Journal of Theoretical and Applied Information Technology
31st January 2020. Vol.98. No 02

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

K MEDOID-LEAST COST COMBINED MODEL IN COMBINED AND SCHEDULED SHIPPING SERVICE

PURBA DARU KUSUMA
School of Electrical Engineering, Telkom University, Indonesia
E-mail: purbodaru@telkomuniversity.ac.id

ABSTRACT

City courier service still meets inefficiency, especially while handling multi destination orders. In the existing system, single transaction order means delivering packet from one sender to one destination. So, when a customer needs to send more than one packets to more than one destination, customer must create order one by one. Then a driver can execute only one delivery order. Besides that, a driver also cannot deliver packets that their destination is close to each other because these packets are ordered by different customers. Based on this problem, in this work, we develop dispatch model that supports combined shipping model. In combined shipping model, customer can create one order that contains more than one destination. Meanwhile, a driver can execute more than one packet concurrently. This work is a continuation of our previous works in combined shipping service and scheduled shipping service. In this work, we develop the dispatch model by combing and modifying k-medoid method and least cost method. In our previous work, the world is a square shaped city. In this work, the virtual work is a circle shaped city. This work is also the improvement of our previous work that uses sequential least cost and Round Robin method. This model then is implemented into city courier simulation application. Based on the simulation result, this proposed model performs better than the conventional method in reducing the total driver distance. By using these proposed models, the total travel distance reduction is above 60 percents. The reduction in total driver distance has positive relation with reduction in total cost. These proposed models also reduce travel distance disparity among drivers as it becomes problems in the previous work. The travel distance standard deviation of least first-medoid model is below 5 kilometer while travel distance standard deviation of Round Robin-medoid model is below 10 percent.

Keywords: *Scheduled Shipping, Least Cost, K-medoids, Round Robin, Least First.*

1. INTRODUCTION

The rise of online motorcycle taxi business in Indonesia has triggered new business opportunity in Indonesia. This business is city courier service. This business becomes the answer to several problems in online motorcycle taxi business. The first problem is the resistance from conventional motorcycle taxi business because the online motorcycle taxi disrupts the traditional players' comfort zone and threatens their existence [1,2]. The second problem is fierce competition between online motorcycle taxi companies, especially between Go-Jek and Grab in Asia [3-5]. By delivering packets, online motorcylce taxi drivers are more utilized because there is new kind of delivery order rather than just transporting people. Courier delivery service can increase demand by keeping drivers busy although in not rush hour [6].

This business is also creates better opportunity for e-commerce players who need to deliver their purchased order faster than by using conventional courier service, especially when the packet destinations are in the same city with the merchant. This business also gives benefit to restaurants or food merchants so that they can provide food delivery service without having their own delivery units [5]. For example, there are approximately 300,000 food merchants that are connected to Go-Food as the brand for Go-Jek food delivery service [7]. Meanwhile, in 2018, there are 529 million food and drink orders that are ordered through Go-Food [7]. Besides Go-Food, there is Grab Food which is a brand for food delivery service that is held by Grab as the answer for the food delivery needs and a response to Go-Food so that Go-Jek is not the only one player in online food delivery service [8].

Unfortunately, order execution model that is used in city courier service is still conventional. Concept in the existing model is one on one model. It means that single transaction or delivery order from a customer or sender contains only one packet for one destination or receiver. This delivery order can only executed by single driver. So, when customer needs to send more than one packets with multi destinations, the delivery orders must be created one by one. It makes the system less efficient.

This problem has been tried to be solved by using combined shipping model [9] and scheduled shipping model [10] in our previous works. In combined shipping model, single delivery order can contains many packets and a driver can execute more than one packet [9]. In this previous work, the number of drivers that is needed to execute multi destinations delivery order can be less than the number of destinations in this delivery order [9]. In this work, we combined the random walk model and least effort model [9].

This previous work is improved by the scheduled shipping model [10]. In scheduled shipping model, packets from one order can be delivered by driver who executes other delivery order as long as their destination is close to each other [10]. In this work, we combined the Round Robin model and least cost model [10].

These previous works have reduced total travel distance successfully. Unfortunately, there is problem in disparity among drivers. Travel distance disparity among drivers is still high. Disparity in travel distance has positive relation with disparity in drivers' revenue.

Based on these problems, there are two purposes in this research. The first purpose is reducing total travel distance in conventional existing dispatch model. The second purpose is increasing travel distance equality among drivers.

In this work, we combine k-medoid method and least cost method. The least cost method has been used in our previous works [9,10]. We use k-medoid method because this method has been used widely in many quantitative clustering works [11-13], for example in parallel computing [12,14] and internet banking customer analysis [15]. In their work, Aryuni, et al compared the performance between k-means method and k-medoids method [15].

In load balancing process to increase equality among drivers, we use Round Robin method and least first method. Round Robin method is used because this method has been used widely in load balancing work, for example in cloud system [16-18] or in web server cluster [19].

This paper organized as follows. In the first section, we describe the background, problem statement, research purpose, and paper organization. In the second section, we explain the proposed model. In the third section, we explain the implementation of the proposed models into the city courier dispatch system simulation application. In the fourth section, we explain the analysis of the simulation result. In the fifth section, we discuss the analysis, result comparison between proposed model and previous model, and the research findings. In the sixth section, we make conclusion and propose future research potentials.

2. PROPOSED MODEL

There are five entities in the scheduled shipping courier system: customer, destination/receiver, collector, central warehouse, and driver. Customer is entity that creates delivery order in the system. Collector is entity that collects packets from customer to warehouse. Warehouse is entity that receives collected packets, dispatches orders and gives packets to selected drivers. Driver is entity that delivers packets from warehouse to destinations or receivers. Receiver/destination is entity that receives packet from driver. The illustration is shown in Figure. 1.

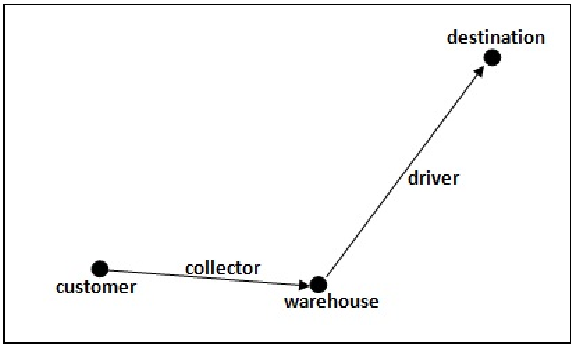

Figure 1. Proposed System Illustration

In conventional system, packet is delivered by driver directly from customer to destination. It is because one delivery order or transaction means order to deliver packet from single customer to single destination. It makes central warehouse is not needed in the conventional system. The conventional system is illustrated in Figure 2.

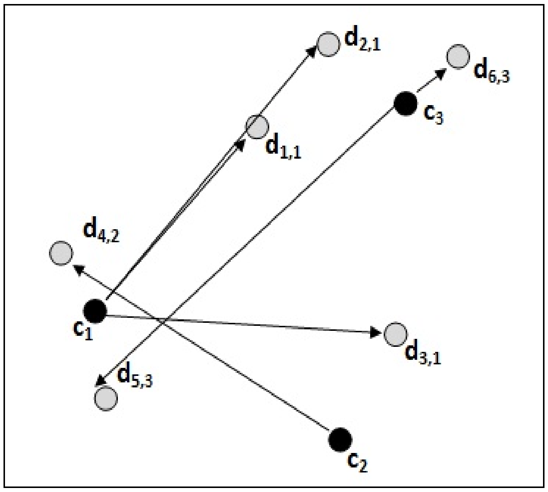

Figure 2.Conventional System Illustration

In Figure 2, it is shown that there are three customers $\{c_1, c_2, c_3\}$ and six destinations $\{d_1, d_2, \ldots, d_6\}$. Customer c_1 has packets that will be sent to $\{d_1, d_2, d_3\}$. Customer c_2 has packets that will be sent to $\{c_4\}$. Customer c_3 has packets that will be sent to $\{d_5, d_6\}$. Because there are six packets so that system should provide six available drivers to these requested orders.

