Fault Damage Zone Fracture Network Connectivity

A graph theory approach towards the assessment of fault damage zone leakage risk using DFN and outcrop studies

Master thesis

Hidde W. Spaa

TU Delft

Fault Damage Zone Fracture Network Connectivity

A graph theory approach towards the assessment of fault damage zone leakage risk using DFN and outcrop studies

by

Hidde W. Spaa

Supervisor:	Prof. Dr. G. Bertotti
Internship supervisor:	Dr. K. Bisdom
Date:	01/09/2022
Faculty:	Civil Engineering and Geosciences, Delft

Cover: Watchet Fault, Watchet (UK), drone image by Dr. R. Rizzo

TU Delft

Summary

Capturing CO_2 directly at industrial complexes and safely storing it in the subsurface is one of the proposed mitigation measures of climate change. One of these challenges is identifying suitable reservoirs that ensure the safe and permanent storage of CO_2. Therefore, detailed integrity assessments should be performed on the caprock to accurately determine and manage the risks involved in the permanent storage of CO_2 (Kaldi et al., 2013). This should include leakage risk quantification of (sub-seismic) fault zones in the primary seal (Zappone et al., 2021). This thesis aims to understand and predict the probability of caprock leakage through a fault-related fracture network. It specifically focuses on the network organisation of fault damage zone (FDZ) fracture networks and the impact of their topology on the connectivity of the fracture network. Topology, in the form of fracture node classification, has been used to determine the connectivity of fracture networks (Sævik and Nixon, 2017). However, this research points out that the connectivity of a fracture network cannot be solely determined by a linear relationship between the fracture node ratio and the connectivity. This methodology does not effectively succeed in capturing connectivity differences of different fracture networks. To research these differences, a measure of connectivity is introduced based on the graph theory concept of the giant component. In this concept, networks are connected by a certain amount depending on their node type. By using the total length of the giant component we find the largest component ratio (LCR). This measure of connectivity and its relationship with percolation was tested using a DFN simulator. The concept of the giant component allows us to study the connectivity of various fracture networks. This is done by developing an algorithm that is based on the concept of robustness that removes fracture segments from a fracture network, which in turn enables us to find a relationship between the topology and the connectivity of the network and assess its uncertainty. From the DFN simulations and the robustness algorithm, it is found that the relationship between connectivity and topology is unique for different fracture networks. However, it was concluded that in general a fracture network significantly starts to increase in its connectivity when the average node has 1.6 edges connected to it (k). It also showed that the highest uncertainity of fracture network connectivity is present at 1.9<k<2.2.

Contents

1

Introduction

Since pre-industrial times, the concentration of greenhouse gases such as CO_2 and methane have increased substantially in the atmosphere. This has can have a severe impact on the stability of the global climate system. Reducing greenhouse gas emissions from human activities is key in mitigating these dangerous effects.

Capturing CO_2 directly at industrial complexes and safely storing it in the subsurface is one of the proposed mitigation measures. The importance of this technology, which is called Carbon Capture and Sequestration (CCS), and its vital role to achieve global climate goals has been underlined by the UN International Panel on Climate Change (IPCC, 2022).

The safe storage of CO_2 in the subsurface has several challenges. One of these challenges is identifying suitable reservoirs that ensure the safe and permanent storage of CO_2. This is an important task as the leakage of CO_2 from a storage location can have severe consequences. These include the release of CO_2 back into the atmosphere and the contamination of natural surroundings.

A large amount of research has been performed on the subsurface characteristics of reservoirs. This research was mostly performed to study the feasibility of oil and/or gas production from these reservoirs. Accurate predictions of reservoir architecture and behaviour is difficult as reservoirs are often situated at several kilometres depth. The only direct data source that is available from these reservoirs are cores. Besides being scarce, due to financial constraints, they only give information of a small area. Therefore, indirect data sources, such as well logs and seismic imaging, are often used to obtain a better understanding of overall reservoir architecture and properties. Another data source are outcrops which can serve as analogues to the reservoir. Outcrop studies allow for direct data collection of rock properties and reservoir characteristics.

Less research of the oil and gas industry focused on the sealing capacity of the layer above the reservoir (seal) that prevents the upward migration of reservoir fluids or gasses. This is because when there is a confirmed presence of oil and/or gas in the reservoir the assumption can be made that the caprock has sufficient sealing capacity. Due to this assumption depleted oil and gas reservoirs are often considered for the permanent storage of CO_2. However, due to the changes in reservoir pressure caused by production of these reservoirs and the different properties of CO_2 compared to the original reservoir fluid/gas, this assumption should be avoided. Therefore, a detailed integrity assessment should be performed on the caprock to accurately determine and manage the risks involved in the permanent storage of CO_2 (Kaldi et al., 2013). These assessments should include leakage risk quantification of (sub-seismic) fault zones in the primary seal (Zappone et al., 2021). Faults that penetrate the caprock can potentially form corridors for fluids or gasses to escape through the caprock. Especially, the fault damage zone (FDZ), which is a zone of faulting-related deformation, is identified as a primary pathway for fluids or gases (Kim and Sanderson, 2010). This is because the level of deformation, of which much is often accommodated by fractures, is far greater in these zones than the rest of the caprock. These fractures, which have a permeability far greater than the surrounding rock, can form fracture networks that span across the caprock. If these fracture networks have sufficient connectivity, leakage can occur. Therefore, it is important to better characterise and model these fault-related fracture networks.

1.1. What are fractures?

A fracture is a discontinuity in the rock that is caused by failure under stress. This process typically involves mode I, II and III deformation (see chapter 2). Fractures are considered joints when there is no displacement between the different bodies. When a notable displacement can be observed it is called a shear fracture. A fracture becomes a fault when large scale displacement (meter scale) is observed.

Fractures are commonly altered after their formation. Exhumation and weathering processes often have a profound impact on fracture aperture and geometry (Gale et al., 2014). Mineral precipitation (i.e. vein formation) and/or dissolution in fractures occur when exogenous fluids are in chemical disequilibrium with the host rock. This can have a severe impact on fracture transmissivity (Jones and Detwiler, 2016). Mineralization and weathering processes are, however, out of the scope of this research.

An arrangement (or system) of interconnected or disconnected geological fractures in a given area is considered to be a fracture network.

1.2. Studying fractures and fracture networks

Fractures can often only be observed directly from log and core data as they are present below seismic resolution (sub-seismic scale). These sources have a major limitation as they only sample a very small proportion of the entire fracture network. Since there often is a large amount of spatial variation in fracture networks, it is difficult to determine the spatial arrangement of fractures beyond the wellbore. To mitigate these limitations and improve the understanding of fracture networks as a whole, outcrop studies can be performed. From outcrop and laboratory studies it is known that fracture networks can often be divided into orientation sets that develop in consistent geometric relationships to one another. However, there is still a gap when it comes to converting that deterministic knowledge into a predictive model that can be used to assess flow behaviour of subsurface fault zones.

Fracture networks, including fault-related fracture networks (Ceccato et al., 2021; Panza et al., 2018; Smeraglia et al., 2021; Voeckler and Allen, 2012), are often studied and modelled using discrete fracture networks (DFN). In this modelling technique, fractures are represented as explicit discrete lines or surfaces. DFNs typically model fracture networks by using distributions of fracture geometries (width, length, height and abutment relationships). Many DFN studies performed assume random placement of fractures without taking into account the complex network organisation that is often found in natural fracture networks (Maillot et al., 2016). As this severely impacts the (connectivity) characteristics of fracture networks, DFN simulations that consider geomechanical rules have been performed (Bonneau et al., 2016; Welch et al., 2009).

A physics-driven approach (forward modelling) models fracture networks by using geomechanical laws and (assumed) mechanical properties of the medium. These simulations thus mimic geological processes. However, 3D fracture propagation modelling and the uncertainty in the geological process responsible for structures seen in nature, are major obstacles (Dijk, 2019), as it is hard to determine the impact of different fracture formation controls. Additionally, these techniques are computationally expensive and are therefore not well suited for statistical analysis for which many fracture networks are required.

Both methodologies described above are used to model large-scale fracture networks and are not specialised to research fault-related fracturing. These fracture networks are hard to model as they are formed under a complex set of formation processes. These formation processes are very case-dependent and often result in complex fracture networks.

1.3. Thesis aim and outline outline

This thesis aims to understand and predict the probability of caprock leakage through a fault-related fracture network. This is an issue of percolation; does a fracture network pathway exist through the caprock?

In this thesis, we specifically focus on the relationship of network organisation on the connectivity. We research this relationship by using topology and other graph theoretical concepts.

The ultimate goal of this thesis is to find a methodology by which the risk of leakage and its uncertainty can be determined using only sparse data obtained by, for example, wells. This goal is best described by the example depicted in figure 1.1. In this figure, a fault is shown with a fault damage zone fracture network in the caprock (seal). A well is drilled through the fault damage zone and a subset

of the fracture network is observed. How can the data from the well say something about the connectivity of the entire network? Do we expect different relationships between the topology and network connectivity for different types of fracture networks?

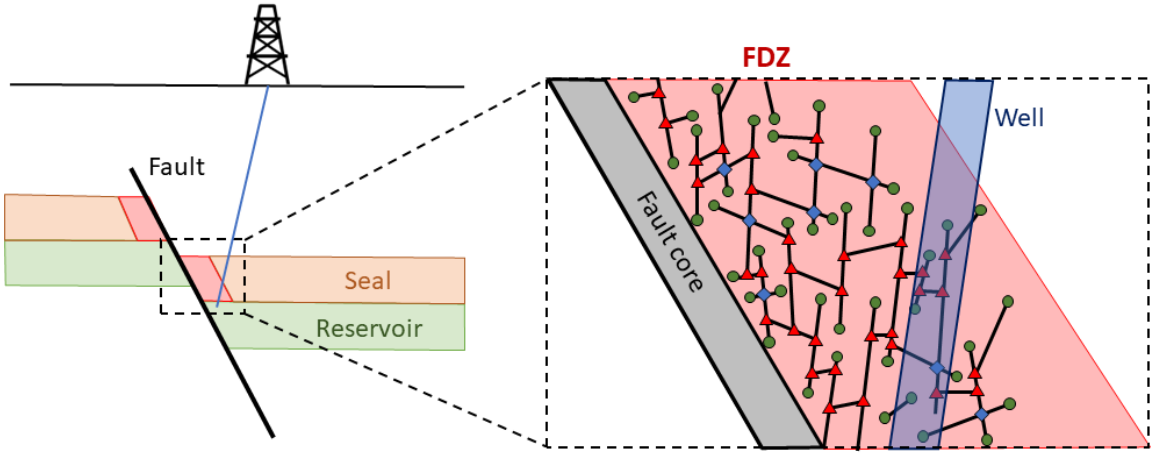


Figure 1.1: Fault with FDZ that is drilled by a well. From this well a part of the fracture network can be observed.

To improve this assessment two research questions are formulated:

- Can we capture the approximate connectivity of FDZ fracture networks in a statistically robust way based on sparse subsurface data?
- Can we use a simplified model based on the topology to make forecasts on the connectivity of specific subsurface fault damage zones?

An attempt is made to answer these questions by first identifying which topology based measures of connectivity exist and whether these measures of connectivity help to identify differences in connectivity of different (FDZ) fracture networks. This analysis is performed by using a DFN generator. Hereafter, a variety of FDZ fracture network outcrops are researched based on the graph theory concept of robustness. This concept is applied to a variety of FDZ's to research the differences in connectivity and its uncertainty.

This thesis focuses on analysing fracture networks as graphs and using graph theory concepts to determine their connectivity. However, first, a basic understanding of FDZ fracture networks needs to be gained.

The basics of fracture development are discussed in chapter 2. Additionally, this chapter goes into the geomechanical formation process that leads to the formation of fracture networks.
Chapter 3 gives an insight into the internal structure and hydraulic properties of fault zones. Additionally, it outlines the challenges related to studying these (subsurface) features.

The topology and graph theory concepts that are used extensively in this thesis are discussed in chapter 4. This chapter discusses graph theory concepts of connectivity and percolation. First, the basic principles of percolation theory are outlined. Thereafter, existing connectivity assessment methodologies based on fracture network topology are discussed. Finally, the connectivity concept of the giant component and its relationship with graph robustness are discussed in detail.

Chapter 5 outlines the methodology used to improve topology based connectivity assessment of FDZ fracture networks. First, the DFN generator that is used to research different connectivity measures is discussed. Thereafter, the robustness algorithm to assess natural fracture networks is explained.
The results of the DFN and outcrop analyses are shown in chapter 6. Chapter 7 discusses the results and evaluates the usage of the newly developed technique.
The findings and conclusions of this thesis are described in chapter 8.

2

Fracture mechanics

To understand the formation of complex fracture networks it is essential to first obtain an understanding of fracture development, as the formation controls and the resulting geometry of these fractures directly impact the overall architecture of the fracture network. This in turn will impact the connectivity of the fracture network. First, two different types of deformation are introduced. Hereafter, the concepts of fracture formation are briefly introduced. Lastly, the development of fracture networks is discussed. Studying these fracture network formation processes will help to understand their resulting characteristics, which in turn can help to deduce relevant geometric relationships.

2.1. Rock deformation

The deformation of rocks is complex and impacted by a range of geomechanical processes which are largely dependent on the behaviour of the protolith. Distributed deformation in granular sedimentary rock is often accommodated by grain crushing and pressure solution. More localised deformation often comes in the form of deformation (shear) bands. These regularly form in coarse-grained high porosity rock (>10-15%) (Fossen et al., 2007). Massive behaviour, which is the focus of this thesis, predominately shows fracturing. Both deformation bands and fractures can be observed on grain- to macro-scale (Faulkner et al., 2010).

2.2. Fracture formation

Three end members of fracture formation exist, namely mode I, II and III fracturing. The stress state under which a particular plane in a body of rock will fracture is explained by using the Mohr-Coulomb failure criteria. Figure 2.1 shows the Griffith and Coulomb failure criterion, brittle-ductile transition and Von Mises ductile failure criterion. Additionally, the related failure types are indicated.

Mode I fractures (A in fig. 2.1) are fractures that open due to loads that are perpendicular to the normal of the fracture plane. Due to the relationship principal component σ_1 it does not have a shear component.

Mode II fractures (C) shear along the fracture plane due to tangential load to the normal of the fracture plane. Hybrid fractures (B) are a combination of mode I and II fractures and form under a small angle to σ_1. Therefore, they show opening and shear.

Mode III (or tearing mode) fractures is a form of out-of-plane shear. The fractures are formed with a shear stress that acts parallel to both the fracture plane and the front of the fracture.

Mode II is easily explained by the Mohr-Coulomb failure criteria and are abundantly observed in nature and can be created with experiments. Conversely, the conditions that control mode I fracture formation are poorly understood. They form under tension, which is unrealistic in the subsurface, and under high fluid pressure, which is present in geological over-pressure situations and during induced hydraulic fracturing. Contrary to mode II failure, experiments are not able to determine the formation controls of mode I failure (Bertotti et al., 2016).

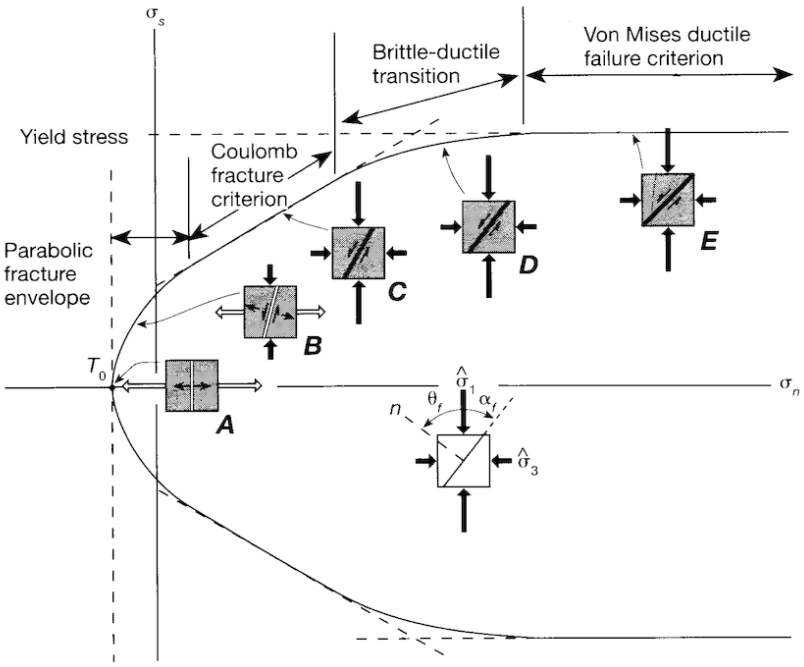


Figure 2.1: Failure envelopes and related fractures. We see that the Mohr diagram mainly consists of three parts; the Griffiths, Coloumb and Von Mises ductile failure criterion. The Griffiths failure criterion (parabolic failure criterion) is relevant for high cohesive rocks. When cohesion is larger than the differential stress, a mode 1 tensional fracture forms when the fluid pressure within the rock is increased. The Coloumb criteria can be reached by increasing the fluid pressure when the differential stress is larger than the cohesion within the bed forming Mode 2 shear fractures along the internal angle of friction. The Von-Mises ductile failure criterion is relevant for ductile materials. After, Twiss et al. (2007)

2.3. Fracture networks

Fracture networks form under a range of processes. These are directly impacted by the formation processes of individual fractures. Therefore, this section is divided into two sections; mode I fracture networks and mode II fracture networks. Both types can of fracture network development can lead to the formation of well-connected networks.

2.3.1. Mode I fracture networks

Abundant examples of well-connected Mode I fracture networks can be found in nature (fig. 2.2). These networks can develop due to a variety of geomechanical processes which are impacted by e.g. bed thickness and fracture spacing.

(a) **(b)**

Figure 2.2: a) Orthogonal cross joints network in the carbonate Monterey Formation, California (US). after (Bai et al., 2002). b, A joint ladder pattern, Somerset (UK). After, Peacock and Sanderson (2018)

Fracture spacing and bed thickness

Fractures are often bounded by mechanical boundaries present in the massive rock (fig. 2.2a). These boundaries often correlate with sedimentary bedding. The spacing of mode I fractures in geomechanical layers are often related to the bed thickness. In a process called sequential infilling (Gross, 1993) fractures develop in the layer until saturation is reached.

Bai and Pollard (2000) used a finite element method to study the fracture-spacing-to-layer-thickness ratio. From numerical modelling, it was shown that four ranges can be identified (Bai and Pollard, 2000); *under-saturation* of the layer, a *saturated* layer, an *over-saturated* layer and a layer that is very *over-saturation*. Over saturation can only occur if other mechanisms are at play (e.g. fluid pressure).

The critical spacing to layer thickness ratio increases non-linearly with increasing Young modulus ratio of the fracture layer to the neighbouring layers, with increasing Poisson's ratio and increasing overburden stress (Bai and Pollard, 2000).

Bai et al. (2000) determined three different mechanisms (fig. 2.3b) that determine the stress state in the finite element model. The first mechanism, which is active in range I, transfers stress from adjacent layers producing a tensile regime. In range (II) the horizontal compressive stress increases due to the opening of the fractures. The third mechanism is the tranction-free fracture surface effect and starts when fracture-spacing-to-layer-thickness ratio drops below 0.3 (range III).

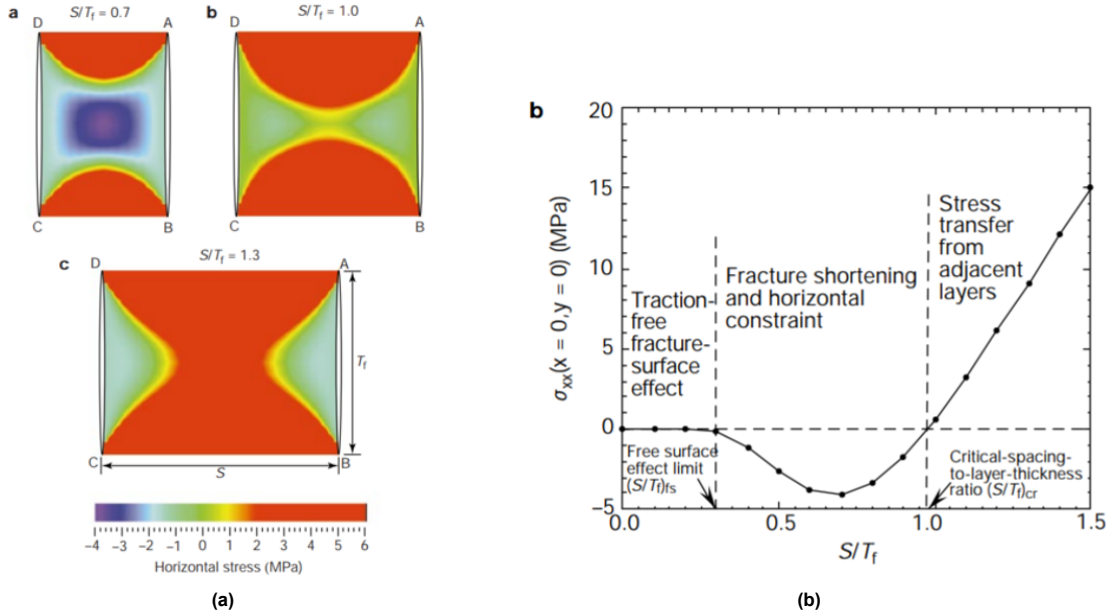


Figure 2.3: a) Horizontal stress state for different fracture-spacing-to-layer-thickness ratios (S/T_f). b) Horizontal stress between the fractures is controlled by three different mechanisms depending on the S/T_f ratio. After, Bai et al. (2000).

Orthogonal fracture networks

Mode I fractures can develop into well-connected orthogonal networks (Bai et al., 2002; Boersma et al., 2018; Gross, 1993) and have been observed in many field studies (Li and Ji, 2021; Peacock and Sanderson, 2018) (fig. 2.2b).

By studying the main orientation trend found in mode I fracture networks, a regional maximum tensile stress orientation can be deduced (Dyer, 1988), since mode I fractures form perpendicular to the local maximum tensile stress following linear elastic fracture mechanics (LEFM) principles. Bai et al. (2002) noted that this approach implies a 90° rotation of the principle stresses when considering an orthogonal mode I fracture network and proposed a local switch in principal stress orientation (fig 2.4a). The *stress shadowing* effect introduced by mode I fractures (fig. 2.3) has less effect on vertical stress in between the fractures, and therefore causes the vertical stress to become more tensile than horizontal stress (stress switch).

From this a critical-spacing-to-height ratio was introduced that indicates the spacing-to-height ratio at which a local stress switch occurs. They found that the critical-spacing-to-height ratio is independent of the Young's modulus, increases nonlinearly with increasing horizontal remote stress ratio and shows

an inverse relationship to Poisson's ratio.

Mode I fracture networks reach this saturation in four different geometric manners: (1) a closely spaced set of parallel fractures, (2) a ladder-like network geometry, (3) an inter-connected nested arrangement and (4) a cross-cutting network (Boersma et al., 2018) (fig. 2.5).

Boersma et al. (2018) noted that the development of closely spaced mode I fractures often occurs under subcritical stress conditions and therefore includes subcritical crack growth (SCCG) (Atkinson, 1984) in their static model. Subcritical crack growth describes the stress concentration on fracture tips which enables the fractures to grow. In the numerical model, they introduced two widely spaced fractures with 17 micro-fractures in between and observed the development of orthogonal fracture networks due to LEFM and SCCG mechanics. The development of these networks is dependent on Poisson's ratio and the remote horizontal stress ratio and therefore impacts the network geometry.

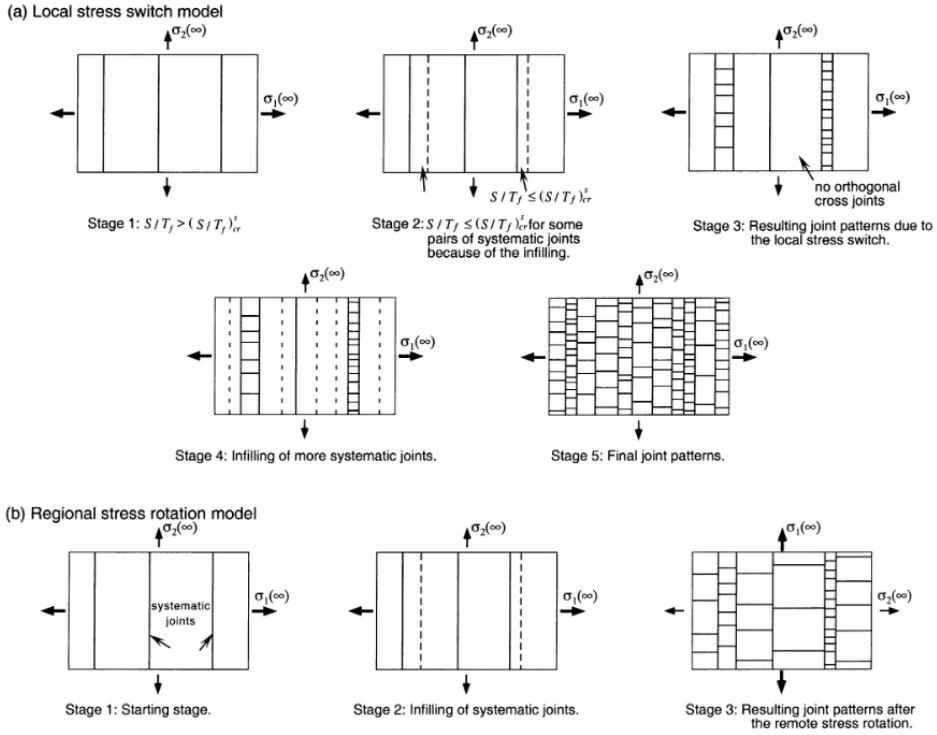


Figure 2.4: Illustration of two different models of how an orthogonal fracture network can develop. a) Shows a local stress switch. In this development process, the cross joints will form in a systematic way between the longer fractures due to a local stress switch. The cross-joints will first fill the more close-spaced fractures (stage 3). However, when the spacing-to-height ratio reaches its critical values, cross-joints can also be initiated in more widely spaced fractures (stage 5). b) Regional stress rotation model to explain cross-joint formation. For this model, the cross-joints do not preferentially form in the more close space fractures. After, Bai et al. (2002)

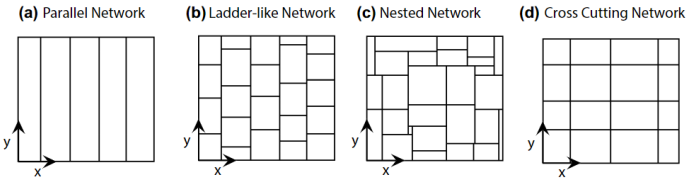
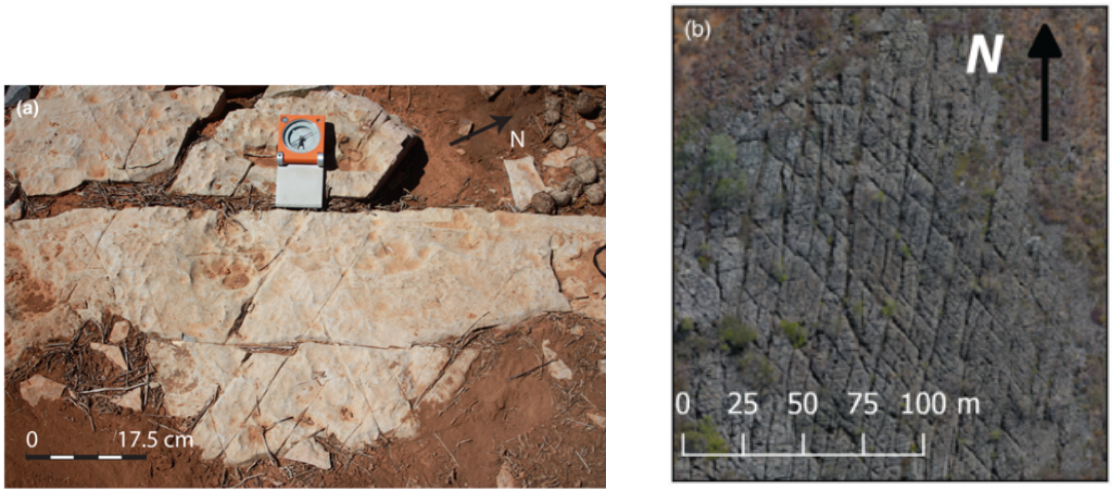


Figure 2.5: 4 different geometric manners how orthogonal fracture networks reach saturation. After, Boersma et al. (2018)

2.3.2. Mode II fracture networks

Like mode I fracture networks, mode II fracture networks can be abundantly observed in nature (fig. 2.6). Mode II (or conjugate) fracture network development, including the mechanical controls and the saturation of the fracture network, is less clearly understood than mode I fracture network development.

Boersma et al. (2018) used LEFM and SCCG principles acquired from mode I fracture development to model conjugate fracture sets. The numerical models showed that micro-fractures develop into fully connected and nested conjugate fracture networks when the critical differential stress is reached. Furthermore, significant stress localization was observed that eventually leads to fracture saturation. Moreover, mode I fractures are highly dependent on applied driving stresses. Boersma et al. (2018) noted several assumptions and limitations in this numerical approach.



(a) (b)

Figure 2.6: a) detailed image of a conjugate fracture sets b) drone image of a fracture pavement (Bahia, Brazil). After, Boersma et al. (2019)

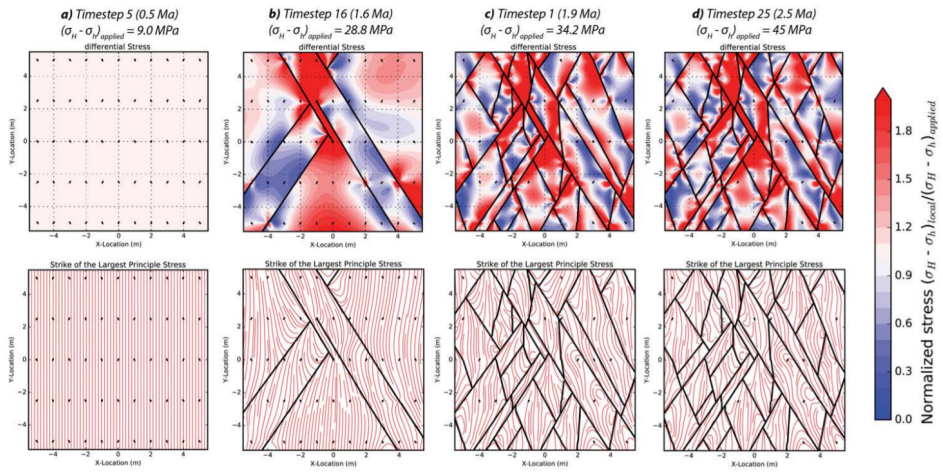


Figure 2.7: Stress localization and the development of conjugate fracture networks. After, Boersma et al. (2018)

3

Fault zone characterisation

Much research has been done on the internal structure, mechanical behaviour and permeability of fault zones (Anderson, 1951; Childs et al., 2009; Faulkner et al., 2010,1; Mitchell and Faulkner, 2009; Peacock et al., 2017; Savage and Brodsky, 2011; van der Zee et al., 2008; Wibberley et al., 2008), since many subsurface processes are controlled by fault zones. Faults are highly complex features with a high degree of variation along the strike and down dip. Therefore, studying faults and understanding their hydraulic properties is a challenging task. Additionally, comparing different studies can be difficult due to inconsistent use of fault zone terminology (Childs et al., 2009). Therefore, it is important to give a clear description of the used terminology.

3.1. Internal structure

In the most simple conceptual model fault zones are characterized as a fault core consisting of a gouge, cataclasite or ultacataclasite surrounded by a damage zone which consists of low-strain features (fig. 3.1a). Strain is concentrated in the fault core and can be distributed either homogeneously or over multiple high-strain slip surfaces (fig. 3.1b).

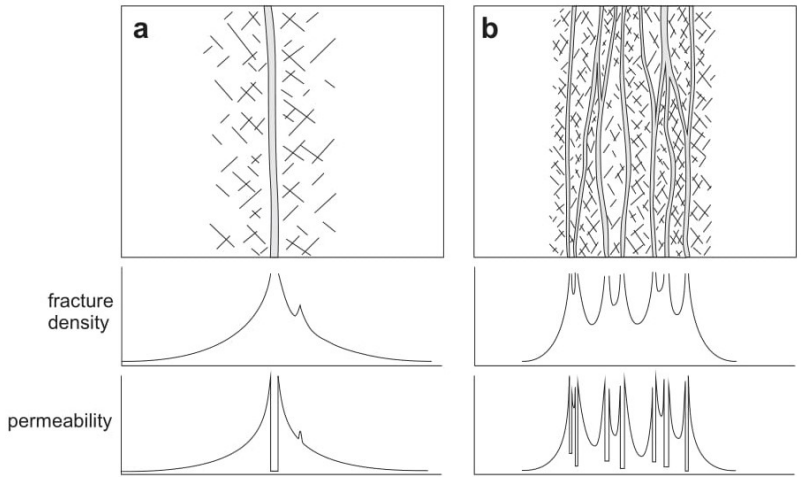


Figure 3.1: Strain is concentrated in the fault core and can be distributed either homogeneously or over multiple high-strain slip surfaces. After, Faulkner et al. (2010)

More complex conceptual models consist of anastomosing fault cores where faults branch off in different directions and link up again creating fractured protolithic lenses (van der Zee et al., 2008). Moreover, the fault rock (deformed rock at the fault core) is not simply a (granular) deformed protolith, but often undergoes complex mechanical, deformational, and fluid-rock interaction processes. Fault rock is often altered by the authigenic growth of minerals such as clay and could be considered as low-

to medium-grade metamorphic rock in some cases. Additionally, the thickness and distribution of fault rock are largely determined by fault geometry (Childs et al., 2009).

Outcrop studies show that the position of the fault plane with respect to the fault zone is variable (Shipton and Cowie, 2001), however, we consider the most simplistic case in which the fault plane is directly adjacent to the fault core (gouge), which is defined as a localised zone of high cataclastic deformation. The FDZ is the volume of deformed wall rock around the fault core. The width of the FDZ is the length between the outer boundary of the fault core and the point where deformation (fracture intensity) falls to background levels.

