Research on Online Car-Hailing Scheduling Problem Based on Resource Allocation

Rongyuan Chen[1], Geng Qian[1], Zhixiong He[1], Yuanxing Shi[1] & Minmin Hu[1]

[1] Institute of big data and Internet innovation, Hunan University of Technology and Business, China

Correspondence: Geng Qian, Institute of big data and Internet innovation, Hunan University of Technology and Business, China. E-mail: 515756288@qq.com

Abstract

In recent years, with the rapid development of Internet technology, various online car-hailing platforms (Uber, DiDi, CaoCao, etc.) have swept the world. The convenient and fast information exchange of "Internet +" greatly alleviates the information obstruction caused by the space-time deviation between passengers and drivers in the traditional taxi system, and becomes a powerful tool to improve the efficiency of the taxi market. In order to improve the utilization of resource allocation, this paper designs a model based on the vehicle's empty driving distance, and uses the Hungarian solution to obtain the shortest vehicle's empty driving distance to optimize the resource allocation problem. Then, according to the position coordinates and the travel route of the vehicle, the vehicle information of eleven times a day is calculated, and it is found that the result obtained by the algorithm is ideal；At the same time, we design the subsidy pricing model and its scheme. Finally, we make a systematic summary of the research content of this paper and look forward to the possible research directions in the future.

Keywords: online car-hailing scheduling, resource allocation, hungarian algorithm, empty driving distance

1. Introduction

Clothing, food, shelter and transportation have always been the most concerned topics for the people. With the development of national economy and the continuous improvement of living standards, people put forward higher requirements for quality travel. Comfort, convenience, safety and appropriate cost have become part of the main factors that people consider when they travel. However, travel difficulties are often caused by various factors in large and medium-sized cities with a population of more than 2 million. For example, it is difficult to take a taxi in the peak period. It often happens that taxi drivers choose whether to pick up the passenger or not according to your destination, refuse to take passengers, share passengers and take detours; The public transport system is overcrowded and uncomfortable; Private car travel cost is high, parking is difficult; The overall traffic situation in the city is rather congested, etc. These factors affect people's travel experience.

In recent years, China's information and communication technology has been greatly developed, 4G wireless network has covered many cities across the country, smart phones have also been greatly popularized in China. According to data released by the media in the first quarter of 2016, China's smartphone users have exceeded 624 million. The development of information and communication technologies and smart phones, together with the new thinking of the "Internet +" entrepreneurial model proposed by the state, has spawned a new business model--O2O. This business model has achieved rapid development and great success in a short period of time. The online car-hailing platform is one of the typical cases of success in the O2O model. In 2012, Mr. Cheng wei founded Beijing xiaoju technology co., ltd. and launched the "DiDi taxi" APP. In order to meet the needs of more people and further reduce people's travel costs, "DiDi express" officially launched on the "DiDi taxi" APP on May 7, 2015, and quickly became popular in China with advantages such as being close to taxi prices and higher than taxi experience. By August 2016, "DiDi express" had covered more than 300 cities in China, with daily orders reaching 14 million, more than 15 million registered drivers and more than 300 million registered users, which has optimized people's travel experience, driven economic development and solved some employment problems. At 3 o'clock on the afternoon of July 28, 2016, the State promulgated the "Guiding Opinions on Deepening Reform and Promoting the Healthy Development of the Taxi Industry" and the "Interim Measures for the Administration of Network Appointment of Taxi Operation Services" to make the online car-hailing officially have legal status. It has also become the first country in the world to officially promulgate the legal terms of the

online car-hailing, which is also an improvement of the state to encourage the sharing of the economy. The integration of online car and traditional parade taxis will help transform and upgrade traditional parade taxis and provide a better service experience.

