CHAPTER V

PARTIAL ASSIGNMENT- LOT SIZE HEURISTIC (PA-LS)

Although the research problem is very complicated as mentioned in chapter 1, the mathematical model has been formulated in the chapter 3. Unfortunately, the model, which is classified as MLCLSP problem, is NP-hard problem. Then the solution method is value to pursuit and need. A solution method is proposed in the chapter 4 which is A-LS heuristic. However, the heuristic still has some drawbacks with its solution performance. As a consequent, this chapter develops the new heuristic that can permit the more solution improvement. The proposed heuristic is called Partial Assignment – Lot size heuristic. With an example for illustration, the heuristic will be discussed at the end of this chapter.

The remainder of this chapter is organized as follows. The introduction is presented in section 5.1. The heuristics are described in Section 5.2. In Section 5.3, we present the heuristics procedures that were used. The computational results are given in Section 5.4. Finally, some concluding remarks are provided in Section 5.5.

5.1 Introduction

In the previous chapter, the Assignment-Lot size heuristic (A-LS) was proposed. Although it takes very computational time for solving, it gives only a satisfactory solution. Therefore, it's encouraged us to find the way to improve the solution. With the concept of lot sizing problem, it is obvious that Lot-for-Lot policy gives only no holding solution and, especially in the case of setup consideration, the solution with more holding has a potential to be the better solution. On basis of this characteristic, in this chapter propose the heuristic method that can change the Lot-for-Lot solution to the solution with consideration holding. The heuristic is called

Partial Assignment- Lot size (PA-LS). This concept based on believes that the closer to original assignment will give the better solution. Although the performance of a solution is better, the solving time consuming is also dramatically increase. Therefore, some of characteristics of the result will be discussed with an example of a problem.

5.2 Heuristics Description

Obviously, lot sizing with setup consideration has a significant trade-off between the setup and the holding. If the setup cost ($g_t^{i,k}$) is very high and the holding cost ($h_t^i \cdot x_t^{i,k}$) is low, the lot size should be as large as possible. In addition, the possible lot size with no stock consideration can be

$$x_t^{i,k} = \begin{cases} \left[0, \min\left\{\left(\dfrac{cap_t^k - g_t^{i,k}}{o_t^{i,k}}\right), \sum_{l=t}^{NT}\left(d_l^i\right)\right\}\right], \forall k \in WP \\ \left[0, \min\left\{Avail_t^k, \sum_{l=t}^{NT}\left(d_l^i\right)\right\}\right], \forall k \in WR \end{cases}$$

In this research, it is clear that the trade-off is depended on the assignment of item-workstation. Changing the assignment will change the availability, capacity, unit time, and unit cost parameters. Therefore, this heuristic will try to find the better assignment to make the suitable lot size.

5.3 Heuristic Procedures

As aforementioned, steps in A-LS heuristic is based on assigning and lot sizing. On given assignment leads to the same local optimum, therefore, there is a need to change the assignment matrix. This heuristic will relax the matrix with the original matrix on believing that the problem should be more closed to the solution of the original problem and will change the lot size to make the more suitable lot size. As the same as A-LS, the heuristics can be separated in two major parts which are the assigning solving and the lot sizing solving. The different is that after first iteration the assignment matrix will be relaxed to the original matrix. The detailed of relaxation will be discussed in the section of Phase 1. The overall flow of this heuristic is shown in Figure 5.1.

