Vehicle Routing Problem and Random Key Encoding

Shan He

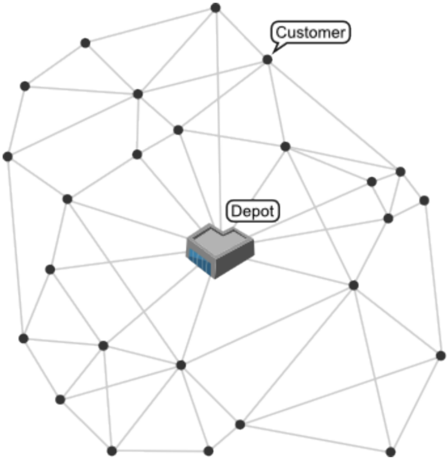
School for Computational Science
University of Birmingham

Advanced Aspects of Nature-Inspired Search and Optimisation

Outline of Topics

VPR: Vehicle routing problem

- Real-world problem: How to design an optimal set of routes for a fleet of vehicles to traverse in order to deliver to a given set of customers?



VPR: Vehicle routing problem

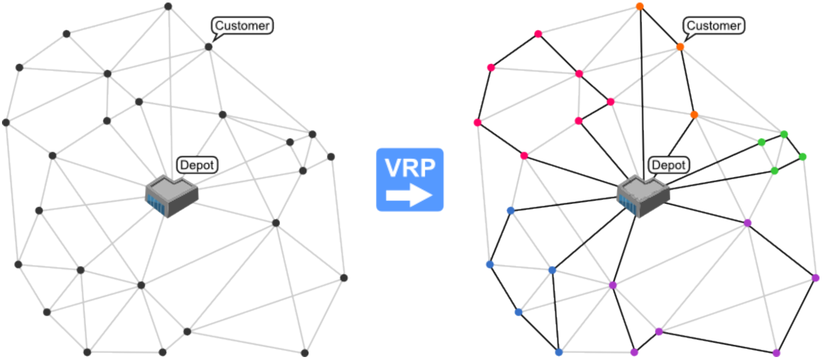
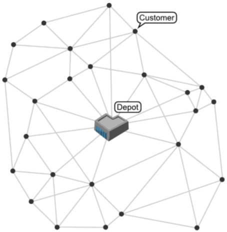


Figure: An instance of a VRP (left) and its solution (right). Figure from VRP webiste with permission.

Vehicle routing problem formulation

- Let's denote
  - $V = \{v_0, v_1, \cdots, v_n\}$ is a vertex set, where v_0 is the depot, and $V' = V \setminus \{v_0\}$ is the set of n customers.
  - $A = \{(v_i, v_j) | v_i, v_j \in V; i \neq j\}$ is an edge (arc) set.
  - C is a matrix of non-negative costs or distances c_{ij} between customers v_i and v_j.
  - $D = \{d_1, d_2, \cdots, d_n\}$ is a vector of the customer demands.
  - R_i is the route for vehicle i, i.e., $R_i = \{v_0, v_1, \cdots, v_{r_i+1}\}$, where $v_i \in V$ and $v_0 = v_{r_i+1} = 0$
  - m is the number of vehicles (all identical). One route is assigned to each vehicle.



Capacitated Vehicle routing problem

- The cost of a given route: $C(R_i) = \sum_{j=0}^{r_i} c_{j,j+1}$.
- **Total cost** of m vehicles (routes): $\sum_{i=1}^{m} C(R_i)$.
- **Constraint**: all vehicles have the same capacity, denoted as s.
- **Objective**: to minimize the total cost, subject to the demand of commodities for each route does not exceed the capacity of the vehicle which serves that route
- More formally:

$$\text{minimise} \sum_{i=1}^{m} C(R_i) \tag{1}$$

$$\text{subject to} \sum_{j=1}^{r_i} d_j \leq s, \qquad i = 1, \cdots, m, \tag{2}$$

Code Example 1: Reading/visualising vehicle routing problem (10 mins)

The CVRP benchmark problem we are going to solve is from VRP website. The original benchmarks were compiled by Augerat et. al.

- Explanation of the file format: open file A-n32-k5.vrp
- Use the code ReadInData.m to read in the file
- Open the optimal solution file opt-A-n32-k5
- Use test.m to plot and visualise the problem and results

Question

Question: What is the relationship between VRP and TSP?

Travelling Salesman and vehicle routing problems

- Answer: TSP is the simplest VRP with the condition that only 1 vehicle or 1 salesman is in operation.
- Vehicle Routing Problem (VRP) was introduced by Dantzig and Ramser in 1959 [1].
- As Dantzig and Ramser noted, the vehicle routing problem "*may be considered as a generalization of the Travelling Salesman Problem*"
- **Question**: can we apply PSO or real-valued coded GA to TSP or VRP?

