Scientific and Business Applications

On Finding Minimum Routes in a Network With Turn Penalties

TOM CALDWELL, *University of Arizona, Tucson, Arizona*

In some highway engineering work it is necessary to find a route between two points in a city's street and freeway network such that a function of time and distance is minimized. Such a route is called a "best" route, and finding such a route is not a difficult task. The Moore Algorithm[1] accomplishes this quite nicely, and using that algorithm and a procedure developed by Hoffman and Pavley[2] (programmed by them for the IBM 650) it is even possible to find the "Nth best path."

It is desirable to redefine the function to be minimized so that it depends on "turn penalties" in addition to time and distance. The purpose of this paper is to show that, while this additional variable complicates the use of the algorithms of Moore, Hoffman and Pavley, they are still applicable.

Borrowing from Hoffman and Pavley,[2] we shall define a network to be "a connected linear graph consisting of points called nodes and directed line segments joining pairs of nodes called links. Associated with each link is a positive real number called the value of the link."

Let us identify the nodes by integers $n = 1, \cdots, N$. Then a link may be identified by an ordered pair (i, j) where i is the link's initial node and j is its terminal node. (If the network contains at least one pair of links connecting the same pair of nodes, we will be forced to use ordered triplets to identify links. We shall assume this is not the case below.)

Let L_{ij} be the value of a link (i, j), in some network, and suppose there is physical justification for saying a route i, j, k in the network has a value $V \geqq L_{ij} + L_{jk}$. Then we may let

$$t_{ijk} = V - (L_{ij} + L_{jk})$$

and call t_{ijk} a turn penalty.

To make the preceding paragraph formal, we define a network with turn penalties to be a network such that with a given pair of links (i, j) and (j, k), $i \neq k$, there is associated a positive real number, t_{ijk}, called a turn penalty. A sequence of nodes $a_1, a_2, \cdots, a_m, a_{m+1}, \cdots, a_n$ is called a route (or path) if to every pair a_m, a_{m+1} there corresponds a link (a_m, a_{m+1}). In a network with turn penalties each route has a value V equal to the sum of the links composing it and the turn penalties associated with pairs (a_m, a_{m+1}), (a_{m+1}, a_{m+2}) of those links.

In a network with turn penalties it isn't necessarily true that the best path from the origin to a node i through a node j, coincides from the origin to j, with the best path from the origin to j. For instance, in the network pictured in Figure 1 and described in Table 1, the best path to node 4 from node 1 is 1, 2, 4; its value is 90. The alternate path is 1, 3, 4 with a value of 95. On the other hand, the value of 1, 2, 4, 5 is 161, and the value of 1, 3, 4, 5 is 153.

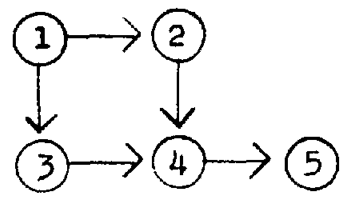

FIG. 1

TABLE 1

From i	*Thru j*	*To k*	L_{ij}	L_{jk}	t_{ijk}
1	2	4	47	42	1
1	3	4	41	46	8
2	4	5	42	58	13
3	4	5	46	58	0

Consider a route $n_1, \cdots, n_r$. Its value will be

$$\begin{aligned} V &= L_{n_1,n_2} + t_{n_1,n_2,n_3} + L_{n_2,n_3} + \cdots + L_{n_{r-1},n_r} \\ &= L_{n_1,n_2} + (t_{n_1,n_2,n_3} + L_{n_2,n_3}) + \\ &\qquad \cdots + (t_{n_{r-2},n_{r-1},n_r} + L_{n_{r-1},n_r}) \\ &= L_{n_1,n_2} + v_{n_1,n_2,n_3} + \cdots + v_{n_{r-2},n_{r-1},n_r} \end{aligned}$$

where $v_{i,j,k} = t_{i,j,k} + L_{j,k}$

Furthermore, if we define $t_{0,i,j} = 0$, then

$$v_{0,i,j} = L_{i,j}$$

and

$$V = \sum_{i=0}^{r-2} v_{n_i,n_{i+1},n_{i+2}} .$$

[1] MOORE, E. F. The shortest path through a maze. In *Proceedings of an International Symposium on the Theory of Switching*, Part II, April 2–5, 1957, The Annals of the Computation Laboratory of Harvard University 30, Harvard University, 1959.