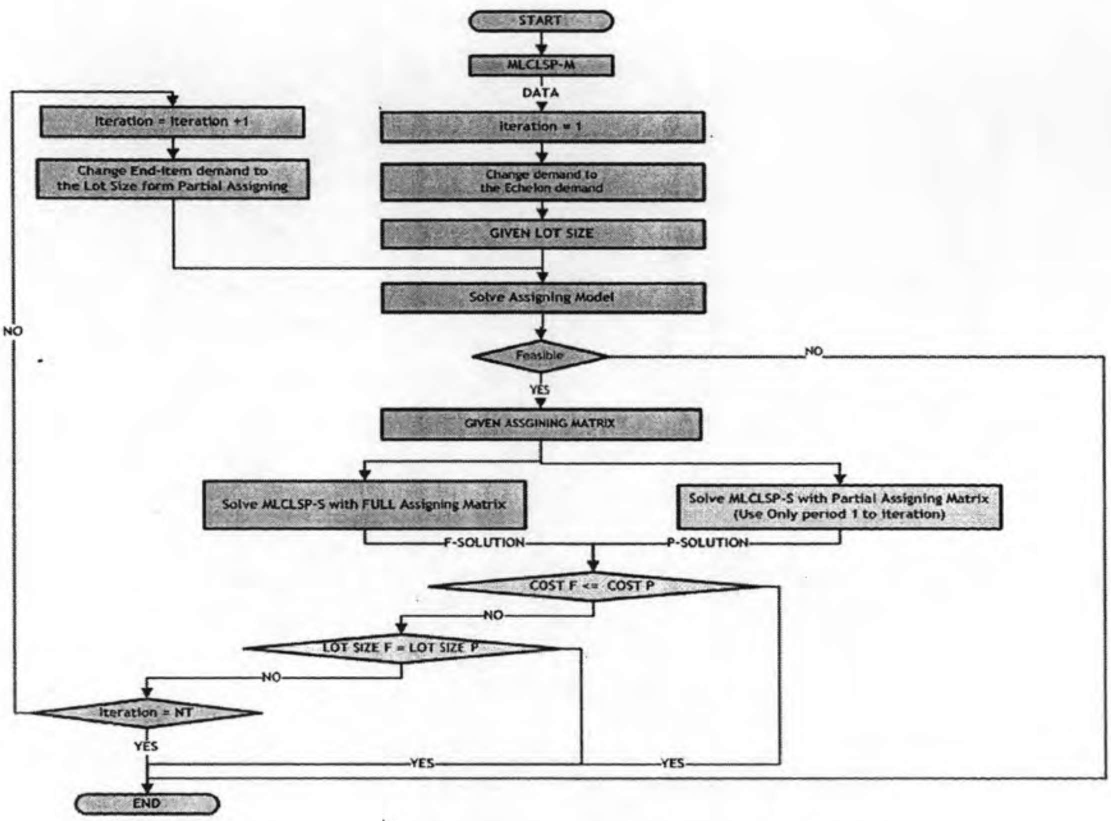

Figure 5.1 A flow diagram of this research's heuristics

5.3.1 Phase 1: Assignment problem with given lot-size

Let $w_t^{i,k}$ represents the status of assignment ($w_t^{i,k} = 1$ if in period t item i is assigned to be operated on workstation k, and $w_t^{i,k} = 0$ for otherwise) and D_t^i represents the given lot size. The formulated model can be represented as follows

$$Min \ \left(p_t^{i,k} \cdot D_t^i + f_t^{i,k}\right) \times w_t^{i,k} \qquad \text{(P1)}$$

Subject to

$$D_t^i = d_t^i + \sum_{j \in S(i)} u^{ij} D_t^j \ , \ \forall i \in I; \forall t \in T \qquad (5.1)$$

$$\left(\sum_{i \in I(k,t)} D_t^i\right) w_t^{i,k} \leq Avail_t^k \ , \ \forall k \in WR, \forall t \in T \qquad (5.2)$$

$$\left(\sum_{i\in I(k,t)}\left(o_t^{i,k}\cdot D_t^i+g_t^{i,k}\right)\right)w_t^{i,k}\le cap_t^k,\ \forall k\in WP,\forall t\in T \tag{5.3}$$

$$\sum_{k\in WS} w_t^{i,k}\le 1,\ \forall i\in I,\forall t\in T \tag{5.4}$$

$$\sum_{i\in I} w_t^{i,k}\le 1,\ \forall k\in WS,\forall t\in T \tag{5.5}$$

$$w_t^{i,k}\in\{0,1\},\ \forall i\in I,\forall k\in WS,\forall t\in T \tag{5.6}$$

In this section, we introduce a set $Q(t)$ to represent original assignment item-workstation in period t. Let $w_t^{i,k}$ represents the status of assignment ($w_t^{i,k}=1$ if in period t item i is assigned to be operated on workstation k, and $w_t^{i,k}=0$ for otherwise) and is calculated from the assignment model in the assignment with given lot size phase. Set $A(t)$, which is a set of the assignment item-workstation in period t, can easily defined by $A(t)=\{(i,k)\mid w_t^{i,k}=1\}$. Similarly, for the original problem, the assignment matrix has been already defined, as a result, $w_t^{i,k}$ can be defined by the equation (O) as follows:

$$w_t^{i,k}=\begin{cases}1 & \forall i\in I(k,t)\\ 0 & otherwise\end{cases} \tag{O}$$

Then the set $Q(t)=\{(i,k)\mid w_t^{i,k}=1\}$ is a set of the original assignment item-workstation in period t. The relaxation will begin in the second iteration. The assignment matrix will be used with relaxation to original problem assignment for all this iteration until max number of iteration. In another word, a given assignment matrix that will be sent to lot sizing part split into 2 types which are the "Full assigned" (as the result from assignment model with given lot size or set $A(t)$) and the "Partial assigned" (as the result from assignment model with given lot size only in iteration period or set $A(t)$, other periods will be replaced with the original assignment of the problem or set $Q(t)$).

