Vulnerable Regions of Networks on Sphere

Balázs Vass*, László Németh†, Martin Zachariasen‡, Amaro de Sousa§, János Tapolcai*
*MTA-BME Future Internet Research Group, Budapest University of Technology and Economics, {vb,tapolcai}@tmit.bme.hu
†Department of Probability Theory and Statistics, Eötvös Loránd University, Budapest, Hungary, lnemeth@caesar.elte.hu
‡Faculty of Science, University of Southern Denmark, Odense, Denmark, dekan-science@sdu.dk
§Instituto de Telecomunicações / DETI, University of Aveiro, Portugal, asou@ua.pt

***Abstract*—Several recent works shed light on the vulnerability of networks against regional failures, which are failures of multiple equipments in a geographical region as a result of a natural disaster. In order to enhance the preparedness of a given network to natural disasters, regional failures and associated Shared Risk Link Groups (SRLGs) should be first identified. For simplicity, most of the previous works assume the network is embedded on an Euclidean plane. Nevertheless, since real networks are embedded on the Earth surface, this assumption causes distortion. In this work, we generalize some of the related results on plane to sphere. In particular, we focus on algorithms for listing SRLGs as a result of regional failures of circular shape.**

***Index Terms*—Shared Risk Link Groups, large scale disasters, disaster resilience**

I. Introduction

Serious network outages are happening with increasing frequency due to disasters (such as earthquakes, hurricanes, tsunamis, tornadoes, etc.) that take down almost every equipment in a geographical area (see [1] for a recent survey conducted within COST Action RECODIS [2] on strategies to protect networks against large-scale natural disasters). Such failures are called *regional failures* and can have many locations, shapes and sizes.

Due to the huge importance of telecommunication services, improving the preparedness of networks to regional failures is becoming a key issue [3]–[11]. Roughly speaking, protecting networks against regional failures is dealt with either by using geometric tools [3], [12]–[17] or by aggressively reducing the problem space to a set of failure candidate locations [5], [7]–[11]. Nevertheless, both approaches require a detailed knowledge of the geometry of the network topology, such as the exact GPS coordinates of nodes and cable conduits' routes, and the statistics of past disasters.

In many works, regional failures are computed by transforming the geographical coordinates of an existing network into a plane, which introduces distortion. Depending both on the geographical area of the network and on the transforming procedure, this distortion can vary from negligible to significant. For example, the backbone network of a small-to-medium size country is not suffering a significant distortion when compared with the uncertainty of the available geographical data, but when turning to networks covering a large country, a continent, or even multiple continents, there is no projection which can hide the spherical-like geometry of the Earth surface (see Fig. 1).

Since distortion of projections is well studied (see [18] for a comprehensive study on this field), we do not concentrate on theoretical aspects of projections and, instead, our focus is on the practical advantages and disadvantages of using a spherical model over a planar one to represent network topologies.

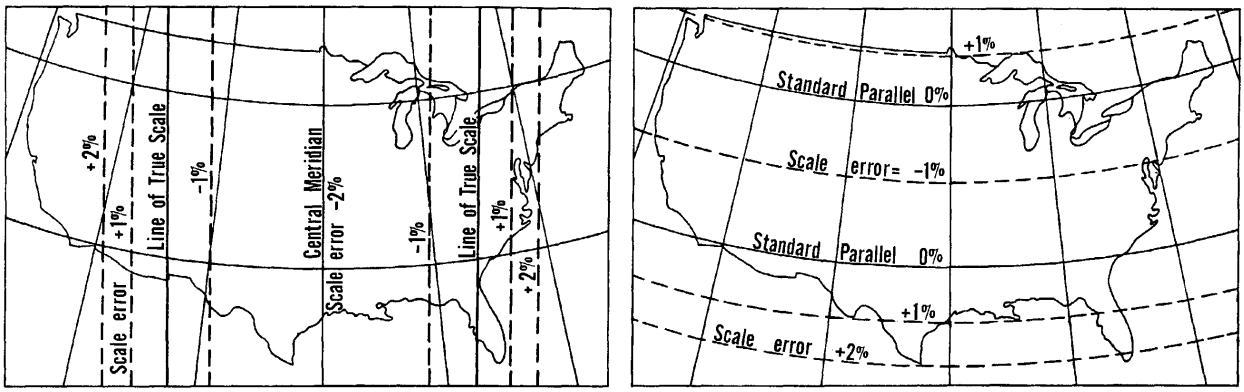
Backbone networks are designed to protect a given predefined list of failures, called *Shared Risk Link Group*s (SRLGs). Network recovery mechanisms are efficient if this SRLG list covers the most probable failure scenarios while having a manageable size.

An SRLG is called *regional* if it aims to characterize a failure damaging the network only in a bounded geographical area. It is still an ongoing research how to define and compute efficiently regional SRLG lists [15]–[17]. A common simplification of these works is that they compute the list of SRLGs on planar representation of the networks; thus, our focus is to generalize these approaches to sphere.

An SRLG consists of a set of network links, while node failures are implicitly defined (a node is considered to be failed with the SRLG if all its adjacent edges are part of the SRLG). If a failure f is listed as an SRLG, it is a common approach to skip listing any subset of f (if the network is protected to f, it is also protected to any of its subsets) and, therefore, it is enough to list the maximal SRLGs caused by disasters.

