Route Optimization of Alternative Fuel Vehicles in Logistics Operations

STUDENTS / UNIVERSITIES
Berfu Aydın / Sabancı University
Talha Balın / Sabancı University
Abdurrahman Demirli / Sabancı University

SUPERVISOR(S)
Bülent Çatay

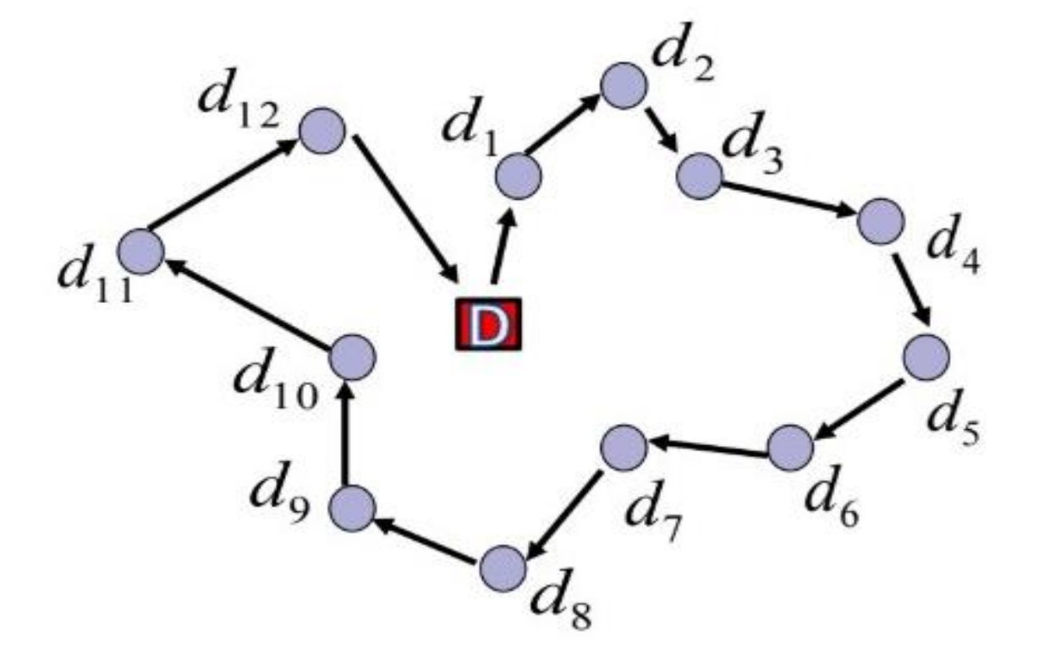
Introduction

What is Route Optimization?

- Route Optimization is the process of determining the **most cost-efficient** route.

What is Traveling Salesman Problem (TSP) ?

- The traveling salesman problem (TSP) is an algorithmic problem tasked with finding the **shortest route** between a set of points and locations that must be visited.