For enterprises, profit is their fundamental purpose. However, for a long time, the biggest marketing method that drip drivers and passengers use their platform is the investment of real money, the driver gives subsidies and rewards, and the passengers give a lot of vouchers and discount coupons, which lowers the entire platform. Profitability, even for quite a while, Didi is losing money, its purpose is to increase reputation. From 2012 to August 2016, Didi has undergone several mergers and financings. Tencent and Alibaba have burned more than 50 billion yuan together, and they still have not made it into a profit stage. Investors have not received any gains. After the price hike in August 2016, most cities started to stop losing money and gradually entered the profit mode. However, according to the data released by relevant institutions, in August 2016 alone, the number of orders dropped by more than 10%, and the number of active users also lost 700,000. Is it that the amount of business without subsidies or discount coupons will be reduced? How to use more effective means to expand its business volume in the profit model? How can we more fully interface and maximize the use of information from service providers and service providers to enhance the value of their service marketing? Waiting for some of the marketing-related issues facing Didi need further research. This paper aims to solve the problem of single stimulation mode and low user loyalty in online car-hailing platform. Practical analysis was carried out on such problems as unreasonable allocation of resources matching supply and demand, lack of pertinences in subsidy distribution, etc., to optimize the empty driving distance of vehicles and passenger waiting time, and to provide theoretical basis for optimization subsidy scheme of online car-hailing platform.

2. Literature Review

Savelsbergh and Sol (1995) raised the " GPDP,pick-up and delivery problem ", Refers to the design of a set of routes to meet the given load, departure point and destination of the transport requirements. Each vehicle that performs these routes has a specific starting point and capacity limit. By ride-taking problem is a special form of transportation problem, the most basic ride-taking problem is as a set with the same starting point and destination of homogeneous vehicle design route and schedule, to ferry a certain quantity of a request for a service homogeneity passengers, passenger ride-taking problem is usually applied to transport to the elderly or disabled for the purpose of system.

In order to make the loading problem more applicable, many scholars have studied the The dial-a-ride problem (DARP) variant problem which is more widely applicable. Psaraftis (1983) and so on to minimize the service time of all customers, considering the time window of picking up and dropping passengers, each passenger has a designated pick-up time upper and lower bounds, designed and improved the corresponding dynamic planning algorithm. Jaw et al. (1986) aimed at minimizing the ride time, also considered the time window for picking up and receiving passengers, and designed a heuristic algorithm for this problem. Cordeau and Laporte (2007) summarized relevant literature on the issue of DARP and put forward the issue of Multiple Depot dial-a-ride problem (MD-DARP). Aiming at minimizing the transportation cost, Carnes et al.(2013)[6] studied the air rescue service, a major part of which was the advance delivery and transportation of medical facilities. By developing a series of complex constraint planning tools, they distributed the requests received in advance to each aircraft, greatly saving the time of flight scheduling. In real life, passengers may have different requirements (such as wheelchair transport) and service providers may schedule different types of vehicles in multiple yards, Braekers et al. (2014) introduced a kind of loading problem that can simultaneously consider multiple yards, heterogeneous passengers and heterogeneous vehicles, and proposed a universal branch-cutting algorithm, which can accurately solve the small problem of MD-H-DARP. In order to solve the larger problem instance, a new deterministic annealing element heuristic algorithm is proposed. Some scholars have also considered the transfer problem, which allows passengers to change vehicles at a fixed transfer point during the journey. Masson et al. (2014a)[8] limited the transfer times to 1 and proposed a solution method based on adaptive large neighborhood search element heuristic algorithm. Review existing literature, we think we can extend the "ride-taking problem" to the taxi problem in our daily life, the model is established with the objective function of minimizing the empty driving distance of vehicles, based on the solution methods based on the above literatures and the actual situation of this paper, a Hungarian algorithm is proposed to solve this model.

The essence of online car-hailing scheduling problem is very similar to the "DARP", such as MD-DARP yard and heterogeneous passengers and vehicles. It can be regarded as a generalization of the classic "DARP" in the transportation generalization. Different from the DARP, in the online car-hailing scheduling problem problem, the number of vehicles and the starting location are not fixed. In addition, for the service provider of the network car, the online car-hailing platform, the network car passengers and Net car drivers are customers, so the

assessment of the quality of the car service depends not only on the passengers, but also on the driver.

3. Problem and Model Construction

3.1 Problem Description

This paper aims at the online car-hailing platform serving a specific region as a scheduling system, and makes an appropriate scheduling strategy for it considering the travel utility of passengers and drivers, so as to maximize the resource utilization rate of the online car-hailing platform. For the driver, when the utility of a single trip is greater than his expected profit, he is willing to accept the orders assigned by the platform; For passengers, when the utility of a single trip is greater than the cost of such a single trip, she is willing to obey the distribution of the platform. Fully considering the utility of the trip of passengers and drivers can help improve user loyalty and the overall matching rate, and theoretically increase the operating income of the online car-hailing platform. However, in order to maintain the balance between supply and demand of the whole service area of the online car-hailing platform and the reasonable income level of the online car-hailing drivers, the investment amount will be increased accordingly. Therefore, an appropriate scheduling strategy should not only encourage passengers and drivers to accept orders assigned by the online car-hailing platform, improve the matching rate, but also help to improve the overall operating income of the online car-hailing platform.

