Application of tabu search method in solving a taxi-sharing problem

View the article online for updates and enhancements.

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Application of tabu search method in solving a taxi-sharing problem

R N Hadiarti[1], Y Satria[1] and H Burhan[1]

[1] Department of Mathematics Universitas Indonesia, Depok, 16424, Indonesia

E-mail : revi.nurfathhiyah@sci.ui.ac.id, ysatria@sci.ui.ac.id, helen.burhan@sci.ui.ac.id

Abstract. Nowadays, a ride-sharing system is a trend among society for traveling. The ride-sharing system is a solution that can be developed to reduce the congestion because of the high amounts of the vehicle on the road. Taxi as an alternative transportation in an urban area can impose the ride-sharing system. Taxi-sharing aims to maximize the utilization of taxi capacity, thereby reduces the fare for passengers, increases the income for taxi operator, and reduces congestion, gas emission, as well as fuel consumption. In order to maximize the benefits of the taxi-sharing system usage, we need to optimize taxi routes and match requests that share taxi service. In this paper, we used a mixed integer programming problem as in Hosni et al (2014) to make a model of optimization of the taxi-sharing problem, then solved the problem by using a tabu search method. The experiments showed that the tabu search method could increase the income of taxi operator up to 10 - 14 %.

1. Introduction

Taxi is a public transportation which provides point to point service in an urban area [1]. The high amounts of a car with taxi services utility that is low will impact traffic load which has high quantities of the car. If the quantities of the car are increased but the road capacity is limited, congestion and its impacts such as fuel wasting and air pollution could happen [2]. Therefore, we need to optimize taxi services usage by applying a taxi-sharing system which is adapted from a ride-sharing system in private vehicles.

The taxi-sharing system unites two or more separate taxi service requests that have similar characters to share taxi service [2]. Requests have the same or nearby destinations in the same or almost the same travel time [3]. The taxi-sharing system has several advantages, such as reduces passengers travel costs because a passenger share taxi services with another passenger and also reduces taxis' operating costs because a taxi will be able to take multiple requests simultaneously [2].

Requests' character of taxi service could be static or dynamic. Dynamic requests allow a new passenger to enter the system (taxi routes) that have been made, while the static one does not [4]. In this paper, the taxis' speed was static and the requests were semi-dynamic by dividing taxis service time periodically. Information of requests was available before taxis started its services. A request was allowed to have delivery time in a different period with the pickup time. So, before the taxi started its service in a time period, it finished the services from the previous period.

In order to maximize the benefits of the taxi-sharing system, we need to optimize taxi routes and match requests that share taxi service for maximizing taxi operator's income and minimizing passengers' fare [4]. This optimization is known in the literature as a dial a ride problem (DARP) which is a problem of transportation requests from specific pickup locations to specific drop off

locations while imposing certain requirements, such as earliest pickup time and latest drop off time of passengers, the capacity of vehicles and precedent requests [5]. In this paper, the taxi-sharing problem was formulated as a mixed integer programming model. Furthermore, by the program, we made sure that whenever passenger shared the taxi service, they would pay less than when they did not. Then the model solved by using tabu search method to find optimal routes and passengers matching for increasing taxi operator income.

The rest of the paper is organized as follows: Section 2 literature review and Section 3 describes problem formulation, Section 4 and Section 5 presents solution methodology and its computational experiment, and Section 6 gives the conclusion of the research.

2. Literature review

Ride-sharing is a trend among society for traveling which is a solution that can be developed to reduce the congestion because of high amounts of a vehicle on the road. In 2013, Furuhata et al. explained the state of the art and future directions of a ride-sharing system [3]. Taxi as an alternative transportation in an urban area can impose that ride-sharing system, thus Hosni et al. made formulation of taxi sharing problem and its solution by using heuristics method in 2014 [4]. The following years researches about taxi sharing were done such as modeling urban level impact of shared taxi market by Paraboschi et al. (2015) [6], an open data approach for quantifying the potential of taxi sharing by Baraan et al. (2017), and understanding the effects of taxi sharing as a case study in Singapore by Wang et al. (2018) [1].

A lot of DARP had been solved by using a tabu search method and its variation. Nanry & Barnes solved the pickup and delivery problem with time windows by using reactive tabu search method in 2000 [7]. As a generalization of pickup and delivery problem, the static multi-vehicle DARP had been tackled by applying tabu search heuristic of Cordeau & Laporte (2003) [8]. The next year, parallel tabu search was utilized for solving an upgraded problem, the dynamic multi-vehicle DARP of Attanasio et al.[9]. Another variation is a granular tabu search algorithm which was used for solving DARP by Kirchler & Calvo (2013) [10].

