Vehicle Routing Problem -

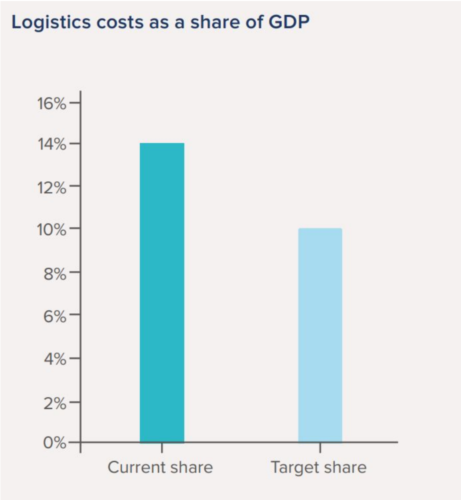
Practical Implementations using OSRM

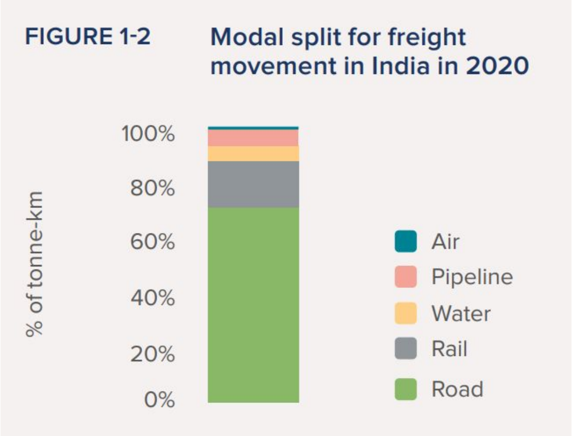
Madhav Mittal CS19B029
Guide : Prof. N.S. Narayanaswamy

UGRC
May 4, 2022

Logistics in India

- Logistics sector - 5% of Indian GDP
- Logistics cost - as high as 14%
- Road transportation - 71 percent of freight
- Truck empty running rate : 40%





Source: Fast Tracking Freight in India: A roadmap for clean and cost effective goods transport. June 2021. NITI Aayog, RMI India, RMI.

The Problem

Vehicle Routing Problem :

Broadly, given a fleet of vehicles and a set of points to be visited, find optimal routes for the vehicles to minimize cost.

UGRC Project Goal

Analysing and implementing variants of VRP for real world problems like goods delivery

- Studying OSRM APIs
- Analyse available VRP sol^{n}
- Integrate and Test on real world data
- Design APIs for modular and easy use of the implementations (pending)

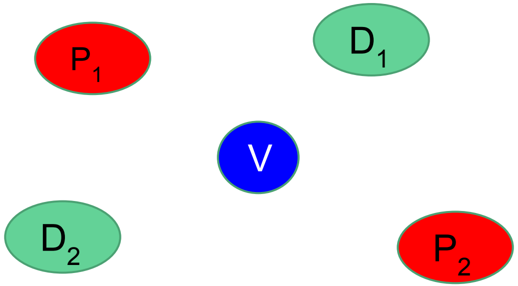
Work undertaken

- Variants of VRP :
  - VRPPD - pickup and delivery
  - MDVRP - multiple depots
  - CVRP - capacitated
  - VRPTW (pending) - time windows
- Knowledge Representation
- Implementation
- Data Generation (later addition)

Variants of VRP

Vehicle Routing Problem with Pickups and Deliveries

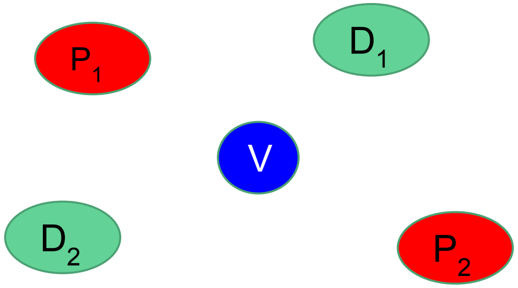
Given a vehicle v located at loc and n consignments c_j with pickup points P_j ($j = 1 \dots n$) and delivery points D_j ($j = 1 \dots n$), route the vehicle to minimize the total distance covered by the vehicle.



VRPPD

Pickup must be done before corresponding delivery.

