TRANSPORT NETWORK VULNERABILITY ANALYSIS BASED ON TRAFFIC ASSIGNMENT METHODS

Miroslav Slivoně[1]

Summary: This paper discusses the possibility of use of two traffic assignment approaches to solve the problem of identification of vulnerable links. The first method is based on Dial's algorithm originally used to solve the stochastic traffic assignment problem. The second method calculates the costs experienced by all the network users according to the user equilibrium assignment (before and after particular link failure). This approach is not new but not commonly used because of its high computation complexity; this complexity can be considerably reduced using the subnetwork approach. There are proposed some new modifications of the subnetwork approach which could reduce the complexity of the original idea.

Key words: transport network vulnerability, traffic assignment, stochastic assignment, user equilibrium.

1. THE BASIC CONCEPT OF THE NETWORK VULNERABILITY

There have been proposed different definitions of the network vulnerability by the scientific community [1], [2]. This paper supposes following concept of network link vulnerability: The link on network is vulnerable when the consequences of its failure are severe, irrespective of the probability of the failure.

The total consequences TC_l of failure of link l can be expresses like the sum of direct and indirect consequences [3]:

$$TC_l = DC_l + IC_l, \tag{1}$$

where:

DC_l are direct financial costs (needed to repair the link),

IC_l are indirect generalized financial costs induced by the link failure.

The measure of direct financial costs is out of scope of this paper. These costs will be extremely high especially for technically difficult infrastructure objects like bridges or tunnels. The repair process of such objects will be also very time consuming, so the duration of the failure is supposed to be long. This fact has negative impact on the total size of indirect costs.

The indirect consequences experienced by network users can be measured mainly by means of additional travel distance and additional travel time.

[1] Ing. Miroslav Slivoně, University of Pardubice, Jan Perner Transport Faculty, Department of Transport technology and Control, Studentská 95, 532 10 Pardubice, Tel. +420 466 036 198, E-mail: Miroslav.Slivone@upce.cz

The total additional travel distance can be expressed like:

$$TD_l^{\Delta} = \sum_i \sum_j f_{ij} (d_{ij}^{\,l} - d_{ij}), \tag{2}$$

where:

f_{ij} is the size of transport demand, i.e. number of trips from zone i to zone j,
$d_{ij}^{\,l}$ is the average travel distance when the link l is broken,
d_{ij} is the average travel distance under normal conditions.

The total additional travel time can be expresses like:

$$TT_l^{\Delta} = \sum_i \sum_j f_{ij} (t_{ij}^{l} - t_{ij}), \tag{3}$$

where:

$t_{ij}^{\,l}$ is the average travel time when the link l is broken,
t_{ij} is the average travel time under normal conditions.

The travel distances and travel times are assumed to be the average values. In dependence on selected traffic assignment model these distances or times are not necessary equal for all users.

The indirect consequences can be then calculated like:

$$IC_l = TD_l^{\Delta} \cdot c_D + TT_l^{\Delta} \cdot c_T, \tag{4}$$

where:

c_D are the costs of driving one distance unit,
c_T are the costs of one unit of driving time.

Failure of some links can lead to separation of some part of the network. These links are called cut-links (which corresponds to the term “bridge” used in graph theory). In case of cut-link failure, both TD_l^{Δ} and TT_l^{Δ} equal to infinity. By this reason the cut-links should be indentified prior to measuring link vulnerability of network. Importance of such a cut-link can be measured by total unsatisfied demand U_l [4]:

$$U_l = \sum_i \sum_j u_{ij}^{l}, \tag{5}$$

where:

$$u_{ij}^{l} = \begin{cases} f_{ij} & if\ t_{ij}^{l} = \infty,\ d_{ij}^{l} = \infty \\ 0 & if\ t_{ij}^{l} < \infty,\ d_{ij}^{l} < \infty \end{cases}. \tag{6}$$

The idea of the concept of network vulnerability presented above is simple; however the measure of the vulnerability of all the links can be very time consuming. In many practical applications of the network vulnerability analysis this required computation time makes hundreds of days [3]. The calculation time depends mainly on:

- the size of network – a realistic transport network contains tens of thousands of nodes and links,

- the traffic assignment model used and its setting – besides the simplest unrealistic models, it takes quite a long time to solve most of variants of traffic assignment problem.

2. EXISTING VULNERABILITY ANALYSIS VARIANTS BASED ON TRAFFIC ASSIGNMENT MODELS

2.1. Stochastic traffic assignment without capacity constraints

This kind of traffic assignment was intensively studied in the literature during past decades. Some of the models assume Gaussian distribution to express the error in the perceived travel time – these methods are called probit based assignment methods and the computation of assignment is usually made by Monte Carlo method. Other models assume the distribution of the perceived travel time error to be Gumbel's – they are called logit based and the assignment can be computed explicitly. Probably the most common logit based method was proposed by Dial [5]; Taylor [1] used Dial's algorithm to identify vulnerable links.

