Coupling Ridesharing and Public Transit System as First-mile/Last-mile Transit Strategy

Yen-chu Wu (yenchuw2), Denissa Purba (dpurba2),
Ruolin Zhang (ruolin3), Shanshan Liu (sl116)

Abstract

Public transit becomes one of the strategic solutions towards sustainable transportation. However, such travel modes are suscept to low accessibility and public dependency issues, such as low level of service due to low operation management and lack of accessibility to connect origin and destination points. This problem is known as the first- and last-mile transit problem. Ridesharing systems have opportunities to tackle such limitations that connect origin points to transit stations and transit stations to destination points. Our project aims to model peer-to-peer ridesharing and the transit system as first-mile/last-mile transit problems. We construct the problem as Capacitated Vehicle Routing Problem with Time Window (CVRPTW) to minimize the total travel cost and maximize the rider's satisfaction. We construct Saving Method with time window constraints to solve the problem heuristically at a larger network size. The model formulation and solution consistency are verified on small problem size using Gurobi (optimal solution) and proposed constructed heuristic methods. Finally, we implement our proposed model on 30 New York City trip request data as our experimental area.

INTRODUCTION

The majority of commuting trips in the US use private cars as travel mode with single-occupant travel. Large numbers of trips with low vehicle occupancy rates often lead to severe traffic congestion in urban areas and produce excess traffic gas emissions. Following the current decarbonized transportation efforts, local governments suggest the use of public transportation as one of the strategies to reduce such negative external effects of car travel. However, there are some challenges to public transit. First, it has low efficiency of operation management, low frequency, and delays. Second, its accessibility is low, such as long commuting time, parking problems, stopping points far away from the starting point or final destination. Therefore, transportation agencies must find better solutions to the first- and last-mile transit problems to attract people to public transportation. A cheaper and more environmentally sustainable strategy for addressing the first-mile/last-mile transit problem is by utilizing the shared mobility of commuting vehicles as shuttles connecting to the public transit. Ridesharing systems, which are sustainable and more flexible, can share passenger mobilities with similar travel patterns (e.g., origin, destination, and schedule), are efficient in improving vehicle occupancy and effective in reducing emissions.

Our project aims to study the potential benefits of integrating peer-to-peer ridesharing and the transit system to improve the first-mile/last-mile transit strategy. Minimizing vehicle miles, minimizing travel time, and maximizing the matchings are common objectives when optimizing the ridesharing system. In this project, we plan to formulate a ride-matching problem coupling ridesharing and transit mobility that minimizes the total travel cost and maximizes the rider's satisfaction. We take New York City as our experimental area. Our problem is that given a set of requests, a road network, and a transit network, find: (1) An optimal match between drivers and riders for ridesharing so that a given objective function is optimized; (2) Optimal ridesharing chain for each driver and rider.

PROBLEM FORMULATION

In this project, we present the formulation of the first- and last-mile transit problem using ridesharing. We use ridesharing to connect the request trips from origin nodes to the transit stations and connect the request trips from transit stations to the destination nodes. It is worth noticing that the problem is separable. We could divide the problem into two separate problems, i.e. (i) First-mile Problem, where we assign drivers to the request trips from origin nodes to transit nodes; and (ii) Last-mile Problem, where we assign drivers to the request trips from transit stations to destination nodes. Using such an approach, we solve the first- and last-mile problem as two ridesharing problems.

The ridesharing problem formulation is constructed as Capacitated Vehicle Routing Problem with Time Window (CVRPTW) (Lin et al., 2012). In the ridesharing problem, we assume that there are two agents in the problem, i.e., the drivers and the riders. Each rider creates a single trip request during trip periods going from their origin to the destination. Each requested trip from the rider may have more than one rider load (group ride). Each rider/group rider will ride in the same vehicle during the requested trip. Each driver will be matched to the requested trip according to the time window. Each rider may serve more than one requested trip (more than one rider in the vehicles) if the total load is still within the vehicle capacity. In the ridesharing problem, we have three objectives:

1. Minimize the transportation cost
2. Minimize the waiting time of each request trip
3. Minimize the extra riding time of each rider due to the ridesharing

For the First-mile Ridesharing problem, the origin nodes are any possible pick-up nodes, and the drop-off nodes are strictly the transit stations. For the Last-mile Ridesharing problem, we have transit stations as the pick-up nodes and the destination nodes as the drop-off nodes.

