CHAPTER VI

ALL ITEM-OUTLETS IN THE SAME OUTLET HEURISTIC (AIOOH)

6.1 Introduction

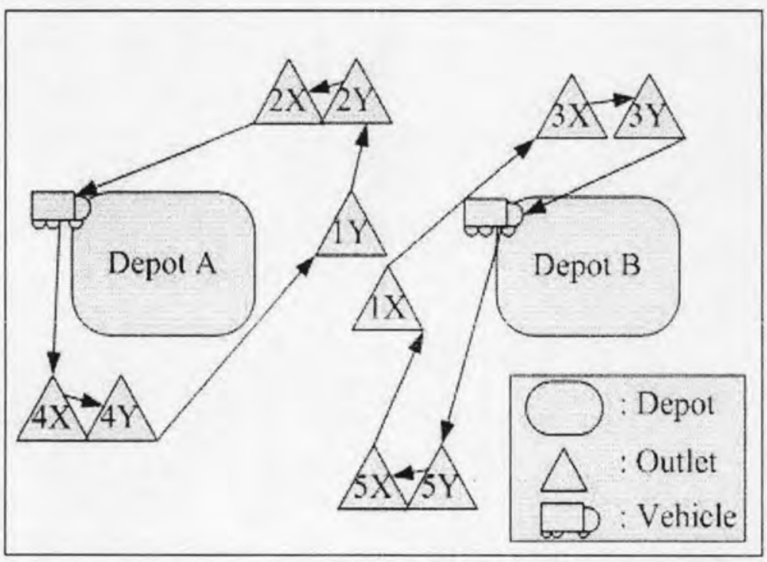
Since the results of the heuristic SIOH proposed in chapter V showed that the heuristic SIOH can slightly improve the solution obtained by Lot-For-Lot policy. Moreover, there are a few cases that SIOH can improve the solution. As a consequence, this chapter develops another heuristic for the multi-item multi-depot inventory routing problem to deal with the case that the heuristic SIOH can not improve total cost. The new heuristic named "AIOOH" is developed follow the analysis of factors for route cost reduction. With an example and illustrations, the analysis of factors will be presented during this chapter.

The remainder of this chapter is organized as follows. Disadvantages of the heuristic SIOH is presented in Section 6.2. The development of AIOOH is presented in Section 6.3. Computational results are reported in Section 6.4. The conclusions of this work and some recommendations for the next are presented in Section 6.5.

6.2 Disadvantages of the heuristic SIOH

According to the results from chapter V, the heuristic SIOH can slightly improve solution obtained from Lot-For-Lot policy. The improvement will occur only in cases that an item-outlet is assigned to a route which is different from the route that serves other item-outlets in the same outlet. An example of these cases is presented in Figure 6.1. The Figure 6.1 shows that there are two routes in the example. The first one starts from depot "A" and visits "4X", "4Y", "1Y", "2Y", "2X" and return to

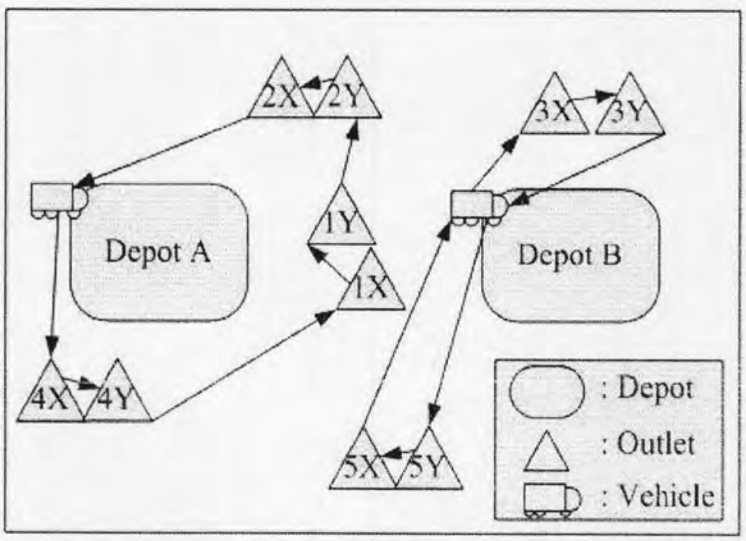
"A". The last one starts from depot "B" and visits "5Y", "5X", "1X", "3X", "3Y" and return to "B". It can be noticed that the item-outlet "1X" and "1Y" which are in the same outlet named "1" are separately assigned to different route. On the other hands, the cases that every item-outlet is assigned in the same route with another item-outlet which is in the same outlet of it, the SIOH can not improve solution from Lot-For-Lot. An example of these cases is presented in Figure 6.2. The Figure 6.2 shows that there are two routes in the example. The first one starts from depot "A" and visits "4X", "4Y", "1X", "1Y", "2Y", "2X" and return to "A". The last one starts from depot "B" and visits "5Y", "5X", "3X", "3Y" and return to "B". It can be noticed that the item-outlet "1X" and "1Y" which are in the same outlet named "1" are assigned to the same route.



Route 1	A-4X-4Y-1Y-2Y-2X-A
Route 2	B-5Y-5X-1X-3X-3Y-B

Figure 6.1 Example of cases that item-outlets in the same outlet are assigned to different route

In order to understand the reason that the SIOH does not work, the concept of the heuristic is revisited. The concept of the heuristic is to iteratively improve the solution from initial solution which is the solution obtained by Lot-For-Lot policy. The heuristic will improve solution by solving the Lot-sizing problem of each item-outlet to obtain replenishment quantity of them. Basically, the Lot-sizing problem is the problem of finding replenishment quantity and period of considered item on an outlet by balancing the holding cost and ordering cost. However, there is a difference between the modified Lot-sizing problem used by the heuristic and the basic Lot-sizing problem in ordering cost. The ordering cost in the basic problem is fixed and identical for all periods while the ordering cost in the modified Lot-sizing problem is dynamic. It depends on the route that the considered item-outlet is assigned to. Because the value of the ordering cost is dynamic, the heuristic SIOH has to approximate its value for solving the modified Lot-sizing problem. The ordering cost of any item-outlet in each period is approximated by calculating the difference on routing cost of the route that contains the considered item-outlet and the route that does not contain it.