The online car-hailing problem can be described as follows: Matching the vehicle with the passenger in an area, the vehicle's driving track and the time of getting on and off the vehicle are known, and the vehicle's driving speed is all uniform. The goal is to give the optimal vehicle scheduling and the matching scheme of supply and demand with the passenger under the condition of minimizing the vehicle's empty driving distance.

To clarify the scope of application of the study, make the following assumptions: (1) The vehicles are of the same type, and the distance from when the passenger gets off the bus to when the next passenger gets on the bus is empty distance; (2) According to the actual situation, the trajectory of different vehicles varies; (3)A car only serves one passenger or a group of passengers, and does not consider carpooling for the time being; (4) Each passenger has only one car to serve them; (5) When the vehicle arrives at the passenger, it gets on the train; (6) Vehicle use cost includes vehicle use time cost, labor cost and fixed departure cost.

3.2 Symbols and Variables

Notation	Explanation
i	The i-th car
j	The j-th passenger
Z	Total empty distance of all vehicles (km)
D_{ij}	Empty driving distance assigned car i to pick up passenger j (km)
S_{ij}	The driving distance of the j passenger in the i car
x_{ij}	Decision variable, 1 when the i car goes to pick up the j passenger, and 0 when the i car does not pick up the j passenger.
d_{ij}	The shortest empty driving distance (km) for assigning the i car to the j passenger
h	Starting price of the vehicle (yuan)
q	Unit mileage price beyond the starting price (yuan/km)
f	Fuel consumption per unit distance (liter)
k	Expect coefficient
λ_f	Unit fuel consumption cost (yuan/liter)
C	Range of vehicle starting price (km)
N	Vehicle income (yuan)
$b\%$	The platform gives the driver a percentage, $(1-b\%)$ is the proportion of the platform
P_{ir}	The total income of the i car in the actual state of the driver (yuan)
P_{ie}	The total income of car i (yuan) under the ideal condition of the driver (each order is the shortest empty driving distance)
P_{if}	Total income under the dynamic profit distribution after optimization (yuan)

3.3 Mathematical Model

This paper designs a model based on the 0-1 planning based on the characteristics of the "ride-taking problem", 0-1 planning is a special form of integer programming. The decision variables of this plan only take the value 0 or 1, so they are called 0-1 variables or binary variables, because a non-negative integer can be expressed in binary notation with several 0-1 variables. Variables 0-1 can quantitatively describe logical relationships, sequential relationships, and mutually exclusive constraints between discrete variables, such as on and off, take and abandon, presence and absence, etc, Therefore, the 0-1 planning is very suitable for describing and solving line design problems and assignment problems. The objective function is the total empty driving distance of all independent vehicles.

Suppose the distance from car i to passenger j is D_{ij}, The design variable in the problem is x_{ij}, Its meaning is as follows:

$$x_{ij} = \begin{cases} 1 & \text{When the } i \text{ car goes to pick up the } j \text{ passenger} \\ 0 & \text{When the } i \text{ car does not pick up the } j \text{ passenger} \end{cases}$$

Then the objective function of the resource configuration is:

$$minZ = \sum_{i=1}^{m}\sum_{j=1}^{n} D_{ij}x_{ij} \tag{1}$$

$$s.t. \sum_{j=1}^{n} x_{ij} = 1 \quad (i=1,2,...,n) \tag{2}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (j=1,2,...,n) \tag{3}$$

$$x_{ij} = 0 \text{ or } 1 \quad (i=1,2,...,n; j=1,2,...,n) \tag{4}$$

The objective function formula (1) represents the idle distance of all minimized vehicles or the minimum time the passenger waits; Equations (2) and (3) indicate that each passenger can only be visited once; Equation (4) is expressed as having a car or carless service status.