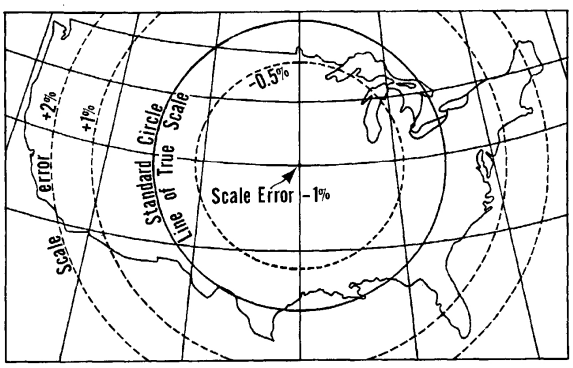
Another issue is that the number of the listed SRLGs has to be kept low. With this aim there is a common practice to fix the shape of the disasters [13], [14]. Among the possible geometric failure shapes the most natural one is the circular disc, as it is compact, and is invariant to rotation. One possibility is to overestimate the possible failures with circular disks, which yields short SRLG lists. However, it is not clear, what is the cost of this overestimation. In this work we choose to overestimate the disasters by circular disks with a maximum size according to one among many possible measures.

When talking about (maximal regional) disk failures, the most natural measure is the disk radius, which represents the maximum geographical coverage of the natural disaster. Nevertheless, since the network density is usually not homogeneous (i.e., there are more nodes and links in geographical crowded areas than in non crowded areas) the number of network elements (either nodes or links) contained by the disk are also two useful measures (it is natural that more SRLGs



(a) Transverse Mercator Projection

(b) Lambert Conformal Conic Projection



(c) Oblique Stereographic Projection

Fig. 1: Distortion patterns on common conformal map projections. Projections are shown with reduction in scale along the central meridian or at the center of projection, respectively. Each of the projections has $>3\%$ scale error over the US. [18]

are needed in crowded areas and less crowded areas can be covered with fewer SRLGs). Therefore, in this work, we will concentrate on the following three types of SRLG lists:

- *maximal r-range SRLG list*: list of maximal link sets which can be hit by a disk with radius at most r.
- *maximal k-node SRLG list*: list of maximal link sets which can be hit by a disk hitting at most k nodes.
- *maximal k-link SRLG list*: list of maximal link sets which can be hit by a disk hitting at most k links.

To distinguish between lists obtained from planar and spherical representation, we will include attribute *planar* or *spherical* in the list names (e.g. maximal spherical r-range SRLG list) when needed for clarification.

It turns out that in all three mentioned cases, the size of maximal SRLG list is linear in the network size in practice, and can be computed in low polynomial time both in planar case (maximal r-range list: [15], k-node: [17], k-link: [19]) and spherical case.

To the best of our knowledge, this paper is the first study on enumerating regional SRLG lists in a spherical model. In addition, the first comparison of the spherical and planar representations is provided. We show through an example that polynomial algorithms could be designed for spherical representation of the networks. In our experience, these algorithms are only 2 times slower than their planar pairs. We also believe that using our approach, fresh promising results as in [20] can be further enhanced.

As there are many mathematical derivations in the rest of the paper, we would like to summarize the concepts in plane text once again here for the sake of readability. As learned from previous studies, all of r-range, k-node and l-link lists can be exactly calculated in low polynomial time of the network size in case of planar representation. Our first goal was to show that considering spherical embeddings of the networks the possibility of designing low-polynomial time algorithms for determining these SRLG lists remains. We demonstrated this phenomenon in Sec. III by designing an algorithm capable of determining the r-range SLRG list both in planar and spherical case in low polynomial time. The existence of fast exact algorithms is good news, however, their drawback is that intuitively the faster the harder they are to implement. This fact motivates our second goal, namely designing a framework of simple and fast algorithms capable of determining all the mentioned SRLG lists in both planar and spherical representation with enough precision, which are presented in Sec. V.

The remaining of the paper is organized as follows. Sec. II describes the network representation model together with

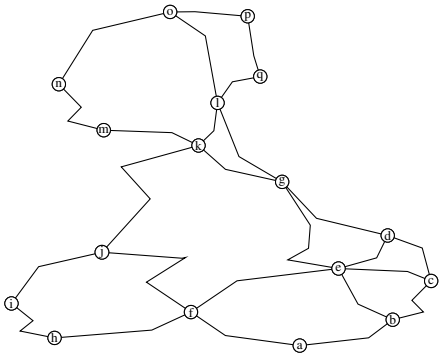


Fig. 2: Input graph $\mathscr{G}(V,\mathscr{E})$ with polylines, $n = 17$, $\gamma = 4$

the assumptions made. In Sec. III, we present an example of polynomial algorithm for computing maximal SRLG list handling both the planar and spherical cases, while in Sec. IV a faster and more flexible heuristic approach is presented for solving the same problem. Simulation results are presented in Sec. V and, finally, we draw the conclusions in Sec. VI.

II. Model and Assumptions

Throughout the paper we will consider two types of embeddings of the network: embedding in euclidean planar and spherical geometry.

The network is modeled as an undirected connected geometric graph $\mathscr{G} = (V,\mathscr{E})$ with $n = |V| \geq 3$ nodes and $m = |\mathscr{E}|$ edges. The nodes of the graph are embedded as points in the Euclidean plane or sphere, and their exact coordinates are considered to be given in 2D and 3D Cartesian coordinate system in planar and spherical case, respectively. Note that if coordinates are given in polar system (in case of spherical geometry), one can easily transform them to Cartesian at the very beginning.

When speaking of planar geometry, for each edge e there is a *polygonal chain* (or simply *polyline*) e^l in the plane in which the edge lies (see Fig 2). Parameter γ will be used to indicate the maximum number of line segments a polyline e^l can have. Naturally, in spherical case the polyline of an edge refers to a series of geodesics. Note that this model covers special cases when edges are considered as line segments (geodesics).

