Algoritmos de aproximação - Problema de cobertura por conjuntos

Marina Andretta

ICMC-USP

22 de setembro de 2015

Baseado no livro Uma introdução sucinta a Algoritmos de Aproximação, de M. H. Carvalho, M. R. Cerioli, R. Dahab, P. Feofiloff, C. G. Fernandes, C. E. Ferreira, K. S. Guimarães, F. K. Miyazawa, J. C. Piña Jr., J. A. R. Soares e Y. Wakabayashi.

Problema de cobertura mínima por conjuntos

Dada uma coleção finita $\mathcal{S}$ de conjuntos finitos, dizemos que uma subcoleção $\mathcal{T}$ de $\mathcal{S}$ cobre um conjunto finito E se todo elemento de E pertence a algum conjunto de $\mathcal{T}$.

Neste caso, dizemos também que $\mathcal{T}$ é uma cobertura de E.

Problema de cobertura mínima por conjuntos

O problema da cobertura mínima por conjuntos (*minimum set cover problem*) consiste no seguinte:

Problema $\textsc{MinCC}(E, \mathcal{S}, c)$: Dados um conjunto finito E, uma coleção finita $\mathcal{S}$ de conjuntos finitos que cobre E e um custo c_S em $\mathbb{Q}_{\geq}$ para cada S em $\mathcal{S}$, encontrar uma cobertura $\mathcal{T}$ de E que minimize $c(\mathcal{T})$.

A exigência de que $\mathcal{S}$ cubra E serve apenas para excluir instâncias inviáveis do problema.

Chamamos o número $c(\mathcal{T})$ de custo da cobertura $\mathcal{T}$.

Assim, o problema consiste em encontrar uma cobertura de custo mínimo.

Exemplo 1

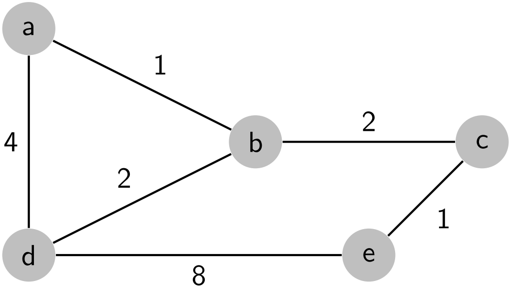
Um caso particular o problema $\textsc{MinCC}$ é o problema de cobertura por arestas.

Neste problema, dado um grafo $G = (V, A)$, e um custo c_a em $\mathbb{Q}_{\geq}$ para cada aresta $a \in A$, desejamos encontrar um sub-conjunto $\mathcal{T}$ de arestas de A de custo mínimo de forma que todo vértice de V seja atingido por ao menos uma aresta de $\mathcal{T}$.

Neste caso, dado $G = (V, A)$, temos que E é o conjunto de vértices de um grafo ($E = V$), $\mathcal{S}$ é o conjunto de arestas ($\mathcal{S} = A$) e c_S é o custo das arestas ($c_S = c_a$).

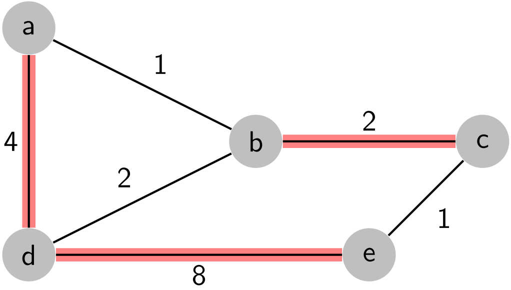
Exemplo 1

Neste exemplo, $E = \{a, b, c, d, e\}$,
$\mathcal{S} = \{\{a, b\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, e\}, \{d, e\}\}$ e c_S é dado pelos números nas arestas da figura.