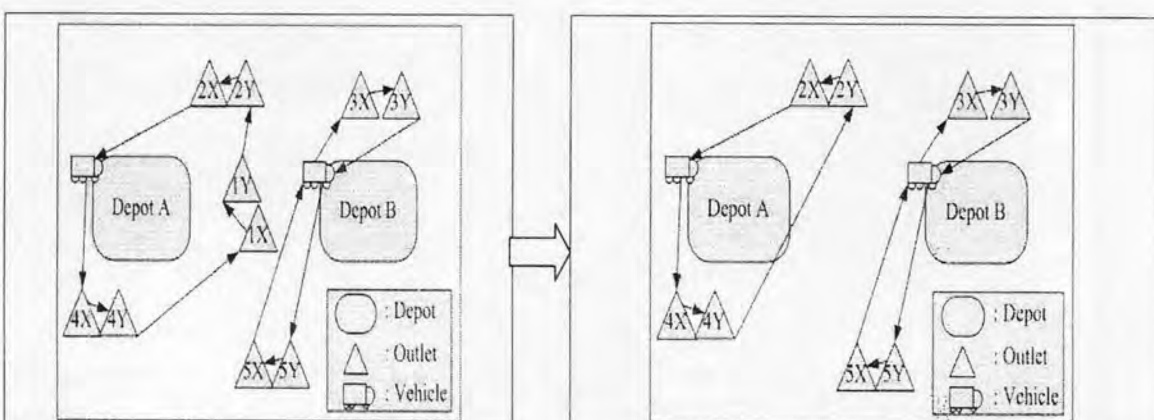


Route 1	**A-4X-4Y-1X-1Y-2Y-2X-A**
Route 2	**B-5Y-5X-3X-3Y-B**

Figure 6.2 Example of cases that all item-outlets in the same outlet are assigned to the same route

However the result of this approximation method leads to the zero ordering cost of all item-outlets in the cases that every item-outlet is assigned in the same route with another item-outlet which is in the same outlet. It is due to there are no difference on route cost between the route that contains the any considered item-outlet and the route that does not contain it. The reason is that even though the item-outlet is not considered to be routed, other item-outlets which are in the same outlet still remain in the route so the vehicle which serve this route will visit the same outlet to make a replenishment to them. Therefore, the routes that contain and do not contain the item-outlet remain visiting the same group of outlets so the total distance of route and route cost remain the same.

With the zero ordering cost the solution of Lot-sizing problem will be the solution obtained by Lot-for-Lot policy because the solution will result in minimum total cost in Lot-sizing problem. The minimum cost is due to the ordering cost is zero and with Lot-For-Lot policy the holding cost is zero also so the solution obtained by Lot-For-lot will be the optimal solution under this approximation method of the heuristic SIOH. It can be concluded that the heuristic SIOH usually results in no improvement on total cost from solution Lot-For-Lot because its method that considers replenishment quantity of an item-outlet. However it can be believed that if another heuristic is developed using similar concept as SIOH except it will consider on a group of item-outlets instead of an item-outlet, it may result in the better solution than solution from SIOH. Since the consideration on a group of item-outlets especially all item-outlets which are in the same route and in the same outlet may cause the change in route cost. For example by removing a group of item-outlets composing of "1X" and "1Y" as shown in Figure 6.3, the first route is changed from "A-4X-4Y-1X-1Y-2Y-2X-A" into "A-4X-4Y-2Y-2X-A" so the route cost is changed from 100 to 80 because the vehicle does not visit the outlet "1" and the approximated ordering cost of the group of "1X" and "1Y" is 100-80 = 20 units



Route 1	**A-4X-4Y-1X-1Y-2Y-2X-A** **Route Cost = 100**
Route 2	**B-5Y-5X-3X-3Y-B**

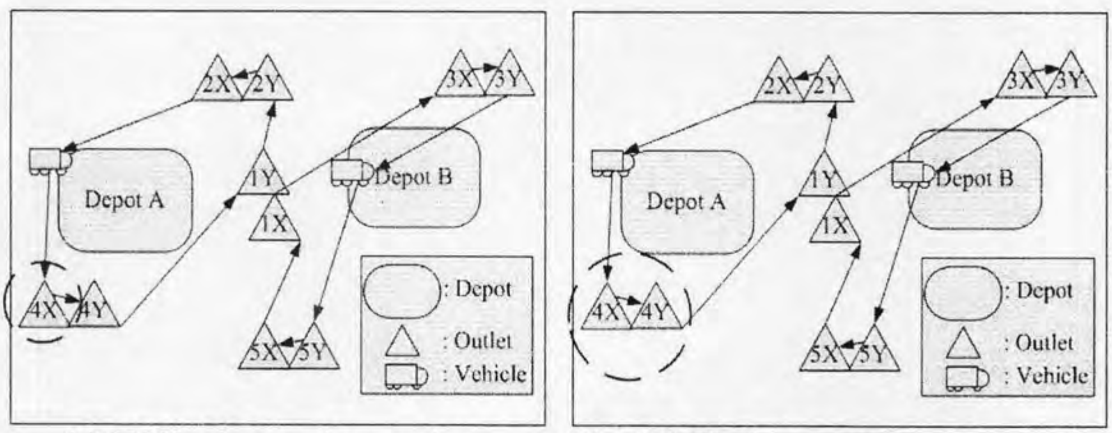
Route 1	**A-4X-4Y-2Y-2X-A** **Route Cost = 80**
Route 2	**B-5Y-5X-3X-3Y-B**

Figure 6.3 Example of removing a group of item-outlets

6.3 Development of AIOOH

6.3.1 Heuristic description

The heuristic procedure of AIOOH will start from initial solution by Lot-For-Lot as the procedure of SIOH. The difference between the heuristic AIOOH and SIOH is mainly in the Step 2. The Step 2 of AIOOH will randomly select a group of item-outlets which are in the same route and in the same outlet while the SIOH will select an item-outlet. The illustration of difference on the selection procedure of the two heuristic is shown in Figure 6.4. Figure 6.4(a) shows the selection procedure of SIOH that select the item-outlet name "4X" while Figure 6.4(b) shows that AIOOH select a group of item-outlet "4X" and "4Y" which is in the same route and in the same outlet named "4". In the later procedure of AIOOH, the replenishment quantity of this group of item-outlets will be determined by solving the Lot-sizing problem of this group.



