Report
R-17-12
February 2018

Methods and workflow for geometric and hydraulic conditioning

DFN-R – status report 2013–2015

Tomas Bym
Jan Hermanson

SVENSK KÄRNBRÄNSLEHANTERING AB

SWEDISH NUCLEAR FUEL
AND WASTE MANAGEMENT CO

Box 3091, SE-169 03 Solna
Phone +46 8 459 84 00
skb.se

SVENSK KÄRNBRÄNSLEHANTERING

ISSN 1402-3091
SKB R-17-12
ID 1668704
February 2018

Methods and workflow for geometric and hydraulic conditioning

DFN-R – status report 2013–2015

Tomas Bym, Jan Hermanson
Golder Associates AB

Keywords: Discrete fracture network (DFN), Conditioning, Modelling.

Abstract

This report summarizes an effort of modelling group run by Golder Associates as a part of DFN-R project. New methods for conditioning Discrete Fracture Network (DFN) models are presented and demonstrated. By applying presented methods it is possible to create DFN models that honour locally observed data while still maintain global model stochasticity. Methods are divided into three main groups: geometric, connectivity and hydraulic conditioning. Geometric conditioning allows modifying a DFN model in such way that it honours observed tunnel traces and borehole intersections. Hydraulic conditioning together with connectivity conditioning assure that conditioned DFN models reproduce observed inflow to borehole intervals and tunnels. All methods have been tested and validated using data from synthetic reality DFN model.

Sammanfattning

Denna rapport sammanfattar insatsen som har utförts av modelleringsgruppen på Golder Associates som en del av DFN-R-projektet. I rapporten presenteras och demonstreras nya metoder för konditionering av diskreta spricknätverksmodeller (DFN-modell). Genom tillämpning av presenterade metoder är det möjligt att skapa DFN-modeller som lokalt överensstämmer med observerade data och samtidigt uppfylla den globala stokastiska modellen. Metoderna är indelade i tre huvudgrupper: geometrisk, konnektivitet och hydraulisk konditionering. Geometrisk konditionering möjliggör modifiering av DFN-modellen för att överenstämma med identifierade/karterade sprickor i tunnlar och borrhål. Hydraulisk konditionering tillsammans med konnektivitetskonditionering garanterar att konditionerade DFN-modeller reproducerar observerade flöden till respektive borrhålsintervall och tunnelsträcka. Alla metoder har testats och validerats med syntetisk data i en DFN-modell.

Contents

1 Introduction

Most Discrete Fracture Network (DFN) models are intrinsically stochastic. The geometric and hydraulic properties are based on statistical distributions evaluated from direct measurements of heterogeneous geometric data from outcrops, boreholes and underground excavations as well as from indirect interpretations from geophysics and hydraulic tests. In the DFN-R project, a single realization of a specified DFN model is used to develop and scrutinize novel methods for local geometric and hydraulic conditioning of unconditional stochastic DFN realizations. The objective of the DFN-R project is to support the development of DFN models that honor local measurements acquired during the construction and subsurface investigations of underground excavations, e.g. during the construction of a geological repository for spent nuclear fuel.

1.1 Background and objectives

The DFN-R project has been in progress since 2013 where two modelling teams from Golder Associates (Golder) and Amec Foster Wheeler (Amec) have been assigned by SKB to develop new methodology for DFN modelling. This report aims to summarize the progress made by the Golder Associates modelling team over the project period from 2013 to 2015.

The main phases of the developments are specified below:

In phase *DFN-R 2013* methods were developed by Amec and Golder to condition stochastic discrete fracture network (DFN) models on fracture trace data on tunnel or deposition hole walls and in boreholes. The set-up of the conditioning process considered a single tunnel, deposition hole, or borehole at a time.

In phase *DFN-R 2014*, the main objective of the geometric conditioning was to extend the analysis to deal with multiple conditioning objects, i.e. several tunnels, deposition holes or boreholes in the same conditioning step and in any combination in a sampling domain.

In phase *DFN-R 2015*, the focus was to include also flow and hydraulic properties in the conditioning process. Presented conditioning methods was to honor not only the geometric data but also static hydraulic test data.

The conditioning methodology has been developed from a theoretical basis but has been tested against synthetic fracture networks. Single realization (referred to as HypoSite) has been used as synthetic reality for testing and validating presented methods.

1.2 Conditioning methods and workflow

Following sections discuss and demonstrate new methods for conditioning to both geometric, hydraulic and connectivity data. Geometric method demonstrated in phase DFN-R 2014 have undergone some major modifications and improvements while some new methods related to hydraulic data have been developed. Following sections describe the three major phases that have been carried out in the conditioning workflow: Geometric conditioning, Connectivity conditioning and Hydraulic conditioning. The last part of this report illustrates the capability of the presented conditioning workflow applied to HypoSite synthetic model.

2 Geometric conditioning

In the previous two years, we presented two different approaches for geometric conditioning. The first method referred to as the Fracture Generation Method, uses the trace/borehole intersections geometries for defining fracture size and orientation. Based on this information new fractures are created to match single or multiple traces on a tunnel wall. Once the fractures are created for all sampled traces the surrounding stochastic DFN model is modified. All stochastic fractures intersecting the sampling object are removed and then replaced by fractures generated by the Fracture Generation Method. While this workflow honors geometric observations on the sampling object further care is needed for obtaining statistically correct connectivity between the stochastic and generated fracture. For this reason, it was decided not to pursue the Fracture Generation Method approach further.

Instead, a Fracture Replacement Method was developed that generates stochastic fractures that are rearranged to fit geometric observations. Modifications and improvements of this method are presented in next section.

2.1 Fracture replacement method

The Fracture Replacement Method compares fractures in a generated stochastic DFN model with geometric observations on multiple tunnels and boreholes and rearranges these fractures to enable an optimal geometric fit while maintaining various statistical measures in the model. The method workflow can be split into different phases as described below.

The Search Method is used to identify if traces from multiple sampling objects can be connected by a single fracture plane. Once this information is processed, the Replacement Method tries to identify a fracture in the stochastically generated model which could match the specified traces. If the selected fracture is already intersecting a sampling object this fracture is simply moved to the new position. In case a selected fracture is not intersecting a sampling object it is moved to match the trace and the original fracture position is replaced by a randomly selected intersecting fracture. This minimizes the potential local change in fracture intensity and keeps number of intersecting fractures per object constant. In case that the number of intersecting fractures is less than number of traces it is not possible to replace all fractures. In a well-calibrated model this should however affect only a small number of fractures. There is also a possibility that a matching fracture is not found in a model. In such case a trace or traces remain unmatched. Number of unmatched traces is reported in a summary report generated at the end of each conditioning simulation.

2.2 Search method

The Search method is used to identify traces from multiple sampling objects that could possibly be connected by a single fracture plane. The method is able to test if traces and borehole intersections from multiple sampling objects could be matched by a single fracture plane. Real fractures are typically not perfectly planar but can exhibit a multitude of geometric behaviors such as small-scale roughness variations, larger scale undulations, curvatures or step-overs etc. It is relatively common that fractures observed in Laxemar and Forsmark can be approximated as planar or undulating over distances at the scale of meters to tens of meters as for example reported by (Hermanson, et al. 2008).

Figure 2-1 shows an illustration of a non-planar fracture intersecting three sampling tunnels. The search method analyses sampling domain intersections and identifies potential fractures candidates in a sequential process as described in the following sub-sections.

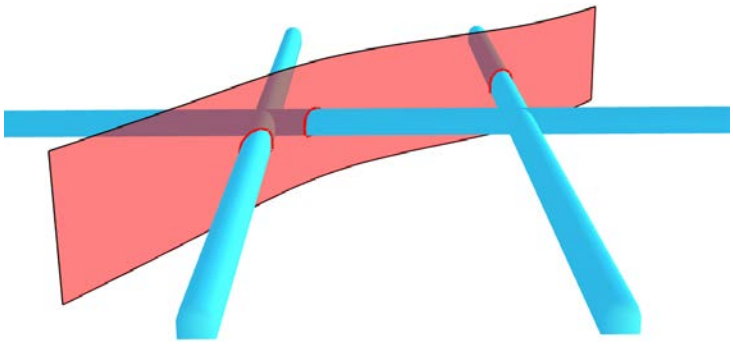

Figure 2-1. *Illustration of three tunnels interpreted to be intersected by a non-planar fracture.*

2.2.1 Initial trace identification

The method randomly selects a trace/borehole intersection from the input trace and intersection data. Only traces fully intersecting the sampling object are considered for further analysis. The assumptions are that larger fractures are more important with regards to connectivity and flow with the assumption that there is a lower probability that small fractures would intersect multiple objects as well as that large fractures have a lower probability to partially intersect a sampling object.

2.2.2 Trace plane estimation

Initially, a plane is calculated for each trace/borehole intersection. In case of borehole intersection a plane is directly calculated from the given dip and strike. The situation can be more complex for tunnel trace data as approximations to the measured trace are made with a single plane. The approach is to match a plane to all trace points with a minimum error using orthogonal linear regression. The results of a point set approximation using orthogonal linear regression can be seen in Figure 2-2.

2.2.3 Search criteria

Once a plane for a trace/borehole intersection is calculated the method searches through all available traces/intersections and tries to identify which could make a reasonable fit with a planar fracture. The method is using three criteria to identify a matching trace: the Alpha angle, Beta angle and a Property value (i.e. a defined geoscientific property attributed to the trace). The user has the ability to control to which level of detail the search method will match these properties. The presented example below is based on the intersections of a single non-planar fracture as illustrated in Figure 2-1.

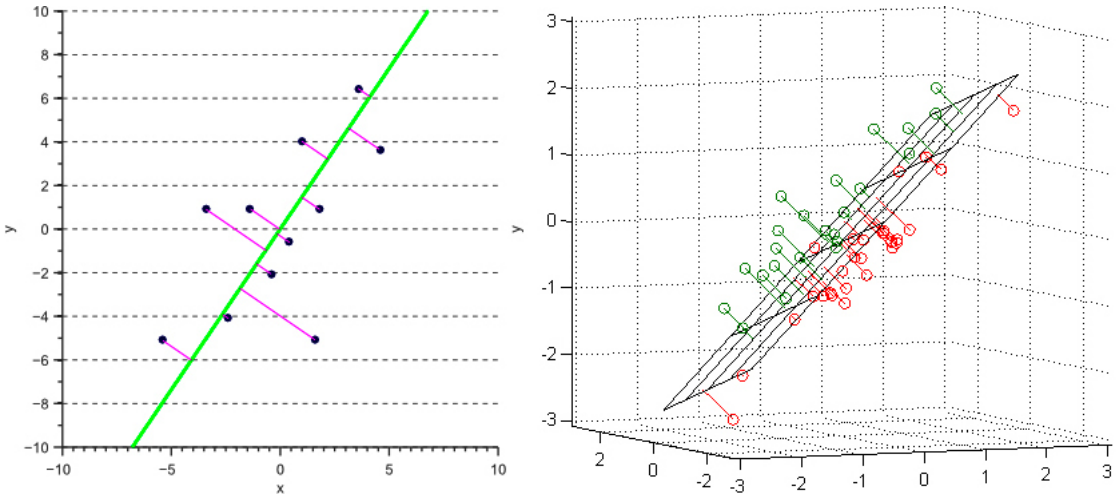

Figure 2-2. *Two examples of approximating a single plane to a set of points by using orthogonal linear regression in 2D and 3D.*

The alpha angle can be also referred to as an initial screening window. It defines a volume around the initial plane within which the process is searching for other traces/borehole intersections. Using a small alpha angle results in smaller screening window, cf. Figure 2-3. It should be noted that in the HypoSite model all traces were formed by planar fractures so the alpha angle can be set to very small angle to identify multiple sampling domain intersections.

Once the traces fulfilling the alpha angle search criteria are identified the orientations of such traces are compared. For each trace a plane and its normal vector is calculated using the method described in Section 2.2.2 and the angle between the initial trace normal and the calculated normal is computed, cf. Figure 2-4. In the method Beta angle refers to the maximum angle between the normal of the individual planes which are considered to be part of the same fracture.

The search criteria can also account for other geoscientific properties to connect traces/borehole intersections by a single fracture plane. A property can be defined based on geological observations such as specific mineralization, fracture roughness etc or from tracer tests or other hydraulic observations such as transmissivity. As an illustration of how properties can complement the search method Figure 2-5 shows a set of traces with different properties which all fulfil the search criteria using Alpha and Beta angles. However only a few traces contain a property considered to be a unique identifier for connected traces (green color in our example). In this case the search method would only identify the green traces to be connected by a fracture.

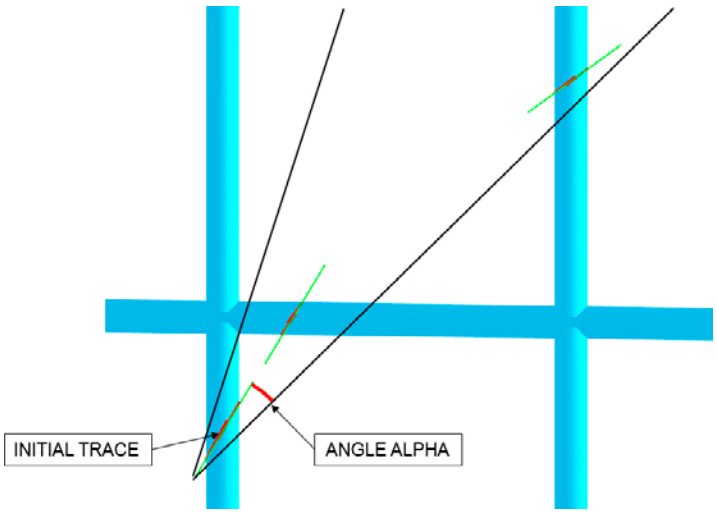

Figure 2-3. *Alpha angle defining a screening window for the "Search Process"*

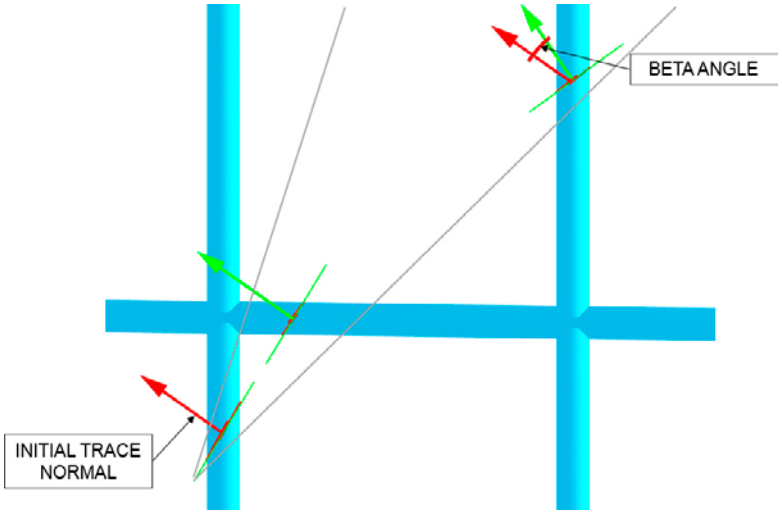

Figure 2-4. *Angle between the initial trace normal and another trace normal is referred to as a Beta angle.*

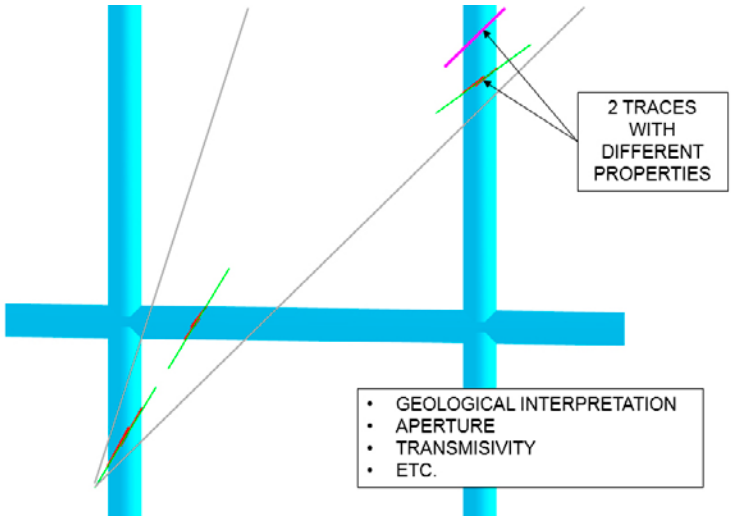

Figure 2-5. *Figure illustrating traces with different properties. Only traces marked in green would be considered by the Search Method.*

All three parameters affect the results of the search method. For the method to yield reasonable results it is important that the search windows are calibrated using appropriate site specific knowledge. It is also important to have a clear understanding of the purpose of the DFN model – is this an application where exact intersections should be reproduced in the model or is this an application more focused on the behavior of the fracture network (connectivity, hydraulic or geomechanical properties) as illustrated in Figure 2-6?

Figure 2-7 and Figure 2-8 illustrate the results of varying the search parameters using the same intersection data. An interference test between borehole A and B have been conducted and the transmissivity results were used as the fracture property parameter (Figure 2-6). The input data consists of intersection points and their corresponding property. Grey traces illustrate how reality may look. A dashed red line in the left side of Figure 2-7 represents the two intersections identified by the search method without constraining the output with fracture property controls. The right side of Figure 2-7 illustrates the output if additional fracture properties are used to constrain the results further. In both cases low alpha and beta angles are used.

If high alpha and beta angles are used, as illustrated in Figure 2-8, fracture connections are possible in many directions (left side) whereas connections with matching properties can be refined by adding fracture property constraints (right side).

