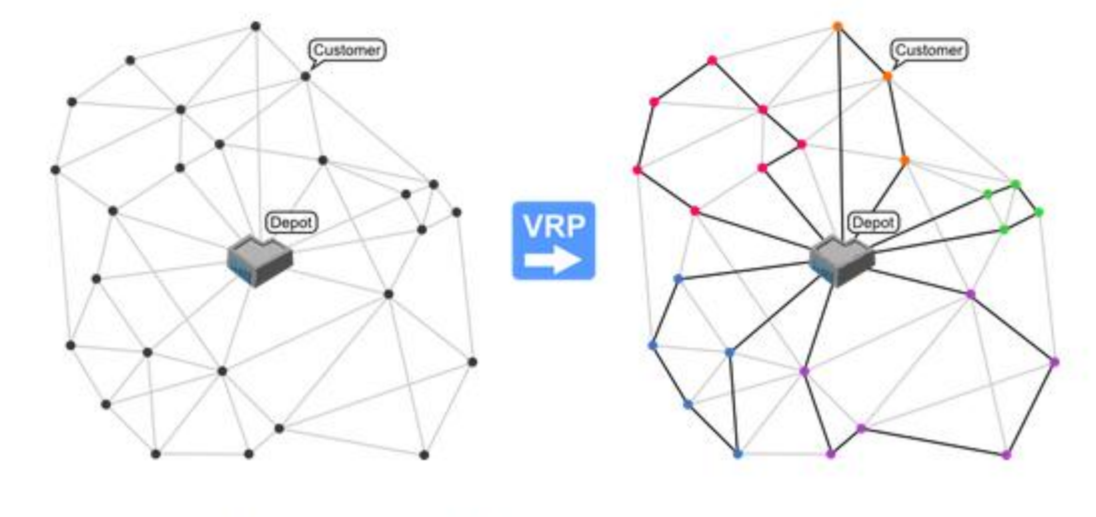
3.5.1 Roteamento de veículos 195

Problema de Roteamento de Veículos

The Vehicle Routing Problem-VRP

O VRP é um problema que consiste em definir rotas para um conjunto de veículos estacionados em um depósito central que irão servir um conjunto de clientes, minimizando os custos de transporte. Este problema corresponde a problemas de otimização no intuito de cobrir os nós de um grafo, contendo um nó representando o depósito, a mínimo custo. Os problemas reais são caracterizados por diversas restrições que limitam a tipologia dos ciclos e tornam o VRP um dos mais difíceis entre os problemas de otimização combinatorial. O VRP é tido com *NP-hard* e os algoritmos exatos propostos na literatura são capazes de resolver apenas problemas de menores dimensões e aparentemente não levam em consideração a complexidade de problemas reais.



An instance of a VRP (left) and its solution (right)

http://neo.lcc.uma.es/vrp/vehicle-routing-problem/

O problema de roteamento de veículos, então, é o problema de construir um conjunto de rotas de um depósito para pontos de demanda, de modo que a soma dos comprimentos seja minimizada. Para solucionar o VRP as heurísticas são classificadas em[1]: i) construtivas; ii) de duas fases; e iii) de melhoria de rotas

i) Heurísticas construtivas: Os métodos de construção de rotas estão entre as primeiras heurísticas para o VRP e ainda formam o núcleo de muitas implementações de software para vários aplicativos de roteamento. Esses algoritmos geralmente partem "do nada" (*empty solution*) e constroem iterativamente rotas inserindo um ou mais clientes em cada iteração, até que todos os clientes sejam servidos por um veículo. Algoritmos de construção são subdivididos em sequenciais e paralelos,

[1] http://dis.unal.edu.co/~gjhernandezp/TOS/ROUTING/VRP1.pdf

dependendo do número de rotas elegíveis para a inserção de um cliente. Os métodos sequenciais expandem apenas uma rota de cada vez, enquanto os métodos paralelos consideram mais de uma rota simultaneamente. Os algoritmos de construção de rotas são totalmente especificados por suas três principais etapas: um critério de inicialização; um critério de seleção especificando quais clientes são escolhidos para inserção na iteração atual; e um critério de inserção para decidir onde inserir os clientes escolhidos nas rotas atuais.

Podemos citar os algoritmos clássicos para o VRP (link).

