Issue 22(1), 2022, pp. 63-92
https://doi.org/10.18757/ejtir.2022.22.1.5657

TU Delft OPEN

EJTIR

ISSN: 1567-7141
http://ejtir.tudelft.nl/

Identification of road links with the gravest network impacts when blocked concurrently

Rostislav Vodák[1]
Faculty of Science, Palacký University, Olomouc, Czech Republic

Zuzana Křivánková[2]
CDV – Transport Research Centre, Brno, Czech Republic

Michal Bíl[2]
CDV – Transport Research Centre, Brno, Czech Republic

The identification of a combination of links which can cause the gravest impact on network performance, when interrupted simultaneously, is of great practical importance. The links can be spatially dispersed due to an event such as an earthquake or due to a combination of accidents and disasters. This task is, thus, extremely computationally demanding when applied to real-world networks. We focused on approaches which are both capable of accomplishing this task and where the computational time is acceptable for application in practice. We tested three algorithms based on known heuristic methods: Simulated Annealing (SA), Guided Local Search (GLS) and Variable Neighborhood Search (VNS). The algorithms were modified in the sense of adjusting the searching neighborhood. All the algorithms were subsequently applied to four actual road networks in order to evaluate the impacts of complete simultaneous blockage of four and ten links. The results suggest that the modified SA algorithm identified scenarios with worse consequences than the algorithms based on GLS and VNS. The SA results, for the setting with four interrupted links, were even comparable with those obtained from a deterministic algorithm (which evaluates the entire state space). The algorithm based on SA was also performing best for situations with ten concurrently blocked links. The approach based on SA is thus suitable when modeling the potential impacts of events where a large number of concurrently blocked links is expected. Network managers will thus be able to monitor the immediate state of the network and potential risks related to network disintegration.

Publishing history

Submitted: 17 March 2021
Accepted: 11 March 2022
Published: 17 March 2022

Cite as

Vodák, R., Křivánková, Z., & Bíl, M. (2022). Identification of road links with the gravest network impacts when blocked concurrently. European Journal of Transport and Infrastructure Research, 22(1), 63-92.

Keywords: *road networks, link interruption, vulnerability, simulated annealing, guided local search, variable neighborhood search*

[1] 17 listopadu 1192/12, 779 00, Olomouc, Czechia T: +420 585 634 604 E: rostislav.vodak@upol.cz
[2] Lisenska 33a, 636 00, Brno, Czechia T: +420 602 361 748 E: michal.bil@cdv.cz

1. Introduction

Quantification of risk for various types of negative events ranks among the main current research topics. Low risk does not mean that such an event cannot occur at all. There is therefore a need to evaluate not only the probability of an event but also its (possible, albeit less probable) impacts beforehand, as many low probability events can have devastating impacts (Hohenemser et al., 1986; Koerblein and Kuechenhoff, 2019).

The identification of the concrete network links (we call them the *weak* links in this work) which, when blocked concurrently, have the gravest network impact, usually means evaluation of all the possible events (i.e., all the combinations of blocked links within the network). Such a task is, for actual road networks, almost always too intensive in terms of computational time, despite the current progress in computer power and parallelization. Determination of the worst impacts for a disaster resulting in ten concurrently blocked links, in a network containing 1,000 links, means, for example, evaluating 2.6×10^{23} possible scenarios. It is not always necessary to evaluate all possible states, as only some kinds of events are of interest. In these cases, certain constraints (such as a reduced set of possible blocked links or concentration on small subregions) are usually applied. The constraints can significantly reduce the overall number of combinations of interrupted links (the size of the state space). Despite this fact, the number of all combinations of interrupted links usually still remains enormous.

The computational intensity is also affected by the choice of loss function (vulnerability measure) which is used to rank the results, according to the selected criterion. The primary demand for such a measure is the speed of the evaluation process. This fact, together with computational difficulty, is the reason, in all probability why many previously published works only paid attention to less extensive networks and/or to scenarios with a single interrupted link. In cases of a higher number of concurrently interrupted links, additional criteria were added to reduce the number of possible combinations.

In this paper, no restrictions on the interrupted links were assumed. The interrupted links can be thus spread across the network. It is assumed that the links are completely blocked and that there is no way to pass through them. Therefore, we do not assume any traffic flow between nodes. The events, to which we pay attention, result in a complete disintegration of the road network. Three algorithms, based on the known heuristics (Simulated Annealing, Guided Local Search, Variable Neighborhood Search), are introduced and tested at four real road networks with 984–2,394 links. four and ten links were blocked concurrently to simulate the impacts of disasters. In the case of the four blocked links, the algorithms were also compared with a parallelized deterministic algorithm (PDA).

1.1 *An overview of existing approaches and basic notions*

The analyses of the impacts were applied to the transportation networks where both local (e.g., traffic accidents) and regional (e.g., rainfall, earthquake) events took place (see for instance Chang and Nojima, 2001; Sohn, 2006; Bono and Gutiérez, 2011; Jenelius and Mattsson, 2012; Bíl et. al, 2017; Zahradníček et al., 2018). Events with a high spatial extent usually affect many network nodes at a time and can lead, in the worst case, to disintegration of the network into several isolated parts (Bíl et al., 2015a).

Any vulnerability analysis requires a (vulnerability) measure which enables an assessment of the impacts of various events. The literature covering the vulnerability and vulnerability measures is extensive. We recommend the reader begin with (Knoop et al., 2012) where the authors tested several vulnerability measures. Highly vulnerable networks often undergo a decrease in their robustness. We refer the reader to (Ellens and Kooij, 2013; Vodák et al., 2015; Barabasi and Posfai, 2016; Schieber et al., 2016) for definitions of various robustness measures. Concerning the traffic on networks, measures based upon travel costs and unsatisfied demands have been presented by (Scott et al., 2006; Jenelius et al., 2006; Jenelius, 2009). Additional measures, which can be used universally, are based upon the accessibility, s-t path availability and the shortest paths and were developed in (Taylor et al., 2006; Chen et al., 2007; Matisziw and Murray, 2009; Hong-ying and Li-gun, 2010). The capacity reduction approach (Sullivan et al., 2010; Watling et al., 2012) should be used if information about actual link capacities exists. Recent approaches to vulnerability in different research areas are for instance (Lakner et al., 2018; Leng et al., 2018; Wei et al., 2018; Berberler and Yigit, 2018). A recent review on road network vulnerability was published by (Taylor, 2017).

The primary drawback of many previously published vulnerability measures is that they cannot be directly applied to a physically disintegrated network. For instance, the Network Robustness Index defined in (Sullivan et al., 2010) requires information about travel time and traffic volume. These values are, however, significantly changed or even unknown or the travel time equals infinity in the case of a large disruption leading to network disintegration. It also holds for other quantities such as flows, traffic demand and traffic patterns (Kurauchi et al., 2009; Helbing, 2001; Helbing, 2002; Lammer et al., 2006; Gottlich and Klar, 2009). This leads to a need for a definition of the measures based on the importance of the particular nodes (see also Taylor et al., 2006).

As mentioned above, the two main complications are the large number of possible scenarios and the time needed to evaluate a loss function. These two obstacles have led many authors to work with only *one* interrupted link or node at a time (Taylor et al., 2006; Latora and Marchiori, 2004; Knoop et al., 2008; Erath et al., 2010; Newman, 2010; Wang et al., 2013). Two or even more concurrently interrupted links often occur, however, as a result of natural disasters (Chang and Nojima, 2001; Bono and Gutiérez, 2011; Bíl et al., 2015a). The impacts of these disasters were evaluated retrospectively. This means that only one state, the outcome of the disaster, was compared to the healthy network to express the damage caused by the particular disastrous event. Certain other authors applied geographic restrictions to the parts of the network which can be affected simultaneously, when modeling the impacts of potential events (Sohn, 2006; Jenelius and Mattsson, 2012; Latora and Marchiori, 2004; Wang et al., 2013). Only small parts of the entire existing state space were evaluated in these cases.

At the end of the chapter, we provide a brief overview of the basic notions related to the topic of the paper and introduce how we use them. *Vulnerability* in the road transportation system is susceptibility to incidents that can result in considerable reduction to road network serviceability, i.e., the possibility of using a link/route/road network during a given time period (Berdica, 2002). These incidents may be more or less predictable, be caused intentionally or unintentionally, by man or by nature. It is not an easy task, however, to calculate the probability of events, especially of the events with a large impact on a road network. This is the reason why we omit the probabilities and only pay attention to the impact on the road network and to the identification of the worst-case scenarios. The second notion is *robustness* (Cats and Jenelius, 2015; Snelder et al., 2012). *Robustness* of a public transport system is the ability to withstand or quickly recover from disturbances such as infrastructural and vehicular malfunctions and planned maintenance closures without significant reduction in the performance of the system (in terms of travel times, etc.). Robustness is defined as the degree to which a system is capable of functioning according to its design specifications in the case of serious disruptions. Despite the fact that we do not directly study the robustness and assume no flow on the road network due to a large disruption, information about the worst-case scenarios can enhance the road network to exclude them completely. *The importance of the link* is related to the impact that the closure of the link itself can have on the general functionality of the network as a whole (Rupi et al., 2015). The notion can also be related to connectivity.

2. Data and Methods

2.1 Road network data

We applied the methods, tested in this work, to four actual road networks representing four regions in the Czech Republic (Table 1).

Table 1. An overview of the regions and related road networks.

Region	Nodes	Links	People
Zlín	734	990	587,624
Karlovy Vary	727	984	366,638
Liberec	925	1,313	430,987
Ústí nad Labem	1,672	2,394	874,736

In order to demonstrate the current situation related to complete road link blockages, we analyzed daily data available from the Zlín region over 7/2014 – 7/2019 (Fig 1). The most frequently occurring were days when exactly one road link was blocked (29%), followed by days with a blockage-free road network (27%). Three and less road links were blocked in 80% of the days. The emergence of situations when more than three links were concurrently interrupted was not as frequent (20%), but still occurred. The links were not always interrupted simultaneously within a day, but the interruptions often temporally overlapped. Generally, the data represent all the disruptions in the network, both individual disruptions (e.g., as a result of a crash) as well as the incidents that caused simultaneous interruption of more than one link. Most of the time, it was not a single incident that caused all the interruptions on the particular day.

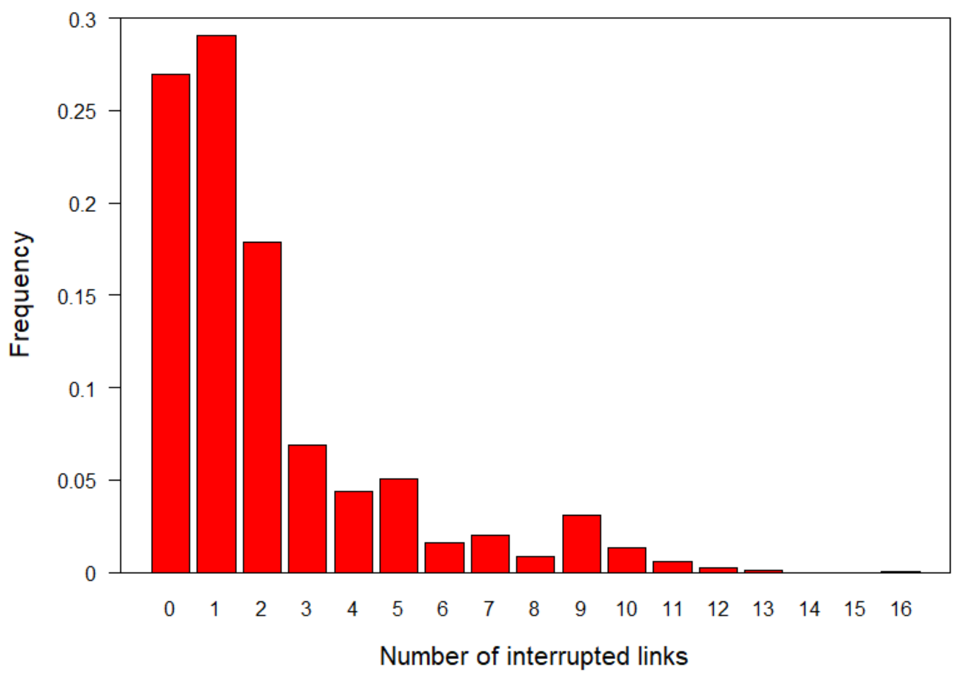

Figure 1. A histogram of interrupted links in the Zlín region for each day over 5 years (7/2014 – 7/2019). Notice: not all the traffic interruptions, which took place in the same day, necessarily overlapped temporarily. More precise data specifying hours when the interruption occurred were not available.

