O Problema de Localização de Instalações

Autores: Giovana, Nereida, Mariana

Universidade de Sao Paulo

4 de dezembro de 2015

Roteiro

1. Introdução
   - Importância do problema
   - Descrição do problema
2. Uncapacitated Facility Location Problem
3. Algoritmo de Aproximação JMS
   - Algoritmo Primal-Dual
   - Razão de aproximação JMS
4. Algoritmo de Metaheurístico $TABU$

Introdução

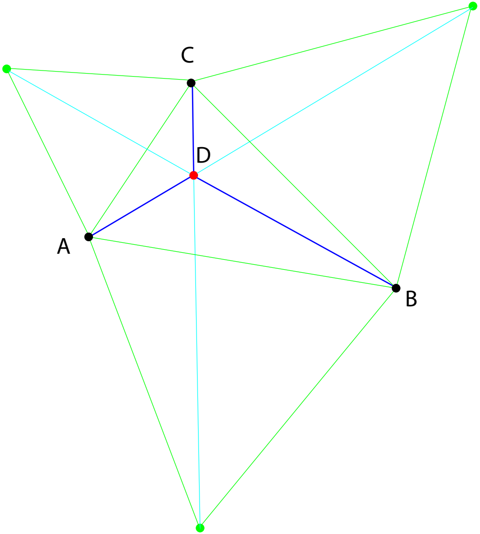
Importância do problema

Selecionar a localização de instalações pode ser visto como um problema de decisão econômica.

Descrição do problema

O problema de localização de instalações pode ser resumido em escolher a posição de um conjunto de instalações tendo como objetivo conectar os clientes com as instalações minimizando os custos.

Descrição do problema



Uncapacitated Facility Location Problem

$UFLP(C, \mathcal{F}, f_i, c_{ij})$ é definido como:

- conjunto finito de m clientes C
- conjunto finito de n instalações $\mathcal{F}$
- custos fixos por abertura f_i
- custo de serviço ou conexão c_{ij}

Onde procuramos:

- subconjunto S de instalações (abertas)
- conectar os clientes às instalações abertas
- minimizar os custos de abertura e de conexão

$$custo(S) = \sum_{i \in S} f_i + \sum_{j \in C} \min_{i' \in S} c_{i'j}$$

NP-Dificil

Problema 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 $\mathbb{Q}$ para cada S em $\mathcal{S}$, encontrar uma cobertura $\mathcal{T}$ de E que minimize $c(\mathcal{T})$.

NP-Dificil

Problema 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 $\mathbb{Q}$ para cada S em $\mathcal{S}$, encontrar uma cobertura $\mathcal{T}$ de E que minimize $c(\mathcal{T})$.

Podemos escrever o $UFLP(C, \mathcal{F}, f_i, c_{ij})$ como um instância $(E, \mathcal{S}, c)$ do problema de cobertura de conjuntos, basta definirmos:
$\mathcal{C} := E$ conjunto de clientes, $\mathcal{F} := \mathcal{S}$ e $f_s := c(S)$, para $S \subset \mathcal{S}$.
O custo de conexão c_{ij} como: zero, se $j \in S \subset \mathcal{S}$ ou ∞, se $j \in E \setminus S$.

Algoritmo de Aproximação JMS

O algoritmo proposto por **Jain**, **Mahdian** e **Saberi**, JMS, tem como ideia central que clientes já conectados, ainda podem oferecer certa quantia a outras instalações. Isso se a possibilidade de reconexão com alguma dessas novas instalações oferece algum benefício ou se elas estão mais próximas.

Algoritmo de Aproximação JMS

j desconectados, i fechadas, $t = 0$, e orçamento de j, $B_j = 0$.

1. Em cada tempo, cada um dos clientes oferece alguma quantia do seu orçamento à cada instalação fechada.
   - se j é não conectado, a quantia oferecida é o $\max(B_j - c_{ij}, 0)$
   - se j está conectado com a instalação i', o cliente oferece para a instalação i o $\max(c_{i'j} - c_{ij}, 0)$
2. Enquanto j não esteja conectado, incrementar t e B_j com mesma taxa, até que:
   - Para i não aberta a quantia oferecida seja igual a f_i.
   - Para j não conectado e i já aberta, B_j seja igual a c_{ij}.

