Algoritmos de Aproximação

Segundo Semestre de 2012

Cobertura por conjuntos

Instância:

- conjunto base finito E
- coleção $\mathcal{S} = \{S_1, \ldots, S_m\}$ de subconjuntos de E
- custo $c_j > 0$ para cada S_j em $\mathcal{S}$

Cobertura por conjuntos

Instância:

- conjunto base finito E
- coleção $\mathcal{S} = \{S_1, \ldots, S_m\}$ de subconjuntos de E
- custo $c_j > 0$ para cada S_j em $\mathcal{S}$

Objetivo: encontrar cobertura $\mathcal{S}' \subseteq \mathcal{S}$ de E de custo mínimo.

Cobertura por conjuntos

Instância:

- conjunto base finito E
- coleção $\mathcal{S} = \{S_1, \ldots, S_m\}$ de subconjuntos de E
- custo $c_j > 0$ para cada S_j em $\mathcal{S}$

Objetivo: encontrar cobertura $\mathcal{S}' \subseteq \mathcal{S}$ de E de custo mínimo.

Relaxação linear: Dados E, $\mathcal{S}$ e c, encontrar x que

minimize $\sum_j c_j x_j$

sujeito a $\sum_{j:e\in S_j} x_j \geq 1$ para cada e em E

$x_j \geq 0$ para $j = 1, \ldots, m$

Cobertura por conjuntos

Instância:

- conjunto base finito E
- coleção $\mathcal{S} = \{S_1, \ldots, S_m\}$ de subconjuntos de E
- custo $c_j > 0$ para cada S_j em $\mathcal{S}$

Objetivo: encontrar cobertura $\mathcal{S}' \subseteq \mathcal{S}$ de E de custo mínimo.

Relaxação linear: Dados E, $\mathcal{S}$ e c, encontrar x que
minimize $\sum_j c_j x_j$
sujeito a $\sum_{j : e \in S_j} x_j \geq 1$ para cada e em E
$x_j \geq 0$ para $j = 1, \ldots, m$

Este é o programa primal, que denotaremos por (P).

Relaxação linear e seu dual

Problema: Dados E, $\mathcal{S}$ e c, encontrar cobertura $\mathcal{S}' \subseteq \mathcal{S}$ de E de custo mínimo.

Relaxação linear: Dados E, $\mathcal{S}$ e c, encontrar x que
minimize $\sum_j c_j x_j$
sujeito a $\sum_{j:e \in S_j} x_j \geq 1$ para cada e em E
$x_j \geq 0$ para $j = 1, \ldots, m$

Relaxação linear e seu dual

Problema: Dados E, $\mathcal{S}$ e c, encontrar cobertura $\mathcal{S}' \subseteq \mathcal{S}$ de E de custo mínimo.

Relaxação linear: Dados E, $\mathcal{S}$ e c, encontrar x que

minimize $\sum_j c_j x_j$

sujeito a $\sum_{j:e\in S_j} x_j \geq 1$ para cada e em E

$x_j \geq 0$ para $j = 1, \ldots, m$

Dual: Dados E, $\mathcal{S}$ e c, encontrar y que

maximize $\sum_e y_e$

sujeito a $\sum_{e\in S_j} y_e \leq c_j$ para $j = 1, \ldots, m$

$y_e \geq 0$ para cada $e \in E$

Algoritmo primal-dual

```
PrimalDual (E, 𝒮, c)        ▷ 𝒮 = {S_1, ..., S_m}
    para cada e em E faça y_e ← 0
    I ← {j : ∑_{e∈S_j} y_e = c_j}
    enquanto existe e' que não é coberto por I faça
        ϵ ← min{c_j − ∑_{e∈S_j} y_e : j tal que e' ∈ S_j}
        y_{e'} ← y_{e'} + ϵ
        I ← {j : ∑_{e∈S_j} y_e = c_j}
    devolva I
```

Algoritmo primal-dual

PRIMALDUAL $(E, \mathcal{S}, c)$ $\quad \triangleright\ \mathcal{S} = \{S_1, \ldots, S_m\}$

para cada e em E **faça** $y_e \leftarrow 0$

$I \leftarrow \{j : \sum_{e \in S_j} y_e = c_j\}$

enquanto existe e' que não é coberto por I **faça**

$\quad \epsilon \leftarrow \min\{c_j - \sum_{e \in S_j} y_e : j$ tal que $e' \in S_j\}$

$\quad y_{e'} \leftarrow y_{e'} + \epsilon$

$\quad I \leftarrow \{j : \sum_{e \in S_j} y_e = c_j\}$

devolva I

Claro que I é uma cobertura.

