depot_delivery.py documentation

/ /

This document explains the class which is responsible to generate user defined number of random delivery locations in the radius of 8 km for Munich City.

Background: Using OSMnx python package a short script was written to download the driveable data of Munich city. The radius of 8000m around city centre was considered. The data was saved in .graphml format for further use.

The class MunichDeliveryNetwork in depot_delivery.py is explained in following steps.

a) init function

- The saved city map is imported and the city centre is located using geocode command of osmnx.
- For simplicity, the map is converted to undirected nature. The undirected means that both way travel is allowed on the roads. This is done to simplify the problem.
- The node closest to the centre is fixed as the depot.

b) select_delivery_location

- This function takes input of number of locations, min and max distance from depot. The maximum distance should be lower than 8000m.
- Using dykstra algorithm, the random delivery locations are selected. The algorithm is also used to find minimum distance between depot and delivery locations and also between delivery locations.
- The distances are crucial to create the distance matrix.
- The distance matrix shows the information about distances between depot and delivery as well as between deliveries.

eg:

	Depot	Delivery 1	Delivery 2
Depot	0	1000	5000
Delivery 1	1000	0	2300
Delivery 2	500	2300	0

→ shows shortest distance between delivery location 1 and delivery location 2.

These distances are calculated by Dykstra algorithm. Study of this is not in scope.