The result of this part is the assignment matrix or the pair of item-workstation to be used as input data for the lot sizing part.

5.3.2 Phase 2: Lot sizing problem with given Partial assignment matrix.

The result of the input data is the both set $W(i,t)$ and set $I(k,t)$ have only one member (Use the MLCLSP-M model for solving). Let set $A(t)=\{(i,k)\mid w_t^{i,k}=1\}$ is a set of the assignment item-workstation in period t.

$$Min \sum_{t\in T}\sum_{(i,k)\in A(t)}\left(p_t^{(i,k)}\cdot x_t^{(i,k)}+f_t^{(i,k)}\cdot y_t^{(i,k)}\right)+\sum_{t\in T}\sum_{i\in I}\left(h_t^i\cdot s_t^i\right) \qquad \text{(P2)}$$

Subject to

$$s_{t-1}^i+\sum_{(i,k)\in A(t)}x_t^{(i,k)}=d_t^i+s_t^i,\forall i\in E,\forall t\in T \qquad (5.7)$$

$$s_{t-1}^i+\sum_{(i,k)\in A(t)}x_t^{(i,k)}=\sum_{j\in S(i)}\left(u^{i,j}\cdot\sum_{(j,k)\in A(t)}x_t^{(j,k)}\right)+s_t^i,\ \forall i,j\in I-E,\forall t\in T \qquad (5.8)$$

$$\sum_{(i,k)\in A(t)}x_t^{(i,k)}\le Avail_t^k,\ \forall k\in WR,\forall t\in T \qquad (5.9)$$

$$\sum_{(i,k)\in A(t)}\left(o_t^{(i,k)}\cdot x_t^{(i,k)}+g_t^{(i,k)}\cdot y_t^{(i,k)}\right)\le cap_t^k,\ \forall k\in WP,\forall t\in T \qquad (5.10)$$

$$\sum_{i\in I}s_t^i\le V,\ \forall t\in T \qquad (5.11)$$

$$s_0^i=s_{NT}^i=0,\ \forall i\in I \qquad (5.12)$$

$$\sum_{(i,k)\in A(t)}y_t^{(i,k)}\le 1,\ \forall i\in I,\forall t\in T \qquad (5.13)$$

$$\sum_{(i,k)\in A(t)}y_t^{(i,k)}\le 1,\ \forall k\in WS,\forall t\in T \qquad (5.14)$$

$$x_t^{i,k}\le M_t^{i,k}\cdot y_t^{i,k},\ \forall t\in T,\forall(i,k)\in A(t) \qquad (5.15)$$

$$x_t^{(i,k)}\ge 0,\ \forall(i,k)\in A(t),\forall t\in T \qquad (5.16)$$

$$s_t^i\ge 0,\ \forall i\in I,\forall t\in T \qquad (5.17)$$

$$y_t^{(i,k)}\in\{0,1\},\ \forall(i,k)\in A(t),\forall t\in T \qquad (5.18)$$

In the first iteration, the assignment problem will be solved with echelon demand ($D_t^i = e_t^i$). The solution of this phase (the assignment matrix) will be given to the lot sizing problem and then we will get the initial solution.

5.3.3 Iteration

The two sub-problems will be done iteratively. Firstly, the problem with the Full assignment result from the first model is solved and kept as the first solution (F-solution). Secondly, the problem is solved with the Partial assignment matrix that changes the data of the first period to the iteration period (using the number of iteration as period number) to the same as the result of the assignment part and the solution will be kept as the P-solution. Thirdly, the F-solution and the P-solution will be compared. If they are equal or the first one is lesser or the lot size solution is equal, we stop and this is the solution for this problem. Otherwise, we continue to next iteration.

For example, if the planning horizon covers 3 periods, the maximum number of the iteration will be 3. The illustration of this example is shown in Figure 5.2.