Algoritmo Primal-Dual

Esse problema pode ser escrito como programação inteira:

$$
\begin{aligned}
\text{minimizar} \quad & \sum_{S \in \mathcal{S}} c_S x_S \\
\text{sujeito a} \quad & \forall j \in \mathcal{C} : \sum_{S : j \in S} x_S \geq 1 \\
& \forall S \in \mathcal{S} : x_S \in \{0, 1\}
\end{aligned}
\tag{1}
$$

Onde $x_s = 1$ se a estrela S é escolhida e $x_s = 0$ caso contrário.

Algoritmo Primal-Dual

A relaxação linear do problema é:

$$
\begin{aligned}
\text{minimizar} \quad & \sum_{S \in \mathcal{S}} c_S x_S \\
\text{sujeito a} \quad & \forall j \in \mathcal{C} : \sum_{S : j \in S} x_S \geq 1 \\
& \forall S \in \mathcal{S} : x_S \geq 0
\end{aligned}
\tag{2}
$$

Algoritmo Primal-Dual

E o dual do problema é:

$$\begin{aligned} \text{maximizar} \quad & \sum_{j \in \mathcal{C}} \alpha_j \\ \text{sujeito a} \quad & \forall S \in \mathcal{S} : \sum_{j \in S \cap \mathcal{C}} \alpha_j \leq c_S \\ & \forall j \in \mathcal{C} : \alpha_j \geq 0 \end{aligned} \tag{3}$$

Onde α_j é quanto j contribuiu no custo total da conexão.

Algoritmo Primal-Dual

Solução para $UFLP$ depende da quota α_j :

$$T = \sum_{j \in \mathcal{C}} \alpha_j$$

Para as estrelas:

$$\sum_{j \in S \cap \mathcal{C}} \alpha_j \leq \gamma \mathcal{C}_S \tag{4}$$

Algoritmo Primal-Dual

Solução para $UFLP$ depende da quota α_j :

$$T = \sum_{j \in \mathcal{C}} \alpha_j$$

Para as estrelas:

$$\sum_{j \in S \cap \mathcal{C}} \alpha_j \leq \gamma \mathcal{C}_S \tag{4}$$

Sendo A a coleção de clientes j conectados a cada instalação i aberta na solução ótima:

$$\sum_{j \in A} \alpha_j \leq \gamma(f_i + \sum_{j \in A} c_{ij}) \tag{5}$$

onde γ, que é um número fixo ≥ 1, é a taxa de aproximação do algoritmo LP.

Razão de aproximação JMS

Estrela S está formada por uma instalação i é um conjunto de m clientes.

- Custo de abertura de instalação é f.
- Custo de conexão entre i e um cliente j, d_j.
- A quota de j do custo total da solução é α_j.

Assumindo que as quotas seguem a ordem:

$$\alpha_1 \leq \alpha_2 \leq ... \leq \alpha_m \tag{6}$$

Razão de aproximação JMS

Momento antes de i ser conectado $t = \alpha_i - \epsilon$.
Para cada $j < i$:

- Se j está conectado a alguma instalação => $r_{j,i} = dj$
- Cc. $r_{j,i} := \alpha_j$ Isso se e somente se $\alpha_i = \alpha_j$

Razão de aproximação JMS

Momento antes de i ser conectado $t = \alpha_i - \epsilon$.
Para cada $j < i$:

- Se j está conectado a alguma instalação => $r_{j,i} = dj$
- Cc. $r_{j,i} := \alpha_j$ Isso se e somente se $\alpha_i = \alpha_j$

j conectou-se com uma instalação:

- B_j é constante,
- j não pode retirar sua contribuição,
- j não pode reconectar-se com is com um maior custo de conexão.

Razão de aproximação JMS

Momento antes de i ser conectado $t = \alpha_i - \epsilon$.
Para cada $j < i$:

- Se j está conectado a alguma instalação => $r_{j,i} = dj$
- Cc. $r_{j,i} := \alpha_j$ Isso se e somente se $\alpha_i = \alpha_j$

j conectou-se com uma instalação:

- B_j é constante,
- j não pode retirar sua contribuição,
- j não pode reconectar-se com is com um maior custo de conexão.

