An Algorithm for Optimal Solutions of Nugent Problems: -Cost Minimization-

Assoc. Prof. Dr. Mehmet ÇINAR*,
Ömer TAHTASAKAL**

ABSTRACT

This paper deals with the minimization of an objective function which related to the minimization of the total material handling cost in the facilities layout problems. The proposed method, **"Cost Minimization"**, and its algorithm, based on Mathematica coding, are introduced in detail and compared to the solutions of the other well-known methods. The results show that the algorithm is efficient, easy to run and very competitive with the other well-known methods.

* Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Öğretim Üyesi.
** Apple Center Mega Genel Müdürü.

1. Introduction

The plant layout problem has long been of interest to industrial engineers, production systems analysts and management scientists. The assignment of facilities to locations has been of particular concern to those working in the field. Being a combinatorial problem, it has been difficult to obtain optimum solution to large or even reasonable-sized problems. Several optimizing and sub-optimizing solutions have been proposed with the optimizing ones which feasible only for small-sized problems [Gavett and Plyter, 1966: Gilmore, 1962: and Lawler 1963]. The computer-based quantitative methods, like CRAFT by Buffa *et al.* [1964], H-63 by Hillier [1963], H-66 by Hillier and Connors [1966], COL by Vollmann *et al.* [1968], MAT by Edwards *et al.* [1970], LSP by Zoller and Adendorff [1972], FRAT by [Khalil 1973] and COFAD by Tompkins [1980] are developed to minimize the total materials handling or communication costs. On the other hand, the computer-based qualitative methods like CORELAP by Lee and Moore [1967] and ALDEP by Seehof and Evans [1967] are improved to maximize the values for desirability of facility units to be near each other, namely closeness ratings.

In this paper, a new heuristic algoritm which considers only the cost aspect of the problem will be introduced.

2. Formulation of Problem

The problem has been formulated by most of the researchers with the identical assumptions and measures of effectiveness in general. Cost as a linear function of distance (such as materials handling cost) has been the dominant criterion to be minimized in almost all the methods proposed so far.

The assumptions made in the proposed method are that the departments are of equal area, the distance between adjacent departments are one unit each, the cost is proportional to the material handling and linear in nature .

The problem can now be stated that of assigning facilities (n=1,2...) to n locations so as to minimize the cost function as;

$$C_{min} = \text{Min} \left\{ \sum_{i=1}^{n} \sum_{j=i+1}^{n} f_{ij}^{s} \, d_{kl}^{s} \right\} \quad \ldots . (1)$$

where f_{ij}= the total weighted flow of material between the i_{th} and j*th* facility; d_{kl}= the distance between k*th* and l*th* locations.

A detailed description of the algorithm is as follows.

1. Input information related to number of the fixed departments, the cost weighted flow matrix, and initial coordinates of the facilities.

2. Compute (A), the difference of the longest and shortest distance between the centres of the facilities.This distance depends on the travel pattern in the plant.

3. For each facility (I) calculate Pi (i=1,2,...,n) the total cost of all possible trips of lenght greater than or equal to A.

4. Find the facility M that yields the highest value of all Pi's computed. This represents the desirability order of moving the facilities.

$$P_i = \sum_{k \in I} f_{ik} \, d_{ik}$$

I' sub-set with distance >= A.

5. Compute the cost of switching the position of each I facility with the M*th* facility and switch the one that yields maximum savings in total cost.

6. Repeat step number 5 until no more saving is possible.

7. Reduce (A) by an amount (L) equal to the shortest distance between the centres of two facilities. (A-L=A').

8. Repeat steps 3 to 7 until A' is less than L.

9. Compute the effect of pairwise switches between facilities I and J and execute the switching if profitable. This is repeated until I=n-1 and J=n where n is the number of facilities.

10. Step number 9 is repeated until no more improvement could be obtained.

3. Testing and Results

In order to test the efficiency of the algorithm, a Mathematica programme has been developed for the model.

