IT 3708: Project 2

Solving The Multiple Depots Vehicle Routing Problem (MDVRP) Using Genetic Algorithm (GA)

Lab Goals

- Implement genetic algorithm (GA) to solve a NP-hard combinatorial optimization problem ─ The Multiple Depots Vehicle Routing Problem (MDVRP).
- Compare the performance of your solutions on several benchmark problems.
- Test and analyze effects of genetic operators and related parameters.

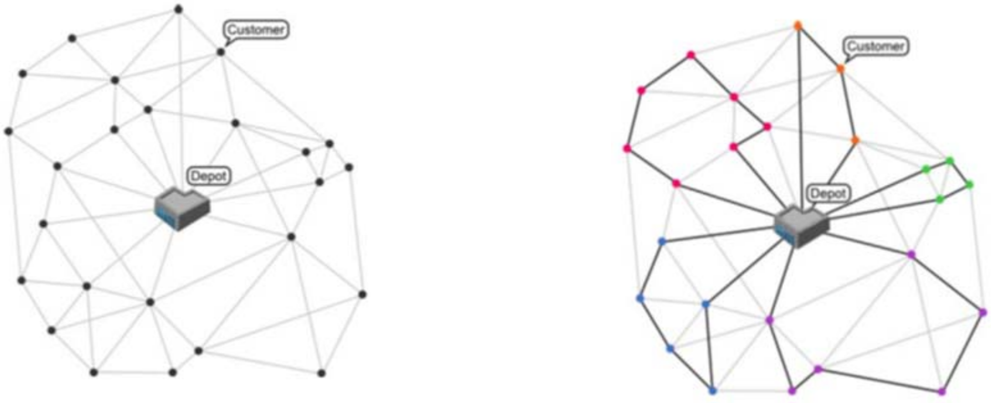
Groups Allowed? No, you have to submit and demonstrate individually.

Deadline: March 02, 2017 (Thursday) at 07: 59 AM.

Assignment Details

Vehicle routing problems (VRPs) are classical combinatorial optimization problems which have received much attention in recent years due to their wide applicability and economic importance. VRP formulations are used to model an extremely broad range of issues in many application fields, transportation, supply chain management, production planning, and telecommunication, to name but a few. Many of these problems are NP-hard, and they usually define challenging search problems that are good for analyzing heuristic search techniques.

A typical VRP can be stated as follows: a set of geographically dispersed customers with known demands are to be serviced by a homogenous fleet of vehicles with limited capacity. Each customer is to be fully serviced exactly once and each vehicle is assumed to start and end at exactly one depot, and the **primary objective is to minimize the total distance travelled by all vehicles**. However, in a large number of practical situations and to satisfy real-life scenarios, additional constraints are usually defined for variants of the VRP.



(a) (b)

Fig. 1: A hypothetical instance of a VRP (a) and its solution (b)

In this project, you need to solve the multi-depot VRP (MDVRP), which is an extension of the classical VRP with the exception that there are multiple depots with a number of vehicles and every customer should be serviced by a vehicle based at one of several depots. Similar to the typical VRP, in the MDVRP every vehicle route also must start and end at the same depot. The MDVRP is NP-hard, which means that an efficient algorithm for solving the problem to optimality is unavailable. Even for relatively small problem sizes, the MDVRP is difficult to solve to optimality. Therefore, solving the MDVRP by an exact algorithm is time-consuming and computationally intractable. For this reason, heuristic/meta-heuristic algorithms are good choices to solve the MDVRP. **In this project, you will solve the MDVRP using a well-known meta-heuristic, called the genetic algorithm (GA)**.

Problem Formulation:

The MDVRP is about how a transportation company decides to make a most efficient schedule for their vehicles. The transportation company has multiple depots from which their vehicles depart and arrive, and has multiple customers being served from the different depots. **The challenge is to make a schedule for each vehicle individually** so that the vehicles drive in the most efficient way optimizing one or several objectives.