- *Savings* de Clark *and* Wright (1964)
- Christofides (1979)
- Mole *and* Jameson (1976)
- *Matching Based*
- *Multi-route Improvement Heuristics*
    - Thompson *and* Psaraftis
    - Van Breedam
    - Kinderwater *and* Savelsbergh

ii) Heurísticas de duas fases: Os métodos de duas fases são baseados na decomposição do processo da solução de VRP nos dois subproblemas separados.

(1) Agrupamento – determina uma partição dos clientes em subconjuntos, cada um correspondendo a uma rota; e

(2) Roteamento – determina a sequência de clientes em cada rota.

Podemos destacar as seguintes heurísticas[2]

- Cluster-First-Route-Second Algorithms (link)
    - Fisher and Jaikumar (link)
    - The Petal Algorithm
    - The Sweep Algorithm
    - Taillard
- Route-First-Cluster-Second Algorithms

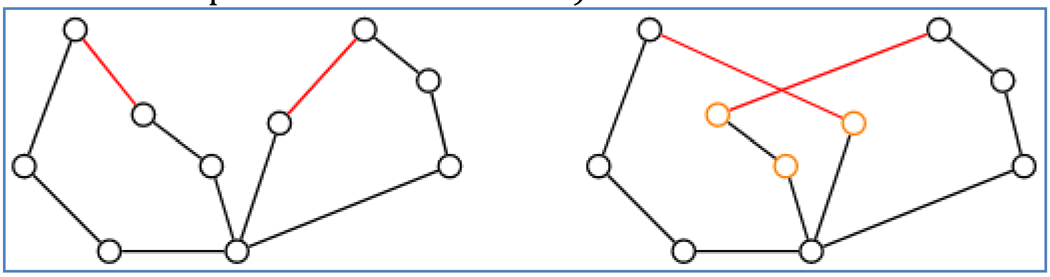
Em um método *cluster-first-route-second*, os clientes são agrupados primeiro em clusters e as rotas são então determinadas sequenciando adequadamente os clientes em cada cluster. Diferentes técnicas foram propostas para a fase de agrupamento, enquanto a fase de roteamento equivale a resolver um TSP.

[2] http://neo.lcc.uma.es/vrp/solution-methods/

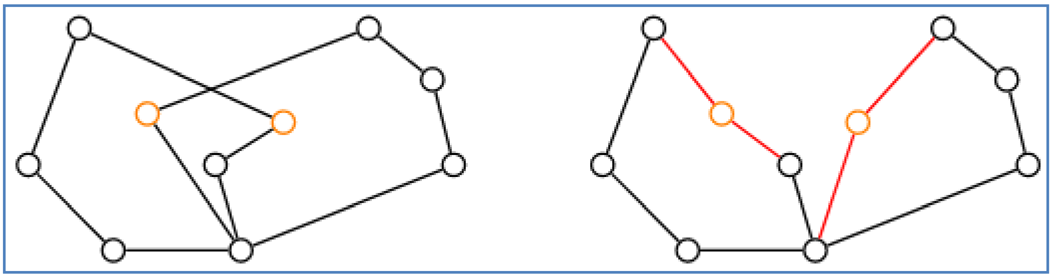
iii) Heurísticas de melhorias de rotas: Os algoritmos de busca local são frequentemente usados para melhorar as soluções iniciais geradas por outras heurísticas. A partir de uma determinada solução, um método de busca local aplica modificações simples, como trocas de arco ou movimentos de clientes, para obter soluções vizinhas de custo possivelmente melhor. Se uma solução melhor for encontrada, ela se tornará a solução atual e o processo será iterado; caso contrário, um mínimo local foi identificado.

Uma grande variedade de heurísticas de melhoria estão disponíveis. Estas podem ser subdivididas: i) intra-rotas-*single-route*, elas operarem em uma única rota de cada vez; ou ii) inter-rotas-*multi-route*, considerarem mais de uma rota simultaneamente.[3]

