Going the Extra Mile with Disaster-Aware Network Augmentation

Oostenbrink, J.; Kuipers, F.A.

Publication date
2021

Document Version
Accepted author manuscript

Published in
INFOCOM 2021 - IEEE Conference on Computer Communications

Citation (APA)
Oostenbrink, J., & Kuipers, F. A. (Accepted/In press). Going the Extra Mile with Disaster-Aware Network Augmentation. In *INFOCOM 2021 - IEEE Conference on Computer Communications* IEEE .

Going the Extra Mile with Disaster-Aware Network Augmentation

Jorik Oostenbrink
Delft University of Technology
J.Oostenbrink@tudelft.nl

Fernando Kuipers
Delft University of Technology
F.A.Kuipers@tudelft.nl

***Abstract*—Network outages have significant economic and societal costs. While network operators have become adept at managing smaller failures, this is not the case for larger, regional failures such as natural disasters. Although it is not possible, and certainly not economic, to prevent all potential disaster damage and impact, we can reduce their impact by adding cost-efficient, geographically redundant, cable connections to the network.**

In this paper, we provide algorithms for finding cost-efficient, disaster-aware cable routes based on empirical hazard data. In contrast to previous work, our approach finds disaster-aware routes by considering the impact of a large set of input disasters on the network as a whole, as well as on the individual cable. For this, we propose the Disaster-Aware Network Augmentation Problem of finding a new cable connection that minimizes a function of disaster impact and cable cost. We prove that this problem is NP-hard and give an exact algorithm, as well as a heuristic, for solving it. Our algorithms are applicable to both planar and geographical coordinates. Using actual seismic hazard data, we demonstrate that by applying our algorithms, network operators can cost-efficiently raise the resilience of their network and future cable connections.

I. Introduction

Communication networks are one of our key critical infrastructures. In fact, as identified by U.S. Presidential Policy Directive 21 [1], communication networks provide "enabling functions" across all other critical infrastructure sectors. Large network outages can have enormous economic and societal impact. Yet, as past events have shown [2], much of our communication infrastructure is still very vulnerable to larger, regional failures such as disasters.

In 2006, an earthquake of the coast of Taiwan damaged 8 submarine cable systems, severely disrupting communications in the region [3]. In 2008 and 2009, new cable systems were installed that deliberately avoided this earthquake region. Thus, when a similar event damaged the same 8 cable systems again in 2009, network operators were able to restore service much quicker [4]. Numerical simulations suggest disaster-aware submarine cable deployments could potentially save society billions of dollars [5].

By installing a new cable connection, a network operator can introduce *geographic redundancy*. In case of a disaster, connections can be routed through the new cable instead of through the disaster area. As a simple example, consider the new cable connection depicted in Fig. 1. By avoiding D1, the new link ensures that nodes 2 and 3 remain connected. Note that avoiding D1 forces the new cable to either go through disaster D2 or make a large detour. Designing disaster-resilient topologies requires operators to make these kind of compromises for hundreds of disaster regions, taking into account cable laying costs, disaster probabilities, the impact of disasters on the network as a whole, as well as the impact of a disaster on the new cable itself.

When designing a real network, only considering two potential disasters is insufficient, and decisions need to be made based on a whole class of potential future disasters. Taking into account all possible combinations of failures and disaster regions would simply be too time-consuming to do manually. Thus, to create truly disaster-resilient network topologies, we need an automated system that can suggest potential cable routes based on actual hazard data.

Although there is a large body of work on finding disaster-resilient cable connections, none of the previous work considers, simultaneously, the impact of a large class of disasters on the cable route itself, as well as on network connectivity as a whole. To fill this gap, we propose a set of algorithms for finding cost-efficient, disaster-aware cable connections based on a large set of representative disasters. The main idea behind our algorithms is to separate the decision of which disaster regions to avoid from the design of the route itself. This allows us to develop exact and heuristic algorithms that search through the problem space and are able to incorporate any pathfinding algorithm for computing the actual routes

Of course, the final decision on the design of a network or cable must be made by the stakeholders, and not by an automated system. By varying an input parameter, our algorithms can quickly generate multiple routes that are Pareto-optimal in cable cost and expected disaster impact. In addition, because our algorithms assign a specific cost to each disaster, and specifically select a set of regions to avoid, they can provide detailed information on *why* a proposed cable connection takes a certain route. Armed with this data, network operators, governments, and other stakeholders can make an informed, disaster-aware decision on any new cable connection.

Our main contributions are as follows:

- We define the Disaster-Aware Network Augmentation Problem of finding a new cable connection that minimizes a cost-function of expected disaster impact and cable cost (Sec. II). By varying a parameter, α, in the objective function we make it possible for network operators to find

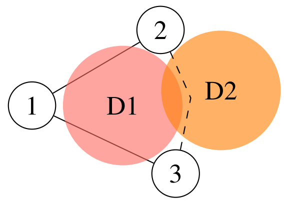


Fig. 1. Example disaster-aware network augmentation. Node 3 is connected to node 2 through node 1. However, if disaster D1 were to occur, this connection would fail. To increase the resilience of this network, we can add a cable connecting nodes 3 and 2 (dotted line), avoiding disaster D1.

various different Pareto-optimal connections for cable cost and expected impact.

- Since the Disaster-Aware Network Augmentation Problem is NP-Hard (Sec. III), we propose both an exact branch & bound algorithm (Sec. IV), as well as a heuristic (Sec. V). Our algorithms are applicable to both the plane, as well as on geographical coordinates.
- We demonstrate our approach by augmenting a real network topology based on actual seismic hazard data (Sec. VI). Given a representative disaster set of 100,000 disasters, our algorithms are able to compute cost-efficient network augmentations within 3 minutes.

II. Problem Statement

A. Network and Disaster Model

We model the network as a directed multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \psi)$, with nodes $v \in \mathcal{V}$ connected by links $e \in \mathcal{E}$, where $\psi : \mathcal{E} \to \mathcal{V} \times \mathcal{V}$ and $e \in \mathcal{E}$ connects v_1 to v_2 iff $\psi(e) = (v_1, v_2)$. Thus, we permit the same pair of nodes to be connected by multiple links.

We model the physical structure and location of network nodes as points in $\mathbb{R}^2$ and denote these points as $p(v)$ for all $v \in \mathcal{V}$. Any link $e \in \mathcal{E}$ between v_1 and v_2 is modelled as a finite sequence of line segments or geodesics connecting v_1 to v_2, $seg(e) = (s_1, s_2), (s_2, s_3), \ldots, (s_{l-1}, s_l)$ where $s_1, \ldots, s_l \in \mathbb{R}^2$, $s_1 = p(v_1)$, and $s_l = p(v_2)$.

To determine how to optimally augment this network, we assume we are given a finite set of representative disasters[1] to protect against, $\mathcal{D}$: We model each potential disaster $d \in \mathcal{D}$ as a disaster region $A(d)$ in the plane or on the globe, with associated probability $P(d)$, and assume exactly one of these disasters occurs (i.e. $\sum\limits_{d \in \mathcal{D}} P(d) = 1$) and destroys all network components intersecting its disaster region. We denote this random disaster by D.