3.4 Vehicle Subsidy Pricing Model and Plan

To alleviate the problem of "difficulty in taking a taxi", we can subsidize drivers to take a single order. The subsidy can not only be accepted by the majority of consumers, but also ensure that the driver's income is above the average level, so as to better serve passengers. In order to provide theoretical basis for the optimization of subsidy scheme for online car-hailing companies, the income function of taxi drivers for single passenger transportation is given as follows:

The driver's actual situation is a fixed profit distribution plan：Basic salary + percentage commission

$$N = \begin{cases} h & , \quad S_{ij} \leq C \\ \sum_{i \in N}\sum_{j \in N}(S_{ij}-C)q \times x_{ij} + h & , \quad S_{ij} > C \end{cases}$$

The driver's ideal income:

$$P_{ir} = N \times b\% - \sum_{j \in N} D_{ij} \times \lambda_f \times f \quad , \quad i \in N;$$

The actual income of the driver:

$$P_{ie} = N \times b\% - \sum_{j \in N} d_{ij} \times \lambda_f \times f \quad , \quad i \in N;$$

The expected difference between the two models:

$$E = P_{ie} - P_{ir} \quad , \quad P_{ie} > P_{if} > P_{ir}$$

$$P_{if} = P_{ir} + kE$$

$$k = \frac{D_{ij}}{D_{ij} + S_{ij}}$$

This model compares the income of the vehicle in the ideal state with the income in the actual state. According to this model, the enterprise can use its own internal data to make a budget for the subsidy of drivers, and then make adjustments to give subsidies according to the actual situation.

4. Hungarian Algorithm

Application object of the Hungarian algorithm: Can be used to solve the assignment problem, the assignment problem is a special form of integer programming. The problem can be summarized as follows: there are n tasks that need n people to complete, each person can only complete one of them, and each task can only be completed by one person. In the problem, the expertise and work efficiency of each person will be given in various forms. The problem that needs to be solved is to consider which person to assign to which task to achieve the highest total efficiency or the shortest total time spent. The Hungarian algorithm is an algorithm to solve the optimization problem whose optimal model can be reduced to the assignment problem. Procedure is as follows:

```python
import pandas as pd
import numpy as np
from scipy.optimize import linear_sum_assignment
from math import radians, cos, sin, asin, sqrt
def haversine(lon1, lat1, lon2, lat2):
    lon1, lat1, lon2, lat2 = map(radians, [lon1, lat1, lon2, lat2])
    dlon = lon2 - lon1
    dlat = lat2 - lat1
    a = sin(dlat/2)**2 + cos(lat1) * cos(lat2) * sin(dlon/2)**2
    c = 2 * asin(sqrt(a))
    r = 6371
    return c * r * 1000
def readValues(filePath):
    f = open(filePath)
    allCCC = pd.read_csv(f,index_col=0)
    #print(allCCC)
    for i in range(0,len(allCCC),2):
        index1,index2 = allCCC.index[i], allCCC.index[i+1]
        A                                                                        =
allCCC.at[index1,"LONGITUDE"],allCCC.at[index1,"LATITIUE"],allCCC.at[index2,"LONGITUDE"],allCCC.at[index2,"LATITIUE"],
        listValue.append(A)
        #print("%f,%f,%f,%f"%(a,b,c,d))
        # ddd(a,b,c,d)
    print("Enter the function ")
listValue = []
readValues("D:/2019-02-20.07.30_all.csv")
list_save = []
for i in range(len(listValue)):
    for j in range(len(listValue)):
print("%f,%f,%f,%f"%(listValue[i][0],listValue[i][1],listValue[j][2],listValue[j][3]))
        values = haversine(listValue[i][0],listValue[i][1],listValue[j][2],listValue[j][3])
        #values = conputeDistance(listValue[i][0],listValue[i][1],listValue[j][2],listValue[j][3])
        list_save.append(values)
    # print("*"*50)
    # print("*"*50)
    # print("*"*50)
    # print("*"*50)
    #   print("*"*50)
```

```
# In[157]:
sum = 0
for i in range(len(listValue)):
        #print("%f,%f,%f,%f"%(listValue[i][0],listValue[i][1],listValue[i][2],listValue[i][3]))
        values = haversine(listValue[i][0],listValue[i][1],listValue[i][2],listValue[i][3])
        sum = sum+values
        #values = conputeDistance(listValue[i][0],listValue[i][1],listValue[j][2],listValue[j][3])
        #list_save.append(values
# In[158]:
aa = np.array(list_save).reshape(int(sqrt(len(list_save))),int(sqrt(len(list_save))))
row_ind,col_ind=linear_sum_assignment(aa)
print(row_ind)
print(col_ind)
print(aa[row_ind,col_ind])
print(aa[row_ind,col_ind].sum())
# In[159]:
print("Vehicle number：%d"%len(listValue),end="")
print("\t Actual total distance：%f"%sum,end="")
print("\t Optimized total distance：%f"%(aa[row_ind,col_ind].sum()),end="")
print("\t Average actual empty distance：%f"%(sum/len(listValue)),end="")
print("\t Optimize the average empty distance：%f"%(aa[row_ind,col_ind].sum()/len(listValue)))
```