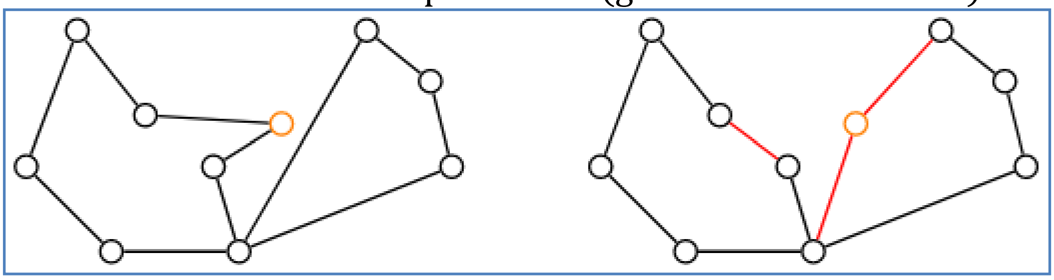
i) O tipo de heurística intra-rota mais comum é a heurística k-opt para o TSP, onde k arestas são removidas da solução atual e substituídos por outras k.
ii) As inter-rotas podem ser divididas em dois tipos (conforme Van Breedam).
   a. *String cross*: duas rotas são mudadas pelo cruzamento de duas arestas de duas rotas diferentes. (2-opt: duas arestas de rotas diferentes são substituídas por duas novas arestas)



   b. *String exchange* (*swap*): porções de duas rotas compostas por no máximo k nós são trocadas entre duas rotas.



   c. *String relocation* (*move*): Uma parte da rota composta por no máximo k nós é movida de uma rota para outra (geralmente k = 1 ou 2).



[3] Laporte and Semet (2002) Classical Heuristics for the CVRP. Chapter 5 of Paolo Toth, and Daniele Vigo (eds) The Vehicle Routing Problem, SIAM

Algoritmo Clarke e Wright para VRP

A primeira e mais famosa heurística para o VRP foi proposta por Clarke e Wright (1964) e é baseada no conceito de economia, uma estimativa da redução de custo obtida ao servir dois clientes sequencialmente na mesma rota, em vez de duas rotas separadas. Se i é o último cliente de uma rota e j é o primeiro cliente de outra rota, a economia associada é definido como $s_{ij} = c_{i0} + c_{0j} - c_{ij}$. Se s_{ij} for positivo, servir i e j consecutivamente em uma rota é lucrativo. O algoritmo de Clarke e Wright considera todos os pares de clientes e classifica as economias em ordem não crescente. Começando com uma solução na qual cada cliente aparece separadamente em uma rota, a lista de pares de clientes é examinada e duas rotas são mescladas sempre que isso for viável. Geralmente, uma mesclagem de rota é aceita somente se a economia associada for não-negativa, mas, se o número de veículos for minimizado, mesclagens de economias negativas também podem ser consideradas. O algoritmo de Clarke e Wright é inerentemente paralelo, já que mais de uma rota está ativa a qualquer momento. No entanto, pode ser facilmente implementado de forma sequencial.

Quando duas rotas $0 - \cdots - i - 0$ e $0 - j - \cdots - 0$ podem ser facilmente mescladas em uma única rota $0 - i - j - \cdots - 0$, uma ecnomia de distância $s_{ij} = c_{i0} + c_{0j} - c_{ij}$ é gerada. O algoritmo funciona da seguinte maneira (as 3 primeiras etapa são iguais nas versões paralela e sequencial):

Algoritmo de Clarke Wright – Economias/*Savings* – para o VRP

Rotule os clientes como nós $1, \ldots, n$ e denomine o armazém de nó 0.

Determine os custos c_{ij} para viajar entre todos os pares de cidades e o armazém $i = 0,1, \ldots, n,\ j = 0,1, \ldots, n$.

1. Selecione o armazém como o nó central.
2. Calcule as economias $s_{ij} = c_{i0} + c_{0j} - c_{ij}$ para $i = 1, \ldots, n,\ j = 1, \ldots, n, i \neq j$. Crie n rotas de veículos 0–i–0 para $i = 1, \ldots, n$.
3. Ordene as economias, s_{ij}, da maior à menor.
4. Tome a aresta (i, j) a partir do topo da lista de economia. Então, executa-se procedimento específico para verificar se os nós i e j já estão em alguma outra rota e por isso possuem restrições para compartilharem da mesma rota. Após este procedimento específico ser executado é verificado a possibilidade de conectar estes dois nós, ainda é necessário verificar se as restrições de tempo e de capacidade estão atendidas. Só então os dois nós poderão passar a fazer parte da nova rota.

