Algoritmos Parametrizados

Programação Linear e Ramificação

Lehilton Pedrosa

Segundo Semestre de 2016

Instituto de Computação – Unicamp

Roteiro

1. Programação Linear
2. Núcleo baseado em programação linear
3. Cobertura por vértice (excedendo PL)
4. Transversal para ciclos ímpares
5. Problema da cadeia de caracteres mais próxima

Programação Linear

Programação Linear

Programa linear

Um programa linear (PL) é formado por

- uma função linear em variáveis reais;
- conjunto de desigualdades lineares nas variáveis;
- um objetivo maximização ou minimização

Exemplo:

$$\begin{array}{rl} \text{minimize} & 2x + y \\ \text{sujeito a} & x - 2y \geq 3 \\ & 2x - y \geq 0 \\ & x \geq 0. \end{array}$$

Teorema (Khachiyan)

Programação linear é polinomial.

Programação linear inteiro

Programa linear inteiro

Um programa linear inteiro (PLI) é um programa linear com a restrição adicional que variáveis devem ser inteiras.

PLI para Cobertura por vértices

PLI é útil para para formular problemas:

Cobertura por vértices

$$\begin{array}{rll} \text{minimize} & \sum_{v \in V(G)} x_v & \\ \text{sujeito a} & x_u + x_v \geq 1 & \text{para todo } uv \in E(G) \\ & 0 \leq x_v \leq 1 & \text{para todo } v \in V(G) \\ & x_v \in \mathbb{Z} & \text{para todo } v \in V(G) \end{array}$$

Teorema

Seja $vc(G)$ o valor ótimo do programa acima. Então existe uma cobertura por vértices de tamanho k se, e somente se, $vc(G) \leq k$.

Corolário

Programação linear inteira é $\mathcal{NP}$-difícil.

Relaxação

Se removemos as restrições de integralidade:

$$\begin{array}{rll} \text{minimize} & \sum_{v \in V(G)} x_v & \\ \text{sujeito a} & x_u + x_v \geq 1 & \text{para todo } uv \in E(G) \\ & 0 \leq x_v \leq 1 & \text{para todo } v \in V(G) \end{array}$$

Observações

- interpretamos x_v como a "parte" escolhida de v
- se $\mathrm{vc}^*(G)$ é a solução ótimo do PL, então $\mathrm{vc}^*(G) \leq \mathrm{vc}(G)$
- se G é o triângulo, então $\mathrm{vc}(G) = \frac{3}{2}\mathrm{vc}^*(G)$
  - → esse fator é chamado de desvio de integralidade de G
  - → o maior desvio de integralidade entre todo G é o desvio de integralidade do PL

Exercício: Mostre que o desvio de integralidade do PL é $2 - \frac{2}{n}$.

Estrutura da solução ótima

Seja x uma solução ótima do PL. Defina:

- $V_0 = \{v \in V(G) : x_v < \frac{1}{2}\}$;
- $V_{1/2} = \{v \in V(G) : x_v = \frac{1}{2}\}$;
- $V_1 = \{v \in V(G) : x_v > \frac{1}{2}\}$;

Teorema (Nemhauser-Trotter)

Existe uma cobertura por vértices mínima S de G tal que

$$V_1 \subseteq S \subseteq V_1 \cup V_{1/2}.$$

Núcleo baseado em programação linear

Redução baseada em PL

Observamos que o PL resolve a parte "fácil" do problema:

Redução VC.4: Seja x uma solução ótima do PL para instância (G, k):

- se $\sum_{v \in V(G)} x_v > k$, então devolva não;
- senão, devolva $(G - V_0 - V_1, k - |V_1|)$.

Lema

A redução VC.4 é segura.

Núcleo baseado em PL

Teorema

Cobertura por vértices admite um núcleo com $2k$ vértices.

Resolvendo o PL usando emparelhamento

Lema

Seja G um grafo com n vértices e m arestas. Uma solução de PL pode ser encontrada em tempo $\mathcal{O}(m\sqrt{n})$.

