IEE 574 – Applied Deterministic Operation Research

Project Report

on

Drone Schedule & Routing

By:

Anirudh Reddy Goluguri (1213278051)

Harish Siva Subramanian (1213355570)

Arvind Ramkumar (1213469749)

Introduction:

As the problem states that we need to minimize the drone scheduling costs and also minimize the transportation cost of the drone. These drones are used to deliver the vaccine to the hospital in the cities nearby the CDC lab located in Philadelphia, PY for different hour and different demand requirements according to the population in those cities. As the vaccines cannot be stored they are manufactured according to the demand specification of the hospital in different cities near the CDC lab in Philadelphia, PY. As we have two type of drones: type I drone of capacity 450 and battery life of 3 hours and type II drones with capacity of 1000 and battery life of 2 hours. Also, each drone must start from the CDC lab and should return back to the CDC lab at last the problem was split into two folds.

First, we tried to find the optimal number of drones (which is considered as the upper bound) that are required to deliver the vaccines to the hospital according to their demand in different hour and our objective is to minimize the cost of number of drones we require.

Secondly, efforts were made to obtain the optimal routes between the CDC labs and the hospitals in different cities where the CDC lab serves the vaccines. Four models were employed to find the best route with minimizing the total distance travelled by the drones.

Assumptions:

1. The cities were chosen that had a minimum population of 5000.
2. The cities chosen where within a radius of 25 miles from the CDC lab.
3. To meet the minimum requirement of the demand of the cities, the demand was ROUNDED UP.
4. A real-time geo-distance was considered to develop the distance matrix.
5. In mathematical formulation and dynamic programming we have assumed that for type I drone battery life drains completely in three hours even if it is not utilized and type II drone's battery life drains completely in two hours even if it is not utilized.

Minimizing the Drone- Scheduling Cost

Data Collection & Demand Calculation:

The population data of the cities was used to estimate the demand for each period of operation using the formula,

$$d_{ij} = p_i * 35 * \log_2(Pop_j)$$

Here p_i is obtained from the Equation,

$$p_i = 1 - e^{-1/i}$$

Table 1: Demand Calculation for each Period

Period 1	Period 2	Period 3	Period 4
289	180	130	101
297	185	133	104
288	179	129	101
315	196	141	110
318	198	143	112
304	190	137	107
302	188	136	106
313	195	141	110
334	208	150	117
323	201	145	113
283	177	127	99
327	204	147	115
293	183	132	103
279	174	125	98
281	175	126	99
311	194	140	109
334	208	150	117
277	172	124	97
297	185	133	104
317	197	142	111
6082	3789	2731	2133

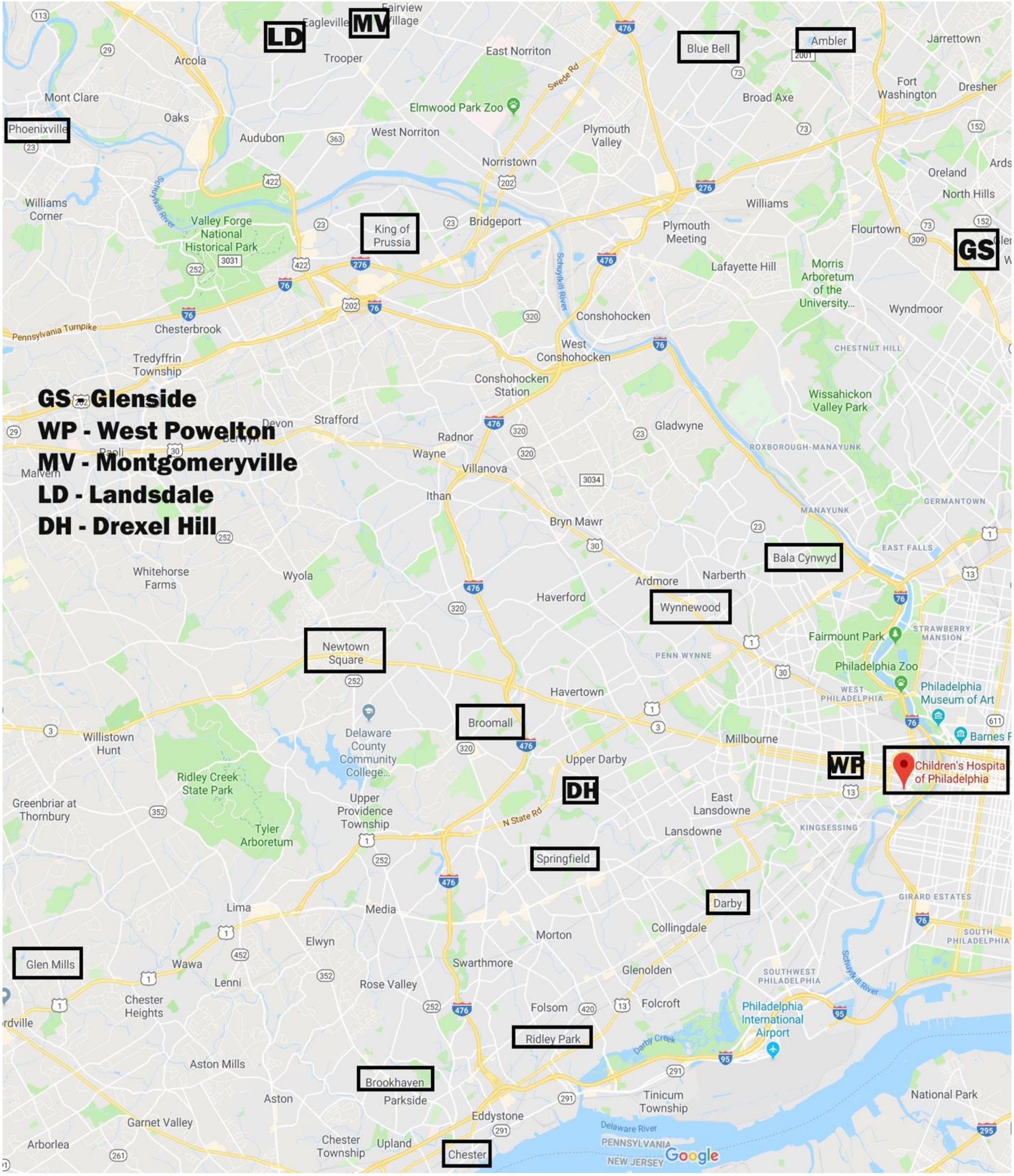


Figure 1: Location of the cities chosen

The demand data calculated using the formula is shown in Table 1. Using the above demand data calculated using the formula, a mathematical program was formulated to minimize the Drone Scheduling Cost. The cost, capacity, battery life and the minimum number to schedule is shown in Table 2.

Table 2: Cost, Capacity, Battery Life and Minimum # to Schedule for each Drone Types

Drone	**Capacity (Vaccine Units)**	**Cost (in $1000)**	**Battery Life**	**Min # to Schedule**
Type I	450	350	3 hours	3
Type II	1000	550	2 hours	2

Objective Function:

Minimize: $\sum_{i=1}^{2} \quad \sum_{j=1}^{4} \quad c_{ij}x_{ij}$

Here x_{ij} is the Decision Variable which is the Number of Drones of Type i that starts working in period j.

C_{ij} is the Cost associated by operating drone type i at period j (350 or 1000)

Constraints:

$\sum_{j=1}^{4} \quad x_{1j} \geq 3$ (Minimum number of Drones of Type I summed over all periods)

$\sum_{j=1}^{4} \quad x_{2j} \geq 2$ (Minimum number of Drones of Type II summed over all periods)

$450x_{11} + 1000x_{21} >= 6082$ (Required Demand Constraint for Period 1)

$450(x_{11} + x_{12}) + 1000(x_{21} + x_{22}) >= 3789$ (Required Demand Constraint for Period 2)

$450(x_{11} + x_{12} + x_{13}) + 1000(x_{22} + x_{23}) >= 2731$ (Required Demand Constraint for Period 3)

$450(x_{12} + x_{13} + x_{14}) + 1000(x_{23} + x_{24}) >= 2133$ (Required Demand Constraint for Period 4)

$x_{ij} \geq 0$ (Non-Negativity Constraint)

Note: The above "Required Demand Constraints" were formulated based on the fact that Type I drones can operate only for 3 hours and that every x_{1j} term will occur once in three consecutive constraints which indicates 3 hours operation. Similarly, Type II drones have a battery life of 2 hours and every x_{2j} term will occur once in two consecutive constraints which indicates 2 hours operation

On solving the above Mathematical program in AMPL, we get an objective function value of $5250 k which is the Scheduling cost for the drones.

```
ampl: display x;
x :=
1 1   3
1 2   0
1 3   1
1 4   0
2 1   5
2 2   0
2 3   2
2 4   0
;
```

Figure 2: Number of Drones

Figure 2 says that the number of drones of Type I which started at period 1 is 3. Similarly, the number of drones of Type II which started at period 2 is 0.

