Eötvös Loránd University

Faculty of Science

Balázs Vass

A Combinatorial Geometric Approach for Network Attack Analysis

Applied Mathematics M. Sc. Thesis

Advisors:

Erika Bérczi-Kovács
ELTE
Department of Operations Research

Attila Kőrösi
BME
Department of Telecommunications and Media Informatics

Budapest, 2016

Acknowledgements

I would like to express my deep gratitude to Erika Bérczi-Kovács, Attila Kőrösi and János Tapolcai, my research supervisors, for their patient guidance, enthusiastic encouragement and useful critiques of this research work.

I am also very thankful to László Gyimóthi for implementation used in the thesis.

Contents

Abstract

Current backbone networks are designed to protect a certain pre-defined list of failures, called Shared Risk Link Groups (SRLG). The list of SRLGs must be carefully defined, because leaving out one likely failure event will significantly degrade the observed reliability of the network. In practice, 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. A common belief is that the number of possible regional failures can be very large, thus simply listing them as SRLGs is not a viable solution. In this study we show the opposite, and provide efficient algorithms for enumerating two types of maximal failures as SRLGs. Both types of failures are caused by disasters having shape of disks. One type of failures is caused by node excluding disks, the other are caused by disks with a fixed radius. According to some practical assumptions and simulations these lists are surprisingly short with $O(|V|)$ SRLGs in total, and can be computed in low-order polynomial time.

Chapter 1

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 are set of links that are 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 links 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 apply numerous techniques to handle failures. 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 the protection of which the network is already pre-configured for. 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 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 caused a power outage that silenced 46% of the network in the New York area [4]. These type of failures are called *regional failures* which are simultaneous failures of nodes/links located in specific geographic areas [1–9]. The number of possible regional failures can be very large, thus simply listing each of 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 this thesis we propose a solution to this problem with two combinatorial geometric techniques that can significantly reduce the number of possible failure states that should be added as an SRLG to cover all regional failures.

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 circles with given size (also called disks) [12]. In our study we assume the regional failure is caused by a geometric shape having a shape of disk. The number of SRLGs is 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 with edges drawn as line segments.

2. The geometric shapes causing the regional failures are assumed to be disks.

As a practical view of point we mainly want to generate quickly a short list of maximal failures caused by relatively 'small' disks. Depending on the exact interpretation of word 'small', we offer two very different solutions of the problem.

One possibility of defining smallness is based on bounding the size of disks causing the failures, thus we can speak about **bounded radius disk failure**s and **unbounded radius disk failure**s. We say that a disk is small if its radius is smaller than a parameter r. It is quite natural to chose this interpretation. However, this way we cannot control the number of nodes covered by disks. Thus the traffic matrix can suffer heavy changes, since if a node becomes covered by a disk, it can no longer send traffic. This has a network wide effect. Thus we offer another interpretation of smallness too.

Another possibility of defining smallness is based on bounding the number of nodes covered by the disks. We could say that a disk failure is small iff it covers at most k nodes. In fact, due to the previously mentioned effect of node deleting on the traffic matrix we treat disks covering k nodes separately from those covering $l \neq k$ nodes. Thus we speak about **disk failures including arbitrary number of nodes** and **disk failures including k nodes** ($k \in \mathbb{N}$). This way we control the number of nodes covered by disks. However, we cannot control the size (radius) of disks, which is a disadvantage of this interpretation.

Both of the previously described definitions have their advantages and disadvantages, thus studying both cases seems equivalently useful. And what about bounded radius disk failures including k nodes, etc.? The next sentence summarises our questions, where one substitute either *(i)* or *(ii)* and either *(a)* or *(b)*.

We may want to investigate on

(i) unbounded radius		(a) including arbitrary number of nodes.
	disk failures	
(ii) bounded radius		(b) including k nodes.

Case *(i)/(a)* is not very interesting: a sufficiently large disk covers graph G, thus the list of SRLGs will have a single element containing all the edges of G.

Case *(i)/(b)* is much more exciting. In Chapter 2, according to some practical assumptions we prove that for node excluding (0 nodes including) disks our list is $O(|V|)$ long, and can be calculated in $O(|V|\log|V|)$ time for typical backbone networks. Chapter 2 contains results from our paper about to appear *Shared Risk ink Group Enumeration of Node Excluding Disaster Failures* [13], which won the best paper award on conference *NaNA*, and contains improved results of our paper *Listing Regional Failures* [14] presented on *Mesterpróba*, a conference for students oranized on Budapest University of Technology and Economics. Our poster *Shared Risk Link Groups of Disaster Failures* [15] with these results was also presented on conference *INFOCOM*.

In our paper *Shared Risk Link Groups of Disaster Failures: The Single Node Failure Case* [16] submitted to conference *RMDN* is proven that for single node including disks ($k = 1$) the list of SRLGs is still $O(|V|)$ long, and still can be calculated in $O(|V|\log|V|)$ time. These results are not included in the thesis.

The case of disks including multiple nodes ($k \geq 2$) can be also solved in polynomial time, thus we plan to write a paper with these results too.

Case *(ii)/(a)* is also interesting. In Chapter 3 we show that the length and calculation complexity of list of maximal SRLGs calculated in this case is bounded by a low-order polynomial of $|V|$. According to our simulations in case of backbone networks its length is $O(|V|)$, and can be calculated in $O(|V|^2)$ time. Results presented in Chapter 3 are based on our work *Hálózatok legfennebb r sugárban ható hibáinak felsorolása* [17] presented at a *Scientific Students' Associations* conference oranized by the Mathematical Institute of the Eötvös Loránd University.

Case *(ii)/(a)* has reasonable motivation and questions too. For this time, it is left for further study.

Let us remind the motivation of the works. Using SRLG lists generated by our proposed algorithms, network operators can design their networks to be protected against regional failures. Backbone networks designed according to our new failure models should have higher reliability, and leave much less failures to be recovered by the convergence of higher layer intra-domain routing protocols (IS-IS, OSPF) in the next few seconds, minutes or hours after the failure. We believe the thesis fills the gap between the conventional SRLG based pre-planned protection and regional failures.

The structure of the thesis is the following. In Chapters 2 and 3 we present our results, and in Chapter 4 we briefly mention some related works. In Chapter 5 we draw our conclusions, and finally, in 6 we present our plans for the future.

Chapter 2

Listing maximal failures caused by node excluding disks

In this chapter the list of maximal failures caused by node excluding disks are calculated. This list is proven to be short in case of backbone networks, and can be calculated quickly.

2.1 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 [18]. 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 [19]. 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. 2.1 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 thesis.

