IE312: Facility Design & Planning

Project - Part 1

Fall 2022 - 28.11.2022

Layout Design for a Hypothetical Flexible Manufacturing System

Tevfik Buğra Türker 2019402120

Hüseyin Emre Bacak 2021402279

Ömercan Mısırlıoğlu 2020402261

CONTENTS

Abstract

The objective of this research is to create the design of a hypothetical flexible manufacturing system (FMS) using the problem description and production data. Heuristic algorithms and improvement methods are used to reach the local optimal solution.

1. Introduction

The system receives and ships the manufactured parts from the receiving and shipping areas, which are two of the 9 locations in the facility plan. The equipment, a few manufacturing zones, and a buffer zone will all be housed at the final 7 locations. The production system is made up of 6 appliances, each of which has a specific function and a different processing time. 5 different part types are intended to be produced by the facility, and each part has a unique operating process that is carried out by a variety of equipment, which can be seen in **Table 2.**. Both the distances between the pick-up and departure zones of the location and the quantity of components that enter the system are known and can be seen in **Table 1. and Figure 1.**. The transportation part will be carried out by the assistance of autonomous guided vehicles.

The goal of the project is to place the equipment in the allotted spaces in a way that minimizes the cost of manufacturing the components. Traffic conditions and transit times have an impact on the whole cost. Throughout the design phase, one constructive and one improvement algorithm are used.

The constructive algorithm assigns values to the classes, classifies the relationships between the machines, such as the volume of part flow, and determines how the machines relate to one another. The more flows there are between two machines, the higher the likelihood that the link will be valued. The Total Closeness Rating (TCR) is then determined for each piece of equipment. Machines are arranged in the right sequence and priority locations in accordance with predefined rules.

The improvement algorithm aims to have a lower overall cost by switching the places of the equipment. According to the resulting costs of 2-way changes, the lowest cost is aimed to be found.

Table 1. Part types and the production rates

Part Type	Process Plan				Arrival rate (units per hour)
A	op1	op2	op3		6
B	op4	op2	op5	op6	5
C	op7	op8	op9	op10	5
D	op11	op12	op13		2
E	op14	op8	op15		2

Figure 1. Candidate machine locations together with P and D stations and the AGV flow path

