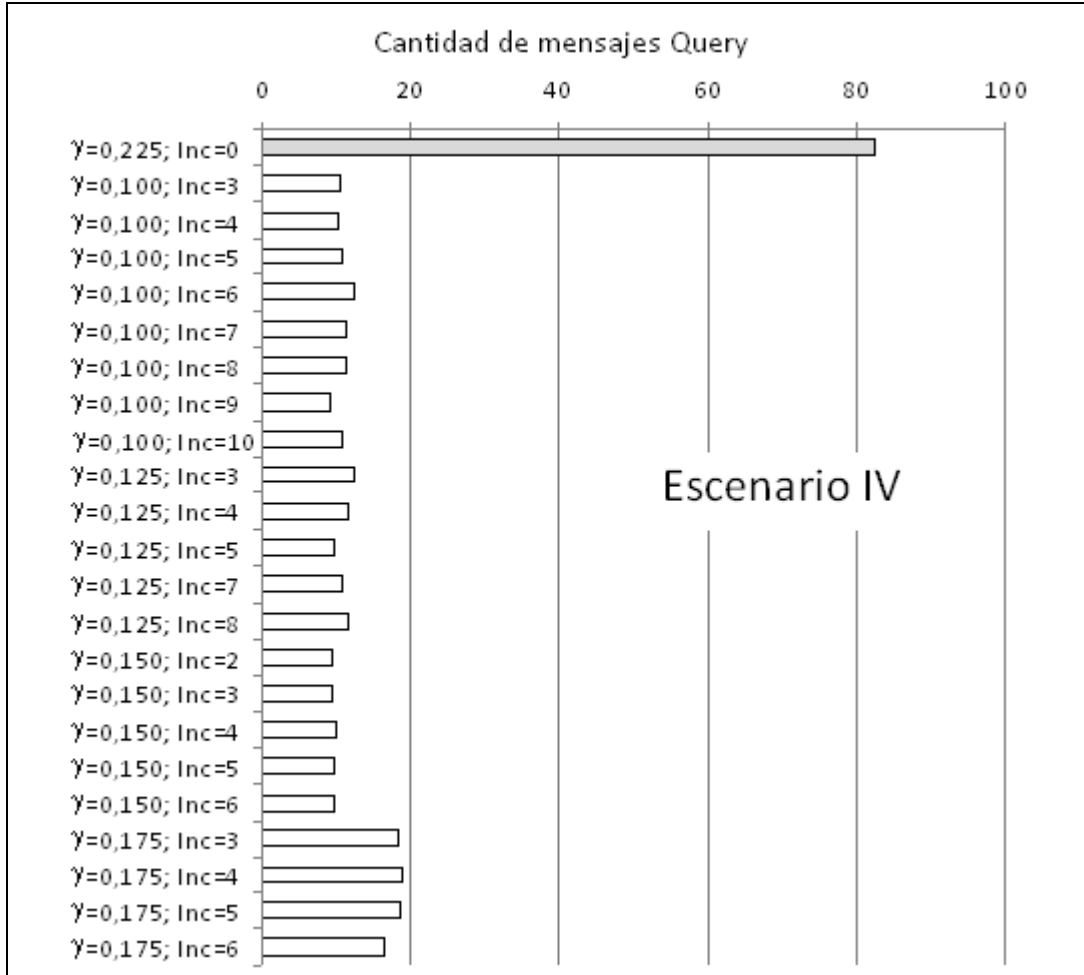


Figura 6-4 Tráfico generado por estrategia incremental y no incremental sobre escenario III con disponibilidad MEDIA de recursos compartidos



**Figura 6-5 Tráfico generado por estrategia incremental y no incremental sobre escenario IV con disponibilidad ALTA de recursos compartidos**

Las figuras 6-2, 6-3, 6-4 y 6-5 permiten apreciar con mayor facilidad las diferencias de rendimiento observadas en los casos de prueba presentados en las tablas anteriores para los cuatro escenarios considerados. Claramente la búsqueda inteligente con exploración incremental mejora notablemente los resultados obtenidos por la estrategia no incremental y se muestra mucho más eficiente con niveles de tráfico muy inferiores. Aunque esta diferencia es más pronunciada en los escenarios con mayor disponibilidad de recursos —III y IV— también es importante en los escenarios I y II dónde las instancias de recursos escasean.

Como desventaja de la estrategia incremental debe considerarse el tiempo necesario para conducir una búsqueda hasta etapas incrementales altas. El tiempo de espera que un nodo debe agotar antes de pasar a la siguiente etapa hace deseable resolver la búsqueda utilizando la menor cantidad posible de incrementos. Por lo tanto, aquí aparece otra métrica importante a tener en cuenta que consiste en el tiempo de respuesta asociado a la cantidad de etapas incrementales utilizadas en una búsqueda.

El parámetro *Inc* especificado en los casos de prueba ( $\gamma$ ; *Inc*) no es un parámetro de la estrategia incremental implementada en BII-P2P sino de la experimentación realizada para evaluar el rendimiento de la metodología propuesta. En BII-P2P son los nodos quienes, de acuerdo a sus intereses, deciden si continúan disparando etapas incrementales hasta agotar todas las posibles. Aunque en el contexto de las pruebas realizadas, el criterio de los nodos ha sido lanzar, de ser necesario, hasta *Inc* etapas incrementales, no debe tomarse este valor como un indicativo directo del tiempo de respuesta. En muchos casos de prueba, todas las búsquedas realizadas se resolvieron utilizando menor cantidad de incrementos que los establecidos por el parámetro *Inc* de la experimentación. Las tablas 6-5, 6-6, 6-7 y 6-8 presentan un análisis detallado sobre las etapas incrementales utilizadas en los casos de prueba presentados anteriormente.

**Tabla 6-5 Detalle sobre etapas incrementales utilizadas en los casos de prueba presentados en la Tabla 6-1 sobre el escenario I.**

$\gamma$	<i>Inc</i>	Cant. de búsquedas que utilizaron i incrementos											Incrementos	
		i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.
0,300	2	37	3	0									0,075	0,267
0,300	4	32	4	2	2	0							0,350	0,802
0,325	4	34	5	1	0	0							0,175	0,446
0,350	1	36	4										0,100	0,304
0,350	3	37	2	1	0								0,100	0,379
0,375	1	39	1										0,025	0,158
0,375	3	39	1	0	0								0,025	0,158

**Tabla 6-6 Detalle sobre etapas incrementales utilizadas en los casos de prueba presentados en la Tabla 6-2 sobre el escenario II.**

$\gamma$	Inc	Cant. de búsquedas que utilizaron i incrementos										Incrementos	
		i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media
0,200	3	31	7	2	0							0,275	0,554
0,200	4	31	7	2	0	0						0,275	0,554
0,200	5	29	8	1	1	1	0					0,425	0,874
0,225	1	35	5									0,125	0,335
0,225	2	37	3	0								0,075	0,267
0,225	3	35	5	0	0							0,125	0,335
0,225	4	37	3	0	0	0						0,075	0,267
0,225	5	35	5	0	0	0	0					0,125	0,335

**Tabla 6-7 Detalle sobre etapas incrementales utilizadas en los casos de prueba presentados en la Tabla 6-3 sobre el escenario III.**

$\gamma$	Inc	Cant. de búsquedas que utilizaron i incrementos											Incrementos	
		i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.
0,100	4	22	13	4	0	1						0,625	0,868	
0,100	5	25	6	4	3	2	0					0,775	1,209	
0,100	6	26	9	3	1	1	0	0				0,550	0,932	
0,100	7	20	12	6	2	0	0	0	0			0,750	0,899	
0,100	8	20	14	3	3	0	0	0	0	0		0,725	0,905	
0,100	10	23	10	5	2	0	0	0	0	0	0	0,650	0,893	
0,125	5	23	10	3	1	2	1					0,800	1,265	
0,125	7	27	5	4	1	3	0	0	0			0,700	1,224	
0,150	3	21	9	6	4							0,825	1,035	
0,150	6	25	9	6	0	0	0	0				0,525	0,751	
0,150	7	27	8	1	2	1	1	0	0			0,625	1,192	
0,175	5	27	9	0	4	0	0					0,525	0,933	
0,200	1	38	2									0,050	0,221	
0,200	2	37	3	0								0,075	0,267	
0,200	3	36	4	0	0							0,100	0,304	
0,200	4	35	4	1	0	0						0,150	0,427	
0,200	5	38	2	0	0	0	0					0,050	0,221	
0,225	1	39	1									0,025	0,158	
0,225	4	38	2	0	0	0						0,050	0,221	
0,225	5	38	2	0	0	0	0					0,050	0,221	

**Tabla 6-8 Detalle sobre etapas incrementales utilizadas en los casos de prueba presentados en la Tabla 6-4 sobre el escenario IV.**

$\gamma$	Inc	Cant. de búsquedas que utilizaron i incrementos										Incrementos		
		i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.
0,100	3	32	5	2	1								0,300	0,687
0,100	4	33	3	4	0	0							0,275	0,640
0,100	5	33	4	0	2	1	0						0,350	0,921
0,100	6	28	7	2	2	1	0	0					0,525	0,987
0,100	7	29	6	4	1	0	0	0	0				0,425	0,781
0,100	8	25	14	0	1	0	0	0	0	0			0,425	0,636
0,100	9	35	4	1	0	0	0	0	0	0	0		0,150	0,427
0,100	10	30	7	2	1	0	0	0	0	0	0	0	0,350	0,700
0,125	3	25	11	1	3								0,550	0,876
0,125	4	29	6	3	1	1							0,475	0,933
0,125	5	34	5	0	0	1	0						0,225	0,698
0,125	7	27	12	1	0	0	0	0	0				0,350	0,533
0,125	8	31	6	0	0	2	1	0	0	0			0,475	1,176
0,150	2	34	5	1									0,175	0,446
0,150	3	33	7	0	0								0,175	0,385
0,150	4	31	8	1	0	0							0,250	0,494
0,150	5	33	5	2	0	0	0						0,225	0,530
0,150	6	34	3	3	0	0	0	0					0,225	0,577
0,175	3	35	3	2	0								0,175	0,501
0,175	4	36	2	2	0	0							0,150	0,483
0,175	5	37	3	0	0	0	0						0,075	0,267
0,175	6	34	6	0	0	0	0	0					0,150	0,362

Las tablas anteriores muestran que en realidad ninguno de los casos de prueba presentados necesitó más de 5 etapas incrementales para alcanzar el 100% de efectividad requerido. A modo de ejemplo considere el caso ( $\gamma=0,100$ ;  $Inc=10$ ) sobre el escenario IV presentado en la tabla anterior. Aunque en las 40 búsquedas realizadas se permitieron hasta 10 etapas incrementales, todas ellas se resolvieron utilizando a lo sumo 3 incrementos. Sólo 1 búsqueda alcanzó la tercera etapa incremental, 2 búsquedas alcanzaron la segunda etapa incremental, 7 búsquedas utilizaron sólo 1 etapa incremental y las 30 búsquedas restantes se resolvieron sin necesidad de disparar ninguna etapa incremental. Esto arroja un promedio de 0,35

etapas incrementales por búsqueda que constituye un valor aceptablemente pequeño desde la perspectiva global del sistema de búsqueda.

En los cuatro escenarios considerados existen casos como el mencionado en el párrafo anterior, el lector puede identificarlos fácilmente en las tablas 6-5, 6-6, 6-7 y 6-8. Estas tablas también nos permiten descubrir el caso más desfavorable en relación al tiempo de respuesta asociado a las etapas incrementales para cada escenario, estos son: el caso ( $\gamma=0,300$ ;  $Inc=4$ ) en el escenario I con un promedio de 0,35 incrementos por búsqueda; el caso ( $\gamma=0,200$ ;  $Inc=5$ ) en el escenario II con un promedio de 0,425 incrementos por búsqueda; el caso ( $\gamma=0,150$ ;  $Inc=3$ ) en el escenario III con un promedio de 0,825 incrementos por búsqueda y el caso ( $\gamma=0,125$ ;  $Inc=3$ ) en el escenario IV con un promedio de 0,550 incrementos por búsqueda. Obsérvese que aún en los casos más desfavorables, el costo asociado al tiempo de respuesta por el uso de etapas incrementales, es relativamente pequeño en relación al beneficio obtenido representado por niveles de tráfico muy inferiores a los registrados por la estrategia no incremental —ver tablas 6-1, 6-2, 6-3 y 6,4—.

Por lo tanto es posible afirmar que la estrategia de exploración incremental implementada en BII-P2P favorece la búsqueda disminuyendo significativamente el tráfico generado y manteniendo un alto grado de efectividad. El costo a pagar por el beneficio obtenido consiste en un aumento del tiempo de respuesta que, sin embargo, desde la perspectiva global del sistema de búsqueda, no resulta considerable.

Obsérvese que los casos de prueba presentados en las tablas anteriores se corresponden con aquellos que resultaron 100% efectivos alcanzando el objetivo propuesto en las 40 búsquedas realizadas. Por ello es que en el párrafo anterior se habla de alto grado de efectividad. Sin embargo, es válido preguntarse si la cantidad de éxitos obtenida en los casos de prueba estudiados es representativa de cualquier otro conjunto de 40 búsquedas elegidas al azar o por el contrario, si este

nivel de efectividad se debe sólo a una inusual conformación del conjunto de búsquedas utilizado en la experimentación. Es importante determinar la validez de las comparaciones antes realizadas verificando que la efectividad de los casos de prueba incrementales analizados en esta sección no resulta inferior a la exhibida por el caso no incremental utilizado como referente comparativo.

Para llevar a cabo este análisis de efectividad, por cada caso ( $\gamma; Inc$ ) se realizaron 40 pruebas independientes  $P_1, P_2, \dots, P_{40}$ . Por cada prueba  $P_i$  se conformó aleatoriamente el conjunto de experimentación  $C_i = \{B_{i1}, B_{i2}, \dots, B_{i40}\}$ , siendo  $B_{ij}$  la  $j$ -ésima búsqueda a realizarse durante la ejecución de la prueba  $P_i$ . Así, para cada prueba  $P_i$  se registró el valor  $E_i$  que representa la cantidad de búsquedas que alcanzaron el objetivo propuesto, es decir que  $E_i = \sum_{j=1}^{40} f(B_{ij})$  donde  $f(B)$  es una función que devuelve 1 si la búsqueda  $B$  resultó exitosa y 0 en caso contrario.

De esta forma, para cada caso de prueba ( $\gamma; Inc$ ) definido sobre los escenarios I, II, III y IV se obtuvo una muestra  $E_1, E_2, \dots, E_{40}$  cuya media y desvío estándar pueden consultarse en las tablas A-27, A-28, A-29 y A-30 a partir de la página 289 en la sección “Apéndice A”. En el presente capítulo, las tablas 6-9, 6-10, 6-11 y 6-12 resumen los resultados obtenidos sólo para los casos de prueba previamente analizados. Con ello se pretende establecer la validez de las comparaciones sobre el rendimiento antes realizadas al demostrar que los casos de exploración incremental resultan al menos igualmente efectivos que los casos no incrementales con los que se comparan.

La Tabla 6-9 presentada a continuación, muestra los resultados obtenidos por la experimentación descrita precedentemente, aplicada a los casos de prueba cuyos rendimientos fueron detallados en la Tabla 6-1 y graficados en la Figura 6-2. Estos casos se corresponden con los mejores rendimientos observados sobre el escenario I con baja disponibilidad de recursos compartidos. La Tabla 6-9 muestra que no existen diferencias significativas respecto de la media de éxitos

alcanzados entre los casos incrementales y el referente comparativo ( $\gamma=0,400$ ;  $Inc=0$ ) a excepción del caso ( $\gamma=0,375$ ;  $Inc=3$ ). Sin embargo, esta diferencia favorece a la estrategia incremental —39,925 contra 39.275—. Por lo tanto, se puede concluir que todos los casos de exploración incremental sobre el escenario I detallados en la Tabla 6-1, además de mejorar notablemente el rendimiento del referente comparativo generando menor cantidad de tráfico —ver Figura 6-2—, mantienen la misma efectividad o incluso la mejoran.

**Tabla 6-9 Análisis sobre la efectividad alcanzada por los casos de prueba presentados en la Tabla 6-1 sobre el escenario I.**

$\gamma$	Inc	Cantidad de búsquedas exitosas			
		Media	Desv. Est.	p-valor	Significativo
0,400	0	39,275	0,905		
0,300	2	38,475	1,467	4,62E-03	NO
0,300	4	38,975	1,143	1,97E-01	NO
0,325	4	39,075	0,859	3,14E-01	NO
0,350	1	39,175	0,874	6,17E-01	NO
0,350	3	39,800	0,608	3,31E-03	NO
0,375	1	39,650	0,622	3,43E-02	NO
0,375	3	39,925	0,350	9,69E-05	SI

La Tabla 6-10 presentada a continuación muestra los resultados sobre la cantidad de búsquedas exitosas alcanzadas por los casos de prueba visualizados en la Tabla 6-2 y Figura 6-3, correspondientes a los mejores rendimientos observados sobre el escenario II con disponibilidad media-baja de recursos compartidos. La Tabla 6-10 muestra que sólo en un caso la diferencia de la media respecto del referente comparativo —caso no incremental ( $\gamma=0,275$ ;  $Inc=0$ )— no es significativa. En el resto de los casos de exploración incremental las medias observadas resultan significativamente mejores a la del caso no incremental. Por lo tanto, se puede concluir que todos los casos de exploración incremental sobre el escenario II analizados precedentemente, superan notablemente el rendimiento del



referente comparativo —ver Figura 6-3—, al mismo tiempo que igualan o mejoran su efectividad.

**Tabla 6-10 Detalle sobre la efectividad alcanzada por los casos de prueba presentados en la Tabla 6-2 sobre el escenario II.**

$\gamma$	Inc	Cantidad de búsquedas exitosas			
		Media	Desv. Est.	p-valor	Significativo
<b>0,275</b>	0	39,075	0,859		
<b>0,200</b>	3	39,700	0,564	2,68E-04	SI
<b>0,200</b>	4	39,800	0,648	6,04E-05	SI
<b>0,200</b>	5	39,925	0,267	3,03E-07	SI
<b>0,225</b>	1	39,125	0,939	8,04E-01	NO
<b>0,225</b>	2	39,775	0,423	2,21E-05	SI
<b>0,225</b>	3	39,850	0,362	2,72E-06	SI
<b>0,225</b>	4	40,000	0,000	3,90E-08	SI
<b>0,225</b>	5	40,000	0,000	3,90E-08	SI

La Tabla 6-11 presentada a continuación, muestra los resultados sobre la cantidad de éxitos alcanzados por los casos de prueba visualizados en la Tabla 6-3 y Figura 6-4 correspondientes a los mejores rendimientos observados sobre el escenario III con disponibilidad media de recursos compartidos. La Tabla 6-11 muestra que para la mayoría de los casos con exploración incremental, las diferencias en las medias con respecto al referente comparativo —caso no incremental ( $\gamma=0,250; Inc=0$ )— no son significativas. Sin embargo el caso de prueba ( $\gamma=0,150; Inc=3$ ) debe excluirse del análisis de rendimiento pues representa el único caso de la tabla cuya media de éxitos alcanzados es significativamente menor a la del caso no incremental. Otros ejemplos dónde la diferencia de medias respecto de ( $\gamma=0,250; Inc=0$ ) resulta significativa lo constituyen los casos ( $\gamma=0,200; Inc=5$ ), ( $\gamma=0,225; Inc=4$ ) y ( $\gamma=0,225; Inc=5$ ). Sin embargo, en todos ellos, la diferencia se inclina a favor del caso incremental mostrando mayor efectividad que la búsqueda no incremental. Por lo tanto, se

puede concluir que, a excepción de ( $\gamma=0,150; Inc=3$ ), todos los casos de exploración incremental sobre el escenario III analizados en este capítulo, además de superar notablemente el rendimiento del referente comparativo —ver Figura 6-4— igualan o incluso mejoran la efectividad alcanzada por este último.

**Tabla 6-11 Detalle sobre la efectividad alcanzada por los casos de prueba presentados en la Tabla 6-3 sobre el escenario III.**

$\gamma$	Inc	Cantidad de búsquedas exitosas			
		Media	Desv. Est.	p-valor	Significativo
0,250	0	39,625	0,628		
0,100	4	39,000	0,987	1,22E-03	NO
0,100	5	39,275	1,012	6,77E-02	NO
0,100	6	39,150	0,834	5,25E-03	NO
0,100	7	39,675	0,572	7,11E-01	NO
0,100	8	39,575	0,594	7,16E-01	NO
0,100	10	39,525	0,679	4,96E-01	NO
0,125	5	39,325	0,764	5,89E-02	NO
0,125	7	39,425	0,781	2,11E-01	NO
0,150	3	38,050	1,280	3,42E-09	SI
0,150	6	39,175	0,931	1,35E-02	NO
0,150	7	39,575	0,747	7,47E-01	NO
0,175	5	39,850	0,362	5,40E-02	NO
0,200	1	39,750	0,494	3,25E-01	NO
0,200	2	39,950	0,221	3,33E-03	NO
0,200	3	39,975	0,158	1,37E-03	NO
0,200	4	39,975	0,158	1,37E-03	NO
0,200	5	40,000	0,000	5,30E-04	SI
0,225	1	39,925	0,267	7,50E-03	NO
0,225	4	40,000	0,000	5,30E-04	SI
0,225	5	39,975	0,158	1,37E-03	SI

Finalmente, la Tabla 6-12 presentada a continuación, muestra los resultados sobre la cantidad de éxitos alcanzados por los casos de prueba visualizados en la Tabla 6-4 y Figura 6-5 correspondientes a los mejores rendimientos observados sobre el escenario IV con disponibilidad alta de recursos compartidos. La Tabla

6-12 muestra que en todos los casos con exploración incremental, las diferencias en las medias con respecto al referente comparativo —caso no incremental ( $\gamma=0,225; Inc=0$ )— no resultan significativas. Por lo tanto, se puede concluir que todos los casos de exploración incremental sobre el escenario IV analizados en este capítulo, además de superar notablemente el rendimiento del referente comparativo —ver Figura 6-5 —, mantienen su misma efectividad.

**Tabla 6-12 Detalle sobre la efectividad alcanzada por los casos de prueba presentados en la Tabla 6-4 sobre el escenario IV.**

$\gamma$	Inc	Cantidad de búsquedas exitosas			
		Media	Desv. Est.	p-valor	Significativo
0,225	0	39,700	0,648		
0,100	3	39,775	0,480	5,58E-01	NO
0,100	4	39,775	0,423	5,42E-01	NO
0,100	5	39,775	0,480	5,58E-01	NO
0,100	6	39,875	0,335	1,35E-01	NO
0,100	7	39,975	0,158	1,25E-02	NO
0,100	8	39,975	0,158	1,25E-02	NO
0,100	9	39,950	0,221	2,53E-02	NO
0,100	10	39,875	0,335	1,35E-01	NO
0,125	3	39,600	0,672	5,00E-01	NO
0,125	4	39,950	0,221	2,53E-02	NO
0,125	5	39,825	0,446	3,19E-01	NO
0,125	7	40,000	0,000	5,70E-03	NO
0,125	8	39,950	0,221	2,53E-02	NO
0,150	2	39,150	0,864	1,92E-03	NO
0,150	3	39,675	0,656	8,64E-01	NO
0,150	4	39,750	0,439	6,88E-01	NO
0,150	5	39,900	0,304	8,29E-02	NO
0,150	6	39,875	0,404	1,52E-01	NO
0,175	3	39,925	0,267	4,76E-02	NO
0,175	4	39,975	0,158	1,25E-02	NO
0,175	5	39,975	0,158	1,25E-02	NO
0,175	6	40,000	0,000	5,70E-03	NO

## 6.5 Conclusiones sobre la Estrategia de Exploración Incremental en BII-P2P

La exploración incremental potencia a la búsqueda inteligente maximizando la eficiencia de los mensajes *Query* y economizando tráfico sobre la red, especialmente cuando el interés por hallar la mayor cantidad de instancias de un recurso no es tan importante como asegurar que todas las búsquedas sean exitosas, es decir, que todas alcancen un determinado nivel de hallazgos. Además facilita la elección del parámetro  $\gamma$  que debe establecerse en un valor pequeño alrededor de 0,2; si el área explorada resultara escasa, las sucesivas etapas incrementales compensarán la situación ampliando la búsqueda convenientemente.

Los datos obtenidos durante la evaluación revelan que la estrategia de exploración incremental aplicada a la búsqueda inteligente, mejora notablemente el rendimiento de esta última, disminuyendo la cantidad de mensajes *Query* generados y manteniendo o mejorando la eficacia para alcanzar el nivel de hallazgos requerido.

Además, se ha comprobado que siempre es ventajoso utilizar la estrategia de exploración incremental independientemente de la disponibilidad de los recursos en la red. Esta ventaja se acentúa considerablemente cuando la cantidad de instancias por recurso presentes en la red es alta.

Por otra parte, se ha observado que el tiempo de respuesta asociado a la cantidad de etapas incrementales de las búsquedas, en la mayoría de los casos no resulta significativo. Ello se debe a que habitualmente no es necesario disparar demasiadas etapas incrementales para alcanzar el nivel de hallazgos requeridos.

Por todo lo expuesto anteriormente puede afirmarse que la estrategia de exploración incremental complementa convenientemente a la búsqueda inteligente. Así se completa la definición de la metodología BII-P2P, un sistema de búsqueda basado en la difusión selectiva de mensajes y ampliaciones sucesivas

del espacio de exploración que se caracteriza por el alto rendimiento y constituye el tema central de la presente tesis.



# Conclusiones y Líneas Futuras de Investigación

En esta tesis se ha analizado la problemática relacionada con la búsqueda de recursos en redes P2P. Partiendo de ese análisis, se ha propuesto, desarrollado y evaluado una solución original que se enmarca dentro del modelo P2P puro no estructurado y que supera ampliamente el rendimiento de la búsqueda convencional utilizada en esta clase de redes.

La búsqueda de recursos en los sistemas P2P puros, a diferencia de los modelos híbridos, se lleva a cabo sin ningún tipo de centralización. Así, estos sistemas se libran de situaciones indeseables presentes en los modelos híbridos como el desequilibrio de carga, la existencia de puntos críticos de falla, la congestión de tráfico y las limitaciones computacionales en relación a los servidores de búsqueda. Sin embargo, como contrapartida, la localización de los recursos es más difícil y constituye un problema a la espera de mejores soluciones.

Los sistemas P2P puros estructurados como *Chord*, nacidos en entornos académicos, resuelven el problema de la búsqueda registrando todos los recursos compartidos en un gran directorio global. Sin embargo, este directorio no se localiza en un sitio puntual de la red sino que se particiona y reparte en tablas de *hash* distribuidas (DHTs) entre los nodos que se organizan en una estructura anular. El rendimiento del sistema de búsqueda basado en DHTs es alto en tanto se mantenga la estructura operativa, estable y actualizada. Ello demanda, aún ante

cambios menores como la incorporación de un nuevo nodo a la red, una carga de trabajo considerable, asociada a la reasignación del espacio de claves y al mantenimiento de las tablas.

Los sistemas P2P puros no estructurados, de los cuales la red *Gnutella* 0.4 sigue siendo el referente más representativo, permiten la creación de entornos dinámicos dónde los nodos entran y salen de la red con mucha frecuencia sin que ello demande una carga de trabajo significativa. En este tipo de redes no existen servidores centrales, roles especializados, estructuras rígidas o responsabilidades globales, todos los integrantes cumplen las mismas funciones agrupándose libremente en vecindades y conectándose en una relación simétrica como pares o iguales.

Entre las ventajas más destacadas de los sistemas P2P puros no estructurados se encuentran la sencillez, robustez y tolerancia a fallas. Su naturaleza distribuida y la ausencia total de control central y de roles diferenciados promueven el equilibrio de carga entre todos los pares de la red. Ningún nodo es imprescindible para el funcionamiento del sistema que no tiene puntos de fallas sensibles. Son fáciles de implementar pues la ausencia de puntos de centralización y de estructuras particulares simplifican su administración. Sin embargo, todas estas ventajas se ven eclipsadas por la falta de un sistema de búsqueda de recursos realmente eficiente y escalable. La metodología BII-P2P —*Búsqueda Inteligente Incremental en redes P2P*— propuesta y desarrollada en esta tesis representa un avance en esa dirección.

La búsqueda de recursos en los sistemas P2P puros no estructurados como *Gnutella* 0.4 se implementa por medio de un sencillo algoritmo BFS que inunda la red con mensajes de peticiones. El nodo interesado en un recurso envía una petición a todos sus nodos vecinos quienes, a su vez, la reenvían a todos sus otros vecinos alcanzando así nodos más distantes. La utilización de un campo TTL limita el alcance de los mensajes evitando el colapso de la red pero circunscribe la



propagación de las solicitudes a los nodos relativamente cercanos, dentro del radio de distancia establecido por este valor. La búsqueda BFS de *Gnutella* no escala bien, perdiendo efectividad cuando los recursos son escasos y la red es grande.

Superar estas deficiencias en las búsquedas de los sistemas P2P puros no estructurados ha sido la motivación principal que impulsó la ejecución de las actividades relacionadas con el presente trabajo. Cumpliendo con el objetivo tempranamente planteado en esta tesis, se ha diseñado, implementado y evaluado la metodología de búsqueda BII-P2P basada en la difusión selectiva de mensajes, con muy buenos resultados.

BII-P2P utiliza en cada uno de los nodos del sistema P2P una red neuronal artificial. Ésta asiste al nodo propietario en el ruteo de peticiones para que, en función del recurso buscado y los perfiles aprendidos de cada vecino, la búsqueda sea conducida sólo hacia el subconjunto más apropiado de ellos, es decir, aquellos con mayor probabilidad de satisfacerla rápidamente. Así es posible aumentar la capacidad de localización de los recursos con exploraciones más profundas y al mismo tiempo disminuir el tráfico generado con exploraciones más angostas al descartar los caminos menos prometedores.

BII-P2P descubre las mejores rutas, más directas y con mayor cantidad de recursos, que conducen las peticiones desde el nodo iniciador de la búsqueda hasta aquellos poseedores del recurso buscado. Estas rutas quedan establecidas por medio de una secuencia de “buenas decisiones” tomadas localmente por todos los nodos involucrados en la trayectoria del mensaje de solicitud en el momento de su propagación. BII-P2P es una solución escalable que mejora significativamente el rendimiento del algoritmo BFS de *Gnutella* alcanzando nodos más lejanos, realizando más hallazgos y generando menor cantidad de tráfico sobre la red P2P.

La vasta experimentación llevada a cabo en el contexto de esta tesis, cuyos

resultados han sido oportunamente expuestos, ha demostrado que, con la parametrización correcta, BII-P2P supera ampliamente a la búsqueda BFS de *Gnutella* independientemente del tamaño de red sobre la que se implemente. Los parámetros TTL y  $\gamma$  de BII-P2P regulan la profundidad y el ancho de la exploración respectivamente, por lo que impactan significativamente en el rendimiento de la búsqueda. En este trabajo se ha presentado una gran cantidad de combinaciones de valores posibles para estos parámetros, todas ellas ventajosas.

Para que BII-P2P funcione correctamente es necesario mantener actualizado el conocimiento que los nodos poseen acerca de su entorno cercano. Este conocimiento consiste en qué clases de recursos pueden accederse fácilmente a partir de cada uno de sus vecinos. El entrenamiento tradicional de las redes neuronales ha sido reemplazado por un algoritmo de aprendizaje basado en el intercambio de información entre nodos vecinos que facilita y acelera muchas veces el proceso de adquisición del conocimiento. En relación a ello, se han definido y evaluado varias políticas que gobiernan la forma y la frecuencia con la que se producen estos intercambios de información.

Las pruebas realizadas demostraron que no es necesario propagar los mensajes de actualización de conocimiento hacia nodos alejados. Este resultado es sumamente conveniente pues favorece el ahorro de tráfico. La política  $DLP_2$ , que limita el alcance de los mensajes de actualización a los nodos que se encuentran a distancia 2 o menos, es la aconsejable en la mayoría de los casos.

La exploración incremental implementada en BII-P2P mejora aún más el rendimiento de la búsqueda inteligente y es la causa de una gran reducción de tráfico en la red P2P. Consiste en iniciar las búsquedas sobre una región acotada, profunda pero angosta, de la red P2P y, sólo en caso de no alcanzar los resultados esperados, ir ampliándola en sucesivas etapas incrementales manteniendo la profundidad pero haciendo cada vez más ancha la región explorada.

Otra ventaja de la exploración incremental es que facilita la elección del parámetro  $\gamma$  que, según resultados experimentales, debe establecerse en un valor pequeño cercano a 0,2. Si el área explorada resultara demasiado reducida, las sucesivas etapas incrementales compensarán la situación ensanchando la región convenientemente.

A diferencia de la búsqueda BFS de *Gnutella*, BII-P2P es escalable y se puede implementar en grandes redes manteniendo la eficacia y la eficiencia con niveles de tráfico reducidos. Ello se debe a que BII-P2P puede realizar exploraciones mucho más profundas que BFS utilizando un valor más alto para el parámetro TTL haciendo alcanzables todos los nodos de la red. De esta forma, BII-P2P se convierte en una solución escalable y de alto rendimiento para la búsqueda de recursos en sistemas P2P puros no estructurados. Sólo es necesario ajustar adecuadamente la profundidad y ancho de las búsquedas.

Durante el proceso de elaboración de esta tesis, llevado a cabo concurrentemente con el diseño, implementación y puesta a punto de la metodología de búsqueda BII-P2P, se consideraron distintas alternativas y esbozaron soluciones que finalmente no se emprendieron y fueron descartadas. Estas ideas que no llegaron a materializarse permiten dejar planteadas algunas líneas futuras de investigación. En otros casos, la obtención de buenos resultados que, sin embargo, no pueden considerarse concluyentes, animan el estudio más detallado de ciertos tópicos.

Se plantea como tarea de investigación futura la realización de mayor cantidad de pruebas y el estudio de nuevas alternativas para implementar eficientemente el aprendizaje en línea introducido en el capítulo 4. Los resultados obtenidos durante las pruebas realizadas sobre el enfoque híbrido de aprendizaje, que se presentaron en el capítulo 5, no fueron concluyentes, sin embargo pueden considerarse como resultados preliminares y un indicio de su factibilidad.

También se plantea el análisis de rendimiento de BII-P2P sobre topologías específicas con características distintivas. Todas las pruebas presentadas en este trabajo fueron realizadas sobre topologías regulares generadas aleatoriamente. Interesa evaluar el comportamiento de BII-P2P en presencia de ciertas irregularidades como por ejemplo la existencia de regiones con nodos altamente interconectados entre sí y otras con vecindarios reducidos.

Otra línea de investigación que se plantea sobre BII-P2P es el estudio minucioso acerca de los valores más adecuados para los parámetros TTL y  $\gamma$ , con el objetivo de implementar un algoritmo de ajuste automático, sin intervención del usuario, que garantice el máximo rendimiento de la búsqueda. Como objetivo preliminar se proyecta la confección de una guía práctica que asista al usuario que pone en funcionamiento al sistema en la correcta elección de estos parámetros.

También se plantea el estudio sobre una implementación de BII-P2P basada en la utilización de parámetros TTL y  $\gamma$  locales, es decir definidos individualmente y manipulados en cada nodo de la red. En esta solución, cada nodo que inicia una solicitud de búsqueda determina, de acuerdo a su experiencia pasada, el valor más adecuado para los parámetros TTL y  $\gamma$  para esa búsqueda en particular. Los demás nodos involucrados en la búsqueda deben respetar los valores elegidos por el nodo iniciador por lo que se hace necesario comunicar también el valor de  $\gamma$  en los mensajes de solicitud.

El concepto de búsqueda con exploración incremental introducido en BII-P2P abre un abanico de nuevas posibilidades que ameritan ser investigadas. Por lo tanto, para finalizar esta enumeración de líneas de investigación se propone el estudio de alternativas de estrategias incrementales que permitan, entre otras cosas, expandir la región explorada haciéndola cada vez más profunda. Se deja planteada esta propuesta junto al desafío de llevarla a cabo evitando la necesidad de repetir los recorridos transitados en etapas anteriores de la búsqueda.

# Bibliografía

- [1] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," *16th international conference on Supercomputing. ACM.*, pp. 84-95, 2002.
- [2] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti, "A local search mechanism for peer-to-peer networks," *The eleventh international conference on Information and knowledge management. ACM.* , pp. 300-307, 2002.
- [3] Beverly Yang and Hector Garcia-Molina, "Improving search in peer-to-peer networks," *22nd IEEE International Conference on Distributed Computing Systems*, pp. 5-14, 2002.
- [4] Dimitrios Tsoumakos and Nick Roussopoulos, "Adaptive Probabilistic Search for Peer-to-Peer Networks," *Proceedings. Third International Conference on. IEEE*, pp. 102-109, 2003.
- [5] Vapa M. et al., "NeuroSearch: evolutionary neural network resource discovery algorithm for Peer-to-Peer networks," *University of Jyväskylä*, 2004.
- [6] Jorge Ardenghi and Javier Echaiz, "Peer-to-Peer Systems: The Present and the Future.," *Journal of Computer Science & Technology JC&T*, vol. 7, no. 3, october 2007.

- [7] Yatin Chawthe, Sylvia Ratnasamy, Lee Breslau, Nick Lanham, and Scott Shenker, "Making Gnutella-like P2P systems scalable," *SIGCOMM'03, August 25-29, 2003, Karlsruhe, Germany. Copyright 2003 ACM 1-58133-735-4/03/0008*.
- [8] Leonardo Corbalán, Laura Lanzarini, and Armando De Giusti, "Resources NeuroSearch in Peer-to-Peer networks," *Information Technology Interfaces. Proceedings of the ITI 2009 31st International Conference on. IEEE*, pp. 597-602, June 2009.
- [9] Leonardo Corbalán, Laura Lanzarini, and Armando De Giusti, "Búsqueda Neuronal de Recursos con Exploración Incremental en Redes Peer-to-Peer," *Jornadas Chilenas de Computación. JCC 2009.*, pp. 11-20, Nov. 2009.
- [10] Leonardo Corbalán, Laura Lanzarini, and Armando De Giusti, "Búsqueda Neuronal de Recursos Adaptativa en Sistemas Peer-to-Peer (BNAP2P)," *XVI Congreso Argentino de Ciencias de la Computación CACIC 2010*, pp. 191-200, Oct. 2010.
- [11] Clay Shirky. (2000, noviembre) Open P2P. [Online]. <http://openp2p.com/pub/a/p2p/2000/11/24/shirky1-whatisp2p.html>
- [12] (1999) home page. [Online]. <http://setiathome.berkeley.edu>
- [13] Dejan Milojevic et al., "Peer-to-Peer Computing," *HP Laboratories Palo Alto*, 2003.
- [14] Javed I. Khan and Adam Wierzbicki, "Foundation of peer-to-peer computing," *Elsevier Journal of Computer Communication*, vol. 31, pp. 187-189, 2008.

