SHARE PROGRAM LIBRARY AGENCY

SHARE

PROGRAM NUMBER

3600-15.6.003

University of Miami

1365 MEMORIAL DRIVE - CORAL GABLES, FLORIDA
(305) - 284-6257

SHARE PROGRAM LIBRARY SUBMITTAL FORM

SHARE PROGRAM LIBRARY AGENCY
Triangle Universities Computation Center
Post Office Box 12076
Research Triangle Park, North Carolina
27709 USA

SPLA CONTROL NUMBER: 159

This form should be completed and submitted with the program package to the SHARE Program Library Agency at the address shown above. Standards and instructions for submitting programs are in the "SHARE Reference Manual".

(1) Program Number (to be filled in by SPLA)........ 360D-15.6.003

(2) System Type (machine)..................... IBM 360/370

(3) Search Key......................... FACILITY LAYOUT

(4) Programming Systems/Languages...... IBM Fortran IV, OS Assembler

(5) Author's Name and Address.......... G.C. Armour

(6) Direct Technical Inquiries to Name & Address (if different than Author)

Thomas L. Ward
Speed Scientific School, IE
University of Louisville
Louisville, KY 40292

(7) Title of Program........... Computerized Relative Allocation of Facilities Technique, CRAFT

(8) Submitter's Installation Membership Code.....

(9) Submitter's Own Program Identification and Suffix(Optional).. CRAFT 4.2

(10) Primary Subject Code..................... 15.6

(11) Minimum System Requirements

(12) New or Revision Code (if revision, show prior Program Number in Item 1) R

(13) Year Completed.................................. 1976

(14) Date of Submittal.............................. April 1976 10/79

(15) Documentation (number of original pages submitted)..........

(16) Abstract (should contain sufficient information for a reader to determine the value of the program). Listed on the reverse side of this form are subjects which may serve as a guide for a descriptive abstract.

Revised 4/74

SHARE PROGRAM LIBRARY SUBMITTAL FORM

Subject Guide:

a. Purpose
b. Programming Language used
c. Version and modification level or release number
d. Field of application
e. Type of routine (main program, subroutine, etc.)
f. Specific description of machine requirements

ABSTRACT:

Computerized Relative Allocation of Facilities Technique (CRAFT) accepts an initial layout pattern for a physical facility and generates improved layouts. The program is governed by heuristic rules which sequentially alter layout patterns while attempting to minimize variable material handling costs. Inputs are material handling flow and cost data, and an initial layout of departmental areas. CRAFT generates the variable cost of material handling for the initial layout. The program then tries combinations of two department exchanges, attempting to find a less costly layout. Modifications continue until no further cost reduction is possible. CRAFT can also be applied to any movement problem that can be represented on a cost-per-foot basis. The flow of people in an office layout is an example. CRAFT was written by Armour (c.1961), revised (CRAFT IV) by Fagnani in 1967, further revised (called CRAFT 4.1 here), and submitted to SHARE in 1974. CRAFT 4.2 modifies CRAFT 4.1 for the IBM 360/370. FORTRAN IV with some assembler subroutines; requires 220K to compile and link-edit and 160K to load without overlays.

(Please attach additional pages if necessary)..........Total pages attached ____

Permission to Publish

"I hereby give the SHARE Program Library Agency permission to reprint, reproduce, and distribute this program."

(17) Signature of Submitter and Date Thomas L. D[illegible] 16 APR 76

(18) Signature of Installation Addressee ____

1.0 Introduction

The structure and theory of CRAFT are described in Armour and Cramer (1965) and Fagnani (1967). It was originally written by Armour in the course of his dissertation research at UCLA. The CRAFT IV revision produced by Fagnani provided for 45 (rather than 40) departments and a 40 by 40 (rather than 30 by 30) spatial array. In addition CRAFT IV permits (at the control of the operator) exchange of departments of approximately equal areas and permits supression of exchanges with low actual cost savings.

The version submitted to SHARE in February 1974 (which we will call CRAFT 4.1) was esentially CRAFT IV as described by Fagnani (1967) but with two additions.

The first difference between CRAFT IV and CRAFT 4.1 was the addition of input variables IC and ISEED together with a pseudorandom number generator (FRAND and FRND1). When IC is greater than or equal to 50, CRAFT 4.1 behaves like CRAFT IV. When IC is less than 50, uncertainty is introduced in the selection of department exchanges. Large IC (say, 2 or 3) emphasize the most recently selected exchanges (recent values of AACOST). Small values of IC (say, 0 or 1) emphasize earlier exchanges at each step. ISEED is the seed for the random number generator.

The second difference between CRAFT IV and CRAFT 4.1 is the provision for new input parameter IREUSE. If IREUSE is zero, two or more data decks may be submitted with each job as before. If IREUSE is 1 on the first control card of the j-th deck, then the volume and cost data and the initial spatial array used by the (j-1)-th deck are reused for the j-th job.

These four decks may be concatonated and run as a single job. Each deck has been duplicated (with the exception of an "END" card) and placed as a psnedo-departmental list (6A or 6B) so that the deck is listed at the conclusion of the problem.

4.0 Tape Key

This volume contains three Files and three Tape Marks arranged as follows:

File 1. FORTRAN Source Deck

EBCDIC

Sequence 00000 through 22780 in columns 76-80; CRA in cols. 73-75; 2279 cards.

2279 card images blocked 80 per block

29 blocks of 6400 characters each

T/M

File 2. OS ASSEMBLER Source Deck

EBCDIC

Sequence 30000 through 30970 in columns 76-80; CRA in cols. 73-75: 98 cards.

98 card images blocked 80 per block

2 blocks of 6400 characters each

T/M

File 3. Sample Data Input

EBCDIC

Sequence 40000 through 41670 in columns 76-80; CRA in cols. 73-75; 168 cards.

168 card images blocked 80 per block

3 blocks of 6400 characters each

T/M

Note: Appendix C refered to in Section 2.0 of this submittal is contained in Fagnani (1967).

CRAFT 4.2 is a revision of CRAFT 4.1 that permits it to run on IBM 360/370. In addition it corrects several problems that limited full use of the input options.

2.0 How to Use The Program

A complete data deck is required for each problem, but the user may insert as many decks as desired in a given run. The n-th data deck is placed behind the (n-1)-th deck.

Two alternate deck set-ups are available. Set-up A is for a job shop or similar layout with many flow relationships:

1A - Title Card
2A - Control Card(s)
3A - Flow Matrix
4A - Cost per Unit Distance Matrix
5A - Interspatial Array Matrix
6A - Department List (optional)
7A - "END" Card

Set-up B is for a more restricted flow pattern.

1B - Title Card
2B - Control Card(s)
3B - I, J Element List
4B - "99" Card
5B - INTERSPATIAL ARRAY Matrix
6B - Department List (optional)
7B - "END" Card

Set-up B allows more input flexibility and is recommended for all but problems with many interdepartmental flows.

1A and 1B - TITLE CARD FORMAT 20A4

This card is the first card in any data set. All 80 columns may be used for problem identification. The project title will be reprinted on the summary sheet exactly as it appears on this card.

2A and 2B - FIRST CONTROL CARD   FORMAT (8(I2, 1X), 2(F5.0,1X),...)

The Control Card is used to describe limiting parameters of the problem. It also specifies program internal processing options.

Columns			Variable
1-2	Number of Departments	MAX 45	NDEPT
4-5	Number of rows in the spatial configuration	MAX 40	IROW
7-8	Number of columns in the spatial configuration	MAX 40	ICOL
10-11	Analyzer control 00 two department moves only 01 three department moves only 02 two department moves followed by three department moves 03 three department moves followed by two department moves 04 choose best of two or three department moves at each iteration (recommended)		ICTL
13-14	Input/output control 00 print first and last layout 01 print first layout and for each iteration the most favorable layout found during the iteration		IOCTL
16-17	Debugging parameter 00 no failure messages 01 write exchange failure and no cost reduction messages 02 same as above but also prints results of search for best move		ICHK
19-20	Number of departments to be fixed in place NOTE: the acutal departments to be fixed are placed on the second control card IF(IFIX.EQ.0), omit the second control card		IFIX

SECOND CONTROL CARD — FORMAT (40I2)

Columns		Variable
1-2	Department number of first department to be fixed in place, right justified	IDFIX (I)
3-40	Same as above 2 columns for each additional department to be fixed in place	

2A and 2B - FIRST CONTROL CARD CONTINUED

Columns		Variable
22-23	00 or blank specified job set-up number one (refer to 3A and 4A). 02 specifies use of element list (refer to 3B and 4B).	IPTS
25-29	Percent of department size variability FORMAT (F 5.0) FOR EXAMPLE: 00.05 = 5% (refer to Appendix C for explanation of this feature)	PONT
31-35	Cost limiting factor FORMAT (F 5.0) FOR EXAMPLE: 45.00= $45.00, all actual exchanges must be greater than $45 (EPSLN) to be accepted (refer to Appendix C for further information)	EPSLN
37-38	Exchange control FORMAT (I2) If IC=50 deterministic (CRAFT IV) exchange rules are used. If IC.LT. 50 uncertainty is introduced.	IC
40-49	Random number seed FORMAT (I10) Any number between 1 and 2147483647 inclusive. Default is 1.	ISEED

51-52 Deck control IREUSE

FORMAT (I2)

00 Complete problem deck including flow and cost data (3A and 4A, or 3B and 4B) and initial layout (5A or 5B) required.

01 Flow, cost, and initial layout from previous deck are reused.

3A FLOW MATRIX FORMAT (20F4.0)

Flow volumes are punched for every interdepartment relationship. Each department (row of the volume matrix (department from)) is started on a new card. Twenty paired department volume relationships may be punched per card. (e.g., for the first card punches in cols. 9-12 would input the volume flowing from department 1 to department 3.)

If more than 20 departments are specified, use additional cards. Volumes are treated uniquely on both sides of the diagonal in order that volumes in different directions can be represented at different travel costs. (See cost per unit distance array below). If travel cost is the same for both directions of flow, place 1/2 of the volume of each side of the diagonal.

There should be as many sets of cards for the array as is indicated in columns 1-2 of the control card. There should be the same number of fields utilized per set as is indicated in columns 1-2 of the control card. There is one card per set if the number of departments (col. 1-2 of control card) is 20 or less. There are two cards per set if the number of departments (col. 1-2 of control card) is greater than 20 and less than 46.

The decimal point is not punched. A decimal point will be considered to immediately follow the right most digit of each field for purposes of computation.

4A COST PER UNIT DISTANCE MATRIX FORMAT (20F4.3)

This array is punched on a one to one relationship with the flow array. (e.g., for the first card of the first act punches in Cols. 9-12 would input the unit cost of moving a unit load a unit distance from department number 1 to department number 3.

The decimal point is not punched. A decimal point will be considered to lie between the first and second positions of each field for purposes of computation. (e.g., a punch of 1015 in a field would be treated as 1.015 for computation).

3B - I, J ELEMENT LIST

Replaces - 3A and 4A in set-up B

FORMAT (2[I2,1X], 2[F5.0, 1X])

Variables: I, J. DST, CVL

I= Department from which product flow is emerging
J= Department destination of product flow emerging from I
DST= The quantity of flow from Department I to Department J
CVL= The cost to move this flow one grid square (i.e., cost per unit distance

EXAMPLE:

01, 05, 010.0, 00.20

Flow from department #1 to department #5 of 10 units, at a cost of $.20 a unit for each grid square moved.

Each card represents one from I to J move relationship. Every such non-zero relationship must be represented by one card. No specific order is required in this section of the data set except the requirement of a "99" card (4B in deck set-up), to signal termination of this section.

4B - "99" CARD

A card having the number 99 in its first 2 columns, signals termination of I, J ELEMENT LIST

5A-5B INTER SPATIAL ARRAY MATRIX   FORMAT (40I2)

Punch each row of the array on a separate card. Department numbers of the departments occupying each space are punched on the card in two column fields. The array is punched by rows. There should be as many cards punched as is specified in columns 4-5 of the control card. There should be as many fields punched on each card as is specified in columns 7-8 of the control card.

Department numbers are converted by the program to alphabetic characters for print out (e.g., 01 = A, 02 = B,..., 26 = Z, 27 = AA, 28 = EB, etc.).

Irregularities in building configuration should be filled in by adding artificial departments until a rectangular building configuration is achieved. These artificial departments may be held fixed. See control card.

The maximum number of times that any one department may be punched is 150. No department may be disjointed. Each department which appears in more than one field or card must be punched so that when the keypunched spatial array is listed each department number has a like department number immediatley adjacent in the same row or column. Departments may have multiple indentations along one axis but not along both axes. A department my not completely surround another department on all sides.

6A-6B DEPARTMENT LIST (OPTIONAL)

Any number of cards may be placed between the last card of section 5A-5B and the "END" card (7A or 7B). All 80 fields of each card may be used. The information is read under an Alphanumeric Format and reproduced at the end of SMRY exactly as it appears on the cards.

7A-7B "END" CARD

A card having the letters E, N, D in columns one, two, and three respectively, signals the termination of a data set.

3.0 Testing

The submittal includes four test decks. These will duplicate the layouts and cost data for the following:

3.1 Armour and Cramer (1965), example 1, sheets 1-15.

3.2 Armour and Cramer (1965), example 2, sheets 16-21.

3.3 Armour and Cramer (1965), example 3, sheets 22-28.

3.4 Fagnani (1967), figures 5-9.

COMPUTERIZED RELATIVE ALLOCATION

OF FACILITIES TECHNIQUE

Roger A. Fagnani

<u>DISCLAIMER</u>

Operations Research Department
Finance Staff
December 1967

CONTENTS

SUMMARY

Computerized Relative Allocation of Facilities Technique (CRAFT) is a computer program designed to produce suboptimal plant layouts based on material flow and handling costs.

The original CRAFT program was written by Dr. G. C. Armour in 1963 and was subsequently made available through the IBM SHARE Program Library.

The Manufacturing Studies group of the Operations Research Department undertook an evaluation of CRAFT to determine its worth as a layout tool for the Company. In order to test the program, it was found necessary to modify the SHARE version of the program (CRAFT-I); three different versions of CRAFT were developed. The final version, CRAFT-IV, was then tested with a series of Company plant layout problems.

The Material and Equipment Engineering Department, Manufacturing Staff, supplied the data for the tests and led in the evaluation of the results.

From the study, the following is concluded: CRAFT-IV is an effective means for producing good relative location patterns (layouts). The savings potential of the program is large enough to merit continued development and Company application.

ACKNOWLEDGMENTS

Mr. James T. Carson, Materials and Equipment Engineering Department, Manufacturing Staff, provided the test data and led in the analysis of the test results. Technical supervision was provided by Mr. Douglas F. Thompson, Operations Research Department. Their efforts, which made possible the successful conduct of the study, are gratefully acknowledged.

DESCRIPTION OF CRAFT

Computerized Relative Allocation of Facilities Technique (CRAFT) is a computer program designed to determine suboptimal layout patterns for physical facilities. The program is governed by a set of heuristic rules which sequentially alter layout patterns while attempting to minimize variable material handling costs. Input consists of material handling flow and cost data, and an initial layout comprised of departmental areas. Using this information, CRAFT generates the variable cost of material handling for the initial layout. The program then tries different combinations of two department exchanges, attempting to find a less costly layout pattern. These modifications continue until no further improvement can be made in the material handling cost.

It should be noted that CRAFT can be applied not only to material handling problems involving product flows but to any movement problem that can be represented on a cost-per-foot basis. The flow of people in an office layout is an example.

Basic Methodology of CRAFT

Consider the problem of determining the relative location of departments in a manufacturing plant in such a manner that an objective function (i.e., variable material handling cost) is minimized. Let:

f_{ij} = number of unit loads of product moving from department i to department j.

c_{ij} = cost to move one unit load a unit distance (length of the side of a grid square) from department i to department j.

l_{ij} = the distance between departments i and j.

n = number of departments.

The following assumptions are made:

(1) product flow (f_{ij}) between departments i and j is constant.

(2) move cost per unit distance (c_{ij}) is constant.

It is possible to represent the departmental cost (DC_{ij}) of material handling between departments i and j as:

$$DC_{ij} = f_{ij}c_{ij}l_{ij}$$

The sum of all the departmental costs will be equivalent to the variable material handling cost for a particular relative location pattern or layout.

CRAFT has three basic elements in its input:

1. A matrix representing the product flow (f_{ij}). It is assumed that no flow exists within departmentsl areas (f_{ij} = 0 when i = j).

FROM \ TO		DEPARTMENTS 1	2	3	...	n
	1	0	f_{12}	f_{13}	...	f_{1n}
	2	f_{21}	0	f_{23}	...	f_{2n}
DEPTS.	3	f_{31}	f_{32}	0	...	f_{3n}
	⋮					
	n	f_{n1}	f_{n2}	f_{n3}		0

2. A matrix representing cost per unit distance (c_{ij}) with $c_{ij} = 0$ when i = j.

FROM \ TO DEPARTMENTS	1	2	3	...	n
DEPTS. 1	0	c_{12}	c_{13}	...	c_{1n}
2	c_{21}	0	c_{23}	...	c_{2n}
3	c_{31}	c_{32}	0	...	c_{3n}
⋮					
n	c_{n1}	c_{n2}	c_{n3}	...	0

3. An initial layout or interspatial array represented in a matrix no larger than 30 x 30 grid squares.

From the initial layout the program computes the rectangular distance (l_{ij}) between the centers of all departments in the layout.

FROM \ TO DEPARTMENTS	1	2	3	...	n
DEPTS. 1	0	l_{12}	l_{13}	...	l_{1n}
2	l_{21}	0	l_{23}	...	l_{2n}
3	l_{31}	l_{32}	0	...	l_{3n}
⋮					
n	l_{n1}	l_{n2}	l_{n3}	...	0

Departments are not point locations and can vary in size and shape. A rectangular measure of the distance between departments is therefore used, based on departmental geographical centers. The rectangular measure approximates aisle movement and the geographical center serves as a convenient origin for computational purposes.

The total variable cost of material handling is TC_o where:

$$TC_o = \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} c_{ij} l_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{n} DC_{ij}$$

CRAFT attempts to minimize this function by exchanging combinations of departments. These exchanges endeavor to reduce l_{ij}. Matrix elements f_{ij} and c_{ij} are constant for any particular problem.

The Computer Program in Brief

The steps taken by CRAFT in its attempt to reduce variable material handling costs are (refer to Figure #1, next page):

1. Read and check the initial data:

   a) The physical dimensions (in grid squares) of the overall facility, and the number of departments within it.

   b) The volume of product flow between all paired departments for some convenient time period.

   c) The cost per unit load per unit distance for movement between all paired departments.

   d) The initial layout pattern.

   e) Designation of departments fixed as to location in the layout.

2. Calculate departmental centers and then compute a matrix of distances between all pairs of centers.

3. Compute the value of objective function for the initial layout (i.e., the total variable material handling cost for the interdepartmental product flows in the initial layout).

4. Tentatively determine how the initial layout can be modified so that a lower cost layout will result.

5. a) Make the exchange of department locations to achieve the new layout determined in the preceding step, and do so in such a manner that the new arrangement is a valid one. (Rules that determine the validity of department shapes can be found in the SHARE library documentation for the CRAFT program, SOA 3391, October 21, 1965.)

   b) If the resulting total interdepartmental cost is greater than the total cost for the initial layout, restore the old layout, invalidate this proposed layout, and return to step 4.

6. Print a scaled diagram of the new layout problem and its corresponding objective function value.

7. Repeat steps 2 through 6 until no further reduction of the objective function can be made.

The Heuristics of CRAFT

CRAFT establishes a list of possible departmental exchanges, each representing a small modification of the present layout which can result from exchanging the locations of two departments. To be eligible for the list, each pair of departments must meet one of the following criteria:

1. They are equal in size (see Figure #2).
2. They have a common border (see Figure #3).
3. They border on a common third department (see Figure #4).

CRAFT chooses the modification which it estimates will yield the greatest improvement over the cost of the present layout. This new configuration then presents another list of possibilities, and so on, until there is no pair of departments on the current list of possibilities which promises an improvement over the cost of the present layout.

The program uses the following procedure. It estimates the change in transportation costs if the center of department"A" were to be located at the present center of department "B", and the center of "B" were at the center of "A". The pair of departments with the greatest estimated cost reduction is then actually switched, yielding a change in transportation cost which may differ from the estimate (refer to "EPSLN" appendix C). Should the new transportation cost be increased, this pair of departments will be rejected and the pair with the next best estimated improvement will be exchanged.

CRAFT FLOW DIAGRAM

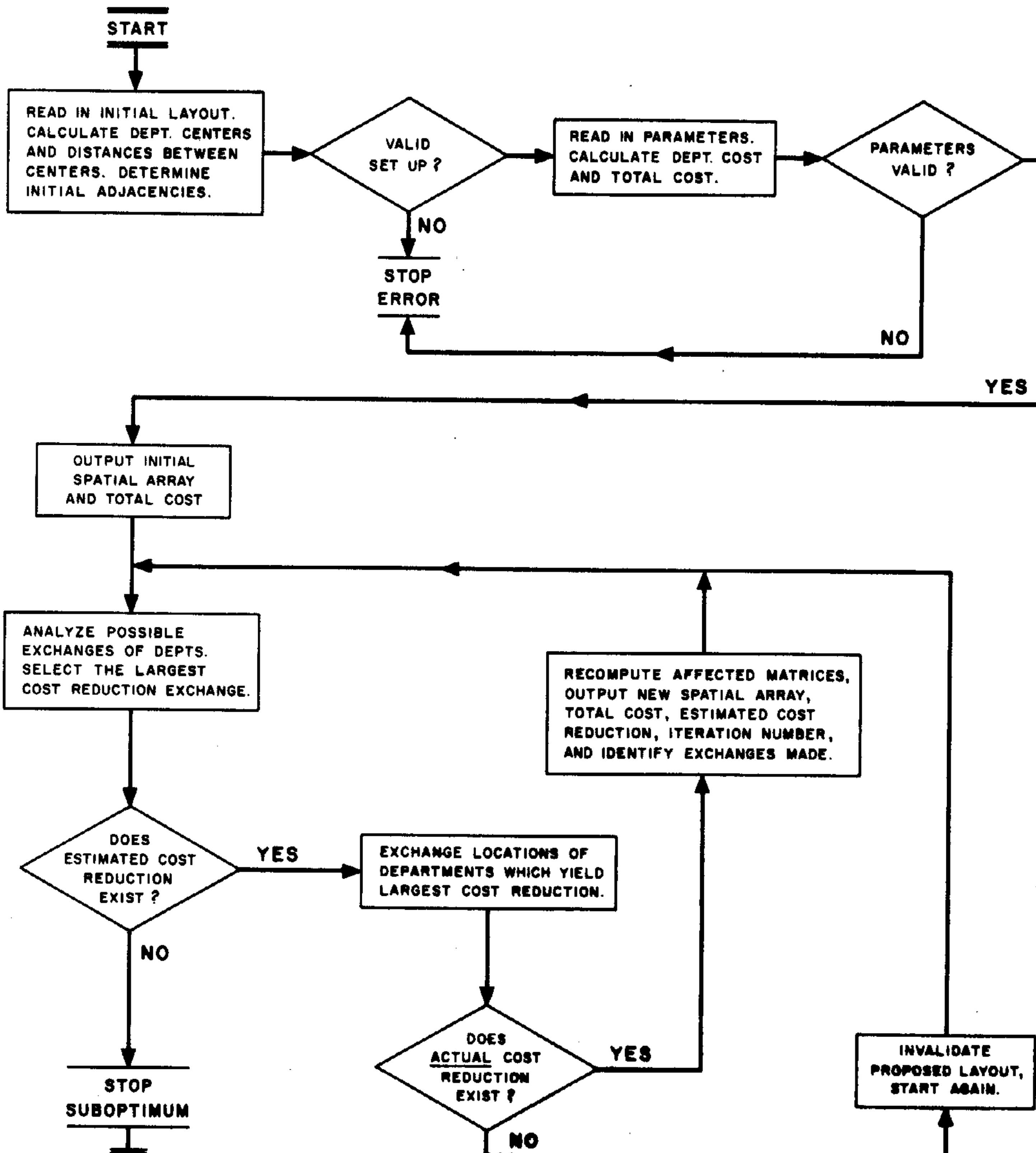
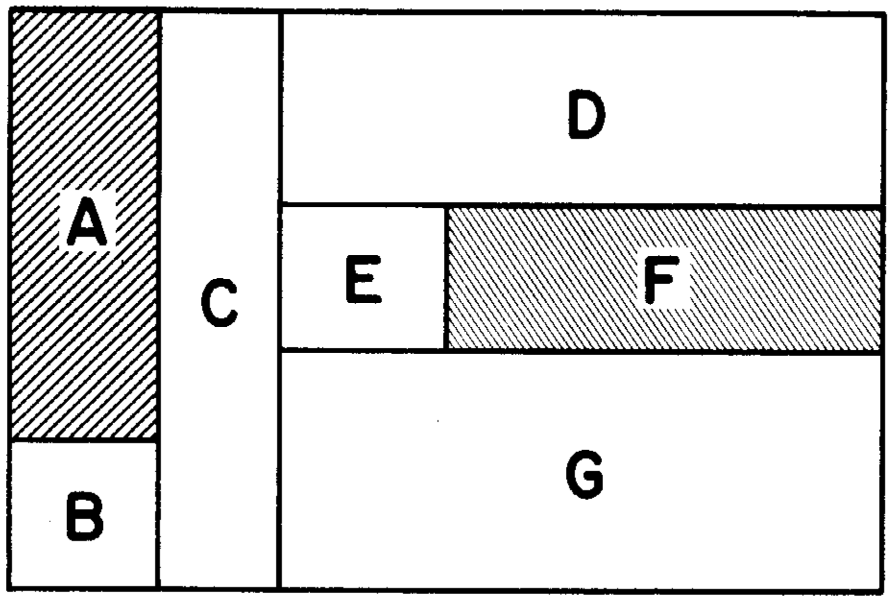


FIGURE # 1

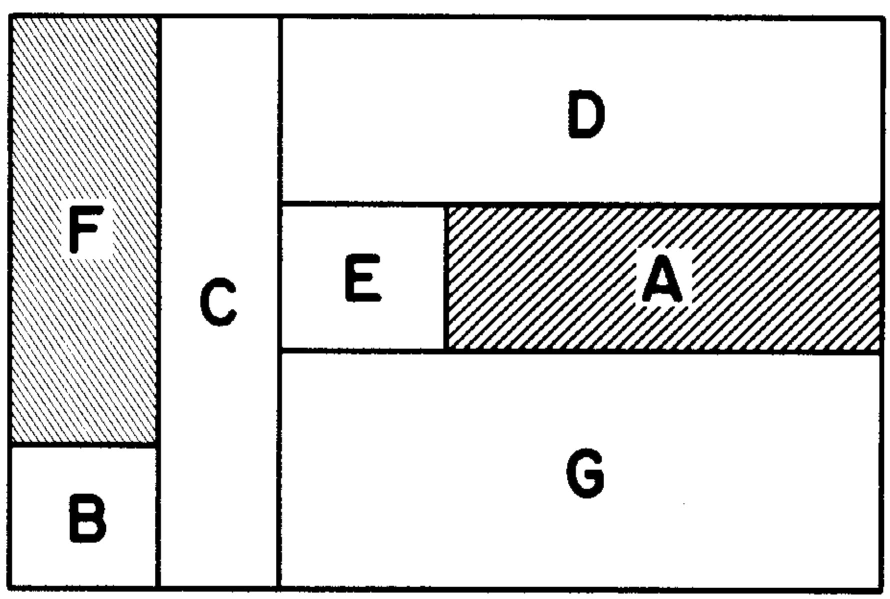
DEPARTMENTAL EXCHANGES

METHOD NO. 1

SWITCHING TWO DEPARTMENTS
EQUAL IN AREA
(NOT ADJACENT)



INPUT LAYOUT



REVISED LAYOUT
AFTER SWITCHING DEPTS. "A" & "F"

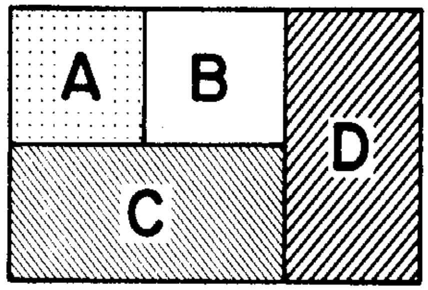
NOTE: IF DESIRED, COMPUTER PROGRAM PERMITS SWITCHING WITHIN A VARIABLE PERCENTAGE AREA ALLOWANCE OF TWO UNEQUAL DEPARTMENTS.