The contribution of this paper is a development of problem formulation of taxi sharing which is discussed in Hosni et al. (2014) [4] and an implementation of a tabu search method for the problem. Note that if passengers are sharing the taxi service, their travel fare should be less than their private ride cost. The remainder characteristics of the problem are defined below [4]:

- Taxi service request could be refused.
- Passengers are picked up and dropped off by the taxi at their corresponding locations.
- The ride time of passengers does not exceed their maximum ride time.
- The pickup and drop off must always happen on passengers' specified time windows.
- The amounts of a passenger on the taxi do not exceed the maximum taxi capacity.

The next section presents the mixed integer programming formulation of the problem.

3. Problem formulation

In this paper, the taxi sharing problem aimed to optimize the taxi operator's income because the taxis could serve more passengers in certain time period rather than serve one request at a time. The taxi operator's income would be increased if the operator could match the requests and choose the taxis' route properly. The optimization of the taxi sharing problem was formulated as mixed integer programming as in Hosni et al. (2014) [4]. Taxis could be in different start location and have different capacity. Then, the taxi could also refuse a request that is assumed to be unprofitable.

Road network representation in the problem will be represented as a graph with vertices are representing taxi or requests location and arcs are representing the roads which connect one location to the others. Given graph $G = (V, A)$ with V is set of vertices dan A is set of arcs. Set V contains subsets V_1, V_2, V_3 and V_4. V_1 is the subset of taxi locations by means of $v(t)$ as the vertex of t, V_2 is the subset of drop off locations of onboard passengers, V_3 is the subset of pickup locations of seekers and V_4 is the subset of drop off locations of seekers.

In the taxi sharing problem, given set of taxi T, set of onboard passengers O, set of seekers S, and set of onboard passengers and seekers P. The notation of taxi which serves passenger p is $t(p)$. Passenger p has pick up point $s(p)$ and drop off point $f(p)$. Then, taxi t has the maximum capacity of W^t. Passenger p is associated with earliest pickup time $s(p)$, latest drop off time $f(p)$, and maximum ride time as U_p^e, U_p^l, dan $\hat{U}_p$. Taxi from vertex i to vertex j through an arc (i,j) takes minimum time $\bar{U}_{ij}$ and spends cost C_{ij}. Taxi t arrived at vertex i at u_{ti} and has delay time b_i for picking up and dropping off passengers in vertex i. Note that passenger fare for each arc is equally divided among the passengers on the taxi.

This formulation use x_{ij}^{pt}, y_{ij}^t, and d_{pt} as decision variables. x_{ij}^{pt} worth 1 if passenger p from vertex i to vertex j through an arc (i,j) by using taxi t and 0 otherwise. Then y_{ij}^t worth 1 if taxi t from vertex i to vertex j through an arc (i,j) and 0 otherwise. While d_{pt} worth 1 if taxi t serve passenger p and 0 otherwise.

The formulation of the taxi sharing problem is presented as follows:

$$\max \sum_{p \in S} R_p \sum_{t \in T} d_{pt} - \sum_{i \in V} \sum_{j \in V} C_{ij} \sum_{t \in T} y_{ij}^t \tag{1}$$

s.t.
$$\sum_{j \in V} x_{v(t(p))j}^{pt(p)} = 1, \forall p \in O \tag{2}$$

$$\sum_{j \in V} x_{jf(p)}^{pt(p)} = 1, \forall p \in O \tag{3}$$

$$\sum_{j \in V} x_{js(p)}^{pt} = \sum_{j \in V} x_{s(p)j}^{pt} - d_{pt}, \forall p \in S, \forall t \in T \tag{4}$$

$$\sum_{j \in V} x_{jf(p)}^{pt} = \sum_{j \in V} x_{f(p)j}^{pt} + d_{pt}, \forall p \in S, \forall t \in T \tag{5}$$

$$\sum_{j \in V} x_{ji}^{pt} = \sum_{j \in V} x_{ij}^{pt}, \forall p \in P, \forall t \in T, \forall i \in V - V_1 - \{s(p)\} - \{f(p)\} \tag{6}$$

$$\sum_{p \in P} x_{ij}^{pt} \le W^t y_{ij}^t, \forall t \in T, \forall i \in V, \forall j \in V \tag{7}$$