We decided to simulate the impacts of two model events which were caused by four and ten concurrently interrupted links. We also assume that the links are interrupted simultaneously as consequences of various events (e.g., natural disasters, road maintenance, traffic accidents, etc.) and are dispersed across the road network. The links are assumed to be impassable for at least several hours with a significant influence on the traffic volume.

2.2 *Methods – definition of network heuristics*

A brief description of the three selected heuristics is provided in this section. The deterministic algorithm used for the comparison is also mentioned. The pseudocodes and technical details of the corresponding algorithms can be found in the Appendix A.

Simulated annealing

Simulated annealing (SA) is physically motivated heuristics for finding the global minima. SA explores the state space using a slow decrease in the probability of accepting worse solutions, which prevents it from being stuck in local minima. SA is based on the Metropolis–Hastings algorithm and the Monte Carlo method to generate the sample states of a thermodynamic system (Metropolis et al., 1953). SA was used in cases where it was important to find at least one acceptable local optimum within a large discrete state space over a limited amount of time. The SA algorithm does not require further assumptions on the loss function, such as continuity, differentiability, or convexity. The SA algorithm was first presented and applied to the Ising spin glass problem in (Kirkpatrick et al., 1983; Cerny, 1985). A description of the algorithm can be found for instance in (Bertsimas and Tsitsiklis, 1993; Spall, 2003). An analysis of its convergence was studied in (Hajek, 1988; Granville et al., 1994). The convergence is closely related to the decrease in temperature (see discussions in Geman and Geman, 1984; Szu and Hartley, 1987; Press et al., 1992; Brooks and Morgan, 1995).

The next issue is how to define the neighborhoods and how to search them during the process (see Szu and Hartley, 1987; Brooks and Morgan, 1995; Bohachevsky et al., 1986; Styblinski and Tang, 1990; for various approaches). One of the possible modifications of the SA algorithm based on Bayesian ideas can be found in (Laud et al., 1989; Laarhoven et al., 1989). SA can be modified to improve its results and shorten the time of computation. In (Ingber, 1989; Ingber, 1993; Ingber, 1996), attention was primarily paid to the computational time. In (Miki et al., 2002; Martins et al., 2008; Tavares et al., 2011; Zhao, 2011), an adaptive neighborhood version of SA was studied and applied mainly in the continuous case. Applications of the SA algorithm in three problems, the graph partitioning problem, the graph coloring problem, and the number partitioning problem, were extensively studied in (Johnson et al., 1989; Johnson et al., 1991). Concerning road network optimization, simulated annealing was used in the network design (Miandoabchi et al., 2013; Farahani et al., 2013) and in the selection of the optimal location for healthcare centers (Jia et al., 2014). Over the years, new metaheuristics and heuristics have appeared with the aim of overcoming SA. SA was thus developed not only as a valuable heuristic but also as a benchmark for further attempts. We denote ANSA as the new version of the algorithm presented in Appendix A.

Guided local search

Guided Local Search (GLS) is based on a clever use of a local search (Voudouris, 1997; Voudouris, 1998; Voudouris and Tsang, 2003). The idea is based on penalization of the undesirable features of a problem or its solutions and a thorough local search. To prevent from becoming stuck in a local minimum, the algorithm can jump to another part of the state space after some time. A fast modification was developed for problems with large neighborhoods (Voudouris, 1997). The set of possible applications is the same as for SA. The typical application is in the Traveling Salesman Problem (Voudoris and Tsang, 1999). Other applications are in the quadratic assignment problem (Mills and Tsang, 2003), job scheduling (Nagata and Ono, 2018), vehicle routing problem (Zhong and Cole, 2005; Tarantilis et al., 2008; Barbucha, 2011; Turky et al., 2017), the multidimensional knapsack problem (Tairan et al., 2015) and network planning (Lucio et al., 2007). Many other applications of the algorithm exist in various fields (see for instance in Naanaa and Belghith, 2018, for the most recent ones). The algorithm is still alive because its modifications or improvements continue to appear (Shi et al., 2018).

Variable neighborhood search
The last approach tested in this paper is Variable Neighborhood Search (VNS) (Mladenovic and Hansen, 1997; Hansen and Mladenovic, 2001; Hansen et al., 2001; Hansen et al., 2008; Hansen et al., 2017). The heuristic is based upon predefined neighborhood structures which are thoroughly searched and rotated. The rotation depends on whether a better solution was found or not. The method was also applied to the classical travelling salesman problem (Mladenovic et al., 2014; Todosijevic et al., 2017; Hore et al., 2018; Wang et al., 2019), vehicle routing problem (Defryn and Sorensen, 2017; Sevkli and Guler, 2017; Xu and Cai, 2018; Simeonova et al., 2018), scheduling problems (Komaki and Malakooti, 2017; Tozzo et al., 2018; Pacheco et al., 2018), continuous optimization (Mladenovic et al., 2008; Carrizosa et al., 2012) and graph theory (Brimberg et al., 2009; Duarte et al., 2012). VNS was successfully used in many areas, see (Hansen et al., 2010; Mladenovic et al., 2016) for an overview.

Parallelized deterministic algorithm
Where possible, we compared the results of the algorithms based on the previously mentioned heuristics with the results of the parallelized deterministic algorithm (PDA) (Vodák et al., 2019). This algorithm is based on a process of removing cycles from the network and was developed for searching and evaluating the disintegrations of the network. Its main advantage is the ability to identify all the possible disintegrations and therefore it can also find exactly the worst combination of interrupted links. Its main disadvantage is that it can only manage problems up to 5 concurrently interrupted links in the networks used in the paper. This means that the most critical scenarios, representing cases with 6 and more concurrently interrupted links, cannot be processed by this algorithm.

2.3 *Methods – vulnerability measure*
The primary aim of these kinds of measures (also called loss functions) is to allow *ranking* of the results according to selected pre-defined criteria. Some measures only pay attention to the topology of the network (Ellens and Kooij, 2013). Others include further information such as traffic flow (Sohn, 2006), capacities of links (Sullivan et al., 2010; Watling et al., 2012), the number of people living in nodes (Vodák et al., 2019), the presence of important facilities in nodes (Novak and Sullivan, 2014), etc.

In short, it is possible to solve many kinds of problems depending on the various kinds of information we need to include. It is also important to consider the size of the respective state space and the related number of loss function evaluations. In this work, we only pay attention to the events when the road network disintegrated into two or more isolated parts.

The loss functions are applied to allow for quantification of various states of the network, i.e., the degree to which the network was damaged. It should therefore capture all relevant network characteristics. At the same time, due to the large number of interrupted links and used heuristics, the loss function should be evaluated as rapidly as possible. We therefore decided to use the simple loss function (also see Vodák et al., 2019):

$$F(G_c) = \sqrt{\frac{1}{m}\sum_{i=1}^{m+1}\left(P_i - \overline{P}\right)^2} \qquad (1)$$

where

$$\overline{P} = \frac{1}{m+1}\sum_{i=1}^{m+1} P_i$$

and m is the number of interrupted links, G_c is a graph representing the road network with m interrupted links from ordered vector c and P_i is the importance of the *i*-th component. The maximal number of components is *m+1*. If the network disintegrates into a smaller number of components, we assume that the importance of the remaining nonexistent components is equal to zero. The loss function is generally discontinuous in the state space. An extreme example can be demonstrated with a situation when two concurrently interrupted links caused the disintegration of the network into two components but repairing one of the links leads to the reconnection of the network. The problem we address in this work can be stated as follows:

For a given number (4 and 10) of (concurrently interrupted) links, a set of links needs to be found which will have, according to a defined loss function (vulnerability measure), the worst impact on the network.

The mathematical formulation of the problem is

$$\min_{c \in G} F(G_c) \tag{2}$$

where G is a graph representing the road network. In view of (1), the worst scenario corresponds to the case when *m* interrupted links cause the network disintegration into *m+1* components with the same importance. The value of the loss function is then zero. Otherwise it is positive. The idea behind the choice of the loss function is that the scenario when the network disintegrates into *m* components with the same importance is better than the above-mentioned worst case. We therefore seek for such a combination of interrupted links which minimizes the value of the loss function.

An example of the importance P_i is the number of people living in the *i*-th component. Another example of the importance P_i is if each component contains a medical facility with limited supplies which cannot be replenished.

The natural question is why not use a different measure. We decided on the measure due to its fast evaluation and the further properties described above. Moreover, we wanted to compare the results of the algorithms with known results for PDA. It is also possible to use another measure which is able to cope with the disintegrated network, for instance the measure based on OD pairs. The choice of the measure thus depends on what we want to measure. The study of the relationship among various vulnerability measures and the tested algorithms can be an interesting topic for future work. The only thing we need to keep in mind is that the time needed to evaluate the disintegrated network can affect the time of the whole computation or the accuracy of the results.

3. Results

The number of all the combinations of the links, which can be interrupted, were computed for each network in Table 2.

In case of four interrupted links, the results were also compared with the results of PDA which is able to determine all the network disintegrations for this setting. The analyses with ten concurrently interrupted links were, however, beyond the scope of the PDA approach.

Table 2. The numbers of all the combinations of links.

Region	4 links	10 links
Zlín	3.98×10^{10}	2.38×10^{23}
Karlovy Vary	3.88×10^{10}	2.24×10^{23}
Liberec	1.23×10^{11}	4.05×10^{24}
Ústí nad Labem	1.37×10^{12}	1.67×10^{27}

3.1 Results for four concurrently interrupted links

Concerning the time of the computations, the PDA algorithm produced its results between 8,900 and 98,000 s, depending on the network. For ANSA, GLS and VNS, we set the computational time at 86,400 s. The results of the simulations may, however, change due to the stochastic nature of the algorithms, but we did not observe fundamental changes after repeated computations. The convergence of the algorithms is only assured for time tending to infinity. This means the shorter the time of computation, the larger the variability in the results. The best ten results for all the algorithms are summarized in Fig 2.