This conventional system may create other problems. The first problem is providing available drivers. When the number of packets is low, system can provide adequate number of available drivers to execute these orders easily. Unfortunately, when the number of packets is high, potential of failed orders will be high too because providing available drivers will be difficult.

The second problem is the total delivery cost is high. As it is shown in Figure 2, because every packet must be delivered by distinct driver, total delivery distance in executing total orders will be high. It makes the total cost that must be paid by customers is high too. So the challenge is reducing the total cost by reducing total delivery distance.

The third problem is the average driver's revenue is relative low. It is because a driver can only deliver one packet in a single process. Although the total cost is high, the average driver's revenue is low. Connecting to the second problem, it is interesting to reduce total cost that must be paid by customers in one side while in the other side to increase average driver's revenue.

Based on these problems, in this work, packets will be grouped. The basic concept is packets that their destination is close to each other will be executed by single driver. Based on illustration in Figure 2, the scenario is as follows. These six packets can be clustered into three groups. The first group contains d_1, d_2, and d_6. The second group contains d_4 and d_5. The third group contains d_3. Based on this concept, the total cost will be reduced and the average drivers' revenue will increase because generally, one driver can delivers more than one packet.

The second concept is that this system adopts both combined shipping and scheduled shipping. In combined shipping [9], several packets from single customer will be delivered together by single driver as we developed previously. Unfortunately, when system implements only combined shipping service, packets from different customer cannot be delivered together by single driver although their destination is close to each other. It is because in basic combined shipping service, dispatch system runs simultaneously when a delivery order from a customer is created. Because other delivery orders are not created at the same time, dispatch system cannot detect whether there are delivery orders that their packets destinations may be near the packets destinations of the current delivery order.

Based on this problem, system should also adopt scheduled shipping. In scheduled shipping [10], dispatch system does not run simultaneously when one delivery order is created. When a delivery order is created, this order will be stored first and enters the waiting list for several periods waiting for other delivery orders that may come. After several periods, dispatch process runs. So, system will process more than one delivery order simultaneously.

The third concept is that the system needs a central warehouse. This central warehouse is used to store packets that are waiting to be delivered. When a delivery order is created, system will process this order by sending a collector to the customer's location to pick up the packets. Then, collector will bring these packets to the central warehouse. After the dispatch process runs, selected drivers will come to the central warehouse to get the packets that he must deliver and then will deliver these allocated packets to the their destination.

In our proposed model, dispatch process is divided into two processes. The first process is clustering the packets. The second process is balancing the delivery plan. In the clustering process, k-medoid method is chosen. The purpose of the first process is determining the first destination for all selected drivers. By using k-medoid method, the initial destination for every driver is the medoid or centroid of every cluster.

The second process is balancing the delivery plan. Load balancing is needed to distribute packets among selected drivers more equally based on the clustering result. When the clustering process ends, sometimes there is unbalance condition among clusters. Some clusters may contain few destinations while other clusters may contain many destinations. Some drivers travel very low miles while other drivers must travel very high miles. In this proposed work, there are two options that system can choose for load balancing. The first option is Round Robin method while the second option is least first method.

As it is explained above, the dispatch process contains two steps. This dispatch process main algorithm is shown in Figure 3. In Figure 3, function set_initialtarget represents the first step while function set_route represents the second step. The input of the first function is the destination set (E). This set contains the packets destination {e_1, e_2, e_3, …, $e_{npacket}$}. Variable n_{packet} represents the number of packets. The output of this function is stored in variable T_1 that represents the set of drivers' initial target. So, variabel T_1 contains {$t_{1,1}$; $t_{1,2}$; $t_{1,3}$; …; $t_{1,ndriver}$}. Variable n_{driver} represents the number of selected drivers in the system.

```
Begin
 $T_1$ ← set_initialtarget(E)
 $T_r$ ← set_route($T_1$, E)
End
```

Figure 3. Dispatch Process Main Algorithm

After the first function runs, the next function is executed. The input of the second function is the set of destination (E) and the set of initial target (T_1). The output of this function then is stored in variable T_r. Variable T_r represents the set of routing plan.

The first process is clustering process by using k-medoid method. This k-medoid method is divided into two steps. The first step is determining the initial medoids' location. The second step is determining the final medoids' location through iteration. Basically, the output of this process is the clusters that contain packet destinations and the medoid location of every cluster. Similar to the previous work, the number of clusters represents the number of drivers. The difference is as follows. In the previous work, the packets that should be delivered by a driver are packets that they are the members of the cluster related to the driver. In this work, we modified this process that the output of the clustering process that is used in the next process is only the medoids' location. This location represents the drivers' first packet destination. Meanwhile, the k-medoid process algorithm is shown in Figure 4.

```
Begin
 $n_m$ ← $n_{driver}$
 for i = 1 to $n_m$ do
  $m_i$ ← setinitiallocation(E,$S_m$)
 end

 status ← "run"
 while status = "run" do
 begin
  grouping()
  $d_{curtot}$ ← calc_totaldistance(E,M)
     if $d_{curtot}$ >= $d_{prevtot}$ then
      status ← "stop"
 end
end
```

Figure 4. K-medoid Process Algorithm

There are several variables in algorithm in Figure 4. Variable n_m represents the number of medoids and it is equal to the number of drivers. Variable m represents the medoid location with i as its index. Variable status represents iteration status where run means iteration still continues and stop represents iteration ends. Variable d_{curtot} represents current total distance between nodes and their medoid. Meanwhile, variable $d_{prevtot}$ represents previous total distance between nodes and their medoid. The number of drivers is determined based on the number of packets and the maximum packets that can be handled by single driver. The number of packets determination is formalized by using Equation 1. In Equation 1, variable $n_{maxperdriver}$ represents maximum number of drivers that can be executed by single driver.

$$n_{driver} = \operatorname{int}\left(\frac{n_{packet}}{n_{\max perdriver}}\right) \qquad (1)$$

Besides variables, there are three sub programs that are used in this algorithm. Function setinitiallocation is used to determine the initial medoids location. Function calc_totaldistance is used to determine the total distance. Procedure grouping is used to grouping the nodes into clusters.