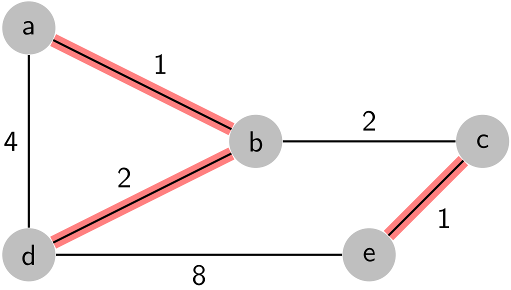
Exemplo 1

Uma possível cobertura (minimal) é $\mathcal{T} = \{\{a, d\}, \{d, e\}, \{b, c\}\}$, com custo $c(\mathcal{T}) = 4 + 8 + 2 = 14$.



Exemplo 1

A cobertura mínima, neste caso, é $\mathcal{T} = \{\{a, b\}, \{c, e\}, \{b, d\}\}$, com custo $c(\mathcal{T}) = 1 + 1 + 2 = 4$.



Exemplo 2

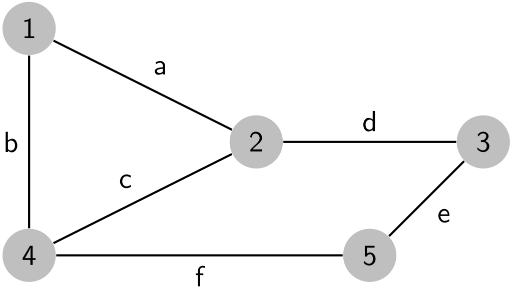
Um outro problema que pode ser escrito como um problema $\textsc{MinCC}$ é o problema de cobertura por vértices.

Neste problema, dado um grafo $G = (V, A)$, e um custo c_v em $\mathbb{Q}_{\geq}$ para cada vértice $v \in V$, desejamos encontrar um sub-conjunto T de vértices de V de custo mínimo de forma que toda aresta de A seja incidente a pelo menos um vértice T.

Neste caso, dado $G = (V, A)$, temos que E é o conjunto de arestas de um grafo ($E = A$), c_S é o custo dos vértices ($c_S = c_v$) e, para cada $v \in V$, temos um conjunto $S \in \mathcal{S}$ formado por todas arestas de A incidentes em v.

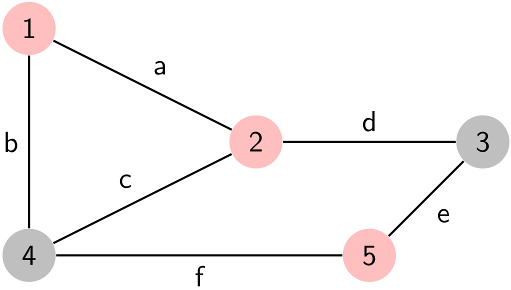
Exemplo 2

Neste exemplo, $E = \{a, b, c, d, e, f\}$,
$\mathcal{S} = \{\{a, b\}, \{a, c, d\}, \{d, e\}, \{b, c, f\}, \{e, f\}\}$ e c_S é 1 para todo $S \in \mathcal{S}$.



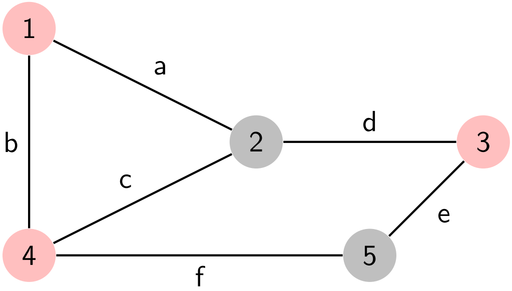
Exemplo 2

Uma cobertura mínima é $\mathcal{T} = \{\{a, b\}, \{a, c, d\}, \{e, f\}\}$, correspondente ao conjunto de vértices $\{1, 2, 5\}$, com custo $c(\mathcal{T}) = 3$.



Exemplo 2

Outra cobertura mínima é $\mathcal{T} = \{\{a, b\}, \{d, e\}, \{b, c, f\}\}$, correspondente ao conjunto de vértices $\{1, 3, 4\}$, com custo $c(\mathcal{T}) = 3$.