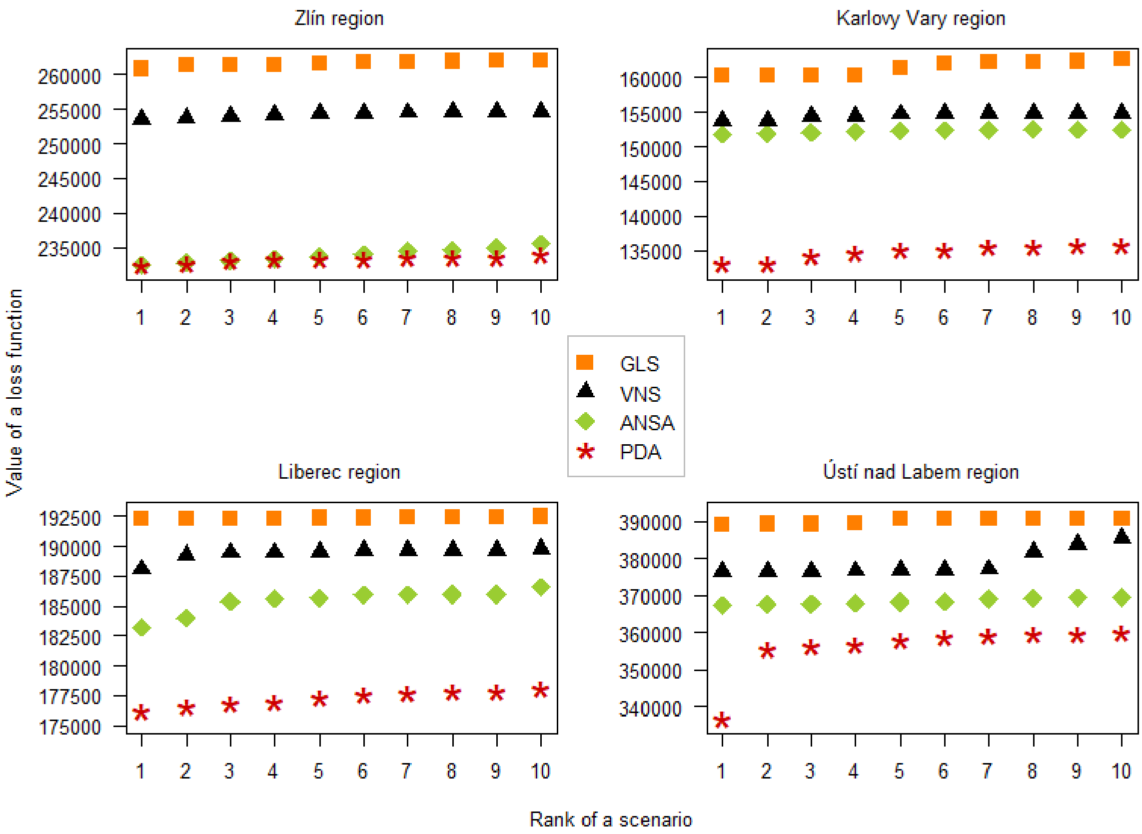

Figure 2. Values of the ten most critical scenarios for four concurrently interrupted links

We are aware that the results based only on the values of the loss function do not show the complete picture. In Table 3, we thus present the optimality of the results based upon the ranking. As PDA finds all the disintegrations of the network, we can assess how good the results of the heuristics are based on their rank within the PDA results. Therefore, PDA serves as a benchmark for the comparison of the heuristics not only using the values of the loss function but also in view of the ranking. The table presents the ranks of the best ten results of the heuristics in percentages, relative to all the results of PDA of the particular network. If *nan* is presented, the algorithm failed to find the disconnection caused by all four links, so the values are significantly higher than all the PDA results.

Table 3. Optimality of the results (in percentages).

Algorithms	Zlín	Karlovy Vary	Liberec	Ústí nad Labem
ANSA	0.0030	0.0020	0.0070–0.0090	0.0003
VNS	0.3590–1.0730	0.0040–0.0940	0.1000–1.2700	0.0070–0.6060
GLS	nan	3.1000–10.0000	36.0000–58.0000	10.3900–33.2400

The values of the optimality in Table 3 are presented in percentages, because the number of PDA results is different for each network, depending on its topology. Therefore, the results of the heuristics need to be compared to PDA on the particular network, not to each other. Fig 3 serves as an example of how the ranking in absolute values would look. We also see the values of the loss function. It needs to be displayed in a logarithmic scale and is still difficult to draw properly.

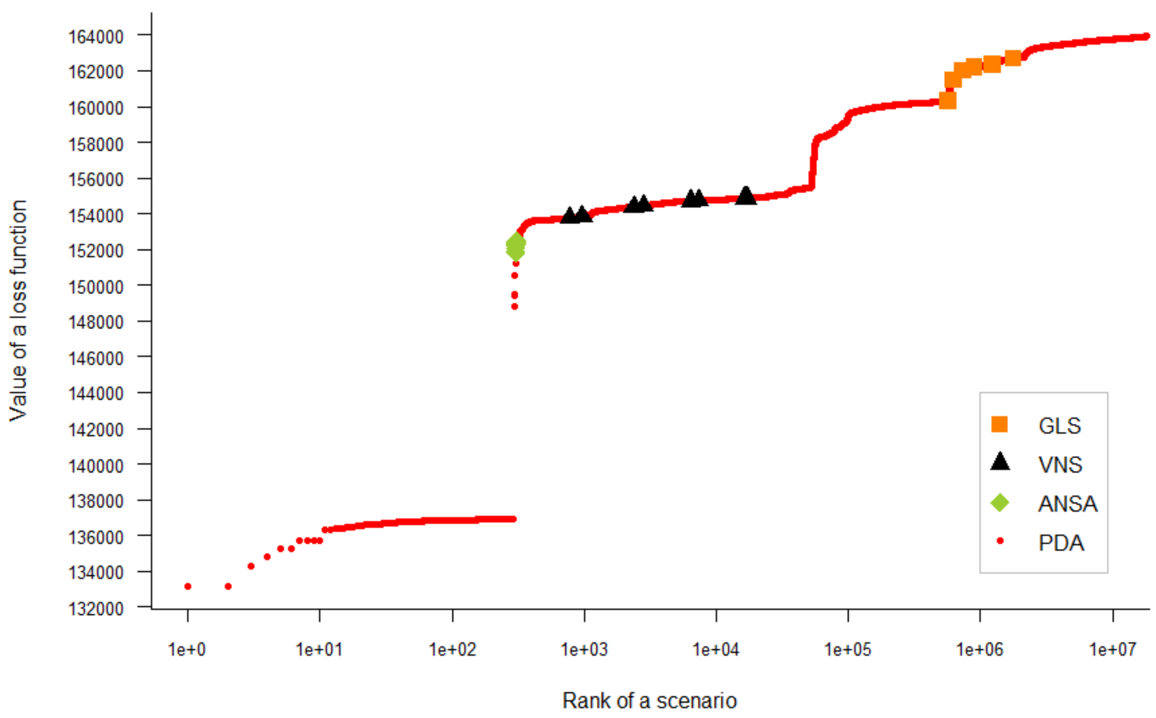

Figure 3. Ranks of heuristics within PDA and values of the loss function for disintegrations with four concurrently interrupted links in the Karlovy Vary region network. All 18,135,132 PDA results are drawn in the graph, due to the logarithmic scale.

We also calculated the rank correlation of links for the Karlovy Vary, Liberec and Zlín regions because the results for ANSA seem to be comparable for them according to Table 3. Since the links can be a part of many combinations and these combinations can have various values of the loss function, we determined for each link the combination with the lowest value of the loss function, i.e., the worst possible scenario. The Spearman's rank correlation was then calculated for ANSA, GLS and VNS versus PDA. We took the 100,000 best saved results for ANSA, GLS and VNS. The results are summarized in the following table. Despite the low correlations, it is apparent that the ANSA algorithm performed best.

Table 4. Rank correlation

	Zlín	Karlovy Vary	Liberec
ANSA/PDA	0.379	0.277	0.171
GLS/PDA	-0.05	0.195	-0.05
VNS/PDA	0.127	0	0.075

Figs 4–7 show the worst (the most critical) scenarios for four interrupted links found by ANSA on the chosen networks.

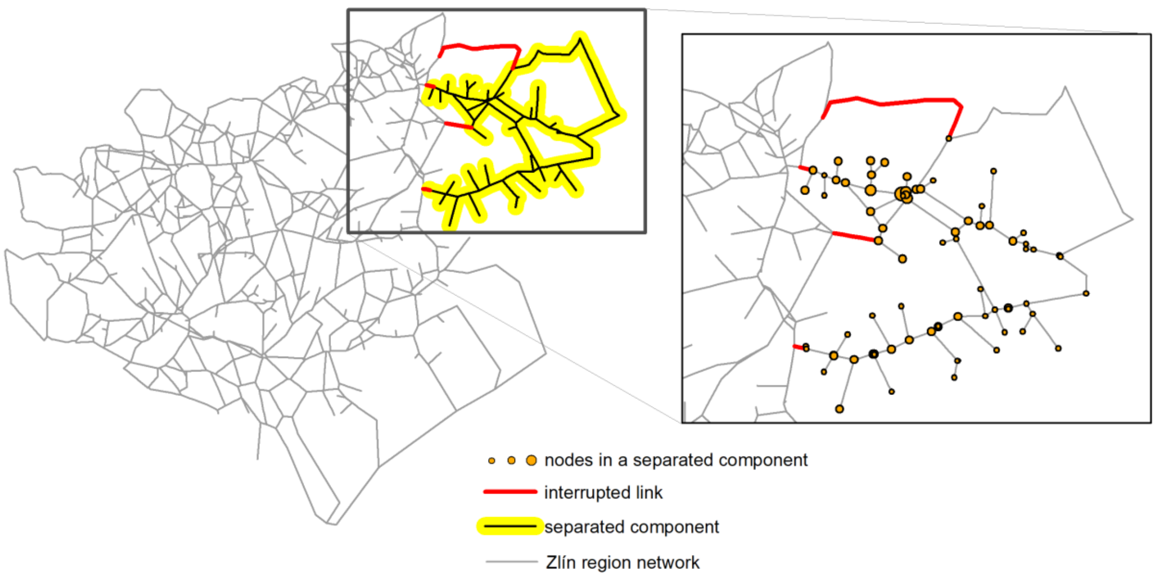

Figure 4. The worst scenario found by ANSA for 4four concurrently interrupted road links in the Zlín region network. 56,707 people (9.65%) cut off from the main network.

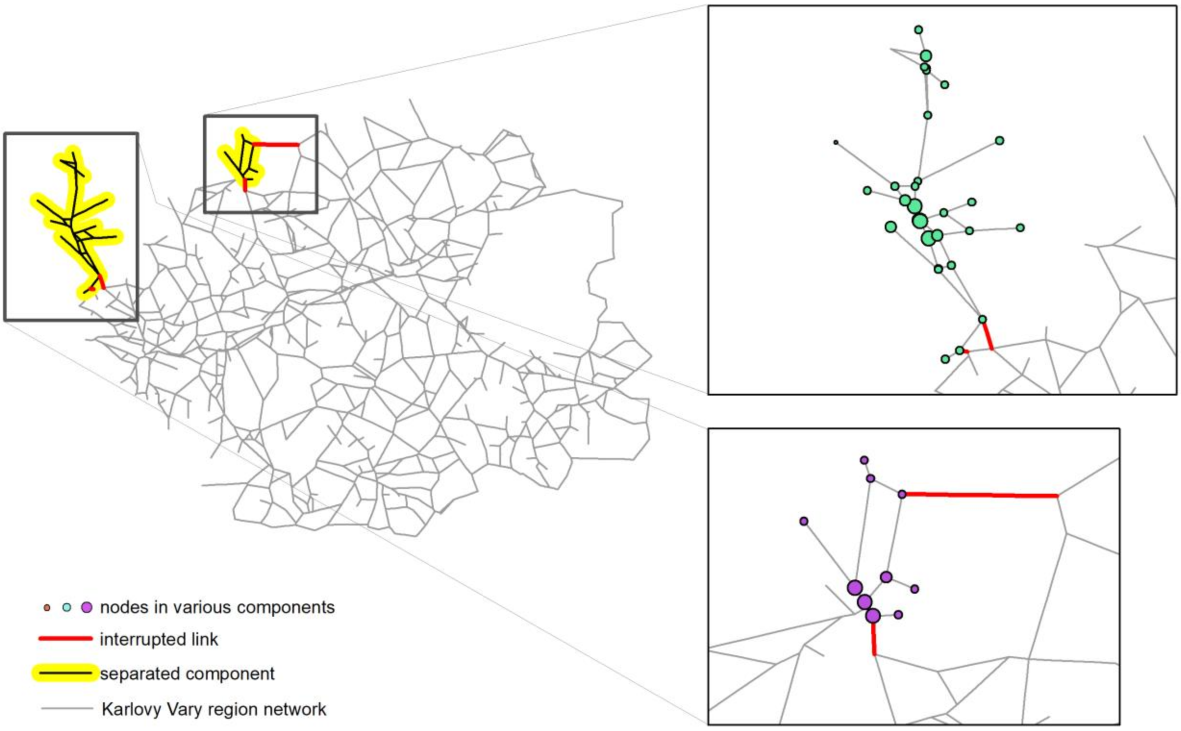

Figure 5. The worst scenario determined by ANSA for four concurrently interrupted road links in the Karlovy Vary region network. 22,024 people (6.01 %) in two components are cut off from the main network.

