

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/303959745>

Protocolo BitTorrent Ciente de Localização em Redes Corporativas

Conference Paper · May 2015

READS

9

2 authors, including:



Marcelo Gonçalves Rubinstein

Rio de Janeiro State University

58 PUBLICATIONS 496 CITATIONS

SEE PROFILE

Protocolo BitTorrent Ciente de Localização em Redes Corporativas

Marco Antonio D'Alessandro Costa¹, Marcelo Gonçalves Rubinstein²

¹ Petrobras – Petróleo Brasileiro S.A – RJ, Brasil

² Programa de Pós-Graduação em Engenharia Eletrônica - Universidade do Estado do Rio de Janeiro (UERJ), RJ, Brasil

marco.dalessandro@gmail.com, rubi@uerj.br

Resumo. *Distribuições de grandes volumes de dados por redes TCP/IP corporativas trazem problemas como a alta utilização da rede e de servidores, que podem ser reduzidos com utilização de redes par-a-par. Este artigo propõe uma modificação no comportamento padrão do protocolo BitTorrent, onde o rastreador identifica o endereço IP do par que está solicitando a lista de endereços IP do enxame e envia somente aqueles pertencentes à mesma rede local e ao semeador original, com o objetivo de reduzir o tráfego em redes de longa distância. Em um cenário corporativo típico, simulações mostraram que a alteração é capaz de reduzir o consumo médio de banda e o tempo médio dos downloads, quando comparados ao BitTorrent padrão.*

Abstract. *Distributions of large amounts of data over enterprise TCP/IP networks bring problems such as high network and server utilizations, which can be reduced with the use of peer-to-peer networks. This paper proposes a change in the default behavior of the BitTorrent protocol, where the tracker identifies the peer IP address requesting the list of IP addresses of the swarm and sends only those belonging to the same LAN and to the original seeder, in order to reduce traffic on wide area networks. In a typical enterprise scenario, simulations showed that the change is able to reduce the average bandwidth consumption and the average download time, when compared with the standard BitTorrent.*

1. Introdução

Hoje em dia, é cada vez mais comum a necessidade de transferência de grandes massas de dados (acima de 100 MB) por redes TCP/IP (*Transmission Control Protocol/Internet Protocol*), muitas vezes para mais de um cliente simultaneamente. Os dados vão desde arquivos de mídia (áudio e vídeo) até replicações de banco de dados, atualização de vacinas de programas de antivírus e sistemas operacionais. Devido ao grande volume de dados que envolvem estas transferências, podem ocorrer problemas específicos, que são detalhados a seguir:

- **alta utilização dos recursos de rede** – Arquiteturas do tipo cliente-servidor, em geral, têm limitações geradas pelas redes que conectam o cliente ao servidor, pois os recursos de rede são finitos. O acesso concorrente de vários clientes a um servidor só piora esta situação, provocando lentidão na transferência;

- **longo período de tempo para conclusão** – Atualizações de sistemas operacionais e, principalmente, de vacinas de antivírus devem ser realizadas o mais rápido possível, pois uma demora na atualização pode expor um computador, um servidor ou até mesmo toda uma rede a vulnerabilidades de segurança, como vírus, *worms* e ataques DDoS (*Distributed Denial-of-Service*);
- **maior sensibilidade a falhas** – Uma arquitetura cliente-servidor com apenas uma fonte de dados (ou com as fontes em um mesmo local) pode gerar problemas quanto à disponibilidade em caso de falha do servidor, já que é esperado que o envio dos dados demore um longo período de tempo;
- **necessidade de recuperação dos dados em caso de falha na transferência** – Em caso de alguma falha (do cliente, do servidor ou da rede), todos os dados já transferidos podem ser perdidos.

Os problemas citados são especialmente impactantes em redes corporativas, onde há uma busca contínua por redução de custos, segurança da informação e alta disponibilidade dos dados. Deste modo, uma distribuição de um grande volume de dados em uma rede corporativa, feita de maneira ideal, deve ser orientada a utilizar os recursos de rede de maneira eficiente, no menor tempo possível e ser tolerante a falhas. Há muito tempo, o problema das grandes transferências de dados já é discutido no ambiente de Internet. Uma das soluções propostas é utilizar as redes Par-a-Par (P2P).

Uma rede par-a-par é formada por pares que dividem a responsabilidade de prover serviço entre eles de um modo cooperativo. Os pares voluntariamente se juntam à rede e contribuem com recursos. Do ponto de vista técnico, as redes par-a-par somente requerem que os pares executem um *software*, e não requerem nenhum serviço do núcleo da Internet além da pilha TCP/IP. As redes par-a-par são, em princípio, auto-escaláveis. A operação de qualquer sistema de distribuição de conteúdo par-a-par depende de uma rede de pares e das conexões entre eles. Esta rede é formada em cima, e de maneira independente, de uma rede física e é chamada de rede de “sobreposição”, de acordo com Androutsellis-Theotokis e Spinellis (2004) e Passarella (2012). As redes par-a-par se diferenciam muito segundo o funcionamento, dependendo da rede de sobreposição escolhida, mas também em função das decisões de como realizar a descoberta de pares, do que compartilhar e como realizar a transmissão das informações.

O BitTorrent é uma rede par-a-par muito popular e amplamente utilizada na Internet para distribuição de arquivos, além de já ser utilizada em redes corporativas, segundo Ernesto Facebook (2010) e Ernesto Twitter (2010). Neste artigo, o BitTorrent é utilizado no estudo do comportamento das redes P2P em um cenário corporativo. Através de simulações, mostramos que, com o funcionamento padrão do BitTorrent, em um cenário onde há duas redes locais conectadas por um enlace de longa distância com pouca banda e alta latência, há uma redução gradativa do desempenho das transferências de arquivo na medida em que uma das redes locais tem mais pares que a outra. Porém, através de uma modificação no comportamento do rastreador, de maneira a forçar as trocas de dados nas redes locais, o desempenho melhora, tanto em termos da redução de utilização dos recursos de rede, quanto do tempo médio necessário para concluir os *downloads*.