Iteration 1

Assign. phase

Full assigned Matrix

T=1	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

T=2	Workstation		
Item	1	2	3
1	0	0	1
2	0	1	0
3	1	0	0

T=3	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

Relax Matrix (Partial assigned)

T=1	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

(original)

T=2	Workstation		
Item	1	2	3
1	1	1	1
2	1	1	1
3	1	1	1

(original)

T=3	Workstation		
Item	1	2	3
1	1	1	1
2	1	1	1
3	1	1	1

Lot size phase

Lot size solution (Given Lot size)

SOLUTION FROM ASSIGN MATRIX (F – SOLUTION) ≠ SOLUTION FROM RELAX MATRIX (R – SOLUTION)

Total Cost F-solution > Total Cost R-solution

Iteration 2

Assign. phase

Assign. Matrix (Given R-Solution)

T=1	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

Differ from Iteration 1

T=2	Workstation		
Item	1	2	3
1	1	0	0
2	0	0	1
3	0	1	0

T=3	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

Relax Matrix (Given Partial assign.)

T=1	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

T=2	Workstation		
Item	1	2	3
1	1	0	0
2	0	0	1
3	0	1	0

(original)

T=3	Workstation		
Item	1	2	3
1	1	1	1
2	1	1	1
3	1	1	1

Lot size phase

Lot size solution (Given Lot size)

SOLUTION FROM ASSIGN MATRIX (F – SOLUTION) ≠ SOLUTION FROM RELAX MATRIX (R – SOLUTION)

Total Cost F-solution > Total Cost R-solution

Iteration 3

Assign. phase

Assign. Matrix

T=1	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

Same as Iteration 2

T=2	Workstation		
Item	1	2	3
1	1	0	0
2	0	0	1
3	0	1	0

T=3	Workstation		
Item	1	2	3
1	0	1	0
2	1	0	0
3	0	0	1

Lot size phase

Lot size solution

END

Figure 5.2. The illustration of three-period problem

In the first iteration, a given assignment matrix that will be sent to lot sizing part will be split in 2 types which are the "Full assigned" (as the result from assignment model with given echelon demand lot size) and the "Partial assigned" (as the result from assignment model with given echelon demand lot size only in period 1, other periods (period 2 and 3) will be replaced with the original assignment of the problem). The Full assigned matrix will be sent and then solved in the lot size part and the solution will be kept as the first solution of the problem. On other hand, the Partial assigned will also sent and solved in the lot size part, the solution will be sent as given lot size in the next iteration. Likewise, the second iteration given lot size will be used the data from Partial assigned and the assignment matrix will be split in two types. The Partial in the second iteration will use the result from assignment model with given previous lot size only in period 1 and 2, other periods (period 3) will be replaced with the original assignment of the problem. This will go on and on until the number of iteration equals to max number of iteration. In this example will be end in the third iteration. In the third iteration, the assignment matrix will not be split and the solution will be the best solution.

With the concept of lot sizing problem, it is obvious that Lot-for-Lot policy gives only no holding solution and, especially in the case of setup consideration, the solution with more holding has a potential to be the better solution. On basis of this characteristic, in this paper propose the heuristic method that can change the Lot-for-Lot solution to the solution with consideration holding. The heuristic is called Partial Assignment- Lot size (PA-LS).

5.3.4 Termination

Understanding by above description, the termination of our heuristics occurs in two ways. First, all periods are considered or we get a solution by finding the same solution from the assigned result and the partial assigned result in any iteration. In these iterations, we use the AMPL/CPLEX 8.0.0 to solve each problem. Second, it's run until meet the user iteration limited.

5.4 An Example

Our goal in this section is to demonstrate how the heuristic developed in the earlier section may be implemented in practice in order to obtain a solution. For the sake of clarity we assume a ten-product ten-workstation five-period situation. An example can be seen in Figure 5.3. There are ten items (seven production items and three raw materials), and ten workstations (seven production workstations and three purchasing workstations). The capacity of each workstation is limited by the availability of its resources. For example, the capacity of production workstations are limited by capacity of available operating time of machines or workers and the capacity of a purchasing workstation are limited by available raw materials in the supply workstation. For the production workstation the capacity will be charged both from lot-size operation time and setup time. On the other hand, the purchasing operation will be charged by only the purchasing lot-size. By way of illustration let us look at the case of item 1; the item 1 can be produced only on workstation 1 which is a production workstation with limited capacity. Whenever item 1 is produced, the capacity of workstation 1 will be charged setup time and operation time (both of which varying by period) for the lot size of item 1. In this case, it has to be a trade-off between fixed cost and variable cost if we want to keep the total cost low. As only one workstation per one item, we aim to find a minimum lot size of particular item in a period over multi-period planning horizon consideration. Here, once a product is ordered, the setup time of the production will be occurred and will be considered as sequence independent between orders of different productions. The setup activities for each production workstation incur setup costs and consume setup time. Within a limited capacity expressed in hours during each time period of the workstation, the operation time and the setup time will reduce capacity in period. Assume that the demands for the part processing vary with time in a deterministic manner and there is no order for a component item from the customers. Therefore, production quantity of a component item can be computed from order quantities of end items that require the component item. If an item will not be planed to be used in the next period, it must be kept in warehouse and charged holding costs. Here, the quality of production is neglected. Moreover, all unit costs, prices, and setup times or costs are assumed to be known with dynamic by period. To address this multiple time period formation problem, a mixed integer programming (MIP) model is formulated. The objective of the model is to minimized total cost (setup cost, purchasing cost, production cost and inventory cost) for the entire planning time horizon with given time varying parameters (external demand, availability of raw materials, capacity of workstations, time parameters and cost parameters).