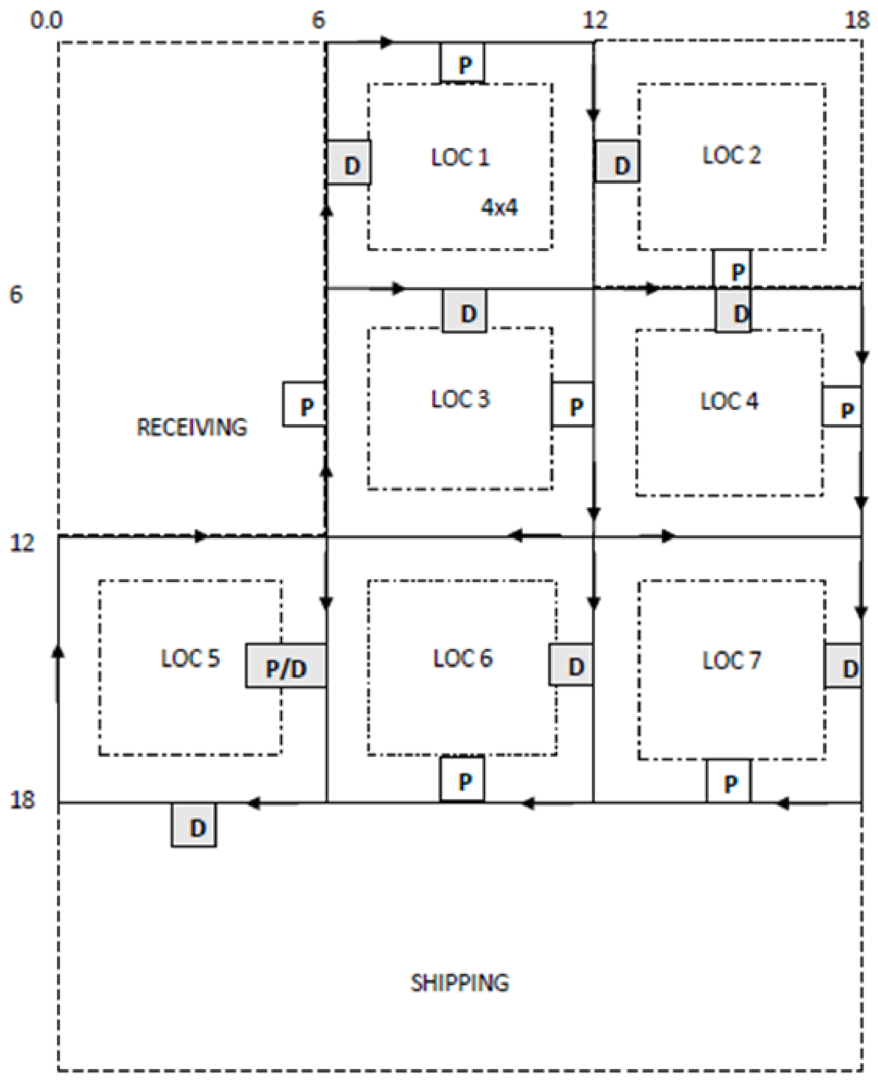


Table 2. Machine allocations and processing times

Operation	Machine center	Processing time (sec.)
op1	VTC2	240
op2	SHP	120
op3	VMC	420
op4	VTC1	260
op5	VTC1	120
op6	HMC	300
op7	VTC2	220
op8	SHP	90
op9	UMC	480
op10	VTC2	120
op11	VTC1	440
op12	VMC	300
op13	HMC	360
op14	VTC2	150
op15	HMC	380

2. Alternative Layout I - Constructive Algorithm

2.1. Introduction

The algorithm to be used is a constructive and heuristic algorithm for designing a hypothetical flexible manufacturing system (FMS) layout. Main objective of this algorithm is to define relationships of different machine centers using qualitative closeness ratings and construct a layout design with respect to these ratings.

The algorithm consists of:

A) Initialization

B) Determination of the placement sequence of machine centers

C) Determination of relative locations of machine centers

2.2. Initialization of activity relationships

The relationships of machine centers should be determined before constructing a placement sequence of departments. There are 4 types of relationships between machine centers (A, E, I, O) and they are assigned as appeared in **Table 3.** and **Table 4.**

Table 3. Relationships of machine centers with each other

Relation	Description of relation	Flow between machine centers
A	Absolutely Necessary	9 <= flow
E	Especially Important	6 <= flow < 9
I	Important	5 <= flow < 6
O	Ordinary	flow < 5

Table 4. Relationships of machine centers with central buffer

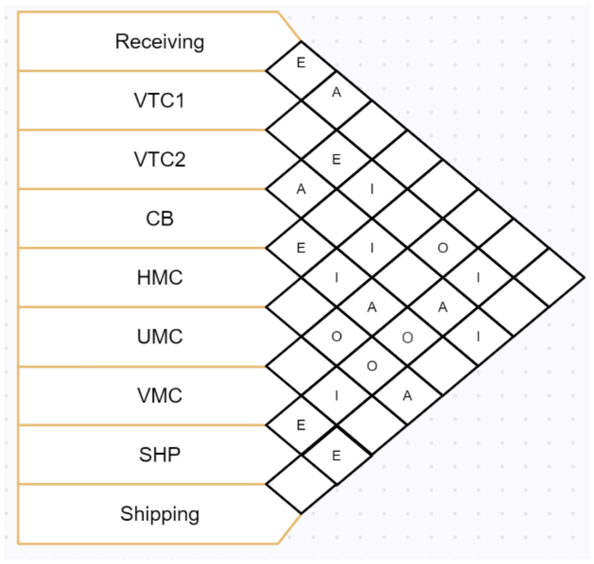
Active minutes per hour for a certain machine type	CB relation with certain machine type
processing mins/hour >= 51	A
42 <= processing mins/hour < 51	E
33 <= processing mins/hour < 42	I
processing mins/hour < 33	O

As seen in **Table 3.**, relationships between each machine center is determined by calculating flow rates between them. Since there are no predetermined flow rates between any machine center and central buffer, a different approach is used.

