

**Distribución de contenidos en redes *ad hoc*
utilizando el paradigma P2P y
conocimiento de redes sociales**

**Idónea comunicación de resultados
que para obtener el grado de Maestro en Ciencias
(Ciencias y Tecnologías de la Información)**

presenta

Ing. Eric Márquez Solís

Asesores:

Dra. Elizabeth Pérez Cortés

Dr. Miguel López Guerrero

Sinodales:

Presidente: Dr. Ricardo Marcelín Jiménez

Secretario: Dr. Miguel López Guerrero

Vocal: Dr. Javier Gómez Castellanos



Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

**Distribución de contenidos en redes *ad hoc*
utilizando el paradigma P2P y
conocimiento de redes sociales**

Idónea comunicación de resultados
que para obtener el grado de Maestro en Ciencias
(Ciencias y Tecnologías de la Información)

presenta

Ing. Eric Márquez Solís

Asesores:

Dra. Elizabeth Pérez Cortés

Dr. Miguel López Guerrero

Sinodales:

Presidente: Dr. Ricardo Marcelín Jiménez

Secretario: Dr. Miguel López Guerrero

Vocal: Dr. Javier Gómez Castellanos

México, D.F. Enero de 2018

Agradecimientos

Agradezco a mis padres por haberme dado la vida y haberme guiado por el buen camino.

A mi esposa por su apoyo incondicional y por ser la compañera de mi vida.

A mis asesores de tesis por guiarme en este camino de la investigación y por su compromiso con el proyecto.

A mi Universidad (UAM) por darme la oportunidad de realizar estos estudios.

A Conacyt por la beca de apoyo que recibí en el periodo de la maestría.

Gracias a Dios por poner los medios para realizar este sueño.

Índice general

Lista de figuras	VII
1. Introducción	1
1.1. Objetivos	2
1.2. Metodología	3
1.3. Aporte del proyecto	3
1.4. Organización del documento	3
2. Marco conceptual	5
2.1. Redes <i>ad hoc</i>	5
2.2. Sistemas P2P: definición y propiedades	6
2.3. Protocolo BitTorrent	7
2.3.1. Mecanismo de selección de par (<i>choking/unchoking</i>)	9
2.3.2. Mecanismos de selección de pieza	9
2.4. Redes sociales	10
3. Trabajo relacionado	13
3.1. Trabajos que no consideran relación social	13
3.1.1. Nivel de cooperación	13
3.1.2. Proximidad	14
3.2. Trabajos que consideran la relación social	15
3.2.1. Datos importados de una red social	15
3.2.2. Datos generados a partir de intereses en común	16
3.2.3. Datos generados con base en intercambios de información	17
3.3. Taxonomía	18

4. ClubBitTorrent	19
4.1. Contexto	19
4.2. Propósito de ClubBitTorrent	20
4.3. Fundamentación	20
4.4. Propuesta ClubBitTorrent	22
5. Modelo de simulación	23
5.1. Elementos del modelo de simulación	23
5.2. Modelo de red social	25
5.3. Algoritmo para generar la ubicación de los usuarios	27
5.4. Algoritmo de ponderación	28
5.5. BitTorrent	28
5.6. Simulación de la red <i>ad hoc</i>	31
5.7. Integración de los componentes del modelo	32
6. Experimentación y análisis de resultados	35
6.1. Protocolos estudiados	35
6.2. Medidas de desempeño	37
6.3. Configuración de simulaciones	37
6.4. Resultados	37
6.4.1. Porcentaje de descarga	37
6.4.2. Tiempo promedio de descarga	44
6.4.3. Efecto de la proporción amigos-no amigos	46
7. Conclusiones y perspectivas	53
7.1. Conclusiones	53
7.2. Perspectivas	55
Lista de referencias	56
A. Tablas	60

Índice de figuras

2.1. Ejemplo de red <i>ad hoc</i> . Los círculos punteados representan los radios de transmisión de los dispositivos.	6
2.2. Funcionamiento de BitTorrent.	8
3.1. Taxonomía de trabajos relacionados.	18
4.1. Ejemplo de distribución de un grupo de amigos en un sitio arqueológico	20
4.2. Algoritmo de selección de par <i>choking/unchoking</i> de BitTorrent.	21
4.3. Ejemplo donde se observa que los nodos que se encuentran a un salto permiten alcanzar grupos de amigos de diferente tamaño (el círculo punteado indica el radio de alcance a un salto, los triángulos con líneas punteadas muestran los grupos de nodos).	22
5.1. Elementos del modelo de simulación	25
5.2. Distribución experimental de enlaces.	27
5.3. Ubicación de nodos.	30
5.4. Arquitectura EbitSim.	32
6.1. Comportamiento de desconexión de EbitSim en red <i>ad hoc</i>	36
6.2. Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 20 nodos	39
6.3. Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 40 nodos	40
6.4. Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 60 nodos	42

6.5. Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 80 nodos	43
6.6. Tiempo promedio de descarga global en escenarios de 20, 40, 60 y 80 nodos . . .	45
6.7. Porcentaje de ganancia en tiempo promedio de descarga del global de nodos <i>CBt_{4,0}</i> vs. BitTorrent <i>ad hoc</i>	45
6.8. Tiempo promedio de descarga de nodos amigos	46
6.9. Porcentaje de ganancia en tiempo promedio de descarga de nodos amigos <i>CBt_{4,0}</i> vs. BitTorrent <i>ad hoc</i>	47
6.10. Tiempo promedio de descarga de nodos no amigos	47
6.11. Porcentaje de ganancia en tiempo promedio de descarga de nodos no amigos <i>CBt_{4,0}</i> vs. BitTorrent <i>ad hoc</i>	48
6.12. Porcentaje de descarga para 40 nodos variando proporción de amigos	49
6.13. Ganancia en tiempo promedio de descarga para 40 nodos <i>Cbt_{4,0}</i> sobre BitTorrent	50
6.14. Tiempo promedio de descarga global 40 nodos variando proporción de amigos .	50
6.15. Tiempo promedio de descarga en amigos 40 nodos variando proporción de amigos	51
6.16. Tiempo promedio de descarga en no amigos 40 nodos variando proporción de amigos	52

Capítulo 1

Introducción

Análisis recientes del mercado de telefonía móvil revelan una gran penetración de los dispositivos de comunicaciones personales entre la población global. Basta decir que a finales del 2010 se vendieron 3 millones de teléfonos inteligentes, lo cual representó ese año el 19% del total de las ventas de dispositivos de comunicación. Se estima que en la actualidad más de 5.3 miles de millones de personas en el mundo usan teléfonos celulares.

Aunque los usuarios de dispositivos de comunicaciones móviles principalmente utilizan la infraestructura preinstalada de red (p. ej. puntos de acceso WiFi o radiobases celulares), algunas de sus necesidades de comunicación pueden satisfacerse a través de la creación de redes a la demanda. Este modo de operación permite el intercambio de información entre los mismos usuarios sin necesidad de infraestructura preexistente. A este tipo de red se le conoce como red inalámbrica *ad hoc* [1]. Las redes *ad hoc* pueden ser utilizadas en varias aplicaciones como distribución de información en aulas, comunicación en campos de batalla y actividades de apoyo en desastres naturales.

El uso de los dispositivos de comunicaciones personales se ha popularizado a la par que el uso de las redes sociales, las cuales han experimentado altos niveles de aceptación. Actualmente de cada 100 personas que tienen acceso a Internet, aproximadamente 70 utilizan alguna de las redes sociales existentes. Esto nos lleva a pensar en el gran potencial que tienen las redes sociales como medio de distribución de información.

El acelerado crecimiento de las redes sociales y su enorme potencial de aplicación explican el hecho de que se han convertido en un objeto de gran interés científico. Es así que en años recientes se han logrado avances significativos en el entendimiento de sus propiedades. Se puede mencionar, por ejemplo, que actualmente existe evidencia de que Facebook y Twitter

cumplen con ciertas propiedades que permiten catalogarlas como redes complejas (ver [2], [3], [4] y [5]).

En este trabajo nos centramos en el problema de la distribución de información en redes, y con esta motivación surge naturalmente la necesidad de considerar el exitoso paradigma par a par ($P2P$). Dentro de los sistemas $P2P$ más populares se puede mencionar a BitTorrent, al cual se le debe una porción significativa del tráfico que circula diariamente en la Internet [6]. En general, un sistema $P2P$ proporciona un ambiente descentralizado para la distribución de contenidos en donde todos los nodos tienen una doble función, pueden solicitar y compartir recursos [7], lo que lo convierte en el candidato ideal para el intercambio de contenido en redes *ad hoc*.

En este panorama, se detecta un nicho de oportunidad que resulta de unir la fortaleza del paradigma $P2P$ para la distribución de contenido y el conocimiento de los vínculos de amistad entre los miembros de una red *ad hoc* inalámbrica. Nuestra hipótesis es que si agregamos el conocimiento de la relación social, entre los individuos de un grupo, al protocolo BitTorrent, entonces es posible mejorar la experiencia en la descarga de contenido.

1.1. Objetivos

En este trabajo se plantea el siguiente objetivo general:

Diseñar un protocolo de descarga de contenido en redes *ad hoc* inalámbricas, utilizando conocimiento de redes sociales.

Y como objetivos específicos:

1. Identificar las variantes del protocolo de descarga BitTorrent para redes *ad hoc* inalámbricas.
2. Identificar las variantes del protocolo de descarga BitTorrent que incorporen conocimiento de redes sociales.
3. Identificar qué conocimiento del comportamiento de los individuos en redes sociales es de utilidad para la distribución de contenidos sobre una red *ad hoc* inalámbrica.
4. Diseñar, implementar y evaluar un protocolo de distribución de contenidos para redes *ad hoc* inalámbricas que incorpore el conocimiento de redes sociales.

1.2. Metodología

La metodología empleada para este trabajo de investigación es la siguiente:

1. Revisión y clasificación de la literatura sobre trabajos relacionados.
2. Elaboración de hipótesis y diseño de una estrategia de distribución de contenido.
3. Desarrollo de un protocolo con base en la estrategia diseñada.
4. Validación del protocolo mediante simulación.
5. Elaboración de conclusiones.
6. Comunicación idónea de resultados.

1.3. Aporte del proyecto

Como primer aporte de este proyecto, y resultado de la revisión de la literatura, se propone una taxonomía que clasifica los trabajos relacionados al intercambio de contenido. La clasificación se basa en el criterio utilizado para el establecimiento de los enlaces lógicos entre los participantes de una red par a par.

La idea principal del proyecto es aprovechar el conocimiento que nos proporciona una red social. Con base en esta idea, se busca beneficiar con descargas más rápidas y completas a un grupo de individuos relacionados socialmente que se encuentre ubicado en un espacio compartido con individuos ajenos al grupo. Entonces, la aportación central del proyecto es la definición de un protocolo que permita mejorar la experiencia de descarga de contenidos para un grupo de individuos que tienen una relación social. Este protocolo podría aplicarse en diferentes ámbitos como escuelas, museos, sitios turísticos, oficinas y en general en donde se requiera que la información se distribuya preferentemente a un subconjunto de individuos.

1.4. Organización del documento

En este trabajo se presenta en el capítulo dos el marco conceptual del proyecto. Posteriormente, en el capítulo tres, se analizan los trabajos relacionados y se presenta la taxonomía propuesta. Enseguida, en el capítulo cuatro, se define el protocolo ClubBitTorrent propuesto para mejorar la experiencia de descarga de contenidos para un grupo de individuos que tienen

relación social entre ellos. En el capítulo cinco se exponen los elementos de los que consta el modelo de simulación utilizado para la validación de la propuesta. Los experimentos realizados y los resultados se reportan en capítulo seis. Y por último, en el capítulo siete se exponen conclusiones y perspectivas.

Capítulo 2

Marco conceptual

En este capítulo se abordan los conceptos necesarios para la comprensión de este trabajo de investigación.

2.1. Redes *ad hoc*

Una red *ad hoc* inalámbrica es un conjunto de nodos inalámbricos que se autoconfiguran para formar una red sin la ayuda de una infraestructura establecida [8]. En la figura 2.1 se ilustra una red *ad hoc* inalámbrica y se observa que sus nodos establecen enlaces entre ellos a través de comunicaciones inalámbricas. Al no depender de una infraestructura, los nodos participantes ejecutan las tareas de administración y control de la red, generalmente a través de algoritmos de control distribuidos. Cuando el nodo destino no se encuentra próximo al nodo origen, las redes *ad hoc* utilizan rutas multisalto, estas rutas se forman de la trayectoria que siguen los mensajes al utilizar a los nodos intermedios que retransmiten paquetes hacia el nodo de destino final. Por esta razón cada nodo tiene dos roles dentro de la red: encaminador y *host*.

Las principales ventajas de una red *ad hoc* son:

- Evitan costos de instalación y mantenimiento de la infraestructura de red.
- Pueden ser implementadas y reconfiguradas rápidamente.
- Muestran gran robustez por su naturaleza distribuida.



Figura 2.1: Ejemplo de red *ad hoc*. Los círculos punteados representan los radios de transmisión de los dispositivos.

2.2. Sistemas P2P: definición y propiedades

Los sistemas par-a-par (*peer-to-peer*, *P2P*) se basan en un modelo de intercambio de recursos en donde los nodos que conforman el sistema son autónomos y cada uno de ellos puede asumir el rol de cliente o de servidor. Por esta característica se les denomina pares a los nodos que forman el sistema. En consecuencia este tipo de sistemas posee una gran escalabilidad ya que la capacidad de distribución de contenido se incrementa al aumentar el número de pares conectados. Gracias a ello, los sistemas *P2P* han llegado a ser muy populares en aplicaciones de Internet. A continuación se mencionan las características más relevantes de un sistema *P2P* [7]:

- Descentralización. Los recursos están distribuidos entre todos los participantes del sistema.
- Escalabilidad. Al incrementar el número de pares, se incrementa la cantidad de recursos y por lo tanto el sistema aumenta su capacidad.
- Mantenimiento dinámico. El sistema se autoconfigura, las máquinas y recursos son agregados o eliminados sin intervención manual o supervisión.
- Tolerancia a fallas. Aunque alguna parte del sistema falle, los recursos disponibles de las máquinas sobrevivientes continúan siendo accesibles.

- Autoestabilización. El sistema tiene un mecanismo de reparación de fallas, ya que detecta inconsistencias locales en un enlace, como la desconexión de un par o fallas en la comunicación.
- Búsqueda eficiente. Uno de los objetivos de un sistema *P2P* es localizar recursos eficientemente. Por ello, soporta un lenguaje de consulta para hacer búsquedas complejas tales como búsquedas en rangos, por coincidencia más cercana a una clave y por coincidencias con palabras clave.
- Seguridad. El sistema *P2P* tiene seguridad ante ataques como el de negación de servicio donde participantes malintencionados pueden inundar el sistema con mensajes sin sentido o erróneos, entorpeciendo así el tráfico legítimo.
- Construcción susceptible a cambios en la topología. El encaminamiento de datos puede adaptarse a factores cambiantes en la red local tales como la distancia viajada o latencia a lo largo del camino de transmisión.

BitTorrent es un protocolo para intercambio de recursos basado en el modelo *P2P* muy popular en Internet y por esa razón se seleccionó como punto de partida de este proyecto de investigación. En la siguiente sección se expondrá el protocolo BitTorrent, donde se proporcionan detalles de su operación, elementos que lo componen y sus estrategias de distribución de contenido.

2.3. Protocolo BitTorrent

BitTorrent es un protocolo de distribución de contenido bajo el paradigma *P2P* basado en una estrategia de cooperación conocida como *tit-for-tat* (ojo por ojo), la cual consiste en atender primero a los pares más rápidos para la descarga de contenido y que hayan compartido recursos recientemente [9].

En BitTorrent el contenido se divide en piezas. Cada par que forma parte de la red tiene dos estados: el estado de descargador (*leecher*), que es cuando está descargando contenido y aún no tiene todas las piezas y el estado semilla (*seed*), que se alcanza cuando ya tiene todas las piezas. En el momento en que un par descargador ya obtuvo una pieza, la puede compartir con otros pares. Los pares que cooperan para descargar el mismo contenido forman un torrente [1].

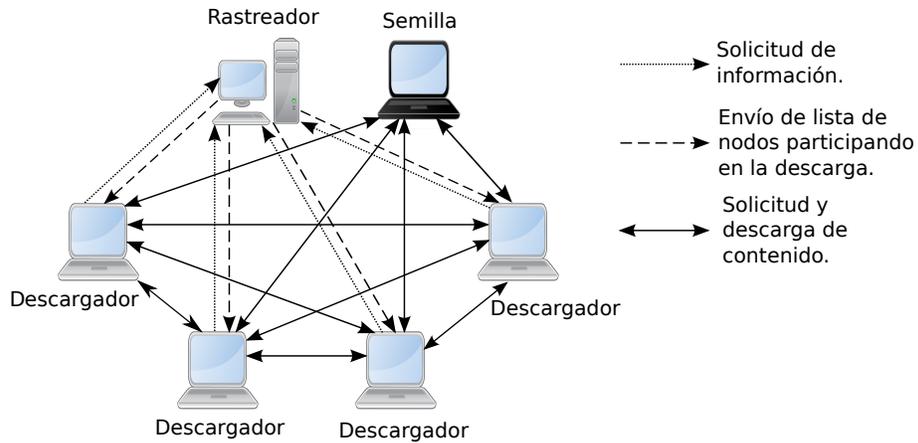


Figura 2.2: Funcionamiento de BitTorrent.

BitTorrent se encarga de la transferencia de contenido desde una fuente inicial a potencialmente un gran número de destinos. Se asume que el contenido está confinado en la red, es decir, que los pares son nodos de la red, como se muestra en la figura 2.2.