Representative sets of disasters are similar to the stochastic event sets used in catastrophe modelling [7] and should be easily obtainable by network operators. In contrast to many approaches that make use of stochastic event sets, we do not assign structural failure probabilities to network components, but instead assume a more pessimistic outcome where every component inside an affected region fails. For network augmentation, such a worst-case perspective should result in more resilient and less overfitted cable connections among the many uncertainties involved in disaster modelling.

If a node lies in a disaster region, all its incident links also intersect this region. Thus, we do not need to explicitly consider node failures, as the failure of all incident links would disconnect nodes from the network as well. We define the *failure state*, $S(d) \subseteq \mathcal{E}$, of a disaster d to be the set of links intersecting the disaster region $A(d)$, where we say a link $e \in \mathcal{E}$ intersects $A(d)$ if and only if one or more of its line segments, $seg(e)$, intersects $A(d)$.

Before augmenting the network, we first need an impact measure over these failure states to optimize towards. For this purpose, we construct the set $\mathcal{E}^+_\mathcal{G}(d) \subseteq \mathcal{V} \times \mathcal{V}$ of all node pairs that are still directly connected by a functioning link:

$$\mathcal{E}^+_\mathcal{G}(d) := \psi[\mathcal{E} \setminus S(d)] \tag{1}$$

We allow any function $M : \mathcal{P}(\mathcal{V} \times \mathcal{V}) \to \mathbb{R}$ over these sets of node pairs as an impact measure, as long as

$$\forall B \subseteq C \subseteq \mathcal{V} \times \mathcal{V}, M(B) \geq M(C) \tag{2}$$

B. Cable Costs

When suggesting the addition of new links to the network, it is imperative to take the costs of installing these links into account. A simple measure of this cost is cable length. However, costs can vary greatly depending on the specific path of the cable, e.g., if it crosses less accessible areas. To take these factors into account, we divide a rectangular area encompassing the network into a grid of $w \times h$ cells and assume we are given the costs of laying a cable from the center of each cell to the centers of all 8 of its neighbors.

We formulate the route of a new link e from node v_1 to v_2 as a sequence of grid cells, $r(e) = c_{x_1,y_1}, \ldots c_{x_l,y_l}$, where any successive cell c_{x_i,y_i} is a neighbor of the previous cell $c_{x_{i-1},y_{i-1}}$, $v_1 \in c_{x_1,y_1}$, and $v_2 \in c_{x_l,y_l}$. The cost of this route is

$$\mathcal{C}(r(e)) = \sum_{i=1}^{l-1} \mathcal{C}(c_{x_i,y_i}, c_{x_{i+1},y_{i+1}}), \tag{3}$$

where $\mathcal{C}(c_{x_i,y_i}, c_{x_{i+1},y_{i+1}})$ is the cost of laying a cable between cells c_{x_i,y_i} and $c_{x_{i+1},y_{i+1}}$. The exact path of the fiber, $seg(e)$, can now be constructed by connecting the centers of the grid cells in $r(e)$:

$$seg(e) = (p(v_1), ctr(c_{x_2,y_2})), \ldots, (ctr(c_{x_{l-1},y_{l-1}}), p(v_2)), \tag{4}$$

where $ctr(c_{x_i,y_i})$ is the center of cell c_{x_i,y_i}.

Note that the existing links $\mathcal{E}$ of $\mathcal{G}$ do not need to adhere to this grid system, and we solely use the grid and $\mathcal{C}$ as a means of computing the path and cable cost of *new* links. Furthermore, our algorithms are also applicable to any other system for computing cable costs, as long as it allows us to compute a shortest path avoiding a given set of disaster regions.

[1] We base our disaster model on our earlier work on determining network resiliency [6].

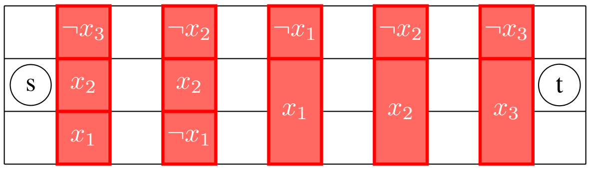


Fig. 2. Example of a reduction from a 3-SAT instance $((x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_2))$ to a Disaster-Aware Network Augmentation Problem instance.

C. Disaster-Aware Network Augmentation Problem

Any augmentation can now be defined by three properties: (1) the source node, v_1; (2) the destination node, v_2; and (3) the route of grid cells connecting these two nodes, r. Given such a *link triple*, we define a network augmentation as follows:

Definition 1 (Network Augmentation): Given a link triple (v_1, v_2, r), where $v_1, v_2 \in \mathcal{V}$ and r is a valid route of cells connecting these nodes,

$$(\mathcal{V}, \mathcal{E}, \psi) + (v_1, v_2, r) = (\mathcal{V}, \mathcal{E} \cup \{e\}, \psi') \tag{5}$$

where

$$\psi'(e') = \begin{cases} (v_1, v_2) & \text{if } e' = e \\ \psi(e') & \text{otherwise} \end{cases} \tag{6}$$

and $seg(e)$ is given by Eq. 4.

Any impact measure M for $\mathcal{G}$ is also applicable to $\mathcal{G} + (v_1, v_2, r)$, giving us a straightforward way of computing the benefit of augmenting the network with any link triple.

Definition 2 (Disaster-Aware Network Augmentation Problem): Given a directed multigraph $\mathcal{G}$, node locations p, link segments seg, cable costs $\mathcal{C}$, metric M, and $\alpha > 0$, find a link triple (v_1, v_2, r) that minimizes

$$\begin{aligned} &\text{cost}(v_1, v_2, r) := \\ &\alpha E[M(\mathcal{E}^+_{\mathcal{G}+(v_1,v_2,r)}(D))] + \mathcal{C}(r) \end{aligned} \tag{7}$$

Remark 1: By varying α, we can find different Pareto-optimal link triples for expected impact and cable cost. If possible, one should choose α such that $\alpha E[M(\mathcal{E}^+_{\mathcal{G}}(D))]$ roughly represents the expected future cost of the class of disasters taken into consideration.

This problem can be divided into two sub-problems:

1) Given two nodes $v_1, v_2 \in \mathcal{V}$, find a route, r, that minimizes cost(v_1, v_2, r);
2) Find the optimal source and destination nodes $v_1, v_2 \in \mathcal{V}$.

III. NP-HARDNESS

Theorem 1: The Disaster-Aware Network Augmentation Problem is NP-hard, even if we restrict ourselves to a single node pair.

Proof: We will provide a polynomial-time reduction from the NP-complete 3-SAT problem [8] to the decision variant of the Disaster-Aware Network Augmentation Problem.

Suppose we are given a Boolean formula f in conjunctive normal form, where each clause contains exactly three literals:

$$f = C_1 \wedge C_2 \wedge \cdots \wedge C_k \tag{8}$$

with

$$C_1 = l_{1,1} \vee l_{1,2} \vee l_{1,3}, C_2 = l_{2,1} \vee l_{2,2} \vee l_{2,3}, \ldots \tag{9}$$

The 3-SAT problem is to determine if this formula is satisfiable. Let V be the set of all variables in the formula. To reduce f to an instance of the Disaster-Aware Network Augmentation Problem, we first create a grid of $(1+2k+2|V|) \times 3$ cells and assign the same cost of $\frac{1}{3(1+2k+2|V|)}$ to each of the possible connections from a cell to its neighbors. We then create a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \psi\}$ of two nodes ($\mathcal{V} = \{s, t\}$) and no links ($\mathcal{E} = \emptyset$) and place s in the middle cell of the left-most column, and t in the middle row of the right-most column.