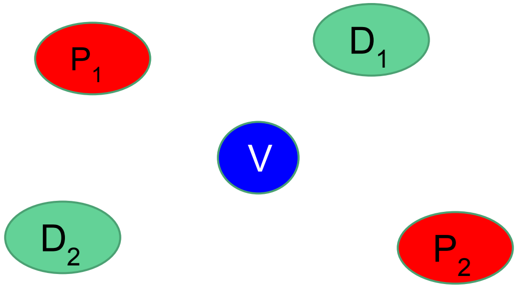
$V \rightarrow P_2 \rightarrow D_1 \rightarrow P_1 \rightarrow D_2$ => INVALID



VRPPD

Method 1 : Use the savings algorithm separately on the pickup points and delivery points. That is, first pickup all the consignments and then deliver all of them.

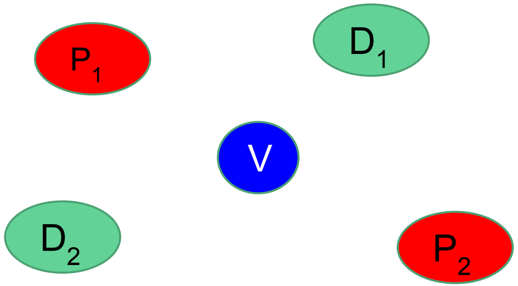
For example : V -> P_1 -> P_2 -> D_1-> D_2



VRPPD

Method 2 : Visit the nearest neighbour from current point in consideration. We start with considering all the pickups as the set of points to choose from. When a pickup point is visited, add its corresponding delivery point to the set of points.

For example : $V \to P_1 \to D_1 \to P_2 \to D_2$



VRPPD

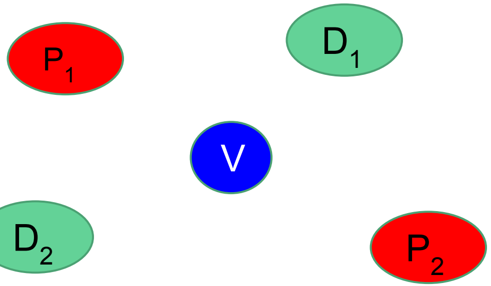
Method 2 : Visit the nearest neighbour from current point in consideration. We start with considering all the pickups as the set of points to choose from. When a pickup point is visited, add its corresponding delivery point to the set of points.

For example : V -> P_1 -> P_2 -> D_2-> D_1 => Not optimal

VRPPD

Method 3 : Use the savings algorithm on the pickup points to generate a route. Then, starting with the last pickup in this route, insert the corresponding delivery point in the route where it causes least detour after its pickup. Similarly, insert all the corresponding deliveries into the route from upto the first pickup.

For example : $V \rightarrow P_1 \rightarrow D_1 \rightarrow P_2 \rightarrow D_2$



VRPPD

Method 3 : Use the savings algorithm on the pickup points to generate a route. Then, starting with the last pickup in this route, insert the corresponding delivery point in the route where it causes least detour after its pickup. Similarly, insert all the corresponding deliveries into the route from upto the first pickup.

For example : V -> P_1 -> D_1 -> P_2-> D_2 => Better than method 2

VRPPD

Table 2. VRPPD : cost of 3 different approaches

case	# cons	1	2	3
1	3	80.64	83.78	80.64
2	7	77.57	114.32	77.01
3	8	71.41	77.89	71.89
4	30	209.72	235.17	193.11

Vehicle Routing Problem with Multiple Depots

Given m depots d_i located at loc_i each having a vehicle v_i with capacity cap_i assigned and n consignments c_j with delivery points P_j ($j = 1 \ldots n$), find routes to minimize the total distance covered by all the vehicles.

MDVRP

Algorithm : 2 phases

1. Allocate the consignments to the depots. Done based on the proximity of depots to the pickup points.
2. Now, the problem has been reduced to CVRP where each vehicle v_i with capacity cap_i has to service all the deliveries allocated to its assigned depot d_i . Hence, used savings algorithm to find all the routes to be travelled.

MDVRP - phase 1

Complexity :

O(n.m^2) considering m depots and n consignments.

Algorithm 1: MDVRP : Phase I

Input: list of depots D, list of consignments C

Output: list of depots with assigned consignments

for *c in C* **do**
    $d =$ depot nearest to c
    while *c not allocated* **do**
        if *c can be allocated to d (constraints)* **then**
            allocate c to d
        else
            $d =$next nearest depot to c

Capacitated Vehicle Routing Problem

(MDVRP Phase 2)

Given a vehicle v with capacity cap and n consignments c_i with delivery points P_i ($i = 1...n$) and a depot D where all the consignments are sourced and where the vehicle starts and must end, find a list of routes to minimize the total distance covered by the vehicle.

CVRP | MDVRP phase 2

Complexity :

$O(n^2 \log n^2)$ where n is number of consignments.