The **CostMinimization** algorithm is tested utilizing the same test problems used in Nugent *et al.* [1968] experimental comparison paper. This allows a fair comparison between the results of **CostMinimization** in and several of the other methods. The starting solutions are given in Nugent *et al.* [1968]. The results of MAT and MAT augmented with CRAFT to the test problems utilized are given by Edvards *et al.* [1970]. The results of FRAT are given in Khalil [1973].

The **Cost Minimization** was written in Fortran, Basic,and Mathematica, in order to solve problems involving equal-area facilities.Therefore, only this type of problem is considered in this paper.

The results of all the principal procedures and of **CostMinimization** comparing the final cost yielded by the solutions are summarized in table 1. Some of the interesting final layouts yielded minimum cost are also given in the Appendix.

4. An Example

In order to indicate the efficiency of the proposed algorithm, it is compared with the MUGHAL algorithm , using the test problem (8X8) presented by [Dutta and Sahu 1982]. The minimum cost value of 194 is found in problem (8X8) from 15 combinations. The cost and initial coordinates of the problem as follow.

```
CostM=
    {{0,6,1,1,8,2,4,4},
    {0,0,1,2,3,3,6,2},
    {0,0,0,5,2,3,1,10},
    {0,0,0,0,2,8,3,3},
    {0,0,0,0,0,4,10,10},
    {0,0,0,0,0,0,8,8},
    {0,0,0,0,0,0,0,2},
    {0,0,0,0,0,0,0,0}};
```

```
CoordM=
{{5,1,2},{4,2,2},{6,3,2},{8,4,2},
{1,1,1},{7,2,1},{2,3,1},{3,4,1}};
```

5	4	6	8
1	7	2	3

INITIAL COST = 226
Swapping the departments 5 <--> 2
TOTAL COST = 212
Swapping the departments 5 <--> 4
TOTAL COST = 204
Swapping the departments 5 <--> 7
TOTAL COST = 201
Swapping the departments 3 <--> 8
TOTAL COST = 199
Swapping the departments 4 <--> 8
TOTAL COST = 194
Swapping the departments 3 <--> 4
TOTAL COST = 179
Optimum layout =
1 2 3 4 5 6 7 8
1 1 4 4 2 3 2 3
1 2 1 2 1 2 2 1
TOTAL COST = 179

226, {{5, 1, 2}, {4, 2, 2}, {6, 3, 2}, {8, 4, 2},
{1, 1, 1}, {7, 2, 1}, {2, 3, 1}, {3, 4, 1}}
212, {{1, 1, 1}, {2, 1, 2}, {3, 4, 1}, {4, 2, 2},
{5, 3, 1}, {6, 3, 2}, {7, 2, 1}, {8, 4, 2}}
204, {{1, 1, 1}, {2, 1, 2}, {3, 4, 1}, {4, 3, 1},
{5, 2, 2}, {6, 3, 2}, {7, 2, 1}, {8, 4, 2}}
201, {{1, 1, 1}, {2, 1, 2}, {3, 4, 1}, {4, 3, 1},
{5, 2, 1}, {6, 3, 2}, {7, 2, 2}, {8, 4, 2}}
199, {{1, 1, 1}, {2, 1, 2}, {3, 4, 2}, {4, 3, 1},
{5, 2, 1}, {6, 3, 2}, {7, 2, 2}, {8, 4, 1}}
194, {{1, 1, 1}, {2, 1, 2}, {3, 4, 2}, {4, 4, 1},
{5, 2, 1}, {6, 3, 2}, {7, 2, 2}, {8, 3, 1}}
179, {{1, 1, 1}, {2, 1, 2}, {3, 4, 1}, {4, 4, 2},
{5, 2, 1}, {6, 3, 2}, {7, 2, 2}, {8, 3, 1}}

2	7	6	4
1	5	8	3

For the same problem,the solution of the proposed algoritm is 179, with one pass.

Table 1. Comparison of the final costs.