Table 3: Results obtained

Type	Starting Period	Number of Drones	Scheduling Cost (in $ 1000)
1	1	3	1050
1	2	0	0
1	3	1	350
1	4	0	0
2	1	5	2750
2	2	0	0
2	3	2	1100
2	4	0	0
Total Scheduling Cost			5250

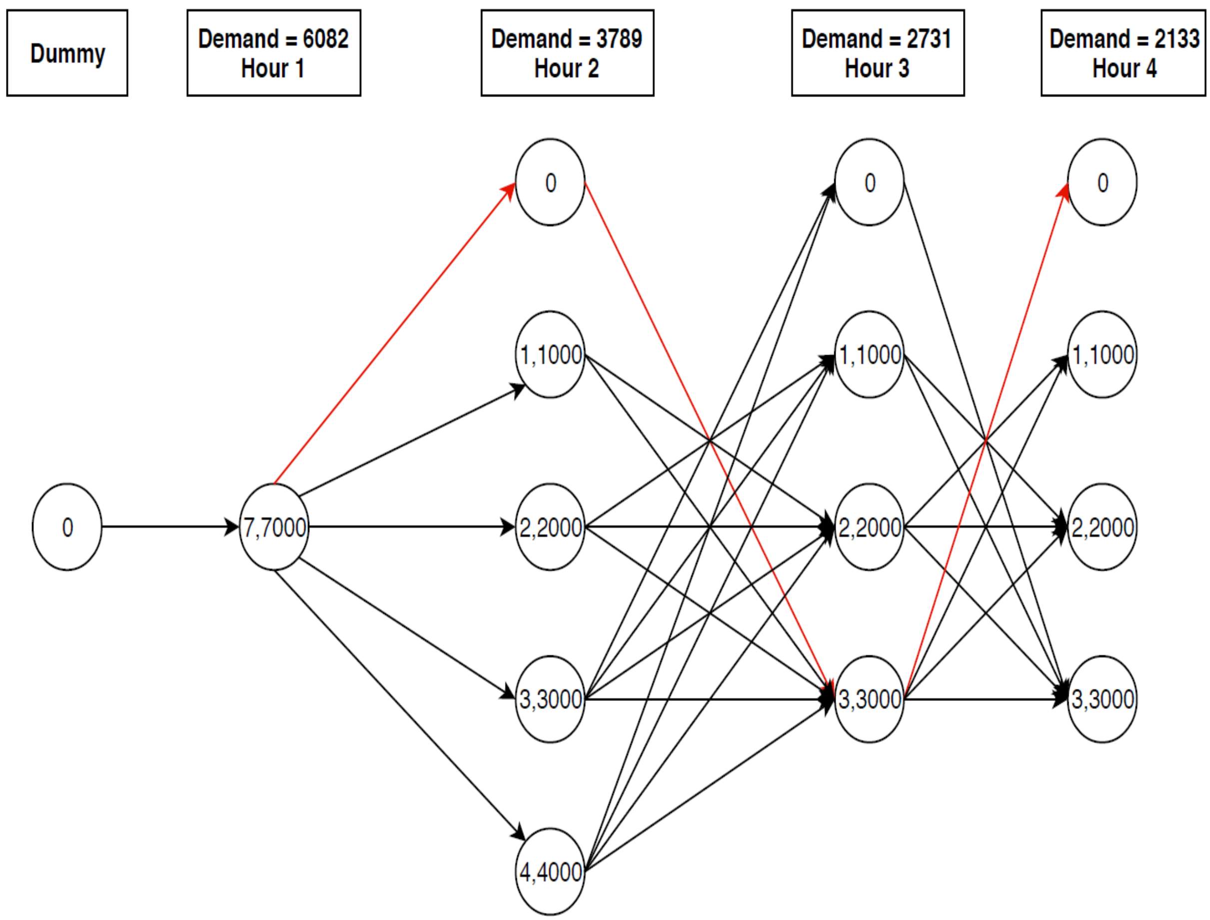
Here, Table 3 shows a clear representation of the numbers taken from the above image. Here the Scheduled Cost associated with each drone is shown and it is calculated using the objective function equation.

Dynamic Programming:

For the dynamic programming the we have assumed that the batteries of the drones will drain even if the drones are idle. It is considered here that the hours as stages and number of drones that are with one hour of battery life left at the end of the period as states. Now in this as the at the state one the demand is 6082 we need seven drones to meet the demand and not less than that. For easier representation we have not showed 1 to 6. While in the stage 2 we can use the drone that did start in the stage 1 and we can also add the 1, 2, 3 or 4 new drones to satisfy the demand 3789. In stage 3 the demand is 2731 so we can add 1, 2, 3 as maximum of three drones are required to satisfy the demand. We can add 1 new drone and utilize the drones with 1 hour battery life left that are employed in the hour 2 and it is similar to the other drone.

First we calculate the minimum drone that are required at hour 4 and then we use these drones to find the minimum drones that are required at the each stage by backtracking using recursive method. Below, we have showed the recursive equations, objective function and the drones that are added at each stage for the type drone 2.

Results:



Type – II Drone:

Recursive Equation

$$f_t(s,v) = \min_{1000(x_t + s) \geq \text{Demand}} \{x_t + f_{t+1}(s,v)\}$$

Constraint:

$1000(x_t + s) \geq \text{Demand}$

Objective Function

Min. $f_0(0)$

Stages: The Time Period (Stage = 1, 2, 3, 4)

States:

S - Number of Drones with remaining battery life 1 hour at the end of the stage.

V – Number of Vaccines Loaded

Stage 4

$f_4(0) = 0$

$f_4(1) = 1$

$f_4(2) = 2$

$f_4(3) = 3$

Stage 3

$f_3(0) = \min \{0 + f_4(3)\} = \min \{0+3\} = 3$

$f_3(1) = \min \{1 + f_4(2), 1 + f_4(3)\} = \min \{3, 4\} = 3$

f3(2) = min {2 + f4(1), 2 + f4(2), 2 + f4(3)} = min {3, 4, 5} = 3

f3(3) = min {3 + f4(0), 3 + f4(1), 3 + f4(2), 3 + f4(3)} = min {3, 4, 5, 6} = 3

Stage 2

$f_2(0) = \min \{0 + f_3(3)\} = \min \{0 + 3\} = 3$

$f_2(1) = \min \{1 + f_3(2), 1 + f_3(3)\} = \min \{4, 4\} = 4$

$f_2(2) = \min \{2 + f_3(1), 2 + f_3(2), 2 + f_3(3)\} = \min \{5, 5, 5\} = 5$

$f_2(3) = \min \{3 + f_3(0), 3 + f_3(1), 3 + f_3(2), 3 + f_3(3)\} = \min \{6, 6, 6, 6\} = 6$

$f_2(4) = \min \{4 + f_3(0), 4 + f_3(1), 4 + f_3(2), 4 + f_3(3)\} = \min \{7, 7, 7, 7\} = 7$

Stage 1

$f_1(7) = \min \{7 + f_2(0), 7 + f_2(1), 7 + f_2(2), 7 + f_2(3), 7 + f_2(4)\} = \min \{10, 11, 12, 13, 14\} = 10$

Stage 0

$f_0(0) = f_1(7) = 10$

Optimal Requirements

Stage 1 – 7 Drones are added

Stage 2 – 0 Drones are added

Stage 3 – 3 Drones are added

Stage 4 – 0 Drones are added

Total number of drones = 10

Objective Function = 10

Overall Cost of using these drones = 10*550 = $ 5500K

Type – I Drone

$$ft(s_1,s_{2,}\,v) = \min_{1000(xt\,+\,s1+s2)\,\geq\,\text{Demand}} \{xt + f_{t+1}(s_1,s_{2,}\,v)\}$$

Objective function

Min. $f_0(0)$

Constraints:

$1000(x_t + s1 + s2) \geq$ Demand

Stages: The Time Period (Stage = 1, 2, 3, 4)

States:

S_1 - Number of Drones with remaining battery life 1 hour at the end of the stage.

