Centro Federal de Educação Tecnológica de Minas Gerais
Departamento de Computação
Engenharia de Computação

Mariane Raquel Silva Gonçalves

Comparação de algoritmos paralelos de ordenação em MapReduce

Orientadora: Profª. Drª. Cristina Duarte Murta

Belo Horizonte
2012

Centro Federal de Educação Tecnológica de Minas Gerais
Departamento de Computação
Engenharia de Computação

Mariane Raquel Silva Gonçalves

Comparação de algoritmos paralelos de ordenação em MapReduce

Projeto apresentado como requisito parcial à aprovação em Trabalho de Conclusão de Curso I do curso de Engenharia de Computação do CEFET-MG.

Orientadora: Profª. Drª. Cristina Duarte Murta

Belo Horizonte
2012

Centro Federal de Educação Tecnológica de Minas Gerais
Departamento de Computação
Engenharia de Computação

Comparação de algoritmos paralelos de ordenação em MapReduce

Trabalho de Conclusão de Curso I

Mariane Raquel Silva Gonçalves
Aluna

Prof[a]. Dr[a]. Cristina Duarte Murta
Orientadora

Aos professores,
aos colegas de curso e
aos meus familiares,
dedico este trabalho.

Agradecimentos

Agradeço aos meus pais, pelo amor incondicional.
Aos meus professores, pelos conhecimentos adquiridos.
E finalmente aos colegas de curso pela convivência e trocas de experiências.

"Pshaw, my dear fellow, what do the public, the great unobservant public, who could hardly tell a weaver by his tooth or a compositor by his left thumb, care about the finer shades of analysis and deduction!"

Sherlock Holmes, in "The Adventure of the Copper Beeches"

Resumo

O presente trabalho tem como objetivo criar um sistema de monitoramento escalável e robusto para telefonia móvel , utilizando *smartphones* com o sistema operacional *Android*. A ideia é agregar à funcionalidade de rastreamento, a possibilidade de execução comandos de forma remota, criando assim uma gama maior de aplicações para o sistema. Um exemplo interessante, seria no caso de roubo. Atualmente as pessoas costumam guardar informações pessoais e sigilosas em seu celular, portanto seria de extremo interesse que uma vez confirmado a perda do aparelho ela pudesse apagar estas informações.

Palavras-chave: *Android*, *Pattern: Command*, *Java*, *Google App Engine*, *Computação em Nuvens*,*Rastreamento*, *Telefonia móvel*.

Abstract

This paper aims to create a scalable and robust tracking system for the mobile phone network, using *smartphones* with the *Android* operational system. The ideia is to add a functionality of execution of remote commands to the tracking system, thus creating a wider range of applications and possibilities. An sample of those new features would be in the theft case, because with this functionality the cellphone owner would easily erase his private information such as contacts, messages and data.

Keywords: Android, Pattern: Command, Java, Google App Engine, Cloud Computing, Tracking, mobile network

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 Definição do Problema

As tendências atuais em design de microprocessadores estão mudando fundamentalmente a maneira que o desempenho é obtido a partir de sistemas computacionais. A indústria declarou que seu futuro está em computação paralela, com o aumento crescente do número de núcleos dos processadores [Asanovic et al. 2009]. O modelo de programação *single core* (sequencial) está sendo substituído rapidamente pelo modelo *multi-core* (paralelo), e com isso surge a necessidade de escrever software para sistemas com multiprocessadores e memória compartilhada [Ernst et al. 2009].

Arquiteturas *multi-core* podem oferecer um aumento significativo de desempenho sobre as arquiteturas de núcleo único, de acordo com as tarefas paralelizadas. No entanto, muitas vezes isto exige novos paradigmas de programação para utilizar eficientemente a arquitetura envolvida [Prinslow 2011]. A ordenação é um exemplo de tarefa que pode ter seu desempenho melhorado com o uso de paralelismo. A ordenação paralela é o processo de reorganizar uma sequência de entrada e produzir uma saída ordenada de acordo com um atributo através de múltiplas unidades de processamento, que ordenam em conjunto a sequência de entrada.

Uso crescente de computação paralela em sistemas computacionais gera a necessidade de algoritmos de ordenação inovadores, desenvolvidos para dar suporte a essas aplicações.

1.2 Motivação

O limite físico de aumento na frequência de operação dos processadores levou a indústria de hardware a substituir o processador de núcleo único por vários processadores eficientes em um mesmo *chip*, os processadores *multi-core*. É preciso, então, criar aplicações que utilizem efetivamente o crescente número de núcleos dos

processadores [Asanovic et al. 2009].

Um grande número de aplicações paralelas possui uma fase de computação intensa, na qual uma lista de elementos deve ser ordenada com base em algum de seus atributos. Um exemplo é o algoritmo de Page Rank [Page et al. 1999] da Google: as páginas de resultado de uma consulta são classificadas de acordo com sua relevância, e então precisam ser ordenadas de maneira eficiente [Kale e Solomonik 2010].

Na criação de algoritmos de ordenação paralela, é ponto fundamental ordenar coletivamente os dados de cada processo individual, de forma a utilizar todas as unidades de processamento e minimizar os custos de redistribuição de chaves entre os processadores. Fatores como movimentação de dados, balanço de carga, latência de comunicação e distribuição inicial das chaves são considerados ingredientes chave para o bom desempenho da ordenação paralela, e variam de acordo com o algoritmo escolhido como solução[Kale e Solomonik 2010]. No exemplo do Page Rank, o número de páginas a serem ordenadas é enorme, e elas são recolhidas de diversos servidores da Google; é uma questão fundamental escolher algoritmo paralelo com o melhor desempenho dentre as soluções possíveis.