O procedimento de verificação se os pontos i e j podem ser conectados é:

a. Nem i nem j já foram atribuídos a uma rota; nesse caso, uma nova rota é iniciada incluindo i e j.

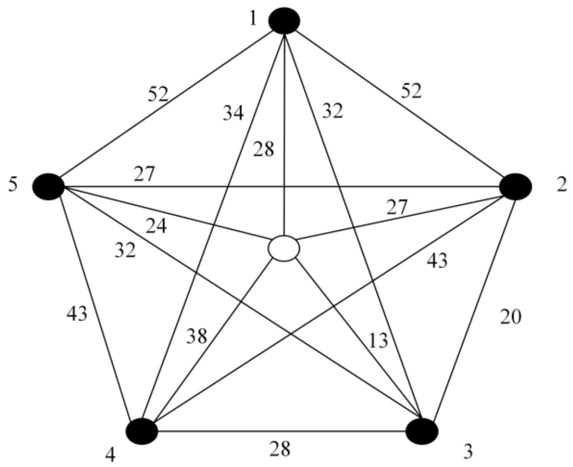
b. Ou, exatamente um dos dois pontos (i ou j) já foi incluído em uma rota existente e esse ponto não é interior a essa rota (um ponto é interior a uma rota se não estiver adjacente ao depósito); nesse caso, o link (i, j) é adicionado à mesma rota.

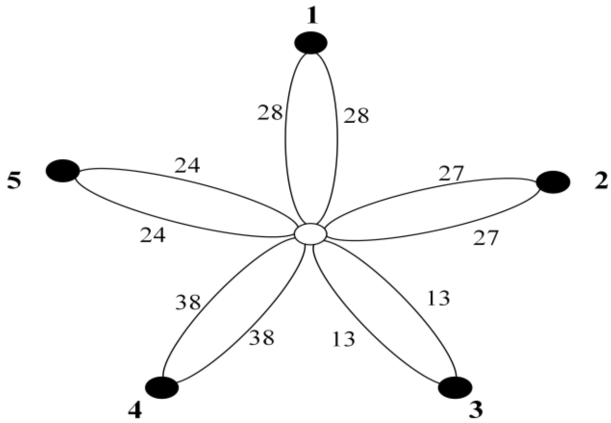
c. Ou, i e j já foram incluídos em duas rotas existentes diferentes e nenhum ponto é interior à sua rota; nesse caso, as duas rotas são mescladas.

Ao término da lista e dos procedimentos, se houver nós P_k desconectados do armazém, realizar a conexão diretamente ao armazém, criando roteiros individuais $0–P_k–0$.

Exemplo 1 www.inf.ufpr.br/aurora/disciplinas/topicosia2/Ico-construtivas.ppt

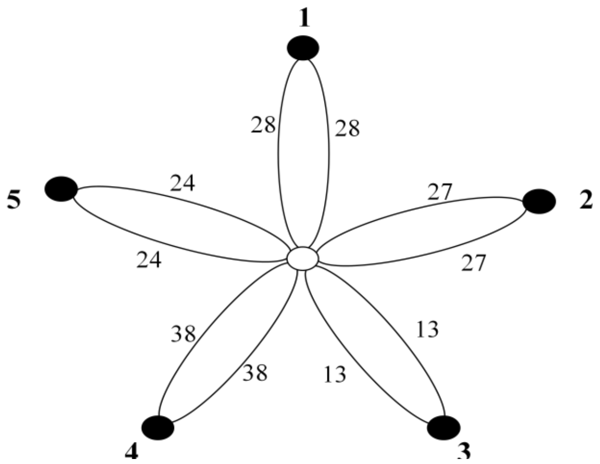
Cidades	1	2	3	4	5	Capacidade
Demanda	15	17	27	12	23	50

