Algoritmos de Aproximação - UFMS

Gabarito da Lista 1

1. Mostre que o algoritmo ESCALONAMENTO-GRAHAM(m,n,t) tem razão de aproximação $2-\frac{1}{m}$. Para cada m, exiba uma instância para a qual o algoritmo produz um escalonamento que atinge tal razão em relação ao ótimo.

   Solução: Seja $\{M_1,\ldots,M_m\}$ o escalonamento produzido pela chamada ESCALONAMENTO-GRAHAM(m,n,t) e k uma máquina tal que $t(M_k)$ é máximo. Seja j o maior índice em M_k e $\tau := t(M_k)-t_j$. Como $j \in M_k$, a máquina k foi escolhida na linha 3 do algoritmo quando $i=j$. Assim, $\tau \le t(M_l)$ para todo l distinto de k e temos que

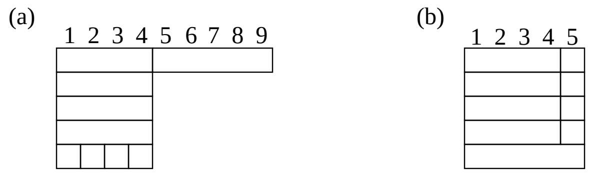
   $$\tau \;\le\; \frac{1}{m}\left(\sum_{l=1}^{k} t(M_l) - t_i\right) \;\le\; \text{opt} - \frac{t_i}{m}\,,$$

   pela segunda desigualdade de (2.1). Logo,

   $$t(M_k) \;=\; \tau + t_i \;\le\; \text{opt} - \frac{t_i}{m} + t_i \;=\; \text{opt} + \left(1-\frac{1}{m}\right) t_i \;\le\; \left(2-\frac{1}{m}\right)\text{opt}\,,$$

   onde a última desigualdade vale por causa da primeira desigualdade de (2.1). Concluímos assim que o algoritmo ESCALONAMENTO-GRAHAM(m,n,t) é uma $(2-\frac{1}{m})$-aproximação.

   Para mostrar que a análise acima não pode ser melhorada, para cada m, considere a instância (m,n,t) onde $n=2m-1$ e $t_i=m-1$ para $1\le i\le m-1$, $t_i=1$ para $m\le i\le 2m-2$ e $t_{2m-1}=m$. O escalonamento devolvido pelo algoritmo de Graham para tais instâncias tem tempo de conclusão $2m-1$ enquanto que um escalonamento ótimo tem tempo de conclusão m. A razão entre estes tempos é de $2-\frac{1}{m}$, que é exatamente a razão demonstrada acima. A figura abaixo ilustra o escalonamento produzido pelo algoritmo de Graham e o escalonamento ótimo para a instância descrita, para $m=5$.

2. O **problema da cobertura máxima** (*maximum coverage problem*) consiste no seguinte:

   Problema MAXCC$(E,w,\mathcal{S},k)$: *Dados um conjunto finito E, um peso w_e em $\mathbb{Q}_{\ge}$ para cada e em E, uma coleção $\mathcal{S}$ de subconjuntos de E e um inteiro não-negativo $k \le |\mathcal{S}|$, encontrar uma subcoleção $\mathcal{T}$ de $\mathcal{S}$ tal que $|\mathcal{T}|=k$ e $w(\bigcup \mathcal{T})$ é máximo.*

   Este problema é NP-difícil [GJ79]. Mostre que o algoritmo guloso que escolhe a cada passo um conjunto que cobre o subconjunto mais pesado de elementos descobertos é uma $(1-(1-k^{-1})^k)$-aproximação polinomial para o problema. Lembre-se de que $1-(1-k^{-1})^k > 1-e^{-1}$, onde e é a base dos logaritmos naturais, e conclua que esse algoritmo é uma 0,63-aproximação para o MAXCC.

   Solução: Abaixo descrevemos o algoritmo descrito no enunciado em pseudocódigo.

Algoritmo MAXCC-GULOSO $(E, w, \mathcal{S}, k)$

```
para i de 1 a k faça
    seja S_i um conjunto em 𝒮 tal que w(S_i \ ∪_{j<i} S_j) é máximo
devolva 𝒞 := {S_1, ..., S_k}
```

Seja $\mathcal{C}^*$ uma solução ótima do MAXCC$(E, \mathcal{S}, k)$. Primeiramente, note que, para todo i em $\{1, \ldots, k\}$, temos que

$$w(S_i \setminus \cup_{j<i} S_j) \;\geq\; \frac{\sum_{S \in \mathcal{C}^*} w(S \setminus \cup_{j<i} S_j)}{|\mathcal{C}^*|} \;\geq\; \frac{w(\cup \mathcal{C}^*) - w(\cup_{j<i} S_j)}{k} \;=\; \frac{\mathrm{opt} - w(\cup_{j<i} S_j)}{k} . \tag{1}$$

Agora vamos provar, por indução em i, que

$$w(\cup_{j \leq i} S_j) \;\geq\; \frac{\mathrm{opt}}{k} \left(1 + \ldots + (1 - k^{-1})^{i-1}\right) . \tag{2}$$

