Ciência da Computação / Sistemas de Informação

ISOMORFISMO DE SUBGRAFOS EM CONJUNTOS DE DADOS DE GRAFOS : METODOLOGIA EXPERIMENTAL PARA OTIMIZAÇÃO DO DESEMPENHO

Hélio Henrique Medeiros Silva - 7° módulo de Ciência da Computação, UFLA

Vinícius Vitor dos Santos Dias - professor do Departamento de Ciência da computação - DCC, UFLA. - Orientador(a)

Resumo

Grafos são estruturas matemáticas utilizadas para modelar informação em diversos domínios, como redes sociais, biológicas, químicas, etc. Isomorfismo de Subgrafos é um problema cujo objetivo é determinar se um dado grafo H é isomorfo (equivalente em estrutura) a algum subgrafo de outro grafo G. Esse problema está presente em uma tarefa importante em bancos de dados: dados um grafo H e uma coleção de grafos G1,...,Gn, determinar quais grafos da coleção possuem subgrafos isomorfos a H. Essa tarefa é intensiva computacionalmente e por isso, a computação paralela é essencial para sua escalabilidade. O objetivo deste trabalho foi identificar os parâmetros de execução ideais de forma a maximizar a eficiência da paralelização. Este trabalho oferece as seguintes contribuições: (1) uma metodologia para análise experimental; (2) uma ferramenta que automatiza essa metodologia; e (3) um estudo de caso em conjuntos de dados reais de diferentes domínios. (1) A partir de um dataset, selecionamos aleatoriamente um grafo, extraímos dele um caminho com K vértices, e adicionamos todas as arestas de cada vértice deste caminho. Este será nosso grafo H. (2) Em seguida, criamos um método que verifica o isomorfismo entre H e cada grafo de um dataset. Para utilizar esse método de forma paralela, dividimos o dataset em batches de tamanho M=5%, 15%, 25%, e chamamos o método de forma que cada um dos N processos criados processem exatamente 1 batch por vez, até que todo o dataset seja computado. Posteriormente, criamos uma ferramenta para analisar o desempenho das execuções. A ferramenta calcula o tempo médio das execuções para cada configuração de entrada e gera métricas de desempenho e eficiência que permitem identificar a parametrização (N e M) ideal para a paralelização. (3) Como estudo de caso, selecionamos 3 datasets: dois representando coleções de moléculas (Mutag, BZR) e um no campo de visão computacional (Cuneiform). Ao todo, foram gerados 9 grafos H para cada caso, variando o parâmetro K. Considerando o tempo como a principal variável, a configuração N=8 e M=25 apresentou o melhor desempenho. No entanto, ao considerar o uso eficiente dos recursos computacionais, a configuração N=4 e M=25 foi superior na maioria dos casos, alcançando uma relação ótima entre eficiência e tempo. Concluímos que aumentar N não necessariamente melhora a eficiência, embora reduza o tempo. O tamanho dos batches também impacta a eficiência, variando conforme o dataset.

Palavras-Chave: paralelização, computação paralela, Otimização.
Link do pitch: https://youtu.be/TMHb-txxwUM