FIGURE #2

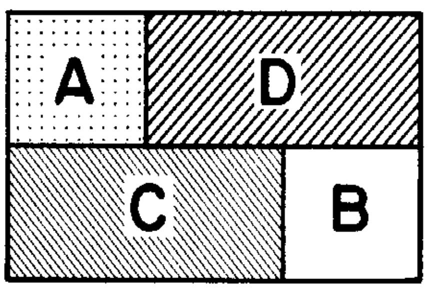
DEPARTMENTAL EXCHANGES

METHOD NO. 2

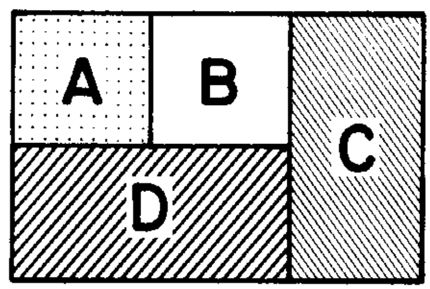
SWITCHING TWO ADJACENT DEPARTMENTS



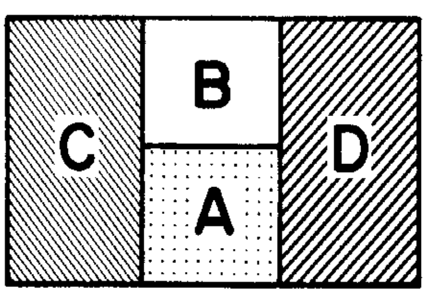
INPUT LAYOUT



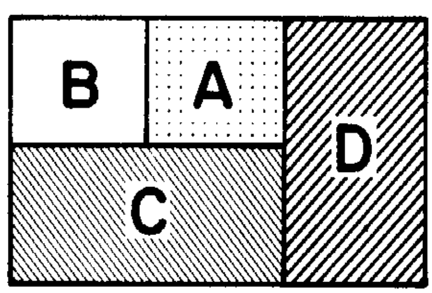
EXCHANGE OF B & D



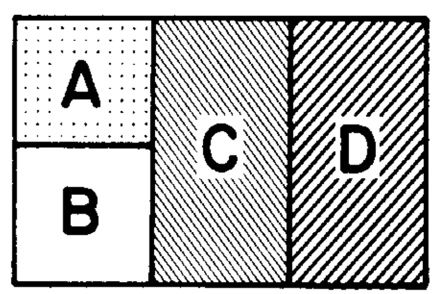
EXCHANGE OF C & D



EXCHANGE OF A & C



EXCHANGE OF A & B



EXCHANGE OF B & C

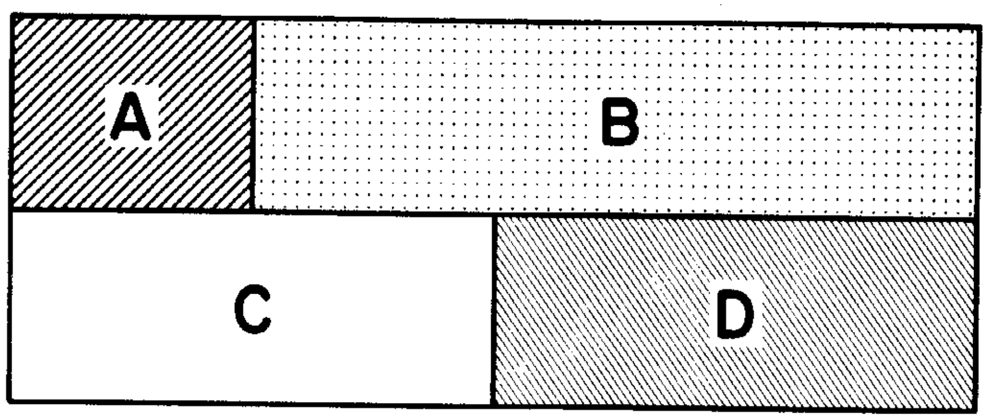
ALTERNATE LAYOUT CONSIDERATIONS

FIGURE # 3

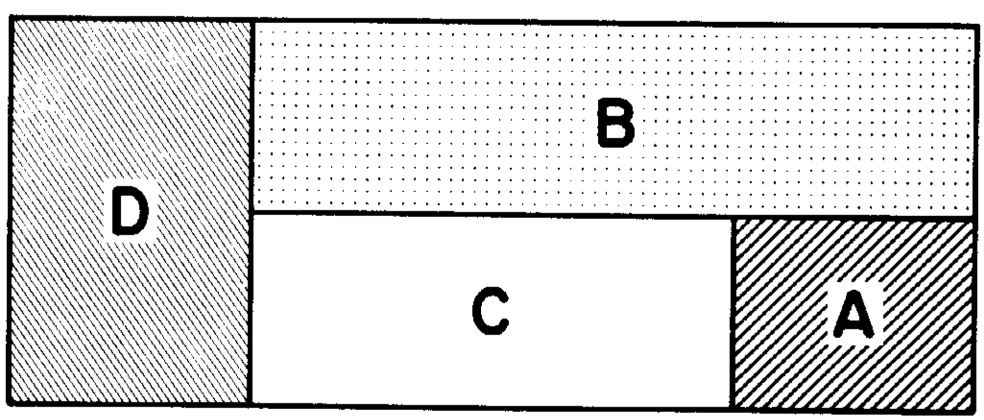
DEPARTMENTAL EXCHANGES

METHOD NO. 3

SWITCHING TWO DEPARTMENTS BORDERING ON COMMON DEPARTMENT



INPUT LAYOUT



REVISED LAYOUT
AFTER SWITCHING DEPTS. "A" & "D"

FIGURE #4

Sample CRAFT Problem

The CRAFT output for a sample layout problem follows (see Figures #5 through #9). The internal search for exchange candidates and the resulting cost estimates are printed before each iteration change. The meaning of the various tabular items is explained below.

I J MA MB M N ACOST BCOST

I, J = two departments being considered for exchange.

MA, MB = best two departments thus far found in search.

ACOST = cost in "E" notation (.54783 + 002 = $.54783 \times 10^2$ or $547.83) of MA and MB exchange.

BCOST = cost in "E" notation of I and J exchange.

M, N = indexing parameters used for error detection purposes.

The second list heading;

I J K MA MB MC M N NN ACOST BCOST

is for Method Number Three switches. Variables K and MC are used for the addition of the third department.

INTERDEPARTMENT PRODUCT FLOW

	A	B	C	D	E	F
A	0.	0.	0.	0.	0.	20.000
B	0.	0.	0.	0.	0.	30.000
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	0.	0.	0.	0.	0.	10.000
F	20.000	30.000	0.	0.	10.000	0.

INTERDEPARTMENT MOVE COST PER UNIT LOAD PER UNIT DISTANCE

	A	B	C	D	E	F
A	0.	0.	0.	0.	0.	1.000
B	0.	0.	0.	0.	0.	1.000
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	0.	0.	0.	0.	0.	1.000
F	1.000	1.000	0.	0.	1.000	0.

COVOL = (MOVE COST/LOAD.X(NO. OF LOADS)

	A	B	C	D	E	F
A	0.	0.	0.	0.	0.	20.000
B	0.	0.	0.	0.	0.	30.000
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	0.	0.	0.	0.	0.	10.000
F	20.000	30.000	0.	0.	10.000	0.

FIGURE *5

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	B	B	D	D	D	D	D	D	D	D	D
3	A	B	B	D	D	D	D	D	D	D	D	D
4	A	C	C	E	E	E	E	E	F	E	F	E
5	A	C	C	E	F	F	F	F	F	F	F	F
6	A	C	C	E	F							F
7	A	C	C	E	F							F
8	A	C	C	E	F							F
9	A	C	C	E	F							F
10	A	C	C	E	F							F
11	A	C	C	E	F							F
12	A	C	C	E	F	F	F	F	F	F	F	F

TOTAL COST 1185.73 EST. COST REDUCTION 0. MOVEA MOVEB MOVEC ITERATION 0

FIGURE #6

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2	0	0	0.54783+002	0.54783+002
1	3	1	3	0	0	0.11043+003	0.11043+003
2	3	2	3	0	0	0.33000+003	0.33000+003
1	4	2	3	0	0	0.33000+003	0.11043+003
2	4	2	3	0	0	0.33000+003	0.33000+003
3	4	2	3	0	0	0.33000+003	0.
3	5	2	3	0	0	0.33000+003	-0.34706+002
4	5	2	3	0	0	0.33000+003	-0.34706+002
5	6	5	6	0	0	0.47647+003	0.47647+003

I	J	K	MA	MB	MC	M	N	NN	ACOST	BCOST
5	3	1	5	3	1	0	0	0	0.89923+002	0.89923+002
5	4	1	5	3	1	0	0	0	0.89923+002	0.89923+002
5	3	2	5	3	2	0	0	0	0.28941+003	0.28941+003
5	4	2	5	3	2	0	0	0	0.28941+003	0.28941+003
6	5	3	6	5	3	0	0	0	0.52573+003	0.52573+003
6	5	4	6	5	3	0	0	0	0.52573+003	0.52573+003

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	B	B	D	D	D	D	D	D	D	D	D
3	A	B	B	D	D	D	D	D	D	D	D	D
4	A	F	F	F	F	F	F	F	E	E	E	E
5	A	F						F	E			E
6	A	F						F	E			E
7	A	F						F	E	E	E	E
8	A	F						F	F	E	C	C
9	A	F						F	C	C	C	C
10	A	F						F	C			C
11	A	F						F	C			C
12	A	F	F	F	F	F	F	F	C	C	C	C

TOTAL COST 851.91 EST. COST REDUCTION 525.73 MOVEA C MOVEB E MOVEC F ITERATION 1

21975

FIGURE # 7

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2		0	0.54783+002	0.54783+002
1	4	1	2		0	0.54783+002	-0.12457+003
2	4	1	2		0	0.54783+002	-0.22500+002
3	4	1	2		0	0.54783+002	0.
3	5	1	2		0	0.54783+002	-0.19608+000
4	5	1	2		0	0.54783+002	-0.13529+002
1	6	1	6		0	0.30705+003	0.30705+003
2	6	1	6		0	0.30705+003	0.36237+002
3	6	1	6		0	0.30705+003	-0.71618+003
4	6	1	6		0	0.30705+003	0.18955+003
5	6	1	6		0	0.30705+003	-0.30551+003

I	J	K	MA	MB	MC	M	N	NN	ACOST	BCOST
5	4	1	1	6	6	0	0	2	0.	-0.48753+002
5	4	2	5	4	2	0	0	2	0.12059+002	0.12059+002

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	F	F	F	F	F	F	F	F	F	F	F	F
2	F	B	B	D	D	D	D	D	D	D	D	D
3	F	B	B	D	D	D	D	D	D	D	D	D
4	F	F	F	F	F	F	F	F	F	E	F	E
5	F							F	F			E
6	F							F	F			E
7	F							F	F	E	F	E
8	F					F	F	F	F	E	C	C
9	F	F	F	F	F	F	A	A	C	C	C	C
10	F	A	A	A	A	A	A	A	C			C
11	F	A						A	C			C
12	F	A	A	A	A	A	A	A	C	C	C	C

TOTAL COST 664.09 EST. COST REDUCTION 307.05 MOVEA A MOVEB F MOVEC ITERATION 2

I	J	MA	MB	M	N	ACOST	BCOST
1	3	1	6	[illegible]	0	0.	-0.19159+003
2	4	1	6	[illegible]	0	0.	-0.80625+002
3	4	1	6	[illegible]	0	0.	0.
3	5	1	6	[illegible]	0	0.	-0.94314+002
4	5	1	6	[illegible]	0	0.	-0.76471+001
1	6	1	6	[illegible]	0	0.	-0.44578+003
2	6	1	6	[illegible]	0	0.	-0.30469+003
3	6	1	6	[illegible]	0	0.	-0.62013+003
4	6	1	6	[illegible]	0	0.	-0.22261+003
5	6	1	6	[illegible]	0	0.	-0.54105+003

I	J	K	MA	MB	MC	M	N	NN	ACOST	BCOST
5	3	1	1	6	0	0	0	2	0.	-0.14834+001
5	4	2	1	6	0	0	0	2	0.	-0.38456+002

MATERIAL HANDLING COST DATA

TEST PROBLEM CRAFT IV 11-7-67

ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
0	1185.73	0.	0.					
1	851.91	333.82	333.82	C	E	F		
2	664.09	187.81	521.64	A	F			

PERCENT IMPROVEMENT = 43.99

FIGURE # 9

CHRONOLOGY OF DEVELOPMENT

The CRAFT program described previously is the basic program available from the SHARE library. The Manufacturing Studies Group of the Operations Research Department undertook an evaluation of CRAFT to determine its value as a layout tool for the Company. Tests of the program with actual manufacturing plant layout problems revealed the need for modifying the original SHARE version. The following sections describe the tests of CRAFT and the resulting program modifications. A more detailed discussion of the modifications is provided in Appendix C.

CRAFT-I

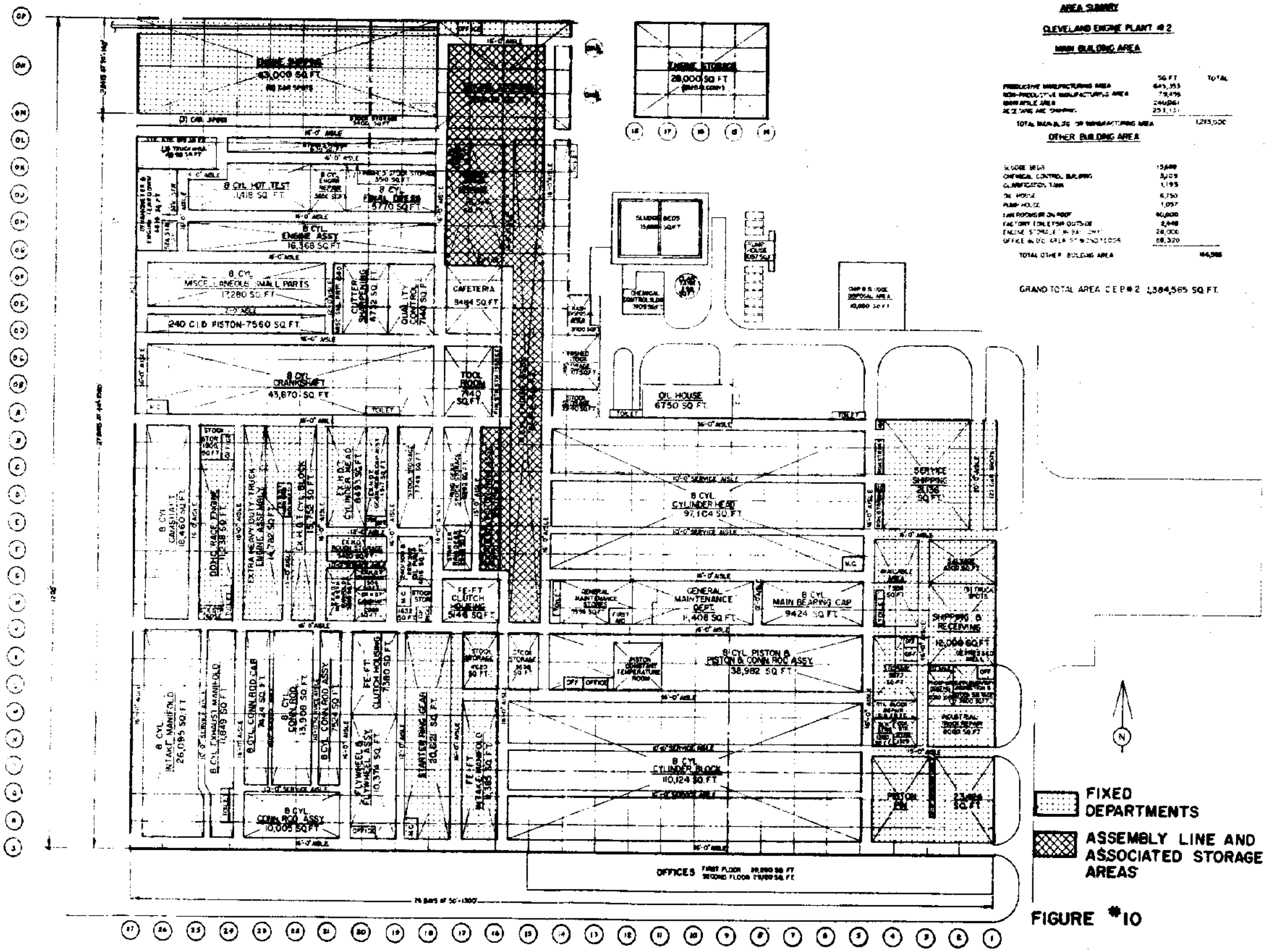
CRAFT-I is the designation given to the computer program obtained from the IBM SHARE PROGRAM LIBRARY. Using CRAFT-I an attempt was made to evaluate the program on Company problems. Mr. J. T. Carson, Materials and Equipment Engineering Department, Manufacturing Staff, supplied the necessary data and led in the evaluation of the various tests. An approved expansion of Cleveland Engine Plant #2 was chosen as the first layout test.

First Solution Attempt

Adapting the Cleveland Plant into a 30 x 30 grid proved more difficult than expected. The various departmental areas differed markedly in size and shape, necessitating a large amount of manual editing before arriving at the initial CRAFT layout (see Figures #10 and #11 for the original Cleveland layout).

CRAFT requires a rectangular plant layout pattern. This meant that the blank areas of the Cleveland plot plan had to be represented with dummy departments (departments AA, BB, CC). Departments J, K, Z, FF, DD, Y, and EE were to remain in their present locations and therefore designated as "fixed".

The first Cleveland tests showed the CRAFT could not maintain suitable departmental rectangularity. The initial run produced 14 iterations, but all the layouts beyond number four were unacceptable. The departmental areas had lost their oroginal shapes, aisle patterns had been destroyed, and the overall layout arrangement was unsuitable for manufacturing.



AREA SUMMARY

CLEVELAND ENGINE PLANT #2

MAIN BUILDING AREA

	SQ FT	TOTAL
PRODUCTIVE MANUFACTURING AREA	645,353	
NON-PRODUCTIVE MANUFACTURING AREA	79,495	
MAIN AISLE AREA	244,061	
RECEIVING AND SHIPPING	253,131	
TOTAL MAIN BLDG. OF MANUFACTURING AREA		1,213,000

OTHER BUILDING AREA

SLUDGE BEDS	15,600	
CHEMICAL CONTROL BUILDING	3,109	
CLARIFICATION TANK	1,195	
OIL HOUSE	6,750	
PUMP HOUSE	1,097	
FAN ROOMS ON ROOF	40,000	
FACTORY TOILETS OUTSIDE	2,448	
ENGINE STORAGE	28,000	
OFFICE & DC AREA 1ST & 2ND FLOORS	68,320	
TOTAL OTHER BUILDING AREA		[illegible]

GRAND TOTAL AREA C.E.P.#2 1,384,565 SQ FT.

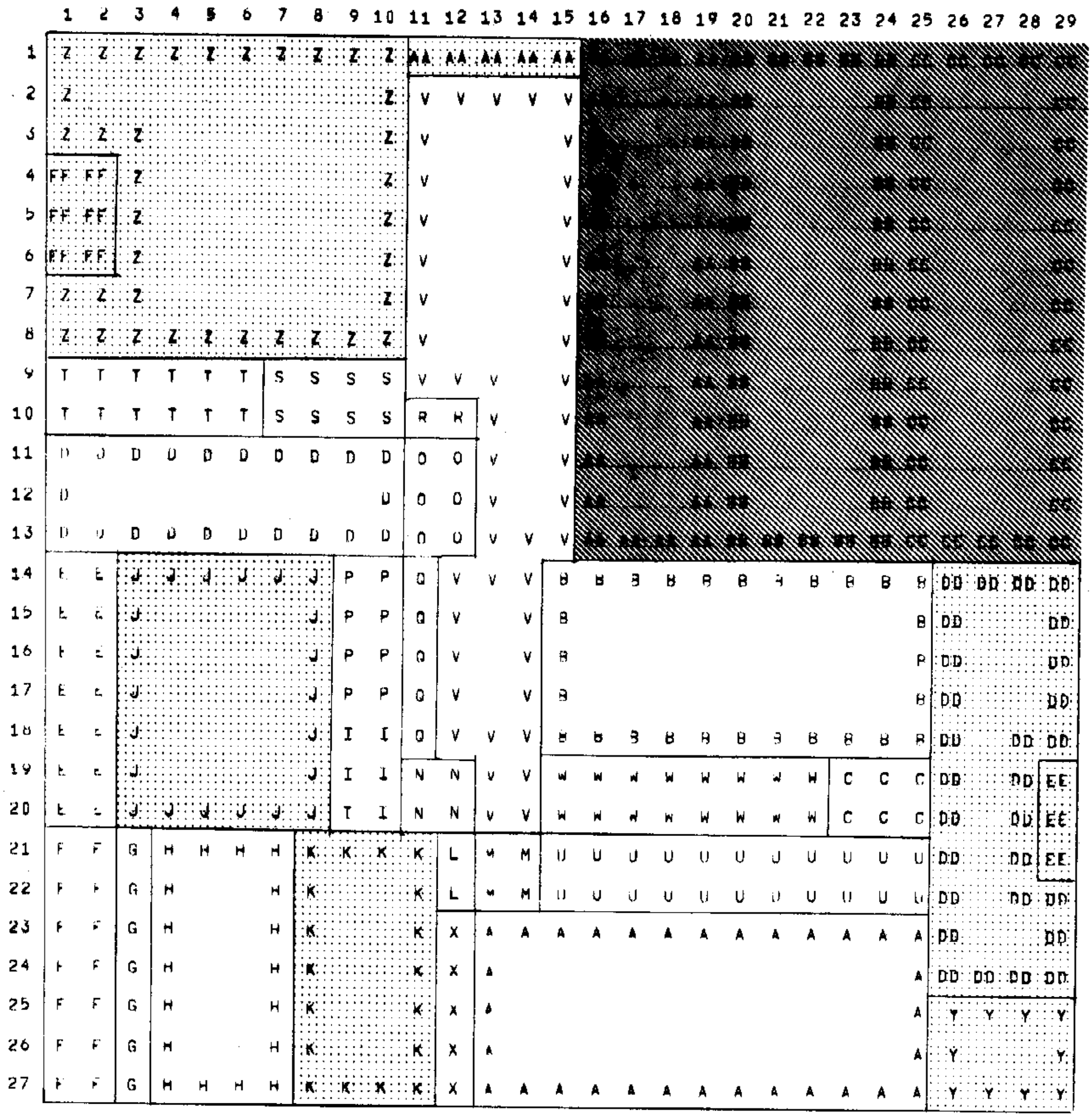
FIGURE #10

CLEVELAND ENGINE PLANT NO. 2

COMPUTER DEPARTMENT DESIGNATION (30 x 30 GRID)

A	335	Cylinder Block
B	335	Cylinder Head
C	335	Bearing Cap
D	335	Crankshaft
E	335	Camshaft
F	335	Int. Manifold
G	335	Exhaust Manifold
H	335	Rod, Cap and Assembly
I		Oil Pump
J		Race Engine and Heavy Duty Truck
K		Fly. and Housing/ Ring Gear
L		Storage
M		Storage
N		Clutch
O		Tool Room
P	"FE"	Int. Manifold
Q	352, 390	Manifold
R		Cafeteria
S		Cutter-Grind
T		Small Parts Machining
U		Piston
V		Assembly Line
W		Maintenance
X	351	Intake Manifold
Y		Piston Pin
Z		Engine Final and Shipping
AA		Dummy Areas
BB		Dummy Areas
CC		Dummy Areas
DD		Storage
EE		Truck Receiving
FF		Truck Shipping

LOCATION PATTERN



CLEVELAND ENGINE PLANT NO. 2

(30 X 30 GRID)

INITIAL INPUT LAYOUT

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

FIGURE #11

CRAFT-II

To improve the quality of the layouts produced by the program, CRAFT-II was developed. The major differences between this program version and CRAFT-I were as follows:

	CRAFT-II	CRAFT-I
1. Maximum layout representation	40 x 40 grid	30 x 30 grid
2. Maximum department limit	45	40
3. Maximum number of grid squares permitted in any one department	150	75

The grid was increased to allow more accurate representation of the initial plant layout. Changes (2) and (3) also allowed more accurate layout representation and expanded the capacity of CRAFT to handle larger problems.

In order to accomplish these problem capacity increases, all the error checking statements in the program that were related to the above parameters had to be altered. These occurred in almost every subroutine in the original program. Realizing that this procedure would have to be duplicated each time that a program limit was changed, a modification of the error checking system was instituted. Each parameter check was changed from a constant to a function of three variables: number of grid squares, maximum number of departments, and maximum department size. The variables were set in the main calling program and placed in common storage. This procedure allowed further modification of these parameters to be accomplished by simply resetting the three basic values and reallocating common storage. For further information refer to Appendix C.

While CRAFT-II was being developed, effort was also made to improve the rough cost estimates used in the initial runs. Mr. J. T. Carson performed analyses of accounting, industrial engineering and material handling data in order to more accurately describe the material handling costs. These costs are comprised of fixed and variable (distance related) components. CRAFT only considers the variable component. Considerable difficulty was encountered in obtaining reliable data that allowed an accurate assessment of the variable costs. In order to provide suitable layout problems for test purposes within a reasonable period of time, cost data were based upon the best information available plus judgment estimates.

CRAFT
September 1965
G. C. Armour
P. A. Cramer

CRAFT data deck
write up

GENERAL:

The user may insert as many complete data decks (four components per data deck - see below) as he pleases for any given run. Simply place the nth data deck behind the (n-1)th data deck.

A data deck consists of four components: Control card(s), volume, unit cost, and spatial information. These components must appear in the order listed within each data deck.

1. Control Card, Format (40I2)

The control card is used to describe limiting parameters of the problem presented by the remaining components of the data deck, and specify how the program is to utilize its various internal options in processing the problem.

Columns

1-2 Number of departments - max. 40.

3-4 Number of rows in the spatial configuration - max. 30.

5-6 Number of columns in the spatial configuration - max. 30.

7-8 Analyzer control
- 00 Two department moves only
- 01 Three department moves only
- 02 Two department moves followed by three department moves
- 03 Three department moves followed by two department moves
- 04 Choose best of two or three department moves at each iteration (recommended)

9-10 Input output control
- 00 Print first and last layout
- 01 Print first layout, and for each iteration the most favorable layout found during the iteration.

11-12 Debugging parameter
- 00 No failure messages (recommended)
- 01 Write Exchange failure and no cost reduction messages
- 02 Same as above, but also prints results of search for best move.

13-14 Number of departments to be fixed in place.

CRAFT data deck
write up

Columns

15-16 Department number of first department to be fixed in place.

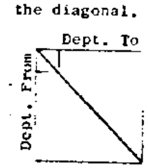
17-18 Department number of second department to be fixed in place.

19-80 Same as above, 2 columns for every additional department to be fixed in place. A second card may be used if necessary starting in columns 1-2.

2. Volume array, format (20F4.0)

Volumes are punched for every interdepartment relationship. Each department (row of the volume matrix (department from)) is started on a new card. Twenty paired department volume relationships may be punched per card. (e.g., for the first card punches in cols. 9-12 would input the volume flowing from department 1 to department 3.)

If more than 20 departments are specified, use additional cards. Volumes are treated uniquely on both sides of the diagonal in order that volumes in different directions can be represented at different travel costs. (See unit cost array below) If travel cost is the same for both directions of flow, place 1/2 of the volume on each side of the diagonal.



(Symmetric row and column headings)

There should be as many sets of cards for the array as is indicated in columns 1-2 of the control card. There should be the same number of fields utilized per set as is indicated in columns 1-2 of the control card. There is one card per set if the number of departments (col. 1-2 of control card) is 20 or less. There are two cards per set if the number of departments (col. 1-2 of control card) is greater than 20 and less than 41.