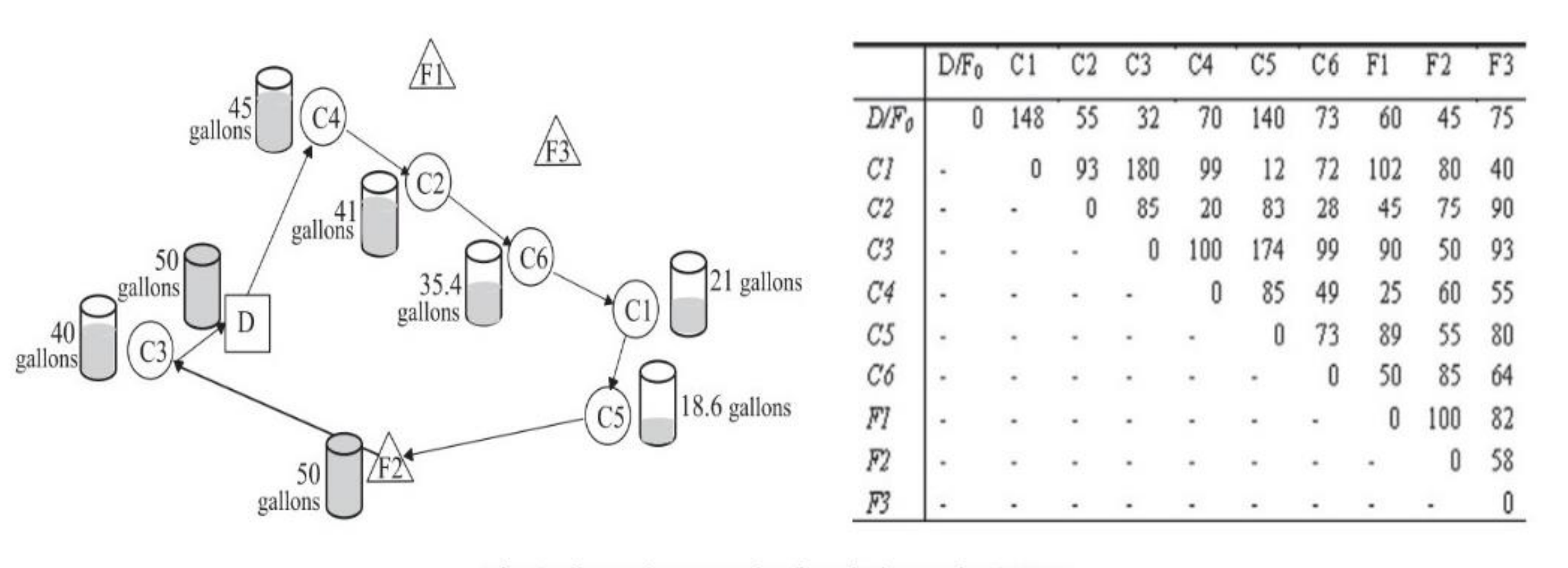
An example of a TSP

What is Vehicle Routing Problem (VRP) ?

- In the Vehicle Routing Problem (VRP), the goal is to find optimal routes for **multiple vehicles** visiting a set of locations. (When there's only one vehicle, it reduces to the Traveling Salesman Problem.)

What is Green Vehicle Routing Problem (GVRP) ?

- In the Green Vehicle Routing Problem (GVRP), the aim is to convert the fleet of service vehicles to include **Alternative Fueled Vehicles** (AFVs) in order to reduce their environmental impact and to meet environmental governmental regulations.



	D/F₀	C1	C2	C3	C4	C5	C6	F1	F2	F3
D/F₀	0	148	55	32	70	140	73	60	45	75
C1	-	0	93	180	99	12	72	102	80	40
C2	-	-	0	85	20	83	28	45	75	90
C3	-	-	-	0	100	174	99	90	50	93
C4	-	-	-	-	0	85	49	25	60	55
C5	-	-	-	-	-	0	73	89	55	80
C6	-	-	-	-	-	-	0	50	85	64
F1	-	-	-	-	-	-	-	0	100	82
F2	-	-	-	-	-	-	-	-	0	58
F3	-	-	-	-	-	-	-	-	-	0

Fig. 1. Illustrative example of a solution to the G-VRP.

Motivation

- In large logistics companies, the complexity of constraints emerge.
  - The increment on the number of customers,
  - The increment on the number of service vehicles,
  - The need of fuel transfer due to the limited range of alternative fueled vehicles (AFVs).
  - Time window of depot's working hours.
- All of these constraints increase the complexity of the particular case.

Thus, it was needed to come up with algorithms to form and optimize particular routes solving vehicle routing problems.

- In order to solve these problems various heuristics have been applied in this project such as Clarke and Wright Savings and Insertion algorithms.

Literature Review

- S. Erdoğan, E. Miller-Hooks (2012) worked on a project of a Green VRP (GVRP) with alternative fueled vehicles servicing to each customers.
- They used a modified Clarke and Wright Savings Heuristics (MCWS) and Density-Based Clustering Algorithm (DBCA).
- In their case and also in this project, several constraints were existent:
- Fuel tank capacity of 60 gallons and fuel consumption rate of 0.2 miles per mile based on average values for biodiesed-powered AFVs (Fraer et al., 2005).
- The average vehicle speed was assumed to be 40 mph and the total tour duration was assumed to be 11 hours.

