Análise de algoritmos de aproximação utilizando o método dual-fitting

Leonardo Marchetti
Orientadora: Cristina Gomes Fernandes

Instituto de Matemática e Estatística
Universidade de São Paulo, Brasil

Trabalho de Conclusão de Curso, 2009

O que é o método dual-fitting?

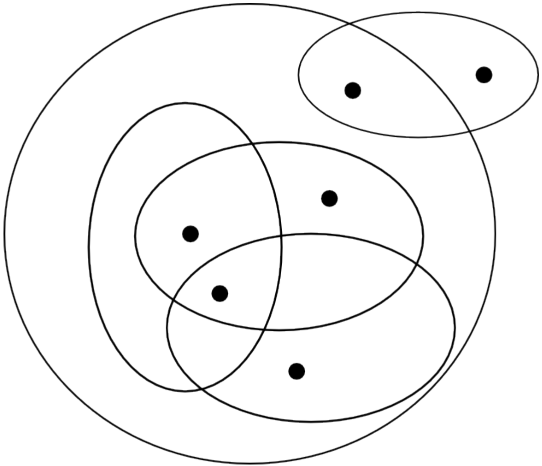
É um sofisticado método para analisar algoritmos de aproximação para problemas de otimização.

Ele se baseia no princípio de dualidade em programação linear.

Cobertura por conjuntos

- Seja E um conjunto finito e $\mathcal{S}$ uma coleção de subconjuntos de E. Uma *cobertura* de E em $\mathcal{S}$ é um subconjunto τ de $\mathcal{S}$ tal que para todo $e \in E$ temos que $e \in S$ para algum $S \in \tau$.
- Se τ é uma cobertura de E em $\mathcal{S}$ 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 $\sum_{S \in \tau} c_S$.

Cobertura por conjuntos - exemplo



Cobertura mínima por conjuntos

Problema MINCC $(E, \mathcal{S}, c)$: *Dados um conjunto finito E, uma coleção finita $\mathcal{S}$ de subconjuntos não-vazios de E (que cobre E) e um custo $c_S \in \mathbb{Q}_{\geq}$ para cada $S \in \mathcal{S}$, encontrar uma cobertura τ de E em $\mathcal{S}$ que minimize $c(\tau)$.*

Algoritmo guloso de Chvátal

Veremos um algoritmo guloso proposto por Chvátal [1] para o MinCC.

Algoritmo MINCC-CHVATAL $(E, \mathcal{S}, c)$

1. $\mathcal{S}' \leftarrow \mathcal{S}$
2. $E' \leftarrow E$
3. $\tau \leftarrow \emptyset$
4. enquanto $E' \neq \emptyset$
5. seja Z em $\mathcal{S}'$ tal que $c_Z/|Z \cap E'|$ é mínimo
6. $E' \leftarrow E' \setminus Z$
7. $\mathcal{S}' \leftarrow \{S \in \mathcal{S}' : S \cap E' \neq \emptyset\}$
8. $\tau \leftarrow \tau \cup \{Z\}$
9. devolva τ

Razão de aproximação

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

Vamos provar o teorema utilizando o método dual-fitting.

Programa linear inteiro

Considere o seguinte programa linear inteiro $PI(E, \mathcal{S}, c)$: encontrar um vetor x indexado por $\mathcal{S}$ que

$$
\begin{array}{rl}
\text{minimize} & cx \\
\text{sob as restrições} & \sum_{S: e \in S} x_S \geq 1 \text{ para todo } e \in E, \\
& x_S \in \{0, 1\} \text{ para todo } S \in \mathcal{S}.
\end{array}
$$

Relaxação linear

Seja $P(E, \mathcal{S}, c)$ uma relaxação linear de $PI(E, \mathcal{S}, c)$.

$P(E, \mathcal{S}, c)$ consiste em encontrar um vetor x indexado por $\mathcal{S}$ que

$$\begin{array}{rl} \text{minimize} & cx \\ \text{sob as restrições} & \sum_{S: e \in S} x_S \geq 1 \text{ para todo } e \in E, \\ & x_S \geq 0 \text{ para todo } S \in \mathcal{S}. \end{array}$$

Dual da relaxação linear

Seja $D(E, \mathcal{S}, c)$ o programa linear dual de $P(E, \mathcal{S}, c)$.

$D(E, \mathcal{S}, c)$ consiste em encontrar um vetor y indexado por E que

$$
\begin{array}{rl}
\text{maximize} & \sum_{e \in E} y_e \\
\text{sob as restrições} & \sum_{e \in S} y_e \leq c_S \text{ para todo } S \in \mathcal{S}, \\
& y_e \geq 0 \text{ para todo } e \in E.
\end{array}
$$

Solução primal

Podemos construir, a partir do que o algoritmo devolve, uma solução x de $P(E, \mathcal{S}, c)$, que é dada pelos conjuntos escolhidos pelo algoritmo.

Se τ é a cobertura devolvida pelo algoritmo,

- $x_S = 1$, se $S \in \tau$
- $x_S = 0$, caso contrário.