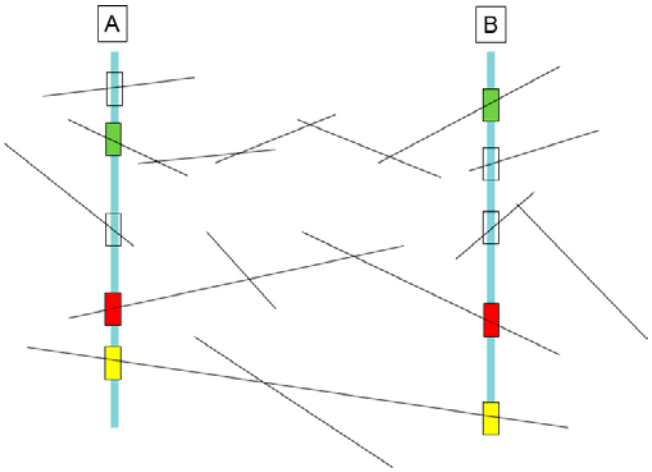

Figure 2-6. *Two parallel boreholes with fracture intersections. Property of each intersecting fracture is based on the connectivity test results.*

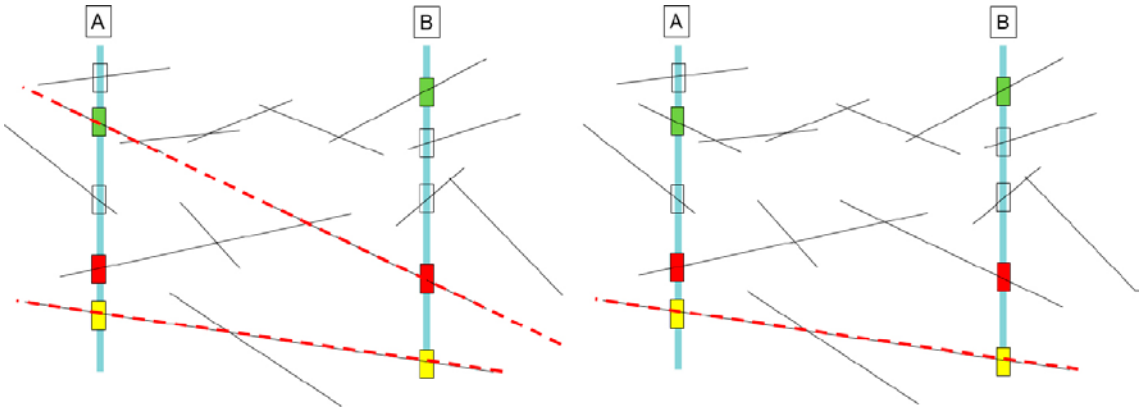

Figure 2-7. *Results of the Search Method using LOW alpha angle and LOW beta angle. Figure on left shows result without considering the property match parameters while the figure on right is using the property match.*

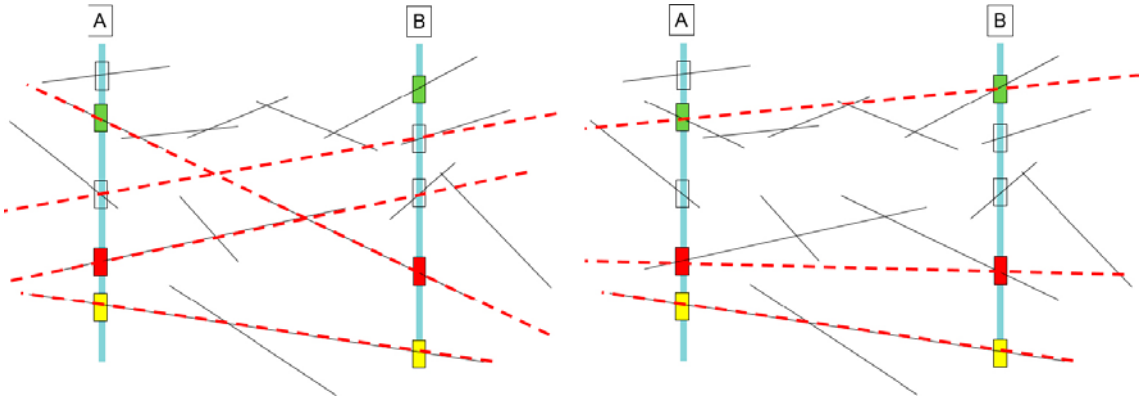

Figure 2-8. *Results of the Search Method using HIGH alpha angle and HIGH beta angle. Figure on left shows result without considering the property match parameters while the figure on right is using the property match.*

2.2.4 Calculation of a resulting plane

The last step of the search method is to define the geometric properties of a plane which fits all identified intersections. Due to software limitations, fractures can only be generated as planar features. If the identified traces/intersections correspond to a planar fracture the resulting plane would be exactly the same as the original fracture. If the traces/intersections deviate from a plane the orthogonal linear regression method is used to calculate the resulting plane (Figure 2-9). All following steps of the method (such as defining size limits and fracture position) are performed in 2D where all objects are projected to this resulting plane.

2.2.5 Defining fracture size limits

Fracture size is estimated in the next step of the geometric conditioning workflow. Presented workflow works for any regular (non-elongated) fracture shape, i.e. fractures can be represented as a 4-node polygon (rectangle) up to n-node polygon (pseudo circle).

A fracture must be large enough to extend beyond all identified trace/intersection points. At the same time if a fracture is too large it can intersect sampling objects where no traces/intersections are observed. Also, the individual size of a fracture should exist within the same statistical range as the total population of fractures in the model. Obviously there exist an unlimited number of combinations that can answer this question, however the presented workflow tries to address the most common geometrical combinations while retaining the statistics of the unconditioned set.

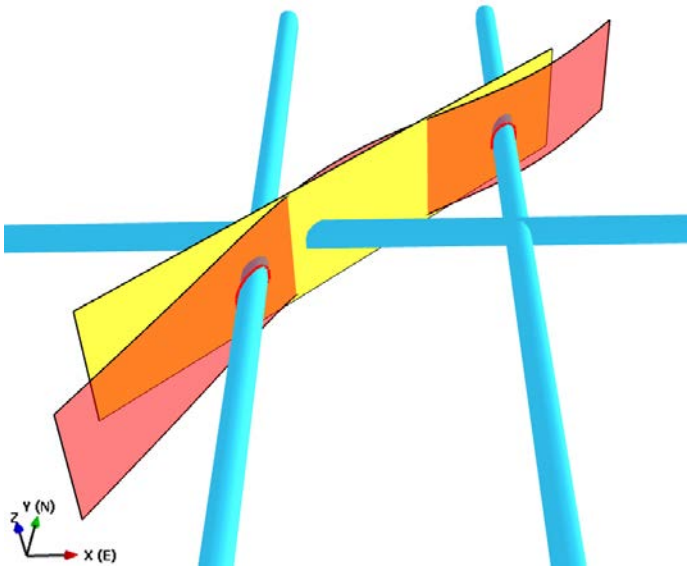

Figure 2-9. *Approximation of the original curved fracture surface (orange) by a single planar surface (yellow)*

For simplification all fractures are considered to be circular. The size of a fracture is estimated in terms of its equivalent circular radius. For a circular fracture *EqR* corresponds to the distance between the fracture center and its vertices (for a circular fracture with unlimited vertices *EqR* corresponds to the exact distance of the radius while with decreasing number of fracture vertices the error increases up to 88 % ($\sqrt{\pi}/4$) of the exact distance).

$$EqR = \sqrt{\frac{A}{\pi}}$$

The simplification is used only in search method. When a fracture is placed to a specific location the actual fracture shape is considered. It should also be noted that this section only describes how the limits R_{min} and R_{max} are calculated. The placement of the actual fracture is described in further sections.

Minimum fracture size (R_{min})

The minimum fracture size is defined by the largest distance between the trace points of a single trace, or the largest distance between multiple traces/intersection points. Figure 2-10 illustrates the example how R_{min} is defined for single and multiple traces.

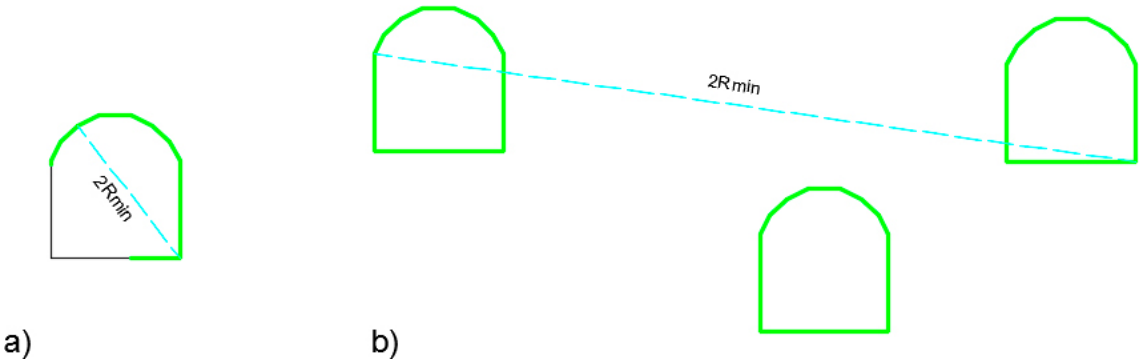

Figure 2-10. *Illustration of the estimation of the minimum radius required to place a fracture.*

Maximum fracture size (R_{max})

Calculating the maximum fracture size is complex as it depends not only on the trace/intersection data but also on their position relative to other sampling objects. Usually a trace is observed only on sampling objects intersecting a plane defined by the search method. The information on which sampling objects a trace appears can help us to estimate the constrains on the maximum fracture size.

The following sections illustrates typical geometrical cases and explains the workflow of the method.

The method first starts by transforming 3D traces into a 2D plane calculated by the search method. This simplifies all geometric computations as well as visualizations shown in this report. In Figure 2-11 below observed traces on the sampling objects are shown in green. Sampling objects without any traces are shown in red and are referenced as No-Trace objects in the text. This initial part of the method only estimates the possible fracture size limits. The latter part of the workflow (Section 2.2.7) tests for avoiding intersections with No-Trace objects (i.e. sampling objects without traces). In some parts of the process geometry of the traces is simplified to a single point. This could potentially lead to over- or under-estimating R_{max} however when a specific fracture is chosen all trace points are checked against fracture geometry, i.e. method makes sure that all trace vertices are connected to the fracture and that no intersections with No-Trace objects exist.

Single trace with two or less no-trace objects

In this case there is no upper limit on fracture size. For a single trace constrained by less than two No-Trace objects there is degrees of freedom to fit a fracture of any size.

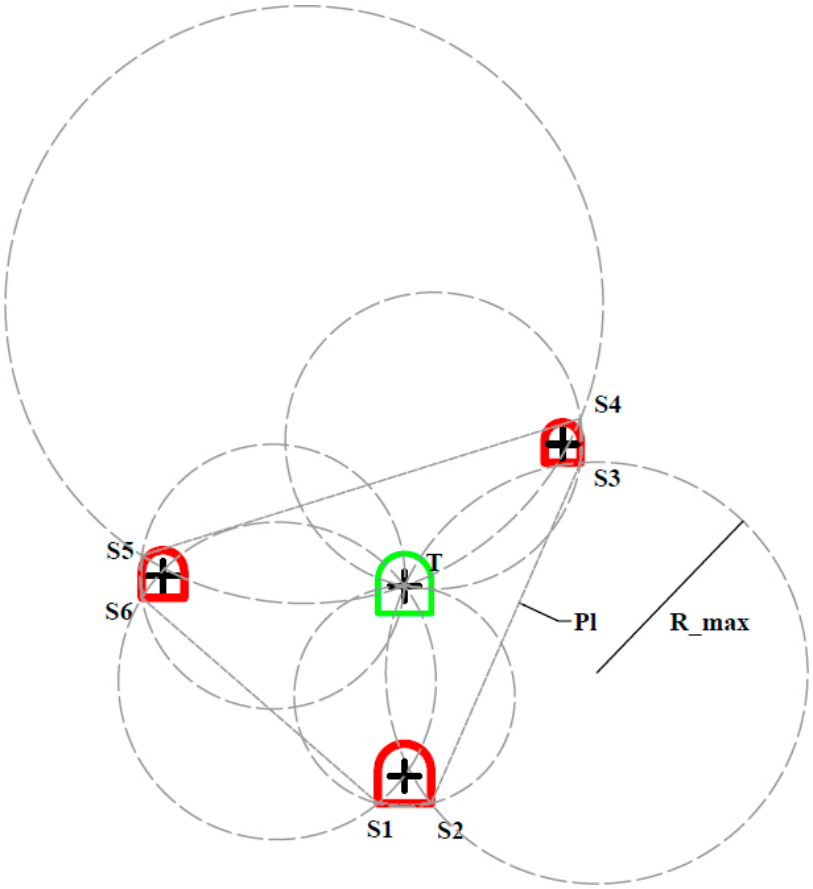

Figure 2-11. *Using No-Trace polygon Pl to find Rmax. The largest circle is not considered as it includes one No-Trace center point.*

Single trace with three or more no-trace objects

If more than two No-Trace objects are identified along a fracture plane the maximum fracture size depends on the relative position of the trace and the No-Trace objects. For computational efficiency trace and No-Trace objects are represented by a single point. The method workflow is as followed:

a) Create a polygon Pl formed by all No-Trace points

b) Check if trace points are inside polygon Pl
   a. If NOT: no upper limit on fracture size
   b. If YES: continue to c)

c) Iterate through all segments of polygon Pl
   a. Create a circle formed by trace point (T1) and segment points (Sn and Sn+1)
   b. Check if No-Trace center points are inside circle
      i. If NOT: save radius of the circle

d) Set the fracture upper size limit as the maximum of the saved radii

Multiple traces with one and more no-trace objects

A case with multiple trace objects and multiple No-Trace objects can be handled in a similar way as the previous case. It is necessary that a correct limit for fracture size is set such that all trace objects are encompassed without creating an intersection with any No-Trace objects. In case the fracture size limit is set incorrectly and it is not possible to condition selected fractures, the Replacement method (Section 2.2.7) would recognize the case and would remove one of the multiple trace objects. The whole workflow for estimating fracture size limits is iterated until a possible fit is achieved. In the tests, a geometric fit was achieved in approximately 95 % of the cases with multiple Trace objects and multiple No-Trace objects.

For simplification the method uses a single point representation of both trace objects and no-trace objects. These points are then projected to a plane where all calculations are done in 2D. The workflow is as follows:

a) Create a polygon Pl formed by all trace points

b) Check if any no-trace point is inside polygon Pl
   a. If NOT: create a polygon Pl2 formed by all no-trace points
      i. Check if any trace point is inside polygon Pl2
         1. If NOT: no upper limit on fracture size
         2. If YES: Iterate through all trace points (T) that are inside Pl2:
            a. Iterate through all segments of polygon Pl2 (S1 and S2)
               i. Create a circle formed by T1, S1 and S2
               ii. Check if all trace points are inside circle and all no-trace points are outside
                  1. If YES: save radius of the circle
            b. Set the fracture upper size limit as the maximum of the saved radii
   b. If YES: Remove furthest trace from the initial trace and rerun algorithm for fracture size limits

Fracture size distribution

In case a fracture for replacement is not intersecting a sampling object it is necessary for find a fracture which would fit the size distribution of intersected fractures. Presented workflow aims to minimize effects of the fracture conditioning method that may cause bias to the size distribution of the fracture population.

All fractures that intersect the sampling objects are identified and their fracture size distribution parameters are calculated. This process is executed independently for each sampling object (tunnel/borehole) to handle orientation sampling bias. This requires the user to specify a parent size distribution type. The distribution parameters are calculated only once at the beginning of conditioning process.

Secondly, the user is required to define a distribution fit ratio (*tol* in Figure 2-12). The smaller the ratio, the better the distribution is preserved. However, a too narrow ratio may lead to an inability of the search method to find a correct fracture. Values between 0.1–0.2 worked well for all tested examples. The workflow is as follows:

c) Calculate the size distribution parameters (based on a user defined distribution type).

d) Generate a random sample (R_0) from the distribution which falls within the limits defined in previous sections (R_{min} and R_{max}).

e) Use the distribution fit ratio to calculate the optimal lower (R_L) and upper (R_U) bound for the fracture size.

f) Check that the optimal bounds are within the limits of R_{min} and R_{max}.

R_0 is a random sample from the defined distribution. Since it is not possible to find a fracture with exact same radius we introduce an upper and lower bound limit for our search criteria based on fit ratio. With reasonably small fit ratio a fracture with radius within the upper and lower bound limits would still fit the required distribution.

2.2.6 Fracture search method

The purpose of the Fracture Search Method is to identify a fracture within the generated fracture population which will be used to fit the observed traces. The method searches through the unconditioned fracture sets and tries to identify a fracture with a similar orientation and size that fits within the optimal fracture size bounds. User specifies the maximum angle by which a fracture is allowed to rotate to fit particular trace/traces. This maximum allowable rotation is used as search criteria. Only fractures with angle between their normal and trace normal less than the maximum angle are considered. The method first searches through fractures intersecting the sampling object(s) and only R_{min} and R_{max} are considered as size limits. If no suitable fracture is found, the method then continues searching through all stochastic fractures using the size optimal limit bounds R_L and R_U, as described above. The method will identify the closest matching fracture that will then be used to fit the identified intersections. If a no-intersecting fracture is selected its original position is replaced by a randomly selected fracture that intersects the sampling object.

