Application of Mean Shift Clustering to optimize matching problems in ridesharing for maximize the total number of match

H Sadewo1*, YSatria1*, HBurhan1

1Department of Mathematics, Faculty of Mathematics and Natural Sciences (FMIPA), University of Indonesia, Depok16424, Indonesia

E-mail*: hario.sadewo@sci.ui.ac.id, ysatria@sci.ui.ac.id, helen.burhan@sci.ui.ac.id

Abstract. The ridesharing system One of the solution can reduce the use of private vehicles so as to reduce congestion. The problem that happened with this ridesharing system is the matching problem between the driver and the passenger (rider). Mean shift clustering will be used in this paper as the first step in optimizing the matching problem in ridesharing. Mean shift clustering is a method of grouping spatial data by iteratively assigning data points to groups by shifting points to mode (mode is the highest density of data points in the region, in the context of mean-shift). So that with clustering it will be easier and more effective in pairing drivers and passengers optimally. After the clustering results are obtained, the driver and passenger will be paired based on the objective function of maximizing the number of pairs that occur (match). The basic idea of this objective function is to find the maximum number of match to do ridesharing. With the help of the Hopcroft Karp algorithm, can find a solution for the maximum number of match to do ridesharing.

1. Introduction

The massive population growth that occurred in Indonesia has shown fairly high growth in the use of private vehicles. Inadequate public transport services in Indonesia are also one of the biggest reasons why many people, especially in big cities, choose to use private vehicles. The peoples still think that public transportation is uncomfortable, travel time is longer, and the capacity of public transportation is not operated properly so that private vehicles are considered more comfortable than using public transportation [1]. Due to these factors, private vehicle users continue to grow. The growth in the use of private vehicles has resulted in congestion

There are many solutions to reduce the level of congestion. May be seen in terms of transportation needs, improving existing public transport infrastructure and traffic engineering and management. In addition, it can also reinforce the odd-even rules for private vehicles, one-way roads, and much more. One solution that will be discussed in this paper that can reduce the use of private vehicles and congestion is to use a ridesharing system.

Ridesharing, an individual traveler who shares the vehicle (ride share) and the cost of the trip with other travelers who have the same or nearly the same place of origin and destination. In this paper, will be discussed is ridesharing with a single driver and a single rider (single request). The pairing mechanism between the driver (single driver) and the passenger (single rider), especially in the case of matching between the two, is known as the matching problem [2].

The first step before optimizing the matching problem on ridesharing is determining possible match pairs. The problem encountered in ridesharing is to get the optimal driver-rider pair, while the number
of participants involved in the system is very large but the optimization time is very short. In this paper, clustering will be carried out as an initial step before determining the possible match pairs. Clustering seeks to cluster the origin and destination of the driver and passenger (rider) based on the proximity between the coordinates of the origin-destination location of each driver and rider. By dividing drivers and riders into a number of smaller sub-problems, so that clustering will reduce the driver-rider pairs that are possible to do ridesharing. The clustering method that will be used for this paper is mean shift clustering.

Mean shift clustering is nonparametric, because there is no need for model assumptions at the initial stage of clustering. There is only one parameter used, namely bandwidth which will determine the number of clusters automatically. So that it can make it easier to determine the number of clusters or grouping automatically.

After the possible match are obtained from the clustering results, a driver and rider pair will be checked whether they are feasible or not. After finding a feasible match, the final step is to optimize the matching problem by using an objective function for maximize the number of match in the ridesharing system. The mathematical model of the matching problem on ridesharing is in the form of maximum weighted bipartite matching.

Methods that can be used to determine maximum weighted bipartite matching include the Hungarian algorithm, the Edmonds’ Blossom algorithm, Ford Fulkerson's algorithm, or the Hopcroft Karp algorithm [3]. Hopcroft Karp algorithm will be used in this paper, because the algorithm aims to find the maximum cardinality of the number of pairs that match according to the objective function to maximize the number of pairs that match to do ridesharing.

2. Literature Review

2.1. Ridesharing

Ridesharing is a model of transportation in which individual traveller share vehicles and travel costs with other traveller who have the same time and destinations that are almost the same [4]. So with ridesharing, the driver maximizes the utility of using an empty seat in his vehicle by offering it to other passengers and in return the passenger helps pay for the travel costs incurred by the driver. Ridesharing is a promising and competitive system for reducing private car ownership [3].