(a) SIOH selects an item-outlet (b) AIOOH selects a group of item- outlet

Figure 6.4 Difference between SIOH and AIOOH in item-outlet selection process

AIOOH modified the shortest path network that is used by SIOH to solve the Lot-sizing problem of a group of item-outlets. The modification is that an arc from k to t means considered inventory level of a group of item-outlets is replenished at period k and then period t consecutively. Moreover, all arcs in the shortest path network must not violate both vehicle carrying capacity and outlet storage capacity. AIOOH considers the violation of vehicle capacity and storage capacity with similar method to the SIOH however there are a few differences among them. For vehicle carrying capacity, AIOOH will check that any arc will not violate vehicle carrying capacity by pre-assigning the group of item-outlet into an origin period of arc. For example, follow the data presented in Table 6.1, an arc "2-4" of the group of item-outlet named "1X" and "1Y" which means there are deliveries at period 2 and then period 4. This arc must be checked by re decision about route in period 2 with addition information of replenishment quantity of each item-outlet in this group with equal to cumulative demand of it from period 2 to period 3. The new route will be acceptable as a feasible route if carrying capacity is not exceeded and this arc will be considered as an available arc. The illustration of the case that vehicle capacity is met is shown in Figure 6.5 and the case that vehicle capacity is violated in shown in Figure 6.6. In the Figure 6.5, the arc 1-3 of item-outlet "1X" is considered by using summation of demand from period 1 to period 2 as replenishment quantity of "1X" and "1Y" in period 1. This amount of replenishment quantity is considered as delivery load in routing decision. In the Figure 6.6, the arc 1-4 of the group of item-outlet "1X" and "1Y" is considered and results in violation of vehicle capacity.

Table 6.1 Example of initial solution

Item-Outlet \ Period	1	2	3
1X	12	8	10
1Y	11	9	10
2X	10	8	9
2Y	8	10	11
3X	12	8	10
3Y	11	9	10
4X	10	8	9
4Y	8	10	11
5X	10	8	9
5Y	8	10	11
Route1	A-4X-4Y-2Y-2X-A	A-4X-4Y-2Y-2X-A	A-4X-4Y-2Y-2X-A
Route2	B-3X-3Y-1X-1Y-5Y-5X	B-3X-3Y-1X-1Y-5Y-5X-B	B-3X-3Y-1X-1Y-5Y-5X-B

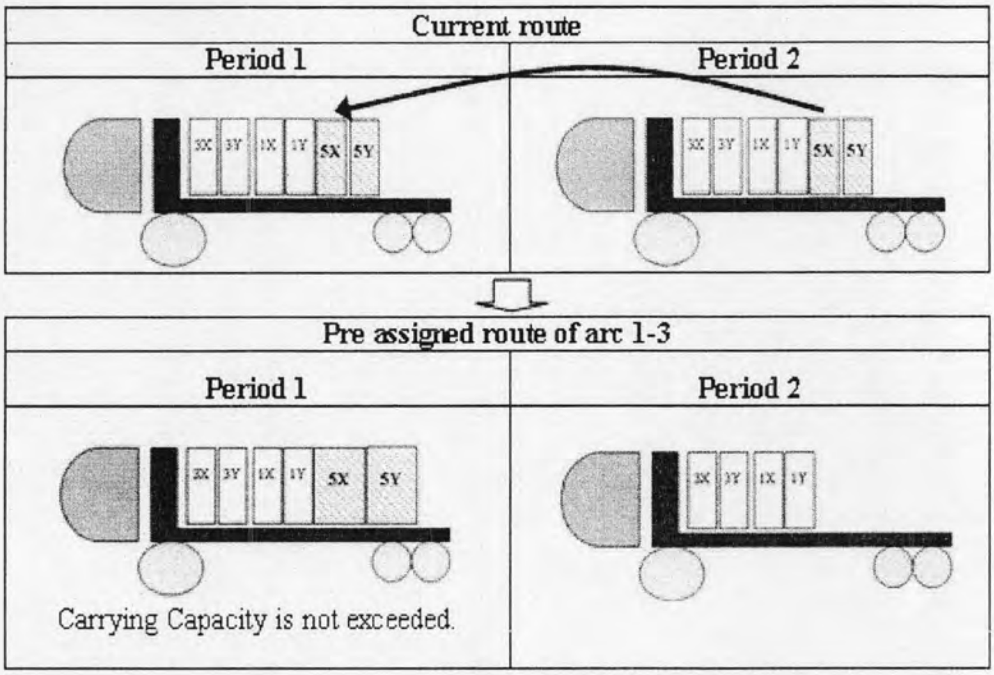


Figure 6.5 Illustration of the case that vehicle capacity is met

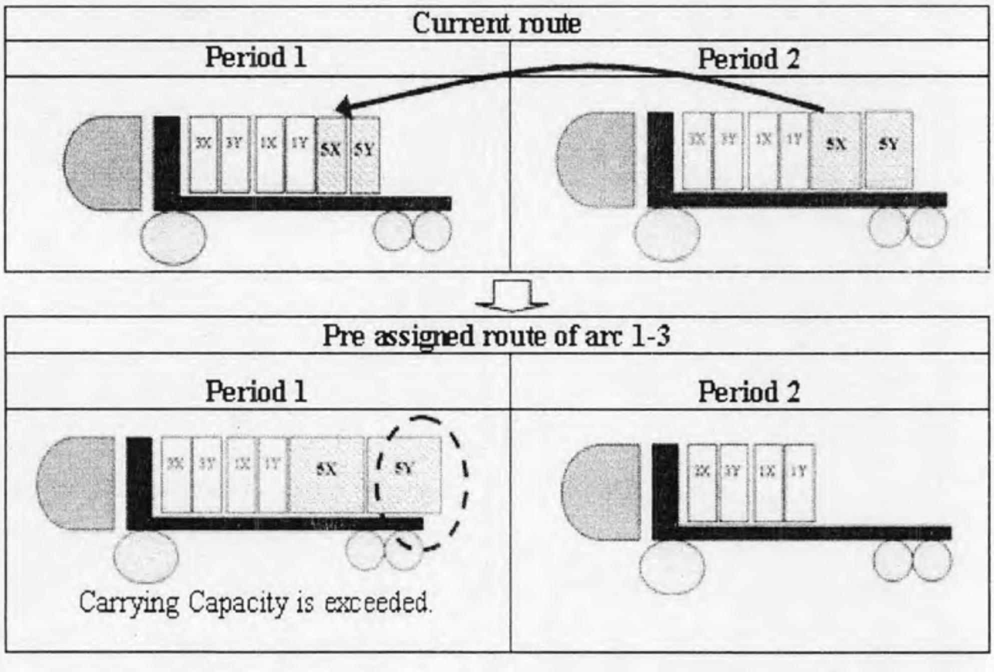


Figure 6.6 Illustration of the case that vehicle capacity is violated

For outlet storage capacity, the outlet must be checked by comparing the summation of inventory level of all item-outlets which are in it with its storage capacity. The arc will be existed if the summation of inventory level does not exceed the outlet storage capacity. The inventory level is equal to the replenishment quantity subtract by the demand in consider period.

