Report
R-17-11
February 2018

Conditioning discrete fracture network models on intersection, connectivity and flow data

Pete Appleyard
Peter Jackson
Steven Joyce
Lee Hartley

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-17-11
ID 1668698
February 2018

Conditioning discrete fracture network models on intersection, connectivity and flow data

Pete Appleyard, Peter Jackson, Steven Joyce, Lee Hartley
Amec Foster Wheeler

Preface

The current report is one out of two concluding reports within the DFN-R project.

The project DFN-R, where ”R” denotes Repository, is aimed at developing a Discrete Fracture Network (DFN) methodology specifically suited for honouring measured data when describing conditions at a repository scale. Both geological and hydrogeological data are considered.

The active work within the project was performed during the time period 2013–2016 and involved two modelling teams, namely Amec Foster Wheeler and Golder Associates, working in close collaboration with SKB project members.

The stochastic DFN models of the fractured rock mass used previously, e.g., within SDM-Site and SR-Site, were based on data obtained from surface-based measurements and from measurements made in surface-drilled boreholes. Moreover, the models derived were unconditioned in that they did not honour data locally, except in a statistical sense. In the review of SR-Site, the authorities and their experts noted that the DFN models were associated with rather large uncertainties, but that these uncertainties were expected to be reduced once underground data become available during the construction phase of the repository.

In DFN-R, synthetic data from a numerical realisation of a hypothetical site, HypoSite, is used. Both geological data and hydrogeological data from steady-state hydraulic tests were provided to the modelling teams. The geological data consisted of intersections between the fractures in a defined DFN realisation, intended to mimic Forsmark conditions at repository depth, and a number of strategically positioned boreholes and tunnels of various orientations. The tunnels represent deposition tunnels with deposition holes, and transport tunnels, using their actual cross sections, spacings and dimensions. The hydraulic data resulted from constant-head inflow and injection tests in pilot holes for modelled deposition tunnels and deposition holes, as well as constant-head inflow tests into deposition holes.

DFN-R developed two independent methodologies (schemes) for performing local conditioning based on geological data or combined geological and hydrogeological data. The conditioning is shown to result in models that honour measurements locally and thereby also reduce the ensemble uncertainty such that a deterministic-stochastic transition is obtained with a higher degree of determinism locally in the areas where measurements are available.

The present report describes the conditioning methodology developed by the Amec Foster Wheeler team.

Jan-Olof Selroos and Raymond Munier
Project leaders DFN-R

Sammanfattning

I denna rapport presenteras en metodik för betingning av realiseringar från en diskret spricknätverks (DFN) modell på geometriska och hydrauliska sprickmätningar gjorda antingen i bergborrhål eller karterade på tunnelytor. Metodiken är implementerad i ConnectFlow. Olika test av metodiken görs, och resultaten visar att metodiken resulterar i god överensstämmelse mellan simulerade observationer av sprick- och hydraulisk data uppmätta i en eller flera tunnlar, deponeringshål eller i borrhål. Då metodiken använder bibliotek för att återskapa empiriska fördelningar av sprickor som kan skära tunnlar, ändrar metodiken inte på ett betydande sätt sprickintensitet, storleksfördelning eller orienteringsfördelning i närheten av tunnlar relativt a priori statistisk modell.

Fördelarna med att skapa betingade DFN realiseringar visas och kvantifieras för olika typer av mätdata. Detta görs med en syntetisk sprickdatabas, HypoSite, som möjliggör skapandet av syntetiska mätningar i en uppsättning av deponeringshål och tunnlar samt i de föregående pilothålen. Tillförlitligheten av betingade DFN modellers probabilistiska prediktioner av typiska mätetal för förvarets funktion såsom kapselflöden och flödesrelaterat transportmotstånd, undersöks genom att användanda olika kombinationer av geologisk kartering, inflöden samt hydrauliska test gjorda i pilothål. Studien bekräftar genom de betingade realiseringarna att implementeringen av betingning generellt är framgångsrik.

Studien visar ytterligare att betingningsmetodiken kan generera probabilistiska prediktioner av flödesförhållandena efter förslutning i specifika deponeringshålslägen. Detta ger en potentiellt substantiell hjälp i att bedöma om ett specifikt deponeringshålsläge är lämpligt för bruk eller inte. Möjligheter för att förbättra den prediktiva styrkan i metodiken testas. Detta inkluderar valet av realiseringar som är kända för att uppvisa likheter med uppmätt data, samt användandet av transient flödesdata för att få information om sprickor som inte korsar tunnlarna.