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

Both on the right and left side of l let us take a complete bipartite graph called $K^r_{j,j}$ and $K^l_{j,j}$. We locate the points of K^r and K^l carefully as follows. For both $K^r_{j,j}$ and $K^l_{j,j}$ 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 $v \in V\setminus\{v_1,\dots,v_c\}$ $|d(v,l)-d(v_1,v_2)/2| \le \varepsilon$ for a very small $\varepsilon > 0$. Though the top partites are much further from v_1 than $d(v_1,v_c)$, they are much closer to them as the bottom partites. The graph is pictured in figure 2.1. 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. 2.1 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 $i \in \{1,\dots,k-1\}$. 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 $|E_\triangledown| \le 3n-6$ and $|\mathcal{M}_0^{u,v}|$ is $O(m)$ (corollary of correctness of Algorithm 3). □

Corollary 7. *The graph illustrated on Fig. 2.1 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 $D_\triangledown = (E_\triangledown, V)$. After that for every $\{u,v\} \in E_\triangledown$ 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 2.2.

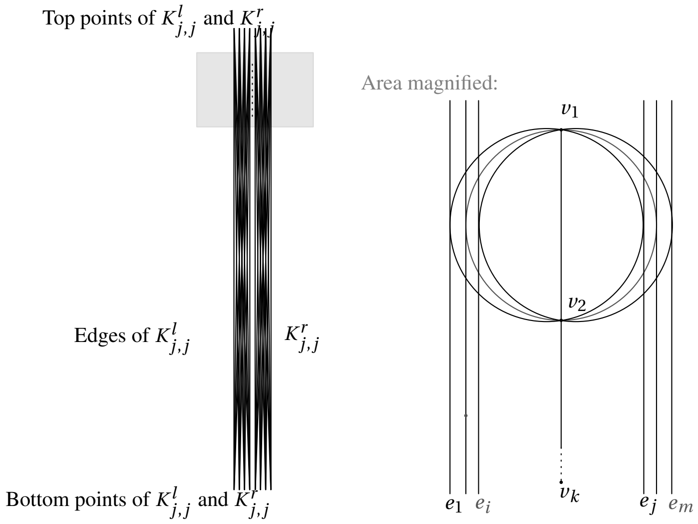

Figure 2.1: Sketch of a graph family with $\Theta(n^3)$ maximal regional link failures

2.2 The Algorithm

Consider the Delaunay triangulation $D_\triangledown = (V, E_\triangledown)$ and the set T_0 of Delaunay triangles given by their vertices. Since $D_\triangledown$ 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^t_0 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 $C^{u,v}_0$ 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 2.2).

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

Theorem 8. *$\mathscr{M}_0$ 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 [18]. Sets T_0 and T^t_0 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 $\{u, v\} \in E_\triangledown$ in $O(n\theta_0^2)$ time according to

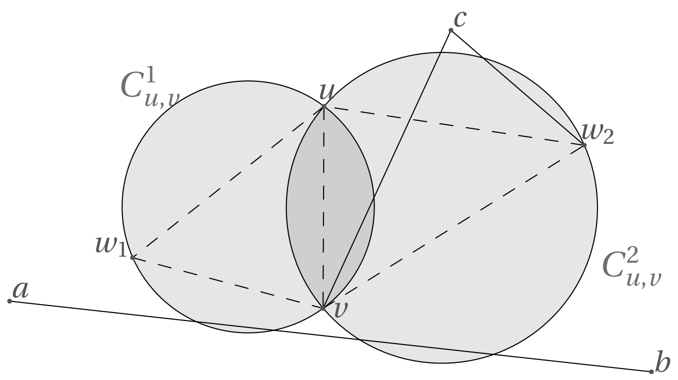

Figure 2.2: Example on a Delaunay-edge $\{u, v\}$ with w_1, w_2, $C^1_{u,v}$ and $C^2_{u,v}$. Here $E^1_{u,v} = \{\{a,b\}\}$, $E^2_{u,v} = \{\{a,b\},\{c,w_2\}\}$, $E^3_{u,v} = \{\{c,v\}\}$.

Lemma 11.

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

Finally, in line 4 $\mathcal{M}_0$ is calculated from lists $\mathcal{M}_0^{u,v}$ 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 $O(\log n)$, $\mathcal{M}_0$ 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 $\mathcal{M}_0$ of maximal single regional failures.

begin

1 $E_\triangledown, T_0, T_0^t (t \in T_0) \leftarrow$DELAUNAY$(V)$;

2 $E^i_{u,v} (i \in \{1,2,3\}, \{u,v\} \in E_\triangledown) \leftarrow$ GETEDGESETS$(V, E, E_\triangledown, T_0)$;

3 $\mathcal{M}_0^{u,v} (\{u,v\} \in E_\triangledown) \leftarrow$ GENERATE $(V, E, E_\triangledown, E^i_{u,v} (i \in \{1,2,3\}, \{u,v\} \in E_\triangledown))$;

4 $\mathcal{M}_0 \leftarrow$ ELIMINATEREDUNTANTS$(\mathcal{M}_0^{u,v}, \forall \{u,v\} \in E_\triangledown)$;

return $\mathcal{M}_0$

2.2.1 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 $|E_\bigtriangledown| \leq 3n-6$. 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

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

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 $m \leq (2n-5)\theta_0$. □

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. 2.1). 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 $|\mathscr{M}_0| \leq cn$.

2.2.2 Algorithm 2 (Method Getedgesets)

Lemma 11. *Algorithm 2 computes sets* $E^i_{u,v}$ *for all* $\{u,v\} \in E_\bigtriangledown$ *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 $C^i_{u,v}$, $i \in \{1,2\}$ 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, $E^2_{u,v}$ 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 (having nonempty part in) circumcirlce of Delaunay tirangle t, in lines 10 - 12 we also get set $E^i_{u,v}$.

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 $E^i_{u,v}$ sets in similar complexity.

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

□

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($t_i, \{a,b\}$)

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. 2.3).

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 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. □

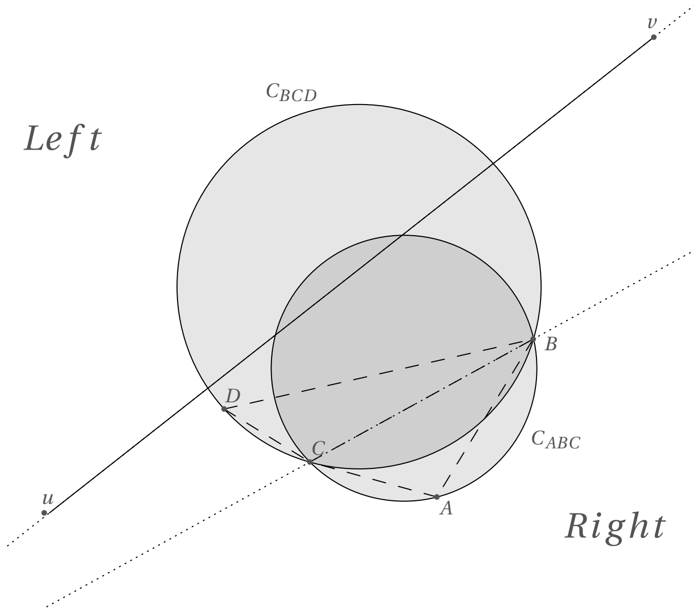

Figure 2.3: Illustration for proof of Lemma 12

2.2.3 Algorithm 3 (Method Generate $\mathcal{M}_0^{u,v}$)

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

Proof. Proposition 14 shows the correctness of Algorithm 3.