Active minutes per hour of each machine type is calculated and the relation between each certain machine type and central buffer is assigned based on predetermined arbitrary intervals and calculations which can be seen from **Table 4.**

After assigning all relationships, a REL chart is constructed. (**Figure 2.**)

Figure 2. REL chart



Subjectively chosen, numerical values are assigned to closeness ratings as

V(A) = 10000

V(E) = 1000

V(I) = 100

V(O) = 10

then using these numerical values, total closeness ratings (TCR) of each machine center is calculated. The TCR of each machine center is used in the next step of the constructive algorithm.

2.3. Determination of the placement sequence of machine centers

In this step of the algorithm, the purpose is to obtain a placement sequence (π) of machine centers. To determine the sequence(π), calculated total closeness ratings (TCR) of machine centers are used. (**Table 5.**)

Table 5. Total closeness ratings of each department

Department	A	E	I	O	TCR
Receiving	1	1	0	0	11000
VTC1	0	2	2	1	2210
VTC2	3	0	2	0	30200
CB	2	2	1	1	22110
HMC	1	1	1	2	11120
UMC	0	0	3	0	300
VMC	1	2	0	2	12020
SHP	1	1	3	2	11320
Shipping	1	1	1	0	11100

In this specific layout, locations of “Receiving” and “Shipping” are assumed to be fixed, which means that TCR and placement of these departments are not going to be considered. After this assumption, other TCR values in **Table 5.** is used for determination of sequence:

π = {VTC2,CB,VMC,SHP,HMC,VTC1,UMC}

2.4. Determination of relative locations of machine centers

In this step of the algorithm, the purpose is to place machine centers and central buffer with respect to the locations of "receiving" and "shipping" departments and the placement sequence (π) obtained in the previous step. "Weighted Placement Value" (WPV) method is used to place each machine center and central buffer in the following manner:

i. If a machine center or central buffer is placed "fully adjacent" with a related department (which means having a common border), it receives all the benefits of that relation.

ii. If a machine center or central buffer is placed "partially adjacent" with a related department (which means having only a common point), it receives ½ of the benefits of that relation.

After the placement of a machine center or central buffer, a new one is chosen from the placement sequence π and WPV method is used again to determine the location until all elements in placement sequence π are placed.

The final layout design is given below. (**Figure 3.**)