Network Definition

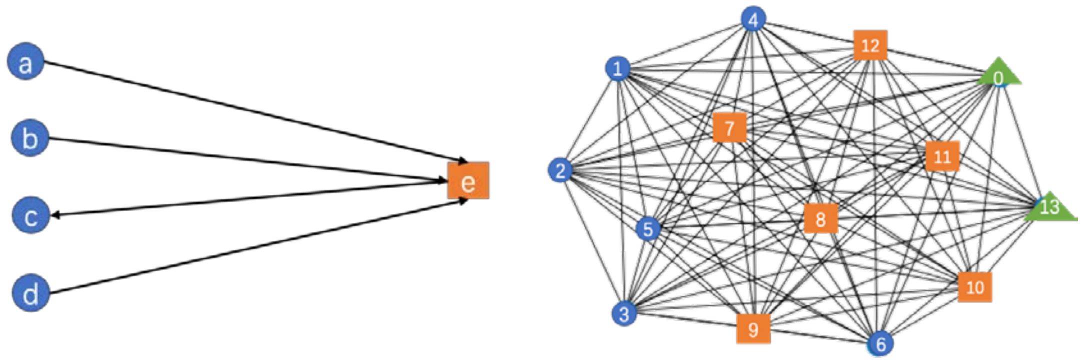
From the problem definition, we consider that each rider will create a trip request during time periods, and we will match the request with the available rider during that time window. Note that it is possible to have more than one trip demand requested in each pick-up node. Also, each rider could make more than one trip for different time windows. It is hard to model the network as a real transportation topology network for the ridesharing problem because more than one request is generated from the same pick-up node. Hence, we construct the trip request network that connects every request trip within the time window.

Let define a complete graph where a node represents one trip/request-id. We define n as the number of requests that need to be completed. We define P as the pickup node of each request-id; $P = \{1,2,\ldots,\mathrm{n}\}$. Define D as dropoff node of each request-id; $\mathrm{D} = \{\mathrm{n}+1, \mathrm{n}+2, \ldots, 2\mathrm{n}\}$. Then, we create two dummy nodes, i.e., node 0 as dummy node for the vehicle to depart, and node $2n+1$ as return dummy node. Overall, the total node in the transformed network is in a set of $N = \mathrm{P} \cup D \cup \{0, 2n+1\}$ with the cardinality of $|N| = 2n+2$.

Figure 1 presents the example of the network construction for our problem. Let us consider the trip request detail as shown in Table 1. For each request, we have the request ID, the pick-up and drop-off nodes, and the pickup time window. Using the pick-up and drop-off nodes, we could manually identify the type of request whether it is a first-mile or last-mile request. In this example, Requests 1, 2, 3, and 6 belong to the first-mile problem where passengers would be picked up on their origins to transit stations e. Requests 4 and 5 belong to the last-mile problem where passengers would be picked up at the transit station e and then dropped off at their destinations c. We could illustrate the network topology as in Figure 1(a). However, as we described, it is more practicable to model the network as the trip request network. For this problem, we modified the problem into a trip request network as in Figure 1(b).

Table 1 Example of trip request details in ridesharing problem

Request ID	Type	Pickup Node	Dropoff Node	Pickup Time Window
1	First-mile	a	e	12:45-13.15
2	First-mile	b	e	13:00-13:30
3	First-mile	b	e	10:00-10:30
4	Last-mile	e	c	08:30-09:00
5	Last-mile	e	c	10:20-10:50
6	First-mile	d	e	09:30-10:00



(a) (b)

Figure 1 (a) Example of the origin-destination network; (b) modified trip request network

Parameters and Notations

M	the set of vehicles
n	the number of nodes to be served
P	the set of pick-up nodes
D	the set of drop-off nodes
N	the set of requests; $N = P \cup D \cup \{0,2n+1\}$ and $\|N\| = 2n+2$
$[e_i, l_i]$	pickup time windows associated with request node i
q_i	the demand of m −th vehicle at request node i
t_{ij}	shortest travel time from node i to j
L_i	the maximum ride time of a passenger, $L = (1+r)t_{i,i+n}$,

r	the acceptance rate to wait extra time
s_i	service duration of node i (unload/load passengers)
Q	vehicle capacity
T_m	maximum travel time for the $m-$th vehicle to complete all tasks on each route
$\alpha_1, \alpha_2, \alpha_3$	the weight of each objective

