Excercise 4
Implementing a centralized agent

Group 5: Ignacio Aguado, Dario Martinez

November 8, 2016

1 Solution Representation

1.1 Variables

Defining $V = v_1, v_2, ..., v_{N_v}$ the set of vehicles owned by the company with Nv the number of vehicles. And let $T = t_{1_p}, t_{1_d}, t_{2_p}, t_{2_p}, ..., t_{Nt_p}, t_{Nt_d}$ the set of to tasks to be picked up and delivered with Nt the number of tasks to be picked up and delivered. Note that for every task t there is a pickup action t_p and a delivery action t_d.

The plan is defined as a set of schedules, and each schedule is a tuple composed by a vehicle and the actions that vehicle must perform:

$$\begin{aligned} Plan = \{ &< v_1, (t_{1_p}, t_{1_d}, t_{2_p}, t_{2_p}, ...) >, \\ &< v_2, (t_{1_p}, t_{1_d}, t_{2_p}, t_{2_p}, ...) >, \\ &\dots, \\ &< v_{Nv}, (t_{1_p}, t_{1_d}, t_{2_p}, t_{2_p} ...) > \} \end{aligned}$$

1.2 Constraints

1. The weight of the picked up tasks in a vehicle must not exceed the capacity of that vehicle.
2. In a plan, the pick up action of task must not appear after the delivery and the delivery action must not appear before the pick up.
3. In a plan, every action must appear once and only once.
4. Two actions belonging to the same task must appear in the same schedule.

1.3 Objective function

The objective function to optimize is, as presented in the recommended paper, the cost of the company because the benefit for delivering a task is going to be constant. The cost of the company is going to be the sum of the marginal cost of each vehicle for carrying the assigned tasks and the cost of the vehicle is defined as the distance between the two tasks in kilometers multiplied by the cost of each kilometer:

$$Cost = \sum_{n=1}^{Nt} \sum_{n=1}^{Nv} dist(v_{city}, t_{city}) * costperkm$$

2 Stochastic optimization

2.1 Initial solution

We have implemented three different strategies for generating the initial solution and we tested which one gives the best results. First, we tried a round-robin assignment of tasks, this will give us the most fair scenario in which the tasks distributes equally depending in the cost of each vehicle.The second strategy was to give all the tasks to the vehicle with the biggest capacity and this seems to be a better solution that converges faster than the round robin but we believed that it was not fair. Finally, we decided that the best strategy was to distribute each task to the nearest vehicle, this allows us to minimize the cost in a fair scenario with decent convergence.

2.2 Generating neighbours

To generate the neighbors we followed a very similar structure that in the paper, two methods were implemented:

- `changeVehicle`: This method takes iterates over every vehicle and takes the first (pickup and delivery) tasks of a vehicle a gives it to another vehicle, checking if that this action don't violate any defined constraint.
- `changeOrder`: For every pair of tasks in a vehicle's schedule the order is changed, checking that thhe the order swap don't break any of the defined constraints.

2.3 Stochastic optimization algorithm

The stochastic optimization algorithm implemented is a Stochastic Local Search. The `localChoice` method selects the assignment with the less cost from the neighbor list generated by the methods explained in the last subsection with a probability p and returns the old plan with a probability $1 - p$. This will prevent the algorithm to get stuck in a local minimum and not find a near more optimal solution. Also if two two assignments have the same cost we chose the new plan with the probability p or we choose the old one with the probability $1 - p$.

3 Results

All the experiments has been performed in a MacBook Pro 2,5 GHz Intel Core i7 with 16 GB DDR3 RAM 1600 MHz.

3.1 Experiment 1: Model parameters

In this experiment we have test the algorithm for the three different strategies implemented for generating the Initial Solution.

3.1.1 Setting

	Strategy	Topology	N tasks	N vehicles	Cost/km	p	Iterations
Test 1	Biggest vehicle	England	30	4	5 5 5 5	0.3	3000
Test 2	RR	England	30	4	5 5 5 5	0.3	3000
Test 3	Nearest vehicle	England	30	4	5 5 5 5	0.3	3000

3.1.2 Observations

	Cost	Distance	N tasks/vehicle
Test 1	29777	5943	30 0 0 0
Test 2	33878	6760	8 9 6 7
Test 3	18600	3705	12 7 4 7

The best strategy for the initial solution is the to distribute each task to the nearest vehicle. Although this strategy is not the fairest, as it can be observed that the Round Robin is, it gives the best cost optimization. Giving all the tasks to the vehicle with the biggest capacity seems to give better results than the Round Robin but it is not fair as it gives all the tasks to one vehicle only.

3.2 Experiment 2: Different configurations

In this experiment we have test the algorithm with different configurations of the environment, changing the number tasks and the number of vehicles. We have selected the distance strategy in the all the tests except for the last one (in which we test the fairness of the two algorithms by changing the cost per kilometer of the vehicles).

3.2.1 Setting

	Strategy	Topology	N tasks	N vehicles	Cost/km	p	Iterations
Test 1	Nearest vehicle	England	5	4	5 5 5 5	0.3	3000
Test 2	Nearest vehicle	England	35	4	5 5 5 5	0.3	3000
Test 3	Nearest vehicle	England	80	4	5 5 5 5	0.3	3000
Test 4	Nearest vehicle	England	30	3	5 5 5 5	0.3	3000
Test 5	Nearest vehicle	England	30	8	5 5 5 5	0.3	3000
Test 6	Nearest vehicle	England	30	12	5 5 5 5	0.3	3000
Test 7	Nearest vehicle	England	30	4	10 1 3 6	0.3	3000
Test 8	Round Robin	England	30	4	10 1 3 6	0.3	3000

3.2.2 Observations

	Cost	Distance	N tasks/vehicle
Test 1	7146	1425	1 3 0 1
Test 2	24900	4963	15 7 5 8
Test 3	71632.5	14291	37 13 13 17
Test 4	23850.5	4758	24 36
Test 5	19288	6760	6 4 4 7 1 4 3 1
Test 6	21333	4254	2 8 4 4 2 4 4 2 4 4 2 10
Test 7	22910.4	3638	12 7 4 7
Test 8	29590.8	6361	6 11 6 7

In this experiment we can conclude that a the nearest vehicle strategy is not fair. If we increase the number of tasks in the model, the majority of the tasks are assigned to the first vehicle even thought all the vehicles have the same cost. Also If we change each vehicle cost all the behaviour is similar but, on the other hand, with the Round Robin strategy the tasks are distributed proportionally to the cost of each vehicle. The reason for this behaviour is that only one vehicle's schedule where its load is maximized approaches an optimal solution. The algorithm is not optimal, as it will always find a local solution but is not guaranteed to be a global maximum. In our case, the first vehicle seems to have no motivation in exchanging tasks with other vehicles but only to change the order in his schedule.