We form a disaster set $\mathcal{D}$ by combining all possible literals, i.e., $\mathcal{D} = \bigcup_{x \in V} \{x, \neg x\}$. Fig. 2 demonstrates how we construct the disaster regions:

1) We create a column of 3 disaster sub-regions for each clause C_i of f that fills up exactly one column of cells: the disaster region of $l_{i,1}$ fills up the first cell of this column, the region of $l_{i,2}$ the second cell, and the region of $l_{i,3}$ the third cell.
2) We also create a column of 2 disaster sub-regions for each variable $x \in V$, where x fills up the first two cells of the column, and $\neg x$ the third cell.
3) We then place all these columns directly to the right of the column containing s, and put a spacing of 1 cell between each successive column.

We set $P(d) = \frac{1}{\mathcal{D}}$ for all $d \in \mathcal{D}$ and choose $\alpha = |\mathcal{D}|$. Finally, we assign an impact of 0 if s is connected to t, and an impact of 1 otherwise:

$$\begin{aligned} M(\{(s,t),(t,s)\}) = 0,\, M(\{(s,t)\}) = 0, \\ M(\emptyset) = 1,\, M(\{(t,s)\}) = 1 \end{aligned} \tag{10}$$

Note that this means that a solution that connects t to s instead of s to t will always have a cost of at least $|\mathcal{D}|$.

Now, suppose there is a route r from s to t with cost

$$\alpha E[M(\mathcal{E}^+_{\mathcal{G}+(s,t,r)}(D))] + \mathcal{C}(r) < \frac{1}{2}\mathcal{D} + 1 \tag{11}$$

This would mean that r intersects at most $\frac{1}{2}\mathcal{D} = |V|$ disasters, as otherwise $\alpha E[M(E^+_{G+(s,t,r)}(D))] \geq \frac{1}{2}\mathcal{D} + 1$. To reach t from s, the route must intersect x or $\neg x$ for each variable $x \in V$. Because the number of intersected disasters is at most $|V|$, this means that for all $x \in V$, the route can not intersect both x and $\neg x$. In addition, the route must also intersect at least one disaster region of the literals of each clause. Thus, the selection of literals intersected by r form a satisfying assignment for f.

Vice versa, suppose f is satisfiable. In other words, there is an assignment of TRUE and FALSE to each variable $v \in V$ such that each of the clauses of f (and thus f itself) is satisfied. We will construct a route r from s to v such that

$$\alpha E[M(\mathcal{E}^+_{\mathcal{G}+(s,t,r)}(D))] + \mathcal{C}(r) < \frac{1}{2}\mathcal{D} + 1 \tag{12}$$

First, our route will need to cross all the columns of clauses. Because each clause is satisfied by the assignment, at least one literal of each clause must evaluate to TRUE. Thus, we construct the route in such a way that we only intersect this literal of the clause. Note that due to the spacing between successive clauses, this is always possible. Next, the route will need to cross the columns of variables as well. Here, as with the clauses, we intersect the literal that evaluates to TRUE. This way, we can construct a route r from s to t that only intersects the $|V|$ literals that evaluate to TRUE. Furthermore, because we pass through each cell at most once,

$$\begin{aligned}\alpha E[M(\mathcal{E}^+_{\mathcal{G}+(s,t,r)}(D))] + \mathcal{C}(r) &< \\ \frac{1}{2}\mathcal{D} + \frac{3(1+2k+2|V|)}{3(1+2k+2|V|)} &= \\ \frac{1}{2}\mathcal{D} + 1\end{aligned} \tag{13}$$

We thus have a polynomial-time reduction from the 3-SAT problem to the decision variant of the Disaster-Aware Network Augmentation Problem, and can conclude that the Disaster-Aware Network Augmentation Problem is NP-hard. ∎

Remark 2: This proof also applies to sub-problem 1 by itself. Thus, determining the optimal route between a given source and destination node is already NP-hard.

IV. Branch and Bound

In this section we describe an exact algorithm for sub-problem 1 based on the branch and bound paradigm. Suppose we are given two nodes $v_1, v_2 \in \mathcal{V}$, our goal is to find a route r of cells from v_1 to v_2 that minimizes cost(v_1, v_2, r). This might seem similar to the shortest path problem. However, the difficulty lies in that, unlike for the shortest path problem, where the sub-path of a shortest path is itself a shortest path, in our case a sub-route of a minimum-cost route is not necessarily a minimum-cost route itself.

The key insight behind our approach is that if we decide on a specific set of disasters to avoid, $R \subseteq \mathcal{D}$, the problem of finding a route with minimum cable cost between v_1 and v_2 that does not intersect any disaster in R *is* a shortest path problem. In the rest of this paper, we call these subsets of representative disasters *restrictions*, and the minimum-cable-cost route avoiding a restriction a *restricted shortest route*. As we show in this section, we can quickly compute the cost of any given route between v_1 and v_2 as a sum of pre-computed disaster *penalties* and cable costs. Thus, to determine the cost associated to a given restriction R, we simply find a restricted shortest path for R, and compute the cost of this path. This allows our algorithm to search for the optimal restriction R instead of the optimal route, greatly simplifying the problem.

We first introduce an indicator value $I(d, r)$:

$$I(d,r) = \begin{cases} 1 & \text{if } seg(r) \text{ intersects } A(d) \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

Regardless of our choice of route r, the impact of any disaster $d \in \mathcal{D}$ is $M(\mathcal{E}^+_{\mathcal{G}}(d))$ if r intersects it, and $M(\mathcal{E}^+_{\mathcal{G}}(d) \cup \{(v_1, v_2)\})$ otherwise. If we take the difference of these values, we get a measure of the benefit of adding a connection from v_1 to v_2 in case of a disaster d:

input: $\mathcal{G}, v_1, v_2, M, \mathcal{C}, \mathcal{D}, \alpha$
output: optimal route from v_1 to v_2, r
compute $M(d)^+ \ \forall d \in \mathcal{D}$ ▷ Equation 15
$W \leftarrow \alpha \sum_{d\in\mathcal{D}} P(d)I(d,r)M(d)^+ + \mathcal{C}(r)$ ▷ Equation 17
$W(\emptyset) \leftarrow \infty$
$\mathcal{D} \leftarrow \{d \in \mathcal{D} | M(d)^+ > 0, p(v_1) \notin A(d), p(v_2) \notin A(d)\}$
$r \leftarrow$ Search$(\mathcal{D}, \emptyset, \emptyset, \emptyset)$

function Search$(\mathcal{D}, R, \mathcal{D}^-, r_*)$
  cutoff $\leftarrow W(r_*) - \alpha \sum_{d\in\mathcal{D}^-} P(d)M(d)^+$
  try to find a restricted shortest route $sp(R)$ from v_1 to v_2 with a cutoff cost of cutoff
  if $sp(R)$ not found **then**
    $r \leftarrow r_*$
  else
    if $W(sp(R)) < W(r_*)$ **then**
      $r \leftarrow sp(R)$
    else
      $r \leftarrow r_*$
    end if
    for all $d \in \mathcal{D} \setminus (\mathcal{D}^- \cup R)$ intersected by $sp(R)$ **do**
      $r \leftarrow$ Search$(\mathcal{D}, R \cup \{d\}, \mathcal{D}^-, r)$
      $\mathcal{D}^- \leftarrow \mathcal{D}^- \cup \{d\}$
    end for
  end if
  return r
end function