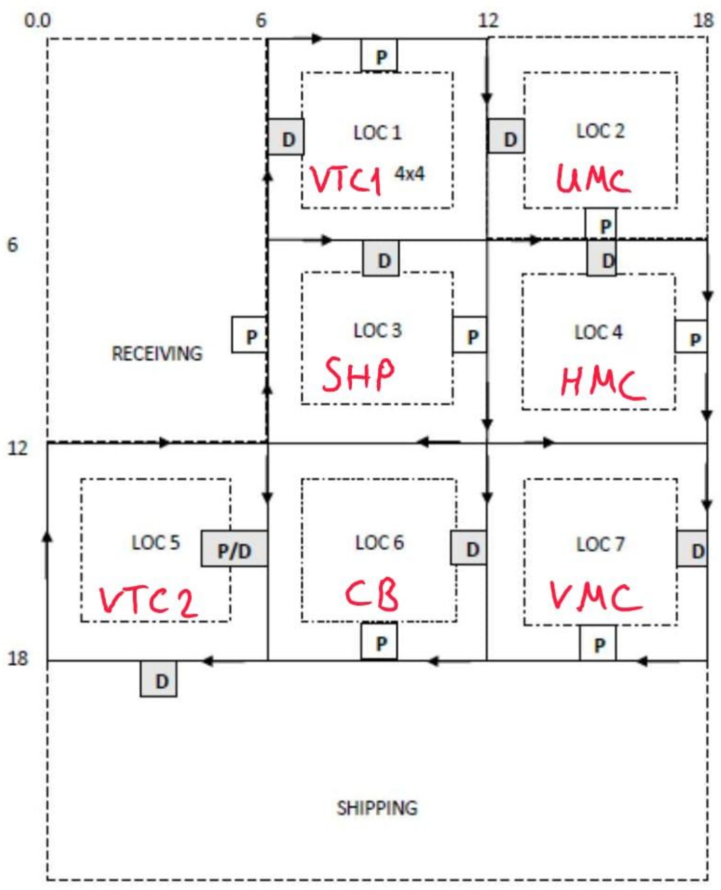


Figure 3. Final layout design for constructive algorithm

2.5. Cost determination of Constructive Algorithm

Cost matrix of the constructive algorithm is determined by multiplication of "From-to matrix" and "Distance matrix" in an element-wise manner. In order to achieve this, "From-to matrix" is reordered by consideration of final layout design.

In order to find a convenient spot for the Central Buffer, its "from-to" values with other machines should be indicated. To represent these values, first, each machine's hourly workload is calculated. Then the results are converted into percentage base. After that, for every machine, **exp**(X*5)/15 is evaluated and rounded. The reason that exponential relation is used is that it represents the relation between machine load and flow units better than linear approximation. Eventually, these are the values that will be used in "From-to matrix" to calculate cost matrix. (**Table 6.**)

Table 6. Table for estimating flow from the central buffer

Machine Types	Needed Time / hour	%	Result	Rounded
VTC1	46,33	0,772	3,167	**3**
VTC2	57,33	0,956	7,920	**8**
HMC	49,67	0,828	4,183	**4**
UMC	40	0,667	1,869	**2**
VMC	52	0,867	5,080	**5**
SHP	32,5	0,542	1,000	**1**

The final result is given below. The total cost is obtained by summing every cell. The result may not be cost-wise optimal and can be improved using an improvement algorithm. (**Table 7.**)

Table 7. Cost matrix for the constructive algorithm

Cost Matrix	VTC1	UMC	SHP	HMC	VTC2	CB	VMC	Shipping	
Receiving	42	0	0	0	312	0	0	0	354
VTC1	0	0	150	60	0	54	48	0	312
UMC	0	0	0	0	240	132	0	0	372
SHP	90	150	0	48	0	6	72	0	366
HMC	0	0	0	0	0	240	0	216	456
VTC2	0	0	390	0	0	336	0	30	756
CB	90	84	30	144	192	0	240	0	780
VMC	0	0	0	84	0	240	0	72	396
								COST =	3792

3. Alternative Layout II - Improvement Algorithm

3.1. Methodology and Principles

As an improvement algorithm, a similar method to CRAFT is used. The main working principle of the algorithm is as follows:

1) First, a base condition is selected. As a base condition, the result of the constructive algorithm can be used.
2) Then, using the base condition, all possible two-way changes are considered. In this specific layout design, there are C(7,2)=21 possible changes and a base condition.
3) All the cost matrices of 22 distinct layouts are calculated.
4) If a cost lower than the base condition among 21 possible changes can be found, the lowest cost is selected and change is implemented.
5) The lowest cost becomes the new base condition and this iteration continues until no cost lower than the base is found.

3.2 Application of the Algorithm

3.2.1. The First Iteration

As the base condition, the result obtained from the constructive algorithm is used. In **Table 8.** every respective change and resulting cost are recorded.

Table 8. Changes and resulting costs for the first iteration

1	3792 no change	12	3744 (2, 7)
2	3852 (1, 2)	13	3936 (3, 4)
3	3732 (1, 3)	14	3534 (3, 5)
4	4080 (1, 4)	15	3672 (3, 6)
5	3882 (1, 5)	16	4128 (3, 7)
6	3480 (1, 6)	17	3690 (4, 5)
7	4080 (1, 7)	18	**3468 (4, 6)**
8	3828 (2, 3)	19	3768 (4, 7)
9	3804 (2, 4)	20	3630 (5, 6)
10	4278 (2, 5)	21	3762 (5, 7)
11	3504 (2, 6)	22	3552 (6, 7)

According to **Table 8.**, among all possible changes, relocating stations in LOC4 and LOC6 caused the lowest cost. For this reason, the new base condition includes that change.