3.2. Fault initiation and growth

The properties and geometry of fault zones are largely determined by their genetic process. There are essentially two growth models, namely the propagation model (fig. 3.2) and the linkage model (fig. 3.3). The propagation model contains three groups of domains; far field, near field and along fault. A strong stress concentration at the fault tip enables the fault to propagate while often creating relatively large areas of distributed low-strain deformation. Clasts are progressively disrupted and fractured in the processes zone associated with the approaching fault tip. Along fault deformation is dominated by frictional processes such as grain sliding and crushing and typically involves volume reduction. The linkage growth model often develops in regions with widespread distributed fractures. The fractures grow and link, and thereby creating faults that can accommodate displacement. The respective fault can then link up with other faults by breaching the relay ramp that separates the two fault segments (fig. 3.3).

The internal structure of FDZ's are largely influenced by the formation process. The next section, therefore, gives an in-depth description of the different fault mechanisms.

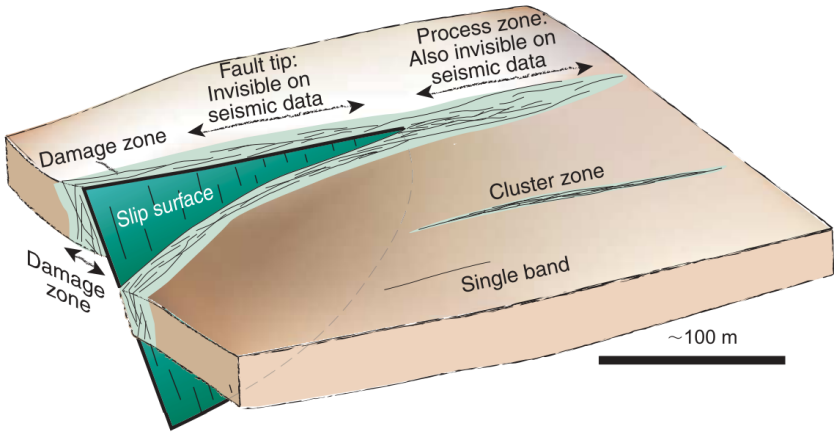


Figure 3.2: The fault propagation model containing three domains; far field, near field and along fault. After, prof. G. Bertotti lecture (TUDelft).

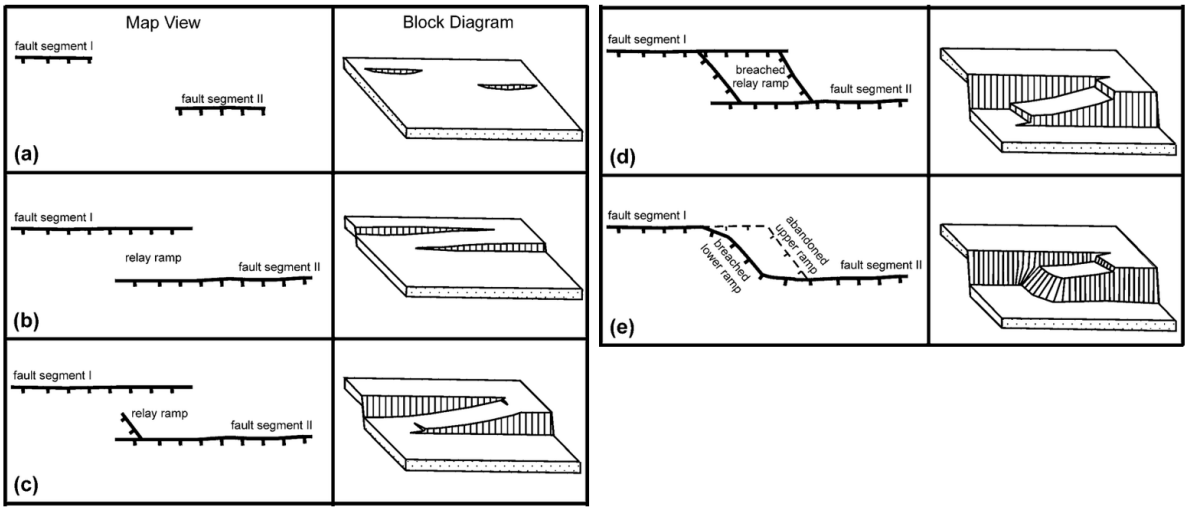


Figure 3.3: The linkage fault growth model. Different fault segments connect by breaching the relay ramp. After, Ciftci and Bozkurt (2007)

3.3. Processes responsible for off-fault damage

Off-fault damage often shows complex structures that are due to a large range of formation processes.

For instance, significant differences in fracture orientation and distribution can be present between the footwall and hanging wall due to the development of an asymmetric stress field during fault propagation and hanging wall folding (Berg and Skar, 2005).

Mitchell and Faulkner (2009) compiled five processes that are responsible for off-fault damage (fig. 3.4). Based on the Anderson model (Anderson, 1951), which assumes a homogeneous stress state and Coulomb failure behaviour in the crust, faults will form under an angle of 25-30° to the maximum compressive stress. These types of faults form due to the coalescence of many tensile microcracks that are orientated parallel to the maximum principle compressive stress (fig. 3.4a) and will therefore be under an angle of 25-30° to the fault plane.

The formation of wing cracks at fracture tips results in the second process as described by Mitchell and Faulkner (2009) (fig. 3.4b). These extensional features may link up to other fractures and can eventually lead to the initiation of a fault. Deformation by this process will result in fractures with the same orientation as the first model, but more concentrated around the fault.

The third process, which results in fractures in a different orientation, is based upon fracturing around the tip of a propagating fault (process zone). Propagation of the fault and thus the propagation of the processes zone, will lead to the development of a damage zone surrounding the fault (fig. 3.4c). The stress magnitude related to the process zone largely determines its size. Since the stress magnitude is related to the fault length (following linear elastic fracture mechanics (LEFM)), the fault length should scale with the size of the process zone. Local tensile stress conditions at the fault tip result in fractures with a high angle to the fault plane. Compressive local stress conditions will lead to the formation of fractures that are under a lower angle to the fault plane. These local stress conditions can often be found at the same fault tip.

Increasing slip of the fault plane leads to the fourth process of off-fault deformation (fig. 3.4d). Fault plane roughness results in stress cycling, which in turn, results in cumulative fracture damage. Therefore, with increasing fault displacement, increased damage zone widths can be expected. The orientation of the fractures varies along the fault plane since maximum compressive stress orientations vary. Damage which resulted from fault slip might overprint damage created in the processes zone by fault tip propagation.

The fifth process (3.1e) is related to earthquake rupture events. The fractures created by this process likely show similar orientations as fractures formed by fault propagation. The rupture velocity influences the dynamic stress field of a propagating slip pulse, and thus, the orientation of the fractures (Rice et al., 2005). The size and the degree of deformation of the damage zone is further controlled by the stress drop during the slip event, friction coefficients, principle pre-stress orientations and poroelastic coefficients. A combination of all five off-fault damage models (fig. 3.4f) results in a complex pattern

of fractures that surround the fault plane.

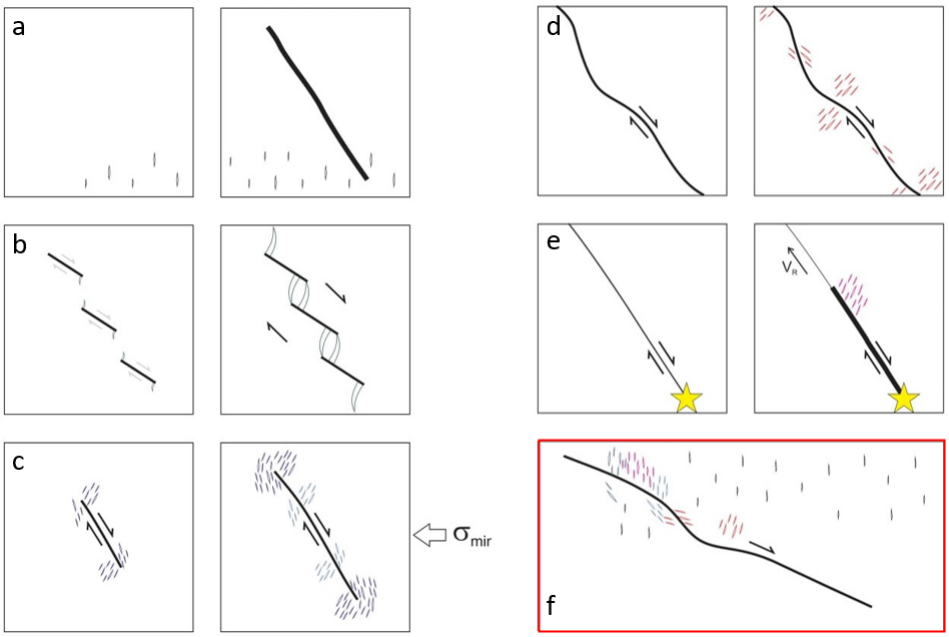


Figure 3.4: 5 different processes that are responsible for creating fault damage zones. For every process, the initial state is shown (box on the left) and the developed state (right). a) Fractures that coalesce and initiate a fault. b) The linking of fractures. c) Process zone fracturing. d) Deformation along kinked faults. e) Fracture damage caused by a seismic event. f) Combination of all processes results in a complex mixture of FDZ structures. After, (Faulkner et al., 2010)

3.3.1. Interacting damage zones

Peacock et al. (2017) introduced an interaction damage zone model where two faults with any orientation and relative age interact. Interaction damage zones include linkage, approaching and intersecting damage zones. Approaching damage zones are present between two approaching faults that are kinematically linked but not geometrically. Intersecting damage zones are geometrically linked, although this does not imply they are kinematically linked. The linkage model is a subcategory of the interaction damage zone model which occurs between two approaching sub-parallel faults.

3.3.2. Specific features of Fault Damage Zones

A large number of features can be found in the damage zone. These features, which are present over a range of scales, include wing cracks, horsetail fractures, synthetic branch faults, antithetic faults, extensional and contractional steps.

Kim et al. (2004) compiled field data and brought forward a geometric classification scheme of strike-slip damage zones based on location along the fault (tip, linking, and wall-damage). They state that other faults types such as normal- and reverse faults show similar overall patterns and development mechanisms.

According to Kim et al. (2004) deformation around fault tips is dominated by wing cracks, horsetail fractures, synthetic branch faults and antithetic faults. Wing cracks are formed due to a rapid decrease of slip at the fault tip. They generally form under low differential and confining pressure. When slip decreases more gradually towards the fault tip horsetail fractures will be formed. These features are similar to wing cracks but are more closely spaced. Both wing cracks and horsetail fractures develop as mode I fractures and are therefore parallel to the local maximum compressive stress (σ_1). Synthetic branch faults (or splay faults) show similar slip movement as the main fault and may cause linkage between different faults. Antithetic faults have an opposite slip movement as the main fault and they typically form under an angle of 30° to σ_1. Combinations of these features are possible and are, most commonly, a mix of synthetic horsetail fractures or synthetic branch faults with antithetic faults (Kim et al., 2004).

More intense and complex deformation is often found at linking damage zones. Linkage zone deformation can be divided into extensional and contractional steps (Kim et al., 2004). Extensional

steps contain extension fractures, pull-aparts, rotated blocks and isolated lenses. Extension fractures form parallel to the local σ_1 and can potentially link two fault segments. Pull-aparts are formed due to slip on the fault and their geometry is controlled by secondary features that form the boundaries of the pull-aparts. Rotating blocks are created by increasing slip of the main fault. Antithetic faults, which create the blocks, and extensional fractures occur around the rotated block. The antithetic faults form with an angle of less than 90° to the main fault, but will rotate with increasing slip. Isolated lenses are formed between two fault segments and are similar to sidewall ripouts and open eye-structures. Contractional steps also show features such as rotating blocks and isolated lenses. Additionally, faults that connect two fault segments are often observed (connection faults).

Wall damage zones is the last along-fault location and can be divided into wedge-shaped, long and narrow, and intense damage zones (Kim et al., 2004). Wedge-shaped damage zones are formed due to progressive propagation of the fault tip and an increase in size from the centre to the tip of the fault. Generally, they are only observed at the end of the fault and are therefore likely created by process zone fracturing. Long and relatively narrow damage zones are dominated by extension fractures and veins. The geometry is partly controlled by the highly varying angle between the extensional fracture and the fault. Antithetic faults and synthetic faults can also be found in wall damage zones. Intense damage zones can be the result of friction due to increasing slip. Faults with a large amount of total slip often show synthetically block rotation (with respect to the main fault) at the intersections between faults.

Another type of damage zones are fracture corridors. They are narrowly spaced lineaments of fractures that have a sub-parallel orientation (Ogata et al., 2014; Souque et al., 2019). They are a result of either folding or faulting related (wall and tip zone) deformation. The corridors, made up of individual fractures on the meter scale, can be continuous for hundreds of meters (Souque et al., 2019).

3.4. Hydraulic properties of fault zones

Predicting the hydraulic behaviour of fault zones is difficult due to the many stress and structural complexities. These complexities often result in significant heterogeneous and anisotropic permeabilities (Lunn et al., 2008). Fault zones, in the simplest concept, are made up of a fault core that acts as an across-fault barrier and a damage zone that acts as an along-fault conduit (Caine and Evans, 1996). Therefore, across-fault flow is dominated by the permeability of the fault core whereas the along-fault flow is dominated by the permeability of the fault damage zone.

3.4.1. Fault zone permeability

Fault zone permeability has been important to study reservoir flow properties. Fault core/gouge permeability plays a significant role in fluid behaviour in the subsurface. Two different types of fault gouges can be distinguished; a granular gouge with broken and irregular grains and a gouge containing phyllosilicates (Faulkner et al., 2010). In general, phyllosilicate-rich fault gouges, tend to have a lower across-fault permeability than granular gouges. This is not due to clay alignment, but rather due to microlayers of granular and clay-rich materials.

However, many simplified permeability estimations consider the fault zone as a whole and do not specifically make a distinction between the fault core and fault damage zone. These simplified models include the Shale Gouge Ratio (SGR), juxtaposition studies and well pressure studies.

Shale or clay smearing and abrasional mixing are the most important processes leading to a higher concentration of phyllosilicates in fault zones. The abrasional mixing theory led to the Shale Gouge Ratio (SGR) (Yielding et al., 1997) which calculates the composition of the fault based on the average of the lithologies along which it slipped. The sealing capacity of the sand-shale gouge can then be studied with capillary sealing theory. This does however not give an actual fault permeability but rather a connectivity/seal ratio.

The sealing capacity of faults is also often determined with hanging-wall footwall juxtaposition studies (Allan, 1989). However, this method simplifies the situation greatly and does not consider the possibility of multiple slip surfaces that divide the total displacement on a sub-seismic scale which could lead to unexpected reservoir-reservoir juxtaposition. Moreover, juxtaposition studies, like SGR, only give a connectivity/seal ratio and not a fault permeability value.

Across-fault flow behaviour is often determined using field data. Well, pressure data on both sides of the fault can either confirm or refute communication between the two fault blocks (Wibberley et al.,

2016). This, however, gives no indication of along-fault permeability.

3.4.2. Fault damage zone permeability

FDZ permeability, which is governed by the permeability of the rock matrix and macro-scale features such as fracture networks, controls the along-fault permeability.

Direct field measurements of fault damage zone are often performed to understand FDZ permeability. However, these measurements have significant limitations due to weathering and exhumation processes that alter the fracture network after formation. Furthermore, direct field observations on a larger scale are hard to perform. Therefore, FDZ permeability is often estimated using numerical flow models and secondary data sources such as borehole measurements.

Fractures can play an important role in FDZ permeability since they have a significantly higher permeability than the surrounding matrix. However, FDZ bulk permeability is not controlled by the number of fractures, but rather by their connectivity (see chapters 4). Moreover, the aperture distribution, which is largely impacted by the orientation of the fractures to the present-day local stress field, is a major control. Fault slip or reactivation of faults can also have an important impact on along-fault permeability (Cappa et al., 2022; Tanikawa et al., 2012; Vrolijk et al., 2016). However, this is not directly relevant for this thesis. For completeness, a discussion on the effect of slip on permeability can be found in Appendix A.

3.5. Describing fault zones using scaling relationships.

Predicting complex sub-seismic fault and off-fault damage structure proves often to be difficult due to the many processes involved in their formation processes. Therefore, to obtain a fault zone model, scaling distributions are often utilized that predict fracture intensity and damage zone width (Savage and Brodsky, 2011). These scaling relations take into account the fracture statistics through the fault zone. These methodologies disregard the complexities of fracture network development discussed in chapter 2 and 3 and is therefore a significant simplification step. Additionally, these relationships are often determined using outcrop data, which increases uncertainty, as explained in the introduction to this chapter.

Therefore, statistical properties, such as fracture intensity, orientation and average length, have been used in numerous studies to generate synthetic representations of natural fracture networks. These synthetic representations allow for analytical or numerical permeability computations which can be used to perform upscaling. These synthetic representations, or DFN's, and how they are used to study fracture network permeability in discussed further in chapter **??**

3.5.1. Fracture length

Fracture length is often measured in the field to obtain a cumulative distribution of length (Vazaios et al., 2017). These fracture length distributions often follow a power law relationship. More complex systems often require two individual power laws to accurately describe the relationship, which has led to multifractal scaling analysis (Lahiri, 2021). The same principles are often used for aperture distribution studies (Dichiarante et al., 2020).

3.5.2. Fracture intensity and damage zone width

Various studies show that fracture intensity decays exponentially with distance from the fault (Mitchell and Faulkner, 2009). This decay of fracture intensity is best defined by equation 3.1 where d is the fracture density, n is the decay exponent, c is a fault-specific constant and r is the distance from the fault (Savage and Brodsky, 2011). Fracture intensity around faulted areas with multiple strands can be best estimated with the superposition of equation 3.1 (fig. 3.1).

$$d = cr^{-n} \tag{3.1}$$

There exists a positive correlation between damage zone width and displacement (Faulkner et al., 2011). Savage and Brodsky (2011) found by compiling literature that for faults with less than 150 m of displacement the damage decays by approximately the inverse of the distance from the fault. Faults beyond 150 m displacement show smaller damage zones than expected (fig. 3.5). This is likely due to the presence of multiple fault strands that show a significant displacement in total. However, the

damage zone width only scales with the individual fault strands (Childs et al., 2009). This also partially explains the log-log relationship often needed to plot the data (Kim and Sanderson, 2005).

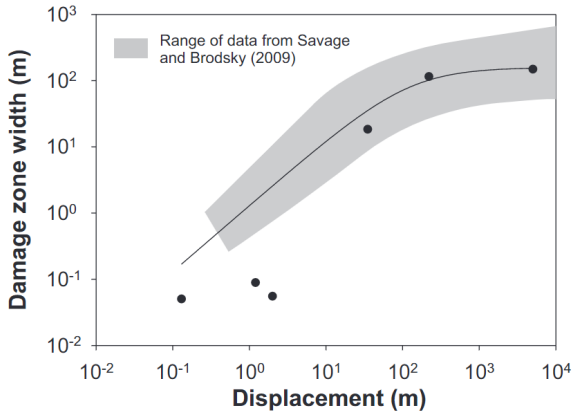


Figure 3.5: Damage zone widht as a funtion of fault displacement. After, Faulkner et al. (2011)

4

Percolation and Graph Theory

Fracture network connectivity and network percolation have been studied using various methods. In this chapter, the concept of percolation theory is discussed first. Hereafter, topology and topological connectivity are discussed. Lastly, the concepts of graph theory are discussed. This includes an explanation of the basic concepts of graph theory, an introduction to the largest component concept and a discussion of the concept of robustness.

4.1. Percolation Theory

Fault damage zone leakage through the caprock (assuming impermeable matrix) is a percolation problem; does an upward flow pathway from the reservoir through the caprock exist? Percolation theory has been used extensively to understand the connectivity of fractured media (Berkowitz, 1995; Berkowitz and Balberg, 1993; Hestir and Long, 1990). A great deal of this research has been performed with the use of lattice systems (fig. 4.1) where bonds exist with a probability p. The probability a connecting network that spans the system (backbone) exist is an increasing function of p and increases sharply from 0 to 1 within a small range of p values. The point where this transition occurs is called the critical percolation threshold (p_c). For an infinite 2D square lattice this critical threshold is reached at $p_c = \frac{1}{2}$ (Kesten, 1982), whereas for a 3D lattice the p_c drops to $p \approx 0.249$ (Stauffer, 1992).

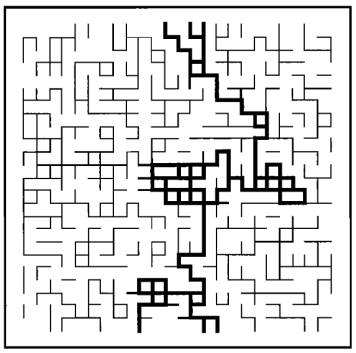


Figure 4.1: Example of a lattice network with a bond probability p = 0.525. The bold connections show a vertical backbone through the network. After, Renshaw (1999).

However, square lattices do not mimic the complex nature and strong spatial variation of fracture networks. Additionally, natural fracture networks are finite. Therefore, DFN are used to predict fracture network connectivity and percolation potential.

Studying fracture network percolation with the usage of DFN simulators can be very useful as it allows for modelling the effects of spatial variations and fracture geometries on the percolation potential. However, percolation modelling using DFN simulators has various issues. One of the most important issues is the fact that DFNs are finite systems. This significantly increases the uncertainty at which percolation is reached. This is illustrated by figure 6.3. In this figure, p indicates the probability that

one of the smaller grid blocks is percolating. In the leftmost figure p = 0.2 (20% of the grid blocks is percolating). Since the grid block that are percolating are perfectly aligned, they form a connecting path from left to right. Therefore, percolation through the entire system is achieved from left to right (P=1). In the rightmost figure p = 0.8 (80% of the grid blocks is percolating). However, they are not positioned in such a way that the allows for percolation through the entire system from left to right. Therefore, P=0. his highlights the issues of researching percolation with finite scale models. Therefore, Masihi et al. (2007) proposed that a cluster of fractures can be spanning through the network when its size becomes comparable to the system size.

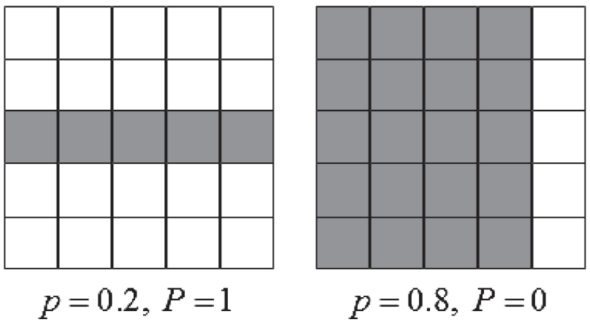


Figure 4.2: Finite size issue of modelling percolation. The case of the left shows an example where the percolation probability of the grid blocks is p=0.2. It shows that at this low p percolation (P=1) can be achieved through the system if they perfectly alligned. The case on the right show how for p=0.8 percolation might not be achieved if the grid blocks are not aligned well (P=0). After, Masihi et al. (2007)

4.2. Topology of Fracture Networks

Many authors have suggested using graph theory to characterise fracture networks (Hansberry et al., 2021; Manzocchi, 2002; Sanderson and Nixon, 2015,1; Sævik and Nixon, 2017). Using topology (a branch of graph theory that concerns strain-invariant relationships) to describe fracture networks ensures that interconnecting relationships between fractures are preserved (Faulkner et al., 2010; Sanderson and Nixon, 2015,1; Sævik and Nixon, 2017). Connections per line (C_L) has also been used as a connectivity measure since it is relatively easy to determine. However, there is no direct connection between C_L and the connectivity of the network as it does not consider topology (Sanderson and Nixon, 2015).

Many critical network properties such as connectivity are determined by topology rather than by fracture geometric properties. Therefore, describing a fracture network in terms of I-, Y- and X-nodes (fig. 4.3a) can be especially useful (Manzocchi, 2002). Additionally, studying exposed fracture networks by using the node classification becomes especially beneficial when fractures extend beyond the sample area (Sanderson and Nixon, 2015). In general three types of nodes can be distinguished; I-nodes, X-nodes and Y-nodes (fig. 4.3a). I-nodes represent fractures that end in the matrix. Y-nodes describe a fracture abutting another fracture. X-nodes are found when two fractures cross each other. It has to be noted that V-nodes, which are formed when two fracture tips meet, and nodes with higher degrees also exist. However, these are less common and play a less significant role in the characterisation of large-scale fracture networks.

Figure 4.4 illustrates why it is important to consider abutment relationships and why stochastic simulations often underestimate the connectivity of the network. Case A (fig. 4.4a) and B (fig. 4.4b) are both fracture networks with the same number of fractures with an equal orientation- and length distribution. However, case A is much more interconnected than case B. The amount of connection points, which is easily plotted in a ternary diagram, clearly illustrates this connectivity difference (fig. 4.3c).

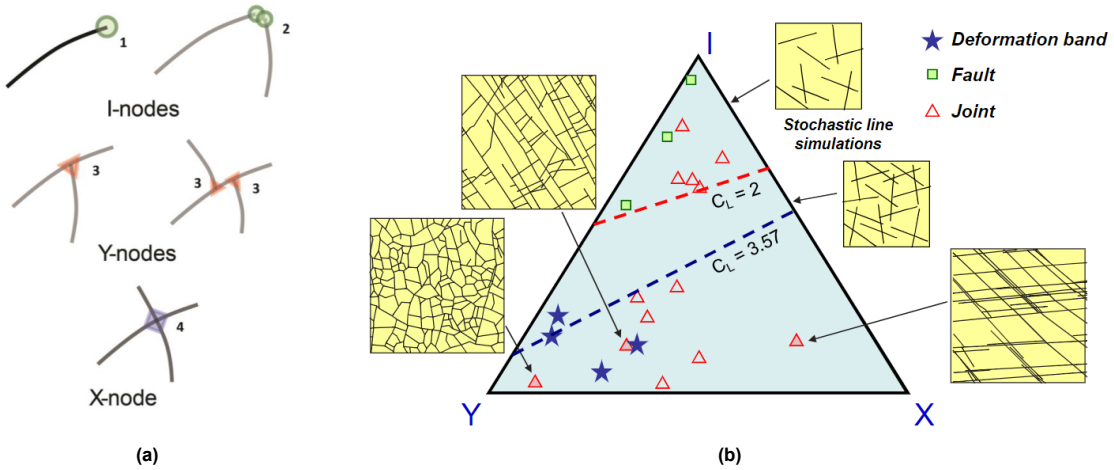


Figure 4.3: a) Different type of nodes, after Peacock et al. (2016). b) Various fracture networks plotted on a node type ternary diagram. After, Sanderson and Nixon (2015).

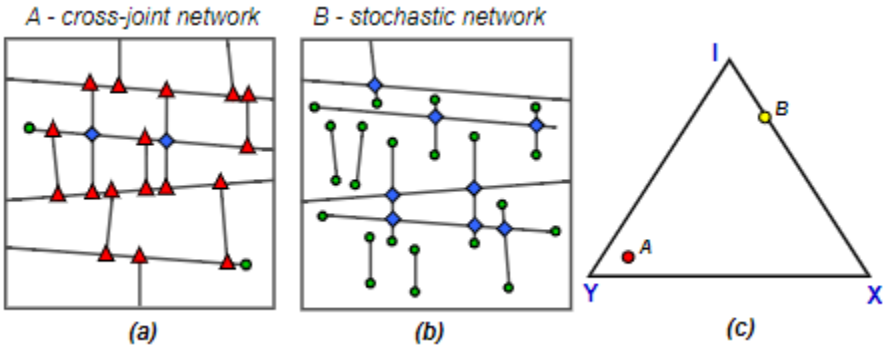


Figure 4.4: Network where there are the same amount fractures of fractures, same orientation and same length. However, they show different topologies. The purely stochastic model (B) does not take the topology into account and thus is not well connected. After, Sanderson and Nixon (2015)

4.3. Topological connectivity

In the previous section, it was explained why characterising fracture networks solely on fracture statistics does not capture the connectivity of the network. Therefore, a methodology taking fracture network topology into account is required.

Network connectivity is an important topic when studying the leakage potential of fracture networks. In subchapter 4.1 a short introduction to percolation theory is given. However, the use of topology allows for a new perspective on fracture network connectivity.

Sævik and Nixon (2017) proposed an analytical upscaling approach to find the network connectivity. With the network connectivity, the effective permeability can be found by using equation 4.1 with N fractures in a domain, V as the volume of the domain, T as fracture transmissivity (mechanical aperture times intrinsic permeability), A as fracture area and n as unit normal vector of the tangential plane. I is the identity matrix.

$$\mathbf{K}_{eff} = \frac{f}{V} \sum_{i=1}^{N} T_i A_i (I - \mathbf{n}_i \mathbf{n}_i) \tag{4.1}$$

The factor f (eq. 4.2), which denotes hydraulic connectivity in equation 4.1, incorporates the difference between the actual effective permeability and the theoretical upper bound and has a value between 0 and 1 depending on the connectivity.

$$f = max(0, 2.94 \cdot \eta'_C - 2.13) \tag{4.2}$$

Sævik and Nixon (2017) analytically derived five different measures of topological connectivity (η) and found through comparison to a numerical model that η'_C (eq. 4.3) most accurately describes the relationship (fig. 4.5). The factor η'_C is based on connection per fracture, but X-nodes are treated as half Y-nodes and V-nodes have no effect (although there is higher tortuosity). When $\eta'_C < 2.13/2.94$ the fracture network is under the percolation threshold and thus has no hydraulic connectivity.

$$\eta'_C = \frac{4n_x + 2n_Y}{4n_X + 2n_Y + n_I} \tag{4.3}$$

Moreover, it is important to note that there is a significant discrepancy between the synthetic data, created by considering a variety of patterns with different fracture intensities, and the natural data obtained from an outcrop (southern margin of the Bristol Channel Basin, north Somerset, UK) in figure 4.5. The reason given by Sævik and Nixon (2017) is as follows: "the outcrop map contains more isolated branches and dangling ends than synthetic networks of comparable fracture intensity, due to unconnected en-echelon arrays and fragmented fracture trace".

Since η'_C, and therefore the hydraulic connectivity, are only dependent on the ratio between X-, Y- and I-nodes it can be plotted in a ternary diagram (fig. 4.6).

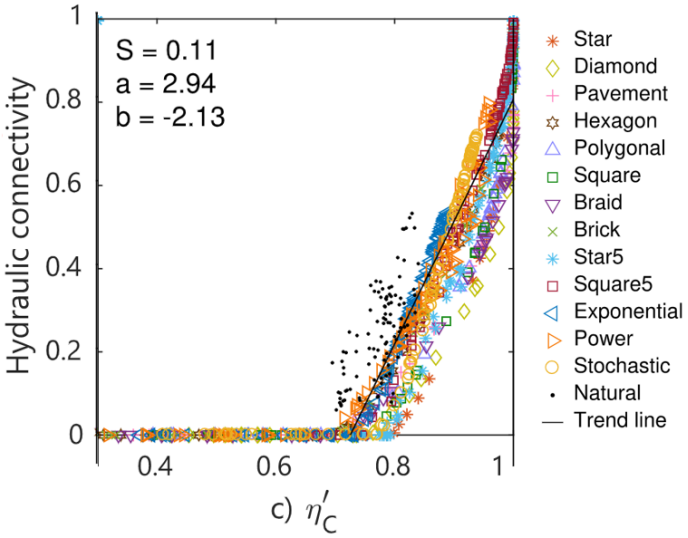


Figure 4.5: Hydraulic connectivity is plotted as a function of topological connectivity. The hydraulic connectivity is determined numerically for 12 different fracture patterns including a natural fracture network (see publication for images of the patterns). Topological connectivity (η'_C) is derived analytically and shown in equation 4.3. From regression equation 4.2 is found. Note the large deviation of the natural network to the linear trend line. After, Sævik and Nixon (2017).

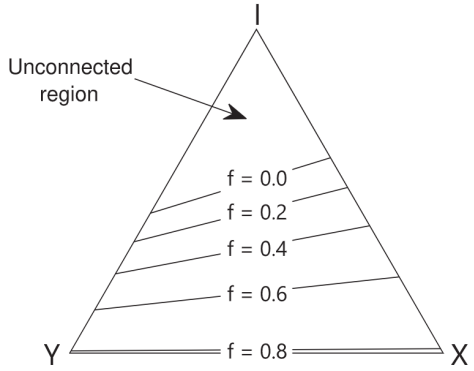


Figure 4.6: The hydraulic connectivity as a function of the number of I-, Y- and X-nodes. After, Sævik and Nixon (2017).

The Sævik and Nixon (2017) analytical equation (eq. 4.2) is an easy and fast method to determine hydraulic connectivity. However, it does have a major limitation. Consider the case illustrated in figure 4.7a with 10 X-nodes, 20 Y-nodes and 16 I-nodes. By applying the Sævik and Nixon (2017) method, we find the hydraulic connectivity to be 0.32. This appears logical since a well-connected example is illustrated. Consider the same network again, but 10 completely isolated fractures (I-I fractures) are added to the study area (fig. 4.7b). The hydraulic connectivity, as defined by Sævik and Nixon (2017) decreases with each additional I-I fracture (fig. 4.7c). When 8 isolated fractures are added the hydraulic connectivity drops below the percolation threshold as defined by Sævik and Nixon (2017). However, does the real hydraulic connectivity of the network actually decrease? If hydraulic connectivity is defined as "the ability of a fluid to flow through a region", it did not decrease. The original network, which allowed for flow to occur, is still intact in the scenario depicted in 4.7b. If this logic is followed the additional I-I fractures should not be considered. The hydraulic connectivity as determined Sævik and Nixon (2017) therefore reduces to a measure that determines the connectivity of the fracture network cluster that spans through the area.

This reasoning also explains the deviation of the numerically and topologically determined hydraulic connectivity of the natural network considered in figure 4.5. The I-I nodes that are present in the natural outcrop decrease the value of η_c' causing a shift in figure 4.5.

Additionally, this issue also affects the capabilities of the usage of the hydraulic connectivity model when using well data. When a well is drilled it is unknown whether the nodes that are found belong to the spanning cluster or are part of the fracture network section that should be disregarded.