- [15] Kai Guo and Zhijng Liu, "A New Efficient Hierarchical Distributed P2P Clustering Algorithm," *IEEE Proceedings of the Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 352-355, 2008.
- [16] Napster. (1999) Napster. [Online]. <http://www.napster.com/>
- [17] William Sears, Zhen Yu, and Yong Guan, "An Adaptive Reputation-based Trust Framework for Peer-to-Peer Applications," *Fourth IEEE International Symposium on Network Computing and Applications (NCA'05)*, pp. 1-8, 2005.
- [18] Nikta Dayhim, Amir Masoud Rahmani, and Sepideh Nazemi , "Towards a Multi-Agent Framework for Fault Tolerance and QoS Guarantee in P2P Networks," *IEEE Proceedings of the Third 2008 International Conference on Convergence and Hybrid Information Technology*, pp. 166-171, 2008.
- [19] Qian Zhang, Yu Sun, Zheng Liu, Xia Zhang, and Xuezh Wen, "Design of a Distributed P2P-based Grid Content Management Architecture," *IEEE Proceedings of the 3rd Annual Communication Networks and Services Research Conference (CNSR'05)*, pp. 1-6, 2005.
- [20] (2000) Freenet. [Online]. <https://freenetproject.org>
- [21] Nelson Minar and Marc Hedlund, "Peer-to-Peer Models Through the History of the Internet," in *Peer-to-Peer: Harnessing the Power of Disruptive Technologies.*: Oram, Andy, 2001.
- [22] Lawrence Lessig, *Free Culture. How Big Media uses Technology and the Law to lock.* New York: Penguin Press, 2004.
- [23] iMesh. [Online]. <http://www.imesh.com/>

- [24] Gnutella. (2000) The Annotated Gnutella Protocol Specification v0.4. [Online]. <http://rfc-gnutella.sourceforge.net/developer/stable/>
- [25] Eduard Aibar and Jairo Ares, "Un análisis socio-histórico de la controversia en torno a las redes P2P para el intercambio de música e imágenes," *X Congreso Español de Sociología. Sociología y sociedad en España: hace treinta años, dentro de treinta años.*, julio 2010.
- [26] Tor Klingberg and Raphael Manfredi. (2003) Souceforge. [Online]. [http://rfc-gnutella.sourceforge.net/src/rfc-0\\_6-draft.html](http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html)
- [27] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems," *International Conference on Distributed Systems Platforms (Middleware)*, pp. 329-350, 2001.
- [28] Ben Y Zhao, John D. Kubiatowicz, and Anthony D. Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area location and routing," *U. C. Berkeley Technical Report CB//CSD-01-1141*, 2001.
- [29] Silvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker, "A Scalable Content-Addressable Network," *ACM SIGCOMM 2001*, pp. 161-172, 2001.
- [30] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," *Proceedings of, ACM SIGCOMM*, pp. 149-160, 2001.
- [31] Petar Maymounkov and David Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric," *Peer-to-Peer Systems. Springer*, pp. 53-65, 2002.



- [32] Yoram Kulbak and Danny Bickson. (2005) DANSS (Distributed Algorithms, Networking and Secure Systems) Lab. [Online]. <http://www.cs.huji.ac.il/labs/danss/presentations/emule.pdf>
- [33] Bram Cohen, "Incentives Build Robustness in BitTorrent," *Workshop on Economics of Peer-to-Peer Systems, Berkeley*, 2003.
- [34] Steffen Gebert, Rastin Pries, Daniel Schlosser, and Klaus Heck, "Internet access traffic measurement and analysis," *Traffic Monitoring and Analysis. Springer Berlin Heidelberg.*, pp. 29-42, 2012.
- [35] Pawel Kopiczko, Wojciech Mazurczyk, and Krzysztof Szczypiorski, "StegTorrent: a Steganographic Method for the P2P File Sharing Service," *arXiv preprint arXiv:1303.4264*, 2013.
- [36] J.A. Pouwelse, P. Garbacki, D.H.J Epema, and H.J. Sip, "The bittorrent p2p file-sharing system: Measurements and analysis," *Peer-to-Peer Systems IV. Springer Berlin Heidelberg*, pp. 205-216, 2005.
- [37] SorceForge. [Online]. <http://aresgalaxy.sourceforge.net>
- [38] R. Bolla, R. Gaeta, A. Magnetto, M. Sciuto, and M. Sereno, "A measurement study supporting P2P file-sharing community models," *Computer Networks*, vol. 53, no. 4, pp. 485-500, 2009.
- [39] Haiying Shen, "An efficient and adaptive decentralized file replication algorithm in P2P file sharing systems," *Parallel and Distributed Systems, IEEE Transactions*, vol. 21, no. 6, pp. 827-840, 2010.
- [40] Ramesh Shahabdkar and Ramachandra V. Pujeri, "Hybrid framework for mitigating illegitimate Peer Nodes in Multimedia file sharing in P2P,"

- International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 1, pp. 263-271, 2012.
- [41] Mayank Raj, Krishna Kant, and Sajal K. Das, "Energy adaptive mechanism for p2p file sharing protocols," *Euro-Par 2012: Parallel Processing Workshops. Springer Berlin Heidelberg*, pp. 89-99, 2013.
- [42] Sandvine. [Online]. <https://www.sandvine.com/trends/global-internet-phenomena/>
- [43] (2004) PPLive. [Online]. <http://www.pptv.com/>
- [44] (2005) PPstream. [Online]. <http://www.ppstream.com/>
- [45] (2005) SopCast. [Online]. <http://www.sopcast.org/>
- [46] Alex Borges Vieira, Ana Paula Couto da Silva, Francisco Henrique, Goncalves Glauber, and Pedro de Carvalho Gomes, "Sopcast p2p live streaming: Live session traces and analysis," *4th ACM Multimedia Systems Conference*, pp. 125-130, 2013.
- [47] (2005) Tvants. [Online]. <http://tvants.es/>
- [48] Di Wu, Yi. Liang, Jian He, and Xiaojun Hei, "Balancing performance and fairness in p2p live video systems," *Circuits and Systems for Video Technology, IEEE Transactions on* , vol. 23, no. 6, pp. 1029 - 1039 , 2013.
- [49] Yi Sun, Yang Guo, Xiaobing Zhang, Zhenyu Li, and Kave Salamatian, "The case for P2P mobile video system over wireless broadband networks: A practical study of challenges for a mobile video provider," *IEEE Network*, vol. 27, no. 2, pp. 22-27, 2013.

- [50] Weijie Wu, John CS Lui, and Richard TB Ma, "On incentivizing upload capacity in P2P-VoD systems: Design, analysis and evaluation.," *Computer Networks*, vol. 57, no. 7, pp. 1674-1688, 2013.
- [51] Lidia Jordanova, Jordan Nenkov, and Dobri Dobrev, "Comparative Analysis between Centralized and P2P System Traffic in Terms of Providing VoD Service to Cable Television Network Subscribers," *IJCSNS International Journal of Computer Science and Network Security*, vol. 13, no. 9, pp. 29-34, 2013.
- [52] F. Liu, S. Shen, B. Li, B. Li, and H. Jin, "Cinematic-Quality VoD in a P2P Storage Cloud: Design, Implementation and Measurements," *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS/SUPPLEMENT*, vol. 31, no. 9, pp. 214 - 226 , 2013.
- [53] A. M. Riad, M. Elmogy, and A. I. Shehab, "A Framework for Cloud P2P VoD System based on User's Behavior Analysis," *International Journal of Computer Applications*, vol. 76, no. 6, pp. 20-26, 2013.
- [54] (2013) F-Talk. [Online]. <http://ftalk.armanasci.com>
- [55] (2003) Skype. [Online]. <http://www.skype.com/es/>
- [56] Saikat Guha and Neil Daswani, "An experimental study of the skype peer-to-peer voip system," *Computing and Information Science Technical Reports. Cornell University*, 2005.
- [57] Kai SHUANG and X. I. E. Jing, "Disaster Recovery Backup System for P2P Based VoIP Application.," *Journal of Computational Information Systems*, vol. 9, no. 20, pp. 8099-8109, 2013.

- [58] J. Gomes, M. Pereira, M. Freire, and P. Monteiro, "Identification of Peer-to-Peer VoIP Sessions Using Entropy and Codec Properties," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, no. 10, pp. 2004 - 2014 , 2013.
- [59] A. Ríos, A. J. González Cela, and J. Á. Alcober Segura, "Prototipo de un sistema P2P de multiconferencia basado en SIP," *XVIII Jornadas Telecom I+D-2008. Bilbao: Universidad del Pais Vasco*, pp. 1-4, 2008.
- [60] J. Rosenberg et al., "SIP: Session Initiation," *IETF RFC 3261*, June 2002.
- [61] Blue Talk. [Online]. [bluetalk.ca](http://bluetalk.ca)
- [62] Nouha Oualha, Jean Leneutre, and Yves Roudier, "Verifying remote data integrity in peer-to-peer data storage: A comprehensive survey of protocols," *Peer-to-Peer Networking Application*, vol. 4, pp. 1-11, 2011.
- [63] Lijiang Chen, Bin Cui, Hua Lu, Linhao Xu, and Quanqing Xu, "iSky: Efficient and Progressive Skyline Computing in a Structured P2P Network," *IEEE Proceedings of the 28th International Conference on Distributed Computing Systems*, pp. 160-169, 2008.
- [64] (2002) BONIC. [Online]. <http://boinc.berkeley.edu>
- [65] Ian Foster and Adriana Iamnitchi, "On death, taxes, and the convergence of peer-to-peer and grid computing," *Peer-to-Peer Systems II. Springer Berlin Heidelberg*, pp. 118-128, 2003.
- [66] Ian Brown and Yumin Oliver. Huang, "An Analysis of Peer-to-Peer Traffic over an Ad-Hoc Network," 2013.

- [67] Chun-Hsin Wang, Chia-Chun Lien, and Shao-Hu Lin, "Collaborative Applications Platform Based on Secure P2P Networks," *The Fifth International Conference on Advances in Future Internet*, pp. 84-89, 2013.
- [68] Tariq Rahim Soomro and Rab Nawaz, "Current Trends, Characteristics, Challenges and Future of P2P," *Lecture Notes on Information Theory*, vol. 1, no. 2, 2013.
- [69] Jigyasu Dubey and Vrinda Tokekar, "Identification of efficient peers in P2P computing system for real time applications," *arXiv preprint arXiv:1212.3074*, 2012.
- [70] Geoffrey Fox, "Peer-to-peer networks," *Computing in Science & Engineering*, vol. 3, no. 3, pp. 75-77, 2001.
- [71] P. Pankaj, M. Hyde, and J. A. Rodger, "P2P Business Applications: Future and Directions," *Communications and Network*, vol. 4, no. 3, 2012.
- [72] H. M. Xu, Y. J. Shi, Y. L. Liu, F. B. Gao, and T. Wan, "Integration of cloud computing and P2P: A future storage infrastructure," *Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE), 2012 International Conference on* , pp. 1489-1492, 2012.
- [73] O. Babaoglu, M. Marzolla, and M. Tamburini, "Design and implementation of a p2p cloud system," *27th Annual ACM Symposium on Applied Computing* , pp. 412-417, 2012.
- [74] Amir H Payberah, "Live Streaming in P2P and Hybrid P2P-Cloud Environments for the Open Internet," *Doctoral Thesis in Information and Communication Technology. KTH ROYAL INSTITUTE OF TECHNOLOGY*, pp. 1-107, 2013.

- [75] Jacob Chakareski, "Cost and profit driven cloud-P2P interaction," *Peer-to-Peer Networking and Applications*, pp. 1-16, 2013.
- [76] Marvin Minsky and Seymour Papert, *Perceptrons: An Introduction to Computational Geometry*. Cambridge: Mit Press, 1969.
- [77] Paul Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis.*: Harvard University, 1974.
- [78] Teuvo Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Ciber.*, pp. 59-69, 1982.
- [79] Teuvo Kohonen, *Self-Organizing Maps*, Third Edition ed.: Springer Verlag, 2000.
- [80] Teuvo Kohonen, *Self-Organization and Associative Memory*. New York: Springer-Verlag, 1984.
- [81] Robert Hecht-Nielsen, "Counterpropagation networks," *Applied Optics*, vol. 26 Issue 23, pp. 4979-4983, December 1987.
- [82] Robert Hecht-Nielsen, "Counterpropagation Networks," in *Proc. of the IEEE International Conference on Neural Networks*, New York, June 1987, pp. 19-32.
- [83] Robert Hecht-Nielsen, *Neurocomputing.*: Addison-Wesley Publishing Company, 1991.
- [84] Warren S. McCulloch and Walter Pitts, "A logical calculus of the ideas immanent in neurons activity," *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943.

- [85] James A. Freeman and David M. Skapura, *Redes neuronales Algoritmos, aplicaciones y técnicas de programación.*: Addison–Wesley Iberoamericana, 1993.
- [86] Frank Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psychological Review*, vol. 65:6, pp. 386-408, 1958.
- [87] Donald O. Hebb, *Organization of Behaviour*. New York: John Wiley & Sons, 1949.
- [88] Alejandro Flores Méndez and Eduardo Gómez Ramírez, "Las Redes Neuronales Celulares y su aplicación en el Procesamiento de Imágenes utilizando MATLAB," *IV Simposio Iberoamericano de Reconocimiento de Patrones.*, marzo 1996.
- [89] David E. Rumelhart, *Parallel Distributed Processing.*: MIT Press, 1986.
- [90] Openprocessing. [Online]. <http://www.openprocessing.org/sketch/54623>





# Apéndice A

En este apéndice se presentan los datos completos recogidos durante la experimentación llevada a cabo para evaluar el rendimiento de BII-P2P.

Las tablas A-1, ..., A-8 presentan los resultados obtenidos durante la experimentación realizada para evaluar la capacidad de búsqueda de la componente inteligente de BII-P2P sin exploración incremental.

Las tablas A-9, ..., A-22, presentan los resultados obtenidos durante la experimentación realizada para evaluar la capacidad de adaptación de BII-P2P bajo las distintas políticas de adquisición y actualización de conocimiento definidas.

Finalmente las tablas A-23, ..., A-30 presentan los resultados obtenidos durante la experimentación realizada para evaluar el rendimiento de la exploración incremental sobre distintos escenarios.



**Tabla A-1 Rendimiento de la búsqueda inteligente sobre red de 2.000 nodos**

Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
<i>Gnutella BFS (TTL=4)</i>	3913,4	1204,5		81,79	15,24		BII-P2P (TTL=12; $\gamma$ =0,55)	4000,3	70,09	0,6512	96,79	4	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,20)	8,35	4,57	< 0.0001	3,59	4,06	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,60)	5336,75	72,02	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,25)	15,25	9,16	< 0.0001	5,38	4,81	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,65)	5964,3	69,16	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,30)	30,38	16,92	< 0.0001	9,1	7,36	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,70)	7190,75	62,17	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,35)	46,9	24,32	< 0.0001	12,56	8,53	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,75)	8680,33	49,18	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,40)	83,55	44,69	< 0.0001	16,67	11,75	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,80)	9575,43	52,3	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,45)	125,13	59,35	< 0.0001	20,64	12,87	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,50)	254,65	105,64	< 0.0001	31,15	16,89	< 0.0001	BII-P2P (TTL=12; $\gamma$ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,55)	291,08	126,48	< 0.0001	32,56	16,58	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,20)	16,6	12,24	< 0.0001	9,1	8,64	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,60)	470,18	186,18	< 0.0001	39,36	17,72	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,25)	87,05	54,3	< 0.0001	32,31	18,2	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,65)	574,18	235,87	< 0.0001	43,33	19,07	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,30)	403,68	145,58	< 0.0001	65,9	18,1	0,0004
BII-P2P (TTL=4; $\gamma$ =0,70)	810,6	321,88	< 0.0001	49,49	20,55	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,35)	871,78	126,61	< 0.0001	84,23	10,2	0,2052
BII-P2P (TTL=4; $\gamma$ =0,75)	1159,65	406,54	< 0.0001	57,56	20,04	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,40)	1634,68	64,6	< 0.0001	91,41	6,32	0,0001
BII-P2P (TTL=4; $\gamma$ =0,80)	1396,1	485,26	< 0.0001	61,15	19,96	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,45)	2288,45	69,88	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,85)	1710,5	599,89	< 0.0001	65	18,58	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,50)	3646,95	72,11	0,1704	96,41	4,68	< 0.0001
BII-P2P (TTL=4; $\gamma$ =0,90)	2186,28	731,08	< 0.0001	70,26	19,07	0,0034	BII-P2P (TTL=13; $\gamma$ =0,55)	4006,7	68,34	0,6275	96,79	4	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,20)	10,35	6,33	< 0.0001	5,64	5,7	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,60)	5339,25	72,55	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,25)	22,55	15,52	< 0.0001	9,87	6,81	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,65)	5965,3	69,08	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,30)	53,1	31,58	< 0.0001	17,69	11,38	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,70)	7190,88	62,1	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,35)	93,25	49,77	< 0.0001	24,36	13,55	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,75)	8680,65	49,4	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,40)	192,43	101,01	< 0.0001	34,36	17,25	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,80)	9575,75	52,41	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,45)	321,1	150,18	< 0.0001	43,59	18,57	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,50)	765,45	278,03	< 0.0001	63,46	19,66	< 0.0001	BII-P2P (TTL=13; $\gamma$ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,55)	903,95	335	< 0.0001	65,38	19,4	0,0001	BII-P2P (TTL=14; $\gamma$ =0,20)	16,65	12,26	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,60)	1580,1	505,86	< 0.0001	78,33	14,14	0,4161	BII-P2P (TTL=14; $\gamma$ =0,25)	93,25	57,74	< 0.0001	34,62	20,33	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,65)	1982,93	653,19	< 0.0001	82,69	12,65	0,5899	BII-P2P (TTL=14; $\gamma$ =0,30)	433,48	143,02	< 0.0001	68,08	17,34	0,0017
BII-P2P (TTL=5; $\gamma$ =0,70)	2913,03	871,61	< 0.0001	88,72	10,83	0,0146	BII-P2P (TTL=14; $\gamma$ =0,35)	908,88	109,09	< 0.0001	84,49	9,44	0,1716
BII-P2P (TTL=5; $\gamma$ =0,75)	4266,53	1023,99	0,1618	94,62	6,11	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,40)	1654,18	56,97	< 0.0001	91,67	5,97	0,0001
BII-P2P (TTL=5; $\gamma$ =0,80)	5141,8	1183,98	< 0.0001	96,79	4,56	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,45)	2300,93	68,29	< 0.0001	94,1	5,93	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,85)	6303,95	1330,5	< 0.0001	98,08	3,75	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,50)	3650,2	71,5	0,1755	96,41	4,68	< 0.0001
BII-P2P (TTL=5; $\gamma$ =0,90)	7897,1	1583,29	< 0.0001	99,49	1,9	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,55)	4008,85	67,01	0,6196	96,79	4	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,20)	12,08	7,81	< 0.0001	6,03	6,34	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,60)	5339,6	72,63	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,25)	30,08	21,37	< 0.0001	12,18	8,8	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,65)	5965,6	69,04	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=6; $\gamma$ =0,30)	84,55	51,11	< 0.0001	22,44	14,15	< 0.0001	BII-P2P (TTL=14; $\gamma$ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=6;γ =0,35)	166,4	84	< 0.0001	32,69	16,58	< 0.0001	BII-P2P (TTL=14;γ =0,75)	8680,85	49,53	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=6;γ =0,40)	389,75	184,74	< 0.0001	51,15	20,49	< 0.0001	BII-P2P (TTL=14;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=6;γ =0,45)	695,08	282,81	< 0.0001	65,77	20,15	0,0004	BII-P2P (TTL=14;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=6;γ =0,50)	1721,38	434,05	< 0.0001	86,92	11,23	0,0504	BII-P2P (TTL=14;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=6;γ =0,55)	2041,65	492,63	< 0.0001	88,72	10,05	0,0097	BII-P2P (TTL=15;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=6;γ =0,60)	3409,2	596,41	0,0211	94,36	5,71	< 0.0001	BII-P2P (TTL=15;γ =0,25)	98,8	60,61	< 0.0001	36,41	20,9	< 0.0001
BII-P2P (TTL=6;γ =0,65)	4127,48	703,4	0,3354	95,64	4,95	< 0.0001	BII-P2P (TTL=15;γ =0,30)	457,48	138,76	< 0.0001	70,38	16	0,0059
BII-P2P (TTL=6;γ =0,70)	5627,35	723,54	< 0.0001	97,69	3,95	< 0.0001	BII-P2P (TTL=15;γ =0,35)	932,88	95,66	< 0.0001	85,26	8,46	0,0992
BII-P2P (TTL=6;γ =0,75)	7484,8	531,83	< 0.0001	98,85	2,48	< 0.0001	BII-P2P (TTL=15;γ =0,40)	1663,88	53,06	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=6;γ =0,80)	8541,03	474,97	< 0.0001	99,62	1,39	< 0.0001	BII-P2P (TTL=15;γ =0,45)	2307,33	67,76	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=6;γ =0,85)	9793,65	363,15	< 0.0001	99,87	0,99	< 0.0001	BII-P2P (TTL=15;γ =0,50)	3651,9	71,62	0,1783	96,41	4,68	< 0.0001
BII-P2P (TTL=6;γ =0,90)	11256	321,3	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=15;γ =0,55)	4010,13	66,89	0,6149	96,79	4	< 0.0001
BII-P2P (TTL=7;γ =0,20)	13,4	8,96	< 0.0001	6,79	7,1	< 0.0001	BII-P2P (TTL=15;γ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=7;γ =0,25)	38	26,57	< 0.0001	14,74	11,09	< 0.0001	BII-P2P (TTL=15;γ =0,65)	5965,75	68,88	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=7;γ =0,30)	124,13	72,92	< 0.0001	31,03	17,32	< 0.0001	BII-P2P (TTL=15;γ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=7;γ =0,35)	269,65	121,47	< 0.0001	45,64	18,78	< 0.0001	BII-P2P (TTL=15;γ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=7;γ =0,40)	673,68	258,26	< 0.0001	68,72	18,2	0,0036	BII-P2P (TTL=15;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=7;γ =0,45)	1189,48	365,85	< 0.0001	83,08	14,49	0,4672	BII-P2P (TTL=15;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=7;γ =0,50)	2679,6	348	< 0.0001	94,36	6,35	< 0.0001	BII-P2P (TTL=15;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=7;γ =0,55)	3085,63	344,85	0,0001	95,26	5,2	< 0.0001	BII-P2P (TTL=16;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=7;γ =0,60)	4613,33	292,54	0,0009	97,31	3,75	< 0.0001	BII-P2P (TTL=16;γ =0,25)	103,73	63,1	< 0.0001	37,56	21,29	< 0.0001
BII-P2P (TTL=7;γ =0,65)	5342,3	297,68	< 0.0001	97,69	3,41	< 0.0001	BII-P2P (TTL=16;γ =0,30)	476,23	132,92	< 0.0001	70,9	15,68	0,008
BII-P2P (TTL=7;γ =0,70)	6754,35	224,02	< 0.0001	98,21	2,81	< 0.0001	BII-P2P (TTL=16;γ =0,35)	949,15	85,26	< 0.0001	85,51	8,23	0,0807
BII-P2P (TTL=7;γ =0,75)	8394,35	135,55	< 0.0001	98,97	2,24	< 0.0001	BII-P2P (TTL=16;γ =0,40)	1669,33	52,3	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=7;γ =0,80)	9346,35	102,32	< 0.0001	99,74	1,25	< 0.0001	BII-P2P (TTL=16;γ =0,45)	2309,78	67,67	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=7;γ =0,85)	10396,8	61,42	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=16;γ =0,50)	3652,58	71,52	0,1794	96,41	4,68	< 0.0001
BII-P2P (TTL=7;γ =0,90)	11643,4	56,52	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=16;γ =0,55)	4011	66,82	0,6117	96,79	4	< 0.0001
BII-P2P (TTL=8;γ =0,20)	14,65	10,17	< 0.0001	7,44	7,34	< 0.0001	BII-P2P (TTL=16;γ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=8;γ =0,25)	46,85	32	< 0.0001	17,82	12,36	< 0.0001	BII-P2P (TTL=16;γ =0,65)	5965,88	68,77	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=8;γ =0,30)	171,1	96,21	< 0.0001	40,38	18,74	< 0.0001	BII-P2P (TTL=16;γ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=8;γ =0,35)	397,55	154,34	< 0.0001	60,64	17,91	< 0.0001	BII-P2P (TTL=16;γ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=8;γ =0,40)	980,2	278,22	< 0.0001	81,41	13,4	0,7398	BII-P2P (TTL=16;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=8;γ =0,45)	1633	334,32	< 0.0001	90,26	8,84	0,0013	BII-P2P (TTL=16;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=8;γ =0,50)	3222,58	198,41	0,0009	95,9	6,08	< 0.0001	BII-P2P (TTL=16;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=8;γ =0,55)	3620,65	185,69	0,1364	96,54	4,5	< 0.0001	BII-P2P (TTL=17;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=8;γ =0,60)	5078	128,43	< 0.0001	97,69	3,25	< 0.0001	BII-P2P (TTL=17;γ =0,25)	107,8	65,16	< 0.0001	39,1	22,22	< 0.0001
BII-P2P (TTL=8;γ =0,65)	5761,6	123,02	< 0.0001	98,08	2,8	< 0.0001	BII-P2P (TTL=17;γ =0,30)	490,43	126,71	< 0.0001	71,67	15,37	0,0128
BII-P2P (TTL=8;γ =0,70)	7064,18	84,5	< 0.0001	98,46	2,66	< 0.0001	BII-P2P (TTL=17;γ =0,35)	959,6	77,89	< 0.0001	85,51	8,23	0,0807
BII-P2P (TTL=8;γ =0,75)	8600,23	55,44	< 0.0001	98,97	2,24	< 0.0001	BII-P2P (TTL=17;γ =0,40)	1672,75	50,93	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=8;γ =0,80)	9518,48	45,61	< 0.0001	99,74	1,25	< 0.0001	BII-P2P (TTL=17;γ =0,45)	2311,15	67,14	< 0.0001	94,23	5,85	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=8;γ =0,85)	10496,6	48,23	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=17;γ =0,50)	3652,93	71,55	0,1799	96,41	4,68	< 0.0001
BII-P2P (TTL=8;γ =0,90)	11685,3	47,01	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=17;γ =0,55)	4011,23	66,79	0,6109	96,79	4	< 0.0001
BII-P2P (TTL=9;γ =0,20)	15,5	11,03	< 0.0001	8,59	8,44	< 0.0001	BII-P2P (TTL=17;γ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=9;γ =0,25)	55,58	37,24	< 0.0001	21,03	13,86	< 0.0001	BII-P2P (TTL=17;γ =0,65)	5965,88	68,77	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=9;γ =0,30)	222,5	116,8	< 0.0001	48,46	20,77	< 0.0001	BII-P2P (TTL=17;γ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=9;γ =0,35)	528,35	169,08	< 0.0001	69,62	16,38	0,0041	BII-P2P (TTL=17;γ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=9;γ =0,40)	1237,78	235,76	< 0.0001	85,77	10,31	0,0708	BII-P2P (TTL=17;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=9;γ =0,45)	1943,23	226,73	< 0.0001	92,56	6,88	< 0.0001	BII-P2P (TTL=17;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=9;γ =0,50)	3468,63	119,13	0,0253	96,41	4,68	< 0.0001	BII-P2P (TTL=17;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=9;γ =0,55)	3852,15	109,04	0,7504	96,67	4,19	< 0.0001	BII-P2P (TTL=18;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=9;γ =0,60)	5246,55	77,12	< 0.0001	97,69	3,25	< 0.0001	BII-P2P (TTL=18;γ =0,25)	111,28	66,99	< 0.0001	39,36	22,54	< 0.0001
BII-P2P (TTL=9;γ =0,65)	5898,68	72,28	< 0.0001	98,21	2,72	< 0.0001	BII-P2P (TTL=18;γ =0,30)	500,93	121,82	< 0.0001	72,69	15,51	0,0272
BII-P2P (TTL=9;γ =0,70)	7151,33	59,43	< 0.0001	98,46	2,66	< 0.0001	BII-P2P (TTL=18;γ =0,35)	965,28	72,4	< 0.0001	85,64	7,94	0,0725
BII-P2P (TTL=9;γ =0,75)	8656,45	44,71	< 0.0001	98,97	2,24	< 0.0001	BII-P2P (TTL=18;γ =0,40)	1674,43	50,76	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=9;γ =0,80)	9558,5	45,92	< 0.0001	99,74	1,25	< 0.0001	BII-P2P (TTL=18;γ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=9;γ =0,85)	10514,6	48,52	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=18;γ =0,50)	3653,05	71,46	0,1801	96,41	4,68	< 0.0001
BII-P2P (TTL=9;γ =0,90)	11690	47,52	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=18;γ =0,55)	4011,4	66,59	0,6103	96,79	4	< 0.0001
BII-P2P (TTL=10;γ =0,20)	16	11,56	< 0.0001	8,97	8,57	< 0.0001	BII-P2P (TTL=18;γ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=10;γ =0,25)	63,75	42,07	< 0.0001	24,74	14,06	< 0.0001	BII-P2P (TTL=18;γ =0,65)	5965,88	68,77	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=10;γ =0,30)	274,08	130,65	< 0.0001	54,62	20,47	< 0.0001	BII-P2P (TTL=18;γ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=10;γ =0,35)	649,9	168,74	< 0.0001	75,9	15,07	0,2044	BII-P2P (TTL=18;γ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=10;γ =0,40)	1419,03	173,84	< 0.0001	89,36	7,81	0,002	BII-P2P (TTL=18;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=10;γ =0,45)	2124,95	135,57	< 0.0001	93,59	6,35	< 0.0001	BII-P2P (TTL=18;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=10;γ =0,50)	3576,08	90,71	0,0851	96,41	4,68	< 0.0001	BII-P2P (TTL=18;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=10;γ =0,55)	3944,58	83,49	0,8711	96,67	4,19	< 0.0001	BII-P2P (TTL=19;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001
BII-P2P (TTL=10;γ =0,60)	5309,2	70,62	< 0.0001	97,69	3,25	< 0.0001	BII-P2P (TTL=19;γ =0,25)	114,53	68,65	< 0.0001	39,74	22,86	< 0.0001
BII-P2P (TTL=10;γ =0,65)	5945,05	67,98	< 0.0001	98,21	2,72	< 0.0001	BII-P2P (TTL=19;γ =0,30)	509,55	117,73	< 0.0001	73,08	15,47	0,038
BII-P2P (TTL=10;γ =0,70)	7180,7	60,86	< 0.0001	98,59	2,46	< 0.0001	BII-P2P (TTL=19;γ =0,35)	969,08	68,68	< 0.0001	85,77	8,04	0,0657
BII-P2P (TTL=10;γ =0,75)	8673,08	47,67	< 0.0001	98,97	2,24	< 0.0001	BII-P2P (TTL=19;γ =0,40)	1674,93	50,66	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=10;γ =0,80)	9571,4	49,63	< 0.0001	99,74	1,25	< 0.0001	BII-P2P (TTL=19;γ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=10;γ =0,85)	10518,2	48,69	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=19;γ =0,50)	3653,2	71,37	0,1804	96,41	4,68	< 0.0001
BII-P2P (TTL=10;γ =0,90)	11691,2	47,36	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=19;γ =0,55)	4011,5	66,48	0,6099	96,79	4	< 0.0001
BII-P2P (TTL=11;γ =0,20)	16,4	11,95	< 0.0001	8,97	8,57	< 0.0001	BII-P2P (TTL=19;γ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=11;γ =0,25)	71,78	46,32	< 0.0001	28,46	15,96	< 0.0001	BII-P2P (TTL=19;γ =0,65)	5965,88	68,77	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=11;γ =0,30)	323,98	140,99	< 0.0001	59,49	19,71	< 0.0001	BII-P2P (TTL=19;γ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=11;γ =0,35)	749,2	157,27	< 0.0001	79,74	14,23	0,8774	BII-P2P (TTL=19;γ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=11;γ =0,40)	1531,63	116,22	< 0.0001	90,64	6,56	0,0004	BII-P2P (TTL=19;γ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=11;γ =0,45)	2217,48	92,83	< 0.0001	94,1	5,93	< 0.0001	BII-P2P (TTL=19;γ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=11;γ =0,50)	3621,05	78,55	0,1336	96,41	4,68	< 0.0001	BII-P2P (TTL=19;γ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=11;γ =0,55)	3985,28	74,75	0,7084	96,67	4,19	< 0.0001	BII-P2P (TTL=20;γ =0,20)	16,7	12,3	< 0.0001	9,23	8,77	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 2.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=11; $\gamma$ =0,60)	5329,75	71,29	< 0.0001	97,69	3,25	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,25)	117,1	70,01	< 0.0001	40,38	23,18	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,65)	5959,83	68,15	< 0.0001	98,21	2,72	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,30)	515,7	114,83	< 0.0001	73,46	15,51	0,0514
BII-P2P (TTL=11; $\gamma$ =0,70)	7188,28	60,88	< 0.0001	98,59	2,46	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,35)	972,08	67,03	< 0.0001	85,9	7,78	0,0555
BII-P2P (TTL=11; $\gamma$ =0,75)	8678,53	48,62	< 0.0001	98,97	2,24	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,40)	1675,1	50,39	< 0.0001	91,79	5,92	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,80)	9573,88	51,08	< 0.0001	99,74	1,25	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,45)	2311,63	67,18	< 0.0001	94,23	5,85	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,50)	3653,3	71,31	0,1805	96,41	4,68	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,55)	4011,5	66,48	0,6099	96,79	4	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,20)	16,53	12,12	< 0.0001	9,1	8,64	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,60)	5339,73	72,75	< 0.0001	97,69	3,25	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,25)	79,63	50,24	< 0.0001	30,51	17	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,65)	5965,88	68,77	< 0.0001	98,21	2,72	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,30)	368,1	145,38	< 0.0001	63,21	18,2	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,70)	7190,98	62,06	< 0.0001	98,59	2,46	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	820,95	142,54	< 0.0001	82,82	11,85	0,4263	BII-P2P (TTL=20; $\gamma$ =0,75)	8681,08	49,71	< 0.0001	98,97	2,24	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	1597,13	82,17	< 0.0001	91,03	6,46	0,0002	BII-P2P (TTL=20; $\gamma$ =0,80)	9575,83	52,37	< 0.0001	99,74	1,25	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,45)	2265,35	74,52	< 0.0001	94,1	5,93	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,85)	10519	48,01	< 0.0001	100	0	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,50)	3638,6	73,49	0,1577	96,41	4,68	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,90)	11691,3	47,35	< 0.0001	100	0	< 0.0001

**Tabla A-2 Rendimiento de la búsqueda inteligente sobre red de 5.000 nodos**

Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
<i>Gnutella BFS (TTL=4)</i>	4284,25	1451,5		56,78	14,96		BII-P2P (TTL=12;γ =0,55)	9746,75	116,77	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=4;γ =0,20)	6,55	3,46	< 0.0001	1,43	1,83	< 0.0001	BII-P2P (TTL=12;γ =0,60)	13045,2	124,49	< 0.0001	97,85	2,54	< 0.0001
BII-P2P (TTL=4;γ =0,25)	12,7	6,63	< 0.0001	2,1	2,14	< 0.0001	BII-P2P (TTL=12;γ =0,65)	14601,5	111,72	< 0.0001	98,36	2,03	< 0.0001
BII-P2P (TTL=4;γ =0,30)	25,58	13,7	< 0.0001	3,07	2,52	< 0.0001	BII-P2P (TTL=12;γ =0,70)	17671,9	109,01	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=4;γ =0,35)	42,33	20,27	< 0.0001	4,51	3,67	< 0.0001	BII-P2P (TTL=12;γ =0,75)	21370,4	110,56	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=4;γ =0,40)	83,93	38,04	< 0.0001	6,76	4,46	< 0.0001	BII-P2P (TTL=12;γ =0,80)	23541,7	100,32	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=4;γ =0,45)	119,05	55,9	< 0.0001	7,89	5,29	< 0.0001	BII-P2P (TTL=12;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=4;γ =0,50)	254,68	96,8	< 0.0001	13,11	7,05	< 0.0001	BII-P2P (TTL=12;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=4;γ =0,55)	299,33	124,22	< 0.0001	14,13	7,41	< 0.0001	BII-P2P (TTL=13;γ =0,20)	11,85	11,58	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=4;γ =0,60)	454,88	192,2	< 0.0001	18,07	9,39	< 0.0001	BII-P2P (TTL=13;γ =0,25)	92,83	67,51	< 0.0001	16,08	11,82	< 0.0001
BII-P2P (TTL=4;γ =0,65)	591,5	251	< 0.0001	21,66	10,48	< 0.0001	BII-P2P (TTL=13;γ =0,30)	658,25	258,19	< 0.0001	55,25	16,41	0,6737
BII-P2P (TTL=4;γ =0,70)	841,25	345,5	< 0.0001	26,16	11,7	< 0.0001	BII-P2P (TTL=13;γ =0,35)	1863,4	441,87	< 0.0001	79,62	14,56	< 0.0001
BII-P2P (TTL=4;γ =0,75)	1225,1	470,88	< 0.0001	32,21	13,08	< 0.0001	BII-P2P (TTL=13;γ =0,40)	3765,63	391,53	0,0344	90,32	4,87	< 0.0001
BII-P2P (TTL=4;γ =0,80)	1463,13	553,97	< 0.0001	35,18	13,45	< 0.0001	BII-P2P (TTL=13;γ =0,45)	5490,3	204,46	< 0.0001	93,8	3,5	< 0.0001
BII-P2P (TTL=4;γ =0,85)	1849,43	692,79	< 0.0001	39,38	14,46	< 0.0001	BII-P2P (TTL=13;γ =0,50)	8933,25	116,36	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=4;γ =0,90)	2364,28	854,61	< 0.0001	43,83	14,81	0,0003	BII-P2P (TTL=13;γ =0,55)	9778,33	115,14	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=5;γ =0,20)	7,88	4,82	< 0.0001	1,79	1,84	< 0.0001	BII-P2P (TTL=13;γ =0,60)	13064,2	125,23	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=5;γ =0,25)	18,08	10,26	< 0.0001	3,33	2,78	< 0.0001	BII-P2P (TTL=13;γ =0,65)	14612,9	112,87	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=5;γ =0,30)	44,58	25,66	< 0.0001	5,73	3,93	< 0.0001	BII-P2P (TTL=13;γ =0,70)	17676,7	109,98	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=5;γ =0,35)	86,3	43,21	< 0.0001	9,22	6,2	< 0.0001	BII-P2P (TTL=13;γ =0,75)	21372,2	109,12	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=5;γ =0,40)	203,15	95,07	< 0.0001	16,33	7,97	< 0.0001	BII-P2P (TTL=13;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=5;γ =0,45)	326,13	153,14	< 0.0001	20,94	10,78	< 0.0001	BII-P2P (TTL=13;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=5;γ =0,50)	833,38	304,84	< 0.0001	37,43	13,1	< 0.0001	BII-P2P (TTL=13;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=5;γ =0,55)	1019,35	401,29	< 0.0001	41,01	14,22	< 0.0001	BII-P2P (TTL=14;γ =0,20)	11,98	11,8	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=5;γ =0,60)	1728,63	676,4	< 0.0001	52,43	16,65	0,2322	BII-P2P (TTL=14;γ =0,25)	103,98	75,42	< 0.0001	17,46	12,26	< 0.0001
BII-P2P (TTL=5;γ =0,65)	2343,6	915,83	< 0.0001	59,19	17,62	0,5101	BII-P2P (TTL=14;γ =0,30)	766,73	262,64	< 0.0001	60,16	16,09	0,3308
BII-P2P (TTL=5;γ =0,70)	3595,4	1339,41	0,0304	69,48	18,14	0,001	BII-P2P (TTL=14;γ =0,35)	2014,53	412,67	< 0.0001	81,21	11,98	< 0.0001
BII-P2P (TTL=5;γ =0,75)	5567,75	1886,05	0,0011	79,83	15,64	< 0.0001	BII-P2P (TTL=14;γ =0,40)	3873,35	287,29	0,0863	90,83	4,44	< 0.0001
BII-P2P (TTL=5;γ =0,80)	6850,88	2233,67	< 0.0001	84,02	12,62	< 0.0001	BII-P2P (TTL=14;γ =0,45)	5556,95	141,3	< 0.0001	93,86	3,55	< 0.0001
BII-P2P (TTL=5;γ =0,85)	8928,28	2773,78	< 0.0001	88,53	11,46	< 0.0001	BII-P2P (TTL=14;γ =0,50)	8953,58	115,93	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=5;γ =0,90)	11669,6	3412,33	< 0.0001	92,88	8,51	< 0.0001	BII-P2P (TTL=14;γ =0,55)	9788,8	114,9	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=6;γ =0,20)	9,05	6,57	< 0.0001	1,95	1,86	< 0.0001	BII-P2P (TTL=14;γ =0,60)	13071,8	125,96	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=6;γ =0,25)	24,78	15,33	< 0.0001	4,56	3,9	< 0.0001	BII-P2P (TTL=14;γ =0,65)	14616,6	113,3	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=6;γ =0,30)	73,05	43,55	< 0.0001	10,09	7,66	< 0.0001	BII-P2P (TTL=14;γ =0,70)	17678,3	109,96	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=6;γ =0,35)	166,55	84,61	< 0.0001	17,1	10,17	< 0.0001	BII-P2P (TTL=14;γ =0,75)	21373,1	108,01	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=6;γ =0,40)	455,1	206,32	< 0.0001	31,08	13,59	< 0.0001	BII-P2P (TTL=14;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=6;γ =0,45)	813,1	364,58	< 0.0001	42,91	17,87	0,0003	BII-P2P (TTL=14;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=6;γ =0,50)	2319,43	729,52	< 0.0001	69,59	16,25	0,0005	BII-P2P (TTL=14;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001

Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=6;γ =0,55)	2882,05	949,5	< 0.0001	73,73	16,58	< 0.0001	BII-P2P (TTL=15;γ =0,20)	12,08	11,94	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=6;γ =0,60)	5094,23	1519,05	0,017	84,69	11,37	< 0.0001	BII-P2P (TTL=15;γ =0,25)	115,53	83,52	< 0.0001	18,74	12,94	< 0.0001
BII-P2P (TTL=6;γ =0,65)	6769,05	1962,42	< 0.0001	89,66	9,56	< 0.0001	BII-P2P (TTL=15;γ =0,30)	866,68	254,93	< 0.0001	64,26	14,09	0,025
BII-P2P (TTL=6;γ =0,70)	10085,4	2548,55	< 0.0001	94,52	6,91	< 0.0001	BII-P2P (TTL=15;γ =0,35)	2126,6	364,55	< 0.0001	82,8	8,04	< 0.0001
BII-P2P (TTL=6;γ =0,75)	14654,6	2743,99	< 0.0001	97,44	3,75	< 0.0001	BII-P2P (TTL=15;γ =0,40)	3934,83	204,94	0,1394	91,19	4,13	< 0.0001
BII-P2P (TTL=6;γ =0,80)	17411,6	2646,28	< 0.0001	98,62	1,71	< 0.0001	BII-P2P (TTL=15;γ =0,45)	5590,43	114,64	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=6;γ =0,85)	21113,2	2626,62	< 0.0001	99,33	1,16	< 0.0001	BII-P2P (TTL=15;γ =0,50)	8962,23	117,38	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=6;γ =0,90)	25304,5	2439,05	< 0.0001	99,64	0,79	< 0.0001	BII-P2P (TTL=15;γ =0,55)	9793,28	115,32	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=7;γ =0,20)	9,9	7,96	< 0.0001	2,25	2,2	< 0.0001	BII-P2P (TTL=15;γ =0,60)	13073,8	125,66	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=7;γ =0,25)	32,53	21,17	< 0.0001	6,09	5,34	< 0.0001	BII-P2P (TTL=15;γ =0,65)	14617,6	113,28	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=7;γ =0,30)	113,95	68,56	< 0.0001	15,16	10,4	< 0.0001	BII-P2P (TTL=15;γ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=7;γ =0,35)	302,1	147,54	< 0.0001	29,24	14,64	< 0.0001	BII-P2P (TTL=15;γ =0,75)	21373,8	106,81	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=7;γ =0,40)	912,98	379,11	< 0.0001	50,95	18,22	0,1458	BII-P2P (TTL=15;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=7;γ =0,45)	1724,45	676,61	< 0.0001	66,31	20,88	0,0194	BII-P2P (TTL=15;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=7;γ =0,50)	4736,13	1043,07	0,1143	88,63	10,08	< 0.0001	BII-P2P (TTL=15;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=7;γ =0,55)	5704,15	1261,41	< 0.0001	90,58	8,59	< 0.0001	BII-P2P (TTL=16;γ =0,20)	12,13	11,97	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=7;γ =0,60)	9283,9	1395,31	< 0.0001	95,75	3,99	< 0.0001	BII-P2P (TTL=16;γ =0,25)	128,03	92,24	< 0.0001	20,89	14,22	< 0.0001
BII-P2P (TTL=7;γ =0,65)	11371,6	1518,76	< 0.0001	97,24	2,84	< 0.0001	BII-P2P (TTL=16;γ =0,30)	952,28	239,26	< 0.0001	67,54	12,81	0,0009
BII-P2P (TTL=7;γ =0,70)	15178,9	1440,9	< 0.0001	98,67	1,66	< 0.0001	BII-P2P (TTL=16;γ =0,35)	2208,2	300,78	< 0.0001	83,67	6,46	< 0.0001
BII-P2P (TTL=7;γ =0,75)	19734,1	815,06	< 0.0001	99,18	1,39	< 0.0001	BII-P2P (TTL=16;γ =0,40)	3972,23	151,43	0,1839	91,24	4,18	< 0.0001
BII-P2P (TTL=7;γ =0,80)	22259,2	588,98	< 0.0001	99,44	1,16	< 0.0001	BII-P2P (TTL=16;γ =0,45)	5606,93	107,02	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=7;γ =0,85)	25109,9	397,65	< 0.0001	99,64	0,87	< 0.0001	BII-P2P (TTL=16;γ =0,50)	8966,78	118,15	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=7;γ =0,90)	28410,6	247,46	< 0.0001	99,8	0,62	< 0.0001	BII-P2P (TTL=16;γ =0,55)	9795,58	114,98	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=8;γ =0,20)	10,55	9,17	< 0.0001	2,46	2,71	< 0.0001	BII-P2P (TTL=16;γ =0,60)	13074,3	125,61	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=8;γ =0,25)	41,1	27,51	< 0.0001	7,99	6,45	< 0.0001	BII-P2P (TTL=16;γ =0,65)	14617,7	113,36	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=8;γ =0,30)	169,28	99,71	< 0.0001	21,35	13,09	< 0.0001	BII-P2P (TTL=16;γ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=8;γ =0,35)	508,33	230,51	< 0.0001	40,96	17,62	< 0.0001	BII-P2P (TTL=16;γ =0,75)	21374,2	105,9	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=8;γ =0,40)	1572,93	553,95	< 0.0001	68,31	18,14	0,0019	BII-P2P (TTL=16;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=8;γ =0,45)	2906,58	910,94	< 0.0001	80,85	16,19	< 0.0001	BII-P2P (TTL=16;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=8;γ =0,50)	6823,35	816,7	< 0.0001	94,32	5,1	< 0.0001	BII-P2P (TTL=16;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=8;γ =0,55)	7873,55	899,14	< 0.0001	95,03	5,03	< 0.0001	BII-P2P (TTL=17;γ =0,20)	12,18	12	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=8;γ =0,60)	11645,5	611,98	< 0.0001	97,44	2,82	< 0.0001	BII-P2P (TTL=17;γ =0,25)	141,13	100,26	< 0.0001	22,63	15,16	< 0.0001
BII-P2P (TTL=8;γ =0,65)	13526,6	563,61	< 0.0001	98,21	2,13	< 0.0001	BII-P2P (TTL=17;γ =0,30)	1028,33	218,64	< 0.0001	70,05	10,97	< 0.0001
BII-P2P (TTL=8;γ =0,70)	16967,5	410,49	< 0.0001	98,98	1,45	< 0.0001	BII-P2P (TTL=17;γ =0,35)	2265,05	235,28	< 0.0001	84,69	5,13	< 0.0001
BII-P2P (TTL=8;γ =0,75)	20959,1	199,5	< 0.0001	99,23	1,38	< 0.0001	BII-P2P (TTL=17;γ =0,40)	3992,4	131,06	0,2127	91,24	4,18	< 0.0001
BII-P2P (TTL=8;γ =0,80)	23233,7	139,12	< 0.0001	99,49	1,14	< 0.0001	BII-P2P (TTL=17;γ =0,45)	5615,85	104,77	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=8;γ =0,85)	25707,9	101,29	< 0.0001	99,64	0,87	< 0.0001	BII-P2P (TTL=17;γ =0,50)	8968,68	118,52	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=8;γ =0,90)	28716,3	73,75	< 0.0001	99,8	0,62	< 0.0001	BII-P2P (TTL=17;γ =0,55)	9796,5	114,84	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=9;γ =0,20)	11,03	10,07	< 0.0001	2,51	2,94	< 0.0001	BII-P2P (TTL=17;γ =0,60)	13074,5	125,32	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=9;γ =0,25)	50,65	34,73	< 0.0001	9,22	7,06	< 0.0001	BII-P2P (TTL=17;γ =0,65)	14617,7	113,36	< 0.0001	98,41	2,05	< 0.0001

Continúa la tabla...



Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=9;γ =0,30)	241,4	136,39	< 0.0001	28,93	15,15	< 0.0001	BII-P2P (TTL=17;γ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=9;γ =0,35)	782,4	314,61	< 0.0001	54,12	19,58	0,5802	BII-P2P (TTL=17;γ =0,75)	21374,4	105,55	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=9;γ =0,40)	2281,55	641,65	< 0.0001	79,11	15,74	< 0.0001	BII-P2P (TTL=17;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=9;γ =0,45)	3944,28	911,95	0,2142	87,97	11,42	< 0.0001	BII-P2P (TTL=17;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=9;γ =0,50)	7990,18	423,13	< 0.0001	95,39	4,17	< 0.0001	BII-P2P (TTL=17;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=9;γ =0,55)	8981,28	430,28	< 0.0001	96,06	3,83	< 0.0001	BII-P2P (TTL=18;γ =0,20)	12,23	12,04	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=9;γ =0,60)	12556,9	246,4	< 0.0001	97,64	2,75	< 0.0001	BII-P2P (TTL=18;γ =0,25)	153,73	107,16	< 0.0001	24,17	15,9	< 0.0001
BII-P2P (TTL=9;γ =0,65)	14246,3	214,58	< 0.0001	98,31	2,06	< 0.0001	BII-P2P (TTL=18;γ =0,30)	1092,28	193,8	< 0.0001	71,63	9,86	< 0.0001
BII-P2P (TTL=9;γ =0,70)	17468,3	143,38	< 0.0001	98,98	1,45	< 0.0001	BII-P2P (TTL=18;γ =0,35)	2306,28	176,44	< 0.0001	84,95	5,13	< 0.0001
BII-P2P (TTL=9;γ =0,75)	21260,1	115,69	< 0.0001	99,28	1,27	< 0.0001	BII-P2P (TTL=18;γ =0,40)	4003	121,18	0,2292	91,3	4,24	< 0.0001
BII-P2P (TTL=9;γ =0,80)	23461	94,34	< 0.0001	99,49	1,14	< 0.0001	BII-P2P (TTL=18;γ =0,45)	5619,58	104,58	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=9;γ =0,85)	25820,1	83,72	< 0.0001	99,64	0,87	< 0.0001	BII-P2P (TTL=18;γ =0,50)	8968,83	118,58	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=9;γ =0,90)	28758	62,58	< 0.0001	99,8	0,62	< 0.0001	BII-P2P (TTL=18;γ =0,55)	9796,7	114,95	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=10;γ =0,20)	11,33	10,55	< 0.0001	2,71	3,5	< 0.0001	BII-P2P (TTL=18;γ =0,60)	13074,5	125,32	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=10;γ =0,25)	60,6	42,69	< 0.0001	11,06	8,72	< 0.0001	BII-P2P (TTL=18;γ =0,65)	14617,7	113,36	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=10;γ =0,30)	330,3	175,82	< 0.0001	35,74	17,06	< 0.0001	BII-P2P (TTL=18;γ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=10;γ =0,35)	1095,18	387,11	< 0.0001	65,13	19,22	0,0292	BII-P2P (TTL=18;γ =0,75)	21374,5	105,41	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=10;γ =0,40)	2878,23	634,11	< 0.0001	84,74	13,72	< 0.0001	BII-P2P (TTL=18;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=10;γ =0,45)	4667,4	749,6	0,1434	90,68	8,52	< 0.0001	BII-P2P (TTL=18;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=10;γ =0,50)	8534,23	206,17	< 0.0001	95,85	3,63	< 0.0001	BII-P2P (TTL=18;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=10;γ =0,55)	9464,6	206,87	< 0.0001	96,31	3,48	< 0.0001	BII-P2P (TTL=19;γ =0,20)	12,25	12,06	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=10;γ =0,60)	12876,2	139,33	< 0.0001	97,75	2,73	< 0.0001	BII-P2P (TTL=19;γ =0,25)	165,78	113,05	< 0.0001	25,81	16,25	< 0.0001
BII-P2P (TTL=10;γ =0,65)	14487,2	126,03	< 0.0001	98,36	2,03	< 0.0001	BII-P2P (TTL=19;γ =0,30)	1143,18	168,64	< 0.0001	73,27	8,26	< 0.0001
BII-P2P (TTL=10;γ =0,70)	17613,9	103,23	< 0.0001	98,98	1,45	< 0.0001	BII-P2P (TTL=19;γ =0,35)	2334,78	132,98	< 0.0001	85,36	5,18	< 0.0001
BII-P2P (TTL=10;γ =0,75)	21338,4	113,03	< 0.0001	99,28	1,27	< 0.0001	BII-P2P (TTL=19;γ =0,40)	4008,35	118,35	0,238	91,3	4,24	< 0.0001
BII-P2P (TTL=10;γ =0,80)	23518,3	97,2	< 0.0001	99,49	1,14	< 0.0001	BII-P2P (TTL=19;γ =0,45)	5621,15	104,15	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=10;γ =0,85)	25844	82,99	< 0.0001	99,64	0,87	< 0.0001	BII-P2P (TTL=19;γ =0,50)	8969,03	118,57	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=10;γ =0,90)	28763,5	63,44	< 0.0001	99,8	0,62	< 0.0001	BII-P2P (TTL=19;γ =0,55)	9796,88	114,96	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=11;γ =0,20)	11,53	11	< 0.0001	2,76	3,62	< 0.0001	BII-P2P (TTL=19;γ =0,60)	13074,5	125,32	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=11;γ =0,25)	70,63	50,53	< 0.0001	12,7	9,8	< 0.0001	BII-P2P (TTL=19;γ =0,65)	14617,7	113,36	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=11;γ =0,30)	432,28	211,25	< 0.0001	43,63	18,09	0,0009	BII-P2P (TTL=19;γ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=11;γ =0,35)	1400,73	433,93	< 0.0001	72,3	18,07	< 0.0001	BII-P2P (TTL=19;γ =0,75)	21374,5	105,41	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=11;γ =0,40)	3302,13	576,86	0,0002	87,66	10,36	< 0.0001	BII-P2P (TTL=19;γ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=11;γ =0,45)	5107,85	526,08	0,0015	92,93	3,8	< 0.0001	BII-P2P (TTL=19;γ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=11;γ =0,50)	8779,75	130,35	< 0.0001	96,01	3,4	< 0.0001	BII-P2P (TTL=19;γ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001
BII-P2P (TTL=11;γ =0,55)	9666,88	130,08	< 0.0001	96,31	3,48	< 0.0001	BII-P2P (TTL=20;γ =0,20)	12,28	12,09	< 0.0001	2,87	3,87	< 0.0001
BII-P2P (TTL=11;γ =0,60)	12998	125,63	< 0.0001	97,8	2,71	< 0.0001	BII-P2P (TTL=20;γ =0,25)	177,73	118,05	< 0.0001	27,09	16,67	< 0.0001
BII-P2P (TTL=11;γ =0,65)	14570,6	114,84	< 0.0001	98,36	2,03	< 0.0001	BII-P2P (TTL=20;γ =0,30)	1183,4	146,22	< 0.0001	74,45	7,47	< 0.0001
BII-P2P (TTL=11;γ =0,70)	17658,5	106,26	< 0.0001	98,98	1,45	< 0.0001	BII-P2P (TTL=20;γ =0,35)	2352,68	103,81	< 0.0001	85,36	5,18	< 0.0001
BII-P2P (TTL=11;γ =0,75)	21362,4	112,25	< 0.0001	99,28	1,27	< 0.0001	BII-P2P (TTL=20;γ =0,40)	4011,38	116,8	0,243	91,3	4,24	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 5.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=11; $\gamma$ =0,80)	23537,6	100	< 0.0001	99,49	1,14	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,45)	5621,75	103,57	< 0.0001	93,96	3,53	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,85)	25849,9	82,53	< 0.0001	99,64	0,87	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,50)	8969,18	118,53	< 0.0001	96,01	3,4	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,55)	9797,03	114,98	< 0.0001	96,31	3,48	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,20)	11,7	11,29	< 0.0001	2,87	3,87	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,60)	13074,5	125,32	< 0.0001	97,9	2,57	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,25)	81,9	59,48	< 0.0001	14,44	11,41	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,65)	14617,7	113,36	< 0.0001	98,41	2,05	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,30)	545,43	241,39	< 0.0001	50,18	17,22	0,083	BII-P2P (TTL=20; $\gamma$ =0,70)	17678,9	109,97	< 0.0001	98,98	1,45	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	1662,43	451,38	< 0.0001	76,91	16,4	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,75)	21374,5	105,41	< 0.0001	99,28	1,27	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	3587,13	498,91	0,006	89,5	6,72	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,80)	23542,4	100,42	< 0.0001	99,49	1,14	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,45)	5359,2	328,52	< 0.0001	93,45	3,57	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,85)	25851,1	82,86	< 0.0001	99,64	0,87	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,50)	8884,65	116,83	< 0.0001	96,01	3,4	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,90)	28764,7	64,26	< 0.0001	99,8	0,62	< 0.0001

**Tabla A-3 Rendimiento de la búsqueda inteligente sobre red de 10.000 nodos**

Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
<i>Gnutella BFS (TTL=4)</i>	4525,53	1297,2		36,14	9,26		BII-P2P (TTL=12;γ=0,55)	19231,7	177,88	< 0.0001	96,22	2,67	< 0.0001
BII-P2P (TTL=4;γ=0,20)	6,78	3,48	< 0.0001	0,53	0,72	< 0.0001	BII-P2P (TTL=12;γ=0,60)	25901,8	141,65	< 0.0001	97,53	2,12	< 0.0001
BII-P2P (TTL=4;γ=0,25)	13,3	7,52	< 0.0001	0,98	1,1	< 0.0001	BII-P2P (TTL=12;γ=0,65)	29073,2	135,44	< 0.0001	97,98	1,88	< 0.0001
BII-P2P (TTL=4;γ=0,30)	27,53	14,03	< 0.0001	1,59	1,42	< 0.0001	BII-P2P (TTL=12;γ=0,70)	35103,9	146,65	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=4;γ=0,35)	42,08	21,46	< 0.0001	2,24	2,1	< 0.0001	BII-P2P (TTL=12;γ=0,75)	42446,7	115,67	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=4;γ=0,40)	82,53	38	< 0.0001	3,43	2,28	< 0.0001	BII-P2P (TTL=12;γ=0,80)	46841,9	131,2	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=4;γ=0,45)	122,43	51,36	< 0.0001	4,36	2,41	< 0.0001	BII-P2P (TTL=12;γ=0,85)	51382,1	123,14	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=4;γ=0,50)	250,5	96,86	< 0.0001	7,14	3,3	< 0.0001	BII-P2P (TTL=12;γ=0,90)	57235,8	112,83	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=4;γ=0,55)	288,35	120,64	< 0.0001	7,59	3,83	< 0.0001	BII-P2P (TTL=13;γ=0,20)	13,85	11,06	< 0.0001	1,51	1,38	< 0.0001
BII-P2P (TTL=4;γ=0,60)	476,88	167,37	< 0.0001	10,37	4,4	< 0.0001	BII-P2P (TTL=13;γ=0,25)	114,35	104,57	< 0.0001	9,46	7,08	< 0.0001
BII-P2P (TTL=4;γ=0,65)	589	211,6	< 0.0001	11,68	4,75	< 0.0001	BII-P2P (TTL=13;γ=0,30)	990,1	447,66	< 0.0001	47,84	16,01	0,0002
BII-P2P (TTL=4;γ=0,70)	829,13	296,23	< 0.0001	14,02	5,63	< 0.0001	BII-P2P (TTL=13;γ=0,35)	3159,85	666,55	< 0.0001	76,77	8,12	< 0.0001
BII-P2P (TTL=4;γ=0,75)	1202,13	406,12	< 0.0001	17,35	6,2	< 0.0001	BII-P2P (TTL=13;γ=0,40)	7288,13	311,98	< 0.0001	90,06	3,41	< 0.0001
BII-P2P (TTL=4;γ=0,80)	1469,58	478,35	< 0.0001	19,7	6,45	< 0.0001	BII-P2P (TTL=13;γ=0,45)	10805	212,13	< 0.0001	93,19	3,28	< 0.0001
BII-P2P (TTL=4;γ=0,85)	1821,1	636,48	< 0.0001	21,97	7,74	< 0.0001	BII-P2P (TTL=13;γ=0,50)	17684,7	166,55	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=4;γ=0,90)	2417,35	777,29	< 0.0001	26,05	8,23	< 0.0001	BII-P2P (TTL=13;γ=0,55)	19339,7	155,93	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=5;γ=0,20)	8,4	4,71	< 0.0001	0,78	0,92	< 0.0001	BII-P2P (TTL=13;γ=0,60)	25952,8	146,02	< 0.0001	97,55	2,12	< 0.0001
BII-P2P (TTL=5;γ=0,25)	19,68	13,38	< 0.0001	1,51	1,44	< 0.0001	BII-P2P (TTL=13;γ=0,65)	29105,2	131,42	< 0.0001	97,98	1,88	< 0.0001
BII-P2P (TTL=5;γ=0,30)	48,83	26,62	< 0.0001	3,23	2,21	< 0.0001	BII-P2P (TTL=13;γ=0,70)	35117,2	147,85	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=5;γ=0,35)	86,4	46,94	< 0.0001	5,17	3,68	< 0.0001	BII-P2P (TTL=13;γ=0,75)	42454,5	115,79	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=5;γ=0,40)	202,33	96,69	< 0.0001	9,66	5,11	< 0.0001	BII-P2P (TTL=13;γ=0,80)	46846,5	132,81	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=5;γ=0,45)	338,18	146,12	< 0.0001	12,79	6,06	< 0.0001	BII-P2P (TTL=13;γ=0,85)	51382,8	123,62	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=5;γ=0,50)	846,2	327,23	< 0.0001	23,03	8,74	< 0.0001	BII-P2P (TTL=13;γ=0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=5;γ=0,55)	1017,25	422,27	< 0.0001	25,15	9,65	< 0.0001	BII-P2P (TTL=14;γ=0,20)	13,98	11,3	< 0.0001	1,56	1,43	< 0.0001
BII-P2P (TTL=5;γ=0,60)	1908,43	656,76	< 0.0001	35,23	10,43	0,6439	BII-P2P (TTL=14;γ=0,25)	131,28	120,11	< 0.0001	11,02	8,11	< 0.0001
BII-P2P (TTL=5;γ=0,65)	2487,18	865,11	< 0.0001	40,05	11,66	0,1231	BII-P2P (TTL=14;γ=0,30)	1204,2	488,48	< 0.0001	53,49	15,97	< 0.0001
BII-P2P (TTL=5;γ=0,70)	3817,75	1305,74	0,0173	49	12,27	< 0.0001	BII-P2P (TTL=14;γ=0,35)	3608,78	541,54	0,0001	80,61	6,15	< 0.0001
BII-P2P (TTL=5;γ=0,75)	6018,88	1888,16	0,0001	59,67	12,21	< 0.0001	BII-P2P (TTL=14;γ=0,40)	7579,78	223,88	< 0.0001	90,54	3,4	< 0.0001
BII-P2P (TTL=5;γ=0,80)	7664,2	2294,08	< 0.0001	65,85	12,17	< 0.0001	BII-P2P (TTL=14;γ=0,45)	10979,7	175,84	< 0.0001	93,27	3,31	< 0.0001
BII-P2P (TTL=5;γ=0,85)	9955,9	3116,29	< 0.0001	72,13	12,54	< 0.0001	BII-P2P (TTL=14;γ=0,50)	17750,1	162,52	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=5;γ=0,90)	13797,8	3931,24	< 0.0001	79,97	11,56	< 0.0001	BII-P2P (TTL=14;γ=0,55)	19383,2	153,95	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=6;γ=0,20)	9,78	6,04	< 0.0001	0,93	1,01	< 0.0001	BII-P2P (TTL=14;γ=0,60)	25970,3	147,43	< 0.0001	97,55	2,12	< 0.0001
BII-P2P (TTL=6;γ=0,25)	27,18	20,66	< 0.0001	2,12	1,78	< 0.0001	BII-P2P (TTL=14;γ=0,65)	29115,5	130,86	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=6;γ=0,30)	80,95	45,99	< 0.0001	5,5	3,48	< 0.0001	BII-P2P (TTL=14;γ=0,70)	35121,6	149,95	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=6;γ=0,35)	168,63	94,49	< 0.0001	9,61	6	< 0.0001	BII-P2P (TTL=14;γ=0,75)	42456,5	116,6	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=6;γ=0,40)	469,03	223,35	< 0.0001	19,95	8,54	< 0.0001	BII-P2P (TTL=14;γ=0,80)	46847,8	132,99	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=6;γ=0,45)	885,63	375,2	< 0.0001	28,95	11,32	0,0021	BII-P2P (TTL=14;γ=0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=6;γ=0,50)	2606,33	932,32	< 0.0001	52,31	13,67	< 0.0001	BII-P2P (TTL=14;γ=0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001

Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=6;γ =0,55)	3232,43	1219,37	< 0.0001	56,37	14,63	< 0.0001	BII-P2P (TTL=15;γ =0,20)	14,1	11,56	< 0.0001	1,56	1,43	< 0.0001
BII-P2P (TTL=6;γ =0,60)	6543,43	1904,69	< 0.0001	74,4	10,34	< 0.0001	BII-P2P (TTL=15;γ =0,25)	148,9	135,06	< 0.0001	12,28	8,96	< 0.0001
BII-P2P (TTL=6;γ =0,65)	8692,18	2460,19	< 0.0001	80,5	9,34	< 0.0001	BII-P2P (TTL=15;γ =0,30)	1413,23	510,09	< 0.0001	58,66	15,76	< 0.0001
BII-P2P (TTL=6;γ =0,70)	13673,1	3558,31	< 0.0001	88,8	7,44	< 0.0001	BII-P2P (TTL=15;γ =0,35)	3953,23	411,06	0,0107	82,6	5,42	< 0.0001
BII-P2P (TTL=6;γ =0,75)	21404,5	4536,95	< 0.0001	94,83	4,06	< 0.0001	BII-P2P (TTL=15;γ =0,40)	7744,1	179,99	< 0.0001	90,72	3,36	< 0.0001
BII-P2P (TTL=6;γ =0,80)	26737,3	4855,98	< 0.0001	97,28	2,08	< 0.0001	BII-P2P (TTL=15;γ =0,45)	11066,7	167,79	< 0.0001	93,27	3,31	< 0.0001
BII-P2P (TTL=6;γ =0,85)	33522,3	5567,7	< 0.0001	98,46	1,43	< 0.0001	BII-P2P (TTL=15;γ =0,50)	17778,8	162,98	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=6;γ =0,90)	43214,4	5959,69	< 0.0001	99,42	0,67	< 0.0001	BII-P2P (TTL=15;γ =0,55)	19403,4	152,45	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=7;γ =0,20)	10,88	7,03	< 0.0001	1,13	1,12	< 0.0001	BII-P2P (TTL=15;γ =0,60)	25976	148,35	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=7;γ =0,25)	35,63	28,92	< 0.0001	2,88	2,04	< 0.0001	BII-P2P (TTL=15;γ =0,65)	29119,1	131,22	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=7;γ =0,30)	129,13	75,16	< 0.0001	8,8	5,62	< 0.0001	BII-P2P (TTL=15;γ =0,70)	35123,7	150,37	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=7;γ =0,35)	315,43	176,38	< 0.0001	16,24	9,48	< 0.0001	BII-P2P (TTL=15;γ =0,75)	42457,4	116,56	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=7;γ =0,40)	1014,53	455,75	< 0.0001	35,96	13,66	0,8763	BII-P2P (TTL=15;γ =0,80)	46848,4	132,97	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=7;γ =0,45)	2100,53	798,79	< 0.0001	52,38	13,96	< 0.0001	BII-P2P (TTL=15;γ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=7;γ =0,50)	6475,7	1796,24	< 0.0001	80,25	10,08	< 0.0001	BII-P2P (TTL=15;γ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=7;γ =0,55)	7994,03	2199,13	< 0.0001	83,51	9,03	< 0.0001	BII-P2P (TTL=16;γ =0,20)	14,18	11,71	< 0.0001	1,59	1,47	< 0.0001
BII-P2P (TTL=7;γ =0,60)	15157	2521,33	< 0.0001	93,27	3,58	< 0.0001	BII-P2P (TTL=16;γ =0,25)	166,73	150,06	< 0.0001	13,27	9,69	< 0.0001
BII-P2P (TTL=7;γ =0,65)	19150,1	2782,19	< 0.0001	95,41	2,96	< 0.0001	BII-P2P (TTL=16;γ =0,30)	1605,65	513,95	< 0.0001	62,57	15,13	< 0.0001
BII-P2P (TTL=7;γ =0,70)	27075,8	2832,75	< 0.0001	97,91	1,88	< 0.0001	BII-P2P (TTL=16;γ =0,35)	4197,63	297,65	0,1265	83,81	4,94	< 0.0001
BII-P2P (TTL=7;γ =0,75)	36739,3	2315,47	< 0.0001	98,94	1,11	< 0.0001	BII-P2P (TTL=16;γ =0,40)	7833,28	161,65	< 0.0001	90,87	3,25	< 0.0001
BII-P2P (TTL=7;γ =0,80)	42380,1	1583,86	< 0.0001	99,34	0,81	< 0.0001	BII-P2P (TTL=16;γ =0,45)	11110,2	166,79	< 0.0001	93,27	3,31	< 0.0001
BII-P2P (TTL=7;γ =0,85)	48500,5	1213,54	< 0.0001	99,6	0,6	< 0.0001	BII-P2P (TTL=16;γ =0,50)	17791,3	162,74	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=7;γ =0,90)	55835,6	774,9	< 0.0001	99,82	0,36	< 0.0001	BII-P2P (TTL=16;γ =0,55)	19410,4	152,68	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=8;γ =0,20)	11,88	8,1	< 0.0001	1,34	1,23	< 0.0001	BII-P2P (TTL=16;γ =0,60)	25977,4	148,42	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=8;γ =0,25)	45,45	38,14	< 0.0001	3,76	2,56	< 0.0001	BII-P2P (TTL=16;γ =0,65)	29119,6	131,37	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=8;γ =0,30)	198,38	116,62	< 0.0001	13,11	7,32	< 0.0001	BII-P2P (TTL=16;γ =0,70)	35124,5	150,55	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=8;γ =0,35)	556,28	297,29	< 0.0001	26,36	13,18	0,0002	BII-P2P (TTL=16;γ =0,75)	42457,7	116,87	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=8;γ =0,40)	1967,95	765,71	< 0.0001	55,51	13,95	< 0.0001	BII-P2P (TTL=16;γ =0,80)	46848,6	132,89	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=8;γ =0,45)	4170,58	1213,42	0,2101	74,93	9,39	< 0.0001	BII-P2P (TTL=16;γ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=8;γ =0,50)	11366	1853,64	< 0.0001	92,31	4,34	< 0.0001	BII-P2P (TTL=16;γ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=8;γ =0,55)	13455,2	2015,01	< 0.0001	93,32	3,91	< 0.0001	BII-P2P (TTL=17;γ =0,20)	14,23	11,84	< 0.0001	1,59	1,47	< 0.0001
BII-P2P (TTL=8;γ =0,60)	21721,3	1299,33	< 0.0001	96,62	2,34	< 0.0001	BII-P2P (TTL=17;γ =0,25)	185,5	164,06	< 0.0001	14,68	10,57	< 0.0001
BII-P2P (TTL=8;γ =0,65)	25619,3	1185,37	< 0.0001	97,45	2,13	< 0.0001	BII-P2P (TTL=17;γ =0,30)	1779,88	504,39	< 0.0001	65,62	14,45	< 0.0001
BII-P2P (TTL=8;γ =0,70)	32828,1	867,1	< 0.0001	98,44	1,9	< 0.0001	BII-P2P (TTL=17;γ =0,35)	4360,13	218,42	0,431	84,41	4,69	< 0.0001
BII-P2P (TTL=8;γ =0,75)	41068,6	547,35	< 0.0001	99,22	1,04	< 0.0001	BII-P2P (TTL=17;γ =0,40)	7881,98	155,46	< 0.0001	90,97	3,17	< 0.0001
BII-P2P (TTL=8;γ =0,80)	45858,8	335,23	< 0.0001	99,52	0,66	< 0.0001	BII-P2P (TTL=17;γ =0,45)	11129,1	165,87	< 0.0001	93,29	3,32	< 0.0001
BII-P2P (TTL=8;γ =0,85)	50858,4	247,18	< 0.0001	99,7	0,53	< 0.0001	BII-P2P (TTL=17;γ =0,50)	17797,1	163,48	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=8;γ =0,90)	57061,9	148,06	< 0.0001	99,87	0,3	< 0.0001	BII-P2P (TTL=17;γ =0,55)	19413,4	153,73	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=9;γ =0,20)	12,58	8,92	< 0.0001	1,39	1,23	< 0.0001	BII-P2P (TTL=17;γ =0,60)	25978,1	148,29	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=9;γ =0,25)	56,6	48,59	< 0.0001	4,56	3,04	< 0.0001	BII-P2P (TTL=17;γ =0,65)	29119,7	131,42	< 0.0001	98,01	1,89	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=9;γ =0,30)	295,5	172,99	< 0.0001	18,54	9,28	< 0.0001	BII-P2P (TTL=17;γ =0,70)	35124,7	150,82	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=9;γ =0,35)	927,88	459,57	< 0.0001	39,42	15,21	0,2461	BII-P2P (TTL=17;γ =0,75)	42457,7	116,87	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=9;γ =0,40)	3304,08	983,18	< 0.0001	72,64	8,77	< 0.0001	BII-P2P (TTL=17;γ =0,80)	46848,6	132,89	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=9;γ =0,45)	6606,73	1211,37	< 0.0001	86,43	4,77	< 0.0001	BII-P2P (TTL=17;γ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=9;γ =0,50)	14749,3	1132,96	< 0.0001	94,88	2,85	< 0.0001	BII-P2P (TTL=17;γ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=9;γ =0,55)	16803,7	1100,93	< 0.0001	95,49	2,8	< 0.0001	BII-P2P (TTL=18;γ =0,20)	14,28	11,98	< 0.0001	1,59	1,47	< 0.0001
BII-P2P (TTL=9;γ =0,60)	24433,5	483,98	< 0.0001	97,28	2,21	< 0.0001	BII-P2P (TTL=18;γ =0,25)	205,15	177,27	< 0.0001	15,91	11,26	< 0.0001
BII-P2P (TTL=9;γ =0,65)	27935,2	416,23	< 0.0001	97,83	2,08	< 0.0001	BII-P2P (TTL=18;γ =0,30)	1928,98	488,35	< 0.0001	68,27	13,99	< 0.0001
BII-P2P (TTL=9;γ =0,70)	34455	278,44	< 0.0001	98,56	1,76	< 0.0001	BII-P2P (TTL=18;γ =0,35)	4463,65	173,08	0,7665	84,84	4,74	< 0.0001
BII-P2P (TTL=9;γ =0,75)	42084	178,68	< 0.0001	99,27	1,03	< 0.0001	BII-P2P (TTL=18;γ =0,40)	7907,83	154,22	< 0.0001	91,02	3,19	< 0.0001
BII-P2P (TTL=9;γ =0,80)	46605,6	145,19	< 0.0001	99,55	0,66	< 0.0001	BII-P2P (TTL=18;γ =0,45)	11138,5	164,65	< 0.0001	93,29	3,32	< 0.0001
BII-P2P (TTL=9;γ =0,85)	51271,9	129,97	< 0.0001	99,72	0,52	< 0.0001	BII-P2P (TTL=18;γ =0,50)	17799,8	164,15	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=9;γ =0,90)	57209	113,81	< 0.0001	99,87	0,3	< 0.0001	BII-P2P (TTL=18;γ =0,55)	19414,7	153,8	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=10;γ =0,20)	13,05	9,61	< 0.0001	1,39	1,23	< 0.0001	BII-P2P (TTL=18;γ =0,60)	25978,3	148,19	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=10;γ =0,25)	69,58	61,45	< 0.0001	5,5	3,88	< 0.0001	BII-P2P (TTL=18;γ =0,65)	29119,7	131,42	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=10;γ =0,30)	425,3	242,1	< 0.0001	25,55	11,74	< 0.0001	BII-P2P (TTL=18;γ =0,70)	35124,7	150,82	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=10;γ =0,35)	1431,15	620,37	< 0.0001	51,55	15,47	< 0.0001	BII-P2P (TTL=18;γ =0,75)	42457,7	116,87	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=10;γ =0,40)	4734,33	925,16	0,41	82,17	5,76	< 0.0001	BII-P2P (TTL=18;γ =0,80)	46848,6	132,89	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=10;γ =0,45)	8571,85	844,36	< 0.0001	91,1	3,3	< 0.0001	BII-P2P (TTL=18;γ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=10;γ =0,50)	16446,4	573,09	< 0.0001	95,61	2,81	< 0.0001	BII-P2P (TTL=18;γ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=10;γ =0,55)	18340,1	512,94	< 0.0001	96,04	2,77	< 0.0001	BII-P2P (TTL=19;γ =0,20)	14,28	11,98	< 0.0001	1,59	1,47	< 0.0001
BII-P2P (TTL=10;γ =0,60)	25413,6	219,71	< 0.0001	97,45	2,13	< 0.0001	BII-P2P (TTL=19;γ =0,25)	225,78	189,93	< 0.0001	17,5	11,73	< 0.0001
BII-P2P (TTL=10;γ =0,65)	28713,5	198,86	< 0.0001	97,96	1,95	< 0.0001	BII-P2P (TTL=19;γ =0,30)	2055,28	470,43	< 0.0001	70,19	13,63	< 0.0001
BII-P2P (TTL=10;γ =0,70)	34919,5	160,44	< 0.0001	98,61	1,69	< 0.0001	BII-P2P (TTL=19;γ =0,35)	4528,5	148,44	0,9886	85,12	4,68	< 0.0001
BII-P2P (TTL=10;γ =0,75)	42351,1	121,59	< 0.0001	99,27	1,03	< 0.0001	BII-P2P (TTL=19;γ =0,40)	7922,03	154,15	< 0.0001	91,02	3,19	< 0.0001
BII-P2P (TTL=10;γ =0,80)	46782,7	127,63	< 0.0001	99,57	0,64	< 0.0001	BII-P2P (TTL=19;γ =0,45)	11144,4	164,74	< 0.0001	93,29	3,32	< 0.0001
BII-P2P (TTL=10;γ =0,85)	51359,2	124,3	< 0.0001	99,72	0,52	< 0.0001	BII-P2P (TTL=19;γ =0,50)	17801,4	165,17	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=10;γ =0,90)	57232	112,6	< 0.0001	99,87	0,3	< 0.0001	BII-P2P (TTL=19;γ =0,55)	19415,8	154,75	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=11;γ =0,20)	13,4	10,19	< 0.0001	1,51	1,38	< 0.0001	BII-P2P (TTL=19;γ =0,60)	25978,4	148,13	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=11;γ =0,25)	83,53	74,86	< 0.0001	6,96	5,11	< 0.0001	BII-P2P (TTL=19;γ =0,65)	29119,7	131,42	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=11;γ =0,30)	588	317,4	< 0.0001	33,34	14,35	0,277	BII-P2P (TTL=19;γ =0,70)	35124,7	150,82	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=11;γ =0,35)	2020,35	729,88	< 0.0001	62,65	14	< 0.0001	BII-P2P (TTL=19;γ =0,75)	42457,7	116,87	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=11;γ =0,40)	5942,73	692	< 0.0001	87,49	3,62	< 0.0001	BII-P2P (TTL=19;γ =0,80)	46848,6	132,89	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=11;γ =0,45)	9793,43	506	< 0.0001	92,36	3,23	< 0.0001	BII-P2P (TTL=19;γ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=11;γ =0,50)	17206,8	298,79	< 0.0001	95,71	2,86	< 0.0001	BII-P2P (TTL=19;γ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001
BII-P2P (TTL=11;γ =0,55)	18970,5	252,18	< 0.0001	96,14	2,79	< 0.0001	BII-P2P (TTL=20;γ =0,20)	14,28	11,98	< 0.0001	1,59	1,47	< 0.0001
BII-P2P (TTL=11;γ =0,60)	25772,6	147,53	< 0.0001	97,5	2,11	< 0.0001	BII-P2P (TTL=20;γ =0,25)	247,03	202,17	< 0.0001	18,94	12,47	< 0.0001
BII-P2P (TTL=11;γ =0,65)	28979,1	147,83	< 0.0001	97,98	1,88	< 0.0001	BII-P2P (TTL=20;γ =0,30)	2157,75	453,89	< 0.0001	71,88	13,21	< 0.0001
BII-P2P (TTL=11;γ =0,70)	35060,4	146,07	< 0.0001	98,64	1,69	< 0.0001	BII-P2P (TTL=20;γ =0,35)	4567,9	136,37	0,8383	85,37	4,63	< 0.0001
BII-P2P (TTL=11;γ =0,75)	42423,8	116,38	< 0.0001	99,27	1,03	< 0.0001	BII-P2P (TTL=20;γ =0,40)	7930,33	155,12	< 0.0001	91,05	3,2	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 10.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=11; $\gamma$ =0,80)	46828,7	130,08	< 0.0001	99,57	0,64	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,45)	11147,3	164,34	< 0.0001	93,29	3,32	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,85)	51378,4	122,74	< 0.0001	99,72	0,52	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,50)	17801,8	165,79	< 0.0001	95,81	2,84	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,90)	57235,4	112,59	< 0.0001	99,87	0,3	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,55)	19416,1	154,85	< 0.0001	96,27	2,59	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,20)	13,65	10,68	< 0.0001	1,51	1,38	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,60)	25978,4	148,13	< 0.0001	97,58	2,11	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,25)	98,48	89,51	< 0.0001	8,2	6,57	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,65)	29119,7	131,42	< 0.0001	98,01	1,89	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,30)	777,93	387,28	< 0.0001	40,78	15,95	0,1239	BII-P2P (TTL=20; $\gamma$ =0,70)	35124,7	150,82	< 0.0001	98,64	1,69	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	2618	742,43	< 0.0001	70,95	10,89	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,75)	42457,7	116,87	< 0.0001	99,27	1,03	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	6778	465,63	< 0.0001	89,23	3,39	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,80)	46848,6	132,89	< 0.0001	99,57	0,64	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,45)	10462	304,64	< 0.0001	92,94	3,25	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,85)	51382,9	123,66	< 0.0001	99,72	0,52	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,50)	17539,2	195,06	< 0.0001	95,79	2,84	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,90)	57236,1	112,98	< 0.0001	99,87	0,3	< 0.0001

