Pathfinding and Routing
NAIL137

Vehicle routing problem

Jiří Švancara

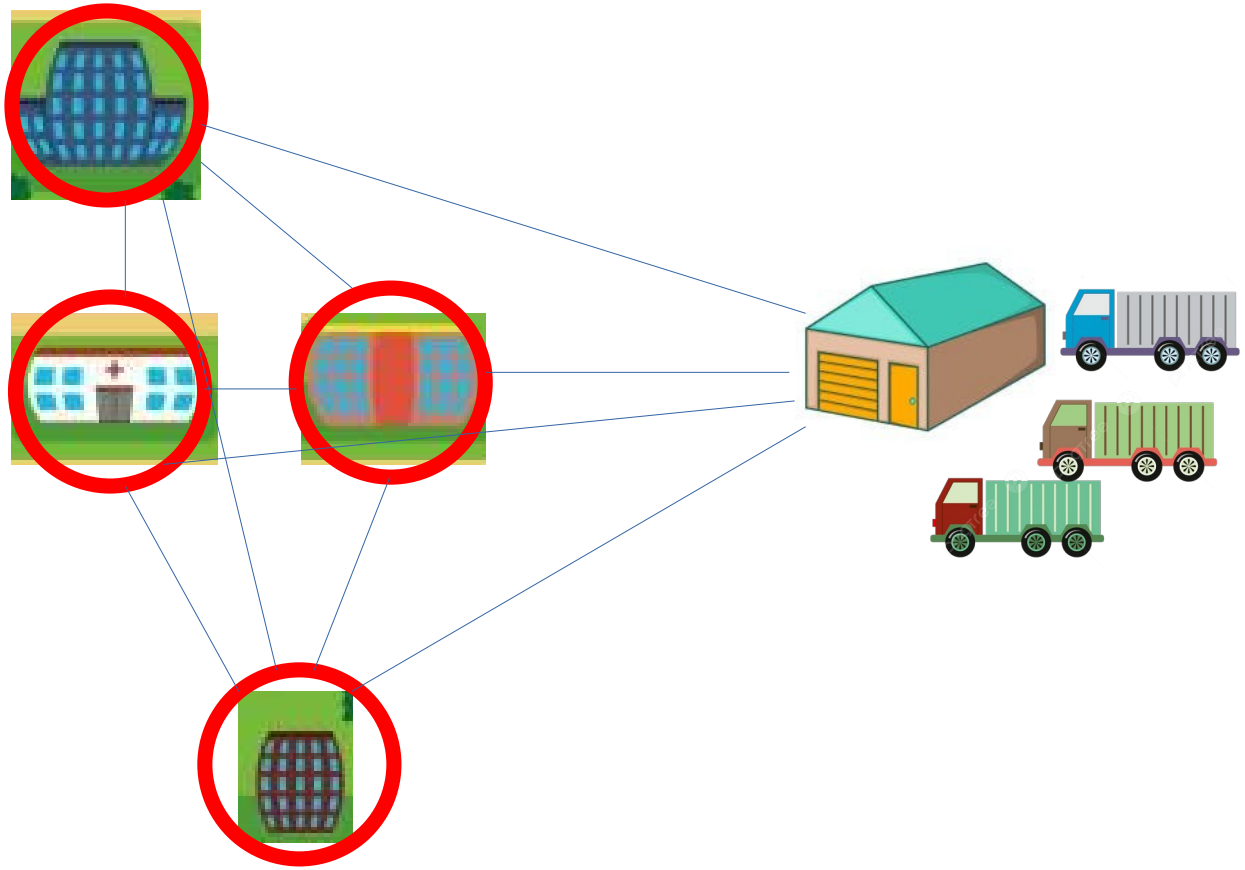
FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