Since it is argued that the Sævik and Nixon (2017) model only helps to determine the connectivity of the cluster that spans through the region, it is only useful to determine the amount of leakage that can be expected when this pathway exists. However, it is not an accurate measure to predict the chance of leakage at a location. This is related to whether a cluster of fractures exists that can span the entire area. Therefore, we require a measure of connectivity that can determine whether a spanning cluster can exist or not.

In an attempt to address this problem, we make use of the graph theory concept of the giant component. In the following chapter, each cluster of fractures that is isolated from other clusters is called a component. The largest, or giant component, will most likely be the component that causes leakage to occur. Therefore, we aim to find a relationship between topology and this large component.

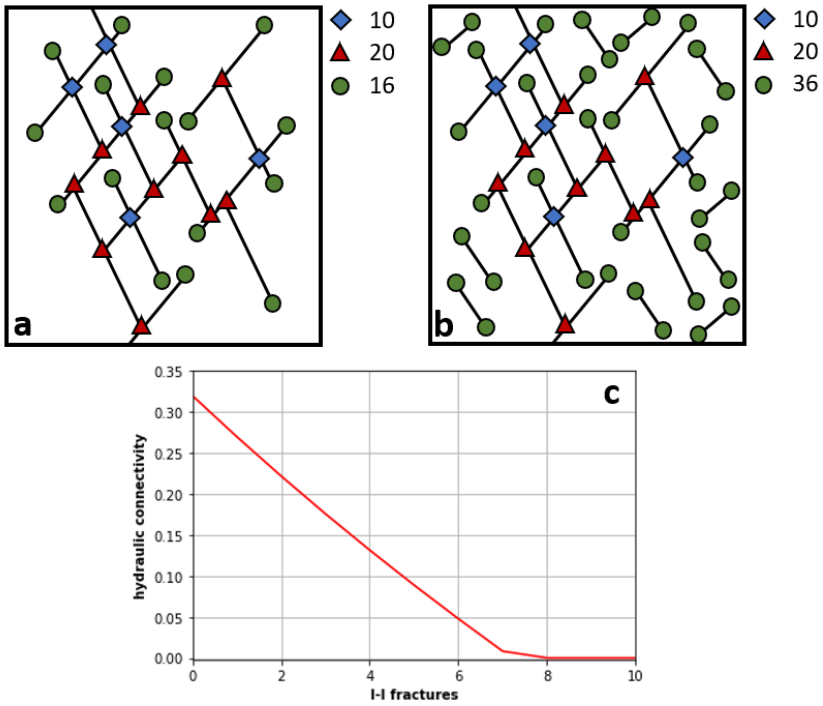


Figure 4.7: a) Connected network with X-, Y- and I-nodes. b) The same network, but with 10 additional I-I fractures surrounding the network. c) Plot that shows the impact of adding I-I fractures on the hydraulic connectivity as defined by Sævik and Nixon (2017)

4.4. Graph Theory

An alternative way of assessing connectivity is through the use of graph theory. In graph theory and network science a graph (G) consists of nodes (v) that are connected by edges (e). Graphs are typically represented in adjacency matrices (A) which, in their most simplistic form, are constructed with the criteria:

$$A_{ij} = \begin{cases} 1 & \text{if node } v_i \text{ is adjacent to node } v_j \\ 0 & \text{otherwise.} \end{cases} \tag{4.4}$$

The degree (k) of a node is equivalent to the number of edges connected to it. In the case of spatial networks, such as fracture networks, constraints exist on the maximum degree of a node (Barthelemy, 2018). The average node degree (k) in a graph can be related to the node classification as used in the previous section by using the following equation:

$$\mathrm{k} = \frac{4 * N_X + 3 * N_Y + N_I}{N_X + N_Y + N_I} \tag{4.5}$$

Fracture networks can be depicted in graphical form in various ways. Firstly, the fracture intersections can be considered as the graph nodes and the fracture segments between fracture intersections as edges. This form of describing the network is also known as the primal representation. Secondly, the fractures themselves can be considered as nodes and their intersection as edges. This is known as the dual representation (Barthelemy, 2018). Both have been used to study the connectivity of fracture networks (Andresen et al., 2013; Bisdom et al., 2017; Valentini et al., 2007). The use of dual form is especially helpful when studying the concept of degree centrality which has proven to be an important parameter for permeability estimations. However, the centrality of a fracture is impossible to determine when the fracture extends beyond the sampling domain. Additionally, significant differences exist between natural and stochastically generated networks (Andresen et al., 2013). Therefore, preference has gone to using the primal form. This approach has led to a classification scheme based on X-, Y- and I-nodes which was discussed in section 4.2. The primal form will be used in the rest of this thesis.

Additionally, weights (w) can be applied to the graph edges. Weights can specify the strength of the connection between two nodes. This results in e_{ij} values between 0 and ∞ in the adjacency matrix and can be based on various characteristics such as distance or specific connectivity between nodes.

A graph is considered a planar graph (G_p) when the spatial location of the nodes in two-dimensional cartesian space is specified.

A collection of connected nodes is known as a component. A network often consists of multiple components. This is especially true for natural fracture networks where large components (fracture clusters) are often isolated from one another. Once a connecting edge is introduced the two components merge into one larger component.

4.4.1. Connectivity in graph theory using the giant component concept

Graph connectivity can be determined in several ways. k-connectivity is one of the simplest measures of connectivity and can be applied to edges (k-edge) and nodes (k-node) (Diestel, 2005). The betweenness centrality (c_B) (Freeman, 1977) is a measure to evaluate the relative importance of a node or an edge for the interconnectedness in a spatial graph (a more extensive discussion on the betweenness centrality is given in appendix B).

Connectivity in graph theory is also associated with percolation theory. A network is completely unconnected when $p = 0$ for e_{ij}. A graph is completely connected when $p = 1$ for e_{ij}. For $0 < p < 1$ different regimes can be identified and the emergence of a so-called giant component can be observed. Consider an Erdős–Rényi random graph (Erdos et al., 1960), which is also known as a Poisson or binomial random graph. For these graphs p varies between 0 and 1 for e_{ij}. A random ensemble of these graphs can be generated with different values of p. This results in a random ensemble of graphs with different connectivity. The connectivity in these graphs is determined by looking at the number of components that are present in the graph. A component is group of edges that are connected to each other but not to the rest of the graph. When two nodes of different components are connected by an edge they merge into one component. So, if $p = 1$ for e_{ij} there is only one component. If $0 < p < 1$ for e_{ij} there are multiple components and a certain type of regime can be found. These different regimes when considering an Erdős–Rényi random graph for $0 < p < 1$ are shown in figure 4.8 where the y-axis represents the size of the giant component as a fraction of all components. On the x-axis average node degree (k) is represented. This is essentially the same as p, because if the number of connections increases, the average degree of all the nodes also increases. For $0 < k < 1$ all components are small, tree-like components. Therefore, the size of the giant component is small compared to all components. This makes sense as most nodes either have 1 or no edge connected (for a more complete explanation the reader is referred to Kang and Petrasek (2015)). For $k > 1$ a giant component emerges. For $k >> 1$ all components are incorporated into the giant component (Kang and Petrasek, 2015).

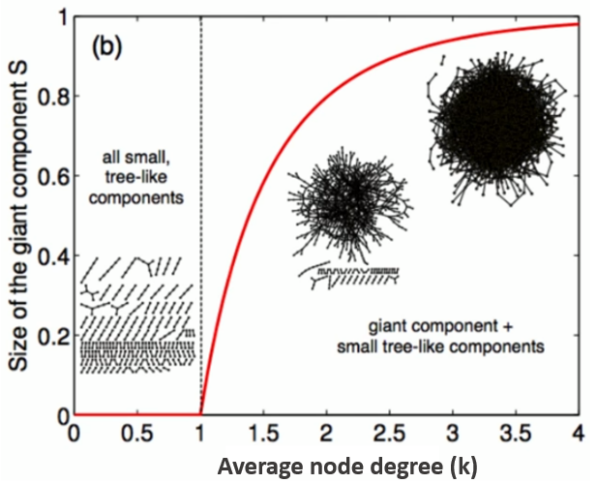


Figure 4.8: Phase transition as function of average node degree for a Erdős–Rényi random graph with N $\rightarrow \infty$. After, Barabási (2022)

4.4.2. The robustness of graphs

The theory of robustness explores the impact of edge (or node) removal on network integrity. Graph robustness is used in a wide range of disciplines, for example, in engineering to design systems that can perform their basic function when certain components fail and in medicine to study virus mutations.

The removal of a single edge has a limited impact on the integrity (or connectivity) of the network. Removing several edges, however, can break the network into multiple components. The threshold where this break-up occurs is given by f_c. At this point, the giant component (the component that contains the majority of the edges) starts to collapse. The speed of this collapse depends on the topology of said network. For random networks (fig. 4.9c), where nodes have comparable degrees, under random failures this can be given by infinite-dimensional percolation theory. For scale-free networks (4.9d), where the degree distribution is given by a power law, the giant component will only disappear when all nodes are removed (Barabási, 2022).

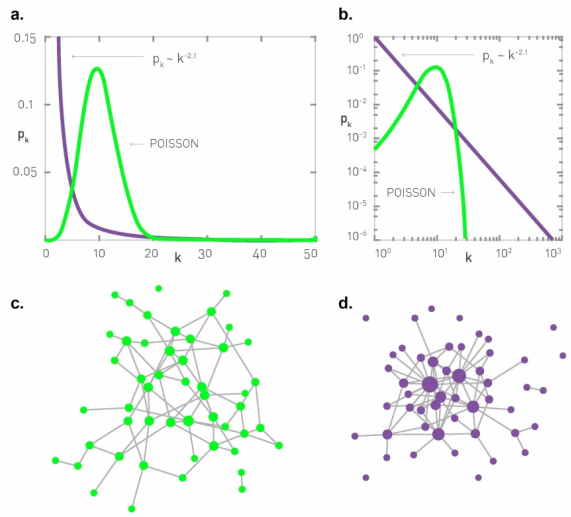


Figure 4.9: a) degree distribution of a random and scale-free network. b) degree distribution of a random and scale-free network on log-log scale. c) random network. d) scale-free network (Barabási, 2022)

4.5. The Largest Component Ratio

From section 4.1 we found that percolation can occur when the cluster becomes comparable to the system size. From section 4.4.1 we found that the connectivity of a graph can be determined with the use of the giant component analysis.

Fracture networks can be represented as a weighted graph with the euclidean distance between nodes (the fracture segments) assigned as weights. Therefore, the cluster that needs to be present in order to for the system to percolate, can be represented as a weighted component. It is most likely that this component is the largest component in the network.

Therefore, in this thesis, the ratio between the largest component and all other components is brought forward as a measure of connectivity. This measure, which is named the Largest Component Ratio (LCR) in this thesis, helps to understand how the connectivity of a fracture network influences the percolation risk. LCR can be formulated as:

$$LCR(l) = \frac{C_L(l)}{\sum_{i=0}^{n} C_i(l)} \tag{4.6}$$

where $C_L(l)$ is the total size of the largest component and $C_i(l)$ is the size of component i. The sum of all components (n) gives the total length of all fractures in the system.

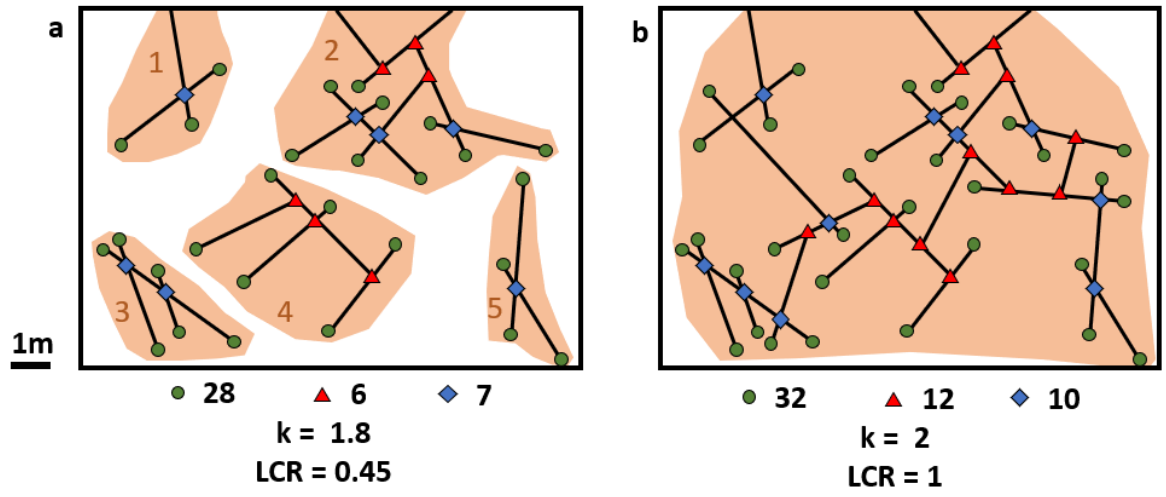


Figure 4.10: a) An example of an relatively unconnected fracture network with 5 components. In this example 2 is the largest component. b) A connected fracture network where only 1 component is present.

To clearly explain the concept of LCR with edge length as weight consider figure 4.10. In figure 4.10a 5 components are shown. For every component, the total length of all the fractures can be calculated. For example, component 1 has a total length of 5m (2.5m+2.5m). By comparing the length of all the components the largest component can be determined. In this case, it is component 2 with a total length of 23m. When the largest component is determined the largest component ratio can be determined as follows:

$$LCR = \frac{23}{5 + 9.5 + 13.5 + 23} = 0.45$$

The average node degree for figure 4.10a can be determined as follows:

$$k = \frac{28 * 1 + 6 * 3 + 7 * 4}{28 + 6 + 4} = 1.8$$

Figure 4.10b only has 1 component due to the connections made between the different components. Therefore, LCR is 1, and the graph is totally connected. This figure also illustrates how spatial connectivity influences the percolation risk. This is shown by the fact that figure 4.10b, with lower spatial connectivity is not percolating and figure 4.10c, with higher spatial connectivity, is percolating.

5 Methodology

This thesis aims to understand and predict the probability of caprock leakage through a fault-related fracture network. The ultimate goal is to find a methodology by which the connectivity and its uncertainty can be determined using only sparse data obtained by, for example, wells.

First, a literature review was performed. Chapter 2 highlighted that fracture networks often form with specific geometric relationships between fractures. Chapter 3 highlighted that FDZ's are often formed by a large combination of processes resulting in complex and unique fracture networks. Chapter 4 showed that in graph theory a relationship between connectivity and topology exists using the giant component concept. It also explained how this concept is related to the robustness of a graph.

To further investigate the connectivity of different (FDZ) fracture networks a study is performed based on both DFN simulations and the robustness of natural FDZ fracture networks. These two methods are illustrated in figure 5.1. It shows how a DFN generator builds up a fracture network by placing fractures into a 2D area. By placing more fractures, more connections are made. This is illustrated by an increase in the number of X-nodes and Y-nodes. By using the number of X-nodes, Y-nodes and I-nodes the hydraulic connectivity can be determined using equation 4.2. The number of X-nodes, Y-nodes and I-nodes can also be converted to the average node degree (k) by using equation 4.5. The LCR can also be determined for these networks using equation 4.6. Lastly, it can be assessed whether the fracture network percolates or not (which is the case for 5.1c).

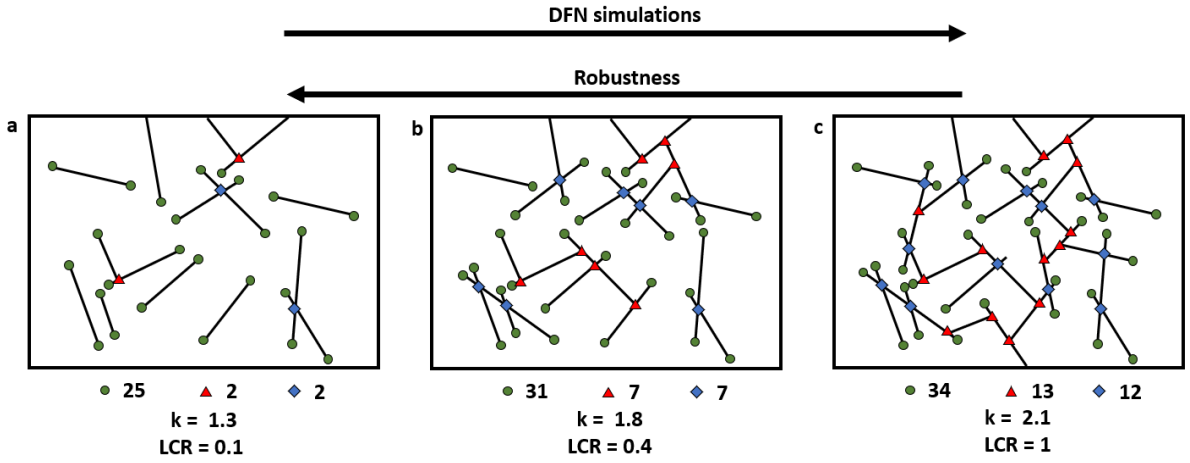


Figure 5.1: DFN simulations build up the fracture network by placing fractures into the 2D area. Throughout this processes the connectivity increase. The principle of robustness works in the opposite way and takes out fracture segment from the network and therefore reduces the connectivity.

So, by using a DFN generator we can determine how the measure of hydraulic connectivity and LCR relate to the topology and the connectivity of the fracture network. Additionally, it can validate whether the percolation threshold as determined by Sævik and Nixon (2017) is accurate. A DFN generator, which is made by Dr. K. Bisdom (Shell Global Solutions) with small adjustments by the author, is used. The inner workings of this DFN generator and the input parameters used for the simulations are described in section 5.1.

When these aspects are determined the robustness methodology can be applied. The robustness methodology essentially does the same as the DFN generator, but then in the opposite direction. It removes fractures from the 2D area. By doing this, the number of fracture connections drops. By taking the fracture network apart the connectivity drops as it breaks into multiple components. This is reflected in a reduction in LCR. Since the robustness methods can start with a full network, real fracture networks can be considered for this study. This allows us to study the connectivity behaviour of various (FDZ) fracture networks and can highlight their differences.

In chapter 5.2, the natural outcrop robustness algorithm is described. This algorithm was completely made by the author of this thesis. In this section, several FDZ fracture network outcrops are shortly introduced which were used for the robustness algorithm. Thereafter, two different variations of the robustness algorithm are discussed; the removal of fractures based on orientation (orientation-based reduction) and length (length-based reduction). Lastly, to study the sensitivity of the robustness methodology a partly synthetic scenario is introduced.

5.1. Discrete Fracture Network generator

A Python-based lightweight 2-dimensional DFN generator is developed by Dr. K. Bisdom with small adjustments from the author. This generator is capable of generating large amounts of DFN which makes it suitable to perform sensitivity analysis of various properties. In this generator, an area (which is set to 20m by 20m) is stochastically populated with 2 sets of fractures. First, the fractures of set 1 are introduced after which the fracture of set 2 are distributed over the area. Each set is populated using a range of criteria and the abutment relationship between the two sets can be specified. The criteria used in populating the area include: mean angle, orientation concentration, size power-law exponent, maximum size, minimum fracture length, P21 and stress shadow radius (the radius around fracture in which no other fracture can be placed).

The orientation distribution used in the model is based on the Von Mises orientation distribution (5.1). With $I_0(\kappa)$ being a Bessel function of order 0, μ being the average orientation, κ the measure of concentration and x the fracture orientation.

$$f(x|\mu,\kappa) = \frac{e^{\kappa \cos x-\mu}}{2\pi I_0(\kappa)} \tag{5.1}$$

The fracture length distribution is based on a Pareto power law distribution (eq. 5.2) where x_m denotes the minimum fracture length and α is the shape parameter (or tail index). x is the fracture length. Furthermore, $f = 0$ for $x < x_m$.

$$f_X(x) = \frac{\alpha x_m^{\alpha}}{x^{\alpha+1}} \tag{5.2}$$

Using the relationships described above the entire area can be populated with fractures. However, the DFN generator is also capable of generating FDZ. This is done by specifying whether set 1 and/or set 2 are fault-related deformation. If this is specified then the fractures are not populated homogeneously over the 2D area, but with the usage of the exponential decay function as defined by Savage and Brodsky (2011) (chapter 3.5). By using this function more fractures are placed near the fault plane. Therefore, a location of a fault plane needs to be assumed in the 2D area. The fault plane is placed at the leftmost boundary. One FDZ DFN realisation is depicted in figure 5.2 where set 1, indicated in black is distributed homogeneously over the area. Set 2, indicated in red, is distributed following the FDZ scaling. Therefore, significantly more fractures of set 2 are placed on the left side (near the fault plane) of the 2D area.

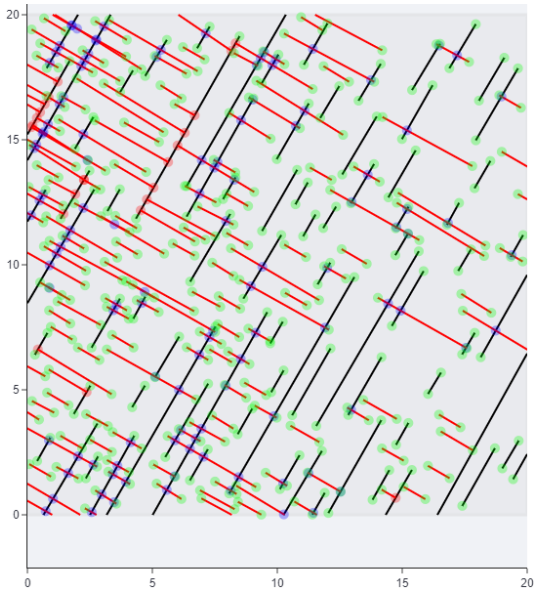


Figure 5.2: Typical FDZ DFN realisation. Set 1 (black) is homogeneously distributed over the area. Set 2 (red) is distributed across the area using a FDZ decay function. Therefore, more fractures are placed near the fault plane (leftmost boundary of the area)

The DFN generator is capable of generating large numbers of DFN. Each DFN can have its own combination of input parameters. Therefore, a diverse set of fracture networks can be studied. Several properties are found at the end of every realisation. These properties are related to one of the two sets or the entire network. Data that is collected for the two individual sets include average fracture length, maximum length, fracture intensity (P21), and average angle. For the entire fracture network the following properties are found: node distribution (X- Y- I-nodes), average node degree, Saevik and Nixon hydraulic connectivity (following eq. 4.2), LCR (eq. 4.6) and percolation potential.

The percolation potential is determined by using the Dijkstra path-finding algorithm (Dijkstra, 1959) which finds a pathway from the bottom to the top of the 2D area. The percolation potential is then calculated by first binning the simulations based on their average node degree. These bins all have a size of 0.1. So, for example, one bin spans from 2.0k to 2.1k. For every bin, the ratio between the amount of realisation that are percolating to all realisations in this bin is calculated. For example, if there are 100 simulations that have an average node degree between 2.0 and 2.1, and 10 of those are percolating. Then the percolation potential is 0.1 for this bin.

5.1.1. Finding Correlations

The DFN is capable of generating a large range of fracture networks with a diverse set of input parameters. To study the impact of different parameters on each other we can study the correlation coefficients. For all simulations above the correlation between different parameters can be depicted using a correlation matrix. The method that is used in this analysis is the Spearman's rank correlation. In this correlation, the values are ranked from high to low. From this ranking, a correlation coefficient is determined. This allows for the recognition of non-linear correlations. The Spearman's rank correlation can be defined as (Gauthier, 2001):

$$r_s = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n^3 - n} \tag{5.3}$$

The interpretation of the resulting Spearman's rank correlation coefficient is given in figure 5.3 (Schober et al., 2018).

Absolute magnitude of the observed correlation coefficient	Interpretation
0.00-0.10	Negligible correlation
0.10-0.39	Weak correlation
0.40-0.69	Moderate correlation
0.70-0.89	Strong correlation
0.90-1.00	Very strong correlation

Figure 5.3: Interpretation of the Spearman's rank correlation coefficient. After, Schober et al. (2018)

5.1.2. DFN generator input paramters

To study the relationship between topology, connectivity and percolation a range of DFN's are generated. First, the input parameters of a range of simple fracture networks are discussed. The fractures are distributed 'homogeneously' over the entire 2D area for these realisations. These realisations form the basis to compare different types of connectivity.
Second, the input parameters of a range of fracture networks with constant background fracturing are discussed. In these fracture networks, one fracture set (set 1) has constant input parameters and is orientated with a constant angle to the vertical. The other fracture set (set 2) has varying input parameters. Additionally, set 2 is considered to be fault-related deformation and thus follows the FDZ fracture density decay function.
Third, two consistent geometric relationships are researched; orthogonal and conjugate. Similar to the case above, set 1 is considered to be background fracturing whereas set 2 is considered to be fault-related deformation.

Simple fracture networks

The DFN generator is used to realise 550 different simple fracture networks. These realisations are subdivided into 4 groups that test the effect of a specific parameter on connectivity (fig. 5.4). The first group (indicated in blue, fig. 5.4) research the impact of different maximum fracture lengths. The second group (grey) studies the impact of different abutment ratios. Group three (orange) again researches the impact of the maximum fracture size, but with an abutment ratio of 0.5. The last group (green) researches the impact of different power-law size exponents.

			Set 1					Set 2				
	code	Abutment ratio	Mean angle	Orientation Concentration	power-law size exponent	max size	P21	Mean angle	Orientation Concentration	power-law size exponent	max size	P21
Max. length	m1	0	0-180	10000	2	1	1	0-180	10000	2	1	1
	m5	0	0-180	10000	2	5	1	0-180	10000	2	5	1
	m10	0	0-180	10000	2	10	1	0-180	10000	2	10	1
Abutment	m10a.2	0.2	0-180	10000	2	10	1	0-180	10000	2	10	1
	m10a.5	0.5	0-180	10000	2	10	1	0-180	10000	2	10	1
	m10a.8	0.8	0-180	10000	2	10	1	0-180	10000	2	10	1
Max. length & Abutment	m1a.5	0.5	0-180	10000	2	1	1	0-180	10000	2	1	1
	m5a.5	0.5	0-180	10000	2	5	1	0-180	10000	2	5	1
	m10a.5	0.5	0-180	10000	2	10	1	0-180	10000	2	10	1
Size exponent	m10p1	0	0-180	10000	1	10	1	0-180	10000	1	10	1
	m10p1.5	0	0-180	10000	1.5	10	1	0-180	10000	1.5	10	1
	m10p2	0	0-180	10000	2	10	1	0-180	10000	2	10	1

Figure 5.4: Overview of all the input parameters used in the first stage of DFN simulations.

Constant background fracturing

The DFN generator is used to realise 4000 FDZ fracture networks (fig. 5.5). In these fracture networks set 1 is considered as background fractures. Therefore, these are distributed 'homogeneously' over the 2D area. They have consistent input parameters and a high orientation concentration. 4 different angles of the background fracture set to the vertical are considered; 0°, 30°, 60° and 90°. Set 2 fractures are considered to be FDZ fractures. Therefore, they are populated using the FDZ decay function (eq. 3.1).

They do not have constant input parameters and have varying mean angle, orientation concentration, size exponents, maximum size and P21.

			Set 1						Set 2					
	code	Abutment ratio	Mean angle	Orientation Concentration	power-law size exponent	max size	P21	FDZ	Mean angle	Orientation Concentration	power-law size exponent	max size	P21	FDZ
Background 0° with FDZ	B0	0-1	0	10000	2	10	0.5	no	0-180	1-1000	1-2	1.1 to 10	0-2	YES
Background 30° with FDZ	B30	0-1	30	10000	2	10	0.5	no	0-180	1-1000	1-2	1.1 to 10	0-2	YES
Background 60° with FDZ	B60	0-1	60	10000	2	10	0.5	no	0-180	1-1000	1-2	1.1 to 10	0-2	YES
Background 90° with FDZ	B90	0-1	90	10000	2	10	0.5	no	0-180	1-1000	1-2	1.1 to 10	0-2	YES

Figure 5.5: Overview of all the input parameters used in the FDZ stage of DFN simulations.

Consistent geoemtric relationships

The final DFN model that is concerned again divides the two fracture sets into background fractures and FDZ fractures. The background fractures are orientated with 30° to the vertical and have a high orientation concentration. The fractures of set 2 have varying orientation concentrations, size exponents, maximum size and P21. However, they do not vary in mean angle. Two mean angles are considered for the FDZ fractures. For the first scenario, the fractures are orthogonal to the background fractures (replicating an orthogonal fracture network). For the second scenario, the fractures are conjugate to the background fractures (replicating an orthogonal fracture network).

			Set 1						Set 2					
	code	Abutment ratio	Mean angle	Orientation Concentration	power-law size exponent	max size	P21	FDZ	Mean angle	Orientation Concentration	power-law size exponent	max size	P21	FDZ
Background with FDZ orthogonal	ORTH	0 - 1	30	10000	2	10	0.5	no	120	1 - 1000	1 - 2	1.1 - 10	0 - 2	YES
Background with FDZ conjugate	CONJ	0 - 1	30	10000	2	10	0.5	no	150	1 - 1000	1-2	1.1 - 10	0 - 2	YES

Figure 5.6: Overview of all the input parameters used in the FDZ geometric stage of DFN simulations.

5.2. Robustness analysis of outcrop fracture networks

FDZ fracture networks are diverse and are formed under a wide range of geomechanical processes which influence their topology and connectivity. To research their specific connectivity, an approach is introduced which is based on the concept of robustness, as discussed in chapter 4. The goal of this algorithm is to study the relationship between topological measures and the connectivity of a variety of fracture networks. Firstly, the robustness algorithm steps are discussed. Secondly, the outcrops that are considered in this analysis are introduced. Thirdly, two different variations of the robustness algorithm are discussed; orientation- and length-based reduction. Lastly, to study the sensitivity of the robustness methodology a partly synthetic scenario is introduced.

5.2.1. Robustness algorithm

To study the robustness, a Python-based algorithm is developed (Appendix C). The algorithm is schematically shown in figure 5.7. First, a digitised version of a fracture network line drawing is required. During the digitisation processes (done with Inkscape in this research) special attention needs to be given to drawing the contact points of fractures. This digitised fracture network can then be loaded into the robustness algorithm. The algorithm then removes one fracture segment from the fracture network. This fracture segment is randomly selected (orientation- and length-based selection are discussed in section 5.2.2). After the segment is removed several properties are recorded. These include node types, P21, LCR, orientation distribution and length distribution. Then, the algorithm removes another fracture segment and again records all the data. This process is repeated until all fracture segments are removed. When all fracture segments are removed one realisation is completed. From this realisation, we can obtain a relationship between different parameters. In figure 5.7 the relationship between LCR and the average node degree is shown. Because the removal processes of fracture segments is random every realisation is unique. Therefore, the process of fully removing an entire fracture network

is performed multiple times. This is done 10 times for all the analyses performed in this research. When the 10 realisations have been performed a curve can be fitted using a non-linear least squares method.

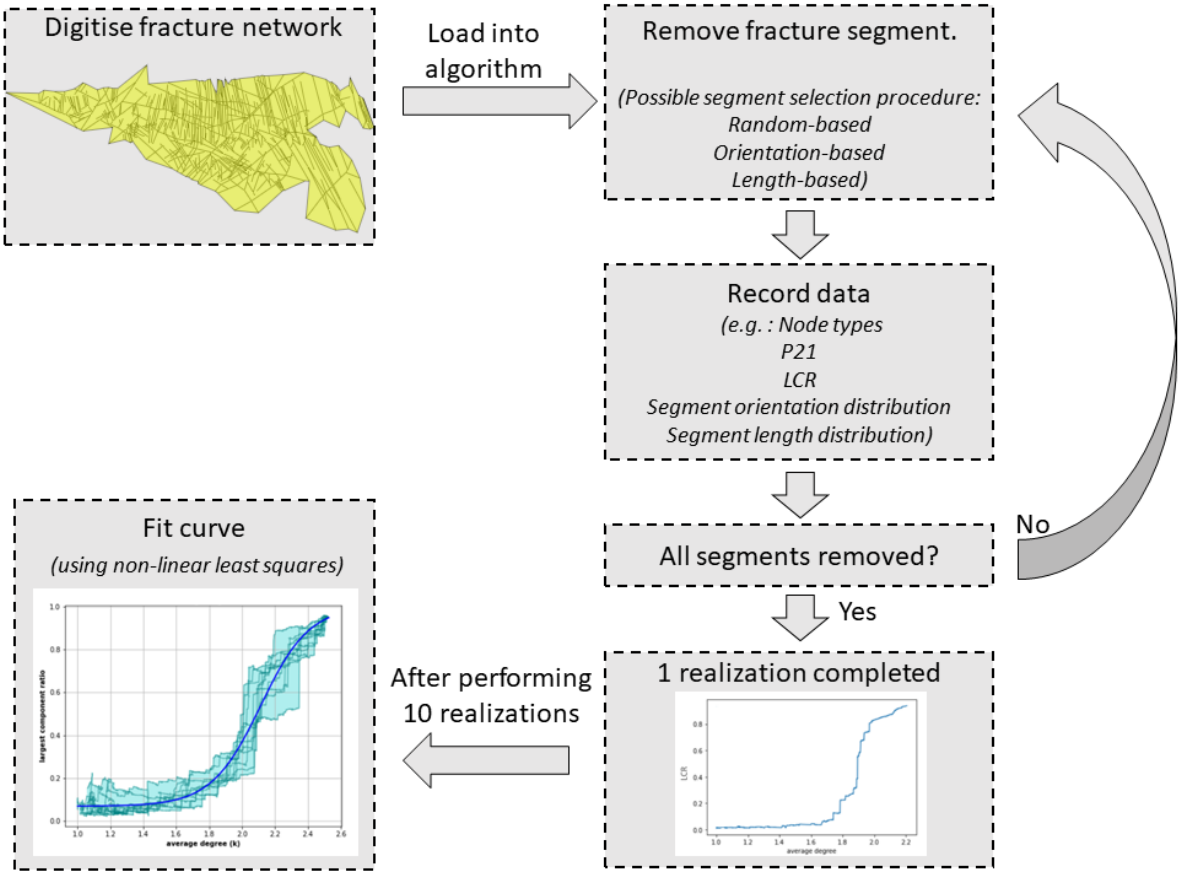


Figure 5.7: Flow chart of the robustenss algorithm. Full code available in Appendix C.