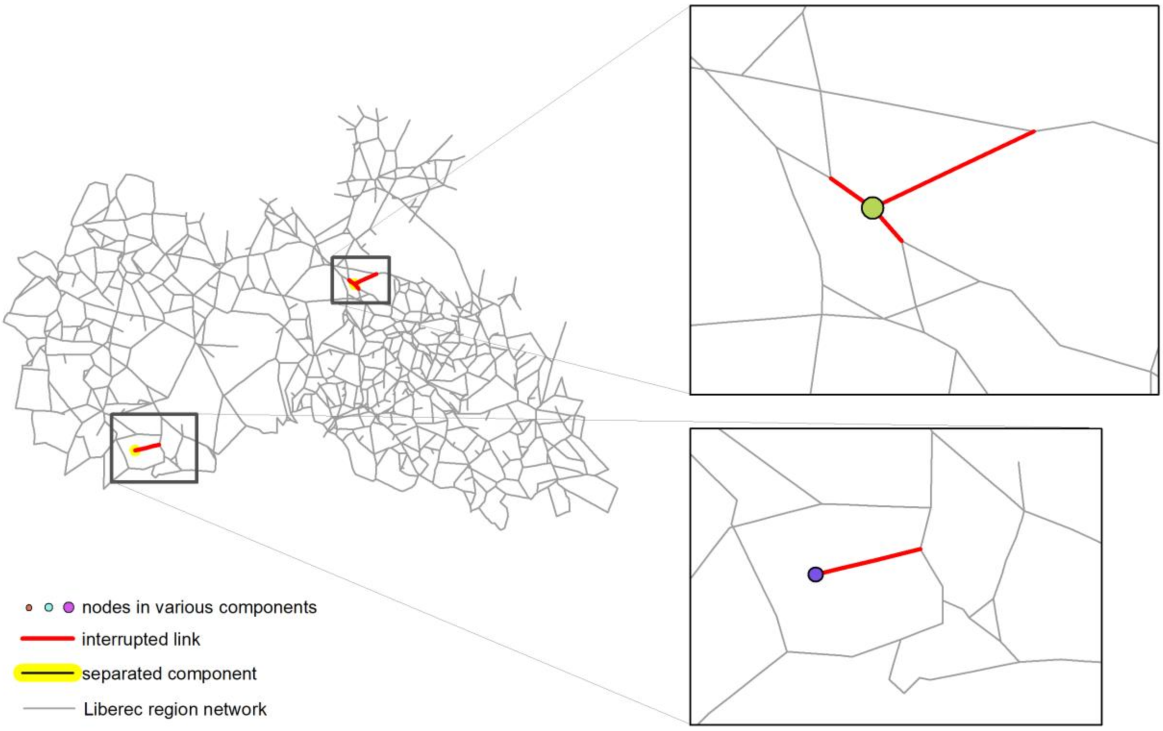

Figure 6. The worst scenario found by ANSA for four concurrently interrupted road links in the Liberec region network. 17,140 people (3.98%) cut off from the main network.

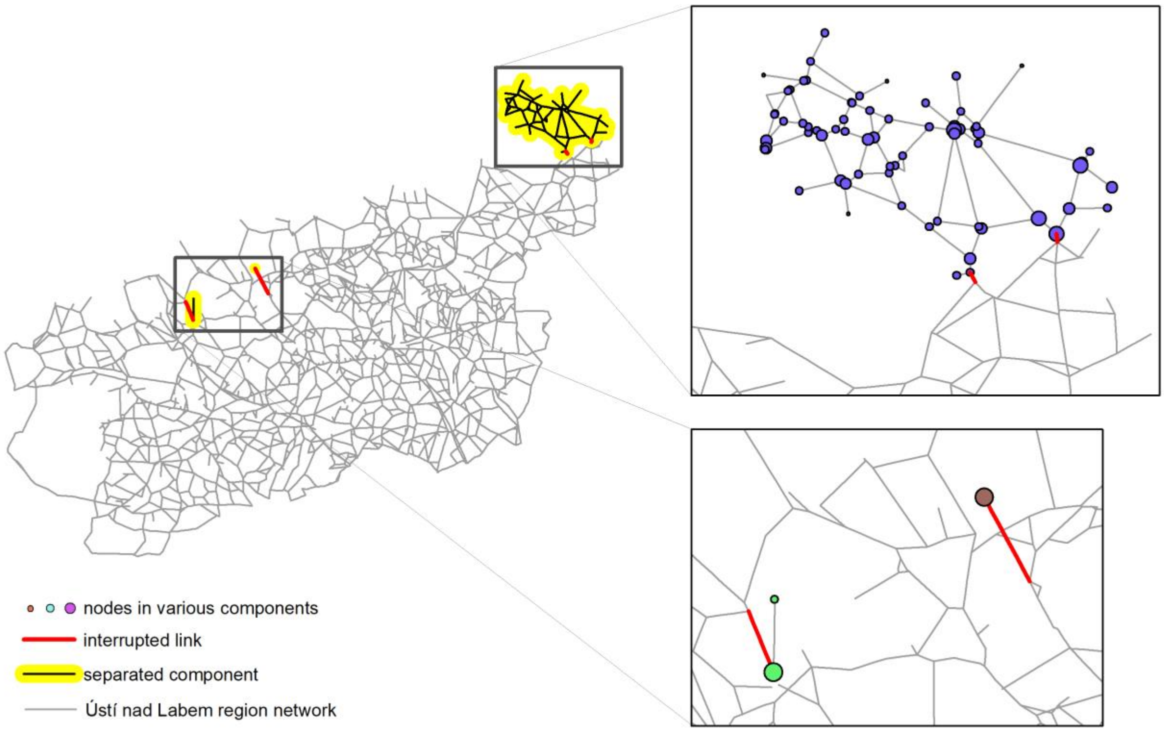

Figure 7. The worst scenario found by ANSA for four concurrently interrupted road links in the Ústí nad Labem region network. 42,712 (4.88%) people cut off from the main network.

3.2 Results for ten concurrently interrupted links

The computational times for ANSA, GLS and VNS were set at 259,200 s (i.e., 3 days). The results (Fig 8) were not compared with the outputs from the PDA, as the PDA approach is not applicable for more than five concurrently interrupted links on actual-size road networks due to the high computational demand.

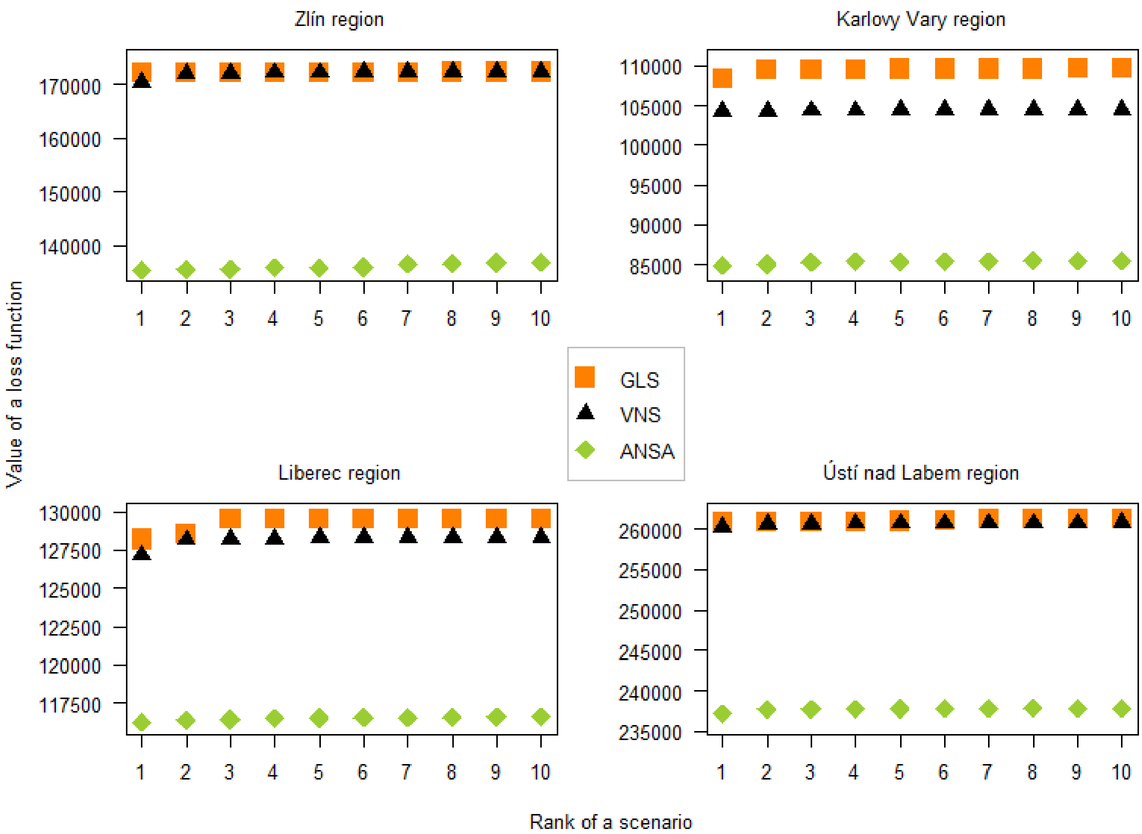

Figure 8. Values of the ten most critical scenarios determined by all the algorithms for ten concurrently interrupted links.

The shape of the loss function was similar as in Fig 3. but the curve profile was much steeper. Figs 9–12 show the worst scenarios found by ANSA.

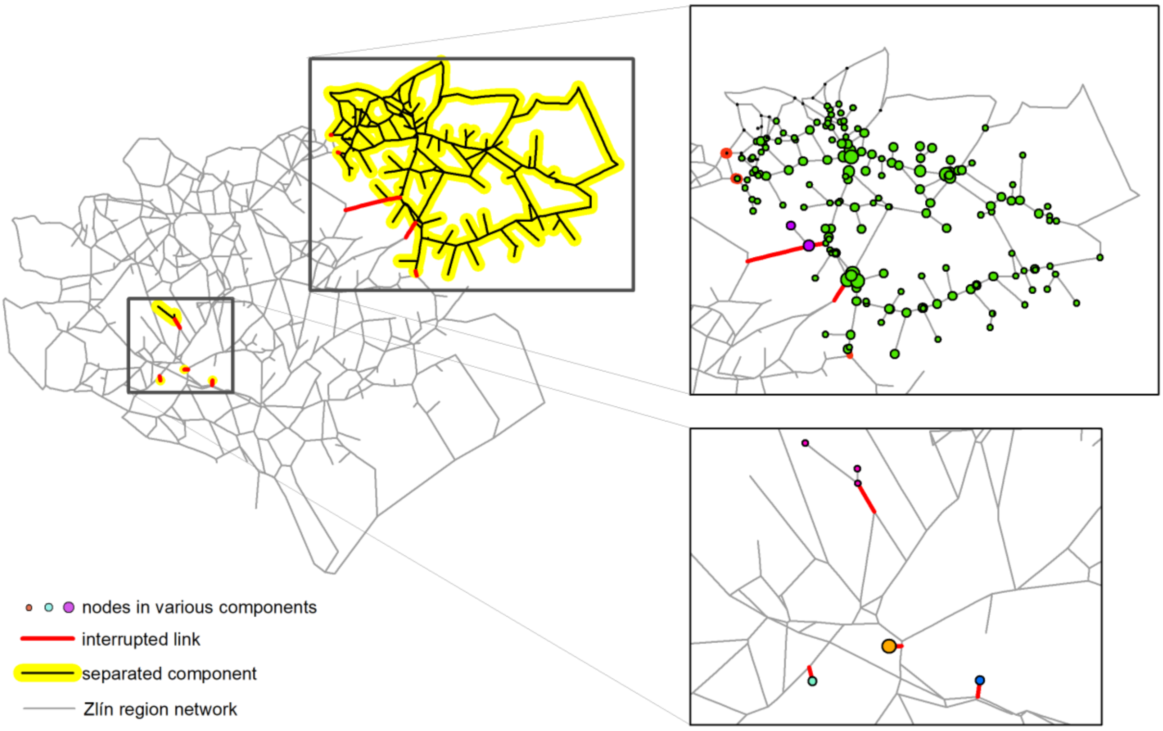

Figure 9. The worst scenario found by ANSA for ten concurrently interrupted road links in the Zlín region network. 142,136 people (24.19%) in six components cut off from the main network.