Both d_{curtot} and $d_{prevtot}$ can be formalized by using variable d_{tot}. If the current time is t then the previous time is t-1. The formalization of current total distance, previous total distance and initial total distance is shown in Equation 2 to Equation 5. In Equation 5, it is shown that the distance between destination and medoid is Euclidean distance.

$$d_{prevtot} = d_{tot,t-1} \tag{2}$$

$$d_{curtot} = d_{tot,t} \tag{3}$$

$$d_{tot,0} = \sqrt{(2r^2)}.n_{packet} \tag{4}$$

$$d_{tot,t} = \sum_{i=1}^{n_{packet}} \left\| e_i - m_{j,i} \right\| \tag{5}$$

The initial medoid location is determined stochastically. The initial medoid location is determined randomly among destinations location and it follows uniform distribution. The rule is that a destination cannot be occupied by more than one medoid. The formalization of this initial medoid location is shown in Equation 6. In Equation 6, function s_e is used to get the status of the destination. Value 0 means the destination is still available. After initial medoid location is determined, the next process is iteration.

$$m_j = rand\left(e_1, e_{n_{packet}}\right) \wedge s_e(e_i) = 0 \tag{6}$$

The first activity inside the iteration is grouping the destinations into clusters. Similar to k-means method, in k-medoid method, destinations will be connected into the nearest medoid. This process is formalized by using Equation 7 and Equation 8. In Equation 8, it is shown that the distance between medoid and destination is calculated by using Euclidean distance.

$$m_{j,i} = m \mid m \in M \wedge \min\left(d(m, e_i)\right) \tag{7}$$

$$d(m,e) = \left\| m - e \right\| \tag{8}$$

After grouping process ends, the next process is changing the medoid location. In k-medoid method, medoid will move to other destinations inside the cluster randomly. This process follows uniform distribution. This process is formalized by using Equation 9.

$$m_{j,t+1} = rand(E_{j,t}) \tag{9}$$

The last process is determining whether the iteration still continues or stops. The simulation still runs as long as the current total distance is lower than previous total distance. This process is formalized in Equation 10. In Equation 10, s_{it} is the status of the iteration.

$$s_{it} = \begin{cases} "run", d_{tot,t} < d_{tot,t-1} \\ "stop", else \end{cases} \tag{10}$$

After the clustering process finishes, the next step is creating route plan. Different to our current work where the driver will deliver packets that their destination is in the driver's cluster, in this work, we implements least cost method. In this least cost method, the driver's next destination is packet destination that is the nearest to the current driver's position and the packet has not been occupied by other drivers. Although this system uses least cost method too, the initial driver's location is not in the central warehouse as in previous work [10], but in the related medoid location. The initial driver location is formalized by using Equation 11. Meanwhile, the next driver destination is formalized in Equation 12.

$$p(d_{j,0}) = m_j \tag{11}$$

$$p(d_{j,t}) = e \mid \min\left(d\left(e, p(d_{j,t-1})\right)\right) \wedge s_o(e) = 0 \tag{12}$$

After the least cost method is chosen, the next work is determining the scheduling method among drivers. We propose two options in giving equal opportunity for drivers and increasing equality in driver's revenue aspect. The first method is Round Robin method. The second method is least first method.

In the Round Robin method, drivers are sorted based on their index. The lowest indexed driver has the first turn to find his next destination. After this driver gets one next destination, this destination is blocked so that other drivers cannot acquire this destination. The next turn is driver whose index is one point after him. This process runs until the driver with the last index. After the last indexed driver gets a next destination, the next turn is the lowest indexed driver. This process runs until all destinations are blocked. In this work, we use non weighted Round Robin rather than Weighted Round Robin (WRR) as in other works [18,19]. This Round Robin algorithm is shown in Figure 5.

```
begin
 token ← 1
 $n_{avpacket}$ ← $n_{packet}$ - $n_{driver}$
 while $n_{avpacket}$ > 0 do
 begin
  next ← findnearest(p($c_{token}$),E))
  p($c_{token}$) ← p($e_{next}$)
  $s_o$($e_{next}$) ← 1
  $n_{avpacket}$ ← $n_{avpacket}$ - 1

  if token < $n_{driver}$ then
   token++
  else
   token ← 1
 end
end
```

Figure 5. Round Robin Algorithm

Several new variables are used in this Round Robin algorithm. Variable $n_{avpacket}$ represents the number of available packets or packets that has not been held by any driver. Variable next is used to store the index of destination that is chosen for the next destination. Its value is determined by using function findnearest. Then the new location of the current driver is moved to the next destination. After that, the status of this destination turns to 1 or it means occupied. Then, the number of available packets decrements.

The second scheduling method is least first method. The concept of this method is prioritizing driver with the lowest travel distance to get the next turn in getting next destination. This least first algorithm is shown in Figure 6.

```
begin
 $n_{avpacket}$ ← $n_{packet}$ - $n_{driver}$
 while $n_{avpacket}$ > 1 do
 begin
  token ← finddriverleast()
  next ← findnearest(p($c_{token}$),E))
  dist ← calcdist($c_{token}$,$e_{next}$)
  $d_{trav}$($c_{token}$)← $d_{trav}$($c_{token}$) + dist
  p($c_{token}$) ← p($e_{next}$)
  $s_o$($e_{next}$) ← 1
  $n_{avpacket}$ ← $n_{avpacket}$ - 1
 end
end
```

Figure 6. Least First Algorithm

The explanation of this algorithm is as follows. Different to the Round Robin algorithm, the first process inside the loop is determining the token or the other word is determining the driver that will get the opportunity to get his next destination. This process is held by function finddriverleast. Then, similar to the Round Robin algorithm, the next process is finding the available packets which its destination is the nearest to the selected driver. Then, system calculates the distance between the selected driver and the selected destination by using function calcdist. The result then is stored in variable dist and it is used to accumulate the selected driver's travel distance.

3. IMPLEMENTATION

The proposed model is then implemented into city courier dispatch system simulation application. The simulation is developed by using PHP language so that it is a web based application. The environment of the simulation is a virtual city. This virtual city shape is circle with a specific radius. The central warehouse is located in the center of the circle.

In the beginning, there are several processes. The first process is generating customer. Customer location is generated randomly and it follows uniform distribution. The customer location is described as pair (u_x,u_y) with x represents location in x coordinate and y represents location in y coordinate. The customer location determination is formalized by using Equation 13 to Equation 16. In Equation 13, r_u represents the radius of customer from the central warehouse. In Equation 14, α_u represents the angle of customer related to the central warehouse and it is represented in degree. Equation 15 is used to determine the x location of customer while Equation 16 is used to determine the y location of customer.

$$r_u = rand(0, r) \qquad (13)$$

$$\alpha_u = rand(0,360) \qquad (14)$$

$$u_x = r_u.\cos(\alpha_u) \qquad (15)$$

$$u_y = r_u.\sin(\alpha_u) \qquad (16)$$

After the customers are generated, the next process is generating the packets. There are two attributes for the packets: the owner and the destination. Similar to the customer location, packet destination determination is also generated randomly and it follows uniform distribution. The packet destination is determined by using Equation 17 to Equation 20. Equation 17 is used to determine the distance between packet destination and the central warehouse. Equation 18 is used to determine the angle of packet destination related to the central warehouse. Equation 19 is used to determine x location of the packet destination while Equation 20 is used to determine y location of the packet destination.

$$r_e = rand(0,r) \quad (17)$$

$$\alpha_e = rand(0,360) \quad (18)$$

$$e_x = r_e.\cos(\alpha_e) \quad (19)$$

$$e_y = r_e.\sin(\alpha_e) \quad (20)$$

After the customers and packets are generated, the next process is running the dispatch process. In this simulation, there are five models that are used so that the output among models can be compared to each others. The first two models are the proposed model. The last three models are models from previous works or the existing conventional models. The first model is the k-medoid-Round Robin combined model. The second model is the K-medoid-least first combined model. The third model is the conventional one to one model. The fourth model is the sequential model. The fifth model is the Round Robin model.

