A Capacitated Time Constrained Vehicle Routing Problem with Crew Allocation for White Gloves Service: A Multiple Pickup, Delivery and Installation Problem for Last Mile Delivery

Final Year Project Review – May 9th 2021

Soorya Sriram (MDM18B049)

Academic Supervisor:
Dr Kalpana P

Dept of Mechanical Engineering,
IIITD&M Kancheepuram

Last mile Delivery(LMD)

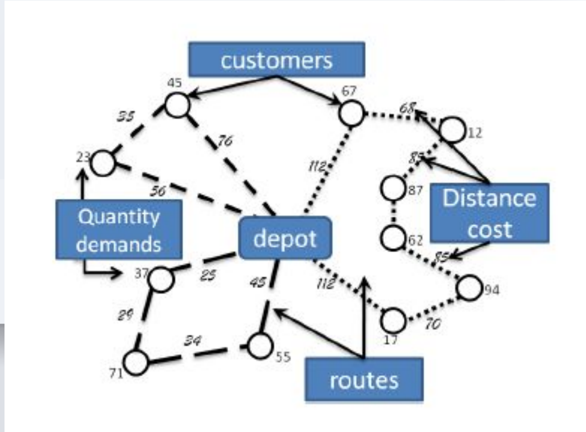
Vehicle Routing Problem is a major problem in Operations Research and Industrial Engineering.

One sub problem is LMD. The last mile refers to the final step that any company undertakes to make sure the final good ordered by the customer reaches the desired destination.

The global e-commerce market for furniture and appliances alone that require white glove service is valued at approximately **$199 billion**. That’s roughly **17 percent** of the existing e-commerce market.

White Gloves Service

- **Special elite service to customers used to deliver specialty goods**
- **Installation, assembly of goods like furniture or specific equipment and disposal excess package material**
- **It is a delivery and installation service provider for valuable, fragile and climate-sensitive items**



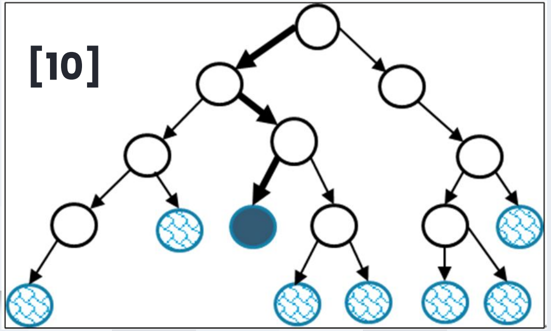
Problem being addressed

- **A novel closed Vehicle Routing Problem involving multiple pickup, delivery and installation.**
- **A single fleet allocation is done where crews with specific skill sets start from a single depot and return to the starting location at the end of service.**
- **Constraints suchs as time windows, vehicle capacity, installation demands and working hour have to be satisfied.**

A VRP Network Map [9]

Branch and Bound Algorithm

1. Used for Solving Combinatorial Mathematical Optimization Problems
2. A systematic approach to find candidate solutions using tree method with the full set at the root
3. Each branch is checked against upper and lower estimated bounds on the optimal solution.
4. Solution is discarded if it cannot produce a better solution than the best one found so far by the algorithm.



Branch and Cut Algorithm

1. A faster method than the branch and bound algorithm
2. Use of Gomory Cutting plane method in addition to Branch and Bound to reduce the feasible solution space where all solutions for the unknowns are integer values
3. Branch and Cut algorithm can be used to find the lower bounds even if optimal solution can not be found

Haversine Formula – Used to calculate the distance between nodes

Haversine formula:

$$a = \sin^2(\Delta\varphi/2) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot \sin^2(\Delta\lambda/2)$$

$$c = 2 \cdot \operatorname{atan2}(\sqrt{a}, \sqrt{(1-a)})$$

$$d = R \cdot c$$

where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km); note that angles need to be in radians to pass to trig functions!

1. **Travelling Salesman Problem**
2. **Vehicle Routing Problem with capacity constraints**
3. **CVRP with time windows**
4. **Pickup and Delivery VRP**

Problem Solved : CVRPTW Multiple Pickup Delivery and Installation with Crew Allocation

Mathematical Model

First, the Pickup, Delivery and Installation using a single fleet satisfying the constraints mentioned has to be captured.