Decision Variables

x_{ij}^m	1 if vehicle m serve the request i then request j; 0, otherwise
Q_i^m	the load of vehicle m after serving the request i
B_i^m	the time when the vehicle m starts serve request i
L_i^m	the ride time of request i on the m-th vehicle

Mathematical Formulation

$$\min \alpha_1 \sum_{m\in M} \sum_{(i,j)\in A} t_{ij} x_{ij}^m + \alpha_2 \sum_{m\in M} \sum_{i\in P} (L_i^m - t_{i,n+i}) + \alpha_3 \sum_{m\in M} \sum_{i\in P} q_i (B_i^m - e_i) \tag{1}$$

$s.t.$

$$\sum_{m\in M} \sum_{j\in P\cup D} x_{ij}^m = 1, \forall i \in P \tag{2}$$

$$\sum_{j\in N} x_{ji}^m = \sum_{j\in N} x_{ij}^m, \forall i \in P \cup D, \forall m \in M \tag{3}$$

$$\sum_{j\neq 0\in N} x_{0j}^m = \sum_{j\neq 2n+1\in N} x_{j2n+1}^m = 1, \forall m \in M \tag{4}$$

$$\sum_{j\in N} x_{ij}^m = \sum_{j\in N} x_{jn+i}^m, \forall i \in P, \forall m \in M \tag{5}$$

$$e_i \leq B_i^m \leq l_i, \forall i \in N, \forall m \in M \tag{6}$$

$$B_{2n+1}^m - B_0^m \leq T_m, \forall m \in M \tag{7}$$

$$B_j^m \geq (B_i^m + s_i + t_{ij})\ x_{ij}^m, \forall i \in N, \forall j \neq i \in N, \forall m \in M \tag{8}$$

$$L_i^m = B_{n+i}^m - (B_i^m + s_i), \forall i \in P, \forall m \in M \tag{9}$$

$$t_{i,n+i} \leq L_i^m \leq L_i, \forall i \in P, \forall m \in M \tag{10}$$

$$Q_j^m \geq (Q_i^m + q_j)\ x_{ij}^m, \forall i \in N, \forall j \neq i \in N, \forall m \in M \tag{11}$$

$$\max\{0, q_i\} \leq Q_i^m \leq \min\{Q, Q + q_i\}, \forall i \in N, \forall m \in M \tag{12}$$

$$x_{ij}^m \in \{0, 1\}, \forall i \in N, \forall j \in N, \forall m \in M \tag{13}$$

$$Q_i^m, B_i^m, L_i^m \geq 0, i \in N, \forall\ m \in M \tag{14}$$

The objective function in (1) aims to minimize the total transportation cost, the total waiting time of each request trip, and the total extra riding time due to ridesharing. Constraints (2) guarantee that each request is served exactly once. Constraints (3) are the flow conservation constraints, if we visit a requesting node, then we will leave that request to. Constraints (4) control that if the vehicle departs, then the vehicle must return. Constraints (5) manage that each one request's origin and destination must be visited by the same vehicle. Constraints (6) define the pickup time windows constraints.

Constraints (7) bound the duration of each route of a single vehicle. Constraints (8) guarantees that vehicles at request i should not arrive the request j before the request at node j start, B_j^m. The ride time of each passenger is controlled by constraints (9) and (10). The consistency of the load variable is guaranteed by constraints (11). Constraints (12) state that before scheduling a vehicle, it should be examined that the residual capacity can satisfy the needs of the next request. Lastly, constraints (13) and (14) define the decision variable definitions.