Para cada j:

$$r_{j,j+1} \geq r_{j,j+2} \geq ...r_{j,m} \tag{7}$$

Razão de aproximação JMS

Em $t = \alpha_i - \epsilon$, a quantidade que j oferece a f é :

$$\begin{array}{ll} \max(r_{j,i} - d_j, 0) & \text{se } j < i, \text{e} \\ \max(t - d_j, 0) & \text{se } j \geq i. \end{array} \tag{8}$$

Razão de aproximação JMS

Em $t = \alpha_i - \epsilon$, a quantidade que j oferece a f é :

$$\begin{aligned} &\max(r_{j,i} - d_j, 0) \quad \text{se } j < i, \text{e} \\ &\max(t - d_j, 0) \qquad \text{se } j \geq i. \end{aligned} \tag{8}$$

A \$ que j oferece a i não pode ser $>$ que f. Assim, para todo i:

$$\sum_{j=1}^{i-1} \max(r_{j,i} - dj, 0) + \sum_{j=i}^{m} \max(\alpha_i - dj, 0) \leq f \tag{9}$$

Razão de aproximação JMS

Para $j < i$:

Sendo f' a instalação na qual j está conectado em $t = \alpha_i - \epsilon$.

- O custo de conexão $c_{f'i}$ é no máximo $r_{j,i} + d_i + d_j$
- $c_{f'i}$ não pode ser menor que t

Razão de aproximação JMS

Para $j < i$:

Sendo f' a instalação na qual j está conectado em $t = \alpha_i - \epsilon$.

- O custo de conexão $c_{f'i}$ é no máximo $r_{j,i} + d_i + d_j$
- $c_{f'i}$ não pode ser menor que t

Quando j não esta conectado, $\alpha_i = \alpha_j$

- $r_{j,i} + d_i + d_j$ não é maior que t

Assim para $1 \leq j < i \leq k$:

$$\alpha_i \leq r_{ji} + d_i + d_j \tag{10}$$

Razão de aproximação JMS

Programa de fator de revelação LP.

$$
\begin{array}{rl}
\text{maximizar} & \dfrac{\sum_{i=1}^{m} \alpha_i}{f + \sum_{i=1}^{m} d_i} \\
\text{sujeito a} & \forall\ 1 \le i < m : \alpha_i \le \alpha_{i+1} \\
& \forall\ 1 \le j < i < m : r_{j,i} \ge r_{j,i} + 1 \\
& \forall\ 1 \le j < i < m : \alpha_i \le r_{j,i} + d_i + d_j \\
& \forall\ 1 \le i \le m : \displaystyle\sum_{j=1}^{i-1} \max(r_{j,i} - d_j, 0) + \sum_{j=i}^{i-1} \max(\alpha_i - d_j, 0) \le f \\
& \forall\ 1 \le j \le i \le m : \alpha_j, d_j, f, r_{j,i}
\end{array}
\tag{11}
$$

Algoritmo de Aproximação JMS

Lema 1

O custo da solução encontrada pelo algoritmo é igual a soma de α_j's (soma das quotas de cada cliente).

Algoritmo de Aproximação JMS

Lema 1

O custo da solução encontrada pelo algoritmo é igual a soma de α_j's (soma das quotas de cada cliente).

Lema 2

Se z_m é a solução do programa de revelação LP, para cada estrela S composta por uma instalação e m clientes, a soma de $\alpha_j's$ dos clientes S no algoritmo JMS é no máximo $z_m c_s$

Algoritmo de Aproximação JMS

Lema 1

O custo da solução encontrada pelo algoritmo é igual a soma de α_j's (soma das quotas de cada cliente).

Lema 2

Se z_m é a solução do programa de revelação LP, para cada estrela S composta por uma instalação e m clientes, a soma de $\alpha_j's$ dos clientes S no algoritmo JMS é no máximo $z_m c_s$

Lema 3

Seja z_m o conjunto de soluções do fator revelação LP e $\gamma := sup_m z_m$. Logo, o algoritmo resolve o problema métrico de localização de instalações com um fator de aproximação γ.

Algoritmo de Metaheurístico $TABU$

Ideia Geral

Começando por S uma solução factível, o algoritmo altera o estado (aberta ou fechada) de instalações para ver se essa mudança diminui ou não o custo de S, respeitando uma lista tabu de instalações que não podem ser alteradas.

