Análise de algoritmos de aproximação utilizando o método dual-fitting

Aluno: Leonardo Marchetti

Orientadora: Professora Cristina Gomes Fernandes

Trabalho de conclusão de curso

Departamento de Ciência da Computação
Instituto de Matemática e Estatística
Universidade de São Paulo

1. Introdução

Neste trabalho iremos descrever um sofisticado método para analisar algoritmos de aproximação para problemas de otimização chamado *dual-fitting* e apresentar algumas aplicações desse método. Este método se baseia no conceito de dualidade em programação linear.

Uma descrição completa do método e dos conceitos necessários para seu entendimento pode ser vista no trabalho em si. Para os propósitos desse pôster, começaremos apresentando um exemplo de aplicação do dual-fitting. Este mesmo exemplo pode ser visto com muito mais detalhe no trabalho, e seu estudo se baseou principalmente no livro de Vazirani [3].

2. Multi-cobertura mínima por conjuntos

Para apresentarmos o *Problema da Multi-Cobertura Mínima por Conjuntos Restrita*, doravante também chamado de MinMCCR, são necessárias algumas definições.

Seja E um conjunto finito e $\mathcal{S}$ uma coleção de subconjuntos de E. Seja ainda $r : E \to \mathbb{N}$ uma função que associa a cada elemento $e \in E$ um natural r_e. Uma *multi-cobertura* de E em $\mathcal{S}$ sujeita a r é um subconjunto τ de $\mathcal{S}$ tal que

$$|\{S \in \tau : e \in S\}| \geq r_e.$$

Se τ é uma multi-cobertura cobertura de E em $\mathcal{S}$ sujeita a r e temos um custo c tal que $c_S \in \mathbb{Q}_{\geq}$ para cada $S \in \mathcal{S}$, definimos $c(\tau)$ como o número $\sum_{S\in\tau} c_S$.

Problema MinMCCR $(E, \mathcal{S}, c, r)$: *Dados um conjunto finito E, uma função $r : E \to \mathbb{N}$ que associa a cada elemento $e \in E$ um número r_e de vezes que e deve ser coberto, uma coleção finita $\mathcal{S}$ de subconjuntos não-vazios de E que é uma multi-cobertura de E em $\mathcal{S}$ sujeita a r e um custo $c_S \in \mathbb{Q}_{\geq}$ para cada $S \in \mathcal{S}$, encontrar uma multi-cobertura τ de E em $\mathcal{S}$ sujeita a r que minimize $c(\tau)$.*

Para esse problema, podemos escrever o seguinte algoritmo que utiliza uma estratégia gulosa.

Algoritmo MinMCCR-Guloso $(E, \mathcal{S}, c, r)$

1 . $\mathcal{S}' \leftarrow \mathcal{S}$
2 . $E' \leftarrow E$
3 . para cada e em E faça
4 . $\quad r'_e \leftarrow 0$
5 . $\tau \leftarrow \emptyset$
6 . enquanto $E' \neq \emptyset$
7 . $\quad$ seja Z em $\mathcal{S}'$ tal que $c_Z/|Z \cap E'|$ é mínimo
8 . $\quad$ para cada e em $Z \cap E'$ faça
9 . $\quad\quad r'_e \leftarrow r'_e + 1 \quad \triangleright \quad y_{(e,r'_e)} = c_Z/|Z \cap E|$
10 . $\quad E' \leftarrow \{e \in E' : r'_e < r_e\}$
11. $\quad \mathcal{S}' \leftarrow \mathcal{S}' \setminus \{Z\}$
12. $\quad \mathcal{S}' \leftarrow \{S \in \mathcal{S}' : \sum_{e\in S\cap E'} r_e > \sum_{e\in S\cap E'} r'_e\}$
13. $\quad \tau \leftarrow \tau \cup \{Z\}$
14. devolva τ

Note que a coleção τ devolvida é de fato uma multi-cobertura de E em $\mathcal{S}$ sujeita a r e que o tempo de execução do algoritmo MinMCCR-Guloso é polinomial em $(|E| + |\mathcal{S}|)$.

