

Universidade Federal de Pernambuco
Centro de Informática
Graduação em Ciência da Computação

Um algoritmo eficiente em tempo e espaço para o problema do ancestral comum mais profundo em árvores e grafos acíclicos

Proposta de Trabalho de Graduação

Aluno: Daniel Cândido Cauás
(dcc4@cin.ufpe.br)
Orientador: Paulo Gustavo Soares da Fonseca
(paguso@cin.ufpe.br)
Área: Teoria da Computação / Algoritmos

Recife, Março de 2019

Resumo

Um problema algorítmico recorrente em áreas diversas como Engenharia de Software ou Biologia Computacional, é o de encontrar o ancestral comum mais profundo (*Lowest Common Ancestor*—LCA) em árvores e/ou grafos dirigidos. As principais soluções existentes consistem em pré-processar a árvore para facilitar a consulta, e portanto, são avaliadas em termos do tempo e espaço de pré-processamento e de consulta. O objetivo de desse trabalho é propor uma solução simples para consultas de LCAs (queries) sucessivas e avaliar o seu desempenho em termos teóricos e práticos com respeito ao tempo de pré-processamento, consulta e uso de memória, em comparação a algumas soluções consagradas encontradas na literatura.

Introdução

Em ciência da computação, as árvores são estruturas ubíquas, utilizadas nas mais variadas aplicações, devido à sua versatilidade. As árvores formam a base de implementação de diversas estruturas de dados de várias linguagens de programação. Na área de Engenharia de Software, por exemplo, são usadas para modelar a herança em linguagens orientadas a objetos. Na área de Biologia Computacional, são usadas para representar árvores evolutivas. Nessas e em outras aplicações, um problema recorrente é o de determinar o ancestral comum mais profundo entre dois nós, em Inglês *Lowest common ancestor*—LCA.

Quando a árvore de entrada é estática, uma solução frequentemente utilizada consiste em pré-processá-la para construir estruturas auxiliares que permitem consultas posteriores sem a necessidade de percorrer novamente a árvore. Bender e coautores [1] apresentam algoritmos para o problema LCA baseados nessa abordagem, analisando-as em termos de tempo de pré-processamento e consulta. A solução ótima apresentada possui complexidade $\langle \Theta(n), \Theta(1) \rangle$, isto é, em tempo/espço linear de pré-processamento e tempo constante de consulta.

Objetivos

Este trabalho tem por objetivo geral analisar o desempenho prático relativo de diferentes algoritmos para o problema LCA. Especificamente, além de algoritmos descritos na literatura, nós pretendemos propor otimizações práticas para estruturas de dados resultantes do pré-processamento da árvore com o objetivo de obter uma solução eficiente em termos de espaço e tempo, permitindo ainda que as consultas sejam respondidas em tempo ótimo ou sub-ótimo na maioria dos casos, conforme verificado em estudos preliminares já realizados.

Metodologia

O projeto será organizado nas atividades descritas abaixo e desenvolvidas conforme o cronograma a seguir.

T0. Preparação da proposta

Nesta fase inicial, aluno e orientador farão uma série de reuniões para definição do problema e escopo do projeto. O orientador apresentará a bibliografia básica sobre o tema e os possíveis pontos a serem abordados, e ambos decidirão sobre aqueles a serem desenvolvidos com base no interesse mútuo.

T1. Revisão e acompanhamento bibliográfico

Essa tarefa será executada de maneira mais acentuada no início do projeto. A atividade consiste num estudo dirigido centrado no material bibliográfico mais diretamente relacionado às estruturas de dados e algoritmos a serem implementados no projeto. Ao final dessa fase inicial, espera-se que o aluno possa manter-se atualizado e aprofundar-se em pontos específicos de maneira mais autônoma.

T2. Implementação das estruturas

Esta tarefa corresponde ao principal componente de desenvolvimento do projeto. O aluno deverá, em interação com o orientador, estudar detalhadamente algoritmos e estruturas de dados relativos a

árvores, LCA, sparse tables, entre outros, e desenvolver uma implementação de referência em nível de produção para os mesmos, em C++.

T3. Realização dos Testes

Nesta tarefa, o aluno deverá fazer uma análise experimental comparativa do desempenho do método proposto e implementado relativamente a outras alternativas identificadas num levantamento inicial na tarefa T1. Pode ser necessário obter dados reais e produzir dados sintéticos que simulem uma situação limite de estresse para os métodos.

T4. Redação e revisão da monografia

A monografia produzida deverá conter: (1) uma breve revisão bibliográfica do estado da arte, fruto da tarefa T1, (2) descrição detalhada dos seus desenvolvimentos, incluindo a análise teórica dos algoritmos e estruturas propostas em termos de tempo e espaço, (3) uma análise crítica com base em resultados experimentais da tarefa T3, e (4) uma discussão com conclusões gerais sobre o projeto.

T5. Preparação da Apresentação

Finalmente, o aluno preparará a apresentação da defesa do TG com o resumo dos desenvolvimentos e resultados obtidos.

O aluno já possui familiaridade com a área, portanto poderá por-se rapidamente em desenvolvimento. O acompanhamento será feito pessoalmente através de reuniões semanais. Todo material desenvolvido é compartilhado entre orientador e aluno através de um repositório privado na plataforma GitHub.

Cronograma de atividades

	Mar		Abr				Mai				Jun				Jul	
Preparação da proposta	●	●														
Revisão bibliográfica [†]	●	●	○	○	○	○	○	○								
Implementação das estruturas			●	●	●	●	●	●	●	●	●					
Realização dos testes [‡]				○	○	○	○	○	○	●	●	●				
Redação e revisão da monografia											●	●	●	●		
Preparação da apresentação															●	●

([†]) ● = levantamento inicial, ○ = aprofundamento

([‡]) ● = experimentos formais, ○ = testes *ad hoc* unitários

Referências

- [1] Michael A. Bender, Martín Farach-Colton, Giridhar Pemmasani, Steven Skiena, and Pavel Sumazin. Lowest common ancestors in trees and directed acyclic graphs. *Journal of Algorithms*, 57(2):75–94, nov 2005.

Possíveis Avaliadores

1. Prof. Silvio Melo
2. Prof. Gustavo Carvalho

Assinaturas

Recife, 13 de Março de 2019

Aluno: Daniel Cândido Cauás

Orientador: Paulo Gustavo Soares da Fonseca