**Tabla A-4 Rendimiento de la búsqueda inteligente sobre red de 20.000 nodos**

Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
<i>Gnutella BFS (TTL=4)</i>	4791,05	1753,6	< 0.0001	21,26	8,44	< 0.0001	BII-P2P (TTL=12;γ =0,55)	38129,1	355,6	< 0.0001	96,15	1,75	< 0.0001
BII-P2P (TTL=4;γ =0,20)	6,63	3,73	< 0.0001	0,28	0,37	< 0.0001	BII-P2P (TTL=12;γ =0,60)	51610,1	239,65	< 0.0001	97,62	1,16	< 0.0001
BII-P2P (TTL=4;γ =0,25)	12,75	7,12	< 0.0001	0,52	0,58	< 0.0001	BII-P2P (TTL=12;γ =0,65)	57929,8	239,5	< 0.0001	98,11	1,04	< 0.0001
BII-P2P (TTL=4;γ =0,30)	26,2	14,4	< 0.0001	0,82	0,68	< 0.0001	BII-P2P (TTL=12;γ =0,70)	70024,4	182,51	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=4;γ =0,35)	42,63	25,75	< 0.0001	1,19	0,85	< 0.0001	BII-P2P (TTL=12;γ =0,75)	84714,3	189,88	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=4;γ =0,40)	83,73	43,38	< 0.0001	1,88	1,21	< 0.0001	BII-P2P (TTL=12;γ =0,80)	93590	170,22	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=4;γ =0,45)	128,2	62,08	< 0.0001	2,31	1,33	< 0.0001	BII-P2P (TTL=12;γ =0,85)	102595	121,15	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=4;γ =0,50)	258,93	99,86	< 0.0001	3,68	1,81	< 0.0001	BII-P2P (TTL=12;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=4;γ =0,55)	311,6	136,08	< 0.0001	4,21	2,22	< 0.0001	BII-P2P (TTL=13;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=4;γ =0,60)	491,13	207,63	< 0.0001	5,46	2,66	< 0.0001	BII-P2P (TTL=13;γ =0,25)	112,68	95,27	< 0.0001	5,26	4,12	< 0.0001
BII-P2P (TTL=4;γ =0,65)	618,35	267,18	< 0.0001	6,32	3,16	< 0.0001	BII-P2P (TTL=13;γ =0,30)	1159,05	587,19	< 0.0001	32,37	14,28	0,0001
BII-P2P (TTL=4;γ =0,70)	859,65	376,81	< 0.0001	7,8	3,77	< 0.0001	BII-P2P (TTL=13;γ =0,35)	4993,03	1539,82	0,5857	68,46	12,32	< 0.0001
BII-P2P (TTL=4;γ =0,75)	1310,65	516,47	< 0.0001	10,18	4,45	< 0.0001	BII-P2P (TTL=13;γ =0,40)	13677,8	1202,18	< 0.0001	88,15	3,05	< 0.0001
BII-P2P (TTL=4;γ =0,80)	1580,28	561,83	< 0.0001	11,41	4,55	< 0.0001	BII-P2P (TTL=13;γ =0,45)	21191,4	871,85	< 0.0001	92,59	2,13	< 0.0001
BII-P2P (TTL=4;γ =0,85)	1983,05	790,02	< 0.0001	12,92	5,52	< 0.0001	BII-P2P (TTL=13;γ =0,50)	35193,7	264,24	< 0.0001	95,74	1,83	< 0.0001
BII-P2P (TTL=4;γ =0,90)	2556,23	947,63	< 0.0001	14,95	5,95	0,0002	BII-P2P (TTL=13;γ =0,55)	38464,7	272,79	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=5;γ =0,20)	8,05	5,06	< 0.0001	0,45	0,54	< 0.0001	BII-P2P (TTL=13;γ =0,60)	51774,9	199,34	< 0.0001	97,62	1,16	< 0.0001
BII-P2P (TTL=5;γ =0,25)	18,38	11,25	< 0.0001	0,9	0,87	< 0.0001	BII-P2P (TTL=13;γ =0,65)	58035,2	219,58	< 0.0001	98,11	1,04	< 0.0001
BII-P2P (TTL=5;γ =0,30)	45,33	26,36	< 0.0001	1,61	1,31	< 0.0001	BII-P2P (TTL=13;γ =0,70)	70073,8	182,15	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=5;γ =0,35)	87,18	55,47	< 0.0001	2,46	1,85	< 0.0001	BII-P2P (TTL=13;γ =0,75)	84732,6	188,5	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=5;γ =0,40)	204,95	112,14	< 0.0001	4,34	2,53	< 0.0001	BII-P2P (TTL=13;γ =0,80)	93600,9	170,06	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=5;γ =0,45)	354,93	175,99	< 0.0001	6,31	3,18	< 0.0001	BII-P2P (TTL=13;γ =0,85)	102597	121,16	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=5;γ =0,50)	878	334,42	< 0.0001	11,85	4,53	< 0.0001	BII-P2P (TTL=13;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=5;γ =0,55)	1103,03	480,9	< 0.0001	13,78	5,55	< 0.0001	BII-P2P (TTL=14;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=5;γ =0,60)	1993,8	827,65	< 0.0001	19,98	7,88	0,4612	BII-P2P (TTL=14;γ =0,25)	133,38	113,35	< 0.0001	6,23	4,78	< 0.0001
BII-P2P (TTL=5;γ =0,65)	2657,43	1130,3	< 0.0001	23,82	9,11	0,219	BII-P2P (TTL=14;γ =0,30)	1525,53	704,86	< 0.0001	39,81	15,52	< 0.0001
BII-P2P (TTL=5;γ =0,70)	4072,93	1741,84	0,0699	30,41	11,7	0,0002	BII-P2P (TTL=14;γ =0,35)	6133	1415,51	0,0003	74,53	9,29	< 0.0001
BII-P2P (TTL=5;γ =0,75)	6814,48	2582,26	0,0001	41,37	12,93	< 0.0001	BII-P2P (TTL=14;γ =0,40)	14688	745,44	< 0.0001	89,17	2,66	< 0.0001
BII-P2P (TTL=5;γ =0,80)	8633,98	2939,15	< 0.0001	46,51	12,64	< 0.0001	BII-P2P (TTL=14;γ =0,45)	21808,1	502,29	< 0.0001	92,87	2,12	< 0.0001
BII-P2P (TTL=5;γ =0,85)	11442,3	4316,57	< 0.0001	52,66	14,92	< 0.0001	BII-P2P (TTL=14;γ =0,50)	35386,6	233,25	< 0.0001	95,79	1,8	< 0.0001
BII-P2P (TTL=5;γ =0,90)	15632,2	5486,16	< 0.0001	60,57	15,05	< 0.0001	BII-P2P (TTL=14;γ =0,55)	38600,9	251,59	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=6;γ =0,20)	9,18	5,92	< 0.0001	0,48	0,55	< 0.0001	BII-P2P (TTL=14;γ =0,60)	51833,1	189,79	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=6;γ =0,25)	25,18	16,77	< 0.0001	1,26	1,25	< 0.0001	BII-P2P (TTL=14;γ =0,65)	58073,9	216,58	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=6;γ =0,30)	75,13	45,64	< 0.0001	2,75	2,07	< 0.0001	BII-P2P (TTL=14;γ =0,70)	70089,2	181,58	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=6;γ =0,35)	170,13	109,07	< 0.0001	4,63	3,13	< 0.0001	BII-P2P (TTL=14;γ =0,75)	84739,8	189,3	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=6;γ =0,40)	481,03	263,21	< 0.0001	9,95	5,49	< 0.0001	BII-P2P (TTL=14;γ =0,80)	93605,1	170,7	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=6;γ =0,45)	946,7	460,69	< 0.0001	15,77	7,73	0,0031	BII-P2P (TTL=14;γ =0,85)	102598	120,72	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=6;γ =0,50)	2840,38	1028,41	< 0.0001	32,92	11,15	< 0.0001	BII-P2P (TTL=14;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001

Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=6;γ =0,55)	3691,4	1529,69	0,0038	38	13,41	< 0.0001	BII-P2P (TTL=15;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=6;γ =0,60)	7427,58	2846,55	< 0.0001	54,83	15,15	< 0.0001	BII-P2P (TTL=15;γ =0,25)	154,58	131,35	< 0.0001	6,96	5,39	< 0.0001
BII-P2P (TTL=6;γ =0,65)	10244,9	3954,36	< 0.0001	63,07	16,6	< 0.0001	BII-P2P (TTL=15;γ =0,30)	1938,13	799,67	< 0.0001	46,27	15,35	< 0.0001
BII-P2P (TTL=6;γ =0,70)	16622	6213,5	< 0.0001	74,25	16,29	< 0.0001	BII-P2P (TTL=15;γ =0,35)	7100,6	1169,57	< 0.0001	78,73	6,62	< 0.0001
BII-P2P (TTL=6;γ =0,75)	28605	8718,67	< 0.0001	87,43	10	< 0.0001	BII-P2P (TTL=15;γ =0,40)	15273	470,48	< 0.0001	89,65	2,68	< 0.0001
BII-P2P (TTL=6;γ =0,80)	36742,8	9752,92	< 0.0001	91,38	8,17	< 0.0001	BII-P2P (TTL=15;γ =0,45)	22123,3	333,36	< 0.0001	92,96	2,17	< 0.0001
BII-P2P (TTL=6;γ =0,85)	48186,8	13213,8	< 0.0001	94,36	7,01	< 0.0001	BII-P2P (TTL=15;γ =0,50)	35467,9	222,05	< 0.0001	95,81	1,8	< 0.0001
BII-P2P (TTL=6;γ =0,90)	64651,9	15900,4	< 0.0001	97,07	4,77	< 0.0001	BII-P2P (TTL=15;γ =0,55)	38655,4	245,55	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=7;γ =0,20)	9,88	6,53	< 0.0001	0,52	0,58	< 0.0001	BII-P2P (TTL=15;γ =0,60)	51854,7	189,14	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=7;γ =0,25)	32,7	23,22	< 0.0001	1,55	1,33	< 0.0001	BII-P2P (TTL=15;γ =0,65)	58089,3	215,77	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=7;γ =0,30)	119,1	73,77	< 0.0001	4,13	3,01	< 0.0001	BII-P2P (TTL=15;γ =0,70)	70094	180,76	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=7;γ =0,35)	321,55	204,92	< 0.0001	8,15	4,99	< 0.0001	BII-P2P (TTL=15;γ =0,75)	84741,6	188,92	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=7;γ =0,40)	1086,18	577,13	< 0.0001	20,98	10,21	0,8571	BII-P2P (TTL=15;γ =0,80)	93606,4	171,12	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=7;γ =0,45)	2387,53	1110,73	< 0.0001	35,4	13,89	< 0.0001	BII-P2P (TTL=15;γ =0,85)	102598	120,31	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=7;γ =0,50)	8076,85	2522,89	< 0.0001	65,08	12,96	< 0.0001	BII-P2P (TTL=15;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=7;γ =0,55)	10496,5	3597,91	< 0.0001	71,04	13,29	< 0.0001	BII-P2P (TTL=16;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=7;γ =0,60)	21227,2	6048,34	< 0.0001	86,29	8,61	< 0.0001	BII-P2P (TTL=16;γ =0,25)	177,3	150,68	< 0.0001	8,05	6,31	< 0.0001
BII-P2P (TTL=7;γ =0,65)	28286,6	7649,57	< 0.0001	90,79	7,22	< 0.0001	BII-P2P (TTL=16;γ =0,30)	2375,65	854,64	< 0.0001	52,83	14,48	< 0.0001
BII-P2P (TTL=7;γ =0,70)	42957,7	9813,69	< 0.0001	95,58	4,18	< 0.0001	BII-P2P (TTL=16;γ =0,35)	7858,08	886,62	< 0.0001	81,18	4,84	< 0.0001
BII-P2P (TTL=7;γ =0,75)	64585,4	8437,17	< 0.0001	98,61	1,03	< 0.0001	BII-P2P (TTL=16;γ =0,40)	15599,6	331,8	< 0.0001	89,88	2,6	< 0.0001
BII-P2P (TTL=7;γ =0,80)	76994,1	7415,43	< 0.0001	99,24	0,78	< 0.0001	BII-P2P (TTL=16;γ =0,45)	22279,2	265,75	< 0.0001	93	2,15	< 0.0001
BII-P2P (TTL=7;γ =0,85)	90697,8	7118,2	< 0.0001	99,53	0,49	< 0.0001	BII-P2P (TTL=16;γ =0,50)	35503,7	217,43	< 0.0001	95,82	1,79	< 0.0001
BII-P2P (TTL=7;γ =0,90)	107380	5409,65	< 0.0001	99,91	0,22	< 0.0001	BII-P2P (TTL=16;γ =0,55)	38680,7	241,13	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=8;γ =0,20)	10,4	7,03	< 0.0001	0,56	0,57	< 0.0001	BII-P2P (TTL=16;γ =0,60)	51861,7	188,12	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=8;γ =0,25)	41,55	31,03	< 0.0001	1,94	1,57	< 0.0001	BII-P2P (TTL=16;γ =0,65)	58093	217,61	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=8;γ =0,30)	184,45	113,5	< 0.0001	6,07	4,09	< 0.0001	BII-P2P (TTL=16;γ =0,70)	70095,2	180,65	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=8;γ =0,35)	587,5	365,39	< 0.0001	14,61	8,72	0,0007	BII-P2P (TTL=16;γ =0,75)	84742,1	188,78	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=8;γ =0,40)	2284,98	1137,95	< 0.0001	37,22	14,71	< 0.0001	BII-P2P (TTL=16;γ =0,80)	93606,7	171,04	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=8;γ =0,45)	5344,48	2226,83	0,2208	58,33	16,87	< 0.0001	BII-P2P (TTL=16;γ =0,85)	102598	120,22	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=8;γ =0,50)	17416,7	3649,11	< 0.0001	86,9	5,85	< 0.0001	BII-P2P (TTL=16;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=8;γ =0,55)	21547,7	4549,4	< 0.0001	90,05	5,1	< 0.0001	BII-P2P (TTL=17;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=8;γ =0,60)	37773	5074,99	< 0.0001	95,95	1,88	< 0.0001	BII-P2P (TTL=17;γ =0,25)	201,98	171,64	< 0.0001	9,12	7,41	< 0.0001
BII-P2P (TTL=8;γ =0,65)	46126	5276,19	< 0.0001	97,2	1,46	< 0.0001	BII-P2P (TTL=17;γ =0,30)	2817,33	859,92	< 0.0001	57,67	13,1	< 0.0001
BII-P2P (TTL=8;γ =0,70)	61586	4513,3	< 0.0001	98,31	1,02	< 0.0001	BII-P2P (TTL=17;γ =0,35)	8393,75	627,8	< 0.0001	82,64	4,12	< 0.0001
BII-P2P (TTL=8;γ =0,75)	80033,4	2057,55	< 0.0001	99,08	0,82	< 0.0001	BII-P2P (TTL=17;γ =0,40)	15770,9	267	< 0.0001	89,98	2,65	< 0.0001
BII-P2P (TTL=8;γ =0,80)	90171,3	1404,82	< 0.0001	99,5	0,54	< 0.0001	BII-P2P (TTL=17;γ =0,45)	22352,2	241,97	< 0.0001	93,04	2,13	< 0.0001
BII-P2P (TTL=8;γ =0,85)	100665	1002,31	< 0.0001	99,63	0,45	< 0.0001	BII-P2P (TTL=17;γ =0,50)	35517,3	213,99	< 0.0001	95,82	1,79	< 0.0001
BII-P2P (TTL=8;γ =0,90)	113661	427,15	< 0.0001	99,95	0,15	< 0.0001	BII-P2P (TTL=17;γ =0,55)	38691,4	237,06	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=9;γ =0,20)	10,78	7,36	< 0.0001	0,58	0,57	< 0.0001	BII-P2P (TTL=17;γ =0,60)	51864,4	187,32	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=9;γ =0,25)	51,85	40,48	< 0.0001	2,35	2,04	< 0.0001	BII-P2P (TTL=17;γ =0,65)	58094,4	218,05	< 0.0001	98,12	1,04	< 0.0001

Continúa la tabla...



Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=9;γ =0,30)	280,48	171,58	< 0.0001	9,18	5,94	< 0.0001	BII-P2P (TTL=17;γ =0,70)	70095,8	180,33	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=9;γ =0,35)	1032,15	610,3	< 0.0001	23,97	12,18	0,2846	BII-P2P (TTL=17;γ =0,75)	84742,1	188,78	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=9;γ =0,40)	4297,08	1831,94	0,2217	56,17	16,2	< 0.0001	BII-P2P (TTL=17;γ =0,80)	93606,7	171,04	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=9;γ =0,45)	9776,68	3221,99	< 0.0001	76,43	14,08	< 0.0001	BII-P2P (TTL=17;γ =0,85)	102598	120,12	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=9;γ =0,50)	26318,1	2649,33	< 0.0001	93,26	2,53	< 0.0001	BII-P2P (TTL=17;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=9;γ =0,55)	30680,2	2867,66	< 0.0001	94,61	2,27	< 0.0001	BII-P2P (TTL=18;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=9;γ =0,60)	46592	2108,02	< 0.0001	97,21	1,25	< 0.0001	BII-P2P (TTL=18;γ =0,25)	229,33	194,47	< 0.0001	10,12	8,2	< 0.0001
BII-P2P (TTL=9;γ =0,65)	54073,2	1891,58	< 0.0001	97,86	1,14	< 0.0001	BII-P2P (TTL=18;γ =0,30)	3239,4	816,17	< 0.0001	61,97	10,89	< 0.0001
BII-P2P (TTL=9;γ =0,70)	67683,2	1225,29	< 0.0001	98,55	0,95	< 0.0001	BII-P2P (TTL=18;γ =0,35)	8757,73	437,56	< 0.0001	83,56	3,72	< 0.0001
BII-P2P (TTL=9;γ =0,75)	83546,6	515,54	< 0.0001	99,09	0,82	< 0.0001	BII-P2P (TTL=18;γ =0,40)	15867,2	240,27	< 0.0001	90,05	2,64	< 0.0001
BII-P2P (TTL=9;γ =0,80)	92788,9	337,18	< 0.0001	99,53	0,52	< 0.0001	BII-P2P (TTL=18;γ =0,45)	22389,2	232,29	< 0.0001	93,04	2,13	< 0.0001
BII-P2P (TTL=9;γ =0,85)	102211	229,81	< 0.0001	99,68	0,42	< 0.0001	BII-P2P (TTL=18;γ =0,50)	35523,8	212,25	< 0.0001	95,82	1,79	< 0.0001
BII-P2P (TTL=9;γ =0,90)	114286	117,34	< 0.0001	99,96	0,13	< 0.0001	BII-P2P (TTL=18;γ =0,55)	38696,7	235,25	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=10;γ =0,20)	10,95	7,5	< 0.0001	0,59	0,58	< 0.0001	BII-P2P (TTL=18;γ =0,60)	51865,3	186,71	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=10;γ =0,25)	64,13	52,58	< 0.0001	2,95	2,43	< 0.0001	BII-P2P (TTL=18;γ =0,65)	58095,6	218,47	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=10;γ =0,30)	418,9	252,08	< 0.0001	13,66	8,19	< 0.0001	BII-P2P (TTL=18;γ =0,70)	70097	180,09	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=10;γ =0,35)	1716,28	922,65	< 0.0001	35,38	15,62	< 0.0001	BII-P2P (TTL=18;γ =0,75)	84742,1	188,78	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=10;γ =0,40)	6967,33	2309,88	< 0.0001	71,35	13,09	< 0.0001	BII-P2P (TTL=18;γ =0,80)	93606,7	171,04	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=10;γ =0,45)	14336,7	3316,65	< 0.0001	85,92	7,96	< 0.0001	BII-P2P (TTL=18;γ =0,85)	102598	120,12	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=10;γ =0,50)	31374,5	1346,86	< 0.0001	95,02	1,9	< 0.0001	BII-P2P (TTL=18;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=10;γ =0,55)	35320,9	1323,36	< 0.0001	95,79	1,79	< 0.0001	BII-P2P (TTL=19;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=10;γ =0,60)	49937	799,31	< 0.0001	97,46	1,14	< 0.0001	BII-P2P (TTL=19;γ =0,25)	258,68	217,58	< 0.0001	11,27	8,93	< 0.0001
BII-P2P (TTL=10;γ =0,65)	56763,1	656,18	< 0.0001	98,02	1,1	< 0.0001	BII-P2P (TTL=19;γ =0,30)	3623,45	731,89	0,0003	65,59	9,09	< 0.0001
BII-P2P (TTL=10;γ =0,70)	69387,8	376,56	< 0.0001	98,57	0,95	< 0.0001	BII-P2P (TTL=19;γ =0,35)	8993,7	308,03	< 0.0001	84,08	3,63	< 0.0001
BII-P2P (TTL=10;γ =0,75)	84403,1	219,16	< 0.0001	99,12	0,84	< 0.0001	BII-P2P (TTL=19;γ =0,40)	15919,1	228,57	< 0.0001	90,07	2,64	< 0.0001
BII-P2P (TTL=10;γ =0,80)	93391,6	181,6	< 0.0001	99,56	0,45	< 0.0001	BII-P2P (TTL=19;γ =0,45)	22406,5	228,16	< 0.0001	93,04	2,13	< 0.0001
BII-P2P (TTL=10;γ =0,85)	102509	131,46	< 0.0001	99,68	0,42	< 0.0001	BII-P2P (TTL=19;γ =0,50)	35527,8	211,33	< 0.0001	95,82	1,79	< 0.0001
BII-P2P (TTL=10;γ =0,90)	114377	117,01	< 0.0001	99,96	0,13	< 0.0001	BII-P2P (TTL=19;γ =0,55)	38699	235,29	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=11;γ =0,20)	11,05	7,59	< 0.0001	0,59	0,58	< 0.0001	BII-P2P (TTL=19;γ =0,60)	51866	186,14	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=11;γ =0,25)	78,33	65,22	< 0.0001	3,62	2,96	< 0.0001	BII-P2P (TTL=19;γ =0,65)	58095,9	218,57	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=11;γ =0,30)	606,6	350,25	< 0.0001	19,35	10,8	0,3268	BII-P2P (TTL=19;γ =0,70)	70097,5	179,88	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=11;γ =0,35)	2654,45	1246,21	< 0.0001	48,13	17,1	< 0.0001	BII-P2P (TTL=19;γ =0,75)	84742,1	188,78	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=11;γ =0,40)	9739,68	2265,84	< 0.0001	81,35	7,19	< 0.0001	BII-P2P (TTL=19;γ =0,80)	93606,7	171,04	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=11;γ =0,45)	17808,4	2528,06	< 0.0001	90,29	3,68	< 0.0001	BII-P2P (TTL=19;γ =0,85)	102598	120,12	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=11;γ =0,50)	33703,5	666,93	< 0.0001	95,57	1,81	< 0.0001	BII-P2P (TTL=19;γ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001
BII-P2P (TTL=11;γ =0,55)	37308,9	623,23	< 0.0001	96,09	1,77	< 0.0001	BII-P2P (TTL=20;γ =0,20)	11,18	7,74	< 0.0001	0,61	0,61	< 0.0001
BII-P2P (TTL=11;γ =0,60)	51160,8	368,57	< 0.0001	97,55	1,16	< 0.0001	BII-P2P (TTL=20;γ =0,25)	290,03	241,72	< 0.0001	12,58	9,82	< 0.0001
BII-P2P (TTL=11;γ =0,65)	57634,1	316,72	< 0.0001	98,08	1,06	< 0.0001	BII-P2P (TTL=20;γ =0,30)	3954,7	630,23	0,0066	68,19	7,9	< 0.0001
BII-P2P (TTL=11;γ =0,70)	69874,3	207,77	< 0.0001	98,57	0,95	< 0.0001	BII-P2P (TTL=20;γ =0,35)	9143,08	235,02	< 0.0001	84,24	3,56	< 0.0001
BII-P2P (TTL=11;γ =0,75)	84648	192,89	< 0.0001	99,13	0,83	< 0.0001	BII-P2P (TTL=20;γ =0,40)	15948,4	224,65	< 0.0001	90,08	2,63	< 0.0001

Continúa la tabla...

Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas			Búsqueda sobre red de 20.000 nodos	Querys generadas			% de Instancias halladas		
	Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor		Media	Desv.Est.	pvalor	Media	Desv.Est.	pvalor
BII-P2P (TTL=11; $\gamma$ =0,80)	93549,5	171,68	< 0.0001	99,56	0,45	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,45)	22415,4	226,58	< 0.0001	93,04	2,13	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,85)	102581	122,76	< 0.0001	99,68	0,42	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,50)	35529,8	211	< 0.0001	95,82	1,79	< 0.0001
BII-P2P (TTL=11; $\gamma$ =0,90)	114392	118,85	< 0.0001	99,96	0,13	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,55)	38700,2	235,42	< 0.0001	96,19	1,72	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,20)	11,13	7,67	< 0.0001	0,61	0,61	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,60)	51866,2	185,94	< 0.0001	97,63	1,17	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,25)	94,55	79,32	< 0.0001	4,28	3,26	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,65)	58096,1	218,64	< 0.0001	98,12	1,04	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,30)	851,63	464,87	< 0.0001	25,32	12,46	0,1172	BII-P2P (TTL=20; $\gamma$ =0,70)	70097,7	179,99	< 0.0001	98,57	0,95	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,35)	3788,35	1479,45	0,0072	59,99	15,27	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,75)	84742,1	188,78	< 0.0001	99,13	0,83	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,40)	12052,6	1789,46	< 0.0001	86,29	3,88	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,80)	93606,7	171,04	< 0.0001	99,56	0,45	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,45)	19986,3	1555,42	< 0.0001	92,06	2,06	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,85)	102598	120,12	< 0.0001	99,68	0,42	< 0.0001
BII-P2P (TTL=12; $\gamma$ =0,50)	34743,4	370,13	< 0.0001	95,67	1,84	< 0.0001	BII-P2P (TTL=20; $\gamma$ =0,90)	114395	119,53	< 0.0001	99,96	0,13	< 0.0001

**Tabla A-5 Resultados experimentales sobre red P2P con 2.000 nodos. Las celdas sombreadas corresponden a los casos en que BII-P2P supera a *Gnutella* en ambas métricas consideradas.**

TTL	Tasa de selección $\gamma$															
	0,20	0,25	0,30	0,35	0,40	0,45	0,50	0,55	0,60	0,65	0,70	0,75	0,80	0,85	0,90	
4	Q	8,35	15,25	30,38	46,9	83,55	125,13	254,65	291,08	470,18	574,18	810,6	1159,65	1396,1	1710,5	2186,28
	H	3,59	5,38	9,1	12,56	16,67	20,64	31,15	32,56	39,36	43,33	49,49	57,56	61,15	65	70,26
5	Q	10,35	22,55	53,1	93,25	192,43	321,1	765,45	903,95	1580,1	1982,93	2913,03	4266,53	5141,8	6303,95	7897,1
	H	5,64	9,87	17,69	24,36	34,36	43,59	63,46	65,38	78,33	82,69	88,72	94,62	96,79	98,08	99,49
6	Q	12,08	30,08	84,55	166,4	389,75	695,08	1721,38	2041,65	3409,2	4127,48	5627,35	7484,8	8541,03	9793,65	11256,03
	H	6,03	12,18	22,44	32,69	51,15	65,77	86,92	88,72	94,36	95,64	97,69	98,85	99,62	99,87	100
7	Q	13,4	38	124,13	269,65	673,68	1189,48	2679,6	3085,63	4613,33	5342,3	6754,35	8394,35	9346,35	10396,83	11643,43
	H	6,79	14,74	31,03	45,64	68,72	83,08	94,36	95,26	97,31	97,69	98,21	98,97	99,74	100	100
8	Q	14,65	46,85	171,1	397,55	980,2	1633	3222,58	3620,65	5078	5761,6	7064,18	8600,23	9518,48	10496,6	11685,33
	H	7,44	17,82	40,38	60,64	81,41	90,26	95,9	96,54	97,69	98,08	98,46	98,97	99,74	100	100
9	Q	15,5	55,58	222,5	528,35	1237,78	1943,23	3468,63	3852,15	5246,55	5898,68	7151,33	8656,45	9558,5	10514,63	11690
	H	8,59	21,03	48,46	69,62	85,77	92,56	96,41	96,67	97,69	98,21	98,46	98,97	99,74	100	100
10	Q	16	63,75	274,08	649,9	1419,03	2124,95	3576,08	3944,58	5309,2	5945,05	7180,7	8673,08	9571,4	10518,15	11691,18
	H	8,97	24,74	54,62	75,9	89,36	93,59	96,41	96,67	97,69	98,21	98,59	98,97	99,74	100	100
11	Q	16,4	71,78	323,98	749,2	1531,63	2217,48	3621,05	3985,28	5329,75	5959,83	7188,28	8678,53	9573,88	10519	11691,3
	H	8,97	28,46	59,49	79,74	90,64	94,1	96,41	96,67	97,69	98,21	98,59	98,97	99,74	100	100
12	Q	16,53	79,63	368,1	820,95	1597,13	2265,35	3638,6	4000,3	5336,75	5964,3	7190,75	8680,33	9575,43	10519	11691,3
	H	9,1	30,51	63,21	82,82	91,03	94,1	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
13	Q	16,6	87,05	403,68	871,78	1634,68	2288,45	3646,95	4006,7	5339,25	5965,3	7190,88	8680,65	9575,75	10519	11691,3
	H	9,1	32,31	65,9	84,23	91,41	94,1	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
14	Q	16,65	93,25	433,48	908,88	1654,18	2300,93	3650,2	4008,85	5339,6	5965,6	7190,98	8680,85	9575,83	10519	11691,3
	H	9,23	34,62	68,08	84,49	91,67	94,1	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
15	Q	16,7	98,8	457,48	932,88	1663,88	2307,33	3651,9	4010,13	5339,73	5965,75	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	36,41	70,38	85,26	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
16	Q	16,7	103,73	476,23	949,15	1669,33	2309,78	3652,58	4011	5339,73	5965,88	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	37,56	70,9	85,51	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
17	Q	16,7	107,8	490,43	959,6	1672,75	2311,15	3652,93	4011,23	5339,73	5965,88	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	39,1	71,67	85,51	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
18	Q	16,7	111,28	500,93	965,28	1674,43	2311,63	3653,05	4011,4	5339,73	5965,88	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	39,36	72,69	85,64	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
19	Q	16,7	114,53	509,55	969,08	1674,93	2311,63	3653,2	4011,5	5339,73	5965,88	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	39,74	73,08	85,77	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100
20	Q	16,7	117,1	515,7	972,08	1675,1	2311,63	3653,3	4011,5	5339,73	5965,88	7190,98	8681,08	9575,83	10519	11691,3
	H	9,23	40,38	73,46	85,9	91,79	94,23	96,41	96,79	97,69	98,21	98,59	98,97	99,74	100	100

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q=3.913,4 H=81,79%

**Tabla A-6 Resultados experimentales sobre red P2P con 5.000 nodos. Las celdas sombreadas corresponden a los casos en que BII-P2P supera a *Gnutella* en ambas métricas consideradas.**