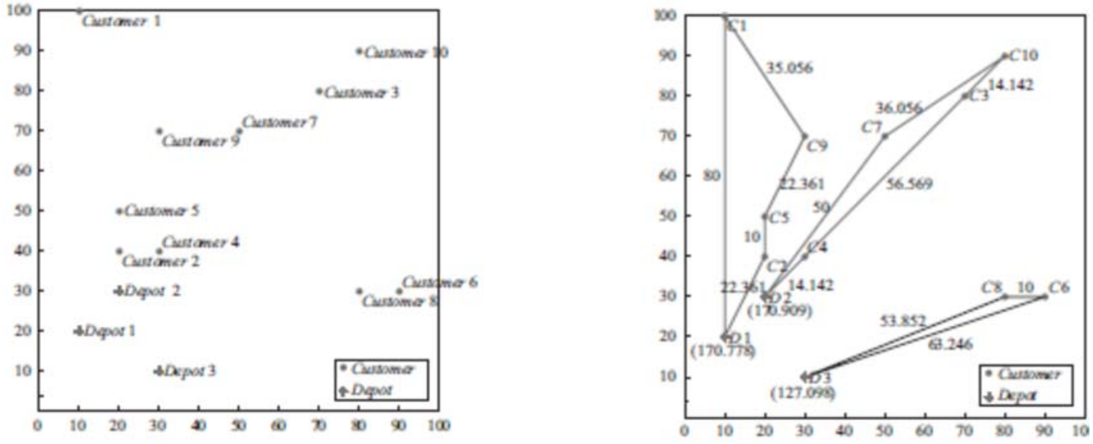
Formally, the MDVRP can be defined as follows. We are given a set of depot locations and a set of customer locations, which are assumed to be disjoint (even if two points share the same physical coordinates, they are still handled as different entities). Each customer is characterized by their own demand. A fleet of vehicles with limited capacity is based at each depot. Each vehicle originates from one depot, service the customers assigned to that depot, and return to the same depot. The MDVRP consists of simultaneously determining the routes for several vehicles from multiple depots to a set of customers and then return to the same depot optimizing predefined objective(s) as well as satisfying the following conditions:

(i) every customer appears on exactly one route.

(ii) every route starts and ends at the depot.

(iii) the total demand of the customers on any route does not exceed a vehicle's capacity.

(iv) the total duration of a route does not exceed a preset value *(for this project, it is only for those problems for which this value is mentioned in the test data).*



(a)

(b)

Fig. 2: A hypothetical instance of a MDVRP: (a) locations of customers and depots, (b) one solution

Fig. 2 presents an example for MDVRP with an arbitrary solution. Generally, the objectives of the MDVRP are to minimize the total delivery distance, to minimize the time spent in serving all customers, or to minimize the number of vehicles needed in serving all customers. In this project, **your main goal is to minimize the total distance travelled by all vehicles across the depots in serving all customers**. At the same time, you should minimize the number of vehicles for each depot, i.e. your solution should minimize the total number of vehicles across the depots in serving all customers (**if possible**).

To solve the MDVRP, the decision makes have to decide three decisions as shown in Fig. 3. The decision makers first need to cluster a set of customers to be served by the same depot, that is, the grouping problem. They then have to assign customers of the same depot to several routes so that the vehicle capacity constraint is not violated. At last, the decision on delivery sequence of each route is made.

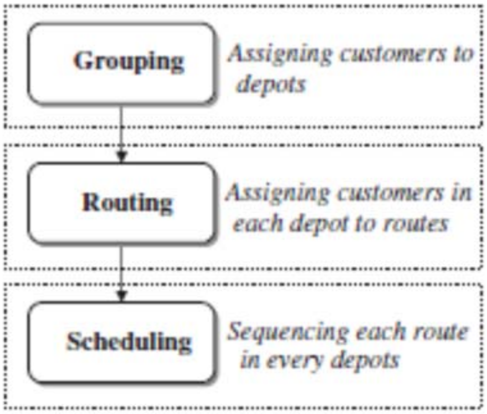

Grouping
Assigning customers to depots
Routing
Assigning customers in each depot to routes
Scheduling
Sequencing each route in every depots

Fig. 3: The hierarchy of decisions in the MDVRP.

Algorithm

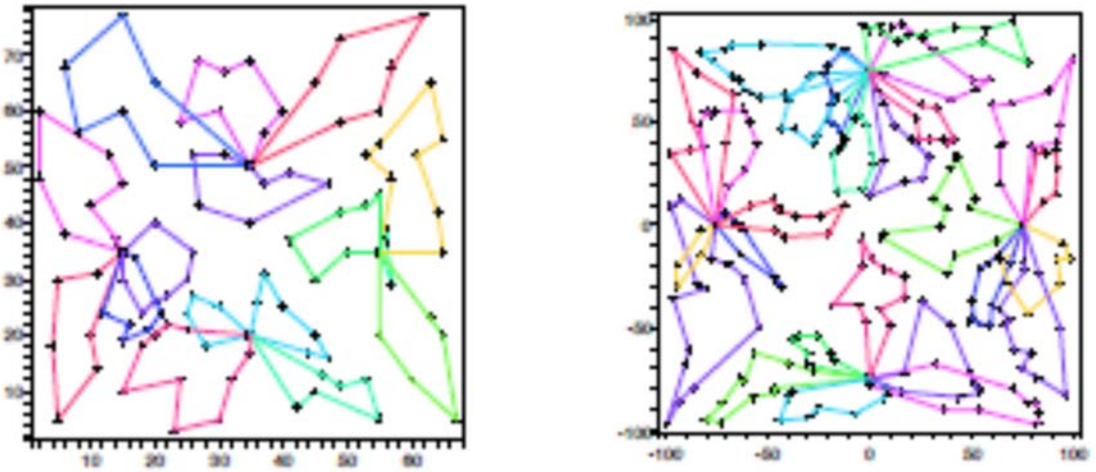
As mentioned earlier, to solve the MDVRP, you need to implement the genetic algorithm (GA) that has been already discussed in lectures. In order to get the optimal/near-optimal results, you may check several forms of representation, genetic operators, and selection mechanism. It would be beneficial to test whether or not *elitism* gives a better solution. However, in demo and report, you only need to mention the final representation, genetic operators, selection and elitism strategy (if implemented) that gave you the optimal/near-optimal solutions.