We assume line 2 runs in constant time. This means that for a given $\{u, v \in E_\triangledown\}$ 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 $M_0^{u,v}$ is calculated in $O(\theta_0^2)$ time. Since $|E_\triangledown| \leq 3n-6$, this gives an overall complexity $O(n\theta_0^2)$.

□

Proposition 14. *It can be shown that if a $k \in C_0^{u,v}$ 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 $\{u, v\} \in E_\triangledown$, $|\mathcal{M}_0^{u,v}|$ is $O(\theta_0)$.*

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

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

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

2.2.4 Algorithm 4 (Method Eliminateredundants)

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

Algorithm 4: ELIMINATEREDUNTANTS

Input: $\mathscr{M}_0^{u,v}$ for all $\{u,v\} \in E_\triangledown$
Output: $\mathscr{M}_0$
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** $f_{u,v} \in \mathscr{M}_0^{u,v}$ **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**
                        $\mathscr{M}_0^{w,z} \leftarrow \mathscr{M}_0^{w,z} \setminus f_{w,z}$
                    else
                        if $f_{u,v} \subset f_{w,z}$ **then**
6                             $\mathscr{M}_0^{u,v} \leftarrow \mathscr{M}_0^{u,v} \setminus f_{u,v}$
7     $\mathscr{M}_0 \leftarrow \bigcup_{\{u,v\} \in E_\triangledown} \mathscr{M}_0^{u,v}$;
8   **return** $\mathscr{M}_0$

Proof. We can prove the correctness of Algorithm 4 by checking that it eliminates all globally non-maximal elements of $\mathscr{M}_0^{u,v}$ and leaves exactly one copy of each element of $\mathscr{M}_0^{u,v}$ 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 $O(n\theta_0^3\tau)$.

□

Chapter 3

Listing maximal failures caused by disks of radius r

3.1 Problem and a primitive solution

Problem:

We have a real number $r > 0$ and a graph $G = (V, E)$ embedded in the $2D$ plane. The edges of G are embedded as line segments, and a failue F is a subset of the set of edges. In this chapter we consider only failures for which is true that the distance of every edge $e \in F$ is at most r from a centre point.

We model the failure caused by disk c as every edge having part inside c is erased from the graph.

We do not track the set of failed nodes since a node v fails or becomes isolated iff all edges incident to v fail, therefore listing the failed nodes beside listing failed edges does not give us additional information from the view of point of connectivity.

We say an edge e is covered by a disk d iff $e \cap d \neq \emptyset$. It is assumed that every disk in the plane with radius r covers at most σ edges. This way for every failure F caused by a disk with radius r, $|F| \leq \sigma$.

Let $n := |V|$ and $m := |E|$. We assume $n \geq 2$ and G is connected.

Definition 17. *Let $\mathscr{S} \subseteq 2^E$ be the set of exclusion-wise maximal failures caused by disks with radius at most r.*

Proposition 18. *$\mathscr{S}$ is a Sperner system.* □

Proposition 19. $\emptyset \notin \mathscr{S}$. □

Our aim is to determine set $\mathscr{S}$. At first glance it is not clear that the cardinality of $\mathscr{S}$ is 'small'. We will prove polynomial upper bounds on $|\mathscr{S}|$, and we will show that $|\mathscr{S}|$ is $O(n)$ in practice.

We observe that if σ is $O(\log n)$ then it is possible to calculate the down set of $\mathscr{S}$ in polynomial time, which is useful while planning networks (see the introduction). Thus we define:

Definition 20. *Let $\mathscr{H}$ be the downward closed set of $\mathscr{S}$ minus the empty set.*

Proposition 21. $\mathscr{H} = \{\emptyset \neq H \subseteq E|$ *there exists a disk c which covers all the edges in $H\}$.*
□

Because $|\mathscr{S}|$ can be up to 2^σ times smaller than $|\mathscr{H}|$, and $\mathscr{S}$ is a compact description of $\mathscr{H}$, we concentrate on calculating $\mathscr{S}$. If $|\mathscr{S}|$ is $O(logn)$, it is easy to compute $\mathscr{H}$ from $\mathscr{S}$ in polynomial time.

Definition 22. *We say a disk covers a set of edges E', if it covers all the edges in E'.*

Definition 23. *Let a disk c be* **smaller** *than disk c', if c has a smaller radius than c', or if they have equal radius and the centre point of c is lexicographically smaller than the centre point of c'.*

Definition 24. *Let $E' \subseteq E$ be a nonempty set of edges. We denote the smallest disk among the disks covering E' by $c_{E'}$ and we say $c_{E'}$ is the* **smallest covering disk** *of E'.*

Proposition 25. *For every $H \in \mathscr{H}$, c_H is a disk with radius at most r covering H.* □

Theorem 26. *For every $H \in \mathscr{S}$ with smallest covering disk c_H there exist 3 (not necessarily different) edges e_1, e_2, e_3 for which the smallest covering disk of set $\{e_1, e_2, e_3\}$ is also c_H.*

For every nonempty $H' \subseteq H$, the set of edges covered by $c_{H'}$ is subset of or equivalent to H.

Theorem 26 would be trivial if smallest covering disks were defined on sets of nodes because a triplet of nodes define a circle. We show that this property holds for edges (considered as line segments) too.

Proof. Let P_{c_H} is the set of points where edges which are located exterior to c_H aside from a singe point of them intersect c_H. Formally, $P_{c_H} = \{p \in e \cap c_H || e \cap c_H| = 1\}$.

It is easy to see that $P_{c_H} \neq \emptyset$ since if it would be empty, we could reduce the radius of c_H while it remains covering disk, which contradicts the fact that c_H is the smallest covering disk of set H.

If $|P_{c_H}| = 1$, then c_H is a single point u, and every element of H goes through it. In this case, c_H is the smallest covering disk of an edge from H having u as an endpoint.

If $|P_{c_H}| = 2$, then say $P_{c_H} = \{u, v\}$. In this case c_H has $[u, v]$ as diameter, and is the smallest covering disk of a set of two edges having u and v as endpoints.

Finally, if $|P_{c_H}| \geq 3$, we argue as follows. All the points of P are on the same circle c (the boundary of c_H). For any point pair $p_1, p_2 \in P_{c_H}$, $p_1 \neq p_2$ there is no disk d which is smaller than c_H and covers set P, because in this case c_H would be different. This means that for p_1 and p_2 there exists a $p_3 \in P$ which prevents the existence of d. Due to this fact the smallest covering disk $c_{H'}$ of edge set $H' = \{e_1, e_2, e_3\}$ (where e_i contains p_i) is not smaller than c_H. On the other hand, $c_{H'}$ is not bigger than c_H either since $H' \subseteq H$. This way it is proven that there exist 3 edges with the same smallest covering disk as whole set H.