[1] G. B. Dantzig and J. H. Ramser, The Truck Dispatching Problem, Management Science Vol. 6, No. 1 (Oct., 1959), pp. 80-91

PSO/GAs for Combinatorial Optimisation

- Answer: for TSP and VRP, we cannot directly apply PSO or GAs without some modifications
- TSP/VRP problem: PSO and GAs (even binary GA) will generate infeasible solutions
- Example: 5 city TSP problem with 5 cities numbered as 1-5
  - Using PSO: $[1.42, 2.22, 2.46, 3.91, 4.82]$
    - Rounding real values to integer: $[1, 2, 2, 3, 4]$
    - Infeasible solution: $1 \longrightarrow 2 \longrightarrow 2 \longrightarrow 3 \longrightarrow 4$
  - Using GA with one point crossover:
    - Two feasible solutions: $[1, 4, 3, | \, 2, 5]$ and $[2, 4, 5, | \, 3, 1]$
    - Infeasible solutions: $[1, 4, 3, 3, 1]$ and $[2, 4, 5, 2, 5]$
- Many solutions, e.g. specific encoding (permutation encoding) and mutation/crossover operators
- **But do we have a generic and efficient solution?**

Random key encoding

- Useful for problems that require permutations of the integers, e.g., Travelling salesman problem
- First proposed by Bean in 1994 [2]
- Individuals are encoded as strings of real-valued numbers (called random keys) in the interval $[0, 1]$
  - Example: $[0.69, 0.34, 0.05, 0.53, 0.42]$
- A deterministic decoder to decode the random keys into integers
  - For TSP: a simple sorting algorithm to sort the random keys A in a sorted array B in ascending order.
  - Use the the indices that describe the arrangement of the elements of A into B

Index	1	2	3	4	5
Random Key A	0.69	0.34	0.05	0.53	0.42
Sorted array B	0.05	0.34	0.42	0.53	0.69
Decode as	3	2	5	4	1

[2] J. C. Bean, Genetic Algorithms and Random Keys for Sequencing and optimization, ORSA Journal on Computing, 1994

Random key encoding

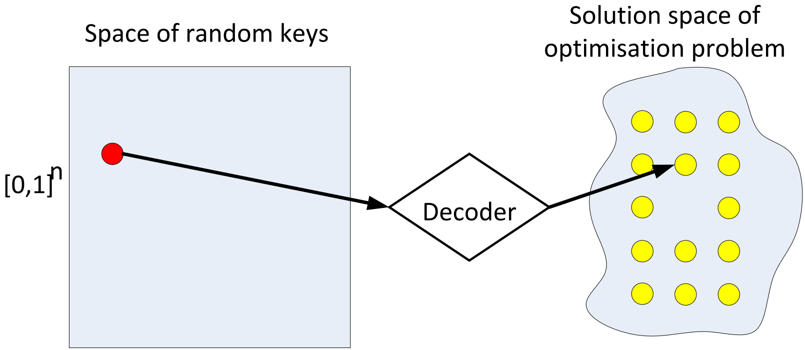


Figure: Random key representation: searches the solution space indirectly by searching the space of random keys and using the decoder to evaluate fitness of the random key.

Exercise 1: Solve TSP problem using PSO with random key encoding (15 mins)

- Open the TSPFitness.m in PSOTSP folder and complete the random key decoding part
  - Hint: execute `[B I] = sort(A)` , you will find `I` essentially describe the arrangement of the elements of A into B
- Solve the USA 48 capital TSP problem using the PSO algorithm with 300 iterations and 50 particles, i.e., execute `PSO_TSP(300, 50)`
- Compare to the global optimal solution which as the minimum distance of 10628

Further readings

- A Random-Key Genetic Algorithm for the Generalized Traveling Salesman Problem
- Biased random-key genetic algorithms for combinatorial optimization

Random key encoding for CVRP

- We only consider CVRP problems where the number of routes are know
- **Question**: how to use a vector of real numbers to represent a set of routes R_i?

Random key encoding for CVRP

- **Question**: how to use a vector of real numbers to represent a set of routes R_i?
- Answer: we use a vector of $n + m - 1$ random key (random real-values):
  - n real-values to represent the permutation of n customers
  - $m - 1$ real-values to represent separators that separate the vector into m routes
  - We then sort the $n + m - 1$ random key values into a sequencing order and use the indices that describe the arrangement as the solution
  - The $m - 1$ largest integer values will be used as route separators to separate the solution into m routes

Random key encoding for CVRP

- Example: Suppose we have a CVRP problem with $n = 5$ customers and $m = 3$ routes
  - We use a vector of $5 + 3 - 1 = 7$ real-values, and using the $m - 1 = 2$ largest integers as separators:

Index	1	2	3	4	5	6	7
Random Key	0.34	0.77	0.45	0.12	0.92	0.28	0.65
Sorted array	0.12	0.28	0.34	0.45	0.65	0.77	0.92
Decoded as	4	6	1	3	7	2	5
Separator	N	Y	N	N	Y	N	N

  - We can then decode the random keys into 3 routes:
    Route 1: 0–4–0
    Route 2: 0–1–3–0
    Route 3: 0–2–5–0

Code example 2: solving CVRP using PSO with random key encoding (20 mins)

- Open the fitness calculation function (CVRPFitness.m) and I will explain
- Run your PSO algorithm. Set the population of particles to 100 and the number of maximum iterations to 500, i.e., execute `PSO_CVRP('A-n32-k5.vrp', 300, 100)`
- You also try my PSO with Passive Congregation (`PSOPC_CVRP('A-n32-k5.vrp', 300, 100)`)