TTL		Tasa de selección $\gamma$														
		0,20	0,25	0,30	0,35	0,40	0,45	0,50	0,55	0,60	0,65	0,70	0,75	0,80	0,85	0,90
4	Q	6,55	12,7	25,58	42,33	83,93	119,05	254,68	299,33	454,88	591,5	841,25	1225,1	1463,13	1849,43	2364,28
	H	1,43	2,1	3,07	4,51	6,76	7,89	13,11	14,13	18,07	21,66	26,16	32,21	35,18	39,38	43,83
5	Q	7,88	18,08	44,58	86,3	203,15	326,13	833,38	1019,35	1728,63	2343,6	3595,4	5567,75	6850,88	8928,28	11669,6
	H	1,79	3,33	5,73	9,22	16,33	20,94	37,43	41,01	52,43	59,19	69,48	79,83	84,02	88,53	92,88
6	Q	9,05	24,78	73,05	166,55	455,1	813,1	2319,43	2882,05	5094,23	6769,05	10085,35	14654,58	17411,58	21113,18	25304,48
	H	1,95	4,56	10,09	17,1	31,08	42,91	69,59	73,73	84,69	89,66	94,52	97,44	98,62	99,33	99,64
7	Q	9,9	32,53	113,95	302,1	912,98	1724,45	4736,13	5704,15	9283,9	11371,6	15178,9	19734,08	22259,18	25109,93	28410,58
	H	2,25	6,09	15,16	29,24	50,95	66,31	88,63	90,58	95,75	97,24	98,67	99,18	99,44	99,64	99,8
8	Q	10,55	41,1	169,28	508,33	1572,93	2906,58	6823,35	7873,55	11645,45	13526,55	16967,45	20959,1	23233,73	25707,93	28716,25
	H	2,46	7,99	21,35	40,96	68,31	80,85	94,32	95,03	97,44	98,21	98,98	99,23	99,49	99,64	99,8
9	Q	11,03	50,65	241,4	782,4	2281,55	3944,28	7990,18	8981,28	12556,85	14246,33	17468,3	21260,13	23460,98	25820,1	28758,03
	H	2,51	9,22	28,93	54,12	79,11	87,97	95,39	96,06	97,64	98,31	98,98	99,28	99,49	99,64	99,8
10	Q	11,33	60,6	330,3	1095,18	2878,23	4667,4	8534,23	9464,6	12876,15	14487,18	17613,88	21338,35	23518,25	25844,03	28763,5
	H	2,71	11,06	35,74	65,13	84,74	90,68	95,85	96,31	97,75	98,36	98,98	99,28	99,49	99,64	99,8
11	Q	11,53	70,63	432,28	1400,73	3302,13	5107,85	8779,75	9666,88	12998	14570,55	17658,48	21362,38	23537,58	25849,93	28764,7
	H	2,76	12,7	43,63	72,3	87,66	92,93	96,01	96,31	97,8	98,36	98,98	99,28	99,49	99,64	99,8
12	Q	11,7	81,9	545,43	1662,43	3587,13	5359,2	8884,65	9746,75	13045,15	14601,45	17671,9	21370,38	23541,73	25851,1	28764,7
	H	2,87	14,44	50,18	76,91	89,5	93,45	96,01	96,31	97,85	98,36	98,98	99,28	99,49	99,64	99,8
13	Q	11,85	92,83	658,25	1863,4	3765,63	5490,3	8933,25	9778,33	13064,2	14612,93	17676,65	21372,23	23542,38	25851,1	28764,7
	H	2,87	16,08	55,25	79,62	90,32	93,8	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
14	Q	11,98	103,98	766,73	2014,53	3873,35	5556,95	8953,58	9788,8	13071,8	14616,58	17678,3	21373,1	23542,38	25851,1	28764,7
	H	2,87	17,46	60,16	81,21	90,83	93,86	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
15	Q	12,08	115,53	866,68	2126,6	3934,83	5590,43	8962,23	9793,28	13073,75	14617,6	17678,93	21373,75	23542,38	25851,1	28764,7
	H	2,87	18,74	64,26	82,8	91,19	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
16	Q	12,13	128,03	952,28	2208,2	3972,23	5606,93	8966,78	9795,58	13074,25	14617,65	17678,93	21374,23	23542,38	25851,1	28764,7
	H	2,87	20,89	67,54	83,67	91,24	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
17	Q	12,18	141,13	1028,33	2265,05	3992,4	5615,85	8968,68	9796,5	13074,48	14617,65	17678,93	21374,43	23542,38	25851,1	28764,7
	H	2,87	22,63	70,05	84,69	91,24	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
18	Q	12,23	153,73	1092,28	2306,28	4003	5619,58	8968,83	9796,7	13074,48	14617,65	17678,93	21374,5	23542,38	25851,1	28764,7
	H	2,87	24,17	71,63	84,95	91,3	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
19	Q	12,25	165,78	1143,18	2334,78	4008,35	5621,15	8969,03	9796,88	13074,48	14617,65	17678,93	21374,5	23542,38	25851,1	28764,7
	H	2,87	25,81	73,27	85,36	91,3	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8
20	Q	12,28	177,73	1183,4	2352,68	4011,38	5621,75	8969,18	9797,03	13074,48	14617,65	17678,93	21374,5	23542,38	25851,1	28764,7
	H	2,87	27,09	74,45	85,36	91,3	93,96	96,01	96,31	97,9	98,41	98,98	99,28	99,49	99,64	99,8

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q=4.284,25 H=56,78%

**Tabla A-7 Resultados experimentales sobre red P2P con 10.000 nodos. Las celdas sombreadas corresponden a los casos en que BII-P2P supera a *Gnutella* en ambas métricas consideradas.**

TTL		Tasa de selección $\gamma$														
		0,20	0,25	0,30	0,35	0,40	0,45	0,50	0,55	0,60	0,65	0,70	0,75	0,80	0,85	0,90
4	Q	6,78	13,3	27,53	42,08	82,53	122,43	250,5	288,35	476,88	589	829,13	1202,13	1469,58	1821,1	2417,35
	H	0,53	0,98	1,59	2,24	3,43	4,36	7,14	7,59	10,37	11,68	14,02	17,35	19,7	21,97	26,05
5	Q	8,4	19,68	48,83	86,4	202,33	338,18	846,2	1017,25	1908,43	2487,18	3817,75	6018,88	7664,2	9955,9	13797,78
	H	0,78	1,51	3,23	5,17	9,66	12,79	23,03	25,15	35,23	40,05	49	59,67	65,85	72,13	79,97
6	Q	9,78	27,18	80,95	168,63	469,03	885,63	2606,33	3232,43	6543,43	8692,18	13673,1	21404,5	26737,28	33522,33	43214,43
	H	0,93	2,12	5,5	9,61	19,95	28,95	52,31	56,37	74,4	80,5	88,8	94,83	97,28	98,46	99,42
7	Q	10,88	35,63	129,13	315,43	1014,53	2100,53	6475,7	7994,03	15157	19150,1	27075,75	36739,25	42380,1	48500,53	55835,63
	H	1,13	2,88	8,8	16,24	35,96	52,38	80,25	83,51	93,27	95,41	97,91	98,94	99,34	99,6	99,82
8	Q	11,88	45,45	198,38	556,28	1967,95	4170,58	11366	13455,18	21721,25	25619,33	32828,1	41068,55	45858,8	50858,43	57061,88
	H	1,34	3,76	13,11	26,36	55,51	74,93	92,31	93,32	96,62	97,45	98,44	99,22	99,52	99,7	99,87
9	Q	12,58	56,6	295,5	927,88	3304,08	6606,73	14749,28	16803,68	24433,45	27935,15	34455	42083,95	46605,58	51271,88	57208,98
	H	1,39	4,56	18,54	39,42	72,64	86,43	94,88	95,49	97,28	97,83	98,56	99,27	99,55	99,72	99,87
10	Q	13,05	69,58	425,3	1431,15	4734,33	8571,85	16446,38	18340,05	25413,6	28713,5	34919,5	42351,1	46782,65	51359,15	57232
	H	1,39	5,5	25,55	51,55	82,17	91,1	95,61	96,04	97,45	97,96	98,61	99,27	99,57	99,72	99,87
11	Q	13,4	83,53	588	2020,35	5942,73	9793,43	17206,75	18970,5	25772,63	28979,13	35060,35	42423,83	46828,68	51378,4	57235,43
	H	1,51	6,96	33,34	62,65	87,49	92,36	95,71	96,14	97,5	97,98	98,64	99,27	99,57	99,72	99,87
12	Q	13,65	98,48	777,93	2618	6778	10461,95	17539,23	19231,7	25901,8	29073,23	35103,85	42446,65	46841,93	51382,05	57235,8
	H	1,51	8,2	40,78	70,95	89,23	92,94	95,79	96,22	97,53	97,98	98,64	99,27	99,57	99,72	99,87
13	Q	13,85	114,35	990,1	3159,85	7288,13	10805,03	17684,73	19339,7	25952,8	29105,18	35117,18	42454,53	46846,45	51382,83	57236,13
	H	1,51	9,46	47,84	76,77	90,06	93,19	95,81	96,27	97,55	97,98	98,64	99,27	99,57	99,72	99,87
14	Q	13,98	131,28	1204,2	3608,78	7579,78	10979,7	17750,13	19383,23	25970,33	29115,53	35121,6	42456,5	46847,83	51382,93	57236,13
	H	1,56	11,02	53,49	80,61	90,54	93,27	95,81	96,27	97,55	98,01	98,64	99,27	99,57	99,72	99,87
15	Q	14,1	148,9	1413,23	3953,23	7744,1	11066,7	17778,83	19403,4	25975,95	29119,05	35123,7	42457,4	46848,4	51382,93	57236,13
	H	1,56	12,28	58,66	82,6	90,72	93,27	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87
16	Q	14,18	166,73	1605,65	4197,63	7833,28	11110,18	17791,28	19410,38	25977,35	29119,63	35124,5	42457,7	46848,55	51382,93	57236,13
	H	1,59	13,27	62,57	83,81	90,87	93,27	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87
17	Q	14,23	185,5	1779,88	4360,13	7881,98	11129,05	17797,05	19413,43	25978,05	29119,65	35124,65	42457,7	46848,55	51382,93	57236,13
	H	1,59	14,68	65,62	84,41	90,97	93,29	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87
18	Q	14,28	205,15	1928,98	4463,65	7907,83	11138,5	17799,8	19414,68	25978,25	29119,65	35124,65	42457,7	46848,55	51382,93	57236,13
	H	1,59	15,91	68,27	84,84	91,02	93,29	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87
19	Q	14,28	225,78	2055,28	4528,5	7922,03	11144,35	17801,35	19415,8	25978,38	29119,65	35124,65	42457,7	46848,55	51382,93	57236,13
	H	1,59	17,5	70,19	85,12	91,02	93,29	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87
20	Q	14,28	247,03	2157,75	4567,9	7930,33	11147,25	17801,83	19416,08	25978,38	29119,65	35124,65	42457,7	46848,55	51382,93	57236,13
	H	1,59	18,94	71,88	85,37	91,05	93,29	95,81	96,27	97,58	98,01	98,64	99,27	99,57	99,72	99,87

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q=4.525,53 H=36,14%

**Tabla A-8 Resultados experimentales sobre red P2P con 20.000 nodos. Las celdas sombreadas corresponden a los casos en que BII-P2P supera a *Gnutella* en ambas métricas consideradas.**

TTL		Tasa de selección $\gamma$														
		0,20	0,25	0,30	0,35	0,40	0,45	0,50	0,55	0,60	0,65	0,70	0,75	0,80	0,85	0,90
4	Q	6,63	12,75	26,2	42,63	83,73	128,2	258,93	311,6	491,13	618,35	859,65	1310,65	1580,28	1983,05	2556,23
	H	0,28	0,52	0,82	1,19	1,88	2,31	3,68	4,21	5,46	6,32	7,8	10,18	11,41	12,92	14,95
5	Q	8,05	18,38	45,33	87,18	204,95	354,93	878	1103,03	1993,8	2657,43	4072,93	6814,48	8633,98	11442,25	15632,15
	H	0,45	0,9	1,61	2,46	4,34	6,31	11,85	13,78	19,98	23,82	30,41	41,37	46,51	52,66	60,57
6	Q	9,18	25,18	75,13	170,13	481,03	946,7	2840,38	3691,4	7427,58	10244,85	16621,95	28605	36742,75	48186,83	64651,85
	H	0,48	1,26	2,75	4,63	9,95	15,77	32,92	38	54,83	63,07	74,25	87,43	91,38	94,36	97,07
7	Q	9,88	32,7	119,1	321,55	1086,18	2387,53	8076,85	10496,45	21227,23	28286,58	42957,65	64585,43	76994,13	90697,8	107379,9
	H	0,52	1,55	4,13	8,15	20,98	35,4	65,08	71,04	86,29	90,79	95,58	98,61	99,24	99,53	99,91
8	Q	10,4	41,55	184,45	587,5	2284,98	5344,48	17416,68	21547,65	37773	46125,95	61586,03	80033,4	90171,3	100664,63	113660,63
	H	0,56	1,94	6,07	14,61	37,22	58,33	86,9	90,05	95,95	97,2	98,31	99,08	99,5	99,63	99,95
9	Q	10,78	51,85	280,48	1032,15	4297,08	9776,68	26318,1	30680,15	46592	54073,18	67683,23	83546,55	92788,93	102211,45	114285,68
	H	0,58	2,35	9,18	23,97	56,17	76,43	93,26	94,61	97,21	97,86	98,55	99,09	99,53	99,68	99,96
10	Q	10,95	64,13	418,9	1716,28	6967,33	14336,65	31374,53	35320,88	49937	56763,1	69387,83	84403,08	93391,55	102508,98	114376,98
	H	0,59	2,95	13,66	35,38	71,35	85,92	95,02	95,79	97,46	98,02	98,57	99,12	99,56	99,68	99,96
11	Q	11,05	78,33	606,6	2654,45	9739,68	17808,35	33703,5	37308,93	51160,78	57634,05	69874,3	84648	93549,48	102581,05	114391,95
	H	0,59	3,62	19,35	48,13	81,35	90,29	95,57	96,09	97,55	98,08	98,57	99,13	99,56	99,68	99,96
12	Q	11,13	94,55	851,63	3788,35	12052,55	19986,33	34743,38	38129,05	51610,1	57929,78	70024,43	84714,25	93589,98	102594,98	114394,65
	H	0,61	4,28	25,32	59,99	86,29	92,06	95,67	96,15	97,62	98,11	98,57	99,13	99,56	99,68	99,96
13	Q	11,18	112,68	1159,05	4993,03	13677,83	21191,4	35193,7	38464,7	51774,85	58035,23	70073,8	84732,6	93600,88	102597,1	114394,9
	H	0,61	5,26	32,37	68,46	88,15	92,59	95,74	96,19	97,62	98,11	98,57	99,13	99,56	99,68	99,96
14	Q	11,18	133,38	1525,53	6133	14688	21808,13	35386,6	38600,88	51833,13	58073,9	70089,2	84739,78	93605,05	102597,7	114394,9
	H	0,61	6,23	39,81	74,53	89,17	92,87	95,79	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
15	Q	11,18	154,58	1938,13	7100,6	15272,95	22123,3	35467,93	38655,38	51854,68	58089,3	70094,03	84741,58	93606,43	102598,18	114394,9
	H	0,61	6,96	46,27	78,73	89,65	92,96	95,81	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
16	Q	11,18	177,3	2375,65	7858,08	15599,6	22279,15	35503,73	38680,68	51861,68	58093,03	70095,2	84742,08	93606,65	102598,3	114394,9
	H	0,61	8,05	52,83	81,18	89,88	93	95,82	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
17	Q	11,18	201,98	2817,33	8393,75	15770,93	22352,23	35517,25	38691,43	51864,38	58094,4	70095,75	84742,08	93606,65	102598,43	114394,9
	H	0,61	9,12	57,67	82,64	89,98	93,04	95,82	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
18	Q	11,18	229,33	3239,4	8757,73	15867,15	22389,23	35523,83	38696,7	51865,28	58095,55	70096,95	84742,08	93606,65	102598,43	114394,9
	H	0,61	10,12	61,97	83,56	90,05	93,04	95,82	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
19	Q	11,18	258,68	3623,45	8993,7	15919,05	22406,45	35527,78	38698,98	51865,98	58095,9	70097,53	84742,08	93606,65	102598,43	114394,9
	H	0,61	11,27	65,59	84,08	90,07	93,04	95,82	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96
20	Q	11,18	290,03	3954,7	9143,08	15948,35	22415,35	35529,83	38700,23	51866,2	58096,13	70097,73	84742,08	93606,65	102598,43	114394,9
	H	0,61	12,58	68,19	84,24	90,08	93,04	95,82	96,19	97,63	98,12	98,57	99,13	99,56	99,68	99,96

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q=4.791,05 H=21,26%

**Tabla A-9 Rendimiento de BII-P2P bajo política de actualización DLPA<sub>(1;75)</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	33,75	5,94	0,1759	7	0,30	193,38	31,58	0,1633	10	0,30	595,88	59,81	0,1004	13	0,30	1026,53	74,44	0,0725
4	0,35	56,23	8,09	0,1439	7	0,35	470,20	49,05	0,1043	10	0,35	1467,63	78,06	0,0532	13	0,35	2104,45	85,37	0,0406
4	0,40	99,28	12,48	0,1257	7	0,40	1198,28	70,91	0,0592	10	0,40	3141,65	88,21	0,0281	13	0,40	3697,75	89,50	0,0242
4	0,45	159,28	15,75	0,0989	7	0,45	2259,43	81,58	0,0361	10	0,45	4619,78	92,25	0,0200	13	0,45	5014,95	92,86	0,0185
4	0,50	304,75	22,46	0,0737	7	0,50	5016,93	92,25	0,0184	10	0,50	7412,70	96,04	0,0130	13	0,50	7566,65	96,13	0,0127
4	0,55	352,78	24,27	0,0688	7	0,55	5898,08	94,06	0,0159	10	0,55	8148,73	96,56	0,0118	13	0,55	8250,10	96,64	0,0117
4	0,60	574,80	31,67	0,0551	7	0,60	8808,03	97,42	0,0111	10	0,60	10265,60	98,11	0,0096	13	0,60	10305,03	98,19	0,0095
4	0,65	696,05	34,68	0,0498	7	0,65	10125,08	98,36	0,0097	10	0,65	11219,48	98,71	0,0088	13	0,65	11234,98	98,71	0,0088
4	0,70	973,50	40,62	0,0417	7	0,70	12249,05	99,31	0,0081	10	0,70	12823,75	99,31	0,0077	13	0,70	12827,00	99,31	0,0077
4	0,75	1357,25	46,39	0,0342	7	0,75	14327,98	99,66	0,0070	10	0,75	14561,03	99,66	0,0068	13	0,75	14561,23	99,66	0,0068
4	0,80	1622,63	50,77	0,0313	7	0,80	15434,80	99,74	0,0065	10	0,80	15555,03	99,74	0,0064	13	0,80	15555,10	99,74	0,0064
5	0,30	63,25	11,45	0,1810	8	0,30	305,03	41,91	0,1374	11	0,30	751,70	66,01	0,0878	14	0,30	1132,20	77,02	0,0680
5	0,35	123,23	16,70	0,1355	8	0,35	778,33	62,48	0,0803	11	0,35	1748,30	81,93	0,0469	14	0,35	2198,93	85,89	0,0391
5	0,40	255,70	26,76	0,1047	8	0,40	1962,20	81,93	0,0418	11	0,40	3443,88	89,33	0,0259	14	0,40	3742,08	89,59	0,0239
5	0,45	454,95	35,71	0,0785	8	0,45	3395,25	87,69	0,0258	11	0,45	4853,78	92,43	0,0190	14	0,45	5038,45	92,94	0,0184
5	0,50	1025,20	53,96	0,0526	8	0,50	6499,93	94,75	0,0146	11	0,50	7511,18	96,13	0,0128	14	0,50	7572,08	96,13	0,0127
5	0,55	1231,70	57,23	0,0465	8	0,55	7382,08	95,96	0,0130	11	0,55	8217,65	96,64	0,0118	14	0,55	8252,88	96,64	0,0117
5	0,60	2203,78	71,26	0,0323	8	0,60	9886,28	97,93	0,0099	11	0,60	10292,83	98,19	0,0095	14	0,60	10305,98	98,19	0,0095
5	0,65	2771,90	75,39	0,0272	8	0,65	10979,50	98,54	0,0090	11	0,65	11231,18	98,71	0,0088	14	0,65	11235,25	98,71	0,0088
5	0,70	4057,35	83,13	0,0205	8	0,70	12727,93	99,31	0,0078	11	0,70	12826,43	99,31	0,0077	14	0,70	12827,00	99,31	0,0077
5	0,75	5829,45	89,50	0,0154	8	0,75	14533,90	99,66	0,0069	11	0,75	14561,23	99,66	0,0068	14	0,75	14561,23	99,66	0,0068
5	0,80	7050,43	92,60	0,0131	8	0,80	15542,48	99,74	0,0064	11	0,80	15555,10	99,74	0,0064	14	0,80	15555,10	99,74	0,0064
6	0,30	113,93	20,14	0,1768	9	0,30	441,98	51,20	0,1159	12	0,30	897,00	70,48	0,0786	15	0,30	1215,85	78,92	0,0649
6	0,35	252,40	33,56	0,1330	9	0,35	1131,98	71,77	0,0634	12	0,35	1957,38	84,25	0,0430	15	0,35	2262,93	86,06	0,0380
6	0,40	596,13	52,93	0,0888	9	0,40	2652,80	86,49	0,0326	12	0,40	3613,30	89,33	0,0247	15	0,40	3765,40	89,59	0,0238
6	0,45	1140,18	65,15	0,0571	9	0,45	4175,55	91,14	0,0218	12	0,45	4963,93	92,69	0,0187	15	0,45	5048,10	92,94	0,0184
6	0,50	2751,10	83,22	0,0302	9	0,50	7150,85	95,78	0,0134	12	0,50	7551,25	96,13	0,0127	15	0,50	7573,85	96,13	0,0127
6	0,55	3329,95	85,28	0,0256	9	0,55	7948,25	96,56	0,0121	12	0,55	8242,00	96,64	0,0117	15	0,55	8253,70	96,64	0,0117
6	0,60	5779,53	94,23	0,0163	9	0,60	10177,58	98,11	0,0096	12	0,60	10302,20	98,19	0,0095	15	0,60	10306,35	98,19	0,0095
6	0,65	7101,13	96,30	0,0136	9	0,65	11172,75	98,71	0,0088	12	0,65	11234,13	98,71	0,0088	15	0,65	11235,25	98,71	0,0088
6	0,70	9604,95	98,45	0,0103	9	0,70	12809,55	99,31	0,0078	12	0,70	12826,93	99,31	0,0077	15	0,70	12827,00	99,31	0,0077
6	0,75	12433,25	99,31	0,0080	9	0,75	14559,00	99,66	0,0068	12	0,75	14561,23	99,66	0,0068	15	0,75	14561,23	99,66	0,0068
6	0,80	14020,50	99,74	0,0071	9	0,80	15553,73	99,74	0,0064	12	0,80	15555,10	99,74	0,0064	15	0,80	15555,10	99,74	0,0064

DLPA (1;75) Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q  
Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-10 Rendimiento de BII-P2P bajo política de actualización DLPA<sub>(2;75)</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	29,50	5,25	0,1780	7	0,30	166,05	29,35	0,1767	10	0,30	501,03	59,04	0,1178	13	0,30	875,23	72,29	0,0826
4	0,35	52,08	7,83	0,1504	7	0,35	402,78	45,35	0,1126	10	0,35	1202,03	75,47	0,0628	13	0,35	1827,23	84,08	0,0460
4	0,40	90,00	11,02	0,1224	7	0,40	1020,00	66,27	0,0650	10	0,40	2794,93	87,95	0,0315	13	0,40	3467,38	90,45	0,0261
4	0,45	142,98	14,97	0,1047	7	0,45	1937,30	78,74	0,0406	10	0,45	4320,13	91,82	0,0213	13	0,45	4830,10	93,03	0,0193
4	0,50	280,58	21,00	0,0748	7	0,50	4627,68	91,57	0,0198	10	0,50	7297,23	96,47	0,0132	13	0,50	7521,33	96,64	0,0128
4	0,55	323,08	22,55	0,0698	7	0,55	5449,60	93,37	0,0171	10	0,55	8056,80	97,16	0,0121	13	0,55	8229,20	97,16	0,0118
4	0,60	530,28	30,03	0,0566	7	0,60	8471,70	98,11	0,0116	10	0,60	10342,15	98,88	0,0096	13	0,60	10403,03	98,88	0,0095
4	0,65	636,50	32,96	0,0518	7	0,65	9865,10	98,71	0,0100	10	0,65	11344,80	99,23	0,0087	13	0,65	11373,70	99,23	0,0087
4	0,70	896,80	38,04	0,0424	7	0,70	12179,08	99,31	0,0082	10	0,70	12977,43	99,57	0,0077	13	0,70	12984,93	99,57	0,0077
4	0,75	1311,18	45,44	0,0347	7	0,75	14430,90	99,74	0,0069	10	0,75	14695,40	99,83	0,0068	13	0,75	14695,98	99,83	0,0068
4	0,80	1572,23	49,57	0,0315	7	0,80	15551,03	99,83	0,0064	10	0,80	15666,45	99,83	0,0064	13	0,80	15666,45	99,83	0,0064
5	0,30	56,10	9,98	0,1779	8	0,30	256,48	40,10	0,1564	11	0,30	636,23	66,18	0,1040	14	0,30	968,43	74,78	0,0772
5	0,35	113,50	15,23	0,1342	8	0,35	640,65	57,06	0,0891	11	0,35	1456,20	79,52	0,0546	14	0,35	1948,83	84,94	0,0436
5	0,40	227,03	24,01	0,1058	8	0,40	1660,93	77,71	0,0468	11	0,40	3139,20	89,07	0,0284	14	0,40	3533,48	90,62	0,0256
5	0,45	400,48	33,48	0,0836	8	0,45	2984,20	86,92	0,0291	11	0,45	4613,68	92,34	0,0200	14	0,45	4861,45	93,12	0,0192
5	0,50	930,80	50,86	0,0546	8	0,50	6186,05	95,44	0,0154	11	0,50	7438,08	96,64	0,0130	14	0,50	7532,35	96,64	0,0128
5	0,55	1111,13	54,04	0,0486	8	0,55	7058,75	96,30	0,0136	11	0,55	8167,78	97,16	0,0119	14	0,55	8235,55	97,16	0,0118
5	0,60	2003,48	68,76	0,0343	8	0,60	9799,00	98,88	0,0101	11	0,60	10384,70	98,88	0,0095	14	0,60	10404,08	98,88	0,0095
5	0,65	2505,10	73,58	0,0294	8	0,65	10972,78	99,23	0,0090	11	0,65	11367,45	99,23	0,0087	14	0,65	11373,95	99,23	0,0087
5	0,70	3706,08	81,58	0,0220	8	0,70	12830,40	99,57	0,0078	11	0,70	12983,78	99,57	0,0077	14	0,70	12984,93	99,57	0,0077
5	0,75	5606,00	88,90	0,0159	8	0,75	14661,70	99,83	0,0068	11	0,75	14695,90	99,83	0,0068	14	0,75	14695,98	99,83	0,0068
5	0,80	6829,03	92,34	0,0135	8	0,80	15656,48	99,83	0,0064	11	0,80	15666,45	99,83	0,0064	14	0,80	15666,45	99,83	0,0064
6	0,30	100,05	18,24	0,1824	9	0,30	370,43	50,52	0,1364	12	0,30	763,05	69,45	0,0910	15	0,30	1043,83	76,59	0,0734
6	0,35	224,95	29,78	0,1324	9	0,35	917,90	68,67	0,0748	12	0,35	1666,50	82,62	0,0496	15	0,35	2032,98	85,37	0,0420
6	0,40	517,95	47,93	0,0925	9	0,40	2289,53	84,42	0,0369	12	0,40	3347,88	89,59	0,0268	15	0,40	3572,33	90,71	0,0254
6	0,45	979,93	61,79	0,0631	9	0,45	3799,68	90,62	0,0238	12	0,45	4762,65	92,69	0,0195	15	0,45	4877,83	93,20	0,0191
6	0,50	2493,65	79,86	0,0320	9	0,50	6967,43	96,13	0,0138	12	0,50	7495,55	96,64	0,0129	15	0,50	7537,10	96,64	0,0128
6	0,55	3010,70	82,19	0,0273	9	0,55	7776,33	96,82	0,0125	12	0,55	8212,60	97,16	0,0118	15	0,55	8238,48	97,16	0,0118
6	0,60	5341,10	93,37	0,0175	9	0,60	10211,95	98,88	0,0097	12	0,60	10398,63	98,88	0,0095	15	0,60	10404,43	98,88	0,0095
6	0,65	6596,70	95,78	0,0145	9	0,65	11258,70	99,23	0,0088	12	0,65	11373,08	99,23	0,0087	15	0,65	11373,95	99,23	0,0087
6	0,70	9108,85	97,76	0,0107	9	0,70	12950,80	99,57	0,0077	12	0,70	12984,78	99,57	0,0077	15	0,70	12984,93	99,57	0,0077
6	0,75	12282,20	99,31	0,0081	9	0,75	14690,85	99,83	0,0068	12	0,75	14695,98	99,83	0,0068	15	0,75	14695,98	99,83	0,0068
6	0,80	13970,80	99,83	0,0071	9	0,80	15665,88	99,83	0,0064	12	0,80	15666,45	99,83	0,0064	15	0,80	15666,45	99,83	0,0064

**DLPA (2;75)**

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166



**Tabla A-11 Rendimiento de BII-P2P bajo política DLPA<sub>(3;75)</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	29,45	5,42	0,1841	7	0,30	151,30	27,37	0,1809	10	0,30	451,78	57,66	0,1276	13	0,30	814,75	74,01	0,0908
4	0,35	50,65	7,40	0,1461	7	0,35	378,00	43,29	0,1145	10	0,35	1160,08	76,42	0,0659	13	0,35	1763,93	85,71	0,0486
4	0,40	87,58	10,93	0,1248	7	0,40	987,75	65,06	0,0659	10	0,40	2692,53	89,24	0,0331	13	0,40	3302,63	91,39	0,0277
4	0,45	136,53	14,20	0,1040	7	0,45	1841,83	78,57	0,0427	10	0,45	4113,55	92,77	0,0226	13	0,45	4596,00	94,15	0,0205
4	0,50	266,90	19,97	0,0748	7	0,50	4421,05	93,55	0,0212	10	0,50	7047,30	96,90	0,0138	13	0,50	7266,78	96,99	0,0133
4	0,55	300,03	21,00	0,0700	7	0,55	5182,53	95,09	0,0183	10	0,55	7839,18	97,42	0,0124	13	0,55	8021,60	97,50	0,0122
4	0,60	502,45	28,23	0,0562	7	0,60	8252,28	98,11	0,0119	10	0,60	10213,13	98,88	0,0097	13	0,60	10287,15	98,88	0,0096
4	0,65	616,55	31,50	0,0511	7	0,65	9742,88	99,05	0,0102	10	0,65	11272,35	99,66	0,0088	13	0,65	11305,73	99,66	0,0088
4	0,70	880,13	37,18	0,0422	7	0,70	12136,95	99,83	0,0082	10	0,70	12961,33	99,91	0,0077	13	0,70	12967,60	99,91	0,0077
4	0,75	1287,90	44,58	0,0346	7	0,75	14461,53	99,91	0,0069	10	0,75	14723,08	99,91	0,0068	13	0,75	14723,30	99,91	0,0068
4	0,80	1548,05	48,97	0,0316	7	0,80	15589,65	99,91	0,0064	10	0,80	15695,28	99,91	0,0064	13	0,80	15695,28	99,91	0,0064
5	0,30	54,23	10,93	0,2016	8	0,30	231,55	37,09	0,1602	11	0,30	577,93	65,92	0,1141	14	0,30	912,90	77,19	0,0846
5	0,35	107,50	15,40	0,1433	8	0,35	609,65	55,85	0,0916	11	0,35	1406,18	81,67	0,0581	14	0,35	1870,65	86,14	0,0461
5	0,40	218,83	23,49	0,1074	8	0,40	1619,70	77,71	0,0480	11	0,40	3005,43	90,36	0,0301	14	0,40	3361,33	91,39	0,0272
5	0,45	379,78	33,30	0,0877	8	0,45	2841,25	87,95	0,0310	11	0,45	4385,85	93,72	0,0214	14	0,45	4627,80	94,15	0,0203
5	0,50	878,58	49,66	0,0565	8	0,50	5961,13	95,78	0,0161	11	0,50	7186,58	96,90	0,0135	14	0,50	7277,63	96,99	0,0133
5	0,55	1025,85	52,50	0,0512	8	0,55	6822,58	96,64	0,0142	11	0,55	7954,38	97,42	0,0122	14	0,55	8029,03	97,50	0,0121
5	0,60	1884,58	68,24	0,0362	8	0,60	9630,35	98,62	0,0102	11	0,60	10264,03	98,88	0,0096	14	0,60	10288,85	98,88	0,0096
5	0,65	2415,08	73,49	0,0304	8	0,65	10877,93	99,57	0,0092	11	0,65	11296,10	99,66	0,0088	14	0,65	11306,58	99,66	0,0088
5	0,70	3629,78	81,84	0,0225	8	0,70	12815,40	99,91	0,0078	11	0,70	12966,60	99,91	0,0077	14	0,70	12967,60	99,91	0,0077
5	0,75	5526,03	89,76	0,0162	8	0,75	14693,93	99,91	0,0068	11	0,75	14723,15	99,91	0,0068	14	0,75	14723,30	99,91	0,0068
5	0,80	6738,00	93,37	0,0139	8	0,80	15688,65	99,91	0,0064	11	0,80	15695,28	99,91	0,0064	14	0,80	15695,28	99,91	0,0064
6	0,30	93,18	17,99	0,1930	9	0,30	334,40	48,54	0,1451	12	0,30	701,60	71,17	0,1014	15	0,30	994,03	78,83	0,0793
6	0,35	210,85	29,26	0,1388	9	0,35	884,40	68,42	0,0774	12	0,35	1609,08	84,42	0,0525	15	0,35	1940,85	86,66	0,0447
6	0,40	498,65	45,52	0,0913	9	0,40	2227,43	85,54	0,0384	12	0,40	3194,43	91,05	0,0285	15	0,40	3394,35	91,39	0,0269
6	0,45	927,40	59,90	0,0646	9	0,45	3624,03	91,14	0,0251	12	0,45	4527,35	94,15	0,0208	15	0,45	4641,83	94,15	0,0203
6	0,50	2351,98	80,03	0,0340	9	0,50	6718,28	96,56	0,0144	12	0,50	7242,65	96,99	0,0134	15	0,50	7281,73	96,99	0,0133
6	0,55	2794,83	83,13	0,0297	9	0,55	7546,08	97,07	0,0129	12	0,55	8002,08	97,42	0,0122	15	0,55	8030,85	97,50	0,0121
6	0,60	5082,43	93,89	0,0185	9	0,60	10068,53	98,71	0,0098	12	0,60	10281,08	98,88	0,0096	15	0,60	10289,60	98,88	0,0096
6	0,65	6421,78	95,87	0,0149	9	0,65	11184,38	99,57	0,0089	12	0,65	11303,23	99,66	0,0088	15	0,65	11307,13	99,66	0,0088
6	0,70	9003,50	98,62	0,0110	9	0,70	12937,18	99,91	0,0077	12	0,70	12967,43	99,91	0,0077	15	0,70	12967,60	99,91	0,0077
6	0,75	12274,68	99,74	0,0081	9	0,75	14719,48	99,91	0,0068	12	0,75	14723,30	99,91	0,0068	15	0,75	14723,30	99,91	0,0068
6	0,80	13960,38	99,91	0,0072	9	0,80	15694,85	99,91	0,0064	12	0,80	15695,28	99,91	0,0064	15	0,80	15695,28	99,91	0,0064

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q  
Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-12 Rendimiento de BII-P2P bajo política DLPA<sub>(4;75)</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	28,08	5,42	0,1931	7	0,30	143,63	25,73	0,1792	10	0,30	441,95	55,77	0,1262	13	0,30	785,93	71,08	0,0904
4	0,35	50,15	7,40	0,1476	7	0,35	373,88	42,34	0,1132	10	0,35	1144,53	75,73	0,0662	13	0,35	1745,93	85,54	0,0490
4	0,40	86,30	10,67	0,1237	7	0,40	985,15	65,15	0,0661	10	0,40	2687,10	88,90	0,0331	13	0,40	3289,98	91,57	0,0278
4	0,45	139,10	14,54	0,1046	7	0,45	1882,13	78,57	0,0417	10	0,45	4136,98	93,89	0,0227	13	0,45	4592,90	94,49	0,0206
4	0,50	268,23	20,40	0,0760	7	0,50	4430,90	93,55	0,0211	10	0,50	7033,80	96,64	0,0137	13	0,50	7253,25	96,90	0,0134
4	0,55	300,98	21,08	0,0701	7	0,55	5186,45	94,66	0,0183	10	0,55	7829,73	97,16	0,0124	13	0,55	8009,58	97,25	0,0121
4	0,60	501,35	28,31	0,0565	7	0,60	8252,30	97,93	0,0119	10	0,60	10209,75	98,80	0,0097	13	0,60	10281,78	98,80	0,0096
4	0,65	615,05	31,41	0,0511	7	0,65	9761,13	99,14	0,0102	10	0,65	11263,23	99,48	0,0088	13	0,65	11301,28	99,48	0,0088
4	0,70	882,25	37,18	0,0421	7	0,70	12129,23	99,74	0,0082	10	0,70	12955,83	99,74	0,0077	13	0,70	12963,23	99,74	0,0077
4	0,75	1293,10	45,09	0,0349	7	0,75	14462,33	99,91	0,0069	10	0,75	14722,98	99,91	0,0068	13	0,75	14723,25	99,91	0,0068
4	0,80	1546,20	48,88	0,0316	7	0,80	15588,48	99,91	0,0064	10	0,80	15694,20	99,91	0,0064	13	0,80	15694,20	99,91	0,0064
5	0,30	51,13	10,84	0,2121	8	0,30	224,13	35,20	0,1570	11	0,30	561,73	62,56	0,1114	14	0,30	877,80	73,75	0,0840
5	0,35	107,05	15,83	0,1479	8	0,35	601,70	55,59	0,0924	11	0,35	1389,48	81,41	0,0586	14	0,35	1853,53	86,32	0,0466
5	0,40	217,23	23,92	0,1101	8	0,40	1618,60	77,80	0,0481	11	0,40	2996,10	90,28	0,0301	14	0,40	3349,40	91,57	0,0273
5	0,45	387,28	34,34	0,0887	8	0,45	2891,00	88,30	0,0305	11	0,45	4396,25	94,23	0,0214	14	0,45	4621,45	94,49	0,0204
5	0,50	880,15	49,40	0,0561	8	0,50	5963,10	96,04	0,0161	11	0,50	7168,33	96,73	0,0135	14	0,50	7265,28	96,90	0,0133
5	0,55	1025,60	51,89	0,0506	8	0,55	6820,20	96,64	0,0142	11	0,55	7944,43	97,16	0,0122	14	0,55	8018,10	97,25	0,0121
5	0,60	1882,00	67,81	0,0360	8	0,60	9624,55	98,54	0,0102	11	0,60	10259,43	98,80	0,0096	14	0,60	10284,83	98,80	0,0096
5	0,65	2415,48	72,72	0,0301	8	0,65	10879,70	99,48	0,0091	11	0,65	11290,43	99,48	0,0088	14	0,65	11302,43	99,48	0,0088
5	0,70	3633,75	82,27	0,0226	8	0,70	12804,80	99,74	0,0078	11	0,70	12961,75	99,74	0,0077	14	0,70	12963,23	99,74	0,0077
5	0,75	5539,13	89,85	0,0162	8	0,75	14694,33	99,91	0,0068	11	0,75	14723,18	99,91	0,0068	14	0,75	14723,25	99,91	0,0068
5	0,80	6731,93	93,29	0,0139	8	0,80	15686,43	99,91	0,0064	11	0,80	15694,20	99,91	0,0064	14	0,80	15694,20	99,91	0,0064
6	0,30	87,38	16,70	0,1911	9	0,30	326,23	46,39	0,1422	12	0,30	678,50	67,21	0,0991	15	0,30	955,50	75,65	0,0792
6	0,35	209,53	28,23	0,1347	9	0,35	870,03	67,81	0,0779	12	0,35	1592,15	84,25	0,0529	15	0,35	1923,30	86,92	0,0452
6	0,40	495,93	44,92	0,0906	9	0,40	2228,93	85,97	0,0386	12	0,40	3184,13	90,96	0,0286	15	0,40	3379,53	91,65	0,0271
6	0,45	948,30	61,10	0,0644	9	0,45	3664,18	92,17	0,0252	12	0,45	4526,78	94,41	0,0209	15	0,45	4635,18	94,49	0,0204
6	0,50	2352,03	80,38	0,0342	9	0,50	6709,30	96,64	0,0144	12	0,50	7227,18	96,82	0,0134	15	0,50	7271,03	96,90	0,0133
6	0,55	2790,93	82,79	0,0297	9	0,55	7539,58	96,99	0,0129	12	0,55	7989,93	97,16	0,0122	15	0,55	8020,85	97,25	0,0121
6	0,60	5077,38	92,94	0,0183	9	0,60	10062,23	98,80	0,0098	12	0,60	10275,48	98,80	0,0096	15	0,60	10286,10	98,80	0,0096
6	0,65	6432,70	95,52	0,0148	9	0,65	11176,95	99,48	0,0089	12	0,65	11298,00	99,48	0,0088	15	0,65	11302,98	99,48	0,0088
6	0,70	8992,58	98,45	0,0109	9	0,70	12929,60	99,74	0,0077	12	0,70	12962,95	99,74	0,0077	15	0,70	12963,23	99,74	0,0077
6	0,75	12275,60	99,74	0,0081	9	0,75	14720,18	99,91	0,0068	12	0,75	14723,25	99,91	0,0068	15	0,75	14723,25	99,91	0,0068
6	0,80	13960,83	99,91	0,0072	9	0,80	15693,75	99,91	0,0064	12	0,80	15694,20	99,91	0,0064	15	0,80	15694,20	99,91	0,0064