5. Analysis of Experimental Results

Through the data, we calculated the 7:30, 8:30, 9:30, 10:30, 11:30, 18:30, 7:30, 20:30, 21:30, 22:30, drove the 11 point data, including vehicle number, the total distance, optimize the total distance, average actual empty driving distance, average optimized empty driving distance. Specific data are shown in table 1.

Table 1. Comparison of experimental results

Moment	Number of vehicles (per vehicle)		Actual total distance (m)	Optimize the total distance(m)	Average actual empty driving distance(m)	Average optimized empty driving distance (m)
7：30	29		333560.873012	202643.805425	11502.099069	6987.717428
8：30	36		379930.082748	242771.683528	10553.613410	6743.657876
9：30	65		453168.016024	169639.607673	6971.815631	2609.840118
10：30	95		494775.563037	195609.924801	5208.163821	2059.051840
11：30	121		508316.748316	228485.996819	4200.964862	1888.314023
18：30	74		422258.848084	184393.024634	5706.200650	2491.797630
19：30	27		100323.068413	74933.799823	3715.669200	2775.325919
20：30	99		342935.393766	238976.590188	3463.993876	2413.904951
21：30	81		272929.865153	169996.354683	3369.504508	2098.720428
22：30	84		427558.796860	169317.688354	5089.985677	2015.686766
23：30	45		470369.105161	162162.383387	10452.646781	3603.608520

According to the above data comparison, when the number of vehicles is 29 at 7:30, the average optimized empty driving distance is about 6.9 kilometers; When the number of vehicles is 121, the average optimized empty driving distance is about 1.8 km. It shows that this algorithm can optimize the empty driving distance of vehicles to some extent at each time point, and obtain very good optimization results.
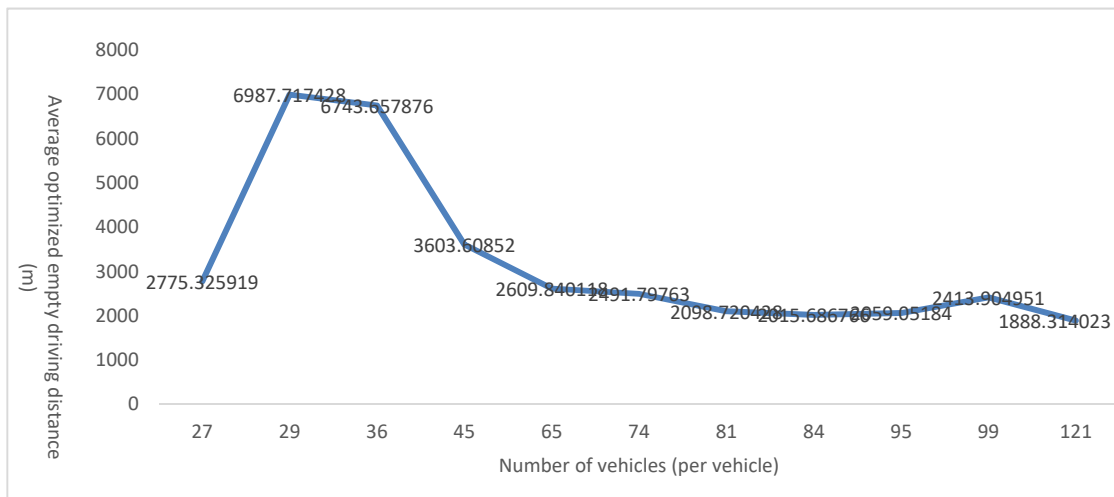
Figure 1. Comparison of average optimized distance under different number of vehicles

It can be seen from Figure 1 that when the Number of vehicles is the abscissa and the Average optimized empty driving distance is the ordinate, as the number of vehicles increases, the average optimized empty driving distance tends to decrease overall, that is, when the number of vehicles increases, the average optimized empty driving distance becomes smaller. This means that more users can be matched, and the empty driving distance of vehicles decreases, which also increases the utilization rate of resources, indicating that this algorithm can get a relatively ideal result.