O restante do texto está organizado da seguinte forma: na Seção 2, apresentamos o protocolo BitTorrent. Na Seção 3, detalhamos o cenário simulado. Os resultados são apresentados na Seção 4. Por fim, concluímos o trabalho na Seção 5.

2.BitTorrent

O BitTorrent é uma das mais populares redes P2P e que desperta muito interesse, como pode ser visto em Mansilha et al. (2010). Em Cohen (2003) e em Theory.org (2014), o protocolo é detalhado e, a seguir, explicaremos sucintamente o seu funcionamento. Na sua versão original, o protocolo BitTorrent utiliza uma arquitetura híbrida, onde um servidor central é utilizado para auxiliar os pares da rede a se encontrarem, ainda que as trocas de arquivo sejam realizadas apenas pelos pares.

A arquitetura do BitTorrent é detalhada na Figura 1.

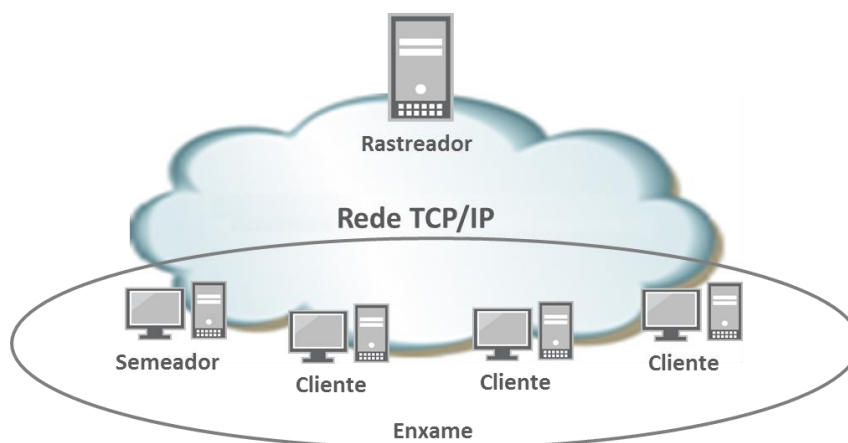


Figura 1. Arquitetura do protocolo BitTorrent.

Os componentes da arquitetura são:

- **clientes** – São os pares que participam da rede par-a-par e realizam o *download* e o *upload* do arquivo. Após obter todo o arquivo compartilhado, eles passam a atuar apenas como servidores do arquivo e, então, enquanto ainda estão na rede, são chamados de semeadores. O par que tem o primeiro arquivo completo na rede é chamado de semeador original;
- **enxame** – É como se denomina o conjunto de clientes que farão a troca do arquivo;
- **rastreador** – É o servidor que mantém a lista dos clientes que estão participando do enxame e a informa, total ou parcialmente, quando solicitado por um par. Segundo Theory.org (2014), o padrão é o envio de uma lista aleatória de 50 clientes. Se houver menos clientes registrados, ele enviará a lista de todos os clientes. Todo cliente para participar do enxame tem que se registrar no rastreador.

No protocolo BitTorrent, os arquivos que serão transferidos são primeiro divididos em partes menores, geralmente, com tamanhos de 256 kB, 512 kB, ou 1 MB, segundo Theory.org (2014).

Toda vez que algum cliente precisa obter um arquivo, ele primeiro precisa de um arquivo de “*torrent*”, tipicamente com extensão *.torrent*. Para obter os arquivos desejados, o cliente se registra no rastreador especificado no arquivo *torrent* e obtém dele a lista dos clientes que estão participando do enxame. Periodicamente, os clientes avisam ao rastreador que ainda estão presentes no enxame. O cliente então estabelece conexões TCP com os outros clientes para obter as partes dos arquivos. Se há apenas o semeador original, o cliente se conecta diretamente a ele para iniciar a transferência dos arquivos. Após acabar de receber os dados, se um cliente se mantiver conectado, ele passa a ser um semeador. Porém, assim que recebe a primeira parte de um arquivo listado no arquivo *torrent*, o cliente já passa a disponibilizar esta parte para os outros clientes, mesmo que ele tenha apenas essa parte. Para decidir quais partes devem ser solicitadas primeiro, os clientes utilizam uma estratégia chamada de *rarest first* (o mais raro primeiro). Periodicamente, eles pedem para os outros membros conhecidos do enxame a lista de partes que cada um tem e determinam quais partes são as mais raras; então cada cliente solicita a quem tem estas partes que inicie as transferências.

Quando uma solicitação de envio chega a um cliente, ele privilegia os quatro membros conhecidos do enxame que estão fornecendo as maiores taxas de transmissão com destino a ele mesmo. Esta estratégia é chamada de *tit-for-tat* (olho-no-olho). Em paralelo, ele também escolhe aleatoriamente outro membro que solicitou o envio para realizar a transmissão, sem nenhum outro critério de escolha. Esta estratégia de escolha aleatória é chamada de liberação otimista (*optimistic unchoking*). Quando um cliente vira semeador, ele mantém a política de liberação otimista, mas para as outras quatro conexões ele passa a privilegiar os clientes para os quais ele tem as maiores taxas de envio.