Algoritmo primal-dual

PrimalDual $(E, \mathcal{S}, c)$ $\quad\triangleright\ \mathcal{S} = \{S_1, \ldots, S_m\}$

para cada e em E **faça** $y_e \leftarrow 0$
$I \leftarrow \{j : \sum_{e \in S_j} y_e = c_j\}$
enquanto existe e' que não é coberto por I **faça**
  $\epsilon \leftarrow \min\{c_j - \sum_{e \in S_j} y_e : j$ tal que $e' \in S_j\}$
  $y_{e'} \leftarrow y_{e'} + \epsilon$
  $I \leftarrow \{j : \sum_{e \in S_j} y_e = c_j\}$
devolva I

Claro que I é uma cobertura.

Teorema: PrimalDual é uma f-aproximação.

Prova:

$$\begin{aligned}
\sum_{j \in I} c_j &= \sum_{j \in I} \sum_{e \in S_j} y_e \\
&= \sum_{e \in E} y_e \, |\{j \in I : e \in S_j\}| \\
&\leq f \sum_{e \in E} y_e \ \leq \ f \, \mathrm{opt}.
\end{aligned}$$

Feedback vertex set problem

Instância:

- grafo não-dirigido $G = (V, E)$
- peso $w_i \geq 0$ para cada vértice i em V

Feedback vertex set problem

Instância:

- grafo não-dirigido $G = (V, E)$
- peso $w_i \geq 0$ para cada vértice i em V

Objetivo: encontrar conjunto de vértices S tal que $G[V \setminus S]$ não tem circuitos e S tem peso mínimo.

Feedback vertex set problem

Instância:

- grafo não-dirigido $G = (V, E)$
- peso $w_i \geq 0$ para cada vértice i em V

Objetivo: encontrar conjunto de vértices S tal que $G[V \setminus S]$ não tem circuitos e S tem peso mínimo.

Formulação como programa linear inteiro:

Dados $G = (V, E)$ e w, encontrar x que

minimize $\sum_{i \in V} w_i \, x_i$

sujeito a $\sum_{i \in C} x_i \geq 1$ para cada C em $\mathcal{C}$

$x_i \in \{0, 1\}$ para cada i em V,

onde $\mathcal{C}$ a coleção de circuitos de G.

Relaxação linear e seu dual

Problema: Dados $G = (V, E)$ e w, encontrar $S \subseteq V$ tq $G[V \setminus S]$ não tem circuitos e S tem peso mínimo.

Relaxação linear (P): Dados G e w, encontrar x que

minimize $\sum_{i \in V} w_i \, x_i$
sujeito a $\sum_{i \in C} x_i \geq 1$ para cada C em $\mathcal{C}$
$x_i \geq 0$ para cada vértice i,

onde $\mathcal{C}$ a coleção de circuitos de G.

Relaxação linear e seu dual

Problema: Dados $G = (V, E)$ e w, encontrar $S \subseteq V$ tq $G[V \setminus S]$ não tem circuitos e S tem peso mínimo.

Relaxação linear (P): Dados G e w, encontrar x que

minimize $\sum_{i \in V} w_i x_i$
sujeito a $\sum_{i \in C} x_i \geq 1$ para cada C em $\mathcal{C}$
$x_i \geq 0$ para cada vértice i,

onde $\mathcal{C}$ a coleção de circuitos de G.

Dual (D): Dados G e w, encontrar y que

maximize $\sum_C y_C$
sujeito a $\sum_{C \in \mathcal{C}:\, i \in C} y_C \leq w_i$ para cada vértice i
$y_C \geq 0$ para cada C em $\mathcal{C}$.