Fig. 3. Pseudocode for the exact depth-first branch and bound algorithm for finding the minimum-cost route from node v_1 to node v_2.

$$M(d)^+ := M(\mathcal{E}^+_{\mathcal{G}}(d)) - M(\mathcal{E}^+_{\mathcal{G}}(d) \cup \{(v_1, v_2)\}) \tag{15}$$

Now, for any route r,

$$\begin{aligned} E[M(\mathcal{E}^+_{\mathcal{G}+(v_1,v_2,r)}(D))] &= \\ \sum_{d\in\mathcal{D}} P(d)(M(\mathcal{E}^+_{\mathcal{G}}(d) \cup \{(v_1,v_2)\}) + I(d,r)M(d)^+) &= \\ E[M(\mathcal{E}^+_{\mathcal{G}}(D) \cup \{(v_1,v_2)\})] + \sum_{d\in\mathcal{D}} P(d)I(d,r)M(d)^+ \end{aligned} \tag{16}$$

If we subtract any constant from our objective function the resulting optimization problem is equivalent to our old one. Thus, we subtract $\alpha E[M(\mathcal{E}^+_G(D) \cup \{(v_1, v_2)\})]$ to obtain the new objective function

$$W(r) := \alpha \sum_{d\in\mathcal{D}} P(d)I(d,r)M(d)^+ + \mathcal{C}(r) \tag{17}$$

As $P(d)M(d)^+$ does not depend on the route of the link itself and can be pre-calculated, $W(r)$ can be seen as the sum of the cable cost, $\mathcal{C}(r)$, and a pre-computed penalty, $\alpha P(d)M(d)^+$, for every intersected disaster region $A(d)$.

We denote a restricted shortest route by $sp(R)$, where $R \subseteq D$ is the set of disasters this route should avoid. Our exact algorithm is a depth-first search for the optimal restriction R. The algorithm starts at $R = \emptyset$, and tries to find the optimal restriction and route from there. The algorithm is provided in pseudocode in Fig. 3. For readability, the algorithm is formulated as a recursive function call. However, our implementation uses an iterative approach.

A. Branching

The number of possible restrictions ($2^{|\mathcal{D}|}$) grows exponentially in $|\mathcal{D}|$. Thus, to keep computation times manageable, reducing the number of considered disasters is essential. Fortunately, it is likely that for many of the representative disasters $d \in \mathcal{D}$, connecting v_1 to v_2 will not bring any benefit and $M(d)^+ = 0$. In addition, disasters that intersect $p(v_1)$ or $p(v_2)$ might have a positive benefit $M(d)^+) > 0$, but can not be avoided. These two sets of disasters do not need to be considered by our algorithm and are excluded.

After computing $sp(R)$, we can limit the number of potential branches even more. If $sp(R)$ does not intersect a disaster $d \in \mathcal{D}$, adding d to the restriction will not change the restricted shortest route. Thus, for any restriction R, we only consider extending R with disasters intersected by $sp(R)$, where we say a route $sp(R)$ intersects a disaster $d \in \mathcal{D}$ if and only if any of the line segments $seg(sp(R))$ intersects $A(d)$.

We choose to branch on individual disasters: If $sp(R)$ intersects k disasters with positive benefit, $d_1, \ldots, d_k \in \mathcal{D}$ from large to small benefit, we create k branches, $R \cup \{d_1\}, \ldots, R \cup \{d_k\}$. If the optimal solution avoids $R \cup \{d_i\}$, our approach will find this solution in branch $R \cup \{d_i\}$. Thus, after having visited branch $R \cup \{d_i\}$, we remove d_i from consideration in branches $R \cup \{d_{i+1}\}, \ldots, R \cup \{d_k\}$. This both prevents the algorithm from visiting the same restriction twice and further limits the number of considered restrictions.

B. Bounding

Throughout our algorithm, we keep track of the best route encountered so far, r_*, and its objective value, $W(r_*)$. Once we know that all further restrictions on R would lead to a worse solution than r_*, we can stop exploring branch R. For every restriction R, $\mathcal{C}(sp(R))$ is a lower bound on the objective value W for any further restrictions. However, by taking into account the disasters we explicitly removed from consideration in our branching approach, we can improve upon this bound.

Let $\mathcal{D}^-$ be the set of all disasters removed from consideration. If the optimal route avoids a disaster $d \in \mathcal{D}^-$, it has already been found previously and r_* is the optimal route. Thus, we can stop exploring a branch when

$$\mathcal{C}(sp(R)) \geq W(r_*) - \alpha \sum_{d \in \mathcal{D}^-} P(d) M(d)^+ \tag{18}$$

C. Shortest Route Computations

Our algorithm needs to compute a new route $sp(R)$ for every considered restriction R. To speed up computations, we pre-compute the minimum distance between each cell and v_2 using Dijkstra's Algorithm, and then use A^* to compute restricted shortest routes from v_1 to v_2. Furthermore, we immediately stop computing a route once the cost to reach the current cell and the minimum distance between the current cell and v_2 exceeds the cutoff value given in Eq. 18.

While computing a restricted shortest route, we constantly need to check if a line segment between two adjacent cells does not intersect any disaster $d \in R$. To reduce the computation time spent on these checks, we use caches to keep track of which line segments intersect which disasters.

D. Global Optimization

To find the optimal link triplet (v_1, v_2, r), we apply our branch & bound algorithm to every pair of nodes in the network. In this context, we make some small adjustments to the algorithm to further reduce computation times. First, we propose pre-computing the failure state $S(d)$ and impact $M(\mathcal{E}_G^+(d))$ of all disasters $d \in \mathcal{D}$. This allows us to compute penalties for each failure state instead of for each disaster. As the number of failure states tends to be much smaller than the number of disasters, this significantly speeds up the pre-computation phase of each node pair.

Second, we keep track of the minimum-cost route r_* and the corresponding upper bound across all node pairs and pass this global upper bound to the branch and bound algorithm. This requires us to transform the global upper bound (on cost) to a local upper bound (on W). Let u_{global} be a global upper bound, we can transform u_{global} to a local upper bound by

$$u_{\text{local}} = u_{\text{global}} - \alpha(E[M(\mathcal{E}_G^+(D) \cup \{(v_1, v_2)\})]) \tag{19}$$

We apply this bound as an initial upper bound for our search function, as well as a limit on the cells we pre-compute A^* heuristics for. Note that the transformed local upper bound might be negative. In this case no possible route from v_1 to v_2 could improve upon our current global best route, and we can skip node pair (v_1, v_2).