2.1. BitTorrent em Ambientes Corporativos

O BitTorrent pode ser utilizado com sucesso para melhorar distribuições de dados em redes corporativas, podendo distribuir arquivos de poucos megabytes até alguns gigabytes. A eficiência da utilização das distribuições através de redes P2P como o BitTorrent é apresentada em detalhes em Mundinger et al. (2008). Nesse trabalho, o tempo de transferência teórico de uma solução em arquitetura Cliente/Servidor é comparado com o da arquitetura P2P utilizada pelo BitTorrent. A análise mostra que quanto maior o número de nós e o número de partes de um arquivo, maior será a eficiência do protocolo BitTorrent quando comparada a de uma arquitetura Cliente/Servidor.

Uma desvantagem da distribuição de arquivos na arquitetura Cliente/Servidor é citada em Pushp e Ranjan (2010). Por exemplo, na Figura 2, para cada cliente que desejamos enviar o pacote de dados, temos no mínimo uma conexão TCP para o servidor. Cada conexão vai enviar dados pelo enlace que interliga o *data center*, onde o servidor está hospedado, à rede local, onde estão os Clientes. Porém, os dados enviados para cada cliente são idênticos, isto é, se envia uma cópia de cada informação para cada um dos clientes. Esta situação gera desperdício de banda nos enlaces e no servidor.

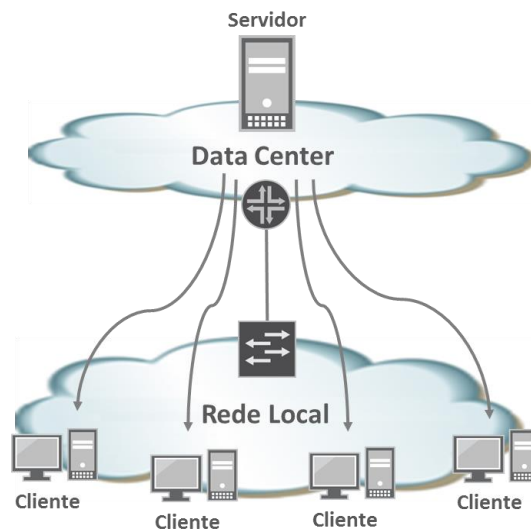


Figura 2. Downloads concorrentes geram uma carga maior nos servidores e nos enlaces da rede.

Na Figura 3, é apresentada uma estratégia mais eficiente utilizando o protocolo BitTorrent. Nele, o servidor que fará o *upload* da distribuição de dados age como um semeador e, com o auxílio de um servidor fazendo o papel de rastreador, envia os dados para os clientes. Porém, assim que os clientes completam o *download* da primeira parte solicitada ao servidor, passam a também ser fornecedores de dados da distribuição para os demais clientes, seguindo as estratégias definidas pelo protocolo BitTorrent. Isto acelera o processo de envio de dados e diminui a utilização do enlace entre a rede local e o *data center*.

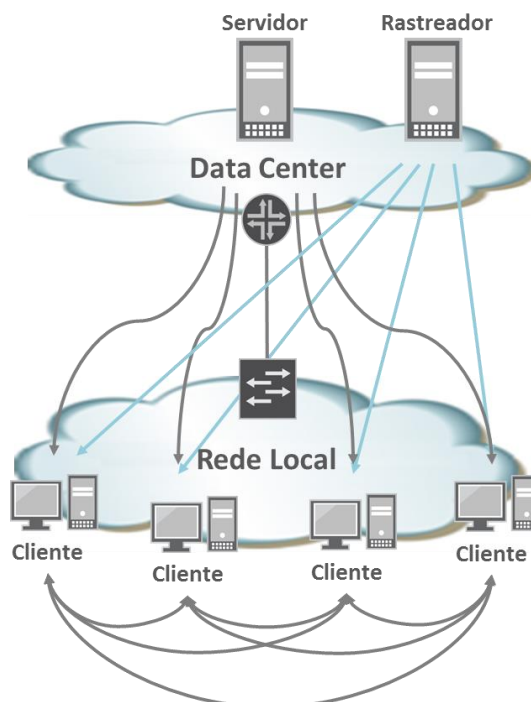


Figura 3. Com o protocolo BitTorrent, os clientes ajudam a acelerar a distribuição e reduzem o tráfego no enlace entre o *data center* e a rede local.

Apesar de o BitTorrent ter sido pensado para um ambiente de Internet, onde não há garantias de disponibilidade e banda, ele pode ser adaptado para ambientes corporativos, onde há mais banda disponível, principalmente em redes locais, baixa latência e os clientes tendem a ser mais confiáveis. Neste caso, até estratégia “olho-no-olho” pode ser repensada, pois pode haver “garantia” que os clientes ficarão na rede por tempo suficiente para realizar o compartilhamento dos dados, pois os *softwares* dos clientes podem ser executados em segundo plano com controle apenas do administrador da rede, dificultando o comportamento do tipo *free rider*.

Em Somani et al. (2012), se discutem algumas técnicas para otimizar as transferências de arquivos em ambientes corporativos, bem como uma avaliação de desempenho do BitTorrent em um cenário deste tipo. Simulações em ambientes similares a *data centers*, com banda de 1 Gbps, latência de poucos milissegundos e nas quais os pares ficam conectados à rede P2P durante todo o tempo, mostraram que a contribuição de *upload* para a rede P2P dos nós que fazem o *download* versus o semeador original varia pouco, ficando entre 75% e 125% do tamanho total do arquivo, em ambos os casos. Isto é, o semeador original, apesar de ficar mais tempo na rede com o arquivo completo, não possui uma contribuição muito maior do que as dos outros pares.

