Readme

Function new_dispatch() contains the following modules:

- **Preprocessing**
    - Nonempty vehicle: dealing with the carrying items of vehicles and update incoming vehicles (these vehicles could be assigned to their destination nodes)
    - Empty vehicle: it has been allocated to the order, but have not yet arrived at the pickup factory

- **Route-level packing**
    - Build dictionaries to check whether the order can be split or not
    - Packing items for each route (pack items into bags, bag capacity equals to vehicle capacity 15. Then try to fill up each bag.)
    - Note that the items will be packing in order of arrival
    - Each bag will contain attributes:
        - 1) route_str: contains the source and destination node strings;
        - 2) total_time: sum of load_time and unload time;
        - 3) min_creation_time: the earliest created item;
        - 4) route_item: contains item objects;
        - 5) route_capacity: total demand of this bag.
    - Reversing the items of this bag: due to the FILO constraint, we prefer to put the earliest item to the last position

- **Compute pickup_time matrix**
    - Use “vehicle_to_source” matrix to record pickup time, compute pickup_time of each available vehicle for each bag

- **Allocating unallocated items**
    - **Add items to preallocated vehicles**. If v_1:

        - 1) has been allocated to route k with bag m;

        - 2) has not yet arrived at the pickup_factory;

        - 3) capacity of bag m < 15;

        - 4) new items which can be add into this bag;

        then we will add items to the preallocated vehicle
    - **Allocate bag/items to available/unallocated vehicles**

        - We will check bags in order of creation time (early to late), and pick the minimum pickup_time vehicle

        - Then we will check whether it meets the allocation conditions:

            - <u>Time threshold:</u>

            - if "bag.total_time(load and unload time) + earliest item live time + 1800*2 (approach time*2) + min_pickup_time + 600 (interval) > 14400"

            - <u>Capacity threshold:</u>

            - for the topside (the most popular site): bag.capacity > 15 - pickup_time//600;

            - other sites: bag.capacity > 9 - pickup_time//600

            - <u>If there is incoming car:</u>

            - assign the incoming car to the bag of this site, it will save one approach time

- **Combine duplicate nodes and write the output**
    - Combine duplicate nodes: combine the pickup_items and delivery_items of the adjacent duplicate nodes
    - Write the "vehicle_id_to_destination" and "vehicle_id_to_planned_route"