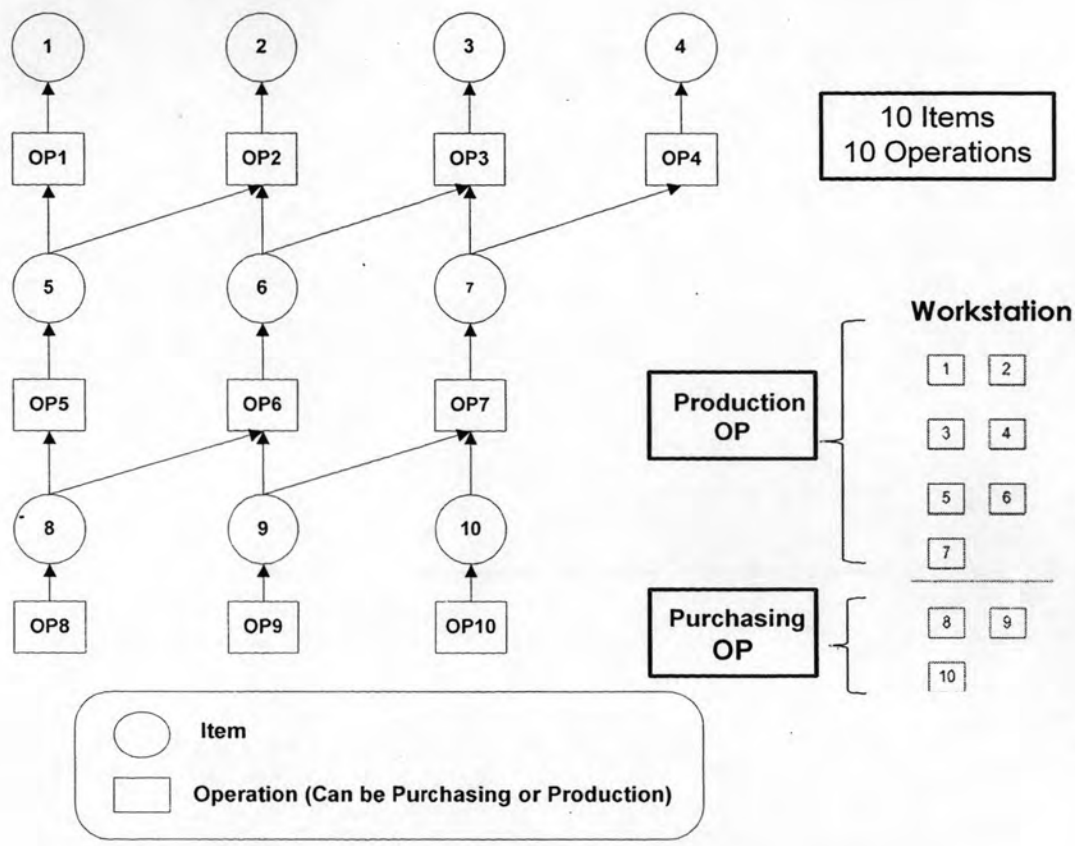

Figure 5.3 BOM for the example of heuristics implemented

The parameters are shown in Table 5.1 to Table 5.9. Item 1 to 7 can be produced on workstation 1 to 7 and item 8 to 10 can purchased on workstation 8 to 10.