O seguinte teorema pode ser mostrado através do método dual-fitting.

Teorema 1: *O algoritmo* MinMCCR-Guloso *é uma H_n-aproximação para o problema* MinMCCR $(E, \mathcal{S}, c, r)$*, onde $n = |E|$ e H_n é o n-ésimo número harmônico.*

Não faremos aqui a demonstração formal desse teorema. Apenas introduziremos alguns ingredientes importantes ao método, omitindo algumas demonstrações. Dessa forma, mostramos uma idéia geral sobre uma aplicação do método dual-fitting.

Considere a seguinte relaxação linear $P(E, \mathcal{S}, c, r)$ do problema MinMCCR: encontrar um vetor x indexado por $\mathcal{S}$ que

$$\begin{aligned} &\text{minimize} && cx \\ &\text{sob as restrições} && \textstyle\sum_{S: e\in S} x_S \geq r_e \text{ para cada } e \in E, \\ &&& x_S \leq 1 \text{ para cada } S \in \mathcal{S}, \\ &&& x_S \geq 0 \text{ para cada } S \in \mathcal{S}. \end{aligned}$$

O dual $D(E, \mathcal{S}, c, r)$ do programa linear acima pode ser escrito como: encontrar um vetor y indexado por E e um vetor z indexado por $\mathcal{S}$ que

$$\begin{aligned} &\text{maximize} && \textstyle\sum_{e\in E} r_e y_e - \sum_{S\in\mathcal{S}} z_S \\ &\text{sob as restrições} && \textstyle\sum_{e\in S} y_e - z_S \leq c_S \text{ para cada } S \in \mathcal{S}, \\ &&& y_e \geq 0 \text{ para cada } e \in E, \\ &&& z_S \geq 0 \text{ para cada } S \in \mathcal{S}. \end{aligned}$$

O algoritmo determina uma solução primal x, que é dada pelos conjuntos escolhidos a cada iteração. Se τ é a multi-cobertura devolvida pelo algoritmo, $x_S = 1$ se $S \in \tau$ e $x_S = 0$ caso contrário.

Podemos identificar um candidato a solução dual determinado implicitamente pelo algoritmo que será conveniente para a análise. Definimos $y_{(e,j)} = c_S/|S \cap E'|$, sendo S o conjunto Z escolhido na linha 6 do algoritmo no início da iteração em que o elemento e é coberto pela j-ésima vez, para cada e em E e $j = 1, 2, \cdots, r_e$. Podemos imaginar que os $y_{(e,j)}$ são calculados pelo algoritmo como mostrado no comentário da linha 9;

Um par (α, β) que é candidato a solução dual, onde α é indexado por E e β é indexado por $\mathcal{S}$, será definido a seguir. Para todo e em E, tome $\alpha_e = y_{(e,r_e)}$. Para todo $S \in \tau$, tome $\beta_S = \sum_{e\in C_S} (\alpha_e - y_{(e,j_e)})$, onde j_e é a vez em que e foi coberto por S, e $C_S = S \cap E'$ no início da iteração em que S é escolhido como o conjunto Z na linha 6 do algoritmo. Para todo $S \in \mathcal{S} \setminus \tau$, tome $\beta_S = 0$.

Podemos provar que o valor deste par (α, β) é pelo menos o valor da solução primal, ou ainda, é igual ao custo da multi-cobertura τ devolvida pelo algoritmo.

