# Enumerating Maximal Shared Risk Link Groups of Circular Disk Failures Hitting k Nodes

Balázs Vass, János Tapolcai, Erika Bérczi-Kovács

Abstract-Many recent studies shed light on the vulnerability of networks against large-scale natural disasters. The corresponding network failures, called regional failures, are manifested at failing multiple network elements that are physically close to each other. The recovery mechanisms of current backbone networks protect failures listed as Shared Risk Link Groups (SRLGs). We aim to design an algorithm for the routing engines, which can generate a reasonable list of SRLGs based on the limited geometric information available. As a first step towards this direction, in this paper, we propose a limited geographic information failure model for the network topology that enables efficient algorithms to compute the set of links that are expected to be close to each other. More precisely, we work with (1) relative node positions without knowing the real distances, (2) an area in the map defines the route of each physical cable, and (3) a regional failure is a circular disk with k = 0, 1, ... nodes in its interior. We describe an efficient algorithm for listing SRLGs based on our limited geographic information failure model and show that under realistic assumptions, the obtained list of SRLGs is short, having approximately 1.2n and 2.2n elements for k = 0and k = 1, respectively, where *n* is the number of nodes of the network.

## I. INTRODUCTION

N etworks can efficiently survive a single point of failure (we refer the interested readers to [1], [2]); however, we are still witnessing severe network outages [1]–[11] 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. These type of failures are called *regional failures*, which are simultaneous failures of nodes/links located in at most a few hundred kilometer wide geographic area [1]– [11]. Therefore, this paper aims to increase the service availability for the network connections where the distance between the endpoints is thousands of kilometers; by allocating backup

Balázs Vass and János Tapolcai are with MTA-BME Future Internet Research Group and MTA-BME Information Systems Research Group, Department of Telecommunication and Media Informatics, Faculty of Electrical Engineering and Informatics (VIK), Budapest University of Technology and Economics (BME). Contact them on {balazs.vass,tapolcai}@tmit.bme.hu.

Erika Bérczi-Kovács is is with MTA-ELTE Egerváry Research Group on Combinatorial Optimization, and Department of Operations Research, Eötvös Loránd University, Budapest, Hungary. Contact her on koverika@cs.elte.hu.

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). J.T. and B.V. are partially supported by the National Research, Development and Innovation Fund of Hungary (grant No. 124171, 128062, 134604 and 135606). The research reported in this paper was supported by the BME Artificial Intelligence TKP2020 IE grant of NKFIH Hungary (BME IE-MI-SC TKP2020). Research supported in part by National Research, Development and Innovation Office - Grant No. NKFI-115288.

paths with more significant physical distance, say hundreds of kilometers, from the primary path.

Note that handling the geometric information with the network topology is not part of the network protocols. Furthermore, in many cases, the Internet Service Providers hire the links as a service from an independent company [12], called the Physical Infrastructure Provider, and thus, operators often have no information about the route of the links, or the physical coordinates of the intermediate routing nodes. As a solution, vendor specification of core network equipment suggests ensuring that the primary and backup paths assigned to a connection are simply edge or node disjoint (e.g., Huawei [13, Sec. 4.5.4], Alcatel-Lucent [14, pp. 46-50], Cisco Systems [15, Chpt. 19], Juniper [16, Chpt. 3], Infinera [17]). With nodedisjointness, operators ensure that the distance between the nodes of the primary and backup paths (except at the terminal nodes) are at least 1-hop-distance from each other (Fig. 1). The intuitive reason behind this is that a link in a backbone network is typically a few hundred kilometers long [18], while natural disasters are never larger than a few hundred km. Unfortunately, this is not enough, and networks may suffer serious outages after natural disasters [1]-[11] because of

- · close network nodes, e.g., in highly populated areas, and
- *parallel links*, e.g., links traversing the same bridge over a river, or crossing a mountain range through a valley.

In the routing protocols of backbone networks (e.g., optical mesh, MPLS, and IP) Shared Risk Link Group (SRLG) (also referred to as shared risk resource group, or failure states) is the concept for defining a set of links of which's failure the network should be prepared. Close network nodes and parallel links should have been identified at network planning and listed as SRLGs in the service level agreement (SLA) [23], which is the contract between a service provider and a service user on the quantity, quality, and price of a service. Then, the routers' routing engine would compute SRLG-disjoint primary and backup paths to survive these failures<sup>1</sup>. In other words, the network can survive a failure of an SRLG or a subset of links in an SRLG; 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 means a significant degradation in the observed reliability of the network. At the same time, the

<sup>&</sup>lt;sup>1</sup>The routing algorithms modify the SRLGs, whose failure isolates the source and destination nodes: those SRLGs are replaced with a smaller non-isolating SRLG according to the failure model.



Fig. 1: Strategies for separating the primary and backup paths in increasing strength (the more right the better the separation is). By utilizing only a limited geometric information on the network topology, this paper offers an SRLG list  $M_k$  that ensures single regional link  $k = 0, 1, \ldots$  node failure disjointness. SRLG list  $M_k$  fills the gap between *h*-hop disjointness and *r*-distance disjointness. Arcs are directed from weaker separations towards the stronger ones (according to Apprendix A, protecting regional link 0-node failures ensures node disjointness).

number of SRLGs should be kept low. Since an SRLG is defined as a set of links, we say a node is part of an SRLG if every adjacent edge to the node is part of the SRLG.

Several studies investigated how to collect historical data about the vulnerabilities (e.g., past natural disasters) and combine it with the geometry of the network topology (e.g., the exact GPS coordinates of the nodes and the routes of the cable conduits) to list SRLGs (or vulnerability regions). They are either focusing on the geometric approaches [3], [20], [21], [24]–[28] or on past natural disasters [5], [7]–[11], [29]. Note that the RFCs [30], [31] on automatically collecting SRLGs are for higher (MPLS) layer. Overall, we see the lack of information on the topologies is the reason why networks are still not properly prepared to survive natural disasters.

This paper aims to provide an alternative approach where the SRLGs are guessed based on the available topology information. Roughly speaking, the SRLGs are derived by deeply analyzing the topology graph and investigating whether a set of links could be physically close. We borrow the idea of Delaunay graphs to classify "close nodes": a node-pair i, jis close if the circular disk through i and j that has no nodes interior. For "parallel links", inspired by the Voronoi diagrams [32], a set of links S is classified as close if there is a point in the plane from which the links of S are the nearest network elements, and thus there could be a disaster that will destroy all the links of S while the rest of the network remains operating.

Inspired by the above, in Subsec. II-A, we define a *limited* geometric information failure model to estimate the links that are potentially close to each other based on the logical topology and a little information a router may know about the network geometry. The model represents the outcome of a possible topology discovery software module running in the routing engines, which embedded the nodes and links into a schematic map according to the measured propagation delays. As also depicted in Fig. 1, protecting single regional link k-node failures (i.e., failures caused by circular disk shaped disasters that hit k network nodes) is a good compromise for regional failure protection if the exact embedding of the network is not known, as 1) these failures cover the "close-node" and "parallel link" type failures, 2) using pure graph-theoretic approaches, close-node and parallel-link failures could not be protected, and 3) approaches based on abundant input data on the network geometry or the disasters are not applicable due

to the lack of information.

The main contribution of the paper is providing a failure model and a related SRLG enumerating algorithm that covers the most important possible regional failure states of the network based on the little geometric information on the topology available.

More concretely, we show that all possible failures caused by a circular disk with at most k nodes interior can be represented by O((k+1)|V|) SRLGs in a typical backbone network topology (see Coroll. 25), and might be at most  $O((k+1)|\mathcal{E}||V|)$  in an artificial worst-case scenario (shown in Appendix I), where |V| denotes the number of nodes in the network and  $|\mathcal{E}|$  the number of links. We propose a systematic approach based on computational geometric tools that can generate this list of SRLGs in  $O(|V|^2(k^5 + k\log(|V|k+1) + 1))$ steps on typical networks. Besides the excellent asymptotic behavior, the obtained SRLG lists are short enough for network operators to prepare their networks to be resilient to such failure situations. We believe backbone networks operated to survive SRLG failures according to our new failure model should have significantly higher reliability.

In our previous studies [33] and [34] regional disasters hitting exactly k = 0 and k = 1 nodes have been investigated. The lists proposed for protecting these failures have a length linear in the number of network nodes and can be calculated in  $O(|V|\log|V|)$  and  $O(|V^2|)$  for k = 0 and k = 1, respectively, on typical networks. Our paper [35] provides a simpler algorithm for arbitrary k with time complexity  $O((k^2 + 1)|V^3|)$  in case of stronger assumptions than present paper's. Note that this is the first study containing simulation results on the proposed SRLG lists.

On the technical side, we note that however, our study heavily relies on former computational geometric results on structure k-Delaunay [36], or on its dual, the k-Voronoi diagram [32], as we are interested in the maximal edge sets hit by disks hitting k nodes, our results are not trivial consequences of the previous knowledge on neither the k-Voronoi, nor the kedge-Voronoi diagram [37]. To the best of our knowledge, no previous study addressed the problem presented in the paper ( [38] and [39] deal with somewhat loosely related topics).

The rest of the paper is organized as follows. In Sec. II our model will be presented. In Sec. III a polynomial algorithm for computing the SRLG list is introduced, and both nonparametrized and parametrized upper bounds on the running time of the algorithm are given. Next, simulation results are presented in Sec. IV. We also discuss some open theoretical problems in Sec. V, and finally, Sec. VI concludes our paper.

# II. PROBLEM FORMULATION AND ASSUMPTIONS

#### A. The Limited Geometric Information Failure Model

We have the following design goals in defining the limited geometric information failure model.

- (a) Do not underestimate the set of links involved in a possible regional failure. We believe the operator's damage in case of an unprotected regional failure is much higher than the extra cost of protecting networks against larger SRLGs.
- (b) *Relative link distances* are given, the exact route of the cables are unknown, and the nodes are embedded in a schematic map.
- (c) Provide a *fast and space efficient* way of calculating the set of SRLGs.

According to our first design goal, we deal with circular disk failures and define the size of the regional failure through the number of nodes it covers. Although the regional failures can have any location, size, and shape, without any background information on the regional failures, it is a common practice to overestimate the size of the regional failure by ignoring its shape and rather focus on its radius only [26], [40]<sup>2</sup>. According to our second design goal, the scaling of the topology map is not known, thus we cannot define a fixed maximum radius for the regional failures, but instead, we define a limit on the number of nodes interior to the circular disk.

Now we can define the limited geometric information failure model, which is based on the following assumptions:

- (1) The network is a graph  $\mathcal{G}(V, \mathcal{E})$  embedded in a 2D plane.
- (2) The exact route of the conduits of the network links are not known, but contained by a known polygonal region.
- (3) The shape of the regional failure is assumed to be a circular disk with arbitrary radius and center position.
- (4) We focus on *regional link k-node failures*, failures that hit k nodes for k ∈ {0, |V| 2}.

