Meeting 1 :

Folder :

1. Proposal
2. Project Diary - include meeting discussion and what we explore

Possible direction for exploration.

→ Our Idea, Our Work

→ Easiest Thing : Toy Example

→ Battery Decreases Linearly

→ Formulate the Problem

→ Sanity Check

→ For first - two weeks.

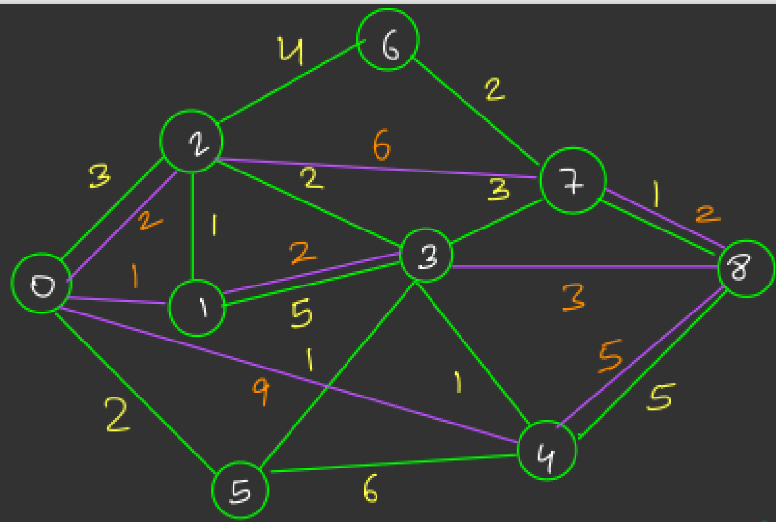
Assump. 1 : Infinite Battery

→ Complete Graph

→ Model to start from warehouse, sending some trucks then sending some drones.

Meeting 2 with TA :

For the 2nd meeting with the TA, we made a sample graph for the hybrid truck drone delivery system. Decided the weight of the edges and paths through which truck can travel or drone can travel. Wrote the assumptions and then found the shortest path for only truck, only drone. And thought of cost function which includes both cost to travel and time taken.



$\boxed{\text{Note}}$: In the above graph, Green lines are paths that trucks can travel, Purple lines are paths that Drones can travel. There is only 1 Depot and 1 customer. Assuming, traffic is uniform for truck in all paths. Yellow weights for edges is for truck and Orange weights for edges is for Drones.

→ 0 is the Depot location
8 is the Customer or Final Location

→ Now, let x_1 be the cost per unit path for trucks, similarly x_2 for Drones.

→ Now, let t_1 be the time per unit path for trucks, similarly t_2 for Drones.

→ Let v be the cost per unit time.

→ $\boxed{\text{Truck}}$:
Now, in the above graph, shortest path for truck will be nodes ⓪ → ⑤ → ③ → ⑦ → ⑧

So total Distance = 7 units
(adding edge weights)

So total cost for travelling $= 7x_1$
And total travel time $= 7t_1$

$\boxed{\text{Drones}}$: shortest path ⓪ → ① → ③ → ⑧

So total Distance = 6 units
(adding edge weights)

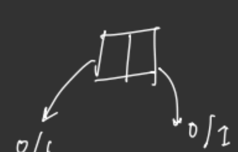
Total Cost for Drones $= 6x_2$
Total Time for Drones $= 6t_2$

→ $\boxed{\text{Assumption}}$: Cost x_2 for Drone is larger than cost x_1 for truck and time t_1 for truck is larger than t_2 for Drone. Now, for Hybrid Truck-Drone Model, one main assumption is that, total time taken by Truck is very large and total cost taken by Drone is very large for Delivery from node ⓪ → ⑧.

So only Only Truck or only Drone Model is not feasible in the above case.