It will be assumed that basic arithmetic functions $(+,-,\times,/,\sqrt{\ })$ have constant computational complexity. For simplicity, we assume that nodes of V and the corner points of the containing polygons defining the possible route of the edges are all situated in general positions of the plane, i.e. there are no three such points on the same line, and no four points on the same circle, and in the spherical case there are no antipodal nodes or breakpoints and no great circles of geodesics of polylines cross the North pole.

We will often refer to circular disks simply as disks. The disk failure model is adopted, which assumes that all network elements that intersect the interior of a circle c are failed, and all other network elements are untouched.

Notation	Denomination	Short name
M_r^p	maximal planar r-range SRLG list	planar r-range list
M_k^p	maximal planar k-node SRLG list	planar k-node list
M_l^p	maximal planar l-links SRLG list	planar l-link list
M_r^s	maximal spherical r-range SRLG list	spherical r-range list
M_k^s	maximal spherical k-node SRLG list	spherical k-node list
M_l^s	maximal spherical l-links SRLG list	spherical l-link list

TABLE I: Notations and denominations of the list types

Definition 1. *A circular disk failure c hits an edge e if the polyline of the edge e^l intersects disk c. Similarly node v is hit by failure c if it is in the interior of c. Let $\mathscr{E}_c$ (and V_c) denote the set of edges (and nodes, resp.) hit by a disk c.*

We emphasize that in this model when we say e is hit by c, it does not necessarily mean that e is destroyed indeed by c, instead it means that there is a positive chance for e being in the destroyed area. In other words this modeling technique does not assume that the failed region has a shape of a disk, but overestimates the size of the failed region in order to have a tractable problem space.

In this study our goal is to generate a set of SRLGs, where each SRLG is a set of edges. Note that from the viewpoint of connectivity, listing failed nodes besides listing failed edges has no additional information. We consider SRLGs that represent worst case scenarios the network must be prepared for and, thus, there is no SRLG which is a subset of another SRLG.

Definition 2. *Let $\mathscr{C}^p$ and $\mathscr{C}^s$ denote the set of all disks in the plane and the set of all disks on the sphere, respectively. For both geometry types $g \in \{p,s\}$, let $\mathscr{C}_r^g$, $\mathscr{C}_k^g$ and $\mathscr{C}_l^g$ denote the set of disks part of $\mathscr{C}^g$ having radius at most r, hitting at most k nodes of V and hitting at most l links of $\mathscr{E}$, respectively.*

Based on the above definition, we define the set of failure states that a network may face after a disk failure, with a maximal measure.

Definition 3. *For all geometry types $g \in \{p,s\}$ and SRLG type $t \in \{r,k,l\}$, let set $F(\mathscr{C}_t^g)$ denote the set of edges which can be hit by a disk $c \in \mathscr{C}_t^g$, and let $M_t^g = M(\mathscr{C}_t^g)$ denote the set of maximal edge sets in $F(\mathscr{C}_t^g)$.*

Table I gives an overview on the corresponding notations and denominations of the SRLG list types we focus on on this paper. Note that for every SRLG type $t \in \{r,k,l\}$ if $f \in M_{t=s}^g$ there is an $f' \in M_{t=s'}^g$ such that $f \subseteq f'$ where $s \leq s'$.

The aim of this study is to propose fast algorithms computing these lists for various sizes of m.

III. Exact Algorithmic Approaches for Enumerating Maximal Failures

As mentioned before, determining the planar lists M_r^p, M_k^p, M_l^p is relatively well studied. It remains a question how much the distortion of maps can affect the calculated SRLG lists. The answer is that it heavily depends on the projection used to

Algorithm 1: Refreshing M with failure f

Input: M, f
Output: M refreshed with f
begin
  maximal:=True
  for $f_M \in M$ **do**
    if $f \subseteq f_M$ **then**
      maximal:=False
  if *maximal* **then**
    $M := M \cup \{f\}$
    for $f_M \in M$ **do**
      if $f \supset f_M$ **then**
        $M := M \setminus \{f_M\}$
  return M

Algorithm 2: Determining maximal r-range SRLG lists

Input: $\mathcal{G}(V,\mathcal{E})$, r, geometry g, coordinates of nodes and edge polylines
Output: M_r^g
begin
  $M_r^g := \emptyset$
  Store $\mathcal{E}$ as a list,
  for $i_1 \in \{1,\dots,m\}$ **do**
    for $i_2 \in \{i_2, m\}$ **do**
      for $i_3 \in \{i_3, m\}$ **do**
        $c_{i_1,i_2,i_3} := c_{\{\mathcal{E}[i_1],\mathcal{E}[i_2],\mathcal{E}[i_3]\}}$
        if *radius of* c_{i_1,i_2,i_3} *is* $\le r$ **then**
          $f := F(c_{i_1,i_2,i_3})$
          refresh M_r^g with f `// as in Alg. 1`
  return M_r^g

make the map. For example, while the *stereographic projection* affect significantly the distances, but in contrast to many other projections it has the nice property of mapping spherical disks to planar disks (Theorem in [21]) (fact also used in Appendix A). One approach for calculating spherical lists would be to adapt existing algorithms to spherical geometry demonstrating the interoperability between these geometries. However, in this paper we follow an approach simpler to present and avoiding trigonometric calculations via applying the projection in both directions for numerous times. In other words, some steps of the algorithm are performed on the plane, while others on the sphere.