O MapReduce[Dean e Ghemawat 2008] é um modelo de programação paralela criado pela Google para processamento de grandes volumes de dados em *clusters*. Esse modelo propõe simplificar a computação paralela e ser de fácil uso, abstraindo conceitos complexos da paralelização - como tolerância a falhas, distribuição de dados e balanço de carga - e utilizando duas funções principais: map e reduce. A complexidade do algoritmo paralelo não é vista pelo desenvolvedor, que pode se ocupar em desenvolver a solução proposta. O Hadoop [White 2009] é uma das implementações do MapReduce, um *framework open source* desenvolvido por Doug Cutting em 2005 que provê o gerenciamento de computação distribuída.

O desenvolvimento de soluções capazes de lidar com grandes volumes de dados é uma das preocupações atuais, tendo em vista a quantidade de dados processados diariamente, e o rápido crescimento desse volume de dados. Não é fácil medir o volume total de dados armazenados digitamente, mas uma estimativa da IDC colocou o tamanho do "universo digital"em 0,18 zettabytes em 2006, e previa um crescimento dez vezes até 2011 (para 1,8 zettabytes). *The New York Stock Exchange* gera cerca de um terabyte de novos dados comerciais por dia. O Facebook armazena aproximadamente 10 bilhões de fotos, que ocupam mais de um petabyte.

The Internet Archive armazena aproximadamente 2 petabytes de dados, com aumento de 20 terabytes por mês [White 2009]. Estima-se que dados não estruturados são a maior porção e de a mais rápido crescimento dentro das empresas, o que torna o processamento de tal volume de dados muitas vezes inviável.

MapReduce e sua implementação *open source* Hadoop representam uma alternativa economicamente atraente oferecendo uma plataforma eficiente de computação distribuída para lidar com grandes volumes de dados e mineração de petabytes de informações não estruturadas [Cherkasova 2011].

Devido ao grande número de algoritmos de ordenação paralela e variadas arquiteturas paralelas, estudos experimentais assumem uma importância crescente na avaliação e seleção de algoritmos apropriados para multiprocessadores.

1.3 Objetivos

Os objetivos deste trabalho são:

- Estudar a programação paralela aplicada à algoritmos de ordenação;
- Implementar um ou mais algoritmos de ordenação paralela no modelo MapReduce, com o software Hadoop;
- Comparar duas ou mais implementações de algoritmos paralelos de ordenação.

O trabalho desenvolvido por [Pinhão 2011] apresentou um estudo sobre a computação paralela e algoritmos de ordenação no modelo MapReduce, através da implementação do algoritmo de Ordenação por Amostragem feita em ambiente Hadoop.

Este projeto busca continuar o estudo sobre ordenação paralela feito no trabalho citado, com a da análise de desempenho de dois ou mais algoritmos de ordenação - sendo um deles o algoritmo de ordenação por amostragem. A análise busca compará-los com relação à quantidade de dados a serem ordenados, variabilidade dos dados de entrada e número máquinas utilizadas.

1.4 Estado da Arte

1.5 Organização

2 Referencial Teórico

2.1 Computação Paralela

2.2 Ordenação Paralela

2.3 Algoritmos

2.3.1 Sample Sort

2.3.2 Quick Sort

3 Metodologia

3.1 Recursos necessários

3.2 Desenvolvimento //

3.3 Descrição dos experimentos

4 Resultados Preliminares

5 Conclusões e Propostas de Continuidade

Referências

[Asanovic et al. 2009]ASANOVIC, K. et al. A view of the parallel computing landscape. *Commun. ACM*, ACM, New York, NY, USA, v. 52, n. 10, p. 56–67, out. 2009. ISSN 0001-0782.

[Cherkasova 2011]CHERKASOVA, L. Performance modeling in mapreduce environments: challenges and opportunities. In: *ICPE'11*. [S.l.: s.n.], 2011. p. 5–6.

[Dean e Ghemawat 2008]DEAN, J.; GHEMAWAT, S. Mapreduce: simplified data processing on large clusters. *Commun. ACM*, ACM, New York, NY, USA, v. 51, n. 1, p. 107–113, jan. 2008. ISSN 0001-0782.

[Ernst et al. 2009]ERNST, D. et al. Preparing students for ubiquitous parallelism. In: *SIGCSE*. [S.l.: s.n.], 2009. p. 136–137.

[Kale e Solomonik 2010]KALE, V.; SOLOMONIK, E. Parallel sorting pattern. In: *Proceedings of the 2010 Workshop on Parallel Programming Patterns*. New York, NY, USA: ACM, 2010. (ParaPLoP '10), p. 10:1–10:12. ISBN 978-1-4503-0127-5.

[Page et al. 1999]PAGE, L. et al. *The PageRank Citation Ranking: Bringing Order to the Web*. 1999.

[Pinhão 2011]PINHÃO, P. de M. *Ordenação Paralela no Ambiente Hadoop*. 2011.

[Prinslow 2011]PRINSLOW, G. *Overview of Performance Measurement and Analytical Modeling Techniques for Multi-core Processors*. 2011.

[White 2009]WHITE, T. *Hadoop: The Definitive Guide*. first edition. O'Reilly, 2009. Disponível em: <http://oreilly.com/catalog/9780596521981>.