OPTIMAL SEQUENCING OF SERIAL MEMORY TRANSFERS

S. M. Johnson
Mathematical Analysis Department
The RAND Corporation

P-2151

December 5, 1960

SUMMARY

The problem of finding an optimal sequence of word transfers on a serial memory drum was discussed and solved iteratively by Arthur Gill [1]. An improved method is presented here which requires only a few simple adjustments after the initial trial sequence is formed. Mathematically it is of interest in that it is a special form of a traveling salesman type problem which can easily be solved.

OPTIMAL SEQUENCING OF SERIAL MEMORY TRANSFERS

S. M. Johnson

INTRODUCTION

In a recent paper [1] Arthur Gill describes an iterative listing technique for finding the optimal sequence of word transfers on a memory drum of a digital computer in the sense of minimizing the total distance the drum travels. Gill shows that the problem is of the traveling salesman type. He proposes a systematic listing of all infeasible routes according to increasing total length until a feasible route or tour is found. This would involve considerable work for a problem of some size.

The purpose of the present note is to show that by utilizing the special structure of the distance matrix this traveling salesman problem can be restated in terms of an equivalent distance matrix of zeros and ones only. This leads to an almost immediate construction of an optimal solution.

DESCRIPTION OF THE PROBLEM

We wish to find an optimal sequence of word transfers on a memory drum of a digital computer. For a more detailed technical description see [1].

The drum has a capacity of N words. There are K words to be read in K different initial positions and transferred one at a time to their terminal positions. We seek a sequence of word transfers which minimizes the total distance the drum must rotate from a given entry position to a given exit position.

Consider a circular array of positions numbered clockwise from 1 to N. Let S_i indicate the initial position of the i^{th} word to be transferred to its terminal position F_i. Let F_0 be the entry position where the transfer program starts, and let S_0 be the exit position where the transfer program terminates.

We wish to start at F_0 moving clockwise to pick up the i^{th} word at S_i, transfer it to F_i, then to S_j where the j^{th} word is picked up and transferred to F_j, etc. until all the words are transferred. We then move around to S_0 and are through. We wish to find a sequence of word transfers which minimizes the total distance traveled.

As Gill mentions in [1], there may be time lag restrictions due to machine instructions of reading and recording, etc.

Let the coding instructions require r_i word-times or bits to read the i^{th} word and q_i bits to record or write it. Thus if S_i follows F_i by less than r_i, the drum must make a complete rotation from F_i before it could handle the j^{th} word.

Define D_{ii} as the distance the drum moves from S_i to $F_i - q_i$ to F_i. Likewise define D_{ij} as the distance from F_i to $S_j - r_j$ to S_j.

We want to find the path from F_0 to S_a to S_b to ... to S_0 which minimizes the total distance. Since $\Sigma\, D_{ii}$ is common to all such paths we can minimize the path

$$(A) \qquad D_{0a} + D_{ab} + \dots + D_{m0}$$

over all permutations of subscripts.

Gill in [1] proposes a systematic listing technique which applies to any general traveling salesman type problem but could lead to lengthy calculations even when done by an electronic computer.

We propose a special technique for the problem at hand which utilizes the special properties of the distance matrix and which gives an optimal tour almost immediately.

By adding the common value D_{00} to the path (A) we close the loop, making the total distance an integral multiple of n, i.e. so many complete rotations of the drum.

To count these rotations for a given tour choose an arbitrary fixed reference point P on the drum circle.

Define an equivalent distance matrix whose entries

$$d_{ij} = 1 \quad \text{or} \quad 0$$

depending on whether P is passed or not while moving from F_i to $S_j - r_j$. Similarly for d_{ii} but since we are not interested in the values d_{ii} common to all tours we label each $d_{ii} = x$. d_{ii} includes the count of passing P as the drum moves from $S_i - r_i$ to S_i to $F_i - q_i$ to F_i. Thus $d_{ii} = 0$, 1 or 2.

It is now clear that only the relative position of the points on the circle matters. Since the counting point P can be placed anywhere, for convenience we place it immediately preceding the greatest accumulated excess of F's over the $S_i - r_i$'s. We want a tour with minimum total count $W = d_{0a} + d_{ab} + \ldots + d_{m0}$.

For expository reasons, renumber the S's in order of their positions after P, and renumber the corresponding F's accordingly. Thus we are lead to a typical distance matrix (d_{ij}) as in Figure 2 below.

Note that excluding the d_{ii} each row consists of a sequence of ones followed by a sequence of zeros. The square matrix is divided by a step-like line into a region of zeros and a region of ones. The region of ones all lie below the main diagonal from upper left to lower right as a consequence of our choice of P.

We are now in a position to construct an optimal tour. Remember the columns now number 1, 2, ..., K + 1.

<u>Step</u> 1. Choose the last zero in the first column. Then choose d_{ij} for the first zero entry in the i^{th} row, etc. Until a cycle of zero entries is reached. If this cycle is a (K + 1) cycle then it is an optimal tour with total W = 0.

If it is only a subcycle of length less than K + 1, start another cycle in the same way from the rows and columns not yet chosen.

In this way we may get several subcycles with total count $W_1 + W_2 + \ldots = 0$. Each row and column is chosen once.