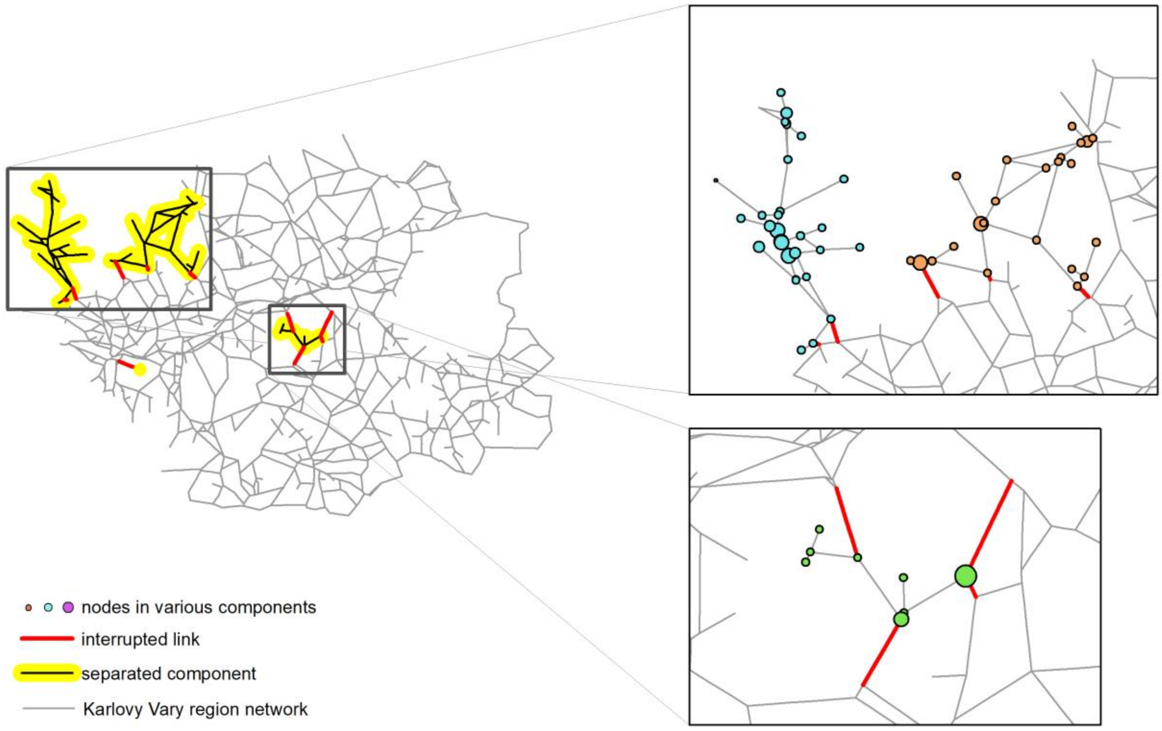

Figure 10. The worst scenario found by ANSA for ten concurrently interrupted road links in the Karlovy Vary region network. 82,427 people (22.48%) cut off from the main network.

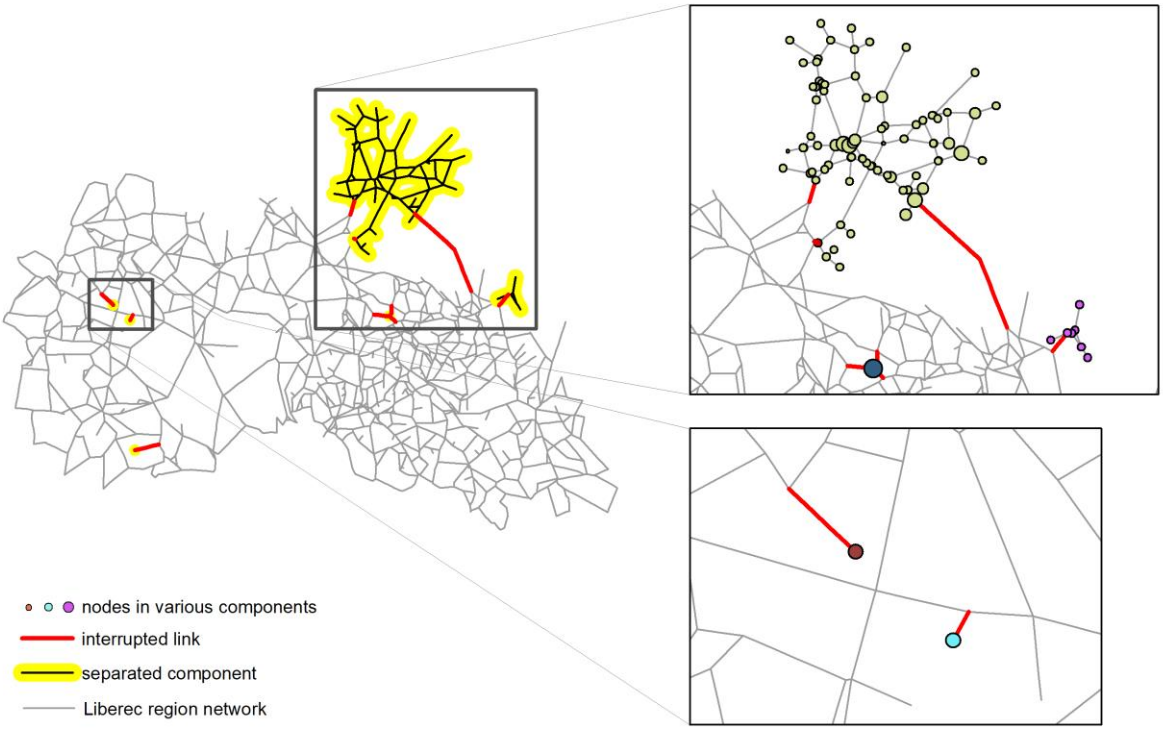

Figure 11. The worst scenario found by ANSA for ten concurrently interrupted road links in the Liberec region network. 41,877 people (9.72%) cut off from the main network.

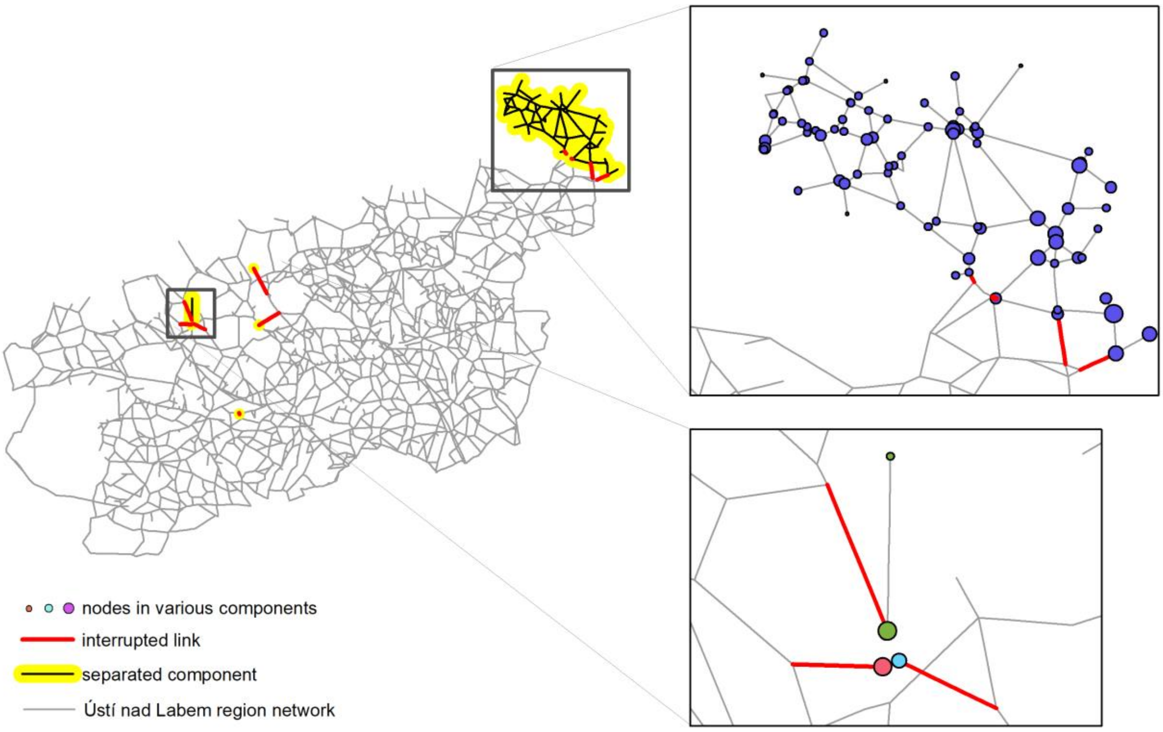

Figure 12. The worst scenario found by ANSA for ten concurrently interrupted road links in the Ústí nad Labem region network. 81,424 people (9.31%) cut off from the main network.

4. Discussion

In this work, we introduced and tested three algorithms (ANSA, GLS, VNS). We then compared their results with a deterministic algorithm (PDA), where possible. As test networks, we used four road networks of actual Czech regions with 984 – 2,394 road links. The general advantages and disadvantages of the algorithms can be summarized as follows (Table 5).

Table 5. Advantages and disadvantages of the algorithms.

Algorithms	Advantages	Disadvantages
PDA	able to find the most critical state / scenario, a smaller state space	incorporation of the restrictions for the smaller state space, it is only capable of working with a limited number of concurrently interrupted links
ANSA, GLS, VNS	finding values close to local minima in finite time, no restrictions on the state space, no limit on the number of interrupted links	finding the best solution for time tending to infinity, variability of the results in finite time, large state space

PDA, using circles (see Vodák et al., 2019 for detailed description), finds all the disintegrations of the network for a given number of links. The results for the entire state space are then evaluated. This means that PDA always finds the most critical solution. ANSA, GLS and VNS work differently. They choose a combination of links and evaluate the state. They then try to find a better solution (a solution with worse impacts in this case) in a certain neighborhood. If they find it, they continue searching in the neighborhood. Otherwise, they try to jump to another part of the state space. As a consequence, there is no expectation that these algorithms will determine the best solution (i.e., the scenario with the worst network impact in this particular application). The computational time for the algorithms can be limited, however, and we are still aware that their results are close to a local minimum. The problem is if the computational time is too short, different results for subsequent computations can be obtained due to the stochastic nature of the algorithms.

All the algorithms were tested for the same loss function. Its main qualification is that it is reasonable and fast to evaluate due to the large number of combinations of interrupted links. The algorithms evaluate respective combinations of interrupted links, but only the links, whose interruption causes a disintegration of the network (i.e., at least two separated subnetworks emerge), change the values of the loss function. This makes the problem more challenging.

We tested the algorithms for two situations with four and ten concurrently interrupted links on four road networks. The numbers of interrupted links were considered examples of damage caused by either moderate or more severe events. The computational times for all the algorithms (except PDA) were the same: 86,400 s (one day) for four interrupted links and 259,200 s (three days) for ten interrupted links. Despite this fact, we still have to take into account the possible variability of the results in finite time.

Considering the four interrupted links, the results were compared with the outputs of PDA as this approach was able to determine all the results. The best performance was reached with ANSA, although satisfactory results were also produced by VNS, although GLS failed (see Fig 2 and Fig 3). The same situation was repeated in case of the ten interrupted links. The results were influenced by the parameters set for the algorithms. We used the settings which were generally recommended, (see Appendix A for more details).

The main difference between the algorithms is in the way they explore the neighborhoods of a given combination of interrupted links and use the loss function. ANSA and VNS are based on the changes in the size and structure of the neighborhood which are growing if no better solution is found. ANSA also accepts with decreasing probability the worst solution to overcome local minima. A similar strategy is adopted by GLS which adds a penalization term to the loss function to obtain new combinations of interrupted links. GLS and VNS also incorporated a deterministic local search which analyzes combinations of neighboring links to the interrupted links (see Appendix A for more details). The different approach to the local search in a neighborhood seems to be responsible for the different result. In case of ANSA, the algorithm spends only a small portion of time in a neighborhood, if it is unable to improve the solution, and then jumps elsewhere. This not the case for GLS and VNS because they must spend the given time in a neighborhood. The natural question to answer is whether the result is network-related or not and how the results change for different types of networks. In Fig 13, we can see how the local minima of the loss function are found in time.