Algorithm 2: CVRP : savings algorithm

Input: integer cap, list c, list P, location D
Output: list of routes r

$savings = [\,]$
for P_i *in* P **do**
    for P_j *in* P **do**
        if $saving > 0$ **then**
            $saving = \text{dist}(P_i, D) + \text{dist}(D, P_j) - \text{dist}(P_i, P_j)$
            $savings.\text{append}([saving, P_i, P_j])$
sort $savings$ in decreasing order considering the first element of each value as key
for $[saving, P_i, P_j]$ *in savings* **do**
    r_k = route in r containing P_i
    r_m = route in r containing P_j
    if $r_k == None$ *and* $r_m == None$ *and* $c_i + c_j \leq cap$ **then**
        $r.\text{append}([P_i, P_j])$ // add new route
    else if P_i *is last stop in* r_k *and* $r_m == None$ *and weight*$(r_k)+c_j \leq cap$ **then**
        $r_k.\text{append}(P_j)$
    else if $r_k == None$ *and* P_j *is first stop in* r_m *and weight*$(r_m)+c_i \leq cap$ **then**
        $r_m.\text{append}(P_i)$
    else if P_i *is last stop in* r_k *and* P_j *is first stop in* r_m *and weight*(r_k)+*weight*$(r_k) \leq cap$ **then**
        $r.\text{remove}(r_m)$
        $r.\text{append}(r_k.\text{extend}(r_m))$
for r_i *in* r **do**
    add D at start and end of r_i
return r

MDVRP

Overall Complexity :

Considering m depots and n consignments.

$O(n.m^2) + O(\sum_{i \in [1,m]} n_i^2 \log n_i^2) \leq O(n.m^2) + O(n^2 \log n^2)$

Hence, total $O(n.m^2 + n^2 \log n^2)$ operations.

Knowledge Representation

Structured representation of objects, actions, constraints etc. of a problem that can be fed to a computer system to solve the problem.

Have presented a knowledge representation that can consider multiple variants of VRP.

Knowledge Representation

Example :

Object / Data entities

- Vehicle : sometimes called carrier or container.
  - Unique identifier - id
  - Physical properties - capacity (numerical), volume (numerical or set of numerical dimensions)
  - Configuration in some state - free_cap, free_vol
  - Assigned Consignments - list of consignments
  - Assigned Route - route
  - Location at current instant - loc
  - Constraints - fuel left, serviceable area, assigned depot, refrigeration etc.

Knowledge Representation

Example :

Actions

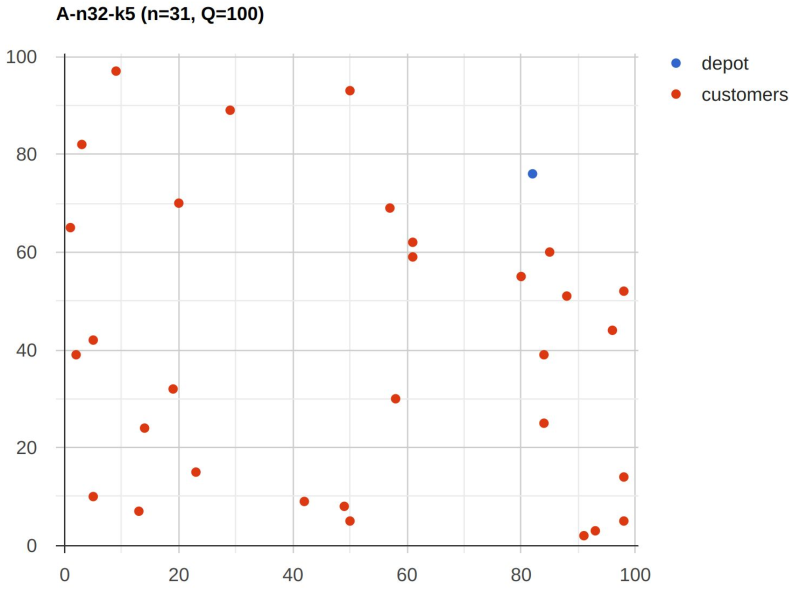
- Move(vehicle v, point loc)
  - Precondition : $location(v, loc_1) \wedge running(v)$
  - Effect : $location(v, loc)$
  - Cost : $p_1 * time(loc_1, loc) + p_2 * dist(loc_1, loc) + p_3$

Data generation

- Felt lack of standardised data with benchmarks for the real world.
- A large number of datasets available for the 2 dimensional Euclidean system with euclidean distances as the edge costs. (Recent work by Queiroga et.al. generated 10000 instances for CVRP)
- Scope for novel ways of generating artificial test data for real world VRP problems.