5.2.2. Orientation- and length-based removal

The selection process for which fracture to remove can be based on multiple selection criteria.

Orientation-based removal allows for the removal of fracture segments based on their orientation. This process allows for a geologically realistic reduction. This is because it can replicate the formation processes of a fracture network in the opposite direction. Consider figure 5.8 where fractures that are orientated sub-vertically are formed first. Hereafter, fractures with a sub-horizontal are formed. To accurately mimic the formation process in an opposite manner the fractures should be removed in the opposite direction. So, fractures that are sub-horizontally, should be removed first. When these are fully removed, the fractures that are orientated in the sub-vertical direction should be removed.

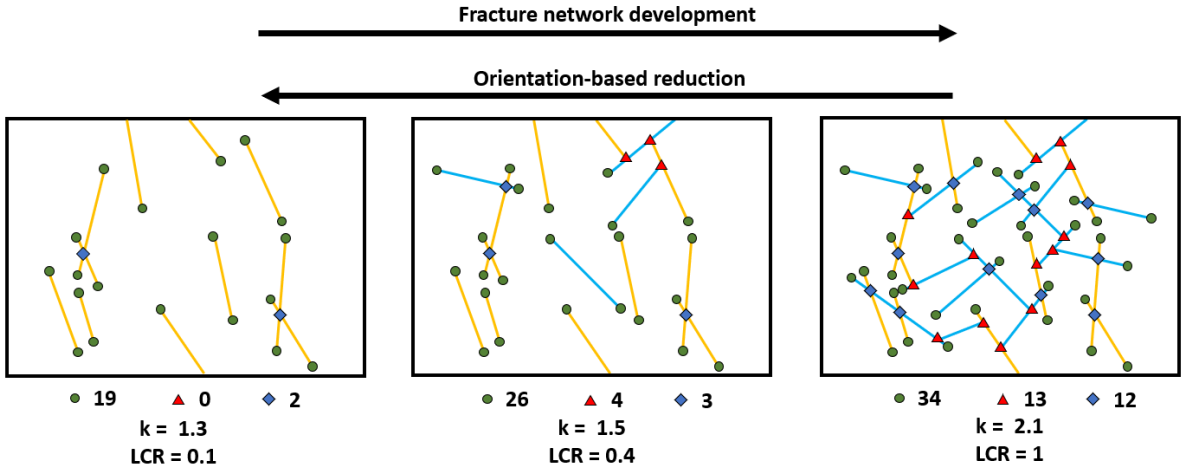


Figure 5.8: The fracture network intially only contains sub-vertical fractures. Later in the proces sub-horizontal fractures are formed. To replicate the relationship between k and LCR in an accurate way, the fractures should be removed in order of orientation.

Length-based removal allows for the removal in the order of length (fig. 5.9). Length-based segment reduction has less clear geological relevance. However, it could improve our understanding of the robustness method. It can show how the largest component reduces in size as the longest fracture segments are removed first.

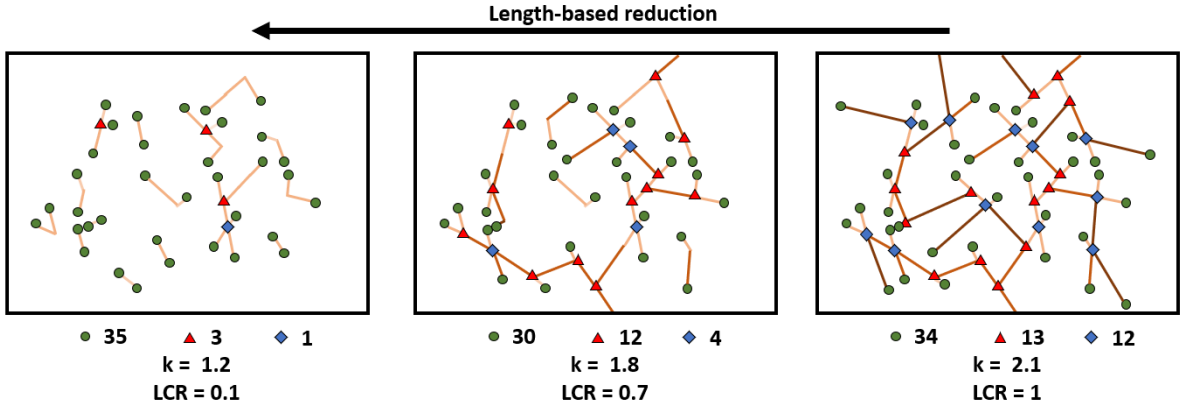


Figure 5.9: Image showing a fracture network with varying segment length. If the reduction process is length based the longest fractures are removed first.

5.2.3. Considered Outcrops

A variety of fault-related fracture networks are studied (fig. 5.10). It is important to note that, as we only consider the FDZ network in this research, we only analyse this part of the fault zone. Therefore, the faults are not depicted in these figures.

The analysed FDZ fracture networks can be subdivided (following the classification in 3.3.2) in fault tip zones, wall damage zones and linking damage zones. Most of these fracture networks were initially interpreted on a different scale which makes comparing fracture networks in terms of P21 difficult. Therefore, the areas (indicated in yellow) were normalised.

Fault tip

One fault tip zone is analysed using the robustness algorithm. This fault tip zone is shown in figure 5.10a. It shows an antithetic fault tip located at Dasht-e Bayaz, Iran (Kim et al., 2004; Tchalenko and Ambraseys, 1970). The fracture network, which is interpreted on a horizontal plane, is 500m in width

and 1km in length. It shows a large variation in fracture lengths and orientations. Additionally, certain regions of the fracture network are more densely populated with fractures than other regions.

Wall-damage

6 different wall damage fracture networks are analysed. A wall damage fracture network from Utah (USA) is shown in figure 5.10b. The fracture network, which is present the Moab formation (Johannessen, 2017) is 3m in height by 4m in length. Two sections are studied that are related to the extensional Konusdalen fault (Svalberg, Norway) (Rizzo et al., 2020). They are especially relevant for this research as the Konusdalen unit is a low permeable caprock analogue. The interpretation of these networks was done by Dr. R. Rizzo. Figure 5.10c shows the first FDZ section which is 30m in height and 50m in length. Figure 5.10d shows the second FDZ section and which is 150m in length and 50m in height.

A fracture corridor network from Languedoc (Southern France) was also analysed using the robustness algorithm. This fracture network is shown in figure 5.10e and is 2.5m in width and 4m in height. The FDZ, which is related to a small-scale normal fault, is formed in a tight carbonate lithology.

5.10f and 5.10g show fracture networks that are developed under low differential stress in Somerset (UK)(Miliorizos and Ruffell, 1998; Philipp, 2008). The fractures developed into veins due to the precipitation of gypsum. Both outcrops are 3m in height and 5m in length.

Linking damage zone

Two linking damage zones were studied using the robustness algorithm. Both these FDZ were interpreted from the top down (map view). Figure 5.10h shows a breached relay ramp from western Malta (Nixon et al., 2020). The fracture network, which is formed in carbonates, is 50cm in length and width.

5.10i shows a lens structure (Kim et al., 2004) that is part of a strike-slip fault in Dasht-e Bayaz, Iran (Tchalenko and Ambraseys, 1970). The fracture network is 1km in width and 4km in length.

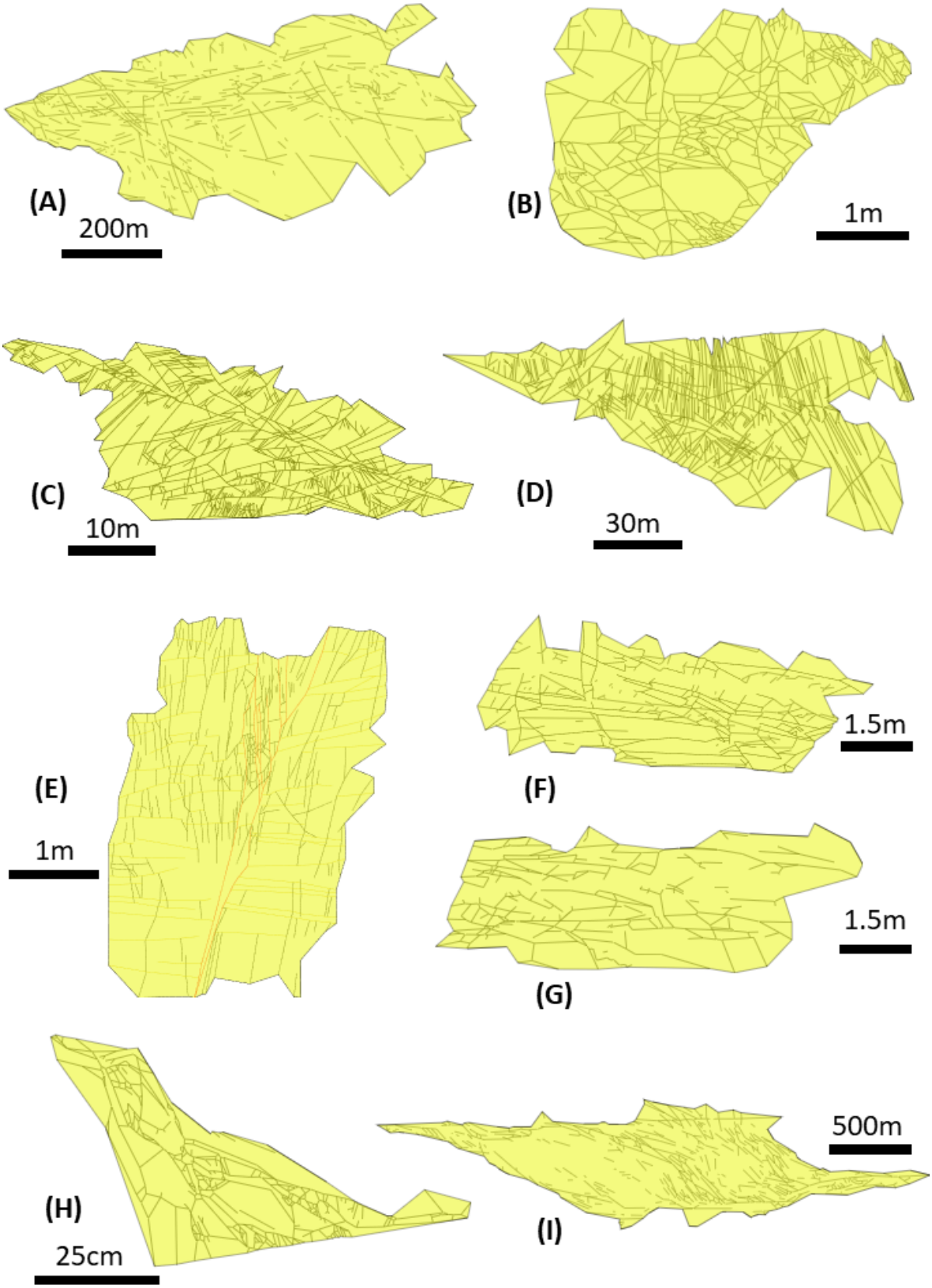


Figure 5.10: (A) Antithetic fault tip (B) Moab Fm. wall damage, (C) Konusdalen II wall damage, (D) Konusdalen I wall damage, (E) Fracture cluster damage zone, (F) Vein network at somerset B, (G) Vein network at Somerset A. (H) Breached relay ramp, (I) Lens structure.

5.2.4. Synthetic connecting fractures

To illustrate the behaviour of the LCR a fracture network in Somerset (UK) is considered (figure 5.11a). The outcrop has an abundance of fractures (white veins) that form a network. The fractures are predominately present in the red mudstone and the majority appear to stop on the silt-rich layers (Miliorizos and Ruffell, 1998; Philipp, 2008). Nonetheless, there are fractures present that are through-going. These fractures are important features that connect the lower fracture network section to the upper network section.

This fracture network is used to study the behaviour of LCR when the robustness algorithm is applied. This is done by artificially making the fracture network more connected. This is achieved by adding synthetic fractures through the silt-rich layer. These fractures (numbered and highlighted in blue) are shown in figure 5.11b. The robustness algorithm is used to generate 9 results. First, the original network is analysed by adding no synthetic. For the second simulation, 1 synthetic fracture is added and so forth.

In this way, we can study what happens to LCR when a fracture network is a little bit more connected. This can help us to assess the results of this thesis.

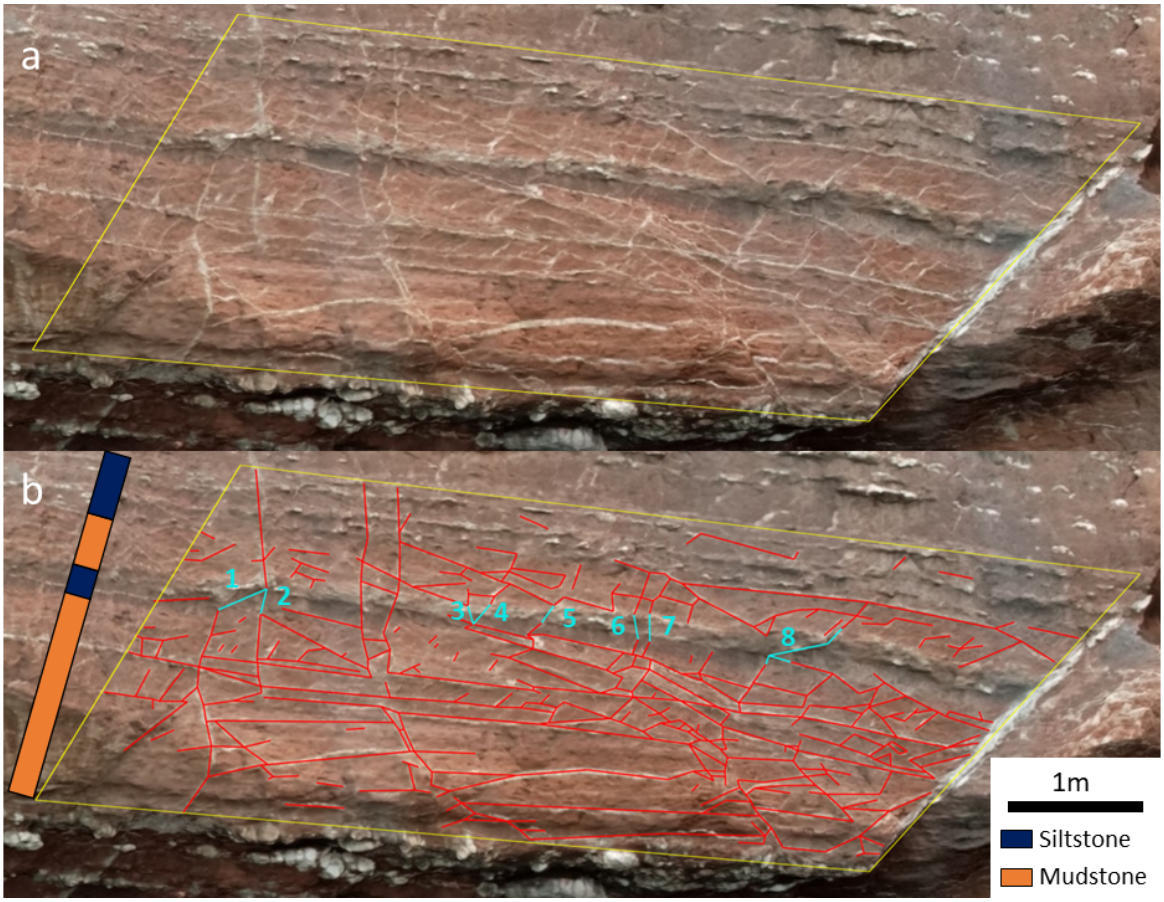


Figure 5.11: a) The fracture network present at somerset (UK). b) Line interpretation of the fracture network. The mudstone layer where most fractures are present is highlighted. The silt-rich layer, where almost no fractures go through is also highlighted. Through this silt layer, a total of 8 fractures are added that were originally not there. These fractures each increase the connectivity of the network by connecting the lower to the upper part of the network.

6

Results

This chapter shows the result of both the DFN simulations and the outcrop robustness methodology. The DFN simulations highlight how the connectivity of the network is captured based on topology. The robustness methodology helps to study and quantify the connectivity for a range of different FDZ outcrops.

First, the resulting DFN simulations are discussed. In this section, the relationship between topology and the two different measures of connectivity are tested; hydraulic connectivity as determined by Sævik and Nixon (2017) and LCR.

Hereafter, the results of the robustness algorithm are discussed. The results of the random-based analysis are shown for various FDZ outcrops. Then, the length- and orientation-based analyses are discussed. Lastly, synthetic through-going fractures are considered to study the behaviour of the LCR as a measure of connectivity.

6.1. FDZ generator

A DFN generator is used to study the impact of a variety of parameters on the connectivity of fracture networks. Special attention is given to the relationship between connectivity and topology.

Three different fracture network cases are tested. These cases include simple fracture networks, FDZ fracture networks with consistent background fractures and FDZ fracture networks with consistent geometric relationships.

6.1.1. Simple fracture networks

A total of 550 simple fracture networks are generated to analyse the relationship between different parameters and the percolation of a fracture network. Figure 6.1 shows the range of simple fracture networks that have been generated. These fracture networks range in maximum length, abutment relationship and power-law size exponent.

For all these simulations the angles of set 1 and set 2 were calculated, the number of I-, Y- and X-nodes were determined, the mean fracture length was determined and the network P21 was recorded. The distribution of these parameters are shown in figure 6.2. It can be seen that the angle between set 1 and set 2 range from 0° to 180°, network P21 ranges from 1 to 1.8 and the mean fracture length is between 1.0 and 3.5. For all simulations there a significantly more I-nodes present than Y-nodes or X-nodes.

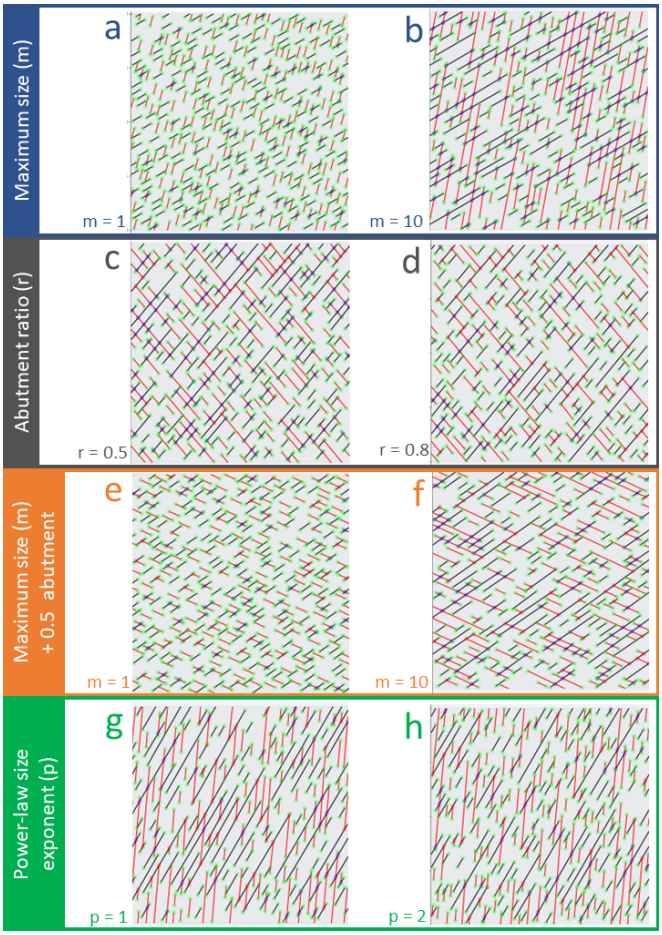


Figure 6.1: a) shows are realisation with a maximum fracture length of 1 b) shows a realisation with a maximum fracture length of 10 c) shows a fracture network with an 0.5 abutment ratio between set 1 and set 2. d) shows a fracture network with an 0.8 abutment ratio between set 1 and set 2. e) shows a realisation with a maximum size of 1 with a 0.5 abutment ratio. f) shows a realisation with a maximum size of 10 with a 0.5 abutment ratio. g) shows a realisation with a power-law size component of 1. h) shows a realisation with a power-law size exponent of 2.

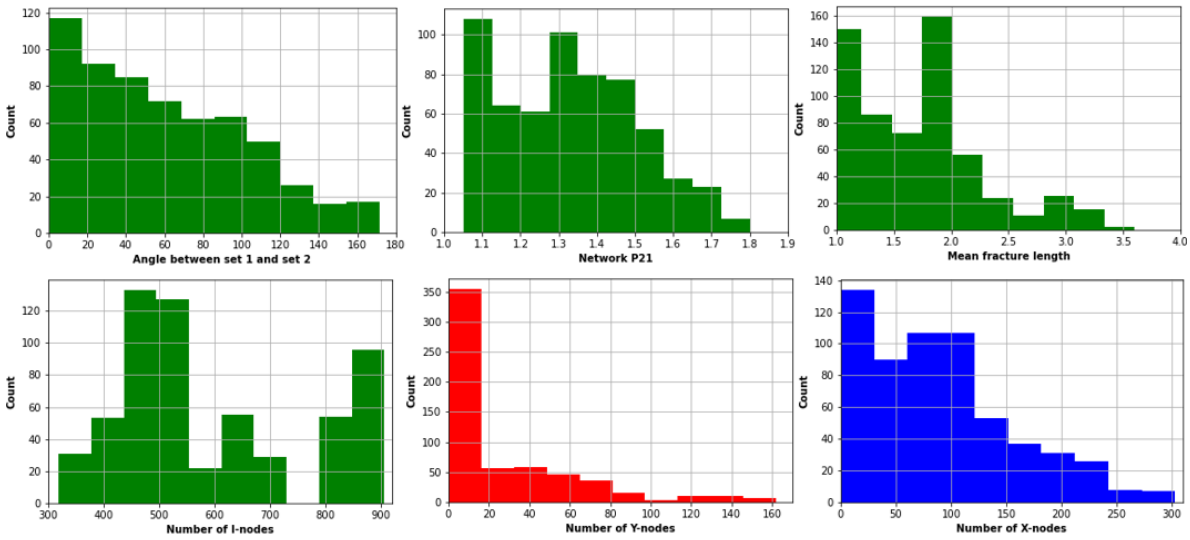


Figure 6.2: Distribution of parameters of the different fracture network. The angles between set 1 and set 2 range from 0° to 180°. The network P21 ranges from 1 to 1.8. The mean fracture length ranges from 1 to 5.5. The number of I-, Y- and X-nodes are also shown.

All the fracture networks were analysed on their percolation probability, hydraulic connectivity and LCR. Next, these parameters were related to the topological measure of average node degree (k).

The percolation probability was determined by binning the realisations based on their average node degree. The bins have a size of 0.1 degree. For every bin the share of simulations that were percolating was calculated. This results in the percolation probability curve shown in figure 6.3.

It can be seen that the percolation probability for k<1.5 is nearly zero. For k>1.5 the percolation probability increases significantly. For k>2.1 all simulations percolate.

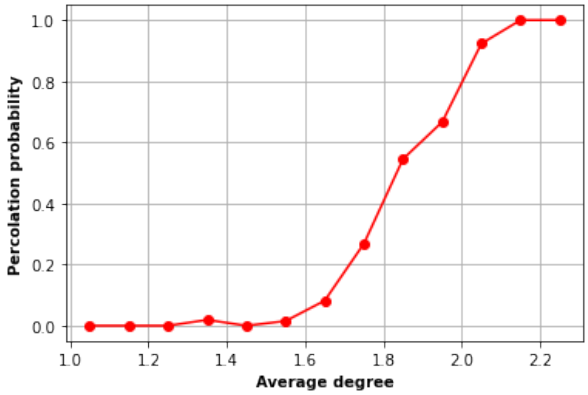


Figure 6.3: Percolation probability of the simple networks combined. The percolation probability start to rise significantly for k>1.6

Next, the hydraulic connectivity as determined by Sævik and Nixon (2017) (eq. 4.2) is calculated. The results are best depicted in a ternary diagram, as the hydraulic connectivity is only dependent on the ratio of I-, Y- and X-nodes. This hydraulic connectivity was calculated in two different ways. By including all the fractures in the network (fig. 6.4a) and by excluding the fractures that are not part of the (main) spanning component (fig. 6.4b). This will give two different results as removing the fractures that are not part of the spanning cluster, which typically have many I-nodes, increases hydraulic connectivity. In these plots, the blue points indicate realisations that are not percolating. The yellow data points indicate DFN realisations that are percolating. It can be observed that simulations that are present lower in the ternary diagram show percolation more often. However, many simulations that percolate are present below the hydraulic connectivity line of 0. Additionally, no clear trend can be found between the hydraulic connectivity and the percolation potential.

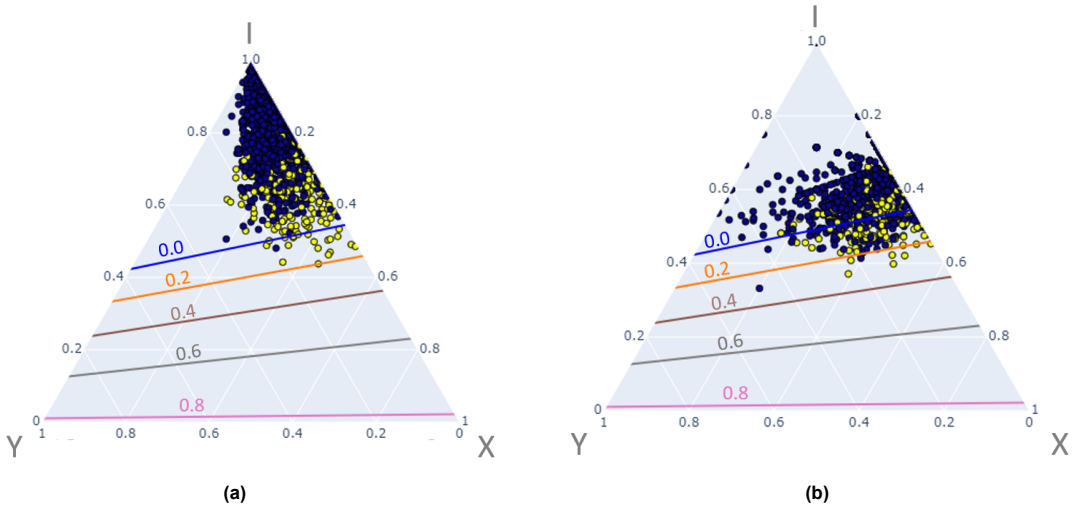


Figure 6.4: Yellow data points indicate simulations with a backbone. The blue data points indicate simulations without a backbone. The regions of the hydraulic connectivity are after (Sævik and Nixon, 2017) and use the simple relationship shown in equation 4.2

In order to compare the different measures of connectivity more effectively, the hydraulic connectivity relationship with k is illustrated in figure 6.5. Again, the hydraulic connectivity is plotted that was determined by using all the nodes in the entire area (fig. 6.5a) and only the nodes of the spanning component (fig. 6.5b). These plots more clearly illustrate that for k<2.0 (or k<2.2) almost all simulations are below the percolation threshold as determined by Sævik and Nixon (2017). In figure 6.5 the hydraulic connectivity shows a linear relationship with average node degree. This is because for simulations m10p1, m10p1.5 and m10p2 (indicated in green) there is an abutment ratio of 0. Therefore, the network only has I- and X-nodes.

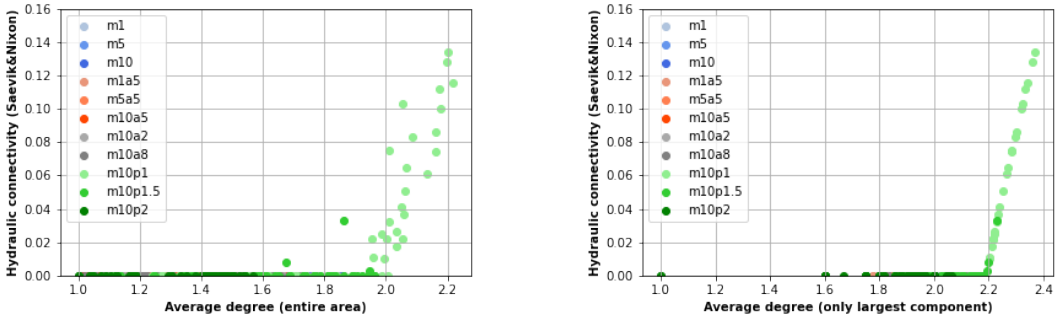


Figure 6.5: Hydraulic connectivity as defined by Sævik and Nixon (2017). a) In this calculation all the fractures in the network are used. b) In this calculation only the fractures in the largest component are used.

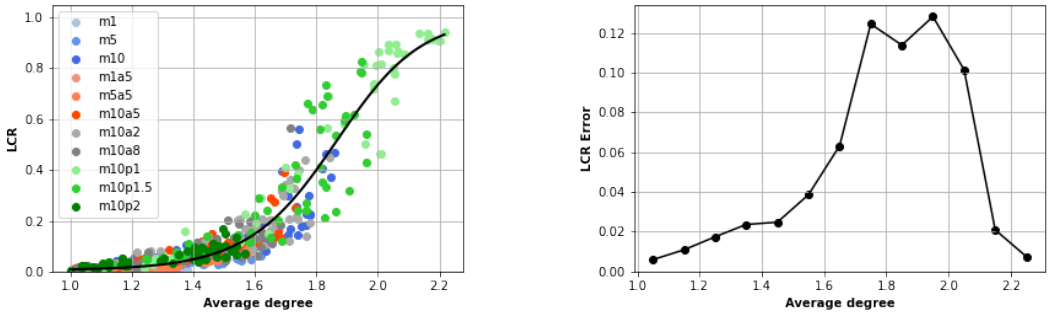
Figure 6.6a shows the relationship of LCR with k. It can be seen that for k>1.6 the LCR rises and it starts to plateau at k>2.1. This is similar to the percolation probability shown in figure 6.3.

By using a non-linear least squares method an s-cruve relationship (fig. 6.6a) between the average node degree and LCR is found (eq. 6.1). In this relationship, LCR denotes the largest component ratio and k denotes the average node degree of the network. k_m denotes the midpoint of the average degree s-curve. c and a control the shape of the curve.

$$LCR(\mathrm{k}) = \frac{1}{1 + e^{-c(\mathrm{k}-\mathrm{k}_m)^a}} \tag{6.1}$$

The deviation between the s-curve relationship and the data points is illustrated in figure 6.6b. This deviation plot is obtained by binning the realisations based on their average node degree in bins of 0.1 k. Then the average of the absolute deviation of the realisations to the s-curve is calculated. This plot, therefore, highlights the average deviation of data points from the s-curve relationship.

It can be seen that the spread in the data around the s-curve is relatively low for k<1.4. For k>1.5 the deviation of the data points to the s-curve start to increase. It reaches a maximum uncertainty at 1.7<k<2.1. For k>2.1 the uncertainty of the LCR estimation starts to decrease significantly.



(a) (b)

Figure 6.6: a) The relationship between the LCR connectivity measure and the average node degree (k). The data point all represent one simulated DFN. From the datapoints a s-curve relationship is found that starts to increase significantly for k>1.6 b) The deviation of the data points shown in (a) around the found s-curve. This illustrates the uncertainty in the connectivity at different average node degree. The uncertainty of the LCR-k relationship is highest for 1.7k<2.0. It has a maximum uncertainty of 0.13 LCR.

6.2. FDZ simulations with constant background fracturing

4 cases of FDZ's with constant background fracturing were generated. The first case has the background fractures (set 1) at an angle of 0° to the fault plane (fig. 6.7a). The second, third, and fourth case have a background fracturing with an angle of 30°, 60°, and 90° to the fault plane respectively (fig. 6.7 b,c,d). Each case is run 1000 times with varying input parameters for set 2 fractures. The distribution of the angles between set 1 and set 2, P21, mean fracture length and node types are shown in figure 6.8. The angles between set 1 and set 2 range from to 0° to 180°. The network P21 spans from 0.5 to 2.5 and the mean fracture length is between 1.5 and 3.0 for most simulations.