6.3.2 Pre-solved procedure

According to the results of the heuristic SIOH proposed in chapter V, there are many cases that the heuristic can not improve the solution obtained by Lot-For-Lot policy. In addition of the reason that described in section 6.2, there are some cases that there is no improvement due to the structure of problem. Therefore, this section analyzed some structures of problems by consider factors that the total cost can be improved in order to develop the pre-solved procedure. The pre-solved procedure can avoid the unnecessary use of the proposed heuristic in cases that the heuristic achieves insignificant improvement of solution from Lot-For-Lot. The consideration on factors is the analysis of the cases that routing cost can be reduced.

It is founded that in cases of the remaining carrying capacity of at least one vehicle is enough for at least summation of demand of a group of item-outlets which if they are removed from route in later period they will cause the reduction on routing on that period. The remaining capacity of vehicle must be enough to take this amount of demand of this group of item-outlets. Especially, the cases that the group of item-outlets has already routed in prior period, the cost reduction will depend mainly on the remaining capacity. The illustration of this case will be provided in Figure 6.7. In the other hand, the cases that remaining capacity is not enough it may cause the additional vehicle for service additional demand and may result in increasing on transportation cost. This case can be shown in Figure 6.8

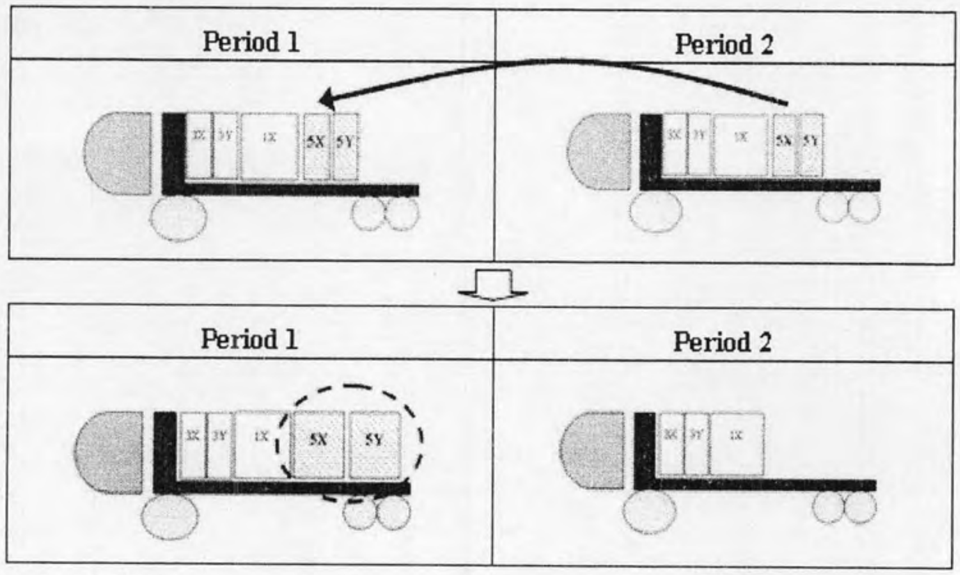


Figure 6.7 Illustration of the case that remaining capacity is enough for demand of a group of item-outlets

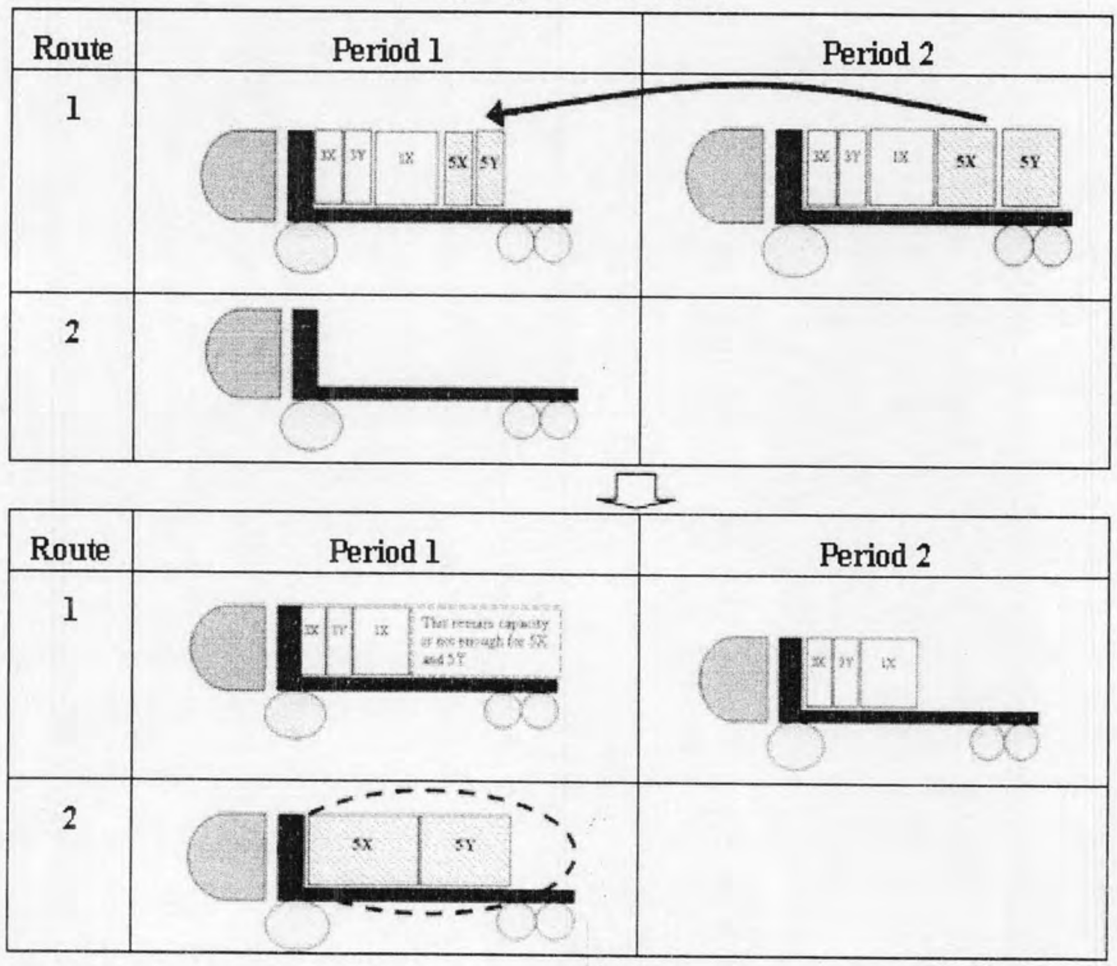


Figure 6.8 Illustration of the case that remaining capacity is not enough for demand of a group of item-outlets