2.2. O Conceito de Localização no BitTorrent

Quanto se lida com a distribuição de conteúdo BitTorrent na Internet, um dos grandes problemas é a grande quantidade de tráfego nos enlaces inter-ISP (*Internet Service Providers* – Provedores de Serviços de Internet) ou entre os ASs (*Autonomous Systems* – Sistemas Autônomos) durante a distribuição. Mesmo que dentro de um ISP vários pares estejam realizando o *download* do mesmo conteúdo, isto é, estão no mesmo enxame, eles não estarão necessariamente conectados entre si. Como consequência, os pares vão realizar desnecessariamente o *download* de conteúdo de pares localizados fora do seu ISP. Os enlaces inter-ISP geram altos custos para os ISPs, então é interessante que o tráfego nestes enlaces seja baixo para diminuir a necessidade de ampliação. Segundo resultado de estudo apresentado em Oechsner et al. (2009), enxames típicos na Internet tem apenas de 1% a 10% de pares no mesmo AS, o que agrava o problema. O problema se repete em ambientes corporativos, quando os pares podem enviar e receber dados através de enlaces de longa distância, ao invés de dar preferência a pares que estão na mesma rede local.

Uma solução para este problema é usar o conceito de localização P2P, isto é, tentar conter o tráfego da rede par-a-par dentro do “local” onde os clientes estão. O “local” pode ser um AS ou ISP no caso da Internet ou um prédio, *data center*, rede local ou faixa de endereços IP no caso de uma rede corporativa. Em uma rede corporativa, a localização do tráfego reduz a utilização dos enlaces de longa distância e acelera o *download*, pois privilegia as trocas de dados na rede local, que, geralmente, possui maior banda. Por padrão, o BitTorrent não aplica nenhuma política de localização e usa uma política aleatória na escolha dos pares que serão informados pelo rastreador. A política de escolha de pares pode ser aplicada nos rastreadores ou nos próprios pares. Os sistemas de BitTorrent que se utilizam de alguma informação de localização para tomada de decisão durante o funcionamento (na escolha de pares, por exemplo) são chamados de sistemas de BitTorrent cientes de localização. Em [Cuevas et al. \(2014\)](#), é

apresentado um estudo sobre os efeitos da localização nas distribuições por BitTorrent na Internet, que mostra que para grandes e médios ISPs, as políticas cientes de localização produzem bons resultados, pois reduzem tanto o tráfego inter-ISP, quanto o tempo de *download*.

Alguns parâmetros podem ser utilizados para definir a localização de um cliente BitTorrent, como o AS ao qual o cliente pertence ou a sua localização geográfica. Neste artigo propomos que seja utilizada a sub-rede IP.

Podemos ver em Choffnes e Bustamante (2008) e Xie et al. (2008) propostas de como melhorar o compartilhamento de arquivos por BitTorrent reduzindo o tráfego inter-ISP. Porém são abordagens complexas, que precisam de modificações significativas nos clientes BitTorrent, que passam a usar informações de redirecionamento de CDNs para escolha orientada de pares ou de infraestrutura específica funcionando em paralelo com a rede BitTorrent.

Na nossa proposta, foi realizada apenas uma modificação simples no rastreador, de modo que quando um cliente BitTorrent solicita uma lista de pares do rastreador, ele verifica quais são os pares que estão na mesma sub-rede e envia uma lista aleatória contendo apenas estes pares da mesma sub-rede do cliente solicitante, com exceção do semeador original, que sempre é enviado para as redes às quais ele não pertence. Esta abordagem se aproveita do fato de que, em uma rede corporativa bem estruturada, as faixas de endereço estão relacionadas a localizações geográficas. Sem perda de generalidade, utilizamos para a definição da sub-rede a máscara 255.255.255.0 (/24); logo cada rede local tem uma sub-rede com esta máscara. Então, com a alteração, estamos forçando que o tráfego da distribuição se mantenha dentro das redes locais, com exceção do tráfego de controle dos protocolos e do semeador original. Fora a já citada simplicidade, esta abordagem oferece outra vantagem: como o endereço IP é inerente ao cliente, não é necessária configuração adicional no cliente para que protocolo se aproveite da ciência de localização.

Outro ponto interessante desta alteração é a importância que o semeador original tem para as redes. A modificação proposta segrega as redes de maneira que, se ocorrer alguma falha e o semeador original sair do enxame sem que haja todas as partes da distribuição em cada uma das redes, a distribuição não será concluída. Em um ambiente corporativo, isto não é problema, pois o semeador original pode estar hospedado em um sistema com alta disponibilidade, com proteção contra falhas de *software*, *hardware* e energia. Porém uma eventual falha de rede, que ocasione um isolamento de alguma rede do semeador original, pode parar a distribuição nesta rede, se todas as partes ainda não estiverem presentes. No funcionamento padrão do BitTorrent, assim que todas as partes são transmitidas para o enxame, o semeador original não é mais fundamental para que se conclua a distribuição; porém antes disso, a distribuição pode parar caso o semeador original fique isolado.

3. Cenário Simulado

Com o objetivo de estudar o desempenho de uma distribuição de dados (pacote de atualização de sistema operacional, por exemplo) utilizando o protocolo BitTorrent em ambientes corporativos, foi modelado no simulador de redes NS-3 (2014) um cenário típico de redes corporativas. Nesse cenário foram criadas uma rede local chamada de

Matriz e outra chamada de Filial, sendo essas redes conectadas através de um enlace que possui características de pequena banda e alta latência quando comparadas às das redes locais.

Foi utilizada uma implementação do BitTorrent para o NS-3, apresentada em Weingärtner et al. (2012). Além dos testes utilizando o comportamento padrão do rastreador na resposta a solicitação de pares do enxame, isto é, a seleção aleatória dos pares registrados, também foram realizados testes utilizando a seleção de pares por sub-rede no rastreador, conforme explicado na Seção 2.2.