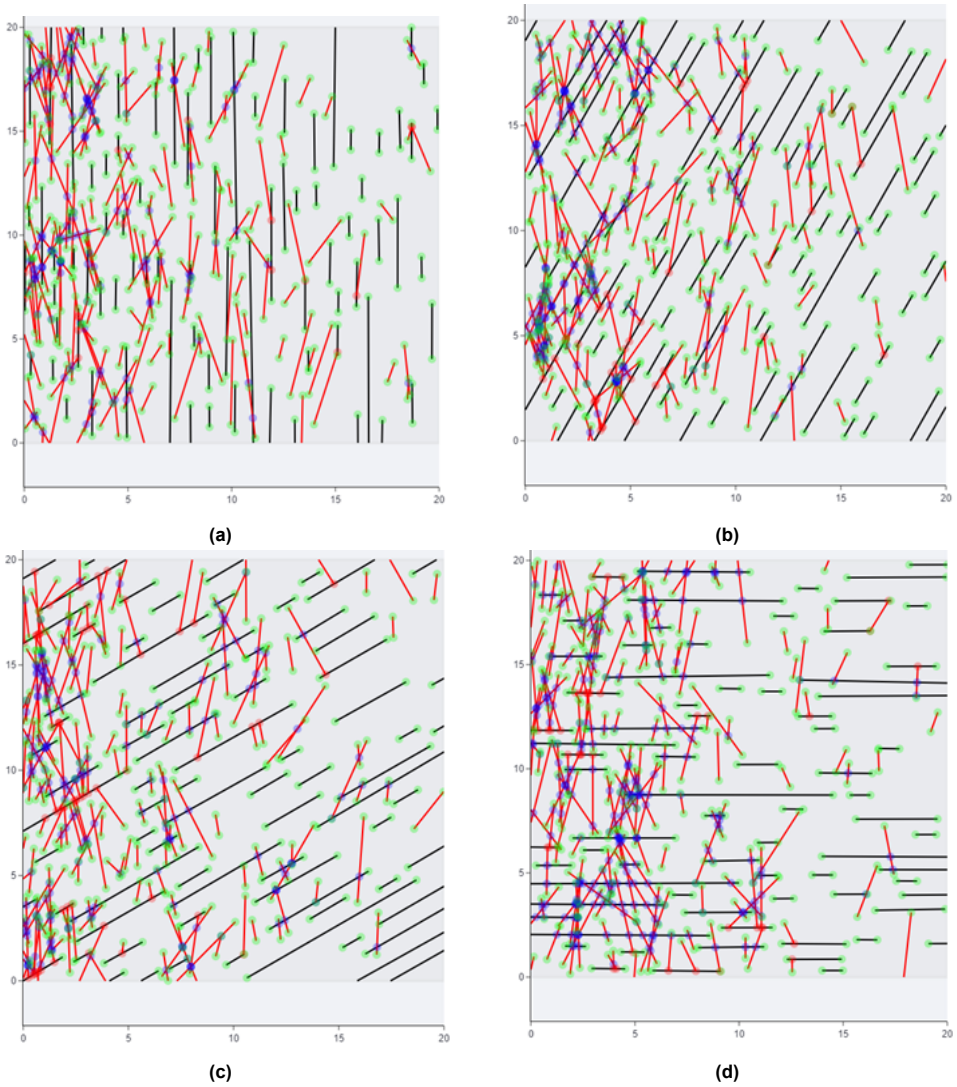


Figure 6.7: 4 cases of FDZ's with constant background fracuring. a) 0° to the fault plane, 30° to the fault plane, 60° to the fault plane, 90° to the fault plane

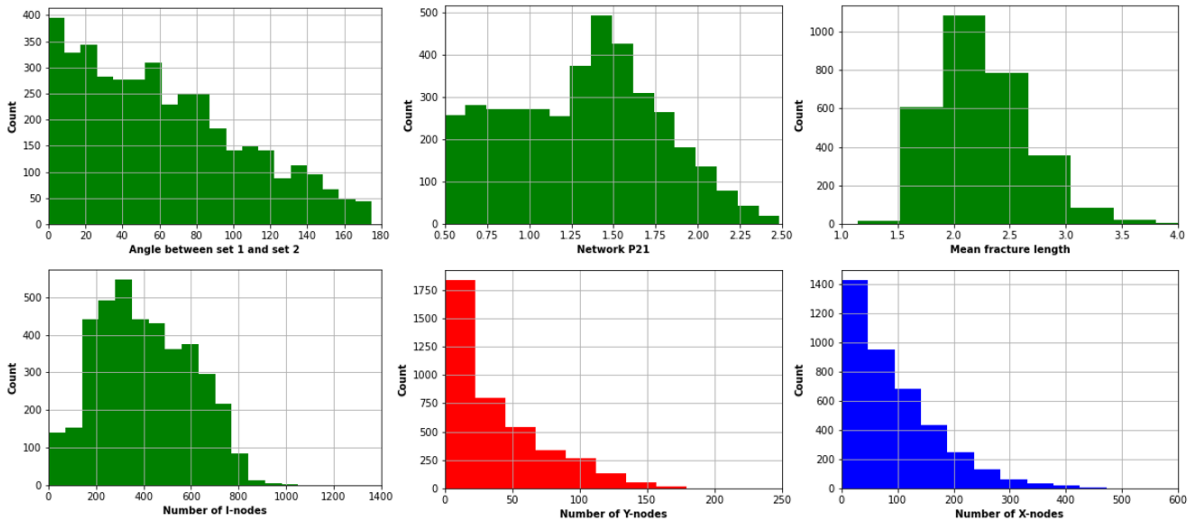


Figure 6.8: Distribution of parameters of the different fracture network. The angles between set 1 and set 2 range from 0° to 180°. The network P21 ranges from 0.5 to 2.5. The mean fracture length ranges from 1 to 4. The number of I-, Y- and X-nodes are also shown.

All the FDZ fracture networks were analysed on their percolation probability, hydraulic connectivity and LCR.

Figure 6.9 shows the percolation probability of the considered FDZ fracture network. The percolation probability is near zero for k<1.5. For k>1.6 the percolation probability increases significantly. For k>2.3 all simulations percolate.

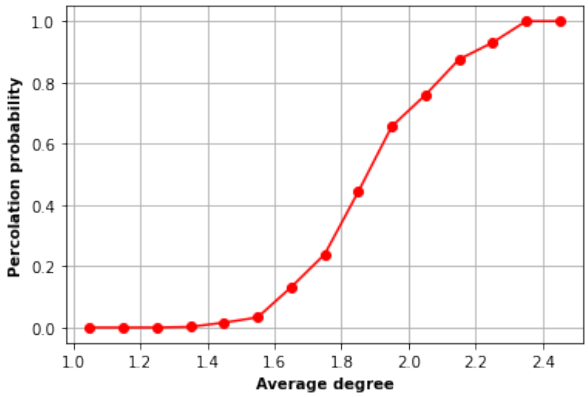


Figure 6.9: Percolation probability of the simulations combined.

The hydraulic connectivity is again calculated using two different methods; excluding and including the nodes from the fractures that are not part of the spanning cluster (fig. 6.10). It can be observed that taking all the nodes that are present in the simulated area gives unclear and chaotic results for the hydraulic connectivity. Excluding these fractures result in a clear linear trend for k>2.2.

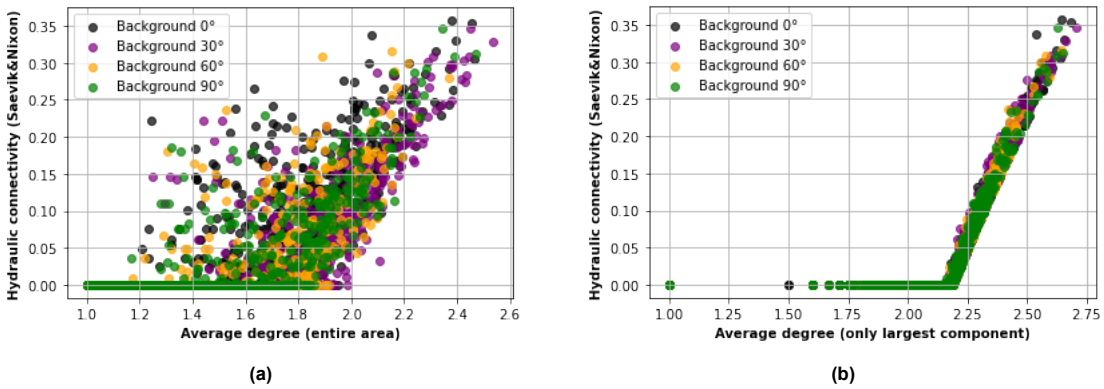


Figure 6.10: Hydraulic connectivity as defined by Sævik and Nixon (2017). a) In this calculation all the fractures in the network are used. b) In this calculation only the fractures in the largest component are used.

The relationship between LCR and average node degree is depicted in figure 6.11a for the case with 0° background fracturing.

The deviation of the data points around the s-curve are shown in figure 6.11b. It can be observed that the spread of the data is around the s-curve for k<1.6 is larger than for the previously considered fracture network. For k>1.6 the spread of the data around the s-curve increases. The largest deviation of the data points from the derived s-curve is 0.14 and is present at 2.0<k<2.2. For k>2.2, the deviation sharply decreases.

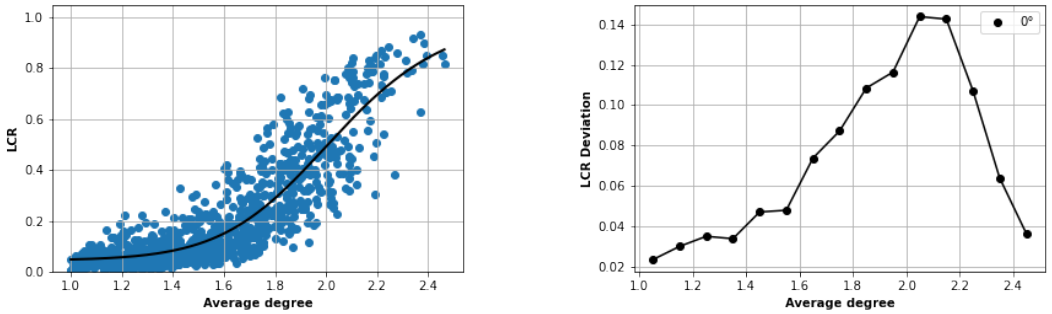
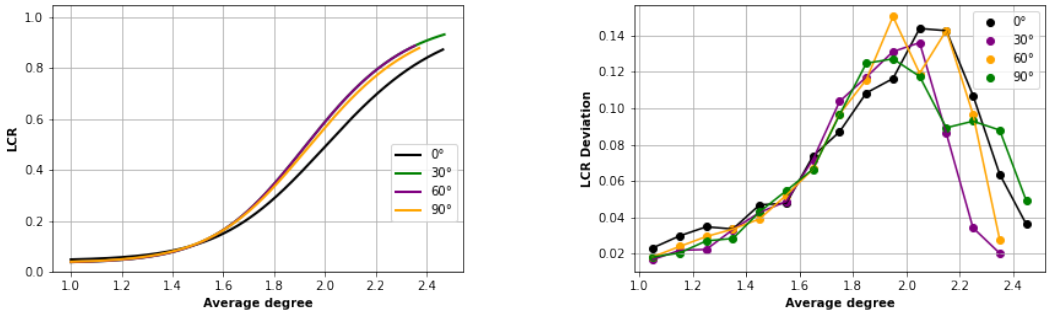


Figure 6.11: a) The relationship between the LCR connectivity measure and the average node degree (k). The data point all represent one simulated DFN. From the data points the s-curve relationship is found that starts to increase significantly for k>1.6 b) The deviation of the data points shown in (a) around the found s-curve. This illustrates the uncertainty in the connectivity at different average node degree. The uncertainty of the LCR-k relationship is highest for 2.0<k<2.2. It has a maximum uncertainty of 0.14 LCR.s

The relationship between LCR and the average node degree for the 4 considered background orientations are shown in figure 6.12a. For all simulations, the LCR average degree relationship shows a similar relationship. LCR is relatively low for k<1.6. For k>1.6 LCR starts to rise for all the considered cases. This highlights that generally speaking the percolation risk of these fracture networks considerably increases for k>1.6

The respective deviation of these plots are depicted in figure 6.12b. The deviation of the data points from the s-curve are all quite similar. The highest deviation, which have values in between 0.12 and 0.14, are present at 1.9<k<2.2. This shows that the uncertainty in the connectivity estimations are generally highest for 1.8<k<2.2.

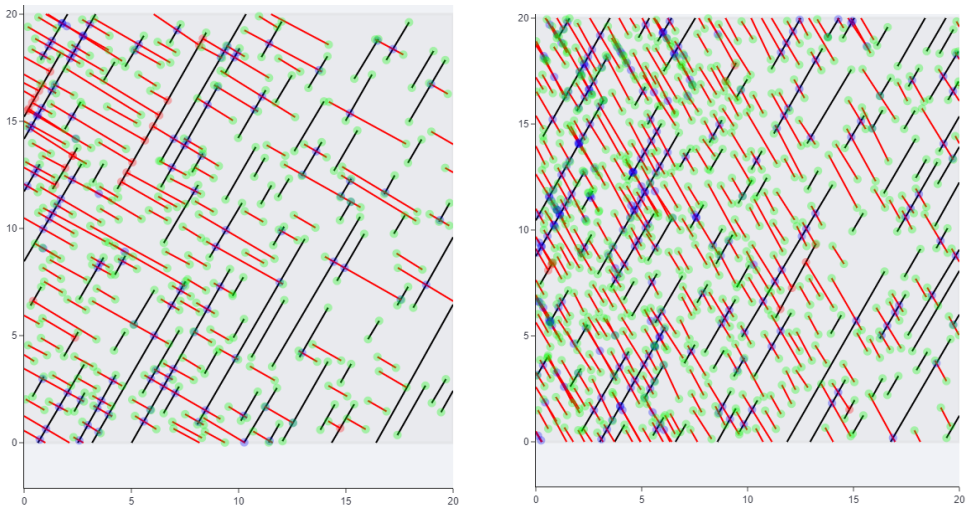


(a) (b)

Figure 6.12: a) The relationship between the LCR connectivity measure and the average node degree (k) for different background fracture orientations b) The deviations of the data points shown in (a) around the found s-curves. This illustrates the uncertainty of all simulations is highest for 1.9k<2.2. It has a maximum uncertainty of 0.14 LCR.

6.3. FDZ simulations with consistent geometric relationships.

Two FDZ's were tested where set 1 and set 2 fractures have consistent geometric relationship to each other. These geometric relationships are orthogonal (fig. 6.13a) and conjugate (fig. 6.13b). For both cases, 1000 realisations have been performed. Figure 6.14 shows the distribution of parameters obtained from these realisations. Logically, the angles between the two sets are clustered around 2 values; 90° for the orthogonal network and 60° (Simulation is run for 120° which results in fractures that are orientated 60° to each other) for the conjugate network. Most realisations have between 200 and 800 I-nodes. The number of Y-nodes and X-nodes range from 0 to 200 and 0 to 800 respectively. The mean fracture lengths are concentrated around a value of 2 and network P21 ranges from 0.5 to 2.5.



(a) (b)

Figure 6.13: 2 cases of FDZ's with constant geometric relationship. a) Orthogonal fracture network. b) Conjugate fracture network.

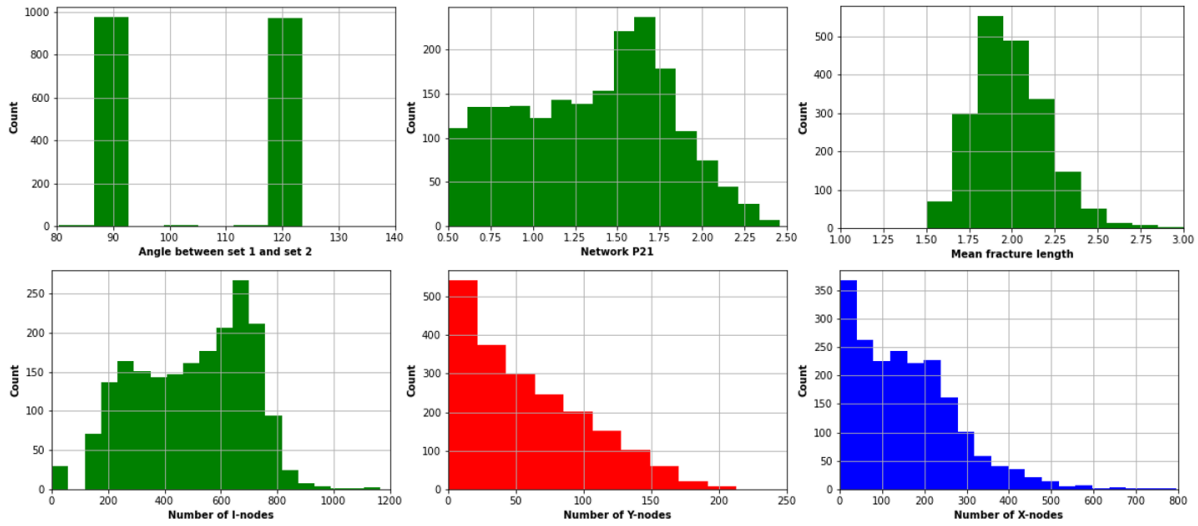


Figure 6.14: Distribution of parameters of the different fracture network. The angles between set 1 and set 2 are 90° and 120°(or 30°). The network P21 ranges from 0.5 to 2.5. The mean fracture length ranges from 1.5 to 5.5. The number of I-, Y- and X-nodes are also shown.

Figure 6.9 shows the percolation probability of the considered geometric consistent FDZ fracture networks.

The percolation probability is near zero for k<1.6. For k>1.7 the percolation probability increases significantly. For k>2.4 all simulations percolate.

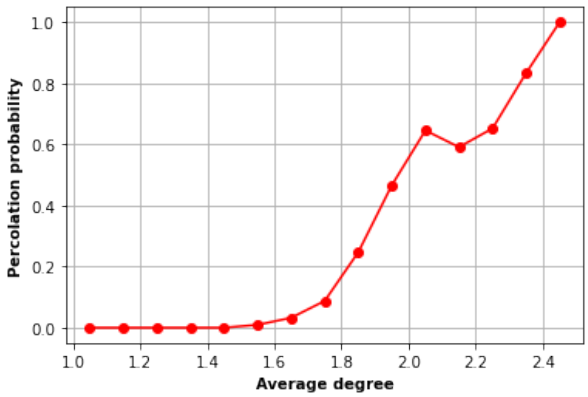


Figure 6.15: The percolation probability of both the orthogonal and conjugate simulations combined.

The hydraulic connectivity for the geometric consistent FDZ are depicted in figure 6.16). We again observe chaotic results if we take the node distribution of the entire area. When we just use the average node degree of the spanning cluster we again obtain the linear relationship. This linear relationship is observed for so well the orthogonal as the conjugate network.

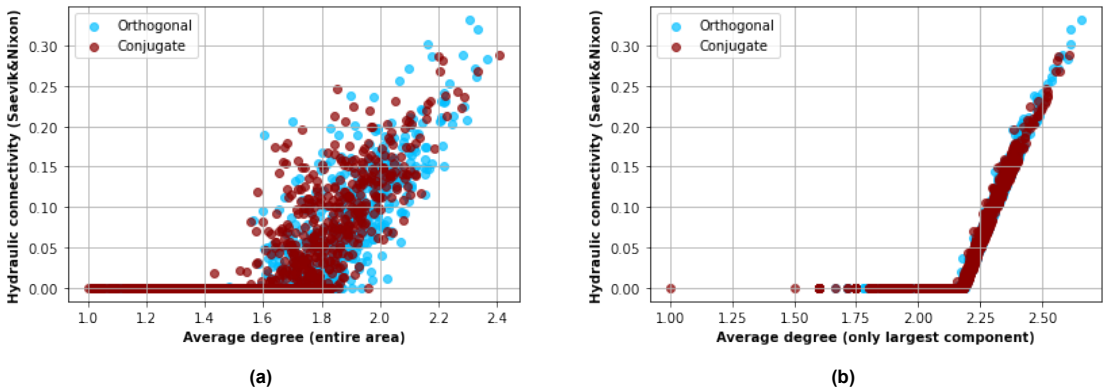


Figure 6.16: Hydraulic connectivity as defined by Sævik and Nixon (2017). a) In this calculation all the fractures in the network are used. b) In this calculation only the fractures in the largest component are used.

The relationship between the average node degree and LCR and its uncertainty for orthogonal networks and conjugate networks are depicted in figure 6.17 and figure 6.18 respectively. For both cases, the maximum LCR value is approximately 0.8. The uncertainty of the predicted relationship for both cases increases significantly for k>1.4 and has its maximum at approximately 2.2. A declining uncertainty is observed for conjugate networks for 1.9<k<2.1. However, this is due to the number of data points in this interval. Note that LCR shows a more inclined relationship with k compared to the results of the simple fracture networks, and the FDZ with constant background fracturing.

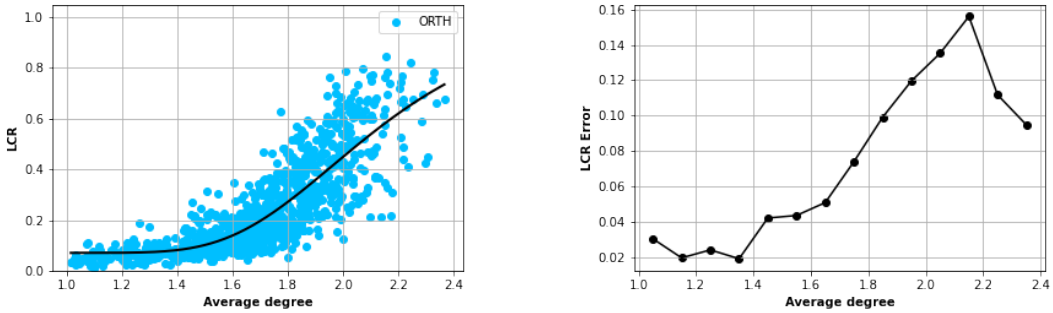


Figure 6.17: a) The relationship between the LCR connectivity measure and the average node degree (k). The data point all represent one simulated DFN. From the data points an s-curve relationship is found that starts to increase for k>1.6. Note that LCR shows an inclined relationship with k. b) The deviation of the data points shown in (a) around the found s-curve. This illustrates the uncertainty in the connectivity at different average node degree. The uncertainty of the LCR-k relationship is highest for 1.7k<2.0. It has a maximum uncertainty of 0.13 LCR.

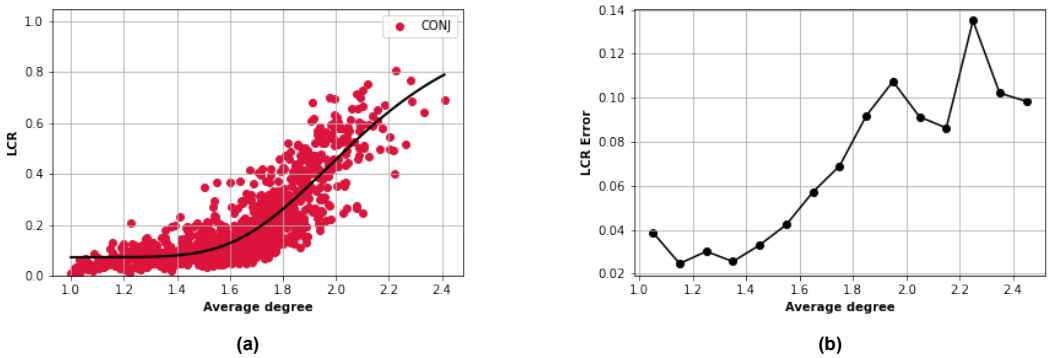


Figure 6.18: a) The relationship between the LCR connectivity measure and the average node degree (k). The data point all represent one simulated DFN. From the data points an s-curve relationship is found that starts to increase for k>1.6. Note that LCR shows an inclined relationship with k. b) The deviation of the data points shown in (a) around the found s-curve. This illustrates the uncertainty in the connectivity at different average node degree. The uncertainty of the LCR-k relationship is highest for 1.7k<2.0. It has a maximum uncertainty of 0.13 LCR.

6.3.1. Connectivity correlation

The correlation matrix of LCR with varying parameters is shown in figure 6.19. It clearly illustrates that the connectivity of the fracture networks tested in this DFN simulation is best captured by using their topology.

	Average degree (k)	X-nodes	Y-nodes	I-nodes	Average fracture length	Network P21	Angle between set 1 and set 2
Simple fracture networks	0.88	0.59	0.09	-0.77	0.76	0.71	0.29
FDZ with constant background fracturing	0.89	0.62	0.53	-0.06	0.42	0.37	0.52
Consistent geometric relationship	0.85	0.73	0.43	0.47	0.45	0.73	-0.07

Spearman's rank correlation coeff.

-1 0 1

Figure 6.19: Spearman's rank correlation between the LCR measure of connectivity and different parameters. It shows that topology is the best indicator if connectivity of fracture networks.

6.4. Analysis of outcrop fracture networks

The DFN simulations have shown how the LCR can approximate the connectivity of different fracture networks. It has also shown that the hydraulic connectivity only gives a simple linear trend and cannot be used to assess the uncertainty of the fracture network connectivity. Therefore, the LCR is used to analyse a range of fracture network outcrops using the robustness methodology. This can help to identify differences in the connectivity of different types of FDZ fracture networks. It can also help to quantify the uncertainties related to this connectivity.

6.4.1. Robustness analysis of fracture networks

Figure 6.20 shows iterations 0, 500 and 1000 of the robustness algorithm. It can be observed that the relative amount of I-nodes increases as the network transitions into a more disconnected state. The amount of X-nodes decreases more significantly. Y-nodes show a relative increase in iteration 0 to 500. From iteration 500 to 1000 the relative amount of Y-nodes decreases. Since this simulation is random-based, the orientation distribution of the fracture segment remains relatively constant.

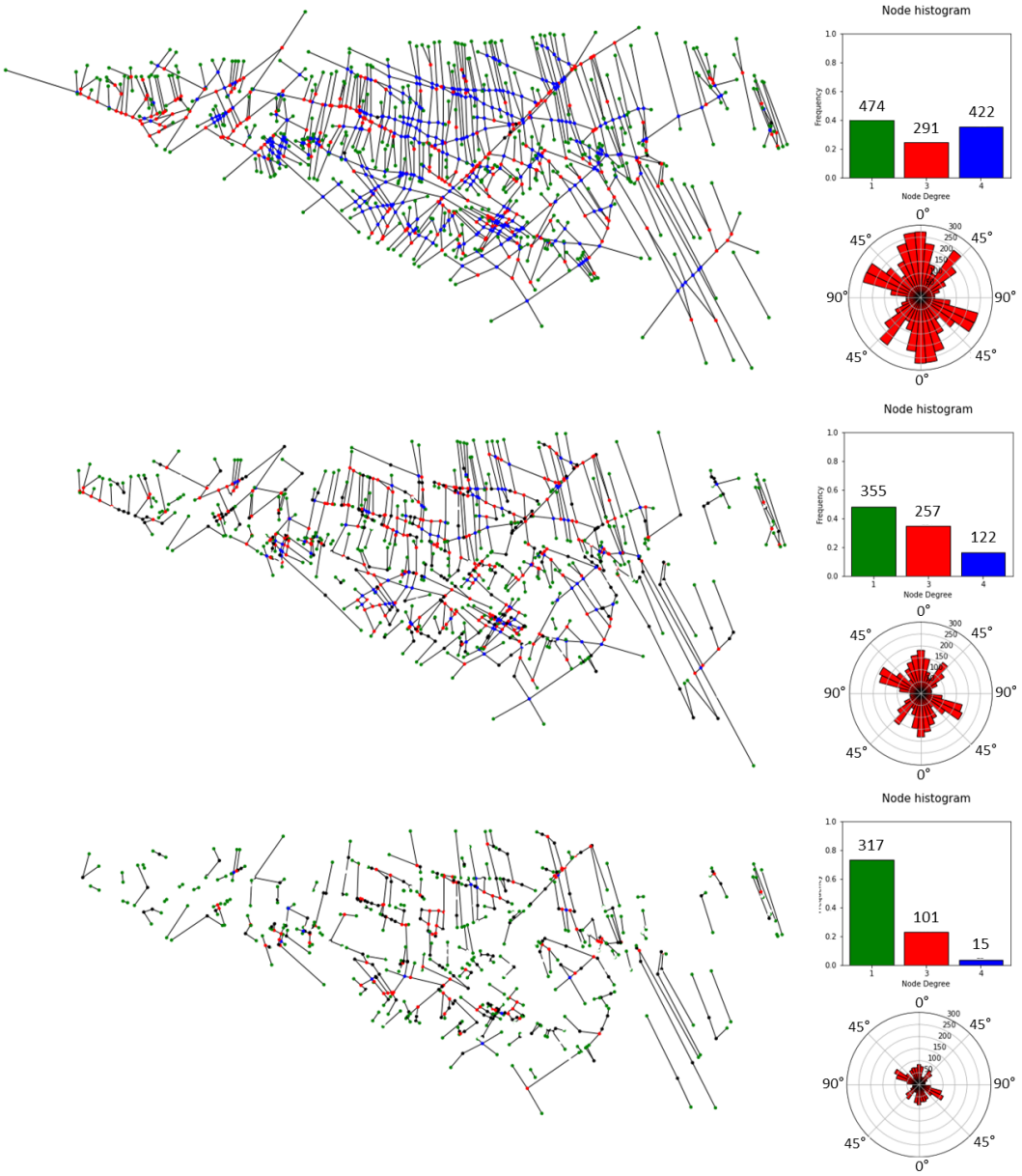


Figure 6.20: Iteration 0, 500 and 1000 of random-based reduction of Konusdalen I. A significant change in the node distribution is observed. The orientation distribution remains relatively constant. Interpretation (shown in iteration 0) modified from Dr. R. Rizzo

The process of segment removal enables us again to obtain a relationship between the LCR and the average node degree. Additionally, a relationship between P21 (fig. 6.21a) and LCR can be determined (fig. 6.21b).

Due to the random nature of fracture segment removal, every realisation of the same fracture network is unique. The curves for every realisation (light blue) show a step-wise increase for increasing k.

It is found that the relationship between the LCR and the average degree can be found using the s-curve relationship (eq. 6.1) as discussed in the DFN simulation chapter. The relationship of the LCR and P21, which could not be defined using DFN simulations, can also be found using the reduction methodology (eq. 6.2).

$$LCR(P_{21}) = \frac{1}{1 + e^{-c(P_{21} - P_{21m})^a}} \tag{6.2}$$

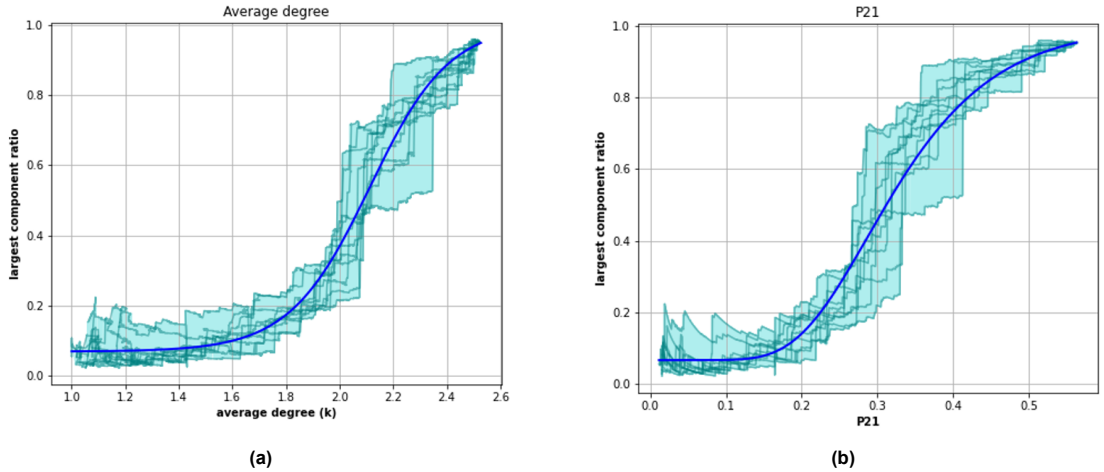


Figure 6.21: a) relationship between average degree and largest component ratio for random-based reduction. b) relationship between P21 and largest component ratio for random-based reduction

6.4.2. Studying various types of FDZ

In chapter 5, 9 different FDZ fracture network outcrops are introduced. These fracture networks were analysed on their connectivity with the use of the robustness algorithm. Analysing these networks helps us to understand the difference in the connectivity relationship with topology. Additionally, it helps us to understand the sensitivity of the robustness algorithm for capturing these differences.

The results for all the FDZ fracture networks are shown in figure 6.22a and b. It can be observed that there is a difference in the relationship between topology and connectivity for all these networks. The Moab Fm. wall damage shows the highest connectivity for 1.8<k<2.25. The cluster damage zone shows the lowest connectivity in this interval. The antithetic, lens, and somerset I fracture networks show an 'incomplete' relationship of LCR and average node degree. This is because the original fracture network that was used in this analysis had a low value of k and had a small LCR. Therefore, these fracture networks are not well suited to test their connectivity using this methodology.

The relationship between LCR and P21 shows more significant differences. The Moab Fm. wall damage shows a sharp LCR increase within a small range of P21. The Konusdalen I fracture network shows a far more gentle increase in connectivity with increasing P21.

Figure 6.23 shows that for all fracture networks the uncertainty of the relationship between the LCR and k is highest between an average degree of 1.9<k<2.2. The Somerset II has the lowest uncertainty (0.09 LCR) in its connectivity estimation and the Moab Fm. wall damage has the highest uncertainty (0.14 LCR).

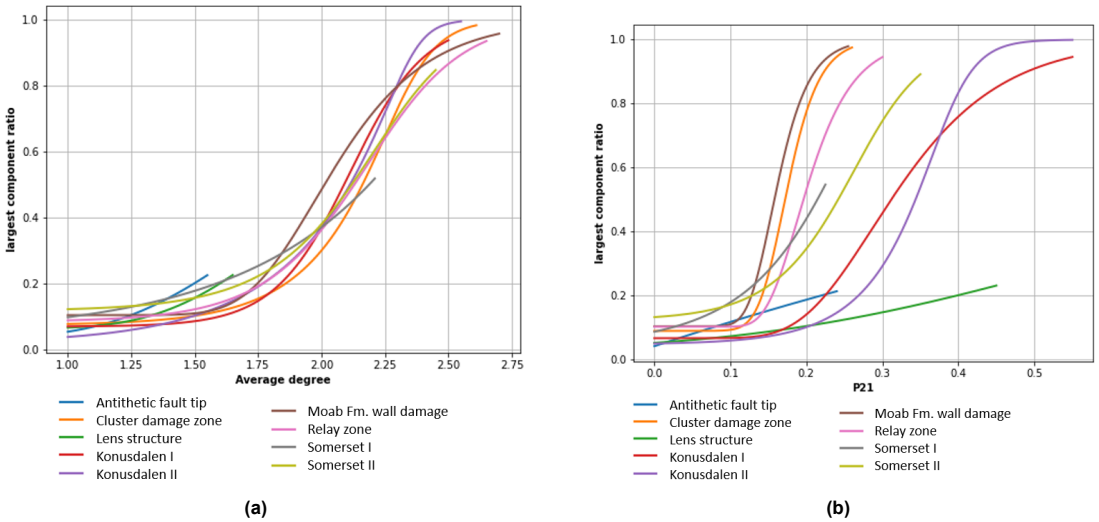


Figure 6.22: a) The LCR-k relationship for different fracture networks based on the random-based robustness algorithm. It shows that different fractures have different relationships between topology and connectivity. However, they all follow similar trends. They generally increase for k>1.75 and reach high connectivity at k=2.5. b) The LCR-P21 relationship shows significant differences between fracture networks. Highlighting that connectivity estimates based on P21 are not universal.