Note that, in our experiments with practical network topologies protecting against single regional link 0-node failures resulted in at least 2-hop-distance between the nodes of the primary and backup paths, except the terminal nodes. We believe the proposed approach well captures the possible regional network failures based on the little geographic information available at network devices.

#### B. Formal Graph and Disaster Models

The network is modeled as an undirected connected geometric graph  $\mathscr{G} = (V, \mathscr{E})$  with  $n = |V| \ge 3$  nodes and  $m = |\mathscr{E}|$ edges<sup>3</sup>. The nodes of the graph are embedded as points in the Euclidean plane, and their exact coordinates are considered to be known. In contrast to this, precise positions of edges are not known, instead we assume that for each edge *e* there is a *containing polygon* (or simply *polygon*)  $e^p$  in the plane in which the edge lies (see Fig. 2a). Parameter  $\gamma$  will be used to indicate the maximum number of sides a containing polygon  $e^p$  can have. Note that this model covers special cases when edges are considered as polygonal chains or line segments (thus at first reading, for ease, the reader may consider the edges as line segments and  $\gamma$  equals two).

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 on the same circle.<sup>4</sup>

We will often refer to circular disks simply as disks. The disk failure model will be adapted, which overestimates the area of a disaster such that all network elements that intersect the interior of a circle c are failed, and all other network elements are untouched. It is important to note that this modeling technique does not assume that the failed region has a shape of a disk, but overestimates the size of the failed region to have a tractable problem space.

**Definition 1.** A circular disk failure c hits an edge e if the polygon of the edge  $e^p$  intersects the interior of disk c. Similarly, node v is hit by failure c if it is in the interior of c. Let  $\mathcal{E}_c$  (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 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 beside listing failed edges has no additional information. We consider SRLGs that represent worst-case scenarios the network must be prepared for; thus, there is no SRLG which is a subset of another SRLG.

**Definition 2.** Let  $\mathscr{C}$  denote the set of all circular disks in the plane, and let  $\mathscr{C}_k \subseteq \mathscr{C}$  denote the set of those hitting exactly k nodes from V.

Now we can define the set of failure states the network may face after a disk failure hitting exactly k nodes.

**Definition 3.** Let set  $F(\mathcal{C}_k)$  denote the set of edge sets which can be hit by a disk  $c \in \mathcal{C}_k$ , and let  $M_k = M(\mathcal{C}_k)$  denote the set of maximal edge sets in  $F(\mathcal{C}_k)$ .

Note that for every  $l \in \{0, ..., k-1\}$  and  $f \in M_l$  there is an  $f' \in M_{l+1}$  such that  $f \subseteq f'$ , because any disk hitting  $l \le k$  nodes could be overestimated by a disk hitting k nodes.

As mentioned before, only the maximal edge sets will be listed as SRLGs. This study aims to offer fast algorithms

<sup>&</sup>lt;sup>2</sup>Clearly, extreme overestimation of the failed link set should be avoided. <sup>3</sup>Graph  $\mathcal{G} = (V, \mathcal{E})$  is not necessarily planar.

<sup>&</sup>lt;sup>4</sup>All of the results of the paper could be extended to geometric objects in non-general position, however, this would complicate our arguments lowering the paper's readability, while by an insignificant perturbation of the data one can make sure that the geometric objects are in general position.



Fig. 2: Input topology  $\mathscr{G}(V,\mathscr{E})$  and k-Delaunay graphs  $D_k(V, E_k)$  for k = 0, 1, 2 on the same node set V. Polygons filled with gray represent the containing polygons of network edges. Line segments between nodes represent Delaunay edges. Edges of  $E_k \setminus E_{k-1}$  are depicted in red on Subfig. 2c and 2d, respectively, while the rest of the edges are in blue.

computing this list for various values of k, more precisely, throughout the paper, we will assume  $k \in \{0, n-2\}$  since if a failure hits n-1 nodes, there is no node pair to communicate.

#### **III.** Algorithm for Enumerating Maximal Failures

In this section, a polynomial-time algorithm is presented for computing  $M_k$ . The basic idea is that determining  $M_k$ can be decomposed into several simpler tasks, as illustrated in Fig. 3. Informally, after determining the so-called k-Delaunay graph (Def. 9 in III-B), data structures *apple* (Subsec. III-C) and *seesaw* (Appendix F) are computed, and finally  $M_k$  is determined by merging lists  $M_k^{u,v}$  and  $M_k^w$  resulting from querying the apples and seesaws.

#### A. Basic Observations

Our first observation is the following.

**Claim 4.** For every  $f \in M_k$   $(k \le n-2)$  there exists a disk  $c \in \mathcal{C}_k$  such that f is hit by c, and c has at least one node of V on its boundary.

*Proof:* Let f be hit by a disk  $c_0 \in \mathcal{C}_k$  with center point p. Since there are nodes of V not inside  $c_0$ ,  $c_0$  can be magnified from p until its boundary reaches a node u from V. This disk  $c_1$  is also from  $\mathcal{C}_k$  and has at least one point on its boundary, still hitting f.

Disk  $c_1$  described in the proof can be further magnified while keeping its center point on ray [up. Here we consider two cases: either there exists a node  $v \in V$ , which gets on the boundary while magnifying disk  $c_1 \in \mathcal{C}_k$ , or the open halfplane *h* having *p* inside, *u* on the boundary, and having the normal vector  $\vec{up}$  hits *k* nodes.

**Definition 5.** Let  $M_k^2$  be the set of maximal failures that can be hit by a disk from  $\mathscr{C}_k$  nodes having 2 nodes on its boundary. Let  $M_k^1$  be the set of maximal failures that can be caused by a half-plane having a node on its boundary hitting exactly k nodes.

# **Proposition 6.** $M_k$ is the set maximal sets in $M_k^1 \cup M_k^2$ .

In the following, we will present a way of computing  $M_k^2$  in detail using a data structure called *apple* defined in the present paper. Determining  $M_k^1$  can be done using similar ideas, thus



Fig. 3: Visual sketch of Algorithm 3 for determining  $M_k$ 

we present it only briefly in Appendix F-G, where the same data structure called *seesaw* is defined.

# B. Graphs k-Delaunay

**Definition 7.** For a node pair  $u, v \in V$  let  $\mathscr{C}_k^{u,v}$  denote the set of disks from  $\mathscr{C}_k$  having nodes u, v on their boundary. Let  $\mathscr{C}^{u,v}$  denote the set of disks from  $\mathscr{C}$  having nodes u, v on their boundary. Let  $M_k^{u,v} = M(\mathscr{C}_k^{u,v})$  be the the set of failures which contain exactly the elements of  $M_k$  that can be hit by a disk  $c \in \mathscr{C}_k^{u,v}$ .

Discussion after Claim 4 suggests the following simple method to compute  $M_k^2$ . First, for every node pair  $\{u, v\} \subset V$ , we compute a set of failures  $M_k^{u,v}$ .

**Definition 8.** Let  $E_k$  denote the set of node-pairs  $\{u, v\} \subset V$  for which  $\mathscr{C}_k^{u,v} \neq \emptyset$ .

Fig. 2 shows an example of the input topology  $\mathscr{G}$  and the corresponding set of node-pairs  $E_k$  for k = 0, 1, 2.

We can observe that by definition,  $M_k^2$  can be computed by merging these sets  $M_k^{u,v}$ , formally  $M_k^2$  is the set of maximal elements from the union of sets  $M_k^{u,v}$ .

Our second observation is that  $E_k$  is the edge set of the so-called *k*-Delaunay graph [41].

**Definition 9.** Let  $D_k = (V, E_k)$  denote the k-Delaunay graph induced by node set V and edge set  $E_k$ .<sup>5</sup>

The k-Delaunay graph  $D_k$  is a so-called *geometric proximity* graph (other examples are the Voronoi diagram or the Gabriel

<sup>&</sup>lt;sup>5</sup>We note that  $E_k$  is different from the set  $\mathscr{E}$  of network links.