The ridesharing system based on the type of implementation is divided into two, namely organized and unorganized ride sharing [4]. An unorganized ride sharing system occurs in everyday life that does not use a clear calculation scale, for example, ride sharing with family, colleagues, neighbour, or closest friends. Meanwhile, an organized ride-sharing system is managed by an operator that connects ridesharing actors who may not know each other. Ridesharing is not a new idea. Current technology advances should increase the popularity of the ridesharing system, as previously explained. Of course, ridesharing must be easy, safe, flexible, efficient and economical before it is more widely adopted [2].

The ridesharing system can allow pairing one driver with more than one rider. According to [2], a driver may pick up one or more riders (single or multiple riders) and so riders may ask to share a ride with a single driver or multiple drives. Table 1 provides a clearer explanation regarding the types of ridesharing based on rider and driver requests.

| Single Driver | Multiple Rider |
|---------------|----------------|
| Single Rider  | Driver and rider matching pair | Routing from the driver to pick up and drop off the rider |
| Multiple Driver | Routing from rider to change driver | Routing from drivers and riders |

2.2. Mean Shift Clustering

The concept of clustering is a data analysis method for finding and dividing data into groups that have similar characteristics (similarity) between one data and another [5]. So that data in one cluster has a maximum similarity level while data between clusters has a minimum similarity [6]. Data grouping
based on objectives can be divided into two, namely grouping as understanding and grouping aimed at use [7].

Mean Shift clustering including under the category of clustering algorithms that assign data points to groups iteratively by shifting the points to mode (mode is the highest density of data points in the region. So it is known as the Mode search algorithm. The basic idea in mean-shift clustering is to run an initialized mean-shift loop at each data point and then each mode defines one cluster, with all points convergent to the same mode on the same cluster [8]. In general, Mean Shift Algorithm has image processing and computer vision applications.

Mean shift is built on the concept of kernel density estimation (KDE). Therefore, the user does not need to select the cluster number. Focus on clusters defined by KDE mode (although this is not the only way to define clusters). Mode is the maximum local density. The natural algorithm for finding KDE mode is the iteration of mean shifts [8]. Interestingly, the number of clusters is not required for its implementation and, being density based, can detect clusters of any kind. Instead, the algorithm relies on a bandwidth parameter, which determines only the size of neighbourhood over which the density will be calculated. In this paper the bandwidth used is radius.

Small bandwidth can result in excessive clusters, while high values can mistakenly combine multiple clusters. Mean shift clustering takes a set of background points and a set of starting points and requires finding centroids of background points contained in spheres with a certain radius R centered on the starting point, or centered on centroid found at the previous step. Finding the points in the sphere requires the search points within distance R of the sphere centers [9]. The Mean Shift procedure consists of two steps: 1.) the construction of a probability density which reflects the underlying distribution of points in some feature space, and 2.) the mapping of each point to the mode(maximum) of the density which is closest to it [10].

3. Model formulation

3.1. Pre-processing

This paper will be used divided into three time periods of request. Drivers and riders will be grouped according to their request time. The first step before optimizing the matching problem on ridesharing is determining possible match pairs. In this paper clustering method will be used and without clustering.

Every time period the request will be carried out the clustering process. Mean shift clustering will be used to determine the driver-rider pair is considered as possible match or not. To fulfil the set of possible matches for the result of clustering, the coordinates (origin and destination) of the driver and rider must be in the same cluster. The algorithm to determine either driver-rider pair is a possible match or not will be shown in figure 1. Without clustering the possible match is to pair one driver with all existing riders.
After the possible match are obtained from the clustering results, a driver and rider pair will be checked whether they are feasible or not. Set of drivers who will be represented by D who drive a vehicle that still has an empty capacity in it and the group of riders who will be symbolized by R who need to share-trips with drivers who make a similar trip [11]. The coordination mechanism between the driver and the passenger (rider) especially in the matching problem has become the focus of research on ridesharing [12].

Assumed \( a \in D \cup R \) represent trip requests from drivers and riders or ridesharing participants. For the time feasible requirements of each travel request \( a \), it is assumed that the time announcement (time when the driver and rider is requested) is \( \tau(a) \). The time window of \( a \) is \( [e(a), q(a)] \) where \( e(a) \) is the earliest time of participant to depart from the origin and \( q(a) \) is the latest departure time of participant. Where \( T(\omega_a, \delta_a) \) is the trip time of each participant departs from its origin \( \omega_a \) and arrive at its destination \( \delta_a \). [3]. The latest time to the participant can arrive at its destination \( l(a) \). Distance between the origin-destination of each driver and rider \( S(\omega_a, \delta_a) \) has its origin \( \omega_a \), destination \( \delta_a \).