Decision Variables

Decision Variables are the unknown that we are trying to solve. They are as follows:

(i) Location i to j using with crew c (3 index variable) - a binary variable with the value 1 if crew c travels from node i to node j and is zero otherwise - **xijc**

(ii) The load being carried by crew c from location i to j - **lijc**

(iii) The time taken by crew c to deliver and install from depot to location i to j and back to depot - **tijc**

Parameters

Parameters are known values that have to be provided as input

Number of Pickup Locations - **picknum**

Number of Drop Locations - **dropnum**

Number of Vertex Locations - **vertexnum**

Number of Crews - **crewnum**

Product ID - **prodid**

Vertex Names - **vertexname**

Pickup Nodes - **pickupnode**

Drop Nodes - **dropnode**

Earliest Time - **ET**

Latest Time - **LT**

Vehicle Capacity - **200**

Start of time window - **a**

End of time window - b

Service time at each node - **s**

Graph arcs - **arcs**

Distance between each node - **dist**

Cost of travelling from 1 node to another by a crew - **cost**

Time taken to travel from 1 node to another - **time**

Load at each node - **load**

Installation time for each - **installtime**

Crew Skillset - **crewskills**

Installation or only Delivery Required defined by binary variable - **installation**

Objective Function

Minimise the total transportation cost.

$$Minimize \sum_{c=1}^{C} \sum_{i=1}^{vertex_{num}} \sum_{j=1}^{vertex_{num}} cost_{ijc} * x_{ijc}$$

Constraints - (14)

- Each pickup customer is allocated to only one crew
- Each request is served exactly once by a crew from the depot or starting point
- Each pickup and delivery/installation is visited only once by the same crew
- No return to previous node, starts from the depot and ends at depot
- Closed VRP where every crew returns back to the depot
- Service Time at customer i should be lesser than customer j
- Start of Service time after start of customer time window
- 8hr working hr constraint for closed VRP
- Precedence of time at pickup node should be less than drop node
- Load at every pickup node should be less than vehicle capacity
- Load at drop node should be lesser than vehicle capacity and greater than 0
- Load at depot for every crew should be zero
- Vehicle Capacity Constraint - the load of a vehicle never exceeds its capacity
- Matching Installation skill set required with the skill set of servicemen in crew

Base Experiment

- **Three large products** such as Air Conditioner,Bed and Washing Machine were considered of **product weight of 30kg, 60kg and 50kg.**
- The Standard vehicle capacity was taken to be **200kg** with a average speed of **50km/hr** to create the **time matrix** between 2 nodes using the **distance matrix generated using Haversine Formula.**

Locations and their Time windows

ID	Location	Latitude	Longitude	StartTW	EndTW	ServiceTime
1	Depot	12° 50' N	80° 8' E	0	480	0
2	P1	13° 2' N	80° 13' E	30	92	10
3	P2	13° 4' N	80° 11' E	15	67	10
4	C1	13° 5' N	80° 9' E	169	224	30
5	C2	12° 58' N	80° 7' E	166	235	30

Product Request

ID	PickupID	CustomerID	ProductID	Del/Ins
1	3	4	1	1
2	3	4	2	0
3	3	4	3	1
4	2	5	2	1
5	2	5	3	0

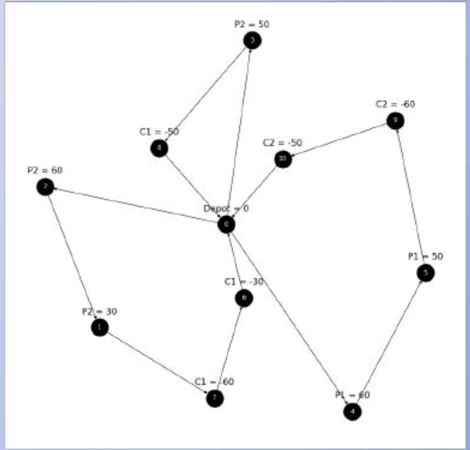
Crew Installation Time

CrewID	ST1	ST2	ST3
1	30	0	30
2	30	30	0
3	0	30	30

Crew Skill Set

CrewID	S1	S2	S3	Crew Cost
1	1	0	1	100
2	1	1	0	100
3	0	1	1	100

Network Map



Experiment 1.