Consequências

Corolário

Cobertura por vértices admite um núcleo com $2k$ vértices que pode ser computado em tempo $\mathcal{O}(m\sqrt{n})$.

Corolário

Existe uma solução ótima do PL tal que $x_v \in \{0, \frac{1}{2}, 1\}$ para todo v e essa solução pode ser encontrada em tempo $\mathcal{O}(m\sqrt{n})$

A solução acima é chamada de semi-integral.

Cobertura por vértice (excedendo PL)

Mudando o parâmetro

Situação atual: núcleo com $2k$ vértices

Analisando:

1. se k é pequeno: núcleo é muito bom
2. se k é grande: núcleo não adianta muito

Fatos

- Para instância sim, $\mathrm{vc}(G) \geq \mathrm{vc}^*(G)$
- Podemos calcular $\mathrm{vc}^*(G)$ em tempo polinomial

Parâmetro alternativo: $\mu(G, k) := k - \mathrm{vc}^*(G)$

$\rightarrow$ o que excede ou está acima do PL.

Revisitando nossa redução

Lema

Suponha que obtemos (G', k') *após aplicar* VC.4 *sobre* (G, k) *(utilizando uma solução semi-integral* x*). Então* $vc^*(G) - vc^*(G') = vc(G) - vc(G') = V_1^x = k - k'$.

Consequência: $\mu(G', k') = \mu(G, k)$.

Reduzindo iterativamente

Lema

Dado G, existe um algoritmo que em tempo $\mathcal{O}(mn^{3/2})$:

- *encontra solução semi-integral de $PL(G)$ que não é toda $1/2$;*
- *conclui que $x_v = 1/2$ para $v \in V(G)$ é a única solução de $PL(G)$.*

Redução VC.5: Se existe solução que não é $x_v = 1/2$ para $v \in V(G)$, então aplique VC.4.

Algoritmo de ramificação

Teorema

Existe um algoritmo para Cobertura por vértices (excedendo PL) com tempo $4^{k-vc^*(G)} \cdot n^{\mathcal{O}(1)}$.

Usando emparelhamento

Observações

Seja M um emparelhamento máximo de G:

- $|M|$ é um limitante inferior da cobertura
- $\mathrm{vc}^*(G) \geq |M|$ (por dualidade)

  $\Rightarrow\ k - \mathrm{vc}^*(G) \leq k - |M|$

Teorema

Existe um algoritmo para Cobertura por vértices (excedendo emparelhamento) com tempo $4^{k-\mathrm{vc}^*(G)} \cdot n^{\mathcal{O}(1)}$.

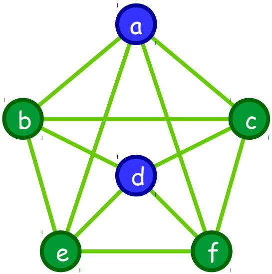
Transversal para ciclos ímpares

Transversal para ciclos ímpares

Dado G, um conjunto de vértices X é uma transversal para ciclos ímpares se $G - X$ é um grafo bipartido.

Problema da transversal para ciclos ímpares mínima

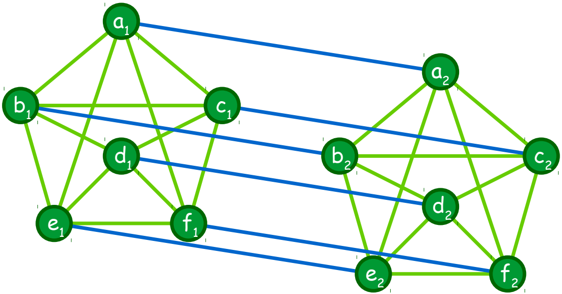
Dado G um grafo não direcionado e um inteiro k. Existe uma transversal para ciclos ímpares de tamanho k?



Redução para Cobertura por vértices

Crie um grafo $\tilde{G}$ tal que:

- $\tilde{G}$ contém duas cópias de $V(G)$: $V(\tilde{G}) = V_1 \cup V_2$
- $u_1v_2 \in E(\tilde{G})$ para toda aresta $uv \in E(G)$
- $u_1u_2 \in E(\tilde{G})$ para todo vértice $u \in V(G)$



Redução (cont.)