S_2 - Number of Drones with remaining battery life 2 hour at the end of the stage.

V – Number of Vaccines Loaded

Minimize Transportation Distance:

Distance Matrix

For calculating the distance between two cities, we used R software to create a function and a loop to create a distance matrix. We took the real time geographical distance between two cities considering the latitudes and longitudes. We also verified the distance by manually checking the distance between two cities in the Internet using the Latitudes and longitudes. Also the distance specified in the Distance matrix is in Miles.

One more important thing to note is that, while creating the distance matrix we obtained values of 0 for distance between CDC to CDC and for distances between the same cities which is obviously true. In order for the AMPL to avoid taking the route from CDC to CDC or any other route where from and to cities are same, we gave a high value (10000) in those places. The distance matrix is provided in the “Initial” Excel file.

Table 4: Representation of Nodes for the city

Nodes Represented	City Names
1	CDC Lab
2	Glenside
3	Darby
4	Brookhaven
5	Glen Mills
6	Newton Square
7	Wynnewood
8	Montgomeryville
9	West Powelton
10	Chester
11	Springfield
12	Ridley Park
13	Drexel Hill
14	Bala Cynwyd
15	Blue Bell
16	Ambler
17	Phoenixville
18	Norristown
19	Lansdale
20	Broomall
21	King of prussia

Models Used:

A. Exact Formulation:

We worked on minimizing the total distance traveled by the drones between the cities from the CDC lab and returns back to the CDC lab after satisfying the demands of all cities for the second hour. In this we tried to eliminate the sub tours between the cities by using two types of formulation. They are 1. MTZ type constraints and 2. Sub tour elimination constraints.

Decision Variables:

$$x_{ijk}$$

Where

$$x_{ijk} = \begin{cases} 1 \; if \; a \; drone \; k \; visits \; city \; i \; from \; city \; j \\ 0 \; if \; a \; drone \; k \; doesnot \; visit \; city \; i \; from \; j \end{cases}$$

i = from which city the drone is departed = 1…21

Here 1 = CDC lab and 2…21 = The 20 hospitals whose demand needs to be satisfied.

j = City visited by the drone from the city I to satisfy its demand = 1…21

k = Drone used to visit from city i to city j in hour 2 = 1…8

Here the Total number of drone that are available for utilization in hour two are 1…3 type I drones and 1…5 type two drones.

Objective Function:

$$\text{Minimize Z: } x_{ijk} * distance_{ij}$$

Where,

$distance_{ij}$ = Distance between city i to city j

1. MTZ – type constraints:

We have set of restriction which we need to include and they are as follows:

1. A drone must start from the CDC lab and return back to the CDC lab which means that if a type 1 drone leaves from CDC lab it can go to as many cities as it can and at last it should return to the CDC lab (ex: CDC→ Wynnewood → Glen Mills → Newton Square → CDC Lab).
2. A drone must visit a hospital at most once and each hospital must be visited exactly once which means that if demand of hospital in Glen mills should be satisfied only one drone can visit the hospital with the demand no more than one drone can visit the each hospital.

3. The demand assigned to each drone must not exceed the capacity of each drone.

Decision Variables:

1.

$$x_{ijk}$$

Where i = From which city the drone is departed = 1…21

Here 1 = CDC lab and 2…21 = The 20 hospitals whose demand needs to be satisfied.

j = City visited by the drone from the city I to satisfy its demand = 1…21

k = Drone used to visit from city i to city j in hour 2 = 1…8

here the Total number of drone that are available for utilization in hour two are 1…3 type I drones and 1…5 type two drones.

2. MTZ variable

$$u_i \geq 0 \qquad \forall\, i = 2 \ldots 21$$

a. Constraints:

1. From city i → city j can be visited only once and by only one drones.

$$\sum_{i=1}^{21} \sum_{k=1}^{8} x_{ijk} = 1 \qquad \forall j = 2 \ldots 21$$

2. From city j → city i can be visited only once and by only one drone.

$$\sum_{j=1}^{21} \sum_{k=1}^{8} x_{ijk} = 1 \qquad \forall i = 2 \ldots 21$$

   Above two equations where used to constrain the problem by saying that hospital in each city must be visited by only one drone and at most only once a city can be visited.

3. Demand Satisfying Constraint:

$$\sum_{i=1}^{21} \sum_{j=1}^{21} x_{ijk} * Demand_j \leq Capacity_k \quad \forall\, k = 1 \ldots 8$$

   As we have 8 drone available where 3 drones have a capacity of 450 and 5 drones have capacity of carrying the 1000 vaccines.

   For every tour a drone takes, the total number of vaccines should not exceed the total capacity of the drone and the above constraint ensures the same.

4. Each drone can fly to one or more than one city from the CDC Lab.
   Since we have the requirement that each drone must start flying from the CDC lab to different hospital in different cities.

$$\sum_{j=2}^{21} x_{1jk} \leq 1 \quad \forall\ k = 1 \ldots 8$$

5. Each drone comes back to the CDC lab from maximum of one city.

$$\sum_{i=2}^{21} x_{1jk} \leq 1 \quad \forall\, k = 1 \dots 8$$

6.

$$\sum_{i=1}^{21} x_{irk} = \sum_{j=1}^{21} x_{rjk} \quad \forall\, k = 1 \dots 8, \forall\, r = 2 \dots 21$$

The above constraint is used to form a route between the hospitals in different cities. The function of this constraint is it make a route such as if a drone starts in CDC → Ambler then the next city the drone must visit should start from the Ambler → Brookhaven.

b. Description of Solution Methods:

7. MTZ – type Constraints:

$$u_i - u_j + N \sum_{k=1}^{8} x_{ijk} \leq N - 1 \qquad \forall \quad i = 2 \dots 21, \forall\, j = 2 \dots 21, \qquad i \neq j, \qquad N = 20$$

$$u_i - u_j + 20 \sum_{k=1}^{8} x_{ijk} \leq 19 \qquad \forall \quad i = 2 \dots 21, \forall\, j = 2 \dots 21, \qquad i \neq j, \qquad N = 20$$

This method is used to eliminate the subtour that are generated without involving the CDC labs in between the remaining cities. Subtour elimination method of MTZ is used because if subtour elimination was not utilized then the solutions we obtained will have the route which may not involve the CDC labs and the having the tours in between the cities like tour starts from Glen mills → Wynnewood → Newton Square and ends at the newton square which instead should be like this CDC → Glen mills → Wynnewood → Newton Square → CDC and this is one of the possible subtour which may happen and there will be different types of subtours that might happen and the route we obtain does not satisfy the requirements.

Here u_i is a variable and this has $(n-1)^2 = (20-1)^2 = 361$ constraints.

c. Results:

Initially, we tried to solve the Formulation with in an indefinite time limit, but because of the computational constraints, the problem could not converge to a result, yielding a memory error. So, we did impose the time limit to the computation. The time limit given was 3 hours (10800 Secs) and a feasible solution was obtained which is 19.95% away from the optimal solution. The corresponding statistics with the routes and results are displayed below.

Objective Value: 150.1007368
MIP simplex iterations: 97081054
Branch-and-bound nodes: 1929891
absmipgap: 29.9563
relmipgap: 0.199575
Route Obtained:
Drone 1: Nil
Drone 2: Nil
Drone 3: Nil
Drone 4:
CDC → Springfield → Glen mills → Brookhaven → Chester → Ridley Park → CDC
Drone 5:
CDC → Ambler → Blue Bell → Lansdale → Montgomeryville → Glenside → CDC
Drone 6:
CDC → Drexel Hill → Broomall → Wynnewood → Bala Cynwyd → West Powelton → CDC
Drone 7: Nil
Drone 8:
CDC → Darby → Norristown → Phoenixville → King of Prussia → Newton Square → CDC

From this we could interpret that the objective value can have a slight improvement as there is a relative gap of the 19.95 % and this also increases MIP iterations and branch and bounds nodes.