As was stated above, the network users in Dial's algorithm do not always decide for the minimum costs path – they decide according to logit formula:

$$p_r = \frac{e^{-\alpha t_r}}{\sum_{j=1}^{m} e^{-\alpha t_j}}, \tag{7}$$

where:

p_r is the probability that user choose the route r,

t_r is the travel time spend on route r (can be also distance or generalized costs),

α is the parameter which expresses the user's sensitivity to travel time.

To avoid the need of the enumeration of all the paths from origin to destination, the path acceptability criterion is added to the model. The path from origin i to destination j is acceptable only when each link (r, s) on this path satisfies $t_{ir} < t_{is}$; i.e. travel time from origin i to initial node r of this link must be higher than travel time from origin i to terminal node s of the link.

The algorithm works this way:

STEP 1

Calculate the link weights w_{rs} and node weights W_s according to given formulas:

$$w_{rs} = \begin{cases} e^{-\alpha t_{rs}} \cdot W_s & \textit{for link which satisfies acceptability criterion} \\ 0 & \textit{for other links} \end{cases} \tag{8}$$

$$W_s = \begin{cases} \sum_{r,s\in\beta^+(s)} w_{rs} & \text{if } s \neq j;\ \beta^+(s) \text{ is a set of links which can be used to leave node } s \\ 1 & \text{for destination node } j;\ \text{i.e. } s = j \end{cases} \quad (9)$$

These weights can be computed recursively from the destination node j in the forward topological order – it means by increasing the value of t_{sj} from node j.

STEP 2

Compute conditional probabilities $P\{(r, s) / r\}$:

$$P\{(r, s)/r\} = \frac{w_{rs}}{W_r}. \quad (10)$$

Then assign flow to network according to these probabilities.

Discussion, possibilities of use in vulnerability analysis

These two steps must be repeated for each origin – destination pair. Taylor [1] does not assign the flow to network; the conditional probabilities are used as indicators of link vulnerability. The higher is the link probability, the greater are the consequences on the origin – destination pair i, j if the link is broken.

Dial's method can be used even in the realistic large networks. The algorithm has two time demanding parts: the computation of the distances of all nodes from path's initial node and the sorting of nodes in forward topological order. This can be done in acceptable time using fast shortest path algorithm (e.g. Dijkstra or A* with binary heap) and fast sorting algorithm (e.g. binary heap).

The described method of stochastic traffic assignment can be used to compute indirect link failure consequences according to equitation (4). However the results will not be realistic because the link capacity restraint (congestion effect) is not included.

2.2. Deterministic traffic assignment with capacity constraints

The model which involves the link capacity restraints and the first Wardrop's principle is called deterministic user equilibrium (DUE) traffic assignment.

The link capacity constraints can be expressed by some link travel time function; probably the most frequently used function is this BRP formula:

$$t_a = \alpha_a + \beta_a \left(\frac{v_a}{c_a} \right)^{\gamma_a} \quad (11)$$

where:

- v_a is the actual link flow,
- α_a is the free-flow travel time,
- c_a is the capacity of link a,
- β_a, γ_a are parameters calibrated for link a.

The Wardrop's principle mentioned above says: "Under equilibrium conditions traffic arranges itself in congested networks in such a way that no individual trip maker can reduce his path costs by switching routes."
The DUE traffic assignment problem is stated like:

$$\min_{v} \sum_{a \in A} \int_{0}^{v_a} t_a(\omega, c_a)\, d\omega \tag{12}$$

$$\sum_{r \in R_w} f_w^r = q_w \qquad w \in W \tag{13}$$

$$v_a = \sum_{w \in W} \sum_{r \in R_w} f_w^r \delta_{ar}^w \qquad a \in A \tag{14}$$

$$f_w^r \geq 0 \qquad r \in R_w,\ w \in W \tag{15}$$

where:

R is the set of all routes in the network,
W is the set of all origin - destination (O-D) pairs,
R_w is the set of all routes between O-D pair $w \in W$,
q_w is the demand between O-D pair $w \in W$,
f_w^r is the flow on the route $r \in R$,
δ_{ar}^w equals 1 if route r between O-D pair w uses link a and 0 otherwise.

The DUE problem can be solved using well-know Frank-Wolfe algorithm or some other alternative like DSD, MSA, Dial's algorithm B [6]. This approach matches best the idea of indirect consequences according to formula (4). Unfortunately the computation time will be rather high in large network.

Discussion, possibilities of use in vulnerability analysis

Interesting approach how to reduce time complexity of vulnerability analysis based on DUE assignment was presented in [3]. The authors used local subnetworks instead of the whole network – the whole idea is based on the fact that the main part of the links serves only little demand with rather short average path distances. Hence the redistribution effect in dense parts of network will be spatially restricted.