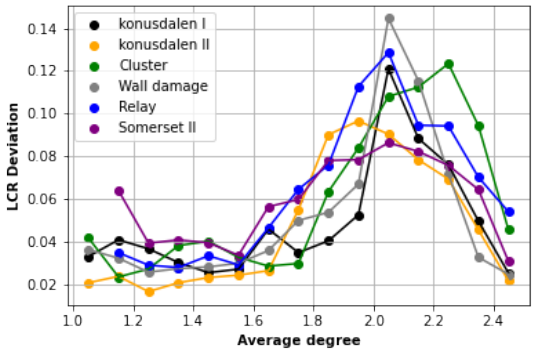


Figure 6.23: The deviations of the data points shown in 6.22a around the found s-curves. This illustrates the uncertainty of all simulations is highest for 2.0k<2.3. It has a maximum uncertainty of 0.14 LCR.

6.4.3. Orientation- and length-based reduction

All the robustness analyses performed until this point were random-based. Now the orientation-based and length-based reduction methodologies are discussed for the Konusdalen I outcrop. These analyses are performed to see how the connectivity of this fracture network responds to different selection processes. The orientation-based reduction shows how different fracture sets have different impacts on connectivity. The length-based reduction can highlight the importance of fracture length in fracture network connectivity.

Orientation-based reduction

Figure 6.24 shows the orientation-based robustness method. The original network shows roughly 2 sets of fracture segment orientations which an average angle of 5° (set 1) and 40° (set 2) to the vertical. The fracture segments of set 1 are removed first by selecting all the fractures that have an angle that deviates with a maximum of 20° to the average angle of set 1. The fracture segments that have to be removed from these sets are selected in random order. When all the fracture segments of set 1 are removed, the fracture segments of set 2 are removed. These are all the remaining fractures.

It can be seen from iterations 0 to 500 that the absolute number of I-, Y- and X-nodes decreases. However, it can be observed the number of X-nodes reduces more considerably than the number of

Y-nodes. From iteration 500 to 1000 the absolute amount of I-nodes increases, as the fracture network becomes more disconnected.

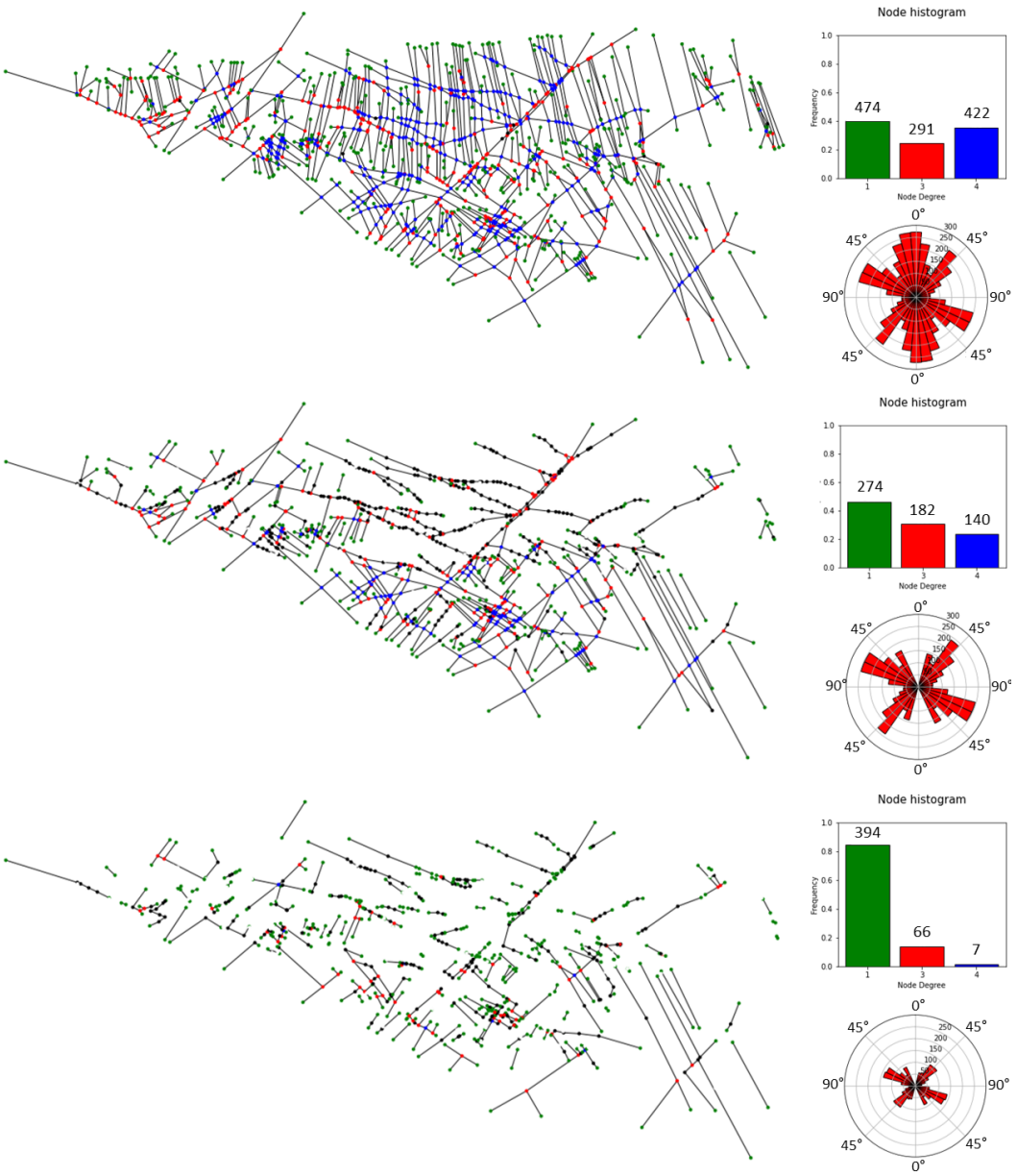


Figure 6.24: Iteration 0, 500 and 1000 of orientation-based reduction (335-015). from the orientation distribution, it can be seen that one orientation is removed first. The node distribution changes less dramatically then orientation-based

Figure 6.25 and **??** show the result of the orientation-based robustness method. In figure 6.25 the fractures of set 2 (sub-vertical) are removed from the fracture network. This removal process occurs at 2.35<k<2.5. It can be observed that all reduction realisation cluster together at k=2.35. This is logical because at this point all the realisations have the same fracture network since all the same fracture segments are removed. After this point set 1 fractures are removed. For k<2.35 we have different relationships between the topology and the connectivity.

Figure **??** shows the effect of first removing the sub-horizontal fractures (set 1) and thereafter the fractures of set 2 (sub-vertical). It clearly shows a different trend than the previous result. Note the more significant drops in connectivity within small ranges of k.

The impact on the LCR-P21 relationship is more significant. During the removal processes of set 2 fractures, the P21 decreases significantly. However, almost no decrease in LCR is observed. This

means that most of the fracture segments of set 2 are not the important connecting fractures that hold the largest component together.

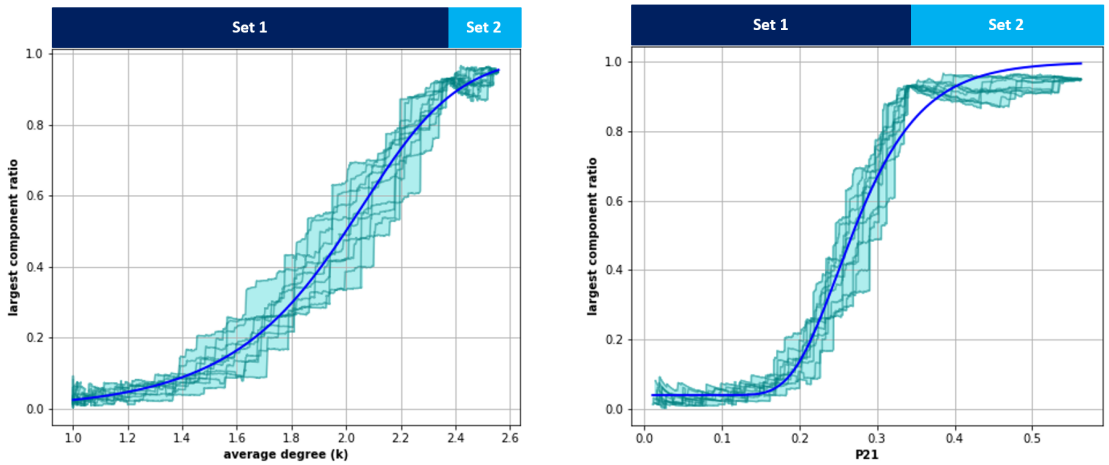


Figure 6.25: a) relationship between the average degree and largest component ratio for the vertical orientation-based reduction. b) relationship between P21 and largest component ratio for orientation-based reduction

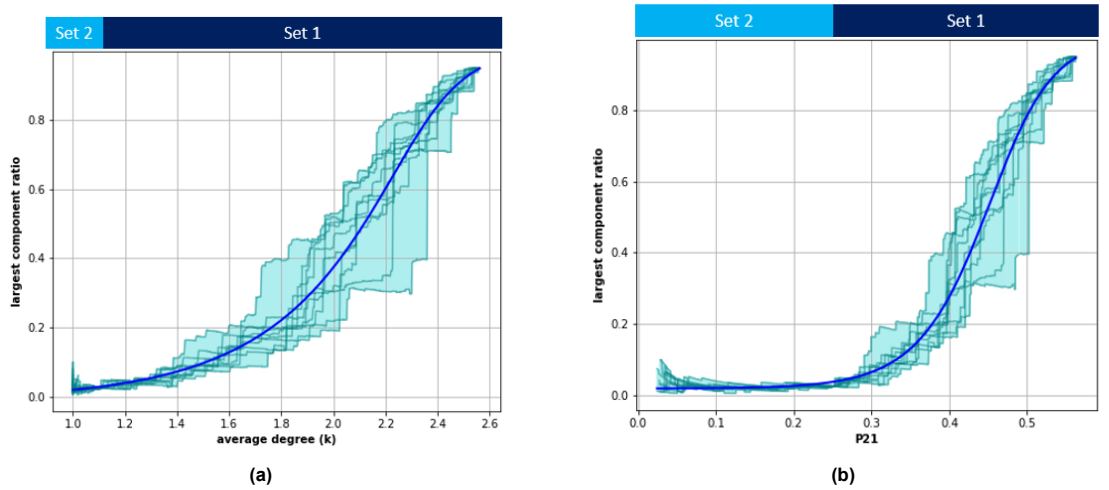


Figure 6.26: a) relationship between the average degree and largest component ratio for the horizontal orientation-based reduction. b) relationship between P21 and largest component ratio for orientation-based reduction

length-based reduction

The Konusdalen I outcrop is considered once more and is used for the length-based reduction analysis. The fracture network throughout one realisation is shown in figure 6.27. It can be seen that angle distribution removes fracture segments from all orientations. However, it can be observed from iterations 0 to 500 that more fracture segments with a sub-vertical direction are removed. This indicates that the fracture segments in this orientation are generally longer than in the sub-horizontal direction. From iteration 0 to 500 it can also be observed that the number of I-, Y- and X-nodes decrease. From iteration 500 to 1000 the number of I-nodes increases and the number of Y- and X-nodes decrease.

The results of the length-based reduction are shown in figure 6.28. It can be seen that LCR is low for 1<k<2. For k>2 the LCR start to increase drastically until it reaches an LCR of 0.9 at k=2.55. Figure 6.28b shows the relationship of LCR with P21. for P21>0 the LCR immediately start to rise with a gentle slope until it reaches a LCR of 0.9 at a P21=0.55.

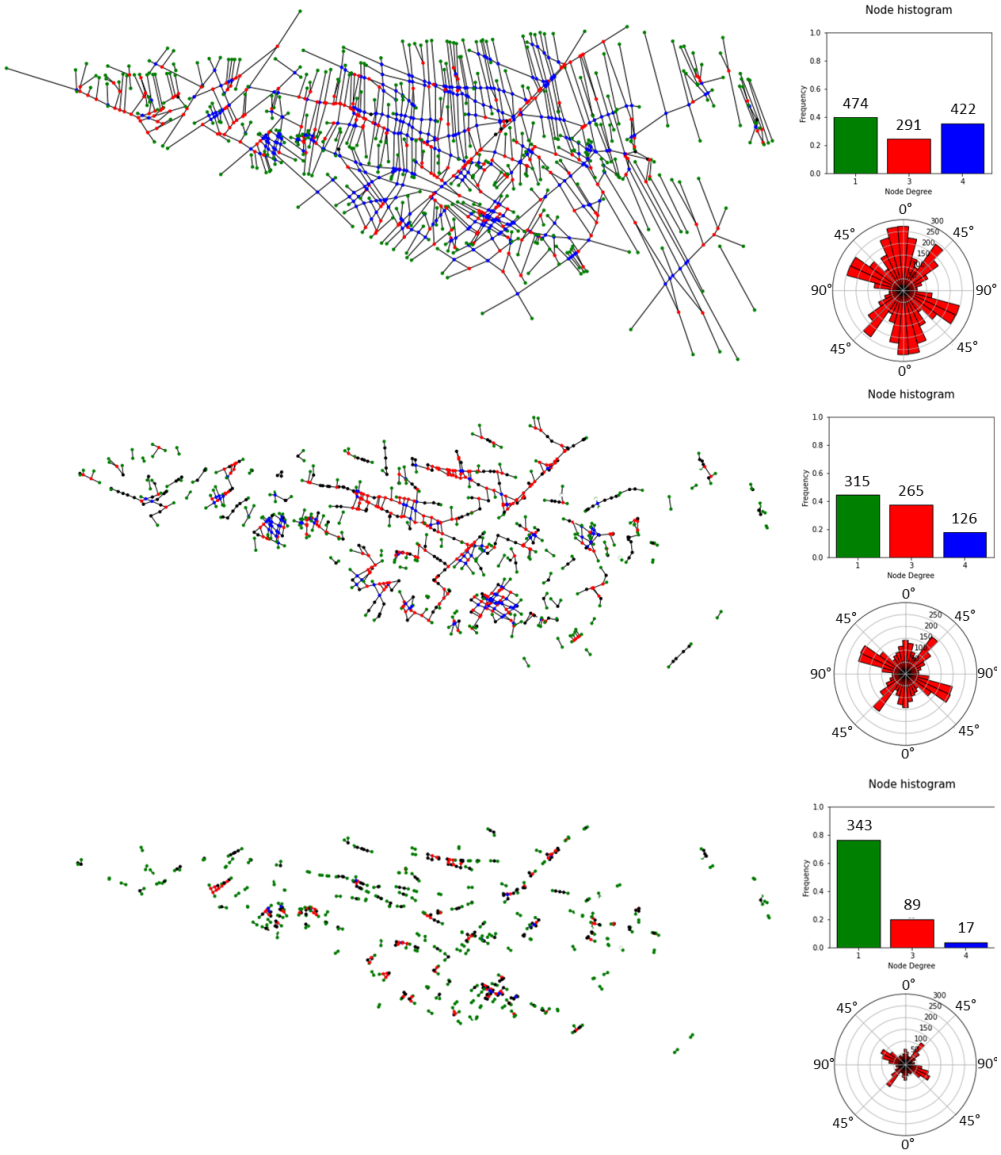
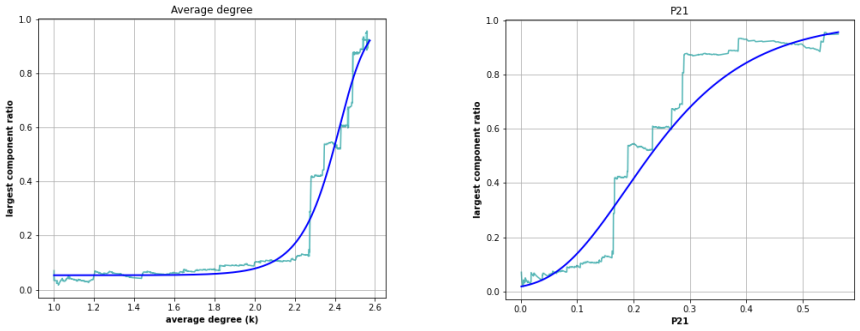


Figure 6.27: Iteration 0, 500 and 1000 of length-based reduction.



(a) **(b)**

Figure 6.28: a) relationship between average degree and largest component ratio for length-based reduction. b) relationship between P21 and largest component ratio for length-based reduction.

6.4.4. Impact of synthetic connecting fractures

To test the impact of important connecting fractures on the fracture network robustness the Somerset II outcrop is considered (Fig. 5.11). This outcrop is considered for this analysis because it clearly shows a region in the network where connectivity is low (the silt-rich layer). This makes it a good outcrop to study what happens to the LCR connectivity when the network is more connected.

9 realisations were completed, each with a different amount of connecting fractures through the silt-rich layer that divides the fracture network into two parts. These additional fractures would lead to higher connectivity without impacting the node distribution significantly. Figure 6.29 shows the result of the analysis. The original network is shown in black. The numbers 2 to 8 correspond to the number of fractures added through the silt-rich layer. When 2 fractures are added a slight shift of the average degree and P21 curve is observed. At k = 2.08 the LRC curve drops below the s-curve of the original fracture network. When 4 or more synthetic fractures are added the curves start to shift more significantly. These realisations intersect the curve of the original network between at k=1.8 and k=1.9, and at a P21=0.16. All curves flatten out at k<1.8 and a for P21<0.16.

The realisation where 8 connecting fractures are added shows the most significant shift. The shape of curve 8 is also different from the original. A steeper rise is observed for k>1.7 (and P21>0.15). The steepness of the curve reduces at k<2.2 and at P21<0.26.

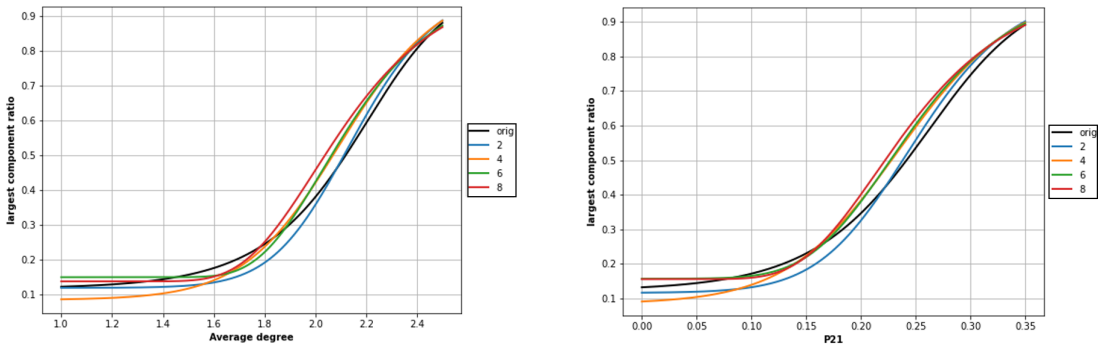


Figure 6.29: a) The LCR-k relationship for the Somerset II outcrop. The number of fractures that were added to the fracture network through the dividing silt-rich layer are highlighted. It can be seen that, as more fractures are added, the curve shifts to the left. This indicates the different relationship of toplogy and connectivity based on the original network. This analysis can help to discuss the difference between different fracture outcrops. b) The LCR-P21 relationship shows a similar shift as observed in the LCR-k relationship.

7 Discussion

Assessing the connectivity FDZ fracture network is a challenging task as FDZ are diverse and there is often little information available of subsurface fracture networks. This research, therefore, aims to find a methodology to approximate the connectivity of fault-related fracture zones when only sparse subsurface data is available. Furthermore, it aims to identify and quantify the uncertainties related to this prediction.

An approach based on topology and graph theory was proposed to tackle these questions. First, a DFN analysis was performed to study the ability of the Sævik and Nixon (2017) hydraulic connectivity to highlight differences in fracture network connectivity. Hereafter, the connectivity concept of LCR, which is formulated from graph theory, was tested as a measure of connectivity using the DFN generator. This measure is directly related to the topology of the network by using the average node degree.

After the DFN simulations, an outcrop analysis was performed based on the graph theory concept of robustness. In this analysis, random-based, orientation-based and length-based reduction of FDZ fracture networks was performed. From these analyses, the relationship between the topology and LCR was studied and its uncertainty was quantified.

In this chapter, a discussion on the results of the DFN generator is provided first. This includes an evaluation of the measures of connectivity and their uncertainties. Hereafter, the results of the outcrop robustness analysis are discussed.

7.1. FDZ DFN generator discussion

In this section, three different topics will be discussed. First, the relationship between the hydraulic connectivity, LCR and the topology of the fracture network is debated. Hereafter, the uncertainty in the relationship between topology and LCR relationship is discussed.

7.1.1. Connectivity and percolation

Two different measures of connectivity were used to study fracture networks.

Figure 7.1 shows how the LCR connectivity and hydraulic connectivity compare to the percolation risk of a fracture network. The hydraulic connectivity was determined using two methods; by excluding or including the fractures that are not part of the large cluster.

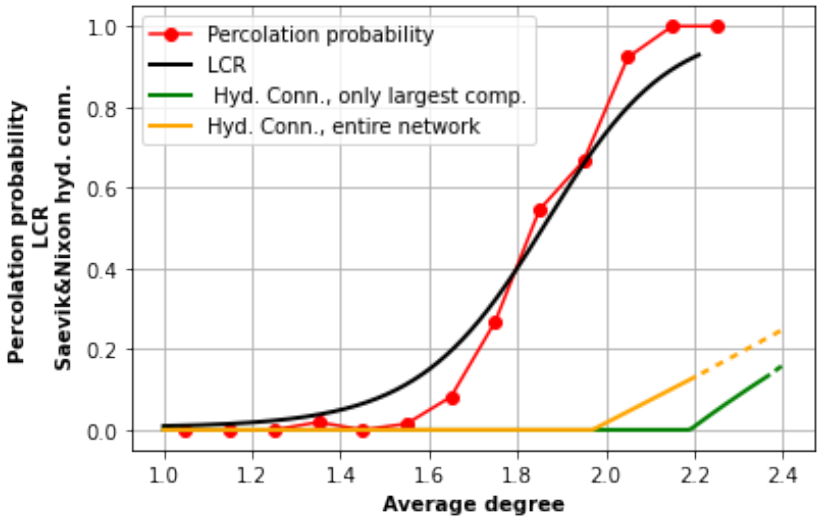


Figure 7.1: Comparison of the LCR measure of connectivity and the percolation probability

The figure shows that the percolation probability is 0 for k<1.35. This is logical as an average node degree of 1.35 or less means that the network is dominated by I-nodes and therefore the majority of the fractures are isolated.

At a k=1.35 a small number of realisations are percolating (2 out of the 105 realisations). This highlights the fact that there is a chance percolation is achieved at these low average node degrees. When k=1.35 the corresponding LCR is 0.05. The LCR at k=1.35 is 0.05 which indicates that a largest component is present, however, it generally only makes up a small amount of the total network. Therefore, as indicated by this measure of, there is low connectivity, and thus a small change of percolation. This is logical for these low values in average node degree. However, there are possibilities at these low values of k that a large component exists that is far bigger than expected.

Figure 7.2 shows two of these cases; inhomogeneous topology (7.2a) and inconsistent fracture length (7.2b). The inhomogeneous topology case is present when there are fractures that have similar lengths and orientations, but there is a significant difference in topology in one area to the rest of the network. This section that has far different topology than the rest of the network and creates a large component when there is only an average node degree of 1.4 in the fracture network. This also highlights why it is essential to have an understanding of the spatial distribution of connectivity of a specific fracture network before making a connectivity assessment based on topology.

Inconsistent fracture size is another scenario where a spanning component can exist at low values of k. The topology over the entire area is homogeneous. However, the two fractures that make up the largest component are significantly larger in size than the other fractures.

These two examples highlight the fact that, although the average node degree can be a reliable indicator of connectivity, it is important to incorporate fracture length variation and topological homogeneity assessment in the connectivity assessment.

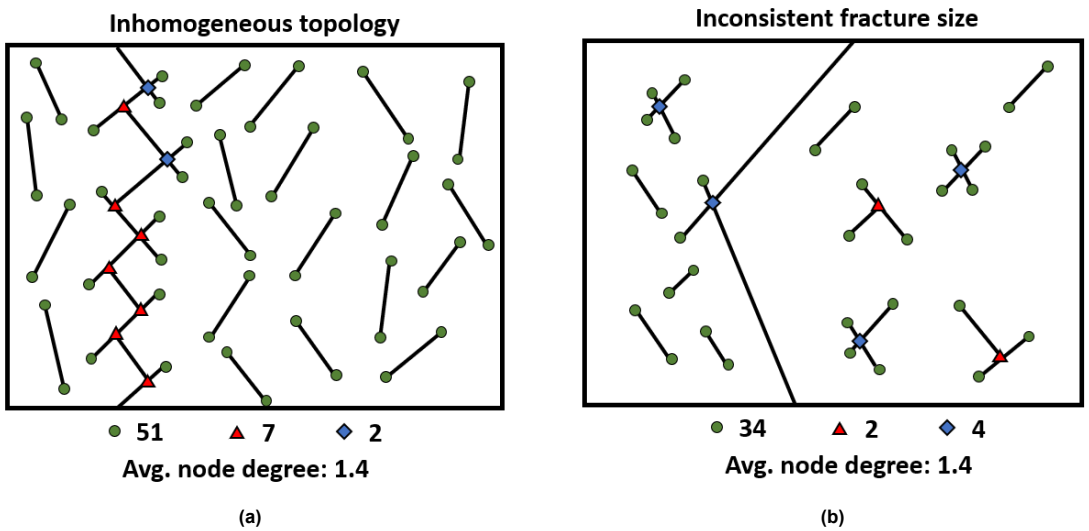


Figure 7.2: a) The case of inhomogeneous topology which may cause unexpected percolation at low values of average node degree. b) The case of incosisten fracture size which may cause unexpected percolation at low values of average node degree.

For k>1.6 the number of simulations that are percolating starts to rise significantly. This increase, which follows an s-curve, reaches 1 at an average node degree of about 2.1. LCR also start to increase more significantly for k>1.6. This is logical as a large spanning component needs to be present in order for the fracture network to percolate. At a k value of 1.6 the LCR is however already at a value of 0.18 whereas the percolation probability is still close to zero. This shows that the largest component is present, but is not orientated in such a way that percolation is reached. The LCR increases to a value of 0.93 at k = 2.2. Contrary to the percolation probability it never reaches the value of 1. This is logical, as there are often parts of the fracture network that are not connected to the largest component.

Both LCR and the percolation probability rise significantly within a range of only 0.5 average node degree. This highlights the fact that only a small variation in the node distribution of the fracture network can have a significant impact on fracture network connectivity.

The Sævik and Nixon (2017) methodology for hydraulic connectivity, which was calculated with and without the fractures that are not part of the spanning cluster, predicts a percolation threshold much higher. It predicts percolation to occur when the average node degree is as high as 2.0. At this average node degree, the percolation probability of approximately 1 is expected.

When only the nodes in the spanning cluster are used to calculate the hydraulic connectivity, the relationship shows a near linear relationship with k for k>2.0 (or k>2.2). This makes highlighting differences in connectivity difficult as it does not take the whole network into consideration when analysing the connectivity. It is therefore not well suited to perform a topology based analyse of the connectivity of different fracture networks.

Two other scenarios were tested with the DFN generator. The scenario with constant background fracturing is shown in figure 7.3a. The percolation probability starts to rise for k>1.6, which is comparable to the distributed fracture networks. It does, however, rise less steeply for 1.6<k<2.1 and only reaches a probability of 1 at k>2.3. This more gradual increase is also observed in LCR. This shows that LCR is able to follow the connectivity behaviour to a certain amount.

Figure 7.3b shows the percolation probability and the LCR relationship with the average node degree of the orthogonal and conjugate fracture network test case. The percolation probability starts to rise at k>1.7. These simulations show a more gentle increase and reach a percolation probability of 1 at an average node degree of 2.4. This gentler increase is also shown by LCR relationship with the average node degree.

This highlights, that LCR can be used to approximate the connectivity of the network based only on topological data. This makes the LCR a better measure of connectivity than hydraulic connectivity for studying different fracture networks based on topological connectivity. LCR is also preferred over other methods of connectivity because it is easy to determine. Therefore, it can be a quick and

computationally effective way of highlighting connectivity differences of fracture networks.

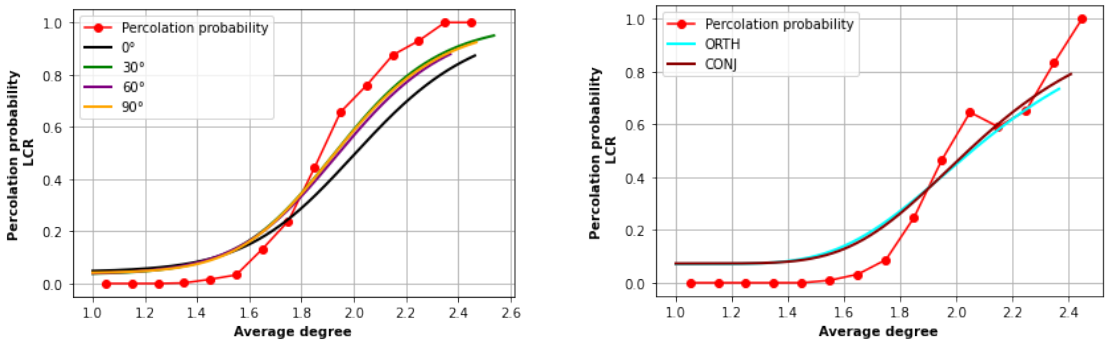


Figure 7.3: Comparison of the LCR measure of connectivity and the percolation probability. Not that both the percolation probability and LCR have a more gentle slope in the relationship shown in (a) than shown in (b).

7.1.2. Uncertainty in the connectivity

The LCR-topology relationship was determined by fitting an s-curve relationship using a least squares method. Figure 6.6a shows the spread in the data around this curve. When the data are binned based on their k-value a plot can be made that quantifies the uncertainty of the prediction at different values of k. For all DFN simulations, there is relatively little uncertainty for 1.0<k<1.6. For these values of k the fracture network is mostly dominated by I-nodes. Therefore, there it is nearly impossible for a large network to exist.

When k>1.6 the uncertainty in LCR starts to rise. The highest uncertainty appears to consistently to be at 1.9<k<2.1. The uncertainty at these peaks ranges from 0.12 to 0.16. At k = 2.2 the uncertainty decreases rapidly. This is because at these high average node degrees there are many connections. These many connections almost certainly lead to the formation of a large component.

7.2. Natural fracture network discussion

A new methodology was brought forward to study the relationship between topology and the connectivity of the network; the robustness algorithm. With this methodology, real FDZ fracture networks can be analysed on their topology-based connectivity. The random-based robustness of Konusdalen I is discussed first. Hereafter, it is evaluated whether different FDZ fracture networks show different LCR-k relationships. Lastly, the effect of the orientation-based and length-based reduction methodologies are discussed.

7.2.1. LCR analysis of natural fracture networks

The Konusdalen I FDZ outcrop is analysed using the robustness algorithm. Throughout the random-based reduction process, the number of X-nodes decreases more rapidly than the number of I- and Y-nodes. This is logical as X-nodes contain 4 segments and therefore have a higher chance of being dismantled. Y-nodes show a relative increase in iteration 0 to 500. This is due to the dismantling process of X-nodes which are converted into Y-nodes if one fracture segment is removed. From iteration 500 to 1000 the relative amount of Y-nodes decreases. This is because less X-nodes are converted into Y-nodes as most X-nodes were already removed. During the whole process, the amount of I-nodes logically increases as the network becomes more disconnected. As the process is random-based the orientation distribution remains relatively constant. No fracture segments with a particular orientation are being removed preferentially.

The LCR-k relationship shows a step-wise decrease for decreasing k for every realisation (fig. 6.21). These steps are caused when important connecting fracture segments are removed. These connecting fracture segments hold the largest component together. So when they are removed the largest component reduces in size depending on the importance of the fracture segment. This highlights that certain fracture connections, and thus nodes, are more important for the connectivity of the fracture network than others. The relationship between LCR and average node degree plateaus when less important

small segments are removed. These small less important segments will not decrease the size of the largest component, nor will they have a significant impact on the LCR. They do, however, reduce the average node degree.

In some instances, it is also observed that the LCR increases in size with decreasing values of average node degree. This occurs when fracture segments are removed that are not connected to the largest component. This potentially reduces the average node degree, if they contain X- and Y-nodes, but also increases the LCR.

The curves for every realisation show important characteristics of these fracture networks. Fracture networks that are dependent on a few important connecting fracture segments will show large drops in LCR when they are removed. Fracture networks that are less dependent on few fracture segments (and therefore better interconnected), will show smaller drops. These drops would also generally occur later in the reduction process. This is because a more interconnected fracture network would first need to be dismantled to a certain point. When the number of connections is decreased, fewer fracture segments keep the largest component together. LCR connectivity drops significantly when these important connecting fracture segments are removed. When fracture networks are more disconnected in the first place, then the important connecting fracture segments can be removed immediately.

The P21 is also recorded during the reduction processes. As every fracture segment has a certain length the P21 is decreased as it is removed. This relationship between P21 and LCR shows many similarities with the relationship between average node degree and LCR as it also shows drops in connectivity when important fractures are removed.

As the reduction process is random-based every realisation is unique. By performing the reduction processes multiple times a relationship between the average node degree (or P21) and LCR is found. The accuracy of the prediction would be increased with the number of realisations.

7.2.2. Differences between various FDZ fracture networks

The robustness algorithm allows us to directly study different FDZ types for their connectivity. Figure 6.22 shows that there is some variation in the relationship between the LCR and the average node degree of the different FDZ fracture networks. Generally, LCR starts to increase significantly from k>1.7. This is very similar to the increase in LCR we observe from the DFN simulations that generally show an increase in LCR for k>1.6. However, clear differences can be observed for 1.7<k<2.5 between different fracture networks. For example, the curve from the Moab Fm. wall damage zone increases far more steeply than the relay fracture network. This highlights the fact that at approximately the same topology, different fracture networks can have different connectivity.