The global upper bound is essential in reducing computation times. However, depending on the order we traverse node pairs in, it might take a long time before we have obtained a low upper bound. Thus, to obtain a reasonable upper bound a priori, we initially compute the shortest route for each node pair. We then select the route with minimum cost and use it as the initial value for r_*.

V. Heuristic

Although our branch & bound algorithm is fast enough for many practical use cases, its runtime is still exponential in the number of representative disasters, $\mathcal{D}$. Since the Disaster-Aware Network Augmentation Problem is NP-hard, we propose a heuristic for sub-problem 1 that can find near-optimal solutions for larger disaster sets in a fraction of the runtime of the branch & bound approach.

In the previous section, we have reduced the problem of finding the optimal route between two nodes to the problem of determining which regions to avoid. We use this same concept to create a heuristic and apply simulated annealing to find an approximate solution for the following optimization problem:

$$\min_{R \subseteq \mathcal{D}} W(sp(R)) = \alpha \sum_{d \in \mathcal{D}} P(d) I(d, sp(R)) M(d)^+ + \mathcal{C}(sp(R)), \tag{20}$$

where $sp(R)$ is a restricted shortest route from v_1 to v_2 avoiding R. To find the global link triple, we use the same approach as described in Sec. IV-D, and simply replace the branch & bound approach with our heuristic.

A. Simulated Annealing

A simulated annealing algorithm searches for an optimal solution by randomly selecting and evaluating neighboring solutions. If the neighbor has a lower cost, the algorithm directly switches to this solution. If not, it does so with a probability depending on the current *temperature* as well as the difference in costs [9]. By starting with a high temperature and slowly decreasing it over time, simulated annealing initially searches throughout the problem space and then gradually "locks in" to a local minimum.

An important consideration when applying simulated annealing to any problem is the selection of neighbors. We take the same approach as described in Sec. IV-A (but do not exclude $\mathcal{D}^-$). The neighbors of a restriction R are

$$\{R \cup \{d\} | d \in \mathcal{D} \setminus R \wedge sp(R) \text{ intersects } A(d)\} \tag{21}$$

and

$$\{R \setminus \{d\} | d \in R\} \tag{22}$$

For our experiments, we have made the following implementation choices for our simulated annealing algorithm.

- **Initial solution**: As in the branch & bound approach, we start at the empty restriction $R \leftarrow \emptyset$, and compute $sp(\emptyset)$.
- **Transition probability**: Let δ be the increase in cost of the neighboring solution. We transition to this solution with probability 1 if $\delta < 0$ and with probability $e^{\frac{-\delta}{T}}$ otherwise, where T is the current temperature.
- **Temperature** T: We set the initial temperature to $\frac{-\alpha \sum_{d \in \mathcal{D}} P(d) I(d, sp(\emptyset)) M(d)^+}{\ln 0.25}$.
- **Temperature Reduction**: Every 10 "repetitions" of considering a neighboring solution we reduce the temperature by $T \leftarrow 0.9T$.
- **Freezing Point** and **Stopping Condition**: If none of the 10 repetitions resulted in a move to a different solution, we set the temperature T to 0 (thus switching to hill climbing). If none of the repetitions resulted in a move, and the temperature is already 0, we submit the restricted shortest path with the lowest cost we have encountered up till this point as the solution.

We make the following modification to the standard simulated annealing implementation: If the cable cost of a restricted shortest path $sp(R)$ exceeds $W(r^*)$, where r^* is the best route we have found up till this point, we outright reject R as a potential solution and set $\delta = \infty$. As discussed in Sec. IV-B, this prevents the algorithm from moving to restrictions that are too restrictive to improve upon r^*, while not blocking off the optimal solution from the search space. More importantly, it allows us to save computation time by setting an upper bound on the maximum cable cost and cutting of the pathfinding algorithm if this upper bound is exceeded.

VI. Experiments

A. Setup

We demonstrate our methods on the undirected Italian subtopology of Interoute, a 25-node, 35-link network traced and made publicly available by the authors of [10]. All experiments were conducted on commodity hardware: a AMD Ryzen 7 3700X 3.6 GHz processor with 32 GB of available RAM.

We augment Interoute with respect to the publicly available earthquake dataset from [10]. This dataset was purpose-built by seismologists for analyzing the resiliency of communication networks. It essentially consists of a set of 1,196,037 disk disasters, which together represent all possible earthquakes that can strike Italy.

To demonstrate the applicability of our algorithms to geographic coordinates, we do not transform coordinates to the plane. Instead, we construct a grid of around 0.05 x 0.05 degree covering the longitude-latitude coordinates of all nodes padded by 0.05 degrees on all sides (resulting in a 232 by 194 grid). To compute cable costs and determine which network components are affected by a disaster, we use the great circle distance for a sphere with radius 6,371 km.

We filter out all disasters that do not damage any network components and reweight the remaining probabilities to make them sum to 1, as it is not necessary to protect the networks against these events and they would be filtered out at the penalty computation stage. This leaves us with 454,433 out of 1,196,037 disasters.

Our impact metric, M, is the number of disconnected node pairs, divided by the total amount of node pairs. Thus, $M = 0$ if all node pairs are connected, and $M = 1$ if all node pairs are disconnected. The expected impact of earthquakes on Interoute is approximately 0.0141. This might seem small, but the disaster set is simply so extensive that it also contains many disasters that barely affect the network. In fact, the total disaster rate is 1.6006 per year. This means that on average, more than 2% of connections fail due to earthquakes *per year*. In practice, there will be some events where a large part of the network is disconnected at once, and in most years none or nearly none of the connections are affected.

B. Connecting Node Pairs

We first consider sub-problem 1: finding the optimal route between two given nodes $v_1, v_2 \in \mathcal{V}$. To decrease computation times, we sample 10 sets of 50,000 disasters and use these as representative disaster sets. We randomly select 20 node pairs, and then use our simulated annealing heuristic to compute a route between these nodes for each sampled disaster set:

- on a cost grid where cost is equal to the distance in km (**uniform cost case**);

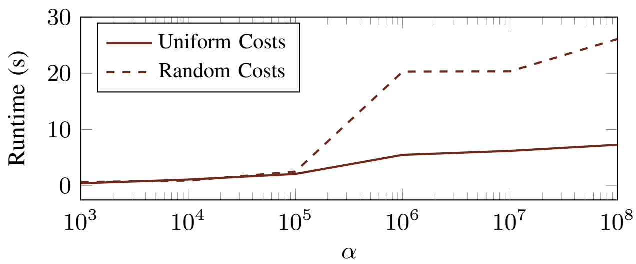


Fig. 4. Mean computation time of a route between two random nodes (using simulated annealing).

- and on grids where the cost of traversing a cell is a random uniformly distributed value from $[0, 2)$ times the traversed distance (in km) (**random cost case**).

We repeat this experiment (for the same disaster sets and node pairs) for different α and evaluate all routes on the *full* disaster set. We do not consider the time required to compute the initial failure states and impacts[2], and compute these separately before running any experiments.

