Digital Computer Laboratory
Massachusetts Institute of Technology
Cambridge 39, Massachusetts

To: Scientific and Engineering Computation Group

From: Jack B. Dennis

Date: August 28, 1956

SUBJECT: A PROGRAM TO SOLVE THE CLASSICAL TRANSPORTATION PROBLEM

A program has been written to solve the classical transportation problem on the Whirlwind computer using the stepping stone method.[1]

The transportation problem may be stated as follows: A company operates m plants producing a commodity, the i^{th} of which can supply S_i units of the commodity. The company sells its production to n customers, the j^{th} of which desires D_j units of the commodity. The cost of manufacturing and transporting a unit of the commodity from plant i to customer j is C_{ij}. It is desired to find the number of units X_{ij} that should be shipped from each plant to each customer, in order to have the total cost of the operation be a minimum.

Thus the problem may be stated mathematically as:

$$\text{minimize } C = \sum_{i,j} C_{ij} X_{ij} \qquad (1)$$

subject to the constraints

$$\sum_j X_{ij} = D_j \qquad (2)$$

$$\sum_i X_{ij} = S_i \qquad (3)$$

$$X_{ij} \geq 0 \qquad (4)$$

This is a special case of the general linear programming problem.

A set of X_{ij} which satisfies equations 2, 3, and 4 is called a feasible solution to the problem. If the set also minimizes C, it is an optimum feasible solution containing, in general, exactly m+n-1 non-zero X_{ij}.

The process of solution consists of:

1) generating a feasible solution having exactly m+n-1 non-zero X_{ij}.

MIT COMPUTATION CENTER
DOCUMENT ROOM

2) finding a zero X_{ij} which, if allowed to be positive, would yield a decrease in C. In the process of increasing this X_{ij}, one of the non-zero X_{ij} must go to zero in order that equations 2 and 3 remain satisfied. Thus the new feasible solution will again have exactly m+n-1 non-zero X_{ij}.

Step 2 is repeated until there are no zero X_{ij} which can be changed so as to reduce C. The set of X_{ij} is then an optimum feasible solution.

The Whirlwind program is unique in that a logical search procedure is used in entering the new element into the feasible solution whereas other transportation codes use auxiliary tables through which many passes must be made for each step towards the solution. Also, in finding which zero X_{ij} should be made positive, the usual procedure is to search through all of the cost data to find the X_{ij} which could be made non-zero to the greatest advantage. The Whirlwind program takes the first X_{ij} found that would lower the total cost if made non-zero. This avoids much of the slow and time consuming process of searching through cost data at the expense of a greater number of iterations.

Because of these features this program is considerably faster than other procedures.[2] A number of problems with 9 plants and 69 customers have been solved in about 1.5 minutes of computer time each. The average number of iterations was 150. Also, seven problems with $m = 60$, $n = 291$, have been solved requiring about 45 minutes and 1800 iterations each.

The program will handle problems with $m \leq 127$, $m+n \leq 401$. In many practical problems not all mn possible C_{ij} are significant. No error will result if the costs for shipping routes that are known to be absurd or impractical are assumed to be infinite. The program has therefore been arranged to consider as infinite all costs not specified in the data. A considerable saving in storage is made this way. Problems with up to 10,000 significant costs can be solved by the program.

The appendix to this report contains detailed instructions for the preparation of data tapes for the transportation routine and directions for its use.

References:

1. A. Charnes and W.W. Cooper, "The Stepping Stone Method of Explaining Linear Programming Calculations in Transportation Problems," Management Science, vol. 1, no. 1, October 1954.
2. For example, see C.W. Swift and S. Poley, "The Transportation Problem," IBM Technical Newsletter No. 10, October 1955.

Appendix

Contents:

A. Data Tape Preparation

The data that must be supplied to the routine to solve a problem includes:

a. the size of the problem (m and n)

b. the cost data C_{ij}

c. Supply capacities S_i

d. Customer demands D_j

The data tape for a sample problem is given in Fig. A-1. Specifications for the data tape are given below. The symbol →| denotes a tabulation, ↵ denotes a carriage return. The important parts of the data tape are discussed in their order of appearance:

1.

"fc TAPE xxx-xx-xxxxx"

Tape number and title in conventional CS II format.

2.