Focusing on constraints (12), we notice that these constraints are nonlinear constraints, which can be linearized as the following,

$$Q_i^m \leq Q, \forall i \in N, \forall m \in M \tag{15}$$

$$Q_i^m \leq Q + q_i, \forall i \in N, \forall m \in M \tag{16}$$

$$Q_i^m \geq 0, \forall i \in N, \forall m \in M \tag{17}$$

$$Q_i^m \geq q_i, \forall i \in N, \forall m \in M \tag{18}$$

SOLUTION METHOD

We present the constructed heuristic method to solve the ridesharing problem CVRTW. We construct the saving method with time window constraints to control the pick-up time window feasibility in our solution. The modified saving method with time window is presented as follows:

- **Step 1:**
  Assume that each trip/request is served by an individual vehicle and calculate the objective value
- **Step 2:**
  Divide the trips/requests into different groups based on their destination. Eliminate the pairs (i, j) that cannot meet the pick-up time of requests constraints or vehicle capacity constraints or vehicle's time window, from which a candidate set of pairs that can be merged is obtained.
- **Step 3:**
  In the candidate set of pairs that can be merged, compute whether the pairs meet all the time window constraints to obtain the final set of pairs (i, j) can be merged.

- **Step 4:**
  Compute savings $s(i,j) = 0.6s_{o1} + 0.2s_{o2} + 0.2s_{o3}$. Sort savings and merge the pairs (i,j) from the top of the saving list.
- **Step 5:**
  From the final set of pairs (i,j) can be merged, select pairs (m,i) and (j,k) to be the new candidate set of pairs and back to Step 3 until the new candidate set of pairs is $\emptyset$.

DATA

The New York City Taxi and Limousine Commission (NYC TLC), in partnership with the NYC Department of Information Technology and Telecommunications (DOITT), has published millions of trip records from yellow medallion taxis and green SHLs for several years (https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page). These records include attributes such as pick-up and drop-off dates, times, and locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts.

Our study area is the Manhattan area, NYC. We select trips that are geographically concentrated in the Manhattan area to design ridesharing and public transit system problems. The study area consists of 184 stations where riders and drivers start or end their trips. These stations are selected at locations with high trip demand. Stations are distributed in the network to ensure that at least one station is within walking distance (< 0.15 miles) of a typical trip's origin/destination.

The data used in this project belongs to the evening peak hour (19:00-20:00) of Feb 19th in 2016. We select trips that are geographically concentrated in the Manhattan area, NYC. For each trip, they contain information such as the trip number, their start time (19:00-20:00), their end time, the origin station (1-184), and the destination station (1-184). The total number of trips is 21,330. Since our project focuses on the first/ last mile problem, we assume that transit stations with the most requests. Our project selected three transit stations whose index number is 95, 97, and 123. Then, set the remaining 181 station nodes as the potential origin pick-up and destination drop-off nodes for our first- and last-mile transit problem.

Lastly, we conducted data preprocessing to obtain the adjacency matrix and travel time matrix within each location node. We assume that all trips will be completed on their shortest travel time paths. We obtain the travel times from the Google Maps API.

NUMERICAL EXAMPLES

Sample Network

We construct a sample network to verify that the model and algorithm are correct and consistent. Table 2 illustrates the sample network with 14 trips which is selected from the NYC dataset. The sample network has 14 trips, which include 7 first-mile trip requests and 7 last-mile trip requests. The number of passengers in each pickup node, the pickup and drop-off time window are also given.

As for this sample network, we utilize the Gurobi package and constructed a heuristic algorithm (i.e., modified saving method) to solve the first- and last-mile problem. In the first-mile problem, we consider Trip ID 1-7 where the drop-off nodes are the transit stations (i.e., nodes 95, 97, and 123). For the last-mile problem, we consider Trip ID 8-14, and the station nodes become the pick-up nodes. If the vehicle does not serve any passengers, it will go from the origin dummy node to the destination dummy node directly. Thus, we can easily tell that the number of vehicles is needed for the trip requests.

Table 2 Trip information in the sample network

Trip ID	Pickup Node	Number of Passengers	Pickup Time Window	Drop off Node	Number of Passengers	Drop off Time Window
1	126	2	[40,57]	95	-2	[60,77]
2	98	1	[24,47]	95	-1	[30,53]
3	79	1	[15,50]	95	-1	[33,68]
4	127	1	[30,45]	97	-1	[44,64]
5	33	2	[16,45]	97	-2	[36,70]
6	77	2	[12,41]	123	-2	[17,55]
7	94	3	[38,57]	123	-3	[45,73]
8	95	2	[2,26]	124	-2	[21,45]
9	95	2	[13,43]	100	-2	[26,56]
10	95	1	[3,17]	157	-1	[9,23]
11	97	2	[20,48]	49	-2	[26,59]
12	97	1	[13,35]	76	-1	[26,53]
13	123	2	[2,28]	50	-2	[20,55]
14	123	2	[39,67]	152	-2	[59,96]