In the followings we extend an exact algorithm for determining M_r^p (see [15]) to an algorithm computing M_r^p or M_r^s depending on the geometry of the input. In the rest of this section we present this extended algorithm.

A. Smallest Enclosing Disks

Let us make the following definition for the sake of clarifying the intuition.

Definition 4. *Let a disk c be smaller than disk c_0, if c has a smaller radius than c_0, or if they have equal radius and the center point of c is lexicographically smaller than the center point of c_0. Among a set of circles S_c let c be the smallest if it is smaller than any other circle in S_c.*

Definition 5. *Let $F \subseteq E$ be a finite nonempty set of edges (not necessarily a failure). We denote the smallest disk among the disks enclosing the polylines of F by c_F and we say c_F is the smallest enclosing disk of F.*

It is not difficult to see that c_F always exists for line segments or geodesics (depending on the geometry), and thus, by mapping the corresponding segments/geodesics together we can deduct that the definition is correct for polylines too. The key idea of our approach is that we can limit our focus only on the smallest enclosing disks c_F. The consequence of the next proposition is that the number of smallest enclosing disks c_F is not too large.

Proposition 6. *Let H be a nonempty set of polylines of edges with smallest enclosing disk c_H. Then there exists a subset $H_0 \subseteq H$ with $|H_0| \le 3$ such that $c_H = c_{H_0}$.*■

Definition 7. *Let $\mathcal{S}$ denote the set of maximal edge sets hit by a smallest enclosing disk.*

By Prop. 6 we have:

Corollary 8. $|\mathcal{S}| \le \binom{m}{3} + \binom{m}{2} + m = \frac{m^3}{6} + \frac{5m}{6}$.■

Lemma 9. *Let H be a set of line segments in the plane or geodesics on the sphere, $|H| \le 3$. Then c_H can be determined in $O(1)$ time.*

The proof of the Lemma is relegated to the Appendix A.

Theorem 1. *Let H be a set of polylines of edges, $|H| \le 3$. Then c_H can be determined in $O(\gamma^3)$ time.*

Proof: First, unpack each polyline into the $\le \gamma$ line segments/geodesics it is consisting of. Then, for each element h_i in H, pick a segment s_i. For each triplet (couple) of segments calculate the smallest enclosing disk (which by Lemma 9 can be done in $O(1)$), and lastly chose the smallest from among the resulting disks. ■

B. Polynomial algorithm for determining maximal failures

In this subsection, we repeat an extension of the basic algorithm provided by [15] which handles both spherical and planar inputs. There are two key facts inspiring this algorithm. Firstly, based on Prop. 6:

Corollary 10 (of Prop. 6)**.** *For both $g \in \{p,s\}$ and every $f \in M_r^g$ there exist $\{e_1,e_2,e_3\} \in f$ such that $c_{\{e_1,e_2,e_3\}} = c_f$.*■

Secondly, according to Theorem 1, smallest enclosing disks can be computed in $O(\gamma^3)$ both in plane and on sphere. Based on these, Alg. 2 is presented, which is a straightforward basic polynomial algorithm. Here, the key idea is to maintain a list M' of maximal failures detected so far while scanning though the link sets f covered by the smallest enclosing disks of at most 3 edges. If there is no $f_M \in M'$ containing f, then f is appended to M' and all $f_M \in M'$ which are part of f

are removed as presented in Alg. 1. This process is called *refreshing*.

The following theorem gives a very loose bound on the complexity of calculating M_r^g.

Theorem 2. *Alg. 2 computes M_r^g in $O(m^3(\gamma^3+m^4))$. M_r^g has $O(m^3)$ elements.*

Proof: Based on Prop. 6 the algorithm is correct, it is computing M_r^g. There are $O(m^3)$ smallest enclosing disks to calculate, each in constant time. We claim that for each disk the calculation time of refreshing M_r^g with the resulting failure (according to Alg. 1) is $O(m^4+\gamma^3)$ in case of each disk, because after the computation of the smallest enclosing disk in $O(\gamma^3)$ and determining f in $O(m)$ there has to be done $O(m^3)$ comparisons of link set, and each can be done in $O(m)$. ■

C. On improved complexity bounds

The results from the previous subsection can be easily improved using parametrization and some computational geometric tricks.

The first observation is that for any meaningful radius of the disk failure most of the network will remain intact. However, failures with the same radius taking place in a crowded area tend to take down more equipment than the ones in sparsely inhabited areas. This motivates the introduction of a graph density parameter:

Definition 11. *For every $r \in \mathbb{R}_0^+$, let ρ_r be the maximum number of edges which can be destroyed by a disk with radius at most r.*

This ρ_r is considered to be small in case of small r values. Another observation is that there are not much more network edges than nodes. This is formalized in the upcoming Claim 13.

Informally speaking, we denote the set of crossing points of the edges by X. A more formal definition follows.

Definition 12. *Let X be the set of points P in the plane on which no node element of V lies and there exist at least 2 edges which have polylines having a finite number of common points crossing each other in P. Let $x = |X|$.*

Despite the fact that on arbitrary graphs x can be even $O(n^4)$, in backbone network topologies typically $x \ll n$ because a node is usually installed if two cables are crossing each other. This gives us the intuition that G is 'almost' planar, and thus it has few edges.

Claim 13. *The number of edges in G is $O(n+x)$. More precisely for $n \geq 3$ we have $m \leq 3n+x-6$.*

Proof: If $\mathscr{G}$ is embedded in plane, do the followings. Let $G_0(V \cup X, E_0)$ be the planar graph obtained from dividing the polylines of edges of G at the crossings. Since every crossing enlarges the number of edges at least with two, $|E_0| \geq m+2x$. On the other hand, $|E_0| \leq 3(n+x)-6$ since G_0 is planar. Thus $m \leq |E_0|-2x \leq 3n+x-6$.