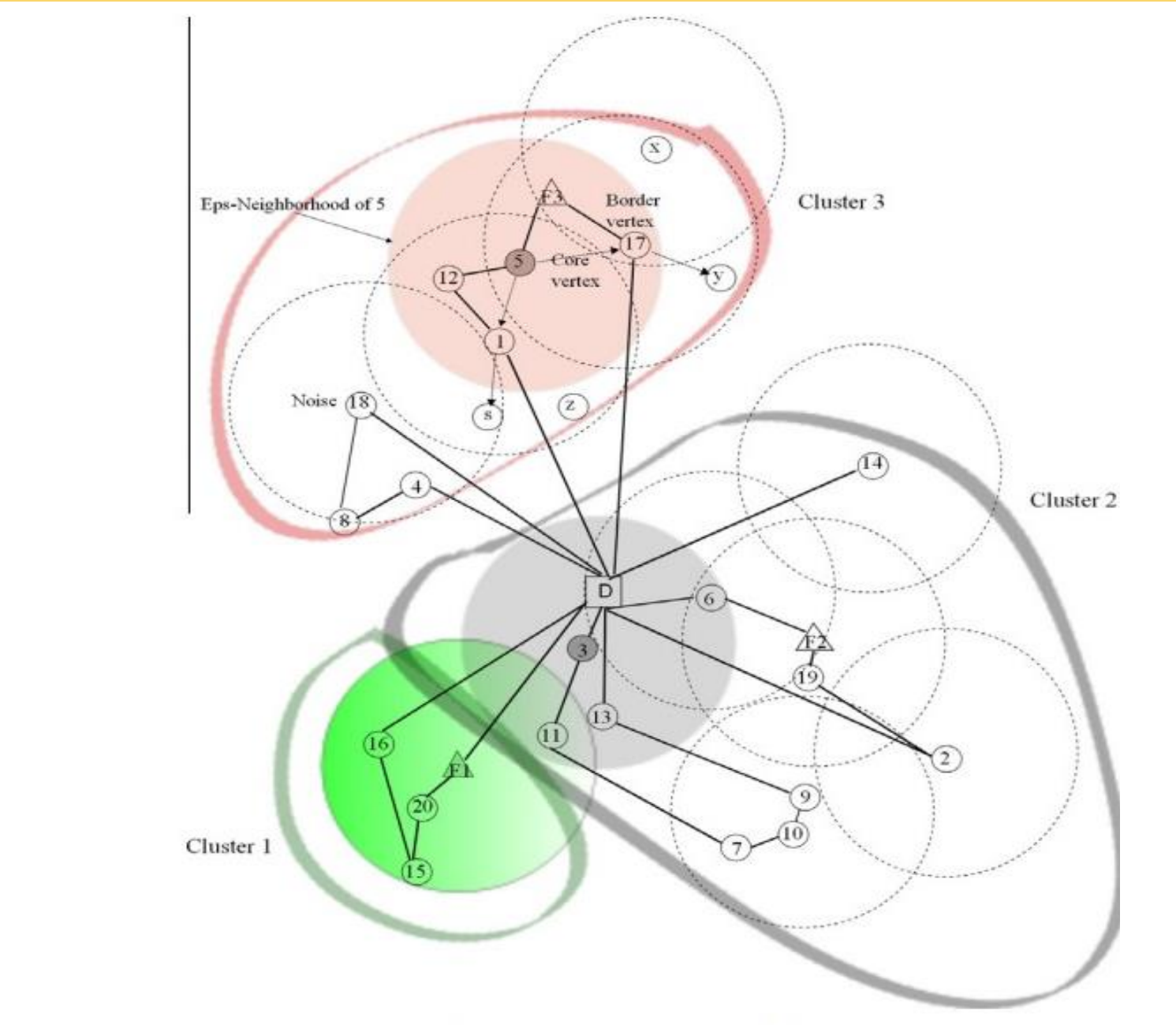


Fig. 4. Forming clusters by DBSCAN Algorithm.