4. RESULT ANALYSIS

In this section, we will discuss the models performance evaluation. The performance data is acquired based on the simulation result. In this work, parameters that are measured include: total travel distance, total travel distance reduction, travel distance standard deviation, total customers cost, and total customers cost reduction. Total travel distance is the accumulation of travel distance from all drivers in delivering all packets. This parameter is formalized by using Equation 21. In Equation 21, d_{tot} represents total drivers travel distance while d_{trav} represents driver's travel distance. Total drivers distance is presented in kilometer.

$$d_{tot} = \sum_{i=1}^{n_{driver}} d_{trav,i} \quad (21)$$

Total drivers travel distance reduction is the reduction of the total drivers travel distance when the system implements non conventional model compared to the total drivers distance when the system implements conventional model. This total driver distance reduction is presented in percent. This reduction is formalized by using Equation 22. In Equation 22, $d_{totnoncon}$ represents the total drivers distance when system implements non conventional method. In the other hand, d_{totcon} represents the total drivers distance when system implements conventional distance. Variable η_{dis} represents the distance reduction. Positive value of this parameter means that the non conventional model is more efficient rather than conventional model.

$$\eta_{dis} = \left(\frac{d_{totcon} - d_{totnoncon}}{d_{totcon}} \right).100 \quad (22)$$

Travel distance standard deviation is the standard deviation of the accumulated travel distance among drivers. This parameter is used to analyze the equality among drivers in travel distance aspect. Higher standard deviation means wider disparity or lower equality in drivers travel distance.

Besides these non financial aspects, in this work, we also analyze the financial aspects. These aspects include total customers cost and total customers cost reduction. The total customers cost is formalized by using Equation 23 and Equation 24. Equation 23 is used to calculate total customers cost for system that implements conventional method. Meanwhile, Equation 24 is used to calculate total customers cost for system that implements non conventional method

$$m_{totcon} = d_{totcon}.m_{trav} \quad (23)$$

$$m_{totnon} = d_{totnon}.m_{trav} + n_{packet}.m_{pick} \quad (24)$$

In Equation 23, variable m_{totcon} represents total customer cost for conventional model. In Equation 24, variable m_{totnon} represents total customer cost when system implements non conventional model. Variable m_{trav} represents unit travel cost while m_{pick} represents pickup cost. Total cost and pickup cost is presented in rupiah. Unit travel cost is presented in rupiah per kilometer. In Equation 24, pickup cost is used to cover pickup cost from customer location to warehouse. It is different to in conventional method where packet is delivered directly from customer location to delivery location.

In this simulation, the independent or adjusted variables include: city radius, number of packets per user, number of users (customers), tolerance, and maximum number of packets per user. When a simulation runs in order to analyze certain adjusted variable, other adjusted variables are set as their default value. The adjusted variables default value is shown in Table 1.

The first test group is analyzing the relation between city radius and the output parameters. In this test, city radius ranges from 5 kilometer to 10 kilometer. The step size is 0.5 kilometer. There are five simulation sessions in every city radius value. The non financial result is shown in Table 2 to Table 4 while the financial result is shown in Table 5 and Table 6.

Table 1. Adjusted Variables Default Value

Variable	Default Value	Unit
r	7.5	kilometer
n_{user}	25	persons
$n_{packetuser}$	50	packets/user
Δ	5	packets
$n_{maxdriver}$	20	packets/driver
m_{pick}	20,000	rupiah/packet
m_{trav}	2,000	Rupiah/km

Table 2. Relation Between City Radius and Total Travel Distance

r (km)	Total Travel Distance (km)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5.0	4,746	326	665	650	619
5.5	4,942	368	714	728	677
6.0	5,456	397	774	839	761
6.5	5,919	430	838	869	766
7.0	6,322	462	867	852	826
7.5	6,623	499	1,058	968	863
8.0	7,005	529	1,115	1,110	964
8.5	7,637	563	1,153	1,157	1,020
9.0	8,191	598	1,196	1,124	1,063
9.5	8,655	624	1,254	1,219	1,141
10.0	9,119	664	1,263	1,302	1,144

Table 3. Relation Between City Radius and Total Travel Distance Reduction

r (km)	Total Travel Distance Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5,0	93.13	85.99	86.29	86.95
5,5	92.56	85.55	85.26	86.30
6,0	92.72	85.81	84.62	86.06
6,5	92.74	85.85	85.32	87.05
7,0	92.70	86.28	86.53	86.94
7,5	92.46	84.03	85.39	86.97
8,0	92.45	84.09	84.15	86.24
8,5	92.62	84.90	84.85	86.64
9,0	92.70	85.40	86.28	87.03
9,5	92.79	85.52	85.92	86.82
10,0	92.72	86.15	85.72	87.46

Table 4. Relation Between City Radius and Total Travel Distance Standard Deviation

r (km)	Total Travel Distance Standard Deviation (km)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5.0	4.31	3.92	4.09	2.09
5.5	4.95	4.30	4.12	2.59
6.0	5.36	4.52	5.24	3.08
6.5	5.96	4.95	5.67	3.10
7.0	6.26	5.23	5.95	2.77
7.5	6.88	6.25	6.14	3.03
8.0	7.03	6.76	7.23	3.71
8.5	7.90	6.96	6.76	3.78
9.0	7.69	7.36	6.92	3.87
9.5	8.67	7.41	7.38	4.37
10.0	9.10	7.32	8.36	4.15

Table 2 shows that city radius has positive relation with the total travel distance. The total travel distance increases due to the increasing of the city radius. Meanwhile, there is difference in the increasing of the total travel distance among models. Table 3 shows that all combined and scheduled shipping models perform better than the conventional one on one shipping model. This condition occurs in all of city radius values. The combined and scheduled shipping model performs much more efficient compared with the conventional model. In all models, the reduction is higher than 80 percent.

Compared among combined and scheduled shipping model, the pure sequential model [9] performs the best one. The total travel distance reduction is higher than 90 percent and it ranges from 92 to 93 percent. Meanwhile, the travel distance reduction of other models ranges from 84 to 87 percent. The least first-medoid model performs the second best in total travel distance reduction. Meanwhile, the Round Robin model [10] performs the worst in the total travel distance reduction. The travel distance reduction tends to fluctuate due to the increasing of the city radius. It means that there is not any relation between city radius and total travel distance reduction.

In total travel distance standard deviation, Table 4 shows that there is positive relation between city radius and total travel distance standard deviation. The total travel distance standard deviation increases due to the increasing of the city radius. Comparing among models, the performance rank in creating total travel distance standard deviation from the best to the worst are: Least First-Medoid, Round Robin [10], Round Robin-Medoid, and Sequential [9].

Table 5. Relation Between City Radius and Total Cost

r (km)	Total Cost (rupiah)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least Cost Medoid
5.0	9,490,400	1,151,200	1,829,200	1,800,000	1,737,600
5.5	9,882,800	1,234,400	1,926,800	1,955,600	1,852,800
6.0	10,911,200	1,293,600	2,047,200	2,178,000	2,020,000
6.5	11,836,800	1,358,800	2,174,800	2,236,800	2,031,600
7.0	12,642,000	1,422,400	2,234,000	2,202,400	2,150,400
7.5	13,244,800	1,497,600	2,615,200	2,434,000	2,225,200
8.0	14,008,333	1,557,333	2,728,000	2,719,667	2,426,667
8.5	15,273,200	1,625,600	2,804,800	2,812,800	2,539,200
9.0	16,381,200	1,694,400	2,891,200	2,746,800	2,624,400
9.5	17,308,333	1,747,667	3,006,333	2,937,000	2,780,667
10.0	18,236,000	1,825,600	3,025,600	3,102,400	2,786,400

Table 6. Relation Between City Radius and Total Cost Reduction

r (km)	Total Cost Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least Cost Medoid
5,0	87.87	80.73	81.03	81.69
5,5	87.51	80.50	80.21	81.25
6,0	88.14	81.24	80.04	81.49
6,5	88.52	81.63	81.10	82.84
7,0	88.75	82.33	82.58	82.99
7,5	88.69	80.25	81.62	83.20
8,0	88.88	80.53	80.59	82.68
8,5	89.36	81.64	81.58	83.37
9,0	89.66	82.35	83.23	83.98
9,5	89.90	82.63	83.03	83.93
10,0	89.99	83.41	82.99	84.72

Table 5 shows that there is positive relation between city radius and total cost. Total cost increases due to the increasing of the city radius. This condition occurs in all models, both conventional and non conventional models. Generally, all non conventional shipping models create lower total cost rather than the conventional models.

