Shared Risk Link Group Enumeration of Node Excluding Disaster Failures

Balázs Vass[*†], Erika Bérczi-Kovács[*], János Tapolcai[†]
[*]Department of Operations Research, Eötvös University, Budapest, Hungary, vassbalazs@gmail.com,koverika@cs.elte.hu
[†]MTA-BME Future Internet Research Group, Budapest University of Technology and Economics, tapolcai@tmit.bme.hu

***Abstract*—Current backbone networks are designed to protect a certain pre-defined list of failures, called Shared Risk Link Groups (SRLG). During network design and operation protecting a failure not part of an SRLG is ignored as they assume to be extremely rare events. The list of SRLGs must be defined very carefully, because leaving out one likely failure event will significantly degrade the observed reliability of the network. The list of SRLGs is typically composed of every single link or node failure. It has been observed that some type of failure events manifested at multiple locations of the network, which are physically close to each other. Such failure events are called regional failures, and are often caused by a natural disaster. In this study we focus on link failures only and assume nodes are never part of the failure. We provide a fast systematic approach for generating a list of SRLGs the protection of which is essential to increasing the observed reliability of the network. According to some practical assumptions this list is very short with $O(|V|)$ SRLGs in total, and can be computed very fast, in $O(|V|\log|V|)$ time.**

I. Introduction

Current backbone networks are built to protect a certain list of failures. Each of these failures (or termed failure states) are called *Shared Risk Link Groups* (SRLG), which is a set of links that is expected to fail simultaneously. The network is designed to be able to automatically reconfigure in case of a single SRLG failure, such that every connection further operates after a very short interruption. In practice it means the connections are reconfigured to by-pass the failed set of nodes and links. Thus the network can recover if an SRLG or a subset of link and nodes in SRLG fails simultaneously; however, there is no performance guarantee when a network is hit by a failure that involves links not a subset of an SRLG. Nevertheless, the list of SRLGs must be defined very carefully, because not getting prepared for one likely simultaneous failure event the observed reliability of the network significantly degrades.

Operators have numerous answers what simultaneous failures means. One extreme is to list every *single link or node failure* as an SRLG. Here the concept is that the failure first hits a single network element for whose protection the network is already pre-configured. Often there is a known risk of a simultaneous multiple failure that can be added as an SRLG, for example if two links between different pair of nodes traverse the same bridge, etc. On the other hand, we have witnessed serious network outages [1]–[9] because of a failure event that takes down almost every equipment in a physical region as a result of a disaster, such as weapons of mass destruction attacks, earthquakes, hurricanes, tsunamis, tornadoes, etc. For example the 7.1-magnitude earthquake in Taiwan in Dec. 2006 caused simultaneous failures of six submarine links between Asia and North America [10], the 9.0 magnitude earthquake in Japan Earthquake on March 2011 impacted about 1500 telecom switching offices due to power outages [5] and damages of undersea cables, the hurricane Katrina in 2005 caused severe losses in Southeastern US [11], hurricane Sandy in 2012 cased a power outage that silenced 46% of the network in the New York area [4]. These type of failures are called *regional failures* which is a simultaneous failures of nodes/links located in specific geographic areas [1]–[9]. The number of possible regional failure can be very large, thus simply listing them as an SRLG is not a viable solution. It is still a challenging open problem how to prepare a network to protect such failure events, as their location and size is not known at planning stage. In the paper we propose a solution to this problem with a technique that can significantly reduce the number of possible failure states that should be added as an SRLG to cover all regional failures.

In practice, regional failures can have any location, size and shape. It is a common best practice to fix the size or shape of regional failures, for example as cycles with given size (also called disk) [12]. In our study we assume the regional failure has a shape of cycle but do not fix its size. Instead we classify the regional failures according to the network elements it has. For example if the failure hit a network node, the node is no longer going to send traffic in the network which has a network wide effect. To treat the node failure separately form link failures, the first class of failure are disk failures that hits links only. Clearly, a disk failure that does not have a node cannot be too large. The second class of failures are disk failures that affects nodes as well which is ignored in this paper and left for further study.

In this study the number of SRLG are significantly reduced applying computational geometric tools based on the following assumptions:

1) The network is a geometric graph $G(V,E)$ embedded in a 2D plane.
2) The shape of the regional failure is assumed to be a circle with arbitrary radius and center position.
3) We focus on **regional link failures**, failures that does not affect nodes.

We will show with these assumption the number of SRLGs is small, $O(|V|)$, in typical backbone network topology, and can be at most $O(|E||V|)$ in an artificial worst case scenario, where $|V|$ denotes the number of nodes in the network, and $|E|$ the number of links. We propose a systematic approach based on computational geometric tools that can generate the list of SRLGs in $O(|V|\log|V|)$ steps on typical networks.