The lowest α we consider is $\alpha = 10^3$. For such an α, a network operator would only be willing to install up to around 14 km of additional cable to *completely* mitigate the impact of all potential disasters. The maximum α we consider is 10^8. Here, an operator would be willing to install around $100,000$ as much cable to achieve the same goal.

As can be seen in Fig. 4, computation times increase with α, but stay within 30 seconds even for an α as high as 10^8. At lower α, our approach only needs to test a few, small restrictions, but for higher α the number and size of considered restrictions, and thus the runtime, rises.

In Fig. 5, we compare the mean reduction in expected disaster impact due to the routes computed by our simulated annealing heuristic to that of the shortest route. As we are connecting *random* node pairs, we can not expect a major decrease in expected impact. Nevertheless, the improvement of a disaster-aware route over the shortest route is quite impressive. For random costs, the mean reduction in disaster impact due to adding a new route to the network is improved by more than 50% just by adding a small detour to the cable route. This is on top of any planned benefits of the cable in terms of, e.g., capacity. For lower α, the mean cost of connecting two random nodes is negative, and it is not worth it to deviate from the shortest path (see Fig. 6). However, as α increases, the simulated annealing solution starts outperforming the shortest path. Note that in practice, network operators will have enough time to compute routes based on the full representative disaster set, which would result in an even larger improvement over the shortest route.

C. Global Solution

Next, we consider the Disaster-Aware Network Augmentation Problem itself and try to find optimal link triples across *all* node pairs. We run 10 experiments on the full disaster set: 5 for the uniform cost case, and 5 for the random cost case. In each experiment, we add new cable connections to the network in a greedy fashion (i.e., we iteratively compute and add the next solution for the Disaster-Aware Network Augmentation Problem to the network) until doing so would not be worth it anymore (cost $\geq \alpha E[M(\mathcal{E}^+_\mathcal{G}(D))]$. As before, we exclude the time required to compute initial failures.

Fig. 7 shows the mean computation times of the first link triple. Again, computation times increase with α. Using simulated annealing, we manage to find a solution within 20 minutes, even for $\alpha = 10^8$. This is fast enough for network operators to vary α and compare different potential routes.

By repeating the same experiment for different α, we can get an idea of how much we can reduce the expected impact given a certain cable budget. Fig. 8 shows the mean normalized reduction in impact against the mean total cable cost of greedily augmenting Interoute. By adding new cable connections to the network, we significantly reduce the expected impact of disasters. The biggest reduction in impact comes from some cheaper, very effective cable connections. If we want to reduce the expected impact even further, we need to invest progressively more for smaller reductions in impact.

D. Resilience Against New Disasters

Our algorithms extend network topologies based on a set of representative disasters, $\mathcal{D}$. This raises the question of how our new routes perform on disasters that are *not* included in this input set $\mathcal{D}$. Does our approach actually increase the resilience of the network to a whole class of disasters, or does it overfit routes to the set of input disasters?

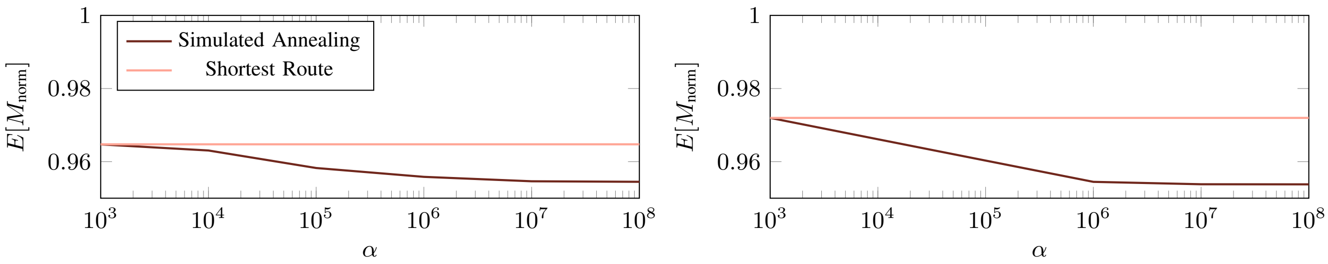
To answer this question, we take an approach that is similar to 10-fold cross-validation: (1) We randomly split our disaster set, $\mathcal{D}$, into 10 groups, or folds, of disasters; (2) For each of these groups, $\mathcal{D}^*$, we greedily augment Interoute by applying our simulated annealing algorithm to $\mathcal{D} \setminus \mathcal{D}^*$ until there is no improvement in cost anymore and (3) compute the expected impact of $\mathcal{D}^*$ on this augmented network (re-weighting probabilities where required). For the purpose of this experiment, we assume uniform cable costs and set $\alpha = 5,000,000$.

If we take the full set of 454,433 disasters into account, the greedy simulated annealing approach reduces the expected impact from around 0.01411 to 0.00916 by adding 3 new cable connections with a total length of around 1235 km. In comparison, the average expected impact over all 10 folds is around 0.00917. This is achieved by adding 3 new connections with an average total length of around 1232 km. We conclude that as long as the set of input disasters is representative of the disasters we want to protect the network against, the routes we compute based on $\mathcal{D}$ also manage to effectively reduce the impact of disasters that were not included in $\mathcal{D}$.

E. Number of Representative Disasters

We have shown that by increasing the resilience of a network to a set of representative disasters, we also increase its resilience to disasters that were not explicitly considered. But

[2]Roughly 310 (26) seconds for the full Interoute disaster set on 1 thread (all cores), and around 35 (3) seconds on the smaller disaster sets.



(a) Uniform Costs

(b) Random Costs

Fig. 5. Mean expected impact after adding a route between two random nodes divided by the expected impact on the initial network topology. The routes were computed based on a sampled set of 50,000 disasters, and evaluated on the full set of 454,433 disasters.

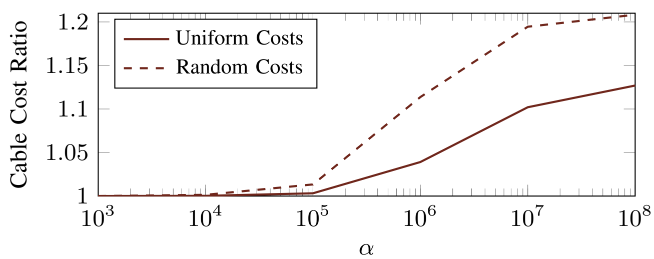


Fig. 6. Mean cable cost of a route between two random nodes, v_1, v_2, computed by simulated annealing divided by the cable cost of the shortest route between v_1 and v_2.

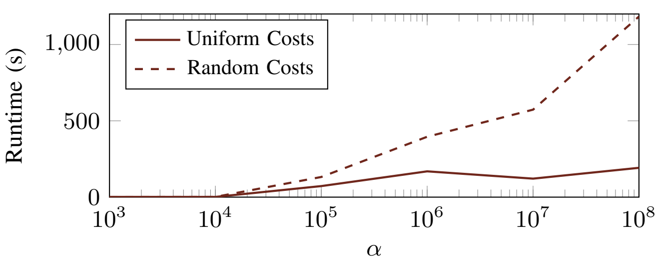


Fig. 7. Mean simulated annealing computation time of a new link triple (Disaster-Aware Network Augmentation Problem).