Algoritmo de Metaheurístico $TABU$: Definições previas

Uma vizinhança $N(S) = \{S(i) | S(i) = S \setminus \{i\}$ fechando i

ou $S(i) = S \cup \{i\}$ abrindo i $\}$

$$MelhorGanho(S) = Custo(S) - \max_{S(i) \in N(S)} Custo(S(i))$$

$$MelhoresAlteracoes(S) = \{i | Custo(S(i)) = \max_{S(i) \in N(S)} Custo(S(i))\}$$

Algoritmo de Metaheurístico $TABU$: Pseudocodigo

Algoritmo 1: Busca Tabu

```
início
    S ← uma solução viável;
    custo(S*) = ∞ ;
    repita
        se MelhorGanho > 0 então
            Aplica alteração aleatória com MelhoresAlteracoes, atualizar a lista tabu e seu
            tamanho;
        senão
            Fecha instalação aleatória;
            Atualiza S - conexões de cidades e estruturas de dados;
        fim
        se (custo(S) < custo(S*)) então
            do S* ← S ;
    até S* mudou nas últimas 500 iterações;
fim
```

Comparação Numérica

- Galvão e Raggi : Casos métricos menores, onde $n = m$, c_{ij} caminho mais curto e f_i é dadas por uma distribuição normal.
- K -median: instâncias de grande escala, onde $n = m$ são pontos independentes num plano, c_{ij} são as distâncias euclidiana e f_i são mesmos para todas as i.

Comparação Numérica

Tabela: Galvão

Instância	Ótimo	JMS			TABU		
		Tempo	< Custo >	Erro %	Tempo	< Custo >	Erro %
50	175802	0,001	175802	0,000	0,002	175802	0,000
70	238515	0,000	238515	0,000	0,003	238515	0,000
100	623551	0,001	623551	0,000	0,004	623551	0,000
150	3250461	0,002	3258471	0,246	0,006	3256208	0,177
200	1975734	0,003	1977268	0,078	0,011	1978038	0,117

Comparação Numérica

Tabela: K-median

Instância	Ótimo	JMS			TABU		
		Tempo	< Custo >	Erro %	Tempo	< Custo >	Erro %
500	798577	0,034	808090	1,191	0,071	799045	0,059
1000	1434154	0,168	1448286	0,985	0,324	1436322,6	0,151
1500	2000801	0,411	2029316	1,425	0,714	2007909	0,355
2000	2558118	0,775	2587422	1,146	1,341	2574153	0,627
3000	3570766	2,165	3620604	1,396	3,299	3583415	0,354

Conclusão

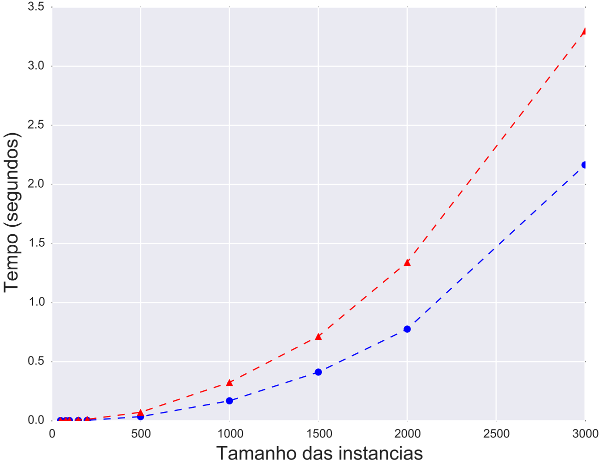


Figura: Gráfico de comparação de tempo de execução

Conclusão

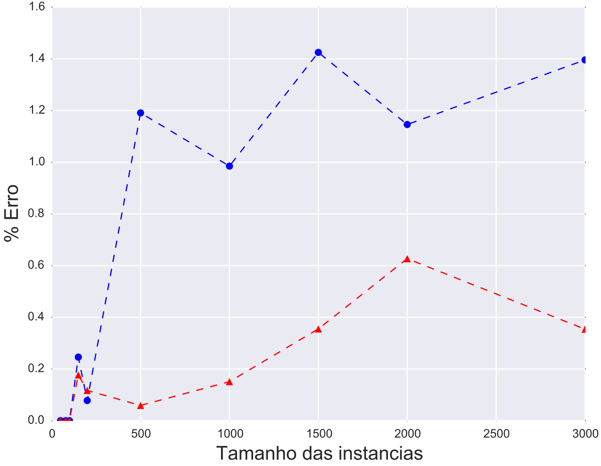


Figura: Gráfico de comparação de erro em relação ao ótimo

Conclusão

Em vista dos resultados obtidos, chegamos na conlusão que em questão de **qualidade de solução** o $TABU$ possui melhores resultados. Porém em questão de **tempo** é preferível o JMS.

Obrigada!