TABLE I:	Table of	f symbols
----------	----------	-----------

Notation	ation Meaning					
	General					
$\mathscr{G}(V,\mathscr{E})$	the network modeled as an undirected connected geometric graph					
<i>n</i> , <i>m</i>	the number of nodes $ V  \ge 3$ and edges $ \mathcal{E} $ , respectively					
$e^p$	$e^p$ the containing polygon of edge <i>e</i> (see Fig. 2a)					
$V_c, \mathscr{E}_c$	set of nodes and edges, resp., hit by a disk c					
C	the set of all circular disks in the plane					
$\mathscr{C}_k$	the set of circular disks in the plane hitting exactly $k$ nodes					
$F(\mathscr{C}_k)$	the set of edge sets which can be hit by a disk $c \in \mathcal{C}_k$					
$M_k$	the set of maximal edge sets in $F(\mathscr{C}_k)$					
the set of maximal failures which can be hit by a disk from c						
l <sup>IVI</sup> k	having 2 nodes on its boundary					
the set of maximal failures which can be caused by a half-p						
M <sub>k</sub>	having a node on its boundary hitting exactly k nodes					
C <sup>u,v</sup>	the set of disks from $\mathscr{L}_{L}$ having nodes $\mu, \nu$ on their boundary					
- <sup>k</sup> <sup>k</sup> u,v	the set of disks from $\mathscr{C}_k$ having nodes $u, v$ on their boundary					
6	the set of disks from 6 having nodes $u, v$ of their boundary					
$M_1^{u,v}$	the the set of families which contain exactly the elements of $M_k$					
ĸ	that can be not by a disk $c \in \mathcal{C}_k$					
· · ·	Parameter					
k	we are interested in circular disk shaped disasters hitting $k$ nodes					
γ	the maximum number of sides a containing polygon $e^{\rho}$ can have					
$\rho_k$	the maximum number of edges hit by a disk hitting $k$ nodes.					
	Apple					
$E_k$	the set of node-pairs $\{u, v\} \subset V$ for which $\mathscr{C}_k^{u,v} \neq \emptyset$					
$D_k(V, E_k)$	the k-Delaunay graph induced by node set V and edge set $E_k$					
	For $u, v \in V$ , a Cartesian coordinate system is placed in the plane					
	such that line $uv$ be identical to the vertical axis y, u and v have					
c(x),	ordinates (y coordinates) 1 and $-1$ , respectively (see Fig. 4).					
$h_+,$	In this coordinate system, $h_{\pm}$ and $h_{\pm}$ are the right and left open					
$h_{-}$ half plane determined by line $\mu\nu$ respectively and $c(r) = c(r)$						
	denotes the unique disk c in $\mathcal{C}^{u,v}$ , which has centre point					
	$(x, 0)$ , and $c(+\infty) := h_{+}$ and $c(-\infty) := h_{-}$ , respectively.					
$I^{u,v}$ the set of those numbers x for which $c(x) \in \mathcal{G}^{u,v}$						
$\frac{1}{k}$ the set of mose numbers x, for which $c(x) \in \mathcal{O}_k$						
xmax	the maximum of $T_k$ , if exists, else if $T_k \neq \emptyset$ , $x_{max} = +\infty$					
x <sub>min</sub>	the minimum of $I_k^{u,v}$ , if exists, else if $I_k^{u,v} \neq \emptyset$ , $x_{min} = -\infty$					
$\mathcal{E}_{c(x_{max})},$	the edge sets hit has $a(n \to n)$ and $a(n \to n)$ representiable.					
$\mathcal{E}_{c(x_{min})}$	the edge sets fit by $c(x_{max})$ and $c(x_{min})$ , respectively					
D	for $u, v \in V$ , the right side of $c(x_{max})$ cut by the vertical line $uv$					
d	for $u, v \in V$ , the left side of $c(x_{min})$ cut by the vertical line $uv$					
$x_+(e)$	for edge e, the leftmost disk which hits $e^p \cap D$					
$x_{-}(e)$	for edge e, the rightmost disk which hits $e^p \cap ($					
	For nodes $w_+ \in h_+$ and $w \in h$ , let $x_+(w_+)$ and $x(w)$					
$x_{+}(v),$	denote the abcissa of the centre point of circle going through					
$x_{-}(v)$	$u, v$ and $w_{\perp}$ or $w_{\perp}$ , respectively.					
E+	the list of edges hit by $D$ ordered descending by the $x_{\perp}$ values					
8_	the list of edges hit by () ordered descending by the $x_{-}$ values					
 V+	the list of nodes hit by $D$ ordered descending by the $x_{\pm}$ values					
	the list of nodes hit by $\bigcirc$ ordered descending by the $x_{-}$ values					
	the list of nodes $w$ in $h_{+}$ ordered decreasingly by the abscissa					
$V'_+$	$x_{+}(z)$ of their leftmost hitting circles (going through $u, v, z$ )					
the list of nodes $z$ in $h$ ordered decreasingly by the abso						
V	$r(z)$ of their rightmost hitting disk (going through $\mu \mu z$ )					
	For an edge $(u, v) \in E_v$ apple $A^{U,V}$ is an ordered system					
AU.V	For an edge $\{u, v\} \in E_k$ , apple $A_k$ is an ordered system					
$A_k^{A,a,a}$	$A_k = (V_+, V\mathcal{E}_+, \mathcal{E})$ . For each element of each list its					
	appropriate $x_+()$ or $x()$ value is also stored.					
Ak	$A_k$ the set of apples $A_k^{u,v}$					
L	Seesaw (only in Appendix F)					
For data s	tructure Seesaw for determining $M^1$ please check Appendix F					
	, preuse encentrippendix 1.					

graph [42]). In that paper, the *k*-Delaunay graph is determined as an intermediate step from the *k*-Voronoi diagram while determining the *k*-Gabriel graph in polynomial time following [42]. An upper bound on  $|E_k|$  is derived in [36]. These statements are the following:

**Theorem 1** (Thm. 2.4 of [42]). Graph k-Delaunay  $D_k = (V, E_k)$  can be constructed in  $O((k^2 + 1) n \log n)$ .

**Theorem 2** (Thm. 2 of [36]).  $|E_k| \le 3(k+1)n - 3(k+1)(k+2)$ .

These theorems give that for small values of k graph  $D_k$ is sparse (in other words,  $\mathscr{C}_k^{u,v} = \emptyset$  for most node pairs u, v), and it can be computed fast.

#### C. Data Structure Apple

Let node pair  $\{u, v\} \in E_k$  be given. Let us place a Cartesian coordinate system in the plane such that line uv be identical to the vertical axis y, u and v have ordinates (y coordinates) 1 and -1, respectively (see Fig. 4). Obviously, this way the center point of any disk  $c \in \mathscr{C}_k^{u,v}$  has ordinate 0.

**Definition 10.** For a given node pair  $\{u, v\}$ , the previously described coordinate system and real number x, let c(x) denote the unique disk c in  $\mathcal{C}^{u,v}$ , which has center point (x, 0).

Trivially, c() is a bijective function between  $\mathbb{R}$  and  $\mathscr{C}^{u,v}$ .

Let  $I_k^{u,v}$  denote the set of those numbers x, for which  $c(x) \in \mathcal{C}_k^{u,v}$ . If  $\mathcal{C}_k^{u,v}$  is empty, then trivially  $I_k^{u,v}$  is empty too. In the case when  $\mathcal{C}_k^{u,v}$  is not empty, we can observe that  $I_k^{u,v}$  is the union of closed intervals.

If the number of nodes in both half planes determined by line uv is not equal to k, then there exists a rightmost and a leftmost element of  $\mathscr{C}_k^{u,v}$ , i.e. there is a maximum  $x_{max}$  and minimum  $x_{min}$  real number in  $I_k^{u,v}$ , that is  $x_{max} = \max I_k^{u,v}$ , and  $x_{min} = \min I_k^{u,v}$ . Disks  $c(x_{max})$  and  $c(x_{min})$  have a third node  $w_+$  and  $w_-$  on their boundary, respectively. If there are exactly k nodes on the right side of uv, then let  $h_+$ be the right open half-plane determined by line uv, and let  $x_{max}$  be sufficiently large to  $\mathscr{E}_{c(x_{max})}$  to contain all the edges having polygon having a point with positive abscissa. For simplicity, sometimes  $x_{max} = +\infty$  and  $c(x_{max}) = h_+$  is used. Same applies for the left side of uv.

Let  $\mathscr{E}_{c(x_{max})}$  and  $\mathscr{E}_{c(x_{min})}$  denote the edge sets hit by  $c(x_{max})$ and  $c(x_{min})$ , respectively. To compute  $M_k^{u,v}$  we use the following observation.

# **Claim 11.** For all $f \in F(\mathscr{C}_k^{u,v})$ , $f \subseteq \mathscr{E}_{c(x_{max})} \cup \mathscr{E}_{c(x_{min})}$ .

*Proof:* It is easy to see that for every disk  $c \in \mathscr{C}_k^{u,v}$ ,  $c \subseteq c(x_{max}) \cup c(x_{min})$ .

According to Claim 11, a first step towards computing  $M_k^{u,v}$  is to determine the edge sets hit by  $c(x_{max})$  and  $c(x_{min})$ . Trivially, this can be done in  $O(m\gamma)$ . The remaining question is how to calculate  $M_k^{u,v}$  from  $\mathscr{E}_{c(x_{max})} \cup \mathscr{E}_{c(x_{min})}$ . Some additional notations and definitions precede the solution.

Let D denote the right side of disk  $c(x_{max})$  cut by the vertical line uv, and let G denote the left side of disk  $c(x_{min})$  cut by the vertical line uv. For each edge  $e \in \mathscr{E}_{c(x_{max})} \cup \mathscr{E}_{c(x_{min})}$  we will compute two disks: the leftmost disk which hits  $e^p \cap D$  and the rightmost disk which hits  $e^p \cap G$ , which have center points  $x_+(e)$  and  $x_-(e)$  respectively.

Let  $\mathscr{E}_+$  denote the list of edges hit by  $\square$ , and similarly, let  $\mathscr{E}_-$  be the list of edges hit by  $\square$ . Thus, we have  $\mathscr{E}_+ \subseteq \mathscr{E}_{c(x_{max})}$  and  $\mathscr{E}_- \subseteq \mathscr{E}_{c(x_{min})}$ , and also  $\mathscr{E}_+ \cup \mathscr{E}_- = \mathscr{E}_{c(x_{max})} \cup \mathscr{E}_{c(x_{min})}$ .

Let  $\mathscr{E}_+$  and  $\mathscr{E}_-$  be ordered descending by the  $x_+$  and  $x_-$  values of their elements, respectively.

Note that according to Claim 39 from the Appendix, both  $x_+(e)$  and  $x_-(e)$  can be computed in  $O(\gamma)$ .

For nodes  $w_+ \in h_+$  and  $w_- \in h_-$ , let  $x_+(w_+)$  and  $x_-(w_-)$  denote the abscissa of the center point of circle going through



Fig. 4: Illustration of an apple with k = 2. Apple  $A_k^{u,v}$  consists of ordered lists of nodes  $V_+$  and  $V_-$  and ordered lists of edges  $\mathcal{E}_+$  and  $\mathcal{E}_-$ , where  $V_+ = \{n_4, n_3\}, V_- = \{n_2, n_1\}, \mathcal{E}_+ = \{e_4, e_3, e_2\}$  and  $\mathcal{E}_- = \{e_3, e_2, e_1\}$ . Given  $\mathcal{G} = (V, \mathcal{E})$  and  $\{u, v\} \subseteq V, A_k^{u,v}$  can be determined in  $O(n\rho_0\gamma + k\log k + \rho_k \log \rho_k)$  (proof of Lemma 30). By querying  $A_k^{u,v}, M_k^{u,v}$  can be computed in  $O(\rho_k^{\alpha})$  (proof of Lemma 32).

u, v and  $w_+$  or  $w_-$ , respectively. We introduce  $V_+$  and  $V_$ similarly to  $\mathscr{E}_+$  and  $\mathscr{E}_-$ , but instead of edges we store nodes hit by  $\square$  ordered descending by their  $x_+$  values, while in  $V_$ nodes in  $\square$  are stored ordered also descending, but by their  $x_-$  values. Trivially, for a node  $v \in V$  both  $x_+(v)$  and  $x_-(v)$ can be determined in O(1).

Note that while every node  $v \in V$  is part of at most one of lists  $V_+$  and  $V_-$ , edges can be part of both  $\mathcal{E}_+$  and  $\mathcal{E}_-$ .

Now we can define the data structure *apple* for each edge of the *k*-Delaunay graph.

**Definition 12.** For an edge  $\{u, v\} \in E_k$ , apple  $A_k^{u,v}$  is an ordered system  $A_k^{u,v} = (V_+, V_- \mathcal{E}_+, \mathcal{E}_-)$ , where its composing lists are as described in the subsection before. For each element l of each list we also store its appropriate  $x_+(l)$  or  $x_-(l)$  value.

# D. Concept of Sweep Disk Algorithms

1) Concept: In this subsection, we highlight the paradigm of sweep disk algorithms, which is similar to the algorithmic paradigm of sweep line (sweep surface) algorithms in computational geometry.