The old base condition: [VTC1, UMC, SHP, **HMC**, VTC2, **CB**, VMC]

The new base condition: [VTC1, UMC, SHP, **CB**, VTC2, **HMC**, VMC]

3.2.2. Consecutive Iterations

In the manner displayed above, consecutive iterations are conducted one by one. In **Table 9.** the lowest costs obtained and the corresponding changes are exhibited.

Table 9. Costs and corresponding changes for iterations

The Second Iteration	**The Third Iteration**	**The Fourth Iteration**	**The Fifth Iteration**
3258 (5, 7)	**3174 (1, 3)**	**3102 (4, 7)**	**2646 (1, 7)**

3.2.3. The Last Iteration

For the last iteration, when all possible changes are investigated, no cost lower than the base condition can be found (**Table 10.**).

Table 10. Changes and resulting costs for the sixth iteration

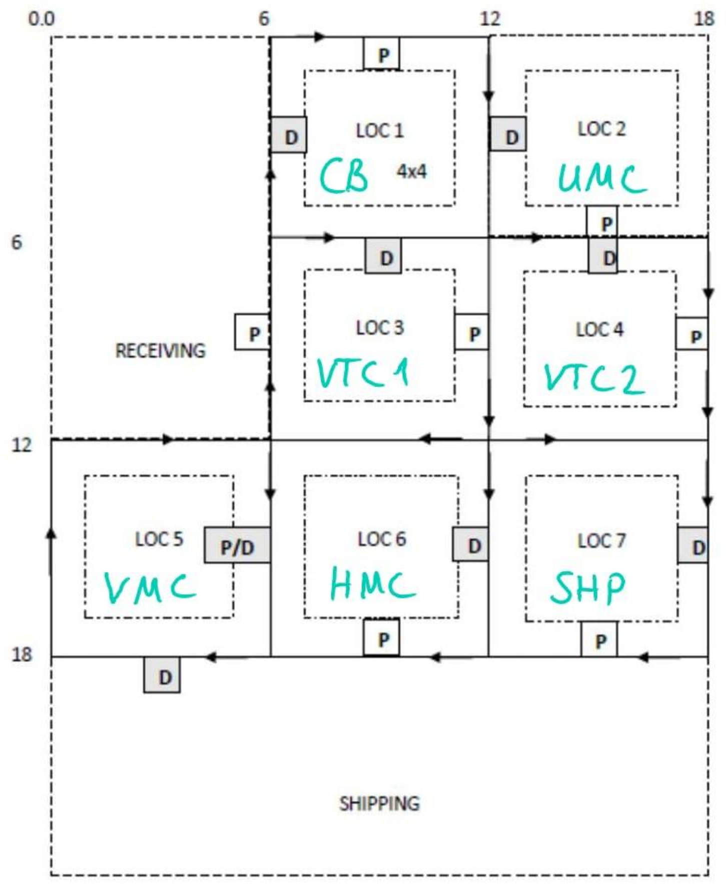
1	**2646 no change**	12	3702 (2, 7)
2	3174 (1, 2)	13	3198 (3, 4)
3	2814 (1, 3)	14	3162 (3, 5)
4	3474 (1, 4)	15	3174 (3, 6)
5	3066 (1, 5)	16	3306 (3, 7)
6	3318 (1, 6)	17	3720 (4, 5)
7	3102 (1, 7)	18	3690 (4, 6)
8	3234 (2, 3)	19	3546 (4, 7)
9	3102 (2, 4)	20	2694 (5, 6)
10	3090 (2, 5)	21	3222 (5, 7)
11	3186 (2, 6)	22	3354 (6, 7)

Since no cost lower than the base can be found, iterations stop and current base condition is concluded to be the local minimum value. The final cost matrix is in **Table 11.**