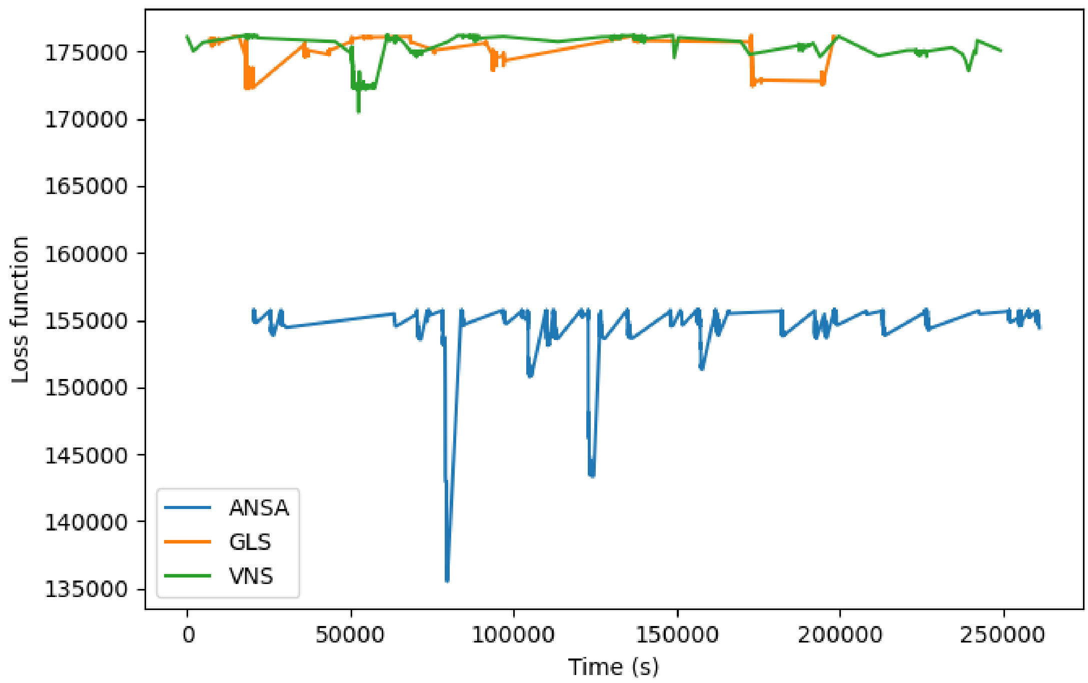

Figure 13. The value of the loss function for ten blocked links in the Zlín region over the course of computation. The graph shows how the values of the local minima emerged over time.

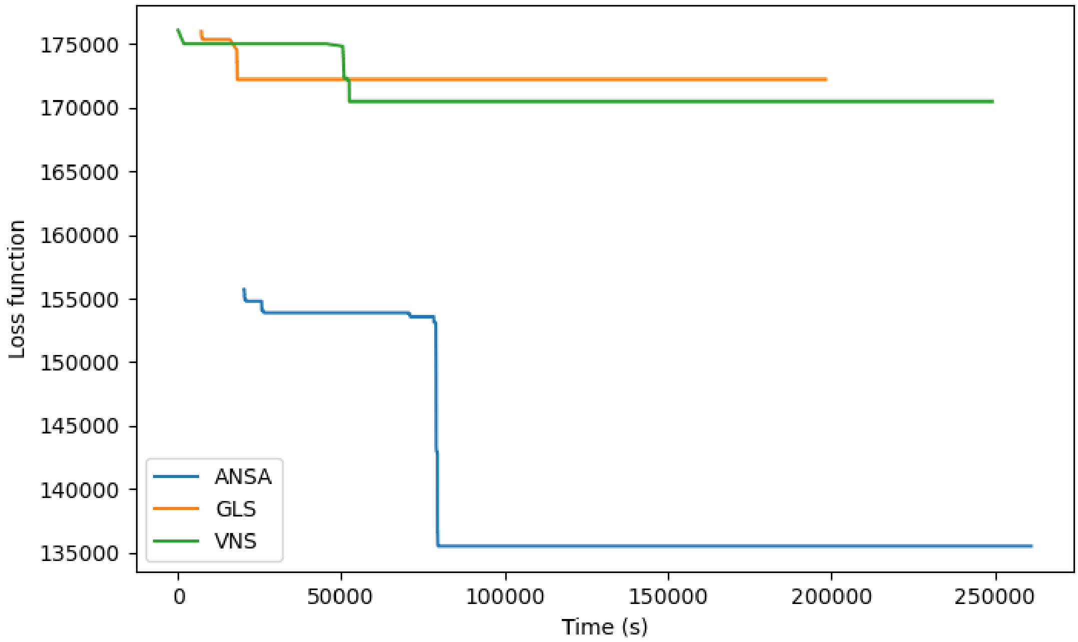

Figure 14. Time evolution of loss function minimum for ten blocked links in the Zlín region

The evolution of the best-found minimum value of the loss function in time can be seen in Fig 14. First, we must recall that the result is related to a particular simulation with ten interrupted links and can be different for another simulation. However, it is also apparent that ANSA seems to be more apt for further improvement of the values of the loss function than the other algorithms. We also presented Fig 3 with the shape of the loss function on part of the state space. It is apparent in the figure that the number of the worst cases for the loss function is strictly limited and finding them is not an easy task.

Our approach can be compared to those previously published. Many works (Taylor et al., 2006; Latora and Marchiori, 2004; Knoop et al., 2008; Erath et al., 2010; Newman, 2010; Wang et al., 2013) only considered a single interrupted link at a time, whereas we worked with four and ten concurrently blocked links. We did not apply any additional constraints as was done in (Chang and Nojima, 2001; Sohn, 2006; Bono and Gutiérez, 2011; Jenelius and Mattsson, 2012; Bíl et al., 2015a; Latora and Marchiori, 2004; Wang et al., 2013), where the authors usually only analyzed the impacts of either concrete negative or even disastrous events (e.g., concrete flooding events), but did not compare them with the worst possible network destruction. To compare the used road networks, interrupted links and the number of combinations, we summarized the respective information in Table 6.

Table 6. Road networks, number of interrupted links and state space used in the previously published papers.

Paper	Number of links in the network	Number of concurrently interrupted links	Number of combinations of interrupted links (state space)
Erath et al., 2010	30,289	1	30,289
Jenelius and Mattsson, 2012	86,940	It depends on the used grid.	Not mentioned explicitly but it should correspond to the number of the grid cells (3,170; 853; 241).
Knoop et al., 2008	468	1	468
Latora and Marchiori, 2004	96; 93	1	96; 93
Sohn, 2006	Unknown, 1,995 nodes	1	Unknown
Taylor et al., 2006	Unknown	1	Unknown
Wang et al., 2013	135; 38	Gradual removal of the links according to a given criterium	9,180; 741
Vodák et al., 2019	990	4	3.9×10^{10}
This work	984 to 2,394	4;	$3.9 \times 10^{10} 2.26 \times 10^{12}$,
		10	$2.2 \times 10^{23} 5.88 \times 10^{27}$

The added value of this paper is that there is no need to apply any restrictions on the number of links, their locations, and the size of the network. The price we pay for this is that we do not know the worst scenarios but only sufficiently bad ones. It is also natural to expect a lower quality of the results for large networks and/or a higher number of interrupted links than studied in the paper.

5. Conclusions

We focused on the development of algorithms capable of identifying those combinations of road links which, when interrupted concurrently, will have the gravest network impacts. This means, in the context of this work, a kind of network disintegration when a significant number of people lose their accessibility to other parts of network. The main advantage of the presented algorithms is that they do not require any additional mathematical properties of the loss function. The best results were achieved by the ANSA algorithm.

The algorithms cannot be used for real-time evaluation of networks during an emergency because of the long computation time spanning between one and three days. However, the current state of the network can be continuously evaluated in real time against a database of the worst-case scenarios which should be prepared beforehand. Managers will thus be able to identify whether the current state of the network is close to a critical state, i.e., close to network disintegration. Any application of such approaches would only require an online road network information system providing data on the current state of the network.

This approach can also be utilized in the phases of road maintenance planning and preparedness for emergency situations. A knowledge of the worst combinations of the weakest links is essential for road managers as it enables them to be prepared even for low-probability events with serious network impacts. The weak network links can further be, as part of preventive measures, physically strengthened in order to increase their physical robustness. Similarly, the entire network topology can be upgraded by adding new (redundant) links as alternatives to the weakest links.

Acknowledgements

This work was produced with the financial support of the Ministry of Transport within the program of long-term conceptual development of research institutions. Rostislav Vodák was further supported by the Ministry of Education, Youth and Sports of the Czech Republic under the project CZ.02.1.01/0.0/0.0/17_049/0008408 "Hydrodynamic design of pumps". Computations were carried out at CERIT-SC computing and storage facilities provided under the program Center CERIT Scientific Cloud, part of the Operational Program Research and Development for Innovations, reg. no. CZ.1.05/3.2.00/08.0144. Access to the CERIT-SC is greatly appreciated.

References

Barabasi AL, Posfai M. (2016). Network robustness. *Network Science*, 271-320.

Barbucha D. (2011). An Agent-Based Guided Local Search for the Capacited Vehicle Routing Problem. *Agent and multi-agent systems: Technologies and application,* 6682, 476-485.

Berberler ZN, Yigit E. (2018). Link vulnerability in networks. *International Journal of Foundations of Computer Science,* 29, 447-456.

Berdica K. (2002). An introduction to road vulnerability: what has been done, is done and should be done. *Transport Policy*, 9, 117-127.

Bertsimas D, Tsitsiklis J. (1993). Simulated Annealing. *Statistical Science,* 8(1), 10-15.

Bíl M, Andrášik R, Nezval V, Bílová, M. (2017). Identifying Locations along Railway Networks with the Highest Tree Fall Hazard. *Applied Geography*, 87, 45-53.

Bíl M, Vodák R, Kubeček J, Bílová M, Sedoník J. (2015). Evaluating road network damage caused by natural disasters in the Czech Republic between 1997 and 2010. *Transportation Research Part A: Policy and Practice*, 80, 90-103.

Bíl M, Vodák R. (2015). The stochastic approach in road network vulnerability analysis. *Safety and Reliability: Methodology and Applications – Proceedings of the European Safety and Reliability Conference, ESREL,* ;201, 929-932.

Bohachevsky I, Johnson ME, Stein ML. (1986). Generalized simulated annealing for function optimization. *Technometrics*, 28, 209-2017.

Bono F, Gutiérez E. (2011). A network-based analysis of the impact of structural damage on urban accessibility following a disaster: the case of the seismically damaged Port Au Prince and Carrefour urban road networks. *Journal of Transport Geography,* 19, 1443-1455.

Brimberg J, Mladenovic N, Urosevic D, Ngai E. (2009). Variable neighborhood search for the heaviest k-subgraph. *Computers & Operations Research*, 36(11), 2885-2891.

Brooks SP, Morgan BJT. (1995). Optimization using simulated annealing. *The Statistician,* 44, 241-257.

Carrizosa E, Drazic M, Drazic Z, Mladenovic N. (2012). Gaussian variable neighborhood search for continuous optimization. *Computers & Operations Research,* 39(9), 2206-2213.

Cats O, Jenelius E. (2015). Planning for the unexpected: The value of reserve capacity for public transport network robustness. *Transport Research Part A*, 81, 47-61.

Cerny V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications,* 45, 41-51.

Chang SE, Nojima N. (2001). Measuring post-disaster transportation system performance: the 1995 Kobe earthquake in comparative perspective. *Transportation Research Part A: Policy and Practice,* 35, 475-494.

Chen A, Yang C, Kongsomsaksakul S, Lee M. (2007). Network-based Accessibility Measures for Vulnerability Analysis of Degradable Transportation Networks. *Networks and Spatial Economics*, 7, 241-256.

Defryn C, Sorensen K. (2017). A fast two-level variable neighborhood search for the clustered vehicle routing problem. *Computers & Operations Research*, 83, 78-94.