Table 3 and Figure 2a showcase the optimal results of the first-mile problem obtained from Gurobi. 5 vehicles are needed to serve all trips to serve the first-mile requests. The first 3 vehicles, only pick up one group of passengers and go to the transit stations directly. The fourth vehicle picks up passengers at node 79 and 98 and then drop off them together at transit station 95, which is a ridesharing trip. The fifth vehicle picks up passengers at node 77 and drop off them at transit station 123, and then picks up another group of passengers at node 126 and drops off them at the transit station 123 again. The optimal objective value of the first-mile problem is 227.

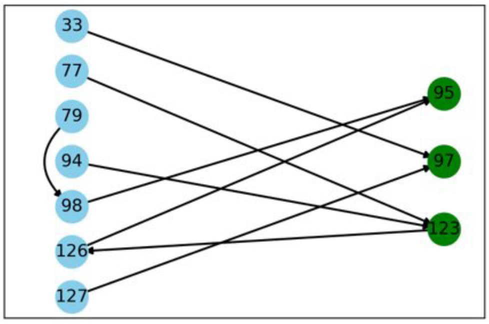
As for the last-mile problem, the pickup nodes are the transit stations, which is the reverse trip of the first-mile trip. Table 4 presents the optimal results of the last-mile problem. We need 4 vehicles to satisfy the requests of passengers from the transit stations to their destinations. Vehicle 1 and 4 pick up one group of passengers and drop of them and then serve another group of passengers. Vehicle 2 pick up 2 groups of passengers at the transit station together, and then drop off them separately, which is a ride-sharing trip. The optimal objective value of the last-mile part is 216. The optimal ridesharing chain for the last-mile problem is shown in Figure 2b

Table 3 First-mile problem results in the sample network

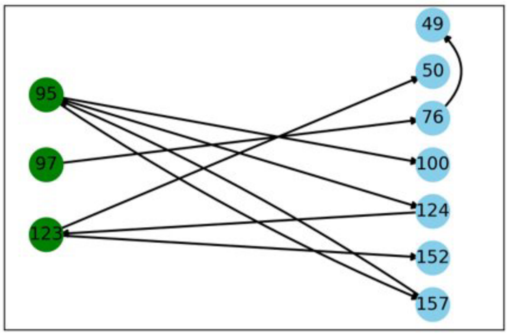
Vehicle ID	Request path	Actual trip path	Pickup time of first client	Trip Type
1	5-12	33-97	16	Non-ridesharing
2	4-11	127-97	30	Non-ridesharing
3	7-14	94-123	38	Non-ridesharing
4	3-2-10-9	79-98-95	15	Ridesharing
5	6-13-1-8	77-123-126-95	13	Non-ridesharing

Table 4 Last-mile problem results in the sample network

Vehicle ID	Request path	Actual trip path	Pickup time of first client	Trip Type
1	1-8-7-14	95-124-123-152	9	Non-ridesharing
2	5-4-12-11	97-76-49	13	Ridesharing
3	6-13	123-50	2	Non-ridesharing
4	3-10-2-9	95-157-95-100	3	Non-ridesharing



(a)



(b)

Figure 2 Optimal results of (a) first-mile problem; (b) last-mile problem of the sample network

When we use the Constructed Heuristic Algorithm to solve the first-mile problem, we cannot get the optimal results. Table 5 and Figure 3a present the heuristic results of the first-mile problem. The objective value is 252 and is greater than 227 of the optimal solution. This is because, in the algorithm, we only merge trips with the same destination. Only the requests on nodes 79 and 98 are merged and this would be the ride-sharing trip. Compared to the optimal solution, the fifth vehicle picks up passengers at node 77 and drop off them at the transit station, and then picks up another group of passengers at node 126 and drops off them at the transit station. But now, we need two vehicles to serve the passengers at nodes 77 and 126 respectively. Using the heuristic method, we need 6 vehicles in total to serve the first-mile requests.