Cuando un par semilla desea compartir algún contenido, se crea un archivo *.torrent*, el cual se aloja en un servidor *web*. Esta acción es básicamente un anuncio de que el contenido se encuentra disponible. El archivo *.torrent* cuenta con la información del contenido a descargar, su longitud, nombre y el *URL* de un rastreador (*tracker*). Los rastreadores son responsables de ayudar a los descargadores a encontrar a los pares semilla.

Si un nodo descargador requiere algún contenido, entonces aquél localiza y consulta el archivo *.torrent* para obtener información acerca de dicho contenido y del nodo rastreador correspondiente.

Una vez que el nodo descargador se comunica con el rastreador, recibe una lista de pares que participan en la descarga del contenido. Posteriormente el descargador se comunica con estos pares para tratar de obtener piezas.

Los mecanismos centrales para la distribución del contenido en BitTorrent son: el usado por un par para seleccionar a los pares a los cuales les va a enviar piezas, conocido como mecanismo de selección de par; y el usado por un par para decidir qué pieza solicitar, llamado mecanismo de selección de pieza. Ambos mecanismos se describen a continuación.

2.3.1. Mecanismo de selección de par (*choking/unchoking*)

El ancho de banda es un recurso limitado, por esta razón un nodo no puede enviar piezas a todos los pares interesados en sus piezas de contenido al mismo tiempo. El algoritmo *choking/unchoking*¹ de BitTorrent selecciona un número fijo de pares para enviarles piezas (se dice que los desahoga), el número de pares por defecto es cuatro; el resto de los pares es bloqueado (ahogado). Cada 20 segundos, en promedio, se eligen pares para desahogar. La elección de tres de los cuatro pares está basada estrictamente en la velocidad de descarga que tengan los pares solicitantes. Como los pares más lentos quedan en desventaja, el cuarto par se elige de manera aleatoria. A esta selección aleatoria se le conoce como desahogo optimista (*optimistic unchoke*) [9], [10].

2.3.2. Mecanismos de selección de pieza

En BitTorrent un contenido se divide en piezas y estas a su vez en subpiezas. Un par envía subpiezas a los pares que desahoga, en el momento en que un par termina de descargar una pieza, esta la puede compartir con otros pares.

El orden en el que se descargan las piezas es importante para el buen desempeño del sistema [9]. Los mecanismos de BitTorrent para selección de la pieza a descargar son:

- La más rara primero. Los pares generalmente descargan primero piezas de las cuales existen pocas copias en sus pares vecinos.
- Pieza aleatoria primero. Cuando la transferencia de la pieza más rara es lenta, se selecciona de forma aleatoria una pieza cuya transferencia sea rápida y posteriormente la pieza más rara.
- Modo de fin de juego. Algunas veces una pieza será solicitada de un par con baja velocidad de transferencia. Esto no es problema si se presenta a la mitad de una descarga, porque al mismo tiempo se pueden estar descargando otras piezas; pero si se presenta al final, podría retrasar la terminación de la descarga. Cuando un par «A» solicita una pieza a sus pares vecinos, probablemente más de uno se la envíe, es por eso que en el momento de que el par «A» termina de descargar la pieza, envía una señal de cancelación

¹En la descripción original de BitTorrent este algoritmo se conoce como *choking*, por claridad aquí nos referimos a él como algoritmo *choking/unchoking*.

de solicitud de la pieza a los otros vecinos. De esta manera se evita ocupar el ancho de banda en envíos redundantes.

2.4. Redes sociales

El estudio de algunas redes sociales ha podido identificar en estas algunas propiedades asociadas al concepto de redes complejas [11]. Una red compleja está formada por un gran número de entidades o componentes que interactúan sin la intervención de un control central, para lograr su comunicación se basan en un conjunto de reglas de operación. Este tipo de redes ha sido modelado con la teoría de grafos, la red es descrita por un grafo en donde los vértices son los usuarios y las aristas son los enlaces. Así, si dos usuarios tienen interacción, entonces existe un enlace entre ellos.

Algunos hallazgos relevantes relacionados a este tema se describen a continuación. En 1967 Milgram encontró que el diámetro de una red social puede ser muy pequeño comparado con el grafo subyacente, es decir, que existe un camino corto entre dos personas cualesquiera. Esta característica fue llamada fenómeno de mundo pequeño (*small-world*) [4]. En [12] Erdős *et al.* proponen un método para generar redes aleatorias denominado modelo Erdős-Renyi (ER). Este modelo indica que un nuevo usuario se puede enlazar con la misma probabilidad a cualquier otro usuario existente en la red. La cantidad de enlaces que pueden tener los vértices en una red aleatoria sigue una distribución binomial que tiende a una distribución de Poisson cuando la red es muy grande. Posteriormente en 1998, Watts y Strogatz [13] desarrollaron un modelo para crear redes cuyas características principales son: primero tener un diámetro pequeño del orden $O(\log(n))$, segundo, un alto coeficiente de agrupamiento $\frac{3(k-2)}{4(k-1)}$ donde k es el número de enlaces que tiene cualquier vértice y finalmente una distribución exponencial.

Las redes complejas también pueden ser descritas con el modelo libre de escala. A diferencia del modelo ER, la probabilidad de enlace de un nuevo usuario a los demás sigue una ley de potencias. Se considera a una red libre de escala [5] si se comporta de acuerdo a las siguientes características:

- La red está en constante crecimiento por la adición de nuevos vértices.
- Los nuevos vértices se enlazan preferentemente a los vértices existentes que cuentan con mayor cantidad de enlaces.

Este alto grado de autoorganización es una propiedad característica de las redes complejas libres escala.

Explorando las bases de datos que describen la topología de las grandes redes en campos tan diversos tales como los establecidos por los hipervínculos entre los documentos residentes en la *WWW* o los patrones de referencias en publicaciones científicas, se encontró que independientemente del sistema y de la identidad de sus constituyentes, la probabilidad $P(k)$ de que un vértice en la red interactúe con otros k vértices decae de acuerdo a una ley de potencias, dada por $P(k) \sim k^{-\gamma}$ donde $2 < \gamma \leq 4$ [5]. Este resultado indica que las grandes redes se autoorganizan en un estado «libre de escala». Para confirmar esta tendencia se observaron algunas redes, como por ejemplo:

- La red de colaboración de actores en las mismas películas. En el grafo los actores son los vértices y la colaboración en una misma película son las aristas, se obtuvo $\gamma = 2.3 \pm 0.1$.
- La *WWW* con $\gamma = 2.1 \pm 0.1$, en este caso las páginas *HTML* son los vértices y las ligas que las conectan son las aristas.
- La red de energía eléctrica del oeste de los Estados Unidos con $\gamma = 4$, donde el conjunto de transformadores y subestaciones son los vértices y las líneas cableadas que las comunican son las aristas.
- La red formada por las referencias bibliográficas entre artículos científicos. En esta red los vértices son los artículos y las aristas son las referencias entre ellos. En este caso $\gamma = 4$.

Por lo anterior se observa que muchas redes aleatorias de gran tamaño comparten la característica de que la distribución de su conectividad local es libre de escala.

Las redes sociales informáticas son comunidades virtuales en las que sus integrantes interactúan entre sí a través de Internet. En las redes sociales se permite a los usuarios registrar y crear su propia página con información personal, con el objetivo de establecer conexiones públicas de amistad con otros miembros. La comunicación dentro de la red social se puede dar de varias formas, por ejemplo a través de correo electrónico privado, mensaje públicos, blogs, imágenes o mensajería instantánea. Un ejemplo de red social informática es *Facebook*, siendo esta una de las redes sociales más populares en la actualidad.

Los miembros de una red social pueden dividirse en grupos o comunidades. La estructura de los grupos nos da una idea del grado de correlación entre los usuarios dado por intereses en

común [14]. Esta característica nos muestra patrones de comunicación, los cuales se basan en la dinámica de difusión de información y la interacción de procesos, ayudando a comprender mejor los sistemas sociales complejos como las sociedades humanas [15] [16].

Capítulo 3

Trabajo relacionado

En este capítulo se reportan los resultados de la investigación documental realizada en este proyecto de investigación, en donde:

- Se adapta BitTorrent para operar en redes inalámbricas.
- Se tienen variantes de BitTorrent que aprovechan el conocimiento de redes sociales.

El foco de la investigación fue el criterio utilizado en cada caso para establecer el enlace lógico entre nodos. Con base en este criterio, los resultados se clasifican en dos categorías: los trabajos que no consideran la relación social y los trabajos que sí la consideran. Al final se presenta una taxonomía que resume esta clasificación.

3.1. Trabajos que no consideran relación social

En la primera categoría, la estrategia para establecer el enlace lógico no considera la relación social entre los nodos. Esta categoría a su vez se puede clasificar en las subcategorías descritas en las siguientes subsecciones, según el criterio utilizado para establecer el enlace.

3.1.1. Nivel de cooperación

En el primer criterio se establecen los enlaces con base en la estrategia *tit-for-tat* (ojo por ojo), la cual consiste en establecer enlaces con los tres nodos de mayor velocidad de descarga y aporte de contenidos. Dentro de esta clasificación se encuentra el proyecto BitTorrent [9].

3.1.2. Proximidad

El criterio de proximidad surge de una propuesta en donde se adapta el sistema BitTorrent de un ambiente Internet cableado a uno de red inalámbrica *ad hoc* móvil o *MANET* (*mobile ad hoc network*). En un ambiente inalámbrico la señal de transmisión se va atenuando al recorrer mayor distancia, por lo que es posible que se pierda la información enviada a través de este medio. Para disminuir este problema existen los protocolos de encaminamiento que crean rutas multisalto entre dos nodos permitiendo así la posibilidad de que cada nodo pueda retransmitir la información hasta alcanzar al nodo destino.

El criterio de proximidad consiste en establecer enlaces con nodos que se encuentren a lo máximo a dos retransmisiones, es decir, los destinos deben de encontrarse próximos al origen.

En el año 2008 Sbai *et al.* [1] presentaron una propuesta para adaptar BitTorrent de un ambiente Internet a uno *MANET*. En este trabajo el conjunto de nodos que se comunican entre sí a dos retransmisiones es llamado vecindario. Se encontró que cuando se limita el alcance del vecindario a dos retransmisiones mejora la velocidad de transferencia; pero disminuye la cantidad de piezas disponibles en el vecindario (diversidad). Por lo que propusieron una estrategia llamada «la pieza ausente». Esta estrategia consiste en ir a buscar fuera del vecindario (área de diversificación) aquellas piezas que no se encuentren en el vecindario. La estrategia de la pieza ausente se implementa cambiando la selección del cuarto nodo en el algoritmo de *choking/unchoking* de BitTorrent. En BitTorrent tradicional el cuarto nodo se elige aleatoriamente, en la estrategia propuesta el cuarto nodo se elige en el área de diversificación, en un intento por buscar piezas ausentes.

Para la validación del protocolo propuesto se consideró un escenario en donde los nodos eran estáticos en una red inalámbrica. Como resultado encontraron que limitar el vecindario mejora la velocidad de transferencia y que la estrategia de la pieza ausente mejora la diversidad de las piezas.

Posteriormente en el año 2009, el mismo equipo de investigadores [17] estudió el escenario en el que los nodos se mueven de acuerdo al modelo *random waypoint*, conservando la estrategia de generación de enlace lógico. Los resultados indican que aun con la variante de movilidad, limitar el número de retransmisiones continua mejorando el tiempo de descarga. Con base en estos estudios se implementó la aplicación BitHoc [18] [19].

BitHoc es una variante de BitTorrent adaptada al entorno de redes *MANET*. Consta de dos componentes, un servidor de gestión de miembros (*BitHoc tracker*) y un servidor de

intercambio de contenido (*BitHoc client*). Para eliminar el nodo rastreador, propusieron un protocolo de «Administración de la membresía». Observaron que la administración de la membresía es una tarea costosa y pesada para las *MANET*. Por lo que el protocolo propuesto se basa en construir un árbol de amplitud mínima, en términos del número de saltos, para garantizar costos mínimos. Con esto se permite a los pares descubrirse unos a otros y mantenerse informados de las llegadas y salidas de los pares.

En ambiente *MANET*, el desempeño de BitHoc supera la versión clásica de BitTorrent en términos de tiempo de descarga.

3.2. Trabajos que consideran la relación social

En la segunda categoría, la estrategia para generar enlaces lógicos considera la relación social entre los nodos. Tomando en cuenta el origen de los datos que indican la relación social, esta categoría se puede dividir en las subcategorías descritas a continuación.

3.2.1. Datos importados de una red social

En la primera subcategoría se pueden ubicar los trabajos que importan datos de una red social existente.

Maarten Fonville [20] propone mezclar las fortalezas de BitTorrent y de *Facebook*. Utiliza BitTorrent para compartir grandes cantidades de datos en un ambiente distribuido y *Facebook* para el control de acceso a datos confidenciales. Para unir las dos plataformas utilizaron como interfaz al sistema *OneSwarm* [21]. Este sistema almacena las identidades de usuarios. Cada usuario tiene una lista de usuarios confiables que es llamada «comunidad». *OneSwarm* crea los enlaces lógicos para envío de contenido a los integrantes de la comunidad. Las identidades y su relación social (lista de usuarios confiables) se importan de *Facebook*. Como resultado de este trabajo se logró el intercambio *P2P* de contenido con el control de acceso de *Facebook* en una red cableada.

Xiao Han *et al.* [22] proponen utilizar la información social para el descubrimiento de contenido en una red *P2P*. El algoritmo propuesto pondera a los nodos de un grupo de amigos con base en dos tipos de atributos sociales: conocimiento y similitud. El algoritmo elige desahogar a los amigos con mayor cantidad de contenidos almacenados y con mayor similitud de intereses. Las listas de amigos y de intereses se descargan de *Facebook*. Para evaluar la

propuesta se capturó información social real de 384,494 usuarios de *Facebook*. Los resultados demostraron que este método mejora el descubrimiento de contenido. Esta propuesta se aplicó en un ambiente cableado.

SocialP2P [23] agrupa nodos con intereses múltiples en común y fomenta la conexión con nodos socialmente cercanos en un grupo. Los nodos se agrupan con base en sus intereses múltiples en común. Dentro de cada grupo los nodos están conectados con base en sus enlaces sociales. Se seleccionan a los nodos con más permanencia en la red como embajadores de su propio grupo en una DHT (tabla *hash* distribuida) [24] para un intercambio de archivos eficiente entre grupos. Este sistema aumenta la eficiencia y fiabilidad en el intercambio de archivos. El algoritmo utiliza BitTorrent e importa datos de *Facebook*.

SPROUT [25] [26] es un algoritmo de encaminamiento DHT (tabla *hash* distribuida) que aumenta significativamente la probabilidad de éxito de encaminamiento mediante el uso de enlaces sociales. El algoritmo forma dos anillos, el primero con una DHT tradicional, y otro con una DHT basada en la amistad. Cuando un nodo requiere contenido, primero hace la búsqueda en el anillo por sus amigos, si no encuentra el contenido en un amigo cercano entonces utiliza el otro anillo para establecer el enlace con un nodo no amigo. Para implementar el algoritmo se utilizaron dos fuentes de datos: un sitio web de la comunidad del Club Nexus de la universidad de Stanford y una red social sintética basada en un algoritmo de mundos pequeños. La propuesta fue aplicada en un ambiente cableado. El algoritmo de encaminamiento SPROUT logra disminuir las fallas de transmisión de mensajes de un par a otro, comparado con el encaminamiento tradicional.

3.2.2. Datos generados a partir de intereses en común

En la segunda subcategoría los datos se generan con base en intereses comunes determinados de acuerdo con el contenido almacenado. Dos ejemplos de sistemas de este tipo se describen a continuación.

SPOON (*Social network-based P2P cOntent-based file sharing in disconnected mObile ad hoc Networks*) [27] es un sistema de red social basado en el intercambio de contenido *P2P* en redes móviles *ad hoc*. Consta de cuatro componentes:

- Algoritmo de extracción de interés. Un nodo infiere sus intereses de palabras clave de cada uno de sus archivos utilizando la técnica de agrupamiento de documentos. Si un grupo de palabras aparece frecuentemente en distintos archivos, entonces estos

pertenecen a una misma categoría, la cual puede ser utilizada para indicar el interés del nodo.

- Algoritmo de construcción de comunidad. Un nodo genera un vector de acuerdo a las categorías de cada uno de sus archivos. A través de un criterio de similitud se calcula qué tan similares son los vectores entre dos nodos, y si el valor de la evaluación excede un umbral, entonces los nodos involucrados forman parte de la misma comunidad.
- Algoritmo de asignación de rol. Para aprovechar la movilidad se eligen los nodos coordinador y embajador. El nodo coordinador es un nodo que permanece en la red y tiene el mayor número de contactos. El embajador es el que tiene mayor número de contactos en otras comunidades.
- Algoritmo de encaminamiento orientado a interés IRA (*interest oriented routing algorithm*). El nodo destino está representado por un vector, el algoritmo de encaminamiento IRA evalúa y elige el nodo destino de menor costo.

SPOON se aplica en *MANET* y toma ventaja de la movilidad declarando sus nodos embajadores, gracias a lo cual logra mejorar la tasa de éxito en intercambio de contenidos.

NetTube [28] es un modelo que explora las redes sociales de videos, los pares vuelven a distribuir los videos que ya tienen almacenados. Cuando un par nuevo se une al sistema NetTube, solicita su primer video al servidor. El servidor tiene un registro del grupo de nodos que contienen ese video, entonces redirecciona el nuevo par al grupo correspondiente enviándole una lista de proveedores potenciales; cuando el cliente termina de ver su video y empieza a descargar otro video, entonces se une a otro grupo, pero sigue perteneciendo al primero, de esta forma se va formando la red social. NetTube mejora el aprovechamiento de la red, en particular para compartir videos cortos.