Overview

- Definitions
- Variations
- Optimal solvers
- Suboptimal solvers
- Use cases

Motivation

Motivation - simplification



Definition

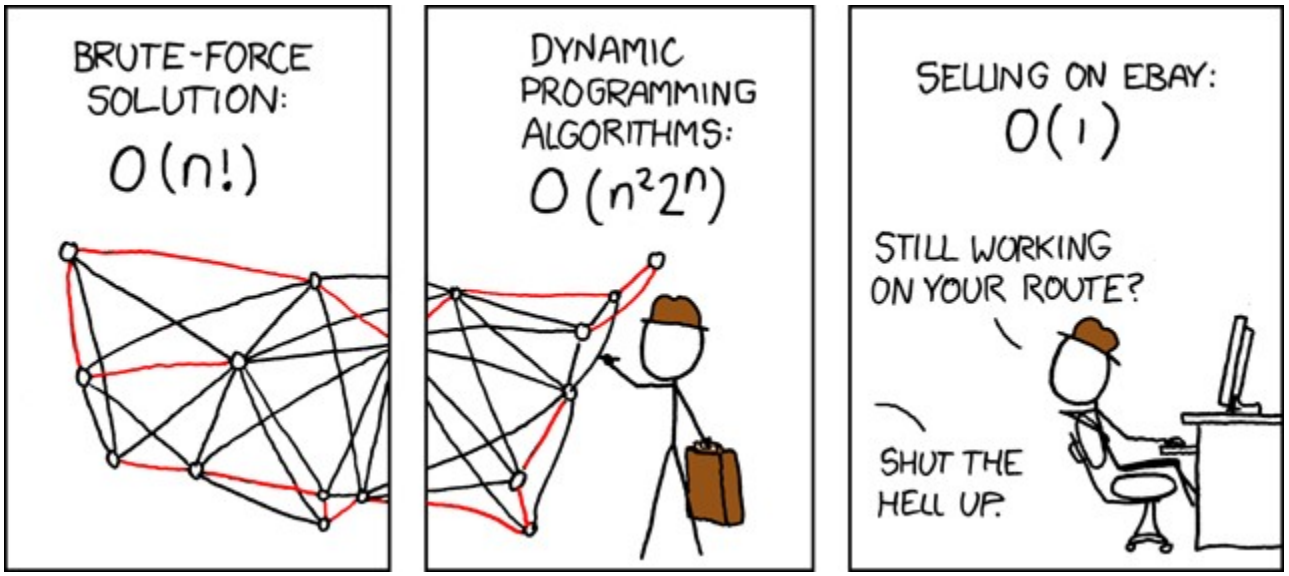
- A complete weighted graph $G=(V,E)$
- v_0 is a depot, $v_1, \ldots v_n$ are customers
- A set of k identical vehicles

Find a set of paths, one for each vehicle, such that each customer is visited exactly once.

Introduced in 1959 as „Truck Dispatching Problem“

Special case - single vehicle

Traveling salesman problem



Complexity

- TSP is NP-Hard
- VRP is even harder, first decide partition, each vehicle is solving a separate TSP problem

  (This is not a formal proof)

Variations

- Distance constrained VRP (DCVRP)
  - Each vehicle has a maximum distance it can travel (cost, safety, balance, …)
- Capacity-constrained VRP (CVRP)
  - Each vehicle has a specific capacity, each customer demands specific amount
- VRP with Time Window (VRPTW)
  - Demands must be met in specific time-window

B. Aghaabdollahian, B. Javadi, A. Aazami. An overview of vehicle routing problems: A comprehensive classification and description of types. Transportation Research Interdisciplinary Perspectives, 2026.

Variations cont.

- VRP with Pickup and Delivery (VRPPD)
  - Each delivery needs to be picked first
- Multiple depot VRP (MDVRP)
  - There are more depots
  - Vehicles can start anywhere
- VRP with stochastic (VRPS)
  - Some parts of the problem is probabilistic
    - Travel time
    - Demand
    - Customer presence
    - ...

Optimization functions

- Total distance traveled (=cost)
- Total number of vehicles used
- Number of unfulfilled demands
- Number of broken constraints on time

Exact approaches

- Bruteforce
- Dynamic programming
- Reduction to ILP, CP

Dynamic programming - TSP

- subset $S \subseteq \{1,\ldots,n\}$
- C(S,j) – cost of path starting at depot, ending at j

- $C(\{j\},j)=c_{0j}$
- $C(S,j)=\min_{i \in S \setminus \{j\}}[C(S \setminus \{j\},i)+c_{ij}]$
- **$T(S)=\min_{j \in s}[C(S,j)+c_{j0}]$**

Dynamic programming - VRP

- F(S,q) – cost to serve customers from S via q vehicles

- $F(S,1)=T(S)$
- $F(S,q)=\min_{A \subseteq S}[T(A)+F(S \setminus A,q-1)]$
- **$F(\{1,\ldots,n\},m)$**

Naive ILP model

x_{ijk} - vehicle *k* is using edge *(i,j)*

$$\text{minimise} \sum_{i,j} c_{ij} \sum_{k} x_{ijk}$$

subject to

$$\sum_{i}\sum_{k} x_{ijk} = 1 \quad \forall j$$

One incoming visit

$$\sum_{j}\sum_{k} x_{ijk} = 1 \quad \forall i$$

One outgoing visit

$$\sum_{i} x_{ihk} - \sum_{j} x_{hjk} = 0 \quad \forall k, h$$

Flow conservation for each vehicle

$$\sum_{ijk} x_{ijk} = |S| - 1 \quad \forall S \subseteq P(N), 0 \notin S$$

Subtour elimination

$$x_{ijk} \in \{0, 1\}$$

ILP - column generation

$$\min \quad \sum c_k x_k$$

subject to

$$\sum_k a_{ik} x_k = 1 \quad \forall i$$

$$x_k \in \{0,1\}$$

All customers are covered by all routes exactly once

x_k - route of vehicle k

c_k - cost of route

a - 0/1 matrix of visited customers

1) Start with a set of routes (columns)
2) Create a dual problem to generate new columns (pricing problem)
3) Repeat until no improving column (route) exists