Using the obtained SRLG list network operators can design their networks to be protected against regional and random failures. Backbone networks designed according to our new failure model should have higher reliability, and leave way less failures to be recovered with the convergence of higher layer intra-domain routing protocols (IS-IS, OSPF) with the next few seconds, minutes or hours after the failure. We believe the paper fills the gap between the conventional SRLG based pre-planned protection and regional failures.

The paper is organised as follows. In Section II we present the core mathematical model with several observations. In Section III we show the main result the $O(|V|\log|V|)$ algorithm for generating the list of SRLG covering every regional link failures with a shape of disk that do not contain network nodes. Finally, in Section IV we draw the conclusions.

II. Model and Assumptions

We model the network as an undirected geometric graph $G(V,E)$ with $n=|V|$ nodes and $m=|E|$ edges, we assume $n\geq 3$. The nodes of the graph are embedded as points in the Euclidean plane, and the edges are embedded as line segments. The position of node v is denoted by (v_x, v_z). A *disk* $c(x,y,r)$ is a circle with a centre point (x,y) and radius r. The failure caused by a disk is modelled as every interior node and edge with interior part is erased from the graph.

For every disk c let E_c denote the set of edges and nodes erased by c.

Proposition 1. *For any $c_1, c_2 \in C$, $c_1 \subseteq c_2$ it holds that $E_{c_1} \subseteq E_{c_2}$.* □

Let C_0 denote the set of disks that do not have any node of V in their interior. Clearly, $|C_0|$ is infinite.

Claim 2. *For any $c_1(x,y,r) \in C_0$ there exists a $c_2 \in C_0$ such that $E_{c_1} \subseteq E_{c_2}$ and c_2 has at least 2 nodes of V on its boundary.*

Proof. If $E_{c_1} = \emptyset$, than we can choose c_2 arbitrarily from among the non-empty set of disks with at least 2 points of V on their boundary.

If there exists an edge $e=\{a,b\}$ in E_{c_1}, than we generate c_2 as follows. We start with disk $c_1(x,y,r)$ and start to increase its radius. We do it until we reach a node $u \in V$. We can further blow the circle larger without loosing any covered area by moving the central point along the line $(x,y)-(u_x,u_y)$ while keeping u on the boundary.

Assume indirectly that it never reaches a second node. We get a contradiction because c_1 intersects line ab and $a,b \in V$.

Thus the circle will reach a second node $v \in V$. Let $c_2 \supseteq c_1$ be this circle having $u,v \in V$ on its boundary. Clearly, $E_{c_2} \supseteq E_{c_1}$. □

For nodes u and v, let $C_0^{u,v}$ be the set of disks from C_0 which have both u and v on the boundary.

Firs let us ignore the edges of the network and focus only on the nodes. We are searching for disks of maximum size that do not have any nodes in interior. Clearly, each disk of maximum size passes through at least three nodes, otherwise its size could be further increased. By simplicity we assume that the nodes are in general position i.e. no four nodes are on the same cycle and no three nodes are on the same line. In this case connecting the three nodes we get triangles. The problem was deeply investigated in the past and it was shown the union of these triangles results a triangulation of the graph, called Delaunay triangulation [13]. Let $D_\triangledown = (E_\triangledown, V)$ denote the Delaunay triangulation on the set of nodes, where $E_\triangledown$ denotes the edges of the triangulation, which can be very different form the edges of the network. In Delaunay triangulation no circumcircle of any triangle contains node in interior. Another interesting property that the dual graph of the Delaunay triangulation is called Voronoi diagram [14]. An important observation is the following.

Proposition 3. *$C_0^{u,v}$ is non-empty iff $\{u,v\}$ is an edge of the $D_\triangledown = (V, E_\triangledown)$ Delaunay triangulation.* □

Let $\mathscr{F}_0$ and $\mathscr{F}_0^{u,v}$ be the set of failures caused by elements of C_0 and $C_0^{u,v}$, respectively. Formally, $\mathscr{F}_0 = \{E_c | k \in C_0\}$ and $\mathscr{F}_0^{u,v} = \{E_c | k \in C_0^{u,v}\}$. We call the elements of $\mathscr{F}_0$ *regional link failures*, or simply *link failures*.

Let denote $\mathscr{M}_0$ and $\mathscr{M}_0^{u,v}$ the exclusion-wise maximal elements of $\mathscr{F}_0$ and $\mathscr{F}_0^{u,v}$, respectively. Our goal is to determine $\mathscr{M}_0$.

Claim 4. $\mathscr{M}_0 \subseteq \bigcup_{\{u,v\}\in E_\triangledown} \mathscr{M}_0^{u,v}$.