Problema de cobertura mínima por conjuntos

O MinCC é *NP*-difícil mesmo quando cada conjunto em $\mathcal{S}$ não tem mais que três elementos.

Uma estratégia gulosa simples para o problema consiste em selecionar repetidamente o conjunto em $\mathcal{S}$ que é mais "promissor" em termos de custo com relação ao número de elementos ainda não cobertos que ele contém.

Essa estratégia dá origem ao seguinte algoritmo, proposto por Chvátal, que apresentamos em sua versão recursiva.

Algoritmo de aproximação

Algoritmo $\textsc{MinCC-Chvátal}(E, \mathcal{S}, c)$:

se $E = \emptyset$

então devolva $\emptyset$;

senão seja Z em $\mathcal{S}$ tal que $c_Z/|Z \cap E|$ é mínimo;

faça $E' \leftarrow E \setminus Z$;

faça $\mathcal{S}' \leftarrow \{S \in \mathcal{S} : S \cap E' \neq \emptyset\}$;

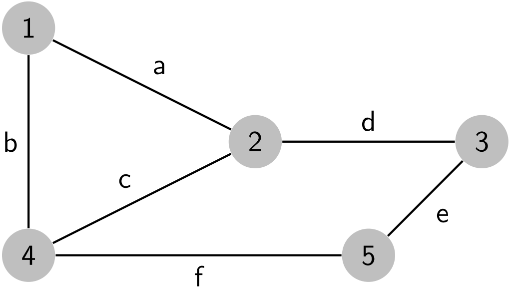
seja c' a restrição de c a $\mathcal{S}'$;

faça $\mathcal{T}' \leftarrow \textsc{MinCC-Chvátal}(E', \mathcal{S}', c')$;

devolva $\{Z\} \cup \mathcal{T}'$.

Exemplo

Considere a instância do exemplo 2. Temos $E_0 = \{a, b, c, d, e, f\}$, $\mathcal{S}_0 = \{\{a, b\}, \{a, c, d\}, \{d, e\}, \{b, c, f\}, \{e, f\}\}$ e c_S é 1 para todo $S \in \mathcal{S}_0$.



Exemplo

Como E_0 não é vazio, iremos para a linha 3 do algoritmo.

Temos que

Z	$\{a, b\}$	$\{a, c, d\}$	$\{d, e\}$	$\{b, c, f\}$	$\{e, f\}$
$\frac{c_Z}{\lvert Z \cap E_0 \rvert}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{2}$

Vamos escolher $Z_0 = \{a, c, d\}$.

Exemplo

Definimos

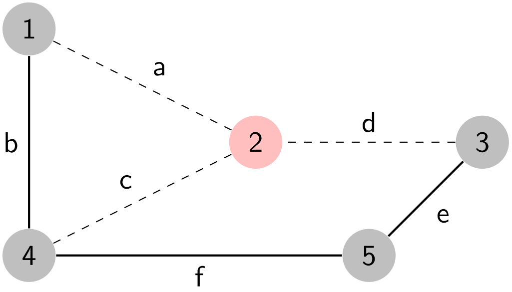
$$E' \leftarrow E_0 \setminus Z_0 = \{b, e, f\},$$

$$\mathcal{S}' \leftarrow \{S \in \mathcal{S}_0 : S \cap E' \neq \emptyset\} = \{\{a, b\}, \{d, e\}, \{b, c, f\}, \{e, f\}\},$$

e chamamos o algoritmo novamente com estes parâmetros.

Exemplo

Temos agora $E_1 = \{b, e, f\}$, $\mathcal{S}_1 = \{\{a, b\}, \{d, e\}, \{b, c, f\}, \{e, f\}\}$ e c_S é 1 para todo $S \in \mathcal{S}_1$.



Exemplo

Como E_1 não é vazio, iremos para a linha 3 do algoritmo.