In the case of sweep line algorithms, it is imagined that a line is moved across the plane, keeping its orientation and stopping at some event points. Geometric operations are restricted to the immediate vicinity of the sweep line whenever it ends, and the complete solution is available once the line has passed over all objects. For example, Fortune's algorithm for computing the Voronoi diagram of a point set is a sweep line algorithm [32].

Our sweep disk algorithms will scan through disk sets  $\mathscr{C}^{u,v}$ . In this sense, in contrast to the sweep surface paradigm, our disks have different diameters, and instead of keeping orientation, the invariant will be that all disks have u and v on the boundary. Thus our disk to sweep is "elastic," in the sense that it can change its diameter, but not its shape.

2) *Example:* Our first sweep disk algorithm is used for determining  $x_{max}$  and  $x_{min}$  for a given  $A_k^{u,v}$ . The algorithm works as follows. Starting from a disk  $c(x) \in \mathcal{C}^{u,v}$  having

**Algorithm 1:** Determining  $x_{max}$  and  $x_{min}$  while sweeping through  $\mathcal{C}^{u,v}$ 

<b>Input:</b> V and $u, v \in V$ ; <b>Output:</b> $x_{max}$ and $x_{min}$ begin				
1	Compute ordered lists $V'_{\pm}$ and $V'_{\pm}$			
2	Merge $V'_+$ and $V'$ into descending ordered list $V'_+$ using values			
	$x_+$ for $V_+$ and $x$ for $V$			
3	$n_+, n \leftarrow 0$			
4	for $l \in \{1,,  V'_+ \}$ do			
5	if $V'_+[l] \in \vec{V}'_+$ then $n_+ + = 1$			
	else $n_{+}=1$			
6	$\#_l :=  V'_+  - n_+ + n / / \#$ currently hit nodes			
7	if $ V'_{+}  = k$ then $z_{+} = v_{\phi}$ ; $x_{max} = +\infty$			
	else $w_+ := V'_+[\min l: \#_l = k]; x_{max} := x_+(w_+)$			
8	if $ V'_{-}  = k$ then $z_{-} = v_{\phi}$ ; $x_{min} = -\infty$			
	else $w_{-} := V'_{+} [\max l : \#_{l} = k]; x_{min} := x_{-} (w_{-})$			
9	<b>return</b> $x_{max}$ and $x_{min}$			

center point with abscissa  $x = +\infty$  (or sufficiently large), *c* is swept throughout the elements of  $\mathscr{C}^{u,v}$  until  $x = -\infty$  (or sufficiently small). Meanwhile, the number of nodes hit is followed. Numbers  $x_{max}$  and  $x_{min}$  can be determined at the first and last state when *c* hits exactly *k* nodes, respectively. (Non-existence of such moments would mean that  $\{u, v\} \notin E_k$ .)

Technically this can be done as follows. Let  $V'_+ \subseteq V$  be the list of nodes w from  $h_+$  ordered decreasingly by the abscissa  $x_+(z)$  of their leftmost hitting circles (going through u, v, z). Similarly, let  $V'_-$  be the list of nodes z in  $h_-$  ordered decreasingly by the abscissa  $x_-(z)$  of their rightmost hitting disk (going through u, v, z). Applying the fact that a node pair  $z_+ \in V'_+$  and  $z_- \in V'_-$  can be hit by the same disk  $c \in \mathscr{C}^{u,v}$  iff  $x_+(z_+) \ge x_-(z_-)$ , sweeping can be imitated as in Algorithm 1. Note that for every node z in  $V'_+$  or  $V'_-$ ,  $x_+(z)$  or  $x_-(z)$  is stored as part of function  $x_+$  or  $x_-$ .

From the following Proposition 13, one can check that the number of hit nodes can be easily followed with the help of an additional variable.

**Proposition 13.** Let  $c \in \mathcal{C}^{u,v}$ . If  $V'_+[i-1]$  is not hit by c, then all the preceding elements in  $V'_+$  are not hit by c. If  $V'_+[i]$  is hit by c, then all the following elements in  $V'_+$  are hit by c.

Similarly, if  $V'_{-}[i-1]$  is hit by c, then all the preceding elements are hit by c. If  $V'_{-}[i]$  is not hit by c, then all the following elements are not hit by c.

**Claim 14.** For a given edge  $\{u, v\} \in E_k$ , both  $x_{max}$  and  $x_{min}$  can be determined in  $O(n \log n)$  time.

*Proof:* According to those written in this subsection, both  $V'_+$  and  $V'_-$  can be determined in  $O(n \log n)$  the dominant step being a sorting algorithm. Sweeping can be trivially done in O(n); meanwhile, both  $x_{max}$  and  $x_{min}$  can be determined.

**Proposition 15.** Both  $V_+$  and  $V_-$  can be determined in  $O(n \log n)$ .

#### E. Determining Apples

**Claim 16.** For a given  $\{u, v\} \in E_k$ , apple  $A_k^{u,v}$  can be determined in  $O(m(\log m + \gamma))$ .

**Proof:** If  $\{u, v\} \in E_k$ , then  $x_{max}$ ,  $x_{min}$ ,  $V_+$  and  $V_-$  can be determined in  $O(n \log n)$  according to Claim 14 and Prop. 15, thus it remains to determine  $\mathcal{E}_+$  and  $\mathcal{E}_-$ . With this aim it is enough to compute the  $x_+(e)$  and  $x_-(e)$  values for every edge, then collect in  $\mathcal{E}_+$  those edges e for which  $x_+(e) \le x_{max}$  and similarly in  $\mathcal{E}_-$  those edges e for which  $x_-(e) \ge x_{min}$ . Finally, edges in  $\mathcal{E}_+$  and  $\mathcal{E}_-$  have to be sorted descending according to their  $x_+$  and  $x_-$  values, respectively. m polygons of edges (each having at most  $\gamma$  sides) have to be checked and sorted which gives a total complexity of  $O(m(\gamma + \log m))$ .

**Definition 17.** Let  $A_k$  be the set of apples  $A_k^{u,v}$ .

**Corollary 18.** For a given k, knowing  $E_k$ , the set of apples  $A_k$  can be determined in  $O((k+1)nm(\gamma + \log m))$ .

*Proof:* Since by Thm. 2  $|E_k| < 3(k+1)n$ , we deduct that O((k+1)n) apples have to be determined. According to Claim 14, an apple can be constructed in  $O(m(\log m + \gamma))$ , which completes the proof.

F. Computing the Set of SRLGs by Sweeping Through Each Apple

**Claim 19.** Let  $e \in \mathcal{E}_+$ , and  $f \in \mathcal{E}_-$ . They can be hit by the same  $c \in \mathcal{C}_k^{u,v}$  if  $x_+(e) \le x_-(f)$  or  $x_+(f) \le x_-(e)$ .

*Proof:* An edge *e* can be hit by circle c(x) if  $x_+(e) \le x$  or  $x \le x_-(e)$ .

Determining  $M_k^{u,v}$  from apple  $A_k^{u,v}$  can be done with the help of a sweep disk algorithm as a subroutine of Algorithm 2 similar to Algorithm 1, the only difference is that here we have to check both the set of currently hit edges and the number of currently hit nodes at the same time.

On the one hand, while sweeping through  $\mathscr{C}^{u,v}$  with c(x) (while x decreasing), nodes are also getting hit or not hit by the actual c(x), thus it is not necessarily permanently part of  $\mathscr{C}_k^{u,v}$  during the sweep disk algorithm. On the other hand, any edge *e* having  $e^p$  intersecting segment [u, v] or for which  $x_+(e) \le x_-(e)$  should be stored exactly once in any element of  $M_k^{u,v}$ .

# **Claim 20.** Querying $A_k^{u,v}$ , Alg. 2 calculates $M_k^{u,v}$ in $O(m^3)$ .

**Proof:** The correctness of the algorithm can be easily checked. Since while sweeping an edge can get hit or unhit at most once on one side of line uv, there are at most O(m) failures with locally maximal cardinalities, each of them having O(m) edges, thus  $(M_k^{u,v})'$  has O(m) elements of O(m) size. Trivially, the number of currently hitting nodes can be monitored in O(n) total time as in Alg. 1. Since we have an ordering of the edges, every pair of sets from  $(M_k^{u,v})'$  can be compared in O(m). This implies that from  $(M_k^{u,v})'$ ,  $M_k^{u,v}$  can be determined in  $O(m^3)$ . It can be shown that all the other operations have complexity at most  $O(m^3)$ .

**Corollary 21.** Known  $E_k$ , lists  $M_k^{u,v}$  for all  $\{u, v\} \in E_k$ , can be determined in  $O((k+1)nm^3)$ .

Here we use the assumption that the corner points of the polygons of the edges are in general position.

A	Algorithm 2: Querying an apple				
I	<b>Input:</b> Apple $A_k^{u,v}$ ; <b>Output:</b> Set $M_k^{u,v}$ of locally maximal				
	failures. begin				
1	Merge $V_+$ , $V$ , $\mathcal{E}_+$ and $\mathcal{E}_+$ into descending ordered list G using				
	values $x_+$ for $V_+$ and $\mathcal{E}_+$ and $x$ for $V$ and $\mathcal{E}$ ;				
2	$n_+, n, e_+, e \leftarrow 0;$				
3	for $l \in \{1,,  G \}$ do				
4	if $l \in V_+$ then $n_+ + = 1$ ;				
5	if $l \in V$ then $n+=1$ ;				
6	if $l \in \mathcal{E}_+$ then $e_+ += 1$ ;				
7	if $l \in \mathcal{E}$ then $e += 1$ ;				
8	$\#_{n,l} :=  V_+  - n_+ + n / / \#$ curr. hit nodes				
9	$\#_{e,l} :=  \mathcal{E}_+  - e_+ + e / / \#$ curr. hit edges				
10	$e_{+,l} := e_+ / / \mathscr{E}_+[i]$ is hit iff $i \ge e_{+,l}$				
11	$e_{-,l} := e_{-} / / \mathscr{E}_{-}[i]$ is hit iff $i \leq e_{-,l}$				
12	Det. L, the set of indexes l, for which $\#_{n,l} = k$ ;				
13	Det. $I_e$ , the sequence of numbers $\#_{e,l}: l \in L$ ;				
14	Det. $M_e$ , the set of indexes l of local maximums of $I_e$ ;				
15	$(M_{l_e}^{u,v})' :=$ hit edge sets in $l \in M_e$ disk positions;				
	// Can be det. using $\mathcal{E}_+, \mathcal{E}, \mathcal{e}_+, \mathcal{E}, \mathcal{e}$				
16	$M_{L}^{u,v} \leftarrow$ maximal elements of $(M_{L}^{u,v})'$ ;				
17	return $M^{u,v}$				

## G. Algorithm for Computing Maximal Failures

As presented before  $M_k^2$  can be calculated by determining and querying the apples, and finally merging the obtained lists  $M_k^{u,v}$  in  $M_k^2$ . Note that there is a valid need of comparing lists of locally maximal failures (see Fig. 5).  $M_k^1$  can be computed very similarly to  $M_k^2$  (as shown in Appendix F-G). Finally, in order to get  $M_k$ ,  $M_k^2$  and  $M_k^1$  have to be merged. This way, the scheme of our algorithm could be written as in Alg. 3.