**DLPA (4;75)**

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-13 Rendimiento de BII-P2P bajo política DLPA<sub>(5;75)</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	28,50	5,85	0,2053	7	0,30	144,08	26,08	0,1810	10	0,30	442,80	56,02	0,1265	13	0,30	785,40	71,43	0,0909
4	0,35	50,70	7,83	0,1545	7	0,35	376,55	42,94	0,1140	10	0,35	1155,88	75,30	0,0651	13	0,35	1745,53	85,46	0,0490
4	0,40	86,25	10,84	0,1257	7	0,40	977,88	64,29	0,0657	10	0,40	2674,48	89,67	0,0335	13	0,40	3280,63	92,00	0,0280
4	0,45	136,65	14,37	0,1052	7	0,45	1855,55	78,49	0,0423	10	0,45	4118,95	93,72	0,0228	13	0,45	4571,70	94,32	0,0206
4	0,50	264,53	20,65	0,0781	7	0,50	4407,98	93,80	0,0213	10	0,50	7026,28	96,73	0,0138	13	0,50	7237,63	96,90	0,0134
4	0,55	297,55	21,69	0,0729	7	0,55	5176,90	95,18	0,0184	10	0,55	7824,25	97,25	0,0124	13	0,55	8000,63	97,25	0,0122
4	0,60	499,30	28,49	0,0571	7	0,60	8248,70	98,02	0,0119	10	0,60	10205,90	98,80	0,0097	13	0,60	10272,95	98,80	0,0096
4	0,65	611,65	31,84	0,0521	7	0,65	9748,25	99,05	0,0102	10	0,65	11251,95	99,48	0,0088	13	0,65	11288,68	99,48	0,0088
4	0,70	883,38	37,35	0,0423	7	0,70	12131,15	99,83	0,0082	10	0,70	12959,20	99,83	0,0077	13	0,70	12964,98	99,83	0,0077
4	0,75	1295,43	44,84	0,0346	7	0,75	14466,78	99,91	0,0069	10	0,75	14721,63	99,91	0,0068	13	0,75	14722,28	99,91	0,0068
4	0,80	1546,20	48,80	0,0316	7	0,80	15591,13	99,91	0,0064	10	0,80	15695,03	99,91	0,0064	13	0,80	15695,03	99,91	0,0064
5	0,30	51,63	11,36	0,2200	8	0,30	224,83	35,71	0,1589	11	0,30	563,43	63,08	0,1120	14	0,30	875,08	74,10	0,0847
5	0,35	107,38	16,35	0,1523	8	0,35	608,30	56,97	0,0937	11	0,35	1397,43	80,81	0,0578	14	0,35	1852,95	86,49	0,0467
5	0,40	216,00	24,10	0,1116	8	0,40	1608,35	77,62	0,0483	11	0,40	2982,15	90,96	0,0305	14	0,40	3341,35	92,00	0,0275
5	0,45	379,48	33,13	0,0873	8	0,45	2868,30	88,81	0,0310	11	0,45	4377,35	93,80	0,0214	14	0,45	4603,30	94,41	0,0205
5	0,50	871,05	49,83	0,0572	8	0,50	5942,28	95,96	0,0161	11	0,50	7158,45	96,90	0,0135	14	0,50	7247,75	96,90	0,0134
5	0,55	1016,23	52,67	0,0518	8	0,55	6808,98	96,56	0,0142	11	0,55	7934,38	97,25	0,0123	14	0,55	8007,03	97,25	0,0121
5	0,60	1875,95	67,90	0,0362	8	0,60	9625,13	98,36	0,0102	11	0,60	10253,80	98,80	0,0096	14	0,60	10274,13	98,80	0,0096
5	0,65	2398,45	72,46	0,0302	8	0,65	10873,18	99,48	0,0091	11	0,65	11277,75	99,48	0,0088	14	0,65	11289,38	99,48	0,0088
5	0,70	3636,83	82,19	0,0226	8	0,70	12805,73	99,83	0,0078	11	0,70	12964,03	99,83	0,0077	14	0,70	12964,98	99,83	0,0077
5	0,75	5544,25	89,76	0,0162	8	0,75	14694,93	99,91	0,0068	11	0,75	14722,13	99,91	0,0068	14	0,75	14722,28	99,91	0,0068
5	0,80	6731,03	93,29	0,0139	8	0,80	15687,48	99,91	0,0064	11	0,80	15695,03	99,91	0,0064	14	0,80	15695,03	99,91	0,0064
6	0,30	88,33	16,95	0,1919	9	0,30	326,48	47,25	0,1447	12	0,30	680,18	67,56	0,0993	15	0,30	950,93	75,56	0,0795
6	0,35	210,73	28,31	0,1344	9	0,35	879,30	68,85	0,0783	12	0,35	1595,68	83,91	0,0526	15	0,35	1920,73	87,18	0,0454
6	0,40	493,73	44,23	0,0896	9	0,40	2212,38	86,06	0,0389	12	0,40	3170,83	91,74	0,0289	15	0,40	3372,55	92,08	0,0273
6	0,45	930,75	61,27	0,0658	9	0,45	3647,80	92,77	0,0254	12	0,45	4506,05	94,23	0,0209	15	0,45	4619,83	94,41	0,0204
6	0,50	2336,65	80,46	0,0344	9	0,50	6698,55	96,73	0,0144	12	0,50	7214,80	96,90	0,0134	15	0,50	7251,85	96,90	0,0134
6	0,55	2781,00	83,05	0,0299	9	0,55	7536,05	97,07	0,0129	12	0,55	7981,33	97,25	0,0122	15	0,55	8008,75	97,25	0,0121
6	0,60	5071,15	93,12	0,0184	9	0,60	10062,33	98,71	0,0098	12	0,60	10268,53	98,80	0,0096	15	0,60	10274,55	98,80	0,0096
6	0,65	6406,88	95,52	0,0149	9	0,65	11165,93	99,48	0,0089	12	0,65	11285,85	99,48	0,0088	15	0,65	11289,58	99,48	0,0088
6	0,70	9000,58	98,62	0,0110	9	0,70	12932,55	99,83	0,0077	12	0,70	12964,80	99,83	0,0077	15	0,70	12964,98	99,83	0,0077
6	0,75	12288,88	99,74	0,0081	9	0,75	14718,00	99,91	0,0068	12	0,75	14722,28	99,91	0,0068	15	0,75	14722,28	99,91	0,0068
6	0,80	13956,60	99,91	0,0072	9	0,80	15694,28	99,91	0,0064	12	0,80	15695,03	99,91	0,0064	15	0,80	15695,03	99,91	0,0064

**DLPA** (5;75)

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-14 Rendimiento de BII-P2P bajo política DLP<sub>1</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	31,55	5,51	0,1746	7	0,30	149,00	29,35	0,1970	10	0,30	360,75	49,40	0,1369	13	0,30	568,40	60,41	0,1063
4	0,35	55,03	7,92	0,1439	7	0,35	373,23	44,58	0,1194	10	0,35	931,00	67,90	0,0729	13	0,35	1285,93	73,92	0,0575
4	0,40	97,90	12,31	0,1257	7	0,40	953,55	64,20	0,0673	10	0,40	2077,30	79,86	0,0384	13	0,40	2435,65	81,76	0,0336
4	0,45	156,70	16,52	0,1054	7	0,45	1756,05	75,30	0,0429	10	0,45	3201,98	84,17	0,0263	13	0,45	3505,80	84,77	0,0242
4	0,50	293,93	22,81	0,0776	7	0,50	3815,35	85,89	0,0225	10	0,50	5470,05	88,98	0,0163	13	0,50	5615,63	89,07	0,0159
4	0,55	339,58	24,70	0,0727	7	0,55	4481,05	87,69	0,0196	10	0,55	6079,90	89,93	0,0148	13	0,55	6198,48	90,19	0,0146
4	0,60	561,15	33,82	0,0603	7	0,60	6862,98	91,57	0,0133	10	0,60	8140,75	93,20	0,0114	13	0,60	8203,55	93,20	0,0114
4	0,65	680,83	36,32	0,0533	7	0,65	8022,40	93,12	0,0116	10	0,65	9148,08	94,32	0,0103	13	0,65	9198,83	94,32	0,0103
4	0,70	967,93	42,77	0,0442	7	0,70	10156,70	95,27	0,0094	10	0,70	11001,23	95,78	0,0087	13	0,70	11027,78	95,78	0,0087
4	0,75	1380,85	49,57	0,0359	7	0,75	12623,53	97,33	0,0077	10	0,75	13180,03	97,68	0,0074	13	0,75	13194,83	97,68	0,0074
4	0,80	1649,23	53,79	0,0326	7	0,80	14015,83	98,11	0,0070	10	0,80	14438,08	98,36	0,0068	13	0,80	14447,20	98,36	0,0068
5	0,30	57,30	10,67	0,1862	8	0,30	213,65	36,49	0,1708	11	0,30	435,98	54,04	0,1240	14	0,30	619,98	62,39	0,1006
5	0,35	114,85	17,47	0,1521	8	0,35	560,45	56,02	0,1000	11	0,35	1083,03	71,08	0,0656	14	0,35	1345,93	74,35	0,0552
5	0,40	242,48	27,62	0,1139	8	0,40	1416,90	73,92	0,0522	11	0,40	2260,15	80,81	0,0358	14	0,40	2474,58	82,10	0,0332
5	0,45	431,43	38,21	0,0886	8	0,45	2449,38	81,33	0,0332	11	0,45	3367,20	84,51	0,0251	14	0,45	3530,53	84,77	0,0240
5	0,50	945,88	54,39	0,0575	8	0,50	4782,90	87,78	0,0184	11	0,50	5562,15	88,98	0,0160	14	0,50	5623,05	89,07	0,0158
5	0,55	1137,23	58,26	0,0512	8	0,55	5452,88	89,07	0,0163	11	0,55	6155,60	90,02	0,0146	14	0,55	6203,98	90,19	0,0145
5	0,60	2033,85	71,77	0,0353	8	0,60	7721,80	92,86	0,0120	11	0,60	8183,65	93,20	0,0114	14	0,60	8205,53	93,20	0,0114
5	0,65	2557,93	75,99	0,0297	8	0,65	8801,30	93,98	0,0107	11	0,65	9180,50	94,32	0,0103	14	0,65	9200,30	94,32	0,0103
5	0,70	3785,65	82,10	0,0217	8	0,70	10757,58	95,44	0,0089	11	0,70	11020,78	95,78	0,0087	14	0,70	11028,28	95,78	0,0087
5	0,75	5583,18	88,21	0,0158	8	0,75	13032,50	97,50	0,0075	11	0,75	13190,58	97,68	0,0074	14	0,75	13195,73	97,68	0,0074
5	0,80	6764,03	91,39	0,0135	8	0,80	14334,93	98,11	0,0068	11	0,80	14444,53	98,36	0,0068	14	0,80	14447,25	98,36	0,0068
6	0,30	96,35	20,14	0,2090	9	0,30	286,00	43,98	0,1538	12	0,30	506,35	57,92	0,1144	15	0,30	661,45	63,51	0,0960
6	0,35	218,25	31,50	0,1443	9	0,35	752,95	63,25	0,0840	12	0,35	1202,55	72,63	0,0604	15	0,35	1385,78	74,87	0,0540
6	0,40	526,98	50,17	0,0952	9	0,40	1803,03	77,88	0,0432	12	0,40	2369,10	81,58	0,0344	15	0,40	2496,53	82,10	0,0329
6	0,45	992,63	61,79	0,0622	9	0,45	2916,08	83,48	0,0286	12	0,45	3457,83	84,60	0,0245	15	0,45	3544,13	84,77	0,0239
6	0,50	2309,70	77,88	0,0337	9	0,50	5256,50	88,73	0,0169	12	0,50	5600,15	88,98	0,0159	15	0,50	5626,78	89,07	0,0158
6	0,55	2792,58	80,55	0,0288	9	0,55	5894,75	89,85	0,0152	12	0,55	6186,05	90,19	0,0146	15	0,55	6206,63	90,19	0,0145
6	0,60	4810,75	88,55	0,0184	9	0,60	8028,70	93,20	0,0116	12	0,60	8197,93	93,20	0,0114	15	0,60	8206,65	93,20	0,0114
6	0,65	5915,10	90,02	0,0152	9	0,65	9057,73	94,23	0,0104	12	0,65	9193,65	94,32	0,0103	15	0,65	9200,98	94,32	0,0103
6	0,70	8091,28	93,46	0,0116	9	0,70	10940,20	95,78	0,0088	12	0,70	11026,75	95,78	0,0087	15	0,70	11028,38	95,78	0,0087
6	0,75	10860,25	96,73	0,0089	9	0,75	13145,63	97,68	0,0074	12	0,75	13193,38	97,68	0,0074	15	0,75	13196,05	97,68	0,0074
6	0,80	12494,28	97,59	0,0078	9	0,80	14417,10	98,19	0,0068	12	0,80	14446,48	98,36	0,0068	15	0,80	14447,25	98,36	0,0068

DLP<sub>1</sub>Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / QReferencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-15 Rendimiento de BII-P2P bajo política DLP<sub>2</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	28,50	6,54	0,2295	7	0,30	124,88	24,87	0,1992	10	0,30	310,03	47,42	0,1529	13	0,30	501,13	58,86	0,1175
4	0,35	51,28	8,69	0,1695	7	0,35	320,80	41,82	0,1304	10	0,35	813,05	66,61	0,0819	13	0,35	1175,60	75,04	0,0638
4	0,40	85,83	11,62	0,1354	7	0,40	798,10	61,62	0,0772	10	0,40	1931,33	82,87	0,0429	13	0,40	2413,88	85,54	0,0354
4	0,45	141,23	15,75	0,1115	7	0,45	1538,48	73,06	0,0475	10	0,45	3109,60	87,95	0,0283	13	0,45	3522,95	89,67	0,0255
4	0,50	271,35	21,60	0,0796	7	0,50	3589,53	88,12	0,0246	10	0,50	5504,30	92,94	0,0169	13	0,50	5745,68	93,29	0,0162
4	0,55	313,88	23,58	0,0751	7	0,55	4206,95	89,93	0,0214	10	0,55	6135,88	93,55	0,0152	13	0,55	6331,08	93,80	0,0148
4	0,60	524,50	30,64	0,0584	7	0,60	6664,08	94,84	0,0142	10	0,60	8256,38	96,39	0,0117	13	0,60	8361,38	96,47	0,0115
4	0,65	628,90	33,91	0,0539	7	0,65	7858,05	96,30	0,0123	10	0,65	9298,83	97,07	0,0104	13	0,65	9373,50	97,07	0,0104
4	0,70	885,55	38,55	0,0435	7	0,70	10086,45	97,42	0,0097	10	0,70	11160,45	97,68	0,0088	13	0,70	11199,60	97,76	0,0087
4	0,75	1290,83	45,87	0,0355	7	0,75	12640,05	98,88	0,0078	10	0,75	13306,35	98,88	0,0074	13	0,75	13325,30	98,88	0,0074
4	0,80	1561,23	51,29	0,0329	7	0,80	14064,53	99,14	0,0070	10	0,80	14573,73	99,31	0,0068	13	0,80	14590,03	99,31	0,0068
5	0,30	50,38	11,62	0,2306	8	0,30	179,80	32,70	0,1819	11	0,30	377,85	52,24	0,1382	14	0,30	551,45	61,19	0,1110
5	0,35	105,28	17,38	0,1651	8	0,35	478,10	52,58	0,1100	11	0,35	960,03	70,40	0,0733	14	0,35	1245,73	76,68	0,0616
5	0,40	205,70	26,08	0,1268	8	0,40	1218,50	72,38	0,0594	11	0,40	2159,73	84,68	0,0392	14	0,40	2474,48	85,89	0,0347
5	0,45	377,85	36,66	0,0970	8	0,45	2219,43	82,96	0,0374	11	0,45	3328,45	88,64	0,0266	14	0,45	3560,33	89,76	0,0252
5	0,50	859,60	53,61	0,0624	8	0,50	4623,50	91,57	0,0198	11	0,50	5648,68	93,12	0,0165	14	0,50	5760,45	93,29	0,0162
5	0,55	1027,78	56,20	0,0547	8	0,55	5301,78	92,43	0,0174	11	0,55	6255,98	93,55	0,0150	14	0,55	6340,93	93,80	0,0148
5	0,60	1874,28	70,57	0,0377	8	0,60	7679,90	95,70	0,0125	11	0,60	8323,40	96,39	0,0116	14	0,60	8366,38	96,47	0,0115
5	0,65	2341,08	74,87	0,0320	8	0,65	8820,33	96,64	0,0110	11	0,65	9349,05	97,07	0,0104	14	0,65	9374,83	97,07	0,0104
5	0,70	3470,18	82,53	0,0238	8	0,70	10843,25	97,68	0,0090	11	0,70	11187,73	97,68	0,0087	14	0,70	11200,03	97,76	0,0087
5	0,75	5234,75	89,67	0,0171	8	0,75	13129,18	98,88	0,0075	11	0,75	13320,10	98,88	0,0074	14	0,75	13325,78	98,88	0,0074
5	0,80	6420,58	92,69	0,0144	8	0,80	14439,85	99,23	0,0069	11	0,80	14585,33	99,31	0,0068	14	0,80	14590,18	99,31	0,0068
6	0,30	82,63	17,73	0,2146	9	0,30	242,23	40,45	0,1670	12	0,30	443,05	56,37	0,1272	15	0,30	590,85	62,99	0,1066
6	0,35	193,95	29,35	0,1513	9	0,35	646,83	59,90	0,0926	12	0,35	1081,35	72,38	0,0669	15	0,35	1299,88	77,71	0,0598
6	0,40	440,28	44,58	0,1013	9	0,40	1612,08	79,52	0,0493	12	0,40	2313,18	85,11	0,0368	15	0,40	2511,48	86,06	0,0343
6	0,45	858,88	57,23	0,0666	9	0,45	2754,85	86,66	0,0315	12	0,45	3455,35	89,07	0,0258	15	0,45	3581,93	89,76	0,0251
6	0,50	2114,65	77,11	0,0365	9	0,50	5207,08	92,43	0,0178	12	0,50	5714,53	93,20	0,0163	15	0,50	5766,55	93,29	0,0162
6	0,55	2536,80	80,12	0,0316	9	0,55	5868,83	93,29	0,0159	12	0,55	6309,45	93,80	0,0149	15	0,55	6346,50	93,80	0,0148
6	0,60	4485,75	90,53	0,0202	9	0,60	8089,45	96,13	0,0119	12	0,60	8348,73	96,47	0,0116	15	0,60	8368,73	96,47	0,0115
6	0,65	5547,03	93,55	0,0169	9	0,65	9165,28	96,99	0,0106	12	0,65	9366,55	97,07	0,0104	15	0,65	9375,18	97,07	0,0104
6	0,70	7761,90	96,13	0,0124	9	0,70	11081,03	97,68	0,0088	12	0,70	11196,63	97,76	0,0087	15	0,70	11200,40	97,76	0,0087
6	0,75	10653,20	98,28	0,0092	9	0,75	13267,65	98,88	0,0075	12	0,75	13324,20	98,88	0,0074	15	0,75	13326,13	98,88	0,0074
6	0,80	12325,88	98,45	0,0080	9	0,80	14541,05	99,23	0,0068	12	0,80	14588,90	99,31	0,0068	15	0,80	14590,25	99,31	0,0068

DLP<sub>2</sub>Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / QReferencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-16 Rendimiento de BII-P2P bajo política DLP<sub>3</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	28,05	5,16	0,1841	7	0,30	116,65	22,20	0,1903	10	0,30	289,18	43,55	0,1506	13	0,30	496,40	61,45	0,1238
4	0,35	49,23	7,57	0,1538	7	0,35	303,40	41,22	0,1359	10	0,35	826,03	70,83	0,0857	13	0,35	1204,95	81,15	0,0673
4	0,40	85,18	11,45	0,1344	7	0,40	792,88	63,77	0,0804	10	0,40	1971,68	85,71	0,0435	13	0,40	2397,63	88,47	0,0369
4	0,45	137,85	15,75	0,1142	7	0,45	1535,43	76,68	0,0499	10	0,45	3103,80	90,71	0,0292	13	0,45	3430,13	91,57	0,0267
4	0,50	260,35	21,43	0,0823	7	0,50	3474,80	90,62	0,0261	10	0,50	5347,75	94,58	0,0177	13	0,50	5541,68	94,92	0,0171
4	0,55	303,58	22,55	0,0743	7	0,55	4123,78	92,69	0,0225	10	0,55	5941,25	95,35	0,0160	13	0,55	6089,50	95,61	0,0157
4	0,60	505,30	30,03	0,0594	7	0,60	6526,63	95,87	0,0147	10	0,60	8003,53	96,99	0,0121	13	0,60	8090,08	97,16	0,0120
4	0,65	612,58	32,62	0,0532	7	0,65	7711,80	96,82	0,0126	10	0,65	8985,90	97,50	0,0109	13	0,65	9042,08	97,50	0,0108
4	0,70	867,93	38,64	0,0445	7	0,70	9896,50	98,45	0,0099	10	0,70	10808,50	98,80	0,0091	13	0,70	10839,60	98,80	0,0091
4	0,75	1265,80	45,27	0,0358	7	0,75	12446,10	99,48	0,0080	10	0,75	13041,83	99,48	0,0076	13	0,75	13055,33	99,48	0,0076
4	0,80	1519,65	49,74	0,0327	7	0,80	13917,15	99,57	0,0072	10	0,80	14357,43	99,57	0,0069	13	0,80	14367,93	99,57	0,0069
5	0,30	48,90	10,41	0,2129	8	0,30	164,48	28,83	0,1753	11	0,30	359,90	50,09	0,1392	14	0,30	555,65	65,75	0,1183
5	0,35	98,63	16,52	0,1675	8	0,35	463,33	52,75	0,1139	11	0,35	983,58	74,78	0,0760	14	0,35	1277,15	82,27	0,0644
5	0,40	201,03	25,56	0,1271	8	0,40	1233,85	76,16	0,0617	11	0,40	2182,63	87,35	0,0400	14	0,40	2447,58	88,64	0,0362
5	0,45	364,45	35,54	0,0975	8	0,45	2251,00	87,01	0,0387	11	0,45	3284,50	91,14	0,0277	14	0,45	3457,78	91,57	0,0265
5	0,50	813,53	51,98	0,0639	8	0,50	4537,55	93,55	0,0206	11	0,50	5465,88	94,75	0,0173	14	0,50	5551,05	94,92	0,0171
5	0,55	986,78	55,85	0,0566	8	0,55	5215,15	94,58	0,0181	11	0,55	6032,83	95,52	0,0158	14	0,55	6096,43	95,61	0,0157
5	0,60	1795,75	70,91	0,0395	8	0,60	7498,95	96,82	0,0129	11	0,60	8057,95	97,16	0,0121	14	0,60	8092,75	97,16	0,0120
5	0,65	2269,53	74,87	0,0330	8	0,65	8586,83	97,16	0,0113	11	0,65	9025,48	97,50	0,0108	14	0,65	9044,23	97,50	0,0108
5	0,70	3382,95	83,13	0,0246	8	0,70	10564,10	98,71	0,0093	11	0,70	10829,53	98,80	0,0091	14	0,70	10840,23	98,80	0,0091
5	0,75	5127,68	90,28	0,0176	8	0,75	12888,80	99,48	0,0077	11	0,75	13052,05	99,48	0,0076	14	0,75	13055,50	99,48	0,0076
5	0,80	6275,73	92,60	0,0148	8	0,80	14255,45	99,57	0,0070	11	0,80	14365,03	99,57	0,0069	14	0,80	14367,93	99,57	0,0069
6	0,30	77,38	16,18	0,2091	9	0,30	221,88	36,23	0,1633	12	0,30	431,30	56,20	0,1303	15	0,30	608,75	68,07	0,1118
6	0,35	179,68	29,17	0,1624	9	0,35	646,30	63,34	0,0980	12	0,35	1110,13	78,06	0,0703	15	0,35	1331,73	82,87	0,0622
6	0,40	427,28	45,27	0,1059	9	0,40	1649,63	82,87	0,0502	12	0,40	2316,53	87,87	0,0379	15	0,40	2476,55	88,90	0,0359
6	0,45	832,88	59,04	0,0709	9	0,45	2776,05	89,67	0,0323	12	0,45	3381,38	91,22	0,0270	15	0,45	3470,98	91,74	0,0264
6	0,50	1998,85	78,49	0,0393	9	0,50	5087,80	94,32	0,0185	12	0,50	5518,05	94,84	0,0172	15	0,50	5554,60	94,92	0,0171
6	0,55	2438,80	82,44	0,0338	9	0,55	5725,58	95,09	0,0166	12	0,55	6071,93	95,61	0,0157	15	0,55	6098,93	95,61	0,0157
6	0,60	4346,88	91,74	0,0211	9	0,60	7865,08	96,90	0,0123	12	0,60	8080,08	97,16	0,0120	15	0,60	8094,03	97,16	0,0120
6	0,65	5416,18	93,63	0,0173	9	0,65	8880,35	97,50	0,0110	12	0,65	9037,13	97,50	0,0108	15	0,65	9045,70	97,50	0,0108
6	0,70	7617,18	96,73	0,0127	9	0,70	10751,95	98,80	0,0092	12	0,70	10836,98	98,80	0,0091	15	0,70	10840,63	98,80	0,0091
6	0,75	10523,40	98,71	0,0094	9	0,75	13007,60	99,48	0,0076	12	0,75	13054,73	99,48	0,0076	15	0,75	13055,50	99,48	0,0076
6	0,80	12230,40	98,80	0,0081	9	0,80	14335,58	99,57	0,0069	12	0,80	14367,18	99,57	0,0069	15	0,80	14367,93	99,57	0,0069

**DLP<sub>3</sub>**

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-17 Rendimiento de BII-P2P bajo política DLP<sub>4</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,40	4,65	0,1760	7	0,30	107,45	21,26	0,1978	10	0,30	277,70	44,06	0,1587	13	0,30	486,20	60,50	0,1244
4	0,35	47,20	7,57	0,1604	7	0,35	295,55	40,19	0,1360	10	0,35	828,28	69,97	0,0845	13	0,35	1225,95	81,41	0,0664
4	0,40	84,40	11,27	0,1336	7	0,40	796,13	63,77	0,0801	10	0,40	1975,70	87,09	0,0441	13	0,40	2381,30	89,93	0,0378
4	0,45	137,63	15,83	0,1151	7	0,45	1554,15	78,23	0,0503	10	0,45	3111,08	91,82	0,0295	13	0,45	3406,58	92,86	0,0273
4	0,50	257,18	20,83	0,0810	7	0,50	3476,83	91,05	0,0262	10	0,50	5275,55	94,49	0,0179	13	0,50	5436,58	94,92	0,0175
4	0,55	298,30	22,72	0,0762	7	0,55	4102,08	93,29	0,0227	10	0,55	5855,43	95,52	0,0163	13	0,55	5992,33	95,78	0,0160
4	0,60	503,15	30,38	0,0604	7	0,60	6514,73	96,30	0,0148	10	0,60	7871,08	97,16	0,0123	13	0,60	7933,13	97,16	0,0122
4	0,65	612,63	32,79	0,0535	7	0,65	7684,70	96,90	0,0126	10	0,65	8845,43	97,33	0,0110	13	0,65	8892,65	97,33	0,0109
4	0,70	873,68	40,19	0,0460	7	0,70	9833,43	98,11	0,0100	10	0,70	10677,03	98,36	0,0092	13	0,70	10702,70	98,36	0,0092
4	0,75	1274,00	47,07	0,0369	7	0,75	12369,33	99,23	0,0080	10	0,75	12921,50	99,31	0,0077	13	0,75	12936,95	99,31	0,0077
4	0,80	1527,83	50,43	0,0330	7	0,80	13845,25	99,57	0,0072	10	0,80	14250,10	99,66	0,0070	13	0,80	14258,50	99,66	0,0070
5	0,30	44,88	9,04	0,2014	8	0,30	154,23	29,26	0,1897	11	0,30	349,10	50,26	0,1440	14	0,30	546,20	64,72	0,1185
5	0,35	94,05	16,01	0,1702	8	0,35	455,08	51,81	0,1138	11	0,35	994,30	75,22	0,0756	14	0,35	1298,70	83,13	0,0640
5	0,40	200,28	25,82	0,1289	8	0,40	1241,70	75,65	0,0609	11	0,40	2181,65	89,07	0,0408	14	0,40	2426,05	90,28	0,0372
5	0,45	365,55	35,89	0,0982	8	0,45	2270,25	87,44	0,0385	11	0,45	3275,65	92,34	0,0282	14	0,45	3427,65	92,86	0,0271
5	0,50	799,48	51,38	0,0643	8	0,50	4522,30	93,72	0,0207	11	0,50	5373,38	94,75	0,0176	14	0,50	5444,58	94,92	0,0174
5	0,55	963,23	55,68	0,0578	8	0,55	5158,83	95,09	0,0184	11	0,55	5941,75	95,70	0,0161	14	0,55	5998,60	95,78	0,0160
5	0,60	1784,73	70,05	0,0393	8	0,60	7430,68	96,90	0,0130	11	0,60	7912,93	97,16	0,0123	14	0,60	7935,28	97,16	0,0122
5	0,65	2272,80	74,10	0,0326	8	0,65	8493,98	97,16	0,0114	11	0,65	8877,30	97,33	0,0110	14	0,65	8893,38	97,33	0,0109
5	0,70	3397,03	82,96	0,0244	8	0,70	10456,48	98,19	0,0094	11	0,70	10695,03	98,36	0,0092	14	0,70	10703,38	98,36	0,0092
5	0,75	5152,03	89,76	0,0174	8	0,75	12785,40	99,31	0,0078	11	0,75	12933,40	99,31	0,0077	14	0,75	12937,18	99,31	0,0077
5	0,80	6294,23	93,46	0,0148	8	0,80	14165,90	99,66	0,0070	11	0,80	14256,50	99,66	0,0070	14	0,80	14258,50	99,66	0,0070
6	0,30	71,20	15,32	0,2151	9	0,30	211,90	36,83	0,1738	12	0,30	420,25	55,85	0,1329	15	0,30	600,88	67,38	0,1121
6	0,35	173,95	27,97	0,1608	9	0,35	640,55	62,39	0,0974	12	0,35	1125,90	79,52	0,0706	15	0,35	1347,93	83,73	0,0621
6	0,40	427,48	44,66	0,1045	9	0,40	1660,08	82,87	0,0499	12	0,40	2306,03	89,59	0,0388	15	0,40	2450,78	90,53	0,0369
6	0,45	840,10	60,76	0,0723	9	0,45	2796,70	90,71	0,0324	12	0,45	3364,23	92,51	0,0275	15	0,45	3438,15	92,86	0,0270
6	0,50	1985,43	78,57	0,0396	9	0,50	5041,43	94,23	0,0187	12	0,50	5416,58	94,84	0,0175	15	0,50	5447,93	94,92	0,0174
6	0,55	2409,90	82,62	0,0343	9	0,55	5643,88	95,44	0,0169	12	0,55	5976,08	95,78	0,0160	15	0,55	6001,55	95,78	0,0160
6	0,60	4350,60	92,25	0,0212	9	0,60	7756,48	97,07	0,0125	12	0,60	7928,00	97,16	0,0123	15	0,60	7935,70	97,16	0,0122
6	0,65	5441,15	93,63	0,0172	9	0,65	8758,28	97,33	0,0111	12	0,65	8888,33	97,33	0,0110	15	0,65	8893,55	97,33	0,0109
6	0,70	7619,35	96,73	0,0127	9	0,70	10626,68	98,36	0,0093	12	0,70	10701,03	98,36	0,0092	15	0,70	10703,55	98,36	0,0092
6	0,75	10516,13	98,62	0,0094	9	0,75	12894,08	99,31	0,0077	12	0,75	12935,88	99,31	0,0077	15	0,75	12937,18	99,31	0,0077
6	0,80	12212,28	99,31	0,0081	9	0,80	14233,33	99,66	0,0070	12	0,80	14258,23	99,66	0,0070	15	0,80	14258,50	99,66	0,0070

**DLP<sub>4</sub>**

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-18 Rendimiento de BII-P2P bajo política DLP<sub>5</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,43	4,99	0,1889	7	0,30	110,98	21,77	0,1962	10	0,30	289,80	45,09	0,1556	13	0,30	502,43	61,27	0,1220
4	0,35	47,20	7,49	0,1586	7	0,35	296,28	40,36	0,1362	10	0,35	842,33	69,79	0,0829	13	0,35	1240,33	82,27	0,0663
4	0,40	82,38	10,76	0,1306	7	0,40	781,73	63,17	0,0808	10	0,40	1975,95	86,83	0,0439	13	0,40	2379,85	89,85	0,0378
4	0,45	136,45	15,32	0,1123	7	0,45	1548,35	78,23	0,0505	10	0,45	3106,23	91,82	0,0296	13	0,45	3390,90	92,60	0,0273
4	0,50	259,68	20,91	0,0805	7	0,50	3510,13	91,74	0,0261	10	0,50	5266,68	94,66	0,0180	13	0,50	5419,33	94,84	0,0175
4	0,55	301,88	22,20	0,0736	7	0,55	4134,78	93,37	0,0226	10	0,55	5835,23	95,35	0,0163	13	0,55	5968,33	95,44	0,0160
4	0,60	500,13	30,03	0,0601	7	0,60	6501,15	95,96	0,0148	10	0,60	7868,88	97,07	0,0123	13	0,60	7921,33	97,16	0,0123
4	0,65	608,88	32,62	0,0536	7	0,65	7667,80	96,64	0,0126	10	0,65	8823,93	97,42	0,0110	13	0,65	8865,75	97,42	0,0110
4	0,70	872,15	40,02	0,0459	7	0,70	9825,70	97,76	0,0099	10	0,70	10656,40	98,02	0,0092	13	0,70	10680,80	98,02	0,0092
4	0,75	1262,75	46,73	0,0370	7	0,75	12366,08	99,14	0,0080	10	0,75	12913,40	99,23	0,0077	13	0,75	12927,13	99,23	0,0077
4	0,80	1517,85	50,52	0,0333	7	0,80	13839,98	99,48	0,0072	10	0,80	14241,43	99,48	0,0070	13	0,80	14247,65	99,48	0,0070
5	0,30	45,60	9,38	0,2057	8	0,30	160,90	29,52	0,1835	11	0,30	362,28	51,29	0,1416	14	0,30	563,68	65,66	0,1165
5	0,35	94,20	16,35	0,1736	8	0,35	460,85	52,32	0,1135	11	0,35	1008,45	75,56	0,0749	14	0,35	1309,28	83,22	0,0636
5	0,40	194,10	24,10	0,1241	8	0,40	1237,05	76,33	0,0617	11	0,40	2182,20	88,81	0,0407	14	0,40	2420,33	90,02	0,0372
5	0,45	362,63	34,51	0,0952	8	0,45	2269,93	87,44	0,0385	11	0,45	3266,08	92,43	0,0283	14	0,45	3414,20	92,60	0,0271
5	0,50	810,83	52,07	0,0642	8	0,50	4543,90	94,15	0,0207	11	0,50	5364,30	94,75	0,0177	14	0,50	5425,85	94,84	0,0175
5	0,55	975,58	55,59	0,0570	8	0,55	5171,35	95,01	0,0184	11	0,55	5921,40	95,44	0,0161	14	0,55	5972,83	95,44	0,0160
5	0,60	1774,88	70,74	0,0399	8	0,60	7426,93	96,82	0,0130	11	0,60	7906,63	97,07	0,0123	14	0,60	7922,35	97,16	0,0123
5	0,65	2260,45	75,04	0,0332	8	0,65	8479,38	97,16	0,0115	11	0,65	8852,35	97,42	0,0110	14	0,65	8866,83	97,42	0,0110
5	0,70	3389,43	82,96	0,0245	8	0,70	10433,78	97,85	0,0094	11	0,70	10673,30	98,02	0,0092	14	0,70	10681,33	98,02	0,0092
5	0,75	5118,15	89,76	0,0175	8	0,75	12784,80	99,23	0,0078	11	0,75	12923,00	99,23	0,0077	14	0,75	12927,20	99,23	0,0077
5	0,80	6272,90	92,69	0,0148	8	0,80	14152,13	99,48	0,0070	11	0,80	14246,60	99,48	0,0070	14	0,80	14247,65	99,48	0,0070
6	0,30	73,00	15,32	0,2098	9	0,30	221,10	37,61	0,1701	12	0,30	434,93	55,94	0,1286	15	0,30	616,83	67,64	0,1097
6	0,35	173,28	28,14	0,1624	9	0,35	652,20	62,22	0,0954	12	0,35	1140,73	79,52	0,0697	15	0,35	1357,25	83,82	0,0618
6	0,40	413,75	43,72	0,1057	9	0,40	1662,78	83,13	0,0500	12	0,40	2306,18	89,24	0,0387	15	0,40	2444,50	90,19	0,0369
6	0,45	835,78	60,33	0,0722	9	0,45	2799,78	90,36	0,0323	12	0,45	3348,60	92,51	0,0276	15	0,45	3425,88	92,60	0,0270
6	0,50	2014,90	79,69	0,0396	9	0,50	5041,43	94,49	0,0187	12	0,50	5402,70	94,84	0,0176	15	0,50	5428,98	94,84	0,0175
6	0,55	2434,38	82,53	0,0339	9	0,55	5633,98	95,27	0,0169	12	0,55	5954,05	95,44	0,0160	15	0,55	5974,80	95,44	0,0160
6	0,60	4334,15	92,17	0,0213	9	0,60	7756,75	97,07	0,0125	12	0,60	7917,80	97,16	0,0123	15	0,60	7922,55	97,16	0,0123
6	0,65	5424,83	93,55	0,0172	9	0,65	8736,75	97,42	0,0112	12	0,65	8862,03	97,42	0,0110	15	0,65	8867,35	97,42	0,0110
6	0,70	7617,33	96,64	0,0127	9	0,70	10604,78	98,02	0,0092	12	0,70	10678,45	98,02	0,0092	15	0,70	10681,33	98,02	0,0092
6	0,75	10487,43	98,62	0,0094	9	0,75	12888,00	99,23	0,0077	12	0,75	12926,03	99,23	0,0077	15	0,75	12927,20	99,23	0,0077
6	0,80	12207,28	99,05	0,0081	9	0,80	14221,55	99,48	0,0070	12	0,80	14247,53	99,48	0,0070	15	0,80	14247,65	99,48	0,0070