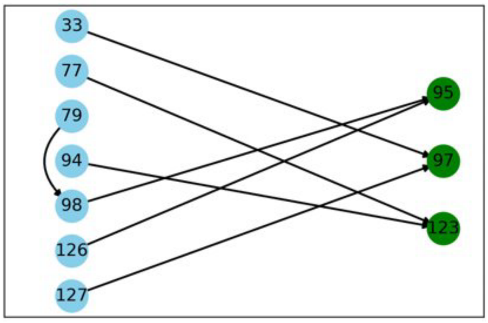
For the last-mile problem, the ridesharing chain result using Constructed heuristic Algorithm is presented in Table 6. The objective value is 237.5 which is greater than the optimal value at 216. In this heuristic result, we need 6 vehicles to serve the 7 trip requests. Note that the first vehicle will pick up riders from node 95 with Trip ID 8 and 9 at once because their request starts due to a relatively close time window and drop them at each destination. All the vehicle is matched to the riders within the time window.

Table 5 First-mile problem results in the sample network

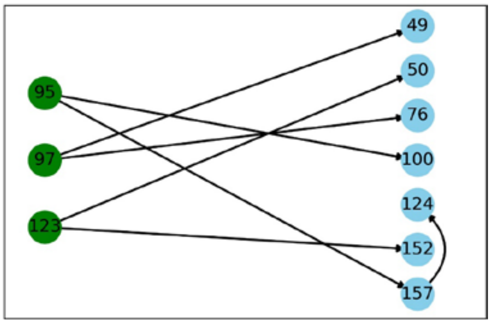
Vehicle ID	Request path	Actual trip path	Pickup time of first client	Trip Type
1	5-12	33-97	16	Non-ridesharing
2	4-11	127-97	30	Non-ridesharing
3	7-14	94-123	38	Non-ridesharing
4	3-2-10-9	79-98-95	15	Ridesharing
5	6-13	77-123	13	Non-ridesharing
6	1-8	126-9	40	Non-ridesharing

Table 6 Last-mile problem results in the sample network

Vehicle ID	Request path	Actual trip path	Pickup time of first client	Trip Type
1	3-1-10-8	95-157-124	11	Ridesharing
2	2-9	95-100	13	Non-ridesharing
3	4-11	97-49	20	Non-ridesharing
4	5-12	97-76	13	Ridesharing
5	6-13	123-50	17	Non-ridesharing
6	7-14	123-152	39	Non-ridesharing



(a)



(b)

Figure 3 Heuristic results of (a) first-mile problem; (b) last-mile problem of the sample network

These small examples confirm the consistency of our ridesharing problem result at optimality using Gurobi and heuristic result using the modified saving method. In the optimal result, we demonstrate the trip chain of ridesharing and non-ridesharing. Without the ridesharing system in our problem, every trip request will be satisfied by one vehicle. For example, 7 requests will require 7 vehicles. However, as we enforce the ridesharing system, we observe that the need for the vehicle can be reduced. This reduction will lead to an increased vehicle occupancy rate per vehicle which is coherent to reduced congestion and tailpipe emission.

30 Trip Requests NYC Example

We apply the constructed algorithm to solve the first-mile ridesharing problem with 30 trip requests from NYC. We demonstrate that our modified saving algorithm can be implemented in a large network. The results are shown in Table 8. Without considering the ridesharing system, we need one vehicle to serve one request. The total transportation cost of the non-ridesharing system is 1314.6. However, if we enforce the ridesharing system, the costs of the ridesharing plan can reduce the objective value to 833.6. Additionally, in total, only 16 vehicles are needed to satisfy the first-mile requests, which leads to lower vehicle total trips and higher vehicle occupancy rates. This result shows that the ridesharing system provides benefits in reducing vehicle occupancy rate which is beneficial in supporting sustainability.