Complexity bounds on the non-parametrized computing time and length of  $M_k$  are summarized as part of the table in Alg. 3. Although in Appendix I it is shown that there are some artificial networks where these asymptotic bounds are relatively good estimations, we would like to focus on the running time and output size on the real networks, which are nearly planar. Thus, after introducing a new parameter, we present parametrized bounds proven in Appendix B-G.

Intuitively, a  $c \in \mathcal{C}_k$  cannot hit too many edges. We introduce graph density parameter  $\rho_k$ , to describe this phenomenon.

**Definition 22.** For all  $i \in \{0, n-2\}$ , let  $\rho_i$  be the maximum number of edges hit by a disk from  $C_i$ .

The parametrized bounds are the following:

**Lemma** 23. 
$$M_k^2$$
 can be computed in  
 $O\left(n^2\left(\left(k^2+1\right)\rho_k^3+\left(k+1\right)\left(\rho_0\gamma\right)\right)\right)$ .  $M_k^2$  has  $O\left(n\left(k+1\right)\rho_k\right)$ 



Fig. 5: In the setting above, circle  $c_{ABC}$  hits the whole link set  $\mathscr{E}$ , while no  $c \in \mathscr{C}_0^{B,D}$  hits link f. Thus, however  $M_0^{B,D}$  is not empty, it does not contain any elements of  $M_0$ .

Algorithm 3: Algorithm for computing  $M_k$  with a table on time complexities. (Refer to Table I for notations.)

<b>Input:</b> $\mathcal{G}(V,\mathcal{E}), k$ <b>Output:</b> $M_k$ begin		Complexity	Non-parametrized	Parametrized					
	Determining $M_k^2$	STEP 1	$O((k^2+1)n\log n)$	$O((k^2+1)n\log n)$					
1	Determine $E_k$	STEP 2	$O(n(k+1)m(\log m + \gamma))$	$O((k+1)n(n\rho_0\gamma + k\log k + \rho_k\log \rho_k))$					
2	Determine set $A_k$ of nonempty apples	STEP 3	$O(n(k+1)m^3)$	$O(n(k+1)\rho_k^3)$					
3	Query apples from $A_k$	STEP 4	$O(n^2(k^2+1)m^3)$	$O(n^2(k^2+1)\rho_{1}^3)$					
4	Merge lists $M_k^{\mu\nu}$ into $M_k^{\mu}$			<u> </u>					
	Determining $M_{L}^{1}$ // See App. F-G for details	STEP 5	$O(nm\gamma \log m)$	$O(n^2 \rho_0 \gamma \log(n \rho_0))$					
5	Determine set $S_k$ of nonempty seesaws	STEP 6	$O(nm^3)$	$O(n\rho_0 + \rho_k^3)$					
6	Query seesaws from $S_k$	STEP 7	$O(n^2 m^3)$	$O(n^2 \rho_k^3)$					
7	Merge lists $M_k^w$ into $M_k^1$	STEP 8	$O(n^2(k+1)m^3)$	$\frac{n}{O(n^2(k+1)\rho_1^3)}$					
8	Merge lists $M_k^2$ and $M_k^1$ into $M_k$	STER 0	0(11 (11 1))))	<u> </u>					
9	return M <sub>k</sub>	Total for $\gamma = O(1)$	$O(n^2(k^2+1)m^3)$	$O(n^2((k^2+1)\rho_k^3+\rho_0\log(n\rho_0)))$					

#### elements with at most $\rho_k$ edges.

Proof of Lemma 23 can be found in Appendix E.

Besides computing  $M_k^2$ , one have to deal with computing  $M_k^1$ . When computing  $M_k^1$ , the vague idea is to give a geometric algorithm in a way similar to the sweep disk algorithm for querying the apples. Now instead of imaginary sweeping a disk, we rotate a half-plane around every node  $v \in V$  until it makes a total turn, and meanwhile, check for hit edge sets with locally maximal cardinalities hit by half-planes hitting exactly k nodes. After this, the maximal elements of the obtained lists are collected in  $M_k^1$ . Now follows Lemma 24 for computing  $M_k^1$ . We refer the reader to see Appendix F-G for its detailed proof.

**Lemma 24.**  $M_k^1$  can be constructed in  $O(n^2(\rho_0\gamma \log n\rho_0 + \rho_k^3))$ and has  $O(n\rho_k)$  elements, each containing at most  $\rho_k$  edges.

**Theorem 3.**  $M_k$  can be computed in  $O(n^2((k^2+1)\rho_k^3+\rho_k\gamma+(k+1+\log(n\rho_0))\rho_0\gamma)))$ .  $M_k$  has  $O(n(k+1)\rho_k)$  elements with at most  $\rho_k$  edges.

*Proof:* Based on Lemmas 23 and 24, both  $M_k^2$  and  $M_k^1$  can be computed in the proposed time, have at most the proposed amount of elements containing at most  $\rho_k$  edges. The proof will be completed by showing that the merger of  $M_k^2$  and  $M_k^1$  can be done in  $O(n^2(k+1)\rho_k^3)$ , which is true because of the followings.

We only have to compare all the pairs  $\{p_2, p_1\}$  made up of a  $p_2 \in M_k^2$  and  $p_1 \in M_k^1$ , which means  $O((n(k+1)\rho_k)(n\rho_k))$ pairs. Each comparison can be made in  $O(\rho_k)$ ,<sup>6</sup> which gives the proposed complexity.

**Corollary 25.** If  $\rho_k$  is O(k+1), then  $M_k$  has O(n(k+1)) elements. If in addition  $\gamma$  is upper bounded by a constant,  $M_k$  can be computed in  $O(n^2(k^5 + \log(nk+1) + 1)).^7$ 

# **IV. SIMULATION RESULTS**

In this section, we present numerical results that validate our model and demonstrate the use of the proposed algorithms on some realistic physical networks. The algorithms were implemented in Python version 3.5 using various libraries. No

<sup>6</sup>Since link sets are ordered lexicographically.

<sup>7</sup>To be exact,  $\gamma = O\left(\max\left\{1, \frac{k^5 + 1}{k + \log n}\right\}\right)$  yields the proposed complexity.

special efforts were made to make the algorithm space or time optimal. The output of the algorithm is a list of SRLGs so that no SRLG contains the other. The network topologies with the obtained list of SRLGs for various k are available online<sup>8</sup>.

First, 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.

The second interpretation is artificial, and we mainly use it for verification. Intuitively, the two interpretations result in very different results, as the latter has much more nodes in the network, and thus the regional failures with k nodes interior must be smaller. Fig. 6 shows example results for both interpretations of the US ATT-L1 network. The US fiber network has 126 nodes and 208 links as polygonal chains, where the links have 36 corner nodes in total. After transforming it into a network of line segments, we will have a larger network with 162 nodes and 244 links. The transformed network has 30% more nodes; however, the number of SRLGs required for k = 0 is just 14% more, which is a sub-linear increase. Surprisingly, after the transformation, the SRLGs became a bit smaller (average number of links  $2.98 \rightarrow 2.79$ ), and the variance in the size of SRLGs is increased from 0.7 to 0.83. It is because in the transformed, a great number of very small SRLGs appeared, having the two adjacent links for most of the degree 2 nodes.

TABLE II: Results of physical backbone topologies of [18].

Name	V		8		# SRLG	k = 0	# SRLG	k = 1
	Polygon	Line	Polygon	Line	Polygon	Line	Polygon	Line
Pan-EU	10	16	16	22	14	19	27	35
EU (Optic)	17	22	40	45	44	59	57	71
EU (Nobel)	19	28	32	41	36	46	53	81
US [7]	21	24	39	42	48	49	57	64
NAmerican	28	39	50	61	65	76	83	97
US (NFSNet)	44	79	73	108	88	124	128	172
US (Fibre)	81	170	141	230	137	189	177	249
US (Deltacom)	103	103	302	302	158	158	218	218
US (Sprint-Phys)	111	264	160	313	156	232	208	307
US (ATT-L1)	126	162	208	244	190	216	255	285
US (Att-Phys)	209	383	314	488	256	352	322	457

<sup>8</sup>https://github.com/jtapolcai/regional-srlg



(a) Polygon: there are 190 SRLGs with average of 2.98 links and  $\rho_0 = 5$ . (b) Line: there are 216 SRLGs with average of 2.79 links and  $\rho_0 = 5$ .

Fig. 6: The SRLGs of k = 0 are visualized for the two cases (a) links are polygonal line segments, and (b) the corner points of the polygonal line segments are treated as degree two nodes and all links are straight lines. In order to have a perspicuous illustration, each SRLG is drawn with the smallest possible circular disk that covers all of its links, even if the disk has nodes interior.

#### A. The list of SRLGs in practice

Table II shows a comparison for k = 0 and k = 1 among eleven physical backbone topologies taken from [18]. The columns are: network name, the number of nodes, the number of links, the number of SRLGs, all for both cases where links are polygonal line segments or each corner point of the polygonal links are substituted with a degree 2 node.

The 126-node US (ATT-L1) network was covered with 190 SRLGs, which is less than listing every single node and link as an SRLG. Fig. 6 shows these SRLGs, intuitively each corresponds to a mid-size regional failure. The SRLGs meet our intuition that there are more network nodes in the crowded areas, and thus it generates more SRLGs for them, while in the less crowded areas are covered with SRLGs corresponding to bigger areas. In practice, it is important to have small SRLGs because it strongly influences the performance of the survivable routing algorithms. On Fig. 6 the SRLGs are relatively small, each SRLG contains a bit less than 3 links on average. Fig. 7c shows how the average size of SRLGs with respect to k for all networks. It has a slightly sub-linear increase with k. Note that the length of the list of SRLGs never exceeded 6000 items in any networks and parameter settings examined.

Next, we have evaluated what would be the radius of the circular disk with k = 0, 1, 2, 3 nodes when we know the GPS positions of the nodes (in case of network US ATT-L1). We have performed a Monte Carlo simulation where



Fig. 8: The cumulative distribution function of the radius of the disk having a given k number of nodes interior.

we pick random locations and compute the maximum radius with k = 0, 1, 2, 3 nodes, which is the distance of the closest, the second, third, and fourth closest nodes. Fig. 8 shows the cumulative distribution function of the actual radius of the circular disk failures. For example, if we cut the smallest and largest 20% the SRLGs generated for k = 0 nodes corresponds to diameter 80-200km.