**DLP<sub>5</sub>**

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166



**Tabla A-19 Rendimiento de BII-P2P bajo política DLP<sub>6</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,53	4,91	0,1849	7	0,30	110,13	22,03	0,2001	10	0,30	288,03	45,44	0,1578	13	0,30	497,68	61,10	0,1228
4	0,35	46,98	7,40	0,1576	7	0,35	293,98	39,85	0,1355	10	0,35	839,60	70,48	0,0839	13	0,35	1236,08	81,58	0,0660
4	0,40	82,58	10,84	0,1313	7	0,40	782,15	63,34	0,0810	10	0,40	1977,98	87,18	0,0441	13	0,40	2378,05	90,28	0,0380
4	0,45	137,18	15,40	0,1123	7	0,45	1563,65	78,23	0,0500	10	0,45	3112,28	91,82	0,0295	13	0,45	3390,90	92,51	0,0273
4	0,50	260,75	21,08	0,0809	7	0,50	3509,35	91,74	0,0261	10	0,50	5262,05	94,66	0,0180	13	0,50	5415,58	94,84	0,0175
4	0,55	302,15	22,46	0,0743	7	0,55	4138,15	93,46	0,0226	10	0,55	5832,20	95,27	0,0163	13	0,55	5959,45	95,35	0,0160
4	0,60	501,45	30,38	0,0606	7	0,60	6517,58	95,70	0,0147	10	0,60	7853,73	96,90	0,0123	13	0,60	7905,40	96,90	0,0123
4	0,65	610,88	32,79	0,0537	7	0,65	7681,08	96,56	0,0126	10	0,65	8813,05	97,25	0,0110	13	0,65	8853,68	97,25	0,0110
4	0,70	875,25	40,36	0,0461	7	0,70	9821,13	97,93	0,0100	10	0,70	10644,00	98,19	0,0092	13	0,70	10671,00	98,19	0,0092
4	0,75	1265,30	46,73	0,0369	7	0,75	12364,28	99,05	0,0080	10	0,75	12904,75	99,23	0,0077	13	0,75	12920,33	99,23	0,0077
4	0,80	1520,68	50,26	0,0330	7	0,80	13841,63	99,40	0,0072	10	0,80	14233,83	99,48	0,0070	13	0,80	14241,85	99,48	0,0070
5	0,30	45,63	9,55	0,2094	8	0,30	159,65	30,12	0,1887	11	0,30	360,25	51,81	0,1438	14	0,30	558,75	65,23	0,1167
5	0,35	93,50	16,27	0,1740	8	0,35	457,50	52,58	0,1149	11	0,35	1007,05	75,73	0,0752	14	0,35	1305,40	82,96	0,0636
5	0,40	194,45	24,70	0,1270	8	0,40	1238,33	76,08	0,0614	11	0,40	2183,83	88,98	0,0407	14	0,40	2418,55	90,45	0,0374
5	0,45	366,03	35,20	0,0962	8	0,45	2285,53	87,44	0,0383	11	0,45	3266,30	92,43	0,0283	14	0,45	3413,50	92,51	0,0271
5	0,50	813,55	52,58	0,0646	8	0,50	4543,55	94,23	0,0207	11	0,50	5358,45	94,75	0,0177	14	0,50	5422,98	94,84	0,0175
5	0,55	977,38	56,37	0,0577	8	0,55	5173,20	95,01	0,0184	11	0,55	5914,80	95,35	0,0161	14	0,55	5963,95	95,35	0,0160
5	0,60	1779,05	70,40	0,0396	8	0,60	7423,98	96,64	0,0130	11	0,60	7890,48	96,90	0,0123	14	0,60	7906,13	96,90	0,0123
5	0,65	2267,05	74,35	0,0328	8	0,65	8478,20	97,07	0,0114	11	0,65	8841,10	97,25	0,0110	14	0,65	8854,83	97,25	0,0110
5	0,70	3405,10	83,30	0,0245	8	0,70	10424,35	98,11	0,0094	11	0,70	10662,38	98,19	0,0092	14	0,70	10671,25	98,19	0,0092
5	0,75	5132,45	89,85	0,0175	8	0,75	12772,20	99,23	0,0078	11	0,75	12915,40	99,23	0,0077	14	0,75	12920,53	99,23	0,0077
5	0,80	6281,68	92,69	0,0148	8	0,80	14147,80	99,48	0,0070	11	0,80	14239,90	99,48	0,0070	14	0,80	14241,95	99,48	0,0070
6	0,30	72,65	15,40	0,2120	9	0,30	219,78	38,12	0,1735	12	0,30	430,98	56,20	0,1304	15	0,30	611,73	67,73	0,1107
6	0,35	171,90	27,80	0,1617	9	0,35	648,95	62,74	0,0967	12	0,35	1137,93	79,26	0,0697	15	0,35	1352,63	83,73	0,0619
6	0,40	415,30	43,89	0,1057	9	0,40	1664,25	83,48	0,0502	12	0,40	2306,40	89,50	0,0388	15	0,40	2442,08	90,53	0,0371
6	0,45	844,63	60,84	0,0720	9	0,45	2811,35	90,71	0,0323	12	0,45	3347,33	92,51	0,0276	15	0,45	3424,28	92,51	0,0270
6	0,50	2020,35	79,78	0,0395	9	0,50	5040,68	94,41	0,0187	12	0,50	5398,33	94,84	0,0176	15	0,50	5426,53	94,84	0,0175
6	0,55	2439,18	82,87	0,0340	9	0,55	5634,50	95,18	0,0169	12	0,55	5946,65	95,35	0,0160	15	0,55	5965,95	95,35	0,0160
6	0,60	4349,83	91,91	0,0211	9	0,60	7742,25	96,90	0,0125	12	0,60	7902,63	96,90	0,0123	15	0,60	7906,60	96,90	0,0123
6	0,65	5438,83	93,20	0,0171	9	0,65	8729,30	97,25	0,0111	12	0,65	8850,58	97,25	0,0110	15	0,65	8855,08	97,25	0,0110
6	0,70	7635,93	96,64	0,0127	9	0,70	10592,48	98,19	0,0093	12	0,70	10668,43	98,19	0,0092	15	0,70	10671,38	98,19	0,0092
6	0,75	10498,30	98,62	0,0094	9	0,75	12877,63	99,23	0,0077	12	0,75	12919,03	99,23	0,0077	15	0,75	12920,53	99,23	0,0077
6	0,80	12216,13	99,14	0,0081	9	0,80	14215,58	99,48	0,0070	12	0,80	14241,40	99,48	0,0070	15	0,80	14241,95	99,48	0,0070

DLP<sub>6</sub>Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / QReferencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-20 Rendimiento de BII-P2P bajo política DLP<sub>7</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,80	4,82	0,1798	7	0,30	111,63	21,94	0,1966	10	0,30	290,05	45,70	0,1575	13	0,30	499,48	61,45	0,1230
4	0,35	47,10	7,40	0,1571	7	0,35	294,08	39,76	0,1352	10	0,35	838,53	70,40	0,0840	13	0,35	1233,05	81,67	0,0662
4	0,40	82,50	10,84	0,1314	7	0,40	781,40	63,34	0,0811	10	0,40	1974,85	86,83	0,0440	13	0,40	2375,85	90,36	0,0380
4	0,45	136,73	15,40	0,1127	7	0,45	1559,28	78,14	0,0501	10	0,45	3109,53	91,91	0,0296	13	0,45	3390,25	92,51	0,0273
4	0,50	260,35	20,83	0,0800	7	0,50	3506,18	91,74	0,0262	10	0,50	5260,98	94,58	0,0180	13	0,50	5416,33	94,75	0,0175
4	0,55	301,85	22,12	0,0733	7	0,55	4134,03	93,63	0,0226	10	0,55	5832,20	95,27	0,0163	13	0,55	5958,13	95,35	0,0160
4	0,60	502,00	30,29	0,0603	7	0,60	6519,58	95,61	0,0147	10	0,60	7854,03	96,82	0,0123	13	0,60	7905,95	96,82	0,0122
4	0,65	611,58	32,87	0,0538	7	0,65	7678,65	96,47	0,0126	10	0,65	8812,43	97,16	0,0110	13	0,65	8852,75	97,16	0,0110
4	0,70	875,20	40,36	0,0461	7	0,70	9824,25	97,93	0,0100	10	0,70	10644,78	98,19	0,0092	13	0,70	10671,95	98,19	0,0092
4	0,75	1263,73	46,82	0,0370	7	0,75	12363,25	98,97	0,0080	10	0,75	12905,93	99,05	0,0077	13	0,75	12920,78	99,05	0,0077
4	0,80	1518,93	50,34	0,0331	7	0,80	13842,05	99,40	0,0072	10	0,80	14233,73	99,48	0,0070	13	0,80	14240,80	99,48	0,0070
5	0,30	46,33	9,55	0,2062	8	0,30	161,33	30,03	0,1862	11	0,30	362,08	51,89	0,1433	14	0,30	559,78	65,40	0,1168
5	0,35	93,90	16,18	0,1723	8	0,35	458,25	52,24	0,1140	11	0,35	1005,35	75,99	0,0756	14	0,35	1301,93	82,96	0,0637
5	0,40	194,40	24,78	0,1275	8	0,40	1236,25	75,82	0,0613	11	0,40	2180,73	88,90	0,0408	14	0,40	2416,95	90,45	0,0374
5	0,45	364,25	35,03	0,0962	8	0,45	2280,05	87,44	0,0383	11	0,45	3265,13	92,43	0,0283	14	0,45	3412,93	92,51	0,0271
5	0,50	811,88	52,67	0,0649	8	0,50	4540,93	94,15	0,0207	11	0,50	5358,33	94,66	0,0177	14	0,50	5424,15	94,75	0,0175
5	0,55	975,00	56,37	0,0578	8	0,55	5174,18	95,01	0,0184	11	0,55	5913,68	95,35	0,0161	14	0,55	5962,85	95,35	0,0160
5	0,60	1780,28	70,65	0,0397	8	0,60	7425,30	96,56	0,0130	11	0,60	7891,53	96,82	0,0123	14	0,60	7906,80	96,82	0,0122
5	0,65	2267,63	74,70	0,0329	8	0,65	8478,40	96,99	0,0114	11	0,65	8840,03	97,16	0,0110	14	0,65	8853,43	97,16	0,0110
5	0,70	3404,75	83,48	0,0245	8	0,70	10430,95	98,11	0,0094	11	0,70	10662,78	98,19	0,0092	14	0,70	10672,45	98,19	0,0092
5	0,75	5126,28	89,85	0,0175	8	0,75	12772,95	99,05	0,0078	11	0,75	12915,65	99,05	0,0077	14	0,75	12921,03	99,05	0,0077
5	0,80	6276,28	92,77	0,0148	8	0,80	14146,93	99,48	0,0070	11	0,80	14239,40	99,48	0,0070	14	0,80	14240,80	99,48	0,0070
6	0,30	73,88	15,32	0,2074	9	0,30	221,75	38,30	0,1727	12	0,30	432,85	56,54	0,1306	15	0,30	611,90	67,38	0,1101
6	0,35	172,55	27,71	0,1606	9	0,35	648,70	62,56	0,0964	12	0,35	1136,05	79,26	0,0698	15	0,35	1349,93	83,91	0,0622
6	0,40	415,38	43,55	0,1048	9	0,40	1660,83	83,30	0,0502	12	0,40	2302,90	89,41	0,0388	15	0,40	2440,88	90,53	0,0371
6	0,45	841,15	60,50	0,0719	9	0,45	2806,58	90,71	0,0323	12	0,45	3346,10	92,51	0,0276	15	0,45	3423,35	92,60	0,0270
6	0,50	2016,40	80,03	0,0397	9	0,50	5039,25	94,32	0,0187	12	0,50	5398,13	94,75	0,0176	15	0,50	5428,20	94,75	0,0175
6	0,55	2433,05	83,22	0,0342	9	0,55	5633,40	95,18	0,0169	12	0,55	5944,58	95,35	0,0160	15	0,55	5964,75	95,35	0,0160
6	0,60	4353,03	91,82	0,0211	9	0,60	7744,15	96,82	0,0125	12	0,60	7902,78	96,82	0,0123	15	0,60	7907,00	96,82	0,0122
6	0,65	5441,05	93,12	0,0171	9	0,65	8730,75	97,16	0,0111	12	0,65	8849,28	97,16	0,0110	15	0,65	8853,60	97,16	0,0110
6	0,70	7638,93	96,64	0,0127	9	0,70	10595,40	98,19	0,0093	12	0,70	10669,35	98,19	0,0092	15	0,70	10672,58	98,19	0,0092
6	0,75	10490,68	98,62	0,0094	9	0,75	12878,00	99,05	0,0077	12	0,75	12919,40	99,05	0,0077	15	0,75	12921,03	99,05	0,0077
6	0,80	12211,50	99,14	0,0081	9	0,80	14215,85	99,48	0,0070	12	0,80	14240,68	99,48	0,0070	15	0,80	14240,80	99,48	0,0070

DLP<sub>7</sub>Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / QReferencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-21 Rendimiento de BII-P2P bajo política DLP<sub>8</sub>. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,80	4,82	0,1798	7	0,30	111,73	22,03	0,1972	10	0,30	289,88	45,61	0,1573	13	0,30	497,18	61,53	0,1238
4	0,35	47,10	7,40	0,1571	7	0,35	293,68	39,85	0,1357	10	0,35	836,65	70,40	0,0841	13	0,35	1232,45	81,58	0,0662
4	0,40	82,40	10,84	0,1316	7	0,40	780,15	63,08	0,0809	10	0,40	1973,78	86,75	0,0439	13	0,40	2375,90	90,36	0,0380
4	0,45	136,88	15,40	0,1125	7	0,45	1559,80	77,97	0,0500	10	0,45	3110,35	91,91	0,0295	13	0,45	3390,20	92,51	0,0273
4	0,50	260,18	20,91	0,0804	7	0,50	3504,03	91,91	0,0262	10	0,50	5262,10	94,58	0,0180	13	0,50	5415,43	94,75	0,0175
4	0,55	301,83	22,20	0,0736	7	0,55	4139,23	93,72	0,0226	10	0,55	5832,25	95,27	0,0163	13	0,55	5958,25	95,35	0,0160
4	0,60	501,83	30,29	0,0604	7	0,60	6520,58	95,61	0,0147	10	0,60	7856,30	96,82	0,0123	13	0,60	7907,75	96,82	0,0122
4	0,65	611,58	32,87	0,0538	7	0,65	7682,10	96,47	0,0126	10	0,65	8814,98	97,16	0,0110	13	0,65	8854,78	97,16	0,0110
4	0,70	874,88	40,36	0,0461	7	0,70	9825,38	97,93	0,0100	10	0,70	10647,45	98,19	0,0092	13	0,70	10674,53	98,19	0,0092
4	0,75	1264,58	46,82	0,0370	7	0,75	12364,55	98,97	0,0080	10	0,75	12906,50	99,05	0,0077	13	0,75	12921,58	99,05	0,0077
4	0,80	1519,68	50,34	0,0331	7	0,80	13841,85	99,40	0,0072	10	0,80	14232,08	99,48	0,0070	13	0,80	14239,15	99,48	0,0070
5	0,30	46,35	9,55	0,2061	8	0,30	161,55	30,21	0,1870	11	0,30	361,08	51,89	0,1437	14	0,30	557,90	65,49	0,1174
5	0,35	93,83	16,09	0,1715	8	0,35	457,30	52,32	0,1144	11	0,35	1003,05	75,99	0,0758	14	0,35	1302,23	82,96	0,0637
5	0,40	194,13	24,70	0,1272	8	0,40	1233,73	75,90	0,0615	11	0,40	2180,83	88,81	0,0407	14	0,40	2416,53	90,45	0,0374
5	0,45	364,90	34,94	0,0958	8	0,45	2279,98	87,52	0,0384	11	0,45	3265,25	92,43	0,0283	14	0,45	3413,20	92,51	0,0271
5	0,50	811,60	52,67	0,0649	8	0,50	4540,18	94,15	0,0207	11	0,50	5358,85	94,66	0,0177	14	0,50	5422,85	94,75	0,0175
5	0,55	976,13	56,20	0,0576	8	0,55	5175,53	95,01	0,0184	11	0,55	5913,90	95,35	0,0161	14	0,55	5962,78	95,35	0,0160
5	0,60	1780,78	70,57	0,0396	8	0,60	7425,65	96,56	0,0130	11	0,60	7893,60	96,82	0,0123	14	0,60	7908,45	96,82	0,0122
5	0,65	2269,20	74,61	0,0329	8	0,65	8479,25	96,99	0,0114	11	0,65	8842,58	97,16	0,0110	14	0,65	8855,75	97,16	0,0110
5	0,70	3403,50	83,39	0,0245	8	0,70	10430,33	98,11	0,0094	11	0,70	10665,58	98,19	0,0092	14	0,70	10675,03	98,19	0,0092
5	0,75	5129,60	89,85	0,0175	8	0,75	12773,78	99,05	0,0078	11	0,75	12916,30	99,05	0,0077	14	0,75	12921,83	99,05	0,0077
5	0,80	6277,40	92,77	0,0148	8	0,80	14146,20	99,48	0,0070	11	0,80	14237,75	99,48	0,0070	14	0,80	14239,15	99,48	0,0070
6	0,30	73,88	15,32	0,2074	9	0,30	221,90	38,47	0,1734	12	0,30	430,95	56,45	0,1310	15	0,30	610,05	67,56	0,1107
6	0,35	172,13	27,54	0,1600	9	0,35	647,00	62,48	0,0966	12	0,35	1134,83	79,35	0,0699	15	0,35	1350,88	83,91	0,0621
6	0,40	414,43	43,46	0,1049	9	0,40	1659,38	83,22	0,0502	12	0,40	2303,10	89,41	0,0388	15	0,40	2440,70	90,53	0,0371
6	0,45	842,13	60,41	0,0717	9	0,45	2807,50	90,71	0,0323	12	0,45	3346,53	92,51	0,0276	15	0,45	3423,95	92,60	0,0270
6	0,50	2015,23	80,03	0,0397	9	0,50	5039,80	94,32	0,0187	12	0,50	5398,28	94,75	0,0176	15	0,50	5426,95	94,75	0,0175
6	0,55	2438,08	83,22	0,0341	9	0,55	5633,20	95,18	0,0169	12	0,55	5945,53	95,35	0,0160	15	0,55	5965,03	95,35	0,0160
6	0,60	4352,20	91,82	0,0211	9	0,60	7745,65	96,82	0,0125	12	0,60	7905,13	96,82	0,0122	15	0,60	7908,65	96,82	0,0122
6	0,65	5442,73	93,12	0,0171	9	0,65	8731,75	97,16	0,0111	12	0,65	8851,55	97,16	0,0110	15	0,65	8856,00	97,16	0,0110
6	0,70	7638,70	96,64	0,0127	9	0,70	10596,85	98,19	0,0093	12	0,70	10671,95	98,19	0,0092	15	0,70	10675,15	98,19	0,0092
6	0,75	10493,30	98,62	0,0094	9	0,75	12878,25	99,05	0,0077	12	0,75	12920,05	99,05	0,0077	15	0,75	12921,83	99,05	0,0077
6	0,80	12213,18	99,14	0,0081	9	0,80	14214,30	99,48	0,0070	12	0,80	14239,03	99,48	0,0070	15	0,80	14239,15	99,48	0,0070

**DLP<sub>8</sub>** Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q  
Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-22 Rendimiento de BII-P2P bajo política Simple. Las celdas sombreadas corresponden a valores que superan la marca registrada por el algoritmo BFS de *Gnutella* para dicha columna.**

TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E	TTL	$\gamma$	Q	H	E
4	0,30	26,80	4,82	0,1798	7	0,30	111,78	22,03	0,1971	10	0,30	289,90	45,61	0,1573	13	0,30	497,10	61,45	0,1236
4	0,35	47,10	7,31	0,1553	7	0,35	294,08	39,76	0,1352	10	0,35	836,45	70,40	0,0842	13	0,35	1232,13	81,58	0,0662
4	0,40	82,40	10,76	0,1305	7	0,40	780,33	63,08	0,0808	10	0,40	1974,00	86,75	0,0439	13	0,40	2376,20	90,36	0,0380
4	0,45	136,88	15,40	0,1125	7	0,45	1559,40	77,97	0,0500	10	0,45	3109,75	91,91	0,0296	13	0,45	3388,85	92,51	0,0273
4	0,50	260,18	20,91	0,0804	7	0,50	3503,25	91,91	0,0262	10	0,50	5261,28	94,58	0,0180	13	0,50	5415,63	94,75	0,0175
4	0,55	301,78	22,20	0,0736	7	0,55	4133,33	93,72	0,0227	10	0,55	5831,45	95,27	0,0163	13	0,55	5957,98	95,35	0,0160
4	0,60	501,80	30,29	0,0604	7	0,60	6519,25	95,61	0,0147	10	0,60	7856,55	96,82	0,0123	13	0,60	7908,55	96,82	0,0122
4	0,65	611,58	32,87	0,0538	7	0,65	7681,10	96,47	0,0126	10	0,65	8814,85	97,16	0,0110	13	0,65	8854,88	97,16	0,0110
4	0,70	875,05	40,36	0,0461	7	0,70	9825,48	97,93	0,0100	10	0,70	10647,58	98,19	0,0092	13	0,70	10674,73	98,19	0,0092
4	0,75	1264,45	46,82	0,0370	7	0,75	12364,43	98,97	0,0080	10	0,75	12906,60	99,05	0,0077	13	0,75	12921,80	99,05	0,0077
4	0,80	1519,75	50,34	0,0331	7	0,80	13842,03	99,40	0,0072	10	0,80	14233,78	99,48	0,0070	13	0,80	14241,08	99,48	0,0070
5	0,30	46,35	9,55	0,2061	8	0,30	161,60	30,21	0,1869	11	0,30	360,98	51,81	0,1435	14	0,30	557,73	65,40	0,1173
5	0,35	93,80	16,27	0,1734	8	0,35	457,70	52,15	0,1139	11	0,35	1002,73	75,99	0,0758	14	0,35	1301,70	82,96	0,0637
5	0,40	194,08	24,78	0,1277	8	0,40	1233,93	75,90	0,0615	11	0,40	2181,20	88,81	0,0407	14	0,40	2416,80	90,45	0,0374
5	0,45	364,93	34,94	0,0957	8	0,45	2280,30	87,52	0,0384	11	0,45	3264,85	92,43	0,0283	14	0,45	3411,98	92,51	0,0271
5	0,50	811,58	52,67	0,0649	8	0,50	4539,33	94,15	0,0207	11	0,50	5358,45	94,66	0,0177	14	0,50	5423,05	94,75	0,0175
5	0,55	975,08	56,20	0,0576	8	0,55	5173,93	95,01	0,0184	11	0,55	5913,28	95,35	0,0161	14	0,55	5962,50	95,35	0,0160
5	0,60	1780,25	70,40	0,0395	8	0,60	7425,35	96,56	0,0130	11	0,60	7894,13	96,82	0,0123	14	0,60	7909,25	96,82	0,0122
5	0,65	2269,00	74,53	0,0328	8	0,65	8480,03	96,99	0,0114	11	0,65	8842,33	97,16	0,0110	14	0,65	8855,98	97,16	0,0110
5	0,70	3404,18	83,39	0,0245	8	0,70	10430,40	98,11	0,0094	11	0,70	10665,43	98,19	0,0092	14	0,70	10675,23	98,19	0,0092
5	0,75	5130,13	89,85	0,0175	8	0,75	12773,60	99,05	0,0078	11	0,75	12916,53	99,05	0,0077	14	0,75	12922,05	99,05	0,0077
5	0,80	6279,18	92,77	0,0148	8	0,80	14147,33	99,48	0,0070	11	0,80	14239,68	99,48	0,0070	14	0,80	14241,08	99,48	0,0070
6	0,30	73,90	15,32	0,2073	9	0,30	221,95	38,47	0,1733	12	0,30	430,90	56,37	0,1308	15	0,30	609,85	67,56	0,1108
6	0,35	172,43	27,71	0,1607	9	0,35	646,93	62,39	0,0964	12	0,35	1134,55	79,35	0,0699	15	0,35	1350,18	83,91	0,0621
6	0,40	414,40	43,46	0,1049	9	0,40	1659,83	83,22	0,0501	12	0,40	2303,80	89,41	0,0388	15	0,40	2440,95	90,53	0,0371
6	0,45	841,98	60,41	0,0718	9	0,45	2807,28	90,71	0,0323	12	0,45	3345,58	92,51	0,0277	15	0,45	3422,65	92,60	0,0271
6	0,50	2014,93	80,03	0,0397	9	0,50	5038,80	94,32	0,0187	12	0,50	5398,30	94,75	0,0176	15	0,50	5427,15	94,75	0,0175
6	0,55	2434,33	83,13	0,0342	9	0,55	5632,35	95,18	0,0169	12	0,55	5945,13	95,35	0,0160	15	0,55	5964,75	95,35	0,0160
6	0,60	4351,33	91,82	0,0211	9	0,60	7745,53	96,82	0,0125	12	0,60	7905,60	96,82	0,0122	15	0,60	7909,45	96,82	0,0122
6	0,65	5441,85	93,12	0,0171	9	0,65	8732,30	97,16	0,0111	12	0,65	8851,50	97,16	0,0110	15	0,65	8856,23	97,16	0,0110
6	0,70	7638,68	96,64	0,0127	9	0,70	10597,43	98,19	0,0093	12	0,70	10672,03	98,19	0,0092	15	0,70	10675,35	98,19	0,0092
6	0,75	10494,18	98,62	0,0094	9	0,75	12878,15	99,05	0,0077	12	0,75	12920,28	99,05	0,0077	15	0,75	12922,05	99,05	0,0077
6	0,80	12213,13	99,14	0,0081	9	0,80	14215,63	99,48	0,0070	12	0,80	14240,95	99,48	0,0070	15	0,80	14241,08	99,48	0,0070

### Simple

Q = Cantidad promedio de mensajes *Query* por búsqueda. H = Porcentaje promedio de instancias halladas por búsqueda. E = Eficiencia del mensaje = H / Q

Referencia comparativa: Algoritmo BFS de *Gnutella* para TTL=4: Q = 4372,65 H = 72,80% E = 0,0166

**Tabla A-23 Rendimiento de la exploración incremental sobre una red con disponibilidad BAJA de recursos compartidos. Se han sombreado los casos de prueba no incrementales. Se ha recuadrado el mejor rendimiento no incremental —40 éxitos con menor tráfico— que se utiliza como referente comparativo para calcular la significancia estadística sobre la diferencia de medias observadas respecto de los demás casos. En todos los casos se ha utilizado TTL=8 sobre una red de 5.000 nodos.**

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,100	0	3 de 40	40												8,000	0,000	0	SI	
0,100	1	3 de 40	1	39										0,975	0,158	15,800	1,265	1,05E-25	SI
0,100	2	3 de 40	2	1	37									1,875	0,463	22,975	3,704	1,11E-25	SI
0,100	3	3 de 40	0	0	2	38								2,950	0,221	31,575	1,767	1,20E-25	SI
0,100	4	8 de 40	3	1	3	0	33							3,475	1,219	35,450	9,954	1,23E-25	SI
0,100	5	6 de 40	5	0	0	0	0	35						4,375	1,675	41,150	13,600	1,29E-25	SI
0,100	6	10 de 40	1	3	2	2	1	0	31					5,075	1,831	44,925	13,944	1,33E-25	SI
0,100	7	10 de 40	4	1	0	0	3	1	0	31				5,875	2,334	47,950	18,510	1,35E-25	SI
0,100	8	11 de 40	2	1	4	0	1	1	1	0	30			6,600	2,639	49,100	19,934	1,36E-25	SI
0,100	9	5 de 40	0	0	0	2	1	0	1	0	1	35		8,475	1,569	53,350	17,965	1,42E-25	SI
0,100	10	8 de 40	2	1	0	0	0	2	0	2	0	1	32	8,850	2,760	58,150	22,063	1,46E-25	SI
0,125	0	2 de 40	40												8,000	0,000	0	SI	
0,125	1	3 de 40	2	38										0,950	0,221	15,600	1,766	1,04E-25	SI
0,125	2	8 de 40	6	0	34									1,700	0,723	21,600	5,786	1,10E-25	SI
0,125	3	6 de 40	3	1	0	36								2,725	0,847	29,750	6,766	1,18E-25	SI
0,125	4	9 de 40	3	1	3	1	32							3,450	1,218	34,925	9,754	1,22E-25	SI
0,125	5	10 de 40	0	2	2	3	1	32						4,475	1,154	41,775	9,483	1,30E-25	SI
0,125	6	7 de 40	0	0	2	2	0	0	36					5,650	1,075	48,600	10,310	1,37E-25	SI
0,125	7	10 de 40	2	0	3	1	2	1	0	31				5,975	2,082	46,500	16,517	1,34E-25	SI
0,125	8	15 de 40	1	1	2	4	2	3	1	1	25			6,325	2,464	51,800	19,000	1,40E-25	SI
0,150	0	3 de 40	40												8,000	0,000	0	SI	
0,150	1	7 de 40	5	35										0,875	0,335	14,925	2,693	1,04E-25	SI
0,150	2	5 de 40	3	0	37									1,850	0,533	22,775	4,264	1,11E-25	SI
0,150	3	7 de 40	3	0	3	34								2,700	0,823	29,600	6,582	1,17E-25	SI
0,150	4	9 de 40	2	4	2	0	32							3,400	1,257	34,200	9,921	1,22E-25	SI
0,150	5	13 de 40	2	2	2	1	3	30						4,275	1,485	39,575	11,890	1,27E-25	SI
0,150	6	11 de 40	1	3	1	4	1	0	30					5,025	1,819	43,500	14,408	1,31E-25	SI
0,150	7	12 de 40	2	1	3	1	0	2	1	30				5,900	2,182	46,400	16,525	1,34E-25	SI
0,175	0	3 de 40	40												16,375	9,325	1,05E-25	SI	
0,175	1	5 de 40	3	37										0,925	0,267	27,275	9,711	1,15E-25	SI
0,175	2	8 de 40	4	4	32									1,700	0,648	41,625	17,038	1,28E-25	SI

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos										Incrementos por Búsqueda		Querys generadas			Significativo
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	
0,175	3	14 de 40	5	5	3	27							2,300	1,114	51,950	24,295	1,38E-25	SI
0,175	4	10 de 40	1	3	1	4	31						3,525	1,037	70,250	32,684	1,58E-25	SI
0,175	5	14 de 40	2	2	4	2	2	28					4,100	1,566	65,825	23,722	1,56E-25	SI
0,175	6	12 de 40	2	0	2	3	2	0	31				5,175	1,693	91,750	43,147	1,83E-25	SI
0,200	0	6 de 40	40												65,575	43,719	1,46E-25	SI
0,200	1	18 de 40	10	30									0,750	0,439	99,775	52,584	1,88E-25	SI
0,200	2	18 de 40	11	7	22								1,275	0,877	137,600	76,828	2,27E-25	SI
0,200	3	25 de 40	12	3	5	20							1,825	1,338	143,925	83,239	2,29E-25	SI
0,200	4	25 de 40	13	5	3	2	17						2,125	1,800	154,925	93,032	2,33E-25	SI
0,200	5	24 de 40	7	6	2	4	3	18					3,100	2,048	223,900	152,199	2,30E-25	SI
0,225	0	14 de 40	40												84,525	45,923	1,70E-25	SI
0,225	1	19 de 40	12	28									0,700	0,464	166,700	96,240	2,52E-25	SI
0,225	2	24 de 40	12	8	20								1,200	0,883	220,125	120,354	3,22E-25	SI
0,225	3	32 de 40	15	8	6	11							1,325	1,248	217,650	128,399	2,89E-25	SI
0,225	4	29 de 40	10	10	3	2	15						2,050	1,694	259,300	176,341	2,28E-25	SI
0,225	5	27 de 40	6	7	9	3	2	13					2,675	1,900	273,450	158,698	3,32E-25	SI
0,250	0	21 de 40	40												218,125	155,361	2,09E-25	SI
0,250	1	27 de 40	18	22									0,550	0,504	347,650	225,473	2,40E-25	SI
0,250	2	35 de 40	23	6	11								0,700	0,883	415,950	184,730	8,89E-25	SI
0,250	3	34 de 40	22	7	3	8							0,925	1,207	409,025	309,706	8,70E-26	SI
0,250	4	35 de 40	25	5	5	0	5						0,875	1,381	393,825	320,105	5,95E-26	SI
0,275	0	25 de 40	40												475,750	237,130	6,96E-25	SI
0,275	1	34 de 40	26	14									0,350	0,483	600,975	356,220	2,64E-25	SI
0,275	2	35 de 40	26	9	5								0,475	0,716	655,850	565,015	9,62E-27	SI
0,275	3	37 de 40	29	4	2	5							0,575	1,059	608,400	357,316	2,81E-25	SI
0,275	4	37 de 40	25	8	2	2	3						0,750	1,235	804,425	526,865	1,29E-25	SI
0,300	0	34 de 40	40												777,375	382,654	1,21E-24	SI
0,300	1	38 de 40	34	6									0,150	0,362	939,325	487,930	1,46E-24	SI
0,300	2	40 de 40	37	3	0								0,075	0,267	931,875	448,010	2,55E-24	SI
0,300	3	39 de 40	32	6	0	2							0,300	0,723	1036,075	498,163	4,72E-24	SI
0,300	4	40 de 40	32	4	2	2	0						0,350	0,802	1020,550	438,162	9,60E-24	SI
0,325	0	32 de 40	40												842,875	387,999	2,42E-24	SI
0,325	1	35 de 40	32	8									0,200	0,405	923,025	542,040	5,30E-25	SI
0,325	2	36 de 40	29	5	6								0,425	0,747	1278,025	668,941	4,52E-23	SI
0,325	3	39 de 40	34	3	2	1							0,250	0,670	886,750	483,169	7,82E-25	SI
0,325	4	40 de 40	34	5	1	0	0						0,175	0,446	932,950	443,864	2,77E-24	SI

Continúa la tabla...

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,350	0	32 de 40	40												1770,925	737,620	2,91E-19	SI	
0,350	1	40 de 40	36	4									0,100	0,304	2129,900	1049,141	2,18E-14	SI	
0,350	2	40 de 40	40	0	0								0,000	0,000	2078,700	591,774	4,73E-17	SI	
0,350	3	40 de 40	37	2	1	0							0,100	0,379	2119,225	953,810	3,36E-15	SI	
0,375	0	39 de 40	40												3164,250	1079,396	3,68E-06	SI	
0,375	1	40 de 40	39	1									0,025	0,158	3472,675	1195,347	7,15E-04	SI	
0,375	2	40 de 40	40	0	0								0,000	0,000	3335,875	978,195	2,56E-05	SI	
0,375	3	40 de 40	39	1	0	0							0,025	0,158	3332,225	1025,327	3,40E-05	SI	
0,400	0	40 de 40	40												4376,975	1097,114		***	
0,400	1	40 de 40	39	1									0,025	0,158	4475,250	1112,557	6,92E-01	NO	
0,400	2	40 de 40	39	1	0								0,025	0,158	4524,425	1090,577	5,48E-01	NO	
0,400	3	40 de 40	40	0	0	0							0,000	0,000	4428,625	944,460	8,22E-01	NO	

Escenario I

Disponibilidad de recursos

en la red: BAJA

Cantidad promedio de instancias existentes en toda la red por cada recurso buscado: 2,52.

INC = Cantidad máxima de incrementos permitidos.

Referente comparativo: 4.376,975 Querys —Estrategia no incremental con  $\gamma = 0.400$ —

**Tabla A-24 Rendimiento de la exploración incremental sobre una red con disponibilidad MEDIA-BAJA de recursos compartidos. Se han sombreado los casos de prueba no incrementales. Se ha recuadrado el mejor rendimiento no incremental —40 éxitos con menor tráfico— que se utiliza como referente comparativo para calcular la significancia estadística sobre la diferencia de medias observadas respecto de los demás casos. En todos los casos se ha utilizado TTL=8 sobre una red de 5.000 nodos.**

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas				
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo	
0,100	0	16 de 40	40												8,000	0,000	1,80E-13	SI		
0,100	1	22 de 40	16	24											0,600	0,496	12,800	3,969	2,65E-13	SI
0,100	2	26 de 40	9	13	18										1,225	0,800	17,775	6,379	3,94E-13	SI
0,100	3	31 de 40	13	10	4	13									1,425	1,259	19,325	10,106	4,41E-13	SI
0,100	4	36 de 40	14	8	8	4	6								1,500	1,450	20,000	11,600	4,63E-13	SI
0,100	5	34 de 40	17	7	3	2	4	7							1,750	1,984	21,175	14,650	5,02E-13	SI
0,100	6	39 de 40	14	10	5	5	1	3	2						1,650	1,819	21,175	14,579	5,02E-13	SI
0,100	7	35 de 40	12	9	5	2	5	1	1	5					2,275	2,407	24,000	16,204	6,27E-13	SI
0,100	8	37 de 40	16	7	5	2	0	6	1	0	3				2,075	2,536	22,800	16,951	5,65E-13	SI
0,100	9	38 de 40	19	6	5	5	0	2	1	0	0	2			1,625	2,339	18,900	13,677	4,18E-13	SI
0,100	10	37 de 40	18	10	1	2	3	1	1	1	0	0	3		1,950	2,943	20,550	16,928	4,69E-13	SI
0,125	0	13 de 40	40												8,000	0,000	1,80E-13	SI		
0,125	1	20 de 40	14	26											0,650	0,483	13,200	3,864	2,73E-13	SI
0,125	2	23 de 40	14	7	19										1,125	0,911	16,900	7,218	3,66E-13	SI
0,125	3	30 de 40	16	7	3	14									1,375	1,334	18,950	10,727	4,27E-13	SI
0,125	4	34 de 40	11	10	6	4	9								1,750	1,532	21,800	12,015	5,36E-13	SI
0,125	5	36 de 40	10	12	7	4	2	5							1,775	1,656	22,000	12,906	5,42E-13	SI
0,125	6	35 de 40	15	7	5	2	3	3	5						2,000	2,196	22,475	15,201	5,56E-13	SI
0,125	7	37 de 40	18	4	8	2	3	1	1	3					1,750	2,193	20,875	15,681	4,86E-13	SI
0,125	8	37 de 40	17	7	6	3	0	1	1	2	3				1,925	2,586	20,525	15,733	4,72E-13	SI
0,150	0	19 de 40	40												7,900	0,441	1,79E-13	SI		
0,150	1	23 de 40	17	23											0,575	0,501	12,550	4,076	2,59E-13	SI
0,150	2	28 de 40	15	8	17										1,050	0,904	16,400	7,235	3,51E-13	SI
0,150	3	37 de 40	18	11	5	6									0,975	1,097	15,725	8,721	3,31E-13	SI
0,150	4	32 de 40	9	9	10	4	8								1,825	1,430	22,400	11,197	5,65E-13	SI
0,150	5	37 de 40	15	7	5	7	3	3							1,625	1,659	21,000	13,274	4,98E-13	SI
0,150	6	33 de 40	11	10	3	3	4	2	7						2,325	2,258	25,575	16,713	7,12E-13	SI
0,150	7	36 de 40	11	9	7	3	2	3	0	5					2,250	2,340	25,050	17,477	6,78E-13	SI
0,175	0	28 de 40	40												15,275	10,903	3,16E-13	SI		
0,175	1	29 de 40	19	21											0,525	0,506	29,475	15,655	9,92E-13	SI
0,175	2	32 de 40	14	17	9										0,875	0,757	32,950	19,393	1,29E-12	SI

Continúa la tabla...