Note that, GA parameter values (population size, generation number, crossover rate, and mutation rate) are correlated and your GA will successfully find the optimal values if you use appropriate parameter values. However, there is no definite rule to find the appropriate parameter values. Therefore, you should test different set of parameter values to decide the appropriate values.

Things To Do

The 20 points total for this project is 20 of the 100 points available for this course. The 20 points will be distributed on two parts: (*i*) demo and (*ii*) report. **The demo can give you a maximum of 14 points and the report can give you a maximum of 06 points**.

To test your code, we uploaded 23 benchmark problems and their solutions. The description of problem and solution data file formats is also included. Note that, for the demo session, **your code must have the option to read the test data according to the given format**. Also, you need to show the output (solution) as described in the provided solution data files as well as graphically (Fig. 4). **You must use different colors to represent different routes from a single depot** as shown in the figures.



(a) Solution for a test problem with 100 customers, 2,3,4 depots with the capacity limit of 100

(b) Solution for a test problem with 249 customers, 4 depots with capacity limit of 500 and route limit of 310

Fig. 4: Solution representation

(a) Demo (14p) :

There will be a demo session where you will show us the running code and we will verify that it works. In the demo session, you need to describe how you designed and implemented your GA. Also, you have to test you code by running 03 (three) test problems that we will be supplied during the demo. The point distribution for the demo is as follows:

(1) Testing 3 test problems (12p = 4 x 3)

- Does your solution find the known optimal value for travel-distance value considering all vehicles across the depots in serving all customers? (3p)
    - If your value is within 10% of the known optimal value, you will get full points.
    - If your value is within 20% of the known optimal value, you will get 2.5 point.
    - If your value is within 50% of the known optimal value, you will get 1 point.
    - Otherwise, you will get 0.
- Present your solution strictly following the graphical format mentioned in Fig. 4 using different colors to represent different routes from a single depot. (0.5p)
- Present your solution strictly following the format mentioned in provided solution data files (also mentioned below). (0.5p)

```
576.87
1  1  60.06  71  0 44 45 33 15 37 17 0
1  2  66.55  79  0 42 19 40 41 13 0
1  3  47.00  78  0 25 18 4 0
2  1  53.44  73  0 6 27 1 32 11 46 0
2  2  79.47  80  0 48 8 26 31 28 22 0
2  3  81.40  77  0 23 7 43 24 14 0
2  4  23.50  54  0 12 47 0
3  1  50.41  75  0 9 34 30 39 10 0
3  2  25.22  54  0 49 5 38 0
4  1  47.67  67  0 35 36 3 20 0
4  2  42.14  69  0 21 50 16 2 29 0
```

(2) Describing your GA, such as representation, genetic operators, selection, elitism strategy, parameters, and other related issues. (2p)

(b) Report (06p):

You should write a report answering the points below. Your report must not exceed 02 (two) pages in total. **Over length reports will result in points being deducted from your final score**. Print on both sides of the sheet, preferably. **Bring a hard copy of your report to the demo session**.

1. Describe your implementation for the following. (2p = 0.5 x 4)
   - Chromosome representation.
   - Crossover operator.
   - Mutation operator.
   - Selection mechanism.
2. Mention the parameter values (population size, generation number, crossover rate, and mutation rate) that gave you the best result. (0.5p)
3. Describe whether or not the crossover and mutation operators will produce infeasible off-spring(s) after executing. If yes, how did you handle that? If not, why? (1p)
4. Describe your fitness function. (1p)
5. Describe whether or not you implemented elitism. If yes, how? If not, why? (0.5p)
6. Using the parameter values mentioned in (b), present the solution of any one of the test problems. The solutions should be presented as ***(i)* graphically, and (*ii*) in the provided solution data file format**. You should follow the solution presentation guidelines mentioned earlier. You must present the solutions using screen shots from your computer. (1p)

Delivery

You should deliver your report + a zip file of your code on *itslearning*. The submission system will be closed at 07:59 AM on March 02, 2017.

You must attend the demo on the scheduled demo date which has been declared on *itslearning***.** Since the demo dates were declared at the beginning of the semester**, no early or late demo will be entertained** except for extreme emergency like sickness with medical certificate, job interview, attending funeral or the like**.** Traveling or holidays will not be considered as emergency situation.