### B. Tightness of Corollary 25

In this subsection we compare the presented parameterized worst case analysis with the obtained simulation results. Corollary 25 assumes that  $\rho_k$  increases linearly with (k+1) for all networks. Fig. 7a shows that the edge density increases linearly with (k+1) as we expected. In Corollary 25 a near



Fig. 7: The edge density, number and size of SRLGs for each network and  $k = \{0, \dots, 5\}$  in case of polygonal chain links.



(a) The number of SRLGs with respect to (b) Measured time complexity. (c) The average the number of (d) The number of SRLGs and the number of the number of nodes, for k = 0. (c) The average the number of (d) The number of SRLGs and the number of edges in graph *k*-Delaunay for US Deltacom

Fig. 9: Comparison of the two interpretation of the input topologies: links are polygonal line segments, or the corner points of the polygonal chains are treated as degree two nodes and all links are line segments.

quadratic worst case increase of computation time was showed with respect to the network size n for fixed k (more precisely it is  $O(n^2 \log n)$ ). It is well reflected in Fig. 9b. Note that we measured the algorithms' runtime on a commodity computer with a CPU at 2.4 GHz and 32 GB of RAM without optimizing the implementation for speed. Corollary 25 provides a linear upper bound on the number of SRLGs with respect to the network size n for fixed k. It is also reflected in Fig. 9a which is a graphical illustration for k = 0, where we may even have the intuition of a sub-linear growth. Based on Table II the slope of the curve can be estimated as the number of SRLGs is roughly  $\approx 1.2n$  for k = 0, and  $\approx 2.2n$  for k = 1. Corollary 25 provides a linear upper bound on the number of SRLGs with respect to k if the network (n) is fixed. This is illustrated in Fig. 9c where the average number of SRLG is shown for all networks for small k. Here we can experience a slightly sub-linear increase. Fig. 7b shows the increase in the number of SRLGs for each network independently for the same range of k. We experienced sub-linearity for all networks, which we further discuss later. Overall, numerical evaluation supports the parameter selection used in the parametrized complexity analysis. We conjecture that the Corollary 25 is close to the experienced performance, and there is little hope for further improving it analytically.

To reveal the reasons why the number of SRLGs increases sub-linearly with respect to k, Fig. 9d shows the function graphs of both  $|M_k|$  and  $\frac{|E_k|}{3}$  for all k values on Network US Deltacom, respectively. As we can see, for  $k \le 15$ ,  $|M_k| \approx \frac{|E_k|}{3}$ , while for bigger k values  $|M_k| < \frac{|E_k|}{3}$ . We can deduct that  $|M_k| \le \frac{|E_k|}{3}$  for every  $k \in \{0, ..., n-2\}$ . In other words, the average number of SRLG-s per apple is  $\le \frac{1}{3}$ . By Thm. 2, this also means that  $|M_k|$  is O((k+1)(n-k)), wich induces a sub-linear growth of  $|M_k|$  with respect to k.

### V. OPEN PROBLEMS

The presented algorithm computes  $M_k$  in  $O(n^2(k^5 + k\log(nk+1) + 1))$  (if  $\gamma$  is constant and  $\rho_k$  is O(k+1)). A natural question is whether it is possible to compute  $M_k$  quicker. For k = 0 the answer is yes in case of usual networks: as presented in [33],  $M_0$  can be determined in  $O(n(\log n + \rho_0^3 \tau_0))$ , where  $\tau_0$  is an additional

parameter depending on local properties of the embedding of the network in the plane (and edges are considered as line segments). In other words, there exists an algorithm to compute  $M_0$  in  $f(\rho_0, \tau_0)O(n\log n)$ .

We believe that there exist similar algorithms for determining  $M_k$  in  $f(k, \rho_k, \gamma, ...) O(n \log n)$ , where f depends only on k and on 'local' properties of the embedding. This near-linear complexity in n would allow  $M_k$  to be computed even quicker in case of huge networks too. Unfortunately, presenting this kind of algorithms would exceed the limits of this paper; thus it will be part of future work.

# VI. CONCLUSIONS

In this paper, we propose a fast and systematic approach to enumerate the list of possible regional failures the network should be prepared for. We define regions as circular disks with maximum size such that the number of nodes inside is at most k. The proposed failure model gives a reasonable conservative description of the possible regional failures with a minor drawback of overestimating the size of the failed region, but with the following three benefits that

- It can be computed efficiently, even for very large networks.
- The list of SRLGs is short enough to fit in the current router configurations.
- A schematic topology map discovered by automatic tools is sufficient to compute the list of SRLGs and does not require precise knowledge on the geometry of the network, i.e. the exact physical location of the nodes and links, the physical distances in the network topology, etc.

Using the proposed algorithm, operators can protect their networks against regional failures by running the current selfhealing mechanisms with the obtained list of SRLGs.

# REFERENCES

- 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.
- [2] J. Rak, "Resilient routing in communication networks," in Computer Communications and Networks, 2015.
- [3] S. Neumayer, G. Zussman, R. Cohen, and E. Modiano, "Assessing the vulnerability of the fiber infrastructure to disasters," *IEEE/ACM Transactions on Networking*, 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?" *IEEE Communications Magazine*, 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," *IEEE Communications Magazine*, 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," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 230–238, 2014.
- [11] R. S. Couto, S. Secci, M. M. Campista, K. Costa, and L. Maciel, "Network design requirements for disaster resilience in iaas clouds," *IEEE Communications Magazine*, vol. 52, no. 10, pp. 52–58, 2014.
- [12] "Shaping Europe's digital future, Actors in the broadband value chain," 2019, European Commission, https://ec.europa.eu/digital-single-market/ en/actors-broadband-value-chain.
- [13] "Huawei NE20E-S Universal Service Router, feature description -MPLS," 2018, v800R010C10SPC500.
- [14] "Alcatel-lucent 7705 service aggregation router OS, MPLS guide," 2013, release 6.0.R4.
- [15] "Cisco WAE design 7.1.2 user guide, chapter 19 : LSP disjoint path optimization," 2018.
- [16] "Juniper networks junos OS MPLS applications user guide, chapter 3 MPLS traffic, shared risk link groups for MPLS," 2020.
- [17] K. Pithewan, M. Yaaseen, R. P. Ramamoorthy, and M. Misra, "Disjoint path computation for arbitrary directed graph," 2016, US Patent 9,253,032.
- [18] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, "SNDlib 1.0: survivable network design library," *Networks*, vol. 55, no. 3, pp. 276– 286, 2010.
- [19] B. Vass, J. Tapolcai, D. Hay, J. Oostenbrink, and F. A. Kuipers, "How to model and enumerate geographically correlated failure events in communication networks," in *Guide to Disaster-Resilient Communication Networks*. Springer, July 2020.
- [20] S. Trajanovski, F. Kuipers, P. Van Mieghem et al., "Finding critical regions in a network," in Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2013, pp. 223–228.
- [21] J. Tapolcai, L. Rónyai, B. Vass, and L. Gyimóthi, "Fast enumeration of shared risk link groups representing regional failures with limited size," *IEEE Transactions on Networking*, 2020.
- [22] Y. Kobayashi and K. Otsuki, "Max-flow min-cut theorem and faster algorithms in a circular disk failure model," in *IEEE INFOCOM*, 2014, pp. 1635–1643.
- [23] L. Shen, X. Yang, and B. Ramamurthy, "Shared risk link group (SRLG)diverse path provisioning under hybrid service level agreements in wavelength-routed optical mesh networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 4, pp. 918–931, 2005.
- [24] 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.
- [25] 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*, vol. 21, no. 5, pp. 1525–1538, 2013.
- [26] 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.
- [27] 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, Apr. 2018.
- [28] C. Cao, Z. Wang, M. Zukerman, J. H. Manton, A. Bensoussan, and Y. Wang, "Optimal cable laying across an earthquake fault line con-

sidering elliptical failures," *IEEE Transactions on Reliability*, vol. 65, no. 3, pp. 1536–1550, 2016.

- [29] 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.
- [30] F. Zhang, O. G. de Dios, M. Hartley, Z. Ali, and C. Margaria, "RSVP-TE Extensions for Collecting Shared Risk Link Group (SRLG) Information," RFC 8001, Jan. 2017. [Online]. Available: https://rfc-editor.org/rfc/rfc8001.txt
- [31] Z. Ali, G. Swallow, F. Zhang, and D. Beller, "RSVP-TE Path Diversity Using Exclude Route," RFC 8390, Jul. 2018. [Online]. Available: https://rfc-editor.org/rfc/rfc8390.txt
- [32] F. Aurenhammer, "Voronoi diagrams: a survey of a fundamental geometric data structure," ACM Computing Surveys (CSUR), vol. 23, no. 3, pp. 345–405, 1991.
- [33] B. Vass, E. Bérczi-Kovács, and J. Tapolcai, "Shared risk link group enumeration of node excluding disaster failures," in *Int. Conference on Networking and Network Applications (NaNA)*, 2016.
- [34] B. Vass, E. Bérczi-Kovács, and J. Tapolcai, "Enumerating circular disk failures covering a single node," in *Proc. 8th International Workshop on Resilient Networks Design and Modeling (RNDM)*, 2016, pp. 189–195.
- [35] —, "Enumerating shared risk link groups of circular disk failures hitting k nodes," in *Proc. International Workshop on Design Of Reliable Communication Networks (DRCN)*, Munich, Germany, march 2017.
- [36] M. Abellanas, P. Bose, F. H. J. García, C. M. Nicolás, and P. A. Ramos, "On structural and graph theoretic properties of higher order Delaunay graphs," in *International Journal of Computational Geometry* & Applications, Volume 19, 2009.
- [37] E. Papadopoulou and M. Zavershynskyi, "The higher-order voronoi diagram of line segments," *Algorithmica*, vol. 74, no. 1, pp. 415–439, Jan 2016. [Online]. Available: https://doi.org/10.1007/s00453-014-9950-0
- [38] R. Pinchasi and G. Rote, "On the maximum size of an anti-chain of linearly separable sets and convex pseudo-discs," *Israel Journal of Mathematics*, vol. 172, no. 1, pp. 337–348, 2009.
- [39] T. M. Chan, "On levels in arrangements of surfaces in three dimensions," Discrete & Computational Geometry, vol. 48, no. 1, pp. 1–18, 2012.
- [40] S. Neumayer, A. Efrat, and E. Modiano, "Geographic max-flow and mincut under a circular disk failure model," *Computer Networks*, vol. 77, pp. 117–127, 2015.
- [41] F. Aurenhammer, R. Klein, and D. Lee, Voronoi diagrams and Delaunay triangulations. World Scientific Publishing Co Inc, 2013.
- [42] T.-H. Su and R.-C. Chang, "The k-Gabriel graphs and their applications," in Algorithms. Lecture Notes in Computer Science, vol. 450., Berlin, Heidelberg, 1990.
- [43] "Notes on stereographic circles," http://www.cs.bsu.edu/homepages/ fischer/math345/stereo.pdf, accessed: 2018-01-15.