If $\mathscr{G}$ is embedded in sphere, we can project it to the plane with stereographic projection, repeat the former arguments then apply an inverse projection to the sphere. ■

A third trick lies on the fact that in practice $|M_r^g|$ is $O(n)$ (as presented in planar case in [15]), thus in Alg. 1 typically there has to be done only $O(n)$ comparisons. Thus we introduce a third parameter:

Definition 14. *Let λ be the maximum cardinality of the list of maximal failures detected so far in Alg. 2.*

Combining the former three observations lower parametrized complexity can be achieved:

Theorem 3. *Alg. 2 computes M_r^g in $O((n+x)^3(n+x+\lambda\rho_r+\gamma^3)$.*

Proof: Based on Prop. 6 the algorithm is computing M_r^g. There are $O((n+x)^3)$ smallest enclosing disks to calculate, each in constant time. We claim that for each disk the calculation time of Alg. 1 is $O((n+x)^3\rho_r+\gamma^3)$ in case of each disk, because after the computation of the smallest enclosing disk in $O(\gamma^3)$ and determining f in $O(n+x)$ there has to be done $O(\lambda)$ comparisons of link set, and each can be done in $O(\rho_r)$. ■

Corollary 15. *If both x and λ is $O(n)$, Alg. 2 computes M_r^g in $O(n^3(n\rho_r+\gamma^3))$ time.*

Cor. 15 proposes that M_r^g can be determined in quartic time of n in practice. On the other hand, Alg. 2 has its limits of speed: because of the three nested for-loops it runs in $\Omega(n^3)$. In order to achieve better results, the algorithm would have to be changed. For the planar case, [15] gives an algorithm which runs in $O((n+x)^2\rho_r^5)$ for $\gamma = 1$ (i.e. the edges are considered as line segments there). Furthermore, we are convinced that an algorithm with parametrized running time near linear in network size could be achieved for determining M_r^g (and also for determining M_k^g and M_l^g, despite they can be computed based on very different theories). However, presenting this kind of algorithms would exceed the limits of this paper.

IV. Heuristics and Implementation Issues

It is always good to have fast exact polynomial algorithms for solving a given problem. However, this approach also has some disadvantages: 1) the lower complexity an exact algorithm for determining a maximal circular SRLG list has, the harder to implement and prove its correctness and complexity; 2) designing algorithms for computing different types of maximal SRLG lists need totally different mathematics.

Moreover, in most cases, the available geographical data of networks is inaccurate. Adding this fact to the inconveniences of the exact approach results into the idea of designing some heuristics that are able to compute these lists *with enough precision*.

In this section we present a heuristic approach suitable for computing all types of maximal SRLG lists defined in Sec. II.

Definition 16. *For a point P (in the plane or on the sphere) and node $v \in V$, let the node-distance couple be $[v, d(v,P)]$,*

Algorithm 3: Heuristic for determining the maximal r-range SRLG lists

Input: $\mathcal{G}(V,\mathcal{E})$, r, geometry type g, coordinates of nodes and polylines of edges
Output: M_r^g
begin
1 Store $\mathcal{E}$ as a list,
2 **for** $P \in \mathscr{P}$ **do**
3 determine $e(P)$
4 **if** $e(P)[1][distance] < r$ **then**
$i := \max\{j | e(P)[j][\text{distance}] < r\}$
5 refresh M_r^g with $\{e(P)[1][\text{edge}],\ldots,e(P)[i][\text{edge}]\}$
// according to Alg. 1
6 **return** M_r^g

where $d(v,P)$ is the distance of v and P. Let $v(P)$ be the list consisting of the node-distance pairs of all nodes $v \in V$, sorted increasingly by the distance values.

We define $e(P)$ to be the list consisting of the edge-distance pairs defined similarly.

Proposition 17. *For a given point P, $v(P)$ can be computed in $O(n\log n)$, while $e(P)$ in $O((n+x)\log(n+x))$.*■

Clearly, node-distance lists, and edge-distance lists can be determined quickly. The plan is to determine these lists for enough points which are also placed well enough to be able to determine the maximal SRLG lists based on these node-distance and edge-distance lists.

Definition 18. *Let $\mathscr{P}$ denote the set of points P for which we want to construct $v(P)$ and $e(P)$.*

Let us stick to planar geometry for a moment. Intuitively, we can calculate M_r^p by including the grid points of a sufficiently fine grid (let's say containing 1 km × 1 km squares) in $\mathscr{P}$. On sphere we should choose a similar nice covering. It is possible that we have some extra short links, thus for calculating the k-node and k-link list we should include some extra points in $\mathscr{P}$. For example, by adding some random points of each polyline of edge and some point near every node we can solve this issue.

Algorithm 3 is an example heuristic for determining M_r^g.

Theorem 4. *Alg. 3 approximately computes M_r^g in $O(|\mathscr{P}|[(n+x)(\log(n+x)+\lambda)])$.*

Proof: For an element P of $\mathscr{P}$ we have to construct $e(P)$, which can be done in $O((n+x)\log(n+x))$, then refresh the list of suspected maximal failures with the resulting set $\{e(P)[1][\text{edge}],\ldots,e(P)[i][\text{edge}]\}$ in $O(\lambda(n+x))$. The proof follows. ■

Corollary 19. *If both x and λ is $O(n)$, Alg. 3 approximately computes M_r^g in $O(|\mathscr{P}|n^2)$ time.*

Comparing Cor. 19 and 15 we can see that despite the heuristic Alg. 3 is much simpler to implement, it clearly outperforms the exact Alg. 2.