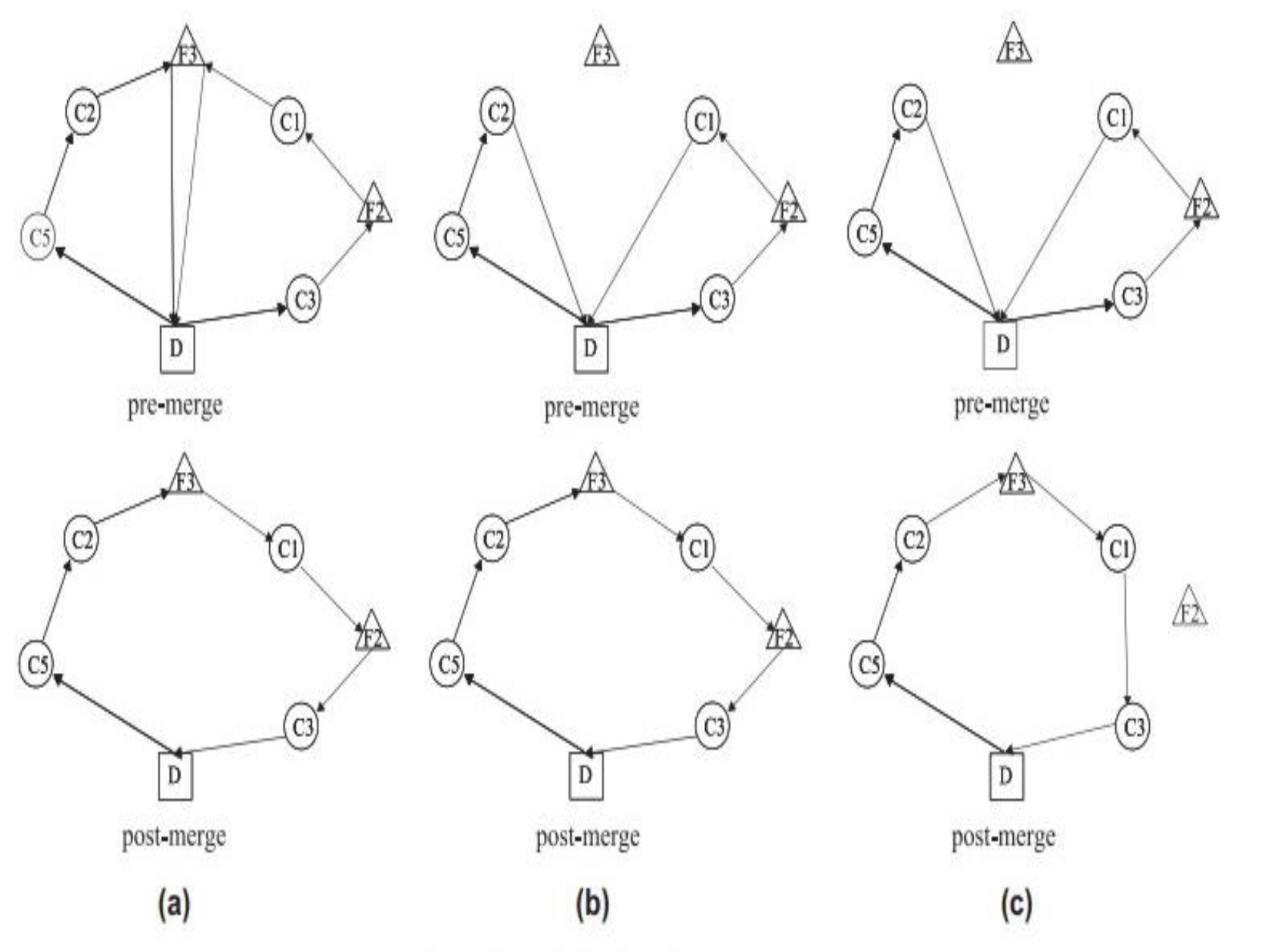