In summary after the initial solution is obtained, the pre-solved procedure will compare the remaining carrying capacity with the summation of demand of any group of item-outlets. In case that the remaining capacity can be carry the summation of demand of any group of item-outlets, the considered problem will not be solved by the proposed heuristic because there are high possibility that the heuristic can not improve the initial solution from Lot-For-Lot.

6.3.3 Numerical example

In this section, a simple problem is illustrated. This example consists of two depots and five outlets with two items in each outlet. Each depot has a vehicle for delivery items to outlets. All vehicles have a capacity limit at one hundred units Decisions are made for three periods. All outlets are transformed into item-outlets and shown in Figure 6.9. Figure 6.9 illustrates the graphical location of all depots and item-outlets in the example problem. Table 6.2 shows the distance between all locations. Demand of all outlets is given in Table 6.3. Table 6.4 gives the information about the holding cost of the outlets. Beginning inventory of all outlets are assumed to be zero.

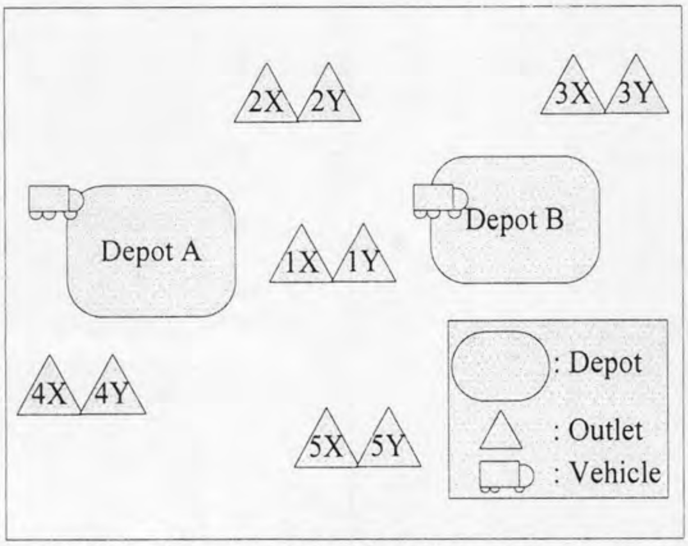


Figure 6.9 Illustration of the graphical location of all depots and outlets in the example problem.

Table 6.2 Distance between all locations

		Depot		Outlet				
		A	B	1	2	3	4	5
Depot	A	0	3	7	10	9	9	11
	B	3	0	9	12	7	7	13
Outlet	1	9	7	0	13	16	11	16
	2	12	10	13	0	15	19	4
	3	7	9	16	15	0	11	13
	4	7	9	11	19	11	0	20
	5	13	11	16	4	13	20	0

Table 6.3 Demand of all outlets

		Period Item	1	2	3
Outlet	1	X	20	19	20
		Y	19	20	19
	2	X	20	21	20
		Y	17	20	21
	3	X	21	22	20
		Y	24	20	15
	4	X	21	18	25
		Y	20	20	21
	5	X	16	20	23
		Y	22	22	17

Table 6.4 Holding cost of all outlets

		Item	Holding Cost
Outlet	1	X	0.10
		Y	0.15
	2	X	0.25
		Y	0.30
	3	X	0.07
		Y	0.10
	4	X	0.35
		Y	0.40
	5	X	0.14
		Y	0.17

Step 1: Set up an initial phase

The example of initial solution is presented in Table 6.5 and Figure 6.10. Table 6.5 presents the replenishment quantity of each item-outlet, for example item-outlet "1X" will receive replenishment quantity 12 units at period 1, 8 units at period 2 and 10 units at period 3. There are two delivery routes for all periods. The first route starts from Depot "B" then visit "5Y", "5X", "1X", "3X", "3Y" and return to Depot "B". The second route starts from Depot "A" then visit "4X", "4Y", "1Y", "2Y", "2X" and return to Depot "A".

Table 6.5 Replenishment quantity and delivery route of initial solution

Item-Outlet \ Period	1	2	3
1X	12	8	10
1Y	11	9	10
2X	10	8	9
2Y	8	10	11
3X	12	8	10
3Y	11	9	10
4X	10	8	9
4Y	8	10	11
5X	10	8	9
5Y	8	10	11
Route1	B-5Y-5X-1X-3X-3Y-B	B-5Y-5X-1X-3X-3Y-B	B-5Y-5X-1X-3X-3Y-B
Route2	A-4X-4Y-1Y-2Y-2X-A	A-4X-4Y-1Y-2Y-2X-A	A-4X-4Y-1Y-2Y-2X-A

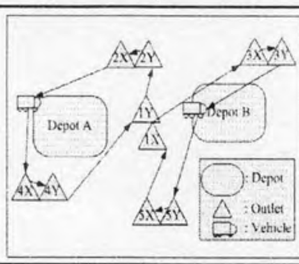
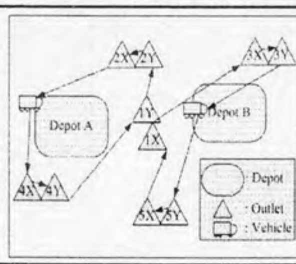
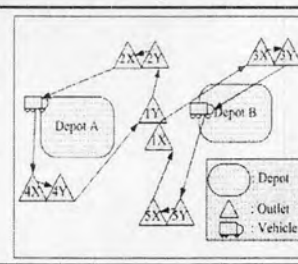
Period	**1**	**2**	**3**
Route			

Figure 6.10 Initial route

Step 2: select an outlet

The illustration of selection of an item-outlet is shown in Figure 6.11. In Figure 6.11, a set of item-outlet "4X" and "4Y" is selected to be considered for determining replenishment quantity of it. The shortest path network is used to determine the replenishment quantity.