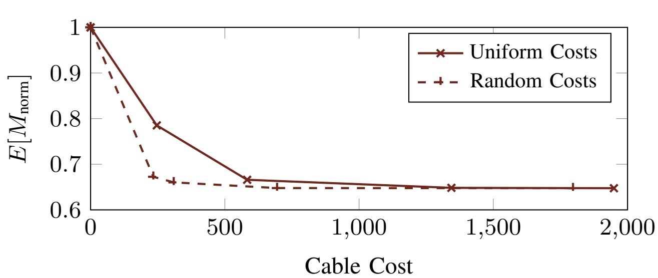


Fig. 8. Mean expected impact after greedily augmenting the network using simulated annealing divided by the expected impact on the initial topology, against the total cable cost.

how many representative disasters should a network operator include in their input set $\mathcal{D}$? In this section, we study the effect of $|\mathcal{D}|$ on the reduction in expected impact, as well as on the computation time of our approaches.

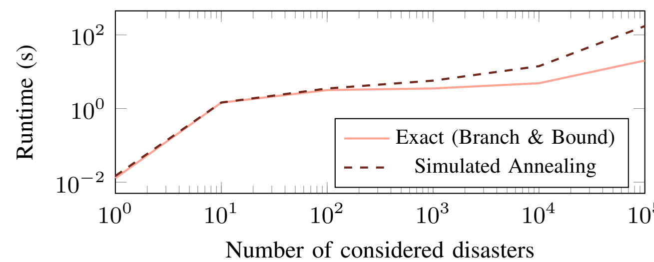


Fig. 9. Mean computation time of the first new link triple for Interoute against the number of considered disasters. $\alpha = 5,000,000$.

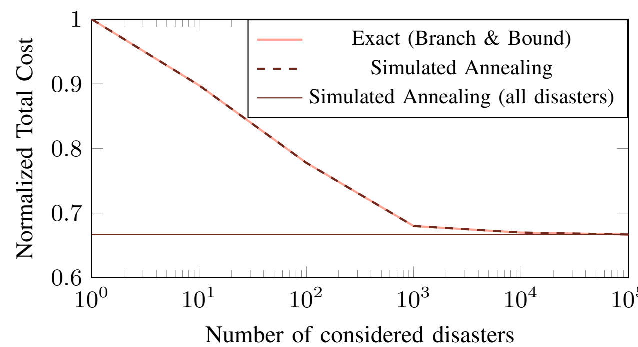


Fig. 10. Mean total cost of a greedily computed set of new Interoute cable routes divided by the cost of not augmenting the network ($(\alpha E[M(\mathcal{E}_{\mathcal{G}}^{+}(D))])$) against the number of considered disasters. $\alpha = 5,000,000$.

We start with the set of 454,433 disasters, $\mathcal{D}$, and will treat this as the set of all potential disasters that can affect the network. Clearly, if we sample disasters from $\mathcal{D}$, the set of sampled disasters is representative of $\mathcal{D}$. Thus, we can create a representative disaster set of any size simply by sampling disasters from our initial disaster set.

To be more precise, to create an input set $\mathcal{D}^*$ of N disasters, we sample N disasters from $\mathcal{D}$ with replacement (where the probability of sampling a disaster $d \in \mathcal{D}$ is $P(d)$) and assign each disaster a probability of $\frac{1}{N}$. We then apply our algorithms on $\mathcal{D}^*$ to greedily extend Interoute with new cable connections until there is no benefit in cost. Finally, we compute the expected impact of the full set $\mathcal{D}$ on this augmented network to check how well we have managed to increase the resilience of the network. We set $\alpha = 5,000,000$ and repeat this process 20 times for each N.

As can be seen in Fig. 9, the runtime of both algorithms increases with the number of considered disasters. Remarkably, at these representative disaster set sizes, the branch & bound algorithm is faster than the simulated annealing algorithm, and both manage to find the initial link triple in less than 3 minutes. In comparison, even if $\alpha = 100,000$, the branch & bound algorithm takes more than an hour to compute a link triple for the 454,433 disasters input set.

Fig. 10 shows the relative improvement in total cost[3] over not augmenting the network against the number of considered disasters. We see that the costs of the link triples found by simulated annealing lie very close to that of the link triples found by the branch & bound algorithm. Importantly, we confirm that we do indeed need approaches that work for larger disaster set sizes (of at least 1,000 disasters), as the total cost drastically decreases as we increase the size of the representative disaster set. However, we also spot some opportunities: by reducing the disaster set size from 454,433 to 100,000 or even 10,000 we significantly reduce computation times, while not sacrificing much of the cost of the final result.

VII. Related Work

For an overview of overall strategies for increasing the survivability of communication networks to disasters, we refer the reader to a recent survey conducted by T. Gomes et al. [11].

Variations of what we call sub-problem 1, finding an optimal cable path while taking into account potential disasters, have been studied extensively [12]–[19]. As these works focus entirely on the path or topology of a single cable connection, they do not take into account the network-wide impact of disasters and solely consider the damage disasters can do to this single cable. Because our algorithms decouple path planning and finding the optimal restriction, approaches such as [14], [15] can be straightforwardly incorporated and used for finding restricted shortest routes. This would extend our approach by allowing network operators to include costs that depend on the length of the cable segment intersecting disaster regions, such as repair rates or even shielding [15].

Building upon the spine concept introduced in [20], L. Garrote et al. gave a heuristic for the obstacle-avoiding Euclidean Steiner tree problem [21]. In contrast to our approach, L. Garrote et al. aim to design a high-availability spine that avoids disaster-prone areas, and do not explicitly consider the impact of disasters on network connectivity metrics.

Cao et al. gave a heuristic for optimizing cable costs of a planar N-node topology under constraints on the disconnection probability of any node in the network [22]. The main difference with our work is that we consider the augmentation of an existing network, while C. Cao et al. considered the design of an entirely new submarine network topology. Furthermore, where their heuristic only considered uniformly distributed disk disasters and simplified cable costs, our disaster and cable cost models are more general.

[3]The expected impact on the augmented network multiplied by α plus the total cable cost of all link triples.

Given a desired topology, pre-computed set of candidate cable routes, and the probabilities of failure of each of these routes, D. L. Msongaleli et al. formulated the optimization of submarine cable deployments under disasters as an integer linear optimization problem [5]. Their simulations suggest that disaster-aware submarine topologies could potentially save society billions of dollars.

P. N. Tran and H. Saito proposed an interesting heuristic for optimizing a weighted set of end-to-end disconnection probabilities under cable length constraints by either recomputing the routes of existing links [23] or augmenting the network by adding new links [24] based on actual seismic hazard data. To compute a set of link triples, they first compute a set of potential candidate routes for each considered link, and then feed these to a dynamic programming algorithm for the global optimization problem. Compared to our approach, their earthquake disaster model is more detailed and takes into account link failure probabilities. This does come at a cost, as their approach does not seem to scale well to larger network sizes or disaster sets. In particular, computing the end-to-end disconnection probabilities in their evaluation metric is a well-known NP-hard problem for even a single disaster [25]. Furthermore, although the authors did limit the paths a cable was allowed to take to streets, their approach is based on a uniform cable-cost scenario.