Proof. Clearly, for all $f \in \mathscr{M}_0$ there exists a $c_1 \in C_0$ such that $f = E_{c_1}$. According to Claim 2 and Prop. 3, there exists a $c_2 \in \cup_{\{u,v\}\in E_\triangledown} C_0^{u,v}$ for which $E_{c_2} \supseteq E_{c_1}$. This implies $f \subseteq E_{c_2}$. Since f is an exclusion-wise maximal element of $\mathscr{F}_0$ by definition of $\mathscr{M}_0$, this is possible only if $f = E_{c_2}$.

We get that for every $f \in \mathscr{M}_0$ there exists a $c_2 \in \cup_{\{u,v\}\in E_\triangledown} C_0^{u,v}$ such that $f = E_{c_2}$. This implies $\mathscr{M}_0 \subseteq \cup_{\{u,v\}\in E_\triangledown} \mathscr{M}_0^{u,v}$. □

Before presenting our algorithm for determining $\mathscr{M}_0$, we should take a look on its size. It turns out that $|\mathscr{M}_0|$ is $O(nm)$ (Claim 6), and in case of some artificial network families $|\mathscr{M}_0|$ is $\Theta(n^3)$ (Cor. 7). The details are the following.

On Fig. 1 we can see a sketch of a $G=(V,E)$ graph having $\Theta(n^3)$ maximal single link failures. It has a so long-drawn shape it cannot be drawn precisely in a paper.

Let us consider $v_1, \ldots, v_k \in V$, different points lying on a vertical line l, $d(v_i, v_{i+1}) = d(v_j, v_{j+1})$, for all $i,j \in \{1,\ldots,k-1\}$, and $\{v_i, v_{i+1}\} \in E$ for every $i \in \{1,\ldots,k-1\}$.

Both on the right and left side of l let us take a complete bipartite graph called $K_{j,j}^r$ and $K_{j,j}^l$. We locate the points of K^r and K^l carefully as follows. For both $K_{j,j}^r$ and $K_{j,j}^l$ one class of nodes is located on the top and the other in the bottom are such that their vertices are equidistant on a horizontal line, and for all $v \in V \setminus \{v_1,\ldots,v_c\}$ $|d(v,l) - d(v_1,v_2)/2| \leq \varepsilon$ for a very small $\varepsilon > 0$. Though the top partiles are much further from

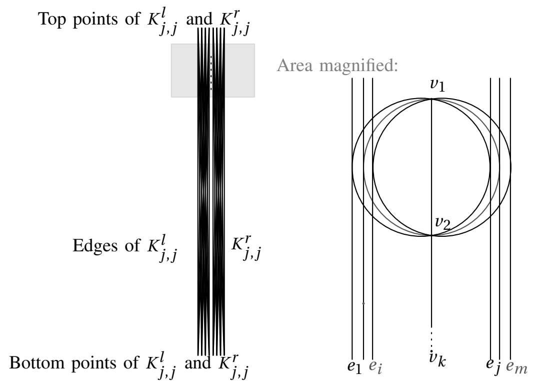

Fig. 1: Sketch of a graph family with $\Theta(n^3)$ maximal regional link failures

v_1 than $d(v_1, v_c)$, they are much closer to them as the bottom partiles. The graph is pictured in figure 1. If we want G to be connected, than take an edge from v_c to the bottom class.

This way we can manage to have $2j^2$ edges nearly parallel with l, which in addition can be arranged nearly equidistant both in the left and right half plane.

Claim 5. *In case that* $6|(n-1)$, $k := (n-1)/3+1$, $j := (n-1)/6$, *the graph sketched in Fig. 1 has at least* $\frac{(n-1)^3}{108}$ *maximal single link failures.*

Proof. It can be shown that the graph has j^2 maximal regional link failures which can have only he v_i, v_{i+1} point pair on the boundary, for every $i \in \{1, \ldots, k-1\}$. This means at least $kj^2 = \frac{(n-1)^3}{108}$ maximal regional link failures. □

Claim 6. *The maximum number of single regional link failures can be at most* $O(nm)$.

Proof. The proof can be made by using Claim 4 and the facts that $|E_\nabla| \leq 3n-6$ and $|\mathcal{M}_0^{u,v}|$ is $O(m)$ (corollary of correctness of Algorithm 3). □

Corollary 7. *The graph illustrated on Fig. 1 has* $\Theta(n^3)$ *regional link failures.* □

The size of $\mathcal{M}_0$ affects the computational complexity of its determination. However $|\mathcal{M}_0|$ can be $\Theta(nm)$, in case of many real-life networks it is $O(n)$. This gives us the idea to use some parameters which are in relation with the size of $\mathcal{M}_0$ and with the computational complexity.

We use parameters θ_0 and τ_0 for the maximum number of edges crossing the circumcircle of a Delaunay triangle, and for the maximum number of circumcircles of Delaunay triangles crossed by an edge, respectively.