6. Conclusion and Prospect

This paper studies the online car-hailing scheduling problem based on resource allocation, review previous literature, on the basis of "ride-taking problem", the online car-hailing problem is derived, and a simpler and more effective theoretical method, model and algorithm are found. This paper considers the optimization of platform resource utilization, user trip utility, and empty driving distance, and establishes related mathematical models for this problem, and proposes corresponding solutions, the final result optimizes the empty distance, as a result, the waiting time of passengers is shorter and the number of passengers served by vehicles in the same time is higher; Reduced the driving time without passengers, optimizes the allocation of resources, saves the time for passengers to wait for vehicles, and improves the benefits of drivers; Reduced fuel consumption for efficient vehicle operation and reduced energy consumption(Carbon emissions); At the same time, it also brings corresponding improvement to the profits of enterprises. Therefore, this paper optimizes the empty driving distance of vehicles to a certain extent and analyzes and designs the subsidy scheme for the resource allocation of online car-hailing. This will be of certain reference for the study of the allocation of online car-hailing resources in the " Internet +" era.

However, as the ride-hailing market is an emerging bilateral market, it contains a wide range of contents, involves many levels and involves complex factors, which need more detailed and in-depth research. In this paper, the directions to be further studied are as follows:

1) In this paper, when considering the matching forms of passengers and drivers in the online ride-hailing market, only the traditional one-to-one matching between passengers and drivers is considered. However, due to the rise of sharing economy, there are many matching forms similar to the one-to-many or many-to-many matching forms such as pooling and hitchhikers, so the model in this paper can be further extended.

2) When considering the passenger and driver's itinerary, this paper only studies the impact of the matching passenger and driver's travel utility on the operating profit and scheduling scheme of the network car platform, and does not consider the potential loss caused by the lost passengers and idle drivers. Can be further studied for the effectiveness of unmatched passengers and drivers.

3) The heuristic algorithm used in this paper is a basic Hungarian solution. It can be further improved with the use of precise algorithms such as branch and bound algorithm, Benders decomposition algorithm or

other heuristic algorithms such as genetic algorithm and tabu search algorithm, in order to achieve higher efficiency.

References

Braekers, K., Caris, A., & Janssens, G. K. (2014). Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots. *Transportation Research Part B: Methodological, 67*, 166-186. https://doi.org/10.1016/j.trb.2014.05.007

Carnes, T. A., Henderson, S. G., & Shmoys, D. B. et al. (2013). Mathematical Programming Guides Air-Ambulance Routing at Ornge. *Interfaces, 43*(3), 232-239. https://doi.org/10.1287/inte.2013.0683

Cordeau, J. F., & Laporte, G. (2007). The dial-a-ride problem: models and algorithms. *Annals of Operations Research, 153*(1), 29-46. https://doi.org/10.1007/s10479-007-0170-8

Jaw, J. J., Odoni, A. R., Psaraftis, H. N. et al. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research, Part B (Methodological)*, *20*(3), 243-257. https://doi.org/10.1016/0191-2615(86)90020-2

Masoud, N., & Jayakrishnan, R. (2017). A decomposition algorithm to solve the multi-hop Peer-to-Peer ride-matching problem. *Transportation Research Part B: Methodological, 99*, 1-29. https://doi.org/10.1016/j.trb.2017.01.004

Masson, R., Lehuédé, F., & Péton, O. (2014). The Dial-A-Ride Problem with Transfers. *Computers & Operations Research, 41*, 12-23. https://doi.org/10.1016/j.cor.2013.07.020

Psaraftis, H. N. (1983). An Exact Algorithm for the Single Vehicle Many-to-Many Dial-A-Ride Problem with Time Windows. *Transportation Science, 17*(3), 351-357. https://doi.org/10.1287/trsc.17.3.351

Savelsbergh, M. W. P., & Sol, M. (1995). The General Pickup and Delivery Problem. *Transportation Science, 29*(1), 17-29. https://doi.org/10.1287/trsc.29.1.17