O cenário utilizado está representado na Figura 4.

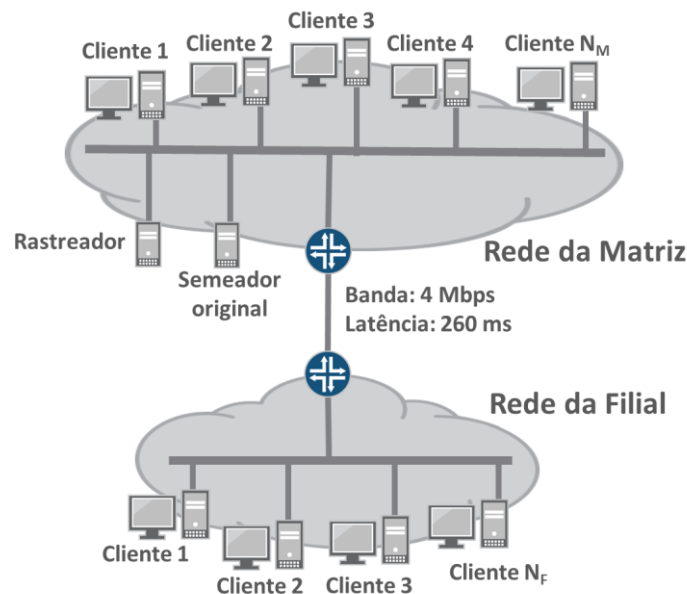


Figura 4. Cenário simulado com as redes locais Matriz e Filial.

A rede Matriz possui N_m nós onde são executados clientes BitTorrent, mais um semeador original, que também é um cliente, porém já possui o arquivo a ser distribuído. O semeador original faz o papel do servidor onde, a princípio, ficam armazenados os dados que devem ser distribuídos pelo protocolo BitTorrent. O rastreador responsável pela divulgação dos nós que pertencem ao enxame também se encontra na rede Matriz. Todos estes nós estão conectados a outro, que faz o papel de roteador, através de uma rede local com alta banda (1 Gbps) e baixa latência (menor que 1 ms).

A rede Filial possui N_f nós onde também são executados clientes BitTorrent. A princípio, não há nenhum nó que possua o arquivo completo na rede Filial. Os nós da rede Filial devem se registrar no rastreador da rede Matriz para conhecer o enxame. Assim como na rede Matriz, os nós da rede Filial estão conectados a um outro, que faz o papel de roteador, através de uma rede local com alta banda (1 Gbps) e baixa latência (menor que 1 ms).

As redes Matriz e Filial são interligadas através de um enlace entre os nós roteadores. Este enlace possui banda de 4 Mbps e 260 ms de latência. Estes parâmetros de banda e latência foram escolhidos para simular um cenário onde a Filial é um *site*

remoto, atendido por um enlace de longa distância via satélite, e com pouca banda disponível, como um posto de gasolina, mercado ou até uma plataforma de petróleo. O arquivo distribuído possui 10 Mbytes de tamanho. Todos os nós da Matriz e da Filial se registram no rastreador como uma *flash crowd*, isto é, assim que a simulação começa, todos os nós tentam se registrar ao mesmo tempo no rastreador.

Durante as simulações, o cenário varia da seguinte maneira: é mantido o valor de N_f em 20 nós e o valor de N_m varia em cada rodada de testes, sendo definido como 40, 60, 80 e 100 nós. Para cada rodada, são executadas 10 repetições. Para a avaliação de desempenho, utilizaremos as seguintes métricas:

- **tráfego da Matriz para Filial e o tráfego da Filial para a Matriz** (em Mbytes)
- É o tráfego que passa pelo enlace de 4 Mbps em cada sentido. Quanto menor, melhor, pois há economia de um recurso caro que é o enlace de longa distância entre a Matriz e a Filial;
- **tempo médio de conclusão do *download*** (em segundos) - Para os nós da Matriz e da Filial, se mede o tempo médio que os nós de cada uma das redes levam para concluir o *download*. Quanto menor, mais eficiente é a distribuição.

Os resultados médios das simulações com e sem a modificação do rastreador são comparados a seguir. Serão também mostrados os intervalos de confiança dos resultados médios com nível de confiança de 90%.

4. Resultados

Nas Figuras 5 e 6, é apresentado o volume de tráfego médio que transita entre a rede da Matriz e da Filial durante a transferência do arquivo, com e sem a modificação do rastreador.