Since $|\mathcal{M}_0|$ can be asymptotically large, it is not possible to give an algorithm which is "really fast" on all graphs. On the other hand, our algorithm computes $\mathcal{M}_0$ in $O(n(\log n + \theta_0^3\tau_0))$ time (Thm. 8), what gives $O(n \log n)$ if θ_0 is constant and τ_0 is $O(\log n)$, which is a natural assumption for many types of networks.

Our algorithm computes $\mathcal{M}_0$ in the following way. First it generates the Delaunay triangulation $D_\nabla = (E_\nabla, V)$. After that for every $\{u,v\} \in E_\nabla$ it generates sets $\mathcal{M}_0^{u,v}$. Finally it computes $\mathcal{M}_0$ by gathering the globally maximal elements of sets $\mathcal{M}_0^{u,v}$. We will realise this plan in Section III.

III. The Algorithm

Consider the Delaunay triangulation $D_\nabla = (V, E_\nabla)$ and the set T_0 of Delaunay triangles given by their vertices. Since D_∇ is a planar graph, for an edge (u,v) there exists one or two nodes in V that are neighbours of both u and v.

If there exist two points of this kind, let us call them w_1 and w_2. In this case both $\{u,v,w_1\}$ and $\{u,v,w_2\}$ are elements of T_0. Let $C^1_{u,v}$ and $C^2_{u,v}$ be the disks with u,v,w_1 and u,v,w_2 on the boundary.

If there exists only one common neighbour w_1 of u and v, let $C^1_{u,v}$ be the same as before, and let $C^2_{u,v}$ be an infinitely large disk with boundary going through u and v not containing w_1. Thus $\{u,v,w_1\} \in T_0$ and $\{u,v,w_2\} \notin T_0$.

Let T_0^t denote the set of Delaunay-triangles which have common edge with t, for all $t \in T_0$.

It is easy to see that all disks in $C_0^{u,v}$ are covered by $C^1_{uv} \cup C^2_{u,v}$.

Let $e \in E^3_{u,v}$ iff $e \in E$ intersects $C^1_{u,v} \cap C^2_{u,v}$, $e \in E^1_{u,v}$ iff $e \in E \setminus E^3_{u,v}$ intersects $C^1_{u,v} \setminus C^2_{u,v}$ and $e \in E^2_{u,v}$ iff $e \in E \setminus E^3_{u,v}$ intersects $C^2_{u,v} \setminus C^1_{u,v}$ (see Figure 2).

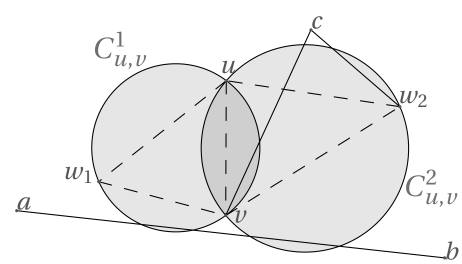

Fig. 2: Example on a Delaunay-edge $\{u,v\}$ with w_1, w_2, $C^1_{u,v}$ and $C^2_{u,v}$. Here $E^1_{u,v} = \{\{a,b\}\}$, $E^2_{u,v} = \{\{a,b\},\{c,w_2\}\}$, $E^3_{u,v} = \{\{c,v\}\}$.

Using the previous plan and the specific properties of the Delaunay triangulation we proved the next theorem.

Theorem 8. $\mathcal{M}_0$ *can be computed in* $O(n(\log n + \theta_0^3\tau_0))$ *using Algorithm 1, and has* $O(n\theta_0)$ *elements, each of them consisting of* $O(\theta_0)$ *edges.*

Proof. At line 1 the Delaunay triangulation can be computed in $O(n \log n)$ time [13]. Sets T_0 and T_0^t for all $t \in T_0$ also can be computed in $O(n \log n)$.

In line 2 we do some preparation in constant time for every Delaunay edge $\{u,v\}$, then we calculate sets $E^i_{u,v}$ simultaneously for all $\{u,v\} \in E_\nabla$ in $O(n\theta_0^2)$ time according to Lemma 11.

In line 3 we generate sets $\mathscr{M}_0^{u,v}$, each in $O(\theta_0^2)$ time (Lemma 13).

Finally, in line 4 $\mathscr{M}_0$ is calculated from lists $\mathscr{M}_0^{u,v}$ in $O(n\theta_0^3\tau_0)$ time (Lemma 16).

According to these results we can derive Theorem 8.