# APPENDIX

A. Protecting regional link 0-node failures ensures node disjointness

**Claim 26.** Let *P* and *B* be an SRLG-disjoint primary and backup paths according to regional link 0-node failures. Then *P* and *B* are node disjoint, apart from the terminal nodes.

**Proof:** Assume indirectly that P and B have a common node v in their interior. Let us pick two edges  $e_p \in P$  and  $e_b \in B$  of G that are adjacent of v. Let p denote the closest corner point of  $e_p$  with node v if there is any, otherwise its endpoint on P. Similarly we define b for  $e_b$ . Note that points v, b and p are not on the same line according to our assumptions in Sec. II; thus, the is a circular disk of any size that covers both  $e_p$  and  $e_b$  but not node v. We can select a small enough radius for this circular disk which covers both  $e_p$  and  $e_b$  and does not have any other nodes interior. The proof follows.

# PARAMETRIZED ALGORITHM FOR ENUMERATING MAXIMAL FAILURES IN $M_k^2$

# B. Parametrized Complexity Bounds for Determining Apples

Up to this point the fact that  $\mathscr{G}(V,\mathscr{E})$  is, in fact, a graph of a communication network, and thus it is 'almost planar' was not used. Intuitively, an almost planar graph has O(n) edges. Parameters  $\rho_i$  ( $i \in \{0, ..., n-2\}$ ) denote the maximum number of edges hit by a disk from  $\mathscr{C}_i$  (Def. 22), thus they help in formalizing this intuition.

Since parameters  $\rho_i$  measure local properties of the networks, often it will be assumed that these parameters are not exceeding a constant. For example,  $\rho_0$  is not going to be large since where there are many edges, a node likely appears.

**Observation 27.** For any  $0 \le i < j \le n-2$ ,  $\rho_i \le \rho_j$ .

**Claim 28.** In any apple  $A_k^{u,v} \in A_k$ ,  $|\mathcal{E}_+| \le \rho_k$  and  $|\mathcal{E}_-| \le \rho_k$ .

*Proof:* All edges in  $\mathcal{E}_+$  are hit by  $c(x_{max})$  which by definition hits at most  $\rho_k$  edges. Similar for  $\mathcal{E}_-$ .

**Lemma 29.** The number of edges is  $O(n\rho_0)$ , more precisely  $m \le (2n-5)\rho_0$ .

*Proof:* Consider the Delaunay triangulation  $D_0$ , which is a planar graph, and thus  $|E_0| \le 3n - 6$ . Since every Delaunay triangle has 3 Delaunay edges and a Delaunay edge is the 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

$$\frac{2|E_0|-3}{3} \le \frac{2}{3}(3n-6) - 1 = 2n - 5.$$

Since the polygon of every edge in  $\mathscr{E}$  intersects at least one triangle, and every triangle can be covered by a disk  $c \in \mathscr{C}_0$ , which intersects at most  $\rho_0$  polygon of edges of the network, we get that the number *m* of edges cannot be larger than  $\rho_0$  times the number of the Delaunay triangles. We get  $m \le (2n-5)\rho_0$ .

**Lemma 30.** If  $E_k$  is given, set  $A_k$  of apples can be calculated in  $O((k+1)n(n\rho_0\gamma + k\log k + \rho_k\log \rho_k))$ .

*Proof:* There are  $|E_k| \leq 3(k+1)n$  apples to determine (Thm. 2). For each,  $V_+$  and  $V_-$  can be determined in  $O(n + k \log k)$ , then (based on Lemma 29)  $O(n\rho_0)$  edges have to be checked if they are in the apple, each in  $O(\gamma)$  time. After this, based on Claim 28, there are  $O(\rho_k)$  edges to order, which gives the proposed complexity.

**Corollary 31.** If  $\rho_k$  and  $\gamma$  is O(1) and  $E_k$  is given, then  $A_k$  can be determined in  $O(n^2(k + \log n))$ .

C. Parametrized Bound for Determining  $M_k^{u,v}$  for Apple  $A_k^{u,v}$ **Lemma 32.** For all the apples  $A_k^{u,v}$  in  $A_k$ , sets  $M_k^{u,v}$  can be determined in  $O(n(k+1)\rho_k^3)$ .

*Proof:* Based on Thm. 2, there are  $|E_k| \le 3n(k+1)$  apples to query. We claim that each of them can be queried in  $O(\rho_k^3)$ . Knowing apple  $A_k^{u,v}$ ,  $(M_k^{u,v})'$  can be determined in

 $O(\rho_k^2)$ , following the steps of Alg. 2. After this,  $M_k^{u,v}$  can be determined by comparing each pair of elements of  $(M_k^{u,v})'$  and eliminating its non-maximal and redundant members. With this purpose,  $O(\rho_k^2)$  comparisons have to be made, each of them has  $O(\rho_k)$  complexity.

D. Parametrized Complexity Bound on Merging Lists  $M_k^{u,v}$ Lemma 33.  $M_k^2$  can be computed in  $O(n^2(k^2+1)\rho_k^3)$  from lists  $M_k^{u,v}$ .

*Proof:* By Thm. 2, there are  $|E_k| \leq 3n(k+1)$  lists containing  $O(\rho_k)$  sets containing  $O(\rho_k)$  edges. First determine an ordering on the set of edges  $\mathscr{E}$ , and sort all the candidate sets of edges according to this ordering, each in  $O(\rho_k \log \rho_k)$  time.  $M_k^2$  can be computed by comparing all the set pairs (and eliminating the redundant or non-maximal elements), which means  $O(n^2(k^2+1)\rho_k^2)$  comparisons. Since (due to the ordering) comparing two sets takes  $O(\rho_k)$  time, the total complexity is  $O(n^2(k^2+1)\rho_k^2)$ .

# E. Parametrized Complexity for Computing $M_k^2$

Now follows the proof of Lemma 23, which states the followings.  $M_k$  can be computed in  $O(n^2((k^2+1)\rho_k^3+(k+1)\rho_0\gamma))$ .  $M_k$  has  $O(n(k+1)\rho_k)$  elements with at most  $\rho_k$  edges.

*Proof:* As presented previously (Thm. 1 and Lemmas 30, 32 and 33), each of the corresponding four phases of Alg. 3 can be examined in the proposed complexity. There are  $|E_k| \le 3n(k+1)$  lists  $M_k^{u,v}$  to merge, each of them has at most  $\rho_k$  edges, completing the proof.

# PARAMETRIZED ALGORITHM FOR ENUMERATING MAXIMAL FAILURES IN $M_k^1$

In this section a sketch of an algorithm will be presented for proving Lemma 24, which states that  $M_k^1$  can be constructed in  $O(n^2(\rho_0\gamma \log(n\rho_0) + \rho_k^3))$  and has  $O(n\rho_k)$  elements, each containing  $O(\rho_k)$  edges.

# F. Data Structure Seesaw

For every node  $w \in V$ , a data structure is built containing both the direction of every node  $z \in V \setminus \{w\}$  related to v and the interval of directions where the polygon of each edge can be seen from w. Nodes and edges also have to be sorted according to this information (similarly to data structure apple). Let us call this data structure *seesaw*, and let  $S_k^w$  denote the previously described seesaw. For a given k, let the set of seesaws be denoted by  $S_k$ . Let the list of locally maximal failures resulting from querying seesaw  $S_k^w$  be denoted by  $M_k^w$ .

**Claim 34.** Any seesaw  $S_k^w$  can be calculated in  $O(n\rho_0\gamma \log(n\rho_0))$  and has a total length of  $O(n\rho_0)$ .

**Proof:** Trivially, the direction of nodes from  $z \in V \setminus \{w\}$  can be determined and sorted in  $O(n \log n)$ . Also, as by Lemma 29, the number of edges is  $O(n\rho_0)$ , intervals of directions corresponding to polygons of edges can be calculated and sorted both by their minimum and maximum values on  $O(n\rho_0\gamma \log(n\rho_0))$ . The proof follows.

# G. Querying Seesaws

**Claim 35.**  $M_k^w$  can be computed from  $S_k^w$  in  $O(n\rho_0 + \rho_k^3)$ , and has at most  $2\rho_k$  elements, each containing at most  $\rho_k$  edges.

*Proof:* Since each set in  $M_k^w$  can be hit by a disk in  $\mathscr{C}_k$  having w on the boundary, the fact that the elements of  $M_k^w$  contain at most  $\rho_k$  edges is trivial by definition.

Now we prove  $|M_k^w| \le 2\rho_k$ . If there is no half-plane having w on the boundary hitting exactly k nodes, then the claim is trivial, otherwise, let  $h_0^w$  be such a half-plane. Let  $h_+^w$  be the unique half-plane which satisfies the followings: it is the rotation of  $h_0^w$  with  $\angle \le \pi$ , it covers exactly k nodes, and no other half-plane covers exactly k nodes, which is the rotation of  $h_+^w$  with an angle  $\in [\angle, \pi]$ . Let  $h_-^w$  be similar, but with rotating towards the negative direction. This way, every half-plane going through w covering exactly k nodes is part of  $h_-^w \cup h_+^w$ , which altogether hit at most  $2\rho_k$  edges. Since while turning a half-plane around w, edges are getting hit or not hit one by one, and an edge at most 2 times, there can exist at most  $2\rho_k$  hit edge sets with locally maximal cardinalities.

Determining  $M_k^w$  can be done by simply turning the halfplane around w. Checking currently hit edge sets, node sets and cardinalities can be done in a total  $O(n\rho_0)$  time, list  $M_k^w$ can be created in  $O(\rho_k^3)$ .

Proof of Lemma 24: Trivially, every element of  $M_k^1$  contains at most  $\rho_k$  elements. Based on Claims 34 and 35, lists  $M_k^w$ can be determined in  $O(n^2\rho_0\gamma\log(n\rho_0) + n\rho_k^3)$  for all  $w \in V$ .  $M_k^1$  can be calculated by merging the previous lists.  $O(n^2\rho_k^2)$ comparisons have to be done, each has a complexity of  $O(\rho_k)$ . This means a total complexity of  $O(n^2(\rho_0\gamma\log(n\rho_0) + \rho_k^3))$ , completing the proof.