Lema

G tem uma transversal para ciclos ímpares de tamanho k sss $\tilde{G}$ tem uma cobertura por vértices de tamanho $k + n$, onde $n = |V(G)|$.

FPT para Transversal para ciclos

Teorema

Transversal para ciclos ímpares pode ser resolvida em tempo $4^k n^{\mathcal{O}(1)}$.

Problema da cadeia de caracteres mais próxima

Problema da cadeia mais próxima

Dadas duas cadeias de caracteres x e y de tamanho L, a distância de Hamming, $d_H(x, y)$ entre x e y é a quantidade de índices em que $x[j] \neq y[j]$.

x = bananada $\qquad d_H(x,y) = 3$

y = camarada $\qquad L := |y| = |x| = 8$

Problema da cadeia de caracteres mais próxima

Dado um conjunto de k cadeias de caracteres $x_1, \ldots, x_k$ com tamanho L e um inteiro d. Existe uma cadeia de caracteres y, de tamanho L, tal que $d_H(x_i, y) \leq d$ para todo $1 \leq i \leq k$?

Uma tal cadeia de caracteres y é chamada de centro.

Exemplo de instância: cadeias

$x =$ bananada $\quad d_H(x,A) = 2$

$y =$ camarada $\quad d_H(y,A) = 1$

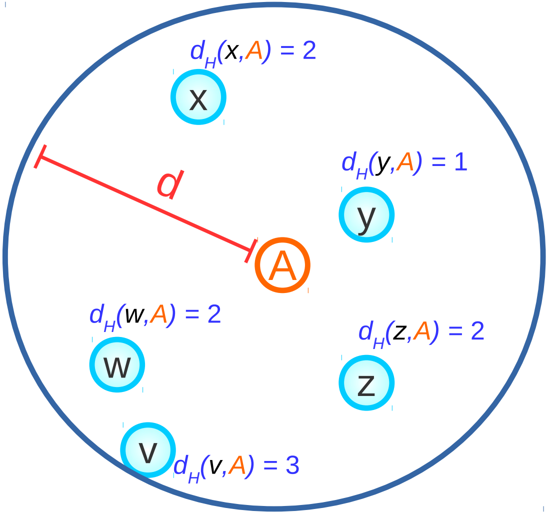
$z =$ rabanada $\quad d_H(z,A) = 2$

$w =$ laminada $\quad d_H(w,A) = 2$

$v =$ laminado $\quad d_H(v,A) = 3$

$A =$ camanada

Exemplo de instância: centro



Matriz de caracteres

coluna 3

linha 2

```
b a n a n a d a
c a m a r a d a
r a b a n a d a
l a m i n a d a
l a m i n a d o
```

Representamos o conjunto de cadeias como uma matriz de caracteres.

Identificando a parte fácil

coluna boa

```
b a n a n a d a
c a m a r a d a
r a b a n a d a
l a m i n a d a
l a m i n a d o
```

coluna ruim

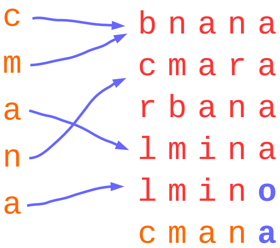
- colunas boas são colunas com um único caracteres
- colunas ruim são colunas com caracteres diferentes

Removendo colunas boas

```
b n a n a
c m a r a
r b a n a
l m i n a
l m i n o
```

Redução CS.1: Remova todas colunas boas.

Contando colunas ruins



Lema

Seja uma instância sim para o Problema de cadeia de caracteres mais próxima. Então o número de colunas ruins é no máximo $k \times L$.

Algoritmo de ramificação para Cadeia mais próxima

Teorema

O problema da cadeia de caracteres mais próxima pode ser resolvido em tempo $\mathcal{O}(kL + kd(d+1)^d)$.