□

Corollary 9. *Assuming θ_0 is upper bounded by a constant and τ_0 is $O(\log n)$, $\mathscr{M}_0$ can be computed in $O(n\log n)$ time, and the total length of it is $O(n)$.*

Algorithm 1: Generating the maximal regional link failures

Input: $G = (V, E)$
Output: The set $\mathscr{M}_0$ of maximal single regional failures.
begin
1 $E_\triangledown, T_0, T_0^t (t \in T_0) \leftarrow$DELAUNAY$(V)$;
2 $E_{u,v}^i (i \in \{1,2,3\}, \{u,v\} \in E_\triangledown) \leftarrow$ GETEDGESETS$(V, E, E_\triangledown, T_0)$;
3 $\mathscr{M}_0^{u,v}(\{u,v\} \in E_\triangledown) \leftarrow$ GENERATE $(V, E, E_\triangledown, E_{u,v}^i (i \in \{1,2,3\}, \{u,v\} \in E_\triangledown))$;
4 $\mathscr{M}_0 \leftarrow$ ELIMINATEREDUNTANTS$(\mathscr{M}_0^{u,v}, \forall \{u,v\} \in E_\triangledown)$;
return $\mathscr{M}_0$

A. On the number of edges and size of θ_0

Lemma 10. *The number of edges is $O(n\theta_0)$, more precisely $m \le (2n-5)\theta_0$.*

Proof. The Delaunay triangulation is a planar graph and thus $|E_\triangledown| \le 3n - 6$. Since every Delunay triangle has 3 Delaunay edges, a Delaunay edge is edge of at most 2 Delaunay triangles, and there are at least 3 Delaunay edges on the convex hull of V, the number of Delaunay triangles is at most

$$\frac{2|E_\triangledown| - 3}{3} \le \frac{2}{3}(3n-6) - 1 = 2n - 5.$$

Since every $c \in C_0$ intersects at most θ_0 edges of the network, and contains a Delaunay triangle and every edge intersects at least one triangle, the m number of the links cannot be larger than θ_0 times the number of the Delaunay triangles. We get $m \le (2n-5)\theta_0$. □

A graph family may have $O(n^3)$ single regional failures and as mentioned before, we gave an artificial graph family, which has $\Theta(n^3)$ of them (see Fig. 1). However, we are convinced that θ_0 is small in case of typical backbone networks and there exists a small constant c that it never exceeds and thus $|\mathscr{M}_0| \le cn$.

B. Algorithm 2 (Method Getedgesets)

Lemma 11. *Algorithm 2 computes sets $E_{u,v}^i$ for all $\{u,v\} \in E_\triangledown$ in $O(n\theta_0)$. If θ_0 is constant, this gives $O(n)$ time.*

Proof. First we have to show the correctness of the algorithm.

Circles $C_{u,v}^i$, $i \in \{1,2\}$ are the circumcircles of the Delaunay triangles, unless $\{u,v\}$ is on the convex hull of V. By definition, if $\{u,v\}$ is on the convex hull of V, $E_{u,v}^2$ is empty set. Therefore, it is easy to see that assuming that in lines 5 - 9 we compute every edge set E_t covered by circumcirlce of Delaunay tirangle t, in lines 10 - 12 we also get set $E_{u,v}^i$.

It remains to prove that in lines 5 - 9 we compute sets E_t correctly. With this object it is enough to prove that for every $\{a,b\} \in E$, $\{a,b\} \in E_t$ iff $[a,b] \cap C_t \neq \emptyset$. It is obvious that if $\{a,b\} \in E_t$ after the last run of function Examine, then $[a,b] \cap C_t \neq \emptyset$. Lemma 12 shows the other way of the statement. Thus Algorithm 2 is correct.

We make an estimation of the complexity of Algorithm 2 as follows.

Clearly, calculations in lines 1-4 can be done overall in $O(n)$ time.

Since a Delaunay triangle t has at most 3 Delaunay triangles having common edge with t, a call of the function Examine runs in constant time, and for every $e \in E$ we examined at most 4 times as many triangles as the number of circumcircles crossing e. This means that in lines 5 - 9 we get sets E_t in $O(m\theta_0)$ time, or by Lemma 10, $O(n\theta_0^2)$ time. If θ_0 is constant, this gives an $O(n)$ complexity.

It is easy to see that lines 10 - 12 we find the desired $E_{u,v}^i$ sets in similar complexity.

The overall complexity of Algorithm 2 is $O(n\theta_0^2)$ time. If θ_0 is constant, this means $O(n)$.

□

Lemma 12. *The set T_e of the Delaunay triangles having circumcircles covering edge e is connected in the sense that from every element of T_e one can reach every element of T_e through triangles having common edge.*

Proof. For an edge $e = \{u,v\}$, let the set of Delaunay triangles with circumcircle intersecting e be T_e. Let the set of Delaunay tringles intersecting e be S_e ($S_e \subseteq T_e$). Trivially, the triangles of S_e are connected.

Indirectly assume that set $X \subseteq T_e \setminus S_e$ of elements of $T_e \setminus S_e$ not connected with S_e is not empty. Let X_r and X_l be the set of elements of X on the right side of line uv, and on the left side of it, respectively. Let X_c be the set of elements of X which have points both on the right and left side of line uv. This way X_r, X_l and X_c is a partition of X.