Since all the cases were checked, the proof is finished. □

Corollary 27. $|\mathscr{S}| \leq \binom{m}{3} + \binom{m}{2} + m = \frac{m^3}{6} + \frac{5m}{6}$. □

Theorem 28. *The smallest covering disk of 1, 2 or 3 edges can be calculated in $O(1)$ time.*

Again, Theorem 28 would be trivial in case of nodes instead of edges. In Subsection 3.3.1 we prove it for edges.

Since the smallest covering disk of a triplet of edges can be calculated in $O(1)$ time, we could solve the problem by processing $O(m^3)$ triplet of edges. However, we will achieve better upper bounds on running time and $|S|$ with the help of some further observations.

Definition 29. *Let X be the set of points p which are not in V and there exist at least 2 edges crossing each other in p. Let $x = |X|$.*

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

Lemma 30. *The number of edges in G is $O(n + x)$. More precisely, $m \leq 3n + x + 6$.*

Proof. Let $G'(V \cup X, E')$ be the planar graph obtained from dividing the edges of G at the crossings. Since every crossing enlarges the number of edges at least with two, $|E'| \geq m + 2x$. On the other hand, $|E'| \leq 3(n + x) + 6$ since G' is planar. Thus $m \leq |E'| - 2x \leq 3n + x + 6$. □

Corollary 31. *Parameter σ is $O(n + x)$. More precisely, $\sigma \leq m \leq 3n + x + 6$.* □

According to Section 3.2, it is enough to process the edge triplets in the neighbourhood with radius $3r$ of every point in $V \cup X$. We will see that this way $O(n\sigma^3)$ edge triplets have to be processed in practice.

It is possible that smallest covering cycles are found even while processing the above mentioned disks with radius $3r$. The number of these 'bad guesses' can be reduced by partitioning the disks with radius $3r$, this speed up has a constant factor. These results are not included in the thesis.

3.2 Improvement by disks with radius $3r$

Theorem 32. *For every failure $H \in \mathcal{H}$ there exists a disk covering H with centre point in the $2r$ radius neighbourhood of a point from $V \cup X$.*

With other words, let $E' \subseteq E$ be a set of edges which can be covered with a disk with radius r. According to Theorem 32, there exists a point $y \in X \cup V$ and a disk c' with radius r such that $c' \in B(y, 3r)$ and c' covers E', where $B(y, 3r)$ denotes the $3r$ radius neighbourhood of y. The proof of the theorem can be found in Section 3.3.

We are ready to word our first algorithm.

Algorithm 5 summarises our method. Edges e_1, e_2 and e_3 are not necessarily different. If they concur, we calculate the smallest covering disk of one or two edges. Algorithm 6 and 7 calculates the smallest covering disk of 2 and 3 edges, respectively.

Let denote D^y the set of smallest covering disks calculated by Algorithm 5 in case of the set of edges E^y incident to $B(y, 3r)$ for a $y \in V \cup X$.

Claim 33. $|D^y| \le \binom{15\sigma}{3} + \binom{15\sigma}{2} + 15\sigma^3 = \frac{1125\sigma^3}{2} + \frac{25\sigma}{2}$.

Proof. Figure 3.1 shows a covering of a disk with radius $3r$ with 15 disks with radius r, thus $|E^y| \le 15\sigma$.

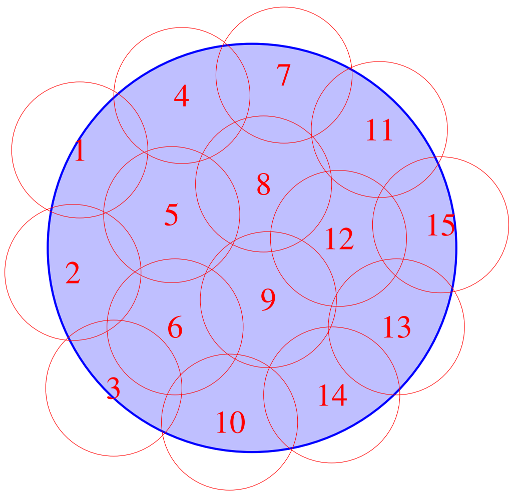

Figure 3.1: A disk with radius $3r$ can be covered with 15 disks with radius r

Smallest covering cycles in D^y are determined by 1, 2 or 3 edges in E^y, and all edges, edge couples and edge triplets determine a single smallest covering disk. From these we can derive the claim, where we consider an edge as an edge triplet with 3 coincident edges, and an edge couple as an edge triplet with 2 coincident edges.

□

Corollary 34. $|\mathcal{S}| < 562.5(n + x)\sigma^3 + 12.5(n + x)\sigma$. □

Algorithm 5: Calculating $\mathscr{S}$ with help of disks with radius $3r$

```
Input: G(V,E), r
Output: Set 𝒮 of maximal failures caused by disks with radius at most r
begin
    X ← EDGECROSSINGS(G);
    𝒮 ← ∅;
    for y ∈ V ∪ X do
        E^y = {e_1, ..., e_l} ← EDGESIN(B(y,3r), E);
        for i ∈ {1, ..., l} do
            for j ∈ {i, ..., l} do
                for k ∈ {j, ..., l} do
                    c ← SMALLESTCOVERINGDISK(e_i, e_j, e_k);
                    if radius of c ≤ r then
                        E_c ← EDGESIN(c, E^y);
                        if E^y is maximal in 𝒮 then
                            𝒮 ← 𝒮 ∪ E^y;
                            Eliminate non-maximal elements of 𝒮;

Function EDGECROSSINGS(G(V,E))
    X ← ∅;
    for e_i = {u_i, v_i} ∈ E do
        for e_j = {u_j, v_j} ∈ E, e_i ≠ e_j do
            if (u_i u_j) ∩ (v_i v_j) ≠ ∅ then
                X ← X ∪ e_i ∩ e_j
    return X

Function EDGESIN(c, E')
    E^y ← ∅ for e ∈ E' do
        if e ∩ c ≠ ∅ then
            E^y ← E^y ∪ {e}
    return E^y
```

Theorem 35. *Algorithm 5 has an $O((n+x)^2\sigma^7)$ time complexity.*

Proof. In line 1 we can calculate the edge crossings in $O(m^2)$ time, which is $O((n+x)^2)$ by Lemma 30.

Computations in Lines 3-9 have to be done for all $y \in V \cup X$, which means an $O(n+x)$ multiplying factor in the next paragraph.