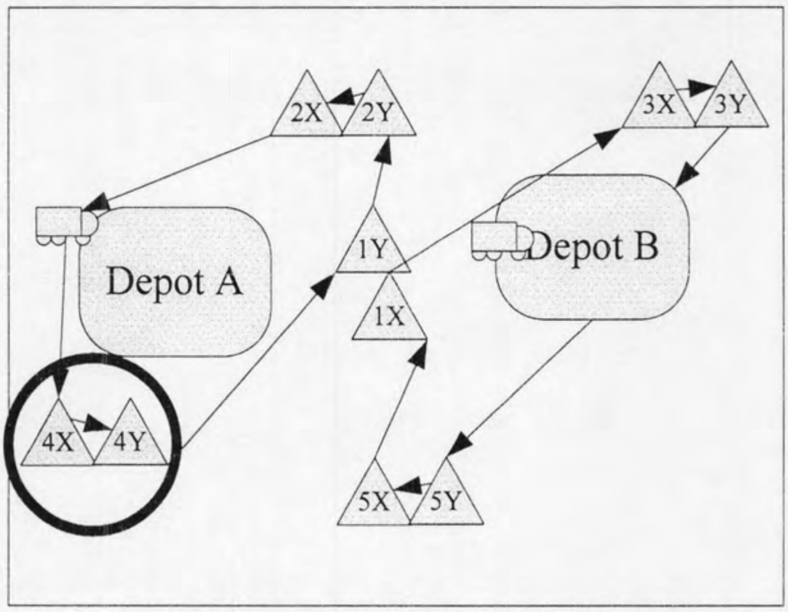


Figure 6.11 A set of item-outlet "4X" and "4Y" is selected

Step 3: Determine delivery period using shortest path algorithm.

Step 3.1 Define Node and Arc.

The illustration of shortest path network is presented in the Figure 6.12.

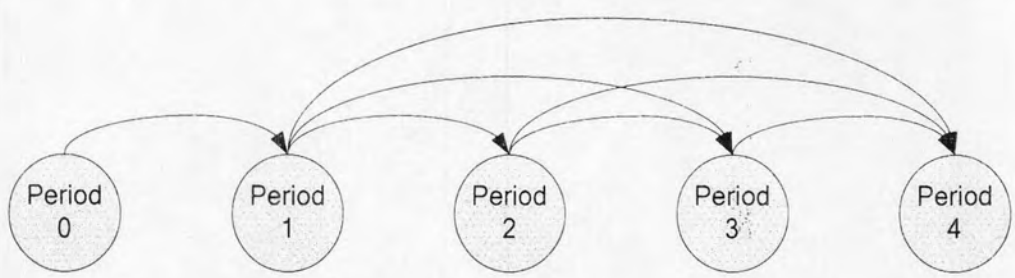


Figure 6.12 Illustration of shortest path network

For Figure 6.12, each circle stands for node, period 0, 1, 2, …, 4 and each arrow line stands for arc, replenishment period 0-1, 1-2,…, 3-4. Hence, we will name node 0, node 1, …, and node 4, and arc 0-1, arc 1-2, arc 1-3, …, and arc 3-4. The meaning of arc k-t is there is replenishment at period t after replenishment at period k. For example arc 2-3 means the set of item-outlet "4X" and "4Y"" is replenished at period 3 after replenishment at period 2. For the case of arc 0-t means the set of item-outlet "4X" and "4Y" is first replenished at period t ex. an arc 0-1 means the first replenishment to the set of item-outlet "4X" and "4Y" is made at period 1. For the case of arc k-4, which node 4 is the last node in this problem, means the set of item-outlet "4X" and "4Y" is last replenished at period k ex. an arc 2-4 means the last replenishment to the set of item-outlet "4X" and "4Y" is made at period 2 to cover demand of period 2 to period 3. It can be notice that there are no arc 0-2, 0-3 and 0-4 because these arcs violate the stock out constraint. For example arc 0-2 violates the stock out constraint since the first replenishment is made at period 2 which the demand at period 1 is not been satisfied.

Step 3.2 Define Arc cost.

Cost of every arc is the summation of increasing holding cost and the difference of route cost of selecting the considered arc. An increasing holding cost of arc k-t is determined by sum all inventory holding cost of the considered outlet from period t to period k+1. Rout cost is determined by calculating the difference between route cost including considered this outlet and route cost without considered this outlet at period k. The illustration of arc cost of shortest path network of the set of item-outlet "4X" and "4Y" is shown in Figure 6.13.

Table 6.6 Holding Cost of Replenishment Period following Arc

To Period / From Period	1	2	3	4
0	0	-	-	-
1	-	0	8x0.35+10x0.4=6.8	(17+9)x0.35+(21+11)x0.4 =21.9
2	-	-	0	9x0.35+11x0.4=7.55
3	-	-	-	0

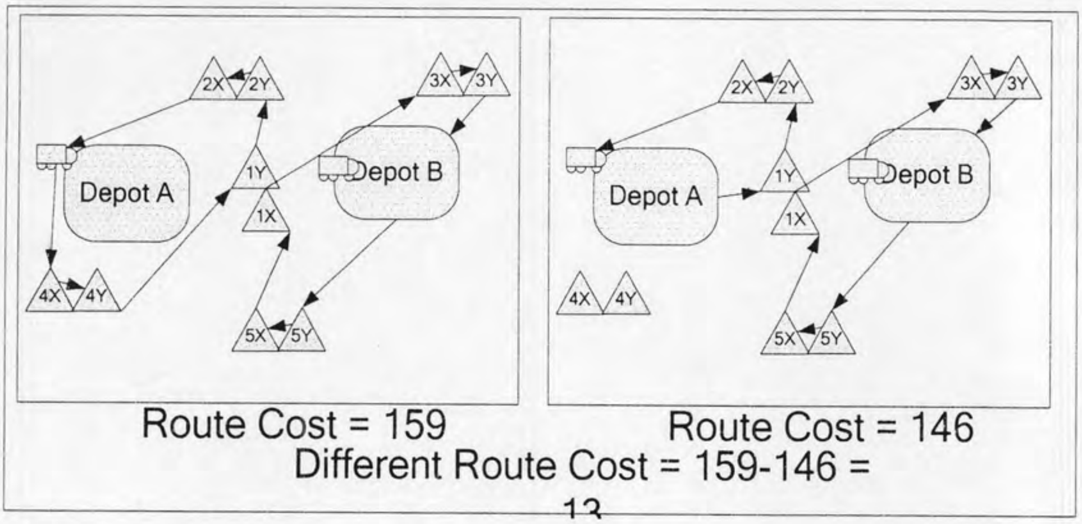


Figure 6.13 Illustration of different route cost in period 2

Table 6.7 Arc Cost of set of item-outlet "4X" and "4Y"