Assume there exists a $t_{XYZ} \in X_c$. Since $t_{XYZ} \notin S_e$, t_c is not intersecting edge $\{u,v\}$. This means that u or v must be intersected by the circumcircle C_{XYZ} of t_{XYZ}. Assume w.l.o.g. that C_{XYZ} covers u. Since t_{XYZ} is a Delaunay triangle, u cannot be in interior of C_{XYZ}, and thus it is situated right on C_{XYZ}, which contradicts the assumption that the nodes of V are in general position. This gives that $X_c = \emptyset$.

Let t_{PQR} be the element of X_r with maximal area of its circumcirlce on the left side of line uv (see Fig. 3).

Trivially, P, Q and R are all in the half plane on the right side of line uv. Assume w.l.o.g. that arc $\widehat{QR}$ of the circumcircle C_{PQR} of t_{PQR} not containing P is intersecting line uv. Now $[QR]$ cannot be on the convex hull of V, because P is situated right from it, and at least one from u and v is

Algorithm 2: GETEDGESETS $E^1_{u,v}, E^2_{u,v}, E^3_{u,v}$ for all $\{u,v\} \in E_\nabla$

Input: V, E, E_∇, T_0
Output: $E^i_{u,v}$ for all $i \in \{1,2,3\}, \{u,v\} \in E_\nabla$
begin
  for $\{u,v\} \in E_\nabla$ **do**
    Determine $w_1, w_2, C^1_{u,v}$ an $C^2_{u,v}$;
    $P^1 \leftarrow$ the half plane having uv line as boundary and containing w_1 ("the half plane on left hand side");
    $P^2 \leftarrow$ the half plane having uv line as boundary and containing w_2 ("the plane on right hand side");
  for $t \in T_0$ **do**
    $E_t \leftarrow \emptyset$
  for $\{a,b\} \in E$ **do**
    for $t \in T_0$ **do**
      $Visited_t \leftarrow false$
    Take a $t = awz_\Delta$ Delaunay triangle;
    Examine($t, \{a,b\}$);
  for $\{u,v\} \in E_\nabla$ **do**
    for $i \in \{1,2\}$ **do**
      if $w_i \in V$ **then**
        $E^i_{u,v} \leftarrow E_{w_i uv_\Delta} \setminus E_{w_{3-i} uv_\Delta}$;
      else if $\{u,v\} \in E$ **then**
        $E^i_{u,v} \leftarrow \{\{u,v\}\}$
    $E^3_{u,v} \leftarrow E_{w_1 uv_\Delta} \cap E_{w_2 uv_\Delta}$;
  return $E^i_{u,v}$ for all $i \in \{1,2,3\}, \{u,v\} \in E_\nabla$
Function *Examine*($t, \{a,b\}$)
  if $Visited_t = false$ *and* $[a,b] \cap C_t \neq \emptyset$ **then**
    $Visited_t \leftarrow true$;
    $E_t \leftarrow E_t \cup \{\{a,b\}\}$;
    for $t_i \in T^t_0$ **do**
      Examine($t_i, \{a,b\}$)

situated on the left side of line QR. This means there must exist a point $S \in V$ on the left part of line QR such that t_{QRS} is a Delaunay triangle. Clearly, $S \notin int(C_{PQR})$, since we have Delaunay triangulation.

If t_{QRS} intersects edge $\{u,v\}$, then $t_{QRS} \in S_e$, thus t_{PQR} is connected to S_e, which contradicts to its choice.

If t_{QRS} does not intersect edge $\{u,v\}$, but it intersects line uv, then $t_{QRS} \in X_c$, since t_{PQR} is not connected with S_e and we can deduct that $\emptyset \neq C_{PQR} \cap [uv] \subset int(C_{QRS})$ from the definition of the Delaunay triangulation. This contradicts the fact that $X_c = \emptyset$.

This means that S is in the half plane right from line uv. Since $\emptyset \neq C_{PQR} \cap [uv] \subset int(C_{QRS})$, t_{QRS} is element of T_r. It is easy to see that the area of disk C_{PQR} left from line QR is contained by the area of disk C_{QRS} left from line QR, which contradicts the choice of t_{PQR}. Thus X_r is empty.

X_l is empty for similar reasons.