Table 11. Final cost matrix for the improvement algorithm

Cost Matrix	CB	UMC	VTC1	VTC2	VMC	HMC	SHP	Shipping	
Receiving	0	0	42	156	0	0	0	0	198
LOC1 - CB	0	12	90	96	120	72	24	0	414
LOC2 - UMC	108	0	0	0	0	0	0	0	108
LOC3 - VTC1	54	0	0	0	24	30	60	0	168
LOC4 - VTC2	384	0	0	0	0	0	78	120	582
LOC5 - VMC	150	0	0	0	0	84	0	36	270
LOC6 - HMC	120	0	0	0	0	0	0	54	174
LOC7 - SHP	36	240	180	0	180	96	0	0	732
								COST =	2646

Figure 4. Final layout design for improvement algorithm



4. Comparison and Summary

The objective of this research is to use specific algorithms and techniques to create a workable and affordable facility layout plan. The from-to and distance matrices are obtained in the initial step of the investigation. To begin the constructive method, the significant links between any two stations are then estimated. A from-to-chart is used to calculate the flow rate, and if it is comparatively higher, the link between them received a higher grade for closeness. Because the flow of the central buffer is not specified, hourly production times for each machine are computed. Their production times and an exponential heuristic formula is used to specify the strength of their relationship with the central buffer. Then, the flow amounts are assigned in light of these production times. After first weighing their weighted placement value in respect to the already decided receiving and shipment, stations are put on the layout.

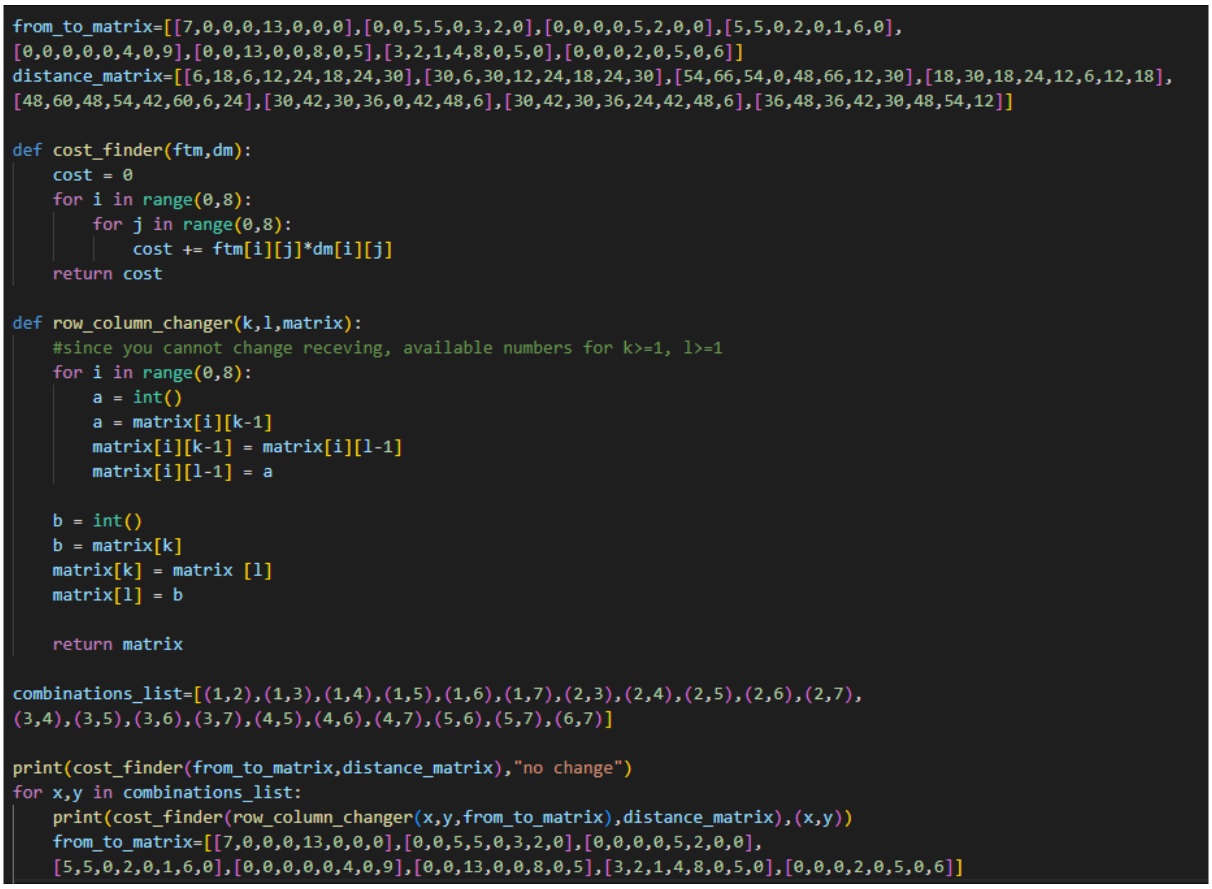
The result obtained from the constructive algorithm provided an initial but not the minimum cost solution. To decrease the cost further down, the improvement algorithm is used. The improvement algorithm's main objective is to lower the cost by interchanging the places of 2 machines. By examining the adjustments, starting with 7 locations and using steepest descent manner, the algorithm resulted in a minimum cost of 2646, which is better than the constructive algorithm’s total cost. Since the layout plan obtained by the improvement algorithm is cost-wise better, the usage of that layout is preferred and recommended.