Such a subnetwork was a limited section cut from the complete network including the section's internal and transit demand (resulting from DUE assignment made using whole network). The subnetworks were generated using 60 km edge length and an offset of half an edge length – so every link was at least 15 km far from nearest subnetwork border. Failure consequences calculated using subnetworks were compared against those consequences assessed using the full network. The subnetwork methodology approved to be very accurate and reliable and the time gain was substantial (in case of the Swiss national transport network with approx. 30 300 undirected links and 24 300 nodes: 40 minutes for the whole network, 35 seconds for one 60 km subnetwork). However, the 60 km subnetwork was not used for national class of roads (which serves long distance O-D relations; in this case the whole

network was considered) and for sparse Alps areas (long length of detour; special large Alps subnetwork used); for cut links was used formula (5).

3. BRIEF PROPOSALS OF MODIFIED LINK VULNERABILITY ANALYSIS APPROACH

Although the method proposed in [3] approved itself to work very well, it could be improved by following modifications. These improvements could positively affect the model precision and its computation complexity.

Modification 1 (use of link radii)

The original method works with national, cantonal and municipal roads different way. The upper level of the hierarchy is supposed to serve for long distance trips and the whole network is involved. The lower level of the hierarchy serves especially for small distance trips and the subnetwork of constant size is used. Although this is valid in most cases, there is a better way how to work with link hierarchy.

The original approach considered a discrete number of hierarchy levels. Algorithm described in [7] uses continuous range of hierarchy levels. Ertl's radii algorithm is originally intended to reduce the searching space during shortest path calculation in large network. However the same idea can be used also for the purposes of link vulnerability analysis – the size of subnetwork can be set according to the link radius.

Each directed link has an associated radius. If a link has the radius R_l, it means that every shortest path over this edge either starts or ends in a node which has a distance smaller than R_l from the starting node of this link.

These link radii can be computed for most of the links exactly using this subnetwork approach: the network is covered by overlapping rectangles so that each rectangle contains about 2 000 nodes. The distance matrixes for these subnetworks are created and the radii can be estimated for 75 - 85 % of links. With this algorithm it is not possible to calculate all link radii – if the radius of a link is larger than the rectangle size it equals to infinity. For these links at least upper radii bounds can be calculated using the entire network which does not contain links with known radii.

The link radius is definitely better representation of the usability of link for long distance trips than a link category alone. The subnetwork rectangle edge size can be estimated from link radius - the rectangle should contain the whole radius plus some addition to cover the redistribution effect. The appropriate subnetwork (its edge size and positioning) can be selected from set of subnetworks created in advance (see modification 2).

Modification 2 (creation of set of subnetworks of various size in advance)

The link radius approach allows to use subnetwork of required size. Prior to subnetwork creation the DUE traffic assignment needs to be solved to build up the subnetwork's transit demand. All the path redistribution is supposed to be made internally - within the subnetwork. The process of aggregation of transit demand from particular paths is quite time demanding –

so it seems like a good idea to create a set of overlaying subnetworks of different sizes in advance and store them in the computer's hard drive. Then for each link the best subnetwork is chosen (the link should be located in the central area and should have the right size).

Use of the stochastic assignment

The stochastic assignment approach works faster compared to the DUE assignment. The subnetwork approach and link radii approach can be also used to accelerate the algorithm. The links identified to be vulnerable could be examined also using DUE assignment. The research of affinity of results reached by both methods in real networks appears to be very interesting.

4. CONCLUSION

The proposals suggested above will be objects of future analysis. Computer model which involves creation of subnetworks and both traffic assignment approaches is created in these days. The results of this analysis will be published soon.

5. ACKNOWLEDGEMENT

This paper has been supported by the Institutional research MSM 0021627505 „Theory of Transport Systems".

LITERATURE

[1] D'ESTE, G. M. D., TAYLOR, M. A. P, *Network Vulnerability: An Approach to Reliability Analysis at the Level of National Strategic Transport Networks*, In Proceedings of the 1st International Symposium on Transport Network Reliability, Oxford (2003).

[2] BERDICA, K., *An introduction to road vulnerability: what has been done, is done and should be done*. Transport Policy, 9, (2002).

[3] ERATH, A., BIRDSALL, J., AXHAUSEN, K.W., HAJDIN, R., *Vulnerability assessment Methodology for Swiss Road Network*. Transportation Research Record: Journal of the Transportation Research Board, vol. 2139 (2009).

[4] JENELIUS, E., PETERSEN, T., MATTSSON, L.G., *Importance and Exposure in Road Network Vulnerability Analysis*. Transportation Research Part A: Policy and Practice (2006).

[5] DIAL, R. B., *A probabilistic multipath traffic assignment model which obviates the need for path enumerationm*, Transporatation Research, 5 (1971).

[6] PATRIKSSON, M. *The Traffic Assignment Problem: Models and Methods*. Utrecht: VSP, (1994).

[7] ERTL, G.: *Shortest Path Calculation in Large Road Networks.* OR Spectrum, Springer-Verlag, (1998).