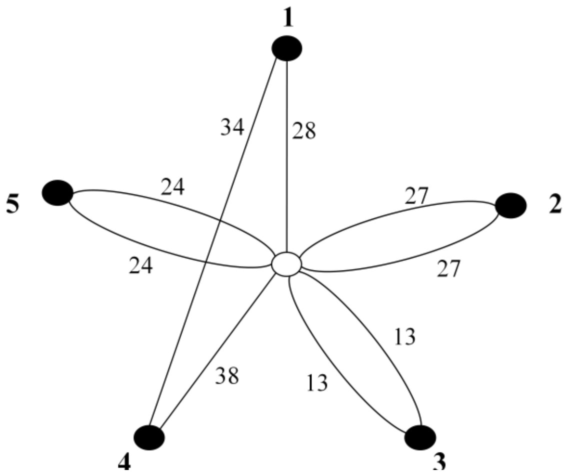




Total percorrido	260
Nº de veículos	5

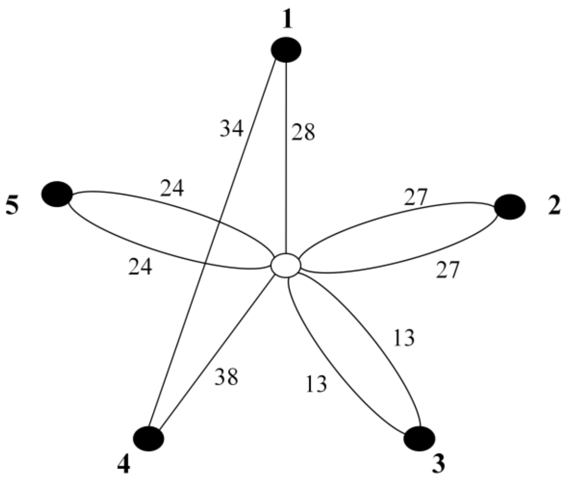
i	j	d_{i0}	d_{j0}	d_{ij}	s_{ij}	Dem
1	2	28	27	52	3	32
1	3	28	13	32	9	42
1	4	28	38	34	32	27
1	5	28	24	52	0	38
2	3	27	13	20	20	44
2	4	27	38	43	22	29
2	5	27	24	27	24	40
3	4	13	38	28	23	39
3	5	13	24	32	5	50
4	5	38	24	43	19	35

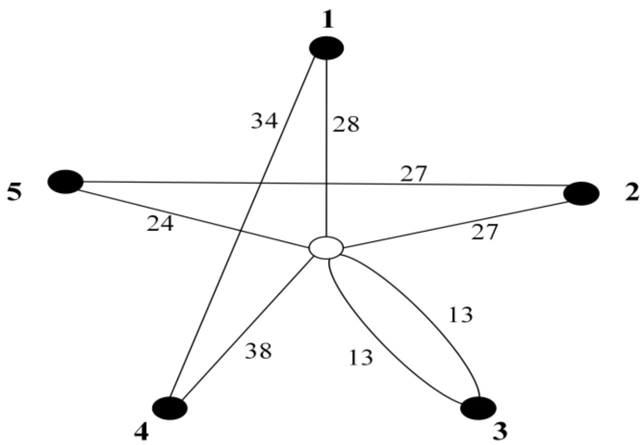




Total percorrido	228
Nº de veículos	4

i	j	s_{ij}	Dem
1	2	3	44
1	3	9	54
1	5	0	50
2	3	20	44
2	4	22	44
2	5	24	40
3	4	23	54
3	5	5	50
4	5	19	50



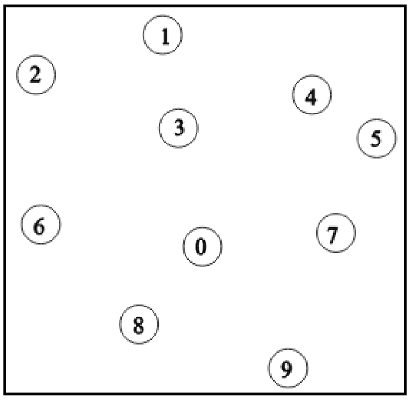


i	j	s_{ij}	Dem
1	2	3	67
1	3	9	54
1	5	0	67
2	3	20	67
2	4	22	67
3	4	23	54
3	5	5	67
4	5	19	67

Total percorrido	204
Nº de veículos	3

Exemplo 2

Dado um garfo completo com um depósito central 0 e 9 clientes. (Os custos não são diretamente proporcionais às distâncias euclidianas do diagrama).