The decimal point is not punched. A decimal point will be considered to immediately follow the right most digit of each field for purposes of computation.

CRAFT-III

With the modifications instituted in CRAFT-II, almost all of core storage had been utilized. Additional core storage was therefore required before further program changes could be made. This section describes the method by which this was accomplished. The method, incorporated along with several input/output changes, was the basis for a new program version called CRAFT-III.

A search for a practical core restructuring technique was instituted. The technique had to be efficient so as not to increase run time. It was also decided that the technique should be, if possible, primarily a reallocation device, having no effect on the CRAFT logic other than the freeing of core. The use of multiple core loads was found to be unsuitable because of the frequent interaction between the program's 25 subroutines. Reduction in array size likewise was not feasible since this would limit the capacity of the program.

It was finally decided that a technique similar to the "ragged tables" concept of SIMSCRIPT* could be used. With the ragged tables feature a variable array storage allocation is made by the program on the basis of need. The logic for the revision was tested on the GE-265 time-sharing computer and then was inserted into CRAFT. The ragged table feature released 4,400 words of computer memory, enabling program revisions to continue unhampered by space limitations. Program running time was not increased. Details of the modification are included in Appendix C.

---

*See the manual SIMSCRIPT, A Simulation Programming Language by Markowitz, Hausner, and Karr, page 123, for a description of ragged tables.

Other CRAFT-III Modifications

Along with the ragged tables feature, several input/output changes were incorporated in CRAFT-III. The input cards representing the necessary flow and cost data were unwieldly because the matrices were large but sparse. In addition, the existing card format necessitated considerable rounding of cost per unit distance factors. As well, a specific card order was required.

All three of these limitations were overcome by the addition of a second, optional input method. This new method has no specific card order and reduces total card input for sparse matrices by approximately 50%. This new procedure also provides the ability to change specific input matrix elements with relative ease. Refer to Appendix C for a more detailed description.

Mr. Carson suggested that the computer should collect specific layout information during the run and print a summary of the data at the end of the final iteration. The result of this suggestion was a new subroutine SMRY which lists on one page the fixed departments and information concerning each CRAFT iteration: variable material handling cost, cost improvement, departments that have moved, and other pertinent data. A detailed description of SMRY is included in Appendix C.

Test runs were made with Cleveland Plant data to verify the CRAFT-III modifications. No documentation is included for these runs since they did not affect layout generation.

CRAFT-IV

CRAFT-IV embodied the first two additions to program heuristics. The additions were designed to: (1) prevent fixed departments from blocking desirable departmental exchanges; (2) improve still further the rectangularity of final departmental shapes.

The first feature permits a switch between two approximately equal departments. The reason for including this feature was that, at least in the early design phase, layout estimates of department sizes are often approximate. This meant that there was good justification for allowing departments of approximately equal size to exchange rather than restricting exchanges only to departments exactly equal in size. The new feature allows the user to specify a percentage range (PCNT) within which the sizes of unfixed departments can vary while still allowing exchanges. Appendices A and C provide further details.

Computational experience so far has indicated that the feature is of value when fixed departmental areas block the exchange of other departments. Running time was increased but is still less than three minutes for the largest problem tested.

A second feature was added to CRAFT to inhibit moves which result in low actual cost savings. As stated earlier, CRAFT predicts expected savings, selects the combination of departments which will result in the largest expected savings, and then performs this exchange. The exchange is rescinded only if the actual cost savings proves to be negative. This can lead to an exchange of departments with little savings but with a marked change in department shapes, particularly if the exchanged departments vary considerably in size. With the new program feature, exchanges are not carried out unless the actual savings are greater than a user-specified variable called EPSLN. This feature has resulted in better department switches in earlier iterations, thereby retaining original shapes and forcing major cost savings to occur earlier. Additional information concerning EPSLN can be found in Appendix C.

TEST RESULTS

The program modifications described in the preceeding sections culminated in the CRAFT-IV program. CRAFT-IV was then tested using a series of actual plant layout problems. The tests are discussed in the material that follows.

Cleveland Engine Plant

The Cleveland Plant test data was for a major expansion of an existing facility (Plant #2). The building outline took the form of an "L" shape with the northwest portion constituting the expansion area (Figures #10 and #13). Initial restrictions were:

1. Shipping and receiving docks and certain production facilities in the existing building were to be fixed in place.
2. The assembly line and its associated storage areas were to remain close to the engine hot test facilities.
3. The engine hot test facilities were to be fixed within the expansion area due to railroad access restrictions imposed by the building dimensions and the plant site.

Results of Cleveland Proposal A - No Float

Figures #12 and #13 show the results of the test run entitled, Cleveland Proposal A - No Float. The cost figures shown on the SMRY table use projected weekly flows and must be multiplied by 50 to give yearly figures. Iteration 0 is the layout that Engine and Foundry Division presently is using for manufacturing planning. The yearly variable material handling cost can be broken down as follows:

(47.5%) $99,800/yr = conveyor capital cost amortized over a 10-year period.

(19.0%) 39,900/yr = yearly conveyor maintenance.

(33.5%) 69,700/yr = variable cost of trailer trains and fork trucks on a yearly basis.

(100.0%) $209,400/yr = total variable material handling cost for one year.

Two thirds of the variable material handling cost for this particular layout is conveyor capital and maintenance cost. Conveyor flow is limited to seven departments: A, B, C, D, H, T, and U. It can be concluded that the layout design should place a high priority on the relative positioning of these seven departments. An examination of the SMRY sheet shows that the first six iterations involve at least one of the seven conveyor departments. The remaining exchanges are split between conveyor and trailer trains. Iteration #16 was chosen by Mr. Carson as the best feasible layout. The layout was not completely suitable for manufacturing and required some manual editing by Mr. Carson. The manually edited layout was then used as input to CRAFT-IV for calculation of material handling costs which amounted to a yearly cost of $179,500:

(44.5%) $80,000/yr = conveyor capital cost amortized over a 10-year period.

(18.0%) 32,000/yr = conveyor maintenance

(37.5%) 67,500/yr = variable cost of trailer trains and fork trucks on a yearly basis.

(100.0%) $179,000/yr = total variable material handling cost.

---

*Appendix E presents cost breakdown details.

CLEVELAND ENGINE PLANT NO. 2

COMPUTER DEPARTMENT DESIGNATION

A	335	Cylinder Block	S	Small Parts Machining
B	335	Cylinder Head	T	Piston
C	335	Bearing Cap	U	Assembly Line
D	335	Crankshaft	V	Maintenance
E	335	Camshaft	W	Piston Pin
F	335	Int. Manifold	X	Hot Test and Ship.
G	335	Exhaust Manifold	Y	Storage
H	335	Rod, Cap, and Assembly	Z	Ring Gear Storage
I		Oil Pump	AA	Race Engine
J		Fly. and Housing	BB	H.D.T. Cylinder Block
K		Ring Gear	CC	H.D.T. Cylinder Head
L		Storage	DD	H.D.T. Misc. Machining
M		Clutch Housing	EE	Truck Receiving
N		Tool Room	FF	Truck Shipping
O		Storage	GG	Tie in Block
P	"FE"	Int. Manifold	HH	Fix 1
Q		Cafeteria	II	Fix 2
R		Cutter Gr.	JJ	Fix 3
			KK	Fix 4

MATERIAL HANDLING COST DATA

F.+F. CLEV ENG - 2 NO-FLOAT 40 X 40

PRSL - A

FIXED DEPARTMENTS = W X Y EE FF GG HH II JJ KK J K AA BB CC DD

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(1)	0	4187,07	0,	0,					
	1	4148,74	38,33	38,33	V	F	A		
	2	3964,74	183,99	222,33	V	T			
	3	3919,11	45,63	267,96	Q	N	D		
	4	3867,37	51,74	319,70	B	A			
	5	3865,61	1,76	321,46	Z	U	B		
	6	3830,79	34,82	356,28	U	H			
	7	3785,96	44,83	401,10	R	S			
	8	3754,75	31,21	432,31	P	Z			
	9	3675,03	79,73	512,04	Z	T			
	10	3656,00	19,03	531,07	M	R			
	11	3648,72	7,28	538,34	N	F			
	12	3637,70	11,02	549,37	N	F			
	13	3630,36	7,34	556,71	L	N			
	14	3629,48	0,88	557,59	M	V			
	15	3619,47	10,01	567,60	S	D			
(2)	16	3616,78	2,68	570,28	S	U			
	17	3601,15	15,63	585,91	I	B			
	18	3598,21	2,95	588,86	I	O	P		
	19	3597,14	1,06	589,92	O	P			
	20	3591,33	5,82	595,74	U	C			
	21	3591,22	0,11	595,85	O	P			

PERCENT IMPROVEMENT = 14.23

MATERIAL HANDLING COST DATA

PRSL - A

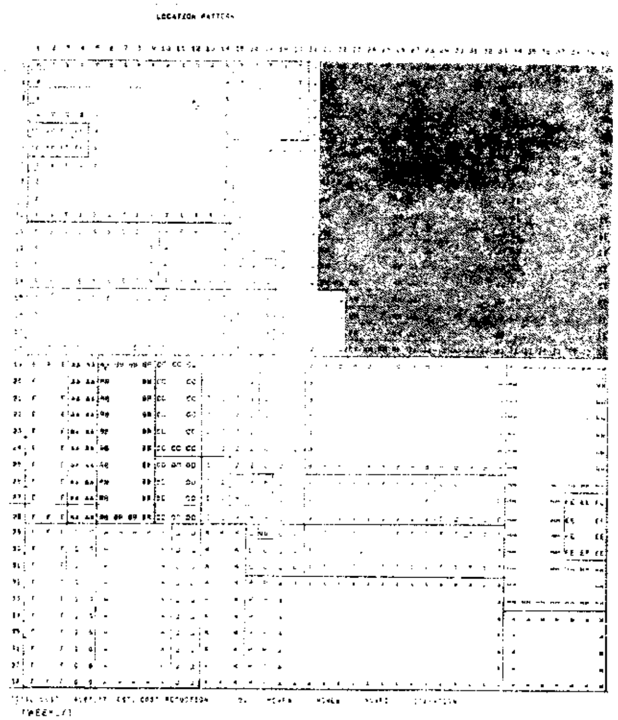
E.+F. CLEV ENG - 2 NO-FLOAT 40 X 40 REVISED

FIXED DEPARTMENTS = W X Y EE FF GG HH II JJ KK J K AA BB CC DD

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(3)	0	3589,57	0,	0,					
	1	3571,33	18,24	18,24	L	B			
	2	3557,38	13,95	32,19	N	Z			
	3	3554,69	2,69	34,88	L	P			
	4	3552,66	2,03	36,91	O	P			
	5	3551,09	1,58	38,49	I	L			
	6	3550,65	0,44	38,93	L	M			

PERCENT IMPROVEMENT = 1,08

FIGURE # 12



CLEVELAND ENGINE PLANT

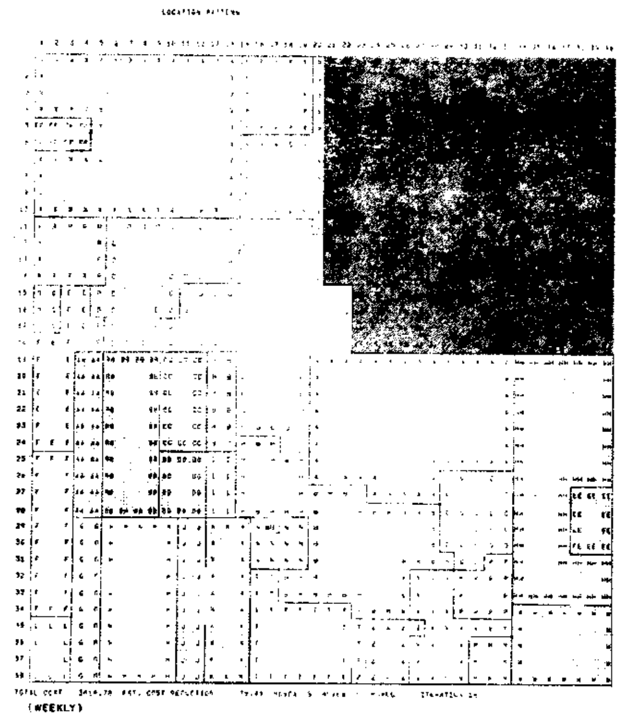
PROPOSAL A

① INITIAL INPUT LAYOUT

LEGEND

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS



CLEVELAND ENGINE PLANT

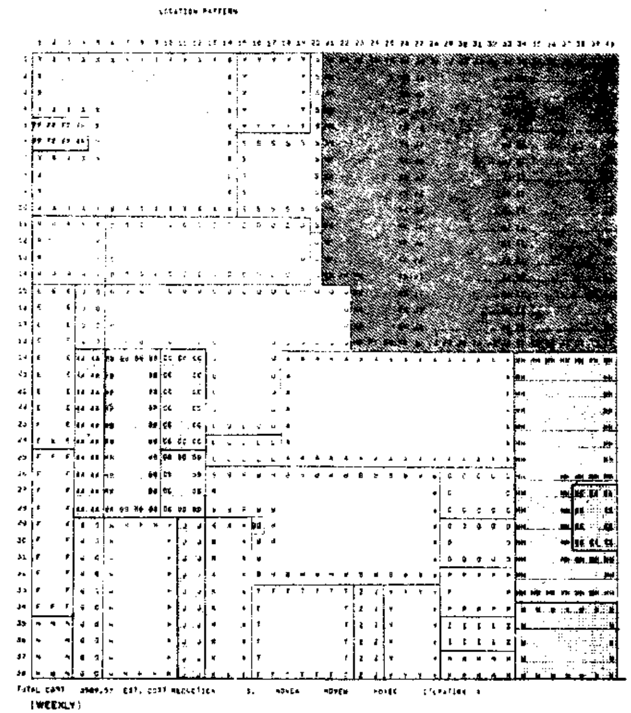
PROPOSAL A

② COMPUTER IMPROVED LAYOUT

LEGEND

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS



CLEVELAND ENGINE PLANT

PROPOSAL A

③ COMPUTER IMPROVED LAYOUT (MANUALLY EDITED)

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

FIGURE # 13

A cost comparison of the initial and final layouts:

ITER #0	Final Layout		Savings	
$99,800	- $80,000	=	$19,800/yr	conveyor capital
$39,900	- $32,000	=	$7,900/yr	conveyor maintenance
$69,700	- $67,500	=	$2,200/yr	trailer train and fork truck

The conveyor capital savings represents the purchase of less conveyor equipment (less length). The maintenance savings accrue from reduced conveyor length.

In the total savings calculations shown below, the 10-year life for conveyor systems is the approximation of equipment life used by the Company. The 3-year value shown for trailer train/fork truck systems is the minimum expected life for such equipments' material handling patterns.

$19,800/yr for 10-year life	=	$198,000
$7,900/yr for 10-year life	=	79,000
Total conveyor savings		$277,000
Trailer train and fork truck time savings, 3 years		$6,600
Total CRAFT savings		$283,600

It should be noted that the trailer train/fork truck savings listed above are only tentative estimates. At present it is safe to assume that if CRAFT reduces travel distance, there will be trailer trains/fork trucks savings. However, there are other factors beyond the geography of the plant layout that must be considered when determining equipment requirements. Many of these factors are not distance related: product characteristics, material handling equipment specifications, production schedules, et al. Further study is therefore required to determine the actual savings provided by CRAFT in this area.

For example, the \$2,200/yr savings cited above for trailer trains/fork trucks is based on a reduction in travel distance which in turn caused a proportionate reduction in such items as labor, maintenance, and depreciation. It might well be, however, that the CRAFT layout actually allowed one entire trailer train or fork truck to be eliminated; the annual savings for a three-shift operation would then be approximately \$35,000. There has been no opportunity to examine this latter possibility and the more conservative value, \$2,200, was therefore used.

Cleveland Proposal B - Full Float

Proposal B differs from Proposal A only in initial restrictions. The Heavy Duty Truck facilities area is not fixed, giving CRAFT greater flexibility (refer to shaded areas of Figure #15). The cost figures shown on the SMRY table use projected weekly flows as in Proposal A.

Proposals A and B use the same initial layout and material handling flows. Therefore, the cost of Proposal B Iteration #0 is identical to Proposal A Iteration #0. Iteration #22 was adjudged by Mr. Carson as the most reasonable layout. After manual editing by Mr. Carson, Iteration #22 had a yearly cost of $168,000:

(45.0%)	$76,000/yr	= conveyor capital cost amortized over a 10-year period.
(18.0%)	$30,500/yr	= conveyor maintenance.
(37.0%)	$61,500/yr	= variable cost of tractor trains and fork trucks on a yearly basis.
(100.0%)	$168,000/yr	= total variable material handling cost.

Using the same assumptions and procedures discussed in Proposal A, the total savings are as follows:

$238,000	conveyor capital avoidance.
94,000	10-year maintenance reduction for conveyors.
$332,000	total conveyor savings.
$ 24,600	trailer train/fork truck savings (3 years).
$356,600	total CRAFT savings.

The difference between Proposals A and B amounts to $73,000. This represents the cost penalty associated with fixing the heavy duty truck facilities in their present location. The $73,000 savings would not defray the loss in production and the cost of moving the facilities. Thus, management's original decision to fix this area was valid from a material handling standpoint.

MATERIAL HANDLING COST DATA

E.+F. CLEV ENG = 2 FULL FLOAT 40 X 40

PRSL-B

FIXED DEPARTMENTS = W X Y EE FF GG HH II JJ KK

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(1) →	0	4187,07	0,	0,					
	1	4148,74	38,33	38,33	V	T	A		
	2	3964,74	183,99	222,33	V	T			
	3	3919,11	45,63	267,96	G	N	D		
	4	3883,13	35,98	303,93	K	J	H		
	5	3831,39	51,74	355,67	B	A			
	6	3831,37	0,02	355,69	Z	U	B		
	7	3796,55	34,82	390,52	C	R			
	8	3757,84	38,71	429,22	P	H			
	9	3713,02	44,83	474,05	R	S			
	10	3682,05	30,97	505,02	P	Z			
	11	3655,27	26,77	531,80	Z	K	G		
	12	3594,77	60,50	592,30	G	H	I		
	13	3485,67	109,10	701,39	D	U			
	14	3440,17	45,50	746,89	CC	R			
	15	3403,73	36,45	783,34	M	T			
	16	3385,71	18,02	801,36	M	K	F		
	17	3334,18	51,53	852,89	S	D			
	18	3310,49	23,69	876,58	G	J			
	19	3290,34	20,15	896,73	CC	J			
	20	3288,90	1,44	898,17	DD	BB	F		
	21	3282,49	6,41	904,58	I	F			
(2) →	22	3252,02	30,47	935,05	DD	H			
	23	3251,11	0,92	935,96	N	AA	BB		
	24	3250,86	0,25	936,21	DD	F	G		
	25	3239,94	10,91	947,12	N	AA			
	26	3236,66	3,28	950,40	U	J			
	27	3228,74	7,92	958,32	N	O			
	28	3224,12	4,62	962,94	L	K			
	29	3221,37	2,76	965,70	I	K			
	30	3215,57	5,80	971,50	P	J			
	31	3213,95	1,62	973,11	CC	V			
	32	3212,97	0,98	974,09	L	CC			
	33	3210,00	2,97	977,06	CC	V	G		
	34	3208,73	1,27	978,33	CC	V			
	35	3208,66	0,08	978,41	L	CC			
	36	3207,64	1,01	979,42	P	C			

PERCENT IMPROVEMENT = 23,39

MATERIAL HANDLING COST DATA

PRSL-B

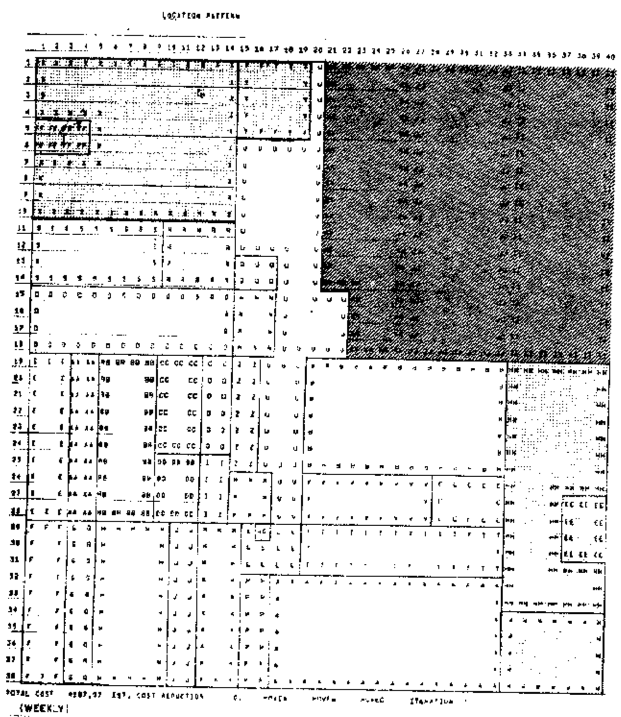
E.+F. CLEV ENG = 2 FULL FLOAT 40 X 40 REVISED

FIXED DEPARTMENTS = W X Y EE FF GG HH II JJ KK

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(3) →	0	3361,25	0,	0,					
	1	3352,63	8,62	8,62	C	F			
	2	3335,27	17,35	25,98	L	A			
	3	3320,34	14,93	40,91	K	V			
	4	3314,81	5,54	46,44	O	J			
	5	3307,53	7,28	53,72	N	E			
	6	3305,14	2,39	56,11	I	G			
	7	3301,36	3,78	59,89	N	DD			
	8	3296,36	5,00	64,89	P	J			

PERCENT IMPROVEMENT = 1,93

FIGURE #14



CLEVELAND ENGINE PLANT

PROPOSAL B

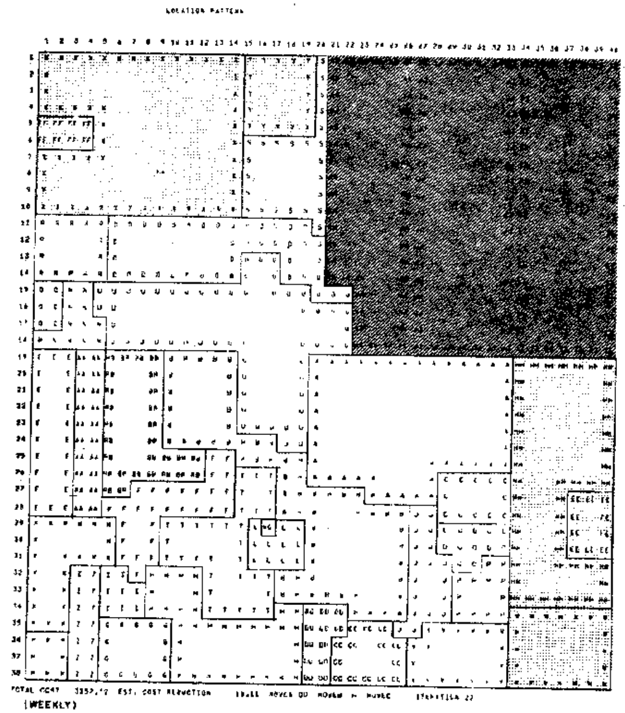
(1) INITIAL INPUT LAYOUT

LEGEND

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

FIGURE #15



CLEVELAND ENGINE PLANT

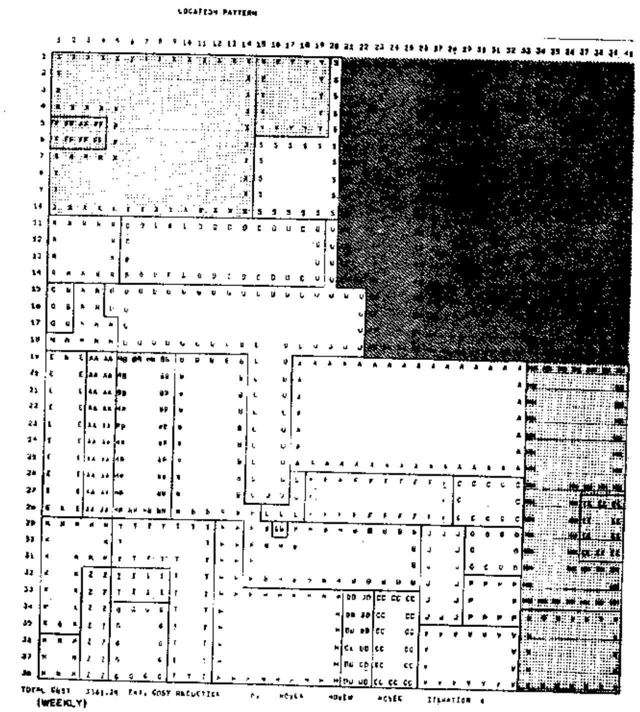
PROPOSAL B

(2) COMPUTER IMPROVED LAYOUT

LEGEND

FIXED DEPARTMENT

"DUMMY" DEPARTMENTS



CLEVELAND ENGINE PLANT

PROPOSAL B

(3) COMPUTER IMPROVED LAYOUT (MANUALLY EDITED)

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

Sterling-Van Dyke Chassis Plant

This is a newly constructed plant from which one of the major product lines was removed at a late stage of planning. The areas that were designated for the missing product line (there were five such areas) were left empty to be utilized at a later date.

The existing production areas do not interact as much as those in the Cleveland Plant; some are almost self-contained. Conveyor lengths are considerably less than at Cleveland and are concentrated in one area of the plant. Trailer train/fork truck cost is therefore more significant in this plant.

Sterling Proposal #1

Proposal #1 recreated the situation immediately following the decision to remove the major product line. The only departmental areas that were fixed were the dummy areas needed to satisfy the rectangular input requirements of CRAFT. Only the flows between production departments were considered. Office facilities and non-production related areas were treated as non-fixed, zero flow departments. (The office areas of the Cleveland plant were located outside the production areas of the plant and therefore were not a part of the CRAFT tests.)

The cost data used for the Sterling runs are yearly variable material handling costs. The conveyor costs are amortized over a 10-year period and are included within the yearly cost.

Sterling-Van Dyke Results - Proposal #1

Iteration #0 (the initial input layout) has yearly variable material handling costs of $134,200 (see Figures #16 and #17):

(16.5%)	$22,000/yr	= conveyor capital cost amortized over a 10-year period.
(8.5%)	$8,800/yr	= yearly conveyor maintenance.
(77.0%)	$103,400/yr	= variable cost of trailer trains and fork trucks on a yearly basis.
(100.0%)	$134,200/yr	= total variable material handling cost for one year.

CRAFT-IV produced 20 iterations with the 19th iteration selected as the best layout. Iteration #19 had one particular flow: the shipping and receiving docks were located in the center area of the layout, rather than along an outside wall. Using the five blank (unused) areas of the plant, Mr. Carson was able to rearrange Iteration #19 into an acceptable production layout (the interdepartmental relationships were retained as much as possible). This new layout was used as input for an additional run which provided the following variable material handling cost:

(17.5%)	$19,500/yr	= yearly capital conveyor cost amortized over a 10-year period.
(7.0%)	$7,800/yr	= yearly conveyor maintenance.
(75.5%)	$84,000/yr	= trailer train and fork truck variable cost.
(100.0%)	$111,300/yr	= total variable material handling cost.

When this layout is compared with the initial layout, the following conveyor savings result:

$25,000	conveyor capital avoidance
10,000	10-year conveyor maintenance reduction
$35,000	total conveyor savings.