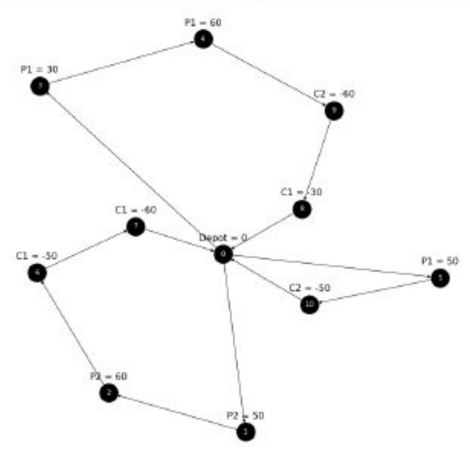
Change in Product Request - 3 experiments

Product Request – (2) & (3)

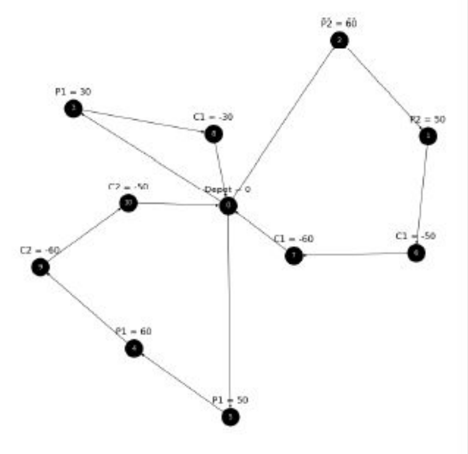
ID	PickupID	CustomerID	ProductID	Del/Ins
1	3	4	3	0
2	3	4	2	1
3	2	5	1	1
4	2	5	2	0
5	2	5	3	1

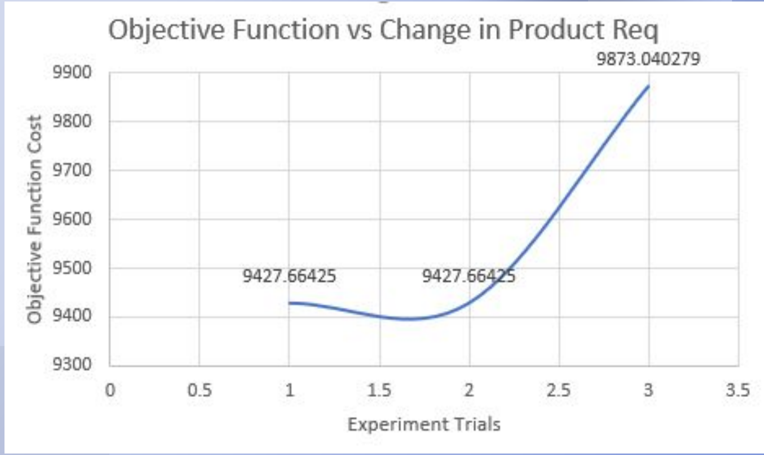
ID	PickupID	CustomerID	ProductID	Del/Ins
1	3	4	3	0
2	3	4	2	1
3	2	4	1	1
4	2	5	2	0
5	2	5	3	1

Network Map – PR (2)



Network Map – PR (3)





Inference

Cost changed drastically due to change in product request by almost 400.

Experiment 2.

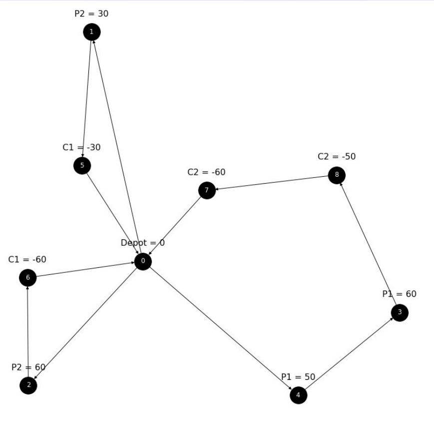
Change in Product Number - 3 experiments

Product Number – (1) & (2)

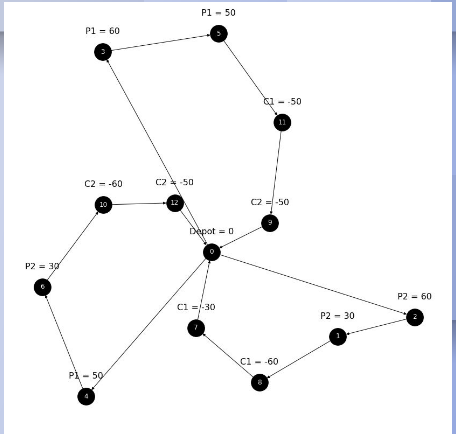
ID	PickupID	CustomerID	ProductID	Del/Ins
1	3	4	1	1
2	3	4	2	1
3	2	5	2	0
4	2	5	3	1

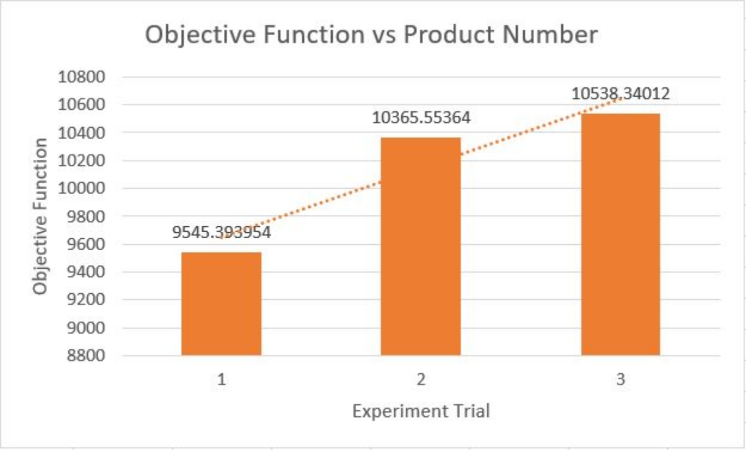
ID	PickupID	CustomerID	ProductID	Del/Ins
1	3	4	1	1
2	3	4	2	0
3	2	5	2	1
4	2	5	3	1
5	2	5	3	0
6	2	5	3	0

Network Map – PR (1)



Network Map – PR (2)





Inference

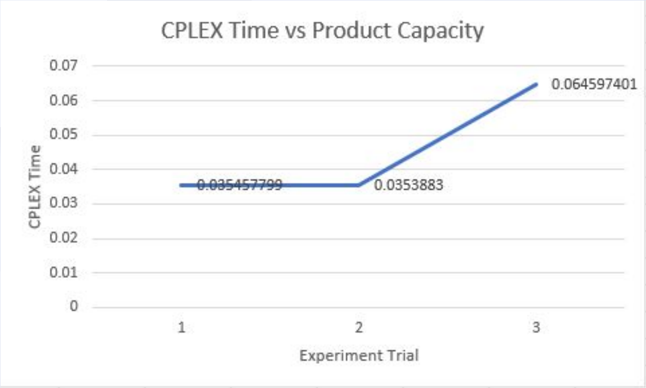
A linear increase in objective function cost value from product number 4 to 7 in the third trial.

Experiment 3.

Change in Product Capacity - 3 experiments

Product Capacity – (1) & (2)

1. Three large products such as Air Conditioner,Bed and Washing Machine were considered of product weight of **100kg, 70kg and 60kg.**
2. Three large products such as Air Conditioner,Bed and Washing Machine were considered of product weight of **80kg, 40kg and 90kg.**



Output – PC (1)

CrewID	Routes	Service Time	Route Locations	Loads
1	0, 1, 6, 0	0.0, 31.8, 224.0, 480.0	Depot, P2, C1, Depot	0.0, 100.0, -100.0, 0.0
2	0, 4, 5, 9, 10, 0	0.0, 30.0, 39.9, 166.0, 226.0, 480.0	Depot, P1, P1, C2, C2, Depot	0.0, 70.0, 60.0, -70.0, -60.0, 0.0
3	0, 3, 2, 7, 8, 0	0.0, 31.8, 41.8, 169.0, 224.0, 480.0	Depot, P2, P2, C1, C1, Depot	0.0, 60.0, 70.0, -70.0, -60.0, 0.0

Output – PC (2)

CrewID	Routes	Service Time	Route Locations	Loads
1	0, 1, 2, 7, 6, 0	0.0, 31.80, 67.0, 169.0, 224.0, 480.0	Depot, P2, P2, C1, C1, Depot	0.0, 80.0, 40.0, -40.0, -80.0, 0.0
2	0, 4, 5, 10, 9, 0	0.0, 30.0, 92.0, 166.0, 235.0, 480.0	Depot, P1, P1, C2, C2, Depot	0.0, 40.0, 90.0, -90.0, -40.0, 0.0
3	0, 3, 8, 0	0.0, 31.8, 224.0, 480.0	Depot, P2, C1, Depot	0.0, 90.0, -90.0, 0.0

Inference

As the number of product capacities increases, that is as number of products increases, the time taken by the solver to find optimal value increases.

Experiment 4.

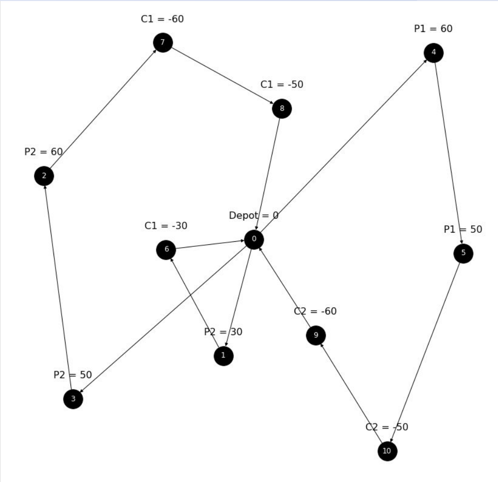
Change in Crew - 3 experiments

Crew Change – (2) & (3)

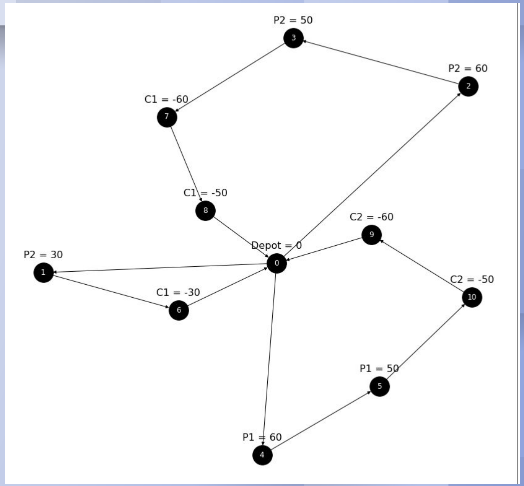
CrewID	S1	S2	S3	Crew Cost
1	0	0	1	100
2	0	1	0	100
3	1	0	0	100

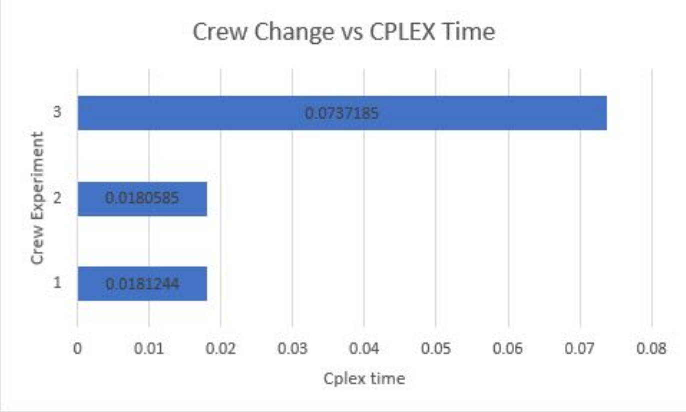
CrewID	S1	S2	S3	Crew Cost
1	1	0	1	100
2	0	1	1	100
3	1	1	0	100

Network Map – CC (2)



Network Map – CC (3)





Inference

The optimal value is reached earlier when the skillset for crews is lesser. As the complexity increases and crews have 2 skillsets each, the optimal value is reached substantially later.