3.2.3. Datos generados con base en intercambios de información

Por último se describen los trabajos donde los enlaces se generan con base en los intercambios reales de información.

PROSA [29] (por sus siglas en inglés *P2P Resource Organization by Social Acquaintances*) trata de imitar la forma en la que se establecen vínculos sociales. En PROSA se tienen tres tipos de enlace:

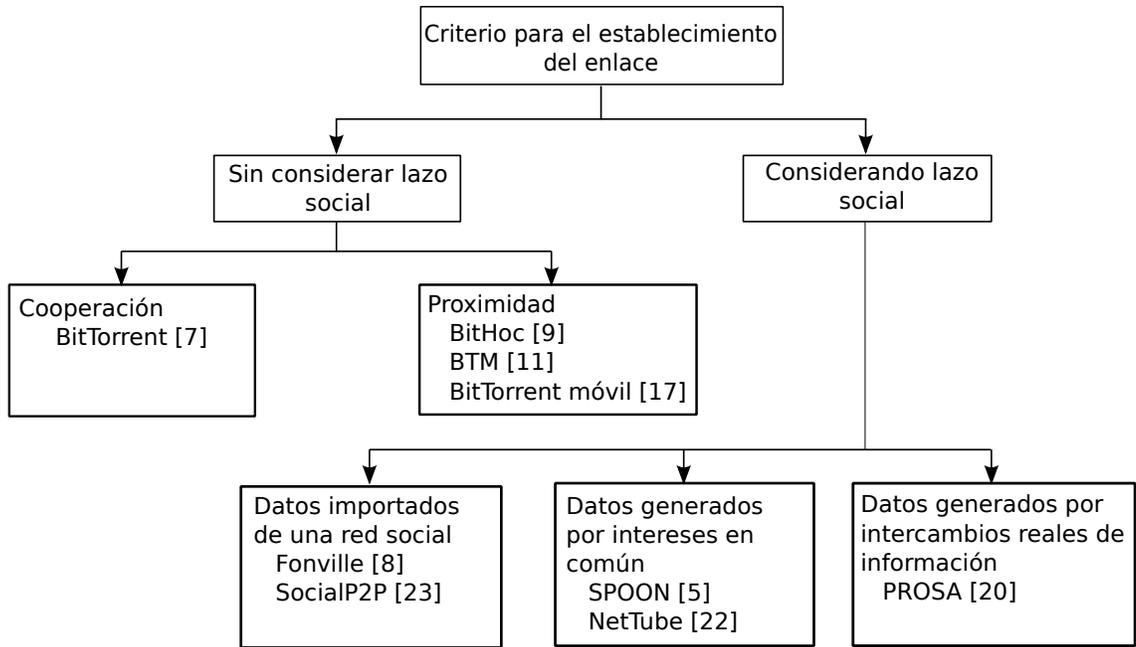


Figura 3.1: Taxonomía de trabajos relacionados.

- Enlace a un conocido AL (*Acquaintance Link*). Este es el enlace más débil, se presenta entre dos nodos que saben de su mutua existencia; pero que nunca han tenido interacción.
- Enlace semántico temporal TSL (*Temporary Semantic Link*). Cuando un nodo envía solicitudes a otro; pero el segundo no contiene lo solicitado, se considera que su enlace es semántico, su ponderación es intermedia.
- Enlace semántico completo FSL (*Full Semantic Link*). Este es el enlace más fuerte porque se presenta en nodos que anteriormente intercambiaron información.

PROSA disminuye el tráfico en la red gracias a que los enlaces se establecen como consecuencia de envío y respuesta de consultas.

3.3. Taxonomía

Como resultado de la clasificación de los trabajos se propone la taxonomía que se muestra en la figura 3.1 en donde se resume lo presentado en el presente capítulo.

Capítulo 4

ClubBitTorrent

En este capítulo se describe el protocolo ClubBitTorrent. Se denominó así porque este protocolo da preferencia en la descarga de contenido a los miembros de un grupo de amigos. Se expone el contexto en el cual se desarrolla, su propósito, su fundamento y por último las modificaciones a BitTorrent para dar origen al protocolo propuesto.

4.1. Contexto

Definimos como un grupo de amigos a un conjunto de individuos que tienen relaciones sociales entre ellos. Por ejemplo: los miembros de un equipo deportivo, los excompañeros de la secundaria o los integrantes de una banda de música.

En un espacio físico los integrantes de un grupo de amigos pueden estar cercanos entre sí o aislados, es decir, en la vida real los amigos no siempre se encuentran juntos. Dependiendo de la ocasión algunos amigos pueden encontrarse alejados del grupo.

Dentro del espacio en donde se encuentre el grupo de amigos frecuentemente existen individuos ajenos al grupo. Suponga, por ejemplo, que un grupo de amigos va de paseo a una zona arqueológica. En la figura 4.1 se ejemplifica esta situación, en donde se puede observar en los círculos amarillos a los individuos que pertenecen al grupo de amigos y en los círculos rojos a los que son ajenos al grupo.



Figura 4.1: Ejemplo de distribución de un grupo de amigos en un sitio arqueológico

4.2. Propósito de ClubBitTorrent

Cuando un integrante de un grupo de amigos quiera compartir información de interés para todos los individuos que se encuentran en un espacio, podría ser deseable que los individuos que pertenezcan al grupo de amigos realicen la descarga del contenido en menor tiempo con respecto a los individuos no amigos. Enfatizamos que se asume que la información es de interés para todos los individuos presentes, sean amigos o no amigos.

ClubBitTorrent tiene como propósito mejorar el tiempo promedio de descarga de los nodos amigos con respecto de los no amigos y mejorar el número de descargas completas en los nodos amigos.

4.3. Fundamentación

En el protocolo BitTorrent, cuando un par comparte una pieza, este atiende las peticiones de los otros pares utilizando el mecanismo *choking/unchoking*. El mecanismo envía piezas a cuatro nodos seleccionados cada cierto tiempo con base en los siguientes criterios. Tres nodos se eligen tomando los de mayor velocidad de descarga y un cuarto nodo se elige de forma aleatoria, como se muestra en la figura 4.2.

Para implementar nuestra propuesta es necesario cambiar el criterio de selección de pares

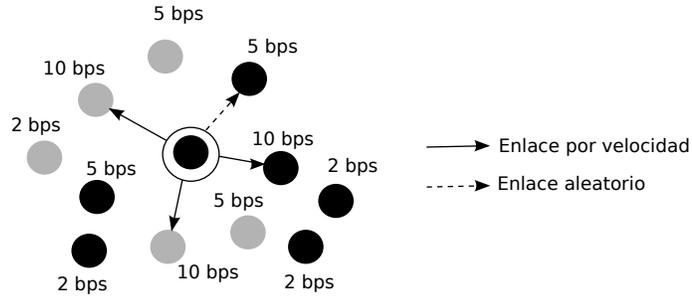


Figura 4.2: Algoritmo de selección de par *choking/unchoking* de BitTorrent.

a desahogar. La propuesta consiste en dar prioridad a los nodos que puedan distribuir el contenido a más nodos amigos. En el ejemplo de la figura 4.3 este criterio llevaría a desahogar a los nodos *a*, *b*, *c* y *d* para enviarles piezas. Dentro de los triángulos punteados se puede observar la cantidad de nodos amigos que recibirían piezas, es decir, al seleccionar:

- El nodo *a*, a pesar de no ser amigo, una vez que tenga alguna pieza la puede transmitir a tres nodos amigos.
- El nodo *b* es amigo y puede enviar piezas a uno más, por esto se considera que recibirían piezas dos amigos.
- El nodo *c* pertenece al grupo de amigos por lo que un amigo recibiría piezas.
- El nodo *d*, cinco amigos recibirían piezas.

En el ejemplo anterior observamos que los nodos *a*, *b*, *c* y *d*, pueden distribuir piezas a distintos números de amigos. Para tomar este factor en cuenta en ClubBitTorrent a cada uno de estos nodos se les asigna un peso (ponderación) en relación directa al número de nodos amigos que permite alcanzar. Es decir, si p_i es el peso del *i*-ésimo nodo, entonces $p_a = 3$, $p_b = 2$, $p_c = 1$ y $p_d = 5$. El valor de la ponderación p del nodo *i* es la cantidad de amigos a los que el nodo *i* pueda enviar piezas, si el nodo *i* pertenece al grupo de amigos se agrega a sí mismo a la suma de la ponderación.

El concepto de ponderación p para un nodo *i* puede formalizarse considerando que si se define el subconjunto de vértices $T_i = \{v | v \in \text{amigos} \ \& \ v \in \text{subárbol con raíz en } i\}$, entonces $p_i = |T_i|$.

En el escenario de la figura 4.1 se conoce quiénes son los integrantes del grupo de amigos y quiénes son los individuos ajenos al grupo en un área determinada. Con base en este tipo

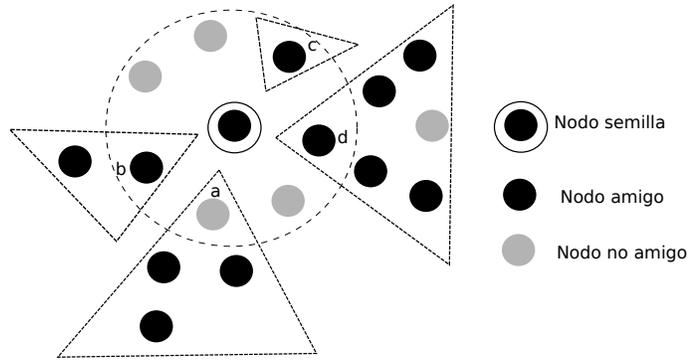


Figura 4.3: Ejemplo donde se observa que los nodos que se encuentran a un salto permiten alcanzar grupos de amigos de diferente tamaño (el círculo punteado indica el radio de alcance a un salto, los triángulos con líneas punteadas muestran los grupos de nodos).

de escenario se diseñó e implementó un algoritmo (sección 5.3) para calcular los valores de ponderación de cada nodo. Este algoritmo recibe como entrada la ubicación e identificación de los nodos amigos, la identidad del nodo semilla y el número total de nodos, y a partir de estos datos se calcula la ponderación de cada nodo.

4.4. Propuesta ClubBitTorrent

Con el objetivo de cambiar el mecanismo de selección de par *choking/unchoking* de BitTorrent tradicional, es necesario incorporar el conocimiento de la relación social entre los miembros del grupo y el valor de la ponderación a cada nodo.

Se propone modificar el algoritmo de *choking/unchoking* teniendo como parámetros el número de nodos a desahogar M , de los cuales se pueden desahogar p nodos por ponderación y $M-p$ nodos por velocidad, dando origen al protocolo ClubBitTorrent denotado como $CBt_{p,M-p}$. Los componentes de la propuesta se detallan en el siguiente capítulo.

Capítulo 5

Modelo de simulación

Existen varias formas de hacer la validación de los algoritmos propuestos, por ejemplo: realizar pruebas en un prototipo, usar simulación por computadora o modelarlos matemáticamente. De las opciones anteriores se seleccionó la simulación por computadora, consideramos que es la manera más adecuada en este caso para generar el ambiente de prueba y hacer las mediciones de desempeño.

La simulación por computadora tiene la ventaja de que solo requiere una computadora, a diferencia de un prototipo en donde se necesitarían varios equipos para intercambiar información entre ellos. Otra ventaja que ofrece la simulación es que los ajustes de los parámetros pueden hacerse rápidamente, lo que tomaría más tiempo en un ambiente real por el número de equipos. Por otro lado modelar el sistema matemáticamente sería poco práctico por su complejidad.

Se decidió utilizar un modelo de simulación porque, además de las ventajas ya mencionadas, nos da la facilidad de incorporar los protocolos de las diferentes capas de la arquitectura de red *TCP/IP*, así como el protocolo BitTorrent con todas sus funcionalidades. Con este nivel de detalle en el modelo, los resultados pueden ser muy cercanos a la realidad. Para este propósito se seleccionó el simulador OMNeT++ versión 4.2.2. [30].

5.1. Elementos del modelo de simulación

Tomando como base el ejemplo de la figura 4.1, en donde un grupo de amigos va de paseo a una zona arqueológica, pretendemos modelar una situación en donde un grupo de amigos llega a un sitio y ellos se intercalan entre otros individuos que se encuentran en el lugar.

Algunos individuos del grupo se encontrarán ubicados cerca de los demás amigos y otros más alejados pero dentro del mismo sitio. En un momento dado, uno de los individuos del grupo decide enviar contenido desde su celular a sus amigos presentes en el lugar vía *WiFi* operando en modo *ad hoc*, todos los individuos portan un equipo de comunicaciones inalámbricas con esta capacidad de conexión y cuentan con las aplicaciones para intercambio de contenido.

En consecuencia, el modelo de simulación está compuesto de los siguientes elementos:

1. Modelo de red social. Para establecer las relaciones sociales de los individuos en la simulación es necesario contar con un modelo que permita generar tales vínculos. Usaremos una red sintética basada en el modelo de red libre de escala [31].
2. Algoritmo para generar la ubicación de los usuarios. La idea principal es establecer la ubicación de los usuarios con base en las relaciones de amistad entre ellos. Por lo que se diseñó el algoritmo 2, para que teniendo como entrada las relaciones sociales de los nodos, genere sus ubicaciones.
3. Algoritmo de ponderación. Se propone un algoritmo que con base en la ubicación de los nodos y la relación social entre ellos, contabiliza el número de nodos amigos a los que se les puede enviar datos a través de él, dato que denominamos ponderación.
4. BitTorrent. Se utilizará el simulador de BitTorrent EbitSim [32], el cual fue diseñado con base en los principales clientes BitTorrent.
5. Red *ad hoc* inalámbrica. Esta se tomará de las bibliotecas del simulador OMNeT++ y sirve como medio de transporte de la aplicación EbitSim.

Se integrarán los componentes anteriores en una sola aplicación que nos proporcionará los datos del comportamiento de la distribución del contenido en los nodos.

En la figura 5.1 se muestran los elementos del modelo de simulación. En el modelo la parte en gris claro contiene los elementos necesarios para la simulación tanto de BitTorrent como de los protocolos derivados de este, que aquí se proponen. Cabe aclarar que, para hacer mediciones en igualdad de condiciones, estos protocolos reciben información de la ubicación de los nodos con base en la red social. En la parte en gris oscuro se presentan los elementos que se adicionan a los protocolos BitTorrent, necesarios para los protocolos *CBt*, siendo la medida de ponderación el dato más relevante para su funcionamiento. A continuación se describen con más detalle los elementos del modelo de simulación.

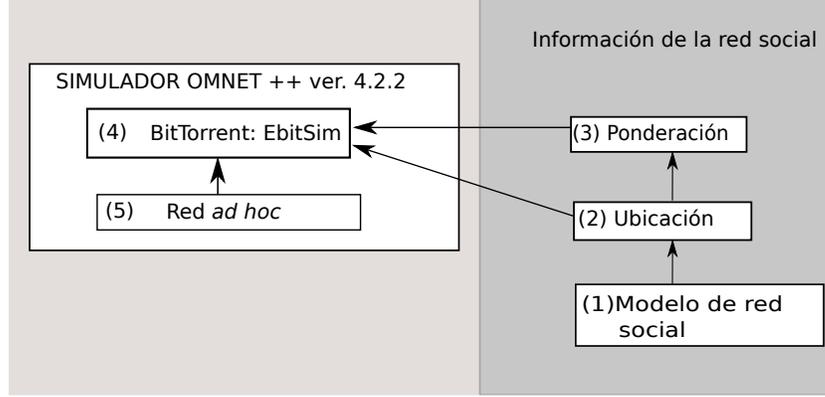


Figura 5.1: Elementos del modelo de simulación

5.2. Modelo de red social

Para la generación de la red social sintética se utilizó el modelo de Chung-Lu [31] de red libre de escala de mundos pequeños. Este modelo especifica que en un grafo aleatorio $G \in G(w)$ con una secuencia de pesos esperados

$$w = (\omega_1, \omega_2, \dots, \omega_n), \quad (5.1)$$

en donde ω_i es el peso esperado del vértice i (número esperado de enlaces que tiene el vértice i hacia otros vértices), la probabilidad P_{ij} de tener un enlace entre los vértices i y j es igual a

$$\omega_i \omega_j \rho \text{ donde } \rho = \frac{1}{\sum_i \omega_i}. \quad (5.2)$$

Se asume que el peso máximo ω_i satisface la condición

$$\omega_i^2 < \sum_j \omega_j \quad (5.3)$$

para que la probabilidad p_{ij} se encuentre estrictamente entre 0 y 1.

Chiasserini *et al.* [33] proponen calcular los pesos ω_i de los nodos y obtener el comportamiento de ley de potencia que caracteriza a las redes libre de escala de acuerdo con:

$$\omega_i = \left(\frac{\gamma - 2}{\gamma - 1} \right) \left(\frac{N}{i + i_0} \right)^{\frac{1}{\gamma + 1}} \quad (5.4)$$

donde $2 < \gamma \leq 4$, este rango del valor de γ se obtuvo con base en observaciones de sistemas reales, i_0 es una constante tal que $i_0 \leq i < N + i_0$, siendo N el total de nodos.

Para validar el modelo de Chung-Lu se generó una red con el valor de $\gamma = 2.3$, se consideró este valor porque cumple con la condición $2 < \gamma \leq 4$ que se obtuvo de escenarios reales. Se utilizó el algoritmo 1 para determinar los enlaces entre los nodos. Este algoritmo tiene como entrada la constante i_0 , el peso promedio esperado, la constante γ y el número de nodos N . Primeramente, se aplica la fórmula de Chiasserini para el cálculo de los pesos. Posteriormente se recorre la matriz $N \times N$, por cada par de elementos de la matriz se calcula la probabilidad de tener un enlace y se genera un número aleatorio entre cero y uno para obtener el valor de probabilidad. Si la probabilidad de enlace es mayor o igual a la probabilidad generada, se establece el vínculo de amistad entre ambos nodos. Este algoritmo tiene una complejidad temporal del orden $O(N^2)$.