Table 5.1 Primary demand, d_t^i.

i	T: 1	2	3	4	5
1	38	32	41	50	47
2	18	15	17	19	23
3	25	27	30	33	43
4	48	55	53	74	80
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0

Table 5.2 Usage parameters, $u^{i,j}$ (BOM in Figure 5.3).

i	J: 1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	1	1	0	0	0	0	0	0	0	0
6	0	1	1	0	0	0	0	0	0	0
7	0	0	1	1	0	0	0	0	0	0
8	0	0	0	0	1	1	0	0	0	0
9	0	0	0	0	0	1	1	0	0	0
10	0	0	0	0	0	0	1	0	0	0

Table 5.3 Ordering/Setup Cost parameters, $f_t^{i,k}$.

t = 1	K									
i	1	2	3	4	5	6	7	8	9	10
1	75	111	60	108	104	57	62	56	73	76
2	34	42	20	24	21	47	50	39	49	28
3	60	70	58	34	31	68	71	73	43	66
4	101	96	120	53	95	125	131	88	85	100
5	359	290	443	522	540	569	500	467	436	452
6	236	355	297	456	276	300	208	199	173	243
7	841	370	413	407	677	505	717	547	343	388
8	683	1244	928	1506	1083	1178	1034	1189	867	1340
9	1897	1665	2061	2466	2222	2248	1759	953	2392	1577
10	2849	1398	1843	1691	2794	3089	2262	1959	2460	1088

Table 5.4 Purchasing /Production Cost parameters, $p_t^{i,k}$.

t = 1	K									
i	1	2	3	4	5	6	7	8	9	10
1	9	19	7	16	18	7	7	8	12	12
2	12	17	6	6	14	12	13	10	11	6
3	19	11	16	9	5	16	14	7	16	6
4	5	9	12	16	18	17	11	13	5	8
5	18	16	11	13	5	10	17	13	18	18
6	13	17	10	6	8	14	16	12	7	9
7	6	16	11	5	11	14	11	6	17	8
8	34	59	48	34	30	54	53	50	46	54
9	43	38	46	49	57	57	37	59	57	47
10	49	44	57	36	58	36	36	35	36	47

Table 5.5 Setup Time parameters, $g_t^{i,k}$.

t = 1	K									
i	1	2	3	4	5	6	7	8	9	10
1	65	57	50	50	79	56	87	94	82	60
2	60	77	67	99	61	68	89	88	59	97
3	66	60	76	61	67	89	80	67	77	81
4	84	83	53	73	91	50	56	71	51	95
5	85	82	62	66	74	79	67	68	56	79
6	98	94	93	57	95	67	91	95	66	98
7	74	90	76	92	79	79	88	64	52	52
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

Table 5.6 Operation Time parameters, $o_t^{i,k}$.

t = 1	K									
i	1	2	3	4	5	6	7	8	9	10
1	16	11	16	18	17	11	17	10	14	14
2	17	15	10	10	10	12	15	15	12	10
3	12	12	14	13	17	11	15	13	16	14
4	19	17	11	18	14	14	18	17	19	12
5	14	19	19	10	11	14	13	14	11	16
6	19	16	12	15	19	17	19	12	10	18
7	13	15	18	12	19	12	12	13	18	18
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

Table 5.7 Holding Cost parameters, h_t^i.

	T				
i	1	2	3	4	5
1	2	3	1	2	1
2	3	1	2	2	2
3	1	1	3	1	3
4	2	2	2	1	2
5	1	3	1	1	2
6	3	2	1	3	2
7	1	2	1	3	1
8	3	3	1	1	1
9	3	2	3	1	3
10	3	2	3	3	2

Table 5.8 Capacity parameters, cap_t^k.

	T				
k	1	2	3	4	5
1	6673	5890	7739	9660	9519
2	5577	6381	6553	9954	9009
3	6059	6006	6596	8107	8697
4	6871	6591	6222	9527	9597
5	5771	5878	6349	8743	10203
6	7542	6030	6524	7029	9688
7	6217	5764	7081	8766	8901
8	-	-	-	-	-
9	-	-	-	-	-
10	-	-	-	-	-

Table 5.9 Availability parameters, $Avail_t^k$.