Para $i = 1$, temos que $w(S_1) \geq \mathrm{opt}/k$ por (1). Então

$$\begin{aligned}
w(\cup_{j \leq i} S_j) &= w(\cup_{j<i} S_j) + w(S_i \setminus \cup_{j<i} S_j) \\
&\geq w(\cup_{j<i} S_j) + \frac{\mathrm{opt} - w(\cup_{j<i} S_j)}{k} && (3) \\
&= (1 - k^{-1}) w(\cup_{j<i} S_j) + \frac{\mathrm{opt}}{k} \\
&\geq \frac{\mathrm{opt}}{k} \left((1 - k^{-1}) + \ldots + (1 - k^{-1})^{i-1}\right) + \frac{\mathrm{opt}}{k} && (4) \\
&\geq \frac{\mathrm{opt}}{k} \left(1 + (1 - k^{-1}) + \ldots + (1 - k^{-1})^{i-1}\right) ,
\end{aligned}$$

onde (3) vale por (1) e (4) vale por indução. Mas (2), para $i = k$, corresponde a

$$\begin{aligned}
w(\cup_{j \leq i} S_j) &\geq \frac{\mathrm{opt}}{k} \Big(1 + \ldots + (1 - k^{-1})^{k-1}\Big) \\
&= \frac{\mathrm{opt}}{k} \cdot \frac{1 - (1 - k^{-1})^k}{1 - (1 - k^{-1})} \\
&= \frac{\mathrm{opt}}{k} \cdot \frac{1 - (1 - k^{-1})^k}{k^{-1}} \\
&= \Big(1 - (1 - k^{-1})^k\Big) \, \mathrm{opt} .
\end{aligned}$$

3. Considere o seguinte algoritmo guloso para o problema da mochila: selecione um a um os objetos j cujo valor de v_j/w_j é máximo até que a capacidade da mochila seja atingida ou todos os objetos já foram considerados. Mostre que este algoritmo não tem razão de aproximação constante, ou seja, exiba uma família de instâncias (m, n, v, w) do problema para as quais a razão entre o valor da solução produzida pelo algoritmo e $\mathrm{opt}(m, n, v, w)$ é arbitrariamente pequena.

   Solução: Para cada m, considere uma instância com dois objetos: um de peso e valor m e outro de peso $\varepsilon > 0$ e valor $(1+\varepsilon)\varepsilon$. O algoritmo guloso escolherá o objeto de peso ε, pois

   $$\frac{(1+\varepsilon)\varepsilon}{\varepsilon} = 1 + \varepsilon > 1 = \frac{m}{m} .$$

   Mas o ótimo, que consiste apenas do objeto de peso m, tem valor m. Assim, nesse caso, a razão entre o valor da solução produzida pelo algoritmo e o valor de uma solução ótima é $\frac{(1+\varepsilon)\varepsilon}{m}$, que pode ser arbitrariamente pequeno, dependendo da escolha de ε e de m.

4. Considere as três seguintes variantes do TSPM. Na primeira, buscamos um caminho hamiltoniano de custo mínimo no grafo dado. Na segunda, queremos um caminho hamiltoniano de custo mínimo dentre os que começam em um dado vértice s. Na terceira, queremos um caminho hamiltoniano de custo mínimo dentre os que começam em um dado vértice s e terminam em um dado vértice t. Modifique o algoritmo de Christofides para que ele seja um algoritmo de aproximação com razão menor que 2 para cada uma destas variantes.

   Solução: O exercício consistia apenas da segunda variante.

   O seguinte algoritmo é uma 1,5-aproximação polinomial para essa variante.

   Algoritmo TSPM-VERSÃO2 (G, c, s)

   1 seja T uma árvore geradora de custo mínimo em que s tem grau 1
   2 seja I o conjunto dos vértices de grau ímpar de T
   3 seja M um emparelhamento máximo em $G[I] - s$ de custo mínimo.
   4 $T' \leftarrow T + M$
   5 seja P' uma trilha euleriana em T' que começa em s
   6 $P \leftarrow$ ATALHO (P')
   7 devolva P

   A linha 1 pode ser executada em tempo polinomial juntando-se a saída da chamada MST $(G - s, c)$ uma aresta de custo mínimo incidente a s. O conjunto I tem um número par de elementos, já que o número de vértices de grau ímpar em um grafo é sempre par. Ademais, o vértice $s \in I$. A linha 3 também consome tempo polinomial pois podemos aplicar o algoritmo de Edmonds ao grafo $G[I]$ com os custos das arestas incidentes a s alterados para uma constante grande (por exemplo, a soma dos custos das demais arestas do grafo) e ignorar a aresta incidente a s do emparelhamento produzido. A linha 5 pode ser executada já que, no (multi)grafo T', apenas s e mais um vértice têm grau ímpar. As linhas 5 e 6 consistem de procedimentos muito semelhantes aos usados no algoritmo de Christofides e também consomem tempo polinomial.

   Para demonstrar uma razão de aproximação para o algoritmo acima, precisamos de algumas delimitações para o valor ótimo do problema. A primeira envolve T e a segunda envolve M:

   $$\text{opt} \geq c(T) \qquad \text{e} \qquad \text{opt} \geq 2\,c(M)\,.$$

   A primeira delimitação vale já que uma solução ótima do problema é uma árvore em que s tem grau 1. Para a segunda, considere uma solução ótima P^* do problema. Sejam $u_1, \ldots, u_t$ os vértices de $I \setminus \{s\}$ na ordem em que aparecem em P^*. Considere os dois seguintes emparelhamentos: $M_1 := \{(u_1, u_2), \ldots, (u_{t-2}, u_{t-1})\}$ e $M_2 := \{(u_2, u_3), \ldots, (u_{t-1}, u_t)\}$. Note que M_1 e M_2 são emparelhamentos máximos em $G[I] - s$. Por isso, $c(M) \leq \min\{c(M_1), c(M_2)\}$ e

   $$2\,c(M) \;\leq\; c(M_1) + c(M_2) \;\leq\; c(P^*) \;=\; \text{opt}\,.$$

   Essa é a segunda delimitação. Das duas delimitações, é fácil concluir que o algoritmo é uma 1,5-aproximação, já que

   $$c(P) \;\leq\; c(T) + c(M) \;\leq\; \text{opt} + \frac{\text{opt}}{2} \;=\; \frac{3}{2}\,\text{opt}\,.$$