Algoritmo 1 Algoritmo Chung-Lu

Entrada: $i_0 = 1$, peso promedio = 17, $\gamma = 2.3$, número de nodos N

Salida: Enlaces sociales entre nodos

sumaPeso=0

para $i = 1$ hasta N **hacer**

Se calcula el peso $w[i]$ de cada nodo con fórmula de Chiasserini

Se suman los pesos $sumaPeso = sumaPeso + w[i]$

fin para

para $i = 1$ hasta $N - 1$ **hacer**

para $j = i + 1$ hasta N **hacer**

$P =$ número aleatorio entre 0 y 1

Se calcula probabilidad de enlace $prob = w[i] \times w[j] / sumaPeso$

si $prob \geq P$ y i, j no amigos e $i \neq j$ **entonces**

genera enlace entre i, j

fin si

fin para

fin para

A fin de validar el algoritmo se implementó un escenario de 300 nodos y el programa se ejecutó diez veces, por lo que se generaron diez redes distintas. Posteriormente se obtuvo el promedio del número de enlaces de cada nodo en las diez ejecuciones. Se hizo un histograma con los resultados para poder observar la distribución de los enlaces.

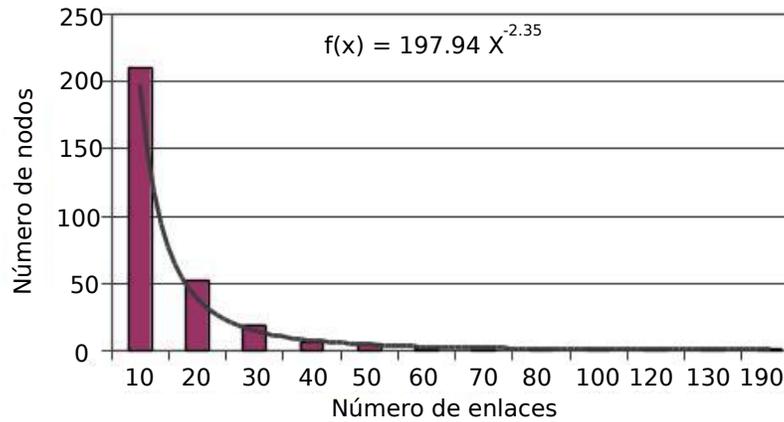


Figura 5.2: Distribución experimental de enlaces.

Al histograma se ajustó un modelo cuya gráfica se muestra como una línea de tendencia en la figura 5.2. La curva obtenida tiene la forma $Cx^{-\gamma}$, donde $C = 197.94$, $x =$ número de enlaces y $\gamma = 2.35$, este valor es muy cercano a 2.3, que es el parámetro de generación de los pesos de los nodos. Con este resultado se confirma que nuestra red cumple con la condición de ser una red libre de escala.

5.3. Algoritmo para generar la ubicación de los usuarios

Con el objetivo de modelar la ubicación de los nodos amigos se desarrolló el algoritmo 2, que toma como entrada la relación de amistad y la utiliza para determinar las posiciones de los nodos en un plano. Proponemos un escenario donde se tiene una probabilidad de 0.8 de que un nodo esté conectado a amigos cercanos al nodo que tiene el contenido a compartir, denominado nodo semilla y una probabilidad de 0.2 de que esté conectado a amigos más alejados del nodo semilla.

En el algoritmo se recibe como entrada la matriz de amistad, que tendrá un uno en la casilla correspondiente a dos nodos que son amigos y cero en otro caso, posteriormente se recorre la matriz y cuando se cumple la condición de que dos nodos son amigos, entonces se genera un número aleatorio que representa la probabilidad de que el nodo se encuentre cerca de sus amigos. Un nodo puede estar próximo a un amigo o a otro que se encuentre un poco más alejado. El primer enlace siempre se hace hacia el primer amigo que encuentra.

Para generar la ubicación se calcula una posición vecina en donde el nodo amigo se encuentra al centro, se genera un radio aleatorio entre uno y cinco metros y ángulo aleatorio entre 0 y 360 grados. Con estos elementos origen, radio y ángulo se determina la posición del nuevo nodo.

En el proceso se puede encontrar que el nodo ya cuenta con un enlace pero puede existir otro amigo que se encuentre a mayor distancia que su amigo de enlace previo, en este caso si la probabilidad de enlace es menor que 0.2, entonces se coloca en la vecindad del nodo de mayor distancia.

Se guarda el dato de número de saltos que indica a cuántos saltos se encuentra el nodo del nodo semilla, este dato se calcula tomando el número de saltos del nodo al que se enlaza más uno. El algoritmo tiene una complejidad temporal del orden $O(N^2)$.

En el algoritmo 3 se utiliza el mismo espacio en donde se encuentran ubicados los nodos amigos y se colocan nodos no amigos en posiciones al azar para modelar a los individuos que no pertenecen al grupo como se muestra en la figura 5.3.

5.4. Algoritmo de ponderación

El algoritmo 3 calcula las ponderaciones de los nodos. Este algoritmo primero simula la presencia de nodos no amigos ubicándolos aleatoriamente. Se hace el cálculo de distancias y se encuentran la rutas mínimas desde el nodo raíz a cada uno de los otros nodos en la red. El nodo raíz es el nodo semilla y el nodo destino es aquel al que se envían piezas. A lo largo de la ruta pueden existir nodos intermedios que descargan y retransmiten la pieza, ayudando a que esta llegue a su destino. Considerando la ruta mínima entre el nodo raíz y un nodo i , los nodos intermedios de esta van sumando valor de ponderación por cada nodo amigo que se encuentre en la ruta.

5.5. BitTorrent

EbitSim [32] es un software que simula el comportamiento de BitTorrent para un ambiente cableado. Su funcionamiento está enfocado a ser realista, sin perder escalabilidad. EbitSim es una implementación exacta del protocolo BitTorrent porque está desarrollado con base en el análisis del código fuente de los principales clientes BitTorrent. EbitSim fue desarrollado en el simulador OMNeT++ versión 4.2.2 y el *framework INET* para dar más exactitud al momento

Algoritmo 2 Algoritmo para ubicación de usuarios

Entrada: N número de nodos, matriz de amistad ($N \times N$)

Salida: Ubicación de los nodos P

Se inicializan posiciones de nodos $P[0] = (x, y)$, donde (x, y) es la posición inicial del nodo semilla, $P[otros] = (0, 0)$, $saltos[otros] = 0$

para $i = 1$ hasta $N - 1$ **hacer**

para $j = i + 1$ hasta N **hacer**

si existe amistad entre i y j **entonces**

 probEnlace = número aleatorio entre 0 y 1

 radio = número aleatorio entre 1 y 5

 grados = número aleatorio entre 0 y 360

si j no enlazado **entonces**

$P[j].x = P[i].x + \cos(\text{grados}) + \text{radio}$

$P[j].y = P[i].y + \sin(\text{grados}) + \text{radio}$

$saltos[j] = saltos[i] + 1$

si no

si enlace i, j más lejano que enlace previo de j y $\text{probEnlace} < 0.2$ **entonces**

 se enlaza a un nodo más alejado

$P[j].x = P[i].x + \cos(\text{grados}) + \text{radio}$

$P[j].y = P[i].y + \sin(\text{grados}) + \text{radio}$

$saltos[j] = saltos[i] + 1$

si no

 Conserva su enlace

fin si

fin si

fin si

fin para

fin para

Algoritmo 3 Algoritmo Ponderación

Entrada: Ubicación de los nodos amigos, identidad de los nodos amigos, identidad del nodo semilla, número total de nodos

Salida: ponderación de cada nodo

Genera posiciones de nodos no amigos

para $i =$ número de amigos hasta TotalNodos **hacer**

$radio =$ número aleatorio entre 1 y 100

$grados =$ número aleatorio entre 0 y 360

$P[i].x = P[0].x + \cos(grados) * radio$

$P[i].y = P[0].y + \sin(grados) * radio$

fin para

para $i = 1$ hasta N **hacer**

para $j = i + 1$ hasta N **hacer**

 Calcula $dist[i][j] =$ distancia euclidiana entre i, j

fin para

fin para

para $i = 2$ hasta N **hacer**

 Encuentra la ruta mínima por Dijkstra entre el nodo semilla y el nodo i

 Suma cantidad de amigos conectados a la ponderación de cada nodo en la trayectoria de la ruta mínima

fin para

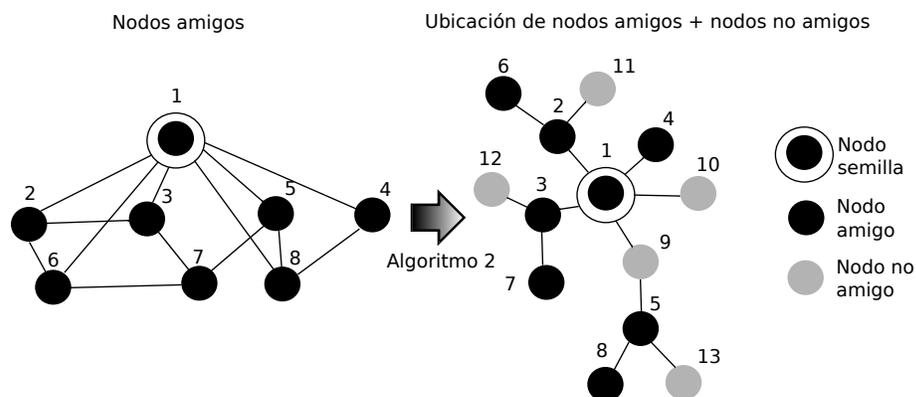


Figura 5.3: Ubicación de nodos.

de interactuar con la capa de red. También se tomaron en cuenta los efectos multitarea, ya que las aplicaciones de BitTorrent adquieren contenido de varias fuentes simultáneamente. El modelo EbitSim puede ser separado en dos partes: cliente y rastreador. La aplicación rastreador es un servidor HTTP simple.

El modelo consta de las siguientes partes: un módulo de aplicación BitTorrent que contiene una instancia con los módulos Controlador de cliente, Administrador de enjambre y el cliente BitTorrent. Además, existe un módulo *choker* y un Administrador de contenido por cada enjambre (*swarm*) en donde el cliente participa. En la figura 5.4 se muestra la arquitectura de EbitSim.

- El módulo «Controlador de cliente» (*ClientController*) hace la función del usuario que solicita el archivo *.torrent* e inicia la aplicación. Este módulo controla el tiempo en el que el cliente entra en el enjambre y el tiempo de permanencia después de convertirse en semilla, es decir, el tiempo que permanece después de haber descargado todo el contenido solicitado.
- El módulo «Administrador de enjambre» (*Swarm Manager*) se encarga de solicitar la lista de pares al *tracker*. Este módulo creará dinámicamente los procesos *choker* y un «Administrador de contenido» para cada enjambre nuevo.
- El cliente BitTorrent (*BitTorrent Client*) establece la conexión entre el cliente y el par. Cada conexión está asociada con un hilo que controla la comunicación entre los pares, cada hilo tiene acceso al *choker* y al «Administrador de contenido».
- El «Administrador de contenido» es responsable de mantener actualizada la información del contenido distribuido entre los pares conectados.
- El módulo *choker* controla el acceso de los clientes para ser atendidos.

5.6. Simulación de la red *ad hoc*

Para la red *ad hoc* se usaron las bibliotecas *INET* del simulador OMNeT++ versión 4.2.2. Estas bibliotecas fueron programadas bajo el estándar IEEE802.11. Se configuró el modelo de propagación de señal de dos rayos, con una potencia de 2 mW (valor típico) y el protocolo de encaminamiento *DSR* (*Dynamic Source Routing*) [34].

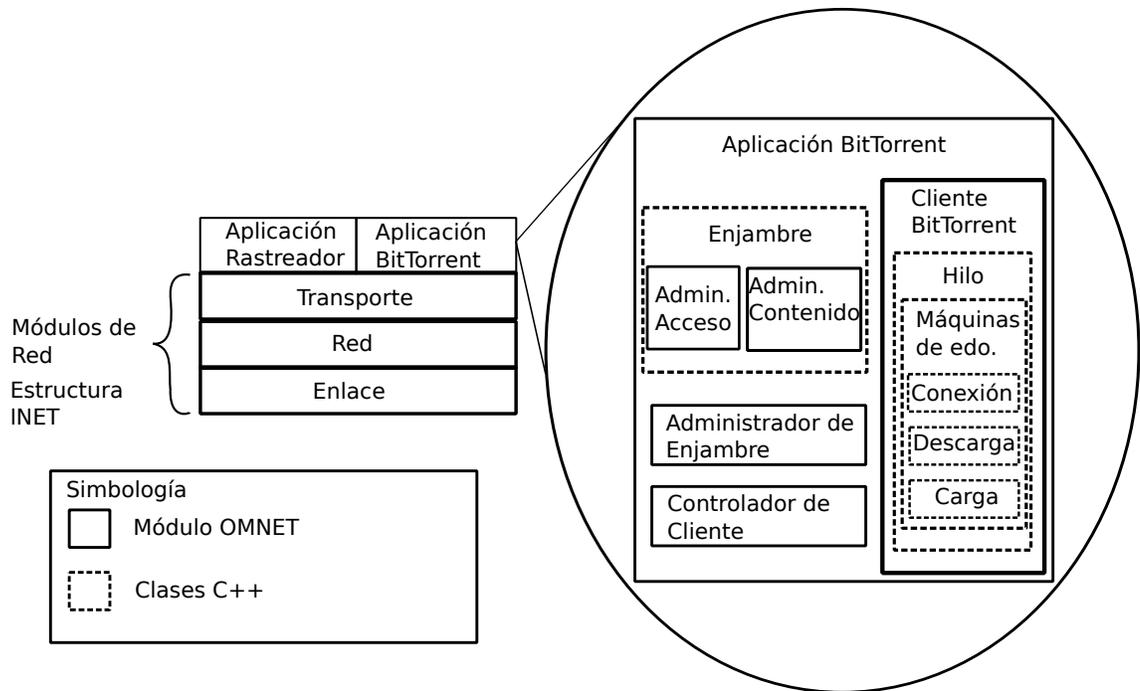


Figura 5.4: Arquitectura EbitSim.

5.7. Integración de los componentes del modelo

Para integrar los componentes anteriormente descritos se llevaron a cabo las siguientes actividades:

- Se modificó el medio de transmisión de EbitSim de cableado a inalámbrico.
- Se identificaron las condiciones para que todos los nodos de la simulación en EbitSim pudieran terminar su descarga de contenido en ambiente *ad hoc*, ya que a diferencia del ambiente cableado en donde la comunicación es estable, en el ambiente *ad hoc* se utilizan a los nodos de la red inalámbrica como encaminadores, esto provoca posibles interrupciones de comunicación en el momento que un nodo se desconecte.
- Se generó la red social sintética utilizando los algoritmos 1, 2 y 3, de los cuales se extrajeron valores de ubicación y ponderación de cada ambiente.
- Se integró la información de la red social al modelo EbitSim agregando los datos de ubicación de nodos y su valor de ponderación en el caso de los protocolos *CBt*.
- Para integrar el valor de ponderación fue necesario modificar el algoritmo de *cho-*

king/unchoking. Con este cambio, el protocolo $CBt_{p,M-p}$ tiene la capacidad de tomar en cuenta el valor de ponderación dependiendo de los parámetros de entrada, es decir, en un ciclo donde se desahogan M nodos, los parámetros indican cuántos nodos se atienden por mayor valor de ponderación (p) y cuántos serán atendidos por velocidad ($M - p$). En todas las pruebas realizadas $M=4$.

Capítulo 6

Experimentación y análisis de resultados

En este capítulo se exponen los protocolos estudiados, las medidas de desempeño que se consideraron para la evaluación de los mismos, la configuración de las simulaciones y los resultados de los experimentos.

6.1. Protocolos estudiados

Los protocolos estudiados son:

- **BitTorrent**. Es el protocolo BitTorrent estándar [9] funcionando en ambiente inalámbrico.
- **BitTorrent *ad hoc***. Este protocolo surge como una variante de BitTorrent, en donde se consideran las posibles desconexiones en una red *ad hoc*. Para ilustrar el problema que surge, considere la figura 6.1 en donde inicialmente el nodo A es semilla tanto para el nodo B como para C (quien recibe piezas a través de B). Cuando B completa todas las piezas de la descarga, este se convierte en semilla y podría decidir desconectarse de todos sus proveedores, siendo uno de ellos el nodo A. En tal situación, dado que el nodo B estaba sirviendo de puente, la comunicación entre A y C se interrumpe, posiblemente evitando así la transferencia completa de una pieza a C. Es conveniente mencionar que una pieza se declara exitosamente recibida cuando todos los bloques que la conforman se han recibido exitosamente. Una vez que la comunicación entre A y C se reestablece,

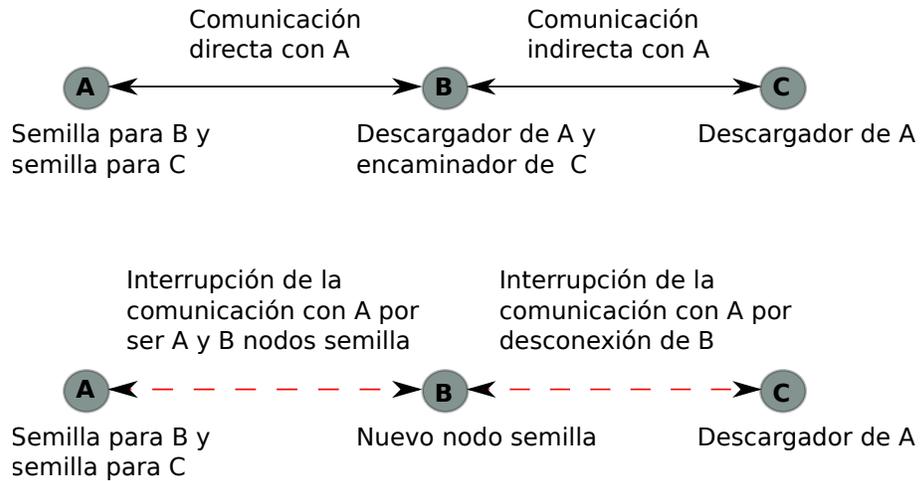


Figura 6.1: Comportamiento de desconexión de EbitSim en red *ad hoc*

a través de otra ruta, el nodo A reinicia el envío de la misma pieza comenzando desde su primer bloque. Así, se pueden generar inconsistencias en la cuenta del número de bloques enviados y recibidos exitosamente. El nodo C, en el peor de los casos, podría quedarse esperando indefinidamente por los bloques faltantes de una pieza. La solución implementada en BitTorrent *ad hoc* es que el nodo C se mantenga lanzando solicitudes mientras se tengan bloques faltantes.