Temos que

Z	$\{a, b\}$	$\{d, e\}$	$\{b, c, f\}$	$\{e, f\}$
$\frac{c_Z}{\lvert Z \cap E_1 \rvert}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$

Vamos escolher $Z_1 = \{b, c, f\}$.

Exemplo

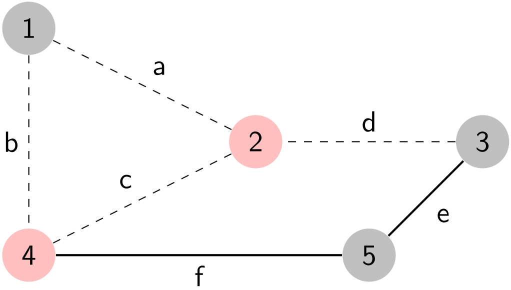
Definimos

$$E' \leftarrow E_1 \setminus Z_1 = \{e\},$$

$$\mathcal{S}' \leftarrow \{S \in \mathcal{S}_1 : S \cap E' \neq \emptyset\} = \{\{d, e\}, \{e, f\}\},$$

e chamamos o algoritmo novamente com estes parâmetros.

Temos agora $E_2 = \{e\}$, $\mathcal{S}_2 = \{\{d, e\}, \{e, f\}\}$ e c_S é 1 para todo $S \in \mathcal{S}_2$.



Exemplo

Como E_2 não é vazio, iremos para a linha 3 do algoritmo.

Temos que

Z	$\{d, e\}$	$\{e, f\}$
$\frac{c_Z}{\lvert Z \cap E_2 \rvert}$	1	1

Vamos escolher $Z_2 = \{d, e\}$.

Exemplo

Definimos

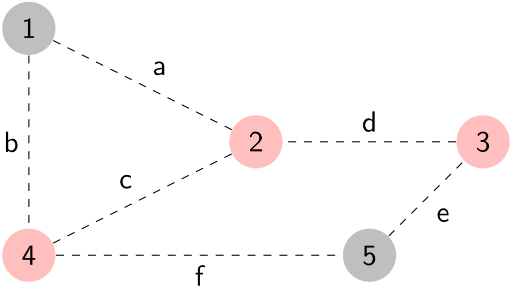
$$E' \leftarrow E_2 \setminus Z_2 = \emptyset,$$

$$\mathcal{S}' \leftarrow \{S \in \mathcal{S}_2 : S \cap E' \neq \emptyset\} = \emptyset,$$

e chamamos o algoritmo novamente com estes parâmetros.

Exemplo

Temos agora $E_3 = \mathcal{S}_3 = \emptyset$. Assim, o algoritmo termina e devolve a cobertura $\mathcal{T} = \{Z_0\} \cup \{Z_1\} \cup \{Z_2\} = \{\{a, c, d\}, \{b, c, f\}, \{d, e\}\}$, que corresponde aos vértices $\{2, 3, 4\}$.



Algoritmo de aproximação

Claramente o algoritmo devolve uma cobertura de E.

Há uma boa delimitação inferior para o valor ótimo do problema $\textsc{MinCC}(E, \mathcal{S}, c)$ relacionada ao algoritmo.

Suponha que Z é um elemento de $\mathcal{S}$ tal que

$$\frac{c_Z}{|Z \cap E|} \leq \frac{c_S}{|S \cap E|}$$

para todo S em $\mathcal{S}$.

Algoritmo de aproximação

Então, para qualquer cobertura $\mathcal{T}$ de E,

$$\begin{aligned}
\frac{c_Z}{|Z \cap E|}|E| &\leq \frac{c_Z}{|Z \cap E|}\sum_{S \in \mathcal{T}} |S \cap E| \\
&\leq \sum_{S \in \mathcal{T}} \frac{c_S}{|S \cap E|}|S \cap E| \\
&= \sum_{S \in \mathcal{T}} c_S \\
&= c(\mathcal{T}).
\end{aligned}$$

Como esta relação vale inclusive para a cobertura ótima $\mathcal{T}^*$, temos que

$$c(\mathcal{T}^*) = opt(E, \mathcal{S}, c) \geq |E|\frac{c_Z}{|Z \cap E|}.$$

Seja H_n o número harmônico $H_n := 1 + \frac{1}{2} + \frac{1}{3} + ... + \frac{1}{n}$.