Data generation

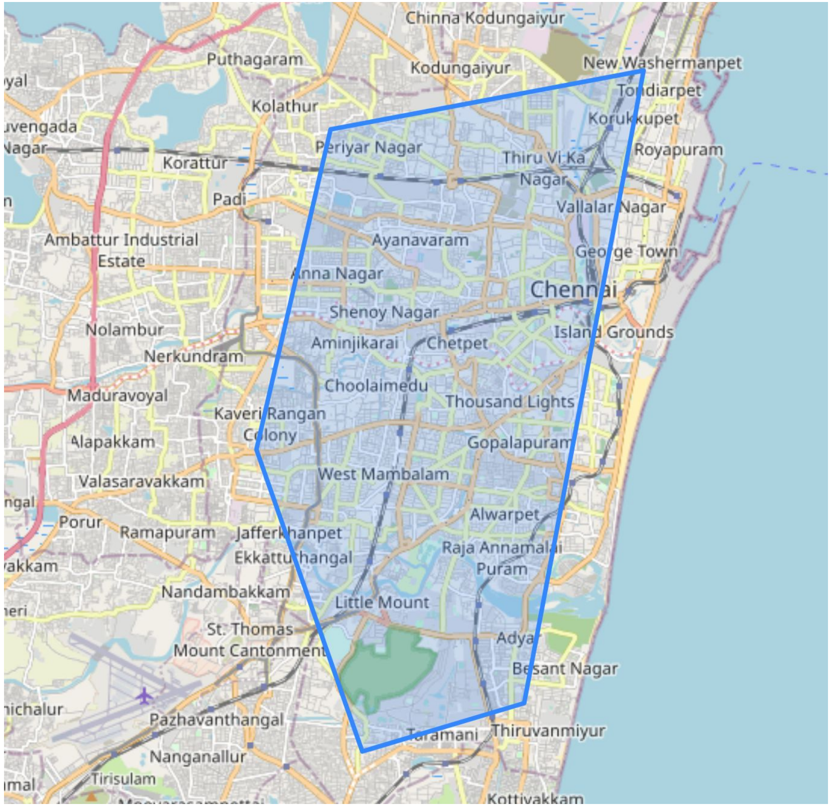
1. Standard Dataset



http://vrp.atd-lab.inf.puc-rio.br/index.php/en/plotted-instances?data=A-n32-k5

Data generation

2. Polygon for geographical area



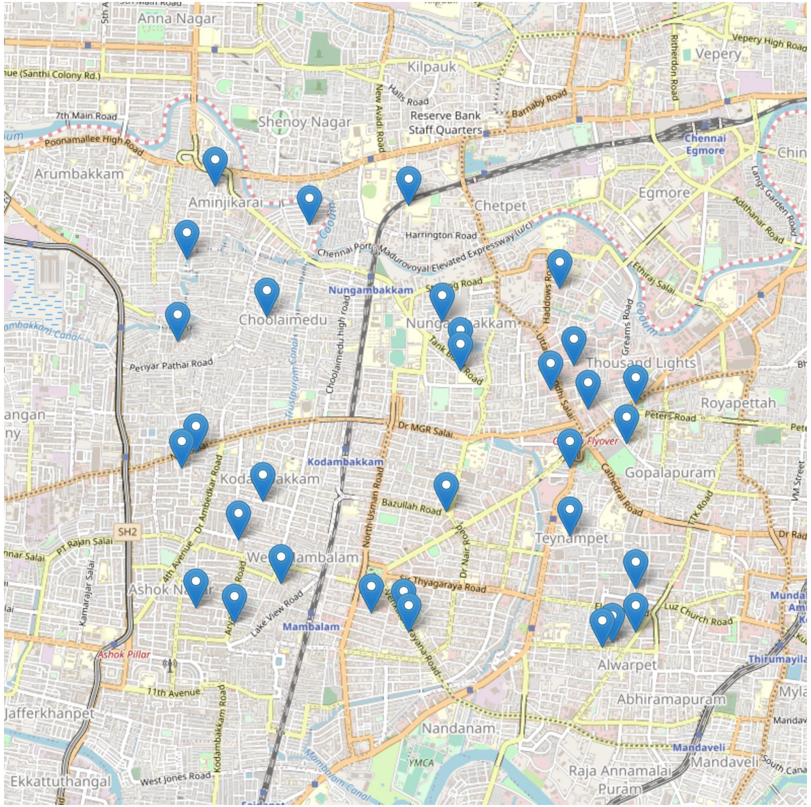
Data generation

3. Lines of latitudes and longitudes -

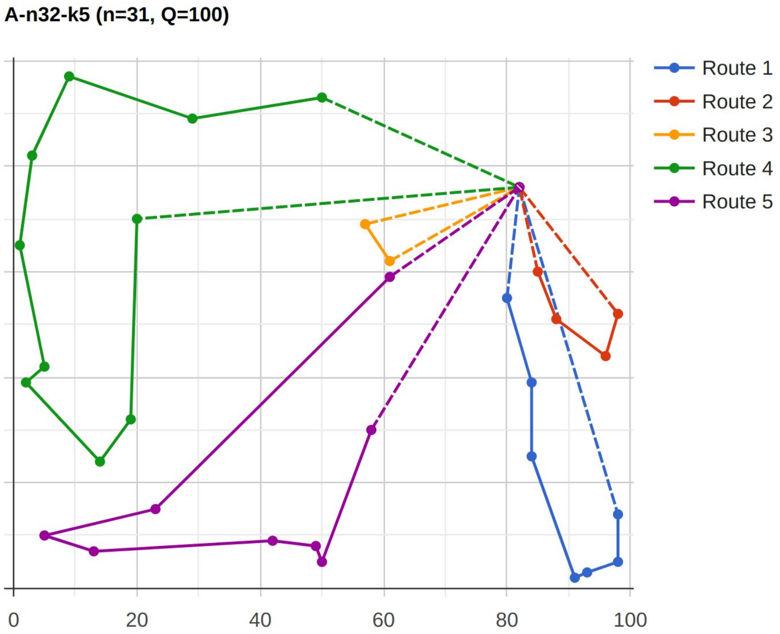
“as the crow flies” distance

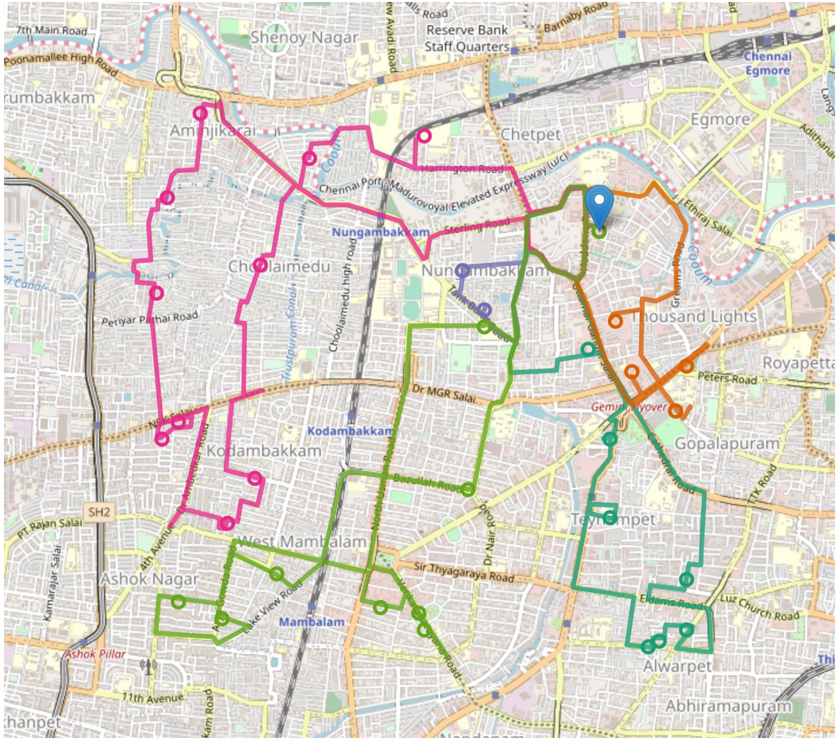
Data generation

4. Embed points into the geographical area.



Data generation





http://vrp.atd-lab.inf.puc-rio.br/index.php/en/plotted-instances?data=A-n32-k5

Data generation

Factor of 1.6x between euclidean and road distances (road distances greater by 60%)

Table 1. difference between embedded and original

Dataset	mean of distances	% rms difference
A-n32-k5	3868	12.9
A-n45-k6	3780	12.8
A-n64-k9	3151	15.7
A-n80-k10	3409	13.4

Further Work

- Comparative studies on other variants of VRP can be undertaken in a similar fashion. Using available solutions to one variant of VRP, new solutions can be devised for other variants also.
- The performance of the method of embedding to generate data can be compared for different geographical entities. Different sets of data such as the CVRP data can also be utilised or combined to generate data for other variants of VRP such as MDVRP or VRPPD.
- Similar ways of generating data can also be explored for the rescheduling and orienteering problems.
- APIs integration.

Thank You