- **BitTorrent *ad hoc* con rango limitado (BitTorrent RL)**. En la investigación documental realizada se identificó el trabajo BitHoc [18], donde se indica que una manera de mejorar el desempeño de BitTorrent en ambiente inalámbrico es limitar el rango de alcance para la transmisión de datos a un vecindario. En este protocolo de rango limitado se propone un vecindario formado por los nodos que se encuentran a una distancia de entre uno y hasta dos saltos.
- **Club BitTorrent $CBt_{4,0}$** . En este protocolo se agrega el conocimiento de la red social. En este caso el protocolo enviará contenido a los nodos que tengan mayor ponderación y no tomará en cuenta la velocidad de los nodos.
- **Club BitTorrent $CBt_{3,1}$** . En esta otra variante en cada ciclo de atención se selecciona a tres nodos con mayor ponderación y a uno con mayor velocidad.

6.2. Medidas de desempeño

Se tomarán las siguientes mediciones en la evaluación de los protocolos:

- Porcentaje de descarga. Se observará la evolución del porcentaje de descarga con respecto al tiempo en cada uno de los protocolos. Se medirá el avance de la descarga en periodos de cien segundos en cada uno de los experimentos. Posteriormente se hará un acumulado del avance de la descarga y se calculará el porcentaje de descarga para cada periodo.
- Tiempo promedio de descarga. Del conjunto de experimentos realizados en cada escenario se calculará el tiempo promedio que le lleva a los nodos terminar la descarga del contenido.

6.3. Configuración de simulaciones

En un área de 400 m x 400 m se consideraron escenarios con cuatro tamaños de red, correspondientes a los casos con 20, 40, 60 y 80 nodos. El número de nodos seleccionado para cada escenario se realizó con base en el ejemplo de la excursión a un sitio arqueológico de la figura 4.1, en donde un autobús de pasajeros puede llevar hasta un promedio de 40 personas que consideramos como el tamaño del grupo de amigos.

Para el primer grupo de experimentos se consideraron escenarios en donde el 50 % de los nodos son amigos y el otro 50 % son no amigos. Posteriormente se efectuaron pruebas adicionales con escenarios de 40 nodos en donde se modificó esta proporción dando lugar a los siguientes casos: a) 25 % amigos y 75 % no amigos, b) 75 % amigos y 25 % no amigos, y c) 100 % de nodos amigos.

La tecnología de conexión de los nodos es *WiFi* con alcance de 50 m en todos los escenarios. Los nodos están distribuidos en el área de acuerdo a la ubicación que arrojó el algoritmo 2, cuyo insumo es la relación social entre los nodos.

6.4. Resultados

6.4.1. Porcentaje de descarga

Se simularon los protocolos mencionados en la sección 6.1 en los diferentes escenarios y por cada protocolo en cada escenario se ejecutaron diez experimentos, obteniendo los siguientes

resultados:

- **Escenario con veinte nodos.** Los resultados del porcentaje global de descarga se muestran en la figura 6.2 (a). Dado que las curvas resultantes están muy próximas entre sí, en la tabla A.1 del apéndice A se anexan los datos de donde proviene la gráfica.

Se puede observar que los cinco protocolos tienen un desempeño similar aunque el protocolo $CBt_{4,0}$ fue el primero en alcanzar el 48 % de la descarga.

Con el objetivo de visualizar la diferencia en el porcentaje de descarga que tienen los nodos amigos y los nodos no amigos, se hicieron gráficas por separado. En la figura 6.2 (b) se muestra el porcentaje de descarga de nodos amigos, en esta gráfica se puede observar que el protocolo $CBt_{4,0}$ le da mayor preferencia a los nodos amigos debido a que tiene mayor porcentaje de descarga, aproximadamente de entre un 15 % a un 20 % en menor tiempo, lo que implica una ganancia significativa.

En la figura 6.2 (c) se muestra el porcentaje de descarga de nodos no amigos. La observación relevante en este caso es que el comportamiento de los cinco protocolos es similar, por lo que los protocolos CBt casi no afectan a la descarga de los nodos no amigos, ya que hasta el 90 % de descarga, el tiempo es similar.

- **Escenario con cuarenta nodos.** Los resultados del porcentaje global se encuentran en la figura 6.3 (a). Las curvas resultantes están muy próximas entre sí, por lo que los datos de donde proviene la gráfica se encuentran en la tabla A.2 del apéndice A. Se puede observar que empieza a haber una diferencia en el desempeño de los protocolos. El protocolo $CBt_{4,0}$ alcanza en menor tiempo mayor porcentaje de descarga con respecto a los otros protocolos. Otra punto importante es que en el protocolo BitTorrent uno de los nodos no termina la descarga.

En la figura 6.3 (b) se observa el porcentaje de descarga de los nodos amigos, donde se puede ver que el protocolo $CBt_{4,0}$ es el que da más preferencia a los nodos amigos y con un comportamiento muy similar a este se encuentra el protocolo $CBt_{3,1}$. Se observa también una ventaja de los nodos amigos sobre los nodos no amigos en el protocolo BitTorrent. Esto se debe a que el mecanismo de *choking* utilizado por BitTorrent da ventaja a los nodos con mayor velocidad de descarga y a los que comparten mayor contenido (estrategia ojo-por-ojo). En el ambiente inalámbrico la distancia entre los

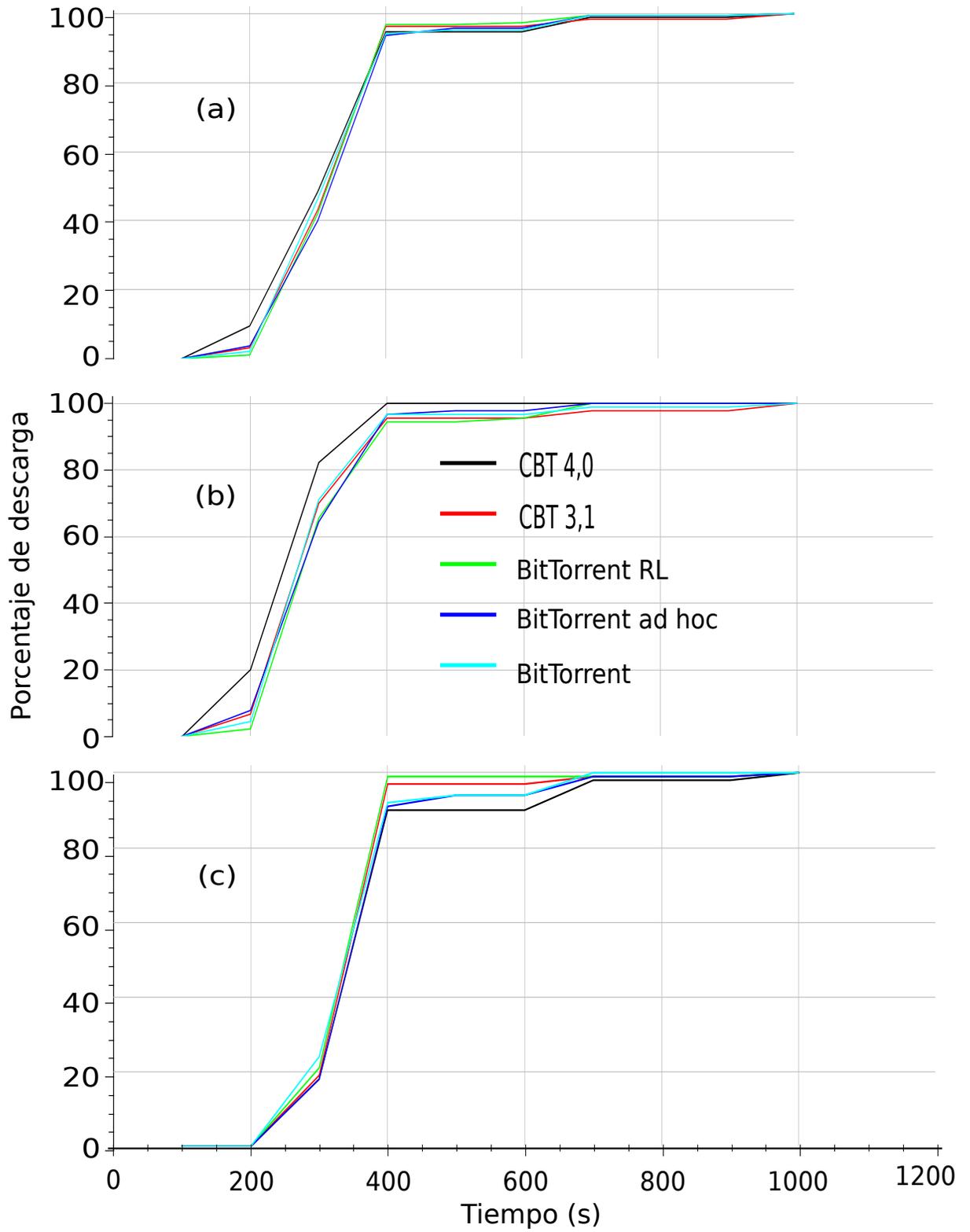


Figura 6.2: Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 20 nodos

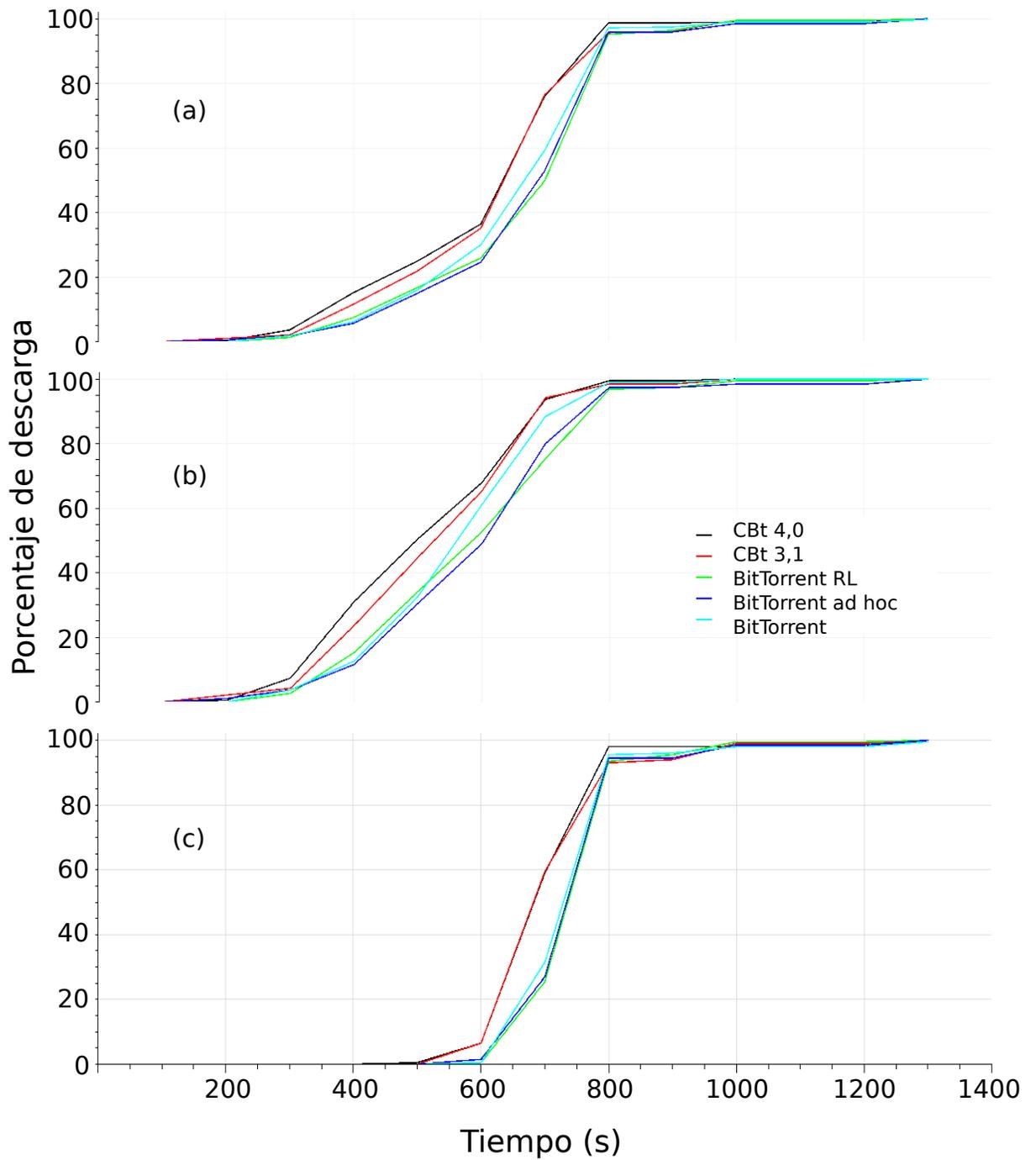


Figura 6.3: Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 40 nodos

nodos juega un papel importante debido al protocolo de encaminamiento. De tal manera que es más rápido enviar información a un nodo que está a un salto de distancia que a dos o más saltos. Los nodos amigos por lo general se encuentran cerca del nodo semilla, debido a que la distribución se realizó considerando la red social, por lo que su descarga es rápida. Se puede afirmar que en el protocolo BitTorrent los nodos amigos usualmente tienen mayor velocidad de descarga y son más cooperativos. En consecuencia los nodos amigos se ven beneficiados. Para la descarga de los nodos no amigos en la figura 6.3 (c) se tiene que los protocolos $CBt_{4,0}$ y $CBt_{3,1}$ tienen mejor tiempo de descarga que el resto de los protocolos, en este caso no se afectaron a los nodos no amigos, sino al contrario, los nodos no amigos también se beneficiaron.

- **Escenario de sesenta nodos.** Los resultados del porcentaje global de descarga se muestran en la figura 6.4 (a). Los datos de la gráfica se encuentran en la tabla A.3 en el apéndice A. En el protocolo BitTorrent nuevamente uno de los nodos no termina la descarga en una de las repeticiones. Tomando como referencia los escenarios anteriores con menor número de nodos, se observa que con sesenta nodos mejora el desempeño del protocolo $CBt_{4,0}$ con respecto a los otros protocolos.

En la figura 6.4 (b) se muestra el porcentaje de descarga de nodos amigos, se observa que los protocolos empiezan a diferenciarse con respecto a la prioridad de descarga de nodos amigos. El protocolo que más ventaja da a los nodos amigos es el $CBt_{4,0}$, en segundo lugar el $CBt_{3,1}$, seguido por el protocolo BitTorrent RL, posteriormente el BitTorrent *ad hoc* y por último el BitTorrent, aunque la diferencia entre estos dos últimos es mínima.

En el porcentaje de descarga de los nodos no amigos mostrado en la figura 6.4 (c) se observa que el comportamiento es similar al caso de los nodos amigos. En este caso los protocolos CBt también benefician a los nodos no amigos.

- **Escenario con ochenta nodos.** Los resultados del porcentaje de descarga global se muestran en la figura 6.5 (a), cuyos datos se pueden observar en la tabla A.4 en el apéndice A. En este caso se observa que aumentó la diferencia en desempeño entre el protocolo BitTorrent *ad hoc* y el $CBt_{4,0}$, este último logrando mayor porcentaje de descarga en menos tiempo.

De la misma forma que en los escenarios anteriores se hicieron gráficas por separado de descarga de nodos amigos y no amigos.