# H. Connection Between Seesaws and Apples

It is known that the inverse of the stereographic projection from the North Pole, which maps geometric objects from the plane to the surface of the sphere, has the property that the image of lines and line segments on the plane are great circles and geodesics on the sphere, respectively (corollary of Theorem in [43]). Known this, if we projected our network topology (back) to a spherical surface, and if we defined the spherical apple and spherical seesaw data structure, we could see that a spherical seesaw  $M_k^w$  is just a spherical apple  $M_k^{w,z}$ , z being the North Pole, thus sweeping through a spherical apple is the same thing as tilting a spherical seesaw, meaning that on the sphere there would not emerge the need to treat separately these two connected data structures.

# EXTREME CASES

#### I. Maximum Number of Maximal Failures

In the following, we show that there are some networks for which the running times are asymptotically not significantly better than our non-parametrized bounds. This motivates the introduction of parameter  $\rho_k$ , which captures the properties of real-life networks.



Fig. 10: Sketch of a graph family where  $|M_k| = \Theta(n^3)$  if k = O(1).

**Theorem 4.**  $\max_{\mathcal{N}} |M_k| = \Theta(n^3)$  if k = O(1), where  $\mathcal{N}$  is the set of all networks on n points.

**Proof:** The proof will be immediate from Cor. 38. It turns out that  $|M_k|$  is O((k+1)nm) (Claim 36), and in case of some artificial network families  $|M_k|$  is  $\Omega((n-k)n^2)$  (Cor. 38). This means that for k = O(1) the maximum number of maximal failures is  $\Theta(n^3)$ . The details are the following.

# **Claim 36.** The number of disk failures is O((k+1)nm).

*Proof:* Since  $|E_k| = O((k+1)n)$  (Thm. 2), there are O((k+1)n) apples. For each apple  $A_k^{u,v}$ ,  $|M_k^{u,v}| \le O(m)$  (proof of Claim 20). Since  $|M_k^2| \le \sum |M_k^{u,v}|$ , this means that  $|M_k^2|$  is O((k+1)nm).

On the other hand, the number of seesaws  $|S_k| \le n$ , and each  $|M_k^w| \le O(m)$ , which gives  $|M_k^1| = O(nm)$ . Since  $|M_k| \le |M_k^2| + |M_k^1|$ , the proof is complete.

**Claim 37.** There is a graph family for which  $|M_k| = \Omega(n^3)$  if k = O(1).

Sketch of proof: On Fig. 10 we can see a sketch of a graph  $\mathcal{G} = (V, \mathcal{E})$  for which  $|M_0| = \Theta(n^3)$ . It has a so long-drawn shape it cannot be drawn precisely in a paper.

Let *l* be a vertical line. Both on the right and left side of *l* let us take a complete bipartite graph called  $K_{b,b}^+$  and  $K_{b,b}^-$ , respectively (see Fig 10). We locate the points of  $K_{b,b}^+$  and  $K_{b,b}^-$  as follows. For both  $K_{b,b}^+$  and  $K_{b,b}^-$  one class of nodes is located at the top and the other at the bottom such that their vertices are equidistant on a horizontal line (i.e for nodes  $w_i$  in top class of  $K_{b,b}^+$ , distance  $d(w_i, w_{i+1})$  is constant *c* for all  $i \in \{1, \dots, b-1\}$ , same for the other classes). Let the *y* coordinates of the top and bottom classes differ in a much larger scale than the former constant *c*. Let  $e_{i,j}$  be the edge incident to  $w_i$  and the *j*<sup>th</sup> node in the bottom class. There is a horizontal line  $l_h$  such that common points  $p_{i,j}$  of edges  $e_{i,j}$  for a given *i* and  $j \in \{1, \dots, b\}$  and line  $l_h$  are located nearly equidistant on  $l_h$ , let this distance be denoted by  $c_2$ . Clearly,  $l_h$  can be chosen such that the abscissa of  $p_{i+1,1}$  is the abscissa of  $p_{i,b}$  plus  $c_2$  (for all  $i \in \{1, \dots, b-1\}$ ).

Let  $a \ge k+2$ . Let us consider  $v_1, \ldots, v_a \in V$ , different nodes lying equidistant on l, such that about half of them are above line  $l_h$  while the rest is below it. Let the distance between these nodes be  $c_v$ , where  $c_v$  is chosen such that the distance between the leftmost edge of  $K_{b,b}^+$  and  $K_{b,b}^-$  to be  $(k+1)c_v + \varepsilon$ , where  $\varepsilon \simeq \frac{c}{2}$ . Finally, we perturb the location of every node to avoid degeneracies.

Due to this setting, considering apples  $A_k^{v_i,v_{i+k+1}}$ , while sweeping from left to right, edges are getting hit and unhit in turn, thus  $|M_k^{v_i,v_{i+k+1}}| = \Theta(b^2)$  for every  $1 \le i \le a - (k+1)$ .

Let  $V_a := \{v_1, \dots, v_a\}$ , and  $\mathcal{E}_a$  be the set of edges induced by  $V_a$ .

Since for all element  $m_i \in M_k^{v_i,v_{i+k+1}}$ ,  $m_i \cap \mathcal{E}_a = \{\{v_i, v_{i+1}\}, \dots, \{v_{i+k}, v_{i+k+1}\}\}$ , we can deduct that there are no duplicates in these lists, i.e.  $|M_k| \ge \sum_{1 \le i \le a-(k+1)} |M_k^{v_i,v_{i+k+1}}|$ , since there are other apples which we have not considered.

This means  $|M_k| \ge (a - (k + 1))\Theta(b^2)$ . The proof can be completed by choosing both *a* and *b* to be  $\Theta(n)$ , for example  $a \ge b \ge \frac{n}{5}$ .

**Corollary 38.** The graph illustrated on Fig. 10. has  $\Theta(n^3)$  maximal link k-node failures if k is O(1).

Although  $|M_k|$  can be  $\Omega(n^3)$ , according to this paper in case of many real-life networks it is O((k+1)n).

COMPUTATIONAL GEOMETRIC CONSIDERATIONS

J. Determining  $x_+(e)$  and  $x_-(e)$  in  $O(\gamma)$ 

**Claim 39.** For any edge  $e = \{a, b\}$  and node pair  $\{u, v\} \subset V$ , both  $x_+(e)$  and  $x_-(e)$  can be calculated in  $O(\gamma)$ .

*Proof:* Let us concentrate on calculation of  $x_+(e)$ , because  $x_-(e)$  can be determined similarly.

*Extreme hitting disk of line:* First, let us compute the leftmost hitting circle in  $\mathscr{C}^{u,v}$  of a point W part of a given line ab. Let  $\{A\} = uv \cap ab$  and let  $x_+(W)$  be the abscissa of the center point of the leftmost hitting disk of W. Clearly, the function  $x_+(W)$  is unimodal in both rays  $(R_+ \text{ and } R_-)$  defined by line ab and point A (i.e., both  $R_+$  and  $R_-$  are consisting of an interval where it is strictly monotone increasing and another interval where strictly monotone decreasing).

Let the two points on the line where the local minimum is reached be  $W_+$  on the right side and  $W_-$  on the left side of uv. Let the center point of disks  $c(uvW_+)$  and  $c(uvW_-)$  be  $X_+$  and  $X_-$ , respectively.

 $X_+$  is located on the Ox axis of the coordinate system of apple  $A_k^{u,v}$ . On the other hand,  $X_+$  is located equidistant from u and line ab, thus it is on parabola  $p_u$  defined by point u and line ab. Similarly, let  $p_v$  be the parabola defined by v and line ab.

Since  $p_u$  and  $p_v$  can be characterized with quadratic expressions, which can be solved in O(1), abscissa of  $X_+$  can be found in constant time by determining their common root.

 $X_{-}$  can be determined similarly. Finally, the center point of the desired leftmost hitting disk is the one from  $X_{+}$  and  $X_{-}$  with smaller abscissa.

*Extreme hitting disk of line segment:* When restricting the domain of a unimodal function, extremes can appear on the new boundary, thus for segment [a,b],  $x_+([a,b]) = \min(x(X_+), x(X_-), x(A), x(B))$ .

Extreme hitting disk of the polygon of edge: For an arbitrary  $e \in \mathcal{E}$  we consider two cases regarding the respective position of [uv] and  $e^p$ .

In the first case segment [uv] is entirely in the interior of  $e^p$ . Trivially, in this case  $x_+(e) = -\infty$ .

In the second case, segment [uv] is not entirely in the interior of  $e^p$ . Since in this model a polygon of edge  $e^p$  consists of at most  $\gamma$  line segments, for determining  $x_+(e)$  one can determine  $x_+$  for all the line segments that take the minimum value of them in  $O(\gamma)$  total time.

Finally, it remains to prove that one can distinguish between the former two cases in  $O(\gamma)$ . Recall that u and v have ordinates 1 and -1, respectively, both lying on the y axis. Let C be the set of ordinates of the intersection points of axis y and the line segments generating the polygonal chain. If  $C \cap [0,1] \neq \emptyset$ , than segment [u, v] is neither entirely inside nor entirely outside of the polygon. Otherwise if  $|C \cap (1, +\infty)|$ is an even or odd number, than [uv] is outside or inside the polygon, respectively. Note that all the required operations can be done in  $O(\gamma)$ .



Balázs Vass received his MSc degree in applied mathematics at Eötvös Loránd University (ELTE), Budapest in 2016. He started his PhD studies in informatics in the same year on the Budapest University of Technology and Economics (BME). His research interests include networking, survivability, combinatorial optimisation, and graph theory. He received the Best Paper Award of NaNa'16, and the Best-in-Session Presentation award on INFO-COM'18. He was an invited speaker of COST RECODIS Training School on Design of Disaster-

resilient Communication Networks in 2019.



Erika R. Bérzci-Kovács received the M.Sc. Degree in Mathematics and the Ph.D. degree in Applied Mathematics from the Eötvös Loránd University (ELTE), Budapest, in 2007 and 2015, respectively. She is currently a senior lecturer at the Department of Operations Research, ELTE, and she is a member of the MTA-ELTE Egerváry Research Group on Combinatorial Optimization. Her research interests are combinatorial optimization and network coding. She was a recipient of NaNA 2016 Best Paper Award.



János Tapolcai received the MSc degree in technical informatics and the PhD degree in computer science from the Budapest University of Technology and Economics (BME), Budapest, in 2000 and 2005, respectively, and the D.Sc. degree in engineering science from the Hungarian Academy of Sciences (MTA) in 2013. He is currently a Full Professor with the High-Speed Networks Laboratory, Department of Telecommunications and Media Informatics, BME. He has authored over 150 scientific publications. He was a recipient of several Best Paper

Awards, including ICC'06, DRCN'11, HPSR'15, and NaNa'16. He is a winner of the MTA Lendület Program and the Google Faculty Award in 2012, Microsoft Azure Research Award in 2018. He is a TPC member of leading conferences, e.g., IEEE INFOCOM 2012-, and the general chair of ACM SIGCOMM 2018.