Table 6 strengthens the argument that combined and scheduled shipping models perform better than the conventional model in reducing total cost. Total cost reduction of all non conventional models range from 80 to 89 percents. The performance rank in total cost reduction among models from the best to the worst are: Sequential [9], Least First-Medoid, Round Robin [10], and Round Robin-Medoid. Meanwhile, there is not relation between city radius and total cost reduction.

The second test group is analyzing the relation between the number of packets per user and the output parameters. In this test, the number of packets per user ranges from 10 to 100 packets per user. The step size is 10 packets per user. There are five simulation sessions for every value of number of packets per user. The non financial result is shown in Table 7 to Table 9. Meanwhile, the financial result is shown in Table 10 and Table 11.

Table 7. Relation Between Number of Packets per User and Total Travel Distance

$n_{packetuser}$ (unit)	Total Travel Distance (km)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First Medoid
10	1,282	185	289	319	285
20	2,738	285	502	495	460
30	4,067	362	664	642	619
40	5,300	424	784	835	737
50	6,900	493	1,009	1,039	919
60	8,062	556	1,095	1,190	1,089
70	9,677	622	1,129	1,330	1,202
80	10,486	684	1,402	1,403	1,341
90	12,355	753	1,552	1,669	1,427
100	13,027	800	1,643	1,800	1,561

Table 8. Relation Between Number of Packets per User and Total Travel Distance Reduction

$n_{packetuser}$ (unit)	Total Travel Distance Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First Medoid
10	85.59	77.48	75.15	77.79
20	89.60	81.66	81.91	83.19
30	91.09	83.68	84.22	84.79
40	92.00	85.20	84.24	86.10
50	92.85	85.38	84.94	86.68
60	93.11	86.42	85.24	86.49
70	93.57	88.33	86.26	87.58
80	93.48	86.63	86.62	87.22
90	93.90	87.44	86.49	88.45
100	93.86	87.39	86.18	88.02

Table 9. Relation Between Number of Packets per User and Travel Distance Standard Deviation Among Drivers

$n_{packetuser}$ (unit)	Total Travel Distance Standard Deviation (km)			
	Sequential	Round Robin	Round Robin-Medoid	Least First Medoid
10	11.61	7.11	7.04	2.70
20	9.12	6.85	5.73	3.34
30	7.72	6.09	5.58	3.29
40	7.04	5.37	5.59	3.54
50	6.63	6.22	6.06	3.78
60	6.13	5.35	5.93	3.33
70	6.13	4.59	6.42	3.28
80	5.81	5.23	5.86	3.61
90	5.74	5.06	6.02	3.34
100	5.57	4.77	5.85	3.65

Table 7 shows that there is positive relation between number of packets per user and the total travel distance. The total travel distance increases due to the increasing of the number of packets per user. This condition occurs in all models, both conventional and non conventional models.

Table 8 shows that there is positive relation between number of packets per user and the total travel distance reduction. The total travel distance increases due to the increasing of the number of packet per user. Table 8 also shows that all non conventional models reduce the total travel distance significantly. The reduction ranges from 75 to 93 percents. The rank of total travel distance reduction among models from the best to the worst are: sequential [9], least first-medoid, Round Robin [10], and Round Robin-medoid model. There is significant disparity in total travel distance reduction between sequential model and other models.

Table 9 shows that there is negative relation between number of packets per user and travel distance standard deviation among drivers in sequential model, Round Robin model, and Round Robin-medoid model. In these three models, the travel distance standard deviation decreases due to the increasing of the number of packets per user. The sequential model [9] produces the highest travel distance standard deviation when the number of packets per user is few. But, when the number of packets per user is many, the travel distance standard deviation that is produced by sequential model is similar to two other models. In the other side, there is not any relation between number of packets per user and travel distance standard deviation among drivers in least first-medoid model. The total travel distance standard deviation that is produced by least first-medoid model is the lowest among other models, tends to stable with small fluctuation.

Table 10. Relation Between Number of Packets per User and Total Cost

$n_{packetuser}$ (unit)	Total Cost (rupiah)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5.0	9,490,400	1,151,200	1,829,200	1,800,000	1,737,600
5.5	9,882,800	1,234,400	1,926,800	1,955,600	1,852,800
6.0	10,911,200	1,293,600	2,047,200	2,178,000	2,020,000
6.5	11,836,800	1,358,800	2,174,800	2,236,800	2,031,600
7.0	12,642,000	1,422,400	2,234,000	2,202,400	2,150,400
7.5	13,244,800	1,497,600	2,615,200	2,434,000	2,225,200
8.0	14,008,333	1,557,333	2,728,000	2,719,667	2,426,667
8.5	15,273,200	1,625,600	2,804,800	2,812,800	2,539,200
9.0	16,381,200	1,694,400	2,891,200	2,746,800	2,624,400
9.5	17,308,333	1,747,667	3,006,333	2,937,000	2,780,667
10.0	18,236,000	1,825,600	3,025,600	3,102,400	2,786,400

Table 11. Relation Between Number of Packets per User and Total Cost Reduction

$n_{packetuser}$ (unit)	Total Cost Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5,0	87.87	80.73	81.03	81.69
5,5	87.51	80.50	80.21	81.25
6,0	88.14	81.24	80.04	81.49
6,5	88.52	81.63	81.10	82.84
7,0	88.75	82.33	82.58	82.99
7,5	88.69	80.25	81.62	83.20
8,0	88.88	80.53	80.59	82.68
8,5	89.36	81.64	81.58	83.37
9,0	89.66	82.35	83.23	83.98
9,5	89.90	82.63	83.03	83.93
10,0	89.99	83.41	82.99	84.72

Table 10 shows that there is positive relation between number of packets per user and the total cost. The total cost increases due to the increasing of the number of packets per user. Fortunately, all of non conventional models perform better in creating lower total cost rather than the conventional model. This disparity is significant. Table 11 also strengthens this condition. In Table 11, it is shown that the total cost reduction ranges from 80 to 89 percents. The sequential model performs the best in total cost reduction and it ranges from 87 to 89 percent. It creates disparity with three other non conventional models.

Table 11 shows that there are various behaviors in relation between number of packets per user and total cost reduction. In sequential model, there is positive relation between the number of packets per user and total cost reduction. In Round Robin model and least first-medoid model, the total cost reduction tends to fluctuate with small increasing due to the increasing of the number of packets per user. Meanwhile, in Round Robin-medoid model, there is not any relation between number of packets per user and the total cost reduction.

The third test group is analyzing the relation between the number of users and the output parameters. In this test, the number of users ranges from 5 to 50 users. The step size is 5 users. There are five simulation sessions for every value of number of users. The non financial result is shown in Table 12 to Table 14. Meanwhile, the financial result is shown in Table 15 and Table 16.