<u>Step</u> 2. If there is a rectangle with zero entries in the opposite corners from two corners which are chosen entries belonging to two different subcycles, make the obvious interchange to the other two corners, thus reducing the number of subcycles with no increase in total count W. (If a cycle of

length one had been formed from d_{ii}, the cycle can be easily combined with another cycle in this way leaving just zero entries.) Repeat as far as possible to reduce the number of subcycles with total W = 0.

If a tour of zero entries is reached, it is clearly an optimal solution. Furthermore, if a zero tour exists, it can be formed by a set of cyclic adjustments to the given non-tour. Each cyclic adjustment can be broken up into a set of rectangular interchanges of the type considered in Step 2. Thus in Figure 1 below the squares represent the old zero choices, the circles represent the new zero choices in a particular cyclic adjustment. Then the diamond must represent a zero entry since no 1 can lie between two 0's in the matrix (d_{ij}).

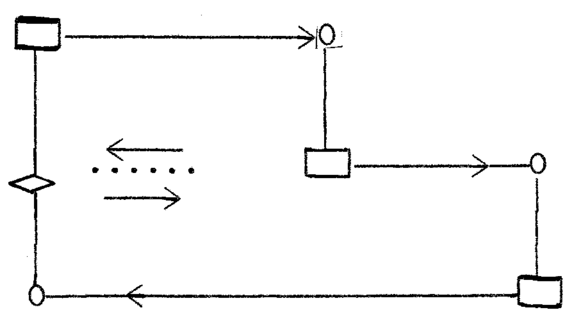

Figure 1

Thus Step 2 gives a zero tour if such a tour exists.

Step 3. If Step 2 no longer applies, we can combine all the remaining subcycles into a tour with W = 1 as follows.

First label the columns according to their subcycles. Then shift the entry choice for the first column in subcycle a to the entry in the same row but in the first column of subcycle b. Then shift from the original entry in the first column of subcycle b to the entry in the same row but in the first column of subcycle c, etc. All of these interchanges involve just zero entries since the shift is to the right from a zero entry to another zero entry. (d_{ii} is never involved here since the i^{th} item cannot belong to two disjoint subcycles.)

To close the tour we must shift from the original entry in the first column of the last subcycle back to the left in the same row, but to the first column of the first subcycle. This last step will increase W from 0 to 1 since otherwise there would have been a rectangle of zero entries involving the first and last subcycles.

Thus we have proved

Theorem 1. We need at most one extra rotation of the drum beyond those indicated by the sum of the d_{ii} entries which are the terms common to all tours.

The following example illustrates the above techniques.

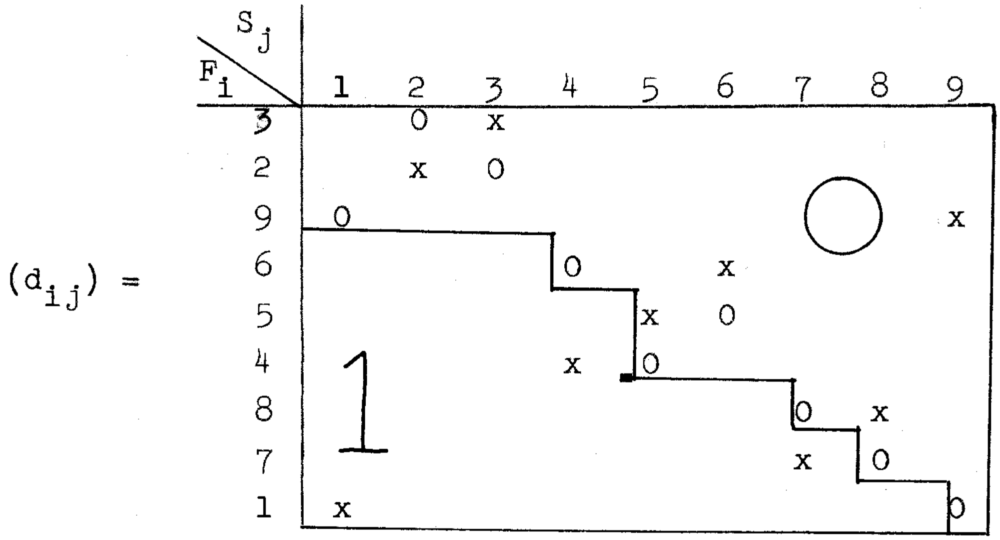

Figure 2

Following the steps outlined above step[redacted]

1. gives subcycles labeled by columns

   a b b c c c d d a ,

2. combines subcycles a and b by shifting from d_{91} to d_{92} and d_{32} to d_{31}, leaving subcycles by columns as follows

   a a a c c c d d a,

3. shifts[redacted]d_{31} to d_{41}, d_{64} to d_{67}, d_{87} to d_{81} = 1 giving an[redacted]optimal tour with total W = 1.

This gives an optimal tour (3,4,5,6,7,8,1,9,2) in cyclic order. Since our columns are renumbered, the actual sequence of word transfers can be read off from this list.

We have assumed throughout, as was implicitly done in [1] that no F_i coincides with an S_j. If $F_i = S_j$, clearly the j^{th} word must be transferred before the i^{th} word. The method of this paper still applies but there is an additional complication easily handled in any numerical case.

REFERENCE

1. Gill, Arthur, "The Optimal Organization of Serial Memory Transfers," I.R.E. Transactions on Electronic Computers, Vol. EC-9, No. 1, March 1960.