

Relatório Trabalho PLI - Análise de Algoritmo II

Vitor Hugo Almeida Marques

30 de junho de 2005

Sumário

1	Problema do EquiCorte	1
2	Problema da Coleta de Lixo	2
3	Problema da Cadeia de Restaurante	3
4	Problema da Distribuição de Combustível	3
5	Conclusão	4

1 Problema do EquiCorte

Variáveis do problema

Variáveis de Entrada

n : número de vértices do grafo de entrada

$arestas(i,j)$: matriz de adjacência que define se existe uma aresta entre os vértices i e j

Variáveis de resultado

$x(i)$: variável binária que diz se o vértice i está ou não no (equicorte)

$d(i,j)$: variável binária que diz se a aresta entre os vértices i e j está ou não no equicorte.

Restrições do problema

- R1: O número de vértices do corte deve ser $n/2$ arredondado para baixo.
(definição de igualdade do corte).
$$\sum_{i=1}^n x(i) = floor(n/2)$$
- R2: As próximas restrições agrupei juntas, pois, as mesmas tem finalidade única: quando uma aresta está no equicorte obrigatoriamente somente um dos vértices da aresta pode estar no equicorte. Caso contrário, os dois podem estar ou nenhum . Claramente o $d(i,j)$ é um resultado do *xor* entre $x(i)$ e $x(j)$.
$$\forall(i,j) \text{ onde } arestas(i,j) = 1$$
$$d(i,j) \leq x(i) + x(j)$$

$$\begin{aligned}
d(i, j) &\geq x(i) - x(j) \\
d(i, j) &\geq x(j) - x(i) \\
d(i, j) &\leq 2 - x(i) - x(j)
\end{aligned}$$

Função Objetivo

Como estamos interessados no menor equicorte (com menor número de arestas), devemos minimizar o somatório do número de arestas.

$$\sum_{i=1}^n \sum_{j=1}^n d(i, j)$$

2 Problema da Coleta de Lixo

Variáveis do problema

Variáveis de Entrada

n: número de setores

m: número de depósitos

custo-km: custo de transporte por km por m^3 de lixo transportado

capacidade: capacidade de cada depósito i

req: produção anual de lixo de cada setor j

custo-fixo: custo fixo de abertura de um depósito i

distancia: distancia entre um setor j e um deposito i

Variáveis de resultado

$x(i)$: variável binária que diz se o depósito de lixo i é ou não aberto.

$d(i,j)$: variável binária que diz se o depósito i atende ou não o setor j .

Restrições do problema

- R1: Esta restrição impõe que um setor só seja atendido por um depósito como definido no enunciado

$$\sum_{i=1}^m d(i, j) = 1, \forall j$$

- R2: Esta restrição impõe para que um depósito atenda um setor o depósito deve estar instalado (relacionar $x(i)$ com $d(i,j)$)

$$d(i, j) \leq x(i), \forall i, j$$

- R3: a capacidade de um depósito deve ser respeitada, ou seja, a soma de todos os setores j depositantes no depósito i deve ser menor ou igual a capacidade do depósito.

$$\sum_{j=1}^n (d(i, j) * req(j)) \leq capacidade(i), \forall i.$$

Função Objetivo

A função objetivo a ser minimizada é a soma do custo fixo de instalação de um depósito, logicamente somente os que são instalados, por isto a multiplicação por $x(i)$ e o custo variável, que depende da distância entre um depósito i e um

setor j e também da quantidade transportada, uma vez que custo-km define o custo de transporte por km por m^3 de lixo transportado.

OBS: É multiplicado por 1000 pois os dados de entradas relacionados a quantidade indicam milhares de metros cúbicos.

$$\sum_{i=1}^m \underbrace{(x(i) * custo - fixo(i))}_{custofixo} + \sum_{i=1}^m \sum_{j=1}^n \underbrace{(d(i,j) * req(j) * distancia(j,i) * 1000 * custo - km)}_{custovariavel}$$

3 Problema da Cadeia de Restaurante

Variáveis do problema

Variáveis de Entrada

n : números de locais para implantação das cadeias de restaurantes

NUM-CADEIAS = 2: parâmetro de entrada

lucro: lucro da abertura no local j da cadeia de restaurantes i

Menos5: se um local dista menos de 5Km de outro local de implantação.