TABLE II: Running times of Alg. 3 on some physical backbone topologies of [22] (in *sec*, $|\mathscr{P}| \simeq 50000$)

Name	$\lvert V\rvert$		$\lvert\mathcal{E}\rvert$		Planar runtime		Spherical runtime	
	Polygon	Line	Polygon	Line	Polygon	Line	Polygon	Line
AboveNet	9	22	15	28	232	156	410	757
LambdaNet	10	33	10	33	282	225	444	410
GARR (Italy)	16	16	18	18	107	92	204	187
GTS (Hungary)	14	15	39	26	175	146	311	291

V. Preliminary Simulation Results

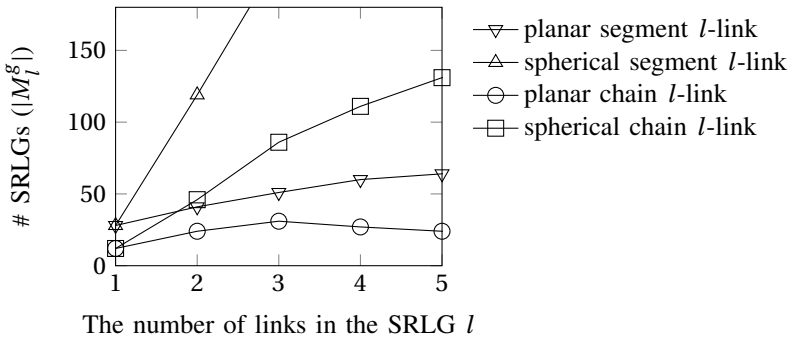
In this section, we present numerical results that validate our heuristic approach presented in Sec. IV and demonstrate the use of the proposed algorithms on some realistic physical networks. The algorithm was implemented in Python3.5 using various libraries. Distance functions were implemented from scratch. No special efforts were made to make the algorithm space or time optimal. Run-times were measured on a commodity laptop with core i5 CPU at 2.3 GHz with 8 GiB of RAM. The output of the algorithm is a list of SRLGs so that no SRLG contains the other.

We interpret the input topologies in two ways: *polygon*, where links are polygonal chains, and *line*, where the corner points of the polygonal links are substituted with nodes (of degree 2). Here links are line segments. We found that running times for spherical representations were ∼ 2 times slower than the planar ones in case of most networks (see Table II). The only exception is when the network has an extreme geographic extension (e.g. AboveNet), in this case the obtained SRLG lists tend to be longer (Fig. 3 demonstrates this in case of k-link lists) causing a slight increase both in parameter λ and in the running time.

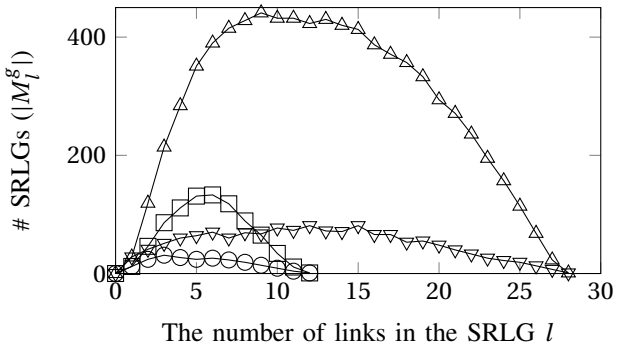
Another issue which can be noticed related to the achievable preciseness using the heuristic approach. Based on Thm. 4, running time is proportional with $|\mathscr{P}|$; given this and the running times collected in Table II, we can deduct that if the drop of price of computation power remains for an additional short time period, one will be able to run these simulations even at home for huge $|\mathscr{P}|$ (e.g. $|\mathscr{P}| \simeq 5 * 10^8$, which number is approximately the Earth's surface in km^2), yielding a high precision. Note that Alg. 3 could be easily parallelized.

The k-link list is chosen as an illustrative example on Fig. 3. In Fig. 3a and 3b we can see that for $k = 1$ there are listed all the single link failures. For $k \geq 2$ there is a higher chance on the sphere for k links to be 'close' to each other than on the plane, thus $|M_l^s| > |M_l^p|$. This phenomenon might appear because mapping the sphere to the plane intuitively lets fewer edges to be next or close to each other. As the number of links in the SRLGs l increases, $|M_l^g|$ first increases too, then after plateauing it starts to decrease, which is just a rephrasing of the intuition that there are the most possible scenarios of a disk hitting exactly l links when $l \simeq m$. Finally, $|M_m^g| = 1$, because there is only one possibility of hitting all the links.

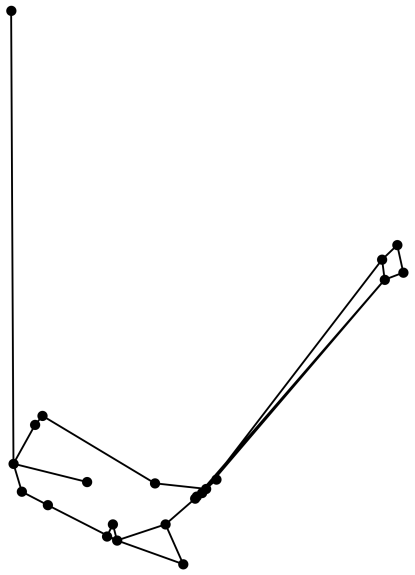
The obtained SRLG lists are different for the two geometries, thus it makes sense to use the much precise spherical model.