Abstract

In this report, an approach is presented for conditioning realisations of a discrete fracture network (DFN) model on geometric and hydraulic measurements of fractures made in the subsurface either in boreholes or mapped on the surfaces of underground openings. The approach is implemented in ConnectFlow. Various tests of the approach are carried out, and it is found that it provides good matches between simulated observations of fractures and associated hydraulic data on the surfaces of one or more tunnels or deposition holes, or in boreholes. Further, because it uses a library to replicate the empirical distribution of fractures that might intersect the tunnel, the approach does not significantly bias the fracture intensity, size distribution or orientation distribution in the vicinity of the tunnel away from the *a priori* statistical model.

The benefits of creating conditioned DFN realisations are demonstrated and quantified for different types of measurement data. This is done using a synthetic fracture database, HypoSite, which allows the creation of synthetic measurements around a small array of deposition holes and tunnels and their preceding pilot holes. The reliability of conditioned DFN models in providing probabilistic predictions of typical repository performance measures, such as canister flow-rates and flow-related transport resistance, using different combinations of tunnel structural mapping, inflows and hydraulic tests performed in pilot holes, is examined. It is found that the resulting conditioned realisations demonstrate that the implementation is generally successful.

It is further demonstrated that the conditioning method is able to provide probabilistic predictions of the post-closure flow conditions in specific deposition holes, thereby potentially providing a very useful aid in the determination of whether a specific deposition hole would be suitable for use. Options for improving the predictive power of the method are tested, including the selection of realisations known to be similar to the observed data and the use of transient flow data to infer information about fractures that do not intersect the tunnels.

Contents

1 Introduction

1.1 Background

In many rocks, groundwater flows primarily through an interconnected network of fractures. For such rocks, models that explicitly represent the flow through a discrete fracture network (DFN) provide a better description of the structurally constrained flow regime than alternative formulations. However, it is generally only possible to determine a limited subset of the fracture properties that would be needed to provide a complete description of the fracture network within the region of interest. Hence, the properties of many fractures have to be based on statistical inference from a limited set of direct measurements.

Numerical realisations of the fractures are set up by statistically sampling fractures in the region of interest and their properties on the basis of the observed distributions, so that the distributions of the fracture properties in the realisations reproduce the inferred distributions (to the extent possible given statistical uncertainties). If all the properties of some fractures have been measured or estimated, such fractures can be included deterministically in each realisation. The stochastic variability of the groundwater flows calculated for the different realisations provides a quantification of the uncertainty in those flows given the available information.

Example applications of stochastic DFN models include those used to model the groundwater flow at Forsmark (Joyce et al. 2010) and Laxemar-Simpevarp (Joyce et al. 2009) in Sweden and Olkiluoto in Finland (Hartley et al. 2013) as part of assessing the long-term safety of potential repositories for spent nuclear fuel. The modelling of Forsmark and Laxemar-Simpevarp carried out so far has used distributions of fracture properties derived from observations of fractures on surface outcrops and in boreholes drilled from the surface. However, it is expected that, in due course, underground facilities will be constructed at Forsmark and further investigations of fracture properties will be undertaken. In particular, it is expected that observations will be made of fracture intersections on parts of the walls of underground facilities, in pilot holes drilled during the construction of an underground facility and ultimately in deposition holes intended for the emplacement of spent fuel canisters. It is expected that hydraulic tests will be carried out and inflow measurements taken. Underground facilities have already been constructed at Olkiluoto and intersection data from tunnels, pilot holes and some demonstration deposition holes is available.

It would be highly desirable to condition stochastic realisations of a DFN on observed intersection data because this would then allow an estimate of the occurrence and extent of fractures connected to deposition holes, and hence reduce uncertainty in location-specific assessments of flow and mechanical stability (see Section 2.6 and Chapter 4). Further constraint of the flow properties in the local fracture system based on flow or hydraulic test data allows the uncertainty to be reduced even further. This reduction in uncertainty makes it possible to generate and apply rejection criteria for deposition holes, significantly reducing the likelihood of unsuitable locations being used, and thus reducing the overall risk to repository safety (see Chapter 4).