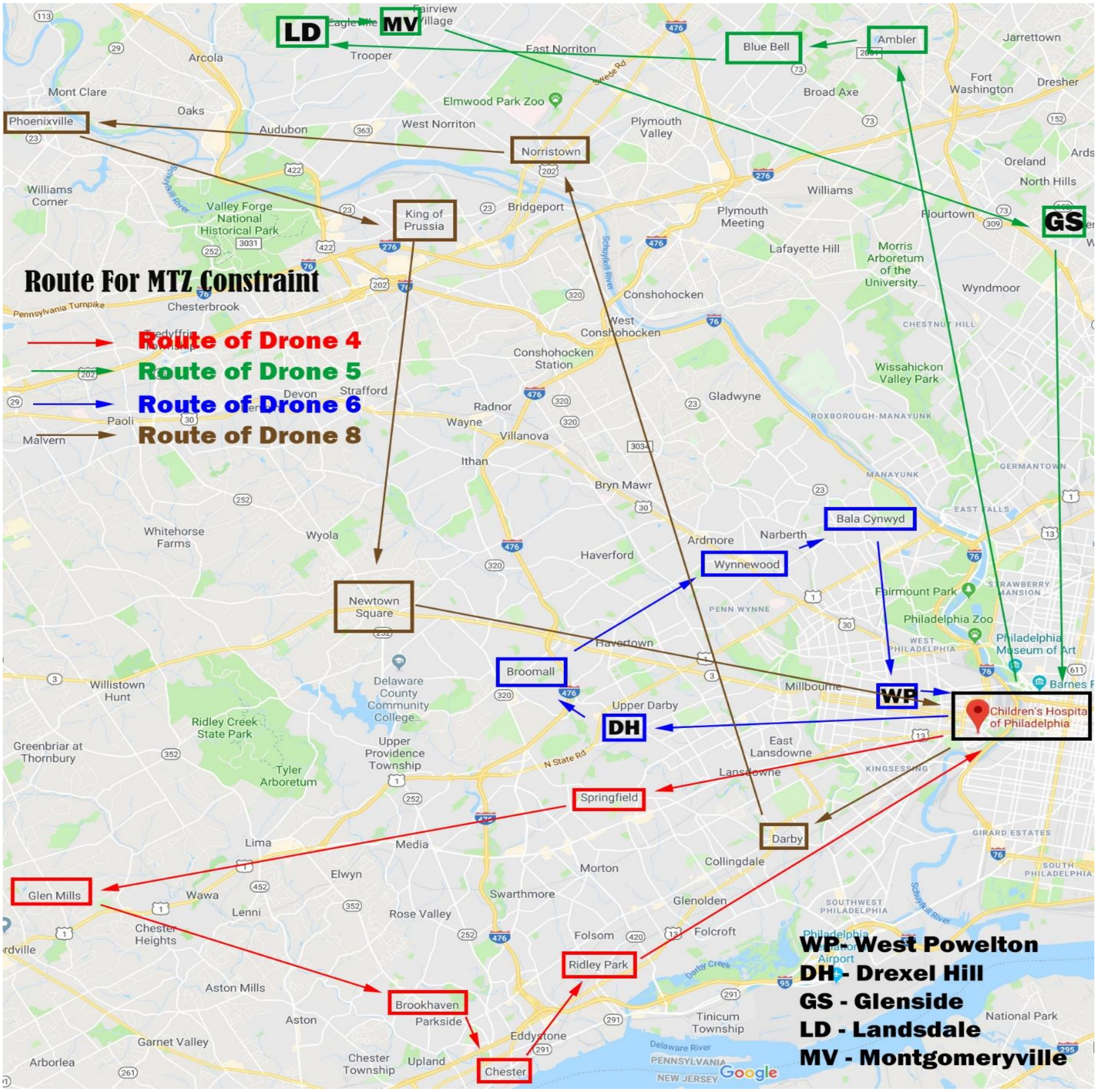


Figure 3: Route for MTZ Constraint

<u>Drone 4:</u> CDC → Springfield → Glen mills → Brookhaven → Chester → Ridley Park → CDC
<u>Drone 5:</u> CDC → Ambler → Blue Bell → Lansdale → Montgomeryville → Glenside → CDC
<u>Drone 6:</u> CDC → Drexel Hill → Broomall → Wynnewood → Bala Cynwyd → West Powelton → CDC
<u>Drone 8:</u> CDC → Darby → Norristown → Phoenixville → King of Prussia → Newton Square → CDC

2. Sub Tour Eliminations:

a. Constraints:

1. From city i → city j can be visited only once and by only one drone.

$$\sum_{i=1}^{21} \sum_{k=1}^{8} x_{ijk} = 1 \qquad \forall j = 2 \dots 21$$

2. From city j → city i can be visited only once and by only one drone.

$$\sum_{j=1}^{21} \sum_{k=1}^{8} x_{ijk} = 1 \qquad \forall i = 2 \dots 21$$

Above two equations where used to constrain the problem by saying that hospital in each city must be visited by only one drone and at most only once a city can be visited.

3. Demand Satisfying Constraint:

$$\sum_{i=1}^{21} \sum_{j=1}^{21} x_{ijk} * Demand_j \leq Capacity_k \quad \forall\, k = 1 \dots 8$$

As we have 8 drone available where 3 drones have a capacity of 450 and 5 drones have capacity of carrying the 1000 vaccines.

For every tour a drone takes, the total number of vaccines should not exceed the total capacity of the drone and the above constraint ensures the same.

4. Each drone can fly to one or more than one city from the CDC Lab.

Since we have the requirement that each drone must start flying from the CDC lab to different hospital in different cities.

$$\sum_{j=2}^{21} x_{1jk} \leq 1 \quad \forall\ k = 1 \dots 8$$

5. Each drone comes back to the CDC lab from maximum of one city.

$$\sum_{i=2}^{21} x_{1jk} \leq 1 \quad \forall\, k = 1 \dots 8$$

6.

$$\sum_{i=1}^{21} x_{irk} = \sum_{j=1}^{21} x_{rjk} \quad \forall\, k = 1 \dots 8, \forall\, r = 2 \dots 21$$

The above constraint is used to form a route between the hospitals in different cities. The function of this constraint is it make a route such as if a drone starts in CDC → Ambler then the next city the drone must visit should start from the Ambler → Brookhaven.

7. Sub tour elimination constraint:

The subtour elimination constraint was built in order to eliminate the subtours that does not involve the CDC lab. This constraint works in order to limit the route in between cities in different set of cities. Here in this constraint the subsets are created with elements of 2 to 18 as we need to eliminate the subtours between the city in between this constraint and this constraint says that the number of routes in between the subset should the less than the number of elements in subsets minus 1.

$$\sum_{i\varepsilon S} \sum_{j\in S} x_{ijk} \leq |S| - 1, \ \forall \ \ 2 \leq |S| \leq 18$$

In general this constraint builds $2^n = 2^{20} = 1048576$.

b. Description of Solution Methods:

In general this constraint builds $2^n = 2^{20} = 1048576$. Owing to the exponentially increasing number of constraints there were lot of computational issues. So from the inferences obtained from the MTZ Constraint solution and the logic reasoning, we can see that no drone can visit more than five or six cities. So we are reducing the number of constraint for making the problem computationally efficient.

Now, we reduce the limit of the set, i.e., $2 \leq |S| \leq 6$. Now, we will have $2^n = 2^8 = 256$ constraints which makes the computation more easy and effective. By using the above constraints model a mod file and distance matrix and other data was modeled as ".dat" file and executed in AMPL with CPLEX solver and we ran the code for a time limit of 14400 secs which is 4hrs we obtained the results as follow:

c. Results

Initially, we tried to solve the Formulation with in an indefinite time limit, but because of the computational constraints, the problem could not converge to a result, yielding a memory error. So, we did impose the time limit to the computation. The time limit given was 4 hours (14400 Secs) and a feasible solution was obtained which is 32.98% away from the optimal solution. The corresponding statistics with the routes and results are displayed below.