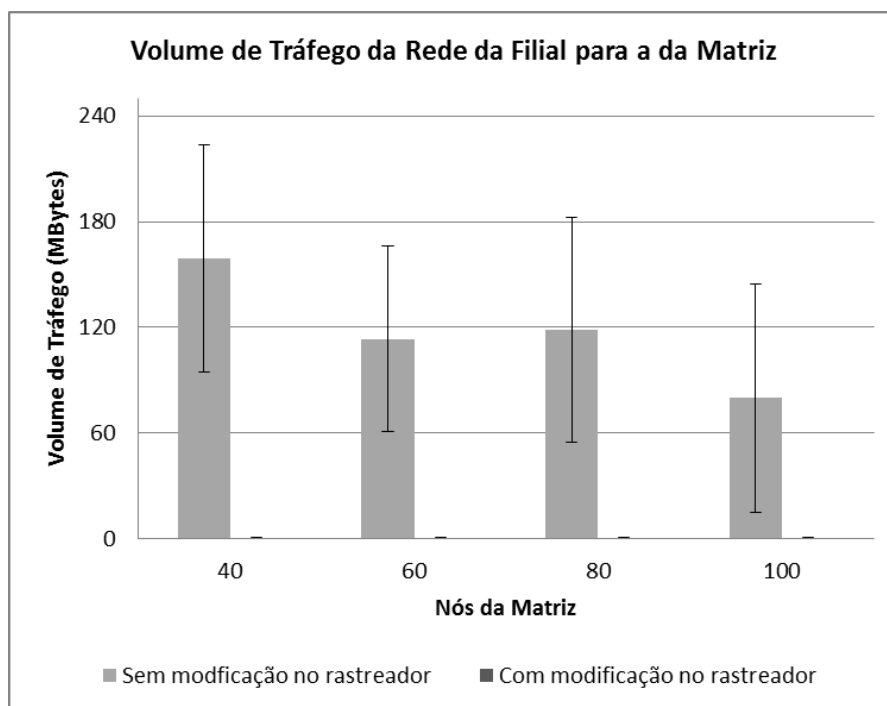


Figura 5. Volume de tráfego da rede da Filial para a da Matriz em função da quantidade de nós da Matriz.

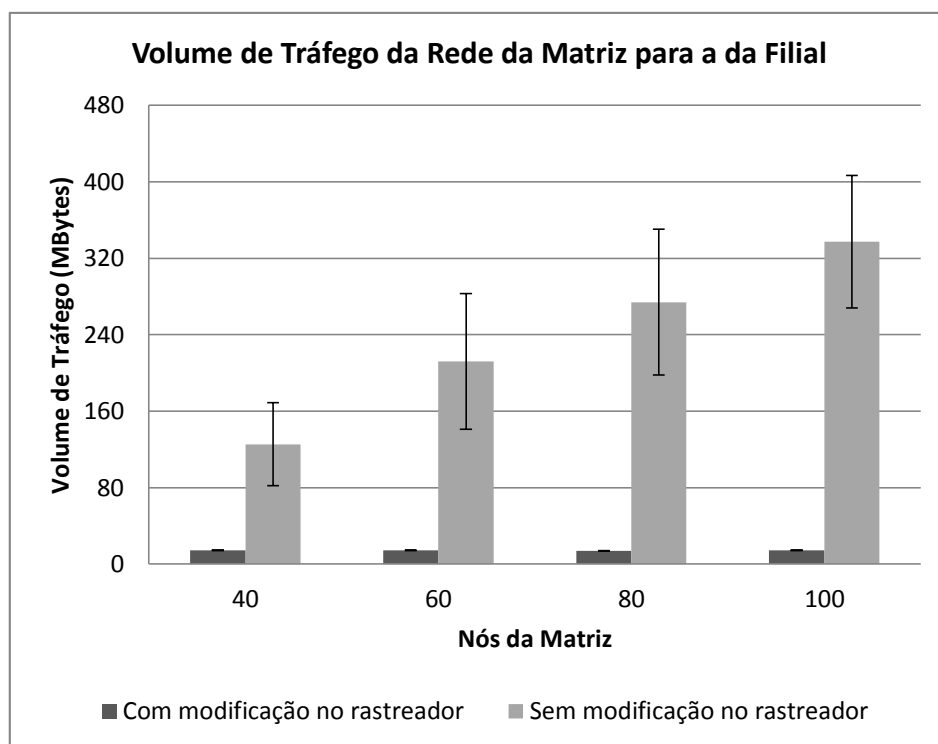


Figura 6. Volume de tráfego da rede da Matriz para a da Filial em função da quantidade de nós da Matriz.

Na Figura 5, podemos observar que, para o tráfego da Filial para a Matriz, sem a modificação do rastreador, o volume varia muito, produzindo intervalos de confiança grandes, mas com tendência de queda na medida em que se aumenta o número de nós da Matriz. Este comportamento pode ser explicado da seguinte forma: quanto maior o número de clientes na Matriz, maior é a probabilidade que eles sejam escolhidos na lista de pares aleatória que o rastreador envia aos clientes, tanto para os clientes da Filial, quanto para os clientes da própria Matriz. Então, há uma tendência de que os clientes da Matriz recebam mais solicitações de partes de ambas as redes, Matriz e Filial, na medida em que aumenta a quantidade de clientes na Matriz. Porém, quando o envio das partes ocorre dentro da própria rede Matriz, o tempo de *download* de cada parte é menor em comparação com as partes oriundas da rede Filial, pois há maior velocidade na rede local do que no enlace entre as redes. Desta forma, durante um mesmo intervalo de tempo, mais partes são enviadas de dentro da própria rede Matriz do que da rede Filial, reduzindo a contribuição total desta última e diminuindo a volume médio de bytes trafegados no sentido da Filial para a Matriz. Porém, como o processo de formação da lista de pares é aleatório, o volume de tráfego medido varia significativamente a cada rodada do experimento, produzindo grandes intervalos de confiança. Com a modificação no rastreador, o comportamento muda radicalmente. Praticamente, não há envio de informações para a Matriz, com o volume de, aproximadamente, 600 kBytes em média e um intervalo de confiança pequeno. Há apenas tráfego de controle para o rastreador e solicitações para o semente original.

Na Figura 6, confirmamos o comportamento estudado, pois, sem a modificação no rastreador, na medida em que aumentamos o número de clientes na Matriz, há

também um aumento no número de solicitações de partes para esta rede pelos clientes da Filial, o que provoca um aumento gradativo do volume médio de tráfego no sentido da Matriz para a Filial. O efeito da maior velocidade no *download* das partes na rede local da Filial não é suficiente para anular o efeito do aumento de solicitações para a rede Matriz. Mais uma vez, devido ao processo aleatório da formação da lista de pares, temos uma significativa variação no intervalo de confiança. Com a modificação no rastreador, os clientes da rede da Filial só conhecem a si mesmos e o semeador original, que fica na Matriz; e apenas os dados deste último são enviados pela rede de interligação entre a Matriz e a Filial. Na média, são enviados, aproximadamente, 14 MBytes, com *overhead* dos protocolos BitTorrent e TCP/IP, bem menos que os 200 MBytes necessários, se fosse utilizada a arquitetura cliente-servidor para o envio do arquivo aos 20 clientes da Filial. Neste caso, o intervalo de confiança é pequeno.