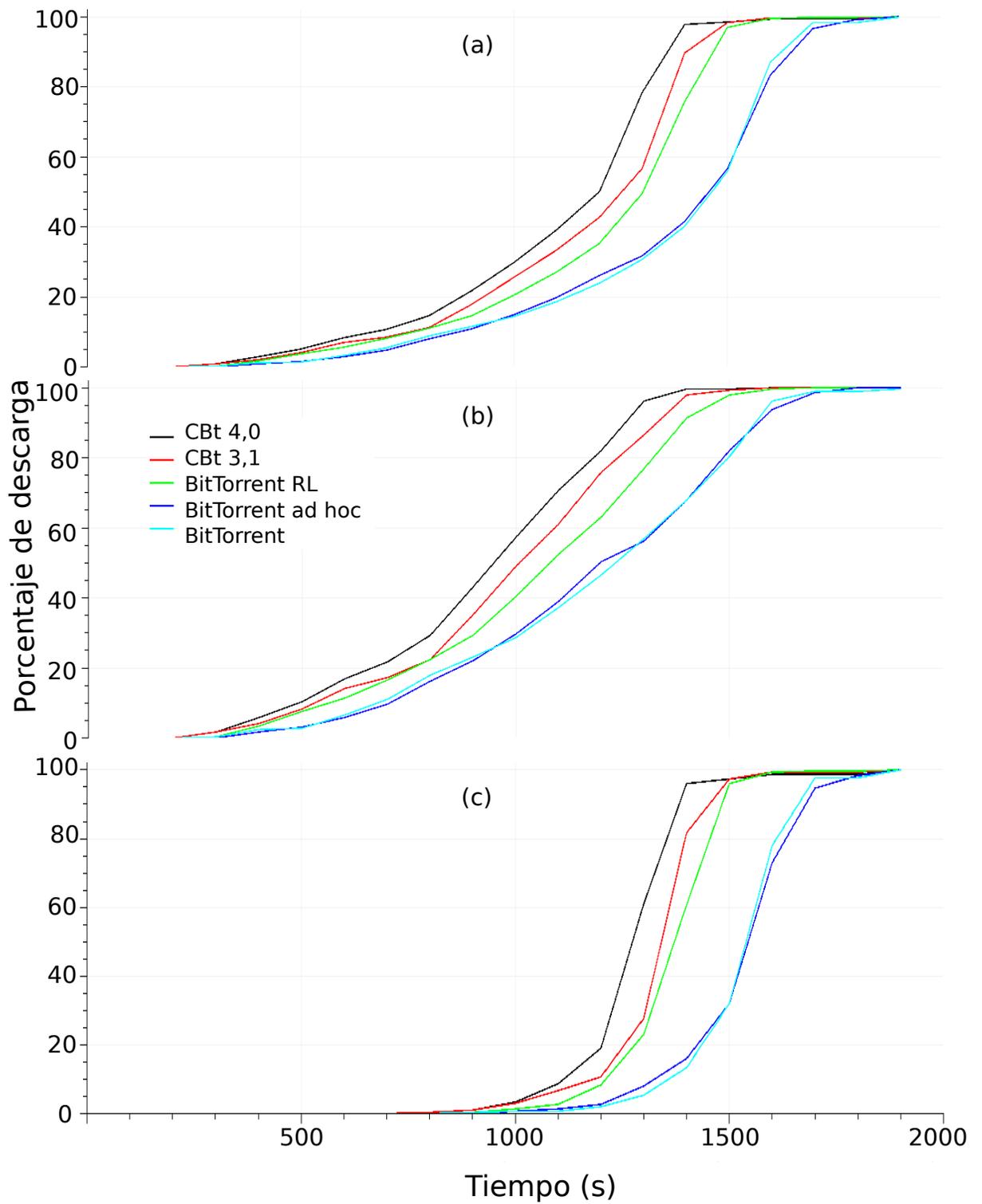


Figura 6.4: Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 60 nodos

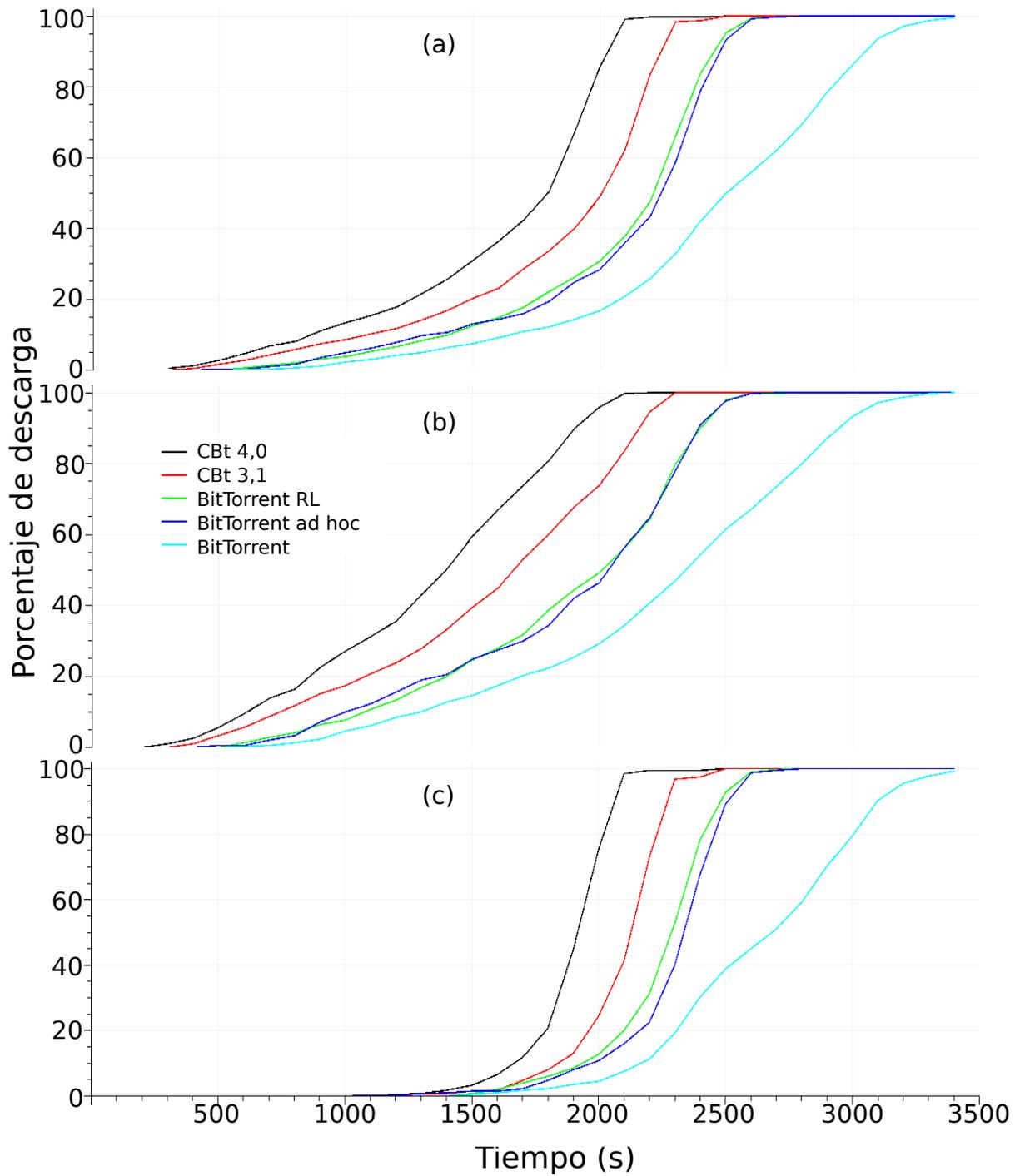


Figura 6.5: Porcentaje de descarga (a) global, (b) amigos y (c) no amigos vs. tiempo, escenario con 80 nodos

Para la descarga de nodos amigos mostrada en la figura 6.5 (b) se observa que se amplía la diferencia entre los protocolos con respecto al beneficio de los nodos amigos. Sin embargo, un punto importante es que la diferencia de los protocolos BitTorrent *ad hoc* y BitTorrent RL es mínima, esto se puede explicar recordando que la mayoría de los amigos se encuentran juntos, por lo que limitar el rango no representa ventaja.

Para el caso de la descarga de los nodos no amigos mostrado en la figura 6.5 (c) se observa que los protocolos *CBt* siguen beneficiando también a los nodos no amigos, y que, a diferencia de la descarga de los nodos amigos, el protocolo BitTorrent RL sí tiene mejor comportamiento que el protocolo BitTorrent *ad hoc* debido a que los nodos no amigos pueden estar más lejos que el rango definido para limitar la transmisión.

6.4.2. Tiempo promedio de descarga

El principal objetivo de este trabajo es verificar la hipótesis de que el conocimiento de la relación social entre los nodos puede mejorar el desempeño de la descarga de contenido par a par. Por esta razón, primeramente se consideró el tiempo promedio de descarga de todos los nodos incluyendo los nodos sin relación social. Posteriormente se observó por separado el tiempo promedio de la descarga en los nodos con relación social o nodos amigos. Y por último se consideraron los nodos sin relación social o no amigos para observar si estos tienen afectación en su desempeño con las modificaciones del protocolo *CBt*.

El tiempo promedio de descarga global se presenta en la gráfica 6.6. En el escenario de veinte nodos los tiempos de descarga globales son similares, se observa que a medida que se aumenta el número de nodos se va incrementando la ganancia de tiempo del protocolo *CBt*_{4,0}. Otra observación importante es que a medida que aumenta el número de nodos, el tiempo promedio de descarga aumenta en todos los protocolos. Sin embargo, el protocolo *CBt*_{4,0} se ve menos afectado, es decir, el tiempo promedio no aumenta en la misma proporción que en los otros protocolos. El porcentaje global de ganancia en tiempo promedio de descarga total del protocolo *CBt*_{4,0} con respecto al protocolo BitTorrent *ad hoc* se muestra en la gráfica 6.7.

En la gráfica 6.8 se muestra el tiempo promedio de descarga de los nodos amigos en cada uno de los escenarios. Se puede observar que el tiempo promedio del protocolo *CBt*_{4,0} en todos los escenarios fue menor que el de los otros protocolos. También se observa que a medida que se aumenta el número de nodos la ganancia en tiempo del protocolo *CBt*_{4,0} con respecto al protocolo BitTorrent *ad hoc*, que es nuestro protocolo de referencia, también aumenta, ver

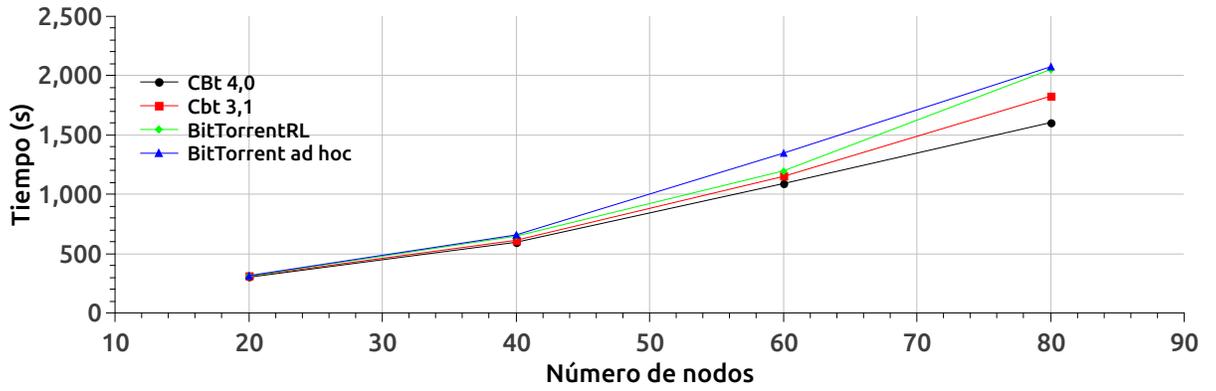


Figura 6.6: Tiempo promedio de descarga global en escenarios de 20, 40, 60 y 80 nodos

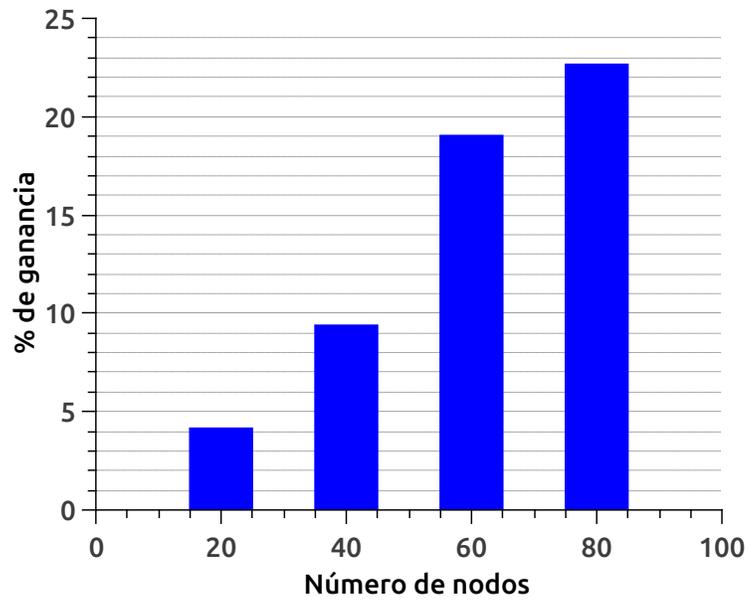


Figura 6.7: Porcentaje de ganancia en tiempo promedio de descarga del global de nodos $CBt_{4,0}$ vs. BitTorrent *ad hoc*

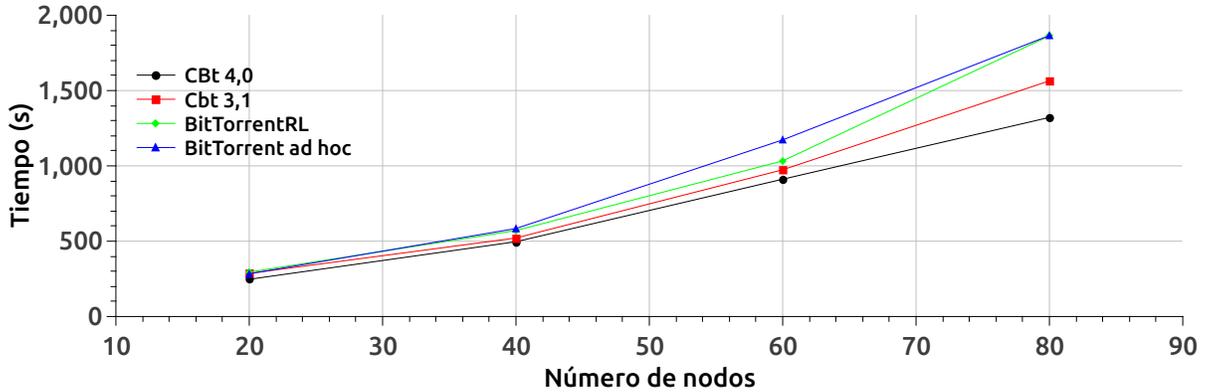


Figura 6.8: Tiempo promedio de descarga de nodos amigos

gráfica 6.9.

Por último se observó el tiempo promedio de los nodos no amigos, es decir, que no tienen relación social con el grupo. Como se muestra en la gráfica 6.10, para los escenarios de veinte y cuarenta nodos los tiempos promedio de los protocolos estudiados son muy parecidos, pero en los escenarios de sesenta y ochenta nodos el protocolo $CBt_{4,0}$ muestra un menor tiempo promedio de descarga que los otros protocolos estudiados. El porcentaje de ganancia en los no amigos se muestra en la figura 6.11, donde se puede observar que para el escenario con 20 nodos no hubo ganancia para los nodos no amigos, sin embargo, a medida que aumenta el número de nodos empezamos a tener ganancia para estos nodos no amigos.

Tomando en cuenta los resultados de las medidas de desempeño se puede resumir que:

- El protocolo $CBt_{4,0}$ beneficia a los nodos amigos.
- El protocolo $CBt_{4,0}$ aumenta el tiempo promedio de descarga en menor proporción que los otros protocolos estudiados al aumentar el número de nodos.
- El principal objetivo de los protocolos CBt es mejorar la experiencia de descarga de contenido en un grupo de amigos, sin embargo, también mejora el tiempo promedio de descarga de los nodos no amigos. Esto se debe a su arquitectura, ya que si un nodo no amigo puede desahogar a nodos amigos entonces se le da prioridad de atención.

6.4.3. Efecto de la proporción amigos-no amigos

En esta sección se muestran los resultados obtenidos cuando se varió la proporción entre nodos amigos y no amigos.