O par (α, β) que definimos em geral não é uma solução de $D(E, \mathcal{S}, c, r)$. Porém, podemos mostrar que, a partir dele, usando um fator de escala, é possível construir uma solução dual (α', β') que o é, como definiremos a seguir:

$$\alpha'_e = \frac{\alpha_e}{H_n} \text{ para cada } e \in E, \quad \beta'_S = \frac{\beta_S}{H_n} \text{ para cada } S \in \mathcal{S}.$$

Portanto, temos

$$\begin{aligned} c(\tau) = \sum_{e\in E} r_e\alpha_e - \sum_{S\in\mathcal{S}} \beta_S &= H_n \cdot [\sum_{e\in E} r_e\alpha'_e - \sum_{S\in\mathcal{S}} \beta'_S] \\ &\leq H_n \cdot \mathrm{opt}(E, \mathcal{S}, c, r) \end{aligned}$$

Sendo assim, temos que o algoritmo guloso apresentado é uma H_n-aproximação para o MINMCCR.

3. Generalização

Podemos utilizar a mesma estratégia que empregamos nessa breve análise do algoritmo MinMCCR-Guloso para obter a razão de aproximação de muitos outros algoritmos de aproximação para diversos problemas de otimização. Essa estratégia é o que é chamado na literatura de método dual-fitting.

De forma geral, dado um problema de minimização (o método é análogo para problemas de maximização) e um algoritmo de aproximação para este problema, o método consiste em:

1. Obter um programa linear inteiro $PI(I)$ para uma instância genérica I do problema, tal que $PI(I)$ seja equivalente ao problema para toda instância I viável.
2. Considerar o programa linear $P(I)$, que á a relaxação linear de $PI(I)$, e seu dual $D(I)$.
3. Determinar uma solução x de $P(I)$ a partir do que o algoritmo devolve.
4. Perceber como o algoritmo determina implicitamente um candidato a solução y de $D(I)$.
5. Mostrar que o valor da função objetivo de $D(I)$ para y é pelo menos o valor da função objetivo de $P(I)$ para x.
6. Mostrar que, para algum $c \geq 1$, $\frac{y}{c}$ é uma solução de $D(I)$.
7. Concluir que o algoritmo é uma c-aproximação para o problema.

Note que c pode ser uma constante ou uma função de $< I >$, onde $< I >$ é o tamanho da instância I.

No trabalho mostramos a prova de que os passos 1-6 de fato levam a conclusão do passo 7.

4. Cobertura mínima por conjuntos

Outro problema que estudamos foi o *Problema da Cobertura Mínima por Conjuntos*, ou MinCC. Para este problema analisamos o algoritmo guloso proposto por Chvátal [1]. Através do método dual-fitting mostramos que este algoritmo é uma H_n-aproximação para o MinCC.

5. Localização de instalações

O último problema que estudamos foi o *Problema da Localização de Instalações* (facility location), que também chamaremos de MinLI, que pode ser definido como visto abaixo:

Problema MinLI $(\mathcal{F}, \mathcal{C}, f, c)$: *Dados um conjunto finito não-vazio de instalações $\mathcal{F}$, um conjunto finito não-vazio de cidades $\mathcal{C}$, um custo de abertura $f_i \in \mathbb{Q}_{\geq}$ para cada instalação $i \in \mathcal{F}$ e um custo de conexão $c_{jk} \in \mathbb{Q}_{\geq}$ para cada $j \in \mathcal{F}$ e $k \in \mathcal{C}$, determinar um conjunto não-vazio $\mathcal{F}' \subseteq \mathcal{F}$ tal que*