	T				
k	1	2	3	4	5
1	-	-	-	-	-
2	-	-	-	-	-
3	-	-	-	-	-
4	-	-	-	-	-
5	-	-	-	-	-
6	-	-	-	-	-
7	-	-	-	-	-
8	542	520	586	698	790
9	542	520	586	698	790
10	542	520	586	698	790

The firm's manufacturing structure has the four products (item 1, item2, item 3 and item 4) and three raw material items (item 8, item 9, and item 10) according to the BOM as shown in Figure 5.3. There is zero initial stock for all items and the warehouse can keep only 542 units. The quantities of the items required to make a unit of another item are assumed to be 1 unit for all items. Table 5.2 shows the quantities of the required items or the usage parameters of all items. Table 5.3 and Table 5.4 show the ordering or setup cost and the purchasing/production cost parameters, respectively. In Table 5.3, the cost in column workstation 8, 9 and 10 are ordering costs, while the costs in column workstation 1 to workstation 7 are setup cost. Similarly, in Table 5.4, the cost in column workstation 8, 9 and 10 are purchasing costs, while the costs in column workstation 1 to workstation 7 are production costs. Table 5.5 and Table 5.6 show the setup time and operation time parameters, respectively. Only production workstations, which are workstation 1 to workstation 7, have the value of setup time and operation time. Table 5.7 shows the holding cost per period for each item. In addition, there are ten workstations in this problem and each workstation has capacity per period of time as shown in Table 5.8 and availability per period of time as shown in Table 5.9.

In summary, the problem considers a ten-component product structure of four end-item constrained by ten workstations over a 5-period planning horizon with the parameters and data presented in Table 5.1 through Table 5.9. The heuristics can be presented as follows:

Iteration 1

Assignment Phase

1. Using the data from Table 5.1 through Table 5.9
2. Calculating the echelon demand.
3. Solving with assignment model. The result will be set as assignment for Lot-size Part

Lot-size Phase

1. Solving the single model with assignment from the assignment part. The total cost is 87109.

2. Solving the single model with assignment from the first iteration only for period 1 and unchanging for others. The total cost is 82854.

3. Comparing the result from 1 and 2. We found that the solutions are unequal. Go to Iteration 2.

Iteration 2

Assignment Phase

1. Using the data from Table 5.1 through Table 5.9 for parameters.

2. Using the Lot-size result from the Lot Size Part of iteration 1 instead of echelon demand.

3. Solving with assignment model.

Lot-size Phase

1. Changing the demand to the result lot size from iteration 1 and solving the single model with assignment from the assignment part. The total cost is 81738.

2. Solving the single model with assignment from the first iteration only for period 1 through 2 and unchanging for others. The total cost is 81617.

3. Comparing the result from 1 and 2. We found that the solutions are unequal. Go to Iteration 3

Iteration 3

Assignment Phase

1. Using the data from Table 5.1 through Table 5.9 for parameters.
2. Using the result of the Lot-size Part of iteration 2.
3. Solving with assignment model, the result can be presented in Table 5.10.

Lot-size Phase

1. Changing the demand to the result lot size from iteration 2 and solving the single model with assignment from the assignment part. The result can be presented in Table 5.11 and the total cost is 80187 (F-solution).
2. Solving the single model with assignment from the first iteration only for period 1 through 2 and unchanging for others. The result can be presented in Table 5.12 and the total cost is 80081 (P-solution).
3. Comparing the result from 1 and 2. We found that the solutions are unequal but the matrixes are same.
4. Then, we stop. The result can be presented in Table 5.13 through 5.14 and the total cost is 83273.

Table 5.10 The assignment form iteration 3, $w_t^{i,k}$

t = 1					k					
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	1	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	1	0	0

t = 2					k					
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	1	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	1	0	0	0	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	0	0	0

t = 3					k					
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	1	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	1	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	1

t = 4					k					
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	1	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

t = 5					k					
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

Table 5.11 F-solution of Iteration 3

	T				
i	1	2	3	4	5
1	38	32	138	0	0
2	18	15	17	42	0
3	25	27	63	0	43
4	103	0	53	154	0
5	56	244	0	0	0
6	43	42	165	0	0
7	155	0	313	0	0
8	56	244	0	0	0
9	99	451	0	0	0
10	240	0	478	0	0

Table 5.12 P-solution of Iteration 3

	T				
i	1	2	3	4	5
1	38	32	138	0	0
2	18	15	17	42	0
3	25	27	63	0	43
4	103	0	53	154	0
5	56	244	0	0	0
6	43	42	165	0	0
7	155	0	313	0	0
8	56	244	0	0	0
9	99	451	0	0	0
10	240	0	478	0	0