"32 a.boocd" Output Code

a = 0	Cost data C_{ij} are supplied on this tape.
a = 1	Cost data of the previous problem are to be used.
b = 0	Supplies and Demands are given on this tape. The computer will stop after reading the data tape.
b = 4	Supplies and Demands are not given on this tape. The computer will stop after reading the data tape.

c = 0 Output will be punched on paper tape.

c = 1 Output will be printed.

d = 0 Final basis only.

d = 2 Initial and final bases.

d = 3 Initial and final bases and step data.

d = 4 Final basis and alternate solution elements.

d = 6 Initial and final bases and alternate solution elements.

d = 7 Initial and final bases, step data and alternate solution elements.

3.

"START AT 56"

(One and only one carriage return!)

4.

"Sample Problem No. 000"

Title of problem. (This title will appear on the results.) The last character of the title line must be a carriage return. Any flexowriter characters except a carriage return may be used elsewhere in the title.

*

5.

"+ m + n "

Size of Problem

---

*Between the asterisks the following characters are ignored by the computer: =, +, color shift, upper case, lower case, space, back space, stop character and delete. All others are <u>illegal</u> except for the digits 0 and 9 and characters specifically mentioned. Also a word enclosed in parenthesis will be ignored, i.e. "(costs for plant 2)." Any flexowriter characters may be used in the word except tabulation or carriage return. The "comment" word must be followed by a tabulation or carriage return.

m = Number of supplies ≤ 127

n = Number of demands $\leq 401\text{-}m$

6.

Cost data C_{ij} if included in this tape. There are four legal word types which must be separated by at least one tabulation or carriage retnrn.

a) "+ .08964"

Cost data word. Plus sign and decimal point are ignored. The decimal point is assumed to be at the extreme left, hence numbers must be less than +1.0. The numbers, which are assumed to be positive, must have no more than 9 digits including zeros. The first cost data word is assumed to be C_{11} unless a j assignment word preceeds.

b) "j = 28"

j assignment word. The next cost data word will be assumed to be for column 28 in this row. The j value specified must be less than or equal to n and must not be zero. The equals sign may be omitted.

c) "|"

Row termination word. Denotes the end of this row. The next cost data word will be assumed to be for column one in the next row unless a j assignment word intervenes.

d) "x"

Termination word. Denotes the end of the last row of cost data. (The vertical bar may not be used to terminate the last row.)

For any row of C_{ij}, twice the number of cost data words plus the number of j assignment words must be less than or equal to 2n. The total amount of cost data must be less than 13,888 registers, that is:

$$2c + m + j \leq 13{,}888 \qquad \text{(A-1)}$$

Here, c is the number of cost data words, j is the number of j assignment words and m is the number of rows in the C_{ij} matrix. Costs which are omitted from the data tape are assumed to be "infinite" (+1.0) by the program.

7.

Supplies and Demands, S_i and D_j if included in this tape. There are three legal word types which must be separated by at least one tabulation or carriage return.

a) Supply or Demand word. Same specifications as for cost data word above. The first m such words will be the supplies, $S_1 \ldots S_m$. The next n such words will be the demands $D_1 \ldots D_n$.

b) "|"
Denotes the end of the supply data words.

c) "x"
Denotes the end of the demand data words.
* (See note, Page 5)

8.

"fc2
START AT 02 "

End of tape. (The "fc" may be the beginning of another data tape which will be processed automatically after the first tape.) There must be <u>no</u> flexowriter character other than feed out between the "x" terminating the data and the "fc". About one inch of feed out is necessary. One carriage return must follow "START AT 0".

B. Output Format

Fig. A-2 shows the output for the sample problem. The "Total supply" and "Total demand" are given as a check on the data. They should not be expected to agree exactly because of error in converting from decimal to binary.

The "Initial basis table" gives the location and value of the X_{ij} in the initial feasible solution. The U's and V's are extra variables that are useful in obtaining the solution. Their values are such that $U_i+V_j = C_{ij}$ for all combinations of i and j for which X_{ij} is non-zero in a feasible solution. They are arranged in order by rows.

The step data gives for each step the row and column of the non-zero X_{ij} dropped and the non-zero X_{ij} added to the feasible solution, the value of the new X_{ij} and the incremental negative cost of making the change.

The "Initial cost" is the sum of $C_{ij}X_{ij}$ for the initial feasible solution. The "Total decrease" is the sum of the decreases in cost at each step. The "Final cost" is obtained by subtraction. The "Summed cost" is given for comparison and is the sum of $C_{ij}X_{ij}$ for the optimum solution. This will not be exactly equal to the final cost because of round-off error in multiplication.