Defined \( \Delta T_d = k_d - \max\{t, e(d)\} \) and \( \Delta T_r = k_d + T(\omega_d, \omega_r) - \max\{t, e(r)\} \). Where \( k_d \) is the latest time driver must pick-up rider \( k_d = \min\{l(r) - T(\omega_r, \delta_r) - T(\omega_d, \omega_r), l(d) - T(\omega_d, \omega_r) - T(\omega_r, \delta_r) - T(\omega_r, \delta_d)\} \) by \( t \) is the optimization time is finished. If the pair of drivers and riders satisfies the following conditions, \( \Delta T_d \geq 0 \) and \( \Delta T_r \geq 0 \), that pair will be said feasible match and we will denote the feasible match by \( \hat{P} \).

### 3.2. Model Formulation

To optimize the use of ridesharing, the objective function used is to maximize the number of suitable driver-rider pairs or match in the ridesharing. The mathematical model used to solve the matching problem in ridesharing could be formulated by following equations [3]:

\[
\max \sum_{(d,r) \in \hat{P}} w_{dr} x_{dr} 
\]  \hspace{1cm} (1)

Subject to:
The following table shows the total participants with 60 percent of the participants' total distance travelled by driver and riders who do not save distance (the total distance travelled during ridesharing is greater than the total distance travelled by drivers and riders without ridesharing). This actually results in drivers and riders feeling disadvantaged. To cover that problem, it is necessary to add conditions that make sure distance savings when ridesharing. The conditions that must be met are: \( \Delta S = S_v(d,r) - S_u(d,r) \geq \epsilon \). \( S_v(d,r) = S(\omega_d, \delta_d) + S(\omega_r, \delta_r) \), shows the total individual trips for drivers and riders. \( S_u(d,r) = S(\omega_d, \delta_r) + S(\omega_r, \delta_d) \), shows the total matched trips for drivers and riders to do ridesharing. The \( \epsilon \) that will be used in this paper is \( \epsilon = 0 \) to make sure that drivers and riders are save distance when they are use a ridesharing system. For each the pair driver and rider in \( \bar{P} \) whose \( \Delta S < 0 \) will automatically be eliminated from \( \bar{P} \).

The results of the mathematical model obtained in the matching problem are equivalent to the Bipartite graph matching problem. Where the nodes in the Bipartite graph are two sets that are linked together, namely the \( D \) driver set and the \( R \) rider set that are disjoint. And the connected nodes describe the relationship between the driver and rider pairs with \( w_{dr} \) as the weight. The number of match to do ridesharing is critical to the long-term sustainability of a ride-sharing service to ensure customer satisfaction and benefits. Hence, in this paper the goal is to maximize the number of matches can be interpreted as a measure of reliability of the ride sharing system [3]. Formally the objective is modelled with uniform weights \( w_{dr} = 1 \) in the objective function. Because the mathematical model of matching problem in ridesharing for maximize the number of match uses uniform weights, then to find a maximum cardinality matching this paper uses Hopcroft Karp’s algorithm [13].

4. Experimental Result
In this paper, python programming language is used. The experimental data used in this paper are 30 drivers and 30 riders, consist of 30 coordinates of drivers’ origins, 30 coordinates of riders’ origins, 30 coordinates of drivers’ destinations, and 30 coordinates of riders’ destinations. The total time period used in this paper is three time periods with 50 seconds as the time steps. Started 07.00.01 until 07.00.50 is the first time period of drivers and riders with time announcement and so on for other’s time announcement with 50 seconds as the time steps. In this paper, will can make sure both participants, by doing ridesharing, will take additional time only add the trip time by 40 percent of the participants’ total trip time without doing ridesharing. So the latest arrival time of participant isl(\( a \)) = e(\( a \)) + T(\( \omega_a, \delta_a \)) + (40% * T(\( \omega_a, \delta_a \))). The travel time or the driver is also assumed to be able to drive a distance of 1 km in 1 minute then we calculate travel time of both participants by multiplying the travel distance with 60 seconds.