Conditioning realisations of a DFN model on observed fracture intersections with engineered openings is not straightforward in practice. In principle, all that needs to be done is to remove all the fractures in a realisation that intersect the engineered openings, and replace them with the fractures that lead to the observed intersections. However, a key issue is that an observation of a fracture intersection does not fully determine the geometric properties of the fracture. In particular, it may determine the fracture orientation, but it does not uniquely determine the size of the fracture and the position of its centre. The same intersection could result from, for example, a small fracture with a centre close to the engineered opening, or a large fracture with a centre some distance from the engineered opening. Thus while it is possible (given modelling assumptions) to preserve the statistical distributions of the fractures added to the tunnel over a series of possible conforming realisations, it is not possible to uniquely create the fracture from the observation.

There are some relevant approaches described in other literature. For example, Hestir et al. (2001) condition three-dimensional models on two-dimensional observed data based on an understanding of the geological processes in the rock, given relatively simple sets of observed data at the ground

surface (using a pixel map) and using a fracture growth model to extend fractures inward from the surface. Tran and Rahman (2006) combine neural networking and stochastic simulation of fractal fractures, again, to grow fractures from the observations. Almeida and Barbosa (2008) describe an approach growing triangular fractures from observations based on mechanical principles. However, none of these approaches are designed based on the case of an underground tunnel complex, and they consider only the geometric aspects of conditional fracture generation based on sparse data. The consequences for flow of fluid through the rock are not explored.

New research was needed to create an approach for conditioning on fracture intersections for a case such as the one considered here, evaluating the consequences for flow of fluid through fractured rock surrounding an underground system of tunnels and holes.

1.2 Objectives

The ultimate objective of conditioning in the context of the modelling of a geological spent nuclear fuel repository is to make accurate predictions of flow and transport in the vicinity of individual deposition holes, rather than general statistical predictions. Such calculations can be used to inform the criteria used for the selection of deposition hole locations and thus reduce uncertainty in repository performance.

In order to successfully make local predictions, it is necessary to use information that is local to the tunnels and deposition holes in question, and create a model that accurately reflects that information. As such, the initial objective is to create realisations of fractures that are consistent with data observed in tunnels and holes. Once these have been generated it will then be possible to determine how well they predict flow and transport around the specific deposition holes, compared with synthetic tests.

1.3 Scope

In order to achieve these objectives the scope of the work, the methods and results provided in this report assume a modelled system. The models used are idealised representations of tunnels and fracture networks; this project does not consider the adaptation of real data into those idealised representations.

Methods for inferring the general distributions of fracture properties in the rock based on the observed properties of fractures found on surface outcrops and in engineered openings such as boreholes and excavations have been established previously (e.g. Follin et al. 2013), and it is assumed that a DFN recipe describing the stochastic properties of the rock has been found that accurately reflects those properties. It is also assumed that the real-world system can be modelled with sufficient accuracy approximating mapped surfaces as planes and boreholes as lines. Just as in any other kind of modelling, it is very likely that a process of converting real-world data (such as intersections) to an idealised model will be required: this is a significant extra step in using a conditioning tool for a real-world problem, but is not within scope of this work; one method is described in Baxter et al. (2016b).

In order to allow generation of tools of this type, SKB have developed HypoSite, described below, a synthetic reality including fractures around a sample repository. Whereas the models provided give complete knowledge of all of the fractures in HypoSite, in order to be realistic it is necessary that data that cannot be determined in a real-world situation be assumed to be unknown when conditioning.

The data that is available therefore is primarily based on the geometric and flow properties of only those fractures that intersect the engineered opening. The flow properties of the fracture measurable at the intersection with the engineered openings do allow some inference to be drawn about the extent of the fracture; however, it is assumed that the modeller does not know any other geometrical information about fractures that do not intersect any engineered opening.

Whereas it is assumed that certain techniques for measuring e.g. inflow at the level of the intersection are possible and practical, it is likely that in a real world scenario, this may be more data than is available in practice. For example, it may be that flow information is only available in some parts of the model, or that short intersections with engineered openings are not mapped, or that mapping precision is not the same on every surface in the structure. It is noted for example that in real-world data in the ONKALO in Finland the tunnel roofs are often mapped with less precision than the tunnel walls, and tunnel floors are not included in the initial mapping, as described in Baxter et al. (2016a). Such differences will need to be taken into account.

