

Algoritmo Distribuído com abordagem em *cache* cooperativo

Pedro Paulo Simões Freitas, Ricardo Augusto Rabelo
PPGCC - Programa de Pós-Graduação em Ciência da Computação
UFOP - Universidade Federal de Ouro Preto
Ouro Preto, Minas Gerais, Brasil
email: pedropufop@gmail.com, rrabelo@gmail.com

Resumo—A utilização da técnica de *cache* é essencial para melhorar a recuperação de dados e desempenho da rede para usuários de dispositivos móveis. Neste artigo serão apresentados algoritmos distribuídos com abordagem em cooperação de *cache* e a análise de complexidade dos mesmos. Em se tratando de algoritmos distribuídos podemos perceber que uma melhor análise será em função da troca de mensagens. O DAJ será utilizado como ambiente de simulação, para que os testes sejam realizados.

Keywords—algoritmos distribuídos, cooperação de *cache*, complexidade, dispositivos móveis.

I. INTRODUÇÃO

Os algoritmos distribuídos são a base dos sistemas distribuídos. De acordo com [1], algoritmos distribuídos são definidos como uma coleção de processos interligados por uma rede de canais de comunicação.

Esses processos geralmente executam o mesmo algoritmo. Dependendo do sistema considerado, uma rede deste tipo pode consistir de conexões ponto-a-ponto, onde cada elo lida com o tráfego de comunicação entre dois processos, ou mesmo representado por grafos onde os vértices representam os processos e as arestas os canais de comunicação.

Nestes tipos de sistemas, os processos não compartilham qualquer tipo de memória. Assim a troca de informações entre os processos deve ser realizadas necessariamente por meio de mensagens. As trocas de mensagens são feitas pelos canais de comunicação existentes entre eles. De acordo com [2], o sentido da comunicação, pode ser:

- *Simplex*: permite o tráfego de mensagens em somente um sentido;
- *Half-Duplex*: permite tráfego de mensagens em ambos os sentidos, mas apenas um sentido por vez;
- *Full-Duplex*: permite tráfego de mensagens em ambos os sentidos simultaneamente.

Com isso vem a ideia do *cache* cooperativo. De acordo com [3], esta estratégia coloca cópias de documentos mais utilizados mais próximos dos clientes, diminuindo o tráfego na rede e a taxa de requisições recebidas pelos servidores de origem, assim ocorre uma redução do tempo de acesso. A utilização desses *caches* é uma boa opção para diminuir o número de requisições feitas a servidores remotos.

Com o grande aumento de dispositivos móveis que possuem várias interfaces de redes, existe a possibilidade de

um grande aumento do número de pessoas que podem estar conectadas a rede, por exemplo a internet. O cenário que analisaremos, considera que várias pessoas com algum dispositivo móvel, estes dispositivos tem acesso a alguma rede, incluindo uma rede de comunicação ad hoc, com acesso local entre as mesmas.

Dessa maneira, o *cache* é compartilhado entre os dispositivos móveis, introduzindo um novo cenário com algumas questões: Como achar a informação que o usuário necessita? É mais barato pesquisar com vizinhos ou ir direto ao servidor?

A outra situação envolve a criação de uma tabela de roteamento, para saber as localizações uns dos outro, e qual a validade. Com isso teremos outra pergunta, é mais barato manter uma tabela ou o usuário ser informado diretamente de quem tem a informação necessária, ou ainda a mesma questão citada anteriormente, acessar direto o servidor que está buscando.

Podemos considerar também que para acessar um servidor, os usuários podem utilizar uma rede 3G, a qual seria a interface de rede para ter uma conexão. A utilização desta interface para acessar o servidor é mais cara, em relação a acessar um vizinho ao lado que pode ter a informação. Este custo está relacionado ao gasto de bateria.

O artigo está organizado da seguinte forma. A Seção II apresenta a justificativa para estar estudando este sistema. Os objetivos são apresentados na Seção III. Na Seção IV é apresentada a metodologia utilizada. E por fim, na Seção V, será descrito quais resultados são esperados com esta pesquisa.