Teorema: O algoritmo $\textsc{MinCC-Chvátal}$ é uma H_n-aproximação polinomial para o $\textsc{MinCC}(E, \mathcal{S}, c)$, com $n := |E|$.

Demonstração: Vamos mostrar que o algoritmo é uma H_n-aproximação por indução em $|E|$.

Se $|E| = 0$, então o algoritmo devolve o conjunto vazio, que é uma cobertura de custo mínimo.

Vejamos agora o caso em que $|E| > 0$ e, portanto, $|\mathcal{S}| > 0$.

Defina $n := |E|$ e $k := |Z \cap E|$, com Z o conjunto escolhido na linha 3 do algoritmo.

Como $|E'| = n - k < |E|$, pela hipótese de indução temos que a coleção $\mathcal{T}'$ produzida na linha 7 do algoritmo é uma cobertura de E' e que

$$c'(\mathcal{T}') \leq H_{n-k} opt(E', \mathcal{S}', c').$$

Além disso, como toda cobertura de E contém uma cobertura de E' formada por conjuntos de $\mathcal{S}'$, temos que

$$opt(E', \mathcal{S}', c') \leq opt(E, \mathcal{S}, c).$$

Algoritmo de aproximação - demonstração

Assim,

$$c(\{Z\} \cup \mathcal{T}') = c_Z + c'(\mathcal{T}') \leq c_Z + H_{n-k} opt(E, \mathcal{S}, c).$$

Como $opt(E, \mathcal{S}, c) \geq |E| \frac{c_Z}{|Z \cap E|} = \frac{n}{k} c_Z$,

$$\begin{aligned} c(\{Z\} \cup \mathcal{T}') &\leq \frac{k}{n} opt(E, \mathcal{S}, c) + H_{n-k} opt(E, \mathcal{S}, c) \\ &\leq \left(\frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{n-k+1} + H_{n-k} \right) opt(E, \mathcal{S}, c) \\ &= H_n opt(E, \mathcal{S}, c). \end{aligned}$$

Ademais, o algoritmo é polinomial, já que consome tempo $O(|E||\mathcal{S}|)$.

O algoritmo MinCC-Chvátal pode produzir coberturas de custo arbitrariamente próximo de $H_n opt(E, \mathcal{S}, c)$, onde $n := |E|$.

Mais precisamente, para cada ϵ positivo, existe uma instância do problema para a qual o algoritmo produz uma cobertura de custo $H_n opt(E, \mathcal{S}, c)/(1 + \epsilon)$.

Algoritmo de aproximação

Basta tomar $E := \{1, ..., n\}$, $\mathcal{S} := \{E, \{1\}, ..., \{n\}\}$, $c_E = 1 + \epsilon$ e $c_{\{i\}} := 1/i$, para cada i.

Uma cobertura de custo mínimo é $\{E\}$ e o custo desta cobertura é $1 + \epsilon$.

Por outro lado, o algoritmo MinCC-Chvátal produz a cobertura $\{\{1\}, ..., \{n\}\}$, cujo custo é H_n.

Algoritmo de aproximação

Assintoticamente, o algoritmo $\textsc{MinCC-Chvátal}$ tem a melhor razão possível para o problema $\textsc{MinCC}$, pois sabe-se que $H_n \leq 1 + \ln(n)$ e existe uma constante positiva α para a qual não existe algoritmo com razão de aproximação menor que $\alpha \ln(n)$, com $n := |E|$, para o $\textsc{MinCC}(E, \mathcal{S}, c)$, a menos que $P = NP$.