It turned out that $X = X_r \cup X_l \cup X_c = \emptyset$, and thus T_e is connected. □

C. Algorithm 3 (Method Generate $\mathscr{M}^{u,v}_0$)

Lemma 13. *Algorithm 3 generates sets $\mathscr{M}^{u,v}_0$ in $O(n\theta_0^2)$ time.*

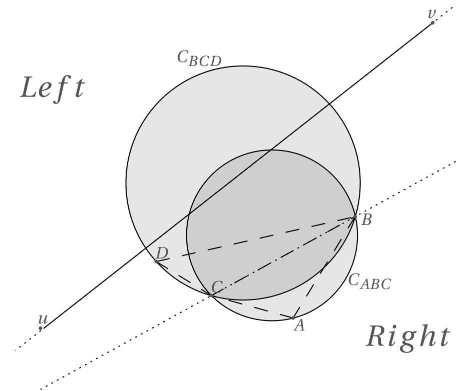

Fig. 3: Illustration for proof of Lemma 12

Algorithm 3: Generaring sets $\mathscr{M}^{u,v}_0$ for all $\{u,v\} \in E_\nabla$

Input: $V, E, E_\nabla, E^i_{u,v} (i \in \{1,2,3\}, \{u,v\} \in E_\nabla)$
Output: Sets $\mathscr{M}^{u,v}_0$ for all $\{u,v\} \in E_\nabla$.
begin
  for $\{u,v\} \in E_\nabla$ **do**
1   **for** $i \in \{1,2\}$ *and* $\{a,b\} \in E^i_{u,v}$ **do**
2       $m^i_{a,b} \leftarrow \max\{m(\widehat{ucv}) | c \in [a,b] \cap P^i\}$
3     $L^1 \leftarrow$SORT(E^1 by $m^1_{a,b}$ values increasingly);
4     $L^2 \leftarrow$SORT(E^2 by $m^2_{a,b}$ values decreasingly);
5     $i \leftarrow 1, j \leftarrow 1$;
6     $\mathscr{M}^{u,v}_0 \leftarrow \emptyset$;
7   **repeat**
8   **while** $m^1_{L^1(i)} + m^2_{L^2(j)} \leq \pi$ *and* $j \leq length(L^2)$ **do**
          $j \leftarrow j+1$
9       $\mathscr{M}^{u,v}_0 \leftarrow \mathscr{M}^{u,v}_0 \cup \{\{L^1(i), \ldots, L^1(n), L^2(1), \ldots, L^2(j-1)\}\} \cup E^3$;
10   **while** $m^1_{L^1(i)} + m^2_{L^2(j)} > \pi$ *and* $i \leq length(L^2)$ **do**
11        $i \leftarrow i+1$
      until $i > length(L^1)$ *or* $j > length(L^2)$;
12 Eliminate the non-maximal elements of $\mathscr{M}^{u,v}_0$;
  return $\mathscr{M}^{u,v}_0$

Proof. Proposition 14 shows the correctness of Algorithm 3.

We assume line 2 runs in constant time. This means that for a given $\{u, v \in E_\nabla\}$ in lines 1 and 2 we get values $m^i_{a,b}$ in $O(\theta_0)$ time, in lines 3 and 4 we sort them in $O(\theta_0 \log \theta_0)$ time, and in lines 7-12 $M^{u,v}_0$ is calculated in $O(\theta_0^2)$ time. Since $|E_\nabla| \leq 3n-6$, this gives an overall complexity $O(n\theta_0^2)$. □

Proposition 14. *It can be shown that if a $k \in C^{u,v}_0$ does not contain $L^1[i-1]$ but contains $L^1[i]$, then $L^1[j]$ is covered by k iff $j \geq i$. Similarly, if k contains $L^2[i-1]$ but does not contain $L^2[i]$, then $L^2[j]$ is covered by k iff $j \leq i-1$. Trivially, E^3 is covered by k, and that is also clear that for any $e_1 \in E^1$ and $e_2 \in E^2$ the edges e_1, e_2 are covered by k iff $m^1_{e_1} + m^2_{e_2} \leq \pi$.* □

Corollary 15. *For every $\{u,v\} \in E_\nabla$, $|\mathscr{M}^{u,v}_0|$ is $O(\theta_0)$.*

Proof. It can be deducted from the description and correctness of Algortithm 3. □

D. Algorithm 4 (Method Eliminateredundants)

Lemma 16. *Algorithm 4 computes $\mathcal{M}_0$ in $O(n\theta_0^3\tau)$ using sets $\mathcal{M}_0^{u,v}$.*