Nas Figuras 7 e 8, é apresentado o tempo médio necessário para a transferência do arquivo na rede da Matriz e na da Filial, com e sem a modificação do rastreador.

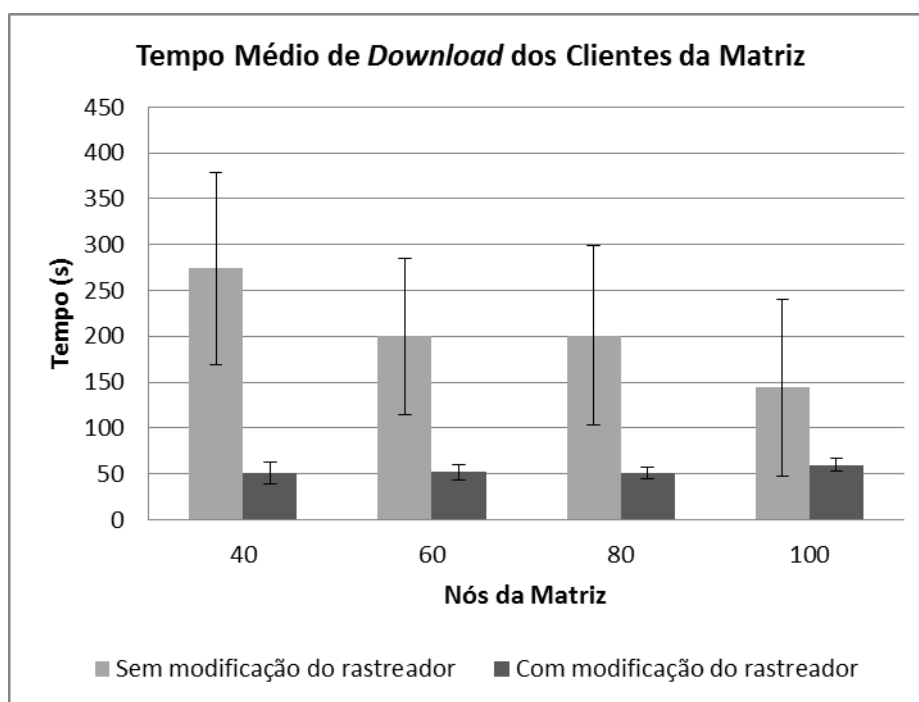


Figura 7. Tempo médio de *download* dos clientes da Matriz em função da quantidade de nós da Matriz.

Na Figura 7, vemos o impacto que a transferência de partes dos arquivos pelo enlace que liga as redes Matriz e Filial produz no tempo médio de *download*, pois, sem modificação no rastreador, vemos que a variação é semelhante a que ocorre na Figura 5. Isto é, o *download* dos arquivos pelos clientes da Matriz fica mais rápido na medida em que há mais clientes na Matriz e, conseqüentemente, menos dados sendo trafegados pelo enlace entre a Matriz e a Filial, que possui menor banda disponível. Novamente, vemos os grandes intervalos de confiança gerados pelo processo aleatório de criação da lista de pares pelo rastreador. Com a modificação no rastreador, não há envio de partes da Filial para a Matriz e, desta forma, não há gargalos que atrasem significativamente o *download* dos arquivos pelos clientes da Matriz. No gráfico, o tempo médio varia entre

50 e 60 s, aumentando lentamente na medida em que se aumenta o número de clientes da Matriz e com o intervalo de confiança pequeno.

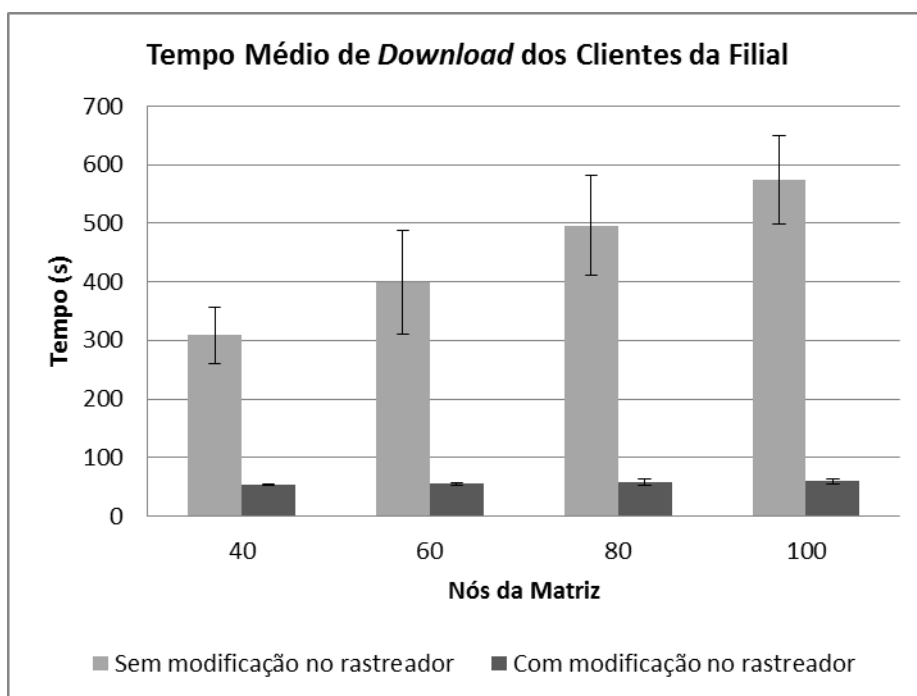


Figura 8. Tempo médio de *download* dos clientes da Filial em função da quantidade de nós da Matriz.