For a given $y \in V \cup X$, E^y is calculated in $O(n+x)$ time in Line 4. We will see in Section 3.3 that a smallest covering disk in Line 5 can be calculated in constant time. Since we have to compute $O(\sigma^3)$ smallest covering disks, this means $O(\sigma^3)$ computation. Line 6 runs in constant time. We can find E_c in $O(\sigma)$ complexity, $O(\sigma^3)$ times in Line 7. Since in Lines 8 and 9 we have to make $O((n+x)\sigma^3)$ comparisons with complexity $O(\sigma)$ in case of a smallest covering disk, while working on a single $y \in V \cup X$, maintaining $\mathcal{S}$ has $O((n+x)\sigma^7)$ complexity.

We can deduct that Algorithm 5 has $O((n+x)^2\sigma^7)$ time complexity. □

Since we are interested mainly in protecting failures which are small compared to the size of networks, and backbone networks have few edge crossings, it is useful to word Corollary 36, which says in this case $\mathcal{S}$ can be calculated in $O(n^2)$ time.

Corollary 36. *If x is $O(n)$ and σ is upper bounded by a constant, then Algorithm 5 runs in $O(n^2)$ time.*

Claim 37. *If Conjecture 44 holds ($|\mathcal{S}|$ is $O(n+x)$), then Algorithm 5 has an $O((n+x)^2\sigma^4)$ time complexity.* □

Proof. The proof can be made by alternating proof of Theorem 35. □

3.3 Proofs for Chapter 3

3.3.1 Proof of Theorem 28

Proof. We prove Theorem 28 by giving algorithms calculating the smallest covering disks in constant time, which is a benefit also from algorithmic view of point.

The expression 'smallest covering disk' will be often abbreviated as SCD.

As a first step, In Lemma 39 we figure out the way of calculating of SCDs for sets of lines and points with at most 3 elements. This lemma is a simpler version of the theorem, which still contains the main ideas. The theorem is proven by giving algorithm for computing SCDs of at most 3 edges. Here not everything will be detailed.

For simplicity in this proof we do not take in count parallel lines and we assume that in graph G there are no parallel edges. Cases rising from parallelism can be handled similarly to those written here.

The SCD of a set of lines and nodes can be defined similarly to the smallest covering disk of a set of edges:

Definition 38. *The smallest covering disk of a set L of points and lines not consisting only of parallel lines is the smallest disk covering all the elements of L.*

Lemma 39. *Smallest covering disk of a set of points and nodes with at most 3 elements can be calculated in $O(1)$ time.*

Proof. We start the proof of Lemma 39 with the following observation.

Observation 40. *If we are searching smallest covering disk for a point, it will have the point on the boundary. If we are searching for smallest covering disk for line, the line will be the tangent of the disk. In other case the point or line can be omitted while searching the smallest covering disk.*

More formally, for every element e of a set L of lines and points the smallest covering disk c_L of the set L is either intersecting e in exactly 1 point, or $c_L = c_{L\setminus\{e\}}$. □

SMALLESTCOVERINGDISK(u, v) *of 2 points:* Disk having $[uv]$ as diameter. (see Fig. 3.2)

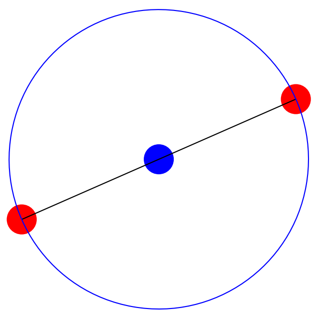

Figure 3.2: Smallest covering disk of 2 points

Comment 41. *The smallest covering disk of and edge and a point or of two edges will be calculated with the help of the previous case.*

SMALLESTCOVERINGDISK(u, v, w) *of 3 points:* Disk having u, v and w on the boundary (see Fig. 3.3). If u, v and w are collinear, we can omit the inner point and we have to compute the smallest covering disk of 2 points.

SMALLESTCOVERINGDISK$(\overline{e}, \overline{f}, \overline{g})$ *of 3 not collinear lines:* Incircle of triangle generated by lines $\overline{e}$, $\overline{f}$ and $\overline{g}$ (see Fig. 3.4).

Comment 42. *The excircles of the triangle are not suitable.*

SMALLESTCOVERINGDISK$(u, \overline{e}, \overline{f})$ *of 1 point and 2 not parallel lines:* The lines will be tangents of the SCD due to Obs. 40, thus the centre point of the disk is on on of the angle bisectors determined by the lines. Lines $\overline{e}$ and $\overline{f}$ cut the plane into four parts. The

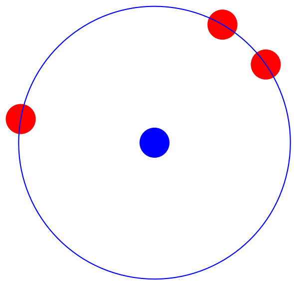

Figure 3.3: Smallest covering disk of 3 points

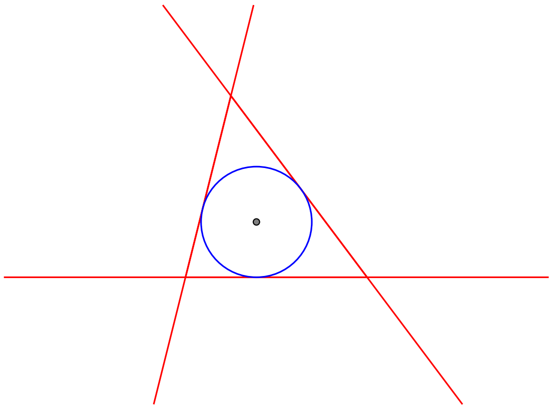

Figure 3.4: Smallest covering disk of 3 lines

centre point of the disk will be in the same part p_c as u. Let the bisector going through this part be $\overline{g}$.

The set of points equidistant from line $\overline{e}$ and point u form a parabola p. We know that line $\overline{g}$ and parabola p may intersect each other in at most 2 points. If the intersection is empty, we can omit the line closer to u, thes search smallest covering disk for 1 point and 1 line. If there is only one point in the intersection, than this will be the centre point of the SCD. If the intersection consists of 2 points, we will keep the smaller disk generated by these points. See Fig. 3.5 for illustration.

SMALLESTCOVERINGDISK$(u, v, \overline{e})$ *of 2 points and 1 line:* The centre point of the disk is on the line perpendicular to $[u, v]$ goin through the midpoint of $[uv]$. The set of points equidistant from line $\overline{e}$ and point u is a parabola p. Starting from this point this case is similar to the last one. See Fig. 3.6 for illustration.

There are still some other cases left. The SCD of a point is the point itself; for a single line it is not defined (see Def. 38); for a point p and a line l it is the disk having $[pp_1]$ as diameter, where p_1 is the intersection point of l and the line perpendicular to l going through p; and finally for 2 non-parallel line it is the intersection point of them.