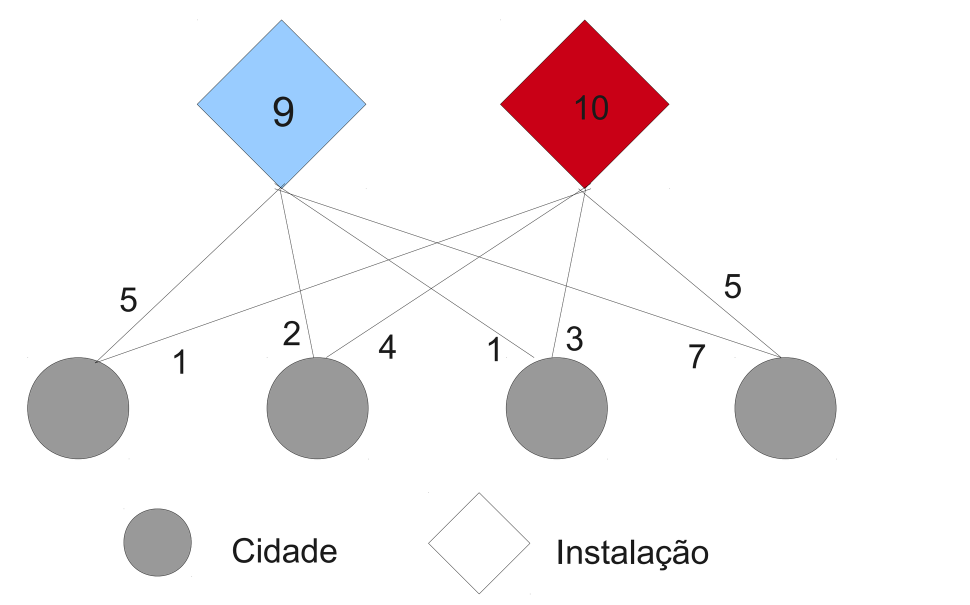
$$\sum_{i\in\mathcal{F}'} f_i + \sum_{j\in\mathcal{C}} \min\{c_{ij} : i \in \mathcal{F}'\}$$

seja mínimo.

Estudamos o caso métrico desse problema, o qual chamaremos de *Problema Métrico da Localização de Instalações* ou ainda MinMLI. No MinMLI os custos de conexão respeitam a desigualdade triangular, ou seja, dadas duas instalações i e i' e duas cidades j e j' quaisquer vale que:

$$c_{ij} \leq c_{ij'} + c_{i'j'} + c_{i'j}.$$

Veja um exemplo de instância para o MINMLI no desenho.



A solução ótima para essa instância corresponde ao conjunto que contém unicamente a instalação vermelha.

Apresentaremos um algoritmo para o MinMLI que possui uma abordagem gulosa. Para entendê-lo são necessárias algumas definições.

Uma estrela $S = (i, C)$ é um par ordenado que possui uma instalação $i \in \mathcal{F}$ e um subconjunto não-vazio C de $\mathcal{C}$.

O algoritmo é iterativo, e consiste em, a cada iteração, escolher e adicionar à solução uma estrela de menor “custo médio” dentre todas as estrelas possíveis de serem formadas com as cidades que ainda não foram conectadas a nenhuma facilidade. O algoritmo pode ser escrito da seguinte forma:

Algoritmo MinMLI-Guloso $(\mathcal{F}, \mathcal{C}, f, c)$

1 . $\mathcal{C}' \leftarrow \mathcal{C}$
2 . $\mathcal{F}' \leftarrow \emptyset$
3 . $f' \leftarrow f$
4 . enquanto $\mathcal{C}' \neq \emptyset$
5 . $\quad$ tome uma estrela (i, C) que minimiza $\frac{f'_i + \sum_{j\in C} c_{ij}}{|C|}$ para $i \in \mathcal{F}$ e $C \subseteq \mathcal{C}'$.
6 . $\quad \mathcal{F}' \leftarrow \mathcal{F}' \cup \{i\}$
7 . $\quad f'_i \leftarrow 0$
8 . $\quad \mathcal{C}' \leftarrow \mathcal{C}' \setminus C$
9 . devolva $\mathcal{F}'$.

Utilizando o método dual-fitting podemos mostrar que o algoritmo MinMLI-Guloso é uma $1,861$-aproximação para o MinMLI, como mostrado inicialmente por M. Mahdian *et al.* [2].

Referências

[1] V. Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[2] M. Mahdian, E. Markakis, A. Saberi, and V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *Journal of the ACM*, 50:127–137, 2001.

[3] V. V. Vazirani. *Approximation Algorithms*. Springer, 2001.