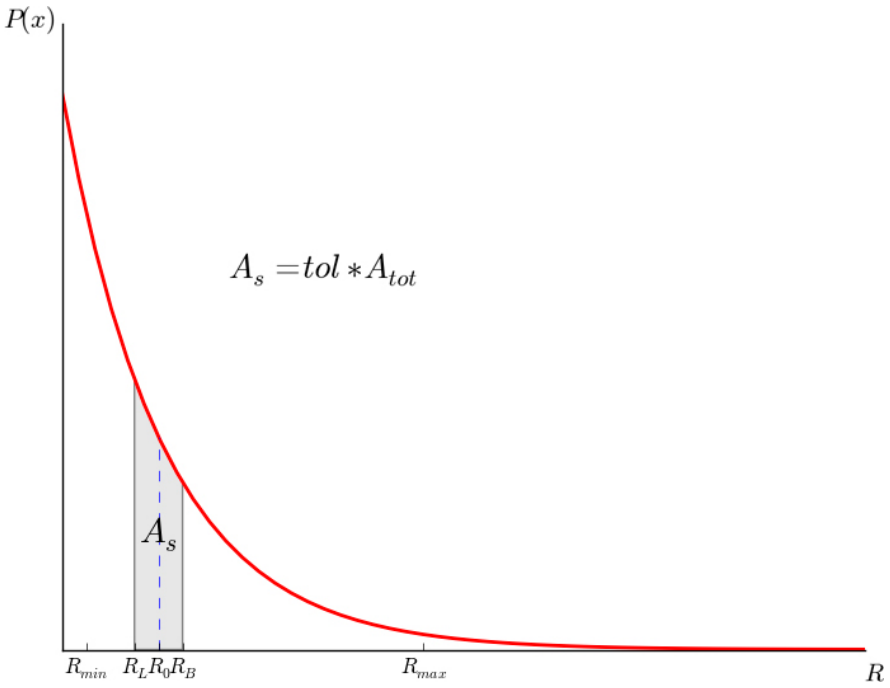

Figure 2-12. *Plot illustrating the method for estimating lower and upper bounds of the fracture size. A_{tot} represents a total area below distribution line and "tol" a distribution fit ratio. From a random sample R_0 and area A_s it is possible to calculate R_L and R_B.*

At the end of the conditioning workflow all fractures that intersect the sampling objects in locations not matching observed traces will be removed from the model. For a good calibrated DFN model, the number of intersecting fractures should statistically match the number of observed traces so that the number of removed fractures should be few. A requirement to remove a large number of fractures is an indication of a poorly executed calibration of the input statistical distributions to the DFN model. This should lead to a reassessment of how data was interpreted and a recalibration of the input parameters.

2.2.7 Replacement method

The Replacement Method computes the required transformation (translation and rotation) for a fracture to fit specified intersection(s) and to avoid no-trace intersections. This method works both for full perimeter intersections (FPI), partial intersections as well as for multiple sampling domain intersections. It also incorporates a degree of stochasticity in terms of fracture location to ensure that repeated results from the conditioning workflow differs in each realization (Figure 2-13). Different approaches for fracture fitting are considered for FPI and partially intersecting traces as described below. Once a pseudo random position is proposed a fracture is tested against no-intersection traces using Ray-casting algorithm (Figure 2-16). If it doesn't fulfill the no-intersection criteria a new random position is proposed. User can specify maximum limit of tries before a new fracture is selected. Alternatively after certain number of tries a similar approach as presented in Section 2.2.5 can be used to find fracture position. (This option is not currently implemented in testing code)

Partially intersecting trace

For a partially intersecting trace the method is to randomly decide between two options how to fit a fracture to a trace. The first option, termed "by side", can be used only if any of the sides of the fracture is larger than the trace length. In such case a fracture is rotated so that single fracture side connect the first and last vertices of the trace (left side of Figure 2-14). The second option, termed "by vertex", first places a random fracture vertex between the start and end trace indices and then rotates a fracture to get the exact fit (right side of Figure 2-14). The different results of the two methods are illustrated in Figure 2-18.

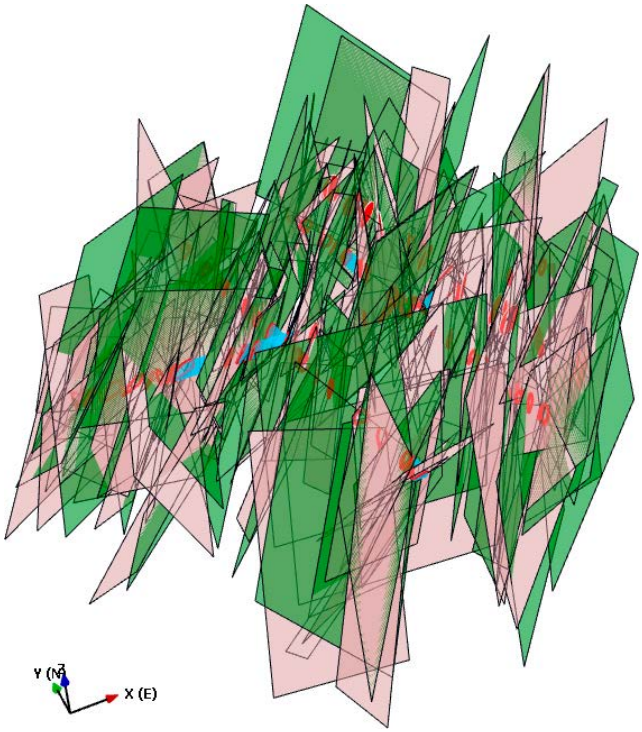

Figure 2-13. *Two conditioned DFN realizations showing the stochasticity of the conditioning method. The green and pink fractures illustrates conditioning results from two separate realizations.*

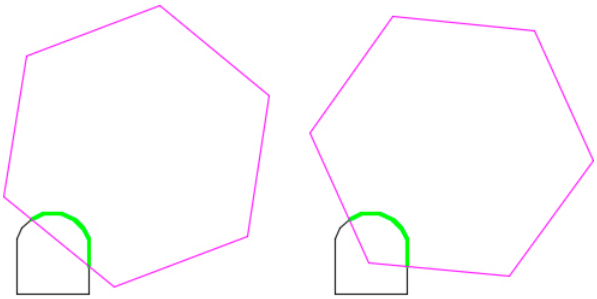

Figure 2-14. *Illustration of the two methods of fitting traces that partially intersect a sampling domain. The "by side" method (left) and the "by vertex" method (right.*

FPI trace

For FPI traces first a space of all possible relative positions are defined and next a poissonian method is applied to select a position within that space. For simplification, the space is defined as an inscribed circle within the vertices of the fracture. A smaller circle that includes all trace points is inserted. The pseudo random fracture position fulfills the criteria that the smaller trace circle is always fully inside the larger inscribed circle (Figure 2-15). Next, a Ray Casting Algorithm is used to control that all trace points are inside of the fracture polygon to eliminate the potential error caused by the circle simplification (Figure 2-16).

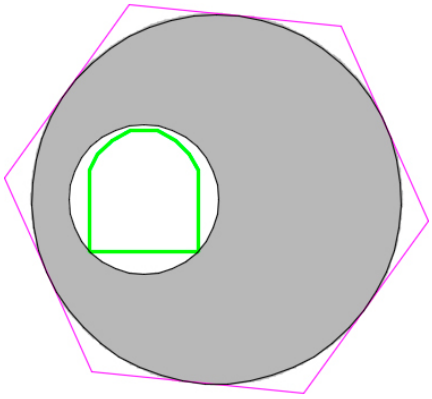

Figure 2-15. *Poissonian method for fitting a fracture to a FPI trace. The shaded region represent a space for a smaller circumscribed circle to all trace vertices.*

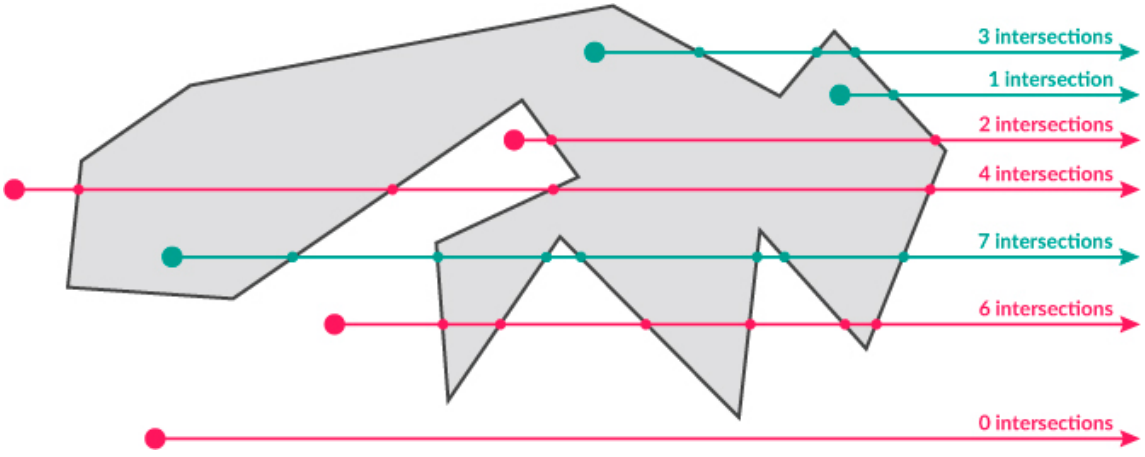

Figure 2-16. *Ray Casting Algorithm. For each tested point a horizontal ray is created and the number of intersections with the polygon is calculated. A point is inside a polygon if the number of intersections is an odd number and outside if it is even.*

Multiple traces

For multiple traces a similar approach as for single FPI-trace is adopted. A circle circumscribing all trace vertices is created and fracture surface is also simplified to a circle. Pseudo-random location is chosen in such way that the smaller circles lies within the larger fracture circle (Figure 2-17). All trace vertices are checked by the Ray Casting Algorithm (Figure 2-16).

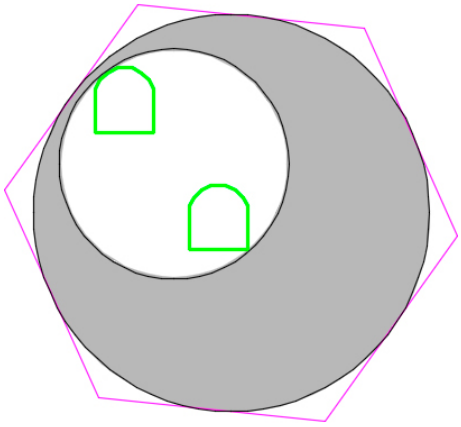

Figure 2-17. *Poissonian method for fitting a fracture to multiple traces. Shaded region represent a possible space for a smaller circumscribed circle to all trace vertices.*

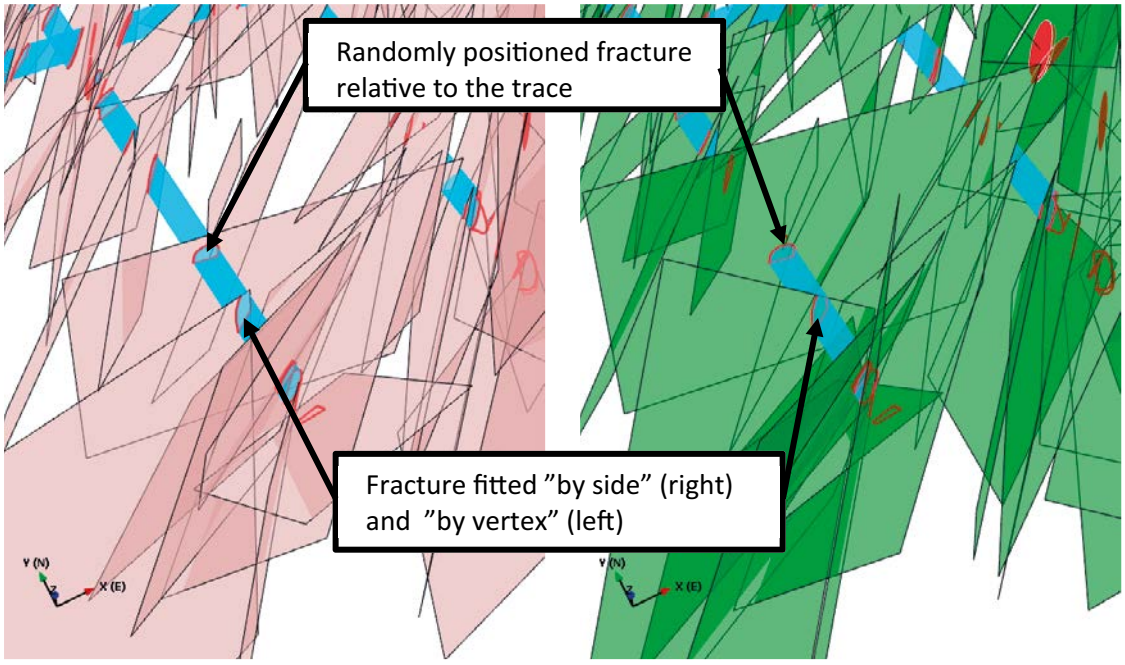

Figure 2-18. *Detail from Figure 2-13 illustrating the example of fitting partially intersecting trace using “by side” and “by vertex” options. For FPI trace a random relative position of a fracture is shown.*

3 Connectivity conditioning

Connectivity Conditioning can be seen as a link between geometric and hydraulic conditioning. When conditioning the model to match hydraulic data it is necessary to ensure that each flowing trace/borehole intersection is connected to a hydraulic boundary. The presented workflow below ensures that at least one connection path exists between a defined inflow point (trace or borehole intersection) and a hydraulic boundary. The workflow is illustrated through a simple case with a tunnel and single surface representing a boundary (Figure 3-1).

3.1 Connectivity method workflow

The first step is to identify traces/borehole intersections that exhibit flow. The Connectivity Method considers only these traces and makes sure connectivity to the boundary exists at these locations. Traces/borehole intersections without any inflow data are processed only through the Geometric conditioning workflow (see Chapter 2). However, it is possible that a geometrically placed fracture will create a connection with a hydraulic boundary and such a trace could potentially exhibit inflow during a hydraulic simulation.

Next step is to identify hydraulic boundary objects. These can be any objects in the model and depends on the aim of the simulation.

Once all hydraulic boundary objects are defined a pathways analysis is conducted on all unconditioned DFN fractures. Based on the results from the pathways analysis a property termed “Connectivity level” (Cx) is assigned to all unconditioned fractures. Fractures that directly connect between a boundary and the sampling object are assigned a connectivity level value of zero (Figure 3-2, left). Fractures that are connected through one additional fracture to the boundary are assigned a connectivity level of one (Figure 3-2, center). Thus, the connectivity level identifies the minimal number of connections between a fracture and a boundary in the model (Figure 3-2, right).

Calculating Connectivity Level for all fractures in a DFN model is computationally extensive but is only required to be run once during the whole conditioning process.

The method then iterates through all traces/borehole intersections that exhibits inflow. If an inflow point is identified the workflow proceeds as follows:

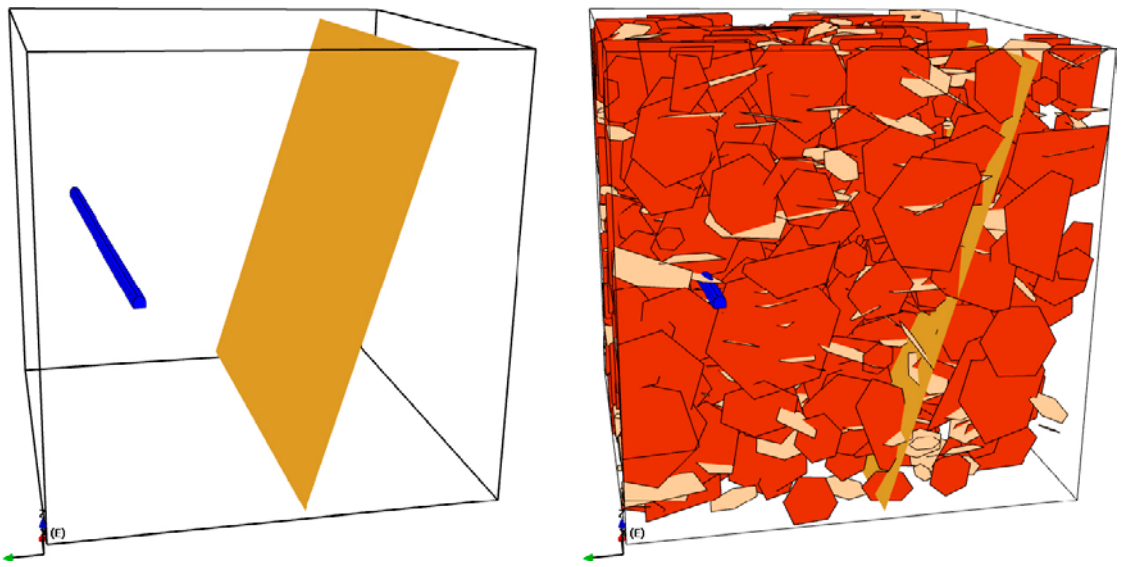

Figure 3-1. *Simple DFN model with a horizontal tunnel (blue) and surface representing a flow boundary (brown) illustrating the Connectivity Method.*

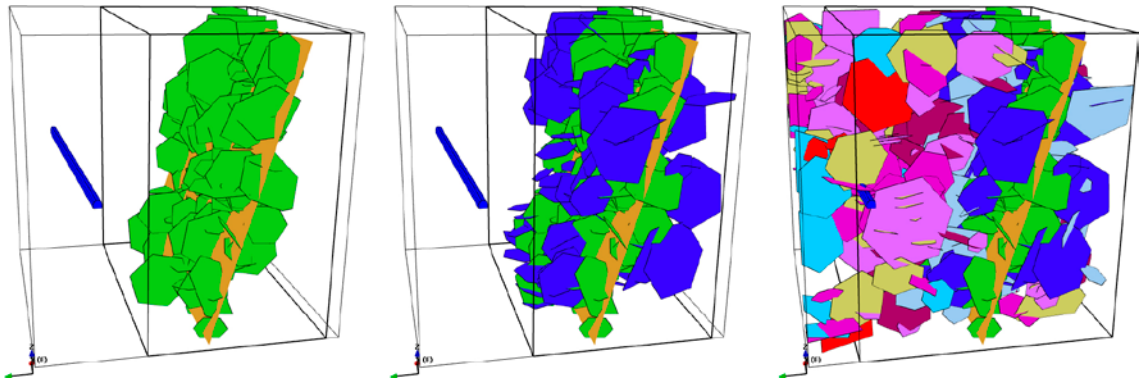

Figure 3-2. *Illustration of the connectivity level property that is assigned to each unconditioned fracture in the model. Green color illustrates connectivity level zero (left), blue connectivity level one (center), all connectivity levels to the boundary (right).*

For calculating a tracemap of all stochastic fractures a temporary plane with the same orientation and position as the trace is created (Figure 3-3, left). This is done using the orthogonal linear regression method as described in Section 2.2.2. Next the intersections between the temporary plane and the stochastic DFN fractures are computed by creating a tracemap on the temporary plane. Each trace contains information of the fracture connectivity level (Figure 3-4).