In every cases we can determine the centre point of the SCD via solving exact linear

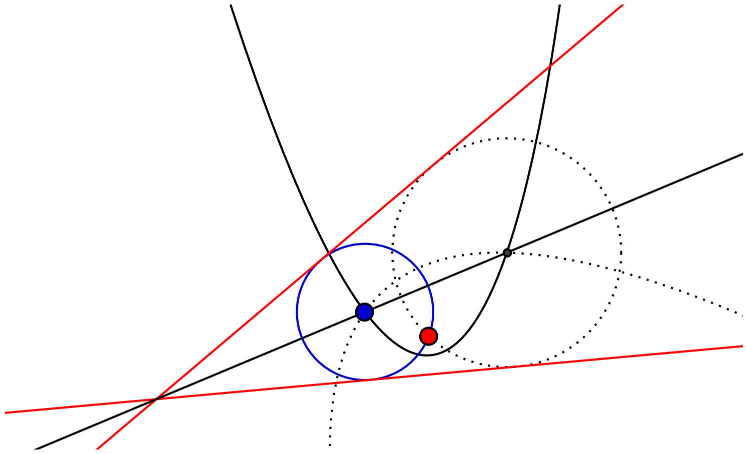

Figure 3.5: Smallest covering disk of 1 point and 2 lines

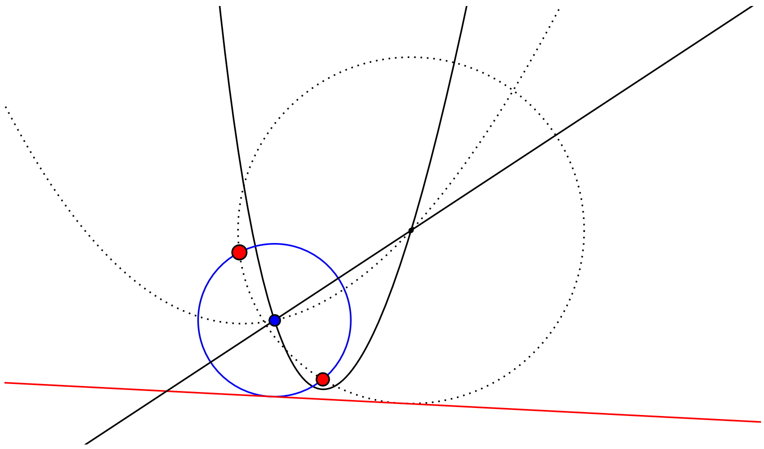

Figure 3.6: Smallest covering disk of 2 points and 1 line

or quadratic formulas, which can be done in constant time. Knowing the centre point we can easily determine the radius of the SCD. Thus it is easy to see that in every case we can determine the SCD in constant time.

Since we took in count all the possibilities, we finished the proof of Lemma 39.

Cases with parallel lines could be handled similarly. □

The proof of the theorem can be completed similarly to Lemma 39's proof, which gives us a taste of computing SCDs.

We will give separate algorithms on computing the smallest covering disk of 1, 2 and 3 edges.

Definition 43. *Line going through endpoints of edge e is called the* ***line of edge*** *e.*

Trivially, the SCD of a single edge e is the lexicographically smaller endpoint of e.

In case of two edges e_1 and e_2 we consider the followings. If $e_1 \cap e_2 \neq \emptyset$, then the intersection will be the SCD. Else, the SCD will go through on at least 1 endpoint v of an edge, let us say, e_1. If the SCD of v and line of e_2 intersects e_2, that will be the SCD of the two edges too. Else, it will be the SCD of v and the endpoint of e_2 closer to v. See Algorithm 7.

Algorithm 6: Smallest covering disk of 2 edges

```
Input: Edges e1 = {u1, v1} and e2 = {u2, v2};
Output: Smallest covering disk of e1 and e2;
begin
    for v ∈ {u1, v1, u2, v2} do
        Let edge not containing v be denoted by e;
        Let intersection of line of e and line perpendicular to e going through v be t;
        if t ∈ e then
            c ←SMALLESTCOVERINGDISK(v, t)
        else
            Let u be the endpoint of e closer to v;
            c ←SMALLESTCOVERINGDISK(v, u)
        if c is smaller than the former ones then
            note c;
    return The smallest disk generated;
```

When we want to compute the SCD of 3 edges, say, of triplet $\{e_1, e_2, e_3\}$, things get more complicated. The SCD can intersect all 3 edges in inner point or in one of the endpoints, which means $3^3 = 27$ cases. In every case we determine the SCD of the set L of points and lines of edges. We say an SCD is allowed iff it intersects not only the lines of edges in L, but it intersects the edges in L themselves (which are part of the lines) too. Algorithm 7 computes the SCD of 3 edges.

Proof of Theorem 28 can be completed via checking the correctness of Algorithm 6 and 7.

□

3.3.2 Proof of Theorem 32

Proof. Since $H \in \mathcal{H}$, c_H (smallest covering disk of H) has radius at most r. It is enough to show that there exists a $y \in V \cup X$ such that the distance between the centre point of c_H and y is at most $2r$, because in this case we can blow up c_H into a disk with radius r inside $B(y, 3r)$.

Algorithm 7: Smallest covering disk of 3 edges

Input: Edges e_1, e_2 and e_3.;
Output: The smallest allowed disk generated.;
for $g_1 \in \{u_1, v_1, (u_1, v_1)\}$ **do**
  for $g_2 \in \{u_2, v_2, (u_2, v_2)\}$ **do**
    for $g_3 \in \{u_3, v_3, (u_3, v_3)\}$ **do**
      $c \leftarrow$SMALLESTCOVERINGDISK(g_1, g_2, g_3);
      if *c is smaller than the former ones* **then**
        note c;
return The smallest allowed disk generated.

If $|H| = 1$, then trivially the centre point of c_H will be the lexicographically smallest point of the edge in H.

If $|H| \geq 2$, then the smallest covering disk of H has 2 or 3 edges on its boundary.

If c_H has 2 points on the boundary, then according to the method of calculating the smallest covering cycle, c_H intersects one of the edges in H in an endpoint v. This point v and blowing up c_H into a disk with radius r while keeping the centre point is suitable for us.

If c_H has 3 (e_1, e_2, e_3) edges on the boundary, then if there exists an edge which is intersected by c_H in an endpoint, then v and c_H suits us as in the previous case.

If c_H intersects all 3 edges in inner points then triangle ABC_Δ determined by $\overline{e}_1, \overline{e}_2$ and $\overline{e}_3$ is not degenerate (see Fig. 3.7). We assume w.l.o.g. that its largest angle is $\widehat{BAC}$. This means the radian measure of $\widehat{BAC}$ is in interval $[\frac{\pi}{3}, \pi)$.