Na Figura 8, o comportamento é semelhante ao já analisado: sem a modificação no rastreador, o envio de dados pelo enlace entre a Matriz e a Filial aumenta o tempo médio necessário para a conclusão dos *downloads* pelos clientes da Filial. Então, quanto mais clientes houver na Matriz, mais lento será o *download* nos clientes da Filial. Com a modificação no rastreador, o tempo médio de *download* varia entre 53 e 60 s, também aumentando lentamente na medida em que se aumenta o número de clientes da Matriz e com o intervalo de confiança pequeno.

5. Conclusão

Neste trabalho, apresentamos resultados de simulações utilizando o protocolo BitTorrent em um cenário corporativo. Com o funcionamento padrão do protocolo BitTorrent, os resultados das simulações mostraram que na medida que se aumenta o número de clientes na Matriz em relação aos clientes da Filial há uma degradação no desempenho da distribuição. Com uma modificação no rastreador para que ele utilize uma estratégia ciente de localização para gerar a lista de pares, os resultados das simulações mudaram. Os resultados se mostraram mais estáveis, com intervalos de confiança menores, e com melhor eficiência na distribuição do arquivo, atingindo tempos médios de *download* menores e com menor utilização do enlace de longa distância. Outros estudos podem ser realizados para analisar cenários mais complexos, com várias filiais, por exemplo, ou com simulação de falha no enlace de longa distância, com a finalidade de verificar o quanto a estratégia de localidade afeta a confiabilidade da distribuição na ausência do semeador original.

Agradecimentos

Agradecemos o apoio da Faperj e do CNPq.

Referências

- Androutsellis-Theotokis, S. e Spinellis, D. (2004), “A survey of peer-to-peer content distribution technologies”, ACM Computing Surveys (CSUR), 36(4), 335-371.
- Cohen, B. (2003), “Incentives build robustness in BitTorrent”, Workshop on Economics of Peer-to-Peer systems (Vol. 6, páginas 68-72).
- Choffnes, D. R., e Bustamante, F. E. (2008), “Taming the torrent: a practical approach to reducing cross-ISP traffic in peer-to-peer systems”, ACM SIGCOMM Computer Communication Review (Vol. 38, No. 4, páginas 363-374), ACM.
- Cuevas, R., Laoutaris, N., Yang, X., Siganos, G., e Rodriguez, P. (2014), “BitTorrent Locality and Transit Traffic Reduction: When, Why, and at What Cost?”, IEEE Transactions on Parallel and Distributed Systems, 25(5), 1177-1189.
- Ernesto Facebook (2010), “Facebook Uses BitTorrent, and They Love It”, TorrentFreak, Junho, 2010, <http://torrentfreak.com/facebook-uses-bittorrent-and-they-love-it-100625/> [Acesso:11/2014]
- Ernesto Twitter (2010), “Twitter Uses BitTorrent For Server Deployment”, TorrentFreak, Fevereiro, 2010, <http://torrentfreak.com/twitter-uses-bittorrent-for-server-deployment-100210/> [Acesso:11/2014]
- Mansilha, R. B., Mezzomo, A., Facchini, G., Gaspary, L. P., e Barcellos, M. P. (2010), “Observando o Universo BitTorrent Através de Telescópios”, Anais do XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC), 2010, 233-246.
- Munding, J., Weber, R. e Weiss, G. (2008), “Optimal scheduling of peer-to-peer file dissemination”, Journal of Scheduling, 11(2), 105-120.
- NS-3 (2014), “NS-3”, NS-3 Consortium, 2012, <http://www.nsnam.org/> [Acesso:11/2014]
- Oechsner, S., Lehrieder, F., Hoßfeld, T., Metzger, F., Staehle, D. e Pussep, K. (2009). “Pushing the performance of biased neighbor selection through biased unchoking”, IEEE Ninth International Conference on Peer-to-Peer Computing, P2P'09, (páginas 301-310), IEEE.
- Passarella, A. (2012), “A survey on content-centric technologies for the current Internet: CDN and P2P solutions”, Computer Communications, 35(1), 1-32.
- Pushp, S. e Ranjan, P. (2010), “A practical incentive towards efficient data sharing in large scale enterprise”. Sixth International Conference on Networked Computing and Advanced Information Management (NCM), (páginas 45-50), IEEE.
- Somani, M., Swamiraj, M., Rengarajan, S. e Shankar, H. (2012), “BitTorrent for large package distribution in the enterprise environment”, International Conference on Recent Advances in Computing and Software Systems (RACSS), (páginas 281-286), IEEE.

Theory.org (2014), “Bittorrent Protocol Specification v1.0”, Theory.org, Julho, 2014, <https://wiki.theory.org/index.php/BitTorrentSpecification> [Acesso:11/2014]

Weingärtner, E., Glebke, R., Lang, M., e Wehrle, K. (2012), “Building a modular BitTorrent model for ns-3”, 5th International ICST Conference on Simulation Tools and Techniques (páginas 337-344), ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Xie, H., Yang, Y. R., Krishnamurthy, A., Liu, Y. G., e Silberschatz, A. (2008), “P4P: Provider portal for applications”, ACM SIGCOMM Computer Communication Review (Vol. 38, No. 4, páginas 351-362), ACM.