The number of "Iterations" is the number of steps required to obtain the optimum solution from the initial solution. The number of "Passes" is the number of times the whole set of cost data had to be searched to find all of the zero X_{ij} that could be brought into the solution profitably.

The "Alternate elements" are those shipping routes which, if used, would not increase the total cost.

C. Performance Requests: Operator Instructions

Once data tapes have been prepared, the problem solutions may be obtained by following the instructions below which are given in the standard form for the Whirlwind machine:

1) si 1 switch off	
2) e, fb 219 - 68 - 2000, ri	Program tape
3) fc xxx-xx-xxxx, ri - - - - - fc xxx-xx-xx, ri	Data tapes

As many data tapes may be run as desired on a single read-in of the program tape. One read-in of a set of cost data may be used for any number of solutions. However, supply and demand data are destroyed by the routine and, hence, must be given for each run.

If the quantity 2c + m + j, in Equation A-1 of section A on data tape preparation, is greater than 13,888 but less than 20,480 the problem can still be handled by transferring the program to the buffer drum. The operating procedure is as follows:

1) si 1 switch off	
2) e, fb 219 - 68 - 2000, ri	Program tape
3) fb 219 - 68 - 3000, ri	Tape to transfer program to buffer drum.
4) fc xxx-xx-x - - -	Data tapes

If this procedure is used, parts 1, 2, 3 and 8 of the data tape must be in 556 binary form. Otherwise the transportation routine on the buffer drum will be destroyed by the CS II conversion program.

Performance Requests: Input-Output Equipment

The amount of input-output equipment used by the program is as follows:

Direct typewriter - two lines for each set of supplies and demands read in.

Auxiliary Storage Drum - groups 0 - 10.

Magnetic Tape - unit #3. The amount of magnetic tape used may be found from the following equation.

Number of feet = N

$$\simeq \frac{1}{24}\left[(1 + a)(m + n)(18 + 2\alpha) + b(25 + 2\alpha)\beta + c(9\gamma) + 500\right] \quad \text{(A-2)}$$

Where: a = 1 if initial basis is requested

b = 1 if step data is requested

c = 1 if alternate elements are requested

α = Maximum number of digits to the right of decimal point in the data.

β = Number of iterations.

γ = Number of alternate elements.

Performance Requests: Time Requirements

The approximate time required to solve a given problem may be estimated from Fig. A-3. About one minute is required to read in the program tape.

Performance Requests: Alarms

There are three possible print-outs on the direct typewriter that can result from a faulty data tape. Their possible causes are listed below:

1. "wrong count"

a. Too much cost data was given in this row of the C_{ij} matrix.

b. Exactly m rows were not given for the cost matrix.

c. Exactly m supplies and exactly n demands were not given.

2. "illegal character"

a. The output code specified in part 2 of the data tape was illegal.

b. An illegal flexowriter character was used improperly.

3. "number too long"

a. More than 3 digits appeared in m or n.

b. A j assignment value greater than n was used.

c. More than 9 digits appeared in a cost or a supply or demand word.

Figure A-1

```
fc TAPE 219 - 68 - 000 DENNIS
32  0.00007

START AT 56
Whirlwind Transportation Routine - Sample Problem

+5      (m)
+8      (n)

(Cost Matrix)
012     009     011     010     007     014     008
003     014     008     011
j=3     008     007     004     j=7     013     010
j=2     010     009     015     006     008     004     005
005     007     011     j=6     011     009     013       x

(Supply)
25      16      06      19      10

(Demand)
04      08      05      20      06      03      26      04
xfc

START AT 0
```

Fig. A-2

Whirlwind Transportation Routine - Sample Problem

Total supply .7599
Total demand .7599

Initial basis table

i	j	x	i	j	x	i	j	x
2	1	.0399	5	2	.0799	2	3	.0499
2	4	.0000	1	5	.0599	4	6	.0099
1	7	.1899	4	8	.0399	2	7	.0699
3	4	.0599	4	4	.1399	5	6	.0199