This way $\sin(\widehat{IAC}) \geq \frac{1}{2}$ since AI is the angle bisector of $\widehat{BAC}$.

$$|AI| = \frac{|LI|}{\sin \widehat{IAC}} \leq \frac{r}{\frac{1}{2}} = 2r.$$

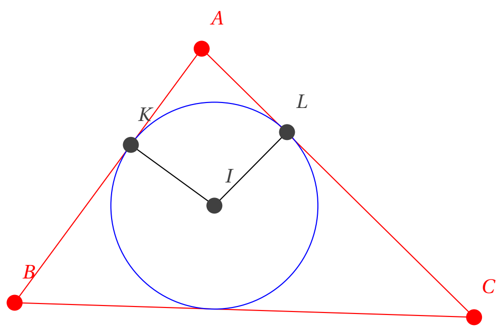

Figure 3.7: Illustration for proof of Theorem 32

If $A \in V \cup X$, then we are done. Else there exists a node v in $[AK] \cup [AL]$. Clearly, $d(vI) \leq 2r$, this way we can get a suitable disk with radius r from v and c_H. □

3.4 Simulations

In this chapter first we visualise set $\mathscr{S}$ on some real backbone networks then we present our test results included in a table.

The red circles go through 2 or 3 nodes. Disks covering 1 node are represented as red disks with radius r. Green disks have 3 edges on the boundary. All other disks are violet. The size of r is the half of the length of the shortest edge in every network on Figures 3.8-3.10 and Table 3.1.

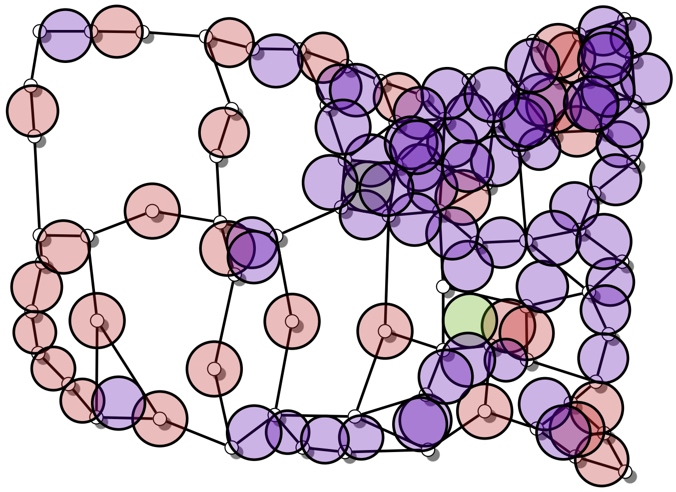

Figure 3.8: American network ($|79\_optic\_nfsnet|$)

We can see that $|\mathscr{S}|$ never exceeds $2n$ in Table 3.1. This gives us the natural question that what are the relations between $|\mathscr{S}|$, r, σ and n or $n+x$. Clearly, for sufficiently small r, $|\mathscr{S}| = n+x$, because all the maximal failures can be caused by a disk which is a small neighbourhood of a node or edge crossing. On the other hand, for sufficiently large r, $|\mathscr{S}| = 1$, because there will exist a disk covering all the edges of G. While determining $|\mathscr{S}|$ for numerous values of r in all our networks we witnessed that $|\mathscr{S}|$ is around $(n+x)$ for any values of r. We show some of our results on the Italian optic network in Figures 3.11 and 3.12 and Table 3.2.

Despite that we have to investigate deeper this question, we conjecture the following.

Conjecture 44. *In practice, $|\mathscr{S}|$ is $O(n+x)$ for any value of r.*

Here 'in practice' means that for those backbone networks we have, the conjecture is true.

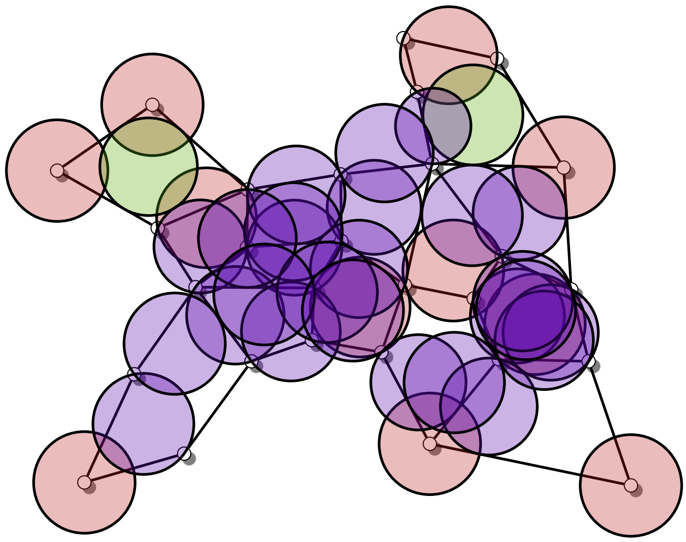

Figure 3.9: European network ($|28\_optic\_eu|$)

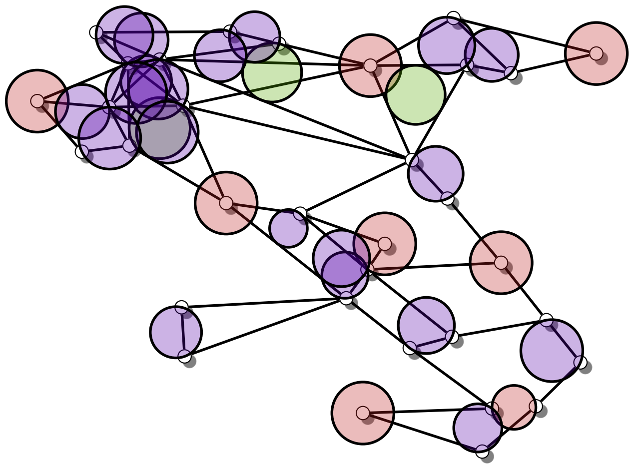

Figure 3.10: Italian network ($|33\_optic\_italian|$)

Table 3.1: Test results on some backbone topologies

Name of network	n	m	x	σ	$\lvert\mathscr{S}\rvert$
10_test_pcycle	10	22	1	16	17
10_test_smallnet	10	22	1	17	15
16_optic_pan_eu	16	22	0	5	14
17_optic_german	17	26	0	7	15
21_test_tnet	21	25	0	6	19
22_optic_eu	22	45	0	13	34
26_optic_usa	26	42	0	10	25
28_optic_eu	28	41	0	9	39
33_optic_italian	33	56	4	13	31
37_optic_european	37	57	0	7	41
39_optic_north_american	39	61	0	7	33
79_optic_nfsnet	79	108	0	9	92

Here we have to mention that while our theoretical upper bound on $|\mathscr{S}|$ is $O((n+x)\sigma^3)$, our examples with highest $|\mathscr{S}|$ values (not presented in the thesis) have much less maximal failures:

Proposition 45. *For every value of σ there exists a graph family and r for which $|\mathscr{S}|$ is $O((n+x)\sigma)$.* $\square$

In the future we want to meet the theoretical complexity results with the complexity results coming from our artificial worst-case graphs.