Table 12. Relation Between Number of Users and Total Travel Distance

n_{user} (person)	Total Travel Distance (km)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	1,370	183	272	289	283
10	2,726	279	455	497	464
15	4,033	362	638	682	617
20	5,271	431	782	855	803
25	6,995	495	962	982	924
30	8,226	558	1,219	1,272	1,058
35	9,578	618	1,329	1,316	1,101
40	10,865	680	1,318	1,570	1,253
45	12,369	740	1,551	1,528	1,321
50	13,953	799	1,597	1,633	1,518

Table 13. Relation Between Number of Users and Total Travel Distance Reduction

n_{user} (person)	Total Travel Distance Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	86.63	80.17	78.88	79.36
10	89.75	83.30	81.76	82.98
15	91.02	84.18	83.10	84.69
20	91.83	85.16	83.78	84.76
25	92.93	86.25	85.96	86.80
30	93.22	85.18	84.53	87.13
35	93.55	86.12	86.26	88.50
40	93.74	87.87	85.55	88.46
45	94.02	87.46	87.65	89.32
50	94.27	88.55	88.29	89.12

Table 14. Relation Between Number of Users and Travel Distance Standard Deviation Among Drivers

n_{user} (person)	Total Travel Distance Standard Deviation (km)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	11.89	5.93	6.81	2.71
10	9.67	6.17	6.17	3.23
15	8.26	5.69	6.61	3.46
20	7.38	5.64	6.37	3.37
25	6.74	5.91	6.36	3.95
30	6.33	6.04	7.43	3.58
35	6.05	5.90	5.53	3.46
40	5.74	5.10	6.01	3.16
45	5.65	5.49	5.32	2.88
50	5.31	4.52	4.85	3.35

Table 12 shows that there is positive relation between number of users and total travel distance. Total travel distance increases due to the increasing of the number of users. This condition occurs in all models. This positive relation also occurs between the number of users and the total travel distance reduction as it is shown in Table 13. This condition occurs in all non conventional models.

Table 13 shows that all non conventional models make efficiency in reducing total travel distance rather than conventional model. Meanwhile, there is difference among models in creating total travel reduction. The sequential model creates the highest total travel distance reduction that ranges from 86 percent to 94 percent. The disparity in making total travel distance between sequential model and other models is wide enough. Meanwhile, disparity among three other models is not significant. Among other models, Round Robin model performs the best when the number of users is low and least first-medoid model performs the best when the number of users is high.

Table 14 shows that in reducing the total driver distance standard deviation, the two proposed models perform better than the two previous models. The sequential model performs the worst one. In sequential model, there is negative relation between number of users and total travel distance standard deviation. The total travel distance standard deviation decreases due to the increasing of the number of users. Meanwhile, in three other non conventional models, there is not any relation between number of users and total travel distance standard deviation. The least first-medoid model becomes the best in reducing the total travel distance standard deviation.

Table 15. Relation Between Number of Users and Total Cost

n_{user} (person)	Total Cost (rupiah)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	2,738,667	464,667	642,333	677,667	664,667
10	5,451,200	757,600	1,109,600	1,193,600	1,127,200
15	8,064,400	1,023,200	1,575,200	1,662,000	1,534,000
20	10,542,000	1,261,200	1,963,200	2,109,600	2,005,200
25	13,989,600	1,488,400	2,422,800	2,464,000	2,346,000
30	16,450,000	1,714,400	3,037,600	3,143,200	2,716,400
35	19,154,000	1,934,800	3,357,200	3,330,400	2,902,000
40	21,729,600	2,159,200	3,434,800	3,939,600	3,306,000
45	24,738,000	2,379,600	4,001,200	3,955,600	3,540,800
50	27,904,400	2,597,200	4,194,000	4,266,000	4,035,600

Table 16. Relation Between Number of Users and Total Cost Reduction

n_{user} (person)	Total Cost Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	83.03	76.55	75.26	75.73
10	86.10	79.64	78.10	79.32
15	87.31	80.47	79.39	80.98
20	88.04	81.38	79.99	80.98
25	89.36	82.68	82.39	83.23
30	89.58	81.53	80.89	83.49
35	89.90	82.47	82.61	84.85
40	90.06	84.19	81.87	84.79
45	90.38	83.83	84.01	85.69
50	90.69	84.97	84.71	85.54

Table 15 shows that there is positive relation between the number of users and the total cost. The total cost increases due to the increasing of the number of users. This condition occurs in all models. This condition is rational as consequence of the increasing of the total travel distance.

Table 16 shows that there is positive relation between the number of users and the total cost reduction. The total cost reduction increases due to the increasing of the number of users. This condition occurs in all non conventional models. So, it can be said that the increasing of the number of users makes system more efficient.

Based on the comparison among models in Table 16, it is shown that the sequential model becomes the most efficient model in total cost reduction aspect. The disparity between this model and three other models is also significant. The least first-medoid model becomes the second most efficient model in reducing total cost.

The fourth test group is analyzing the relation between the tolerance and the output parameters. In this test, the tolerance ranges from 1 to 10. The step size is 1. There are five simulation sessions for every value of tolerance. The non financial result is shown in Table 17 to Table 19. Meanwhile, the financial result is shown in Table 20 and Table 21.

Table 17. Relation Between Tolerance and Total Travel Distance

Tolerance (Δ)	Total Travel Distance (km)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
1	6,778	529	940	1,032	888
2	6,907	520	1,000	919	883
3	6,805	509	1,011	943	917
4	6,942	504	855	969	865
5	6,487	490	968	1,034	887
6	6,595	489	842	980	877
7	6,720	491	986	1,030	918
8	6,837	486	926	904	872
9	6,644	482	967	1,013	965
10	6,742	471	869	1,039	888

Table 18. Relation Between Tolerance and Total Travel Distance Reduction

Tolerance (Δ)	Total Travel Distance Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
1	92.20	86.13	84.78	86.90
2	92.48	85.52	86.69	87.22
3	92.52	85.15	86.14	86.52
4	92.74	87.69	86.04	87.54
5	92.45	85.08	84.06	86.33
6	92.58	87.23	85.13	86.71
7	92.69	85.33	84.67	86.34
8	92.89	86.46	86.77	87.25
9	92.75	85.45	84.76	85.47
10	93.02	87.11	84.59	86.83

Table 19. Relation Between Tolerance and Travel Distance Standard Deviation Among Drivers

Tolerance (Δ)	Total Travel Distance Standard Deviation (km)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
1	5.37	5.36	6.18	3.21
2	6.07	5.75	5.94	3.54
3	5.85	6.24	7.06	3.30
4	6.16	4.76	5.79	2.94
5	6.45	6.37	6.49	3.63
6	6.47	4.55	5.51	3.22
7	6.97	5.86	6.28	3.35
8	7.15	5.40	5.62	3.54
9	7.25	6.28	6.27	3.38
10	7.33	5.05	6.19	3.27

There are various behaviors in the relation between tolerance and non financial output parameters. In Table 17 and Table 18, it is shown that there is not any relation between tolerance and both total travel distance and total travel distance reduction. The total travel distance and total travel distance standard deviation tend to fluctuate due to the increasing of the tolerance. Meanwhile, in Table 19, there is variation in the relation between tolerance and travel distance standard deviation. When system implements sequential model, there is positive relation between tolerance and total travel distance standard deviation. The travel distance standard deviation increases due to the increasing of the tolerance. Meanwhile, when system implements three other non conventional models, the total travel distance standard deviation fluctuates due to the increasing of the tolerance.

Similar to previous test groups, as it is shown in Table 18, the sequential model creates the highest total travel distance reduction among non conventional models. It occurs in any tolerance value. Disparity with three other models is also significant.