Candidato à solução dual

O algoritmo determina implicitamente um candidato à solução dual y.

- Para um elemento $e \in E$, considere a iteração em que e é removido de E'. Seja $C_e := E \setminus E'$ para E' no início desta iteração e S_e o conjunto Z escolhido nesta iteração. Observe que $\tau = \{S_e : e \in E\}$.
- O candidato a solução dual é definido para cada e em E como
$$y_e = \frac{c_{S_e}}{|S_e \setminus C_e|}.$$
- y em geral não é uma solução dual, pois não satisfaz a restrição $\sum_{e \in S} y_e \leq c_S$.

Lema 2

Lema 2: *O valor da função objetivo de $D(E, \mathcal{S}, c)$ para y é maior ou igual o valor da função objetivo de $P(E, \mathcal{S}, c)$ para x.*

Demonstração: Para cada $S \in \tau$, considere a iteração em que S é o conjunto escolhido pelo algoritmo e seja $C = E \setminus E'$ para E' no início dessa iteração. Então,

$$c_S = \sum_{e \in S \setminus C} \frac{c_S}{|S \setminus C|} = \sum_{e \in S \setminus C} \frac{c_{S_e}}{|S_e \setminus C_e|} = \sum_{e \in S \setminus C} y_e.$$

$$\sum_{S \in \tau} c_s \leq \sum_{e \in E} y_e$$

$\square$

Fator de escala

Vamos mostrar que apesar de y não ser uma solução de $D(E, \mathcal{S}, c)$, se escolhermos um fator de escala α adequado, o vetor $\frac{y}{\alpha}$ é uma solução dual.

Seja y' indexado por E tal que,

$$y'_e = \frac{y_e}{H_n} \text{ para todo } e \text{ em } E.$$

Lema 3

Lema 3: *y' é uma solução de $D(E, \mathcal{S}, c)$.*

Demonstração: Temos que $y'_e \geq 0$ para todo $e \in E$ pois $y_e \geq 0$ para todo $e \in E$. Falta mostrar que $\sum_{e \in S} y'_e \leq c_S$ para todo $S \in \mathcal{S}$. Seja $S \in \mathcal{S}$, e sejam $e_1, e_2, \ldots, e_k$ os elementos de S na ordem em que são cobertos pelo algoritmo.
No início da iteração em que o elemento e_i é coberto, existem pelo menos $k - i + 1$ elementos descobertos em S. Assim y_{e_i} é no máximo $c_S/(k - i + 1)$. De fato, como o algoritmo escolhe sempre um conjunto Z tal que $c_Z/|Z \cap E'|$ é mínimo e $\frac{c_S}{|S \cap E'|} \leq \frac{c_S}{k-i+1}$, temos que $y_{e_i} \leq c_S/(k - i + 1)$ e portanto,

$$y'_{e_i} \leq \frac{c_S}{(k - i + 1)H_n}.$$

Lema 3

Lema 3: *y' é uma solução de $D(E, \mathcal{S}, c)$.*

Demonstração: (...)

$$y'_{e_i} \leq \frac{c_S}{(k-i+1)H_n}.$$

Mas então

$$\sum_{e \in S} y'_e \leq \sum_{i=1}^{k} \frac{c_S}{(k-i+1)H_n} = \frac{c_S}{H_n} \sum_{i=1}^{k} \frac{1}{k-i+1} = c_S \frac{H_k}{H_n} \leq c_S.$$

$\square$

Conclusão

Portanto, y' é uma solução viável para o problema dual $D(E, \mathcal{S}, c)$ e consequentemente,

$$V(D(E, \mathcal{S}, c), y') \leq opt(D(E, \mathcal{S}, c)) = opt(P(E, \mathcal{S}, c)) \leq opt(E, \mathcal{S}, c).$$

Finalmente, podemos concluir sobre o custo da cobertura τ devolvida pelo algoritmo MinCC-Chvatal é tal que

$$c(\tau) = V(P(E, \mathcal{S}, c), x) \leq V(D(E, \mathcal{S}, c), y).$$

Como temos $V(D(E, \mathcal{S}, c), y) = H_n \cdot V(D(E, \mathcal{S}, c), y')$, pois a função objetivo de $D(E, \mathcal{S}, c)$ é linear, segue que

$$c(\tau) \leq H_n \cdot V(D(E, \mathcal{S}, c), y') \leq H_n \cdot opt(E, \mathcal{S}, c)$$

e o algoritmo é uma H_n-aproximação para o problema.

Generalização

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 $D(I)$.

Generalização

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:
(...)

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.

Outros problemas estudados

Na monografia estudamos também a análise de algoritmos para os seguintes problemas:

- Problema da Multi-cobertura Mínima por Conjuntos. Analisamos o algoritmo guloso encontrado no livro de V. Vazirani [3].
- Problema Métrico da Localização de Instalações. Estudamos o algoritmo guloso encontrado no artigo de Mahdian *et al.* [2].

Referências

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

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.

V.V. Vazirani.
Approximation Algorithms.
Springer, 2001.

Dúvidas?