(a) $|M_l^p|$ compared to $|M_l^s|$ for small l values



(b) $|M_l^p|$ compared to $|M_l^s|$ for for all possible l values



(c) Best Mercator projection of AboveNet

Fig. 3: Example on extreme geographic extension: AboveNet ($n = 22$, $m = 28$ in *line* case) touching three continents.

VI. Conclusion

We investigated the problem of generating SRLG lists of networks. We found that the known exact low-polynomial SRLG generating techniques can be modified in order to fit the spherical geometry, allowing us to generate SRLG lists with more precision. A heuristic framework of easy-to-implement algorithms for determining the SRLG lists in both planar and spherical representation was also presented.

In our experience, SRLG lists generated using spherical representation of the networks are different from the planar ones, and also they tend to be longer, especially in case of extreme geographical extension. In case of our implementation, enumerating SRLG lists in case of spherical representation was typically 2 times slower than in planar case.

Acknowledgements

This article is based upon work from COST Action CA15127 (”Resilient communication services protecting end-user applications from disaster-based failures - RECODIS”) supported by COST (European Cooperation in Science and Technology). This work was partially supported by the High Speed Networks Laboratory (HSNLab) at the Budapest University of Technology and Economics, by the BME-Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC), and by the Hungarian Scientific Research Fund (grant No. OTKA K 124171).

Appendix

A. Determining Smallest Enclosing Disk of Line Segments or Geodesics in $O(1)$

Proof of Lemma 9:

Proof: For planar geometry, this problem is already solved, see Thm. 3 of paper [15]. It remains to prove it in case of spherical embedding.

Let e_1, e_2, e_3 be three geodesics on the sphere. The endpoints (p_{i1}, p_{i2}) are given by Cartesian coordinates (x_{i1}, y_{i1}, z_{i1}), (x_{i2}, y_{i2}, z_{i2}). Let p_{i3} be an arbitrary point inside e_i. Points p_{i1}, p_{i2} and p_{i3} determine the great circle on sphere containing geodesic e_i.

We will project geometric objects on the sphere to the plane using the stereographic projection from the north pole, which has the property that the image of a spheric circle will be a circle on the plane, or in special case, if it contains the north pole, its image image is a line (Theorem in [21]). Note that for the sake of simplicity it is assumed that the no great circle investtigated crosses the north pole.

Projecting the 9 spherical points onto the plane we receive q_{ij} points given by Cartesian coordinates $(x_{ij}, y_{ij}, z_{ij}) \rightarrow q_{ij} = \left(\frac{x_{ij}}{1-z_{ij}}, \frac{y_{ij}}{1-z_{ij}}\right)$. We denote the images of e_1, e_2, e_3 by arcs f_1, f_2, f_3. Calculating the radius and center point of the containing circle c_i for arc f_i requires constant number of coordinate geometric steps. Let $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ and r_1, r_2, r_3 be the Cartesian coordinates of the center of containing circles, and radiuses, respectively.

The smallest enclosing disk c_H on sphere has an image c'_H on the plane. However the parameters of c'_H are different from the parameters of c_H, the images of the fitting points of c_H and e_i are the fitting points of c'_H and f_i. That inspires the plan to find all the fitting circles of f_i (i.e. those which have exactly 1 common point with each f_i or which have 1 common point with 2 of them and containing the third) on the plane, project them back onto sphere and select the minimal among them, as that is the minimal enclosing disk of e_1, e_2, e_3. Thus we need to find the potential best fitting circles in the plane.

It is possible that the disk fits for two arcs and include some points of the third. We can choose two arbitrary arcs in 3 ways. Choosing f_1, f_2 we must calculate the distance of the two arcs and use it as the diameter of the potential disk. On each arc the distance is determined by an inside point or one of the boundary points. Calculating the distance of two points, a point and a circle or two circles have both constant complexity. So in this case $3 \cdot 3^2 \cdot O(1)$ calculation required.

If the smallest disk touches all of the arcs there are also more different cases. Each arc can be touched on a boundary point or on an inside point (3^3 cases). Fortunately fitting a circle is already solved in all of the cases and called problems of Apollonius [23].

If the smallest enclosing disk touches all three arcs f_1, f_2 and f_3, we have three cases for each arc f_i: the disk either touches the arc in an interior point or at one of its endpoints. In the former case let (x_1, y_1) and r_i be the Cartesian coordinates of center point and radius of the containing circle of arc f_i, respectively. In the latter case, let (x_1, y_1) be coordinates of the endpoint itself, while let r_i be 0. Numbers s_1, s_2 and s_3 are $+/-1$ representing that the fitting circle touches on the outside or on the inside of the containing circles of c_1, c_2 and c_3 (2^3 different possibility to be checked on each case). Parameters x_s, y_s and r_s of the fitting C circle can be calculated by solving the following equation system [24]:

$$(x_s - x_1)^2 + (y_s - y_1)^2 = (r_s - s_1 \cdot r_1)^2$$
$$(x_s - x_2)^2 + (y_s - y_2)^2 = (r_s - s_2 \cdot r_2)^2$$
$$(x_s - x_3)^2 + (y_s - y_3)^2 = (r_s - s_3 \cdot r_3)^2.$$

The system in quadratic, thus it can be solved by constant number of arithmetic calculations. The complexity of these calculations all together are $3^3 \cdot 2^3 \cdot O(1)$.