Objective Value: Minimize Total Distance Traveled: 153.776037
MIP Simplex Iterations: 984275
Branch and Bound Nodes: 55011
Absmipgap: 50.7255
Relmipgap: 0.329866
Routes Obtained:
Drone 1: Nil
Drone 2: Nil
Drone 3: Nil
Drone 4: Nil
Drone 5:
CDC → Springfield → Glen Mills → Brookhaven → Chester → Ridley Park → CDC
Drone 6:
CDC → Glenside → Montgomeryville → Lansdale → Blue bell → Ambler → CDC
Drone 7:
CDC →Bala Cynwyd →Norristown→Phoenixville→King of Prussia→ Darby→ CDC
Drone 8:
CDC →West Powelton →Wynnewood →Newton Square →Broomall → Drexel Hill→ CDC

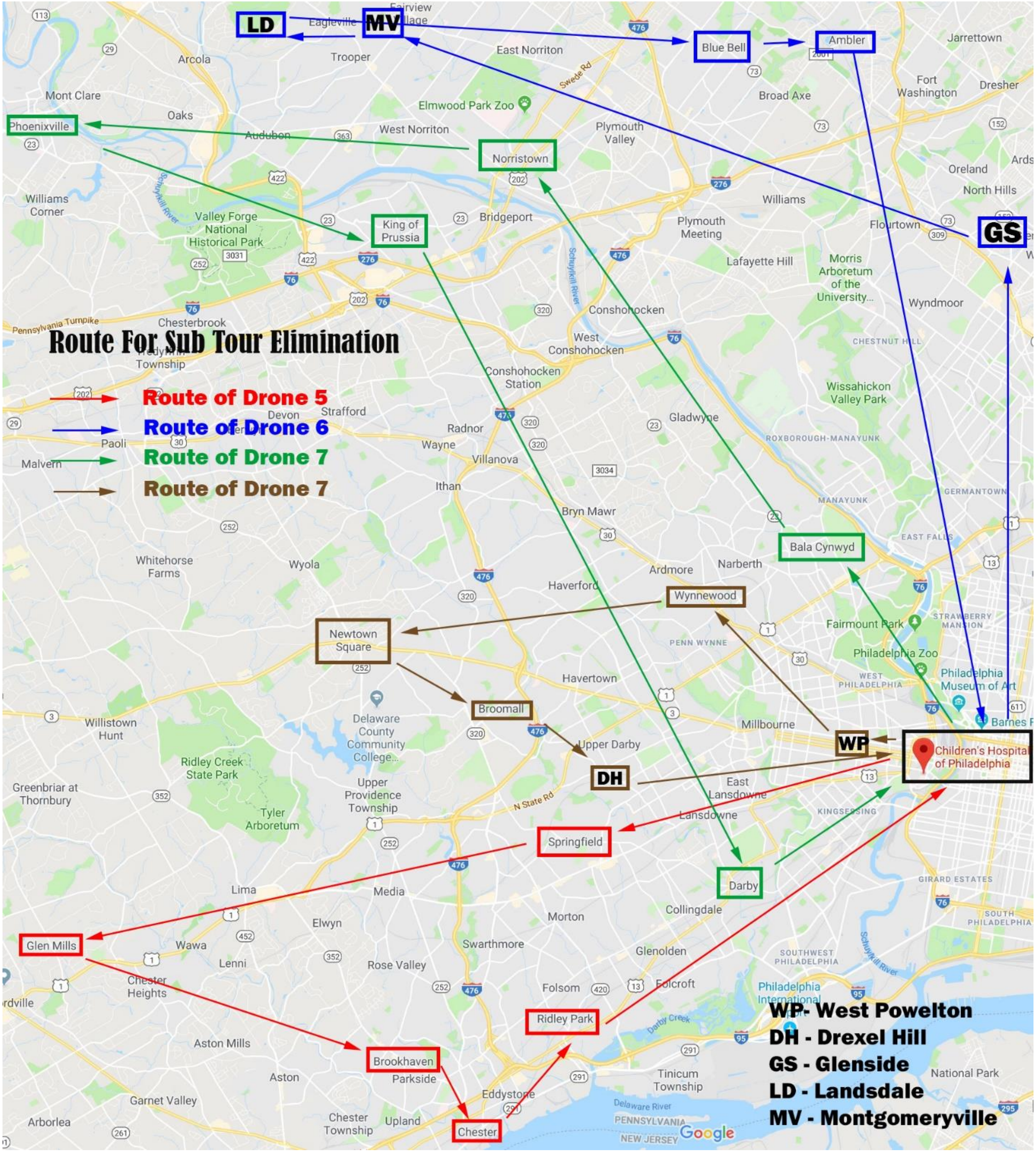


Figure 4: Route for Sub Tour Elimination

<u>Drone 5:</u> CDC → Springfield → Glen Mills → Brookhaven → Chester → Ridley Park → CDC
<u>Drone 6:</u> CDC → Glenside → Montgomeryville → Lansdale → Blue bell → Ambler → CDC
<u>Drone 7:</u> CDC →Bala Cynwyd →Norristown→Phoenixville→King of Prussia→ Darby→ CDC
<u>Drone 8:</u> CDC →West Powelton →Wynnewood →Newton Square →Broomall → Drexel Hill→ CDC

From this we could interpret that the objective value can have a better improvement as there is a relative gap of 32.98 % and this also increases MIP iterations and branch and bounds nodes but it needs high computational equipment to solve it.

B. <u>Relaxation:</u>

Objective Function:

$$\text{Minimize Z: } x_{ijk} * distance_{ij}$$

Where

$$x_{ijk} = \begin{cases} 1 \; if \; a \; drone \; k \; visits \; city \; i \; from \; city \; j \\ 0 \; if \; a \; drone \; k \; doesnot \; visit \; city \; i \; from \; j \end{cases}$$

$distance_{ij}$ = Distance between city i to city j

The primary constrain of the problem is that the demand of each hospital should be satisfied exactly by one and only drone. Now we are relaxing this constraint, and the demand of the hospital can be satisfied by more than one drone.

Here there are so many insights that can be observed.

1. The problem can never be infeasible – We are just relaxing the constraint and since we already have an optimal solution for the restricted problem, the solution of the restricted problem is a feasible solution for the relaxed problem.
2. The relaxed constraint might work as a redundant constraint because based on the demand and our objective function to minimize the total distance, there is highest possibility that the drone will never visit a hospital more than once, since the demand of each hospital is always less than capacity of the drones.

a. Constraints:

With comparison to the MTZ formulation the only constraint that were modified in order to relax the problem are given below and red indicate the changes made to the MTZ constraints in exact formulations to relax the constraints :

1. From city i → city j can be visited only once and by any number of drones.

$$\sum_{i=1}^{21} \sum_{k=1}^{8} x_{ijk} \geq 1 \qquad \forall \, j = 2 \; to \; 21$$

2. From city j → city i can be visited only once and by any number of drones.

$$\sum_{j=1}^{21} \sum_{k=1}^{8} x_{ijk} \geq 1 \qquad \forall \; i = 2 \; to \; 21$$

Earlier with the MTZ formulation this constraint was formulated in such a way that the drone visits the hospital exactly once (= 1). But, here it is formulated in such a way that the drone can visit the hospital more than once (>= 1). This helps us to relax the constraint that the demand of each hospital can be satisfied by more than one drone if necessary.

The rest of the constraints remains the same as MTZ formulation they are mentioned below for reference:

3. Demand Satisfying Constraint:

$$\sum_{i=1}^{21} \sum_{j=1}^{21} x_{ijk} * Demand_j \leq Capacity_k \quad \forall\, k = 1 \ldots 8$$

   As we have 8 drone available where 3 drones have a capacity of 450 and 5 drones have capacity of carrying the 1000 vaccines.

   For every tour a drone takes, the total number of vaccines should not exceed the total capacity of the drone and the above constraint ensures the same.

4. Each drone can fly to one or more than one city from the CDC Lab.
   Since we have the requirement that each drone must start flying from the CDC lab to different hospital in different cities.

$$\sum_{j=2}^{21} x_{1jk} \leq 1 \quad \forall\ k = 1 \ldots 8$$

5. Each drone comes back to the CDC lab from maximum of one city.

$$\sum_{i=2}^{21} x_{1jk} \leq 1 \quad \forall\, k = 1 \ldots 8$$

6.

$$\sum_{i=1}^{21} x_{irk} = \sum_{j=1}^{21} x_{rjk} \quad \forall\, k = 1 \ldots 8, \forall\, r = 2 \ldots\ 21$$

   The above constraint is used to form a route between the hospitals in different cities. The function of this constraint is it make a route such as if a drone starts in CDC → Ambler then the next city the drone must visit should start from the Ambler → Brookhaven.