The method then identifies the trace on the temporary plane that have the optimal connectivity level. This is done by assigning a maximum limit (Cx_max) of the connectivity level. This limit is a user-input value and should be based on an understanding of the test data, the geology and experiences from previous tests performed on the site. The connectivity between the sampling object (tunnel) and boreholes and the hydraulic boundaries is directly affected by the chosen value of Cx_max. It is also possible to specify Cx_max for each trace separately so that traces with higher inflow have lower Cx_max values, thus indicating a faster path to the boundary.

Based on the Cx_max value the method selects the closest trace with Cx < = Cx_max. In theory any trace with Cx smaller that Cx_max can be chosen but it most cases it is easier to match the closest to current sampling object. This trace is then sent to the Geometric Conditioning Method and is treated as an additional trace that should be matched geometric conditioning (Chapter 2). If the trace is matched the conditioned fracture becomes connected to specified boundaries with at least Cx+1 connections.

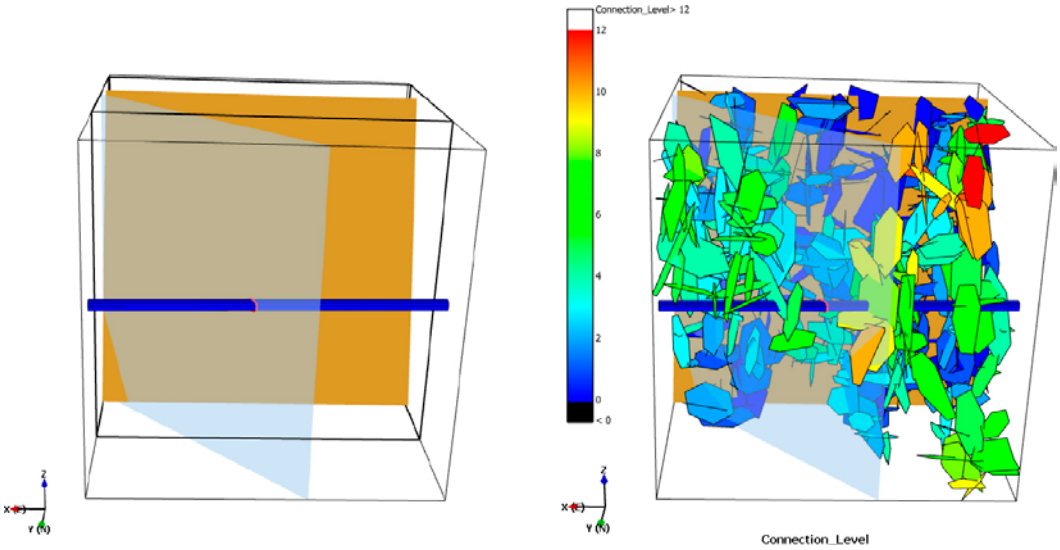

Figure 3-3. *Illustration of the temporary plane of a trace location (left) and the connected fractures between the boundary and the plane (right).*

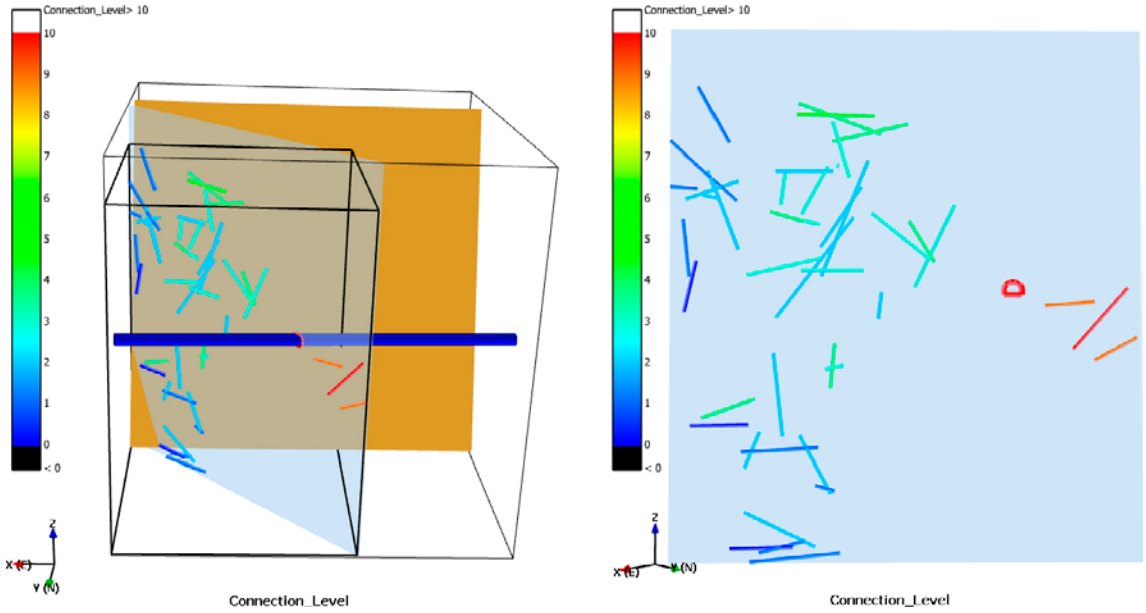

Figure 3-4. *Illustration of fracture traces and their corresponding connectivity level (coloring) intersecting the temporary plane. Round red trace represents the intersection trace in the example tunnel.*

4 Hydraulic conditioning

The geometric conditioning considers only observed traces and borehole intersection geometries to condition the model whereas the hydraulic conditioning uses the inflow information for conditioning the DFN model. Once the geometric connectivity method, described above, is completed and there is adequate connection between the observed inflow locations and the flow boundaries, the hydraulic conditioning workflow tries to match the observed inflow by calibrating fracture hydraulic properties.

Hydraulic conditioning method is modifying each individual fracture transmissivity using the following equation,

$$y = A + Bx^c + De^{(Ex)} \qquad \text{Equation 4-1}$$

where x is mostly taken as fracture equivalent radius. It is possible to use all 5 parameters (A–E) or just some of them. If for example only parameter A is used there is no correlation between fracture size and transmissivity. In all our demonstration cases we are using a relation defined as $y = Bx^C$.

In certain DFN modeling approaches a relation between size-transmissivity is defined and should not be modified. In such case it is not possible to use hydraulic conditioning method as described in this section. Instead it is suggested to run multiple realizations and apply geometric and connectivity conditioning methods and compare flow results with observed values. If after running statistically representative number of simulations a match with the observed values is not found it is necessary to change other model parameters such as fracture size or intensity.

The hydraulic conditioning workflow is more complex than the geometric or connectivity workflows presented above. It requires an iterative method of running hydraulic simulations, analyzing the results, adjusting DFN hydraulic properties and re-running the hydraulic simulations to obtain a hydraulically conditioned DFN model. A schematic workflow for this method is presented in Figure 4-1. Each step in the presented workflow is discussed in following sections.

4.1 Observation data

The term Observation Data refers to all information that the conditioning workflow needs to consider during the conditioning method. A conditioned DFN model should ideally match all observed data both spatially (geometrically) and in absolute sense (hydraulically).

Flow can be evaluated under different conditions – steady state or transient. So far the DFN-R study has only encompassed steady state flow analysis in the conditioning workflow.

Preparing the observation data requires not only extracting the actual test results but also analyzing all the information on how the tests were conducted. For a steady state flow test it requires having information about in situ conditions at the time of a test and estimating boundary conditions. The presented conditioning method involves all these steps in the workflow.

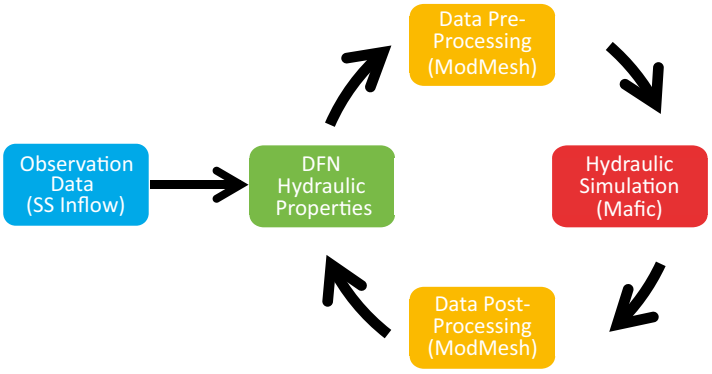

Figure 4-1. *Illustration of the hydraulic conditioning workflow.*

4.2 Adjusting DFN hydraulic properties – PEST

An important part of the conditioning method involves adjusting the DFN properties and analyzing how these changes affect the flow simulation results. In the presented workflow the tool PEST® (www.pesthomepage.org) is utilized but other Parameter Estimation Analyses codes may be possible to use. PEST is considered the industry standard software package for parameter estimation. The code is well documented and due to its popularity across the industry many utility programs that automate and simplify common tasks are available and the software is free to use. PEST is a very powerful package with many features and since only basic PEST estimation modes were used for this demonstration the authors are aware that there might be improved ways how to apply PEST in the conditioning workflow.

PEST requires three input files. A template file is used for generating the input information for the simulation code (in our case the input file for ModMesh). The instruction file contains the information for how to read the simulation results and the control file contains all parameters for the PEST simulation such as observed values, conditioning parameters, estimation method etc.

PEST is trying to optimize input parameters to minimize error between observed and estimated values. The error difference is calculated using the least square method. In our conditioning method we use for boreholes inflow values from each individual borehole intervals; and for tunnels total inflow values all with same weight. If for example somebody wants to emphasize a total inflow to a borehole rather than to individual intervals it is possible to add additional target value as a sum of all intervals and use larger weight than weight for individual intervals.

4.3 Communication layer – ModMesh

ModMesh is a code written for the purpose of demonstrating the conditioning workflow. It serves as a communication layer between PEST and a flow simulator (Mafic). ModMesh is capable of modifying mafic meshes properties by changing the fracture hydraulic properties and by turning on and off specified flow boundaries. It should be noted that mesh geometry remains unchanged and it is only node and element properties that are affected by ModMesh. The input file contains information about which Mafic file should be used in the calculation as well as information about which boundary condition should be applied.

ModMesh is written in Python 2.7 and does not requires any other additional non-standard modules. Since Mafic is a single thread code ModMesh launches multiple instances of Mafic for each available core. This is achieved by implementing a thread-pool technique. First a thread-pool is filled with all commands needed to be performed during one iteration method (in our case each command includes the modification of a mesh properties and the execution of a Mafic simulation on a specified borehole interval). The pool runs the commands on each available core. Once a command is executed and the results are stored, the core continues executing commands until the pool is empty. Results from all commands are compiled together and exported to a single text file which can be read directly by PEST.

The thread-pool technique maximizes the capacity of any workstation while still using a single-thread code. Total time for conditioning workflow is almost linearly reduced by the number of processors.

Description of the ModMesh input file format can be found in Appendix A.

4.4 Demonstration case

A simple DFN model was created to test the hydraulic conditioning method. First the model was used to calculate inflow to specific locations along two horizontal boreholes and next, without changing fracture geometry, a conditioning workflow was applied to estimate fracture transmissivity.

The DFN model used for demonstration contained two fracture sets with the hydraulic properties summarized in Table 4-1. Two horizontal boreholes were introduced to the model and the fracture intersections were analyzed. The boreholes contained five and six fracture intersections respectively, cf. Figure 4-2.

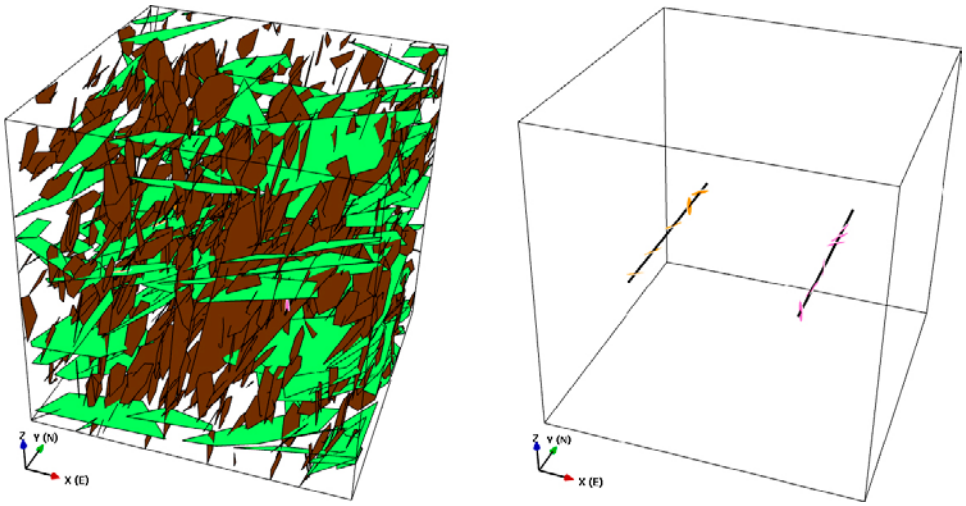

Figure 4-2. Validation DFN model with 2 fracture sets and 2 horizontal boreholes.

Table 4-1. Total inflow to NE main tunnel.

Property	Function
Aperture	0.0001 × EqRadius [m]
Transmissivity	1E–7 × EqRadius$^{0.5}$ [m^2/s]
Storativity	0.0001 [–]

The steady state flow was calculated for each borehole intersection. In the simulation a constant head of 400 m was applied to all six sides of the region box. For each case a tested borehole intersection was assigned a constant head of 0 m while all other intersections had a free head boundary. This approach is identical to the one used in HypoSite model. In total eleven flow simulations were run to obtain the inflow to all intersections along two boreholes. The results are summarized in Table 4-2.

Table 4-2. Inflow to intersections in 2 boreholes in validation model.

Interval	Flux [m^3/s]
BH-1_35.0	3.59E–004
BH-1_65.0	3.30E–004
BH-1_95.0	2.75E–004
BH-1_115.0	3.04E–004
BH-2_5.0	6.95E–004
BH-2_25.0	6.57E–004
BH-2_50.0	4.35E–004
BH-2_80.0	4.02E–004
BH-2_90.0	1.89E–004
BH-2_100.0	1.33E–004

Next the same DFN model with same geometry was used for conditioning. The aim of the conditioning was to adjust fracture transmissivity to match inflow data at the identified borehole intersections. In our setting PEST is comparing inflow values at each location individually using least square method. Results from the analysis, presented in Table 4-2, were used as the target values for the conditioning. Transmissivity was set to follow Equation 4-2 where b and c are the two unknown parameters targeted by our conditioning workflow. Initial values for b and c were set to 1E-5 and 1.0 respectively.

$$T = b \times R^c \qquad \text{Equation 4-2}$$

After eighteen iterations, b and c values were estimated with almost perfect match to values used in the original target model. Results can be seen in Table 4-3. Each iteration contained eleven separate flow simulations as described earlier. Profiting of the multi core architecture of both PEST® and ModMesh the overall time of running 18x11 = 198 realizations was about 30 minutes on an eight-core machine.

Table 4-3. Estimated and original values for parameters b and c in Equation 5-1.

Parameter	Original Value	Estimated Value
b	1.0E–7	7.5112E–8
c	0.5	0.49998

This validation example was used to test the capability of the hydraulic conditioning method using PEST. It was demonstrated that for a model with same geometry the method were able to estimate the original fracture transmissivity with a high level of accuracy.

5 HypoSite model

HypoSite is an acronym for Hypothetical site(s). HypoSite is a collage of controlled datasets representing artificial geological settings. The main purpose of HypoSite is benchmarking codes and concepts, evaluate methodologies and test new implementation. Originally intended to address all aspects of geology, it is in the DFN-R project focused on DFN modeling. Because of the substantial spatial variability in the geometric and hydraulic data used for DFN modeling we need a way

- to compare different DFN tools and concepts; i.e. common test- or benchmark cases, and
- to test new algorithms in our tools.

HypoSite aims to meet these needs by constructing test cases from controlled datasets. Some of the advantages of using HypoSite are:

- Each ”test case” is a deterministic (single) realization of a “DFN recipe”, i.e. a defined set of statistical distributions, rules assumptions, etc.
- We can construct “realistic” data samples from the deterministic single realization by mimicking the data acquisition methods used on outcrops, in boreholes and in underground excavations, and use these samples as input to the DFN tools.
- Depending on the chosen level of complexity we can work in stages starting with simple models. In the end, we need to mimic real world applications.
- Perfectly known geometries and parameters enable us to evaluate performance.