After finding the $3^3 \cdot 2^3 + 3 \cdot 3^2$ possible minimal disks, we must project them back to the surface of the sphere. We are allowed to use only the two endpoint of an arbitrary diameter from each possible circle. This requires $2 \cdot 3^5$ number of coordinate transformations $(x, y) \to \left(\frac{2x^2}{1+x^2+y^2}, \frac{2y^2}{1+x^2+y^2}, \frac{-1+x^2+y^2}{1+x^2+y^2}\right)$.

Finding the minimal radius of potential disks requires $3^5 - 1$ comparisons between diameters. Using this method the minimal disk for e_1, e_2, e_3 can be determined in $O(1)$ time. However the algorithm could be improved by using preconceptions for the edges, exclude some possible disks already on plane instead of transforming back or fixing s_i in case of boundary points. Note that only basic arithmetic functions $(+, -, \times, /, \sqrt{\ })$ were used during the computation. ■

REFERENCES

[1] 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 *RNDM*, Sept 2016.

[2] J. Rak, D. Hutchison, E. Calle, T. Gomes, M. Gunkel, P. Smith, J. Tapolcai, S. Verbrugge, and L. Wosinska, "RECODIS: Resilient communication services protecting end-user applications from disaster-based failures," in *ICTON*, July 2016.

[3] S. Neumayer, G. Zussman, R. Cohen, and E. Modiano, "Assessing the vulnerability of the fiber infrastructure to disasters," *IEEE/ACM Transactions on Networking (TON)*, vol. 19, no. 6, pp. 1610–1623, 2011.

[4] O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" *Communications Magazine, IEEE*, vol. 50, no. 2, pp. s12–s20, 2012.

[5] M. F. Habib, M. Tornatore, M. De Leenheer, F. Dikbiyik, and B. Mukherjee, "Design of disaster-resilient optical datacenter networks," *Journal of Lightwave Technology*, vol. 30, no. 16, pp. 2563–2573, 2012.

[6] J. Heidemann, L. Quan, and Y. Pradkin, *A preliminary analysis of network outages during hurricane Sandy*. University of Southern California, Information Sciences Institute, 2012.

[7] F. Dikbiyik, M. Tornatore, and B. Mukherjee, "Minimizing the risk from disaster failures in optical backbone networks," *Journal of Lightwave Technology*, vol. 32, no. 18, pp. 3175–3183, 2014.

[8] I. B. B. Harter, D. Schupke, M. Hoffmann, G. Carle *et al.*, "Network virtualization for disaster resilience of cloud services," *Communications Magazine, IEEE*, vol. 52, no. 12, pp. 88–95, 2014.

[9] X. Long, D. Tipper, and T. Gomes, "Measuring the survivability of networks to geographic correlated failures," *Optical Switching and Networking*, vol. 14, pp. 117–133, 2014.

[10] B. Mukherjee, M. Habib, and F. Dikbiyik, "Network adaptability from disaster disruptions and cascading failures," *Communications Magazine*, vol. 52, no. 5, pp. 230–238, 2014.

[11] R. Souza Couto, S. Secci, M. Mitre Campista, K. Costa, and L. Maciel, "Network design requirements for disaster resilience in iaas clouds," *Communications Magazine, IEEE*, vol. 52, no. 10, pp. 52–58, 2014.

[12] M. T. Gardner and C. Beard, "Evaluating geographic vulnerabilities in networks," in *IEEE Int. Workshop Technical Committee on Communications Quality and Reliability (CQR)*, 2011, pp. 1–6.

[13] P. K. Agarwal, A. Efrat, S. K. Ganjugunte, D. Hay, S. Sankararaman, and G. Zussman, "The resilience of wdm networks to probabilistic geographical failures," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 5, pp. 1525–1538, 2013.

[14] S. Trajanovski, F. Kuipers, P. Van Mieghem *et al.*, "Finding critical regions in a network," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2013, pp. 223–228.

[15] J. Tapolcai, L. Rónyai, B. Vass, and L. Gyimóthi, "List of shared risk link groups representing regional failures with limited size," in *Proc. IEEE INFOCOM*, Atlanta, USA, May 2017.

[16] J. Tapolcai, B. Vass, Z. Heszberger, J. Biró, D. Hay, F. A. Kuipers, and L. Rónyai, "A tractable stochastic model of correlated link failures caused by disasters," in *Proc. IEEE INFOCOM*, Honolulu, USA, 2018.

[17] B. Vass, E. Bérczi-Kovács, and J. Tapolcai, "Enumerating shared risk link groups of circular disk failures hitting k nodes," in *DRCN*, Munich, Germany, March 2017.

[18] J. P. Snyder, *Map projections: A working manual*. U.S. Government Printing Office, 1987.

[19] E. Papadopoulou and M. Zavershynskyi, "The higher-order Voronoi diagram of line segments," *Algorithmica*, vol. 74, no. 1, pp. 415–439, 2016.

[20] A. Kushwaha, D. Kapadia, A. Gumaste, and A. Somani, "Designing multi-layer provider networks for circular disc failures," in *ONDM*, Dublin, Ireland, May 2018.

[21] [Online]. Available: http://www.cs.bsu.edu/homepages/fischer/math345/stereo.pdf

[22] "The internet topology zoo." [Online]. Available: http://www.topology-zoo.org/dataset.html

[23] H. S. M. Coxeter, "The problem of apollonius," in *The American Mathematical Monthly*, vol. 75, no. 1, pp. 5–15.

[24] C. G.W., "Analytical solutions of the ten problems in the tangencies of circles; and also of the fifteen problems in the tangencies of spheres," in *The Mathematical Monthly*, vol. 2, pp. 116–126.