Duarte A, Escudero LF, Marti R, Mladenovic N, Pantrigo JJ, Sanchez-Oro J. (2012). Variable neighborhood search for the Vertex Separation Problem. *Computers & Operations Research*, 39(12), 3247-3255.

Ellens W, Kooij RE. (2013). Graph measures and network robustness. *arXiv,* 1311.5064.

Erath A, Birdsall J, Axhausen KW, Hajdin R. (2010). Vulnerability assessment methodology for Swiss road network. *Transportation Research Record*, 2137, 118-126.

Farahani RZ, Miandoabchi E, Szeto WY, Rashidi H. (2013). A review of urban transportation network design problems. *European Journal of Operational Research*, 229, 281-302.

Geman S, Geman D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine intelligence,* 6(6), 721-741.

Gottlich S, Klar A. (2009). Modeling and Optimization of Scalar Flows on Networks. In Ambrosio, L, Bressan, A, Helbing, D, Klar, A, Zuazua, E, *Modelling and Optimisation of Flows on Networks*, Cetraro, Italy;2062,395-461.

Granville V, Krivanek M, Rasson J-P. (1994). Simulated annealing: A proof of convergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6), 652-656.

Hajek B. (1988). Cooling schedules for optimal annealing. *Mathematics of Operations Research,* 13, 311-329.

Hansen P, Mladenovic N. Variable neighborhood search: Principles and applications. European Journal of Operational Research 2001;130(3):449-467.

Hansen P, Mladenovic N, Perez-Britos D. (2001). Variable neighborhood decomposition search. *Journal of Heuristics,* 7(4), 335-350.

Hansen P, Mladenovic N, Moreno Perez JA. (2008). Variable neighbourhood search: methods and applications. *4OR-A Quarterly Journal of Operations Research,* 6(4), 319-360.

Hansen P, Mladenovic N, Todosijevic R, Hanafi S. (2017). Variable neighborhood search: basics and variants. Euro *Journal on Computational Optimization,* 5(3), 423-454.

Hansen P, Mladenovic N, Moreno P, Jose A. (2010). Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175(1), 367-407.

Helbing D. (2001). Traffic and related self-driven many-particle systems. *Reviews of Modern Physics*, 73(4), 1067-1141.

Helbing D. (2002). Order and disorder in traffic and self-driven many-particle systems. In Boccaletti S, Gluckman BJ, Kurths J, Pecora LM, Spano ML. *Experimental Chaos*. 622, 239-250.

Hohenemser C, Deicher M, Hofsass H, Lindner G, Recknagel E, Budnick J. (1986). Agricultural impact of Chernobyl – a warning. *Nature*, 321(6073), 817-817.

Hong-ying Y, Li-gun X. (2010). Measuring the Structural Vulnerability of Road Network: A Network Efficiency Perspective. *Journal of Shanghai Jiaotong University (Science),* 15(6), 736-742.

Hore S, Chatterjee A, Dewanji A. (2018). Improving variable neighborhood search to solve the traveling salesman problem. *Applied Soft Computing*, 68, 83-91.

Ingber L. (1989). Very fast simulated re-annealing. *Mathl. Comput. Modelling,* 12(8), 967-973.

Ingber L. (1993). Simulated annealing: Practice versus theory. *Mathl. Comput. Modelling*, 18(11), 29-57.

Ingber L. (1993). Adaptive simulated annealing (ASA): Lessons learned. *Control and Cybernetics*, 25(1), 33-54.

Jenelius E, Mattsson LG. (2012). Road network vulnerability analysis of area-covering disruptions: A grid-based approach with case study. *Transportation Research Part A: Policy and Practice,* 46(5), 746-760.

Jenelius E, Pettersen T, Mattson LG. (2006). Importance and exposure in road network vulnerability analysis. *Transportation Research Part A: Policy and Practice,* 40, 537-560.

Jenelius E. (2009). Network structure and travel patterns: explaining the geographical disparities of road network vulnerability. *Journal of Transport Geography*, 17, 234-244.

Jia T, Tao H, Qin K, Wang Y, Liu C, Gao Q. (2014). Selecting the optimal healthcare centers with a modified P-median model: a visual analytic perspective. *International Journal of Health Geographics,* 13, 42.

Johnson DS, Aragon CR, McGeoch LA, Schevon C. (1989). Optimization by simulated annealing: An experimental evaluation: Part I, graph partitioning. *Operations Research,* 37(6), 865-892.

Johnson DS, Aragon CR, McGeoch LA, Schevon C. (1991). Optimization by simulated annealing: An experimental evaluation: Part II, graph coloring and number partitioning. *Operations Research,* 39(6), 378-406.

Kirkpatrick S, Gelatt Jr CD, Vecchi MP. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671-680.

Knoop VL, Snelder M, van Zuylen HJ, Hoogendoorn SP. (2012). Link-level vulnerability indicators for real-world networks. *Transportation Research Part A: Policy and Practice*, 46, 843-854.

Knoop V, Zuylen H, Hoogendoorn S. (2008). The need of spillback modelling when assessing consequences of blockings in a road network. *European Journal of Transport and Infrastructure Research*, 8, 287-300.

Koerblein A, Kuechenhoff H. (2019). Perinatal mortality after the Fukushima accident: a spatiotemporal analysis. Journal *of Radiological Protection,* 39(4), 1021-1030.

Komaki M, Malakooti B. (2017). General variable neighborhood search algorithm to minimize makespan of the distributed no-wait flow shop scheduling problem. *Production Engineering–Research and Development,* 11(3), 315-329.

Kurauchi F, Uno N, Sumalee A, Seto Y. (2009). Network evaluation based on connectivity vulnerability. In Lam HK, Wong SC, Lo HK (Eds.), *Transportation and Traffic Theory*. Springer. 637-649.

Lakner Z, Szabo E, Szucs V, Szekacs A. (2018). Network and vulnerability analysis of international spice trade. *Food Control,* 83, 141-146.

Lammer S, Kori H, Peters K, Helbing D. (2006). Decentralised control of material or traffic flows in networks using phase-synchronisation. *Physica A: Statistical mechanics and its applications.* 363(1), 39-47.

Latora V, Marchiori M. (2004). How the science of complex networks can help developing strategies against terrorism. *Chaos, Solitons and Fractals,* 20, 69-75.

Laud P, Berliner LM, Goel PK. (1989). A stochastic probing algorithm for global optimization. Computing Science and Statistics: *Proceedings of the 21st Symposium on the Interface*, American Statistical Association, Alexandria, VA. 220-229.

Leng JQ, Zhai J, Li QW, Zhao L. (2018). Construction of road network vulnerability evaluation index based on general travel cost. *Physica A: Statistical mechanics and its applications,* 493, 421-429.

Lucio GF, Reed MJ, Henning ID. (2007). Guided local search as a network planning algorithm that incorporates uncertain traffic demands. *Computer networks.* 51(11), 3172-3196.

Martins, TC, Sato AK, Tsuzuki MSG. (2008). Adaptive Neighborhood Heuristics for Simulated Annealing over Continuous Variables. In Cher Ming Tan *Chapter in Simulated Annealing*, IntechOpen.

Matisziw TC, Murray AT. (2009). Modeling s-t path availability to support disaster vulnerability assessment of network infrastructure. *Computers & Operations Research*, 36, 16-26.

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. (1953). Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics,* 21(6), 1087-1092.

Miandoabchi E, Daneshzand F, Szeto WY, Farahani RZ. (2013). Multi-objective discrete urban road network design. *Computers & Operations Research,* 40, 2429-2449.

Miki M, Hiroyasu T, Ono K. (2002). Simulated annealing with advanced adaptive neighborhood. *Computational Intelligence and Applications,* 113-118.

Mills P, Tsang E. (2003). Applying an extended guided local search to the quadratic assignment problem. *Annals of operations research*, 118(1-4), 121-135.

Mladenovic N, Hansen P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11), 1097-1100.

Mladenovic N, Todosijevic R, Urosevic D. (2014). Two level General variable neighborhood search for Attractive traveling salesman problem. *Computers & Operations Research*, 52, 341-348.

Mladenovic N, Drazic M, Kovacevic-Vujcic V, Cangalovic M. (2008). General variable neighborhood search for the continuous optimization. *European Journal of Operational Research*, 191(3), 753-770.

Mladenovic N, Melian B, Maria B. (2016). Special issue: Applications of variable neighbourhood search. *IMA Journal of Management Mathematics,* 27(1), 1-2.

Naanaa A, Belghith S. (2018). An efficient guided local search approach for multiuser detection in UWB systems. *Physical communication,* 26, 141-148.

Nagata Y, Ono I. (2018). A guided local search with iterative ejections of bottleneck operations for the job shops scheduling. *Computers & Operations Research,* 90, 60-71.

Newman M. (2010). *Networks: An introduction.* Oxford University Press.

Novak DC, Sullivan JL. (2014). A link-focused methodology for evaluating accessibility to emergency services. *Decision Support Systems,* 57, 309-319.

Pacheco J, Porras S, Casado S, Baruque B. (2018). Variable neighborhood search with memory for a single-machine scheduling problem with periodic maintenance and sequence-dependent set-up times. *Knowledge-Based Systems,* 145, 236-249.

Press WH, Teukolsky SA, Vetterburg WT, Flannery BR. (1992). *Numerical recipes in C.* Cambridge University Press, Cambridge.

Rupi F, Angelini S, Bernardi S, Danesi A, Rossi G. (2015). Ranking links in a road transport network: a practical method for the calculation of link importance. *Transportation Research Procedia,* 5, 221-232.

Schieber TA, Carpi L, Frery AC, Rosso OA, Pardalos PM, Ravetti MG. (2016). Information theory perspective on network robustness. *Physics Letters A,* 380(3), 359-364.

Scott, DM, Novak, DC, Aultman-Hall L, Guo F. (2006). Network Robustness Index: A new method for identifying critical links and evaluating the performance of transportation networks. *Journal of Transport Geography,* 14, 215-227.

Sevkli AZ, Guler B. (2017). A multi-phase oscillated variable neighbourhood search algorithm fora real-world open vehicle routing problem. *Applied Soft Computing*, 58, 128-144.

Shi J, Zhang Q, Tsang E. (2018). EB-GLS: an improved guided local search based on the big valley structure. *Memetic computing,* 10(3), 333-350.

Simeonova L, Wassan N, Salhi S, Nagy G. (2018). The heterogeneous fleet vehicle routing problem with light loads and overtime: Formulation and population variable neighbourhood search with adaptive memory. *Expert Systems with Applications,* 114, 183-195.

Snelder M, van Zuylen HJ, Immers LM. (2012). A framework for robustness analysis of road networks for short term variations in supply. *Transportation Research Part A: Policy and Practice,* 46, 828-842.

Sohn J. (2006). Evaluating the significance of highway network links under flood damage: An accessibility approach. *Transportation Research Part A: Policy and Practice,* 40, 491-506.

Spall JC. (2003). *Introduction to Stochastic Search and Optimization*. Wiley.

Styblinski MA, Tang TS. (1990). Experiments in Nonconvex Optimization: Stochastic Approximation with Function Smoothing and Simulated Annealing. *Neural Networks,* 3, 467-483.

Sullivan JL, Novak DC, Aultman-Hall L, Scott DM. (2010). Identifying critical road segments and measuring system-wide robustness in transportation networks with isolating links: A link-based capacity-reduction approach. *Transportation Research Part A: Policy and Practice,* 44, 323-336.

Szu H, Hartley R. (1987). Fast Simulated Annealing. *Physics Letters A*, 122, 157-162.

Tairan N, Algarni A, Varghese J, Jan MA. (2015). Population-Based Guided Local Search for Multidimensional Knapsack Problem, *Conference: 4th International Conference on Future Generation Communication Technologies*.

Tarantilis C, Zachariadis E, Kiranoudis CT. (2008). A hybrid guided local search for the vehicle-routing problem with intermediate replenishment facilities. *Informs Journal on Computing,* 20(1), 154-168.

Tavares RS, Martins TC, Tsuzuki MSG. (2011). Simulated annealing with adaptive neighborhood: A case study in off-line robot path planning. *Expert Systems with Applications,* 38(4), 2951-2965.