ILP - branch and bound

- The model may be solved as LP
  - This creates non-integer solution!
- Branch and bound to extract integer solution
- Combine with the pricing of column generation → branch and price algorithm

CP

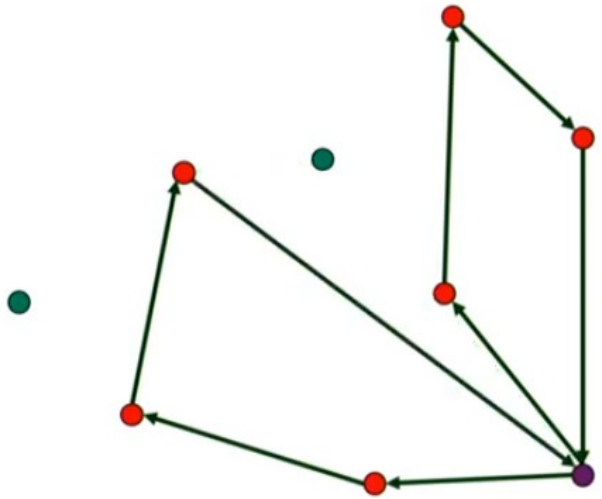
- Useful constraint types
  - AllDifferent
  - Circuit

Suboptimal solvers

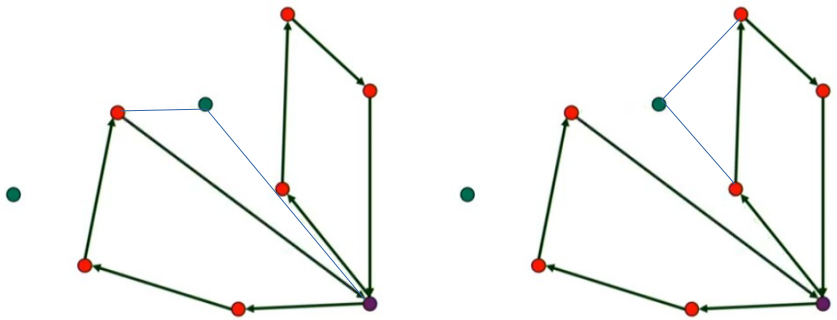
- Construct by insertion
  1) Start with empty solution
  2) Choose a customer to insert
  3) Choose a place to insert it

  - Greedy - choose the cheapest customer to include and the cheapest place to insert
  - Regret - insert the customer that would cause the largest regret if postponed

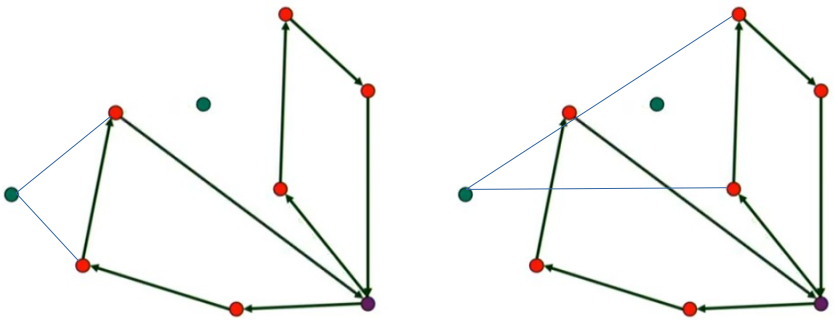
Regret example



Current routes



Second best option is not much worse
→ low regret



Second best option is much worse
→ high regret

Suboptimal solvers cont.

- Improvement methods
- Operators to change solution
  - Move one customer to different position in the vehicle route or different route
  - Swap one customer with another from different vehicle route
  - Swap tails of routes
  - Remove 2 arcs and replace them (reverse order of the customers in between)
  - Select a sequence of customers and try to place them everywhere

Suboptimal solvers cont.

- Improvement methods
  - Local search (such as hill climb, simulated annealing)
  - Genetic algorithms
  - Escape local optimum → take larger neighborhoods

Suboptimal solvers cont.

- Large neighborhood search

  1) Delete part of solution (unassign customers)

  2) Construct the solution back (any construction algorithm, even optimal or a portfolio of solvers)

  3) Keep if better

Suboptimal solvers cont.

- Large neighborhood search improvement
  - Keep worse solution with some prob.
    - Depends on temperature (like SA)
  - Adaptive LNS
    - Change the prob. of selecting an operator based on previous success (in this run)

Use cases

- Logistics
  - Deliveries (ordered packages, recurring deliveries)
  - Trash collection
  - Post delivery
  - Maintenance
- Robotics
  - Task assignment for agents
- Search and rescue

Use cases – winter road maintenance in Plzeňský kraj

Trasy okruhů zimní údržby v okrese DO

Ask Jiří Fink for more details