5.1 Reality realization

Figure 5-1 shows a view of the single realization used as “Reality realization” in the DFN-R project. The model domain is a cube of (200 m^3). Geometric and hydraulic data are specified for three sets of fractures, two sub-vertical and one sub-horizontal. The fractures are generated as polygons with 12 nodes, where the location of the fracture centers follows a Cartesian rectangular distribution, i.e. the fracture centers are random and uncorrelated in 3D. Fracture size (radius) follows a power law distribution with 1 m ≤ r ≤ 250 m, and the intensity (the fracture surface area per unit volume of rock) is set to ca. 0.3 m^{-1}. Approximately 47,000 fractures form clusters of fractures that connect to the outer boundaries of the model domain.

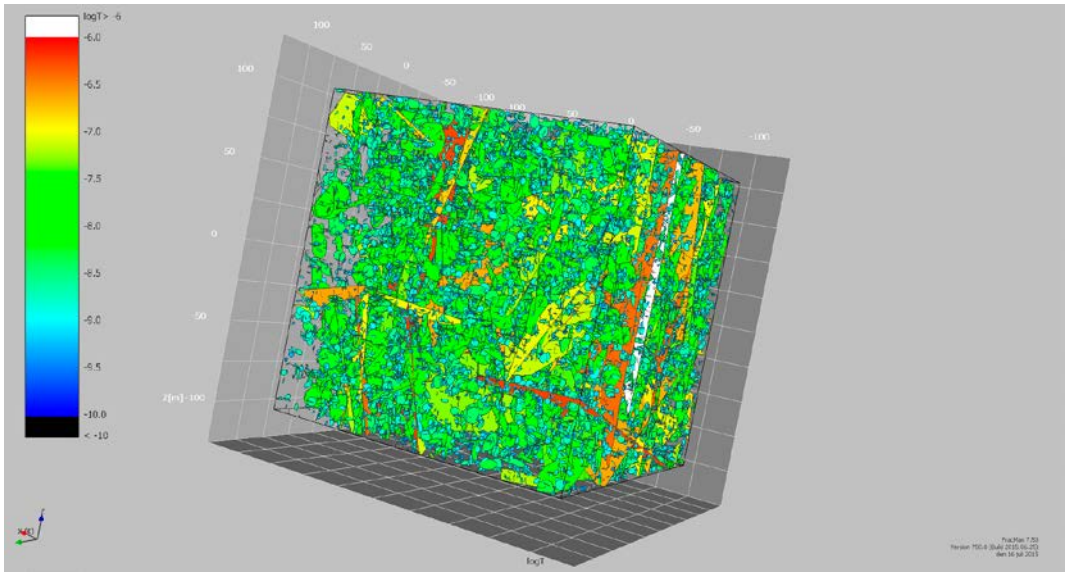

Figure 5-1. *View of the “reality realization”.*

The relationship between fracture aperture (a) and fracture radius (r) is based on Klimzac et al. (2010) and modified to:

$$a = N(m,s)\pi\sqrt{r/8} \qquad \text{Equation 5-1}$$

where N(m,s) is a random deviate from a normal distribution of mean m = 2.4E–4 m and standard deviation s = 2.4E–5 m.

The relationship between fracture transmissivity (T) and fracture aperture (a) is based on Zimmerman and Bodvarsson (1996) and modified to:

$$T = 41.5a^3 \qquad \text{Equation 5-2}$$

The relationship between storativity (S) and transmissivity (T) is based on Rhén et al. (2006) and modified to:

$$S = 7 \times 10^{-4} T^{0.35} \qquad \text{Equation 5-3}$$

5.2 Underground excavations and hydraulic tests

In the center of the reality realization there is one 100 m long main tunnel, two 100 m long deposition tunnels, and thirty-two 8 m long vertical deposition holes (Figure 5-2). Prior to the excavation of the deposition tunnels and deposition holes two 100 m long pilot holes are drilled and investigated hydraulically. After the two deposition tunnels are excavated, but prior to the excavation of the deposition holes, thirty-two 8 m long vertical pilot holes are drilled and investigated hydraulically. Both transient and steady state hydraulic tests are simulated in the Reality realization using MAFIC (Miller et al. 2001) although only the results from the steady state simulations are used for hydraulic conditioning in the DFN-R project.

The hydraulic test simulations in the pilot holes are mimicking constant head inflow/injection tests between packers. The test sections are 5 m long in the horizontal pilot holes and 7 m long the vertical pilot holes. The inflow test simulations assume atmospheric pressure between the packers; the hydraulic head in the test section equals the elevation head of the test section. The injection test simulations assume an excess pressure of ca. 200 kPa; the hydraulic head in the test section equals the ambient hydraulic head plus approximately 20 m excess pressure head. The choice of inflow tests versus injection tests may seem arbitrary from a theoretical point of view, since the key output parameter is the specific flowrate, i.e. the flowrate (Q) normalized by the change in head (dh), or the reciprocal quantity, specific drawdown (dh/Q). The choice of inflow tests versus injection test is not always arbitrary, however, because some fractures are connected to the tunnels, which have atmospheric pressure conditions (of course). In the hydraulic simulations performed for the DFN-R project it is assumed the inflow to the tunnels is at steady state prior to the hydraulic test simulations. From a hydraulic boundary condition point of view the tunnels are perceived as negative boundaries for the inflow tests and positive boundaries for the injection tests. Running both tests in each test section was found to help resolve the boundary conditions at hand. Furthermore, for the geometric and hydraulic properties assigned to the Reality realization it was found that a total test time of a few days could give a significant different interpretation of the hydrogeological conditions at hand compared to e.g. 20 min of testing which is the typical duration of constant head injection tests (Figure 5-3). The results from the steady state simulations are used for hydraulic conditioning in the DFN-R project. However, 80 % of the 5 m tests and 68 % of the 7 m tests were found to be non-conductive leaving a very small number of tests to condition/calibrate against (Figure 5-4).

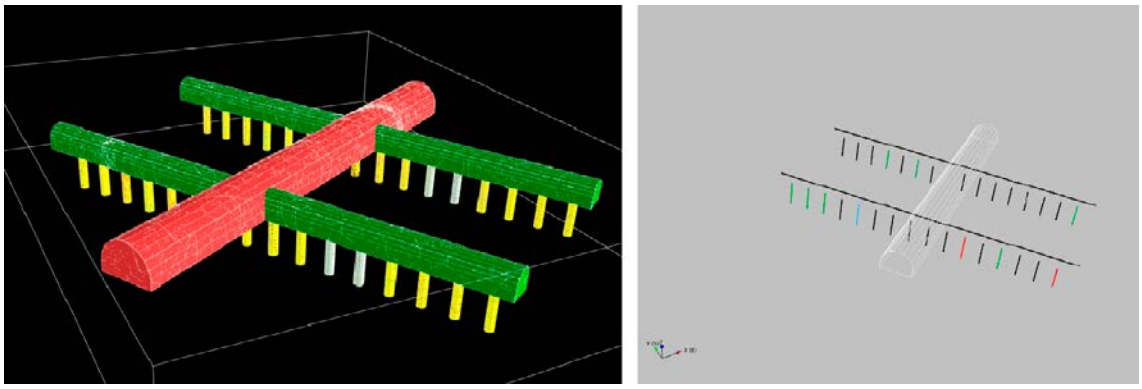

Figure 5-2. HypoSite deposition tunnels and deposition holes (left) and corresponding boreholes (right).

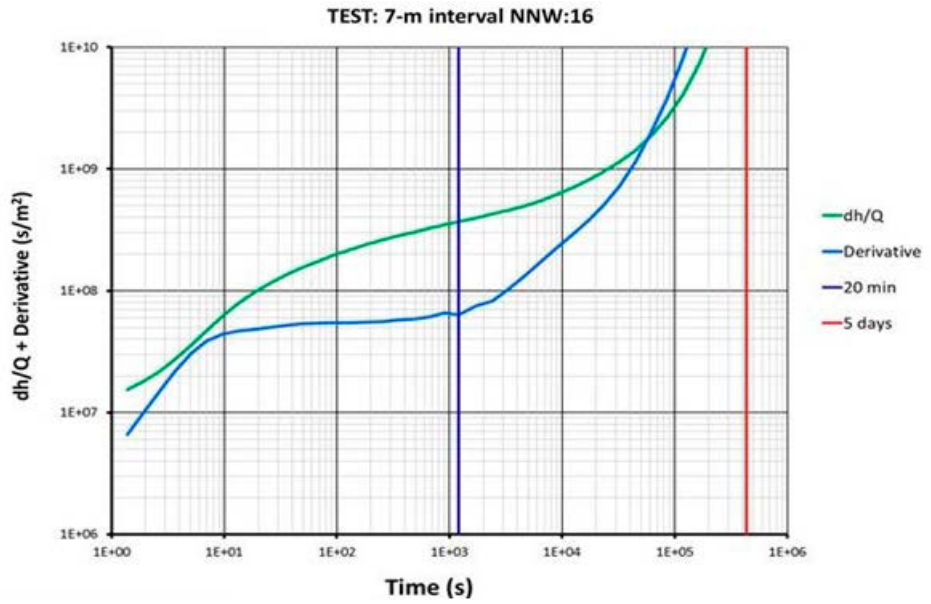

Figure 5-3. Example of a constant head injection test result from HypoSite.

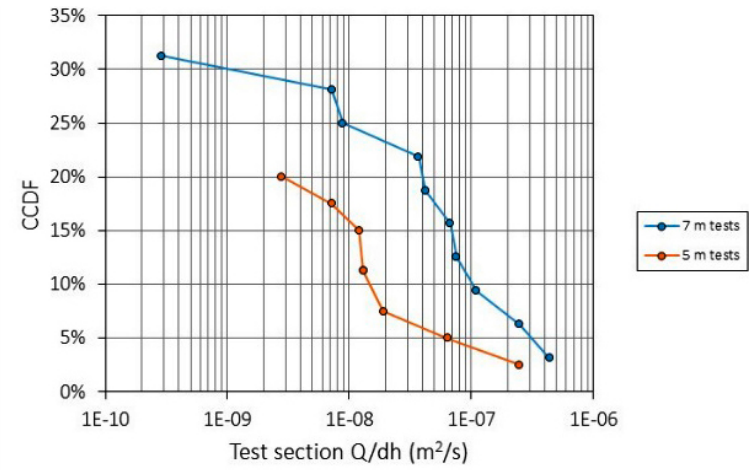

Figure 5-4. CDF plot of all performed constant head injection tests.

6 HypoSite example

This section contains a demonstration of the presented methods (Sections 2 to 4), applied to the synthetic DFN model HypoSite (Chapter 5). The model was used for providing input information about the geometries of fracture tracemaps from tunnels and boreholes intersections. Hydraulic test data were provided from steady state and transient flow tests performed within the model. The provided data set represents an input data set that could have been measured during a real construction project. The aim was to demonstrate the capabilities of the full suite of developed conditioning methods and is not an attempt to describe the whole method of creating a DFN model from observed data. For this reason, fracture recipes used to create HypoSite, with respect to size distribution, fracture geometry and intensity were known to the modeling team so all generated, so all generated unconditioned DFN models were statistically similar to the HypoSite model.

6.1 HypoSite tunnels and boreholes geometry

The HypoSite model contains one main tunnel oriented in NE direction (NE Main Tunnel) with two crossing deposition tunnels in the NNW direction (NNW Deposition Tunnel A and NNW Deposition Tunnel B). Each horseshoe shaped tunnel is 100 m long and with a width of 5.0 and 2.5 m for the NE main tunnel and the deposition tunnels respectively. Below each deposition tunnel are 16 deposition holes with a diameter of 1.5 m and a depth of 8.0 m. Each object in the model also contains a pilot borehole at its center represented as red, green and black lines in Figure 6-1.

The data in the presented conditioning workflow are used in a sequence representing the construction stages. The following list contains data accessible at each sequence, similar to what would be measured in the field.

1. NE Main Tunnel Pilot Hole – borehole log with fracture intersections.
2. NE Main Tunnel excavation – tunnel tracemap, total inflow to the tunnel.
3. NNW Deposition Tunnel A Pilot Hole – borehole log with fracture intersections, inflow measurements for each 5 m interval along the borehole.
4. NNW Deposition Tunnel B Pilot Hole – borehole log with fracture intersections, inflow measurements for each 5 m intervals along the borehole.
5. NNW Deposition Tunnel A excavation – tunnel tracemap.
6. NNW Deposition Tunnel B excavation – tunnel tracemap.
7. Deposition hole pilot holes – borehole log with fracture intersections, total inflow into the pilot holes.
8. Deposition holes excavation – tracemap, total inflow to the deposition holes.

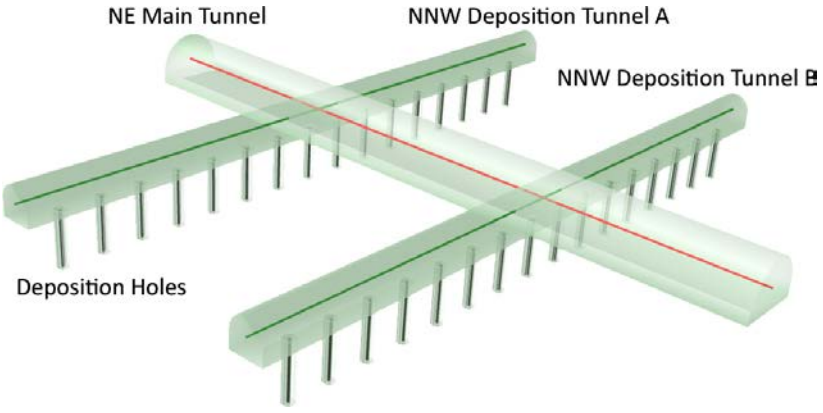

Figure 6-1. *A 3D view of NE Main Tunnel, the two parallel NNW deposition tunnels and the 32 deposition holes.*

In the following sections we focus on data from step two (the main tunnel excavation) and steps three and four (the pilot holes for the deposition tunnels).

6.2 HypoSite NE main tunnel

6.2.1 Geometric conditioning

The first step is to generate a stochastic unconditioned model using the same recipes that were used for creating the HypoSite model BM1-R4. The stochastic, unconditioned, model is statistically identical to model BM1-R4. This is illustrated in Figure 6-2 showing tunnel tracemaps on the NE Main tunnel from the BM1 model (a), the unconditioned stochastic model (b) and the conditioned model (c).

The intersecting fractures for all three realizations are presented in Figure 6-3. From a visual observation it is apparent that the intersecting fractures in the original model are larger than fractures in the other two models. Figure 6-4 shows the histograms of the equivalent fracture radii. It can be seen that in the BM1-R4 realization there are three fractures with radii larger than 50 m while the largest fracture in the conditioned model has a radius of 45 m. The reason for this is that the BM1-R4 realization is, in a statistical sense, an "outlier" as it contains more larger fractures intersecting the sampling tunnels.

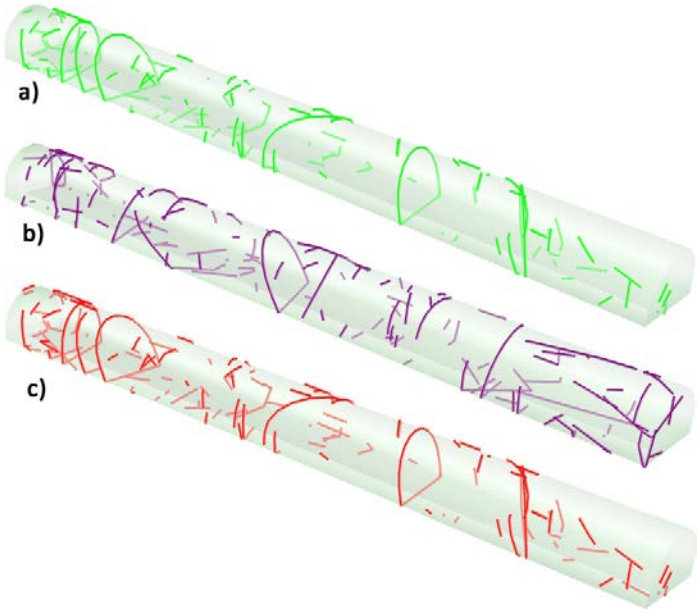

Figure 6-2. *NE Main Tunnel tracemaps. a) Original traces in BM1-R4 b) Traces from unconditioned model c) Traces from model after conditioning.*

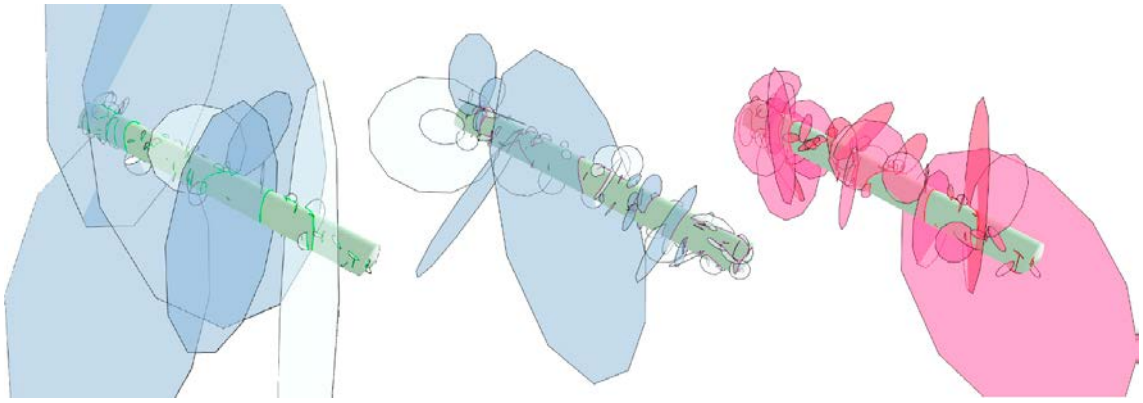

Figure 6-3. *NE Main Tunnel intersecting fractures. a) Original BM1 b) Unconditioned model c) Geometrically conditioned model.*