It is worthwhile to note that the resulting design may not be the optimal choice since the improvement algorithm uses the steepest descent algorithm and there is always a chance that the current cost represents a local optimum rather than a global optimum. Besides, throughout the stages of both algorithms, heuristic methods are widely utilized. Consequently, these are the few of the factors that give rise to the possibility of current solutions being not globally optimal and allow for a better design.

5. Appendix

The procedure of swapping locations of machine centers is implemented in Python, which can be seen below. (**Figure 5.**)

Figure 5. Python code written for swapping locations of machine centers



```python
from_to_matrix=[[7,0,0,0,13,0,0,0],[0,0,5,5,0,3,2,0],[0,0,0,0,5,2,0,0],[5,5,0,2,0,1,6,0],
[0,0,0,0,0,4,0,9],[0,0,13,0,0,8,0,5],[3,2,1,4,8,0,5,0],[0,0,0,2,0,5,0,6]]
distance_matrix=[[6,18,6,12,24,18,24,30],[30,6,30,12,24,18,24,30],[54,66,54,0,48,66,12,30],[18,30,18,24,12,6,12,18],
[48,60,48,54,42,60,6,24],[30,42,30,36,0,42,48,6],[30,42,30,36,24,42,48,6],[36,48,36,42,30,48,54,12]]

def cost_finder(ftm,dm):
    cost = 0
    for i in range(0,8):
        for j in range(0,8):
            cost += ftm[i][j]*dm[i][j]
    return cost

def row_column_changer(k,l,matrix):
    #since you cannot change receving, available numbers for k>=1, l>=1
    for i in range(0,8):
        a = int()
        a = matrix[i][k-1]
        matrix[i][k-1] = matrix[i][l-1]
        matrix[i][l-1] = a

    b = int()
    b = matrix[k]
    matrix[k] = matrix [l]
    matrix[l] = b

    return matrix

combinations_list=[(1,2),(1,3),(1,4),(1,5),(1,6),(1,7),(2,3),(2,4),(2,5),(2,6),(2,7),
(3,4),(3,5),(3,6),(3,7),(4,5),(4,6),(4,7),(5,6),(5,7),(6,7)]

print(cost_finder(from_to_matrix,distance_matrix),"no change")
for x,y in combinations_list:
    print(cost_finder(row_column_changer(x,y,from_to_matrix),distance_matrix),(x,y))
    from_to_matrix=[[7,0,0,0,13,0,0,0],[0,0,5,5,0,3,2,0],[0,0,0,0,5,2,0,0],
    [5,5,0,2,0,1,6,0],[0,0,0,0,0,4,0,9],[0,0,13,0,0,8,0,5],[3,2,1,4,8,0,5,0],[0,0,0,2,0,5,0,6]]
```