Variáveis de resultado

$x(i,j)$: variável binária que diz se uma cadeia i é ou não instalada no local j

Restrições do problema

- R1: um local não pode ter a duas cadeias instaladas

$$\sum_{i=1}^{NUM-CADEIAS} x(i,j) \leq 1, \forall j$$

- R2: restaurantes de mesma cadeia não podem ser instalados em locais que distam menos de 5km.

$$(x(i,j_1) + x(i,j_2)) \leq 1, \forall (i,j_1,j_2) | Menos5(j_1,j_2) = 1$$

- R3(Restrição adicional letra b) : Se for instalada uma cadeia num local então obrigatoriamente instalar outra cadeia em um local que dista menos de 5 km. Ou seja, basta um local que dista 5km, por isto o uso do somatório. Os índices i e $3-i$ faz a intercalação das cadeias.

$$x(i,j_1) \leq \sum_{j_2=1}^n (x(3-i,j_2) * Menos5(j_1,j_2)), \forall (i,j_1)$$

Função Objetivo

A função objetivo representa o lucro da abertura das cadeias de restaurante no locais segundo as restrições e como almejamos o maior lucro a mesma deve ser maximizada.

$$\sum_{i=1}^{NUM-CADEIAS} \sum_{j=1}^n (x(i,j) * lucro(j,i))$$

4 Problema da Distribuição de Combustível

Variáveis do problema

Variáveis de Entrada

n : números de compartimentos do caminhão

m: número de combustíveis
capacidade: capacidade do compartimento j
demanda: demanda do combustível i .

Variáveis de resultado

$x(i,j)$: variável binária que diz se um combustível i é ou não levado pelo compartimento j

$d(i,j)$: variável inteira que diz quanto de um combustível i é levado pelo compartimento j

Restrições do problema

- R1: Esta restrição impõem que um compartimento só leve um tipo de combustível

$$\sum_{i=1}^m x(i, j) \leq 1, \forall j$$

- R2: A quantidade de combustível de um tipo não deve ultrapassar a necessidade do mesmo.

$$\sum_{j=1}^n d(i, j) \leq demanda(i), \forall i.$$

- R3: relacionar $x(i,j)$ com $d(i,j)$. $d(i,j)$ só pode ser maior que zero, limitado pela capacidade do compartimento, caso $x(i,j)$ seja 1.

$$d(i, j) \leq (x(i, j) * capacidade(j)), \forall (i, j).$$

Função Objetivo

O objetivo do problema é deixar de levar a menor quantidade de combustível possível. Logo, deve-se minimizar a diferença entre o total da demanda e o total levado, ou maximizar o total levado, como devo exibir a quantidade não levada, optei pela primeira opção.

$$\sum_{i=1}^m demanda(i) - \sum_{i=1}^m \sum_{j=1}^n d(i, j)$$

5 Conclusão

A principal dificuldade do trabalho foi modelar as restrições para corte. Por bastante tempo tentei fazer algo que esbarrava nas limitações da linguagem. Consegui obter a resposta para o que eu queria na Internet, buscando pela modelagem de XOR em PLI e achei algo que eu já me aproximava, que é o conjunto de restrições (R2) do problema do EquiCorte.

A outra dificuldade foi em alguns teste, que diferia um pouco o resultado. Acontece que a forma de leitura das matrizes independe da forma que a mesma esta visivelmente disposta no arquivo, e isto me fez interpretar duas vezes equivocadamente os indices($distancia(i,j)$ - P2 e $lucro(i,j)$ - P3)

Quanto aos problemas os mesmos se mostraram interessantes e complementaram bem o aprendizado da primeira parte da matéria.