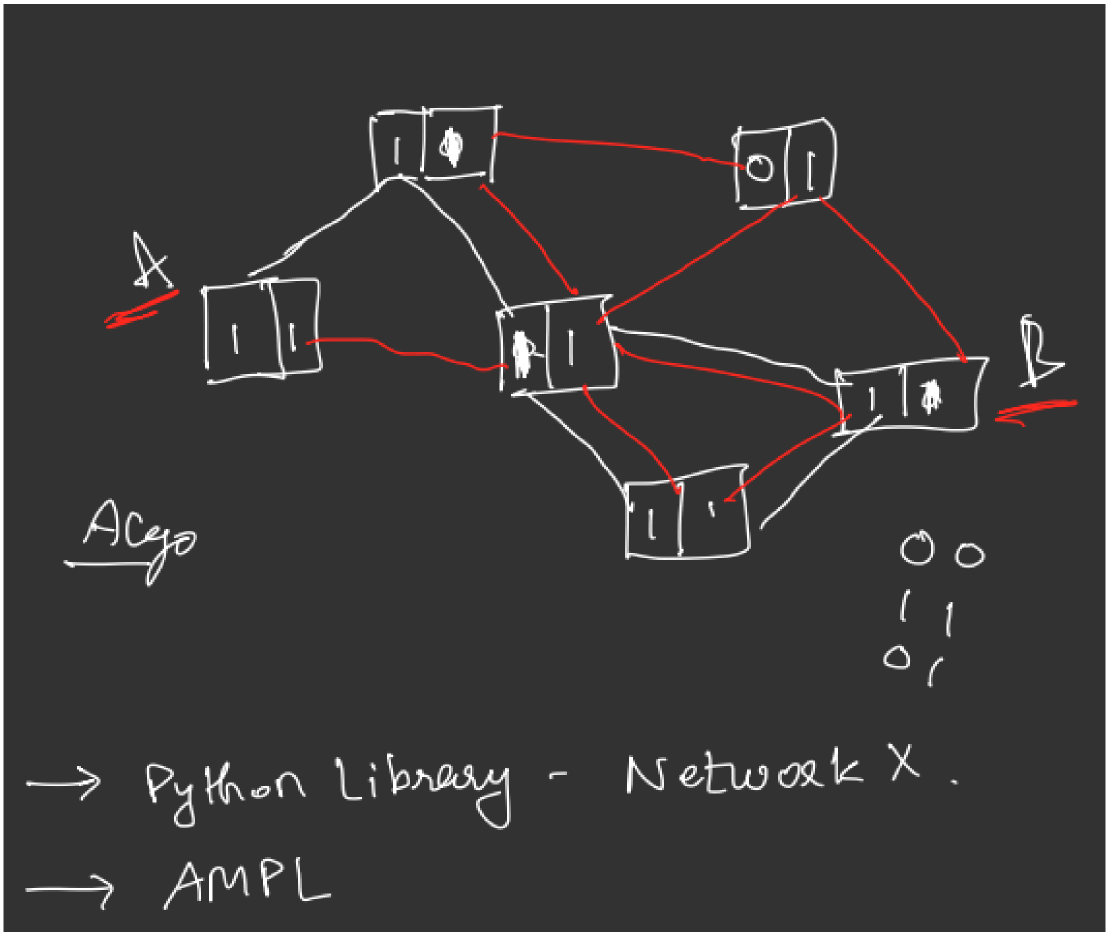
→ Now for Hybrid-Truck Drone Model, there is another variable for converting time to cost say v. cost/time

→ let ratio of cost $\frac{x_2}{x_1} = K$.
and ratio of time $t_2/t_1 = L$.
∴ Objective is to minimize cost.



node is available for truck [1] or [0]

→ Same for drone [1] or [0]



Hybrid Truck + Drone Delivery

1. **Assumptions:**

- Truck and drone move at constant speed.
- The flight path of a drone is a straight line, and the path length is expressed by Euclidean distance.
- Infinite drone battery.
- Truck is cheaper than Drone in terms of cost
- Drone is faster than trucks in delivery due to traffic and other constraints
- All customer demands are known before the start of operations and all customers must be served
- Customer specific delivery cost is estimated based on the desired time to delivery; Truck + Drone combination is selected accordingly