STERLING VAN DYKE PLANT

COMPUTER DEPARTMENT DESIGNATION

A	Steel Storage and Blanking
B	Arm Storage
C	Socket Storage
D	Maintenance
E	Arm Forming
F	Socket Forming
G	Maintenance
H	Small Parts Washing
I	Formed Arm Storage
J	Ford Arm Assembly
K	Light Vehicle Arm Assembly
L	Arm Packing
M	Purch. Parts
N	Service Parts Assembly
O	Vacant
P	Rail Shipping Dock
Q	Casting Storage
R	Spider Machining
S	Race Machining
T	Tube Mill
U	Weld Yoke
V	Slip Yoke
W	Heat Treat
X	Quality Control
Y	Interia Disc
Z	Spider Grinding
AA	Race Grinding
BB	Maintenance
CC	Tube Storage
DD	Tube and Dampener Assembly
EE	Yoke Sub-Assembly
FF	Driveshaft Assembly
GG	Tool Room
HH	General Stores
II	Service Packaging
JJ	Purchased Parts
KK	Non-Production
LL	Vacant
MM	Vacant
NN	Fixed
OO	Vacant
PP	Receiving Dock
QQ	Vacant
RR	Vacant
SS	Fixed

MATERIAL HANDLING COST DATA

STERLING VAN DYKE CHASSIS PLANT **PRSL - 1**

FIXED DEPARTMENTS = NN SS

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(1) →	0	134236,37	0,	0,					
	1	130601,60	3634,77	3634,77	G	E			
	2	126606,54	3995,05	7629,82	RR	GG	P		
	3	121411,90	5194,64	12824,46	N	L			
	4	119957,70	1454,20	14278,67	Y	T			
	5	117450,54	2507,16	16785,83	HH	JJ	PP		
	6	114862,06	2588,48	19374,31	QQ	P			
	7	114526,31	335,75	19710,06	II	JJ			
	8	113243,84	1282,47	20992,53	RR	T			
	9	111447,50	1796,33	22788,86	BB	S	G		
	10	110459,47	988,04	23776,90	AA	FF			
	11	110243,78	215,69	23992,59	JJ	KK	GG		
	12	108952,86	1290,92	25283,51	II	GG			
	13	108285,52	667,34	25950,85	D	H			
	14	108020,09	265,43	26216,28	U	V			
	15	106409,65	1610,44	27826,72	U	T			
	16	106068,69	340,96	28167,68	HH	GG			
	17	105282,57	786,12	28953,80	II	FF	Z		
	18	105206,06	76,51	29030,31	S	X			
(2) →	19	105089,28	116,78	29147,09	S	D			
	20	105079,20	10,08	29157,17	D	OO			

PERCENT IMPROVEMENT = 21,72

MATERIAL HANDLING COST DATA

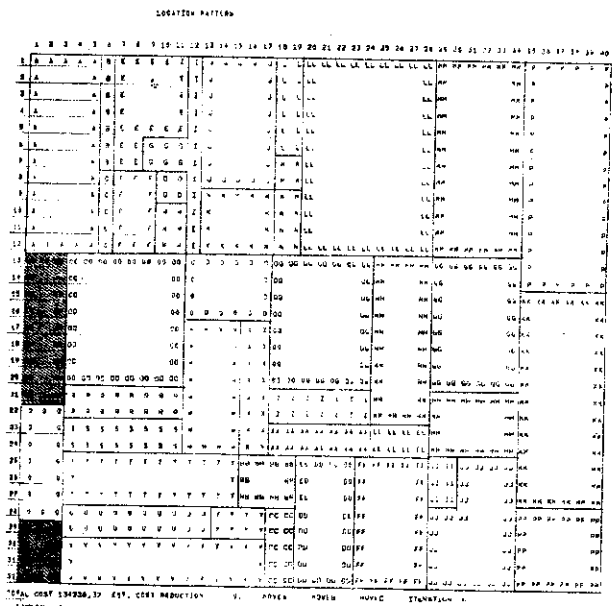
STERLING VAN DYKE CHASSIS PLANT REV PRSL - 1

FIXED DEPARTMENTS = A P Q W NN PP SS

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(3) →	0	111287,92	0,	0,					
	1	111042,80	245,12	245,12	H	L			
	2	110328,01	714,79	959,91	U	T			
	3	109945,57	382,44	1342,35	Z	EE	JJ		
	4	109574,70	370,87	1713,22	QQ	KK			
	5	109358,64	216,07	1929,28	AA	HH			
	6	109030,44	328,20	2257,48	U	X			
	7	108866,34	164,10	2421,58	X	AA			
	8	108856,26	10,08	2431,66	D	OO			

PERCENT IMPROVEMENT = 2,19

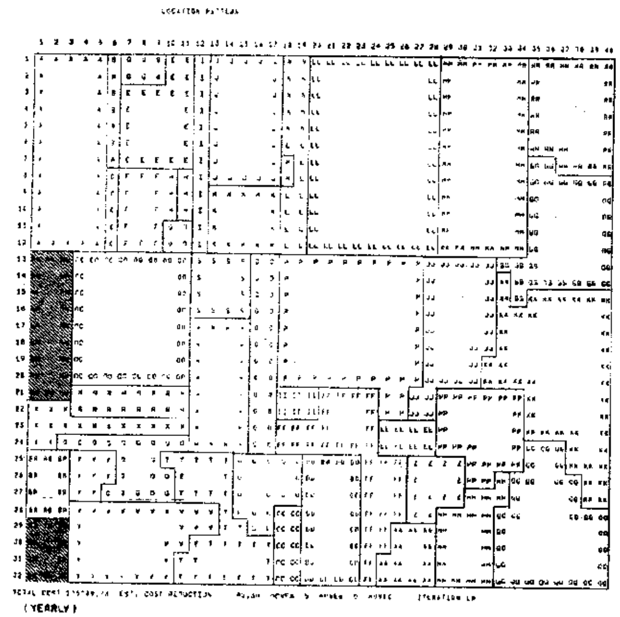
FIGURE # 16



STERLING VAN DYKE PLANT
PROPOSAL NO. 1

(1) INITIAL INPUT LAYOUT

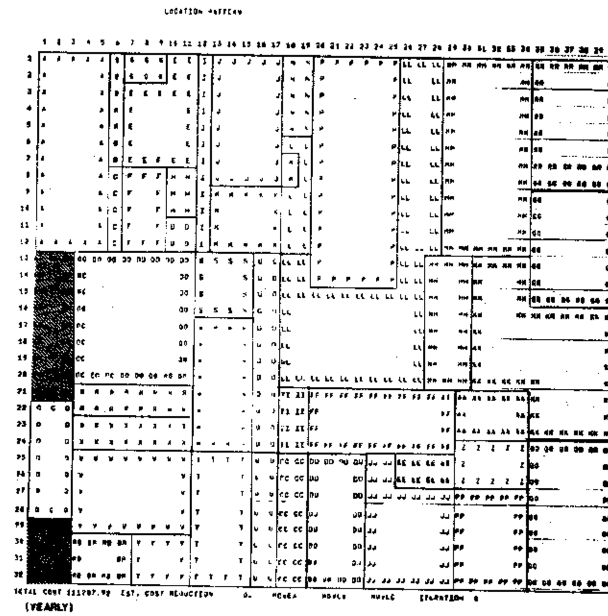
LEGEND:
FIXED DEPARTMENTS
"DUMMY" DEPARTMENTS



STERLING VAN DYKE PLANT
PROPOSAL NO. 1

(2) COMPUTER IMPROVED LAYOUT

LEGEND:
FIXED DEPARTMENTS
"DUMMY" DEPARTMENTS



STERLING VAN DYKE PLANT
PROPOSAL NO. 1

(3) COMPUTER IMPROVED LAYOUT (MANUALLY EDITED)

LEGEND:
FIXED DEPARTMENTS
"DUMMY" DEPARTMENTS

FIGURE 17

The trailer train and fork truck savings amount to $19,400 per year as contrasted with the values of $2,200 and $8,200 per year obtained for the two Cleveland proposals. (The possibility of trailer train/fork truck reduction would seem greater at Sterling than at Cleveland.)

The total cost savings for proposal #1, Sterling-Van Dyke Chassis plant:

$35,000	conveyor savings
58,200	trailer train savings for 3-year period
$93,200	total savings

Sterling-Van Dyke Results - Proposal #5*

Proposal #5 used the same flows and initial layout as proposal #1 but with these areas fixed: the shipping and receiving docks, the hot test area, and the five unused areas mentioned previously. The cost for iteration #0 was the same as for proposal #1: $134,200/yr. (see Figures #18 and #19)

CRAFT-IV produced 15 iterations and Mr. Carson selected the 15th as the basis for his revised layout. Because large portions of the plant had been fixed throughout the run, the final layout was much closer to an acceptable layout than Iteration #19 of proposal #1. The following cost breakdown was obtained for the manually edited Iteration #15.

(16.0%)	$19,500/yr	= yearly capital conveyor cost amortized over a 10-year period.
(6.5%)	$7,800/yr	= yearly conveyor maintenance.
(77.5%)	$93,500/yr	= trailer train and fork truck variable cost.
(100.0%)	$120,800/yr	= total variable material handling cost.

Both proposals #1 and #5 provided the same conveyor savings. This occurred because CRAFT gave conveyor length reduction an early priority (high cost) and because the conveyor facilities were concentrated in one area. CRAFT then proceeded to rearrange the rest of the plant. The fixed areas limited rearrangement possibilities, fewer iterations and smaller savings resulted.

The following conveyor savings resulted from proposal #5:

$25,000	conveyor capital avoidance.
10,000	10-year conveyor maintenance reduction.
$35,000	total conveyor savings.

*Proposals #2 through #4 were used only for program testing purposes and therefore are not described.

MATERIAL HANDLING COST DATA

STERLING VAN DYKE CHASSIS PLANT **#5**

FIXED DEPARTMENTS = A P O NN SS O LL MM OO QQ RR P PP

ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
0	134236,37	0,	0,					
1	130601,60	3634,77	3634,77	G	E			
2	129147,39	1454,20	5088,97	Y	T			
3	125995,53	3151,87	8240,84	N	L			
4	124714,07	1281,45	9522,29	HB	T			
5	124468,96	245,12	9767,41	M	L			
6	123801,62	667,34	10434,75	U	H			
7	123536,19	265,43	10700,18	U	V			
8	122421,02	1115,17	11815,34	U	T			
9	122116,27	304,76	12120,10	BB	V			
10	121400,87	715,40	12835,50	S	V			
11	121170,85	230,02	13065,52	S	X			
12	121072,39	98,46	13163,98	Z	AA			
13	121058,71	13,67	13177,66	II	HH			
14	120847,73	210,98	13388,64	Z	FF			
15	119869,30	978,43	14367,07	Z	FF	HH		

PERCENT IMPROVEMENT = 10,70

MATERIAL HANDLING COST DATA

STERLING VAN DYKE CHASSIS PLANT REV PRSL - 5

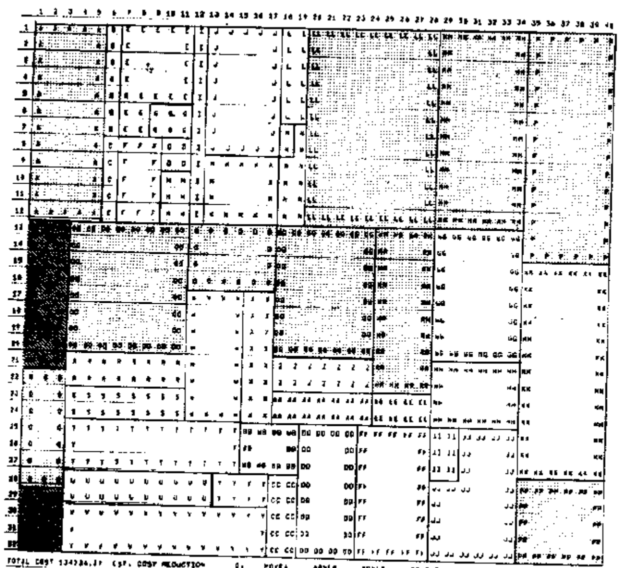
FIXED DEPARTMENTS = A P O NN SS O LL MM OO QQ RR P PP

ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
0	120810,81	0,	0,					
1	120559,15	251,66	251,66	U	T			
2	120367,76	191,38	443,05	Y	V	R		
3	120258,36	109,40	552,45	II	HH			
4	120018,94	239,43	791,87	Z	FF			
5	119784,51	234,43	1026,30	Z	AA			
6	119596,89	187,62	1213,92	Y	V			

PERCENT IMPROVEMENT = 1,00

FIGURE #18

LOCATION PATTERN



(YEARLY)

STERLING VAN DYKE PLANT

PROPOSAL NO. 5

① INITIAL INPUT LAYOUT

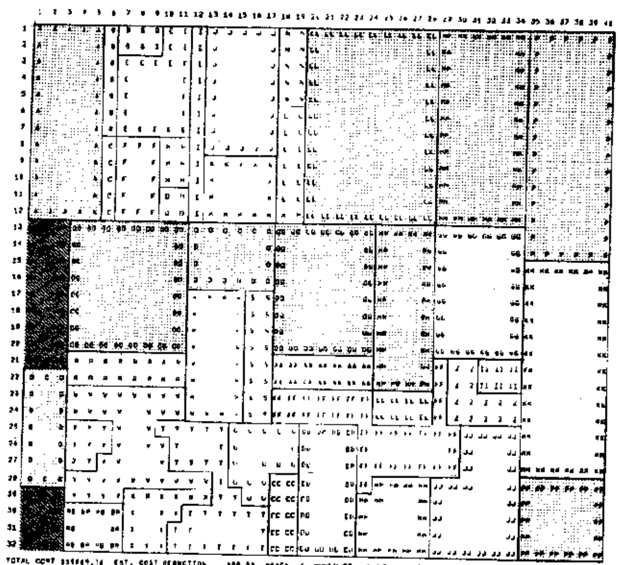
LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

FIGURE # 19

LOCATION PATTERN



(YEARLY)

STERLING VAN DYKE PLANT

PROPOSAL NO. 5

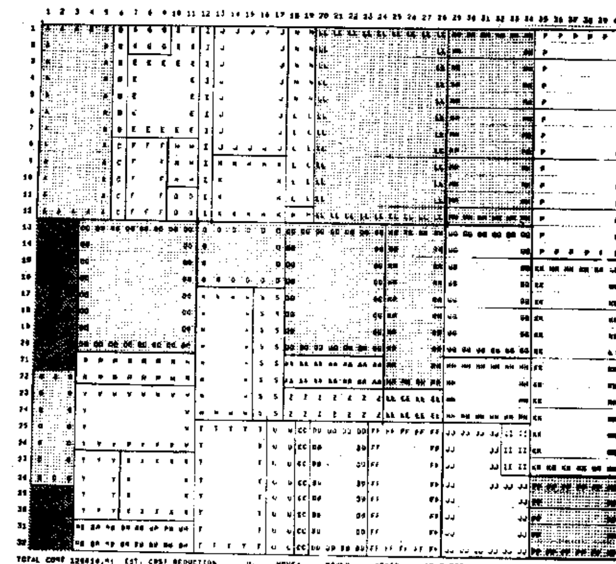
② COMPUTER IMPROVED LAYOUT

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

LOCATION PATTERN



(YEARLY)

STERLING VAN DYKE PLANT

PROPOSAL NO. 5

③ COMPUTER IMPROVED LAYOUT (MANUALLY EDITED)

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

Total savings for the Sterling-Van Dyke Chassis plant proposal #5:

$35,000	conveyor savings
29,700	trailer train savings for 3-year period
$64,700	total savings

Saline Plant

An expansion of the existing Saline facilities was planned for 1970. Two layout proposals were selected by General Parts Division for further study. CRAFT-IV runs were made for both proposals in order to develop comparative cost data. Flows unaffected by the expansion (i.e., product flows between existing facilities) were ignored. The results of the CRAFT-IV runs (see Figures #20 through #23):

	Proposal #1	Proposal #2
initial layout	$91,000/yr	$103,000/yr
final layout	$83,500/yr	$97,000/yr

These figures are variable material handling costs associated only with the expanded portion of the plant. Proposal #1 is cheaper and CRAFT-IV was able to suggest further improvement. A further breakdown of these figures was not made since only one conveyor was involved in the expansion and its portion of the total variable material handling costs was estimated as minor.

SALINE PLASTICS PLANT

COMPUTER DEPARTMENT DESIGNATION

A	Warehouse	O	Radiator Grille Stg.
B	Miscellaneous Stampings	P	Radiator Grille Stg.
C	Stamping Storage	Q	Radiator Grille
D	Stamping Receiving	R	Lamp & Body Stg.
E	Stamped Masks and Dials	S	Decorating
F	Fixed	T	Fender Apron Stg.
G	Odometer	U	Compr. Mold. Stg.
H	Lamp Sockets	V	Fender Apron Ship.
I	Printed Circuits	W	Compr. Mold
J	Fixed	X	Decorating Stg.
K	Fixed	Y	Fixed
L	Fixed	Z	Fixed
M	Receiving	AA	Fixed
N	Fender Apron	BB	Refuse Areas

MATERIAL HANDLING COST DATA

SALINE PROPOSAL NO, 1 11-1-67

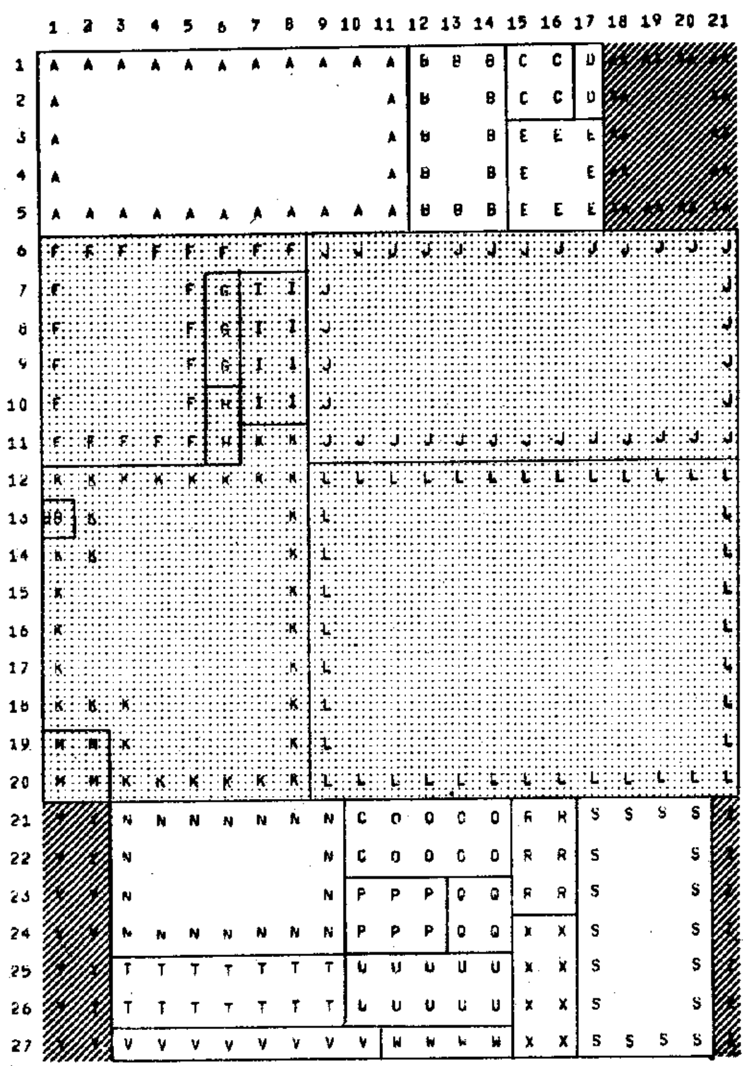
(PRESENT PLANNING)

FIXED DEPARTMENTS = A F G H I J K L M Y Z AA BB

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(1) →	0	90947,87	0,	0,					
	1	87985,59	2962,27	2962,27	R	S			
	2	86239,49	1746,10	4708,38	D	C	B		
	3	84782,55	1456,94	6165,32	D	B			
	4	84615,03	167,52	6332,84	X	S			
(2) →	5	83444,00	1171,03	7503,87	R	S	X		

PERCENT IMPROVEMENT = 8,25

FIGURE # 20



ITERATION 0

TOTAL COST 90947.87 EST. COST REDUCTION 0. MOVEA MOVEB MOVEC

(YEARLY)

SALINE PLANT

PROPOSAL NO. I

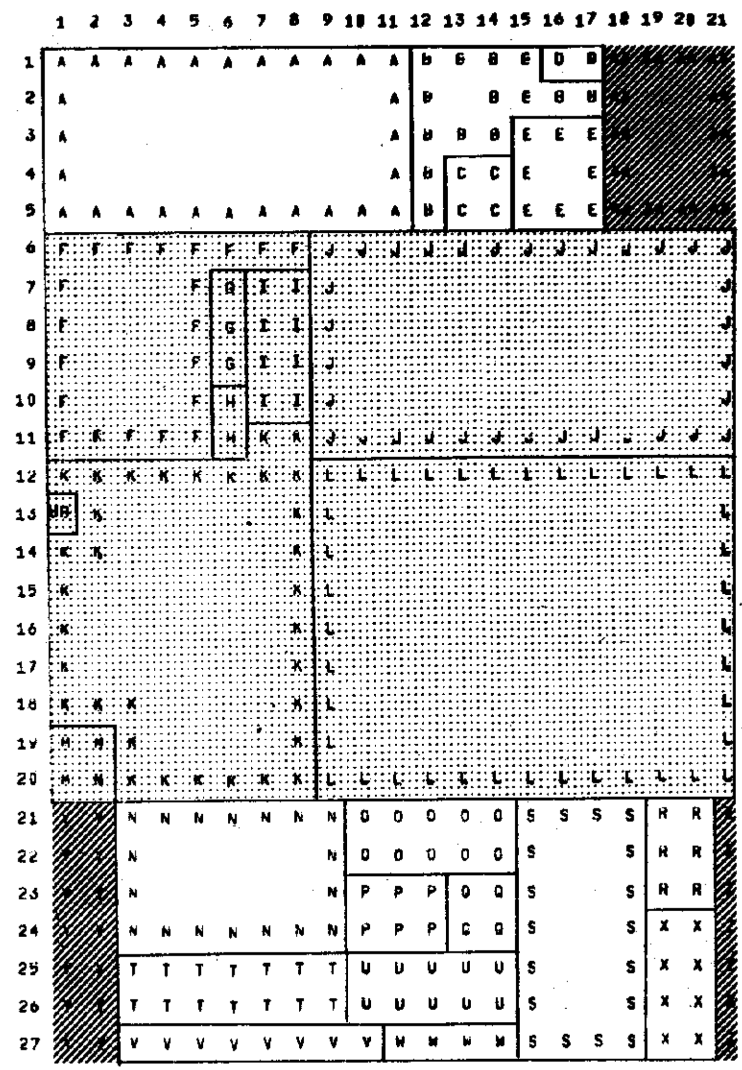
(1) INITIAL INPUT LAYOUT

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

FIGURE #21



ITERATI

TOTAL COST 83444.00 EST. COST REDUCTION 175.98 MOVEA R MOVEB S MOV

(YEARLY)

SALINE PLANT

PROPOSAL NO. I

(2) COMPUTER IMPROVED LAYOUT

LEGEND:

FIXED DEPARTMENTS

"DUMMY" DEPARTMENTS

MATERIAL HANDLING COST DATA

SALINE PROPOSAL NO. 2 11-1-67

(INITIAL PLANNING)

FIXED DEPARTMENTS = A F G H I J K L M Y Z AA BB

	ITERATION	TOTAL COST	IMPROVEMENT	CUM IMPRVMT	MOVEA	MOVEB	MOVEC	CHNGA	CHNGB
(1)	0	103559.13	0.	0.					
	1	100596.85	2962.27	2962.27	R	S			
	2	99682.24	914.61	3876.89	D	B			
	3	98461.04	1221.20	5098.08	E	B			
	4	98293.53	167.52	5265.60	X	S			
(2)	5	97122.50	1171.03	6436.63	R	S	X		

PERCENT IMPROVEMENT = 6.22

FIGURE # 22

RECOMMENDATIONS

The CRAFT technique, and in particular the CRAFT-IV version, is a useful plant layout tool but not a perfect one. Limitations of varying importance remain and are discussed in the following three sections devoted to: 1) program communication, 2) data problems, and 3) CRAFT program logic. Included in these sections are specific recommendations for additional program improvements as well as suggestions for useful areas for further research.

Program Communication

Several suggestions for improvement in communication to and from the CRAFT program are described below:

Modify the program so that multiple runs for a given layout problem could be made using only one complete data set plus specific change cards for all additional runs. The program presently requires complete data sets for all runs of a multiple run series.

CRAFT-IV allows use of a 40 x 40 grid representation. Providing a still finer grid size (say 60 x 60) might further improve the departmental rectangularity of CRAFT layouts; as previously mentioned, such effect was noted when the 40 x 40 grid was implemented. Means for obtaining the required additional core memory are discussed in a following section.

The computer printout of CRAFT layouts does not provide lines defining departmental boundaries, thus hindering visual interpretation of the layouts. To solve this problem, CRAFT output could be used as input to the Cal Comp drafting equipment now available within the Company.

The present CRAFT output lists total costs only and appreciable manual effort is therefore required to develop component cost breakdowns. There is no reason why the program should not perform this task.

A CRAFT layout requires analysis and manual editing by an experienced layout engineer. The manually revised layout is then used as input for another program run to obtain the impact upon costs of the editing changes. A time-share computer of sufficient size (the GE-265 is too small) with graphic display equipment would measurably shorten the time required for this kind of activity. (Quite likely, time-share interaction between an experienced layout engineer and CRAFT would also lead to development of better CRAFT heuristics.)

Data Problems

Among CRAFT's present limitations are several approximations used for convenience in representing material flow.

First, flows are considered to begin or terminate at the geographical center of a departmental area. This can cause an error in travel length calculations. For example, an assembly department is usually long and narrow; flows to and from the department occur at both ends and at intermittent points along the axis of the assembly line.

Second, CRAFT uses a single value for each given flow between departments. Such a flow, of course, actually varies through time with the production schedule.

Third, trailer train/fork truck movement is assumed to be a one-stop round trip cycle of some constant number of unit loads. For example, a train delivers three cars to some destination point and returns to its starting point with three empty cars. In actual practice the train might haul from one to four cars as it begins a trip, then drop off and pick up varying numbers of cars at several stops before returning to the start point.

It is possible to overcome the above limitations by careful planning during run preparation: manually editing and rerun can provide a final accurate portrayal of flow points. A range of product flow levels, and different train movement patterns, can be examined by using multiple CRAFT runs. At present is is believed that these are minor limitations of CRAFT but the user should be aware that they exist.

Prior reference has been made to the difficulty of establishing the actual savings that result when CRAFT reduces trailer train/fork truck travel distance. A possible solution to this problem would be a simulation program which determines the number of equipment units needed for a specific CRAFT layout. The program would recognize varying production schedules, actual destination points, unit load variances, et al. Such a simulator, essentially a material handling equipment fleet sizing and scheduling tool, could be a valuable aid for material handling personnel and could be used with any layout, CRAFT or manually generated.

Another problem concerned with the determination of cost savings is that of getting good estimates of the life of a layout and/or the life of the material handling equipment servicing the layout. The values used in this study (ten-year life for conveyor systems, three-year life for trailer trains/fork trucks material handling patterns) are subject to further refinement. For example, there is knowledge that fork truck material handling patterns can remain stable for as long as eight to ten years.

There may also be a need for refinement of some of the other material handling cost data. For example, it is presently assumed that a reduction in conveyor length results in a proportionate reduction in conveyor maintenance expense. Such may not be the case, however, if conveyor drive unit maintenance represents the major portion of total maintenance expense.

CRAFT Program Logic

CRAFT employs heuristic rules to reach a suboptimal solution to the problem of relative allocation of facilities. Because of these rules, CRAFT is path oriented: a given input data set always generates one unique set of results. Since CRAFT solutions are suboptimal, it would be desirable to explore several different solution paths. There are at least two ways by which this could be done: 1) run several different initial layouts of the same facility keeping the costs and product flows constant; 2) input some given initial layout having no fixed departments and use a large PCNT factor (25%).

Method number (2) will produce a layout possessing the desired inter-departmental material flow relationships but the department area allocations may be unsuitable. Manual editing will then provide a practical layout which can be used as input for a second CRAFT run. This procedure can lead to better solutions than when no PCNT or only a small PCNT value is used.

An expansion of the method for generating exchange candidates might be of value. The present exchange method considers only two department exchanges. The ability to make such exchanges as "3 for 3" or "2 for 1" would increase the list of exchange combinations and could therefore lead to larger cost savings. The programming is likely to be difficult.

Another desirable feature is an EPSLN factor that could vary with specific departments. This would allow layout changes to existing facilities to be based on a return on investment criterion. The feature would insure that a layout change is not made by CRAFT unless the resultant material handling savings can cover the costs of making the change and yield a profit.