In contrast to our approach, none of the previous work can be applied to a large set of disaster inputs *and* optimize cable routes for a network-wide impact metric, let alone incorporates detailed cable laying costs.

VIII. Conclusion

In this paper, we have presented the Disaster-Aware Network Augmentation problem of finding a cost-efficient link triple (pair of nodes and a route between these nodes) to increase the resilience of networks to a large set of representative disasters. The solutions to this problem are Pareto-optimal for expected disaster impact and cable cost.

As the Disaster-Aware Network Augmentation Problem is NP-hard, we have given both an exact algorithm, as well as a heuristic. The main idea behind our algorithms is to split the problem of finding a disaster-aware route into the problem of deciding which disasters to avoid (a *restriction*) and finding a *restricted shortest route* avoiding these disasters.

We have demonstrated the effectiveness of our algorithms by computing disaster-aware cable connections for a real topology using actual seismic hazard data. Using our approach, operators "going the extra mile" can increase the disaster-resilience of their network by adding a cost-efficient selection of new cable connections. By making disaster-aware design decisions, instead of planning based on cable costs only, network operators can simultaneously increase the capacity of their network, as well as reduce the impact of future disasters.

References

[1] (2013, February) Presidential policy directive – critical infrastructure security and resilience. The White House. [Online]. Available: obamawhitehouse.archives.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure-security-and-resil

[2] Z. E. Khaled and H. Mcheick, "Case studies of communications systems during harsh environments: A review of approaches, weaknesses, and limitations to improve quality of service," *International Journal of Distributed Sensor Networks*, vol. 15, no. 2, 2019. [Online]. Available: https://doi.org/10.1177/1550147719829960

[3] W. Qiu. (2011) Submarine cables cut after taiwan earthquake in dec 2006. Submarine Cable Networks. [Online]. Available: www.submarinenetworks.com/en/news/cables-cut-after-taiwan-earthquake-2006

[4] ——. (2011) Submarine cables cut by taiwan earthquake and typhoon morakot. Submarine Cable Networks. [Online]. Available: www.submarinenetworks.com/en/news/cables-cut-by-taiwan-earthquake-and-typhoon-morakot

[5] D. L. Msongaleli, F. Dikbiyik, M. Zukerman, and B. Mukherjee, "Disaster-aware submarine fiber-optic cable deployment for mesh networks," *J. Lightwave Technol.*, vol. 34, no. 18, pp. 4293–4303, Sep 2016. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-34-18-4293

[6] J. Oostenbrink and F. Kuipers, "Computing the Impact of Disasters on Networks," *ACM SIGMETRICS Performance Evaluation Review*, vol. 45, no. 2, pp. 107–110, oct 2017.

[7] K. Mitchell-Wallace, M. Jones, J. Hillier, and M. Foote, *Natural catastrophe risk management and modelling: A practitioner's guide*. John Wiley & Sons, 2017, ch. 4, pp. 297–388.

[8] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[9] K. A. Dowsland and J. M. Thompson, *Simulated Annealing*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1623–1655. [Online]. Available: https://doi.org/10.1007/978-3-540-92910-9_49

[10] A. Valentini, B. Vass, J. Oostenbrink, L. Csák, F. Kuipers, B. Pace, D. Hay, and J. Tapolcai, "Network resiliency against earthquakes," in *2019 11th International Workshop on Resilient Networks Design and Modeling (RNDM)*, 2019, pp. 1–7.

[11] T. Gomes, J. Tapolcai, C. Esposito, D. Hutchison, F. Kuipers, J. Rak, A. de Sousa, A. Iossifides, R. Travanca, J. André, L. Jorge, L. Martins, P. O. Ugalde, A. Pašić, D. Pezaros, S. Jouet, S. Secci, and M. Tornatore, "A survey of strategies for communication networks to protect against large-scale natural disasters," in *2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM)*, 2016, pp. 11–22.

[12] C. Cao, Z. Wang, M. Zukerman, J. H. Manton, A. Bensoussan, and Y. Wang, "Optimal cable laying across an earthquake fault line considering elliptical failures," *IEEE Transactions on Reliability*, vol. 65, no. 3, pp. 1536–1550, 2016.

[13] M. Zhao, T. W. S. Chow, P. Tang, Z. Wang, J. Guo, and M. Zukerman, "Route selection for cabling considering cost minimization and earthquake survivability via a semi-supervised probabilistic model," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 502–511, 2017.

[14] Z. Wang, Q. Wang, M. Zukerman, J. Guo, Y. Wang, G. Wang, J. Yang, and B. Moran, "Multiobjective path optimization for critical infrastructure links with consideration to seismic resilience," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 10, pp. 836–855, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/mice.12287

[15] Z. Wang, Q. Wang, B. Moran, and M. Zukerman, "Application of the fast marching method for path planning of long-haul optical fiber cables with shielding," *IEEE Access*, vol. 6, pp. 41 367–41 378, 2018.

[16] Q. Wang, J. Guo, Z. Wang, E. Tahchi, X. Wang, B. Moran, and M. Zukerman, "Cost-effective path planning for submarine cable network extension," *IEEE Access*, vol. 7, pp. 61 883–61 895, 2019.

[17] Z. Wang, Q. Wang, B. Moran, and M. Zukerman, "Terrain constrained path planning for long-haul cables," *Opt. Express*, vol. 27, no. 6, pp. 8221–8235, Mar 2019. [Online]. Available: http://www.opticsexpress.org/abstract.cfm?URI=oe-27-6-8221

[18] Z. Wang, Q. Wang, B. Moran, and M. Zukerman, "Optimal submarine cable path planning and trunk-and-branch tree network topology design," *IEEE/ACM Transactions on Networking*, pp. 1–11, 2020.

[19] T. Tsubaki, M. Ishizuka, and S. Yasukawa, "A new algorithm of route design against large-scale disasters," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018, pp. 1–5.

[20] A. Alashaikh, T. Gomes, and D. Tipper, "The spine concept for improving network availability," *Computer Networks*, vol. 82, pp. 4 – 19, 2015, robust and Fault-Tolerant Communication Networks. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128615000687

[21] L. Garrote, L. Martins, U. J. Nunes, and M. Zachariasen, "Weighted euclidean steiner trees for disaster-aware network design," in *2019 15th International Conference on the Design of Reliable Communication Networks (DRCN)*, 2019, pp. 138–145.

[22] C. Cao, M. Zukerman, W. Wu, J. Manton, and B. Moran, "Survivable topology design of submarine networks," *J. Lightwave Technol.*, vol. 31, no. 5, pp. 715–730, Mar 2013. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-31-5-715

[23] P. N. Tran and H. Saito, "Geographical route design of physical networks using earthquake risk information," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 131–137, 2016.

[24] P. N. Tran and H. Saito, "Enhancing physical network robustness against earthquake disasters with additional links," *J. Lightwave Technol.*, vol. 34, no. 22, pp. 5226–5238, Nov 2016. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-34-22-5226

[25] M. O. Ball, "Computational complexity of network reliability analysis: An overview," *IEEE Transactions on Reliability*, vol. 35, no. 3, pp. 230–239, 1986.