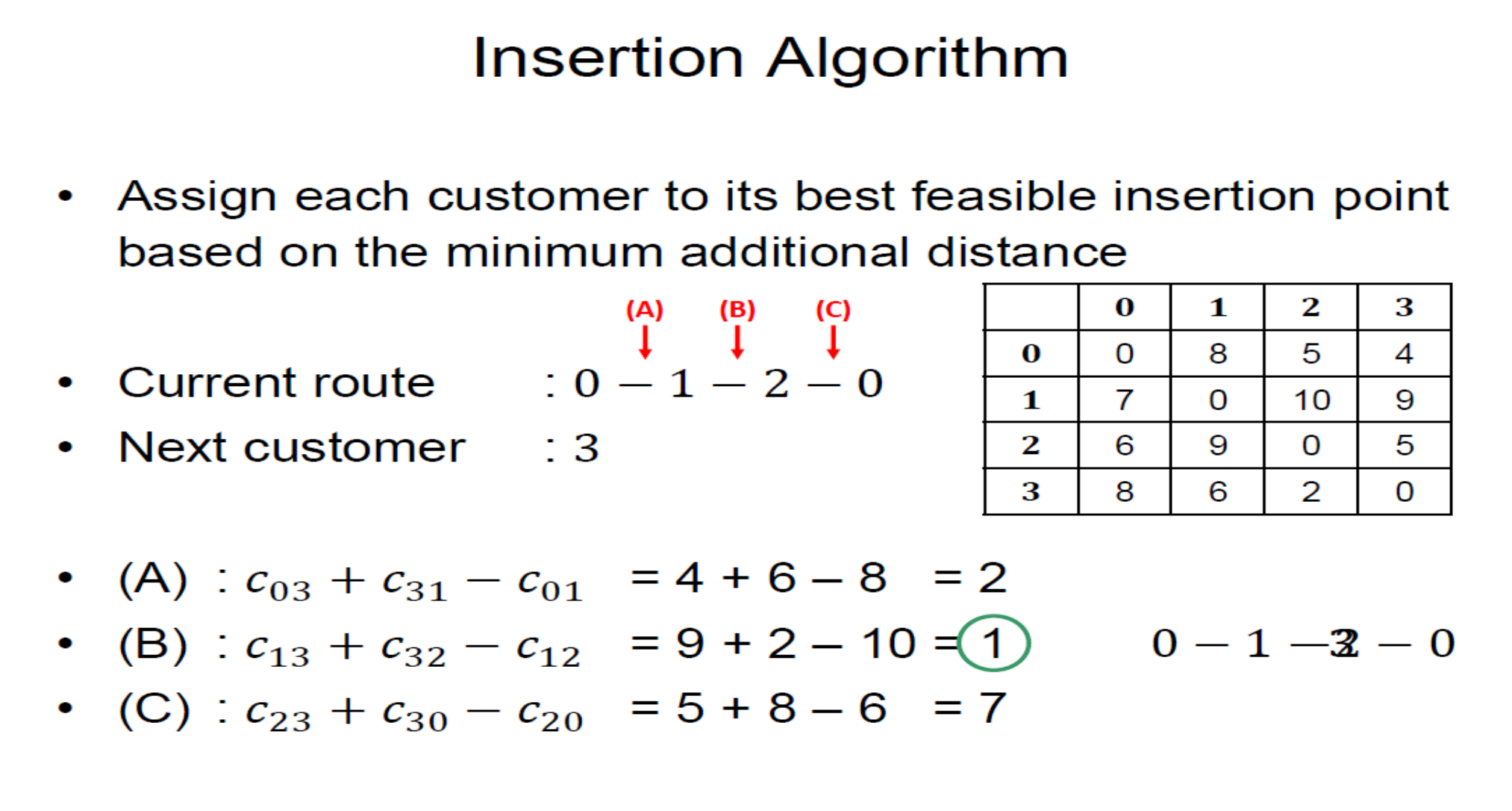


