Algoritmos de Aproximação
Problema do Caixeiro Viajante

Willian Muniz do Nascimento[1]

[1] Programa de Pós-Graduação em Computação Aplicada (PPGCA)
DAINF - Universidade Tecnológica Federal do Paraná (UTFPR)

willianmuniz@ymail.com

Resumo. *Este artigo apresenta a aplicação de um algoritmo de aproximação no problema do caixeiro viajante, porém em um caso específico onde é respeitado o triangulo de desigualdade. Também é apresentado a aplicação ao problema de cobertura de vértices com cardinalidade.*

1. Introdução

Problemas NP-completos em sua maioria são muito importantes para serem ignorados somente por não haver uma solução ótima em tempo polinomial. Temos como uma alternativa de solução os algoritmos de aproximação. Cormen, Thomas H (2009, p.1106).

Os algoritmos de aproximação não trazem uma solução ótima, no entanto alcançam uma solução boa o suficiente em tempo polinomial para um problema NP-Completo.

2. Revisão teórica

Na revisão teórica será apresentado o que é uma relação de aproximação, definição de problemas P, NP e NP-Completo, além do problema Cobertura de vértices com cardinalidade.

A classe de problemas P são aqueles que contêm solução ótima em tempo polinomial. Os problemas que podem ser verificados em tempo polinomial e que ainda não foram encontrados algoritmos para resolvê-los em tempo polinomial são chamados de NP. Também há os problemas NP-Completos, nesta classe de problemas ao encontrar solução para um deles, todos poderão ser resolvidos em tempo polinomial.

2.1. Relação de aproximação

Em problemas que as soluções possíveis tenham custo positivo, buscando o custo máximo ou custo mínimo um algoritmo para um problema tem uma relação de aproximação de $p(n)$ se para qualquer entrada de tamanho n, o custo C de uma solução produzida pelo algoritmo está dentro de um fator de $p(n)$ do custo C* de uma solução ótima:

$$\max\left(\frac{C}{C^*}, \frac{C^*}{C}\right) \leq p(n)$$

Se um algoritmo alcança uma relação de aproximação de $p(n)$, chamamos este de $\boldsymbol{p(n)}$**-algoritmo de aproximação.** Para o problema de maximização, $0 < C \leq C^*$, e a relação C^*/C dá o fator pelo qual o custo da solução ótima é maior do que o custo da solução aproximada. De forma similar, para um problema de minimização, $0 < C^* \leq C$ e a relação C/C^*dá o fator pelo qual o custo da solução aproximada é maior que o custo da solução ótima.

A relação de aproximação nunca é menor que 1, um algoritmo com *1-algoritmo de aproximação* produz uma solução ótima, e caso essa relação de aproximação for muito grande pode retornar uma solução severamente pior que a ótima. Cormen, Thomas H (2009, p.1106-1107)

Quando estamos tratando de problemas NP-Completude, uma meta razoável é procurar uma relação de aproximação menor possível. Mas isto pode ser um pouco enigmático, visto que precisamos chegar próximo a solução ótima sem poder determinar o a solução ótima. Papadimitriou, Christos, and Umesh Vazirani (2009, p.275).

2.2. Cobertura de vértices com cardinalidade

Dado um grafo $H = (U, F)$, um subconjuntos de arestas $M \subseteq F$ é *correspondente* se não há duas arestas de M compartilhando uma extremidade. Uma correspondência de máxima cardinalidade em H é chamada *máxima correspondência*, e uma correspondência que é máxima sob inclusões é chamada correspondência máxima. Uma correspondência máxima pode ser facilmente computada utilizando um algoritmo guloso. Vazirani, Vijay V. (2013, p.3).

O tamanho de uma máxima correspondência em G providencia um limite inferior, visto que qualquer cobertura de vértice tem que selecionar pelo menos uma extremidade de cada aresta combinada. Este limite inferior sugere um simples algoritmo, encontrar a correspondência máxima em G e exibir o conjunto de vértices combinados.

Teorema: Um conjunto de vértices combinados de uma correspondência máxima de G é um fator 2-algoritmo de aproximação para o problema de cobertura de vértices com cardinalidade.

Prova: Nenhuma aresta pode ser descoberta pela esquerda do conjunto de vértices selecionado – Deixe M ser a correspondência selecionada. $|M| \leq OPT$

O fator de aproximação segue da observação que a cobertura selecionada pelo algoritmo tem cardinalidade $2|M|$, que é no máximo *2-algoritmo de aproximação*.

3. Problema do Caixeiro Viajante

O problema do caixeiro viajante é um dos problemas mais bem estudados de otimização combinatória.

Problema: Dado um grafo completo com custos não negativos para arestas, encontrar um ciclo de custo mínimo visitando cada vértice somente uma vez.

O problema tratado respeita o triangulo de desigualdade.

3.1. Um fator 2-algoritmo de aproximação

Conforme apresentado por Vazirani, Vijay V. (2013, p.31-32) o limite inferior que será utilizado para obter o fator 2-algoritmo de aproximação será o custo de uma árvore geradora mínima de G, visto que removendo qualquer aresta de uma solução ótima para o problema do caixeiro viajante temos uma árvore geradora de G.

Algoritmo para encontrar o fator 2-algoritmo de aproximação:

1. Encontre uma árvore geradora mínima, T, de G.
2. Duplique cada aresta da árvore geradora mínima para obter um grafo Euleriano.
3. Encontre um percurso Euleriano, T, neste grafo.
4. Exiba o percurso que visita os vértices de G na ordem de suas primeiras aparições em T, deixe C ser este percurso.

Teorema: O algoritmo 1 é um fator 2-algoritmo de aproximação para o problema do caixeiro viajante

Prova: O $custo(T) \leq OPT$ desde que T contenha cada aresta de T duas vezes, $custo(\mathrm{T}) = 2 * custo(T)$. Por causa do triangulo de desigualdade depois da "poda-curta" ("short-cutting"), $custo(C) \leq custo(\mathrm{T})$. Combinando estas desigualdades obtemos $cuto(C) \leq 2 * OPT$.

4. Conclusão

Foram apresentados apenas dois algoritmos que podem ser aplicados a problemas NP-Completos, no entanto esta é uma área grande, com inúmeros tipos de relação de aproximação e aplicações.

É uma área que já atingiu um período de relativa maturidade como disciplina, todavia ainda há bastante trabalho a se fazer.

Para o próprio problema do caixeiro viajante há outro algoritmo de aproximação com fator 3/2-algoritmo de aproximação, criado a aproximadamente 35 anos atrás por Christofides. Este é um dos exemplos para possíveis trabalhos futuros. Williamson, David P., and David B. Shmoys (2011, p.447)

5. Referências

Cormen, Thomas H. *Introduction to algorithms*. MIT press, 2009.

Vazirani, Vijay V. *Approximation algorithms*. Springer Science & Business Media, 2013.

Papadimitriou, Christos, and Umesh Vazirani. "Algoritmos." (2009).

Williamson, David P., and David B. Shmoys. *The design of approximation algorithms*. Cambridge university press, 2011.