7. MTZ – type Constraints:

$$u_i - u_j + N \sum_{k=1}^{8} x_{ijk} \leq N - 1 \qquad \forall \quad i = 2 \ldots 21, \forall\, j = 2 \ldots 21, \qquad i \neq j, \qquad N = 20$$

$$u_i - u_j + 20 \sum_{k=1}^{8} x_{ijk} \leq 19 \qquad \forall \quad i = 2 \ldots 21, \forall\, j = 2 \ldots 21, \qquad i \neq j, \qquad N = 20$$

b. Results:

Initially, we tried to solve the Formulation with in an indefinite time limit, but because of the computational constraints, the problem could not converge to a result, yielding a memory error. So, we did impose the time limit to the computation. The time limit given was 3 hours (10800 Secs) but it didn't stop the execution and it ran for 4 hours and we need to break the program in AMPLIDE and a feasible solution was obtained which is 17.86% away from the optimal solution. The corresponding statistics with the routes and results are displayed below.

Objective Value: 150.1007368
MIP simplex iterations: 119945634
branch-and-bound nodes: 2012234
absmipgap: 26.8177
relmipgap: 0.178665
Route Obtained:
Drone 1: Nil
Drone 2: Nil
Drone 3: Nil
Drone 4: Nil
Drone 5:
CDC → Drexel Hill → Broomall → Wynnewood → Bala Cynwyd → West Powelton → CDC
Drone 6:
CDC → Ridley Park → Chester → Brookhaven → Glen mills → Springfield → CDC
Drone 7:
CDC → Darby → Norristown → Phoenixville → King of Prussia → Newton Square → CDC
Drone 8:
CDC → Ambler → Blue Bell → Lansdale → Montgomeryville → Glenside → CDC

From this relaxation we could interpret that the objective value can have a might not have much better improvement as the is relative gap of the 17.86 %.

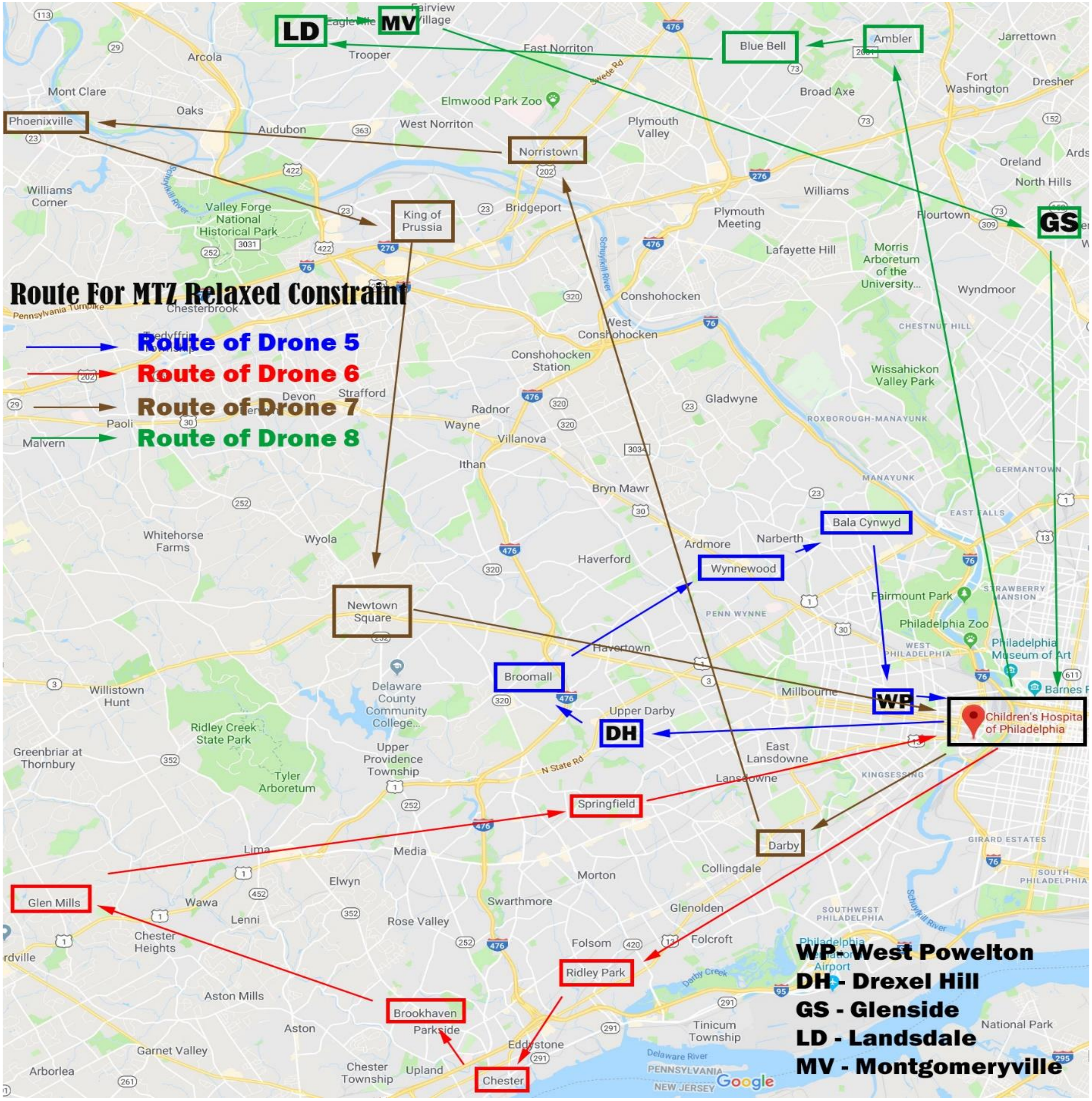


Figure 5: Route for MTZ Relaxed Constraint

<u>Drone 5:</u> CDC → Drexel Hill → Broomall → Wynnewood → Bala Cynwyd → West Powelton → CDC
<u>Drone 6:</u> CDC → Ridley Park → Chester → Brookhaven → Glen mills → Springfield → CDC
<u>Drone 7:</u> CDC → Darby → Norristown → Phoenixville → King of Prussia → Newton Square → CDC
<u>Drone 8:</u> CDC → Ambler → Blue Bell → Lansdale → Montgomeryville → Glenside → CDC

C. Heuristics:

In this, we have combined greedy and local search heuristic algorithm using Excel to generate locally optimal solutions. We have attached the Excel file named “Heuristics” which shows our entire work on Heuristics.
In this problem, heuristics were obtained to assign the Type II drones first and it is followed by the Type I drones. In the Excel file, there is a sheet named “Method”, which shows the approach and technique implemented to obtain the route and the distance simultaneously satisfying the constraints.

Firstly, for every tour, the drone starts its journey from the CDC Lab and ends in CDC.

COLOR CODING	
Color	Meaning
(green)	Currently Chosen
(pink)	Chosen Already

Drone 1 - Type II

	CDC	Glenside	Darby	Brookhaven	Glen Mills	Newton Square	Wynnewood	Montgomeryville	West Powelton	Chester
CDC	10000	12.38779005	10.47642273	11.55978354	18.3985569	9.19008321	4.53499296	20.91960591	0.70480272	11.2754

Figure 6: Minimum Distance from CDC

The above Figure 6 shows the first step and here the distance of CDC to the cities is displayed. In this, the city with the minimum distance is chosen and it is West Powelton here with a value of 0.7048.

Drone	Location	Demand of Loacation	Remaining Capacity of Drone	Distance from previous location
Drone 1 - Type II	CDC			
	West Powelton	195	805	0.70480272

Figure 7: Remaining Capacity and Distance for West Powelton

In the above Figure 7, the demand of the location is subtracted from the capacity of the drone. The demand of West Powelton is 195 and the total capacity of Type II drone is 1000, that implies the remaining capacity of the drone is 1000-195= 805. The drone with an available capacity of 805 will be used to supply the vaccines to the next location chosen. Similarly, the next location is chosen by taking the minimum distance but this time from West Powelton.