Custos simétricos

c_{ij}	0	1	2	3	4	5	6	7	8	9
0	-	12	11	7	10	10	9	8	6	12
1		-	8	5	9	12	14	16	17	22
2			-	9	15	17	8	18	14	22
3				-	7	9	11	12	12	17
4					-	3	17	7	15	18
5						-	18	6	15	15
6							-	16	8	16
7								-	11	11
8									-	10

Demandas

i	1	2	3	4	5	6	7	8	9
d_i	10	15	18	17	3	5	9	4	6

Capacidade de um veículo: K = 40. Solução com Clarke & Wright:

Savings/economias: $s_{ij} = c_{i0} + c_{0j} - c_{ij}$

Simetria: $s_{ij} = s_{ji}$

s_{ij}	1	2	3	4	5	6	7	8	9
1		15	14	13	10	7	4	1	2
2			9	6	4	12	1	3	1
3				10	8	5	3	1	2
4					17	2	11	1	4
5						1	12	1	7
6							1	7	5
7								3	9
8									8

Ordenando as economias: (4,5), (1,2), (1,3), (1,4), (2,6), (5,7), (4,7), (1,5), (3,4), (2,3), (7,9), (3,5), (8,9), (1,6), (5,9), (6,8), (2,4), ...

Solução inicial: ciclos 0–1–0, 0–2–0, ... , 0–9–0.

Aresta (4,5): ligar ciclos 0-4-0 e 0-5-0: resultado 0-4-5-0, carga $d_4 + d_5 = 20<K$.

Aresta (1,2): ligar 0-1-0 e 0-2-0: resultado 0-1-2-0, carregue $d_1 + d_2 = 25<K$.

Aresta (1,3): limite de capacidade: $d_1 + d_2 + d_3 = 43> K$.

Aresta (1,4): limite de capacidade: $d_1 + d_2 + d_4 = 42>K$.

Aresta (2,6): ligar ciclos 0-1-2-0 e 0-6-0: result. 0-1-2-6-0, carga $d_1 + d_2 + d_6 = 30<K$.

Aresta (5,7): ligar ciclos 0-4-5-0 e 0-7-0: result. 0-4-5-7-0, carga $d_4 + d_5 + d_7 = 29<K$.

Aresta (4,7): A condição 4 (i) não é válida: os nós pertencem ao mesmo ciclo.

Aresta (1,5): A condição 4 (iii) não é válida: o nó 5 é um nó interior da sua rota.

Aresta (3,4): limite de capacidade: $d_3 + d_4 + d_5 + d_7 = 47>K$.

Aresta (2,3): A condição 4 (iii) não é válida: o nó 2 é um nó interior.

Aresta (7,9): ligar ciclos 0-4-5-7-0 e 0-9-0: resultado 0-4-5-7-9-0, carga $d_4 + d_5 + d_7 + d_9 = 35<K$

Aresta (3,5): A condição 4 (iii) não é válida: o nó 5 é um nó interior.

Aresta (8,9): ligar ciclos 0-4-5-7-9-0 e 0-8-0: resultado 0-4-5-7-9-8-0, carga $d_4+d_5+d_7+d_9+d_8=39<K$.

Aresta (1,6): A condição 4 (i) não é válida: os nós pertencem ao mesmo ciclo.

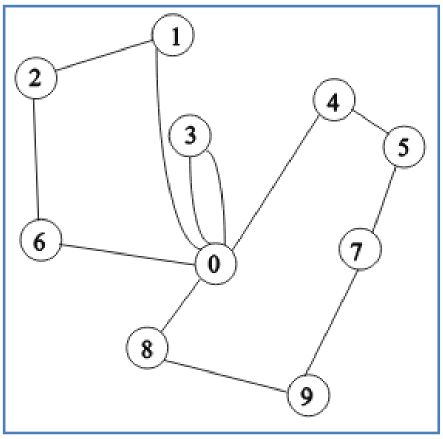
Aresta (5,9): A condição 4 (i) não é válida: os nós pertencem ao mesmo ciclo.

Aresta (6,8): limite de capacidade: $(d_1 + d_2 + d_6) + (d_4 + d_5 + d_7 + d_9 + d_8) = 69>K$.

Resultado: três rotas que não podem ser unidas devido ao limite de capacidade.

Solução:

Rota	Carga	Custo
0-3-0	18	14
0-1-2-6-0	30	37
0-4-5-7-9-8-0	39	46



Custo total: 97

Número de veículos: 3

Sweep Heuristic

Essa heurística é do tipo "*clustering first, route later*".