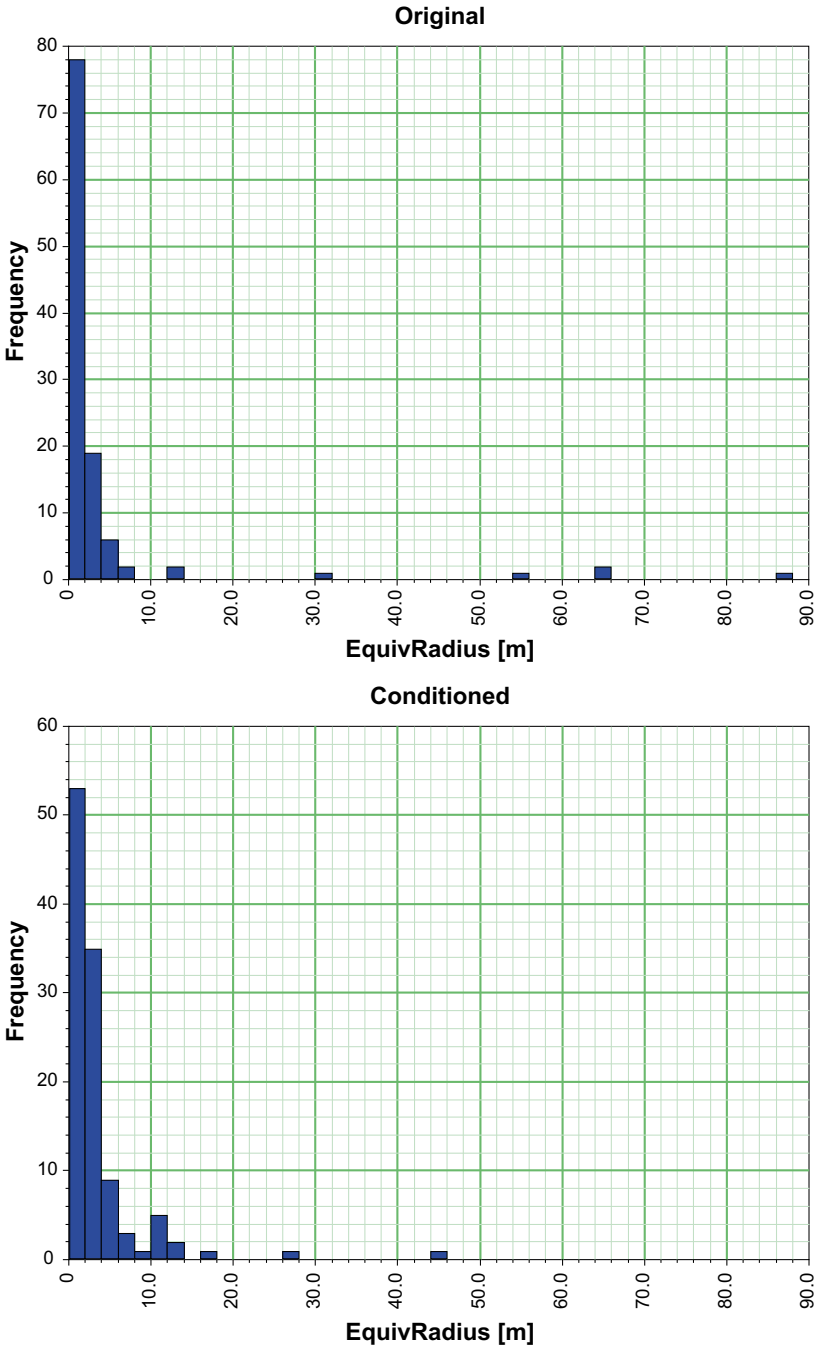

Figure 6-4. *Fracture equivalent radius histograms of the NE Main tunnel intersecting fractures. BM1-R4 is shown on the top histogram and the conditioned model on the bottom.*

As the geometric and connectivity conditioning processes run in parallel, information on required connection level Cx for each trace have to be assigned simultaneously in the method. Since information about exact inflow locations to the tunnel is unknown (as we only have information about total inflow and inflow in each individual trace is unknown) it was decided that all observed traces formed by a fracture with a radius larger than 10 m must be connected to a boundary (Figure 6-5).

The geometric conditioning yields a perfect match between the fracture orientations and almost a perfect match to the trace lengths, cf. Figure 6-6. The difference between conditioned and observed fracture trace lengths is that there are 102 observed traces but 103 conditioned fractures intersecting the tunnel. The difference is mostly caused by conditioned fractures that creates two traces on a same sampling object.

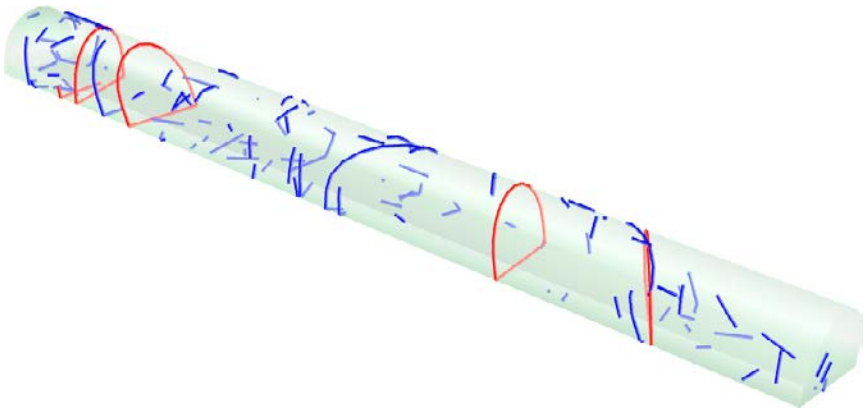

Figure 6-5. *Original traces on NE Main Tunnel. Red traces have a value of connectivity level set to 4 while the blue traces to –1 meaning that they do not need to be connected to any boundary.*

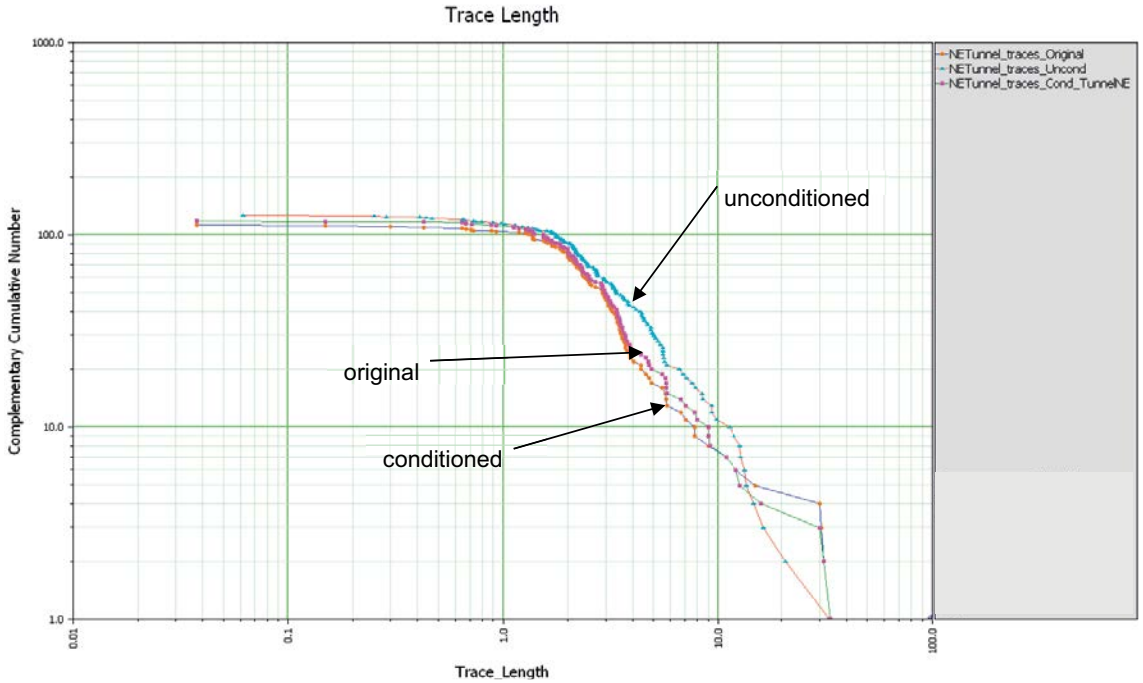

Figure 6-6. *Trace length distribution on NE Main tunnel showing traces from original, unconditioned and conditioned model.*

6.2.2 Hydraulic conditioning

After the DFN realization is geometrically conditioned, hydraulic properties of the fractures are calibrated to match the total measured inflow to the tunnel. Transmissivity of all fractures in the system is optimized using the following formula

$$T = b \times R^c \qquad \text{Equation 6-1}$$

where R is fracture radius and b and c are unknown constants. Using the workflow with PEST and ModMesh as described in Figure 4-1, constants b and c are estimated so that total inflow to the tunnel in the conditioned model is same as the observed total inflow in HypoSite. The model size and the flow boundary condition were identical to the HypoSite model. Table 6-1 summarizes the observed, unconditioned, and conditioned values for total inflow to the tunnel. For unconditioned and geometric conditioned models b and c parameters are same as in Hyposite model (based on Equation 5-1 and Equation 5-2 b = 7.86E–10 and c = 1.5).

Table 6-1. Total inflow to NE main tunnel.

	Target value [m^3/s]	Unconditioned [m^3/s]	Geometric cond. [m^3/s]	Hydraulic cond. [m^3/s]	Estimated b	Estimated c
NE Tunnel	2.78E–04	2.44E–04	2.83E–04	2.78E–04	9.4032E–09	1

The results indicate that the geometric conditioning alone improves the value of total inflow from 2.44E–4 m^3/s to 2.83E–4 m^3/s with the target value being 2.78E–4 m^3/s. After the hydraulic conditioning, total inflow matches the target value which is expected as there are only two degrees of freedom (parameters b and c) to match one target value.

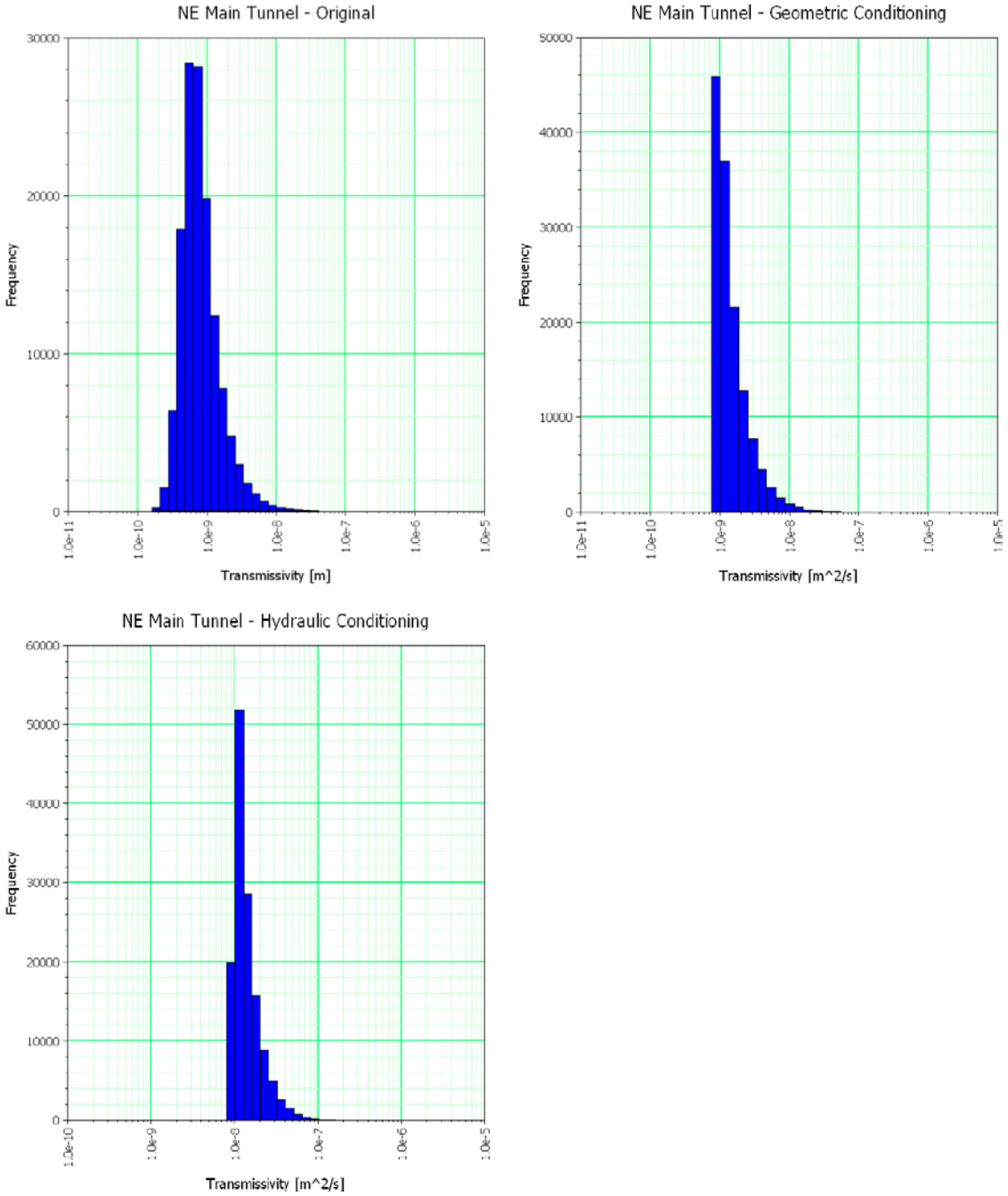

Figure 6-7. *Histograms of fracture transmissivity for original (top-left), geometric (top-right) and hydraulic conditioned models (bottom-left).*

Transmissivity histograms for HypoSite model and conditioned model differ in shape even though the same parameters were used when generating the fractures for the conditioned model. The only difference between the HypoSite and the conditioned model is in the fracture aperture assignment. In the HypoSite model, the aperture distribution contains a random normal distribution N, Equation 5-1. In the hydraulic conditioning method it is not possible to include randomness in each realization as it would prevent PEST to estimate correct parameters. In all conditioned models the fracture aperture is calculated using Equation 6-2. This explains why the left tail of the HypoSite histogram is cut-off in the conditioned realizations.

$$a = 2.4\text{E}{-4} \times \pi \sqrt{r/8} \qquad \text{Equation 6-2}$$

A comparison of transmissivity of all fractures in the Panel size volume in HypoSite indicates that fractures in the hydraulically conditioned model have a transmissivity that is higher by one order of magnitude compared to the target data (and to the geometrically conditioned model). The HypoSite realization is an outlier in the sense that only few large fractures with large transmissivities provide most of the flow but the total fracture transmissivity in HypoSite is lower than in the conditioned model. The conditioned model contains more dispersed pathways that requires a higher transmissivity on average to provide a similar total inflow.

One would expect to see lower or similar T in the hydraulically conditioned model as the total tunnel inflow is larger for the geometrically conditioned model. Obtaining a lower total inflow to a tunnel requires lowering the fracture transmissivities in the flowing pathways. In the hydraulically conditioned realization the overall transmissivity of all fractures is lower than in geometrically conditioned model. This is probably caused by changes in flow pathways in the two models. It is possible that fractures with most inflow in one model might not necessary be the most flowing ones in the other model even though the geometry of both models is identical. This illustrates a general effect of flow in a sparse fracture system where flow pathways are affected by intersections and relative transmissivity difference between neighboring fractures.

6.3 HypoSite NE main tunnel and two pilot boreholes

6.3.1 Geometric conditioning

Two horizontal pilot boreholes were added to the simulation following the geometric and hydraulic conditioning to the main tunnel. The pilot boreholes are parallel to the main deposition tunnel axis. The addition of boreholes simulates the situation of successively adding data during a construction method and allows a continuous update to a model.

Figure 6-8 a to c illustrates the main tunnel traces and the pilot borehole intersections (illustrated as identical disks) for the original (a), unconditioned (b) and conditioned models (c). Boreholes NNW and NNW-B are both 100 m long and contain six and seven (or actually six as well as we ignore the intersection inside the tunnel) observed intersections respectively which illustrates the sparseness of the BM1-R4 fracture system. In the conditioned model traces and borehole intersections match perfectly the original model. Figure 6-9 illustrates the fractures intersecting the tunnel and pilot boreholes in the realization BM1-R4 of the HypoSite model (a) and in the conditioned model (b).

Figure 6-10 shows fracture size histograms of BM1-R4 (left) and the conditioned model (right). Similar to the conditioning method to the NE Main tunnel the BM1-R4 realization contains larger intersecting fractures than in the conditioned simulation. This is due to the fact that the conditioning method prioritizes already intersecting fractures and the chance of bringing in a new large fracture to an intersection has a low probability as explained in Section 2.1.