Taylor MAP, Sekhar SVC, D'Este GM. (2006). Application of Accessibility Based Methods for Vulnerability Analysis of Strategic Road Networks. *Networks and Spatial Economics*, 6, 267-291.

Taylor MAP. (2017). *Vulnerability Analysis for Transportation Networks.* Elsevier, Amsterdam.

Todosijevic R, Mjirda A, Mladenovic M, Hanafi S, Gendron B. (2017). A general variable neighborhood search variants for the travelling salesman problem with draft limits. *Optimization Letters,* 11(6), 1047-1056.

Tozzo E, Cotrim SL, Galdamez EVC, Leal GCL. (2018). A genetic algorithm and variable neighborhood search for the unrelated parallel machine scheduling problem with sequence dependent setup time. *Acta Scientiarum-Technology,* 40 (UNSP e36607).

Turky A, Moser I, Aleti A. (2017). An Iterated Local Search with Guided Perturbation for the Heterogeneous Fleet Vehicle Routing Problem with Time Windows and Three-Dimensional Loading Constraints. In Wagner M, Li X, Hendtlass T (eds). *Artificial life and computational intelligence.* Lecture Notes in Computer Science. 10142:279-290.

Van Laarhoven P, Boender C, Aarts E, Kan A. (1989). ABayesian approach to simulated annealing. *Probability in the Engineering and Informational Sciences.* 3(4), 453-475.

Vodák R, Bíl M, Sedoník J. (2015). Network robustness and random processes. *Physica A: Statistical mechanics and its applications,* 428, 368-382.

Vodák R, Bíl M, Svoboda T, Křivánková Z, Kubeček J, Rebok T, Hliněný, P. (2019). A Deterministic Approach for Rapid Identification of the Critical Links in Networks. *PLOS ONE,* 14(7), e0219658.

Voudouris C. (1997). Guided local search for combinatorial optimisation problems. PhD Thesis, Department of Computer Science, University of Essex, Colchester, UK.

Voudouris C. (1998). Guided Local Search – an illustrative example in function optimisation. *BT Technology Journal*, 16(3), 46-50.

Voudouris C, Tsang E. (2003). Guided Local Search. In Glover F, Kochenberger GA (eds). *Handbook of Metaheuristics. International Series in Operations Research & Management Science*. Springer, Boston, MA.

Voudouris C, Tsang E. (1999). Guided local search and its application to the traveling salesman problem. *European journal of operational research,* 113(2), 469-499.

Watling D, Milne D, Clark S. (2012). Network impacts of a road capacity reduction: Empirical analysis and model predictions. *Transportation Research Part A: Policy and Practice,* 46, 167-189.

Wang S, Hong L, Ouyang M, Zhang J, Chen X. (2013). Vulnerability analysis of interdependent infrastructure systems under edge attack strategies. *Safety Science,* 51, 328-337.

Wang X, Golden B, Wasil E. (2019). A Steiner Zone Variable Neighborhood Search Heuristic for the Close-Enough Traveling Salesman Problem. *Computers & Operations Research*, 101, 200-219.

Wei DJ, Zhang XG, Mahadevan S. (2018). Measuring the vulnerability of community structure in complex networks. *Reliability Engineering and System Safety*, 174, 41-52.

Xu Z, Cai Y. (2018). Variable neighborhood search for consistent vehicle routing problem. *Expert Systems with Applications,* 113, 66-76.

Zahradníček P, Münster P, Bíl M, Skalák P, Štěpánek P, Farda A, Panský, M., Brzezina, J., Bílová, M., Kubeček, J. (2018). The December 2014 glaze event in the Czech Republic: predictability and impacts. *Weather*, 73, 375-382.

Zhao X. (2011). Simulated annealing algorithm with adaptive neighborhood. *Applied Soft Computing*, 11(2), 1827-1836.

Zhong YJ, Cole MH. (2005). A vehicle routing problem with backhauls and time windows: a guided local search solution. *Transportation Research Part E: Logistics and Transportation Review*, 41(2), 131-144.

Appendix A

We present more details about the algorithms used in this paper in the appendix.

Simulated annealing and ANSA

We introduce a modification of SA which can be called *adaptive neighborhood simulated annealing* (ANSA). The presented modifications of the SA algorithm stem from the algorithm developed in (Bíl and Vodák, 2015b). The main issue related to SA is how to ensure that the algorithm explores a sufficiently large part of the state space and at the same time how to improve current solution searching in its neighborhood. First, we introduce the whole algorithm and then we pay attention to the problem related to the neighborhood. ANSA contains the following variables:

numOfLinks	number of concurrently interrupted links
bestResults	a list of a given size of the worst obtained scenarios which should be saved
n	number which controls the adaptive searching in neighborhoods
minPercent	minimal percentage of results which have to be accepted, otherwise the algorithm tests whether it ends or not, depending on *notFoundBetter*
notFoundBetter	maximal number of the successive cases during which the minimal percentage of results is not accepted and a better solution is not found
tempFactor	number which ensures the decrease in the temperature
initTemperature	initial temperature
sizeFactor	number of repetitions before the temperature is decreased

The algorithm ANSA can be described by the pseudocode:

Algorithm A.1 *Simulated annealing – adaptive neighborhood version*

```
while EndCriterion
        S ← GetInitialSolution()
        noBetter ← 0
        nPom ← n
        T ← initTemperature
        bestCost ← Inf
        while noBetter < notFoundBetter
                numNewValuesCost ← 0
        testBetter ← 0
                for j ← 1:sizeFactor
                        numOfChangedLinks ← GenOfNum(nPom, length(S))
                        S′ ← PickRandomNeighbor(numOfChangedLinks, S)
                        δ ← Cost (S′) - Cost (S)
                        if δ ≤ 0 or randomly generated number < e^(-δ/T)
                                nPom ← n
                                numNewValuesCost ← numNewValuesCost + 1
                                if bestCost > Cost(S′)
                                        bestCost ← Cost(S′)
                                        testBetter ← 1
                                        noBetter ← 0
                                if S′ should belong to bestResults
                                        bestResults ← S′
                                S ← S′
                        else
                        nPom ← max (nPom – 1, 1)
                percent ← numNewValuesCost / sizeFactor * 100
                if percent < minPercent and testBetter = 0
                        noBetter ← noBetter + 1
                T ← tempFactor * T
```

The initial solution of the algorithm is generated randomly. The variables *notFoundBetter* and *minPercent* prevent the algorithm from being stuck in a local minimum under low temperature. If such a situation occurs, then one cycle of the algorithm ends, a new initial solution is generated and the entire process is restarted. The key part of the ANSA is the function *GenOfNum()* which generates the number of links which are replaced in solution *S* (an actual combination of interrupted links). In fact, the function determines the size of the neighborhood. The size of the neighborhood is driven by variables *nPom* in Algorithm A.1 and by *par* in Function A.2. The variables determine the probabilities of the number of links which are replaced. It enables combining the local search with a jump to another part of the state space.

Function A.2 *Function GenOfNum(par, numOfLinks) for definition 1*

$$prob \leftarrow \left[\frac{1}{par^1}, \frac{1}{par^2}, \dots, \frac{1}{par^{numOfLinks}}\right]$$
$$prob \leftarrow \frac{prob}{\sum(prob)}$$

return an integer from interval [1, *numOfLinks*] using probabilities in *prob*

One possible distribution of the probabilities is [0.54, 0.26, 0.13, 0.07]. By decreasing the value of the variable *nPom*, we increase the probability that a higher number of interrupted links is replaced and thus the size of the neighborhood increases.

GLS

For GLS, we introduce a modified version of the pseudo-code from (Voudouris and Tsang, 2003) with the variables

graph	a road network represented by a graph
numb_links	number of interrupted links
s_best, s_new	the best found and current interrupted links
s_best_value, s_new_value	values of the loss function
$p = [p_1, \dots, p_{numb_{links}}]$	penalizations

Algorithm A.3 *Guided local search*

while *EndCriterion*:

  s_best ← init_solution(*graph, numb_links*)

  s_best_value ← loss_funct(*graph, s_best*)

  p ← set_penalization(*numb_links*)

  while *BetterSolutionsFound and TimeCriterion*:

    s_new, s_new_value = local_search(*graph, s_best,* loss_funct, loss_funct_aug)

    p ← penalization(s_new, p)

    s_best, s_best_value = select_best(*s_new, s_new_value, s_best, s_best_value*)

We now describe the particular functions in Algorithm A.3. Function *init_solution()* generates an initial solution which consists of a given number of interrupted links. Function *loss_funct()* returns the value of the loss function defined in (1). Function *set_penalization()* sets the initial state of the list p. In the paper, all the initial values of p equal zero. Function *local_search()* searches the neighborhood of the so far best found solution for possible improvements. The function goes through all the combinations of neighboring links to the interrupted links. We only limited the time which the algorithm spends in the types of neighborhood. All the time restrictions are listed in the *Settings* part at the end of Appendix. The new solutions are not evaluated with *loss_funct()* but with *loss_func_aug()*. The function *loss_func_aug()* is computed according to the formula

$$h(s) = f(s) + \lambda \sum_{i=1}^{numb_{links}} p_i I_i(s) \ (3)$$

where f is the original loss function, p_i is a penalization of the i-th link and I_i is an indicator function which equals 1 if the i-th link belongs to s and 0 otherwise and λ is defined as follows

$$\lambda = \frac{\alpha \cdot is}{numb_{edges}} \ (4)$$

For more details about the setting and penalization, we refer the reader to (Voudouris and Tsang, 2003). The neighborhood is understood in a topological sense. This means, for any of the links in a given combination, its end nodes and their outgoing links. Using the links, we construct all possible combinations of a given length *numb_links*. The combinations form a neighborhood which is searched for.

VNS

The algorithm VNS used in this paper can be described by the pseudo-code (compare with basic VNS in (Hansen and Mladenovic, 2001).

Algorithm A.4 *Variable neighborhood search*

```
N ← generate_neigh_structures(numb_links)
while EndCriterion:
        k ← 0
        s_best ← init_solution(graph, numb_links)
        s_best_value ← loss_funct(graph, s_best)
        while k < max_comb  and TimeCriterion:
                s_new = init_sol_neigh(graph, s_best, N[k])
                s_new_value = loss_funct(graph, s_new)
                s_new, s_new_value = local_search(graph, s_new,s_new_value, loss_funct)
                s_best, s_best_value = select_best(s_new, s_new_value, s_best, s_best_va          if
s_new_value < s_best_value:
                        s_best_values = s_new_value
                        k ← 0
        else
                        k ← k+1
```

where the variables are

graph	a road network represented by a graph
numb_links	number of interrupted links
s_best, s_new	the best found and current interrupted links
s_best, s_best_value	values of the loss function
$N = [N_1, \ldots, N_{max_{comb}}]$	neighborhood structures

and

$$max_{comb} \sum_{i=1}^{numb_{links}} \binom{numb_{links}}{i}$$

The list of neighborhood structures *N* is constructed in such a way that its *i*-th component $N[i]$ contains a tuple of numbers. The length of the tuple ranges from 1 to *numb_links* and the numbers represent positions of links in *s_best* which should be replaced. In this manner, we ensure that the algorithm explores a close neighborhood of *s_best*. At the same time, it can jump to another part of the state space, if it is unable to improve the current best solution. The functions *init_solution()*, *loss_funct()*, *local_search()* and *select_best()* are the same as in GLS. The penalization from GLS is replaced here by the function *init_sol_neigh()*. The function randomly generates new links which substitute the links on positions given by $N[k]$

Settings

It is important to establish the parameters for all the algorithms. First, we restricted the time of computation. We use 86,400 s for four interrupted links and 259,200 s for ten interrupted links. In case of ANSA, we put *minPercent* = 2, *notFoundBetter* = 10, *initTemperature* = 1,600, *numb_prob* = 10, *sizeFactor* = 10,000 and *tempFactor* = 0.9 for four and ten interrupted links. In case of GLS, we put α = 0.2. If the algorithm is unable to find a better solution in 3,600 s, then it is restarted. The time restriction for the function local_search() *is 60 s. The same time restrictions are used for VNS.*

The algorithms were realized in Python 3.6 using OS Linux Debian 10, 1 CPU AMD EPYC 7351 2.4 GHz, 300 MB RAM.