Table 7 30 trip requests for NYC example

Trip ID	Pickup Node	Number of Passengers	Pickup Time Window	Drop off Node	Number of Passengers	Drop off Time Window
1	126	2	[40,57]	95	-2	[60,77]
2	98	1	[24,47]	95	-1	[30,53]
3	79	1	[15,50]	95	-1	[33,68]
4	96	1	[37,58]	95	-1	[56,77]

5	29	3	[11,43]	95	-3	[29,61]
6	127	1	[30,45]	97	-1	[44,64]
7	33	2	[16,45]	97	-2	[36,70]
8	53	2	[22,43]	97	-2	[29,55]
9	48	2	[30,52]	97	-2	[46,73]
10	157	3	[18,40]	97	-3	[32,59]
11	49	2	[20,48]	97	-2	[26,59]
12	76	1	[13,35]	97	-1	[26,53]
13	57	3	[22,50]	97	-3	[27,60]
14	55	3	[12,34]	97	-3	[23,50]
15	153	3	[7,38]	97	-3	[26,62]
16	77	2	[12,41]	123	-2	[17,55]
17	94	3	[38,57]	123	-3	[45,73]
18	153	1	[29,49]	123	-1	[46,75]
19	126	3	[1,23]	123	-3	[9,40]
20	50	2	[2,28]	123	-2	[20,55]
21	152	2	[39,67]	123	-2	[59,96]
22	118	3	[2,21]	123	-3	[11,39]
23	154	1	[45,55]	123	-1	[54,73]
24	70	2	[2,20]	123	-2	[13,40]
25	49	1	[15,24]	123	-1	[27,45]
26	48	1	[1,17]	123	-1	[19,44]
27	162	2	[13,30]	123	-2	[27,53]
28	153	3	[2,29]	123	-3	[14,50]
29	49	2	[6,32]	123	-2	[15,50]
30	24	2	[9,42]	123	-2	[17,59]

Table 8 First-mile problem results in the sample network

Vehicle ID	Request path	Actual trip path	Pickup time of first client	Trip Type
1	3-2-33-32	50-47-95	15	Ridesharing
2	5-4-1-35-34-31	43-58-57-95	11	Ridesharing
3	10-6-40-36	40-45-97	18	Ridesharing
4	7-8-37-38	45-43-97	16	Ridesharing
5	13-43	50-97	22	Non-ridesharing
6	14-44	55-97	12	Non-ridesharing
7	15-45	153-97	16	Non-ridesharing
8	12-11-42-41	76-49-97	13	Ridesharing
9	16-30-46-60	77-24-123	13	Ridesharing

10	17-47	94-123	38	Non-ridesharing
11	19-49	126-123	18	Non-ridesharing
12	20-25-29-50-55-59	50-49-123	12	Ridesharing
13	22-52	118-123	8	Non-ridesharing
14	26-28-23-56-58-53	48-153-154-123	10	Ridesharing
15	27-57	162-123	19	Non-ridesharing
16	24-21-18-54-51-48	70-152-153-123	7	Ridesharing

CONCLUSIONS

In this project, our objective is to model the First-mile and Last-mile Problem as an effort of supporting sustainable transportation. We model the First-mile and Last-mile problems as separable problems and formulate them as Capacitated Vehicle Routing Problem with Time Window (CVRPTW). With the consideration of transportation costs, extra riding time due to ridesharing, and waiting time, we construct a multi-objective optimization model to find an optimal or a feasible route of ridesharing system. We verify the consistency and optimality of our formulation with a small example using Gurobi. Gurobi cannot solve the First-mile and Last-mile Problem of ridesharing systems in a large network within a reasonable time. Thus, we propose the constructed heuristics approach (modified saving method) to solve the CVRPTW in a larger network.

From the results, we can observe that as we enforce the ridesharing system, the total vehicles for the trip requests can be reduced. For the 30 trip requests, we only require having 16 vehicles to serve the trips. This reduction will lead to an increased vehicle occupancy rate per vehicle which is coherent to reduced congestion and tailpipe emission.

Lastly, there are several potential extensions that we could address in the future. First, our approach solves the first- and last-mile problem separately. Such an approach leads to non-effective vehicle operation, in which we force every vehicle to end the trip to a dummy node and forbid vehicles to pick up new requests once they complete the trip chain. These assumptions are not effective in the business application, where it is possible for a vehicle to keep taking the new request from the drop-off point. Our CVRPTW model formulation is capable to solve the first- and last-mile problem as one composite problem. We could easily improve it by setting the first- and last-mile problem into a trip network and solving it in one shot for future extension. The other possible improvement is to apply the meta-heuristic algorithm, e.g., Genetic Algorithm and Simulated Annealing, to solve the problem strategically and guarantee the satisfaction level of the heuristic result compared to the constructed heuristics approach.