In this paper, will be compared the result with clustering method and without clustering method. The following table shows the total participants involved in each three time periods, the total \((d, r)\) that is possible match in each period, the total \((d, r)\) that is feasible match in each period, and the final match \((d, r)\) that obtained from each time periods without clustering method and without clustering.
Table 2. Experimental Result in 3 time periods with Mean Shift clustering

| Period | Numbers of Participants | Possible match | Feasible match | Final match |
|--------|-------------------------|----------------|----------------|-------------|
|        | Driver                  | Rider          |                |             |
| 1      | 11                      | 11             | 31             | 12          | 8           |
| 2      | 16                      | 17             | 95             | 22          | 10          |
| 3      | 12                      | 12             | 45             | 5           | 3           |

Table 3. Experimental Result in 3 time periods without clustering

| Period | Numbers of Participants | Possible match | Feasible match | Final match |
|--------|-------------------------|----------------|----------------|-------------|
|        | Driver                  | Rider          |                |             |
| 1      | 11                      | 11             | 121            | 15          | 8           |
| 2      | 16                      | 17             | 272            | 25          | 10          |
| 3      | 12                      | 12             | 144            | 4           | 3           |

Based on table 2 and table 3, obtain the total of matched pairs is 21. The clustering method also able reduce the calculation of the driver-rider pair that will be optimized based on each possible match from each time period, and the running time for clustering method faster than without clustering. This means the clustering method will ensure the driver-rider pairs that will be optimized are in a close proximity so that driver does not need to go too far to pick rider up. It will be more effective and efficient for the next step.

5. Conclusion

Pairing a driver with a rider or a matching problem is a problem in the ridesharing system. By using mean shift clustering, will make sure the driver-rider pairs that will be optimized are in a close proximity so that driver does not need to go too far to pick rider up. Then the matching problem is optimized by maximizing the number of pairs that match with the Hopcroft Karp assignment algorithm. Based on the experimental result, get the same number of matches. If the number of participants involved in the system is very large but the optimization time is very short, we can conclude that with mean shift clustering will be more effective and efficient for the next step than without clustering. Because clustering can also reduce to total calculation of the driver-rider pair that will be optimized by reducing the possible match.

Acknowledgement

This research supported by Publikasi Terindeks Internasional (PUTI) research grant of Universitas Indonesia 2020.

References

[1] Sitanggang R and Saribanon E 2018FAKTOR-FAKTOR PENYEBAB KEMACETAN DI DKI JAKARTA Jurnal Manajemen Bisnis Transportasi dan Logistik 4(3) pp 289-296
[2] Agatz N, Erera A, Savelsbergh Mand Wang X 2012 Optimization for dynamic ride-sharing: A review European Journal of Operational Research 223 pp 295-303
[3] Najmi A, Rey Dand Rashidi T H2017Novel dynamic formulations for real-time ride-sharing systems. Transportation research part E: logistics and transportation review 108 pp 122-140
[4] Furuhata M, Dessouky M, Ordóñez F, Brunet M E, Wang Xand Koenig S 2013 Ridesharing: The state-of-the-art and future directions Transportation Research Part B: Methodological 57 pp 28-46
[5] Februariyanti H and Santoso D B 2017 HIERARCHICAL AGGLOMERATIVE CLUSTERING UNTUK PENGELOMPOKAN SKRIPSI MAHASISWA
[6] Tan P N, Steinbach M and Kumar V 2006 Introduction to Data Mining. Boston:Pearson Education
[7] Prasetyo, Eko 2012 Data Mining konsep dan aplikasi menggunakan Matlab. Jakarta: Andi Offset
[8] Carreira-Perpinán M A 2015 A review of mean-shift algorithms for clustering. arXiv (Preprint arXiv:1503.00687)

[9] DeMenthon D and Megret R 2002 Spatio-temporal segmentation of video by hierarchical mean shift analysis. Computer Vision Laboratory, Center for Automation Research, University of Maryland

[10] Freedman D and Kisilev P 2009 June Fast mean shift by compact density representation. In 2009 IEEE Conference on Computer Vision and Pattern Recognition pp1818-25 IEEE

[11] Lee A and Savelbergh M 2015 Dynamic ridesharing: Is there a role for dedicated drivers?. Transportation Research Part B: Methodological 81 483-97

[12] Kamar E and Horvitz E 2009 Collaboration and Shared Plans in the Open World: Studies of Ride-sharing. In: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI) 9pp 187-194

[13] Izzuddin M 2016 Desain dan Analisis Algoritma Penyelesaian Permasalahan Penugasan Bersyarat dengan Representasi Bipartite Graph (Doctoral dissertation, Institut Teknologi Sepuluh Nopember)