	CDC	Glenside	Darby	Brookhaven	Glen Mills	Newton Square	Wynnewood	Montgomeryville	West Powelton	Chester	Springfield	Ridley Park	Drexel Hill	Bala Cynwyd
West Powelton	0.70480272	11.86763769	9.78883002	11.630736	18.28274658	8.59866774	3.93804792	20.24189187	10000	11.43664	8.36064771	8.49310887	4.50386583	3.57253713

Figure 8: Distance from West Powelton

From Figure 8, it is evident that the minimum distance from West Powelton was found to be 3.5725 which corresponds to Bala Cynwyd. One important thing to note here is that, West Powelton can't be chosen again and it is highlighted in Red to indicate the same. The corresponding distance capacity calculation is shown below,

Drone	Location	Demand of Loacation	Remaining Capacity of Drone	Distance from previous location
Drone 1 - Type II	CDC			
	West Powelton	195	805	0.70480272
	Bala Cynwyd	183	622	3.57253713

Figure 9: Remaining Capacity and Distance for Bala Cynwyd

Here for Bala Cynwyd, the demand is 183 and thus the available capacity would be (805-183=622). The same is highlighted in yellow in Figure 9. This process would be repeated until the capacity of the drone is filled. After the capacity is filled, we will take another new Type II drone to deliver the vaccines to the other unassigned cities. This process will continue until all the cities are covered by the drones.

After calculating all the distances by meeting the capacity constraint, we have obtained the distance travelled to be 171.85.

Swapping

In this, we will try to swap each and every adjacent city to see if we can get a lower distance value than 171.85. If the objective value improves as we swap the cities then we will restart the process again for the next swap. Once all the cities are checked for the improvement and we can no longer find the improvement in the objective value we will stop the swaps.

Drone	Location	Number Reference For Cities	Demand of Loacation	Remaining Capacity of Drone	Distance from previous location
Drone 1 - Type II	CDC				
	West Powelton	1	195	805	0.70480272
	Bala Cynwyd	2	183	622	3.57253713
	Wynnewood	3	190	432	2.54733
	Drexel Hill	4	204	228	2.62393629
	Broomall	5	185	43	2.72228808
	CDC				7.50573891

Figure 10: Distance and Reference Number for the Cities

Table 5: Swapping Cities until Optimality

Swapping	Swapping City Names	Optimal	Improvement in Obj Funtion
(1) - (2)	West Powelton - Bala Cynwyd	No	4.96126728
(2) - (3)	Bala Cynwyd- Wynnewood	No	2.69702658
(3) - (4)	Wynnewood- Drexel Hill	No	3.98208192
(4) - (5)	Drexel Hill- Broomall	Yes	-1.0443752

Swapping
Drexel Hill and Broomal

The formula to find improvement in objective function value when we swap city i and city i+1is given as follows.

New distance(d) between cities (d_{i-1}, d_{i+1} & d_i,d_{i+2}) - Old distance(d) between cities (d_{i-1},d_i & d_{i+1},d_{i+2}).

So, for the first Swap (1)-(2), the New Value will be the distance between (CDC & Bala Cynwyd) + distance between (West Powelton &Wynnewood). Old distance value will be the distance between (CDC & West Powelton) + distance between (Bala Cynwyd & Wynnewood). They are shown in the Table below.

Table 6: Old Distance Value for the First Swap (1)-(2)

Between Cities	Distance
CDC & West Powelton	0.704
Bala Cynwyd & Wynnewood	2.54

Table 7:New Distance Value for the First Swap (1)-(2)

Between Cities	Distance
CDC & Bala Cynwyd	4.2754
West Powelton & Wynnewood	3.938

Here the Improvement in Objective function value is (4.2754+3.938)- (0.704+2.54) which gives a value of 4.961. So, this is not an optimal Swap as there is no improvement in the Objective Value.

There is an improvement in the objective value only if the value is negative. As shown in Table, the value of improvement is -1.044 for the swap (4)-(5) which is Drexel Hill-Broomall. Hence, on calculating the objective value using the newly created swap, we get a value of 170.8. This entire technique and method execution is shown in the same Heuristics file under the Sheet Swap 1. Similarly, the same technique is used to carry out further swaps and in Swap 2, we obtained an objective value of 168.25 by swapping (13)-(14) which is Norristown and Blue Bell. For Swap 3, we obtained the same objective value as there was no optimal swap and all the cities were tested for optimality by swapping the adjacent cities. So, we took the same result as Swap 2 which is 168.25. Also, further swaps are not possible since we tested all the cities and we can take 168.25 as the new objective value from Swapping.

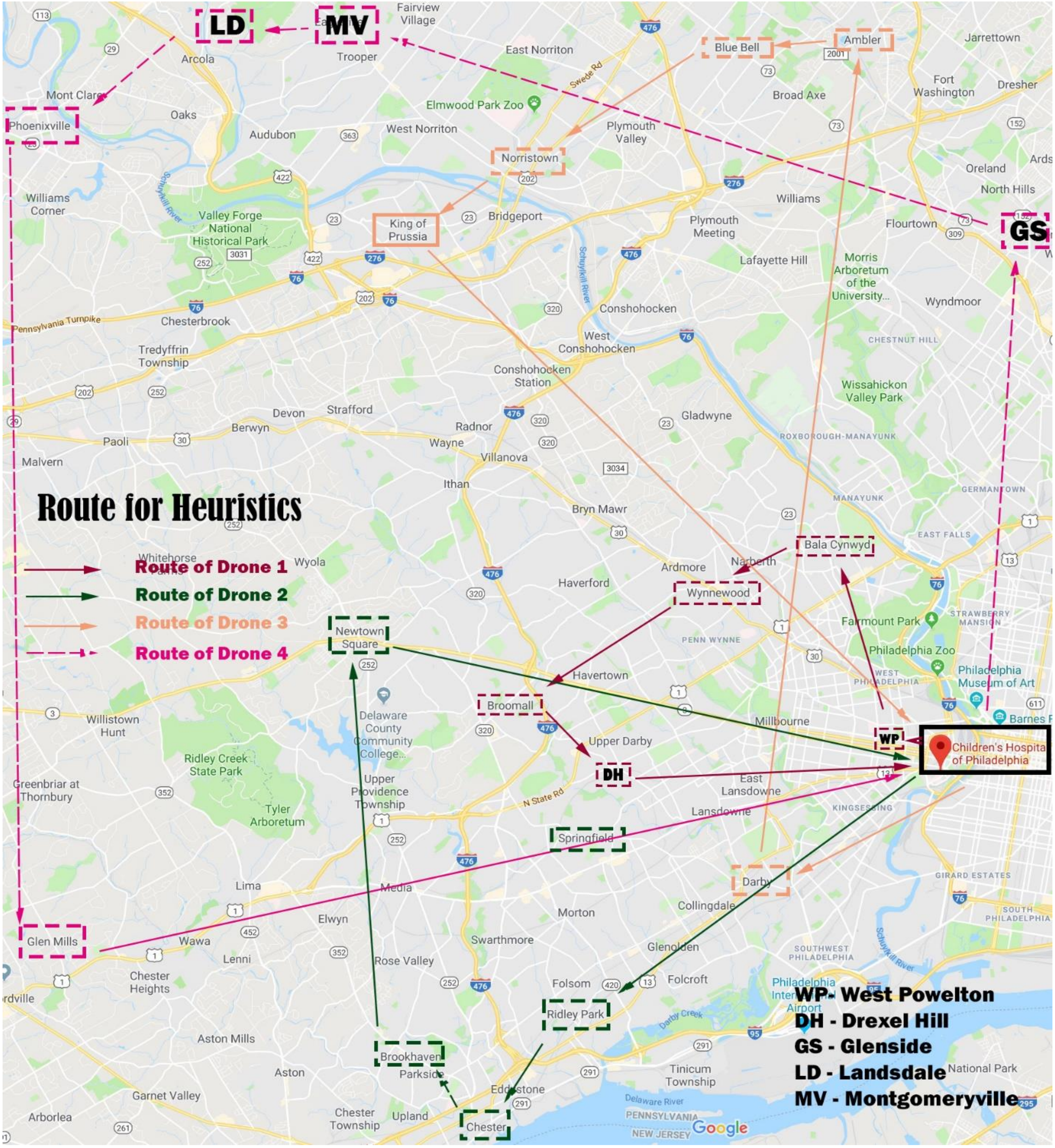


Figure 11: Route Map for Heuristics

<u>Drone 1:</u> CDC → West Powelton → Bala Cynwyd → Wynnewood → Broomall → Drexel Hill → CDC
<u>Drone 2:</u> CDC → Ridley Park → Chester → Brookhaven → Newton Square → CDC
<u>Drone 3:</u> CDC → Darby → Ambler → Blue Bell → Norristown → King of Prussia → CDC
<u>Drone 4:</u> CDC → Glenside → Montgomeryville → Lansdale → Phoenixville → Glen Mills → CDC