Fig. 3. Characteristics of merging in the G-VRP.

Methodology

Insertion Heuristic Algorithm



Insertion Algorithm

- Assign each customer to its best feasible insertion point based on the minimum additional distance
- Current route : 0 – 1 – 2 – 0
- Next customer : 3

	0	1	2	3
0	0	8	5	4
1	7	0	10	9
2	6	9	0	5
3	8	6	2	0

- (A) : $c_{03} + c_{31} - c_{01}$ = 4 + 6 – 8 = 2
- (B) : $c_{13} + c_{32} - c_{12}$ = 9 + 2 – 10 = ① 0 – 1 – 3 – 2 – 0
- (C) : $c_{23} + c_{30} - c_{20}$ = 5 + 8 – 6 = 7

Step 1 : Start with a sub-graph consisting of node i only.

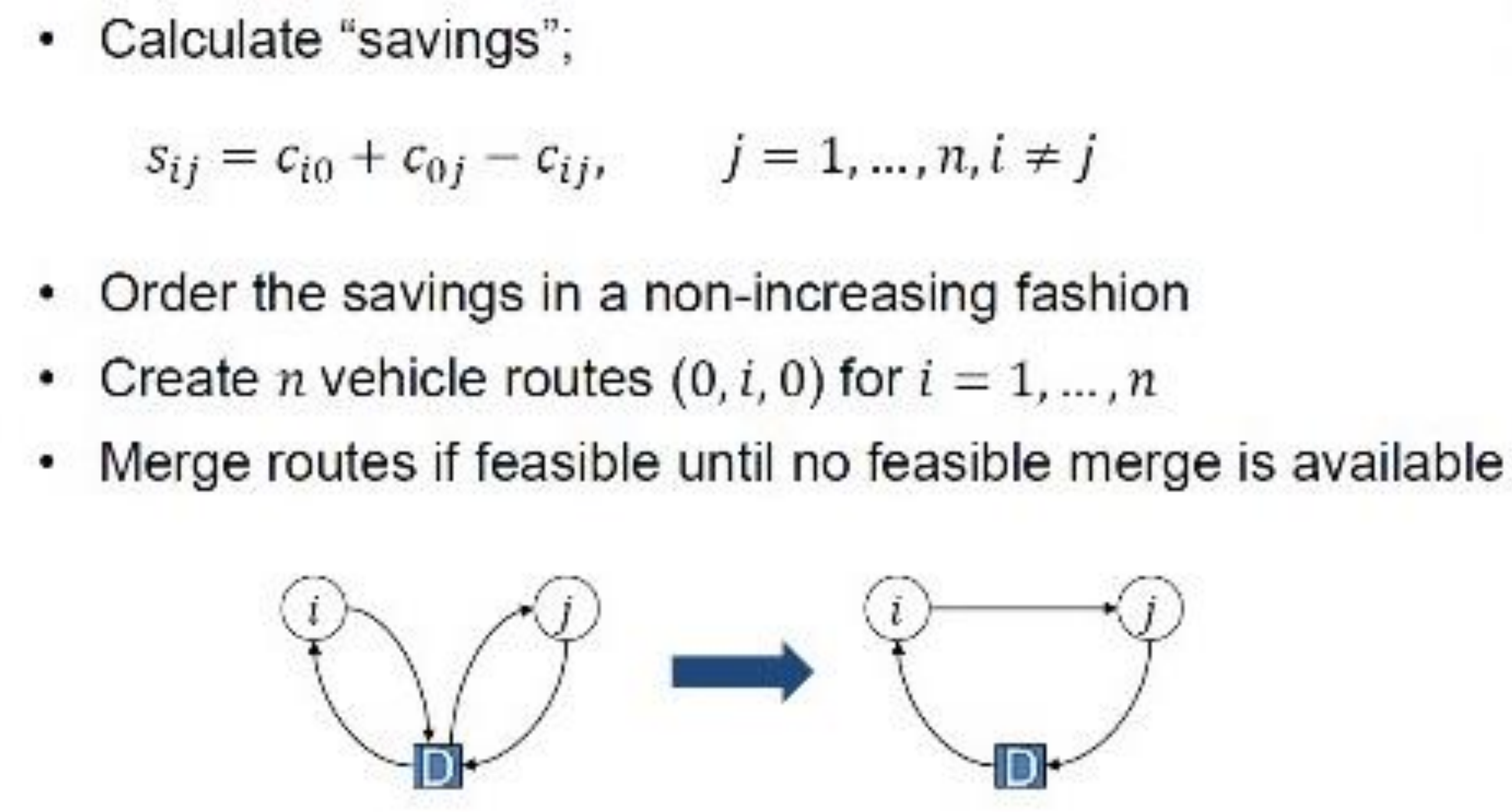
Step 2 : Find node r such that cost is **minimal** and form sub-tour i-r-i.