An additional problem posed by CRAFT is that it considers only material handling costs when making layout decisions. There are other factors, however, to be considered in developing a layout. Some examples are:

1. health and safety considerations

2. supervisory control

3. shared service facilities, i.e., a central coolant system

These factors are difficult to quantify, but the work by Muther and Moore (CORELAP) has provided an approach. It seems desirable to examine the possibility of coupling CRAFT with a program like CORELAP. This would allow the generation of layouts that reflect both the easily quantified factors and the more intangible considerations.

The material handling equipment specified for an initial layout is held constant throughout a CRAFT run. However, the decision between use of trailer trains versus fork trucks in based on distance. Fork trucks are used for movements up to 300 feet and trailer trains for greater distances. It would therefore be desirable to have CRAFT recognize, for example, when it has reduced a trailer train route below 300 feet; the program would then automatically reclassify the route as one using fork trucks and subsequent layout alterations would then be based on fork truck costs. The problem can be handled presently by manually altering the CRAFT output and rerunning.

Relative to the suggestions above for additional changes to CRAFT, the modifications culminating in CRAFT-IV has left unused approximately 3,300 words of core memory. It is possible that additional memory space can be obtained by redesigning the storage and searching patterns of CRAFT. For example array IAJA (used by the program for internal storage) is 45 x 45 words in size and is symmetrical along its diagonal. If all references to this array were confined to the half above the diagonal, the lower half could be freed for other uses. The programming changes would be difficult but would result in over 1,000 additional words.

In addition CRAFT has four unpacked <u>integer</u> arrays (ISP, IT, IU and IV, each 42 x 42 words). These could be packed together into two overlaid arrays to save an additional 3,500 words of memory.

CONCLUSIONS

This report has described a study of the plant layout tool entitled Computerized Relative Allocation of Facilities Technique (CRAFT). After modifying the program and after conducting appropriate plant tests, the following is concluded:

1. CRAFT allows rapid evaluation of layout proposals for new facilities.
2. It suggests further improvements to original layout proposals.
3. It analyzes the effect of layout revisions and expansions to existing facilities.
4. It determines the cost impact of alternate modes of material handling equipment for a specific layout.
5. It quantifies portions of the layout decision-making process that have previously been handled by subjective judgment.

CRAFT despite its limitations has proved to be an effective means for producing good relative location patterns. The savings potential of the program is large enough to merit continued development and Company application.

APPENDIX A

REVISED CRAFT DATA DECK

General

A complete data deck is required for each problem, but the user may insert as many decks as desired in a given run. The nth data deck is placed behind the nth-1 deck.

Two alternate deck set-ups are available. Set-up number 1 is for a job shop or similar layout with many flow relationships:

1A - Title Card
2A - Control Card(s)
3A - Flow Matrix
4A - Cost per Unit Distance Matrix
5A - Interspatial Array Matrix
6A - Department List (optional)
7A - "END" Card

Set-up number 2 is for a more restricted flow pattern.

1B - Title Card
2B - Control Card(s)
3B - I, J Element List
4B - 9999 Card
5B - INTERSPATIAL ARRAY Matrix
6B - Department List (optional)
7B - "END" Card

Set-up number 2 allows more input flexibility and is recommended for all but problems with many interdepartmental flows.

1A and 1B - TITLE CARD FORMAT (10A6)

This card is the first card in any data set. It has a 60-column limitation and is used for problem identification. The project title will be reprinted on the summary sheet exactly as it appears on this card.

2A and 2B - FIRST CONTROL CARD FORMAT (8[I2, 1X], 2[F5.0, 1X])

The Control Card is used to describe limiting parameters of the problem. It also specifies program internal processing options.

Columns			Variable
1-2	Number of Departments	MAX 45	NDEPT
4-5	Number of rows in the spatial configuration	MAX 40	IROW
7-8	Number of columns in the spatial configuration	MAX 40	ICOL
10-11	Analyzer control 00 two department moves only 01 three department moves only 02 two department moves followed by three department moves 03 three department moves followed by two department moves 04 choose best of two or three department moves at each iteration (recommended)		ICTL
13-14	Input/output control 00 print first and last layout 01 print first layout and for each iteration the most favorable layout found during the iteration		IOCTL
16-17	Debugging parameter 00 no failure messages 01 write exchange failure and no cost reduction messages 02 same as above but also prints results of search for best move		ICHK
19-20	Number of departments to be fixed in place NOTE: the actual departments to be fixed are placed on the second control card IF(IFIX.EQ.0), omit the second control card		IFIX

SECOND CONTROL CARD FORMAT (40I2)

Columns		
1-2	Department number of first department to be fixed in place, right justified	IDFIX (I)
3-80	Same as above 2 columns for each additional department to be fixed in place	

2A and 2B - FIRST CONTROL CARD CONTINUED

Columns		Variable
22-23	00 or blank specified job set-up number one (refer to original CRAFT write-up) 02 specifies use of element list (refer to Appendix C for explanation of element list	IPTS
25-29	Percent of department size variability FORMAT (F 5.0) FOR EXAMPLE: 00.05 = 5% (refer to Appendix C for explanation of this feature)	PCNT
31-35	Cost limiting factor FORMAT (F 5.0) FOR EXAMPLE: 45.00 = $45.00, all actual exchanges must be greater than $45 (EPSLN) to be accepted (refer to Appendix C for further information)	EPSLN

3A FLOW MATRIX

Used in job-set up number one (refer to CRAFT write-up for details)

4A COST PER UNIT DISTANCE MATRIX

Used in job set-up number one (refer to CRAFT write-up for details)

3B - I, J ELEMENT LIST

Replaces - 3A and 4A in set-up number two

FORMAT (2[I2, 1X], 2[F5.0, 1X])

Variables: I, J, DST, CVL

I = Department from which product flow is emerging
J = Department destination of product flow emerging from I
DST = The quantity of flow from Department I to Department J
CVL = The cost to move this flow one grid square (i.e., cost per unit distance

EXAMPLE:

01, 05, 010.0, 00.20

Flow from department #1 to department #5 of 10 units, at a cost of $.20 a unit for each grid square moved.

Each card represents one from I to J move relationship. Every such non-zero relationship must be represented by one card. No specific order is required in this section of the data set except the requirement of a 9999 card (4B in deck set-up), to signal termination of this section.

4B - "9999" CARD

A card having the number 9999 in its first 4 columns, signals termination of I, J ELEMENT LIST

5A-5B INTER SPATIAL ARRAY MATRIX

This remains essentially the same as CRAFT-I except for an expansion of the grid from a maximum of 30 x 30 to 40 x 40 (refer to SHARE write up).

6A-6B DEPARTMENT LIST (OPTIONAL)

Any number of cards may be placed between the last card of section 5A-5B and the "END" card (7A or 7B). All 80 fields of each card may be used. The information is read under an Alphanumeric Format and reproduced at the end of SMRY exactly as it appears on the cards.

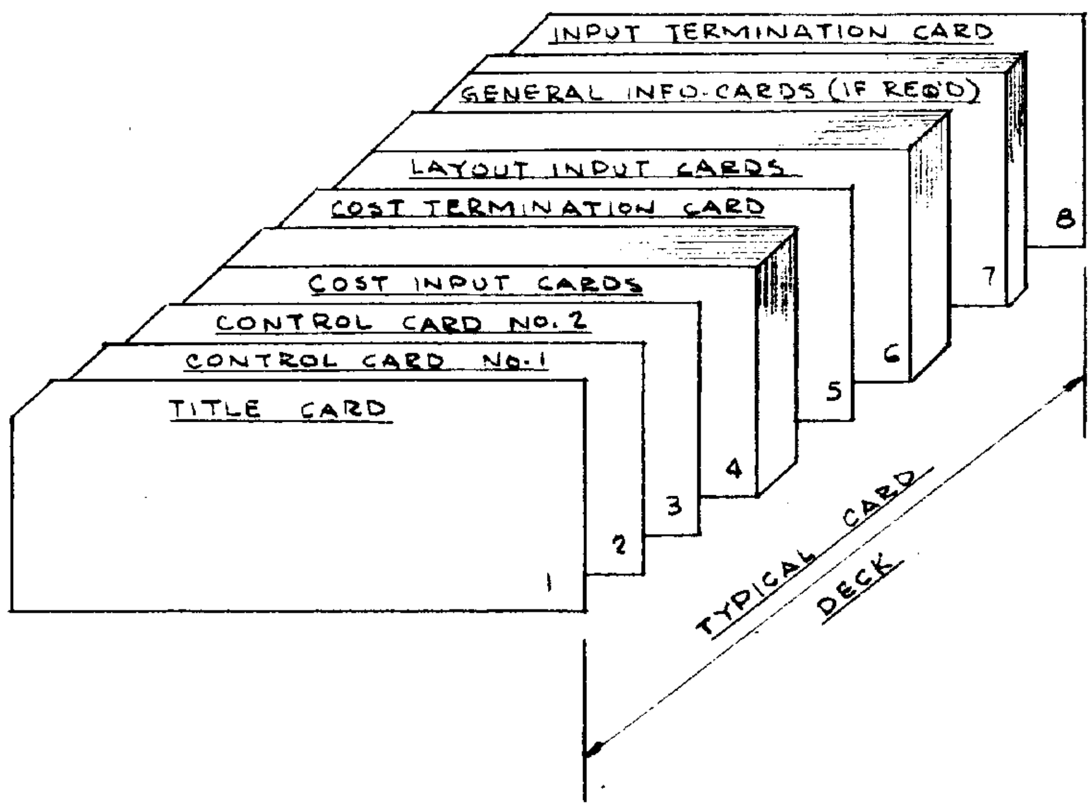
7A-7B "END" CARD

A card having the letters E, N, D in columns one, two, and three respectively, signals the termination of a data set.

APPENDIX B

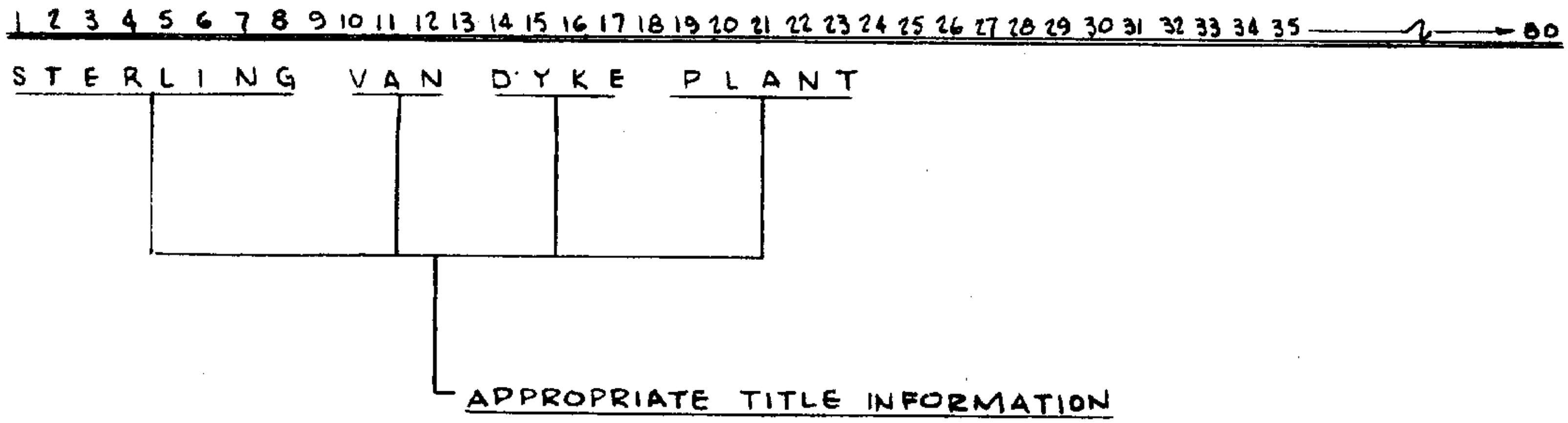
The following information is reproduced with the consent of Mr. James Carson, Material and Equipment Engineering Department, Manufacturing Staff. This material was prepared to assist plant-layout engineers in the deck set up of CRAFT-IV and serves as a supplement to Appendix A.

Input Card Sequence



TITLE CARD

TYPICAL ARRANGEMENT
FOR
I.B.M. CARD



NOTE: MAXIMUM OF 60 SPACES CAN BE USED.

CONTROL CARD No. 1

TYPICAL ARRANGEMENT FOR I.B.M. CARD

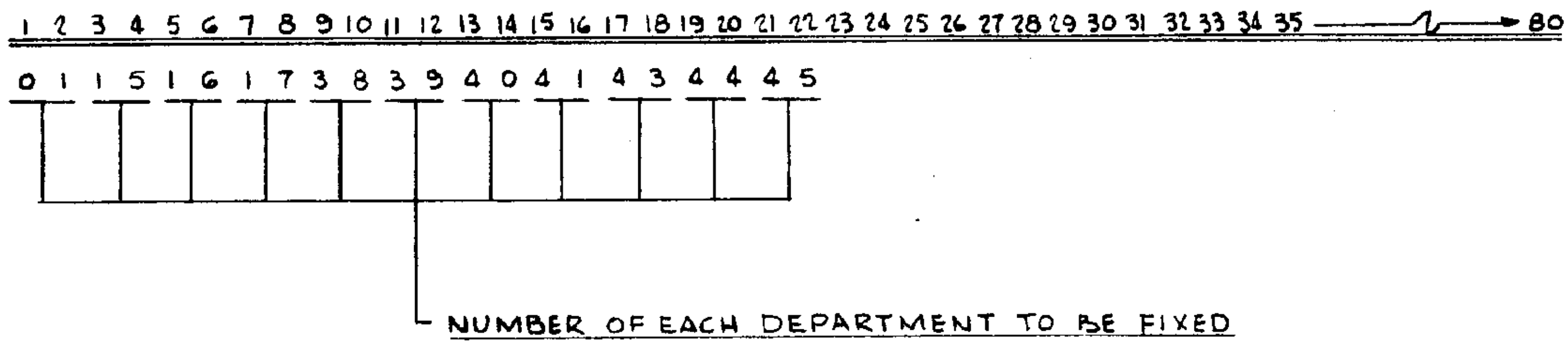
Card columns: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 — 80

45 | 32 | 40 | 04 | 01 | 00 | 11 | 02 | 00·05 | 05·00

- 05·00 (end) — NORMAL END OF CARD INPUT
- 05·00 — MIN. ALLOWABLE COST REDUCTION
- 00·05 — ACTUAL FRACTION VARIANCE ALLOWABLE FOR SWITCHING NON-ADJACENT DEPARTMENTS (DECIMAL NOT %)
- 02 — TYPE OF COST INFORMATION INPUT:
  - 00 – SEPARATE FREQUENCY & COST MATRIX
  - 02 – SINGLE POINT INPUT
- 11 — NO. OF DEPARTMENTS TO BE FIXED
- 00 — PRINT-OUT OF COST CALCULATIONS FOR EACH ITERATION:
  - 00 – NO TEST CALCULATIONS
  - 02 – ALL " "
- 01 — PRINT-OUT OF LAYOUT & RELATED INFORMATION:
  - 00 – FIRST & LAST LAYOUT
  - 01 – FIRST LAYOUT & ALL ITERATIONS
- 04 — TYPE OF DEPARTMENTAL REARRANGEMENT MOVES:
  - 00 TWO DEPTS. ONLY
  - 01 THREE " "
  - 02 TWO " ; THEN THREE DEPTS.
  - 03 THREE " ; " TWO "
  - 04 BEST OF TWO OR THREE DEPTS.
- 40 — NO. OF LAYOUT COLUMNS (MAX. 40)
- 32 — NO. OF LAYOUT ROWS (MAX. 40)
- 45 — NO. OF DEPARTMENTS (MAX. 45)

CONTROL CARD No. 2

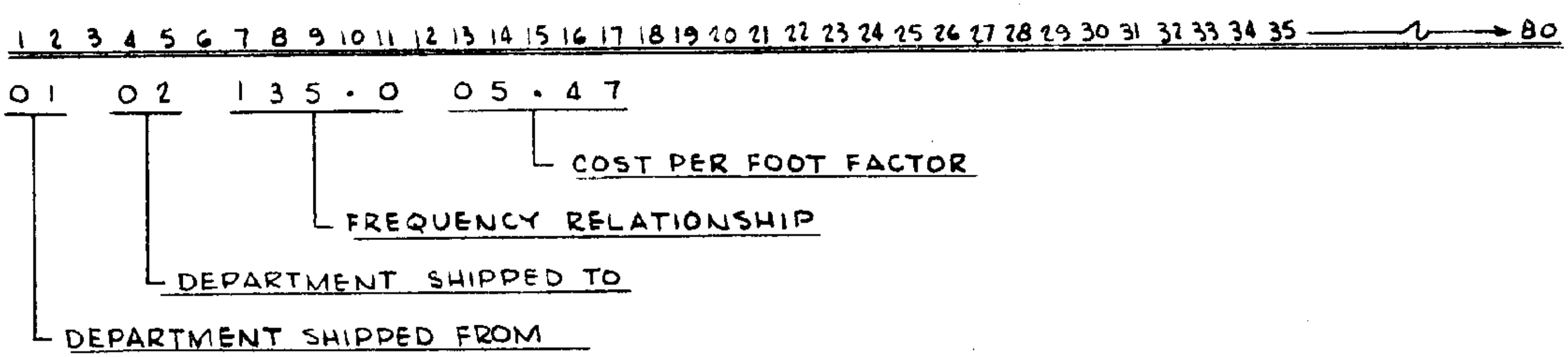
TYPICAL ARRANGEMENT FOR I.B.M. CARD



COST INPUT CARDS

(SINGLE POINT INPUT SYSTEM)

TYPICAL ARRANGEMENT
FOR
I.B.M. CARD



NOTE: SEPARATE CARDS MUST BE PREPARED FOR EACH NUMERICAL VALUE SHOWN ON INTER-DEPARTMENTAL RELATIONSHIP CHART.

Cost Termination Card

Typical Arrangement
for
I.B.M. Card

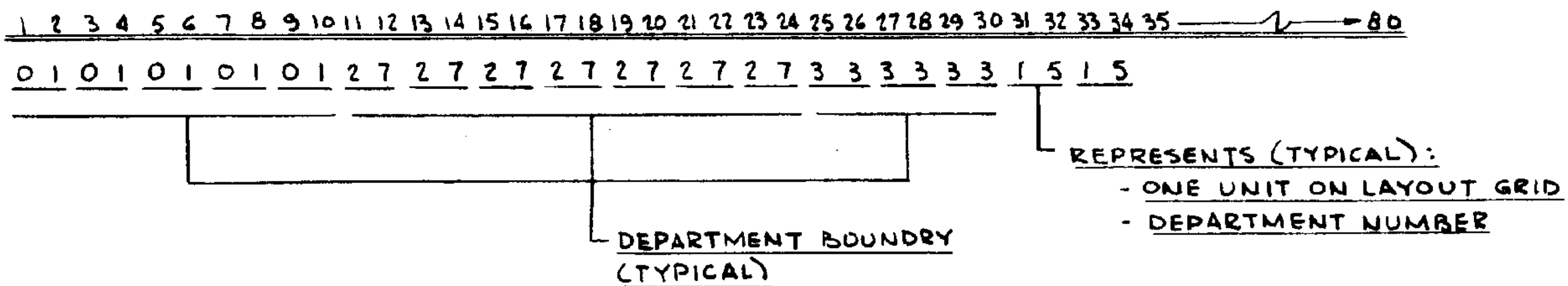
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 → 80

9 9 9 9

THIS NUMBER IS USED TO INDICATE ALL COST INFORMATION HAS BEEN READ BY COMPUTER

LAYOUT INPUT CARDS

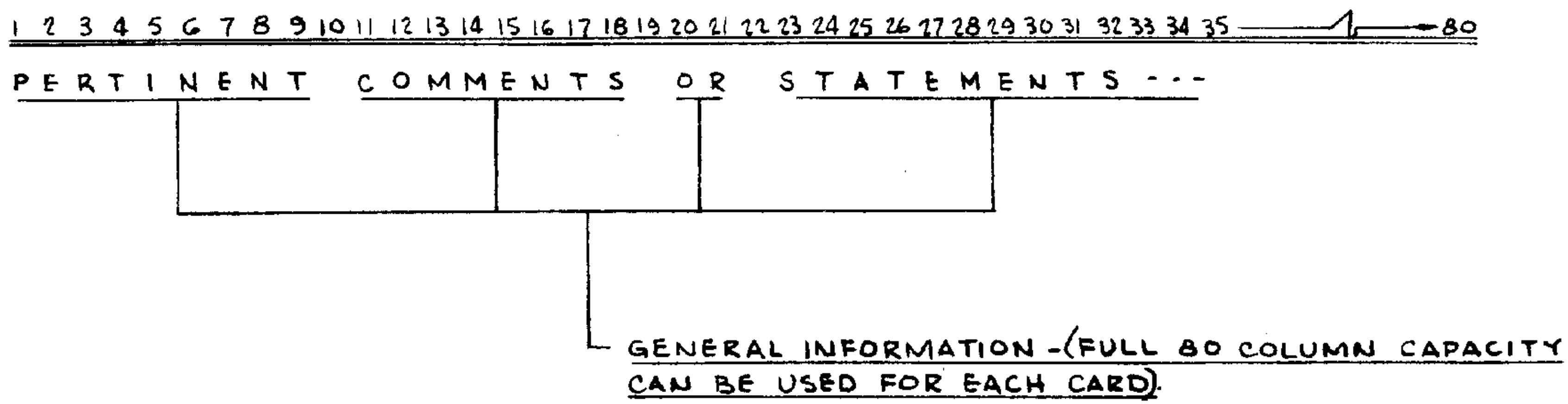
TYPICAL ARRANGEMENT
FOR
I.B.M. CARD



NOTE: SEPARATE CARD IS REQUIRED FOR EACH ROW OF LAYOUT GRID.

General Information Cards

Typical Arrangement
for
I.B.M. Card



INPUT TERMINATION CARD

TYPICAL ARRANGEMENT
FOR
I.B.M. CARD

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 ——— 80

E N D

THE WORD "END" IN THE FIRST THREE COLUMNS INDICATES THAT ALL INPUT INFORMATION HAS BEEN COMPLETED.

APPENDIX C

INTERNAL PROGRAM CHANGES

NOTE: This section assumes prior knowledge of CRAFT-I SHARE write up.

RAGGED TABLES

NOTE: An understanding of IDNDX as used in CRAFT-I is necessary before reading this section.

IDNDX is an array which stores the grid coordinates of each department in the interspatial array. Its original size was (75 x 45); this was later expanded to (150 x 49). This array will not use more words than the number of grid squares in the layout. However, because of its structure, IDNDX never utilizes more than 24% of the storage allocated to it.

Maximum layout grid = 40 x 40 or 1,600 locations

size of IDNDX (150, 45)* = 6,750 locations

$$\frac{1,600}{6,750} \qquad 24\%$$

IDNDX is replaced by IDNDY and IDNDZ (refer to Figure #24). IDNDY stores the starting and ending addresses of each department in the layout. IDNDZ stores the rows and column grid coordinates of every block in each department. For example, if the program needs the layout location of the grid squares in department "E", it references IDNDY (1, E) and IDNDY (2, E). The program now has the starting and ending subscripts of IDNDZ which contain the needed information.

IDNDY and IDNDZ are loaded in Subroutine CKISP and referenced throughout the program.

This feature released 18% of the core storage used for data (approximately 4,400 words). Run time was not increased.

I, J ELEMENT LIST

CRAFT-I employs only one method of input for the cost and flow matrices DIST and COVOL This input method has several limitations: 1) the number of cards required to represent the cost and flow matrices, 2) the complexity of card

*Columns 46-49 are used for department transfers only, refer to SHARE write-up.

RAGGED TABLE CONCEPT

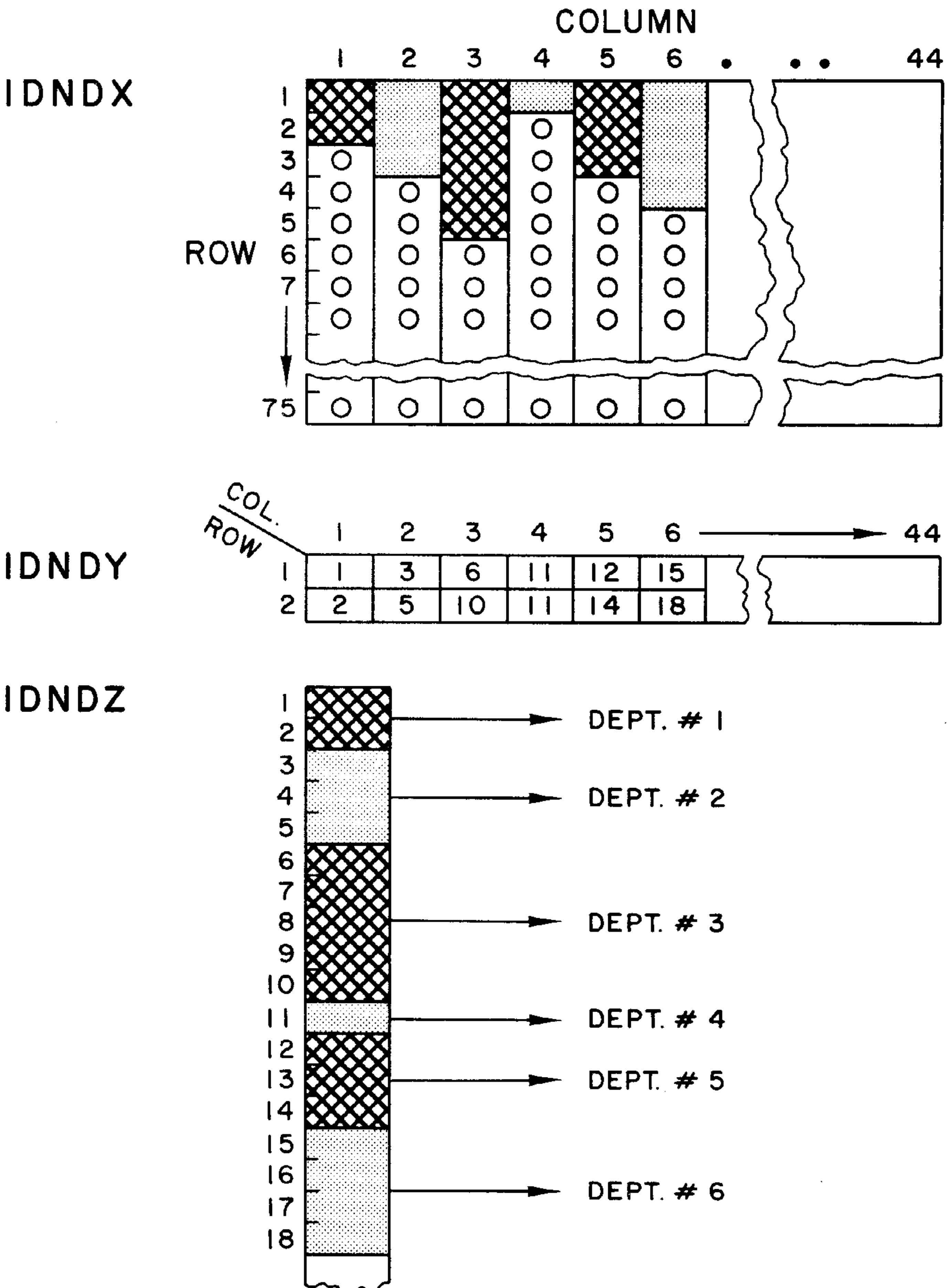


FIGURE #24

field assignment, 3) the specific card order, and 4) the limited accuracy permitted in the representation of cost and flow figures (refer to CRAFT-I write-up for details).

The element list uses one card for each non-zero flow. This card contains 4 pieces of information I, J, DST, CVL. From Department I to Department J a flow of "DST" units occurs at a cost of "CVL" per grid square. These cards may be in any order. The accuracy of "DST" and "CVL" is set at F5.0 but could be easily changed to any "F" format necessary (the element list is read and processed in Subroutine LAYOUT).

Typical cost and flow matrices for the manufacturing layouts in the automobile industry are sparse and large. The use of input method #2 (the element list) will reduce total card input for this type of matrix while allowing increased flexibility and accuracy. (Refer to Appendix A for details of usage.)