$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos										Incrementos por Búsqueda		Querys generadas			Significativo
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	
0,175	3	37 de 40	19	13	3	5							0,850	1,027	31,250	19,640	1,12E-12	SI
0,175	4	36 de 40	21	9	5	1	4						0,950	1,300	30,500	18,848	1,06E-12	SI
0,175	5	37 de 40	23	4	4	3	0	6					1,275	1,840	35,425	27,006	1,50E-12	SI
0,175	6	39 de 40	20	9	6	2	2	0	1				1,025	1,405	34,525	24,164	1,43E-12	SI
<b>0,200</b>	<b>0</b>	<b>28 de 40</b>	<b>40</b>												<b>46,700</b>	<b>30,865</b>	<b>3,83E-12</b>	<b>SI</b>
0,200	1	38 de 40	34	6									0,150	0,362	63,575	35,601	1,63E-11	SI
0,200	2	38 de 40	34	3	3								0,225	0,577	59,425	30,355	1,17E-11	SI
0,200	3	40 de 40	31	7	2	0							0,275	0,554	63,725	32,242	1,69E-11	SI
0,200	4	40 de 40	31	7	2	0	0						0,275	0,554	63,275	40,506	1,54E-11	SI
0,200	5	40 de 40	29	8	1	1	1	0					0,425	0,874	66,550	46,382	1,99E-11	SI
<b>0,225</b>	<b>0</b>	<b>35 de 40</b>	<b>40</b>												<b>89,550</b>	<b>46,162</b>	<b>1,66E-10</b>	<b>SI</b>
0,225	1	40 de 40	35	5									0,125	0,335	99,700	45,346	4,31E-10	SI
0,225	2	40 de 40	37	3	0								0,075	0,267	99,175	52,844	4,08E-10	SI
0,225	3	40 de 40	35	5	0	0							0,125	0,335	104,575	46,979	6,82E-10	SI
0,225	4	40 de 40	37	3	0	0	0						0,075	0,267	102,600	53,792	5,66E-10	SI
0,225	5	40 de 40	35	5	0	0	0	0					0,125	0,335	104,125	45,464	6,54E-10	SI
<b>0,250</b>	<b>0</b>	<b>39 de 40</b>	<b>40</b>												<b>263,750</b>	<b>140,204</b>	<b>6,53E-03</b>	<b>NO</b>
0,250	1	40 de 40	40	0									0,000	0,000	266,950	123,129	6,30E-03	NO
0,250	2	40 de 40	39	1	0								0,025	0,158	324,075	165,823	2,20E-01	NO
0,250	3	40 de 40	39	1	0	0							0,025	0,158	259,625	125,982	3,88E-03	NO
0,250	4	40 de 40	39	1	0	0	0						0,025	0,158	289,275	143,262	3,43E-02	NO
<b>0,275</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>376,900</b>	<b>212,920</b>	<b>***</b>	
0,275	1	40 de 40	38	2									0,050	0,221	368,125	223,219	8,58E-01	NO
0,275	2	40 de 40	39	1	0								0,025	0,158	383,975	254,275	8,93E-01	NO
0,275	3	40 de 40	40	0	0	0							0,000	0,000	410,725	221,232	4,88E-01	NO
0,275	4	40 de 40	38	2	0	0	0						0,050	0,221	432,475	203,295	2,36E-01	NO
<b>0,300</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>687,900</b>	<b>358,987</b>	<b>1,38E-05</b>	<b>SI</b>
0,300	1	40 de 40	39	1									0,025	0,158	675,575	381,131	5,69E-05	SI
0,300	2	40 de 40	40	0	0								0,000	0,000	882,825	377,399	4,82E-10	SI
0,300	3	40 de 40	40	0	0	0							0,000	0,000	881,500	389,760	1,17E-09	SI
0,300	4	40 de 40	40	0	0	0	0						0,000	0,000	778,225	365,573	1,08E-07	SI
<b>0,325</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>754,675</b>	<b>397,763</b>	<b>1,80E-06</b>	<b>SI</b>
0,325	1	40 de 40	40	0									0,000	0,000	851,700	347,111	3,86E-10	SI
0,325	2	40 de 40	40	0	0								0,000	0,000	814,275	423,545	2,60E-07	SI
0,325	3	40 de 40	40	0	0	0							0,000	0,000	861,575	382,989	2,36E-09	SI
0,325	4	40 de 40	39	1	0	0	0						0,025	0,158	809,100	367,055	1,90E-08	SI

Continúa la tabla...

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,350	0	40 de 40	40												1998,975	744,478	0	SI	
0,350	1	40 de 40	40	0										0,000	0,000	1946,575	840,687	8,88E-15	SI
0,350	2	40 de 40	40	0	0									0,000	0,000	2082,675	650,079	0	SI
0,350	3	40 de 40	40	0	0	0								0,000	0,000	1927,575	675,125	0	SI
0,375	0	40 de 40	40													3505,725	869,919	0	SI
0,375	1	40 de 40	40	0										0,000	0,000	3047,800	1046,022	0	SI
0,375	2	40 de 40	40	0	0									0,000	0,000	3532,925	821,785	0	SI
0,375	3	40 de 40	40	0	0	0								0,000	0,000	3101,675	1025,760	0	SI
0,400	0	40 de 40	40													4556,725	916,713	0	SI
0,400	1	40 de 40	40	0										0,000	0,000	4779,700	815,668	0	SI
0,400	2	40 de 40	40	0	0									0,000	0,000	4365,925	1033,920	0	SI
0,400	3	40 de 40	40	0	0	0								0,000	0,000	4458,650	998,599	0	SI

**Escenario II**

**Cantidad promedio de instancias existentes en toda la red por cada recurso buscado: 24,87.**

**Disponibilidad de recursos  
en la red: MEDIA-BAJA**

INC = Cantidad máxima de incrementos permitidos.

Referente comparativo: 376,900 Querys —Estrategia no incremental con  $\gamma = 0.275$ —

**Tabla A-25 Rendimiento de la exploración incremental sobre una red con disponibilidad MEDIA de recursos compartidos. Se han sombreado los casos de prueba no incrementales. Se ha recuadrado el mejor rendimiento no incremental —40 éxitos con menor tráfico— que se utiliza como referente comparativo para calcular la significancia estadística sobre la diferencia de medias observadas respecto de los demás casos. En todos los casos se ha utilizado TTL=8 sobre una red de 5.000 nodos.**

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,100	0	23 de 40	40												7,925	0,474	4,10E-14	SI	
0,100	1	34 de 40	26	14										0,350	0,483	10,800	3,864	5,74E-14	SI
0,100	2	36 de 40	26	7	7									0,525	0,784	12,200	6,272	6,70E-14	SI
0,100	3	39 de 40	27	6	4	3								0,575	0,958	12,600	7,662	6,95E-14	SI
0,100	4	40 de 40	22	13	4	0	1							0,625	0,868	12,925	6,919	7,28E-14	SI
0,100	5	40 de 40	25	6	4	3	2	0						0,775	1,209	14,100	9,535	8,19E-14	SI
0,100	6	40 de 40	26	9	3	1	1	0	0					0,550	0,932	12,400	7,459	6,80E-14	SI
0,100	7	40 de 40	20	12	6	2	0	0	0	0				0,750	0,899	14,000	7,190	8,29E-14	SI
0,100	8	40 de 40	20	14	3	3	0	0	0	0	0			0,725	0,905	13,800	7,244	8,08E-14	SI
0,100	9	39 de 40	21	9	3	4	2	0	0	0	0	1		1,100	1,766	15,800	10,484	9,98E-14	SI
0,100	10	40 de 40	23	10	5	2	0	0	0	0	0	0		0,650	0,893	13,200	7,144	7,52E-14	SI
0,125	0	23 de 40	40												8,000	0,000	4,13E-14	SI	
0,125	1	33 de 40	23	17										0,425	0,501	11,400	4,005	6,17E-14	SI
0,125	2	39 de 40	23	9	8									0,625	0,807	13,000	6,453	7,38E-14	SI
0,125	3	39 de 40	24	11	4	1								0,550	0,783	12,350	6,306	6,82E-14	SI
0,125	4	39 de 40	26	7	6	0	1							0,575	0,931	12,400	6,766	6,84E-14	SI
0,125	5	40 de 40	23	10	3	1	2	1						0,800	1,265	14,400	10,119	8,43E-14	SI
0,125	6	38 de 40	20	10	3	3	1	1	2					1,150	1,673	16,000	10,564	1,02E-13	SI
0,125	7	40 de 40	27	5	4	1	3	0	0	0				0,700	1,224	13,600	9,790	7,67E-14	SI
0,125	8	38 de 40	21	11	2	2	2	0	0	0	2			1,125	1,937	15,600	11,012	9,67E-14	SI
0,150	0	24 de 40	40												7,900	0,632	4,08E-14	SI	
0,150	1	36 de 40	29	11										0,275	0,452	10,200	3,618	5,35E-14	SI
0,150	2	37 de 40	26	9	5									0,475	0,716	11,800	5,725	6,41E-14	SI
0,150	3	40 de 40	21	9	6	4								0,825	1,035	14,575	8,277	8,80E-14	SI
0,150	4	36 de 40	13	11	8	3	5							1,400	1,355	19,125	10,929	1,50E-13	SI
0,150	5	39 de 40	26	7	4	2	0	1						0,650	1,122	13,175	8,924	7,36E-14	SI
0,150	6	40 de 40	25	9	6	0	0	0	0					0,525	0,751	12,150	5,977	6,68E-14	SI
0,150	7	40 de 40	27	8	1	2	1	1	0	0				0,625	1,192	12,900	9,607	7,06E-14	SI
0,175	0	31 de 40	40												15,775	8,453	1,02E-13	SI	
0,175	1	36 de 40	34	6										0,150	0,362	19,850	9,812	1,66E-13	SI
0,175	2	38 de 40	34	4	2									0,200	0,516	23,150	10,098	2,50E-13	SI

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos										Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor
0,175	3	38 de 40	33	5	0	2							0,275	0,716	19,625	13,673	1,54E-13	SI
0,175	4	38 de 40	23	12	3	0	2						0,650	1,001	23,075	11,543	2,44E-13	SI
0,175	5	40 de 40	27	9	0	4	0	0					0,525	0,933	23,500	12,496	2,54E-13	SI
0,175	6	38 de 40	25	8	4	0	1	0	2				0,800	1,488	25,900	18,284	3,15E-13	SI
<b>0,200</b>	<b>0</b>	<b>38 de 40</b>	<b>40</b>												<b>62,300</b>	<b>34,918</b>	<b>3,48E-11</b>	<b>SI</b>
0,200	1	40 de 40	38	2									0,050	0,221	59,425	32,926	2,35E-11	SI
0,200	2	40 de 40	37	3	0								0,075	0,267	64,450	29,138	4,92E-11	SI
0,200	3	40 de 40	36	4	0	0							0,100	0,304	60,000	38,699	2,45E-11	SI
0,200	4	40 de 40	35	4	1	0	0						0,150	0,427	68,400	46,411	8,39E-11	SI
0,200	5	40 de 40	38	2	0	0	0	0					0,050	0,221	61,650	34,550	3,18E-11	SI
<b>0,225</b>	<b>0</b>	<b>38 de 40</b>	<b>40</b>												<b>89,625</b>	<b>45,334</b>	<b>2,03E-09</b>	<b>SI</b>
0,225	1	40 de 40	39	1									0,025	0,158	93,875	47,472	3,98E-09	SI
0,225	2	40 de 40	40	0	0								0,000	0,000	77,500	36,574	3,14E-10	SI
0,225	3	40 de 40	40	0	0	0							0,000	0,000	96,200	43,228	5,35E-09	SI
0,225	4	40 de 40	38	2	0	0	0						0,050	0,221	100,100	42,797	9,61E-09	SI
0,225	5	40 de 40	38	2	0	0	0	0					0,050	0,221	102,175	54,777	1,65E-08	SI
<b>0,250</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>259,725</b>	<b>138,332</b>		<b>***</b>
0,250	1	40 de 40	39	1									0,025	0,158	272,125	167,752	7,19E-01	NO
0,250	2	40 de 40	40	0	0								0,000	0,000	236,900	145,526	4,74E-01	NO
0,250	3	40 de 40	40	0	0	0							0,000	0,000	210,975	133,580	1,13E-01	NO
0,250	4	40 de 40	40	0	0	0	0						0,000	0,000	223,850	131,113	2,37E-01	NO
<b>0,275</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>462,300</b>	<b>248,985</b>	<b>3,14E-05</b>	<b>SI</b>
0,275	1	40 de 40	40	0									0,000	0,000	469,975	265,191	3,95E-05	SI
0,275	2	40 de 40	39	1	0								0,025	0,158	286,325	170,651	4,46E-01	NO
0,275	3	40 de 40	39	1	0	0							0,025	0,158	402,325	271,231	4,42E-03	NO
0,275	4	40 de 40	40	0	0	0	0						0,000	0,000	452,325	263,347	1,30E-04	SI
<b>0,300</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>789,675</b>	<b>444,252</b>	<b>4,26E-09</b>	<b>SI</b>
0,300	1	40 de 40	40	0									0,000	0,000	884,725	306,978	2,22E-16	SI
0,300	2	40 de 40	40	0	0								0,000	0,000	842,975	374,285	2,34E-12	SI
0,300	3	40 de 40	40	0	0	0							0,000	0,000	896,925	448,510	3,86E-11	SI
0,300	4	40 de 40	40	0	0	0	0						0,000	0,000	782,900	309,448	1,60E-13	SI
<b>0,325</b>	<b>0</b>	<b>40 de 40</b>	<b>40</b>												<b>811,000</b>	<b>448,732</b>	<b>2,02E-09</b>	<b>SI</b>
0,325	1	40 de 40	40	0									0,000	0,000	964,350	352,084	4,44E-16	SI
0,325	2	40 de 40	40	0	0								0,000	0,000	876,825	415,453	1,04E-11	SI
0,325	3	40 de 40	40	0	0	0							0,000	0,000	845,525	399,267	1,50E-11	SI
0,325	4	40 de 40	40	0	0	0	0						0,000	0,000	808,375	390,915	5,58E-11	SI

Continúa la tabla...

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,350	0	40 de 40	40												2158,200	781,380	0	SI	
0,350	1	40 de 40	40	0										0,000	0,000	2092,225	776,962	0	SI
0,350	2	40 de 40	40	0	0									0,000	0,000	1936,950	762,834	0	SI
0,350	3	40 de 40	40	0	0	0								0,000	0,000	2121,300	754,220	0	SI
0,375	0	40 de 40	40												3108,275	1046,987	0	SI	
0,375	1	40 de 40	40	0										0,000	0,000	3241,100	898,084	0	SI
0,375	2	40 de 40	40	0	0									0,000	0,000	3261,825	861,607	0	SI
0,375	3	40 de 40	40	0	0	0								0,000	0,000	3146,375	1157,688	0	SI
0,400	0	40 de 40	40												4485,700	1089,029	0	SI	
0,400	1	40 de 40	40	0										0,000	0,000	4219,350	1230,505	0	SI
0,400	2	40 de 40	40	0	0									0,000	0,000	4428,275	1069,211	0	SI
0,400	3	40 de 40	40	0	0	0								0,000	0,000	4226,225	1093,335	0	SI

**Escenario III**

**Cantidad promedio de instancias existentes en toda la red por cada recurso buscado: 49,85.**

**Disponibilidad de recursos**

INC = Cantidad máxima de incrementos permitidos.

**en la red: MEDIA**

Referente comparativo: 259,725 Querys —Estrategia no incremental con  $\gamma= 0.250$ —

**Tabla A-26 Rendimiento de la exploración incremental sobre una red con disponibilidad ALTA de recursos compartidos. Se han sombreado los casos de prueba no incrementales. Se ha recuadrado el mejor rendimiento no incremental —40 éxitos con menor tráfico— que se utiliza como referente comparativo para calcular la significancia estadística sobre la diferencia de medias observadas respecto de los demás casos. En todos los casos se ha utilizado TTL=8 sobre una red de 5.000 nodos.**

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,100	0	32 de 40	40												8,000	0,000	2,62E-13	SI	
0,100	1	39 de 40	29	11										0,275	0,452	10,125	3,695	5,72E-13	SI
0,100	2	39 de 40	32	3	5									0,325	0,694	10,600	5,551	6,41E-13	SI
0,100	3	40 de 40	32	5	2	1								0,300	0,687	10,400	5,495	5,92E-13	SI
0,100	4	40 de 40	33	3	4	0	0							0,275	0,640	10,200	5,120	5,55E-13	SI
0,100	5	40 de 40	33	4	0	2	1	0						0,350	0,921	10,800	7,370	6,30E-13	SI
0,100	6	40 de 40	28	7	2	2	1	0	0					0,525	0,987	12,200	7,894	1,10E-12	SI
0,100	7	40 de 40	29	6	4	1	0	0	0	0				0,425	0,781	11,375	6,163	8,56E-13	SI
0,100	8	40 de 40	25	14	0	1	0	0	0	0	0			0,425	0,636	11,400	5,088	9,10E-13	SI
0,100	9	40 de 40	35	4	1	0	0	0	0	0	0	0		0,150	0,427	9,200	3,413	3,97E-13	SI
0,100	10	40 de 40	30	7	2	1	0	0	0	0	0	0		0,350	0,700	10,800	5,599	6,94E-13	SI
0,125	0	27 de 40	40												8,000	0,000	2,62E-13	SI	
0,125	1	35 de 40	28	12										0,300	0,464	10,300	3,831	6,11E-13	SI
0,125	2	39 de 40	26	8	6									0,500	0,751	11,925	5,976	1,09E-12	SI
0,125	3	40 de 40	25	11	1	3								0,550	0,876	12,375	6,993	1,25E-12	SI
0,125	4	40 de 40	29	6	3	1	1							0,475	0,933	11,750	7,469	9,33E-13	SI
0,125	5	40 de 40	34	5	0	0	1	0						0,225	0,698	9,800	5,580	4,60E-13	SI
0,125	6	39 de 40	31	6	1	0	0	1	1					0,475	1,261	11,150	7,778	7,12E-13	SI
0,125	7	40 de 40	27	12	1	0	0	0	0	0				0,350	0,533	10,800	4,268	7,36E-13	SI
0,125	8	40 de 40	31	6	0	0	2	1	0	0	0			0,475	1,176	11,800	9,411	8,52E-13	SI
0,150	0	33 de 40	40												8,000	0,000	2,62E-13	SI	
0,150	1	39 de 40	32	8										0,200	0,405	9,550	3,281	4,59E-13	SI
0,150	2	40 de 40	34	5	1									0,175	0,446	9,325	3,633	4,14E-13	SI
0,150	3	40 de 40	33	7	0	0								0,175	0,385	9,400	3,078	4,35E-13	SI
0,150	4	40 de 40	31	8	1	0	0							0,250	0,494	9,925	4,015	5,21E-13	SI
0,150	5	40 de 40	33	5	2	0	0	0						0,225	0,530	9,800	4,244	4,90E-13	SI
0,150	6	40 de 40	34	3	3	0	0	0	0					0,225	0,577	9,775	4,627	4,77E-13	SI
0,150	7	39 de 40	31	6	1	1	0	0	0	1				0,450	1,239	10,800	6,157	6,74E-13	SI
0,175	0	34 de 40	40												16,575	8,572	7,02E-12	SI	
0,175	1	38 de 40	30	10										0,250	0,439	17,525	8,674	1,06E-11	SI
0,175	2	40 de 40	40	0	0									0,000	0,000	15,875	7,559	5,40E-12	SI

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos										Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor
0,175	3	40 de 40	35	3	2	0							0,175	0,501	18,325	10,584	1,42E-11	SI
0,175	4	40 de 40	36	2	2	0	0						0,150	0,483	18,875	11,225	1,79E-11	SI
0,175	5	40 de 40	37	3	0	0	0	0					0,075	0,267	18,650	10,736	1,64E-11	SI
0,175	6	40 de 40	34	6	0	0	0	0	0				0,150	0,362	16,400	8,732	6,46E-12	SI
0,200	0	39 de 40	40												52,475	30,482	6,73E-04	SI
0,200	1	40 de 40	40	0									0,000	0,000	61,275	33,969	1,81E-02	NO
0,200	2	40 de 40	40	0	0								0,000	0,000	56,975	30,574	3,52E-03	NO
0,200	3	40 de 40	39	1	0	0							0,025	0,158	61,050	33,625	1,66E-02	NO
0,200	4	40 de 40	38	2	0	0	0						0,050	0,221	54,700	33,432	2,10E-03	NO
0,200	5	40 de 40	39	1	0	0	0	0					0,025	0,158	58,150	33,068	6,49E-03	NO
0,225	0	40 de 40	40												82,350	43,459		***
0,225	1	40 de 40	40	0									0,000	0,000	91,975	46,102	3,40E-01	NO
0,225	2	40 de 40	40	0	0								0,000	0,000	95,350	58,639	2,64E-01	NO
0,225	3	40 de 40	39	1	0	0							0,025	0,158	89,275	49,075	5,06E-01	NO
0,225	4	40 de 40	40	0	0	0	0						0,000	0,000	89,275	43,007	4,76E-01	NO
0,225	5	40 de 40	40	0	0	0	0	0					0,000	0,000	77,000	43,976	5,86E-01	NO
0,250	0	40 de 40	40												239,375	148,687	7,24E-08	SI
0,250	1	40 de 40	40	0									0,000	0,000	259,500	154,537	1,08E-08	SI
0,250	2	40 de 40	40	0	0								0,000	0,000	211,725	122,644	8,63E-08	SI
0,250	3	40 de 40	39	1	0	0							0,025	0,158	245,450	147,532	2,58E-08	SI
0,250	4	40 de 40	40	0	0	0	0						0,000	0,000	267,950	155,919	4,30E-09	SI
0,275	0	40 de 40	40												464,325	256,853	1,21E-11	SI
0,275	1	40 de 40	40	0									0,000	0,000	472,950	251,012	3,29E-12	SI
0,275	2	40 de 40	40	0	0								0,000	0,000	428,725	256,406	1,67E-10	SI
0,275	3	40 de 40	40	0	0	0							0,000	0,000	456,500	297,710	1,08E-09	SI
0,275	4	40 de 40	40	0	0	0	0						0,000	0,000	358,725	234,566	5,20E-09	SI
0,300	0	40 de 40	40												761,675	369,946	2,62E-14	SI
0,300	1	40 de 40	40	0									0,000	0,000	766,950	339,723	1,33E-15	SI
0,300	2	40 de 40	40	0	0								0,000	0,000	799,625	373,238	6,22E-15	SI
0,300	3	40 de 40	40	0	0	0							0,000	0,000	713,425	388,869	1,09E-12	SI
0,300	4	40 de 40	40	0	0	0	0						0,000	0,000	889,000	362,531	0	SI
0,325	0	40 de 40	40												825,675	433,870	2,26E-13	SI
0,325	1	40 de 40	40	0									0,000	0,000	840,950	338,450	0	SI
0,325	2	40 de 40	40	0	0								0,000	0,000	806,350	346,077	4,44E-16	SI
0,325	3	40 de 40	40	0	0	0							0,000	0,000	811,825	469,818	4,01E-12	SI
0,325	4	40 de 40	40	0	0	0	0						0,000	0,000	845,450	373,891	8,88E-16	SI

Continúa la tabla...

$\gamma$	INC	Éxitos	Cantidad de búsquedas que utilizaron i incrementos											Incrementos por Búsqueda		Querys generadas			
			i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8	i=9	i=10	Media	Desv. Est.	Media	Desv. Est.	P-valor	Significativo
0,350	0	40 de 40	40												1983,125	712,582	0	SI	
0,350	1	40 de 40	40	0										0,000	0,000	1906,675	682,775	0	SI
0,350	2	40 de 40	40	0	0									0,000	0,000	1861,800	735,009	0	SI
0,350	3	40 de 40	40	0	0	0								0,000	0,000	2162,000	640,475	0	SI
0,375	0	40 de 40	40													3343,000	977,748	0	SI
0,375	1	40 de 40	40	0										0,000	0,000	3140,400	1026,788	0	SI
0,375	2	40 de 40	40	0	0									0,000	0,000	3177,125	1105,720	0	SI
0,375	3	40 de 40	40	0	0	0								0,000	0,000	3432,775	887,922	0	SI
0,400	0	40 de 40	40													4421,300	1179,538	0	SI
0,400	1	40 de 40	40	0										0,000	0,000	4702,725	774,154	0	SI
0,400	2	40 de 40	40	0	0									0,000	0,000	4639,150	919,505	0	SI
0,400	3	40 de 40	40	0	0	0								0,000	0,000	4453,425	1045,557	0	SI

**Escenario IV**

**Cantidad promedio de instancias existentes en toda la red por cada recurso buscado: 99,08.**

**Disponibilidad de recursos**

INC = Cantidad máxima de incrementos permitidos.

**en la red: ALTA**

Referente comparativo: 82,350 Querys —Estrategia no incremental con  $\gamma = 0.225$ —



**Tabla A-27 Cantidad de éxitos obtenidos por prueba realizada en una red con disponibilidad BAJA de recursos compartidos. Cada prueba consistió en la realización de 40 búsquedas independientes. Se realizaron 40 pruebas independientes registrando la cantidad de éxitos obtenidos en cada prueba para calcular la media y desvío estándar mostrados en la tabla.**

$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos	
		Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.
0,100	0	2,175	1,299	0,150	0	2,350	1,167	0,200	5	25,325	2,645	0,300	3	38,650	1,051
0,100	1	4,350	1,805	0,150	1	4,025	1,860	0,225	0	12,000	2,552	0,300	4	38,975	1,143
0,100	2	5,125	2,300	0,150	2	4,900	2,061	0,225	1	19,725	2,783	0,325	0	31,125	2,524
0,100	3	6,650	2,348	0,150	3	7,475	2,810	0,225	2	23,425	3,630	0,325	1	36,375	1,628
0,100	4	8,100	2,619	0,150	4	8,650	2,486	0,225	3	27,325	2,693	0,325	2	37,775	1,544
0,100	5	9,375	2,967	0,150	5	9,450	2,773	0,225	4	29,425	2,986	0,325	3	38,850	0,834
0,100	6	9,900	2,416	0,150	6	10,250	2,687	0,225	5	30,200	2,821	0,325	4	39,075	0,859
0,100	7	10,100	2,550	0,150	7	10,750	2,853	0,250	0	20,700	2,911	0,350	0	36,700	2,041
0,100	8	10,625	2,993	0,175	0	3,725	1,811	0,250	1	29,075	2,895	0,350	1	39,175	0,874
0,100	9	11,375	2,752	0,175	1	6,275	2,112	0,250	2	32,925	2,586	0,350	2	39,475	0,679
0,100	10	10,775	2,769	0,175	2	8,450	2,459	0,250	3	34,425	1,810	0,350	3	39,800	0,608
0,125	0	2,350	1,460	0,175	3	11,075	2,731	0,250	4	35,250	1,794	0,375	0	38,750	0,981
0,125	1	3,575	1,738	0,175	4	12,325	3,261	0,275	0	24,900	3,128	0,375	1	39,650	0,622
0,125	2	5,150	2,370	0,175	5	15,375	3,232	0,275	1	32,825	2,854	0,375	2	39,825	0,385
0,125	3	6,850	2,656	0,175	6	14,900	2,649	0,275	2	35,325	2,080	0,375	3	39,925	0,350
0,125	4	8,550	2,562	0,200	0	8,775	3,340	0,275	3	36,675	1,700	0,400	0	39,275	0,905
0,125	5	9,225	2,616	0,200	1	14,725	3,021	0,275	4	37,075	1,831	0,400	1	39,975	0,158
0,125	6	9,250	2,499	0,200	2	18,600	3,169	0,300	0	31,525	2,698	0,400	2	39,975	0,158
0,125	7	10,625	2,425	0,200	3	22,675	2,999	0,300	1	36,075	1,845	0,400	3	39,975	0,158
0,125	8	11,025	2,359	0,200	4	24,225	3,000	0,300	2	38,475	1,467				

Escenario I

Disponibilidad de recursos  
en la red: BAJA

INC = Cantidad máxima de incrementos permitidos.

Se han sombreado los casos en que no se ha utilizado la estrategia de exploración incremental

**Tabla A-28 Cantidad de éxitos obtenidos por prueba realizada en una red con disponibilidad MEDIA-BAJA de recursos compartidos. Cada prueba consistió en la realización de 40 búsquedas independientes. Se realizaron 40 pruebas independientes registrando la cantidad de éxitos obtenidos en cada prueba para calcular la media y desvío estándar mostrados en la tabla.**

$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos		$\gamma$	INC	Éxitos	
		Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.
0,100	0	14,750	2,519	0,150	0	13,300	3,014	0,200	5	39,925	0,267	0,300	3	40,000	0,000
0,100	1	21,200	2,997	0,150	1	21,750	2,889	0,225	0	35,450	1,782	0,300	4	40,000	0,000
0,100	2	26,825	3,249	0,150	2	27,725	3,130	0,225	1	39,125	0,939	0,325	0	39,825	0,385
0,100	3	30,875	2,594	0,150	3	30,950	3,021	0,225	2	39,775	0,423	0,325	1	39,975	0,158
0,100	4	33,400	2,285	0,150	4	32,650	2,646	0,225	3	39,850	0,362	0,325	2	40,000	0,000
0,100	5	34,325	2,080	0,150	5	34,425	1,973	0,225	4	40,000	0,000	0,325	3	40,000	0,000
0,100	6	35,675	2,200	0,150	6	35,250	2,329	0,225	5	40,000	0,000	0,325	4	40,000	0,000
0,100	7	36,250	2,085	0,150	7	35,975	1,874	0,250	0	38,650	1,001	0,350	0	39,975	0,158
0,100	8	36,025	1,874	0,175	0	19,300	3,729	0,250	1	39,950	0,221	0,350	1	40,000	0,000
0,100	9	35,825	1,947	0,175	1	29,175	2,735	0,250	2	39,975	0,158	0,350	2	40,000	0,000
0,100	10	36,250	2,133	0,175	2	33,525	2,276	0,250	3	40,000	0,000	0,350	3	40,000	0,000
0,125	0	14,050	2,650	0,175	3	36,325	1,845	0,250	4	39,975	0,158	0,375	0	39,975	0,158
0,125	1	21,050	3,012	0,175	4	37,075	1,095	0,275	0	39,075	0,859	0,375	1	40,000	0,000
0,125	2	27,600	3,463	0,175	5	37,825	1,615	0,275	1	40,000	0,000	0,375	2	40,000	0,000
0,125	3	30,675	3,125	0,175	6	38,525	1,240	0,275	2	40,000	0,000	0,375	3	40,000	0,000
0,125	4	33,200	2,151	0,200	0	31,300	2,857	0,275	3	40,000	0,000	0,400	0	40,000	0,000
0,125	5	34,075	2,411	0,200	1	37,225	1,476	0,275	4	40,000	0,000	0,400	1	40,000	0,000
0,125	6	34,650	1,777	0,200	2	39,125	1,017	0,300	0	39,875	0,335	0,400	2	40,000	0,000
0,125	7	36,400	1,630	0,200	3	39,700	0,564	0,300	1	40,000	0,000	0,400	3	40,000	0,000
0,125	8	36,325	2,030	0,200	4	39,800	0,648	0,300	2	40,000	0,000				

**Escenario II**

**Disponibilidad de recursos  
en la red: MEDIA-BAJA**

INC = Cantidad máxima de incrementos permitidos.

Se han sombreado los casos en que no se ha utilizado la estrategia de exploración incremental

**Tabla A-29 Cantidad de éxitos obtenidos por prueba realizada en una red con disponibilidad MEDIA de recursos compartidos. Cada prueba consistió en la realización de 40 búsquedas independientes. Se realizaron 40 pruebas independientes registrando la cantidad de éxitos obtenidos en cada prueba para calcular la media y desvío estándar mostrados en la tabla.**

γ	INC	Éxitos		γ	INC	Éxitos		γ	INC	Éxitos		γ	INC	Éxitos	
		Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.
0,100	0	23,225	2,922	0,150	0	23,250	3,086	0,200	5	40,000	0,000	0,300	3	40,000	0,000
0,100	1	32,875	1,604	0,150	1	32,650	3,051	0,225	0	39,050	0,932	0,300	4	40,000	0,000
0,100	2	36,200	1,588	0,150	2	36,700	1,937	0,225	1	39,925	0,267	0,325	0	39,900	0,304
0,100	3	37,850	1,673	0,150	3	38,050	1,280	0,225	2	40,000	0,000	0,325	1	40,000	0,000
0,100	4	39,000	0,987	0,150	4	38,850	0,921	0,225	3	40,000	0,000	0,325	2	40,000	0,000
0,100	5	39,275	1,012	0,150	5	39,250	0,809	0,225	4	40,000	0,000	0,325	3	40,000	0,000
0,100	6	39,150	0,834	0,150	6	39,175	0,931	0,225	5	39,975	0,158	0,325	4	40,000	0,000
0,100	7	39,675	0,572	0,150	7	39,575	0,747	0,250	0	39,625	0,628	0,350	0	40,000	0,000
0,100	8	39,575	0,594	0,175	0	28,800	2,884	0,250	1	39,975	0,158	0,350	1	40,000	0,000
0,100	9	39,275	0,751	0,175	1	36,050	1,961	0,250	2	40,000	0,000	0,350	2	40,000	0,000
0,100	10	39,525	0,679	0,175	2	38,650	1,051	0,250	3	40,000	0,000	0,350	3	40,000	0,000
0,125	0	23,575	3,071	0,175	3	39,700	0,608	0,250	4	40,000	0,000	0,375	0	40,000	0,000
0,125	1	32,450	2,037	0,175	4	39,700	0,516	0,275	0	39,850	0,362	0,375	1	40,000	0,000
0,125	2	36,625	1,720	0,175	5	39,850	0,362	0,275	1	40,000	0,000	0,375	2	40,000	0,000
0,125	3	37,975	1,387	0,175	6	39,800	0,516	0,275	2	40,000	0,000	0,375	3	40,000	0,000
0,125	4	39,175	1,059	0,200	0	36,950	1,616	0,275	3	40,000	0,000	0,400	0	40,000	0,000
0,125	5	39,325	0,764	0,200	1	39,750	0,494	0,275	4	40,000	0,000	0,400	1	40,000	0,000
0,125	6	39,625	0,628	0,200	2	39,950	0,221	0,300	0	40,000	0,000	0,400	2	40,000	0,000
0,125	7	39,425	0,781	0,200	3	39,975	0,158	0,300	1	40,000	0,000	0,400	3	40,000	0,000
0,125	8	39,325	0,888	0,200	4	39,975	0,158	0,300	2	40,000	0,000				

**Escenario III**

**Disponibilidad de recursos en la red: MEDIA**

INC = Cantidad máxima de incrementos permitidos.

Se han sombreado los casos en que no se ha utilizado la estrategia de exploración incremental

**Tabla A-30 Cantidad de éxitos obtenidos por prueba realizada en una red con disponibilidad ALTA de recursos compartidos. Cada prueba consistió en la realización de 40 búsquedas independientes. Se realizaron 40 pruebas independientes registrando la cantidad de éxitos obtenidos en cada prueba para calcular la media y desvío estándar mostrados en la tabla.**

γ	INC	Éxitos		γ	INC	Éxitos		γ	INC	Éxitos		γ	INC	Éxitos	
		Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.			Media	Desv.Est.
0,100	0	30,400	3,053	0,150	0	30,250	2,529	0,200	5	40,000	0,000	0,300	3	40,000	0,000
0,100	1	36,875	1,713	0,150	1	37,000	1,783	0,225	0	39,700	0,648	0,300	4	40,000	0,000
0,100	2	39,025	0,800	0,150	2	39,150	0,864	0,225	1	39,975	0,158	0,325	0	39,975	0,158
0,100	3	39,775	0,480	0,150	3	39,675	0,656	0,225	2	40,000	0,000	0,325	1	40,000	0,000
0,100	4	39,775	0,423	0,150	4	39,750	0,439	0,225	3	40,000	0,000	0,325	2	40,000	0,000
0,100	5	39,775	0,480	0,150	5	39,900	0,304	0,225	4	40,000	0,000	0,325	3	40,000	0,000
0,100	6	39,875	0,335	0,150	6	39,875	0,404	0,225	5	40,000	0,000	0,325	4	40,000	0,000
0,100	7	39,975	0,158	0,150	7	39,925	0,267	0,250	0	39,875	0,335	0,350	0	40,000	0,000
0,100	8	39,975	0,158	0,175	0	34,125	2,040	0,250	1	40,000	0,000	0,350	1	40,000	0,000
0,100	9	39,950	0,221	0,175	1	38,900	0,871	0,250	2	40,000	0,000	0,350	2	40,000	0,000
0,100	10	39,875	0,335	0,175	2	39,825	0,385	0,250	3	40,000	0,000	0,350	3	40,000	0,000
0,125	0	29,550	2,819	0,175	3	39,925	0,267	0,250	4	40,000	0,000	0,375	0	40,000	0,000
0,125	1	37,150	1,642	0,175	4	39,975	0,158	0,275	0	39,775	0,480	0,375	1	40,000	0,000
0,125	2	39,000	0,934	0,175	5	39,975	0,158	0,275	1	40,000	0,000	0,375	2	40,000	0,000
0,125	3	39,600	0,672	0,175	6	40,000	0,000	0,275	2	40,000	0,000	0,375	3	40,000	0,000
0,125	4	39,950	0,221	0,200	0	39,000	0,816	0,275	3	40,000	0,000	0,400	0	40,000	0,000
0,125	5	39,825	0,446	0,200	1	39,925	0,267	0,275	4	40,000	0,000	0,400	1	40,000	0,000
0,125	6	39,900	0,304	0,200	2	40,000	0,000	0,300	0	39,975	0,158	0,400	2	40,000	0,000
0,125	7	40,000	0,000	0,200	3	40,000	0,000	0,300	1	40,000	0,000	0,400	3	40,000	0,000
0,125	8	39,950	0,221	0,200	4	40,000	0,000	0,300	2	40,000	0,000				

**Escenario IV**  
**Disponibilidad de recursos**  
**en la red: ALTA**

INC = Cantidad máxima de incrementos permitidos.  
 Se han sombreado los casos en que no se ha utilizado la estrategia de exploración incremental