Number of departments	H 63	HC 66	CRAFT	MAT	FRAT	Cost Minimization
5	25*	29	25*	25*	25*	25*
6	43*	43*	43*	47	43*	43*
7	77	74*	74*	80	74*	74*
8	109	107*	124	124	107*	107*
12	301	304	289*	309	295	289*
15	617	578	583	691	575*	575*
20	1384	1319	1324	1437	1300	1294**
30	3244	3161	3148	3542	3129	3112**

* Optimum solution

** Best solution over all methods

5. Conclusions

The problem of assigning the facilities to locations remains to be one of the most interesting problems to many researches.There has been no method available till today to check the optimality of a solution to layout problems. This can only be done by exhaustive enumaration, which is not feasible for large-size problems. For example, the optimal solutions for the 15, 20 and 30 size problems of Nugent *et al.* [Nugent *et al.* 1968] are not yet known. This has prompted us to develop an efficient heuristic algorithm which would yield good solutions with reasonable less computational effort.

In improvement procedures the final solution depends on the initial assignment of facilities to locations. Initial solutions may be assigned at random or according to a set of criteria. The more the number of solutions examined the better the chance in reaching a near optimal final solution.

The assumptions made in the proposed method are that the departments are of equal area, the distance between adjacent departments are one unit each, the cost is proportional to the material handling and linear in nature . The computation time taken by the proposed algorithm is slightly less than that of the FRAT algorithm. Exact time comparisons, however, are not easy to make due to differences in programming techniques, inherent characteristics in procedure, computing systems and time cost involved.

The problem, as it is usually formulated under the cost criterion is presented. In light of all the criteria discussed and the results obtained, it is concluded that the better results may, however, be achieved through the application of the **Cost Minimization** algorithm.

REFERENCES

BUFFA, E.S., ARMOUR, G.C., and VOLLMANN,T.E. 1964. "Allocating Facilities With CRAFT," *Harward Business Rewiev*, 42,135.

DUTTA, K.N. and SAHU, S., 1982. " A Multigoal Heuristic for Facilities Design Problems: MUGHAL," *International Journal of Production Research*, 20,147.

EDWARDS, H.K., GILLETT, B.E., HALE,M.E. 1970. " Modular Allocation Technique," *Management Science*,17, 161.

HILLIER, F.S. 1963. " Quantitative Tools for Layout Analysis," *Journal of Industrial Engineering*, 14, 33.

HILLIER, F.S. and CONNORS,M.M. 1966. " Quadratic Assignment Problem Algorithms and the Location of Indivisible Facilities," *Management Science*, 13, 42.

KHALIL, T.M. 1973. " Facilities Relative Allocation Technique (FRAT)," *International Journal of Production Research*, 11, 183.

LEE, R.C. and MOORE,J.M. 1967. " CORELAP, Computerized Relationnship Layout Planning," *Journal of Industrial Engineering*, 18, 195.

NUGENT, C.E., VOLLMANN, T.E. and RUML, J. 1968." An Experimental Comparison of Techniques for the Assignment of Facilities to Locations," *Operations Research*, 16,150.

SEEHOF, J.M. and EVANS, W.O. 1967. " Automated Layout Design Problem,"*Journal of Industrial Engineering*, 18, 690.

TOMPKINS, A.J. and REED, R. 1980. "Flexible Facilities Design," *AIIE Transactions*, 12, 200.

VOLLMANN, T.E., NUGENT, C.E. and ZARTLER, R.L. 1968. " A Computerized Model for Office Layout," *Journal of Industrial Engineering*, 19, 321.

ZOLLER, K. and ADENDORFF, K. 1972. " Layout Planning by Computer Simulation," *AIIE Transactions*, 4, 116.