COST LIMITING FEATURE

A variable which is user specified (EPSLN) was included to inhibit moves which result in low actual cost savings. CRAFT predicts expected savings that would occur if present department centers were exchanged. The exchange which will produce the highest estimated savings is carried out. If the exchange produces a cost reduction greater than <u>zero</u>, the exchange is accepted and placed into the current layout. In cases similar to the one depicted on the next page the estimated savings may vary considerably from the actual. (Refer to Figure #25.)

The variable EPSLN replaces <u>zero</u> as the minimum acceptable cost savings. The actual cost reduction must now be greater than EPSLN.

This feature is most effectively used to force the program to make better department exchanges in the earlier iterations, thereby more nearly retaining original shapes.

CENTER DISCREPANCY

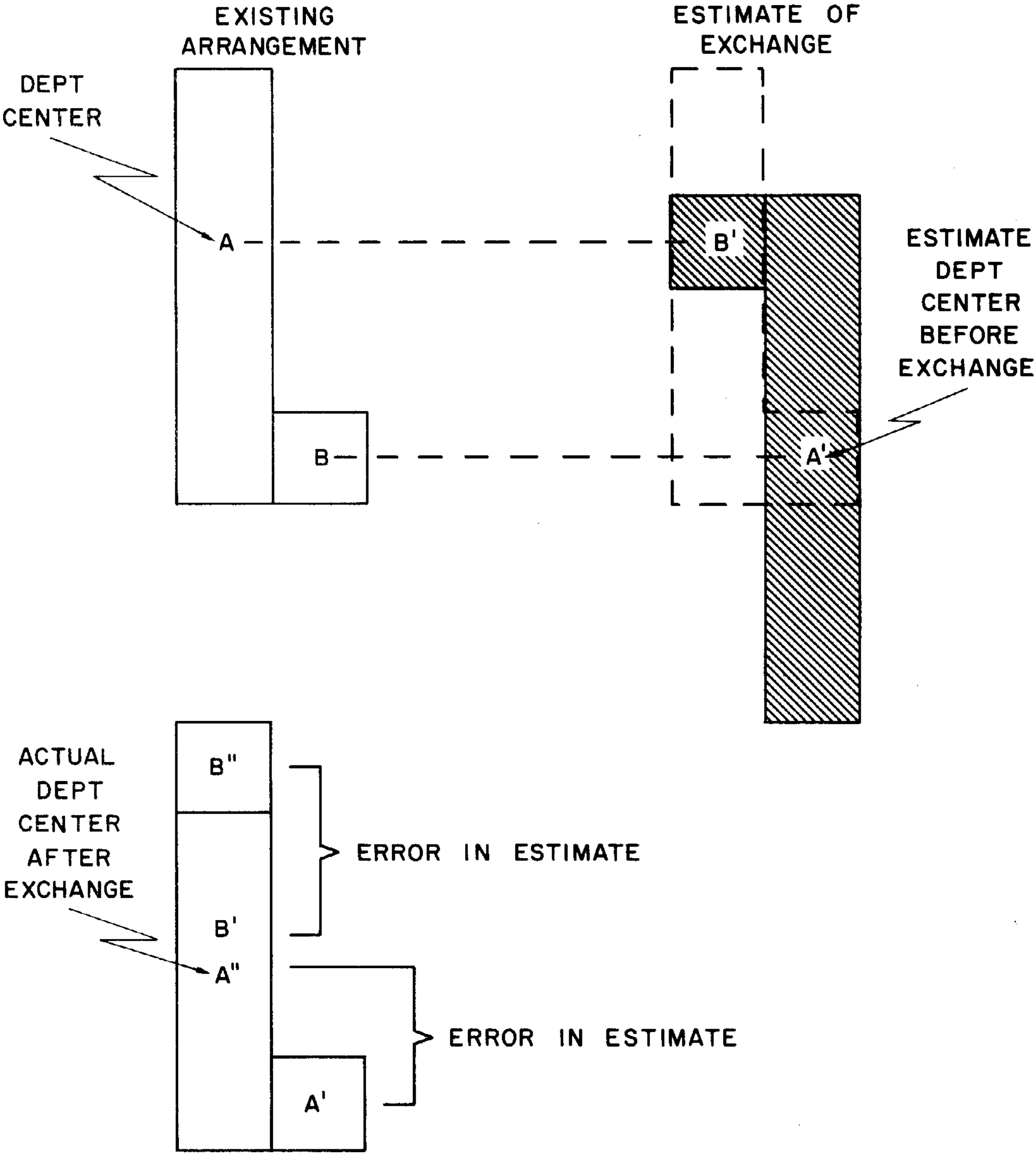


FIGURE #25

The variable EPSLN is read in subroutine LAYOUT and referenced in subroutines ANAN and ANAT.

VARIABLE SIZE DEPARTMENT EXCHANGES

This feature allows departments of approximately equal size to exchange if the criteria of one of the original three exchange methods are not met.

The extent to which a department may vary in size is controlled by a user-specified variable "PCNT". If a department has an original size of 16 grid squares and PCNT = 00.10 (or 10%) the department size could vary from 14 to 18 grid squares (a rounding procedure is incorporated for determining the integer number of grid squares). Once this range of variation is determined, it remains constant throughout a data set, eliminating the possibility of any cumulative increases or decreases in size greater than 10%. The upper and lower bounds of each department are stored in an array named IVRY. This array is 2 rows times the number departments in size. Row 1 stores the lower bound. In Row 2, the upper bound. Each column in IVRY represents a department number. Thus, the lower and upper bounds of Department #5 are located in IVRY (1, 5) and IVRY (2, 5).

The following conditions must be met before an exchange of approximately equal departments can occur:

1. The present size of Department X must lie between the upper and lower bounds of Department Y, and
2. The present size of Department Y must lie between the upper and lower bounds of Department X. (The computer logic is in Subroutine IAJA immediately preceeding the RETURN statement.)

The use of this feature results in a slight increase in run time; however, run time has remained well below 3 minutes for very large manufacturing problems.

The last two columns of the SMRY output list adjusted department sizes if approximately equal departments are exchanged.

INCREASED FLEXIBILITY

Program limits such as maximum grid size, maximum department size, etc., have been changed to variables. In the original program these values were numerical, necessitating considerable program revision if the values were altered.

The following procedure can be used to expand or decrease maximum number of departments, maximum number of grid squares per department, or size of the grid pattern:

1. Reset the desired variable in CRAFT main.

Variable	Function	Present Size
IRFL =	Maximum length and width of the initial layout	40
IRFD =	Maximum number of departments	45
IRFS =	Maximum number of grid squares in any one department	150

2. Reallocate common, adjusting necessary arrays (SHARE write-up will provide array functions, which should simplify this step).

3. Adjust subroutines which contain dimensioned transfer array IMDUM.

   This transfer array is used for temporary storage by the following subroutines:

Subroutine	Transfer Array and Size
CEXC	IMDUM (42, 42)
MATOUT	IMDUM (45, 45)
PERIM	IMDUM (42, 42)
PICKUP	IMDUM (42, 42)

4. Caution: a further expansion of the grid pattern (ISP) will necessitate major output modification. Present output revisions have utilized complete paper width.

   NOTE: There are over 3,300 words of memory available for further program revision or expansion of program parameters.

EXPANSION OF PROGRAM CAPACITY

The grid size was increased from a 30 x 30 layout pattern to a 40 x 40 pattern. The number of departments per layout was increased from 40 to 45. The maximum number of grid squares in any department was expanded to 150, double the original 75 limitations.

This expansion can now be increased even further if necessary (but program tests to date have not indicated the need).

SUBROUTINE SMRY

Subroutine SMRY analyzes the first 39 and the last iteration, shoving both incremental and overall improvement made by each of these 40 iterations.

The columns "CHANGA" and "CHNGB" list the new department sizes resulting from an exchange of approximately equal departments. (The unit of measure is grid squares.)

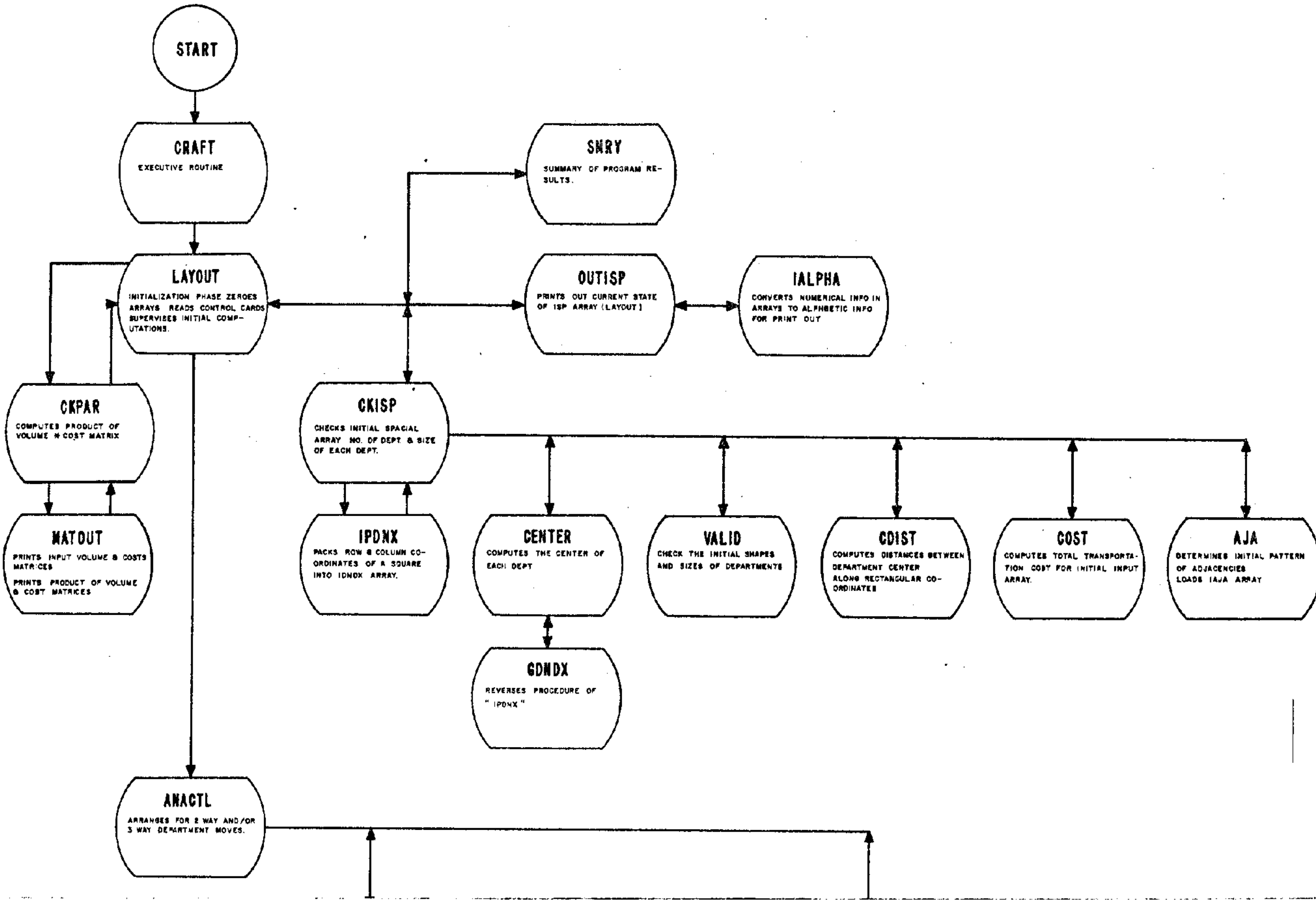
Values for SMRY are loaded in subroutine OUTISP and called from subroutine LAYOUT. Two complete SMRY sheets are printed after each data set.

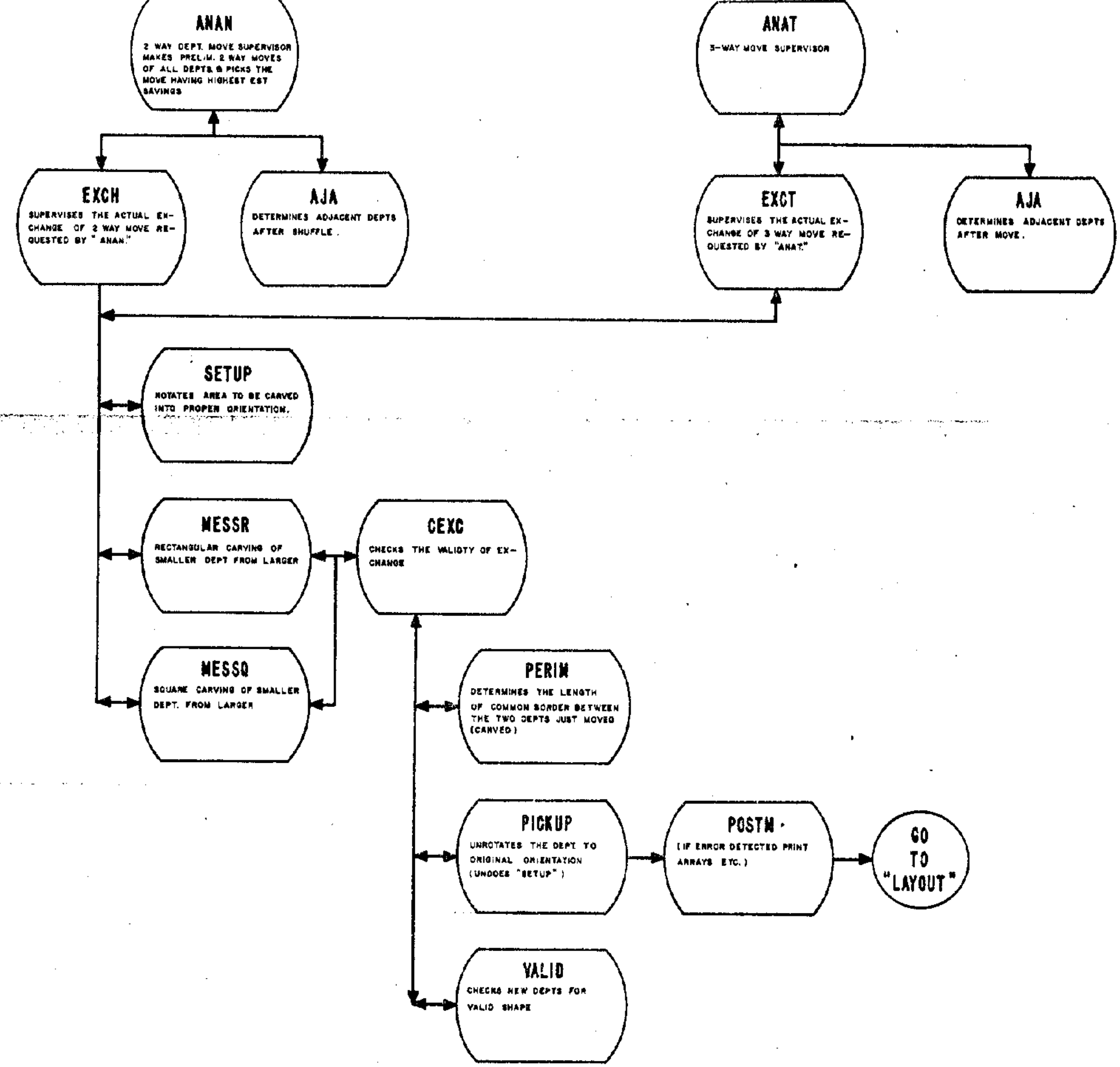
NOTE: A department list or any other desired material may be added to the output by placing this information after the Inter-Spatial Array (ISP) (See Appendix A). Each card placed after ISP will be reprinted one time in a list located between to two SMRY sheets.

APPENDIX D

PROGRAM SUBROUTINE FLOW CHART

CRAFT SUBROUTINE FLOW CHART





ANAN
2 WAY DEPT. MOVE SUPERVISOR MAKES PRELIM. 2 WAY MOVES OF ALL DEPTS & PICKS THE MOVE HAVING HIGHEST EST SAVINGS
ANAT
3-WAY MOVE SUPERVISOR
EXCH
SUPERVISES THE ACTUAL EXCHANGE OF 2 WAY MOVE REQUESTED BY "ANAN."
AJA
DETERMINES ADJACENT DEPTS AFTER SHUFFLE.
EXCT
SUPERVISES THE ACTUAL EXCHANGE OF 3 WAY MOVE REQUESTED BY "ANAT."
AJA
DETERMINES ADJACENT DEPTS AFTER MOVE.
SETUP
ROTATES AREA TO BE CARVED INTO PROPER ORIENTATION.
MESSR
RECTANGULAR CARVING OF SMALLER DEPT FROM LARGER
CEXC
CHECKS THE VALIDTY OF EXCHANGE
MESSQ
SQUARE CARVING OF SMALLER DEPT. FROM LARGER
PERIM
DETERMINES THE LENGTH OF COMMON BORDER BETWEEN THE TWO DEPTS JUST MOVED (CARVED)
PICKUP
UNROTATES THE DEPT TO ORIGINAL ORIENTATION (UNDOES "SETUP")
POSTM
(IF ERROR DETECTED PRINT ARRAYS ETC.)
GO TO "LAYOUT"
VALID
CHECKS NEW DEPTS FOR VALID SHAPE

FIGURE #26

APPENDIX E

COST BREAKDOWN OF ITERATION #0

CLEVELAND ENGINE PLANT #2

COST BREAKDOWN

EXAMPLE: Cleveland proposal (No Float)

The Cleveland proposal has an initial weekly cost of $4,187.07/wk.

ITER#0 . . . . ($4,187.07/wk) x (50 weeks) = $209,353.50/yr or $209,400/yr

Total variable material handling cost for one year equals . . . . $209,400.

This layout has seven departments that use conveyors for material handling: A, B, C, D, H, T and U. Four different types of conveyors are used:

type #1 = #50/ft
type #2 = $75/ft
type #3 = $100/ft
type #4 = $225/ft

The cost of the conveyor used between department A and U is computed as follows:

Rectangular distance between departments A and U	27 grid squares
Length of one grid	33 feet
Type of conveyor used between departments A and U	Type #3 ($100/ft)
Compensation factor for conveyor return, rise and dip	2.5

(33 ft/grid sq) x (27 grid sq's) x (2.5) x ($100/ft) = $222,750

The capital cost of the conveyor system between departments A and U cost $222,750 and has an expected life of 10 years.

Using this calculation procedure, the following chart was derived for the seven conveyor systems that exist within this plant.

From Dept.	To Dept.	Conveyor Tape	Cost
A	U	$100/ft	$222,750
B	U	$50/ft	61,880
C	A	$50/ft	55,690
D	U	$50/ft	53,600
H	T	$50/ft	90,750
T	U	$75/ft	142,350
U	X	$425/ft	371,250
Total capital cost of conveyor system			$998,270

40% of \$998,000 = \$399,000/10 years or

yearly maintenance cost	\$39,900/yr	
yearly capital depreciation	\$99,800/yr	
yearly maintenance cost	+ \$39,900/yr	
yearly conveyor cost	\$139,700/yr	
total variable Material Handling per year		\$209,400/yr
conveyor cost		- 139,700/yr
cost of trailer train; fork truck material handling		\$69,700/yr

The \$69,700/yr trailer train cost is the summation of all variable trailer train Material Handling for this plant. Using the method explained, in Appendix F it is possible to validate this number.

Example of trailer train cost between departments EE and A of this Cleveland proposal.

rectangular distance from department EE to department A	20 grid squares
length of one grid square	33 ft/grid square
number of trips/wk required to supply department A	300 trips/wk
cost/ft of trailer train*	\$.00067/ft

yearly variable trailer train cost for material handling between departments EE and A is:

$$\underbrace{(\$.00067/\text{ft})}_{\text{cost}} \times \underbrace{(33\ \text{ft/grid sq})(20\ \text{grid sq's})}_{\text{distance}} \times \underbrace{(300\ \text{trips/wk})(50\ \text{wk/yr})}_{\text{flow}} = \$6,633/\text{year}$$

A similar exercise for the remaining trailer train patterns, summed together will produce \$69,700/yr, the cost of total variable Material Handling for this proposal.

NOTE: All interdepartmental distances are measured from the geographical center of the departments.

---

*Refer to Appendix F for further information.

APPENDIX F

MATERIAL HANDLING COST DATA

MATERIAL HANDLING DATA

Monorail Conveyor:

average initial cost	$50/ft
estimated life	10 years
replacement and maintenance over a ten-year period	20/ft
adjustment factor to recognize conveyor return chain, and vertical rises and dips	2.5 ft for each foot of rectangular distance between departments
yearly amortized maintenance and replacement	$2/ft
	$7/ft

yearly conveyor cost

($7/ft) x (2.5) = $17.50/ft of distance between departments

($17.50) x (distance) = cost per year

trailer train:

operating cost per hour $6.08

Labor	$4.71
Maintenance, depreciation, etc.	.58
Inefficiency (15%; idle time, etc.)	.79
	$6.08

Average travel speed .0033 min/ft

factor for return of empty train 2

NOTE: when computing trips per year flow between departments, assume:

3 racks/trailer train
1 rack/fork truck

trailer train cost/ft:

($6.08/hr) x (1 hr/60 min) x (.0033 min/ft) x 2 = $.00068/ft of distance between departments

(\$.000668/ft) x (trips/day) x (distance) = variable trailer train costs per year

Load and unload time is constant for any trip. Given a layout, the load and unload costs for trailer trains (or fork trucks) will not vary regardless of the departmental arrangement. This portion of total trailer train cost is therefore not considered by CRAFT.

APPENDIX G

MANUFACTURING ENGINEERING OFFICE BULLETIN

MANUFACTURING ENGINEERING OFFICE

BULLETIN

MANUFACTURING STAFF

FORD MOTOR COMPANY

DATE December 6, 1967 NO. 157

Subject: Computerized Plant Layout

The purpose of this bulletin is to advise that significant advancements have recently been made in the use of computers in the development of plant layouts, and to recommend serious consideration of this technique for future operational planning within the Company.

Computer programming is based on adaptation of the Computerized Relative Allocation of Facilities Technique (CRAFT). This technique provides that the internal arrangement of departments is readjusted within a predetermined building outline, so as to minimize variable costs such as the movement of production and non-production components and personnel.

Refinements to the CRAFT program have resulted from tests involving machining operations within the Company, in conjunction with the Operations Research Department, Finance Staff. Layout alternates, based on computer simulation of departmental rearrangements, are automatically printed out with associated costs and economies to permit appropriate analysis, adjustments and/or selection of the optimum layout.

Current planning includes further efforts to expand application of this technique to additional manufacturing activities and to assembly activities within the Company.

While efforts to date have dealt primarily with program feasibility and refinement, it is now felt that sufficient progress has been made to recommend serious consideration of this technique in operational planning.

It is important to recognize that the CRAFT technique is to be used as a planning "aid" and should not circumvent the need for qualified engineering judgment.

Meetings to present specific program details and techniques are to be scheduled with divisional manufacturing engineering personnel by the Materials and Equipment Engineering Department of this office.

Robert E. Davis, Director
Manufacturing Engineering Office

APPENDIX H

ORIGINAL CRAFT DOCUMENTATION
(Cramer & Armour, 1965)

CRAFT

SHARE PROGRAM CATALOG, PROGRAM DESCRIPTION SUBMITTAL

(For details on the use of this form, see the SHARE Reference Manual, Section 06)

Input Field Number	Field	Entry	Card Column
1	SDA Number (to be filled in by SDA)	S D A 3 3 9 1	A13-21
	Date of Submittal	10/21/65 — New [X] or Revision [ ]	
2	Submitter's Installation Code	R L	ALL CARDS 1-3
3	Program Number or Designation (and Suffix)	C R F T 2	4-8
4	Submitter's Name	P.A. CRAMER	A22-38
5	Submitter's Department (primarily for internal use)		A39-50
6	Author (if different from above)	G.C. ARMOUR	A51-67
7	Year Completed (last 2 digits) or Status Code	64	A68-69
8	Title	COMPUTERIZED RELATIVE ALLOCATION OF FACILITIES TECHNIQUE	Title Card O12-68
9	Field of Application	MGMT. SCIENCE	B12-26
10	Primary Subject Code	H0	B27-29
11	Secondary Subject Codes	T4 T6	B30-41
12	Principal Source Language	FORT4	B42-48
13	Secondary Source Language	MAP	B49-55
14	Type of Routine	MP	B56-57
15	Machine	7090/94	B58-64
16	Monitor or Operating System Required	IBJOB	B65-71
17	Special Machine Requirements		C12-23
18	Non-Library Routines or Subr. Req.		C24-39
19	Documents Available (indicate page counts):		
	Sent to all Installations: Short Write-up	SW000	
	Sent to all Installations: Catalog Cards	15	
	Orderable from SDA: Long Write-up	PA029	C40-49
	Orderable from SDA: Listing	LS028	C50-54
20-21	Program Material Avail.		
20	a. Primary Form	SOURCE DECK	D12-31
	b. Count	01844	D32-39
	c. Medium	BCD	
21	a. Additional Form		D40-59
	b. Count		D60-67
	c. Medium		
22	Search Key	*FACILITY LOCATION*PLANT LAYOUT	Search Card S12-74

Remarks (not to be keypunched)

A

SDA 3391

SHARE PROGRAM CATALOG, PROGRAM DESCRIPTION SUBMITTAL (Continued)

Abstract (Cards 10-99, Columns 12-72)

This program embodies a new methodology for determining suboptimum relative location patterns for physical facilities. It is governed by an algorithm which determines how relative location patterns should be altered to obtain sequentially the most improved pattern with each change, commands their alteration, evaluates the results of alterations, and identifies the sub-optimum relative location patterns. The computer output yields a scaled block diagramatic layout of the facility areas, and the areas need not be equal.

(Please attach additional pages, if necessary)

Pages Attached: keypunchable abstract See Above
Non-keypunchable short write-up X

Signature of Submitter Philip R. Cramer Date 10/21/65

Signature of Installation Addressee SH Jacobs

2

Please use this form for aspects of the program, such as special symbols, charts, etc., that cannot properly be described in the keypunchable abstract.

INST. CODE, PROGRAM NUMBER/DESIGNATION, SUFFIX: R L / C R F T / 2	To be filled in by SDA: SDA NO. S D A 3 3 9 1
DATE SUBMITTED 10/21/65	DATE DISTRIBUTED

G. C. Armour
4681 Jeanean Lane
Yorba Linda, Calif.

P. A. Cramer
2500 Colorado Ave.
Santa Monica, Calif.

CRAFT

CRAFT employs a heuristic approach in solution of problems where multiple entities must interact in some way in order that a measurable result be achieved providing the value or cost of the result so obtained is in some way a measurable function of the relative location pattern of the entities. A manufacturing plant is used as an example of a CRAFT application. It is desired to determine the relative location pattern of entities (departments) which will minimize the sum of the annual costs of transportation between all possible pairs of depts. Input data are an initial layout to scale, the volume of traffic between each pair of departments, the cost/unit-distance/unit-volume for transportation between each pair of departments, any depts. to be held fixed, and parameters describing the magnitude of the economic and spatial data. Neither volume nor cost data need be symmetric. CRAFT makes a series of modifications to the initial layout until it can no longer obtain a reduction in total transportation costs. CRAFT prints out a revised layout to scale, identifies the departments moved, cost reduction obtained, and total cost for each modification made. The suboptimum layout finally obtained is not guaranteed to be mathematically optimum, but this is difficult to obtain.

It is shown in (1), that a simplified version (considering all depts. as points) of the plant layout problem can be formulated as a quadratic integer programming problem, but even then there is no known algorithm which can be used to obtain a solution. It is obvious that an exhaustive search is not feasible for a problem of any size. CRAFT is a compromise between the step-by-step improvement of a linear programming algorithm and an exhaustive search. CRAFT searches through a list of possible configurations each representing a small modification of the present layout which can be obtained from the present layout by switching the locations of two departments, provided that the two departments meet one of the following criteria: (1) They are the same size; (2) They have a common border; and (3) They both border on a common third dept. CRAFT chooses the modification which promises the greatest improvement over the present situation, this new configuration then presents another list of possibilities, etc., until there is no configuration on the current list of possibilities which promises an improvement over the present situation.