Comparison of Solution:

	MTZ Type Constraint	Sub Tour Elimination	MTZ Type Relaxed Constraint	Heuristics
Objective Value	150.1007368	153.776037	150.1007368	168.25
MIP Simplex Iterations	97081054	984275	119945634	-
Branch – And – Bound Nodes	1929891	55011	2012234	-
Absmipgap	29.9563	50.7255	26.8177	-
Relmipgap	0.199575	0.329866	0.178665	-
Time Limit (In Seconds)	10800	14400	14400	-
# Type 1 Drones	0	0	0	0
# Type 2 Drones	4	4	4	4
Total # Drones	4	4	4	4

Although there were various constraints and barrier during the process, with the available Computational power, various routes for the drones were obtained. MTZ and Sub tour elimination was used with a time limit of 10800 and 14400 seconds and two different routes were obtained.

Both the Methods made use of four Type II Drones. Obtaining the optimal solution was impossible and the solutions obtained by using MTZ and Sub tour elimination are 29.95% and 50.72% away from the Optimal Route.

Although we did not get the optimal route, based on the demand of the cities and the routes formed, we see that we cannot reduce the number of drones less than 4. We are concerned about minimizing the cost to schedule the drones and minimizing the distance travelled by the drones. Given the better computational availability, there are chances for improvising the Objective, but the improvement will be very small. So, the route obtained from MTZ constrain could be considered optimal and the route that is proposed for delivering the vaccines is

<u>Drone 4:</u> CDC → Springfield → Glen mills → Brookhaven → Chester → Ridley Park → CDC
<u>Drone 5:</u> CDC → Ambler → Blue Bell → Lansdale → Montgomeryville → Glenside → CDC
<u>Drone 6:</u> CDC → Drexel Hill → Broomall → Wynnewood → Bala Cynwyd → West Powelton → CDC
<u>Drone 8:</u> CDC → Darby → Norristown → Phoenixville → King of Prussia → Newton Square → CDC

Theoretical differences:

1. In the MTZ formulation we introduce an auxiliary variable which creates 21 new variables. On the other hand, in the subtour elimination method we are using the x_{ijk} variable to create subsets by subsets generation method and from the subsets we are eliminating the subtours.
2. MTZ requires $(n-1)^2$ constraints whereas the Sub tour elimination method requires 2^n constraints as the cities increases the constraints of subtour increases exponentially and this makes the computation time consuming. Also, it requires high computational power.

Recommendations:

For our problem we would like recommend that the MTZ with or without relaxation is better even though the sub tour elimination shows a high relmipgap and giving a solution near to the MTZ with and without relaxation objective but to solve the subtour elimination further we also require high computational requirements so that we recommend the MTZ solutions.

There are various relaxations on the drone like, it could travel to any number of cities in a given time period. But in real scale it is not possible and We will need to take that into consideration for a better result. So, given the constrains such as the distance it could travel in one hour, incorporating it will help us in obtaining real scale feasible solution.

As in the heuristics we have used only the adjacent swap operation instead we can also implement all other possible swaps and other heuristics algorithms such as metaheuristics can be used.

We can also implement that the drones battery does not drains even if it is not utilized and can be used for the next hour until the drones runs for exactly equal to the battery life.

This formulation can be extended to any number of cities, even with very high demands, it is possible to obtain a feasible solution given the apt computational efficiency we a few modifications in MTZ and subtour elimination constraints.

References:

1. **Population References**
   Glenside: http://www.city-data.com/city/Glenside-Pennsylvania.html
   Darby: http://www.city-data.com/city/Darby-Pennsylvania.html
   Brookhaven:http://www.city-data.com/city/Brookhaven-Pennsylvania.html
   Glen Millshttps://www.bestplaces.net/people/zip-code/pennsylvania/glen_mills/19342
   Newton Square: https://www.neighborhoodscout.com/pa/newtown-square/demographics
   Wynnewood: https://en.wikipedia.org/wiki/Wynnewood,_Pennsylvania
   Montgomeryville: http://www.city-data.com/city/Montgomeryville-Pennsylvania.html
   WestPowelton:http://www.city-data.com/neighborhood/West-Powelton-Philadelphia-PA.html
   Chester: http://www.city-data.com/city/Chester-Pennsylvania.html
   Springfield:
   https://en.wikipedia.org/wiki/Springfield_Township,_Delaware_County,_Pennsylvania
   Ridley Park: http://www.city-data.com/city/Ridley-Park-Pennsylvania.html
   Drexel Hill: http://www.city-data.com/city/Drexel-Hill-Pennsylvania.html
   BalaCynwyd:https://www.bestplaces.net/people/zip-code/pennsylvania/bala_cynwyd/19004
   Blue Bell: http://www.city-data.com/city/Blue-Bell-Pennsylvania.html
   Ambler: http://www.city-data.com/city/Ambler-Pennsylvania.html
   Phoenixville: http://www.city-data.com/city/Phoenixville-Pennsylvania.html
   Norristown: http://www.city-data.com/city/Norristown-Pennsylvania.html
   Lansdale: http://www.city-data.com/city/Lansdale-Pennsylvania.html
   Broomall: http://www.city-data.com/city/Broomall-Pennsylvania.html
   King of Prussia: http://www.city-data.com/city/King-of-Prussia-Pennsylvania.html

2. MTZ :
   https://neos-guide.org/content/multiple-traveling-salesman-problem-mtsp
3. Multiple Salesman Problem:
   https://dirkschumacher.github.io/ompr/articles/problem-tsp.html
4. Sub tour Elimination Example:
   http://www.academia.edu/13488484/Multitype_ant_colony_system_for_solving_the_multiple_traveling_salesman_problem
5. Multiple Salesman Problem:
   https://ieeexplore.ieee.org/document/7939224
6. https://ampl.com/faqs/how-can-i-get-ampl-to-index-over-the-power-set-consisting-of-all-subsets-of-a-set/

7. http://www.math.uwaterloo.ca/tsp/methods/opt/subtour.htm
8. Gabor, P. (2003). Formulations Using the Traveling Salesman Problem. *SIAM Review*, *45*(1), 116–123.

9. Kaempfer, Y., & Wolf, L. (2018). Learning the Multiple Traveling Salesmen Problem with Permutation Invariant Pooling Networks, 1–17. Retrieved from http://arxiv.org/abs/1803.09621
10. Assaf, M., & Ndiaye, M. (2017). Multi Traveling Salesman Problem Formulation. *International Conference on Industrial Engineering and Applications*, 292–295. http://doi.org/10.1109/IEA.2017.7939224
11. Chan, Y., & Baker, S. F. (2005). The multiple depot, multiple traveling salesmen facility-location problem: Vehicle range, service frequency, and heuristic implementations. *Mathematical and Computer Modelling*, *41*(8–9), 1035–1053. http://doi.org/10.1016/j.mcm.2003.08.011
12. Lee, J., & Raffensperger, J. F. (2006). Using AMPL for Teaching the TSP. *INFORMS Transactions on Education*, *7*(1), 37–69. http://doi.org/10.1287/ited.7.1.37
13. Matai, R., Singh, S., & Lal, M. (2010). Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches. *Traveling Salesman Problem, Theory and Applications*. http://doi.org/10.5772/12909
14. Matai, R., Singh, S., & Lal, M. (2010). Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches. *Traveling Salesman Problem, Theory and Applications*. http://doi.org/10.5772/12909
15. Vali, M., & Salimifard, K. (n.d.). A Constraint Programming Approach for Solving Multiple Traveling Salesman Problem, 1–17.