Part of this difference is due to the random nature of fracture segment removal. However, more important is the difference in connectivity of the fracture network. This is illustrated by section 6.4.4 where synthetic fractures were added at important connection points for the somerset fracture network outcrop. These synthetic fractures were added through a layer that separates the two sections of the fracture network. Since only a few connection points were added the initial average node degree of the fracture network remains similar. However, because they increase the connection between different regions in the network it does have an effect on the relationship of average node degree and LCR. The impact of the first two synthetic connecting fractures is rather small. This is because these two segments only connect a small section of the upper section to the lower section. However, when more important synthetic connecting fractures are added the curve shifts to the left. This means that the fracture network is more interconnected at lower average node degrees. At an average node degree of approximately 2.0 there is a difference of 0.1 LCR between the original fracture network and the fracture network that had 8 connection synthetic fractures added to the network. This illustrates how the robustness algorithm can show the connectivity difference of fracture networks. Additionally, it highlights that the relationship between k and connectivity is case-dependent.

7.2.3. Uncertainty in robustness analysis

Like the results of the DFN simulations, an uncertainty analysis can be performed with the robustness algorithm. This is done by finding the difference between the results of the robustness realisations and the curve that is fitted using the least squares method.

Similar to the results from the DFN simulations the deviation of LCR for k<1.6 are low. For k>1.6

the uncertainty in LCR predictions starts to increase. The uncertainty is highest for 2.0<k<2.2 and has a peak between 0.08 and 0.14. This result is again similar to the results of the DFN realisations where the maximum uncertainty observed is between 0.12 and 0.16.

These uncertainties illustrate that the connectivity of fracture networks with equal topology can be significantly different.

7.2.4. Orientation- and length-based reduction

The length- and orientation-based reduction of Konusdalen I show significant differences compared to the random-based reduction (fig. 7.4).

The relationship between LCR and the average node degree obtained from the length-based reduction shows a very late increase in connectivity. At k>2.2 the curve rises sharply and reaches its maximum at an average node degree of 2.6. This is logical as the largest component often contains longer fractures. So if these are removed at the start of the reduction process, then the LCR drops fast.

The different relationships obtained from the orientation-based reduction method are shown in figure 7.4. The figure compares the results of the reduction where first the sub-horizontally orientated are removed, and the realisation where first the sub-vertical are removed. It shows that the connectivity of the vertical reduction rises significantly faster for 1.2<k<1.9. This is also the case for horizontally reduced results compared to the random-based removal which starts only starts increasing for k>1.7.

This analysis shows that the connections made in the sub-horizontal direction are more important for the connectivity than the fractures in the sub-vertical direction. This is shown in the individual realisations in figure 6.26. It can be seen that the connectivity drops more significantly when these are removed compared to the fracture in the horizontal direction.

This orientation-based method has the potential of quantifying the importance of different fracture sets on the connectivity of the network. Furthermore, it allows for the reconstruction of the formation processes. If the development history of a fracture network is known, it can be analysed by removing the sets first that were formed the last. This allows for a quick and computationally effective way of analysing the connectivity throughout the formation processes. This can highlight the importance of different types of nodes at various stages of the formation processes.

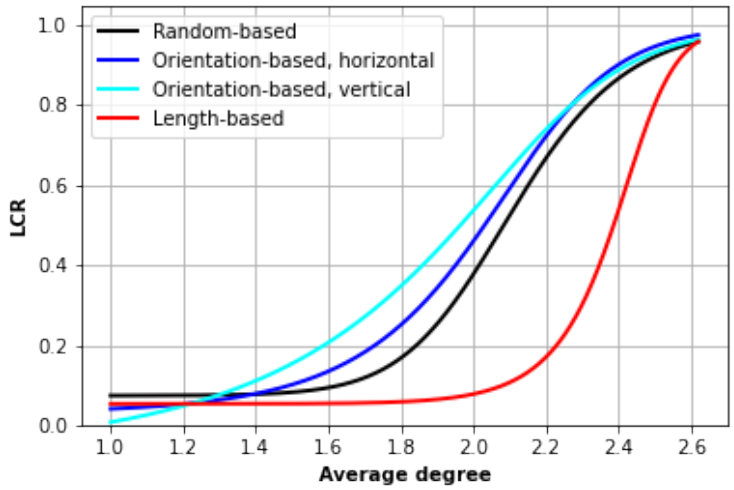


Figure 7.4: LCR results of different reduction processes of the Konusdalen I fracture network.

8

Conclusion

This thesis aims to capture the connectivity of fracture networks in a statistically robust way. We focused on finding a simplified model based on the topology to make a forecast on the connectivity of specific fracture networks when only sparse data of the subsurface is available. It is shown how topology is related to percolation in graph theory by the concept of the giant component. The link between these concepts and connectivity in fracture networks (and percolation) is made by using the largest component ratio which uses fracture segment length as edge weights. To research the connectivity of fracture networks based on topology a DFN generator and a methodology based on robustness were used.

The DFN simulations showed that the percolation threshold determined by Sævik and Nixon (2017) is not reliable. It predicts that percolation occurs at k>2.0 whereas actual percolation can occur far earlier (typically, k>1.6). It also showed that just using a linear trend between topology and connectivity is not sufficient as it is not capable of highlighting the difference in connectivity of different fracture networks. An additional link is required on how these topological measures related to the connectivity of a specific fracture network as a whole.

From the DFN simulations, it has also become clear that the connectivity measure of the largest component ratio is an approximation of the connectivity (and percolation potential) of a fracture network. From the DFN analysis, it can be concluded that the connectivity of a fracture network, and its related percolation risk, start to increase for k>1.6. The highest uncertainty of fracture network connectivity is typically present at 1.9<k<2.2.

Using graph theory to estimate the connectivity of fracture networks allows us to use the concept of robustness. Robustness can be used to study natural fracture networks and can highlight differences in their connectivity. From this analysis, it has become clear that topology can indicate the connectivity of the network. However, this connection between the topology and the network's connectivity is not universal for all fracture networks. This is partially shown by the different fracture networks that were analysed using the robustness algorithm which showed different relationships of topology with connectivity (especially for 1.75<k<2.3). It was also shown by the orientation-based robustness algorithm. This clearly indicates that a connection can be made between topology and the connectivity of the network, however, it is also very much dependent on the formation processes of the fracture network. Additionally, it showed that the relationship between connectivity and P21 is significantly different for various types of fracture networks.

This research can be improved by analysing outcrops that are more specifically related to certain types of fault movement. For this analysis, it is advised to dive deeper into orientation-based reduction as it can relate the topology to the connectivity throughout the network development. This does, however, highlight the need for more line drawings of the FDZ fracture network of which the development processes are (approximately) known. This research can also be improved by studying the relationship between the largest component ratio and robustness to other measures of connectivity such as the betweenness centrality (Appendix B).

Acknowledgements

Firstly, I would like to thank everyone that participated and contributed to this thesis. I want to give special thanks to Dr. Kevin Bisdom (Shell Global Solutions International) and Prof. Dr. Giovanni Bertotti (Delft University of Technology) for the fruitful discussions and their guidance during the project. I greatly appreciate the freedom that was given to explore different fields of study including graph theory.

Secondly, I want to thank Dr. Roberto Rizzo (University of Florence) for contributing to the project by providing a large amount of high-quality data. I also want to thank Dr. Jeroen Snippe (Shell Global Solutions International) for the discussions on the DETECT report and Dr. Rahul Prabhakaran (Friedrich-Alexander-University of Erlangen-Nürnberg) for the discussion on graph theory.

Thirdly, I want to thank my professors and fellow students that supported me during my 2 years in Delft. I especially want to thank my close friends who helped me tremendously during these last 2 years.

Lastly, I wish to thank my parents. I am very grateful for the full support they have given me throughout my entire school career.

References

Allan, U. S. (1989). Model for hydrocarbon migration and entrapment within faulted structures. $The American Association of Petroleum Geologists Bulletin V, 73 : 803 - -811.$

Anderson, E. (1951). The dynamics of faulting and dyke formation with applications to britain.

Andresen, C. A., Hansen, A., Goc, R. L., Davy, P., and Hope, S. M. (2013). Topology of fracture networks. *Frontiers in Physics*, 1.

Atkinson, B. K. (1984). Subcritical crack growth in geological materials. volume 89, pages 4077–4114.

Bai, T., Maerten, L., Gross, M. R., and Aydin, A. (2002). Orthogonal cross joints: do they imply a regional stress rotation?

Bai, T., Pollard, D., and H, G. (2000). Explanation for fracture spacing in layered materials. *Nature*.

Bai, T. and Pollard, D. D. (2000). Fracture spacing in layered rocks: a new explanation based on the stress transition.

Barabási, A.-L. (2022). *Network Science*. Northeastern University.

Barthelemy, M. (2018). *Morphogenesis of spatial networks*. Springer.

Berg, S. S. and Skar, T. (2005). Controls on damage zone asymmetry of a normal fault zone: Outcrop analyses of a segment of the moab fault, se utah. *Journal of Structural Geology*, 27:1803–1822.

Berkowitz, B. (1995). Analysis of fracture network connectivity using percolation theory.

Berkowitz, B. and Balberg, I. (1993). Percolation theory and its application to groundwater hydrology hydrological service, ministry. of agriculture, jerusalem, I•rael.

Bertotti, G., Barnhoorn, A., and Bisdom, K. (2016). Geology of mode i, hybrid and mode ii fractures: what do we really know? *78th EAGE Conference and Exhibition 2016*.

Bisdom, K., Nick, H. M., and Bertotti, G. (2017). An integrated workflow for stress and flow modelling using outcrop-derived discrete fracture networks. *Computers and Geosciences*, 103:21–35.

Boersma, Q., Hardebol, N., Barnhoorn, A., and Bertotti, G. (2018). Mechanical factors controlling the development of orthogonal and nested fracture network geometries. *Rock Mechanics and Rock Engineering*, 51:3455–3469.

Boersma, Q., Prabhakaran, R., Bezerra, F. H., and Bertotti, G. (2019). Linking natural fractures to karst cave development: A case study combining drone imagery, a natural cave network and numerical modelling. *Petroleum Geoscience*, 25:454–469.

Bonneau, F., Pochet, A., Caumon, G., and Renard, P. (2016). A workflow for correlated discrete fracture network simulation constrained by microseismic data.

Byerlee, J. (1978). Friction of rocks.

Caine, J. S. and Evans, J. P. (1996). Fault zone architecture and permeability structure. *Geology*, 24:1025–1028.

Cappa, F., Guglielmi, Y., and De Barros, L. (2022). Transient evolution of permeability and friction in a slowly slipping fault activated by fluid pressurization. *Nature Communications*, 13(1):1–9.

Ceccato, A., Viola, G., Antonellini, M., Tartaglia, G., and Ryan, E. J. (2021). Constraints upon fault zone properties by combined structural analysis of virtual outcrop models and discrete fracture network modelling. *Journal of Structural Geology*, 152:104444.

Childs, C., Manzocchi, T., Walsh, J. J., Bonson, C. G., Nicol, A., and Schöpfer, M. P. (2009). A geometric model of fault zone and fault rock thickness variations. *Journal of Structural Geology*, 31:117–127.

Ciftci, B. and Bozkurt, E. (2007). Anomalous stress field and active breaching at relay ramps: A field example from gediz graben, sw turkey. *Geological Magazine - GEOL MAG*, 144.

Dichiarante, A. M., McCaffrey, K. J., Holdsworth, R. E., Bjørnarå, T. I., and Dempsey, E. D. (2020). Fracture attribute scaling and connectivity in the devonian orcadian basin with implications for geologically equivalent sub-surface fractured reservoirs. *Solid Earth*, 11(6):2221–2244.

Diestel, R. (2005). *Graph Theory (3rd ed.)*. Springer.

Dijk, J. V. (2019). The dmx protocol: A new generation of geology driven 3d discrete fault and fracture network modelling janpieter van dijk, dragon oil.

Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*.

Dyer, R. (1988). Using joint interactions to estimate paleostress ratios.

Erdos, P., Rényi, A., et al. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60.

Faulkner, D. R., Jackson, C. A., Lunn, R. J., Schlische, R. W., Shipton, Z. K., Wibberley, C. A., and Withjack, M. O. (2010). A review of recent developments concerning the structure, mechanics and fluid flow properties of fault zones.

Faulkner, D. R., Mitchell, T. M., Jensen, E., and Cembrano, J. (2011). Scaling of fault damage zones with displacement and the implications for fault growth processes. *Journal of Geophysical Research: Solid Earth*, 116.

Fossen, H., Schultz, R. A., Shipton, Z. K., and Mair, K. (2007). Deformation bands in sandstone: a review.

Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41.

Gale, J., Laubach, S., Olson, J., Eichhubl, P., and Fall, A. (2014). Natural fractures in shale: A review and new observations. *AAPG Bulletin*, 98:2165–2216.

Gauthier, T. D. (2001). Detecting trends using spearman's rank correlation coefficient. *Environmental forensics*, 2(4):359–362.

Gross, M. R. (1993). The origin and spacing of cross joints: examples from the monterey formation, santa barbara coastline, california.

Hansberry, R. L., King, R. C., Holford, S. P., Hand, M., and Debenham, N. (2021). How wide is a fault damage zone? using network topology to examine how fault-damage zones overprint regional fracture networks. *Journal of Structural Geology*, 146.

Hestir, K. and Long, J. C. S. (1990). Analytical expressions for the permeability of random two-dimensional poisson fracture networks based on regular lattice percolation and equivalent media theories. *Journal of Geophysical Research: Solid Earth*, 95(B13):21565–21581.

IPCC (2022). *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, Cambridge, UK and New York, NY, USA.

Johannessen, M. (2017). *Fault core and its geostatistical analysis: Insight into the fault core thickness and fault displacement*. PhD thesis.

Jones, T. A. and Detwiler, R. L. (2016). Fracture sealing by mineral precipitation: The role of small-scale mineral heterogeneity. *Geophysical Research Letters*, 43:7564–7571.

Kaldi, J., Daniel, R., Tenthorey, E., Michael, K., Schacht, U., Nicol, A., Underschultz, J., and Backe, G. (2013). Containment of co2 in ccs: Role of caprocks and faults. *Energy Procedia*, 37:5403–5410. GHGT-11 Proceedings of the 11th International Conference on Greenhouse Gas Control Technologies, 18-22 November 2012, Kyoto, Japan.

Kang, M. and Petrasek, Z. (2015). Random graphs : Theory and applications from nature to society to the brain.

Kesten, H. (1982). *Percolation theory for mathematicians*, volume 194. Springer.

Kim, Y. S., Peacock, D. C., and Sanderson, D. J. (2004). Fault damage zones. *Journal of Structural Geology*, 26:503–517.

Kim, Y. S. and Sanderson, D. J. (2005). The relationship between displacement and length of faults: A review. *Earth-Science Reviews*, 68:317–334.

Kim, Y.-S. and Sanderson, D. J. (2010). Inferred fluid flow through fault damage zones based on the observation of stalactites in carbonate caves. *Journal of Structural Geology*, 32(9):1305–1316. Faulting and fracturing of carbonate rocks: New insights into deformation mechanisms, petrophysics and fluid flow properties.

Lahiri, S. (2021). Estimating effective permeability using connectivity and branch length distribution of fracture network. *Journal of Structural Geology*, 146:104314.

Li, L. and Ji, S. (2021). A new interpretation for formation of orthogonal joints in quartz sandstone. *Journal of Rock Mechanics and Geotechnical Engineering*, 13:289–299.

Lunn, R. J., Shipton, Z. K., and Bright, A. M. (2008). How can we improve estimates of bulk fault zone hydraulic properties? *Geological Society Special Publication*, 299:231–237.

Maillot, J., Davy, P., Le Goc, R., Darcel, C., and de Dreuzy, J. (2016). Connectivity, permeability, and channeling in randomly distributed and kinematically defined discrete fracture network models. *Water Resources Research*, 52(11):8526–8545.

Manzocchi, T. (2002). The connectivity of two-dimensional networks of spatially correlated fractures. *Water Resources Research*, 38:1–1–1–20.

Masihi, M., King, P. R., and Nurafza, P. (2007). Fast estimation of connectivity in fractured reservoirs using

percolation theory. *SPE Journal*, 12(02):167–178.

Miliorizos, M. and Ruffell, . A. (1998). Kinematics of the watchet-cothelstone-hatch fault system: implications for the fault history of the wessex basin and adjacent areas.

Mitchell, T. M. and Faulkner, D. R. (2009). The nature and origin of off-fault damage surrounding strike-slip fault zones with a wide range of displacements: A field study from the atacama fault system, northern chile. *Journal of Structural Geology*, 31:802–816.

Nixon, C. W., Nærland, K., Rotevatn, A., Dimmen, V., Sanderson, D. J., and Kristensen, T. B. (2020). Connectivity and network development of carbonate-hosted fault damage zones from western malta. *Journal of Structural Geology*, 141:104212.

Ogata, K., Senger, K., Braathen, A., and Tveranger, J. (2014). Fracture corridors as seal-bypass systems in siliciclastic reservoir-cap rock successions: Field-based insights from the jurassic entrada formation (se utah, usa). *Journal of Structural Geology*, 66:162–187.

Panza, E., Sessa, E., Agosta, F., and Giorgioni, M. (2018). Discrete fracture network modelling of a hydrocarbon-bearing, oblique-slip fault zone: Inferences on fault-controlled fluid storage and migration properties of carbonate fault damage zones. *Marine and Petroleum Geology*, 89:263–279.

Peacock, D. C., Dimmen, V., Rotevatn, A., and Sanderson, D. J. (2017). A broader classification of damage zones. *Journal of Structural Geology*, 102:179–192.

Peacock, D. C., Nixon, C. W., Rotevatn, A., Sanderson, D. J., and Zuluaga, L. F. (2016). Glossary of fault and other fracture networks. *Journal of Structural Geology*, 92:12–29.

Peacock, D. C. and Sanderson, D. J. (2018). Structural analyses and fracture network characterisation: Seven pillars of wisdom. *Earth-Science Reviews*, 184:13–28.

Philipp, S. L. (2008). Geometry and formation of gypsum veins in mudstones at watchet, somerset, sw england. *Geological Magazine*, 145:831–844.

Prabhakaran, R. (2021). Naturally fractured reservoir characterization advanced workflows for discrete fracture network modeling.

Renshaw, C. E. (1999). Connectivity of joint networks with power law length distributions. *Water Resources Research*, 35:2661–2670.

Rice, J. R., Sammis, C. G., and Parsons, R. (2005). Off-fault secondary failure induced by a dynamic slip pulse. *Bulletin of the Seismological Society of America*, 95:109–134.

Rizzo, R., Fazeli, H., Maier, C., March, R., Egya, D., Doster, F., Kubeyev, A., Kampman, N., Bisdom, K., Snippe, J., Senger, K., Betlem, P., Phillips, T., Forbes Inskip, N., Esegbue, O., and Busch, A. (2020). Understanding fault and fracture networks to de-risk geological leakage from subsurface storage sites.

Sanderson, D. J. and Nixon, C. W. (2015). The use of topology in fracture network characterization. *Journal of Structural Geology*, 72:55–66.

Sanderson, D. J. and Nixon, C. W. (2018). Topology, connectivity and percolation in fracture networks. *Journal of Structural Geology*, 115:167–177.

Sarker, S., Veremyev, A., Boginski, V., and Singh, A. (2019). Critical nodes in river networks. *Scientific reports*, 9(1):1–11.

Savage, H. M. and Brodsky, E. E. (2011). Collateral damage: Evolution with displacement of fracture distribution and secondary fault strands in fault damage zones. *Journal of Geophysical Research: Solid Earth*, 116.

Schober, P., Boer, C., and Schwarte, L. A. (2018). Correlation coefficients: Appropriate use and interpretatio, anesthesia &analgesia.

Shipton, Z. K. and Cowie, P. A. (2001). Damage zone and slip-surface evolution over mm to km scales in high-porosity navajo sandstone, utah.

Smeraglia, L., Mercuri, M., Tavani, S., Pignalosa, A., Kettermann, M., Billi, A., and Carminati, E. (2021). 3d discrete fracture network (dfn) models of damage zone fluid corridors within a reservoir-scale normal fault in carbonates: Multiscale approach using field data and uav imagery. *Marine and Petroleum Geology*, 126:104902.

Souque, C., Knipe, R. J., Davies, R. K., Jones, P., Welch, M. J., and Lorenz, J. (2019). Fracture corridors and fault reactivation: Example from the chalk, isle of thanet, kent, england. *Journal of Structural Geology*, 122:11–26.

Stauffer, D. Aharony, A. (1992). *Introduction To Percolation Theory: Second Edition (2nd ed.)*.

Sævik, P. N. and Nixon, C. W. (2017). Inclusion of topological measurements into analytic estimates of effective permeability in fractured media. *Water Resources Research*, 53:9424–9443.

Tanikawa, W., Mukoyoshi, H., and Tadai, O. (2012). Experimental investigation of the influence of slip velocity

and temperature on permeability during and after high-velocity fault slip. *Journal of Structural Geology*, 38:90–101.

Tchalenko, J. and Ambraseys, N. N. (1970). Structural analysis of the dasht-e bayaz (iran) earthquake fractures. *Geological Society of America Bulletin*, 81(1):41–60.

Twiss, Moores, and Freeman (2007). Structural geology by r. j. twiss and e. m. moores. w. h. freeman co., san francisco, 1992. no. of pages: 532. isbn 0 7167 2252 6. *Geological Journal - GEOL J*, 29:382–383.

Valentini, L., Perugini, D., and Poli, G. (2007). The "small-world" topology of rock fracture networks. *Physica A: Statistical Mechanics and its Applications*, 377:323–328.

van der Zee, W., Wibberley, C. A., and Urai, J. L. (2008). The influence of layering and pre-existing joints on the development of internal structure in normal fault zones: The lodève basin, france. *Geological Society Special Publication*, 299:57–74.

Vazaios, I., Vlachopoulos, N., and Diederichs, M. (2017). Integration of lidar-based structural input and discrete fracture network generation for underground applications. *Geotechnical and Geological Engineering*, 35:2227–2251.

Voeckler, H. and Allen, D. (2012). Estimating regional-scale fractured bedrock hydraulic conductivity using discrete fracture network (dfn) modeling. *Hydrogeology Journal*, 20(6):1081–1100.

Vrolijk, P. J., Urai, J. L., and Kettermann, M. (2016). Clay smear: Review of mechanisms and applications. *Journal of Structural Geology*, 86:95–152.

Welch, M. J., Davies, R. K., Knipe, R. J., and Tueckmantel, C. (2009). A dynamic model for fault nucleation and propagation in a mechanically layered section. *Tectonophysics*, 474(3):473–492.

Wibberley, C., Gonzalez-Dunia, J., and Billon, O. (2016). Faults as barriers or channels to production-related flow: Insights from case studies. *Petroleum Geoscience*, 23:petgeo2016–057.

Wibberley, C. A., Yielding, G., and Toro, G. D. (2008). Recent advances in the understanding of fault zone internal structure: A review.

Yielding, G., Freeman, B., and Needham, D. T. (1997). Quantitative fault seal prediction. *AAPG Bulletin*, 81.

Zappone, A., Rinaldi, A. P., Grab, M., Wenning, Q. C., Roques, C., Madonna, C., Obermann, A. C., Bernasconi, S. M., Brennwald, M. S., Kipfer, R., et al. (2021). Fault sealing and caprock integrity for co 2 storage: an in situ injection experiment. *Solid Earth*, 12(2):319–343.

Zoback, M. D. and Townend, J. (2000). How faulting keeps the crust strong. *Geology*, 28:399–402.

A

Slip tendency

Fault slip or reactivation can have an important impact on along-fault permeability (Tanikawa et al., 2012). Slip along fault planes in brittle porous rocks has been related to an increase in permeability (Cappa et al., 2022), whereas slip in clay-rich shales could lead to a reduction in permeability (Vrolijk et al., 2016). It is therefore useful to shortly discuss the fundamental principles of fault slip.

According to brittle deformation principles fault slip occurs when the critical friction coefficient is reached (Byerlee, 1978). Important controls related to frictional failure are phyllosilicate content in the fault core, the presence of fluids and the local state of stress. Zoback and Townend (2000) showed that faults in the subsurface are generally critically stressed which, counterintuitively, ensures the stability of the subsurface by prohibiting pore pressure higher than hydrostatic pressure.

Earthquake nucleation predictions generally use rate- and state-dependent friction laws to determine periods of stability and aseismic slip before instability (Faulkner et al., 2010). Laboratory based rate and state friction principles explain the response of the dynamic friction coefficient to a sudden increase in sliding velocity (fig. A.1). After an immediate increase of the friction coefficient due to a change of velocity (rate effect) the frictional coefficient evolves in a time-dependent trend over a particular slip displacement (state effect). The phenomenological derived equation A.1 describes the relationship between friction and slip rate with μ as the fictional coefficient and μ_0 the initial friction coefficient, A and B are experimentally determined constant (largely dependent on gouge composition), V and V_0 are the new and initial sliding velocity, θ is the state variable (which is a function of time, normal stress and displacement) and D_C the slip weakening distance.

$$\mu = \mu_0 + Aln(\frac{V}{V_0}) + Bln(\frac{V_0\theta}{D_C}) \tag{A.1}$$

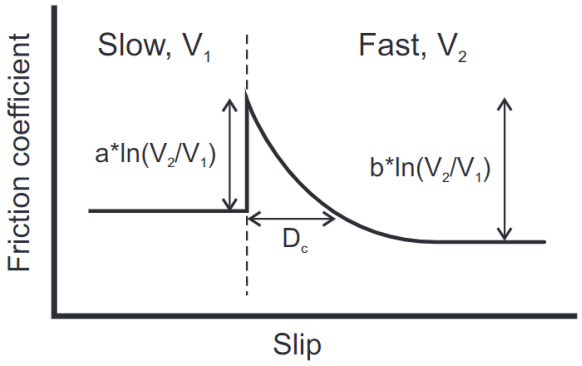


Figure A.1: Faulkner et al. (2010)

B

Betweenness Centrality

The Betweenness Centrality (c_B) (Freeman, 1977) is a measure to evaluate the relative importance of a node or an edge for the interconnectedness in a spatial graph and has been used to analyse fracture networks (Prabhakaran, 2021). The quantity is determined by the ratio between the number of times the node (or edge) lays on the geodesic path between all node pairs combinations in the graph to the total amount of shortest paths between all node pairs. The $c_B(e)$ (edge betweenness centrality) is given by equation B.1 where $\sigma(s,t)$ is the number of shortest paths between node s and t. $\sigma(s,t \mid e)$ is the amount of times edge e lays on these shortest paths. $\mathcal{N}$ is the normalization factor, $\mathcal{N} = \frac{(N-1)(N-2)}{2}$ with N as the number of nodes.

$$c_B(e) = \frac{1}{\mathcal{N}} \sum_{s,t \in V} \frac{\sigma(s,t \mid e)}{\sigma(s,t)} \tag{B.1}$$

In non-lattice spatial graphs the complex relation between distance and topology leads to nontrivial paths of high BC (Barthelemy, 2018). One way to explain the concept of BC is by the example of traffic flow. Consider every node pair in the graph as an origin and destination pair. When it is assumed that traffic will always follow the shortest path between its origin and destination, then the BC will indicate the intensity of traffic flowing through every node and/or edge in the network.

Edge weights can be introduced to increase the accuracy of BC flow path predictions. Euclidean distance between two node pairs can be used as weights. Prabhakaran (2021) proposed using fracture conductivity (F_c) which is a function of aperture (w) and length (l) (eq. B.2).

$$F_c \sim \frac{w^3}{l} \tag{B.2}$$

The betweenness centrality has been used for critical node identification (CNI) in river system studies. CNI problems identify the most important node to be deleted. Sarker et al. (2019) used the betweenness centrality to identify the critical nodes in a river system. They found that there exists a Power-law relationship between the number of connected node pairs in the remaining network and the number of removed nodes.

B.0.1. Result

The edge BC for the Konusdalen I outcrop is calculated for every iteration with euclidean distance as edge weight. Figure B.1 shows 4 iterations (200 to 500). The edges with high BC are highlighted in red. A shift of the high BC paths can be observed as the network is dismantled randomly (iteration 100 to 200). In this stage a concentration of high BC onto one path can be observed. The high BC paths are then broken up between iteration 300 and 500. This is illustrated by figure B.2 where a sharp drop in maximum normalized edge BC coincides with a sharp drop in LCR. Figure B.3 shows another random-based realisation. In this realisation the same sudden drop can be seen. However, it occurs at a later stage of the removal process and shows a less significant drop.

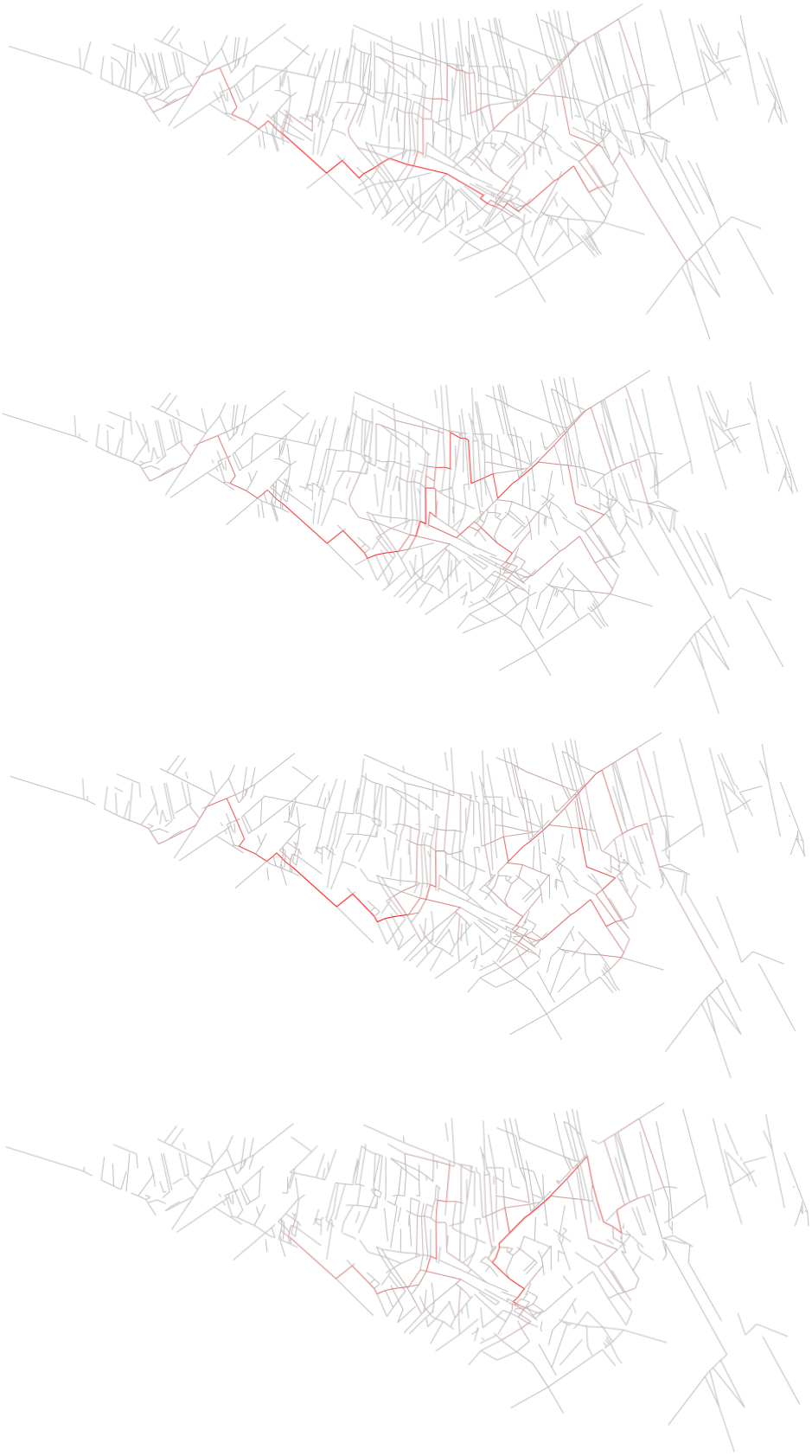


Figure B.1: Iteration 200, 300, 400 and 500 of Konusdalen I outcrop. Red highlights high BC values. A shift of high BC paths can be observed when the path is broken.

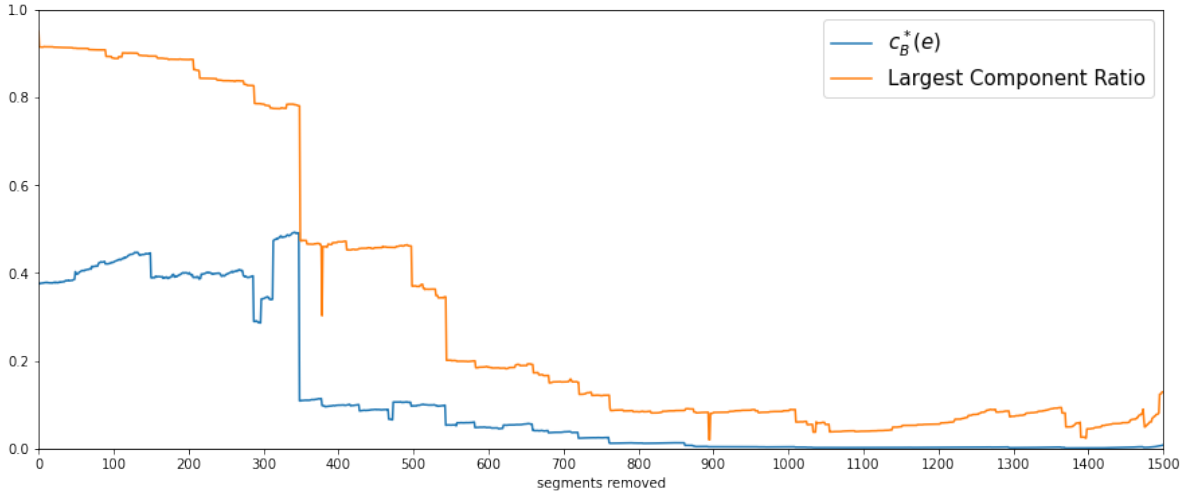


Figure B.2: Maximum normalized edge BC ($c_B^*(e)$) and Largest Component Ratio. Both show relatively stable behaviour between iteration 0 and 350. A sharp drop in both BC and largest component ratio can be observed indicating a co-dependency of between the two.