Algorithm 4: ELIMINATEREDUNTANTS

Input: $\mathcal{M}_0^{u,v}$ for all $\{u,v\} \in E_\triangledown$
Output: $\mathcal{M}_0$
begin
  for $\{a,b\} \in E$ **do**
1     $T_{a,b} \leftarrow \emptyset$
  for $t \in T_0$ **do**
    for $\{a,b\} \in E_t$ **do**
2       $T_{a,b} \leftarrow T_{a,b} \cup \{t\}$
  for $\{a,b\} \in E$ **do**
3     $E^d_{a,b} \leftarrow \bigcup_{t_{uvw_\Delta} \in T_{a,b}} \{\{u,v\},\{v,w\},\{w,u\}\}$
  for $\{u,v\} \in E_\triangledown$ **do**
4     $E^D_{u,v} \leftarrow \bigcup_{\{a,b\} \in E^3_{u,v}} E^d_{a,b}$;
5   **for** $f_{u,v} \in \mathcal{M}_0^{u,v}$ **do**
      for $\{w,z\} \in E^D_{u,v}$ **do**
        for $f_{w,z} \in M_0^{w,z}$ **do**
          if $f_{u,v} \supseteq f_{w,z}$ **then**
            $\mathcal{M}_0^{w,z} \leftarrow \mathcal{M}_0^{w,z} \setminus f_{w,z}$
          else
            if $f_{u,v} \subset f_{w,z}$ **then**
6               $\mathcal{M}_0^{u,v} \leftarrow \mathcal{M}_0^{u,v} \setminus f_{u,v}$
7   $\mathcal{M}_0 \leftarrow \bigcup_{\{u,v\} \in E_\triangledown} \mathcal{M}_0^{u,v}$;
8   **return** $\mathcal{M}_0$

Proof. We can prove the correctness of Algorithm 4 by checking that it eliminates all globally non-maximal elements of $\mathcal{M}_0^{u,v}$ and leaves exactly one copy of each element of $\mathcal{M}_0^{u,v}$ in the end.

The proof of complexity is as follows.

For every $\{a,b\} \in E$ let $T_{a,b}$ be the set of Delaunay triangles with circumcircle intersected by $[a,b]$. In lines 1 and 2 we determine the sets $T_{a,b}$. Obviously the complexity of these lines is $O(n\theta_0)$.

Recall that the maximum number of circumcircles of Delaunay triangles crossed by an $e \in E$ with τ_0. By definition in line 3, for all $\{a,b\} \in E$, $E^d_{a,b}$ is a set containing those $\{u,v\}$ edges of the Delaunay triangulation which are covered by a $k \in C_0^{u,v}$. In line 3 we compute these sets. This step has an $O(n\tau_0)$ complexity.

By definition in line 4, $E^D_{u,v} \supseteq \{\{w,z\} | \exists m \in \mathcal{M}_0^{u,v} : m = E_{c_{u,v}} = E_{c_{wz}}, c_{uv} \in C_0^{u,v}, c_{wz} \in C_0^{w,z}\}$, in other words it contains all $\{w,z\}$ edges of the Delaunay triangulation which may have single regional link failures which can have the point pair w,z, and can have u,v on the boundary, because such failures must contain edge from $E^3_{u,v}$. In line 4 this set is calculated in $O(n\tau_0)$ time.

In lines 5 - 7 we get $\mathcal{M}_0$ by comparing at most $O(n\theta_0^2\tau_0)$ failures, and eliminate the redundant and non-maximal elements. Obviously this can be done in $O(n\theta_0^3\tau_0)$ time.

The overall complexity of algorithm 4 is $O(n\theta_0^3\tau)$. □

IV. Conclusions

In this paper we propose a fast and systematic approach to enumerate the list of possible link failures caused by regional failures. Our approach assumes the regional failure has a shape of circle of any size which does not have a node interior. Although the number of possible regional failure is infinite, we show that the generated list of failures is short, it is basically linear to the network size.

We provide a fast polynomial time algorithm for enumerating the corresponding set of Shared Risk Link Groups. According to our knowledge this is the first study providing a comprehensive solution for this problem. As a future work we plan to extend our model to generate regional failures with exactly one, two, etc., nodes interior. Note that, it is a common practice to distinguish failures involving nodes from failures involving links only. It is because if the failure hits a network node, the node is no longer going to send traffic in the network which has a network wide effect.

References

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

[2] 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.

[3] 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.

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

[5] 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.

[6] 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.

[7] 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.

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

[9] 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.

[10] D. M. Masi, E. E. Smith, and M. J. Fischer, "Understanding and mitigating catastrophic disruption and attack," *Sigma Journal*, pp. 16–22, 2010.

[11] J. P. Sterbenz, D. Hutchison, E. K. Çetinkaya, A. Jabbar, J. P. Rohrer, M. Schöller, and P. Smith, "Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines," *Computer Networks*, vol. 54, no. 8, pp. 1245–1265, 2010.

[12] S. Neumayer, A. Efrat, and E. Modiano, "Geographic max-flow and min-cut under a circular disk failure model," *Computer Networks*, vol. 77, pp. 117–127, 2015.

[13] F. Aurenhammer, "Voronoi diagramsÑa survey of a fundamental geometric data structure," *ACM Computing Surveys (CSUR)*, vol. 23, no. 3, pp. 345–405, 1991.

[14] M. De Berg, M. Van Kreveld, M. Overmars, and O. C. Schwarzkopf, *Computational geometry*. Springer, 2000.