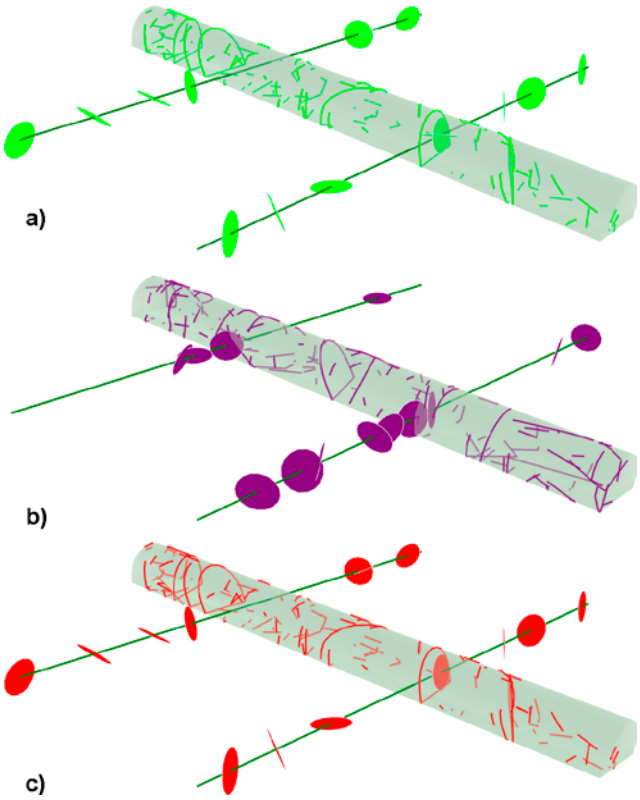

Figure 6-8. *NE Main Tunnel tracemaps and borehole intersections. a) Original BM1-R4 b) unconditioned model c) conditioned model.*

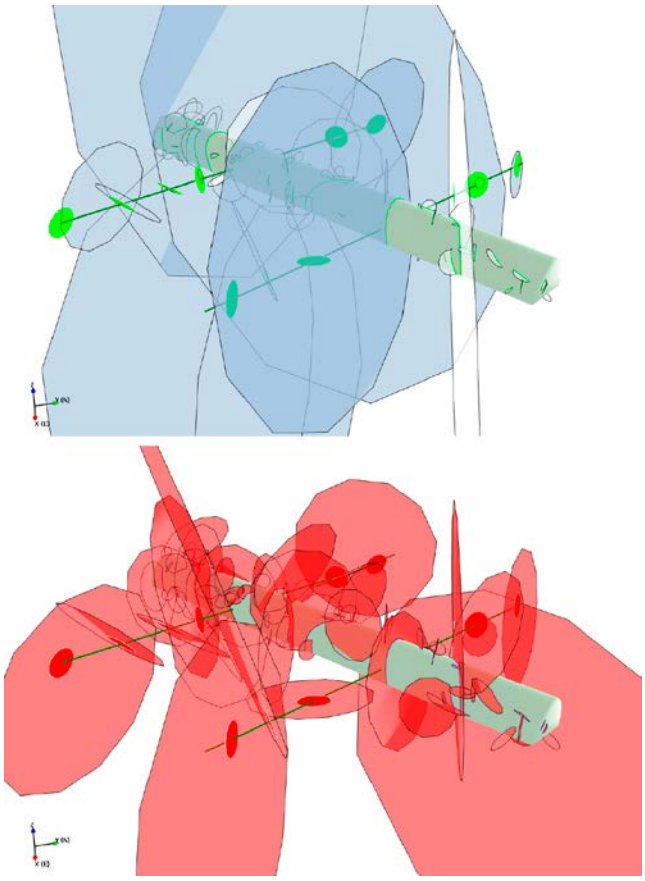

Figure 6-9. *NE Main Tunnel and boreholes intersecting fractures. Original BM1 model (top) and conditioned model (bottom)*

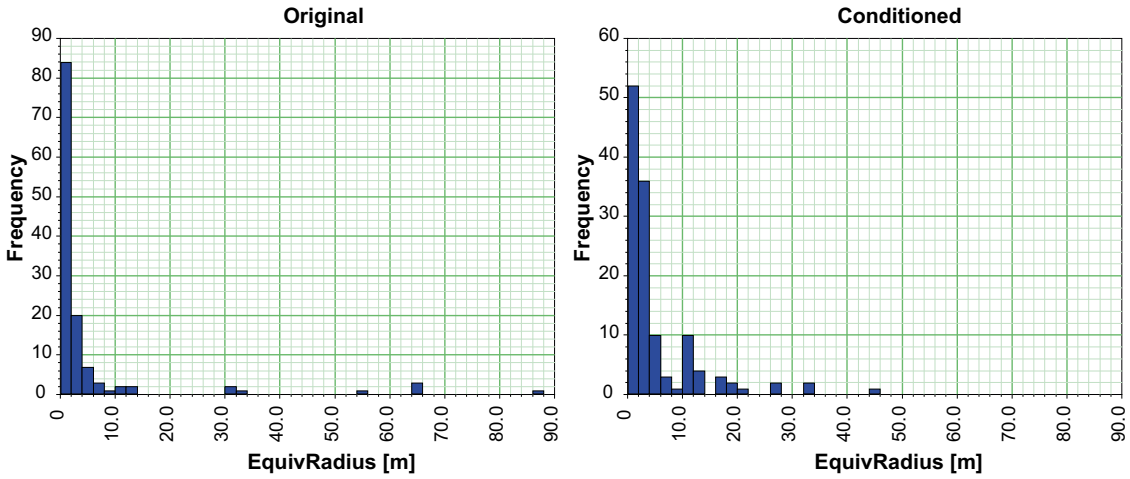

Figure 6-10. *Fracture equivalent radius histograms of NE Main tunnel and pilot boreholes intersecting fractures. BM1-R4 on left and conditioned model on right.*

6.3.2 Hydraulic conditioning

Steady state inflow data from each specific borehole location was used for the hydraulic conditioning of the model. Data from BM1-R4 realization and are summarized in Table 6-2. It should be noted that for each pilot borehole there are only four tested intervals with measured inflow. This indicates that some of the observed fracture intersections are not connected to a flow boundary.

Table 6-2. Steady-state and transient inflow for pilot borehole NNW and NNW-B.

Test interval	dh (m)	Q (m^3/s) dt = 1 200 min	Q/dh (m^2/s) dt = 1 200 min	Q (m^3/s) dt = 5 days	Q/dh (m^2/s) dt = 5 days	Q (m^3/s) Steady-state	Q/dh (m^2/s) Steady-state
NNW:1	226.17	5.60E–06	2.50E–08	3.00E–06	1.30E–08	3.00E–06	1.30E–08
NNW:3	221.69	3.10E–06	1.40E–08	1.60E–06	7.20E–09	1.60E–06	7.20E–09
NNW:16	175.43	4.90E–06	2.80E–08	3.30E–06	1.90E–08	3.31E–06	1.90E–08
NNW:17	175.56	6.80E–07	3.90E–09	5.00E–07	2.80E–09	4.95E–07	2.80E–09
NNWB:2	188.17	5.30E–05	2.80E–07	1.30E–05	6.90E–08	1.20E–05	6.40E–08
NNWB:3	187.83	3.60E–06	1.90E–08	2.30E–06	1.20E–08	2.21E–06	1.20E–08
NNWB:14	112.71	4.10E–05	3.60E–07	2.90E–05	2.50E–07	2.86E–05	2.50E–07
NNWB:20	293.56	5.50E–06	1.90E–08	3.90E–06	1.30E–08	3.81E–06	1.30E–08

Each pilot borehole was divided into 5m long intervals and steady-state flow was calculated for each interval separately. Using PEST® and the method workflow described in Chapter 4 fracture transmissivities were estimated (using b and c parameters in Equation 6-1) matching observed data from BM1-R4. In total 10 realizations (numbered as 3.1–3.10) have been conducted. The results from the best-conditioned realization (v3.1) are summarized in Table 6-3 and Figure 6-11.

The results indicate that the hydraulic conditioning method was able to reproduce the steady-state inflow to certain intervals such as PilotNNW_80 and PilotNNW_B_65 while at some other locations the difference in steady-state inflow was about one order of magnitude. This is reasonable as we are trying to match twelve observed values calibrating only two parameters, b and c. The hydraulic conditioning method is limited to calibrating the fracture transmissivity in the whole model while the geometry (fracture size, position and orientation) remains unchanged.

In three intervals; PilotNNW_25, PilotNNW_30 and PilotNNW_B_20, high inflows are recorded in the conditioned realization v3.1 but none in the BM1-R4 model (interval PilotNNW_B_75 has very little inflow which could be considered as no flow). This indicates that the intersecting fractures in these intervals are connected to flow boundary in the conditioned model while in the BM1-R4 there is no connection.

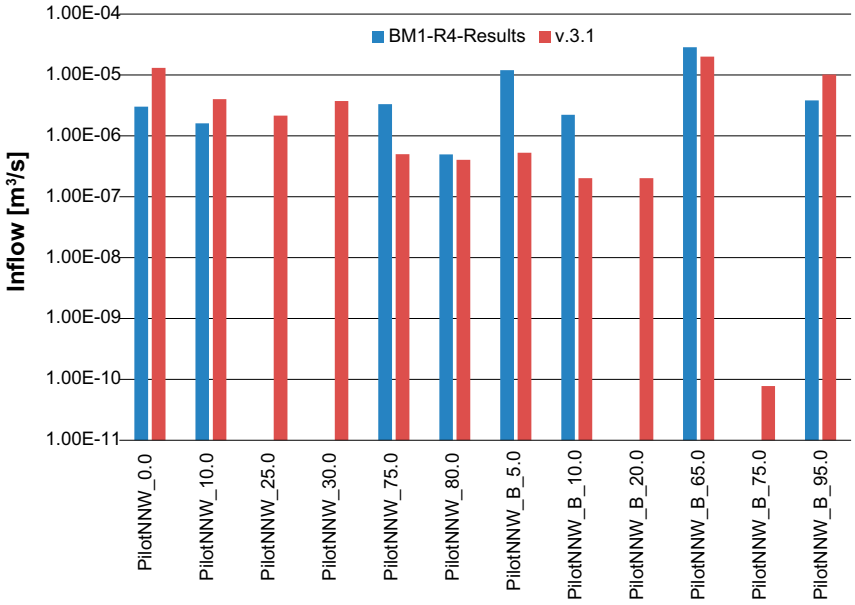

Figure 6-11. *Steady-state flow to pilot borehole intervals measured in the hydraulically conditioned model (red) and in HypoSite model BM1-R4 (blue).*

Table 6-3. Steady-state flow to pilot borehole intervals measured in the hydraulically conditioned model (v3.1) and in HypoSite model BM1-R4.

Interval name	BM1-R4 interval name	V3.1 results Q (m^3/s)	BM1-R4 data Q (m^3/s)
PilotNNW_0.0	NNW:1	4.66E–06	3.00E–06
PilotNNW_10.0	NNW:3	5.92E–06	1.60E–06
PilotNNW_25.0		1.60E–06	
PilotNNW_30.0		2.00E–06	
PilotNNW_75.0	NNW:16	2.53E–06	3.31E–06
PilotNNW_80.0	NNW:17	1.75E–06	4.95E–07
PilotNNW_B_5.0	NNWB:2	2.46E–06	1.20E–05
PilotNNW_B_10.0	NNWB:3	3.22E–06	2.21E–06
PilotNNW_B_20.0		1.35E–06	
PilotNNW_B_65.0	NNWB:14	5.22E–07	2.86E–05
PilotNNW_B_75.0		3.64E–11	
PilotNNW_B_95.0	NNWB:20	3.71E–06	3.81E–06

For this reason a test was performed to improve the connectivity level on each of the intersecting fractures based on the measured inflow. Borehole intersections formed by isolated fractures in the BM1-R4 realization were assigned a connectivity level Cx = –1 indicating that no connection level is specified. For other borehole intersections a connectivity level between two and six was assigned. The resulting levels of connectivity are visualized in Figure 6-12.

Using the updated connectivity levels, ten conditioning realizations were run. As described in Chapter 2 this means that the stochastic fractures in all realizations remain the same and only fractures intersecting sampling objects are geometrically conditioned to match the observed traces/borehole intersections. Results from all ten realizations are summarized in the whisker-plot in Figure 6-13, with the mean, 25 % and 75 % percentile as well as maximal and minimal inflow values.

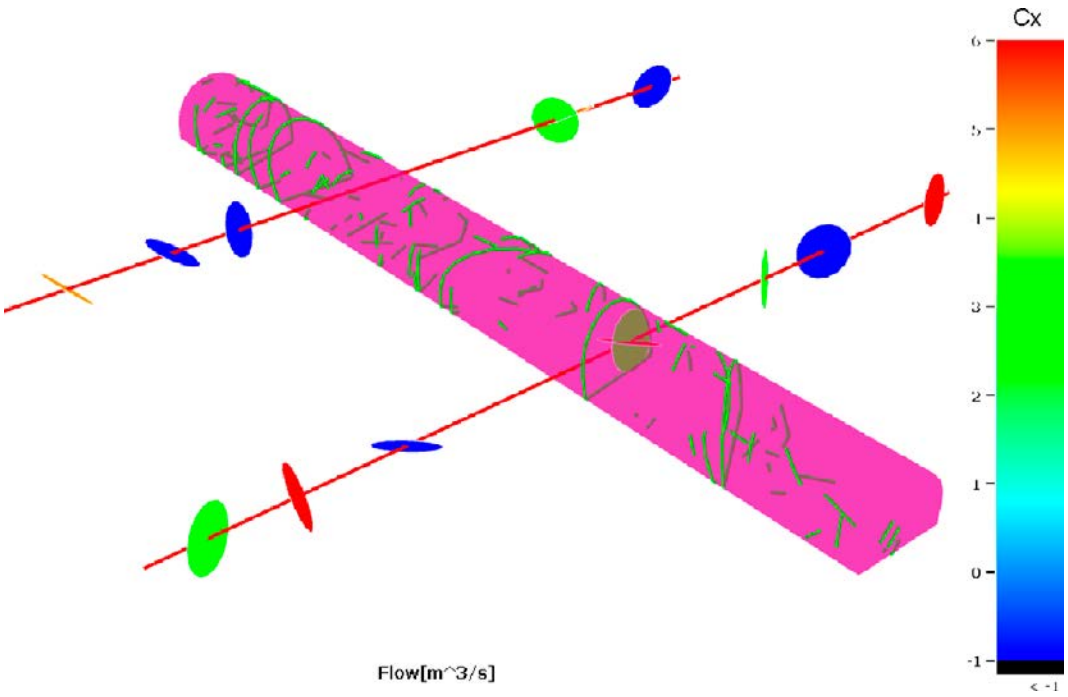

Figure 6-12. *Fracture intersections in the BM1-R4 realization colored based on the connectivity level.*

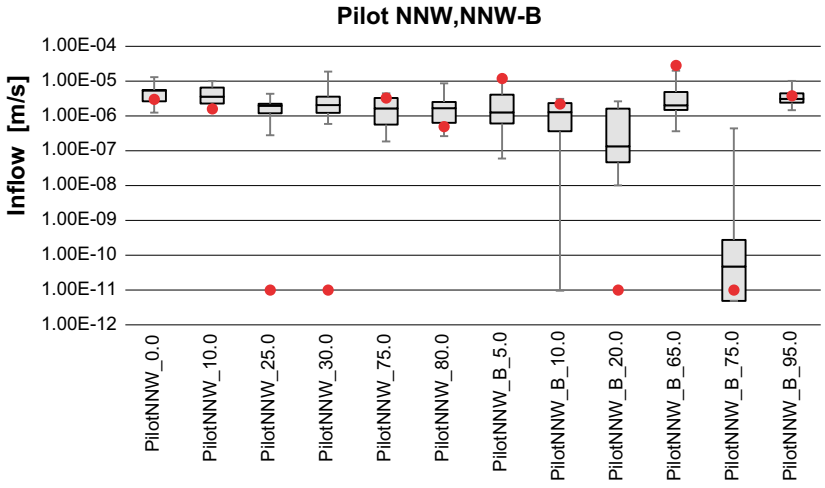

Figure 6-13. *Steady-state inflow for specific intervals in ten realizations. Red dots indicate the observed values from BM1-R4 which were used as conditioning target.*

Results show that flowing intervals in the conditioned simulations remain. An investigation of the stochastic fracture network near the sampling objects revealed sub-parallel fractures very close to the intervals as illustrated on Figure 6-14. These fractures acts as connectors between intersections and the flow boundary and will contribute to inflow to several intervals. Since the geometric conditioning method aiming to match all traces/borehole intersections the chances that some of the conditioned fractures would connect to one of these sub-parallel fractures (in red) is very high. It was also noted that this effect increases closer to Main tunnel which could be caused by tunnel conditioned fractures creating this additional connectivity.

A possible approach to minimize this effect could be to ignore borehole intersections which do not indicate any flow. Such a model would ignore observed local geometries but could be conditioned hydrologically. A more appealing approach would be to modify the connectivity method such that conditioned fractures are prohibited to create this connectivity in no-flow intersections.