To Period / From Period	1	2	3	4
0	0	-	-	-
1	-	0+13=13	6.8+13=19.8	21.9+13=34.9
2	-	-	0+13=13	7.55+13=20.55
3	-	-	-	0+13=13

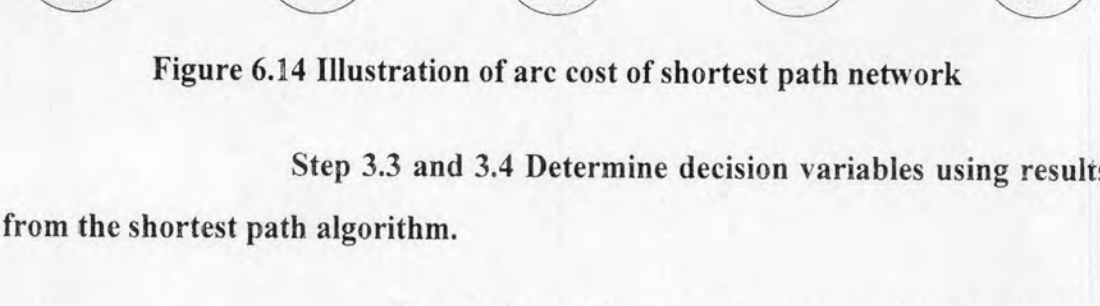


Figure 6.14 Illustration of arc cost of shortest path network

Step 3.3 and 3.4 Determine decision variables using results from the shortest path algorithm.

The solution of shortest path problem of the set of item-outlet "4X" and "4Y" in this iteration is 0-1-3-4 so there is delivery schedule for "1" on period 1 and period 3.

Step 4: Update replenishment quantity and route

The solution after updating is shown in Table 6.8 and Figure 6.15. In Table 6.8, the replenishment quantity is presented. It can be indicated that the replenishment quantity of the set of item-outlet "4X" and "4Y" in period 1 and 2 are changed. For the item-outlet "4X", the value at period 1 is 18 which is derived from summation of demand from period 1 and period 2. The period 2 has no replenishment quantity since the replenishment quantity in period 1 can cover demand in this period already. For the value at period 3 of "4X" is 9. In Figure 6.15 shows the illustration of delivery route in each period. The second route in period 2 is changed from previous state. The second route in period 2 becomes A-1Y-2Y-2X-A that means the vehicle starts from depot indexed by "A" and goes to item-outlet "1Y", "2Y", "2X" and return to "A".

Table 6.8 Replenishment quantity and delivery route after updating

Period / Item-Outlet	1	2	3
1X	12	8	10
1Y	11	9	10
2X	10	8	9
2Y	8	10	11
3X	12	8	10
3Y	11	9	10
4X	10+8=18	0	9
4Y	8+10=18	0	11
5X	10	8	9
5Y	8	10	11
Route1	B-5Y-5X-1X-3X-3Y-B	B-5Y-5X-1X-3X-3Y-B	B-5Y-5X-1X-3X-3Y-B
Route2	A-4X-4Y-1Y-2Y-2X-A	A-1Y-2Y-2X-A	A-4X-4Y-1Y-2Y-2X-A

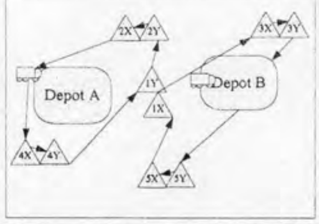
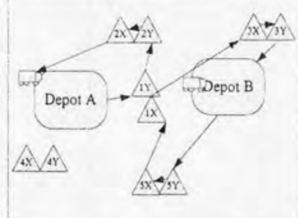
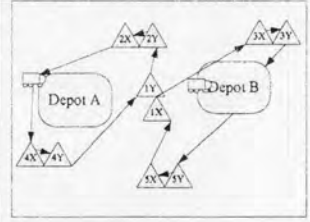
Period	**1**	**2**	**3**
Route			

Figure 6.15 Route after updating

6.4 Experiment and result

The performance of the heuristic AIOOH is tested by the problem instances described in chapter V and compare to the solution obtained by Lot-For-Lot policy.

6.4.1 Performance of algorithms on small-sized test problems

The purpose of these experiments is to evaluate the performance of the proposed algorithm on the small-sized test problems that CPLEX can find solution within 3 hours. The test instances which are described in chapter V are used for performance evaluation of AIOOH. For the performance testing of small-sized problem, this dissertation compares solution from proposed heuristics with solution from CPLEX and from Lot-For-Lot policy (LFL). The performance of heuristics will be tested in two aspects; solution quality, and computational time. The performance of heuristics will be presented in form of the percentage deviation of a particular heuristic from the solution of CPLEX and Lot-For-Lot policy (LFL). Moreover, the performance of heuristic AIOOH will be compared to SIOH in order to show the improvement of AIOOH over SIOH which can be presented in form of percentage deviation from SIOH. The calculation of percentage deviation from SIOH is shown in the following equation:

$$\% \text{ difference from SIOH} = \frac{Sol_{SIOH} - Sol_{AIOOH}}{Sol_{SIOH}} \times 100 \qquad (6.1)$$

where Sol_{AIOOH} is the solution obtained by heuristic AIOOH, and Sol_{SIOH} is the solution obtaining by using heuristic AIOOH.

The heuristic have been implemented in the Visual Basic 6 programming language on a PC with an Intel Pentium 4 1.8 GHz CPU and 256 MB of RAM. The solution obtained by means of the 0-1 mixed linear integer programming formulation given in Chapter III is found by AMPL/CPLEX 8.0.0 solver. However, the CPU time is limited to at most 3 hours.