- Drones may be dispatched from the truck at the depot before the truck moves.
- Drones can serve only a single customer request per dispatch.
- Drones can only serve requests that do not exceed their physical carrying capacity.
- Demand of each customer can be fulfilled by one visit of either the truck or a drone.
- Customers are available during the time of the delivery, and each customer is visited only once.
- There is only one parcel in the delivery task for each customer.
- Drones' charging time is negligible.
- Trucks can serve drones many times.
  // Lines marked in Red are excluded as of now.
- Drones can carry at most one parcel.
- The truck can only stop at depot locations to launch Drones
- The flight range of the drone is limited by time.
- The number of deliveries per trip is limited by the load capacity of the drone.
- There is a set of potential stops from where the truck can launch or collect the drone.
- Drone is inherently faster than Truck due to Traffic and other constraints.
- When drones are launched from the truck, they have to rendezvous with the truck at any of the subsequent stops along the route or at the depot.
- When a drone and the truck reconvene at a customer location, the one arriving earlier waits for the other.
- UAVs are recharged at the truck instantaneously.
- Multiple UAVs can be launched simultaneously.
- The return time to the depot is not considered in the objective value since we focus on minimizing the waiting time of the customers.
- The drone is assumed to have a DL capacity in weight unit, where each customer has a demand of a single weight unit.

2. Decision Variables

- T_{ij}: Distance traveled by the truck between location i and j; i, j = 0, 1, …, I
- d_{ij}: Distance traveled by the drone between node i and j; i, j = 0, 1, …, n + I
- αD: Unit variable cost for drone
- αU: Unit variable cost for truck
- x_{ij}: Binary variable; 1 if truck moves from node i to node j ; 0 otherwise.
- y_{ij}: Binary variable; 1 if UAV u travels from node i to visit node j and returns to the truck at node s ; 0 otherwise.
- z_i: Position of node i in the truck route
- t_i: Arrival time of truck at node i
- L_i: Departure time of truck from node i
- t_{iu}: Arrival time of UAV u at node i
- L_{iu}: Departure time of UAV u from node i
- x_{ij}: Binary variable; 1 if truck traverses the link (i, j); 0 otherwise.
- y_{uijs}: Binary variable; 1 if UAV u travels from node i to visit node j and returns to the truck at node s; 0 otherwise.
- u_{jk}: Number of nodes visited by the drone from depot to node j
- p_{ij}: Binary variable; 1 if node i is visited before node j by truck; 0 otherwise.
- z_i: Position of node i in the truck route
- t_i: Arrival time of truck at node i
- L_i: Departure time of truck from node i
- t_{iu}: Arrival time of UAV u at node i
- L_{iu}: Departure time of UAV u from node i
- Y_{tij}: Binary variable equals 1 if the truck t traverses the link (i, j), and 0 otherwise.
- X_{dij}: Binary variable equals 1 if drone d traverses the link (i, j) onboard of the vehicle, and 0 otherwise.
- E_{dij}: Binary variable equals 1 if drone d dispatches from station i and collects from station j, and 0 otherwise.

- Fdij: Binary variable equals 1 if drone d collected from station i by truck, and 0 otherwise.
- Odijr: Binary variable equals 1 if drone d stops at node i for serving, and 0 otherwise.
- Zdis: Binary variable equals 1 if drone d dispatched from station i travels on link (i, s), and 0 otherwise.
- θd: Time for arriving node i for drone d.
- θd': Time for leaving node i for drone d.
- ϕt: Time for arriving node i for truck t.
- ϕt': Time for leaving node i for truck t.
- Q: The earliest time to return to the depot.
- wdc: The weight of parcel that drone d delivers to customer c.
- l: Number of potential truck locations, indexed by i = 0, 1, …, l
- n: Number of customers nodes, indexed by j = l + 1, …, l + n
- DL: Maximum number of customers visited by the drone in a single trip
- R: Maximum drone's flying range
- Yijk: Binary variable equals 1 if the truck moves from node i to node j to collect the drone, in trip k and 0 otherwise.
- Xijk: Binary variable equals 1 if the drone moves from node i to node j in trip k and 0 otherwise.
- zijk: Binary variable equals 1 if the truck moves from node i to node j to launch the drone in trip k and 0 otherwise.
- pij: Binary variable; 1 if node i is visited before node j by truck; 0 otherwise.
- Tripk: Binary variable equals 1 if the trip k is made and 0 otherwise.

After explaining all this to TA, he approved that we can go forward with this and told us to build the objective function and he suggested a python library Network X for graphs and also AMPL optimization tool to describe and solve optimization problems and get the results of our model and check if it is intuitive or not.

Meeting 3: