Traffic assignment

Shortest path-min. tree building

- what is a tree?
  - If for each origin node in the network are arrived at by one link only, the set of paths from h to the nodes is called "a tree". The process of determining the minimum cost path is called "tree building". The tree together with the minimum path costs from the origin nodes to all destination nodes is referred to as "skimmed tree" (giving rise to the term "skim table" in TDF)

Dijkstra's algorithm

1. for each vertex v in Graph // Initializations
2. dist[v] = ∞ for all v in Graph
3. prev[v] = null for all v in Graph
4. Q = Graph for all v in Graph
5. while Q is not empty
   a. u = vertex v in Q with smallest dist[v]
6. for each neighbor w of u // Relax edges
7.   if dist[w] > dist[u] + dist_between(u, v)
8.   dist[w] = dist[u] + dist_between(u, v)
9.   prev[w] = u
10. return dist
An example
Find the shortest paths from node 1 to all other nodes in the network.

Example solution

<table>
<thead>
<tr>
<th>Step 1/5</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Another example
Find the shortest paths from node A to all other nodes.

Another example
Find the shortest paths from node 6 to all other nodes.

Inputs to assignment
- O-D matrix
- Network

Terminology (1)
- Trip assignment: "loading the network"; volumes are assigned to links
- Free flow speed: speed under no congestion
- Free flow travel time: travel time under no congestion
- Path finding: finding the path with the minimum impedance
- Path loading: loading vehicles to links comprising a path
- Level of service: a qualitative measure describing the operation conditions
Assignment methods

- Non-equilibrium (heuristic) assignments
  - All or nothing: link travel times are determined beforehand and trips are assigned at once
  - Incremental assignment: link travel times are updated through fixed proportions and trips are assigned iteratively

All or nothing assignment

Simple all-or-nothing method is the fundamental building block in traffic assignment procedures.

Incremental assignment

A simple but inconsistent way to account for capacity and congestion effects.

Example

Consider a simple transportation network that has one origin and one destination and two links/paths that provide access from the origin to the destination. One link is 7.5 miles long and has a capacity of 4000 vehicles per hour and a speed limit of 55 miles per hour. The other link is 5 miles long and has a capacity of 2000 vehicles per hour and a speed limit of 35 miles per hour. Assuming that 5000 drivers wish to make the trip from the origin to the destination, find the loaded network.

All or nothing assignment

Link 1:
- Capacity: 4000 vehicles/hour
- Speed: 55 mph
- Distance: 7.5 miles
- Free-flow travel time: 13.5 minutes

Link 2:
- Capacity: 2000 vehicles/hour
- Speed: 35 mph
- Distance: 5 miles
- Free-flow travel time: 9 minutes

Incremental assignment

Let us first assign 1000 vehicles to Link 1 and then update link travel time, which will be:

$t = 13.05 \times \frac{1000}{4000} = 3.26 \text{ minutes}$

The next 1000 vehicles will be assigned to Link 1, which gives a travel time of 6.52.

The next 1000 vehicles will be assigned to Link 2, which results in 9 minutes.

Thus, Link 1 now begins the second path, and therefore, the next 1000 vehicles to Link 2, which gives a travel time of:

$t = 9.05 \times \frac{1000}{4000} = 2.26 \text{ minutes}$
Example: do all-or-nothing and incremental assignment

<table>
<thead>
<tr>
<th>From\To</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

$t_i = t_0 \left(1 - \frac{P_i}{C_i}\right)$

Link characteristics

<table>
<thead>
<tr>
<th>Link</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_0$</td>
<td>10</td>
<td>15</td>
<td>10</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>$C_i$</td>
<td>300</td>
<td>500</td>
<td>150</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>