The performance of heuristic are presented by overall performance of each heuristic for solving problem in all parameter sets, the effect of parameter and interaction of parameter on performance of heuristic. The overall performances of heuristic are presented in Table 6.9-6.11. From the results, the heuristic AIOOH obtains solution with 11.54%-19.31% average improvement from Lot-For-Lot policy and 1.59%-12.01% from solution solved by CPLEX. It can be noticed from table of result that the computation time of heuristic is much better than time consumed by CPLEX. The results in Table 6.11 indicate that the AIOOH can perform better than SIOH between 10.1% and 14.69% in average while AIOOH consumes a few more time than SIOH.

Table 6.9 Overall Performance of AIOOH for small-sized problem compared to LFL

Problem	Improvement (%) vs. Lot-for-Lot			CPU Time (Sec.)	
Name	Max	Min	Average	Lot-for-Lot	Heuristic
E-n4-k2	29.44%	2.50%	11.54%	0.01	0.07
E-n5-k2	38.54%	5.00%	19.31%	0.00	0.09
E-n7-k2	35.18%	4.83%	15.78%	0.01	0.18

Table 6.10 Overall Performance of AIOOH for small-sized problem compared to CPLEX

Problem	Improvement (%) vs. CPLEX			CPU Time (Sec.)	
Name	Max	Min	Average	CPLEX	Heuristic
E-n4-k2	17.48%	0.12%	3.48%	10800	0.07
E-n5-k2	5.94%	0.00%	1.59%	10800	0.09
E-n7-k2	25.33%	1.17%	12.01%	10800	0.18

Table 6.11 Overall Performance of AIOOH for small-sized problem compared to SIOH

Problem	Difference (%) on Total Cost			CPU Time (Sec.)	
Name	Max	Min	Average	SIOH	AIOOH
E-n4-k2	25.44%	0.41%	10.10%	0.07	0.07
E-n5-k2	38.54%	0.76%	14.69%	0.09	0.09
E-n7-k2	35.18%	4.83%	14.58%	0.15	0.18

6.4.2 Performance of algorithms on medium- and large-sized test problems

The purpose of these experiments is to evaluate the performance of the proposed algorithm on the medium- and large-sized test problems that CPLEX can not find solution within 3 hours. The test instances which are described in chapter V are used for performance evaluation of AIOOH.

For the performance testing of medium- and large-sized problem, this dissertation compares solution from proposed heuristics with solution from Lot-For-Lot policy. The performance of heuristics will be tested in two aspects; solution quality, and computational time. The performance of heuristics will be presented in form of the percentage improvement of a particular heuristic from the solution of Lot-For-Lot (LFL). From the results in Table 6.12, it can be observed that the proposed algorithm can yield a good outcome for medium-sized problem. The heuristic AIOOH obtains solution with 9.45%-21.67% average improvement from Lot-For-Lot policy. The results in Table 6.12 show that the heuristics consume computation time more than Lot-For-Lot policy. However the computational time is not over than 24.51 second which is acceptable. The results in Table 6.13 indicate that the AIOOH can improve solution from SIOH between 4.88% and 21.15% in average while AIOOH consumes a few more time than SIOH.

Table 6.12 Overall Performance of AIOOH for medium-sized problem compared to LFL

Problem	Improvement (%) vs. Lot-for-Lot			CPU Time (Sec.)	
Name	Max	Min	Average	Lot-for-Lot	Heuristic
E-n13-k4	38.09%	5.69%	21.67%	0.05	0.57
E-n22-k4	41.78%	1.69%	19.16%	0.22	5.50
E-n31-k7	23.71%	2.18%	9.45%	0.77	18.09
E-n33-k4	30.48%	2.27%	15.48%	0.49	24.51

Table 6.13 Overall Performance of AIOOH for medium-sized problem compared to SIOH

Problem	Difference (%) on Total Cost			CPU Time (Sec.)	
Name	Max	Min	Average	SIOH	AIOOH
E-n13-k4	38.09%	5.69%	21.15%	0.80	0.57
E-n22-k4	35.33%	0.74%	15.57%	7.42	5.50
E-n31-k7	15.67%	0.41%	4.88%	29.73	18.09
E-n33-k4	29.81%	0.15%	11.64%	48.79	24.51

From the results in Table 6.14, the heuristic AIOOH obtains solution with 12.93%-14.52% average improvement from Lot-For-Lot policy. The results in Table 6.14 show that the heuristics consume much more computation time than Lot-For-Lot policy. However, the reduction on total cost is worthy for large-sized problem in which amount of saving on total cost is much more than in small- and medium-sized problem. Hence the increase of computational time to achieve the better solution for these cases is acceptable as long as the computational time does not exceed the reasonable level. The results in Table 6.15 indicate that the AIOOH can perform better than SIOH between 8.29% and 13.78% in average while AIOOH consumes a few more time than SIOH.

Table 6.14 Overall Performance of AIOOH for large-sized problem compared to LFL

Problem	Improvement (%) vs. Lot-for-Lot			CPU Time (Sec.)	
Name	Max	Min	Average	Lot-for-Lot	Heuristic
E-n51-k5	27.00%	3.73%	12.93%	2.35	85.08
E-n76-k7	27.88%	3.21%	14.52%	7.39	213.55
E-n101-k8	33.96%	3.39%	14.35%	18.94	604.65

Table 6.15 Overall Performance of AIOOH for large-sized problem compared to SIOH

Problem	Difference (%) on Total Cost			CPU Time (Sec.)	
Name	Max	Min	Average	SIOH	AIOOH
E-n51-k5	25.66%	1.48%	10.51%	137.08	85.08
E-n76-k7	25.67%	1.53%	13.78%	383.92	213.55
E-n101-k8	25.84%	0.56%	8.29%	1057.52	604.65

6.5 Conclusion

This chapter begins with the analysis of the disadvantages of the heuristic SIOH. After that the development of the heuristic AIOOH which is the improved version of SIOH is presented. This chapter also provides the analysis of the factors which the route cost can be reduced in the section of development of AIOOH. The computational results of AIOOH are provided and discussed in later section. The results show that the heuristic AIOOH can perform better than SIOH in all cases and consumes less time. It can be concluded that the heuristic AIOOH is outperformed SIOH. It is due to the modification in item-outlet selection. AIOOH selects a group of item-outlet which is in the same route and in the same outlet while SIOH selects an item-outlet. This difference of the two heuristics results in much different performance of them. However it is believed that there is an opportunity to improve AIOOH by considering a group of item-outlets in different criteria. The next chapter will present the heuristic that consider a group of item-outlets by consider all item-outlets which are in the same route. With consideration on this group of item-outlets may result in reduction on fixed-charge cost and route cost.