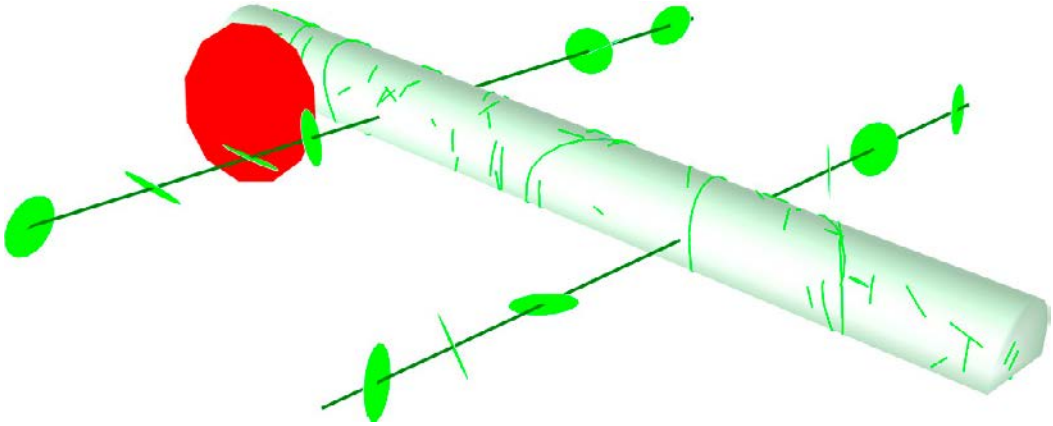

Figure 6-14. *Two parallel boreholes with fracture intersections visualized as green disks. Red fracture is very close almost parallel to a borehole but is not intersecting it.*

In the next step in the validation method four different stochastic realizations were conditioned to the observed data from HypoSite BM1-R4. The difference in this method was that for each realization new stochastic fractures where used in the conditioning method, leading to new connected networks. In the previous conditioning method the stochastic fractures in the model remained the same and only the intersecting fractures were conditioned. That means that fracture network flow channels remained unchanged in each realization and only the local geometry around the tunnels was affected.

In the current step new sets of stochastic fractures were generated and then conditioned to BM1-R4 data for each realization. Figure 6-15 presents the results of steady-state inflow to borehole intervals for four stochastic realizations. Simulated points of inflow to intersections with no recorded flow in the BM1-R4 realization remained but the locations vary through the realizations.

Realization v4.2 (presented in Figure 6-16) shows the closest match to the observed data in BM1-R4. Only at one interval (Pilot_NNW_30) the conditioned model had inflows above 1E–10 m^3/s not present in the BM1-R4 realization. For all other intervals the inflow rates were in similar ranges to the observed data.

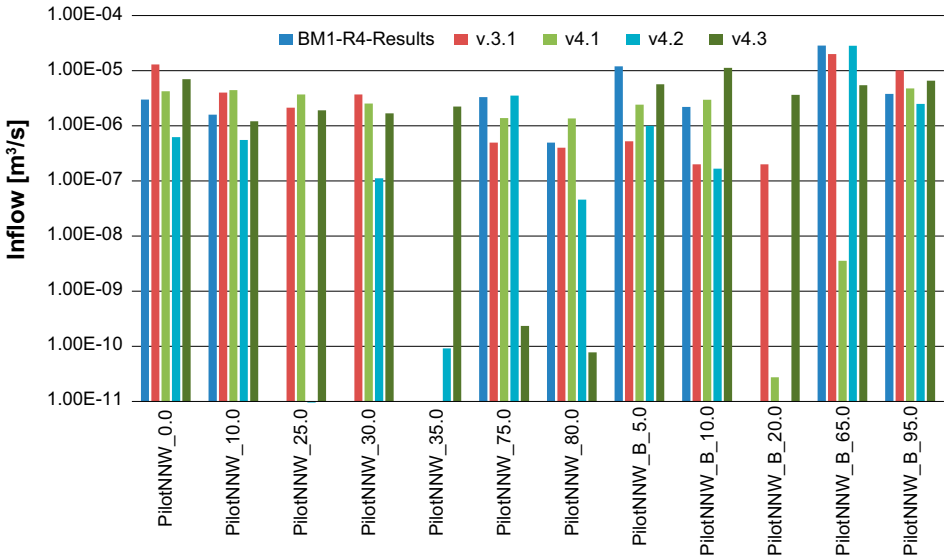

Figure 6-15. *Steady-state inflow to specified intervals in four different stochastic realizations. Blue columns on the left represents the values observed in the BM1-R4 model.*

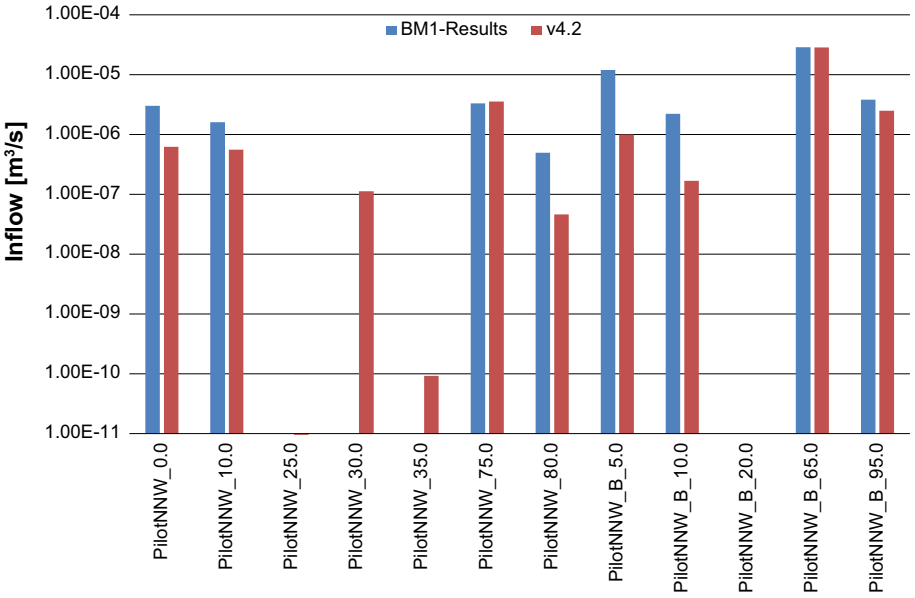

Figure 6-16. *Steady-state inflow for specific intervals in BM1-R4 and realization v4.2.*

6.4 Estimating inflow to deposition holes

The purpose of this study was to investigate the capability of estimating inflow to deposition holes by a model conditioned to two pilot boreholes (for the deposition tunnels) and trace data from the main tunnel in the HypoSite BM1-R4 model. For this analysis realization v4.2 was used.

Two tunnels NNW and NNW-B were inserted into the model at the locations of the corresponding pilot holes. From the bottom floor of each tunnel sixteen vertical deposition holes of 8 m length were tested for steady state inflow (see Figure 6-17). Flow simulations were run independently for each deposition hole, i.e. in each simulation only one vertical borehole was active (had constant head equal to 0).

The results from the simulations compared to observations made in realization BM1-R4 are shown in Figure 6-18. It is apparent from the results that there is very little match between the conditioned model and BM1-R4 realization. The results indicates that for a model to be able to predict flow at specific locations more than eight meters away or more a detailed conditioning needs to be performed using data from much closer to the deposition holes. This is a reasonable result as the predictive powers to such a detailed level is dependent on how far the "first" conditioned fracture geometry extends.

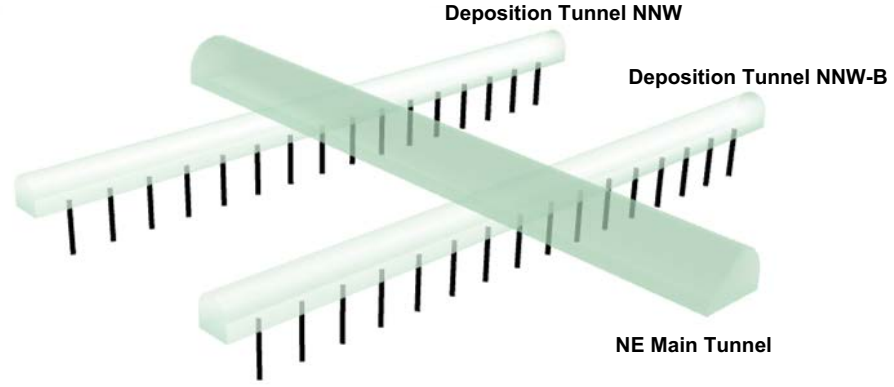

Figure 6-17. *Main NE Tunnel with two deposition tunnels (NNW and NNW-B) together with vertical boreholes (in black) at the location of each deposition hole.*

Obviously, eight meters or more away from the conditioned area, there is a dominance of stochastic fractures. Even though the model has been conditioned to intersections with the pilot boreholes directly above the deposition holes, these conditioned fractures have a low probability to intersect the relatively short and perpendicularly oriented deposition holes.

It is possible to include data from deposition boreholes and apply them in geometrical and hydraulic conditioning in similar manner as described above but this was out of scope of this study.

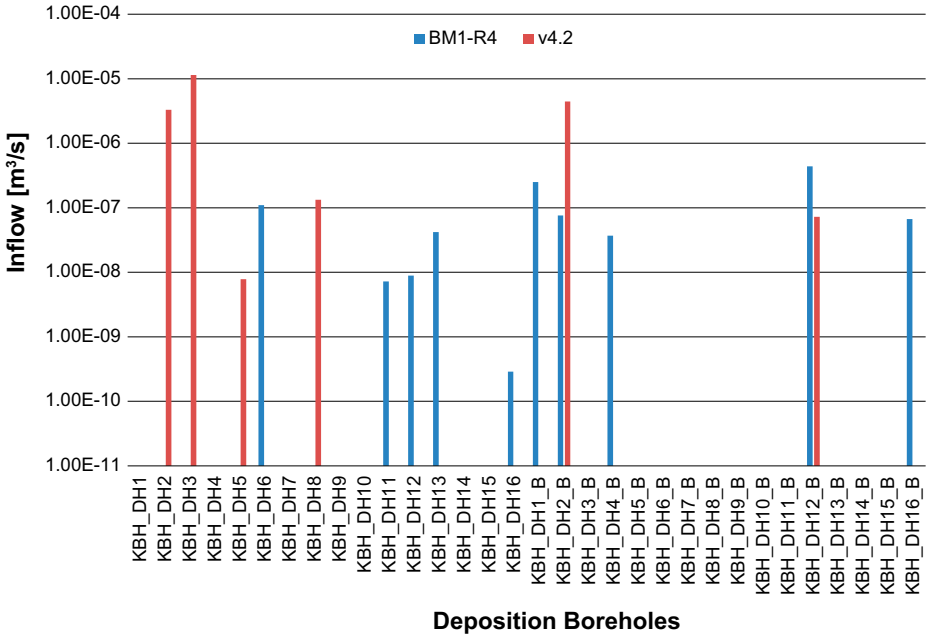

Figure 6-18. *Steady-state inflow to deposition pilot holes below two deposition tunnels.*

7 Summary

The HypoSite model was used for testing and validating presented conditioning methods. Results from the HypoSite realization BM1-R4 were considered as a set of “real observations in the field” and were used to condition the stochastic simulations. Conditioning followed the expected construction workflow starting with a main tunnel, adding information from pilot holes followed by construction of the deposition tunnels and deposition boreholes.

Conditioning to the NE Main tunnel tested the ability to condition to a single sampling object. It was shown that the geometric conditioning is working well under these assumptions as all tunnel traces were perfectly matched. Also using the hydraulic conditioning workflow tested the capability to condition to steady state inflow data in the tunnel.

Next the method was tested on conditioning to multiple objects; NE Main tunnel and two pilot boreholes. The geometric conditioning was made on tunnel traces and borehole intersection data with perfect match between observed and simulated data. Two pilot boreholes were divided into 5 m long intervals where steady-state inflow data was used for conditioning. Various conditioned realizations were able to reproduce observed steady-state inflows with reasonable degree of confidence. However, it was discovered that the conditioned realizations consistently had inflows at locations where no inflows were observed in BM1-R4 model. The reason is likely to be isolated clusters of fractures in the BM1-R4 model, whereas the simulations have clusters beyond the sampling objects that are connected to the boundary. If there is a requirement to perfectly match data both geometrically and hydraulically further improvement to the conditioning method is necessary. Beyond that, the conditioning method was able to reproduce the BM1-R4 data to a very reasonable level.

To test the predictive powers of conditioned models, the realization showing the best match to observed data in the NE Main tunnel and the two pilot boreholes was used to calculate the inflow to deposition boreholes. However, when comparing the results to the BM1-R4 data set there was little evidence of matching the observations. This indicates that for such a detailed prediction it is necessary to base the conditioning on observations made much closer to the deposition holes as the “determinism” in the model only extends as far as the “first” fracture beyond the tunnel wall or pilot borehole.

8 References

SKB's (Svensk Kärnbränslehantering AB) publications can be found at www.skb.com/publications.

Follin S, 2008. Bedrock hydrogeology, SDM-Site Forsmark, SKB R-08-95, Svensk Kärnbränslehantering AB.

Follin S, 2015. DFN-R Numerical simulation of hydraulic tests close to repository tunnels. Golder Associates AB, Draft report 2015-08-18, 78 pp.

Golder Associates, 2014. User Documentation FracMan 7, Version 7.5, Build 2014-12-01, www.fracman.com.

Hermanson J, Fox A, Öhman J, Rhén I, 2008. Compilation of data used for the analysis of the geological and hydrogeological DFN models. Site descriptive modelling. SDM-Site Laxemar, SKB R-08-56, Svensk Kärnbränslehantering AB.

Klimczak C, Schultz RA, Parashar R, Reeves DM, 2010. Cubic law with aperture-length correlation: implications for network scale fluid flow. Hydrogeology Journal, 18(4), pp 851-862. ISSN: 1431-2174, http://dx.doi.org/10.1007/s10040-009-0572-6.

Miller I, Lee, G, Dershowitz W, 2001. MAFIC – MAtrix/Fracture Interaction Code with heat and solute transport user documentation, version 2, Golder Associates Inc. Redmond.

Rhén I, Forsmark T, Forssman I, Zetterlund M, 2006. Evaluation of hydrogeological properties for Hydraulic Conductor Domains (HCD) and Hydraulic Rock Domains (HRD). SKB R-06-22, Svensk Kärnbränslehantering AB.

Zimmerman R W, Bodvarsson G S, 1996. Hydraulic conductivity of rock fractures. Transport in Porous Media, 23:1-30.

Appendix A

ModMesh documentation

This section contains a documentation of the input file format for ModMesh. Each command is specified by a defined keyword on one line and the attribute(s) on the following line(s). Commands can be written in any order. "#" can be used for commenting – for each line characters after # are ignored. Figure A-1 shows an example of a ModMesh input file. Each command is described in order which they appear in the presented example.

```
#Header
#mafic.exe full path
MAFICFULLPATH
C:\DFNR-2015_calc\PEST\mafic-2.exe
#run on multiple threads
MULTICPU
1

#input sab file
SABFILE
C:\DFNR-2015_calc\PEST\BM1-NETunnel\Main_tunnel_SSflow.sab
#input fab file
SABFILE
C:\DFNR-2015_calc\PEST\BM1-NETunnel\Main_tunnel_SSflow.fab
#input mff file
MFFFILE
C:\DFNR-2015_calc\PEST\BM1-NETunnel\Main_tunnel_SSflow.mff
INIT_H
0 0 0 400
INIT_Q
0 0 0 0

TEST_CURRENT #run a simulation with current boundary conditions
2            #boundary output for the results

#each well intersection runs a single steady-state flow test
WELLTEST
2
PilotNNW 5.0
PilotNNW_B 5.0

#change transmissivity y = A + B*x^C + D*e^(E*x)
MODIFY                                          #TRANS, APER, STOR, EQRAD
 ALL                                            #set id (use ALL for all sets)
 TRANS                                          #y
 EQRAD                                          #x
 0  10E-8 2   0       0                         #A B C D E
```

Figure A-1. *Modmesh input file format.*

MAFICFULLPATH – absolute path to a Mafic executable.

MULTICPU – number of threads to be used by ModMesh. Corresponds to the number of Mafic simulations run in parallel.

SABFILE – absolute path to the .sab file. See Mafic manual for more details.

FABFILE – absolute path to the .fab file. See Mafic manual for more details.

MFFFILE – absolute path to the .mff file. See Mafic manual for more details.

INIT_H – initial pressure head for nodes at start of a simulation. For each boundary nodes that are not active in a particular simulation ModMesh assigns an initial head equal to H = Ax+By+Cz+D. Each constant (A, B, C and D) need to be specified on next line and separated by space or tabulator.

INIT_Q – initial flux for nodes at start of a simulation. For each boundary nodes that are not active in a particular simulation ModMesh assigns an initial flux equal to Q = Ax+By+Cz+D. Each constant (A, B, C and D) need to be specified on next line and separated by space or tabulator.

TEST_CURRENT – this keyword performs an additional simulation without changing any boundary condition in an initial .mff. Next line specifies boundary ids for which the results are exported.

WELLTEST – ModMesh allows to perform automatically multiple tests along the borehole intervals. For each specified borehole an interval length is defined. ModMesh then searches each interval along the borehole for intersecting fracture. If an interval contains at least one intersecting fracture ModMesh creates a new mafic simulation where all boreholes boundary conditions except the active interval are deactivated (value of INIT_H and INIT_Q is assigned to the deactivated nodes). All other defined boundaries remain unchanged. Next line after the keyword “WELLTEST” defined the number of borehole to be tested. Each tested borehole must be specified on a single line by its name (same as boundary name in .mff file) and the interval size separated by space or comma.

MODIFY – this command allows to change mesh properties based on specified relationship. The command follow by 4 lines of attributes. First line specifies set to be modified. Single set can be identified by its id in .mff file or using an attribute “ALL” ModMesh allows to modify all fracture sets at once. Next line specifies which property should be modified (output value y). “TRANS”, “APER” and “STORA” keywords can be used for transmissivity, aperture and storativity respectively. Third line defines which variable should be used for calculating the specified property (input value x). Keywords “TRANS”, “APER”, “STORA”, ”AREA” and “EQRAD” where the last two represent area and fracture equivalent radius are accepted. The last line contains the values for 5 coefficients that defines the relation between the input and output variable using the equation $y = A + Bx^C + De^{(Ex)}$.

Arkitektkopia AB, Bromma, 2018

SKB is responsible for managing spent nuclear fuel and radioactive waste produced by the Swedish nuclear power plants such that man and the environment are protected in the near and distant future.

skb.se