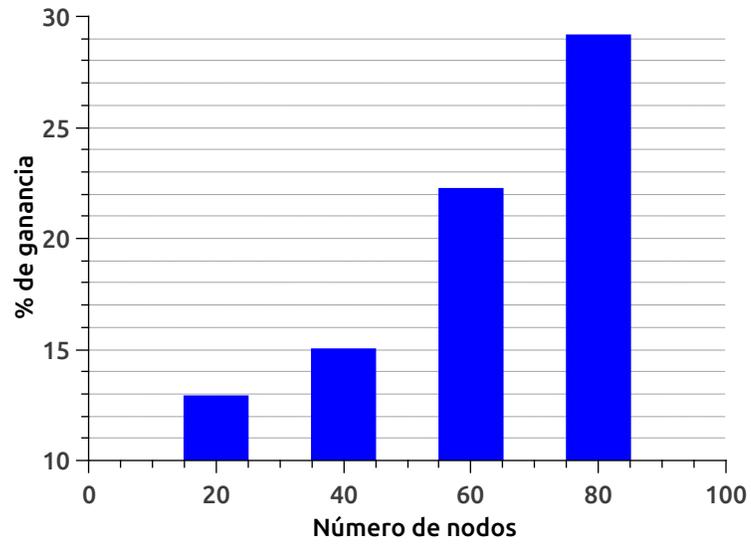


Figura 6.9: Porcentaje de ganancia en tiempo promedio de descarga de nodos amigos $CBt_{4,0}$ vs. BitTorrent *ad hoc*

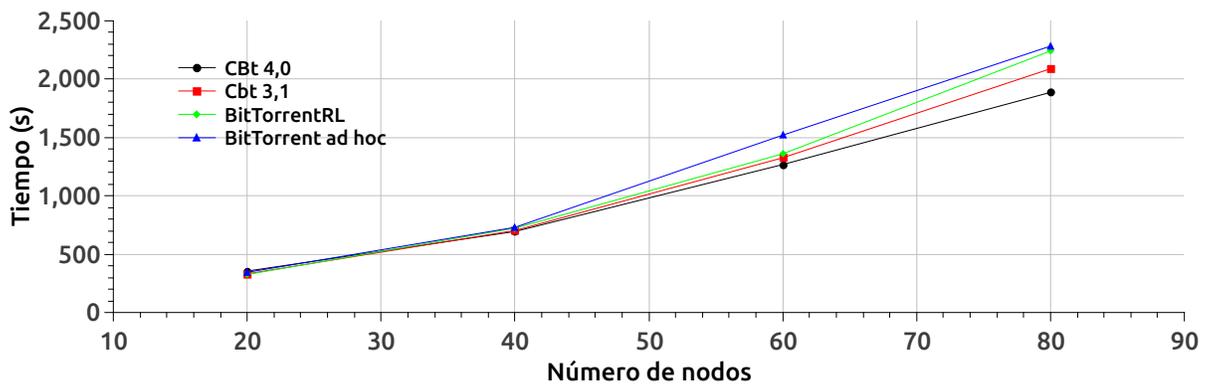


Figura 6.10: Tiempo promedio de descarga de nodos no amigos

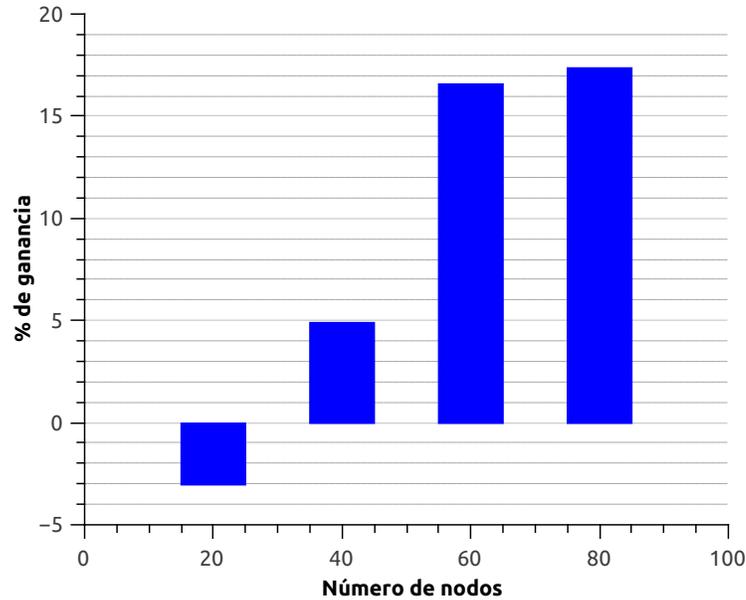


Figura 6.11: Porcentaje de ganancia en tiempo promedio de descarga de nodos no amigos $CBt_{4,0}$ vs. BitTorrent *ad hoc*

Se tomó como base el escenario de cuarenta nodos con los protocolos BitTorrent *ad hoc* y $CBt_{4,0}$. Las proporciones que se tomaron fueron las siguientes considerando que el primer número indica el número de nodos amigos y la segunda el de nodos no amigos: 10-30, 20-20, 30-10 y 40-0.

Los resultados fueron los siguientes:

- Porcentaje de descarga. En la figura 6.12 se muestra el porcentaje de descarga para las variaciones en la proporción de amigos y no amigos. Se puede observar que dicha proporción afecta el desempeño de ambos protocolos de manera similar. El protocolo BitTorrent sufre alteraciones en su comportamiento porque la distribución espacial de los nodos se hizo con base en los datos de la red social. Cuando existe mayor número de amigos la distribución de estos es más densa, los amigos por lo general están juntos, lo que provoca que exista más interferencia entre ellos al momento de la transmisión inalámbrica. El mejor escenario para el desempeño del protocolo $CBt_{4,0}$ es el 20-20. Otra observación es que el protocolo $CBt_{4,0}$ tiene mejor desempeño general en las distintas proporciones de amigos.
- Tiempo promedio de descarga. Se obtuvo el tiempo promedio de descarga de los expe-

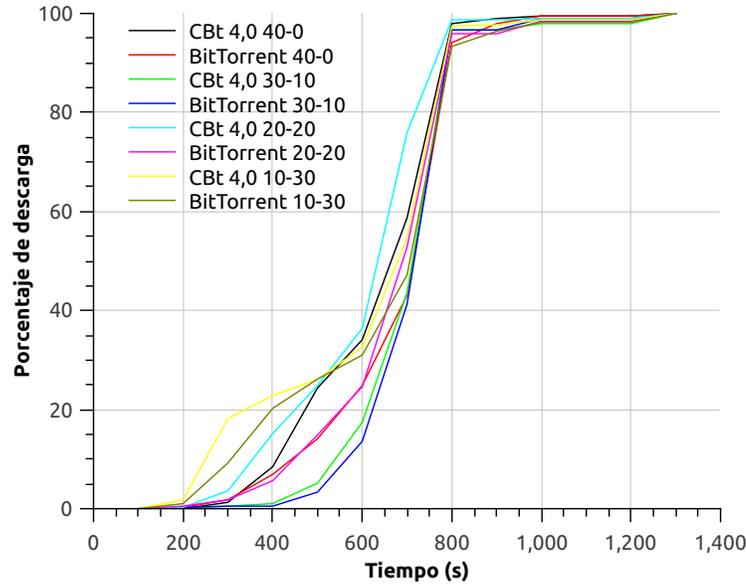


Figura 6.12: Porcentaje de descarga para 40 nodos variando proporción de amigos

rimentos con la variación de la proporción de amigos. El primer dato se muestra en la figura 6.13, donde se observa el porcentaje de ganancia del tiempo promedio de descarga del protocolo $CBt_{4,0}$ sobre el protocolo BitTorrent. En todos los casos existe ganancia a excepción de la proporción de cuarenta nodos amigos porque no hay no amigos. En las tres medidas que son tiempos promedio de descarga global, amigos y no amigos la proporción de 50% amigos y 50% no amigos (20-20) es donde se presenta mayor ganancia.

El tiempo promedio de descarga global se muestra en la figura 6.14. De igual manera que en la medida de porcentaje de descarga, se puede observar que en el tiempo promedio de descarga ambos protocolos sufren una afectación similar debido la distribución espacial de los nodos amigos. En todos los experimentos con variación de la proporción de amigos el protocolo $CBt_{4,0}$ presenta menor tiempo promedio de descarga que BitTorrent. El pico de mayor tiempo promedio de descarga se presenta en la proporción 30-10, debido a que la mayoría de los nodos amigos ya está compitiendo por el medio al estar cerca unos de otros, a esto se le agrega que existen diez nodos no amigos que quedaron alejados, lo que aumenta el tiempo de descarga. En el caso 40-0 existe la competencia por el medio entre los nodos amigos por la cercanía, pero no se tienen nodos lejanos, esto provoca que disminuya el tiempo promedio de descarga con respecto a la proporción 30-10.

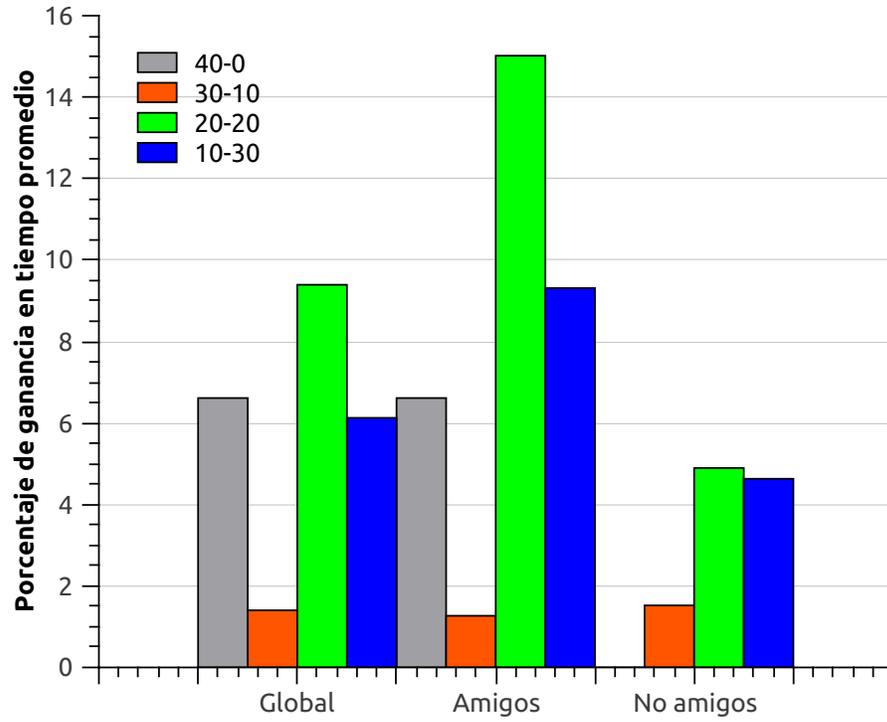


Figura 6.13: Ganancia en tiempo promedio de descarga para 40 nodos $Cbt_{4,0}$ sobre BitTorrent

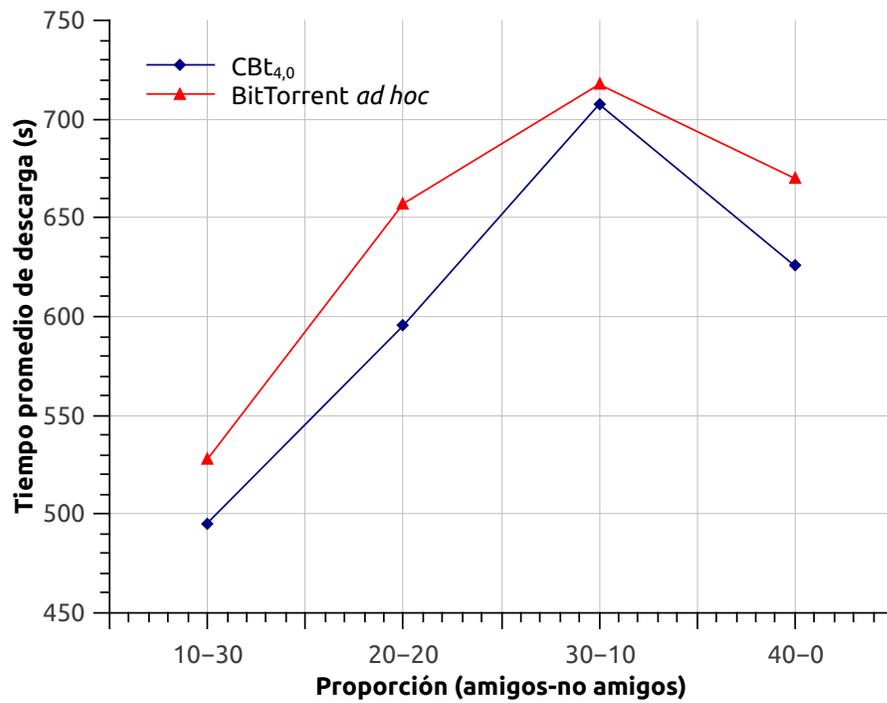


Figura 6.14: Tiempo promedio de descarga global 40 nodos variando proporción de amigos

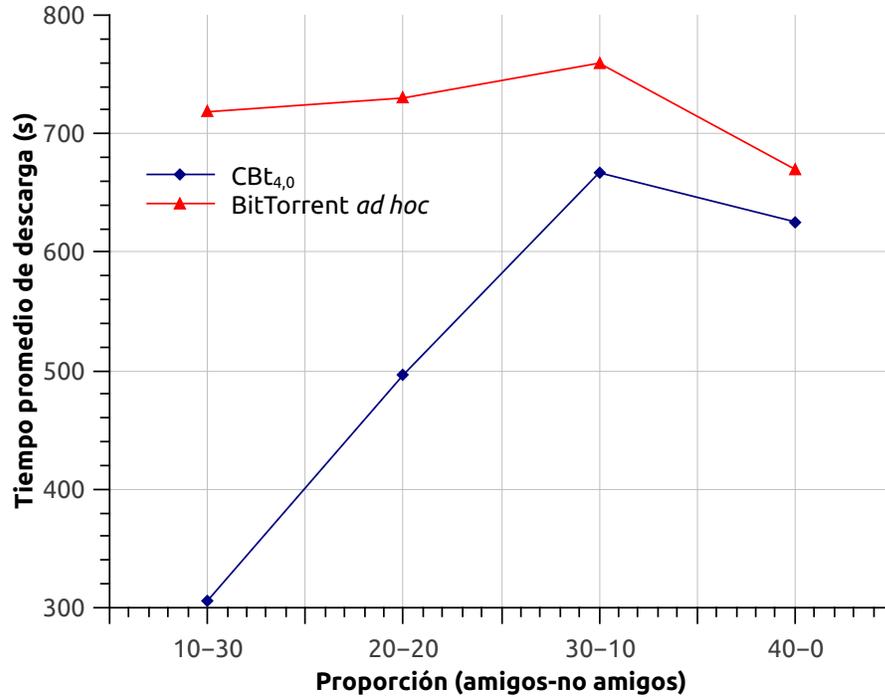


Figura 6.15: Tiempo promedio de descarga en amigos 40 nodos variando proporción de amigos

En la figura 6.15 se presentan los resultados del tiempo promedio de descarga de los nodos amigos. El protocolo $CBt_{4,0}$ presenta menor tiempo promedio de descarga. Y a medida que aumenta la proporción de nodos amigos el tiempo promedio de descarga aumenta hasta llegar al pico en 30-10, posteriormente vuelve a bajar, aunque la diferencia con el protocolo BitTorrent siempre disminuye al aumentar el número de nodos amigos. Los resultados del tiempo promedio de los nodos no amigos se muestran en la figura 6.16. Se observa que el protocolo $CBt_{4,0}$ también tiene un tiempo promedio de descarga menor para los nodos no amigos. Y que el comportamiento de ambos protocolos es similar. La diferencia en tiempo de descarga de no amigos entre ambos protocolos disminuye al aumentar el número de amigos de 20 a 30 nodos.

En el siguiente capítulo se exponen las conclusiones de este trabajo.

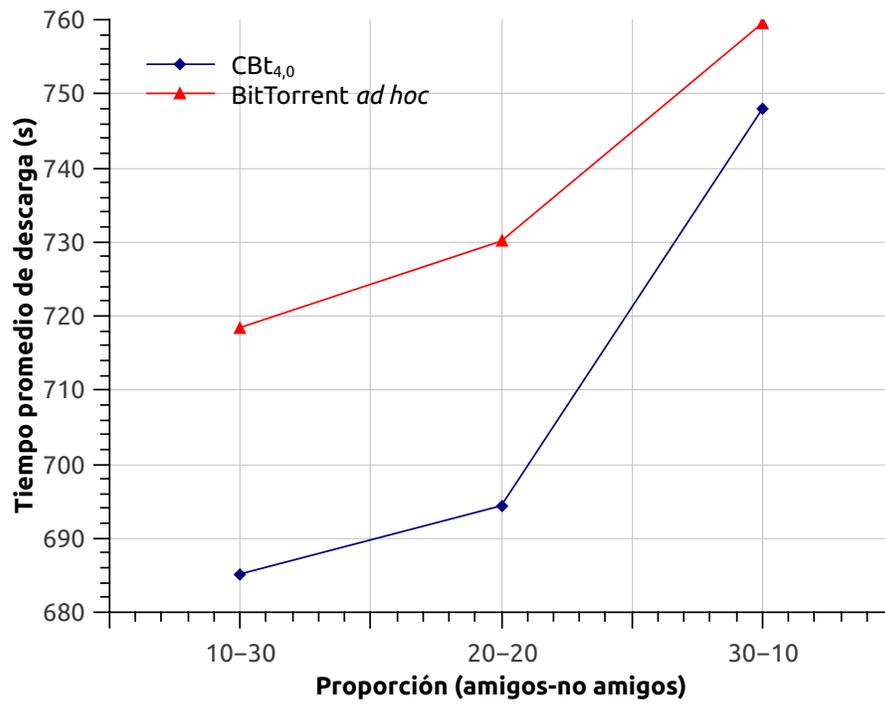


Figura 6.16: Tiempo promedio de descarga en no amigos 40 nodos variando proporción de amigos

Capítulo 7

Conclusiones y perspectivas

7.1. Conclusiones

Nuestro objetivo en este trabajo de investigación fue diseñar un protocolo de descarga de contenido en redes *ad hoc* inalámbricas, utilizando conocimiento de redes sociales. A continuación se mencionan las conclusiones a las que llegamos en el desarrollo de dicha investigación.

Se identificaron las variantes del protocolo de descarga BitTorrent para redes *ad hoc*. Estas variantes se encuentran en el mecanismo *choking/unchoking* utilizado por los nodos con contenido para enviar piezas a sus pares. Mientras que el mecanismo estándar selecciona a tres nodos con mayor velocidad de descarga y un cuarto nodo al azar, en este trabajo se modificó el método de selección de par para integrar el conocimiento de la relación social entre los nodos.

Se observó el comportamiento de la relación social entre individuos, para ello se implementó una red social sintética basada en el modelo de Chung-Lu. El comportamiento que sigue este tipo de red tiene una distribución de ley de potencias, esto se debe a que los individuos que se integran a la red se enlazan preferentemente a los nodos que cuentan con el mayor número de conexiones. De esta red se obtuvo una distribución espacial de los nodos, para modelar grupos de amigos. Una vez que los nodos contaban con su ubicación se empleó el algoritmo de Dijkstra para encontrar rutas mínimas. Se obtuvo la variable de ponderación, que nos indica el número de nodos amigos que un nodo puede conectar sobre una ruta mínima. Esta variable nos proporciona el conocimiento de la relación social entre los nodos, ya que un nodo por lo general está en contacto con sus amigos.