Step 3 : (Selection Step) Given a sub-tour, find node r not in the sub-tour closest to any node j in the sub-tour; minimal cost

Step 4 : (Insertion Step) Find the arc (i,j) in the sub-tour which minimizes cost. Insert r between i and j .

Step 5 : If all the nodes are added to tour, stop. Else go to step 3.

Clarke and Wright Savings Heuristics

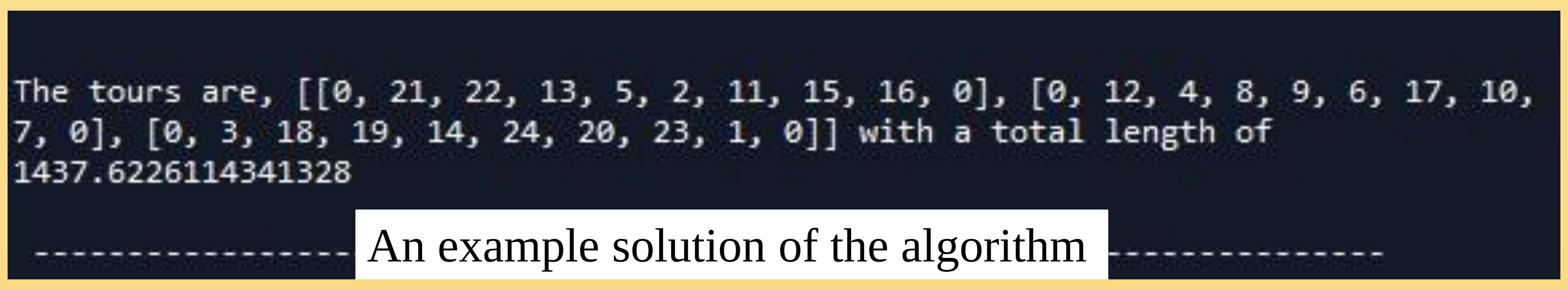


- Calculate "savings";

$$s_{ij} = c_{i0} + c_{0j} - c_{ij}, \qquad j = 1, \dots, n, i \neq j$$

- Order the savings in a non-increasing fashion
- Create n vehicle routes $(0, i, 0)$ for $i = 1, \dots, n$
- Merge routes if feasible until no feasible merge is available

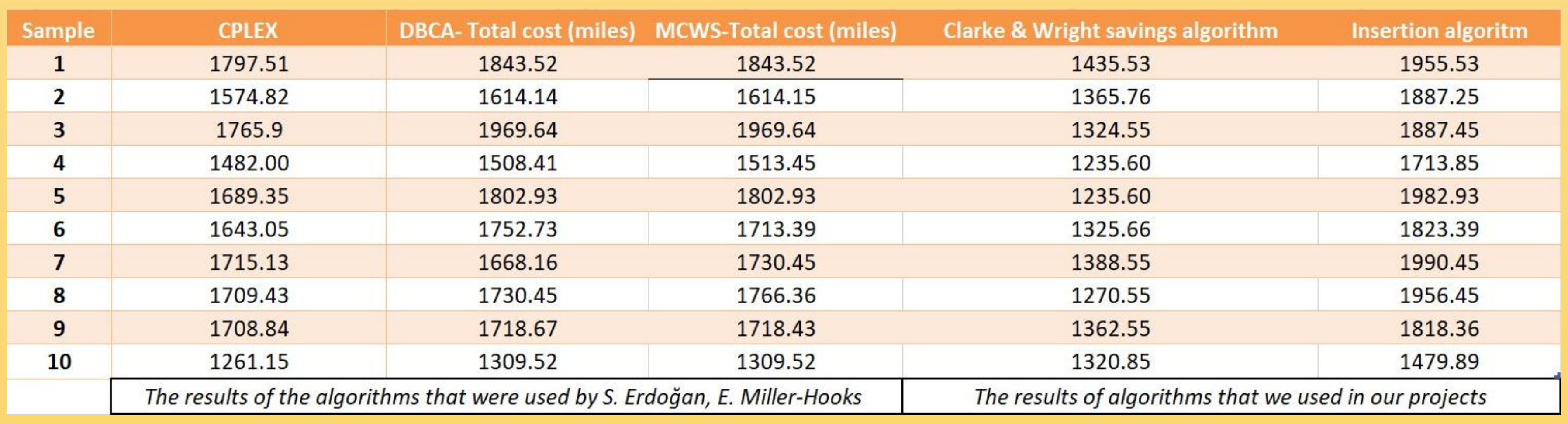
- ***While merging the tours;***
- The ones with more savings from other tour pairs were prioritized.
- the demand of the customers were considered.
- it was controlled that time window of the depot wasn't exceeded.
- The fuel stations were also located, but occasionally may not be working properly.



```
The tours are, [[0, 21, 22, 13, 5, 2, 11, 15, 16, 0], [0, 12, 4, 8, 9, 6, 17, 10,
7, 0], [0, 3, 18, 19, 14, 24, 20, 23, 1, 0]] with a total length of
1437.6226114341328
```