Suponha que os clientes sejam pontos alinhados com distâncias euclidianas. A distância d_{ij} entre (x_i, y_i) e (x_j, y_j) é $d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$.

1. Calcule as coordenadas polares de cada cliente em relação ao depósito. Classifique os clientes aumentando o ângulo polar.
2. Adicione cargas ao primeiro veículo a partir do topo da lista, contanto que a capacidade permita. Continue com o próximo veículo até que todos os clientes sejam incluídos. Agora os clientes foram agrupados por veículos.
3. Para cada veículo, otimize sua rota por um método TSP adequado (exatamente ou por aproximação).

Graficamente, na etapa 1, rotaciona-se um raio centrado no depósito. O ângulo inicial pode ser escolhido arbitrariamente.

Fonte: http://www.bernabe.dorronsoro.es/vrp/

Comentário: Também para o TSP, as técnicas de agrupamento podem ser usadas para dividir problemas do caixeiro viajante de grande escala em partes menores.

Melhoria da Rota via 2-opt e 3-opt

Pode-se utilizar métodos heurísticos de melhoria das rotas

Considerações

Normalmente, existem restrições de capacidade nas rotas, ou o número de rotas é prescrito.

Objetivo

- Em estudos acadêmicos, geralmente uma combinação
  - Primeiro, minimizar o número de rotas
  - Em seguida, minimizar a distância total ou o tempo total

- No mundo real
  - Uma combinação de tempo e distância
  - Deve incluir custos dependentes do veículo e da equipe
  - Normalmente, os números dos veículos são fixos

Complicações

- Vários tipos de veículos
- Várias capacidades do veículo (peso, pés cúbicos, espaço no chão, valor)
- Muitos Custos:
  - Custo fixo.
  - Custos variáveis por quilometro carregado e por quilometro vazio.
  - Tempo de espera; Tempo de carregamento.
  - Custo por parada (manuseio).
  - Custos de carregamento e descarga.
- Prioridades para clientes ou pedidos.
- Janelas de tempo para carregamento e entrega. (difícil versus fácil)
- Compatibilidade Veículos e clientes.
  - Veículos e pedidos.
  - Tipos de pedidos.
  - Motoristas e veículos.
- Regras de direção
  - Duração máxima na direção: 10h antes das 8 horas. pausa.
  - Duração máxima do trabalho: 15h antes das 8 horas de intervalo.
  - Duração máxima da viagem: 144h

Many versions of the VRP have been considered in the literature
http://transp-or.epfl.ch/courses/decisionAid2016/labs/Lab_4/Presentation/Lab11-Presentation.pdf

- *Capacitated VRP*
- *VRP with time windows*
- *Pickup and delivery VRP*
- *VRP with backhauls*
- *VRP with split deliveries*
- *Periodic VRP*
- *Heterogeneous fleet VRP*
- *Dial-a-ride problem (DARP)*
- *Stochastic VRP*
- *Dynamic VRP*
- *Inventory routing problem (IRP)*
- *etc...*

The interested student is referred to Toth and Vigo (2002) or Toth and Vigo (2014). The full text of the former can be accessed online from the EPFL library website if you are on campus or connected through VPN.

- Dynamic Vehicle Routing Problems (DVRP): a part or all of the customers are revealed **dynamically** during the design or execution of the **routes**
- Capacitated VRP (CVRP) - Every vehicle has a limited capacitate
- VRP with Time Windows (VRPTW) - Every customer has to be supplied within a certain time window
- Vehicle routing and scheduling problem with soft time windows (VRPSTW)
- Vehicle Routing Problem with Backhauls and Time Windows (VRPBTW)
- Time Dependent Vehicle Routing Problem with Time Windows (TDVRPTW)
- Open Vehicle Routing Problem (OVRP): after delivering service to the last customer, the **vehicle** does not necessarily return to the initial depot
- Vehicle Routing Problem with Multiple Trips (VRPMT)
- Vehicle Routing Problem with Pickup and Delivery (VRPPD) - Customers may return some goods to the depot
- Periodic VRP - PVRP - The deliveries may be done in some days

- Stochastic VRP - SVRP - Some values (like number of customers, theirs demands, serve time or travel time) are random
- Multiple Depot VRP - MDVRP - The vendor uses many depots to supply the customers
- Split Delivery VRP - SDVRP - The customers may be served by different vehicles