U

+ .0000	+ .9915	+ .9875	+ .9955	+ .9985

V

- .9885	- .9915	- .9835	- .9805	+ .0069
- .9875	+ .0080	- .9905		

Step data

i1	j1	i2	j2	x	c
3	4	3	5	.0599	.9905
2	3	4	3	.0499	.0030
1	5	4	5	.0000	.9965
2	7	4	7	.0099	.9995
5	6	5	1	.0199	.0049
2	1	1	2	.0199	.0040
4	3	1	3	.0499	.0020
4	4	1	4	.0400	.0090
4	5	1	5	.0000	.0030
1	2	2	1	.0199	.0049
1	3	2	3	.0499	.0039
5	1	5	6	.0199	.0040

Initial cost .076005815
Total decrease .070595821
Final cost .005409994

Final basis table

i	j	x	i	j	x	i	j	x
2	1	.0399	5	2	.0799	2	3	.0499
2	4	.0700	1	5	.0000	4	6	.0099
1	7	.1199	4	8	.0399	4	7	.1399
3	5	.0599	1	4	.1299	5	6	.0199

Summed cost .005409994

U

+ .9745	+ .9755	+ .9715	+ .9705	+ .9735

V

- .9725	- .9665	- .9675	- .9645	- .9675
- .9625	- .9665	- .9655		

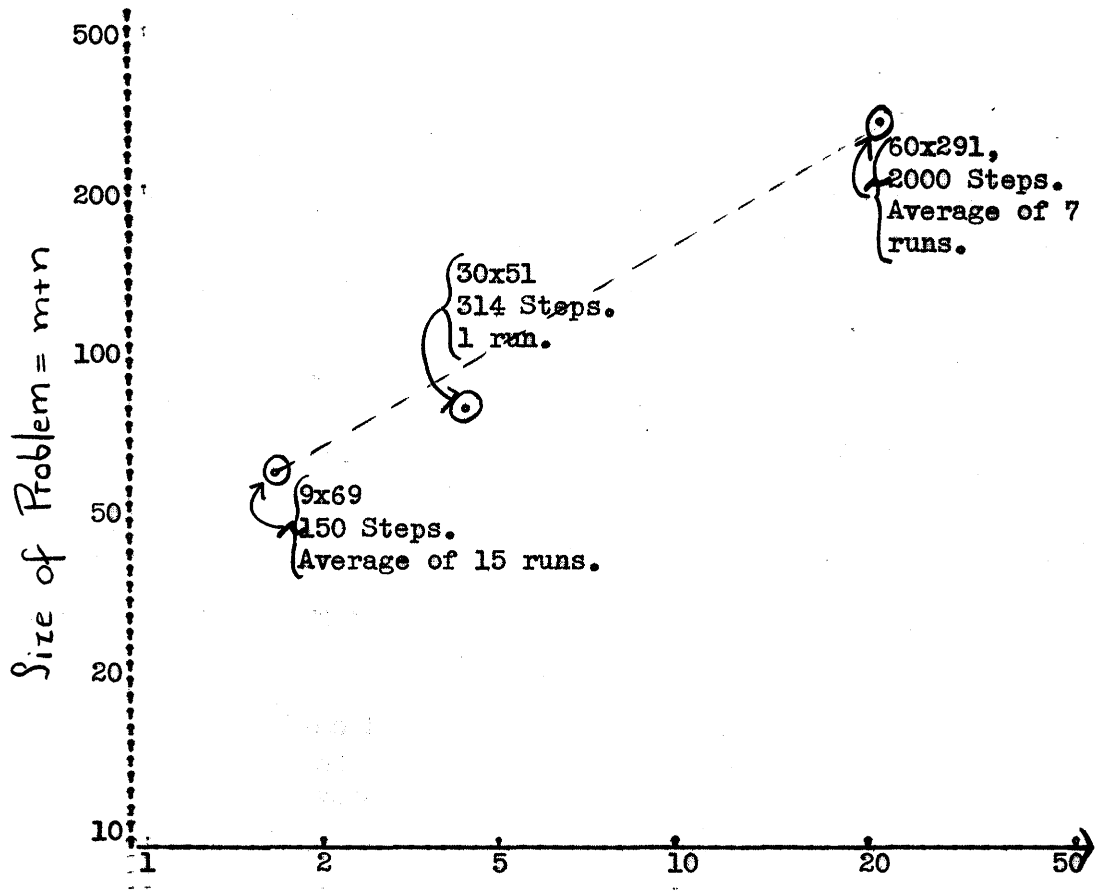
Iterations 12
Passes 3

Alternate elements

i	j	i	j	i	j
3	4				

Time Required for Solution

Fig. A-3



Time for Solution in Minutes