The program uses an estimating procedure to choose the best alternative. It calculates what the change in transportation costs would be if "A" were to be centered at the center of department "B" and vice versa. (All traffic is assumed to travel between department centers.) On the basis of this estimate, the most promising pair of departments are actually switched yielding a change in transportation cost which may differ from the estimate (if it differs so much as to be negative, the program will reject the alternative and try again). The procedure is repeated until no further improvement is indicated.

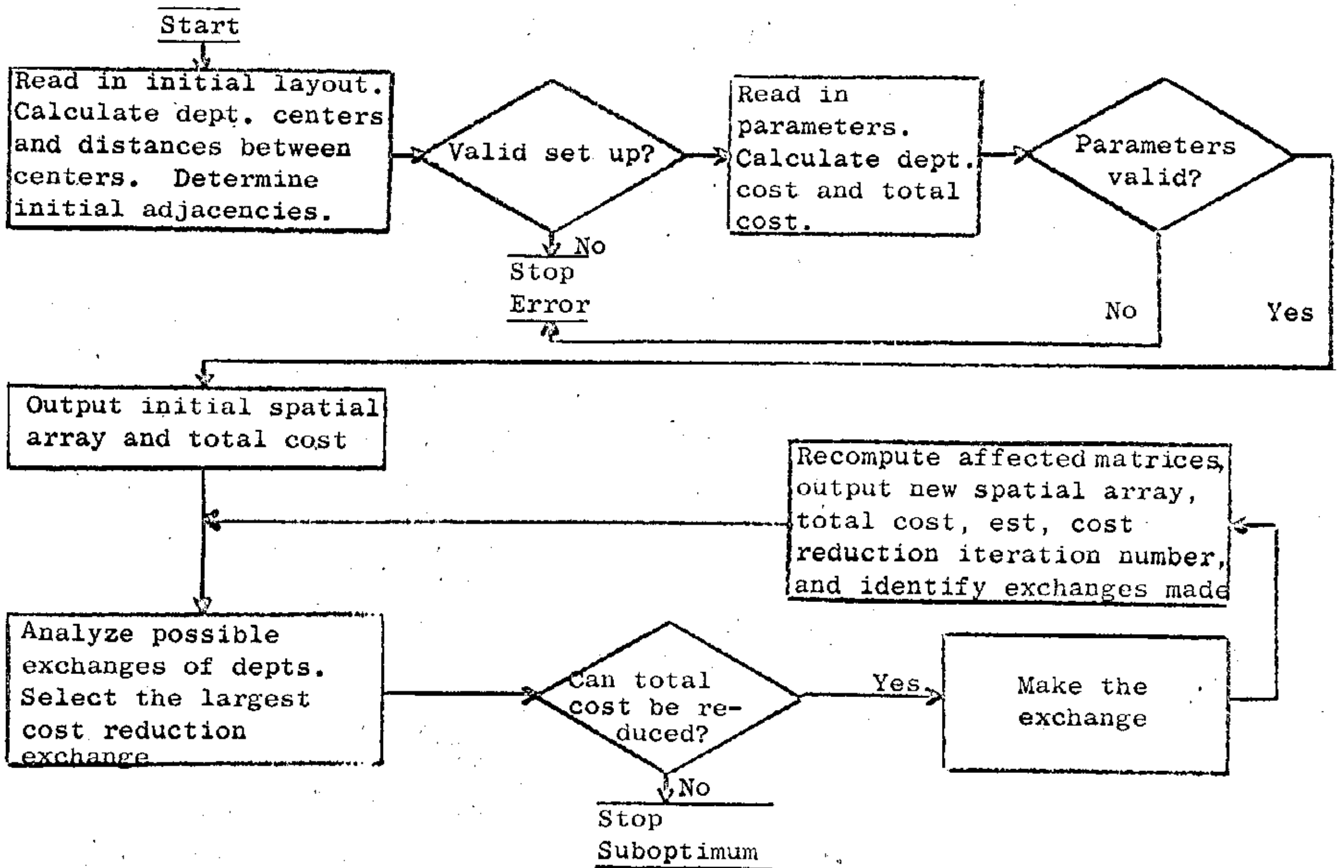
Please use this form for aspects of the program, such as special symbols, charts, etc., that cannot properly be described in the keypunchable abstract.

INST. CODE, PROGRAM NUMBER/DESIGNATION, SUFFIX:	To be filled in by SDA: SDA NO. SDA3391
DATE SUBMITTED	DATE DISTRIBUTED

G. C. Armour
4681 Jeanean Lane
Yorba Linda, Calif.

P. A. Cramer
2500 Colorado Ave.
Santa Monica, Calif.

A description of the input data and the control card is in the long write up.



Flow Diagram of the Computer Program

The user should be able to run the CRAFT program and obtain useful solutions after reading the above description, the input description, and one or both of the articles. CRAFT now considers up to 40 depts, allows the user to fix departments in place at will, and produces a suboptimum layout to scale. No special timing problems or crowding of core storage are involved. Should problems arise or additional insight into the workings of the program be required, there are extensive comment cards in the source deck, and a write-up of all subroutines and arrays in the accompanying program material.

(1) ARMOUR, Gordon C. and Elwood S. Buffa, "A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities," Management Science, Vol. 9, No. 2, January 1963.

(2) BUFFA, Elwood S., G. C. Armour, and T. E. Vollmann, "Allocating Facilities with CRAFT," Harvard Business Review, Vol. 42, No. 2, March-April 1964.

CRAFT data deck
write up

3. Unit cost array, format (20F4.3)

This array is punched on a one to one relationship with the volume array. (e.g., for the first card of the first set punches in Col3. 9-12 would input the unit cost of moving a unit load a unit distance from department number 1 to department number 3.

The decimal point is not punched. A decimal point will be considered to lie between the first and second positions of each field for purposes of computation. (e.g., a punch of 1015 in a field would be treated as 1.015 for computation).

4. Initial spatial array, format (30I2)

Punch each row of the array on a separate card. Department numbers of the departments occupying each space are punched on the card in two column fields. The array is punched by rows. There should be as many cards punched as is specified in columns 3-4 of the control card. There should be as many fields punched on each card as is specified in columns 5-6 of the control card.

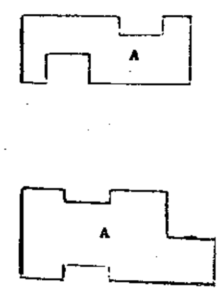
Department numbers are converted by the program to alphabetic characters for print out (e.g., 01 = A, 02 = B,..., 26 = Z, 27 = AA, 28 = BB, etc.).

Irregularities in building configuration should be filled in by adding artificial departments until a rectangular building configuration is achieved. These artificial departments may be held fixed. See control card.

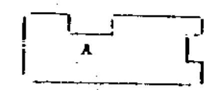
The maximum number of times that any one department may be punched is 75. No department may be disjointed. Each department which appears in more than one field or card must be punched so that when the keypunched spatial array is listed each department number has a like department number immediately adjacent in the same row or column. Departments may have multiple indentations along one axis but not along both axes. A department may not completely surround another department on all sides.

CRAFT data deck
write up

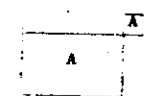
Some
Legal
Configurations



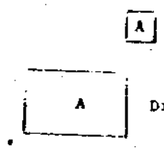
Some
Illegal
Configurations



Indented along both axes

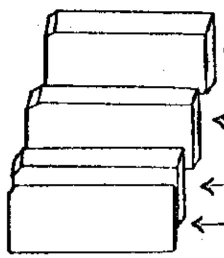


Touches only at a corner



Disjointed

CRAFT - Schematic data deck set up



The user should be able to use the program quite successfully based on the above data deck write up and the preceding short write up. The remainder of this long write up should be of interest to the professional programmer only.

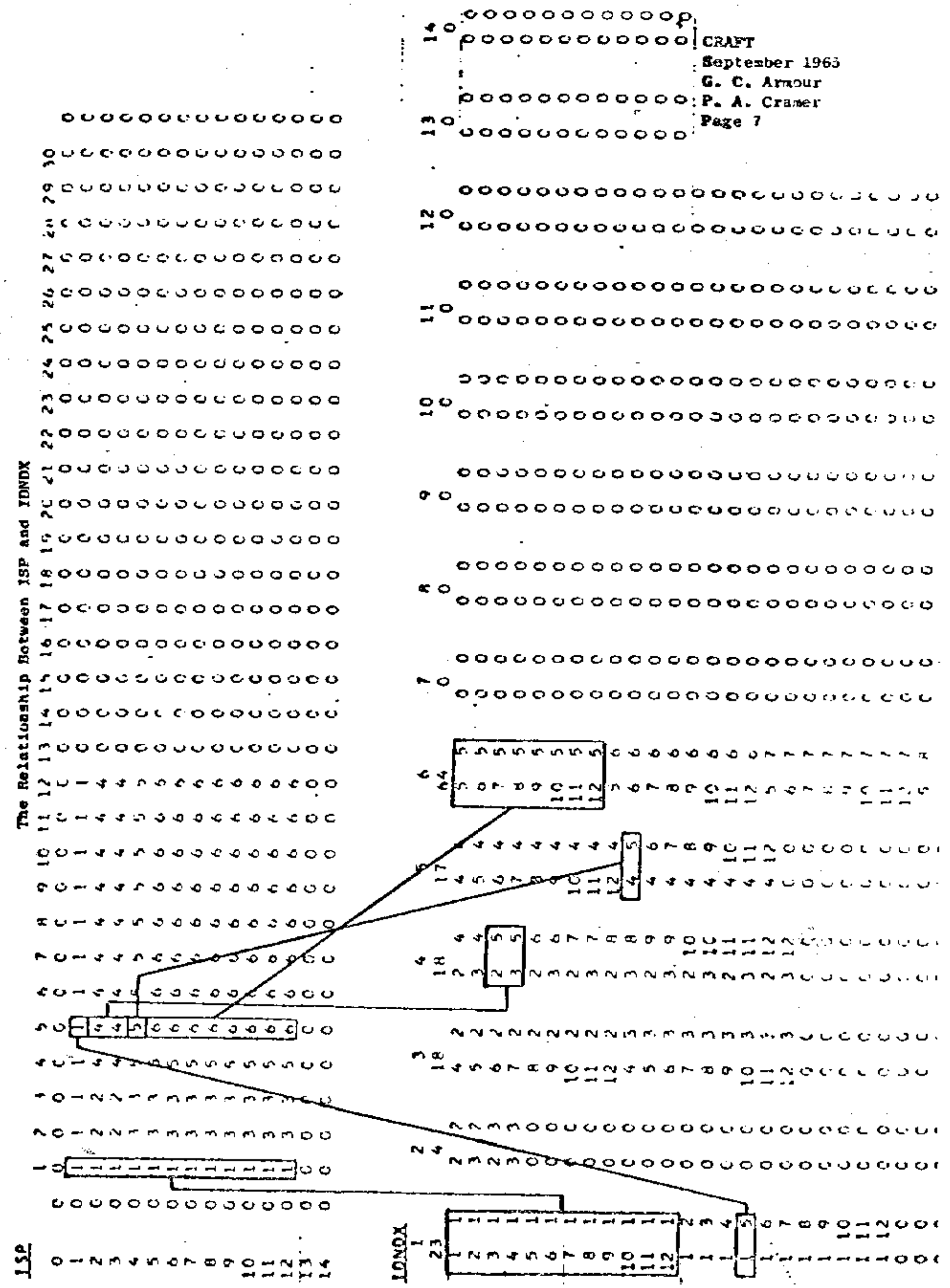
IAJA (40, 40)

IAJA contains codes which indicate whether or not each pair of departments can be exchanged. If a department is fixed its entire row and column contains -1 codes. If departments 3 and 7 have a common border, IAJA (3, 7) and IAJA (7, 3) will contain a +1; if departments 3 and 10 are the same size but do not have a common border, IAJA (3, 10) and IAJA (10, 3) contain a +2. Both of these codes are acceptable for two-way exchanges, and the +1 is acceptable for three-way exchanges. A -2 code indicates that a pair of departments have been switched and either the actual cost saving has turned out to be negative or that the resulting configuration was invalid and hence this pair should temporarily be removed from consideration. Finally, a 0 code indicates that a pair of departments is neither adjacent nor the same size and hence cannot be considered for either exchange.

The relationship between IAJA and ISP (the spatial array - see below) is illustrated in the write-up of the subroutine AJA.

ISP, IT, IU, IV

ISP is the array in which the physical layout is stored. The maximum size of the layout is 30 rows by 30 columns and ISP is 32 x 32; the extra rows and columns are used to form a border of zeros. This simplifies programming by eliminating continuous checks on the value of index parameters, but it has the effect of moving the third row of the input layout to the fourth row of ISP, etc. Hence the program most frequently refers to ISP (I + 1, J + 1), so that the values of I and J can be used directly to refer back to the input layout. IT, IU, and IV are used for temporary storage of portions of ISP, sometimes in a rotated orientation. IT is used by VALID, IU is used by MESSR, and IV is used by MESSQ.



The Relationship Between ISP and IDNDX

IDNDX (75, 44)

IDNDX contains an index by department of the squares of ISP. For example, IDNDX (15, 2) contains the row and column coordinates (packed into one word by IPDNDX) of fifteenth square of department 2. (Initially fifteenth means fifteenth as encountered in a column by column scan of ISP, but the order of squares is not vital; what is important is the fact that column 2 of IDNDX contains a list of all squares in department 2, and hence we do not have to scan ISP repeatedly.) Columns 1-40 contain the indexes of departments 1-40. Columns 41-44 are used for temporary storage of new department configurations. For two-way exchanges, column 41 is used for the new smaller department and column 42 is for the old smaller department plus the remains of the old larger department (from which the new smaller department has been carved) which together form the new larger department. For three-way exchanges (departments A and C being exchanged across department B; A the smaller of A and C) the usage is as follows: First department B is moved to column 42. After A is carved out of C, the new A is stored in column 41, the remainder of C is added to column 42 spilling over into column 43 if necessary. Old department A is added to this, also spilling over if necessary (single departments are limited to 75 squares but this conglomeration is the size of B and C together). The smaller of departments B and C is carved from this amalgamation and the new department B winds up in 42 and new department C in 44.

IDEPT (11) and XDEPT (11)

These contain fixed and floating point representations of the size (number of squares) of each department.

ROWCEN (14) and COLCEN (11)

These contain the row and column coordinates of the center of each department (as computed by CENTER).

DCOST (44)

This contains the total transportation cost involving each department, as computed by COST. Thus the total cost is one-half the sum of the costs (since the transportation costs between departments 5 and 8 are included in both DCOST (5) and DOST (8), etc.).

4

IW (44)

IW is used to contain the alphabetic department codes for printout of the physical layout.

IDFIX (44)

IDFIX contains the list of fixed position departments.

DIST (40, 40)

DIST contains the distance between each pair of departments (as computed by CDIST). Initially it is used to contain the volume array.

COVOL (40, 40)

COVOL contains, for each pair of departments, the cost per unit distance for the total volume of traffic for the two departments. Initially, it contains the unit cost array: this is multiplied element-by-element with initial contents of DIST to obtain the regular contents of COVOL. COVOL (I, J) contains data for traffic from department I to department J, COVOL (J, I) data on traffic from J to I.

AJA

AJA scans the spatial array and enters codes in IAJA to indicate which pairs of departments can be exchanged (-1 codes for fixed departments are put in IAJA by CKISP and left untouched by AJA). First AJA scans ISP, testing at alternate squares (e.g., black squares of a checkerboard) to see whether this square belongs to the same department as the squares immediately above, below, left and right. When it finds adjacent squares in different departments, it obtains the department numbers, say K and L, and enters +1 codes in IAJA (K, L) and IAJA (L, K). When this scan is complete, AJA compares the numbers of squares in each department L (IDEPT (L)) with the number of squares in departments K = 1, 2, L -1, and enters +2 codes in IAJA (L, K) and IAJA (K, L) if they are still zero. In the example below, department 5 is fixed in place, so we have -1 codes in row and column 5 of IAJA. We scan down column one, and at the third square tested we find a difference in departments; the square we are testing is in department 3, and the square above is in department 1, so we enter 1 in IAJA (1, 3) and IAJA (3, 1). The remainder of the 1 codes are arrived at similarly and the 2 codes in IAJA (4, 1) and IAJA (1, 4) are added as a result of the comparison of number of squares in departments.

ISP

1	1	2	2	4	4	4
1	1	2	2	4	4	4
1	1	2	2	6	4	4
1	1	3	3	6	6	6
3	3	3	3	5	5	5
3	3	3	3	5	5	5

IAJA

	1	2	3	4	5	6
1	0	1	1	2	-1	0
2	1	0	1	1	-1	1
3	1	1	0	0	-1	1
4	2	1	0	0	-1	1
5	-1	-1	-1	-1	-1	-1
6	0	1	1	1	-1	0

21

ANACTL

ANACTL arranges for two-way and three-way moves to be attempted in the manner specified in columns 7-8 of the control card. For example, if ICTL = 02 (two department moves followed by three department moves), ANACTL calls ANAN (two-way move supervisor) repeatedly until there are no more profitable two-way moves and then ANAT until there are no further profitable three-way moves.

For ICTL = 04 (choose best of two and three department moves at each stage) it first calls ANAN and ANAT with a parameter of -1, indicating that each routine is to find its best move and then return before actually making the move. ANACTL chooses the move with the higher estimated saving and calls ANAN or ANAT accordingly, this time with a +1 parameter indicating that the previously chosen move is to be made.

ANAN

ANAN is the two-way move supervisor. It pairs each department J with departments I = 1, 2,...,J -1, and checks IAJA to see if they are adjacent or the same size and hence candidates for exchange. For each pair of departments which are candidate for exchange it computes the effect on annual cost which would result if the department exchange meant that the center of the first department became the new center of the second department and vice versa. EXCH is called to attempt to exchange the pair with the highest estimated saving. If the exchange fails, it enters -2 codes in IAJA and tries the second best pair. If the exchange is successful, it computes new actual centers, distances and costs for the new departments (using CENTER, CDIST, and COST), and checks to see if there is actually a cost saving when actual rather than estimated department centers are used. If there is, it moves the new departments into the regular portion of IDNDX, changes ISP to reflect the new department shapes, calls AJA to determine which departments are now adjacent, and returns. If there was no actual cost saving, it recomputes centers, distances, and costs for the old departments, enters -2 codes IAJA and tries the (estimated) next best pair of departments. (To find the next best pair, it repeats the process described in the first two sentences above; the -2 codes eliminate the pair which failed.)

ANAT

ANAT checks all sets of departments I, J, and K, and finds those for which I is adjacent to J and J is adjacent to K, but I is not itself adjacent to K. It then estimates the cost saving that would result if I and K were exchanged. It chooses the set of I, J, and K with the highest estimate and calls EXCT which attempts to exchange I and K across J. If EXCT fails, we set -2 codes in IAJA for departments I and J and go back to the beginning to find the second best set of I, J, K. If EXCT is successful, we calculate centers, distances and costs for the new departments and check to see if there is an actual cost saving. If not, we recalculate centers, distances and costs for the old departments set -2 codes and go back to the beginning. If there was a cost saving (as we expected), we move the new departments into the regular portion of IDNDX, change ISP to reflect the new arrangement of departments, and call AJA to determine which departments are now adjacent.

CDIST

CDIST (J) computes the distance between the center of department J and the centers of each of the other departments. In the program as distributed the method used is to measure distance along rectangular coordinates on the assumption that traffic moves more nearly around corners than diagonally. If this were not the case in a particular problem, the appropriate formula could be substituted in CDIST (CFT03210 and CFT 03220) and the rest of the program would not be affected.

CENTER (I, J)

CENTER calculates the x- and y- coordinates of the center of department I and stores them in COLCEN (J) and ROWCEN (J) respectively. When I ≠ J, I = 41, 42 or 44 - temporary locations of newly carved departments. The method used is as follows: for the x- coordinate, sum the column coordinates of each square in the department and divide by the number of squares in the department. In the example below there are:

2 squares in column 3, 2 x 3 = 6

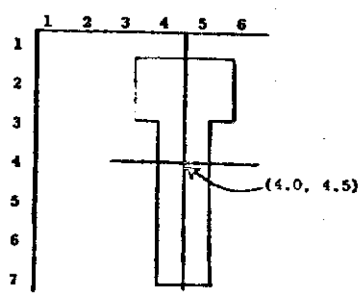
6 squares in column 4, 6 x 4 = 24

6 squares in column 5, 6 x 5 = 30

2 squares in column 6, 2 x 6 = 12

and the sum of the column indexes is 72

Dividing by 16 yields 4.5. For the y- coordinate, a similar process yields (4 x 2) + (4 x 3) + (2 x 4) + (2 x 5) + (2 x 6) + (2 x 7) = 64. Dividing by 16 yields 4, as shown. This is a first moment or center of gravity calculation. This means that if we tried to balance a piece of cardboard the shape of the department on the tip of a pencil, we would find that it would balance when we put the pencil under the point shown. All traffic is assumed to move from department center to department center.



CEXC

CEXC is called by MESSR and MESSQ with arguments to use PICKUP twice to pick up the resulting two areas left by the MESS routine. CEXC calls VALID to check the validity of both pieces, then calls PERIM to compute the length of the common border. If either solution is invalid, this information is passed to the calling routine.

CKISP

CKISP scans the original spatial array, checks to see that the department number in each square is between 1 and NDEPT (column 1-2 control card), keeps a count of the squares in each department, and checks that each department has at least one square but no more than 75 squares, and uses IPDNDX to pack the row and column coordinates of each square into a single word for storage in IDNDX. IDNDX (K, L) thus contains the row and column coordinates of the Kth (as encountered in the scan) square of department L.

It calls CENTER to compute the centers of the initial departments, and VALID to check the initial shapes. It uses the fixed department data on the control cards to set a -1 code in each row and column of IAJA corresponding to a fixed department and calls AJA to determine the initial pattern of adjacencies. It uses CDIST to compute the distances between department centers and COST to compute departmental and total costs for the initial array.

CKPAR

CKPAR calls MATOUT to print volume and unit cost input matrices, checks them and computes their element-by-element product. That is, for each pair of i and j, it computes CV (i, j) = C (i, j) . V (i, j) (In the program: COVOL (I, J) = COVOL (I, J) * DIST (I, J).) This gives the cost/unit distance for the total traffic between each pair of departments. Since we are only varying distances, not volumes or unit costs, this is the most efficient way to store and use the input data. Finally, CKPAR calls MATOUT to print this new matrix.

COST

COST (I, J) computes the total transportation costs between department I and each of the other departments, assuming that department I is centered at department J. Thus if I = J we have the true cost of transportation involving department I. If I ≠ J, we have an estimate of the cost of transportation for department I if its center were the center of department J. It is on the basis of these estimates that we choose a pair of departments to be exchanged. (If department I is not the same size as department J, it is unlikely that the center of the new I will be the same as the center of the old J; hence our actual cost saving computed using actual post exchange department centers may vary from the estimate and occasionally may be negative. When this happens a -2 code is entered in IAJA (I, J) and IAJA (J, I); all this is done in other routines however.)

EXCH

EXCH prepares the departments to be exchanged in a two-way move for the actual carving routines, calls these routines and chooses the better resulting arrangement of departments. If the departments are the same size, carving is unnecessary and it simply exchanges the department indexes and returns. Otherwise, it moves department A (the smaller department) into column 42 of the index where it will later become part of B. It then calls SETUP to rotate B into standard orientation for carving, calls MESSR and MESSQ for rectangular and square carving respectively, and chooses the configuration with minimum common border if both routines succeed. The new department (carved out of B) is placed in column 41 of the index and the remainder of B is added to the old department A in column 42 to form the new department B.

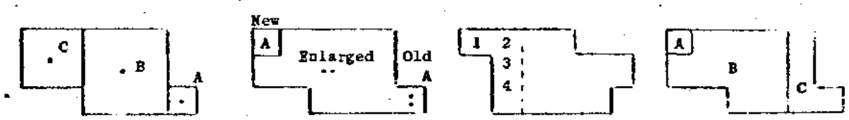
EXCT

EXCT makes the arrangements for the exchange of departments A and C across department B. First it uses SETUP to rotate department C (the larger of A and C) into standard orientation with respect to department B. Then it calls MESSR and MESSQ to carve A out of C. If both are successful, it chooses the arrangement with the smallest common border. The new A is placed in column 41 of IDNDX. Department B, the remainder of C and the old A are placed in column 42 (and 43 if necessary) of IDNDX and the smaller of B and C is carved out of this enlarged department. If C is smaller than B, we use SETUP to rotate the enlarged department into a standard orientation arrived at by using the center of the old department A as the center of the "larger" department and the center of the enlarged department as the center of the "smaller" department. As is explained in the write-up of SETUP the object of this rotation is to enable us to begin carving from the corner and side

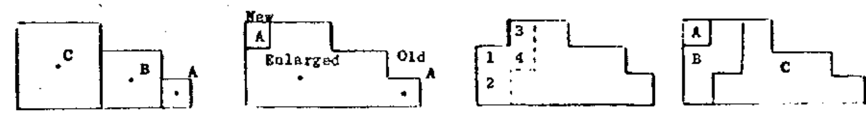
of the "larger" department which are farthest from the "smaller" department. Hence this rather devious use of spurious department centers means that we will be carving department C from the general area of the enlarged department where the old department A is located; since we are exchanging A and C this makes sense.

Conversely, if B is smaller than C we use SETUP to rotate the enlarged department into a standard orientation arrived at by using the center of enlarged B as the center of the "larger" department and the center of old department A as the center of the "smaller" department. This means we will be carving B out of the area of the enlarged department which is farthest from the old A, that is B and the remainder of C; this leaves the area of the enlarged department in and around the old department A as the new department C, which again makes sense.

In any case MESSR and MESSQ are called for rectangular and square carving respectively, and if both are successful, EXCT chooses the configuration with the shortest common border between B and C.



Original | Carve A out of C Combine remains | C smaller than B Rotate, carve C | Rotate back, final arrangement



Original | Carve A out of C Combine remains | B smaller than C No rotation necessary here Hence, carve B | Final arrangement

IPINDX and GINDX

IPINDX is a MAP function which packs the row and column coordinates of a square of the spatial array into one word for storage in IINDX.

GINDX is a MAP subroutine which reverses the above procedure, it takes an entry in IINDX and splits it into row and column coordinates.

IALPHA

IALPHA converts a department number to the corresponding single or double letter code for printout. 1 = A, 26 = Z, 27 = AA, etc. It is used only byOUTISP. A department number of 0 or greater than 40 results in a blank for printing.

MAIN

MAIN is a skeletal program which serves only to call LAYOUT, the first subroutine.

LAYOUT

LAYOUT is the supervisory routine for the initialization phase of the program. It zeros arrays, reads and checks the control cards, reads the volume and unit cost matrices, calls CKPAR to do the necessary initialization with these, reads the original spatial array (floor plan), calls CXISP to do the initial work on this, calls OUTISP to print it, and passes control to AHACTL. At the completion of a problem, AHACTL returns control to LAYOUT which reinitializes for the next problem.

MATOUT

MATOUT prints out a variety of program matrices, such as the initial volume and unit cost matrices, with appropriate headings. Matrices of more than ten columns are printed in ten-column segments due to limitations in paper width.

MESSR

MESSR operates on the rotated copy of the larger department (herein called B); it carves the smaller department (A) out of B by taking vertical slices starting in the northwest corner. If this fails, it takes horizontal slices from the top. All scanning done by MESSR is confined to the smallest rectangle containing B. The carving process is divided into two parts: 1) A fairly uncomplicated process for the first portion of the carving, and 2) A very careful terminal process which takes over when the number of squares left in A is less than the length of a column in the rectangle containing B. The differences in the two processes are most fully utilized when B is irregularly shaped. The first process simply scans B, column by column, relabeling squares in B with the number of department A, and decreasing the count of squares required to carve A out of B. The program proceeds all the way from the top to the bottom of the column, whether or not there are blank squares between squares of B. It does, however, keep track of two things: 1) Does the column being scanned (or just finished) have a gap in it?, and 2) Has there been a previous column with no gap in it? When the number of squares left in A at the end of a column drops below the length of a column, these are interrogated. If there is no column without a gap, we give up and try horizontal slices, otherwise we go to the finishing routine. We now become very careful in order to avoid two errors: 1) Having the remaining squares of A unconnected with the portion already carved, or 2) Creating an indentation in department B by leaving squares of B on both ends of A. If these are unavoidable, we try horizontal slices. The finishing routine is as follows:

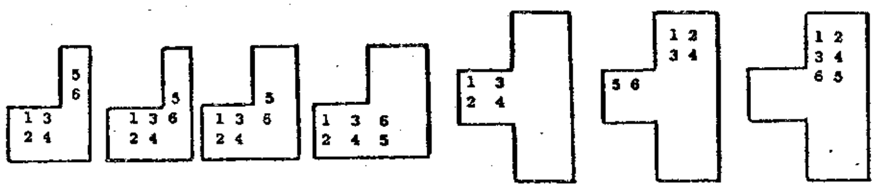
1) Scan down (up) the column just processed, find first square of A (SQ 3 in Fig. 1A).