II. JUSTIFICATIVAS

Devido ao grande aumento da utilização de redes a partir de dispositivos móveis deve-se ter uma maior atenção com relação a complexidade dos algoritmos de roteamento, para tentar minimizar a troca de mensagens entre os membros da rede. Para buscar um menor tráfego de informações a fim de evitar uma sobrecarga na rede. Podemos aumentar a velocidade de acesso utilizando algumas estratégias. Assim o acesso seria bem mais rápido e gastaria menos bateria do dispositivo, já que este tem sua energia limitada.

A comparação destas estratégias se reduz a: qual o custo da troca de mensagens de maneira distribuída para localizar

a informação desejada.

III. OBJETIVOS

O objetivo deste trabalho será descrever o uso de *cache* cooperativo em algoritmos distribuídos. E realizar testes, para obtermos possíveis vantagens ou desvantagens da técnica.

Para fazer a análise dos algoritmos distribuídos, temos que levar em conta o tempo e espaço, e o número de mensagens trocadas. Já que entre estes algoritmos não há compartilhamento de memória e sua comunicação é realizada por mensagens. Cada processo sequencial é modelado com sua sequência de eventos, onde estes eventos podem ser *send(m)* e *receive(m)*, em que *m* é a mensagem.

De acordo com [1], *send(m)*, significa que a mensagem *m* é enfileirada em um canal de saída para ser transmitida ao processo de destino. E *receive(m)*, remove *m* de uma fila de entrada no processo de destino, e assim caso um processo recebe *m*, este verifica se esta mensagem é para ele, caso sim, o processo executa a mensagem, caso contrário ele a encaminha para frente.

Para os testes serão utilizados dois algoritmos. O primeiro será um clássico que é o Propagação da Informação com *Feedback* (PIF), extraído de [1]. Este algoritmo funciona da seguinte maneira, um nó *s* envia uma mensagem para todos os outros nós vizinhos. O fim da execução acontece quando o nó *s*, estiver recebido uma confirmação de recebimento da mensagem dos nós vizinhos, um *feedback*. Um segundo algoritmo seria o *Distributed Group-Based Cooperative Caching* (DGCOCA) apresentado em [4], este apresenta um algoritmo estável de descoberta de vizinho, assim os nós mantêm uma relação de amizade eles. Quanto mais mensagens maior o grau do relacionamento.

IV. METODOLOGIA

Para realizarmos os estudos referentes a este trabalho, será feito um modelo de algoritmo distribuído que utiliza *cache* cooperativo. Iremos executar este modelo no ambiente de simulação DAJ (*Distributed Algorithms in Java*). De acordo com [5], o DAJ é um *framework* para escrever programas em Java que implementam algoritmos distribuídos. Estes programas mostram as estruturas de dados de cada nó, para o nosso caso serão os dispositivos móveis, e permitir que o uma construção interativa do cenário. Em uma situação de aprendizagem, a execução interativa é preferível, pelo fato de poder assistir a simulação. A partir dessa simulação será obtido o número de mensagens trocadas entre os processos.

V. RESULTADOS ESPERADOS

Os resultados esperados serão as complexidades dos algoritmos analisados. E a partir dos resultados obtidos espera-se propor melhorias em algoritmos já existentes ou mesmo criar um novo algoritmo, com um melhor desempenho.

REFERÊNCIAS

- [1] V. C. Barbosa, *An Introduction to Distributed Algorithms*. The MIT Press, 1996, ISBN 0-262-02412-8.
- [2] A. S. Tanenbaum, *Redes de Computadores*, 4th ed. Campus, 2003, ISBN: 8535211853.
- [3] R. F. Brandão and R. de Oliveira Anido, "Cooperação entre caches para web," 1999.
- [4] C.-Y. Chow, H. V. Leong, and A. T. S. Chan, "Distributed group-based cooperative caching in a mobile broadcast environment," in *Proceedings of the 6th international conference on Mobile data management*, ser. MDM '05. New York, NY, USA: ACM, 2005, pp. 97–106.
- [5] M. Ben-Ari, "Interactive execution of distributed algorithms," *J. Educ. Resour. Comput.*, vol. 1, August 2001.