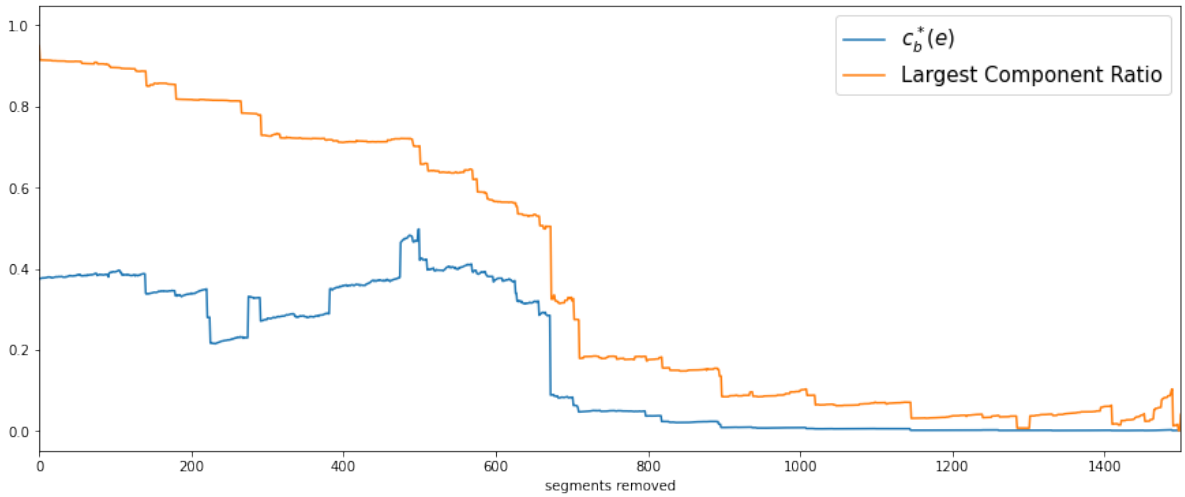


Figure B.3: Maximum normalized edge BC ($c_B^*(e)$) and Largest Component Ratio. The sharp drop for this realization occurs later due to the random nature of fracture segment removal. This realization shows a more gradual network dismantling than illustrated in figureB.2.

B.0.2. Discussion

The betweenness centrality helps to understand the importance of certain fracture segments and therefore the impact on LCR when they are removed. The BC highlights the most important connecting edges. The removal of the edge with the highest BC will therefore have a significant impact on the LCR. When this segment is removed a sharp drop in LCR is expected. This can be associated with the collapse of fracture network connectivity.

This is shown by figure B.2. When the edge with the maximum BC is removed and this edge is the last connecting feature that holds the network together, then a sharp drop in LCR (the giant component is broken up) and a sharp drop in BC (no shortest paths can be formed between node pairs) is observed.

Figure B.3 shows this collapse later in the removal process. Therefore, the sharp drop is less significant, because the largest component is already dismantled to a certain degree.

C

Segment Reduction Algorithm

```
import geopandas
import matplotlib.pyplot as plt
import momepy
import networkx as nx
from contextily import add_basemap
from libpysal import weights
import numpy as np
import math
import pandas as pd
import random
from time import gmtime, strftime
import mplstereonet
import matplotlib.colors
from tqdm import tqdm
from scipy.optimize import curve_fit
import plotly.figure_factory as ff
```

```
#network = geopandas.read_file('outcrops\Shapefiles\Eye.shp')
#network = geopandas.read_file('outcrops\Shapefiles\Relay.shp')
# network = geopandas.read_file('outcrops\Shapefiles\Antithetic.shp')
#network = geopandas.read_file('outcrops\Shapefiles\Konusdalen_I.shp')
#network = geopandas.read_file('outcrops\Shapefiles\Konusdalen_II.shp')
#network = geopandas.read_file('outcrops\Shapefiles\Cluster.shp')
# network = geopandas.read_file('outcrops\Shapefiles\Moab_wall.shp')
#network = geopandas.read_file('outcrops\Shapefiles\Somerset_A.shp')
network = geopandas.read_file('outcrops\Shapefiles\Somerset_B.shp')
```

```
# check reduction mechanism (can only check one):
Remove_random = True
Remove_length = False
Remove_orientation = False

# If remove-orientation is selected:
# devide into two or three sets:

# Two sets
two_sets = False
Orient_1st = [335,15,155,195] # Orientation to remove first, N-S is between 315-45 and
    135-225

# Three sets
three_sets = False
Orient_1st = [315, 355, 135, 175, 5, 45, 185,225] # [T_left    + 180           T_R    + 180]
Orient_2nd = [335,15,155,195]

# Perform betweenness calculation
Calc_betweenness = False

# Give area of outcrop (area in shapefile dimensions)
```

```
conversion = 638.2364835100512
A =  195.29*conversion #4382341.463435001     # area in px**2

# Give data storage location
file = 'data/Somerset_B'

# perform analysis multiple times to obtain best-fit solution in notebook 'Multi_curve_plot'
nruns = 10
```

```
all_fnames = []
for run in range(nruns):  # perform analysis multiple times to obtain best-fit solution in notebook 'Multi_curve_plot'
    G = momepy.gdf_to_nx(network, approach="primal")   # generate a networkx object.

    # alocate memory
    all_graphs = []   # save all networkx object for every iteration
    largest_comp = []  # save the largest comp ratio for every iteration
    xnodes = []       # not used
    ynodes = []       # not used
    inodes= []        # not used
    vnodes = []
    avg_degrees = []  # store average degree of every network generated.
    P21 = []          # save the P21 of every network (fracture intensity m/m2)
    length_set1 = []
    length_set2 = []
    length_set3 = []
    vmaxs = []            # maximum betweeenness centrality in network
    vavg = []
    number_of_comp = []   # total number of components

    # start of loop
    n_edges = len(G.edges)
    itr=0             # keep track of iteration
    for i in range(n_edges): # perform loop until every segment is removed.

        ## GET ALL DATA FROM NETWORK
        # Get node data
        degrees = dict(nx.degree(G)).values()
        xnode = len([i for i in degrees if i == 4])   # get n xnodes of the largest component
        ynode = len([i for i in degrees if i == 3])
        vnode = len([i for i in degrees if i == 2])
        inode = len([i for i in degrees if i == 1])   # get n inodes
        avg_degree = (4*xnode + 3*ynode + 1*inode)/(xnode+ynode+inode)  # determine avg_degree from every node type excluding V-nodes

        xnodes.append(xnode)
        ynodes.append(ynode)
        vnodes.append(vnode)
        inodes.append(inode)
        avg_degrees.append(avg_degree)

        G_matrix = nx.to_pandas_adjacency(G)   # obtain adjacency matrix of the network
        # Obtain total length of largest network
        x = []   # x - coordinates
        y = []   # y -coordinates

        pos = {k: v for k,v in enumerate(G.nodes())}  # get all node coordinates
        for value in pos.values():  # get all the coordinates of the
            x.append(value[0])
            y.append(value[1])

        dist = np.zeros([len(x),len(y)])  # distance between every node in matrix form

        for u in range(len(x)):                       # calculate distance between all points
            for f in range(len(x)):
                dist[u,f] = math.sqrt((x[f] - x[u])**2 + (y[f] - y[u])**2)

        all_lengths = G_matrix * dist    # adjacency matrix * distance matrix = all lengths in network
        all_lengths_matrix = pd.DataFrame(all_lengths)  # all edge lenghts in network

```

```
        # Get the largest component (largest network)
        largest_cc = list(max(nx.connected_components(G), key=len))

        all_lengths = pd.DataFrame(all_lengths)     # convert all_lengths in dataframe, array is not possible for some reason
        all_lengths = all_lengths[largest_cc]       # select the columns of the largest network
        all_lengths = all_lengths.loc[largest_cc]       # select the rows of the largest network
        total_length = all_lengths.to_numpy().sum()/2  # devide by two due to symmetry of the matrix
        total_length_all =  all_lengths_matrix.to_numpy().sum()/2
        largest_network_fraction = total_length / total_length_all  # largest_comp/all_comp ratio
        largest_comp.append(largest_network_fraction) # save in list

        # obtain p21.
        P21.append((total_length_all/A)*100)  # total lenght in px. p21 in 1/px

        ## REMOVING EDGES
        all_mtrx_temp = all_lengths_matrix # need temporal matrix for length removal analysis.
        current_set_removal = []
        while 1:
            if Remove_random:
                edge_start = random.choice(pos) # obtain random edge start
                connected = [to for (fr, to) in G.edges(edge_start)] # find connecting end points.
                if len(connected) != 0:  # if there is an edge
                    edge_end = random.choice(connected) # choose random endpoint connected
                    G.remove_edge(edge_start, edge_end) # remove this edge
                    print(str(len(G.edges())) + ' ' + 'segments remaining' + ' current run: ' + str(run))
                    break
                if len(G.edges()) == 0:
                    print('all edges removed')
                    break

            if Remove_length:
                colm = all_mtrx_temp.max()   # get all max values in columns
                q = colm.to_dict()                # create dictionary
                temp = {val: key for key, val in q.items()}   # remove double values (due to matrix symmetry)
                qq = {val: key for key, val in temp.items()}  # remove double values (due to matrix symmetry)
                sort_segments = dict(sorted(qq.items(), key=lambda x: x[1], reverse=True))  # sort by length
                edge_start = list(sort_segments.keys())[0] # first item of sorted dict, longest segment
                all_mtrx_temp = all_mtrx_temp.replace([list(sort_segments.values())[0]],0)
                    # remove the longest segment from entire length matrix for next iteration
                connected = [to for (fr, to) in G.edges(edge_start)]

                if len(connected) != 0:  # crashes if zero
                    distance = []  # from start point determine all connecting nodes and find furthest away
                    for coord in connected:
                        crd_x = list(coord)[0]
                        crd_y = list(coord)[1]
                        crd_xs = list(edge_start)[0]
                        crd_ys = list(edge_start)[1]
                        distance.append(math.sqrt((crd_x - crd_xs)**2 + (crd_y - crd_ys)**2))
                    edge_end = connected[distance.index(max(distance))]
                    G.remove_edge(edge_start, edge_end)
                    print(str(len(G.edges())) + ' ' + 'segments remaining' + ' current run: ' + str(run))
                    break
                if len(G.edges()) == 0:
                    print('all edges removed')
                    break

```

```
116             if Remove_orientation:
117                 # allocate memory for two fracture sets based on orientation. Fractures are
                        defined by start- and end point
118                 start_1st = []
119                 end_1st = []
120                 start_2nd = []
121                 end_2nd = []
122                 start_3rd = []
123                 end_3rd = []
124 
125                 # seperate in different sets
126                 for p in pos:
127                     edge_start = pos[p]   # all start points
128                     connected = [to for (fr, to) in G.edges(edge_start)] # find all
                            connecting edges.
129                     for edge_end in connected:
130                         compass_brackets, bearing = direction_lookup(edge_end[0], edge_start
                                [0], edge_end[1], edge_start[1]) # find the orientation to all
                                end points
131                         if two_sets:
132                             if Orient_1st[0]<bearing<Orient_1st[1] or Orient_1st[3]<bearing<
                                    Orient_1st[4]:  # Only obtain set 2 segments.
133                                 start_1st.append(edge_start)
134                                 end_1st.append(edge_end)
135                             else:   # only obtain set 1 segments.
136                                 start_2nd.append(edge_start)
137                                 end_2nd.append(edge_end)
138                         if three_sets:
139                             # | T_left                                                + 180
                                                                    |                   T_right
                                                                      +180                        |
140                             if Orient_1st[0] < bearing < Orient_1st[1] or Orient_1st[2] <
                                    bearing < Orient_1st[3] or Orient_1st[4] < bearing <
                                    Orient_1st[5] or Orient_1st[6] < bearing < Orient_1st[7]:
141                                 start_1st.append(edge_start)
142                                 end_1st.append(edge_end)
143                             if Orient_2nd[0] < bearing < Orient_2nd[1] or Orient_2nd[2] <
                                    bearing < Orient_2nd[3]:
144                                 start_2nd.append(edge_start)
145                                 end_2nd.append(edge_end)
146                             else:
147                                 start_3rd.append(edge_start)
148                                 end_3rd.append(edge_end)
149 
150                 if two_sets:
151                     # first randomly remove from 1st list
152                     if len(start_1st) !=0:
153                         edge_start = random.choice(start_1st)   # randomly select edge start
                                from start list.
154                         edge_end = end_1st[start_1st.index(edge_start)]   # find the
                                corresponding end-point to obtain segment.
155                         G.remove_edge(edge_start, edge_end)
156                         length_set1.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (
                                edge_end[1] - edge_start[1])**2))
157                         print(str(len(G.edges())) + ' ' + 'segments remaining. Removing: set1
                                ...' + ' current run: ' + str(run))
158                         break
159 
160                     # remove from 2nd list
161                     else:
162                         edge_start = random.choice(start_2nd) # randomly select edge start
                                from start list.
163                         edge_end = end_2nd[start_2nd.index(edge_start)] # find the
                                corresponding end-point to obtain segment.
164                         G.remove_edge(edge_start, edge_end)
165                         length_set2.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (
                                edge_end[1] - edge_start[1])**2))
166                         print(str(len(G.edges())) + ' ' + 'segments remaining. Removing: set2
                                ...'+ ' current run: ' + str(run))
167                         break
168 
```

```
169                 if three_sets:
170                     # first randomly remove from 1st list
171                     if len(start_1st) !=0:
172                         edge_start = random.choice(start_1st)   # randomly select edge start
                                from start list.
173                         edge_end = end_1st[start_1st.index(edge_start)]   # find the
                                corresponding end-point to obtain segment.
174                         G.remove_edge(edge_start, edge_end)
175                         length_set1.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (
                                edge_end[1] - edge_start[1])**2))
176                         print(str(len(G.edges())) + ' ' + 'segments remaining. Removing: set1
                                ...' + ' current run: ' + str(run))
177                         break
178
179                     if len(start_2nd) != 0:
180                         edge_start = random.choice(start_2nd)   # randomly select edge start
                                from start list.
181                         edge_end = end_2nd[start_2nd.index(edge_start)]   # find the
                                corresponding end-point to obtain segment.
182                         G.remove_edge(edge_start, edge_end)
183                         length_set2.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (
                                edge_end[1] - edge_start[1])**2))
184                         print(str(len(G.edges())) + ' ' + 'segments remaining. Removing: set2
                                ...' + ' current run: ' + str(run))
185                         break
186
187                     # remove from 3rd list
188                     else:
189                         edge_start = random.choice(start_3rd) # randomly select edge start
                                from start list.
190                         edge_end = end_3rd[start_3rd.index(edge_start)] # find the
                                corresponding end-point to obtain segment.
191                         G.remove_edge(edge_start, edge_end)
192                         length_set3.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (
                                edge_end[1] - edge_start[1])**2))
193                         print(str(len(G.edges())) + ' ' + 'segments remaining. Removing: set3
                                ...'+ ' current run: ' + str(run))
194                         break
195
196         # plot the network every i-th iteration. Save to folder to later generate GIF.
197         if Calc_betweenness:
198             print('calculating betweenness centrality...')
199             len_weights = []
200
201             for l in range(len(list(G.edges()))):
202                 edge_start = list(G.edges())[l][0]
203                 edge_end = list(G.edges())[l][1]
204                 len_weights.append(math.sqrt((edge_end[0] - edge_start[0])**2 + (edge_end[1]
                        - edge_start[1])**2))
205
206             weighted_graph = G
207             nx.set_edge_attributes(weighted_graph, {e: {'weight': len_weights[ix]} for ix,e
                    in enumerate(weighted_graph.edges)})
208             centrality = nx.edge_betweenness_centrality(weighted_graph, weight='weight',
                    normalized=False)
209             nx.set_edge_attributes(G, {e: {'betweenness': list(centrality.values())[ix]} for
                    ix,e in enumerate(G.edges)})
210             vmaxs.append(max(list(centrality.values())))
211             vavg.append(np.average(list(centrality.values())))
212             #norm_factor.append()
213
214         if itr % 100 == 0 and run == 0:
215
216             # Plot degree histogram
217             x = ('1','3','4')
218             yy = (inodes[itr],ynodes[itr],xnodes[itr])
219             y = (inodes[itr]/(inodes[itr]+ynodes[itr]+xnodes[itr]),ynodes[itr]/(inodes[itr]+
                    ynodes[itr]+xnodes[itr]),xnodes[itr]/(inodes[itr]+ynodes[itr]+xnodes[itr]))
220
221             fig = plt.figure(figsize=(8,4))
222             ax = fig.add_subplot(121)
```

```
            ax.bar(x,y,align='center', color = ['green','red','blue'], ec="black") # A bar
                chart
            ax.set_ylim([0, 1])
            ax.set_xlabel('Node Degree')
            ax.set_ylabel('Frequency')
            for i in range(len(x)):
                ax.text(i, y[i], yy[i], ha='center', va = 'bottom')
            ax.set_title('Node histogram', y=1.10, fontsize=15)

            # Plot rose diagram
            # http://geologyandpython.com/structural_geology.html
            rose = []
            for p in pos:
                edge_start = pos[p]   # all start points
                connected = [to for (fr, to) in G.edges(edge_start)] # find all connecting
                    edges.
                for edge_end in connected:
                    compass_brackets, bearing = direction_lookup(edge_end[0], edge_start[0],
                        edge_end[1], edge_start[1]) # find the orientation to
                    rose.append(bearing)

            strikes = np.array(rose)
            bin_edges = np.arange(-5, 366, 10)
            number_of_strikes, bin_edges = np.histogram(strikes, bin_edges)
            number_of_strikes[0] += number_of_strikes[-1]
            half = np.sum(np.split(number_of_strikes[:-1], 2), 0)
            two_halves = np.concatenate([half, half])

            ax = fig.add_subplot(122, projection='polar')
            ax.bar(np.deg2rad(np.arange(0, 360, 10)), two_halves,
                   width=np.deg2rad(10), bottom=0.0, color='r', edgecolor='k')
            ax.set_theta_zero_location('N')
            ax.set_theta_direction(-1)
            #ax.set_yticks(np.arange(100, 450, step=50))
            ax.set_title('Fracture rosediagram', y=1.10, fontsize=15)
            fig.tight_layout()
            plt.savefig(file + '/Plots/iter'+str(itr)+'bar_rose'+'.png')
            plt.show()

            # Plot network
            plt.rcParams["figure.figsize"] = (20,20)
            positions = {n: [n[0], n[1]] for n in list(G.nodes)}
            # set color to degree
            d = dict(G.degree)   # Get node degree
            degrr = list(d.values())
            color_map = []
            for degr in degrr:
                if degr == 1:
                    color_map.append('green')
                elif degr == 2:
                    color_map.append('black')
                elif degr == 3:
                    color_map.append('red')
                elif degr == 4:
                    color_map.append('blue')
                else:
                    color_map.append('white')

            # Plot network
            plt.plot(figsize=(80, 80), dpi=100)
            plt.axis('equal')
            # Normal plot
            if Calc_betweenness == False:
                nx.draw(G, positions, node_size=10, node_color=color_map)
            else:
                # Betweenness plot
                # edge color based on edge_betweenness
                #colors = [G[u][v]['betweenness'] for u,v in edges]
                edges, color = zip(*nx.get_edge_attributes(G, 'betweenness').items())
                cmap = matplotlib.colors.LinearSegmentedColormap.from_list("", ["silver","red
                    "])
```

```
290                 vmin = min(list(centrality.values()))
291                 vmax = max(list(centrality.values()))
292                 nx.draw_networkx_edges(G, positions, edge_color=color, edge_cmap=cmap,
                        edge_vmin=vmin, edge_vmax=vmax)
293                 sm = plt.cm.ScalarMappable(cmap=cmap, norm=plt.Normalize(vmin = vmin, vmax=
                        vmax))
294                 sm._A = []
295                 #plt.colorbar(sm)
296             plt.savefig(file + '/Plots/iter'+str(itr)+'.png')
297             #plt.show()
298         number_of_comp.append(nx.number_connected_components(G))
299         all_graphs.append(G)   # save all generated network (networkx object) to list.
300         itr += 1
301     # store data
302     fname = file + "\\" + "Data_"+ "run_" + str(run) +"_"+ strftime("%Y_%m_%d_%H_%M_%S",
            gmtime()) + ".csv"
303     largest_comp_data = pd.DataFrame(largest_comp, columns= ['largest_comp'])
304     avg_degree_data = pd.DataFrame(avg_degrees, columns= ['avg_degree'])
305     P21_data = pd.DataFrame(P21, columns= ['P21'])
306     xnodes_data = pd.DataFrame(xnodes, columns= ['xnodes'])
307     ynodes_data = pd.DataFrame(ynodes, columns= ['ynodes'])
308     inodes_data = pd.DataFrame(inodes, columns= ['inodes'])
309     n_components = pd.DataFrame(number_of_comp, columns= ['n_components'])
310
311     if Calc_betweenness:
312         vmax_data = pd.DataFrame(vmaxs, columns= ['vmax'])
313         vavg_data = pd.DataFrame(vmaxs, columns= ['vavg'])
314         data = pd.concat([largest_comp_data, P21_data, avg_degree_data, xnodes_data,
                ynodes_data, inodes_data, vmax_data, vavg_data], axis=1, join='inner')
315     else:
316         data = pd.concat([largest_comp_data, P21_data, avg_degree_data, xnodes_data,
                ynodes_data, inodes_data], axis=1, join='inner')
317     data.to_csv(fname)
318     all_fnames.append(fname)
```

D

Outcrops

Figure D.1

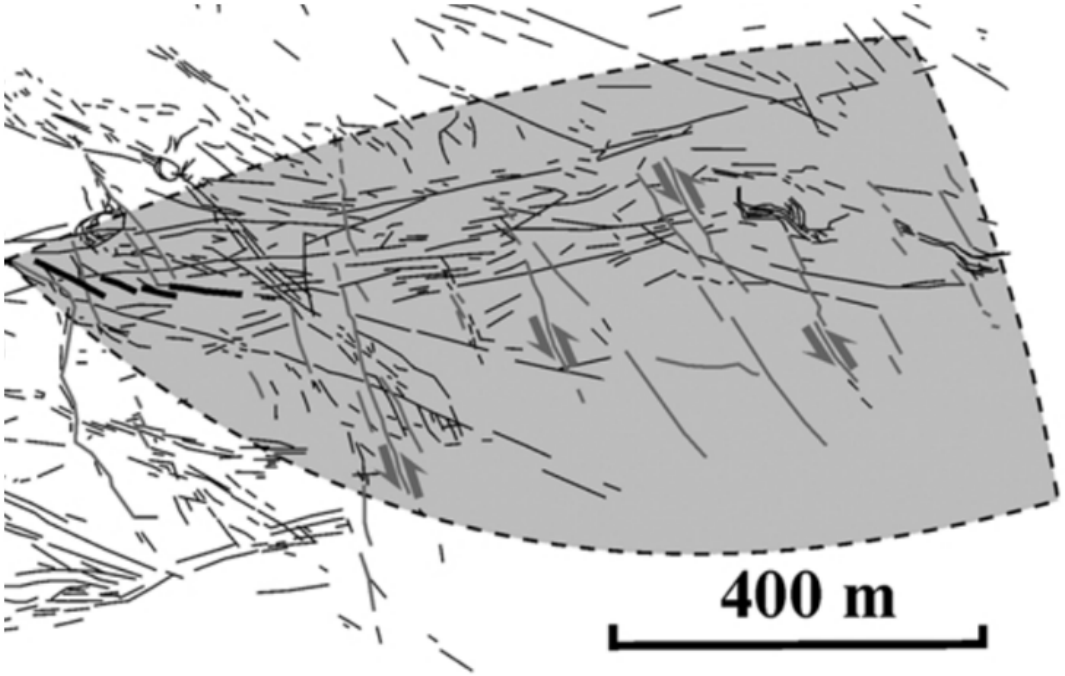


Figure D.2

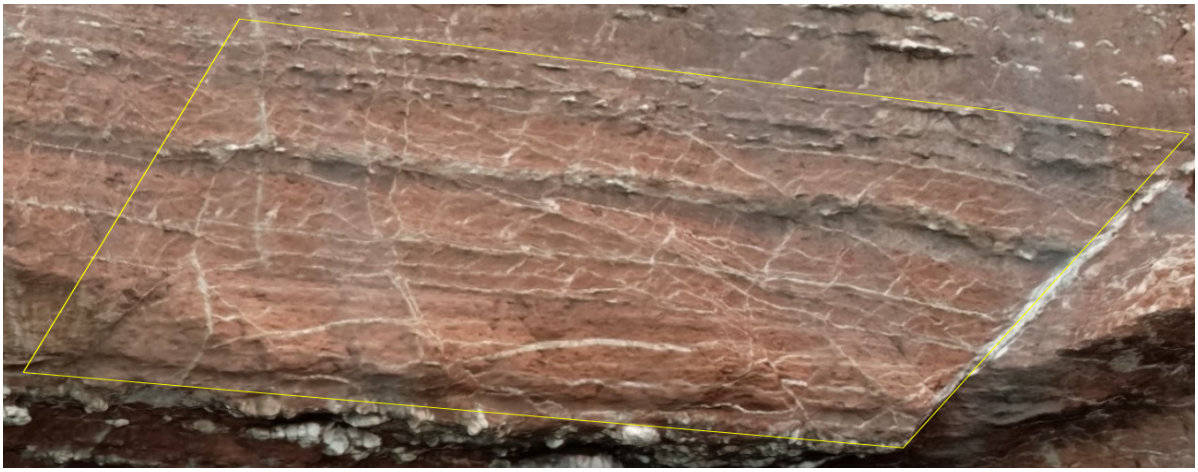


Figure D.3

Figure D.4

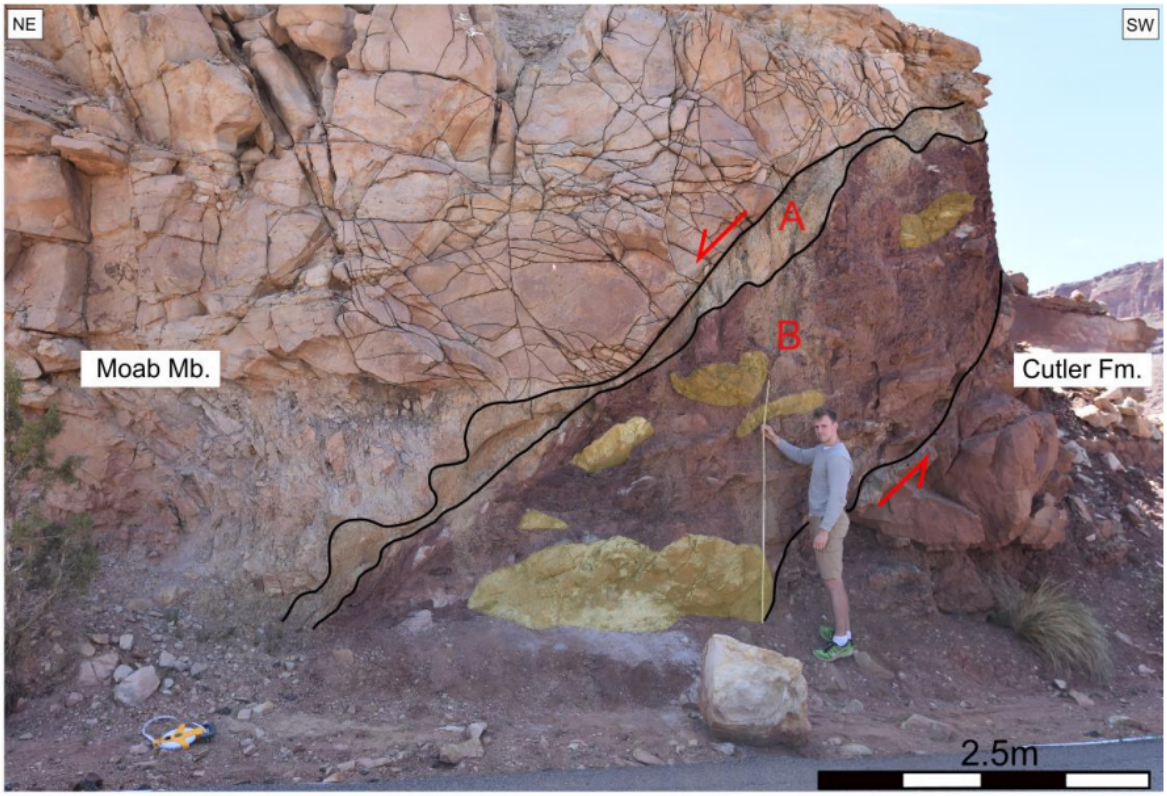


Figure D.5

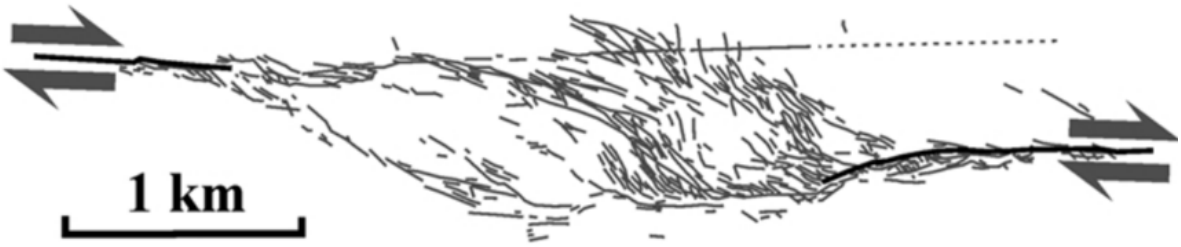


Figure D.6

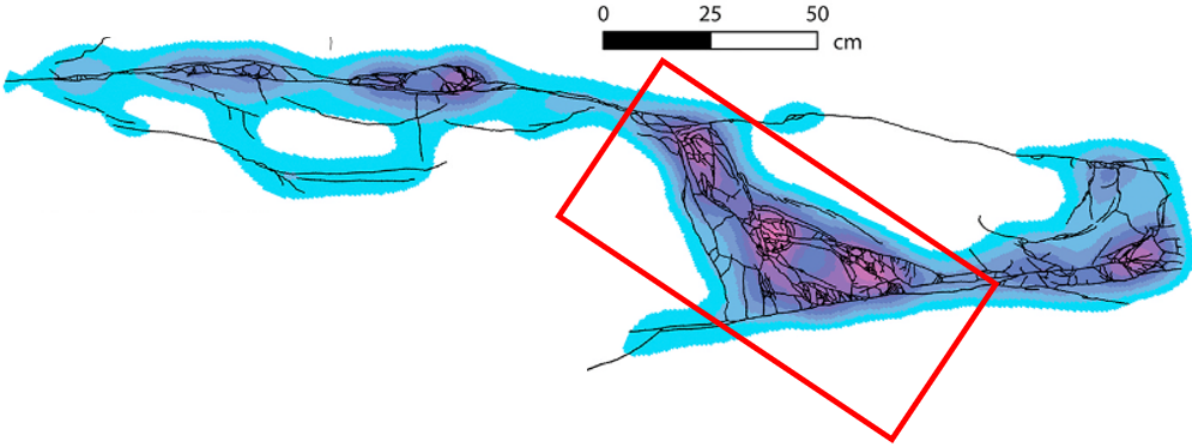


Figure D.7

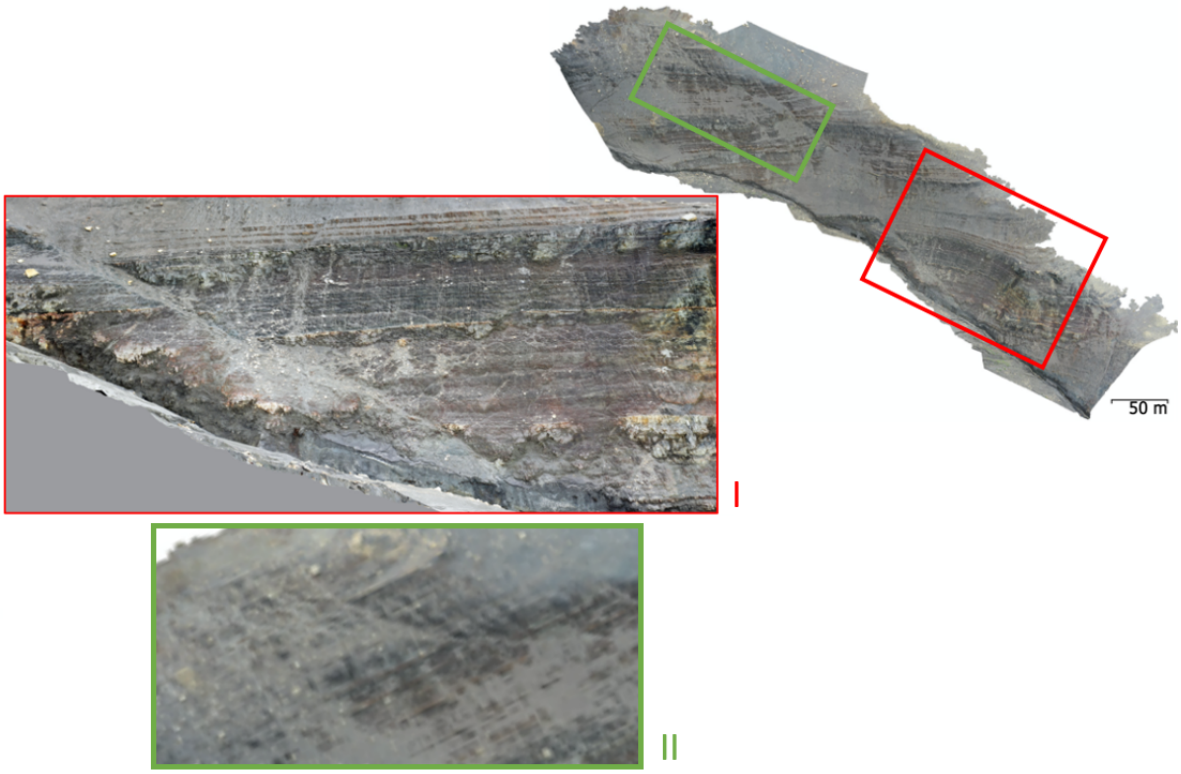


Figure D.8