The Travel Resistance and shortest Path Calculation

Travel Resistance

Generalized time or cost functions

Performing activities and traveling requires spending time and money. Analogously to microeconomic theory, transportation theory adopts the following two equations as constraints to travel choice behavior:

Money budget

$$\sum_n y_{np} k_n = K_p$$

Time budget

$$\sum_n y_{np} t_n = T_p$$

where:

y_{np} = number of trips of person p for activity type (purpose) n
k_n = costs (including travel costs) of performing activity n
t_n = time (including travel time) needed for performing activity n
K_p = money budget (equal to income) of person p;
T_p = time budget of person p.

If the personal income during time period T is denoted with INKp, then it holds

$$K_p = INK_p \cdot T_p$$

Implying that Money budget also may be written as:

$$\sum_n y_{np} \frac{k_n}{INK_p} = T_p$$

If individuals maximize their activity utility within their time and budget constraints, it can be shown that this behavior implies the trading off of time and money. This means that both constraints may be combined into a single weighted summation. Weighted summation leads to the equation:

$$\sum_n y_{np} (t_n + \gamma \frac{k_n}{INK_p}) = (1+\gamma) T_p$$

The travel resistance Zn to performing an activity n thus is:

$$Z_n = t_n + \gamma \frac{k_n}{INK_p}$$

Zn is called generalized time.

If we apply this to a zonal interchange i-j the following equation results

$$Z_{ijv} = t_{ijv} + \gamma \frac{k_{ijv}}{INK}$$

in which:

Z_{ijv} = generalized time between zones *i* and *j* with mode *v*;
t_{ijv} = travel time between zones *i* and *j* with mode *v*
k_{ijv} = travel cost between zones *i* and *j* with mode *v*;
γ = a parameter mostly proportional to income ($\gamma = 3$)

The ratio INK/γ is called value-of-time (VOT). This is the amount of money that travelers are willing to spend in order to save one unit of travel time. We may expect that additional factors will influence the travel resistance, such as e.g. physical effort in bicycling.
Time is a scarce good and thus valuable. In making travel choices, travelers make a tradeoff between time and money: time is valued in money. Car drivers try to save parking costs by walking a longer distance to their destination. In the case of road tolling a lot of car drivers prefer making a detour to save out-of-pocket expenses. The extent to which they do this depends on their value-of time which in turn depends on their income or their gross hourly rate (for business men). Table 1 shows value-of-time figures currently applied by the Dutch Ministry of Transport in their transportation studies.

Table 1. Base value-of-time values for evaluation, in guilders (of 1988) per hour [Source: Ministry of Transport]

income group (averaged over all modes) in guilders/month (gross)	trip purpose commuting	business	other
< 2500	9.20	19.80	7.30
2501 – 4000	9.70	27.80	8.20
4001 – 6000	13.00	37.90	9.30
> 6000	13.40	48.20	11.40
all groups	11.30	37.50	8.70
income group (averaged over all purposes) in guilders/month (gross)	**trip mode car**	**train**	**bus/tram**
< 2500	8.70	6.80	5.00
2501 – 4000	9.80	8.10	6.00
4001 – 6000	13.50	9.70	7.20
> 6000	17.60	12.70	10.10
all groups	12.00	8.90	6.50
mode (averaged over all income groups)	**purpose commuting**	**business**	**other**
car	11.40	37.60	9.10
train	11.60	33.00	7.90
bus/tram	9.50	32.90	5.60
all groups	11.30	37.50	8.70

Before-and-after studies of road traffic tolling show that the car driver is willing to make a detour of about 5 kilometres (or willing to drive extra 5 minutes) in order to save Hfl. 3.50 toll. From these studies it can be derived that one hour in-vehicle time on average is valued at about 8 to 10 guilders (in 1988). The average value-of-time of road freight shipments amounts to about Hfl. 63 (≈ € 29) per hour (in 1992).
The value-of-time of travelers also plays a role in assessing the yield of infrastructure investments. Such investments normally lead to travel time gains; using the aforementioned value-of-time figures these travel time gains can be expressed in money terms and balanced with the investment costs.
The values in the table only apply to persons aged 16 or older; income figures refer to households; one has to take account of real price increases. The VOT-figures give the distribution over three variables: trip purpose, mode and income. In the table two-way splits are shown averaged over the third variable. The VOT-figures are derived from revealed travel behavior. Travel time losses generally are valued higher than travel time gains because of distortions of time plans.

Objective, perceived and modeled resistance

In determining travel resistance we need to distinguish objective, perceived and modeled resistance.
Objective resistance or costs (X_{nk}) can be observed objectively by measurement rules. Time and distance traveled by car or bus for example.
Subjective or perceived resistance or costs result because of individual weighing of objective costs. Decisions of travelers are made on the basis of perceived costs. The assessment of resistance is related to personal characteristics and depends on observational errors in personal estimates of times and distances. Perceived time is judged according to the extent a person can dispose freely of his time, thus whether one has a job or not, whether one is caring for children or not, whether a person goes to school or not.

Modeled resistance or costs ($ß_{nk}$, X_{nk}) are used in a model analyzing travel behavior. The parameter values are derived from observations on actual behavior using estimation and testing models. From empirical analyses using disaggregated estimation models it has been found that time is perceived very differently.

Especially with respect to public transport trip making it appears that access and egress times (that is the time from the door at the origin address to the vehicle, respectively the time from the vehicle to the door at the destination address), waiting times and transfer times are weighed 2 or 3 times higher than the in-vehicle travel time. Also the waiting times in the car at intersections is valued higher than the in-car driving time (Hamerslag, 1979).

Shortest path calculation

The calculation of shortest paths in a network is an essential step in each transportation analysis. This calculation is done very many times in each analysis so that its calculation time is predominant in the analysis duration.

The shortest paths serve several functions within the transportation analysis.

- Shortest paths between nodes or zonal centroids give the travel resistance of trips in a network (expressed in simple or generalized time, in distance, or in costs). These trip resistances are used in modeling choice behavior (mode, destination, and time and route choice).
- Shortest paths also are used in the modeling step of assigning traffic flows between zones to the network in order to find link loads.

Very many algorithms exist to find shortest paths in networks. We may distinguish:

- tree builder algorithms
- matrix algorithms

The tree-builder algorithms search sequentially for each origin node (or centroid) the shortest route to all other nodes (and thus also destination centroids). These algorithms are first introduced by Moore (1957). A special class are the once-through algorithms due to Dijkstra (1959).
Matrix algorithms search simultaneously shortest routes from all origin (centroid) nodes to all destination (centroid) nodes of the network. In general, in larger transportation networks, tree builder algorithms are more efficient: they need less computer memory less calculation time. On the contrary, matrix algorithms are easy to program.

Tree builder algorithm

The network is described by nodes (index n) and links (index k). The resistance of a link is represented by a generalized time z_k.
To each node n a label is attached consisting of 3 components: q_{in}, m_{in} and α:

- q_{in} is the smallest resistance from the origin node i to node n found after each computation step;
- m_{in} is the node number of the last node in the shortest route, the so-called backnode;
- α_n is an (0,1) indicator showing whether still computations have to be carried out with this node n or not, that is, whether the node is active (0) or non-active (1).

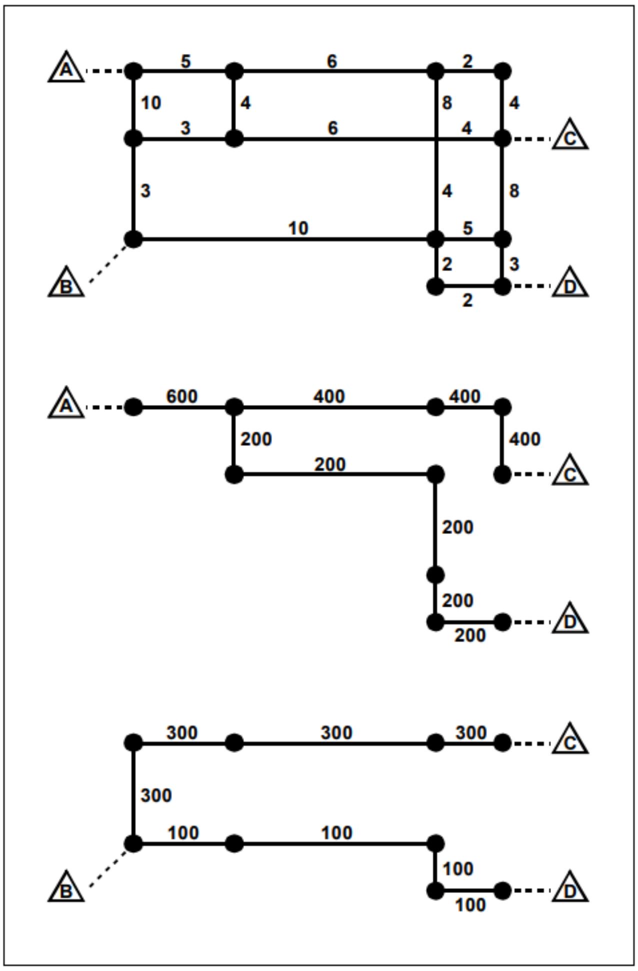


Fig. 1 Illustration of shortest path search.

For each centroid or origin node i the following computation steps are carried out:

1. (Initialization) all nodes are labelled (B, 0, 1) where B is a very large number. The null means that no back node has been found yet.
2. The origin node is labelled (0,0,0)
3. It is checked whether at least one node is active (α=0). At the first step this is the origin node. If no active node is found the calculations for the origin node i are completed.
4. Next, one of the active nodes is chosen (n)
5. Next it is determined which nodes (k) are connected to the active node.
6. For each of these nodes (k) it is determined whether the shortest path is via the active node. This is the case if: $q_{in} + z_k < q_{ik}$ Node k then is labeled with components ($q_{in}+z_k$, n, 0). This means that this node is attached the new shorter resistance, the back node number is changed, and this node now becomes active.
7. The value of α of node n is set to one, thus is made passive.
8. The computation continues with step 3.

The Moore-type algorithms may select an active node in various ways:

- According to the sequence it is put into an auxiliary table
- Taking the node with lowest node number, or
- Randomly.

The once-through algorithms however select the node with the smallest resistance from the origin node. With the once-through algorithm each node becomes active only once. This saves computation time. However, calculation time is needed to find out which node is 'closest' to the origin node. This calculation time often is larger than the extra computation time if nodes become active several times during tree building.

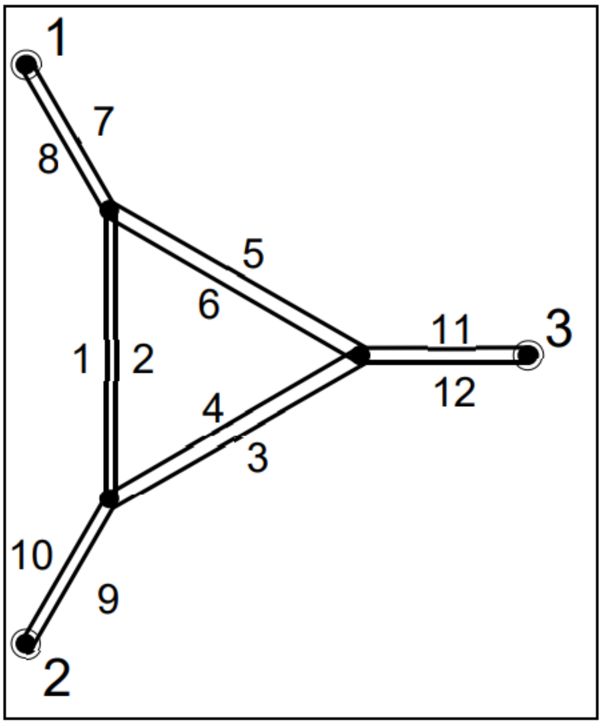
As input to the calculations the network has to be organized into special data structures. Also the output, being the shortest route trees of which one per origin centroid, need to be organized in special data structures. Many alternative ways of organizing these data are possible offering different opportunities for further application such as the reconstruction of alternative routes to the shortest one using the back node information in the shortest tree. For an elaboration of these data possibilities as well on search algorithms, see Ajuha et al (1993).

Assignment Map

There are a number of ways to store the results of a shortest path computation. A very compact way is the shortest path tree. The shortest path tree is a vector with a length that corresponds with the number of nodes in a network. This vector contains for each node its back node: this is the first node that is encountered on the path from the current node to the origin of the shortest path tree.
In one shortest path tree all the information that is needed to reconstruct the paths from one origin to all network nodes can be stored. Hence, if one wants to store the all-to-all shortest path information for a network with N nodes, then N shortest path trees with N elements each are needed.
Albeit very compact, the shortest path tree is not the most suitable form to store paths, when one needs to make computations. In this case it is much more common to use an assignment map. The assignment A is a table with a height that corresponds to the number of network links and a width that corresponds with the number of routes. If element a, r of assignment map A equals 1, i.e. A (a, r) = 1, this indicates that route r traverses arc a.

If we consider the network shown in Figure 2, with origin/destination nodes 1, 2 and 3, then the corresponding assignment map is given by the matrix A. The columns in this matrix correspond with the shortest-paths for OD-pairs 1-2, 1-3, 2-1, 2-3, 3-1 and 3-2.
The assignment map may be used for various types of analysis, such as:

- Computing the total flow on a link
- Determining the routes that pass over a link
- Determining the routes that pass over a particular combination of links
- Determining the links that a route consists of
- Determining the contribution of a particular route-flow to a link flow



$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Fig. 2 Example network with 6 nodes and 12 links. Nodes 1, 2 and 3 function as origin and destination nodes.