On-demand high capacity ride sharing via dynamic trip vehicle assignment

1. Ride-sharing services present enormous potential for positive societal impacts w.r.t pollution, energy consumption, songestion, etc.
2. Dataset used: New York City Taxicab. Rider capacity of upto 10 simultaneous passengersper vehicle.
3. Decoupling: Ride pooling problem = Vehicle-routing problem + dynamic pickup and delivery problem
    a. Arbitrary number of passengers and trips
    b. Anytime optimal rider allocation
    c. Routing dependent on the fleet location
    d. Online rerouting and assignment of riders to existing trips
4. Trade-offs between:
    a. Fleet size
    b. Capacity
    c. Waiting time
    d. Travel delay
    e. Operational costs

Passenger Assignment and Vehicle Routing

1. Each travel request = Time of request + prickup location + drop-off location
    a. Fleet (ŋ) of (m) vehicles and capacity (c).
    b. Requests (R) = {r1,r2,...,rn}
    c. Vehicles (V) = {v1,v2,...,vm}
    d. C = cost function
    e. Z = constraints
    f. Pv = passenger picked up by vehicle v and going towards drop-off location
2. Solved using Integer Linear Optimization (ILP)
3. For each request (r):
    a. [waiting time = pick-up time - request time] < max waiting time
    b. [Total travel delay = drop-off time - request time - earliest possible time destination is reached with shortest path between origin and destination] < max travel delay
4. For each vehicle, No. of passengers <= capacity
5. C = cost function = sum of delays

$$C(\Sigma) = \sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{P}_v} \delta_r + \sum_{r \in \mathcal{R}_{ok}} \delta_r + \sum_{r \in \mathcal{R}_{ko}} c_{ko}.$$

6. Algorithm to compute feasible trips and edges process incrementally in trip size for each vehicle, starting from the request-vehicle edges in the RV graph.
7. Request collected in a time-window, afterwhich they are assigned in batch to different vehicles.
8. Greedy approach: Trips are assigned to vehicles iteratively in decreasing size of trip and increasing cost (sume of travel delays). This is to maximize amount of requests and minimize cost.
9. Rebalance fleet by moving only idle vehicles to unassigned requests, and start again.

Possible ways of using AI

1. For vehicles with larger capacity, heuristic methods such as Lin–Kernighan, Tabu search, or simulated annealing may be used. (Constrained Search Problem)
2. Use A* algorithm to find optimal trip paths and take the sum of delays as the loss function.
3. Use GD with optimizers (ADAM) and reach to an optimal solution by minimizing the cost function.
4. Predict the demand and rebalance idle vehicles to a potentially high-demand area thereby reducing the estimated time arrival (ETA).
5. Do route optimization based on parameters such as congestion and traffic.
6. Work on improving the time complexity of O(mn^v).