Experiment 5.

Change in Time Window - 3 experiments

Time Window – (1) & (3)

StartTW	EndTW
0	480
99	148
68	149
261	316
278	345

StartTW	EndTW
0	480
31	100
12	77
186	257
357	410

Output – TW (1)

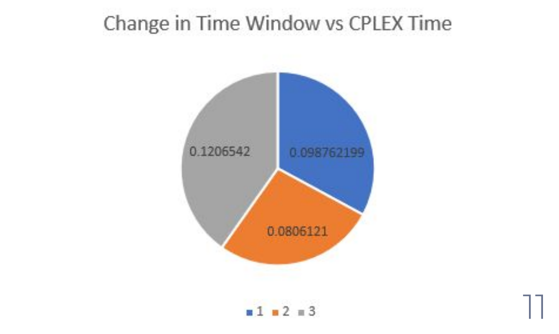
CrewID	Routes	Service Time	Route Locations	Loads
1	0, 4, 5, 10, 9, 0	0.0, 99.0, 148.0, 278.0, 308.0, 480.0	Depot, P1, P1, C2, C2, Depot	0.0, 60.0, 50.0, -50.0, -60.0, 0.0
2	0, 3, 2, 7, 8, 0	0.0, 68.0, 149.0, 261.0, 291.0, 480.0	Depot, P2, P2, C1, C1, Depot	0.0, 50.0, 60.0, -60.0, -50.0, 0.0
3	0, 1, 6, 0	0.0, 68.0, 260.9, 480.0	Depot, P2, C1, Depot	0.0, 30.0, -30.0, 0.0

Output – TW (3)

CrewID	Routes	Service Time	Route Locations	Loads
1	0, 1, 6, 0	0.0, 31.8, 186.0, 480.0	Depot, P2, P2, C1, C1, Depot	0.0, 30.0, -30.0, 0.0
2	0, 4, 5, 10, 9, 0	0.0, 31.0, 41.0, 357.0, 402.07, 480.0	Depot, P1, P1, C2, C2, Depot	0.0, 60.0, 50.0, -50.0, -60.0, 0.0
3	0, 2, 3, 8, 7, 0	0.0, 31.8, 41.8, 186.00, 245.9, 480.0	Depot, P2, C1, Depot	0.0, 60.0, 50.0, -50.0, -60.0, 0.0

Inference

The solving time is independent of the time window values however the optimal routes generated are different for each case.



Other Experiments Conducted

1. Change in the Installation Time for each crew
2. Change in Vehicle Capacity - problem could not be solved when constraint violated
3. Change in Crew Cost
4. Change in Delivery Time

Future Scope

1. The model can be extended to larger industry datasets which is currently unavailable.
2. Near optimal solutions can be achieved by using heuristic techniques like simulated annealing and genetic algorithm.
3. TABLEAU and other visualisation tools could have been used for realistic visualisation of the routes generated.

Conclusion

1. A novel mathematics model was proposed for the multi pickup, delivery and installation vehicle routing problem involving time window and capacity constraints including crew allocation using a single fleet based on the product demand and skill set required for such a service.
2. Various Experiments were conducted and output was cross validated using network maps and by hand.
3. The Branch and Bound algorithm was used to find the accurate solution for a small dataset to verify the validity of the model proposed.
4. TSP, VRP, VRPTW, PDVRP were also understood and executed before Installation using crew allocation was understood and implemented.