El protocolo propuesto *CBt* (Club BitTorrent) integra la variable de ponderación al me-

canismo de *choking/unchoking*. En la propuesta se da prioridad para el envío de piezas a aquellos nodos, ya sean nodos amigos o no amigos, que puedan atender al mayor número de nodos amigos, es decir, los que tengan ponderación más alta. Del protocolo *CBt* se pueden obtener variantes modificando la cantidad de nodos a atender bajo el criterio de ponderación y la cantidad de nodos a atender bajo el criterio de velocidad de descarga, de las cuatro posibilidades, se obtuvieron y presentaron los resultados de las dos con mejores posibilidades *Cbt_{3,1}* atendiendo a tres nodos por ponderación y uno por velocidad y *Cbt_{4,0}* que atiende solo por ponderación, los otros protocolos con menor número de nodos por ponderación se comportan muy similar al protocolo BitTorrent.

En el desarrollo de este trabajo se encontró que en el protocolo BitTorrent se presentan casos en los que no todos los nodos terminan su descarga. Esto es debido al ambiente *ad hoc*, a diferencia del ambiente cableado, para alcanzar a un nodo alejado geográficamente se utiliza un protocolo de encaminamiento, en este caso *DSR*, que usa a otros nodos para alcanzar al nodo objetivo. En la etapa experimental se observó que si dos nodos se desconectaban y uno de ellos era puente para conectar a un tercer nodo, este tercer nodo también perdía conexión. Esto provocaba que se interrumpiera el envío de bloques de piezas y en algunos casos este envío quedaba bloqueado. De lo cual surgió un protocolo alternativo al que se llamó BitTorrent *ad hoc*, que ya contempla esta situación y el cual se usó como referencia para comparar el desempeño de otros protocolos.

Se confirmó que limitando el rango de alcance mejora el tiempo de descarga de acuerdo a lo publicado en el trabajo BitHoc [18].

De los resultados de los experimentos realizados se obtiene que:

- El protocolo *CBt_{4,0}*, que toma en cuenta el conocimiento de la relación social entre los nodos, presenta un mejor desempeño en la descarga de contenido para nodos amigos.
- El protocolo *CBt_{4,0}*, obtiene más alto porcentaje de descarga en menor tiempo y mejora el tiempo promedio de descarga total con respecto al protocolo BitTorrent *ad hoc*.
- También los nodos no amigos son beneficiados al considerar la relación social, esto se debe al diseño del protocolo *CBt_{4,0}*, ya que para hacer llegar contenido a amigos que se encuentran alejados puede utilizar a los nodos no amigos, lo que genera el beneficio.
- A medida que se incrementa el número de nodos, se mejora el desempeño, esto se observó en ambas medidas, tanto en el porcentaje de descarga, como en el tiempo promedio de

descarga, esto se debe a que la información social ayuda a decidir a qué nodos enviar piezas de manera más eficiente, es decir, siempre se sigue la política de atender al mayor número de amigos posible.

- Se investigó qué sucede cuando se varía la proporción de nodos amigos con respecto a los no amigos. De lo cual se concluye que la densidad de nodos juega un papel importante en el desempeño del intercambio de contenido, ya que a mayor densidad de nodos mayor competencia por el medio. También se comprobó que aun variando la proporción de amigos $CBt_{4,0}$ obtuvo ventaja sobre BitTorrent en los casos estudiados.

Los resultados obtenidos contribuyen a demostrar la hipótesis inicial de este trabajo en el sentido de que utilizar la información de las relaciones de amistad en un grupo puede llevar a mejoras en los protocolos de compartición de contenidos tales como BitTorrent.

7.2. Perspectivas

En el estudio realizado en este trabajo de investigación se consideró un ambiente con nodos estáticos. Sin embargo, se considera que los resultados obtenidos son válidos en el ambiente planteado del ejemplo de la excursión para transferencias de archivos pequeños o mensajes de texto. El tiempo de descarga es menor al tiempo que le llevaría a los nodos salir del rango de transmisión considerado inicialmente. Además se considera, por la naturaleza del ejemplo, que los nodos se mueven en grupo, por lo que no se espera que la topología cambie en periodos cortos. Sin embargo, una perspectiva de investigación es precisamente agregar movilidad a los nodos y observar qué condiciones del ambiente estático siguen siendo válidas para un ambiente con movilidad.

Los resultados obtenidos en esta investigación indican que puede haber ganancias en el desempeño de futuros protocolos P2P de distribución de contenido en ambientes ad hoc inalámbricos, si durante su diseño se toma en cuenta la información que se puede obtener a partir de una red social.

Lista de referencias

- [1] Mohamed Karim Sbai, Chadi Barakat, Jaeyoung Choi, Anwar Al Hamra y Thierry Turlletti, “Adapting bittorrent to wireless ad hoc networks,” *Conference Ad hoc Now, Francia*, 2008. [1](#), [7](#), [14](#)
- [2] Catanese S., De Meo P., Ferrara E., Fiumara G., Provetti A., “Extraction and analysis of Facebook friendship relations,” *Computational Social Networks*, 2012. [2](#)
- [3] Grandjean, M., “A social network analysis of Twitter: Mapping the digital humanities community.,” *Cogent Arts Humanities*, *3(1)*, 1171458. <https://doi.org/10.1080/23311983.2016.1171458>, 2016. [2](#)
- [4] Milgram S., “The small world problem,” *Psychology today*, 1967. [2](#), [10](#)
- [5] A.L. Barabási and R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, pp. 509–512, 1999. [2](#), [10](#), [11](#)
- [6] BitTorrent, “BitTorrent,” [Online]. Disponible: <http://www.bitTorrent.com/company/about>. [Accesado: 22-Dic-2017], 2017. [2](#)
- [7] Gauri Shah, “Distributed Data Structures for Peer-to-Peer Systems,” *Ph.D thesis Yale University*, 2003. [2](#), [6](#)
- [8] Andrea Goldsmith, *Wireless communications*. Cambridge University, 2005. [5](#)
- [9] Bram Cohen, “Incentives build robustness in bittorrent,” *Workshop on Economics of P2P systems, Berkeley, CA, USA*, Mayo 2003. [7](#), [9](#), [13](#), [35](#)
- [10] Pietro Michiardi, Guillaume Urvoy-Keller, “Performance analisis of cooperative content distribution in wireless ad hoc networks,” *Fourth annual conference on wireless on demand network systems and services*, pp. 22–29,24–26, Enero 2007. [9](#)

- [11] López Chavira MA, Marcelín-Jiménez R, “ Distributed rewiring model for complex networking: The effect of local rewiring rules on final structural properties. ,” *PLoS ONE* 12(11): e0187538. <https://doi.org/10.1371/journal.pone.0187538>, 2017. 10
- [12] P. Erdos, A. Rényi, B. Bollobás, “Random graphs,” *Academic Press, Lodon*, pp. 5,17, 1985. 10
- [13] Watts DJ, Strogatz SH., “Collective dynamics of ‘small-world’ networks.,” *Nature*, 1998. 10
- [14] N. Ghali, M. Panda, AE Hassanien, A. Abraham, “Social networks analysis: Tools, measures and visualization,” *Computational social networks*, p. 323, 2012. 12
- [15] G. Caldarelli, “Scalefree networks : Complex webs in nature and technology.,” *Oxford University Press*, 2007. 12
- [16] Y. Li, W. Chen, Y. Wang, and Z. Zhang, “Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships,” *CoRR*, 2011. 12
- [17] Salhi E., Sbai M. K., y Barakat C., “Neighborhood selection in mobile p2p networks,” *In Algotel conference, Carry-Le-Rouet, Francia*, 2009. 14
- [18] Krifa A., Sbai M. K., Barakat C., y Turlet T., “Bithoc: A content sharing application for wireless ad hoc networks. in pervasive computing and communications,” *IEEE International Conference on (pp. 1-3)*, 2009. 14, 36, 54
- [19] Krifa A., Sbai M. K., Barakat C. y Turletti T., “A standalone content sharing application for spontaneous,” *Proceeding MobiHeld '09 Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds, (pp. 77-78)*, 2009. 14
- [20] Maarten Fonville, “Confidential peer-to-peer file-sharing using social-network sites,” *[Online]. Available: <http://referaat.cs.utwente.nl/conference/13/paper>*, 2010. 15
- [21] T. Isdal, M. Piatek, A. Krishnamurthy, and T. Anderson, “Privacy-preserving p2p data sharing with oneswarm,” *Technical report*, 2009. 15
- [22] Xiao Han, Ángel Cuevas, Noël Crespi, Rubén Cuevas, Xiaodi Huang, “On exploiting social relationship and personal background for content discovery in p2p networks,” *Journal Future generation computer systems*, vol. 40, 2014. 15

- [23] Ze Li y Haiying Shen, “Social-p2p: Social network-based p2p file sharing system,” *In 2012 20th IEEE International conference on network protocols (ICNP)*, 2012. [16](#)
- [24] Ralf Steinmetz y Klaus Wehrle, *Peer-to-peer systems and applications*, vol. 3485 of *0302-9743*. Springer Berlin Heidelberg, 2005. [16](#)
- [25] Sergio Marti, Prasanna Ganesan y Hector Garcia-Molina, “Dht routing using social links,” *Proceedings of the Third international conference on Peer-to-Peer systems*, 2004. [16](#)
- [26] S. Marti, P. Ganesan y H. Garcia-Molina, “Sprout: P2p routing with social networks,” *EDBT 2004 Workshops PhD, DataX, PIM, P2PDB, and ClustWeb, Heraklion, Crete, Greece*, 2004. [16](#)
- [27] Kang Chen, Haiying Shen y Haibo Zhang, “Leveraging social networks for p2p content-based file sharing in mobile ad hoc networks,” *IEEE Eighth International Conference on Mobile Ad Hoc and Sensor Systems*, 2011. [16](#)
- [28] Xu Cheng , Jiangchuan Liu, “Nettube: Exploring social networks for peer-to-peer short video sharing,” *Proc. IEEE INFOCOM*, 2009. [17](#)
- [29] Vincenza Carchiolo, Michele Malgeri, Giuseppe Mangioni, Vincenzo Nicosia, “An adaptive overlay network inspired by social behaviour”,” *Journal Parallel Distrib. Comput.*, 2010. [17](#)
- [30] András Varga, Rudolf Hornig, “An overview of the omnet++ simulation environment,” *Proceedings of the 1st international conference on simulation tools and techniques for communications, networks and systems workshops*, 2008. [23](#)
- [31] F. Chung y L. Lu, “The average distance in a random graph given expected degrees,” *Internet Math.*, vol. 1, 2004. [24](#), [25](#)
- [32] Pedro Evangelista, Marcelo Amaral, Charles Miers, Walter Goya, Marcos Simplicio, Tereza Carvalho, Victor Souza, “Ebtsim: An enhanced bittorrent simulation using omnet++4,” *Conference Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS), 2011 IEEE 19th International Symposium*, Julio 2011. [24](#), [28](#)

- [33] Carla Fabiana Chiasserini, Michele Garetto y Emilio Leonardi, "Social network deanonymization under scalefree user relations," *IEEE/ACM Transactions on Networking*. [25](#)
- [34] David B. Johnson y David A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile computing*, 1996. [31](#)

Apéndice A

Tablas

Tabla A.1: Porcentaje global de descarga vs. tiempo, escenario con 20 nodos

Tiempo (s)	% BitTorrent	%BitTorrent ad hoc	%BitTorrent RL	%CBt 3,1	%CBt 4,0
100	0.0000	0.0000	0.0000	0.0000	0.0000
200	2.1052	3.6842	1.0526	3.1578	9.4736
300	46.3157	40.0000	42.1052	43.15789	48.4210
400	94.2105	93.6842	96.8421	96.31578	94.7368
500	95.2631	95.7894	96.8421	96.31578	94.7368
600	95.2631	95.7894	97.3684	96.31578	94.7368
700	99.4736	99.4736	99.4736	98.4210	98.9473
800	99.4736	99.4736	99.4736	98.4210	98.9473
900	99.4736	99.4736	99.4736	98.4210	98.9473
1000	100.0000	100.0000	100.0000	100.0000	100.0000

Tabla A.2: Porcentaje global de descarga vs. tiempo, escenario con 40 nodos

Tiempo	% BitTorrent	%BitTorrent ad hoc	%BitTorrent RL	%CBt 3,1	%CBt 4,0
100	0.0000	0.0000	0.0000	0.0000	0.0000
200	0.0000	0.0000	0.0000	1.0256	0.2564
300	1.7948	1.7948	1.2820	2.0512	3.5897
400	6.1538	5.6410	7.4358	11.5384	15.1282
500	15.8974	14.8717	16.6666	21.7948	24.8717
600	30.0000	24.6153	25.8974	35.1282	36.4102
700	59.2307	52.8205	49.7435	76.4102	75.8974
800	97.1794	95.8974	95.1282	95.6410	98.7179
900	97.4358	95.8974	96.4102	96.1538	98.7179
1000	98.9743	98.4615	99.4871	99.4871	98.9743
1100	98.9743	98.4615	99.4871	99.4871	98.9743
1200	98.9743	98.4615	99.4871	99.4871	98.9743
1300	99.7435	100.0000	100.0000	100.0000	100.0000

Tabla A.3: Porcentaje global de descarga vs. tiempo, escenario con 60 nodos

Tiempo	% BitTorrent	%BitTorrent ad hoc	%BitTorrent RL	%CBt 3,1	%CBt 4,0
100	0.0000	0.0000	0.0000	0.0000	0.0000
200	0.0000	0.0000	0.0000	0.0000	0.0000
300	0.1694	0.0000	0.1694	0.8474	0.8474
400	1.1864	0.8474	1.6949	2.0338	2.8813
500	1.3559	1.5254	3.7288	4.0677	5.0847
600	3.2203	2.8813	5.5932	6.9491	8.3050
700	5.4237	4.7457	8.1355	8.4745	10.6779
800	8.8135	7.9661	11.0169	11.1864	14.5762
900	11.5254	10.8474	14.5762	17.7966	21.6949
1000	14.4067	14.9152	20.5084	25.5932	29.8305
1100	18.6440	19.8305	27.1186	33.3898	39.1525
1200	23.8983	26.1016	35.2542	42.7118	50.0000
1300	30.6779	31.6949	49.4915	56.6101	78.3050
1400	40.1694	41.5254	75.7627	89.6610	97.7966
1500	55.7627	56.6101	96.9491	98.3050	98.4745
1600	86.9491	83.2203	99.4915	99.6610	99.3220
1700	98.3050	96.6101	99.8305	99.6610	99.3220
1800	98.3050	99.1525	99.8305	99.6610	99.3220
1900	99.8305	100.0000	100.0000	100.0000	100.0000

Tabla A.4: Porcentaje global de descarga vs. tiempo, escenario con 80 nodos

Tiempo	% BitTorrent	%BitTorrent ad hoc	%BitTorrent RL	%CBt 3,1	%CBt 4,0
300	0.0000	0.0000	0.0000	0.0000	0.5063
400	0.0000	0.0000	0.0000	0.5063	1.2658
500	0.0000	0.2531	0.0000	1.6455	2.7848
600	0.1265	0.2531	0.6329	2.7848	4.6835
700	0.2531	1.0126	1.3924	4.3037	6.8354
800	0.6329	1.6455	2.0253	5.8227	8.1012
900	1.1392	3.5443	3.1645	7.4683	11.1392
1000	2.2784	4.9367	3.7974	8.6075	13.4177
1100	3.0379	6.2025	5.3164	10.2531	15.4430
1200	4.1772	7.8481	6.5822	11.7721	17.8481
1300	4.9367	9.7468	8.3544	14.1772	21.6455
1400	6.3291	10.6329	9.8734	16.8354	25.6962
1500	7.4683	13.0379	12.5316	20.2531	31.0126
1600	9.1139	14.3037	14.8101	23.0379	36.3291
1700	10.8860	15.9493	17.7215	28.6075	42.4050
1800	12.1518	19.3670	22.1518	33.6708	50.3797
1900	14.3037	24.8101	26.2025	40.0000	66.9620
2000	16.7088	28.3544	30.7594	48.8607	85.5696
2100	20.7594	35.9493	37.8481	62.1518	99.1139
2200	25.8227	43.4177	47.5949	83.6708	99.7468
2300	32.9113	58.7341	66.0759	98.3544	99.7468
2400	42.1518	79.2405	84.0506	98.7341	99.7468
2500	50.0000	93.4177	95.3164	100.0000	100.0000
2600	55.9493	99.2405	99.3670	100.0000	100.0000
2700	62.1518	99.7468	99.7468	100.0000	100.0000
2800	69.4936	100.0000	100.0000	100.0000	100.0000
2900	78.6075	100.0000	100.0000	100.0000	100.0000
3400	99.6202	100.0000	100.0000	100.0000	100.0000



Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE EXAMEN DE GRADO

No. 00069

Matrícula: 2153805364

Distribución de contenidos en redes ad hoc utilizando el paradigma P2P y conocimiento de redes sociales

En la Ciudad de México, se presentaron a las 11:30 horas del día 26 del mes de enero del año 2018 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DR. RICARDO MARCELIN JIMENEZ
DR. JAVIER GOMEZ CASTELLANOS
DR. MIGUEL LOPEZ GUERRERO



ERIC MARQUEZ SOLIS
ALUMNO

Bajo la Presidencia del primero y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRO EN CIENCIAS (CIENCIAS Y TECNOLOGIAS DE LA INFORMACION)

DE: ERIC MARQUEZ SOLIS

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobac

REVISÓ



LIC. JULIO CESAR DE LARA ISASSI
DIRECTOR DE SISTEMAS ESCOLARES

Acto continuo, el presidente del jurado comunicó al interesado el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

DIRECTOR DE LA DIVISIÓN DE CBI



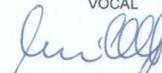
DR. JOSE GILBERTO CORDOBA HERRERA

PRESIDENTE



DR. RICARDO MARCELIN JIMENEZ

VOCAL



DR. JAVIER GOMEZ CASTELLANOS

SECRETARIO



DR. MIGUEL LOPEZ GUERRERO