$$\sum_{t \in T} d_{pt} \le 1, \forall p \in S \tag{8}$$

$$\sum_{j \in V} y_{ji}^t \le 1, \forall t \in T, \forall i \in V \tag{9}$$

$$\sum_{j \in V} y_{ij}^t \le 1, \forall t \in T, \forall i \in V \tag{10}$$

$$\sum_{j \in V} y_{ij}^t \le \sum_{j \in V} y_{ji}^t, \forall t \in T, \forall i \in \{V_3 \cup V_4\} \tag{11}$$

$$u_{tj} - u_{ti} \ge \bar{U}_{ij} - M(1 - y_{ij}^t), \forall t \in T, \forall i \in V, \forall j \in V \tag{12}$$

$$u_{ts(p)} \ge U_{s(p)}^e d_{pt}, \forall p \in S, \forall t \in T \tag{13}$$

$$u_{tf(p)} \le U_{f(p)}^l d_{pt}, \forall p \in P, \forall t \in T \tag{14}$$

$$u_{tf(p)} - u_{ts(p)} \leq \widehat{U}_p + (1 - d_{pt})M, \forall p \in P, \forall t \in T \tag{15}$$

$$x_i^{pt} \in \{0,1\}, y_i^t \in \{0,1\}, d_{pt} \in \{0,1\}, u_{ti} \geq 0, \forall i \in V, \forall j \in V, \forall p \in P, \forall t \in T \tag{16}$$

The optimization function (1) aims to maximize income for the taxi operator. Constraint (2) ensures that onboard passengers leave their current location while Constraint (3) ensures that onboard passengers arrive at their corresponding destinations. Furthermore, Constraint (4) and Constraint (5) ensure that taxi picks up and drops off passengers at their corresponding locations. Constraint (6) is about the conservation of flow. Constraint (7) ensures that the amounts of the passenger in the taxi do not exceed its capacity and Constraint (8) ensures that each passenger is picked up by at most one taxi. Constraint (9) and Constraint (10) ensure that a taxi cannot arrive at a certain point from two different points or head to two different points at the same time. Constraint (11) forces the path continuity for the taxis. Constraint (12) tells the time when taxi reaches a point while Constraint (13) and Constraint (14) ensure that passengers are picked up and dropped off on their time windows. Constraint (15) ensures that travel time for each passenger do not exceed their maximum travel time and Constraint (16) forces the binary and non-negativity conditions.

4. Solution methodology

In this paper, our approach for solving the problem was using tabu search method which is a method that was used for solving a vehicle routing problem founded by Glover in 1986. The processes of finding a solution in this method are in the neighborhood of an initial or current solution. This method allows choosing a new solution which is not better than the previous solution (non-improving move). To prevent cycling back to the previously visited solutions, we use memories called tabu list. The meaning of tabu here is a prohibition to choose the solutions which are on the tabu list. [11][12][13]

The process starts with a set of taxis' routes as an initial solution by assigning a request to a random taxi index until no more request is able to be assigned to the taxi. Then, we build a neighborhood structure of the initial solution by switching two requests from two different routes. Upon the neighborhood, choose the best solution and save the value in the tabu list. If that solution is better than the current solution, set the solution as the current solution. Repeat the process of building neighborhood structure and choose its best solution until certain iteration is fulfilled. In this paper, the number of iterations was 300.

In order to choose the best solution of the neighborhood area, we must check if the solution is tabu or not. If it is not, set the solution as the best solution. Furthermore, we are allowed to enlarge neighborhood structure if there is not any solution that could be chosen in the neighborhood area.

5. Experimental Results

The implementation program in this paper was using C language with Code::Blocks 17.12 and TDM-GCC-64 as the software and compiler. The hardware had processor Intel Core i3 CPU @ 2.00GHz (4CPUs), 2.0GHz with 4096MB RAM and Windows 10 64-bit operating system. The data were generated by using another program besides the main program.

The experiments were using the graph with 512 vertices and 1457 arcs. Taxi service time in three periods from 07.00 until 10.00 with the duration of each period was 60 minutes. In each period, there were 30, 80, 150 taxis and 60, 160, 300 requests. Each request contained either one passenger or two passengers. Taxis capacity was 4 people with the speed 2 minutes for one km. Duration for picking up and dropping off passengers at each vertex was two minutes. The operational cost of the taxi was Rp1.000 per km while passengers fare was Rp3.500 per km. The experiments would give route of taxis, for example, a taxi picked up request A, picked up request B, dropped of request B, and then dropped of request A. These tables below showed experiment result of 10 experiments for the same data:

Table 1. Optimization Values for Each Period in Rupiah

Experiment	Period 1		Period 2		Period 3	
	Initial Solution	Final Solution	Initial Solution	Final Solution	Initial Solution	Final Solution
1	673.744	729.538	1.906.499	2.137.170	4.169.691	4.588.628
2	747.737	775.316	1.988.866	2.177.839	4.213.762	4.718.345
3	595.760	682.137	1.951.269	2.262.861	4.226.224	4.712.037
4	682.364	706.218	2.052.588	2.243.449	4.343.082	4.866.227
5	729.018	793.032	1.972.290	2.187.715	4.106.746	4.674.803
6	700.174	747.125	2.028.882	2.265.647	4.154.559	4.835.110
7	640.052	711.873	1.947.665	2.135.089	4.156.347	4.725.322
8	712.417	746.839	1.998.296	2.279.885	4.152.458	4.753.531
9	716.368	769.028	1.949.980	2.287.788	4.205.370	4.758.212
10	691.333	749.932	1.993.070	2.209.205	4.178.185	4.702.705

Overall, Table 1 shows that there was no relationship between the value of the initial solution and the value of the final solution. If the value of the initial solution was the lowest among the other values, it was not guaranteed that the value of the final solution was also the lowest one, for example in the second period, the lowest value of the initial solution and final solution came from the first and seventh experiment. This interpretation is also applied to the highest one.

We noted that the values of Table 1 were irregular because of a random factor by the program for choosing the first route in initial solution, and also because of the selection of two routes for neighborhood structure. However, in general, there was an improvement of values from the initial solution to the final solution for each experiment which was around 10-14%.

Tabu search method worked based on the initial solution, thus requests that were not served in the initial solution would remain not served. The number of operated taxis and served requests were stable for each experiment. All taxis at first and second period were operated while the third one between 137 and 142 vehicles. Then, requests which were served for each period are 50 to 54, 152 to 156, and 297 to 299. We concluded that the profit of taxi operator did not depend on the number of operated taxis and served request but depended on passengers matching and taxis routing in correlation with passengers' fare and taxi operating cost.

6. Conclusion

This study focused on the taxi sharing problem in the form of mixed integer programming as dial a ride problem which ensured whenever passengers share taxi service, their fare will less than when they did not. Then, the problem was resolved by using a tabu search method that found optimal taxi routes and passengers matching in ordered to increase taxi operator income. The experiments showed that the tabu search method could improve initial routes by maximizing passenger fare and minimizing taxi operating cost in ordered to increase the optimization value. The method improved optimization value from an initial solution to the final solution in the amount of 10-14%. A further problem that could be noticed is a constraint of passengers' saving percentage thus taxi operator could emphasis the taxi-sharing system but with huge income.

References

[1]   Wang Y, Zheng B and Lim E-P 2018 Understanding the effects of taxi ride-sharing – A case study of Singapore *Computers*, *Environment and Urban System* **6** 124-32

[2]   Barann B, Beverungen D and Muller O 2017 An open-data approach for quantifying the potential of taxi ridesharing *Decision Support System* **99** 86-96

[3]   Furuhata M, Dessouky M, Ordonez F, Brunet M, Wang X and Koenig S 2013 Ridesharing: The state-of-the-art and future directions *Transportation Research* B **57** 28-46

[4]   Hosni H, Naoum-Sawaya J and Artail H 2014 The shared-taxi problem: Formulation and solution methods *Transportation Research* B **70** 303-18

[5]   Cordeau J-F and Laporte G 2007 The dial-a-ride-problem: Models and algorithms *Ann Oper Res* **153** 29-46

[6]   Paraboschi A, Santi P and Ratti C 2015 Modelling urban-level impact of a shared taxi market *Computers in Urban Planning and Urban Management* (Massachusetts: Massachusetts Institute of Technology)

[7]   Nanry W P and Barnes J W 2000 Solving the pickup and delivery problem with time windows using reactive tabu search *Transportation Research* B **34** 107-21

[8]   Cordeau J-F and Laporte G 2003 A tabu search heuristics for the static multi-vehicle dial-a-ride-problem *Transportation Research* B **37** 579-94

[9]   Attanasio A, Cordeau J-F, Ghiani G and Laporte G 2004 Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem *Parallel Computing* **30** 377-87

[10]   Kirchler D and Calvo R W 2013 A granular tabu search algorithm for the dial-a-ride-problem *Transportation Research* B **56** 120-35

[11]   Gendreau M 2003 An introduction to tabu search *Handbook of* metaheuristics, ed F Glover and G A Kochen-berger (Boston: Springer) p 37-54

[12]   Glover F, Laguna M and Marti R Principles of tabu search. February of 14th 2018. https://www.uv.es/rmarti/paper/docs/ts1.pdf

[13]   Glover F 1990 Tabu search: A tutorial *Interfaces* **20** 74-94