[2] HOFFMAN, WALTER, AND PAVLEY, RICHARD. A method for the solution of the Nth best path problems. *J. Assoc. Comp. Mach. 6* (1959), 506–514.

Consider the links defined by (i, j) and (m, n). If $j = m$ (so that the links are connected), if $i \neq n$ (so that (i, j) is not the reverse of (m, n)), and if $n \neq n_1$ (so that the pair of links doesn't terminate at the origin), then we shall call the ordered pair $((i, j), (m, n))$ a hook and say $v_{i,j,n}$ is the value of the hook. Thus a route in a network with turn penalties is a sequence of hooks, and the value of the route is the sum of the sequence of hooks.

But this makes a hook in a network with turn penalties the same as a link in a network without turn penalties. Consider a pseudonetwork constructed in the following way: Let there be a node in the pseudonetwork for every link in the network except those whose terminal node is the origin. Thus if (i, j) is a link in the network, then (i, j) is a node in the pseudonetwork if j is not the origin. For each hook in the network, let there be a link in the pseudonetwork joining the nodes corresponding to the links which defined the hook. Thus the link $((i, j), (j, k))$ in the pseudonetwork corresponds to the hook (i, j, k) in the network. Let the link values in the pseudonetwork be the corresponding hook values in the network. The pseudonetwork is clearly a network, and hence the algorithms of Moore, Hoffman, and Pavley may be used to find best routes.

As a simple example, consider the network of Figure 1 and Table 1. The corresponding pseudonetwork is shown in Figure 2 and described in Table 2.

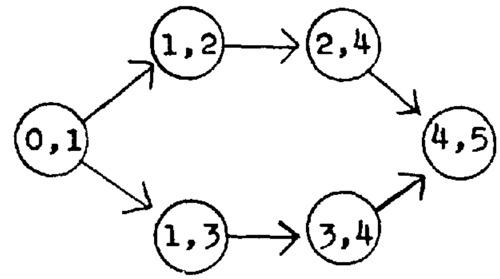

FIG. 2

TABLE 2

From i	*To j*	L_{ij}
0, 1	1, 2	47
0, 1	1, 3	41
1, 2	2, 4	43
1, 3	3, 4	54
2, 4	4, 5	71
3, 4	4, 5	58

Statistical Programs at the University of North Carolina

NORMAN BUSH, *University of North Carolina, Chapel, Hill, N. C.*

The Research Computation Center at the University of North Carolina has access to a UNIVAC 1105 general purpose digital computer for use in connection with data processing problems, theoretical studies, and computer research. With respect to data processing problems, three major statistical programs have been written:

A. General Contingency Table Analysis for Questionnaire Data
B. Analysis of Variance (ANOVA)
C. Multiple Regression and Correlation

Some of the concepts and ideas in these programs are new and may be of interest to other computation centers. Hence they are described below. The programs were written in the Remington Rand UNICODE language. Thus it would not be difficult to translate them into any other algebraic language, such as ALGOL, FORTRAN, or IT.

More details on each of the programs below plus the UNICODE programs, can be obtained by writing to the Research Computation Center, University of North Carolina, Chapel Hill, N. C.

A. General Contingency Table Analysis for Questionnaire Data

This program tabulates two-way tables from coded questionnaire data. The data is appropriately coded and punched on cards. (Only numeric codes are allowed; if there are more than 9 codes per question, then codes 10, 11, 12, 13, etc. are used instead of alphabet.) The coded information on the cards is then put on a magnetic tape, and the magnetic tape is used for input to the computer. Calculations for the two-way tables include:

(i) Counts
(ii) Marginals
(iii) Percentages by rows
(iv) Total chi square
(v) Individual cell contributions
(vi) Identification of questions

The major advantages of this digital computer program over conventional sorting and counting techniques are *speed*, *accuracy*, and the ability to form two-way tables with *any two questions* regardless of the fact that in coding the data on cards these questions appeared on different cards.

There is no restriction on the size of the table generated, and one of the current programs that generates 10×10 tables can produce up to 100 tables, with complete analysis, per hour. If the table is smaller, say 5×8, then up to 240 tables per hour can be produced. In the current programs one is also restricted to a maximum of 5000 questionnaires.

B. Analysis of Variance (ANOVA)

The most attractive feature of this program is the complete generality in computing sums of squares for an F-test