2) Look to right for B; if this square is not in B, we are safe and we proceed immediately to step 4.

3) Scan up (down) this column (one to the right of the column in step 1), counting squares in B. If this is no more than the number of squares left in A, we do step 4. If not, we go to step 6, since step 4 would leave us with error 1.

4) Come down (up) this column replacing B's with A's, and decrementing the count of squares left in A until either 1) we find a square not in B, in which case we return to step 1 and apply it to this new column, or 2) A is completely moved.

5) Call CEXC to check the new configuration:



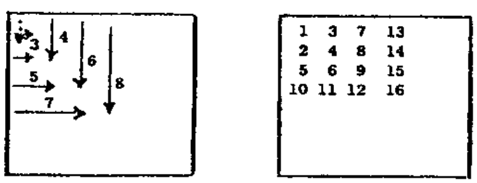
Error 1 Fig. 1a | Error 2 1b | Correct 1c | Vertical: Errors Unavoidable Fig. 2a | Horizontal = Error 1 2b | Horizontal Correct 2c

If it is valid, we return; if not, we try horizontal slices. This process is completely analogous to the vertical process, except that if CEXC indicates a failure here also, we return with an error code set.

6) Set parameters and apply steps 1-5 working from the bottom of the column (this is how we arrive at Fig. 1C). (Substituting the parenthesized up down, and up in steps 1, 3 and 4.)

<u>MESSQ</u>

MESSQ carves the smaller department (A) out of the larger department (B) by taking successively larger squares out of the upper left (see SETUP) corner. For a simple rectangle, the process is illustrated below:



We are not guaranteed simple rectangles, however, so the process is actually as follows, starting in the corner:

1) Check the horizontal failure flag (explained below; off initially). If it is on go to step 6, otherwise we check the first cell in the row just below the square as it has been carved thus far; if this cell is not in B, we go to step 4.

2) Check the cell to the left of this cell to see if it is also in B; if not, go to step 4, otherwise continue scanning left, counting cells in B until we find a cell which is not a member of B.

3) If the cells counted are more than the squares remaining in A, set a horizontal failure flag, indicating that further horizontal carving would result in an error condition (see write-up of MESSR) and go to vertical portion of routine (step 6). (If we already have vertical failure, we set an error code and return, otherwise we do step 4.)

4) Scan horizontally to the right, replacing B's with A's and decrementing the count of squares left in A until this count reaches zero (go to step 12) or until we reach the cell beneath the last column carved. If no cells in B are actually encountered in this scan, an empty-row flag will be left on.

5) If both the empty-row and vertical-failure flags are on at this point we cannot avoid an error condition so we set an error code and return.

6) Steps 6-10 are vertical analogs of steps 1-5. Briefly: 6) if vertical failure flag on go to step 11, otherwise check cell at top of next column, if it is not in B, go to 9.

7) Check cell above this cell, if it is not in B go to step 9, otherwise continue scanning up this column counting cells in B.

8) If cells counted are more than the cells remaining in A, set a vertical failure flag and go to 11.

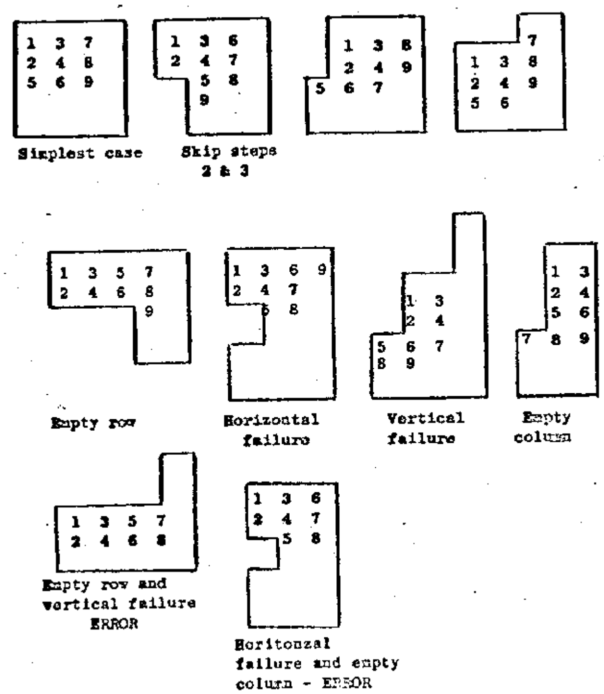
9) Scan (back) down column replacing B's with A's, etc., until we reach the cell to the right of the last row carved. If no cells actually in B an empty column flag is left on.

10) If both empty column and horizontal failure flags on, set an error code and return.

11) Step up bounds of square to reflect latest carving, go to step 1.

12) Call CEXC to check the shapes of new departments.

Some patterns of carving, both successful and otherwise are outlined below; in all cases there are 9 cells to be carved.

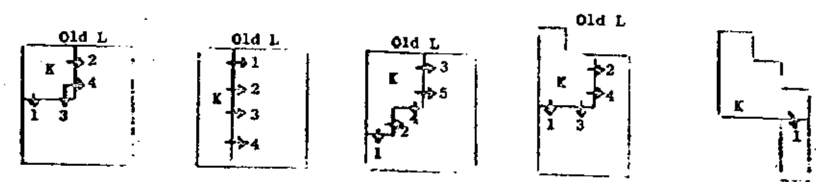


OUTISP

OUTISP prints out the current state of ISP (the spatial array or floor plan) in a readable format. It prints only the edges of each department leaving the interiors blank, and on these edges it substitutes a letter or pair of letters for the department number (department number 1 is outlined in A's, etc.). It also prints summary information indicating which departments have just been switched, the latest cost reduction, the current total cost and iteration count. By specifying 00 in CC 9-10 the printing of the spatial array can be suppressed for intermediate stages, and only the summary information printed.

PERIM

PERIM determines the length of the common border between the new department K and the department (L) out of which it has just been carved. It does this in the following manner: It checks each square in the area occupied by the old department L and for those now in department K, it checks the squares immediately above, below, left and right. For each such square that is in department L, it counts one. The result is the length of the common border. Examples are shown below. The arrows indicate those cases in which a sq in L is found above, below, left or right of a sq in K.



PICKUP

After the carving has been done, PICKUP reverses the rotations performed by SETUP. It uses the parameter IAED computed by SETUP to avoid repeating the analysis of relative orientations. Since part of old department B is now department A, two calls to PICKUP are necessary, one for the new A and one for the remainder of B. PICKUP is told what to scan for and how many cells it should find in order to detect error conditions. Detection of error conditions passes control to POSTM.

POSTM

POSTM is called whenever the program detects an internal error condition. It dumps the following arrays in a readable format: IAJA, ISP, IT, IU, IV, IDNDX (in 15 column segments), ROWCEN COLCEN, XDEPT, DCOST, IDFIX, and IW. It calls MATOUT to print DIST, and PDUMP to dump the undimensioned variables in COMMON. It then calls LAYOUT to initiate another problem.

The use of POSTM, which may be deduced from the appearance of the above arrays in the output indicates one of two things.

1) Input data is invalid or inconsistent (see list of errors below and instructions for preparing input data); in this case the error message preceding the arrays should lend fairly directly to the trouble.

2) Something has gone seriously wrong in the course of program execution.

Foreseeable difficulties, such as invalid department shapes or no actual cost reductions are anticipated by the program and provisions are made to bypass these situations without a call to POSTM. When POSTM is used in the middle of a program it is likely that some portion of the core has been destroyed and this has shown up later in the form of invalid or inconsistent data; the actual cause of the difficulty may be several steps removed from the detection of the error. An examination of the information printed by POSTM should aid in determining what part of core has been affected and may give some clue as to how (for example, a scan may have continued beyond the bounds of an array). It was for this debugging use that POSTM was developed; it should prove quite helpful to any one modifying the program.

POSTM is called from the following routines for the reasons indicated.

CDIST -

1) Message: CDIST PARAMETER NEG OR GREATER THAN NDEPT OR NOT 41-44

- meaning: PARAMETER references column of IDNDX containing list of sqs in a department; hence program has been referred to column which cannot contain valid data.

2) Message: DIST NEG OR GREATER THAN 60.

- meaning: 60 is max possible distance with present max size of spatial array.

CENTER -

1) Message: ICEN NEG OR GREATER THAN NDEPT OR NOT 41-44

- See 1) above.

2) Message: INVALID ROWCEN OR COLCEN COMPUTED

- meaning: A department center has been computed which is outside the boundary of the spatial array.

CKISP -

1) Message: INITIAL CELL ZERO OR GREATER NDEPT

- meaning: input error in spatial array.

2) Message: TOO MANY OR NO CELLS IN A DEPT.

- meaning: input error: MAX CELLS IN A DEPT: 75

3) Message: INITIAL DEPT INVALID

- meaning: input layout contains invalidly shaped department, see description of layout data.

4) Message: FIXED DEPT ZERO OR GREATER THAN NDEPT

- meaning: input error (control card fields inconsistent)

CKPAR -

1) Message: COST OR VOLUME DATA INVALID

- meaning: input error: all cost and volume data must be non-negative and each of the positions on the upper left-lower right diagonal must be zero in one or the other (or both) of the matrices (these positions deal with traffic from dept I to dept I, etc., and logically should be zero).

COST -

1) Message: ICOST OR JCOST NEG OR GREATER THAN NDEPT

- meaning: program asked to compute costs involving an illegal dept number.

EXCH & EXCT -

1) Message: EXCH PARAMETER NEG OR GREATER THAN NDEPT
EXCT PARAMETER NEG OR GREATER THAN NDEPT

- meaning: program asked to exchange invalid dept numbers.

PERIM & PICKUP -

1) Message: PERIM GOOF
PICKUP GOOF

- meaning: in scanning rectangle supposed to contain old larger dept, PERIM or PICKUP failed to find all the squares in the dept it was hunting for.

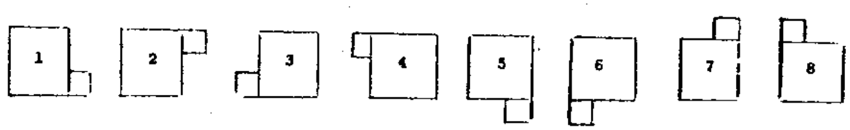
VALID -

1) Message: JVALID NEG OR GREATER THAN NDEPT

- meaning: VALID asked to check the shape of an illegal (nonexistent) department. Is not concerned with validity of department shape.

SETUP

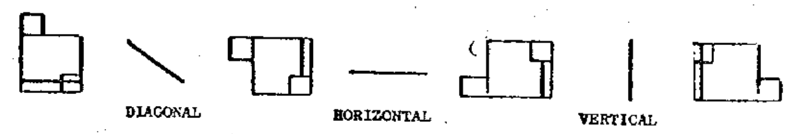
In carving the smaller department out of the larger, we naturally want to carve it from the most distant side or corner in order to obtain the maximum change in location. One way to do this would be separate routines for each orientation of departments (see below). CRAFT takes a different approach. The carving routines are written to operate only on case 1 - the upper left corner for square carving and the left side (followed by the top side if the left side fails), for rectangular carving. SETUP makes it possible to use these routines for all cases. It analyzes the relative locations of the department centers to determine which of the basic orientations is involved.



Basic Orientations (The numbers correspond to values of IAXD in the program)

It then moves the larger department to temporary arrays at the same time rotating it so that it conforms to case 1 - the "far" side to the left and the "far" corner upper left. (It is unnecessary to actually move the smaller

department to a temporary array since no carving is done on it, but the center coordinates are rotated, for purposes explained below.) Cases 5-8 must first be flipped around an upper-left to lower-right diagonal axis; then they correspond to cases 1-4. Cases 2 & 6 must be flipped around a horizontal axis; 3 & 7 around a vertical axis, and 4 & 8 around both. (In the program, one cell is moved at a time and all rotations are performed in a single series of instructions.)



The rotation of case 8 into case 1; "far" side and corner marked

Since the axes used for rotation are the axes of the entire 32 x 32 array, rather than the axes of the department involved, the department may also be rotated into a different general area of the temporary arrays than it occupies in ISP. However, these arrays are zero except for this one department and the program handles this with no difficulty. SETUP determines the boundaries of the rectangle which contains the department in the temporary arrays and passes these to MESSR, PERIM, and PICKUP which use them to minimize their searching.

For a department which is not a simple rectangle, we have one further difficulty: which upper left corner shall we use as a base for the square carving routine. The program compares the rotated coordinates of the smaller department center with the coordinates of each cell placed into the temporary arrays and finds the cell that is the greatest total distance above and to the left of the smaller department center, and passes this information to the square carving routine.

(Note: For two-way exchanges, the words "larger" and "smaller" above apply strictly; for three-way exchanges, SETUP is used more deviously (see write-up of EXCT), but the intent is always the same: Rotate the area which we are going to carve so that the corner and side where we want to begin carving are the Northwest corner and West side.) (Where North and West conform to conventional map orientations, N at the top of thepage or more precisely the top row of the array.)

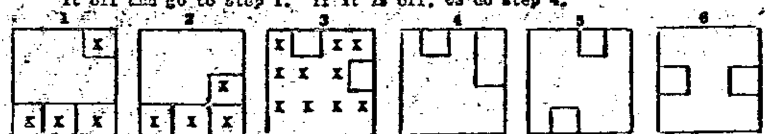
VALID

VALID checks to see that the squares labeled as being in department I do in fact form a reasonably shaped department. Obviously, we want to reject those departments which are not in one piece or which touch only at a corner (such as shapes 1 and 2 below). We also reject departments which have indentations along both axes (such as shape 3). Missing corners and multiple indentations along a single axis are accepted (shapes 4-6 below). The routine proceeds as follows, operating within the rectangle which encloses the department.

1) Start at top of column (left column initially), scan down until we find a square in the department.

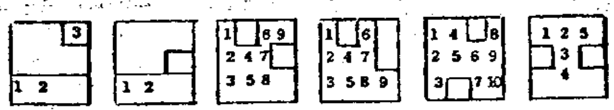
2) Scan on down the column only as long as we continue to find squares in the department. We count each square we find in the department, and we check the square to the immediate right of each square to see if it is also in the department. If it is we set the next column indicator on. This tells us in effect, that the department is connected; that at least one square in the department in this column is adjacent to a square in the next column that is also in the department.

3) When we come to a square that is not in the department or to the bottom of a column, we check the next column indicator. If it is on, we turn it off and go to step 1. If it is off, we do step 4.

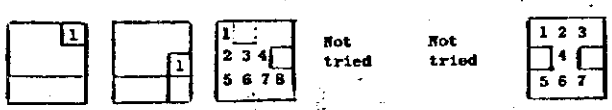


4) Compare the number of squares in the department with the number we have counted. If they agree the shape is valid and we return. If they do not, we try a horizontal scan, going across rows, and checking the square just below, etc. If this count agrees, the shape is valid. If this fails also, the shape is INVALID and we set a failure code before returning. The working of the scan is illustrated below. The numbered squares are the squares which are counted in the order counted.

Sqs counted in vertical scan (in order)



Sqs counted in horizontal scan (in order)



Result	INVALID	INVALID	INVALID	VALID	VALID	VALID

INTERDEPARTMENT PRODUCT FLOW

	A	B	C	D	E	F
A	0.	0.	0.	0.	0.	20.000
B	0.	0.	0.	0.	0.	30.000
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	0.	0.	0.	0.	0.	10.000
F	20.000	30.000	0.	0.	10.000	0.

INTERDEPARTMENT MOVE COST
PER UNIT LOAD PER UNIT DISTANCE

	A	B	C	D	E	F
A	1.000	1.000	1.000	1.000	1.000	1.000
B	1.000	1.000	1.000	1.000	1.000	1.000
C	1.000	1.000	1.000	1.000	1.000	1.000
D	1.000	1.000	1.000	1.000	1.000	1.000
E	1.000	1.000	1.000	1.000	1.000	1.000
F	1.000	1.000	1.000	1.000	1.000	1.000

COVOL = (MOVE COST/LOAD)X(NO. OF LOADS)

	A	B	C	D	E	F
A	0.	0.	0.	0.	0.	20.000
B	0.	0.	0.	0.	0.	30.000
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	0.	0.	0.	0.	0.	10.000
F	20.000	30.000	0.	0.	10.000	0.

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	B	B	D	D	D	D	D	D	D	D	D
3	A	B	B	D	D	D	D	D	D	D	D	D
4	A	C	C	E	E	E	E	E	E	E	E	E
5	A	C	C	E	F	F	F	F	F	F	F	F
6	A	C	C	E	F							F
7	A	C	C	E	F							F
8	A	C	C	E	F							F
9	A	C	C	E	F							F
10	A	C	C	E	F							F
11	A	C	C	E	F							F
12	A	C	C	E	F	F	F	F	F	F	F	F

TOTAL COST 1185.73 EST. COST REDUCTION 0. MOVEA MOVEB MOVEC ITERATION 0

00005

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2........			0.54783E 02	0.54783E 02
1	3	1	3........			0.11043E 03	0.11043E 03
2	3	2	3........			0.33000E 03	0.33000E 03
1	4	2	3........			0.33000E 03	0.11043E 03
2	4	2	3........			0.33000E 03	0.33000E 03
3	4	2	3........			0.33000E 03	-0.
3	5	2	3........			0.33000E 03	-0.34706E 02
4	5	2	3........			0.33000E 03	-0.34706E 02
5	6	5	6........			0.47647E 03	0.47647E 03

00006

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	B	B	D	D	D	D	D	D	D	D	D
3	A	B	B	D	D	D	D	D	D	D	D	D
4	A	C	C	F	F	F	F	F	F	F	F	F
5	A	C	C	F								F
6	A	C	C	F								F
7	A	C	C	F							F	F
8	A	C	C	F				F	F	F	F	E
9	A	C	C	F				F	E	E	E	E
10	A	C	C	F				F	E			E
11	A	C	C	F				F	E			F
12	A	C	C	F	F	F	F	F	E	E	E	E

TOTAL COST 984.26 EST. COST REDUCTION 476.47 MOVEA E MOVEB F MOVEC ITERATION 1

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2	....	..	0.54783E 02	0.54783E 02
1	3	1	3	....	..	0.11043E 03	0.11043E 03
2	3	2	3	....	..	0.33000E 03	0.33000E 03
1	4	2	3	....	..	0.33000E 03	0.11043E 03
2	4	2	3	....	..	0.33000E 03	0.33000E 03
3	4	2	3	....	..	0.33000E 03	-0.
3	5	2	3	....	..	0.33000E 03	-0.34706E 02
4	5	2	3	....	..	0.33000E 03	-0.34706E 02
5	6	5	6	....	..	0.47647E 03	0.47647E 03

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	B	B	D	D	D	D	D	D	C	C	D
3	A	B	B	D	D	D	D	D	D	D	D	D
4	A	C	C	F	F	F	F	F	F	F	F	F
5	A	C	C	F								F
6	A	C	C	F								F
7	A	C	C	F							F	F
8	A	C	C	F				F	F	F	F	E
9	A	C	C	F				F	E	E	E	E
10	A	C	C	F				F	E			E
11	A	C	C	F				F	E			E
12	A	C	C	F	F	F	F	F	E	E	E	E

TOTAL COST 984.26 EST. COST REDUCTION 476.47 MOVEA E MOVEB F MOVEC ITERATION 1

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2........			0.39783E 02	C.39783E 02
1	3	1	2........			0.39783E 02	0.30435E 02
2	3	2	3........			0.16500E 03	0.16500E 03
1	4	2	3........			0.16500E 03	0.55435E 02
3	4	2	3........			0.1650CE 03	-0.
2	6	2	3........			0.16500E 03	-0.14170E 03
3	6	3	6........			0.22043E 03	0.22043E 03
4	6	3	6........			0.22043E 03	-0.35957E 03
5	6	3	6........			0.22043E 03	-0.26801E 03

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	F	F	D	D	D	D	D	D	D	D	D
3	A	F	F	D	D	D	D	D	D	D	D	D
4	A	F	F	F	F	F	F	F	F	C	C	C
5	A	F						F	C	C		C
6	A	F						F	C			C
7	A	F						F	C		C	C
8	A	F						F	C	C	C	E
9	A	F						F	E	E	E	E
10	A	F	F	F				F	E			E
11	A	B	B	F				F	E			E
12	A	B	B	F	F	F	F	F	E	E	E	E

TOTAL COST 759.26 EST. COST REDUCTION 220.43 MOVEA C MOVEB F MOVEC ITERATION 3

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2........			0.39783E 02	0.39783E 02
1	4	1	2........			0.39783E 02	-0.12457E 03
3	4	1	2........			0.39783E 02	-0.
3	5	1	2........			0.39783E 02	0.33824E 02
1	6	1	2........			0.39783E 02	-0.23272E 03
2	6	1	2........			0.39783E 02	-0.18670E 03
3	6	1	2........			0.39783E 02	-0.49000E 03
4	6	1	2........			0.39783E 02	-0.53957E 03
5	6	1	2........			0.39783E 02	-0.49301E 03

ANAN - NO COST REDUCTION - BCOST = 0.10494E 04 TCOST = 0.75926E 03 MOVEA 2 MOVEB 1

I	J	MA	MB	M	N	ACOST	BCOST
1	4	2	1	2	1	0.	-0.12457E 03
3	4	2	1	2	1	0.	-0.
3	5	3	5	2	1	0.33824E 02	0.33824E 02
1	6	3	5	2	1	0.33824E 02	-0.23272E 03
2	6	3	5	2	1	0.33824E 02	-0.18670E 03
3	6	3	5	2	1	0.33824E 02	-0.49000E 03
4	6	3	5	2	1	0.33824E 02	-0.53957E 03
5	6	3	5	2	1	0.33824E 02	-0.49301E 03

LOCATION PATTERN

	1	2	3	4	5	6	7	8	9	10	11	12
1	A	A	A	A	A	A	A	A	A	A	A	A
2	A	F	F	D	D	D	D	D	D	D	D	D
3	A	F	F	D	D	D	D	D	D	D	D	D
4	A	F	F	F	F	F	F	F	F	E	E	E
5	A	F						F	E	E		E
6	A	F						F	E			E
7	A	F						F	E	E	E	E
8	A	F						F	E	E	C	C
9	A	F						F	C	C	C	C
10	A	F	F	F				F	C			C
11	A	B	B	F				F	C			C
12	A	B	B	F	F	F	F	F	C	C	C	C

TOTAL COST 727.20 EST. COST REDUCTION 33.82 MOVEA E MOVEB C MOVEC ITERATION 4

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2	2	1	0.39783E 02	0.39783E 02
1	4	1	2	2	1	0.39783E 02	-0.12457E 03
3	4	1	2	2	1	0.39783E 02	-C.
3	5	1	2	2	1	0.39783E 02	-0.29902E 02
4	5	1	2	2	1	0.39783E 02	-0.18235E 02
1	6	1	2	2	1	0.39783E 02	-0.17301E 03
2	6	1	2	2	1	0.39783E 02	-0.30582E 03
3	6	1	2	2	1	0.39783E 02	-0.44951E 03
4	6	1	2	2	1	0.39783E 02	-0.47986E 03
5	6	1	2	2	1	0.39783E 02	-0.57066E 03

ANAN - NO COST REDUCTION - BCOST = 0.10173E 04 TCOST = 0.72720E 03 MOVEA 2 MOVEB 1

I	J	MA	MB	M	N	ACOST	BCOST
1	4	2	1	2	1	0.	-0.12457E 03
3	4	2	1	2	1	0.	-0.
3	5	2	1	2	1	0.	-0.29902E 02
4	5	2	1	2	1	0.	-0.18235E 02
1	6	2	1	2	1	0.	-0.17301E 03
2	6	2	1	2	1	0.	-0.30582E 03
3	6	2	1	2	1	0.	-0.44951E 03
4	6	2	1	2	1	0.	-0.47986E 03
5	6	2	1	2	1	0.	-0.57066E 03

INTERDEPARTMENT PRODUCT FLOW

	A	B	C	D	E	F
A	0.	0.	0.	0.	50.000	0.
B	0.	0.	0.	0.	0.	0.
C	C.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	60.000	0.	0.	0.	0.	0.
F	0.	C.	0.	0.	0.	0.

INTERDEPARTMENT MOVE COST
PER UNIT LOAD PER UNIT DISTANCE

	A	B	C	D	E	F
A	1.000	1.000	1.000	1.000	1.000	1.000
B	1.000	1.000	1.000	1.000	1.000	1.000
C	1.000	1.000	1.000	1.000	1.000	1.000
D	1.000	1.000	1.000	1.000	1.000	1.000
E	1.000	1.000	1.000	1.000	1.000	1.000
F	1.000	1.000	1.000	1.000	1.000	1.000

COVOL = (MOVE COST/LOAD)X(NO. OF LOADS)

	A	B	C	D	E	F
A	0.	0.	0.	0.	50.000	0.
B	0.	0.	0.	0.	0.	0.
C	0.	0.	0.	0.	0.	0.
D	0.	0.	0.	0.	0.	0.
E	60.000	0.	0.	0.	0.	0.
F	0.	0.	0.	0.	0.	0.

LOCATION PATTERN

	1	2	3	4	5	6	7
1	A	A	B	B	B	B	B
2	A	A	B	B	B	B	B
3	C	C	D	D	D	D	B
4	C	C	D	F	D	D	E

TOTAL COST 880.00 EST. COST REDUCTION 0. MOVEA MOVEB MOVEC ITERATION 0

I	J	MA	MB	M	N	ACOST	BCOST
4	5	4	5	2	1	0.33000E 03	0.33000E 03

EXCH HAS FAILED MOVEA 5 MOVEB 4

00(21

I J MA MB M N ACOST BCOST

00(22

INTERDEPARTMENT PRODUCT FLOW

	A	B	C	D
A	0.	0.	20.000	0.
B	0.	0.	0.	0.
C	40.000	0.	0.	0.
D	0.	0.	0.	0.

INTERDEPARTMENT MOVE COST
PER UNIT LOAD PER UNIT DISTANCE

	A	B	C	D
A	0.	0.	2.000	0.
B	0.	0.	0.	0.
C	3.000	0.	0.	0.
D	0.	0.	0.	0.

COVOL = (MOVE COST/LOAD)X(NO. OF LOADS)

	A	B	C	D
A	0.	0.	40.000	0.
B	0.	0.	0.	0.
C	120.000	0.	0.	0.
D	0.	0.	0.	0.

LOCATION PATTERN

	1	2	3	4	5	6
1	A	A	B	B	B	D
2	A	A	B	B	D	D
3	A	A	B	B	D	D
4	A	B	B	B	C	C
5	A	B	B	B	C	C

TOTAL COST 960.00 EST. COST REDUCTION 0. MOVEA MOVEB MOVEC ITERATION 0

I	J	MA	MB	M	N	ACOST	BCOST
1	2	1	2	5	4	0.39385E 03	0.39385E 03
2	3	2	3	5	4	0.56615E 03	0.56615E 03

LOCATION PATTERN

	1	2	3	4	5	6
1	A	A	B	B	B	D
2	A	A	B	B	D	D
3	A	A	B	B	D	D
4	A	C	C	B	B	B
5	A	C	C	B	B	B

TOTAL COST 480.00 EST. COST REDUCTION 566.15 MOVEA C MOVEB B MOVEC ITERATION 1

I	J	MA	MB	M	N	ACOST	BCOST
1	2	3	2	5	4	0.	-0.36923E 02
1	3	3	2	5	4	0.	0.
2	3	3	2	5	4	0.	-0.61538E 02