Primeira tentativa

PRIMALDUAL (G, w)

para cada C em $\mathcal{C}$ **faça** $y_C \leftarrow 0$
$S \leftarrow \emptyset$
enquanto existe circuito C em G **faça**
  $\epsilon \leftarrow \min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$
  $\ell \leftarrow \arg\min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$
  $y_C \leftarrow y_C + \epsilon$
  $S \leftarrow S \cup \{\ell\}$
  remova ℓ de G
  repetidamente remova folhas de G
devolva S

Primeira tentativa

PRIMALDUAL (G, w)

para cada C em $\mathcal{C}$ **faça** $y_C \leftarrow 0$
$S \leftarrow \emptyset$
enquanto existe circuito C em G **faça**
    $\epsilon \leftarrow \min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$
    $\ell \leftarrow \arg\min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$
    $y_C \leftarrow y_C + \epsilon$
    $S \leftarrow S \cup \{\ell\}$
    remova ℓ de G
    repetidamente remova folhas de G
devolva S

Claro que S é um feedback vertex set.

Primeira tentativa

PRIMALDUAL (G, w)

```
para cada C em 𝒞 faça y_C ← 0
S ← ∅
enquanto existe circuito C em G faça
      ε ← min{w_i − Σ_{C':i∈C'} y_{C'} : i ∈ C}
      ℓ ← arg min{w_i − Σ_{C':i∈C'} y_{C'} : i ∈ C}
      y_C ← y_C + ε
      S ← S ∪ {ℓ}
      remova ℓ de G
      repetidamente remova folhas de G
devolva S
```

Será que PRIMALDUAL é uma ???-aproximação??

$$\begin{aligned} \sum_{i \in S} w_i &= \sum_{i \in S} \sum_{C : i \in C} y_C = \sum_{C \in \mathcal{C}} y_C \, |S \cap C| \\ &\leq\ ??? \sum_{C \in \mathcal{C}} y_C \ \leq\ ???\, \mathrm{opt}. \end{aligned}$$

Algoritmo primal-dual

PRIMALDUAL (G, w) $\quad \triangleright \ n := |V_G|$

para cada C em $\mathcal{C}$ **faça** $y_C \leftarrow 0$

$S \leftarrow \emptyset$

repetidamente remova folhas de G

enquanto existem circuitos em G **faça**

   $C \leftarrow$ circuito com $\leq 2\lceil \lg n \rceil$ vértices de grau ≥ 3

   $\epsilon \leftarrow \min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$

   $\ell \leftarrow \arg\min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$

   $y_C \leftarrow y_C + \epsilon$

   $S \leftarrow S \cup \{\ell\}$

   remova ℓ de G

   repetidamente remova folhas de G

devolva S

Algoritmo primal-dual

PRIMALDUAL (G, w) $\quad \triangleright \; n := |V_G|$

para cada C em $\mathcal{C}$ **faça** $y_C \leftarrow 0$

$S \leftarrow \emptyset$

repetidamente remova folhas de G

enquanto existem circuitos em G **faça**

  $C \leftarrow$ circuito com $\leq 2\lceil \lg n \rceil$ vértices de grau ≥ 3

  $\epsilon \leftarrow \min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$

  $\ell \leftarrow \arg\min\{w_i - \sum_{C' : i \in C'} y_{C'} : i \in C\}$

  $y_C \leftarrow y_C + \epsilon$

  $S \leftarrow S \cup \{\ell\}$

  remova ℓ de G

  repetidamente remova folhas de G

devolva S

Teorema: PRIMALDUAL é uma $4\lceil \lg n \rceil$-aproximação.

Análise

- Se P é um caminho com todos os vértices de grau 2 em G, então no máximo um vértice de P está em S.

Análise

- Se P é um caminho com todos os vértices de grau 2 em G, então no máximo um vértice de P está em S.

- Para todo circuito C, vale que $|S \cap C| \leq 2k$, onde k é o número de vértices de grau pelo menos 3 em C.

Análise

- Se P é um caminho com todos os vértices de grau 2 em G, então no máximo um vértice de P está em S.

- Para todo circuito C, vale que $|S \cap C| \leq 2k$, onde k é o número de vértices de grau pelo menos 3 em C.

- Em qualquer grafo cujos vértices tem grau ≥ 2, existe um circuito com $\leq 2\lceil \lg n \rceil$ vértices de grau ≥ 3.