

## Cap. 2 Problemas de Otimização Combinatória VRP – aula 8

SIMO/MQDEE

MARIA CÂNDIDA MOURÃO

(cmourao@iseg.ulisboa.pt)

## CVRP – Modelos - C0

Problema dos modelos anteriores:

- Pode ser difícil identificar o serviço de cada veículo!
- Modelos com **variáveis de 3 índices!**



Bruce L. Golden  
Univ. Maryland



Tom L. Magnanti  
MIT

B.L. Golden, T.L. Magnanti, H.Q. Nguyen, 1977

*Implementing VR algorithms*

Networks, Vol. 7, pp. 113-148

## CVRP – Modelos - CO



### Modelos de 3 índices

Golden, Magnanti, Nguyen, 1977

#### ➤ Variáveis:

- $x_{ij}^k = \begin{cases} 1 & \text{se o veículo } k \text{ viaja de } i \in V \text{ para } j \in V \\ 0 & \text{c. c.} \end{cases}$
- $y_i^k = \begin{cases} 1 & \text{se o veículo } k \text{ visita } i \in V \\ 0 & \text{c. c.} \end{cases}$
- $u_i^k$  - (minorante da) procura servida pelo veículo  $k$  até chegar a  $i \in N$

$|K|[n(n+3)+1]$  variáveis

## CVRP – Modelos - CO



(VRP3)  $\min \sum_{k=1}^{|K|} \sum_{(i,j) \in A} c_{ij} x_{ij}^k$  Minimização do custo total

3 índices com restrições MTZ

$$\text{s. a: } \left\{ \begin{array}{ll} \sum_{j \in \delta^+(i)} x_{ij}^k - \sum_{j \in \delta^-(i)} x_{ji}^k = 0 & \forall i \in N; \forall k \in K \quad \text{cada } k \text{ entra e sai de } i = n^o \text{ de vezes} \\ \sum_{j \in \delta^+(0)} x_{0j}^k = 1 & \forall k \in K \quad \text{cada } k \text{ sai 1 vez do depósito} \\ \sum_{j \in \delta^+(i)} x_{ij}^k = y_i^k & \forall i \in N; \forall k \in K \quad \text{cada } k, \text{ se sai de } i \text{ visita } i \\ \sum_{j \in \delta^-(0)} x_{j0}^k = y_0^k & \forall k \in K \quad \text{cada } k \text{ regressa ao depósito} \\ \sum_{k \in K} \sum_{j \in \delta^+(i)} x_{ij}^k = 1 & \forall i \in N \quad \text{garantir os serviços} \\ u_i^k - u_j^k + Qx_{ij}^k \leq Q - q_j & \forall (i,j) \in A; \forall k \in K \quad \text{eliminação de subcircuitos ilegais} \\ q_i \leq u_i^k \leq Q & \forall i \in N; \forall k \in K \quad \text{capacidade} \\ x_{ij}^k \in \{0,1\} & \forall (i,j) \in A; \forall k \in K \\ y_i^k \in \{0,1\} & \forall i \in N; \forall k \in K \end{array} \right.$$

$|K|[n(n+5)+2] + n$  restrições

## VRP – Modelos – CO – exemplo 4



Considere de novo o exemplo 4:

Formule o problema utilizando as restrições *MTZ* para eliminação dos subcircuitos ilegais e a formulação de 3 índices, recorrendo a veículos com capacidade igual a 150.

## CVRP – Heurísticas Construtivas



- Greedy
- Cluster 1<sup>st</sup> – Route 2<sup>nd</sup>
  - Identificar uma partição do conjunto de clientes  $N$  em  $|K|$  subconjuntos
  - Identificar uma rota admissível que sirva cada um dos subconjuntos de clientes - TSP
- Route 1<sup>st</sup> – Cluster 2<sup>nd</sup>
  - Identificar uma rota que inclua todos os clientes sem considerar as restrições de capacidade
  - Dividir a rota identificada em  $|K|$  subrotas, uma para cada veículo

## VRP – Savings Algorithm – Clarke & Wright



Clarke, G. & Wright, J.W., 1964  
*Scheduling of Vehicles from a Central  
 Depot to a Number of Delivery Points*  
 Op. Res., Vol. 12, pp. 568-581

### ➤ H. de Savings - Greedy:

Passo 0: Inicialização:

Considerar que cada cliente é servido isolado, ou seja, uma rota por cada cliente

Passo 1: Calcular o *savings* de juntar cada par de clientes por:  $s_{ij} = c_{i0} + c_{0j} - c_{ij}$

Passo 2: Ordenar os *savings* por ordem decrescente numa lista  $L$

Passo 3: Seleccionar as rotas associada à junção de maior *savings* (1ª da lista  $L$ )

Se a junção das rotas identificadas for possível ( $Q$ ),

- juntar as rotas respetivas
- Atualizar  $L$ ; recalculando os *savings* necessários da junção de rotas
- Voltar a 2

**c.c.**, retirar a junção de  $L$

Passo 4: Se ainda há *savings* positivos em  $L$ , voltar a 2

**c.c.** FIM

## VRP – exemplo 5



Considere de novo o exemplo 5.

- a) Utilizando a heurística de *savings* identifique uma SA.
- b) Compare os valores dos problemas resolvidos anteriormente com o de a)

## VRP – Extensões



- ✓ Time Windows – Clientes e Depósito
- ✓ Frota Heterogénea
- ✓ Periódico
- ✓ Pickup & Delivery – restrições de carregamento
- ✓ Com Backhauls – clientes divididos em 2 conjuntos (rotas: 1º procura; 2º oferta)
- ✓ Dial-a-Ride – transporte personalizado de pessoas (táxi partilhado; crianças de escolas para atividades; idosos entre pontos diferenciados)
- ✓ Problemas Estocásticos
- ✓ Green VRP
- ✓ Problemas Dinâmicos: procuras dinâmicas; tempos/procuras dependentes do tempo

## VRP – SOLVER



- Excel Add-in
- [VRP\\_Spreadsheet\\_Solver\\_v3.01](#) – Exemplo Lisboa!

G Erdoğan, 2017  
*An open source Spreadsheet Solver for VRPs*  
 Comp. & Operations Research, Vol. 84, pp. 62-72

## VRP – Trabalho

Proponha um enunciado para um problema de VRP, recorrendo a uma instância com pelo menos 7 clientes e 2 veículos.

- a) Utilize um algoritmo para obter uma SA.
- b) Proponha um modelo para o problema, recorrendo a variáveis com dois índices, e uma sua relaxação. Com base na relaxação proposta determine um minorante para o problema.
- c) Introduza um corte na relaxação proposta para tentar melhorar o valor do minorante.
- d) Utilizando o modelo de três índices e as restrições *MTZ*, para impedir subcircuitos ilegais, resolva o problema.
- e) Utilize o *VRP Solver* para obter uma solução e compare-a com as geradas nas alíneas anteriores.