Other Variants Executed

TSP

```
Route
0     3
1     8
2     0
3     1
4     9
5     2
6     4
7     5
8     6
9     7

Ordered Route
0
3
1
8
6
4
9
7
5
2
0
```

VRP

```
Number of Routes : 3

Ordered Route

Route : 1
0 - Madras (Chennai)
12 - Krishnagiri
20 - Yercaud
8 - Salem
3 - Namakkal
5 - Karur
9 - Tiruchchirappalli
11 - Thanjavur (Tanjore)
0 - Madras (Chennai)

Route : 2
0 - Madras (Chennai)
13 - Vriddhachalam
1 - Kanchipuram
18 - Tiruvallur
0 - Madras (Chennai)
```

```
Route : 3
0 - Madras (Chennai)
14 - Tiruvannamalai
16 - Tiruchengodu
15 - Tiruppur
7 - Satyamangalam
17 - Mettupalaiyam
2 - Coimbatore
6 - Palni Hills
4 - Madurai
19 - Kilakarai
10 - Pudukkottal
0 - Madras (Chennai)
```

VRPTW

```
Routes=[[0, 5, 3, 7, 8, 10, 11, 9, 6, 4, 2, 1, 75, 101],
[0, 81, 78, 76, 71, 70, 73, 77, 79, 80, 101],
[0, 98, 96, 95, 94, 92, 93, 97, 100, 99, 101],
[0, 43, 42, 41, 40, 44, 46, 45, 48, 51, 50, 52, 49, 47, 101],
[0, 101], [0, 13, 17, 18, 19, 15, 16, 14, 12, 101],
[0, 101],
[0, 57, 55, 54, 53, 56, 58, 60, 59, 101],
[0, 32, 33, 31, 35, 37, 38, 39, 36, 34, 101],
[0, 67, 65, 63, 62, 74, 72, 61, 64, 68, 66, 69, 101],
[0, 90, 87, 86, 83, 82, 84, 85, 88, 89, 91, 101],
[0, 20, 24, 25, 27, 29, 30, 28, 26, 23, 22, 21, 101]]
Cplex_Time 83.5109963 Best Cost 827.3
```

PDVRP

```
Vehicles = [[1], [2], [3]]
Routes = [[0, 2, 1, 6, 7, 11],
[0, 3, 8, 11], [0, 5, 4, 10, 9, 11]]
Servicetime = [[0.0, 31.806258416181663, 67.0, 169.0, 199.0, 480.0],
[0.0, 31.806258416181663, 169.0, 480.0],
[0.0, 30.0, 40.0, 166.0, 196.0, 480.0]]
Route Locations = [[Depot, P2, P2, C1, C1, Depot],
[Depot, P2, C1, Depot], [Depot, P1, P1, C2, C2, Depot]]
Loads = [[0.0, 60.0, 30.0, -30.0, -60.0, 0.0], [0.0, 50.0, -50.0, 0.0],
[0.0, 50.0, 60.0, -50.0, -60.0, 0.0]]
Cplex_Time 0.3271329 Best Cost 8445.393954328178
```

References

1. R. Kulkarni and P. Bhave, “Integer programming formulations of vehicle routing problems,”
2. N. Achuthan and L. Caccetta, “Integer linear programming formulation for a vehicle routing problem”
3. G. Desaulniers, J. Desrosiers, A. Erdmann, M. Solomon, and F. Soumis, VRP with Pickup and Delivery
4. I. Harbaoui Dridi, E. Ala¨ıa, P. Borne, and H. Bouchriha, “Optimisation of the multi depots pick-up and delivery problems with time windows and multi-vehicles using pso algorithm
5. C. Tilk, K. Olkis, and S. Irnich, “The last-mile vehicle routing problem with delivery options
6. O. Ali, J.-F. Cˆot´e, and L. Coelho, “The delivery and installation routing problem
7. H. Chu, W. Zhang, P. Bai, and Y. Chen, “Data-driven optimization for last-mile delivery
8. C¸ a˘grı Ko¸c, G. Laporte, and ˙Ilknur T¨ukenmez, “A review of vehicle routing with simultaneous pickup and delivery
9. Konstantinidis, Andreas and Pericleous, Savvas and Charalambous, Christoforos,”Adaptive Evolutionary Algorithm for a Multi-Objective VRP”
10. Shokry, Sherif and Tanaka, Shinji and Nakamura, Fumihiko and Ariyoshi, Ryo and Miura, Shino, “Bandwidth Maximization Approach for Displaced Left-Turn Crossovers Coordination under Heterogeneous Traffic Conditions”

Thank You!