$n = 33$
$m = 56$
$|\mathscr{S}| = 34$

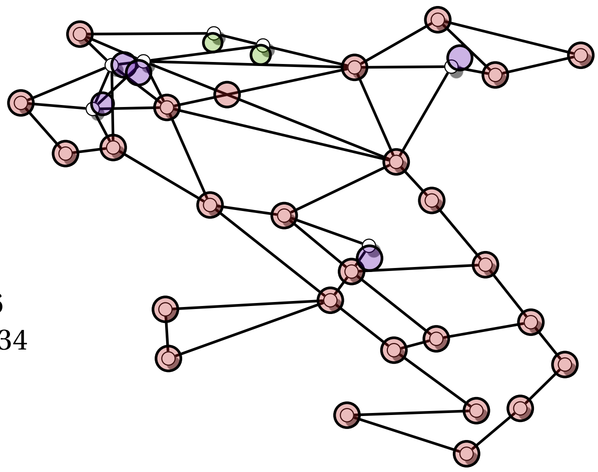

Figure 3.11: Italian optic network $r \simeq 20$km

$n = 33$
$m = 56$
$|\mathscr{S}| = 35$

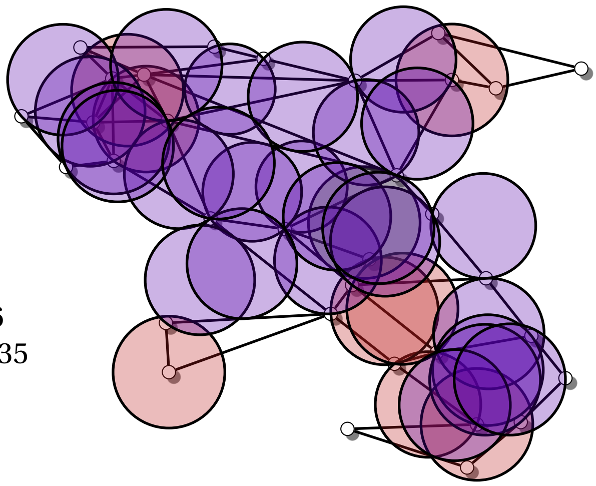

Figure 3.12: Italian optic network $r \simeq 90$km

Table 3.2: $|\mathscr{S}| \simeq (n+x)$ in practice

Network	n	m	x	r	σ	$\|\mathscr{S}\|$
33_optic_italian	33	56	4	5	7	37
33_optic_italian	33	56	4	7	7	37
33_optic_italian	33	56	4	9	7	38
33_optic_italian	33	56	4	11	7	38
33_optic_italian	33	56	4	13	7	38
33_optic_italian	33	56	4	15	8	35
33_optic_italian	33	56	4	17	8	34
33_optic_italian	33	56	4	19	8	37
33_optic_italian	33	56	4	21	8	38
33_optic_italian	33	56	4	23	12	35

Chapter 4

Related works

One approach to analyse the network vulnerability against regional failures is using probabilistic failure models, where each link in the SRLG has some probability to fail [1]. The probabilistic failure models can quantify the network protection schemes through evaluating their end-to-end connection availabilities. However often the end-to-end connection availability is not sufficiently detailed modelling of the failure state, because it ignores the reconfiguration costs, possible traffic changes due to the failure, some limitations in the protocol and failure discovery mechanisms and physical impairments of the network. Therefore during network planing it is preferred to model the network behaviour of each possible failure state independently, and preconfigure the network for fast failure recovery of the failure state. The power of probabilistic failure models to implicitly treating a great number of failure states, but this on the other hand strongly limits the applicability of these models for network planing. Instead, the scope of this study is to define a small number of failure states (as SRLGs) to protect.

In [20] the circulant failure model is generalised and the regional failures are modelled by a given elementary geometric figures as ellipse, rectangle, square, or equilateral triangle with a predetermined size. The problem is similar to ours, and it is showed that there is a polynomial number of non-trivial positions for such a figure that need to be considered. In this study for simplicity we stick to failures of cycles only. Our main contribution compared to [20] is that this polynomial is basically linear in practice for cycles.

Chapter 5

Conclusions

In this thesis we propose two fast and systematic approaches to enumerate lists of possible link failures caused by regional failures. Our approaches assume a failure is caused by a disk as it erases all the network elements it intersects. While one our approaches assumes that the disks are of any size, but do not have a node interior, our other approach fixes their size, but lets disks to cover nodes. Although the number of possible regional failures is infinite, we show that the generated lists of failures are short, they are basically linear to the network size.

We provide fast polynomial time algorithms for enumerating the corresponding set of Shared Risk Link Groups. According to our knowledge this is the first study providing a comprehensive solution for these problems. As our approaches may be applied to similar or more general problems, we plan to extend our model to cover more major classes of regional failures.

Chapter 6

Future work

In this thesis we assumed that the network is a geometric graph embedded in the 2D plane with edges drawn as line segments, and we assumed that the geometric shapes causing the failures are disks.

Thoughts included in this thesis seem to be as a base point starting from where we can push our understanding of regional failures much further. There are so many directions we can go that it is hard to pick up some from them.

As a future work we plan to extend our results from Chapter 2 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.

Our two approaches may be combined: the maximal failures caused by disks with radius (at most) r covering (at most) k nodes may be listed.

A third possibility is to omit the assumption that the links are line segments. Approaches from Chapter 2 can be slightly altered to handle this case too.

A fourth possibility is to apply our algorithms on sphere or in 3D.

Using different metrics the shape of disks might be generalised, even in 2D. This way we could model the effect of regions more resistant to some kind of disasters.

Combining the previous paragraphs we can study networks embedded in special metric spaces, with arbitrary form of edges, and thus we may model a network on a geoid, or even every possible network on the Earth.

Time could be a dimension of metric spaces too, the network could evolve, edges could not necessarily be curves, etc.

Hopefully, we will have the possibility to research on the most worthy cases.

Bibliography

[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] B. Vass, E. R. Kovács, and J. Tapolcai, "Shared risk ink group enumeration of node excluding disaster failures," *NaNA International Conference on Networks and Network Applications*, 2016.

[14] B. Vass, "Listing regional failures," *BME Mesterpróba*, 2016.

[15] ——, "Shared risk link groups of disaster failures," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS): Student Activities*, 2016, pp. 391–392.

[16] B. Vass, E. R. Kovács, and J. Tapolcai, "Shared risk link groups of disaster failures: The single node failure case," *submitted to RNDM International Workshop on Resilient Network Design and Modelling*.

[17] B. Vass, "Hálózatok legfennebb *r* sugárban ható hibáinak felsorolása," *Scientific Students' Associations conference organised by the Mathematical Institute of Eötvös Loránd University*, 2015.

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

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

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