In travel distance standard deviation aspect, least first-medoid model performs the best one. In all tolerance value, the standard deviation is still lower than 4 kilometer. In the other side, the sequential model performs the worst in creating low standard deviation. But, the disparity with the other models is not significant.

Table 20. Relation Between Tolerance and Total Cost

Tolerance (Δ)	Total Cost (rupiah)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
1	13,554,400	1,556,800	2,379,600	2,562,400	2,274,800
2	13,812,400	1,538,000	2,498,400	2,338,000	2,264,400
3	13,609,600	1,516,400	2,520,000	2,385,200	2,333,600
4	13,882,400	1,506,800	2,208,400	2,436,800	2,228,400
5	12,972,400	1,479,200	2,434,000	2,567,200	2,272,400
6	13,189,600	1,477,600	2,183,600	2,460,400	2,252,800
7	13,439,200	1,481,600	2,471,200	2,558,800	2,334,800
8	13,672,400	1,471,200	2,350,400	2,306,800	2,242,400
9	13,286,400	1,462,400	2,432,400	2,524,000	2,429,600
10	13,482,800	1,440,000	2,236,800	2,577,600	2,274,000

Table 21. Relation Between Tolerance and Total Cost Reduction

Tolerance (Δ)	Total Cost Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
1	88.51	82.44	81.10	83.22
2	88.87	81.91	83.07	83.61
3	88.86	81.48	82.47	82.85
4	89.15	84.09	82.45	83.95
5	88.60	81.24	80.21	82.48
6	88.80	83.44	81.35	82.92
7	88.98	81.61	80.96	82.63
8	89.24	82.81	83.13	83.60
9	88.99	81.69	81.00	81.71
10	89.32	83.41	80.88	83.13

Table 20 and Table 21 show that there is not any relation between tolerance and the financial outputs. Both total cost and total cost reduction fluctuate due to the increasing of the tolerance. This condition is the consequence of the fluctuation of the total travel distance aspect due to the increasing of the tolerance.

The fifth test group is analyzing the relation between the maximum packets per driver and the output parameters. In this test, the number of maximum packets per driver ranges from 5 to 50 packets. The step size is 5 packets. There are five simulation sessions for every value of maximum packets per driver. The non financial result is shown in Table 22 to Table 24. Meanwhile, the financial result is shown in Table 25 and Table 26.

Table 22. Relation Between Number of Maximum Packets per Driver and Total Travel Distance

$n_{maxperdriver}$ (unit)	Total Travel Distance (km)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	6,977	731	1,894	1,623	1,884
10	6,736	600	1,176	1,260	1,161
15	6,628	532	1,038	1,007	965
20	7,031	497	966	1,053	904
25	6,587	473	874	878	811
30	6,904	460	841	921	720
35	6,788	446	782	778	691
40	6,911	441	674	818	693
45	6,664	426	686	764	648
50	6,688	429	644	689	633

Table 23. Relation Between Number of Maximum Packets per Driver and Total Travel Distance Reduction

$n_{maxperdriver}$ (unit)	Total Travel Distance Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	89.52	72.86	76.73	72.99
10	91.10	82.55	81.29	82.77
15	91.98	84.34	84.80	85.45
20	92.93	86.25	85.02	87.14
25	92.82	86.73	86.68	87.68
30	93.34	87.82	86.66	89.58
35	93.43	88.48	88.54	89.81
40	93.63	90.25	88.16	89.97
45	93.60	89.70	88.54	90.28
50	93.58	90.37	89.70	90.54

Table 24. Relation Between Number of Maximum Packets per Driver and Travel Distance Standard Deviation Among Drivers

$n_{maxdriver}$ (unit)	Total Travel Distance Standard Deviation (km)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	3.97	2.02	3.66	1.85
10	4.93	2.95	4.54	2.89
15	5.67	4.62	5.25	3.55
20	6.81	5.64	5.79	3.54
25	7.11	6.49	6.63	3.45
30	7.88	6.74	8.31	2.83
35	8.03	7.30	7.55	2.92
40	9.44	6.23	8.66	2.99
45	9.98	7.07	8.09	2.66
50	10.48	6.77	8.72	2.88

In Table 22, it is shown that there is variation in relation between the maximum packets per driver and the total travel distance. This variation depends on the model that is used. When system implements conventional model, there is not any relation between maximum packets per driver and the total travel distance. Meanwhile, when system implements non conventional model, there is negative relation between maximum packets per driver and total travel distance. The total travel distance decreases due to the increasing of the maximum packets per driver. By comparing among non conventional models, the total travel distance decreases slower when system implements sequential model rather than other non conventional models.

Table 23 shows that the sequential model still performs as the most efficient non conventional model in reducing total travel distance rather than other non conventional model. The disparity is wide when the number of maximum packets per driver is low. But, when the number of maximum packets per driver is high, the disparity is not significant. All non conventional models create positive total travel distance reduction in any number of maximum packets per driver. Meanwhile, there is positive relation between the maximum packets per driver and total travel distance reduction.

Table 24 shows that there is different behavior among non conventional models in the relation of the maximum packets per driver and the total travel distance standard deviation. In sequential model, Round Robin model, and Round Robin-medoid model, the standard deviation increases due to the increasing of the maximum packets per driver. Meanwhile, in least first-medoid model, the standard deviation fluctuates due to the increasing of the maximum packets per driver. The least first-medoid model becomes the best model in creating low standard deviation.

Table 25. Relation Between Number of Maximum Packets per Driver and Total Cost

$n_{maxdriver}$ (unit)	Total Cost (rupiah)				
	Conventional	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	13,952,400	1,960,800	4,286,000	3,745,200	4,267,600
10	13,471,600	1,698,000	2,850,400	3,019,600	2,820,400
15	13,255,200	1,562,800	2,574,400	2,514,000	2,428,400
20	14,062,000	1,493,200	2,432,000	2,605,600	2,308,000
25	13,172,800	1,444,800	2,247,600	2,254,400	2,121,200
30	13,807,200	1,418,000	2,181,200	2,341,200	1,938,400
35	13,576,000	1,391,600	2,062,800	2,054,800	1,882,000
40	13,820,800	1,379,600	1,847,200	2,134,800	1,885,600
45	13,327,200	1,352,000	1,871,600	2,026,400	1,795,200
50	13,374,800	1,357,200	1,787,600	1,876,000	1,764,800

Table 26. Relation Between Number of Maximum Packets per Driver and Total Cost Reduction

$n_{maxdriver}$ (unit)	Total Cost Reduction (%)			
	Sequential	Round Robin	Round Robin-Medoid	Least First-Medoid
5	85.95	69.28	73.16	69.41
10	87.40	78.84	77.59	79.06
15	88.21	80.58	81.03	81.68
20	89.38	82.71	81.47	83.59
25	89.03	82.94	82.89	83.90
30	89.73	84.20	83.04	85.96
35	89.75	84.81	84.86	86.14
40	90.02	86.63	84.55	86.36
45	89.86	85.96	84.80	86.53
50	89.85	86.63	85.97	86.81

Table 25 shows that there are different behavior in relation between maximum packets per driver and the total cost. There is not any relation between maximum packets per driver and total cost when system implements conventional model. Meanwhile, there is negative relation between maximum packets per driver and the total cost. Total cost decreases due to the increasing of the maximum packets per driver.