Appendix

Some of the final layout coordinates of the best solutions:

```
(* PR1/20* -1722-1297 *)CoordM=
{{16,1,1},{4,2,1},{11,3,1},{8,4,1},{17,5,1},
{19,1,2},{2,2,2},{15,3,2},{20,4,2},{5,5,2},
{14,1,3},{18,2,3},{12,3,3},{7,4,3},{13,5,3},
{9,1,4},{3,2,4},{10,3,4},{1,4,4},{6,5,4}};
```

9	3	10	1	6
14	18	12	7	13
19	2	15	20	5
16	4	11	8	17

```
(* PR3/20*-1770-1294 *)CoordM=
{{16,1,1},{13,1,2},{6,1,3},{9,1,4},
{1,2,1},{7,2,2},{5,2,3},{3,2,4},
{8,3,1},{20,3,2},{15,3,3},{10,3,4},
{11,4,1},{12,4,2},{2,4,3},{14,4,4},
{17,5,1},{4,5,2},{19,5,3},{18,5,4}};
```

9	3	10	14	18
6	5	15	2	19
13	7	20	12	4
16	1	8	11	17

```
(*PR2/30*-3879 - 3123*)CoordM=
{{15,1,1},{17,1,2},{26,1,3},{1,1,4},{24,1,5},{5,1,6},
{14,2,1},{23,2,2},{18,2,3},{22,2,4},{12,2,5},{6,2,6},
{27,3,1},{29,3,2},{11,3,3},{10,3,4},{9,3,5},{2,3,6},
{4,4,1},{30,4,2},{19,4,3},{7,4,4},{3,4,5},{13,4,6},
{20,5,1},{16,5,2},{8,5,3},{25,5,4},{28,5,5},{21,5,6}};
```

5	6	2	13	21
24	12	9	3	28
1	22	10	7	25
26	18	11	19	8
17	23	29	30	16
15	14	27	4	20

```
(*PR5/30*- 4112--3112*)CoordM=
{{26,1,1},{24,1,2},{1,1,3},{17,1,4},{25,1,5},{15,1,6},
{6,2,1},{12,2,2},{22,2,3},{10,2,4},{8,2,5},{28,2,6},
{21,3,1},{13,3,2},{7,3,3},{19,3,4},{16,3,5},{20,3,6},
{2,4,1},{9,4,2},{11,4,3},{29,4,4},{30,4,5},{27,4,6},
{5,5,1},{3,5,2},{18,5,3},{23,5,4},{4,5,5},{14,5,6}};
```

15	28	20	27	14
25	8	16	30	4
17	10	19	29	23
1	22	7	11	18
24	12	13	9	3
26	6	21	2	5

Appendix. Program Subroutine for CostMinimization of HELP.

(* :Title: A New Algorithm for Plant Layout :Heuristic Evaluation of Layout Planning (HELP) *)

(* :Authors:
Mehmet ÇINAR, Mehmet Emin YÜKSEL, Ömer TAHTASAKAL *)

(* :Summary:
This package intends to briefly introduce the routines for the HELP package*)

(* :Context: HELP` *)

(* :Package Version: 1.0 *)

(* :Copyright: Copyright 1991, Erciyes University *)

(* :History: *)

(* :Keywords: plant layout, multigoal programming, branch and bound approach, heuristic algorithms,utility theory,decision making *)

(* :Warning:
For large size problems, execution time will exponentially increase. *)

(* :Mathematica Version: 1.2 *)

(* :Limitation:
The problem has been formulated by the following assumptions :
1) The material handling cost is the linear function of rectilinear distance between the centroids of the departments.
2) Data for interdeparmental movements (loads) are constant.
3) Departments have an equal area.
4) Data for costs must be consistent with the distance of an edge of a unit square on the layout pattern.
5) Only if the departments share common boundaries are their closeness ratings added to get the score, for the particular assignment.
*)

(* :Discussion:*)

(* BEGIN PACKAGE *)

```
BeginPackage["HELP`","Global`"]

ClearAll[CostMinimization, ClosenessMaximization, Evaluation,TotalCost,
      TotalCloseness,DistanceMatrix,ChangeCoordinateMatrix, AddCost,
      AddCloseness,SquareMatrixQ,Repeat
]
```