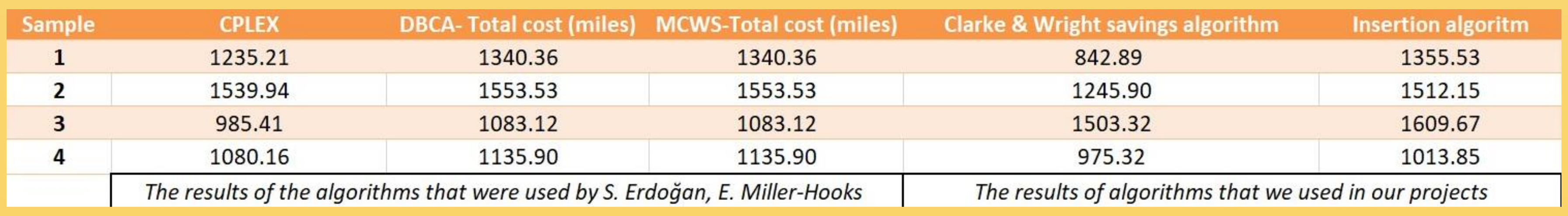
An example solution of the algorithm

Analysis / Results



Sample	CPLEX	DBCA- Total cost (miles)	MCWS-Total cost (miles)	Clarke & Wright savings algorithm	Insertion algoritm
1	1797.51	1843.52	1843.52	1435.53	1955.53
2	1574.82	1614.14	1614.15	1365.76	1887.25
3	1765.9	1969.64	1969.64	1324.55	1887.45
4	1482.00	1508.41	1513.45	1235.60	1713.85
5	1689.35	1802.93	1802.93	1235.60	1982.93
6	1643.05	1752.73	1713.39	1325.66	1823.39
7	1715.13	1668.16	1730.45	1388.55	1990.45
8	1709.43	1730.45	1766.36	1270.55	1956.45
9	1708.84	1718.67	1718.43	1362.55	1818.36
10	1261.15	1309.52	1309.52	1320.85	1479.89
	The results of the algorithms that were used by S. Erdoğan, E. Miller-Hooks			*The results of algorithms that we used in our projects*	

Data-1 from table-2 Results Comparison



Sample	CPLEX	DBCA- Total cost (miles)	MCWS-Total cost (miles)	Clarke & Wright savings algorithm	Insertion algoritm
1	1235.21	1340.36	1340.36	842.89	1355.53
2	1539.94	1553.53	1553.53	1245.90	1512.15
3	985.41	1083.12	1083.12	1503.32	1609.67
4	1080.16	1135.90	1135.90	975.32	1013.85
	The results of the algorithms that were used by S. Erdoğan, E. Miller-Hooks			*The results of algorithms that we used in our projects*	

Data-2 from table-3 Results Comparison

- Same data sets were applied in all algorithms.
- The results by the S. Erdoğan, E. Miller-Hooks are much higher than the algorithms that were used in this project (especially in the Clarke & Wright Savings) this is, in general, because of the fuel nodes that were located in this project doesn't meet the driving range of the service always properly.
- (For this poster we couldn't manage to finish the algorithm with fuel stations meeting fuel demands properly, so, see our paper…)