Table 5.13 The optimal solution of lot size, $x_t^{i,k}$

t = 1	*k*									
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	38	0	0	0
2	0	0	0	0	0	18	0	0	0	0
3	0	25	0	0	0	0	0	0	0	0
4	103	0	0	0	0	0	0	0	0	0
5	0	0	0	0	56	0	0	0	0	0
6	0	0	43	0	0	0	0	0	0	0
7	0	0	0	155	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	56	0
9	0	0	0	0	0	0	0	0	0	99
10	0	0	0	0	0	0	0	240	0	0

t = 2	*k*									
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	32	0	0	0	0	0
2	0	0	15	0	0	0	0	0	0	0
3	0	0	0	0	0	27	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	244	0	0	0	0	0	0	0	0
6	0	0	0	42	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	244	0
9	0	0	0	0	0	0	0	0	0	451
10	0	0	0	0	0	0	0	0	0	0

t = 3	*k*									
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	138	0	0	0
2	0	0	0	0	0	17	0	0	0	0
3	0	0	0	63	0	0	0	0	0	0
4	0	0	0	0	53	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	165	0	0	0	0	0	0	0	0
7	313	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	478

t = 4	*K*									
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	1	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

t = 5	*K*									
i	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	42	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	154	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

Table 5.14 The optimal solution of stock, s_t^i

	T				
i	1	2	3	4	5
1	0	0	97	47	0
2	0	0	0	23	0
3	0	0	33	0	0
4	55	0	0	80	0
5	0	197	42	0	0
6	0	0	85	43	0
7	27	0	197	43	0
8	0	0	0	0	0
9	0	165	0	0	0
10	42	0	0	0	0

The solution from the research algorithm is equal to the optimal solution from AMPL/CPLEX 8.0.0. For more clarification of the solution approach, the relation between the number of iterations and objective function as illustrated in Figure 5.4.

	Full Assignment	Partial Assignment
1	87,109	82,854
2	81,738	81,617
3	80,187	80,081

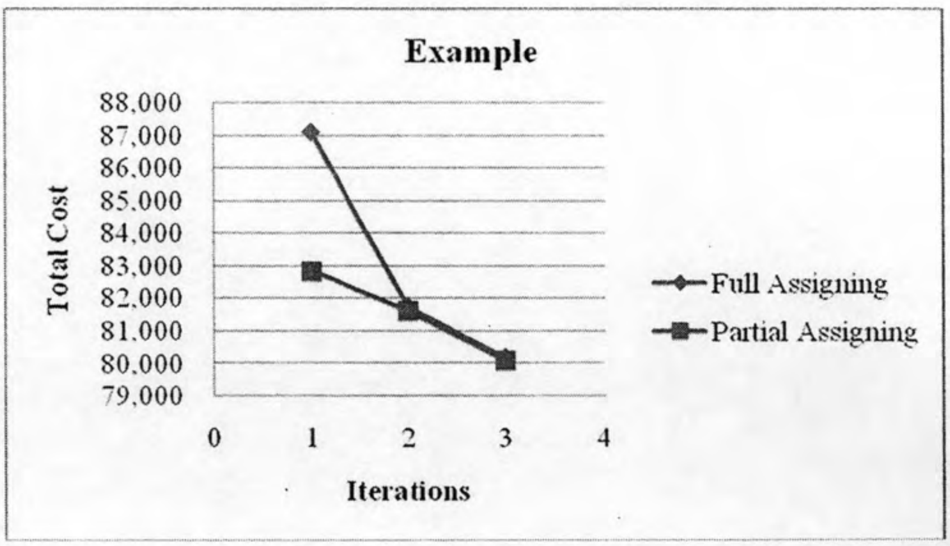

Figure 5.4 The relation between the number of iterations and objective function

It shows that the objective will go to the same point. It's obvious that the solution is very close to the optimal solution. However, the time for solving is very large.

5.5 Conclusion

This research proposes a new decomposition heuristic based on all assumptions of the MLCLSP-M model including that lot-for-lot policy is a feasible solution. The heuristics composes of two phases which are partial assignment with given lot size phase and partial lot size with given partial assignment phase. These phases are solved iteratively and sequentially with AMPL/CPLEX 8.0.0 solver. The assignment solution of the first phase is used as input data for the lot size phase and the lot size solution will be used as input for the assignment phase in the next iteration. The termination of this heuristic will happen whenever the iteration numbers is the maximum iteration limited or the lot solution of the previous iteration equal to this iteration.

As seen in the example, the drawbacks of this heuristic is the Partial steps make the problem reduce the problem size only in period dimensions (The Problem is large in the early iteration). Although the solution is great, it needs long solving time in the early iteration. It's clear that there is a need to find the way to reduce to solving time. Therefore, the heuristic that has lesser solving time and satisfactory solution should be developed in further research.