(* USAGE INFORMATION *)

```
CostMinimization::usage=
      "CostMinimization[coordmat,costmat,ratingsmat] minimizes the total cost and calculates the total closeness value when swapping occurred. CostMinimization returns a list of coordinate matrices and cost values for each of them.The output is returned in the form of {{cost1,coordmat1},{cost2,coordmat2},...,}."

TotalCost::usage=
      "TotalCost[coordinatematrix,costmatrix] gives the total cost of a layout. Coordinatematrix represents the coordinates associated with departments. Costmatrix represents the costs occurred by distance travelled between the departments."

DistanceMatrix::usage=
      "DistanceMatrix[coordinatematrix] gives a matrix of rectilinear distances between the departments"

ChangeCoordinateMatrix::usage=
      "ChangeCoordinateMatrix[coordinatematrix,dept1,dept2] gives a new coordinate matrix by swapping the coordinates of dept1 and dept2."

AddCost::usage=
      "AddCost[closenesslist,costmatrix] adds cost values into the list (closenesslist) returned by ClosenessMaximization[]."
```

(* END USAGE INFORMATION *)

```
Options[CostMinimization]={ FixedDepartments -> None }
Options[ClosenessMaximization]={ FixedDepartments -> None }
Begin["`Private`"]
```

(* BEGIN COST MINIMIZATION *)

```
CostMinimization[coordmat_?MatrixQ,costmat_?SquareMatrixQ,opts___Rule]:=
      Block[
      (* list of local variables *)
{outputlist={}, coordm=coordmat,
      costm=costmat, distm=DistanceMatrix[coordm], m,
```