Table 26 shows that there is positive relation between maximum packets per driver and total cost reduction. Total cost reduction increases due to the increasing of the maximum packets per driver. The sequential model performs the most efficient model. Meanwhile, the performance in total cost reduction among other non conventional models is similar to each other. When the number of maximum packets per driver is low, disparity between sequential model and other models is high. In the other side, when the number of maximum packets per driver is high, disparity in cost reduction among models is low.

5. COMPARISON BETWEEN PROPOSED MODELS AND PREVIOUS MODELS

In the previous section, we have analyzed the performance of the models in every adjusted parameter. We also have analyzed the performance in both financial and non financial aspects. Comparison among models in every adjusted parameters and output parameters also has been done. In this section, we will resume and discuss the model performance, performance comparison between the proposed models and the models from the previous work [9,10], and the linkage between the result and the research purpose.

Generally, all non conventional models have met the basic requirement in reducing total travel distance. These non conventional models are the proposed models and the models from previous works. With the scenario that is set in this work, there is significant reduction in total travel distance. In all aspects, the reduction is higher than 60 percents. The system will be more efficient due to the increasing of the number of packets that are delivered. It occurs when the number of customers or the number of packets per driver increases. City radius and tolerance do not affect the total driver distance reduction. Maximum packets per driver has positive relation with the total travel distance reduction but not significant. The reduction in total travel distance has positive relation with the reduction in total cost.

Performance in reducing total travel distance among models is not equal. Sequential least cost model [9] becomes the best model in reducing total travel distance. Disparity between this model and other non conventional models is also significant. Meanwhile, the least first-medoid model becomes the second best model.

The next concern is creating equality among drivers, both in travel distance and drivers' revenue. Equality in travel distance also has positive relation with equality in revenue. In this aspect, the least first-medoid model performs as the best one in creating the lowest disparity among drivers. This condition occurs in all values of the adjusted parameters. Besides that, by using least first-medoid model, standard deviation in travel distance among non conventional models is below 5 kilometer and this value is stable. In the other side, the sequential model becomes the worst in reducing travel distance disparity among drivers. Performance of the Round Robin model [10] and Round Robin-medoid models is between the sequential and least first-medoid model.

6. CONCLUSION AND FUTURE WORK

Based on the explanation above, this work has proposed two non conventional models by using k-medoid method and least cost method as its basis method. The first model is Round Robin-medoid model and the second model is least first-medoid model. The proposed models have met the research purpose in reducing expensive travel distance and cost that occur in conventional model. The travel distance reduction is more than 60 percent. The proposed models also perform well in creating equality among drivers rather than models in previous work. The travel distance standard deviation of the least first-medoid model is below 5 kilometer while the Round Robin-medoid model is below 10 percent. All these values are below the sequential model.

There are several future research potentials based on this work. In this work, packets destination and sender location are distributed uniformly. So, it is needed to evaluate these proposed models in various distributions, such as Poison or exponential. In this work, packet weight is ignored. In the real world, packets are in various weight and size and so that packets cannot be generalized. So, future research that proposes model where packets size and weight cannot be generalized is needed too.

REFERENCES:

[1] I. Budiari, "Go-Jek Drivers Fearing Violent Resistance from Traditional Ojek", *The Jakarta Post*, June 11, 2015, https://www.thejakartapost.com/news/2015/06/11/go-jek-drivers-fearing-violent-resistance-traditional-ojek.html.

[2] N. Freischlad, "Why Motorcycle Taxi Drivers are Ganging Up Against on-demand Ride Apps in Indonesia", *Techinasia*, July 15, 2015, https ://www.techinasia.com/startups-in-indonesia-indonesia-motorcycle-hailing-taxi-apps-jakarta -gojek.

[3] C. Chandler, "Grab vs. Go-Jek: Inside Asia's Battle of the Super Apps", *Fortune*, March, 20,

2019, https://fortune.com/longform/grab-gojek-super-apps/.

[4] S. Ren, "Why Grab Doesn't Have a Handle on Indonesia", *Bloomberg*, November 6, 2018, https://www.bloomberg.com/opinion/articles/2018-11-05/grab-faces-steep-competition-in-indonesia-from-go-jek.

[5] Y. Lee and H. Suhartono, "Grab and Gojek Square Off in an International Food Fight", *Bloomberg*, September 5, 2019, https:// www.bloomberg.com/news/articles/2019-09-04/ grab-and-gojek-are-squaring-off-in-an-international-food-fight.

[6] J. Davis, "Lesson from Go-Jek: Answer to Uber, Grab", *The Jakarta Post*, April 18, 2018, https://www.thejakartapost.com/academia/2018/04/18/lesson-from-go-jek-answer-to-uber-grab.html.

[7] R.R. Mufti, "Go-Jek Sees Boom in Food Delivery Business", *The Jakarta Post*, January 8, 2019, https://www.thejakartapost.com/news/2019/01/08/go-jek-sees-boom-in-food-delivery-business.html.

[8] Q. Tariq, "Uber Eats Says Goodbye, Grabfood Says Hello", *The Star*, May 28, 2018, https:// www.thestar.com.my/tech/tech-news/2018/05/28/grabfood-to-replace-ubereats.

[9] P.D. Kusuma, "Comparison Between Least Effort and Random Walk in Combined Shipping Service", *Journal of Engineering and Applied Sciences*, vol. 14(1), 2019, 130-138.

[10] P.D. Kusuma and A.B. Osmond, "Round Robin-Least Cost Combined Model in City Courier Scheduled Shipping Service", *Journal of Engineering and Applied Sciences*, vol. 13(15), 2018, 6181-6187.

[11] X. Chen, H. Peng, and J. Hu, "K-Medoids Substitution Clsutering Method and a New Clustering Validity Index Method", *Proceeding of 6th Congress on Intelligent Control and Automation*, June 21, 2006, Dalian.

[12] Y. Zhu, F. Wang, X. Shan, X. Lv, "K-medoids Clustering based on MapReduce and Optimal Search of Medoids", *Proceeding of 9th International Conference on Computer Science and Education*, August 22, 2014, Vancouver.

[13] Y. Tamura and S. Miyamoto, "Two-Stage Clsutering Using One-Pass K-medoids and Medoid-based Agglomerative Hierarchical Algorithms", Proceeding of Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS), December 3, 2014, Kitakyushu.

[14] Y. Jiang and J. Zhang, "Parallel K-medoids Clsutering Algorithm Based on Hadoop", *Proceeding of IEEE 5th International Conferenceon Software Engineering and Service Science*, June 27, 2014, Beijing.

[15] M. Aryuni, E.D. Madyatmadja, and E. Miranda, "Customer Segmentation in XYZ Bank Using K-means and K-medoids Clustering", Proceeding of 2018 International Conferenceon Information Management and Technology (ICIMTech), September 3, 2018, Jakarta.

[16] S. Dave and P. Maheta, "Utilizing Round Robin Concept for Load Balancing Algorithm at Virtual Machine Level in Cloud Environment", *International Journal of Computer Applications*, vol. 94(4), 2014, pp. 23-29.

[17] N. Pasha and R. Rastogi, "Round Robin Approach for VM Load Balancing Algorithm in Cloud Computing Environment", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 4(5), 2014.

[18] D.C. Devi and V.R. Uthariaraj, "Load Balancing in Cloud Computing Environment Using Improved Weighted Round Robin Algorithm for Nonpreemptive Dependent Tasks", *The Scientific World Journal*, 2016, pp. 1-14.

[19] Y. Afrianto, H. Sukoco, and S. Wahjuni, "Weighted Round Robin Load Balancer to Enhance Web Server Cluster in OpenFlow Networks", *Telkomnika*, vol. 16(3), 2018, pp. 1402-1408.