2

```
(* the department causing highest cost *)
        maxdist,
        (* the longest distance *)
        mindist,
        (* the shortest distance *)
        lowerbound,
        (* difference between maxdist and mindist *)
        switchvector, (* see below *)
        total, (* see below *)
        tot, (* mth element of total vector *)
        cred,  (* see below *)
        maxcred, (* the maximum element of cred vector *)
        currentcost, (* the cost of given layout *)
        cost, (* the cost of new layout *)
        fixdepts,  (* list of fixed located departments *)
        maxpos,  (* see below *)
i,j,j1,k,z,depts,coordm1,coordm2,distm1,distm2   (* dummy variables *)
},

(* some details about important local variables used in CostMinimization:
switchvector :
        the total cost of all possible trips of lenght greater than or equal to
        lower bound.
total :
        vector representing the total cost of each layout  when the departments
        are exchanged with the  position of mth department.
cred :
        vector representing the profitability of switching departments.
maxpos :
        the indice of department which yields maximum cost savings
*)
        fixdepts=FixedDepartments /. {opts} /. Options[CostMinimization];
        maxdist=Max[distm];
        mindist=Min[Map[If[#==0,maxdist,#]&,distm,{2}] ];
        lowerbound=maxdist-mindist;
        currentcost=TotalCost[coordm,costm];
        Print["INITIAL COST = " ,currentcost];
outputlist=Append[outputlist,{currentcost,coordm}];
 Repeat[   (* repeat 1 *)
        (* computing switch vector to obtain mth department *)
        distm=Map[ If[#<lowerbound,0,#]& ,distm, {2} ];
        switchvector=Table[
        Plus @@ (distm[[i]]*costm[[i]]),
        {i,Length[distm]} ];
        (* find the department m causing the highest cost
*)
```

```
(* if it appears in the fix departments list then find the next one *)
Repeat[
m=Position[switchvector,Max[switchvector]][[1,1]];
If[MemberQ[fixdepts,m],switchvector=MapAt[Min[switchvector]&,switchvector,{{
m}}]],
!MemberQ[fixdepts,m] ];
(* computes the profitability when the departments m and i are swapped *)
Repeat[ (* repeat 2 *)
total=Table[
coordm1=ChangeCoordinateMatrix[coordm,k,m];
TotalCost[coordm1,costm],
{k,Length[coordm]}];
tot=total[[m]];
cred=Map[(tot-#)&,total];
(* clear the profitabilities of fix departments *)
If[!SameQ[fixdepts,None],cred=MapAt[0&,cred,Map[{#}&,fixdepts] ] ];
maxcred=Max[cred];
If[maxcred>0,
maxpos=Position[cred,maxcred][[1,1]];
coordm=ChangeCoordinateMatrix[coordm,maxpos,m];
    cost=TotalCost[coordm,costm];
    outputlist=Append[outputlist,{cost,coordm}];
    Print["Swapping the departments ",m," <--> ",maxpos];
    Print["TOTAL COST = ",total[[maxpos]] ]
    ],
    maxcred<=0 ]; (* repeat 2 *)

lowerbound=lowerbound-mindist, (* lower bound will be decreased by the minumum
distance *)
lowerbound<mindist ]; (* repeat 1 *)
currentcost=TotalCost[coordm,costm];
(* following loop tests profitability of pairwise exchanges *)
Repeat[        (* repeat 3 *)
    CostDec=False;
    For[i=1, i!=Length[coordm], i++,
    If[!MemberQ[fixdepts,i],
For[j=i+1, j!=Length[coordm]+1, j++,
    If[!MemberQ[fixdepts,j],
    coordm1=ChangeCoordinateMatrix[coordm,i,j];
    cost=TotalCost[coordm1,costm];
    If[cost<currentcost,
    coordm=ChangeCoordinateMatrix[coordm,i,j];
    currentcost=TotalCost[coordm,costm];
    outputlist=Append[outputlist,{currentcost,coordm}];
    Print["Swapping the departments ",i," <--> ",j];
    Print["TOTAL COST = ",currentcost];
```

```
CostDec=True;
      depts={i,j};
Do[
      For[j1=1, j1<depts[[z]], j1++,
      If[!MemberQ[fixdepts,j1],
coordm1=ChangeCoordinateMatrix[coordm,j1,depts[[z]]];
cost=TotalCost[coordm1,costm];
If[cost<currentcost,
      coordm=ChangeCoordinateMatrix[coordm,j1,depts[[z]]];
      currentcost=TotalCost[coordm,costm];
      Print["Swapping the departments ",j1," <--> ",depts[[z]] ];
      Print["TOTAL COST = ",currentcost];
      outputlist=Append[outputlist,{currentcost,coordm}];
] ] ], {z,1,2}
];] (* if *)
] (* if *)
] (* for j *)] (* if *)] (* for i *),
      !CostDec ];(* repeat 3 *)
      Print["Optimum layout = "];
      Print[Transpose[coordm]//MatrixForm];
      Print["TOTAL COST = "];
      Print[currentcost];
      Return[outputlist]
] (* end of block *)

CostMinimization[coordmat_?MatrixQ,costmat_?SquareMatrixQ,ratingsmat_?Squar
eMatrixQ,opts___Rule]:=
      AddCloseness[CostMinimization[coordmat,costmat,opts],ratingsmat]
(* E N D   C O S T   M I N I M I Z A T I O N *)
Sort[cmat]
      ] /; Length[Transpose[coordmat]]==3
AddCloseness[costs_List,ratingsmat_?SquareMatrixQ]:=
      Map[Insert[#,TotalCloseness[ #[[2]], ratingsmat],2]&,costs]
AddCost[ratings_List,costmat_?SquareMatrixQ]:=
      Map[Insert[#,TotalCost[ #[[2]], costmat],1]&,ratings]
SquareMatrixQ[mat_?MatrixQ]:=
      MatrixQ[mat] && (Length[mat]==Length[Transpose[mat]])
Attributes[Repeat]={HoldAll}
Repeat[body_,test_]:=While[True,body;If[test,Break[] ] ]

(* E N D S U P P O R T I N G   R O U T I N E S *)
(* E N D   P A C K A G E *)

End[]
EndPackage[]
```