1.3.1 HypoSite

HypoSite is a fracture network model created by SKB that represents a synthetic reality that can be used to test conditioning methods and other modelling approaches. HypoSite is intended to be a series of models of increasing complexity, starting at BM-0. The version of HypoSite used in this work is BM-1b.

HypoSite models the fractures around a small sample repository. This consists of a horizontal main tunnel, two horizontal deposition tunnels that cross the main tunnel perpendicularly, and 32 deposition holes, 16 for each deposition tunnel, equally spaced along the tunnels as illustrated in Figure 1-1.

Version BM-1b of HypoSite contains a network of fractures, including sets of fractures at varying scales around engineered openings in the model, boreholes and outcrops of rock at ground surface level. Ten realisations of fractures based on the same DFN recipe have been provided, numbered 1 to 10. Of these, realisation number 4 is the preferred realisation for this work because it has the largest number of deposition holes that are connected to the external surfaces of the model.

More detail on HypoSite and the specific modelled data as used in this work is given in Section 3.1.

1.4 Overview of the approach

On the face of it, there are a number of different approaches that could be taken to condition a model.

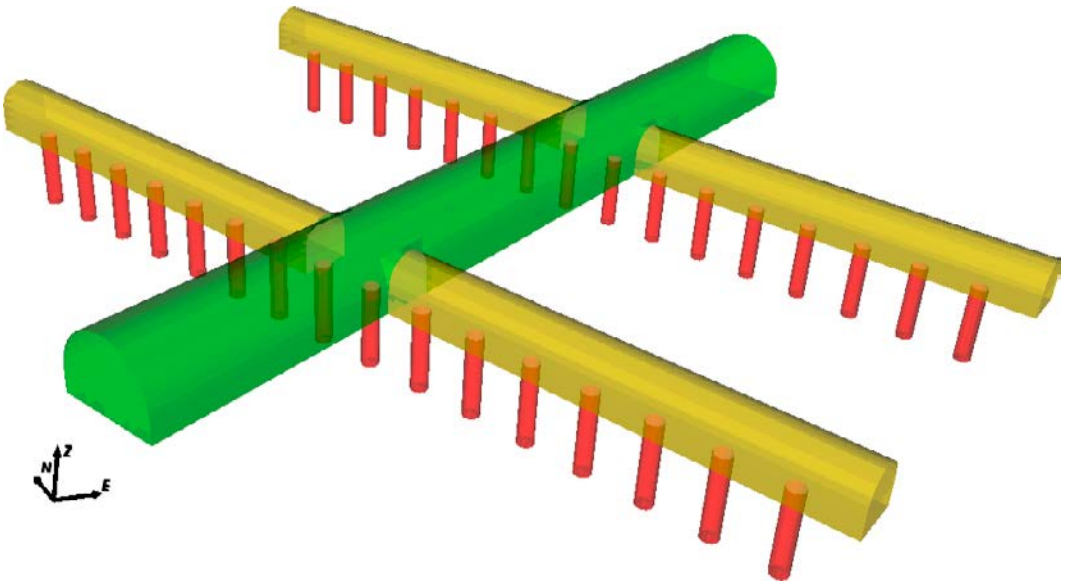

Figure 1-1. *Image of the tunnels and deposition holes in HypoSite. The main tunnel is in green, the deposition tunnels are in yellow and the deposition holes are in red.*

For example, it might be thought that one could determine by mathematical analysis, given a DFN recipe, the distributions of fracture properties corresponding to an observed intersection and then sample the properties of the fracture taken to correspond to the observed intersection on the basis of these distributions. This would ensure that the distributions of the properties of the fractures corresponding to the observed intersections would correspond to the DFN recipe provided. However, this method is not practicable. It may in theory be possible to determine mathematically the distributions of fracture properties corresponding to an observed intersection in a few highly idealised cases (such as, a case where circular disc-shaped fractures intersect a cylindrical opening), but this does not work in the general case where fractures may not be idealised as circular, openings may be of different shapes and may intersect, fractures may create intersections that are not either straight lines or simple curves, and the DFN recipe may include multiple fracture sets with different orientation distributions.

Another approach that might be suggested is a brute force approach. In this approach, for each observed intersection, realisations of fractures would be generated by sampling from the underlying distributions of fracture properties (without taking into account whether a fracture intersects the walls of the engineered openings) until a fracture is generated that is considered to give an acceptable match to the observed intersection. The model would then proceed to the next observed intersection and repeat the process. This would ensure that the fracture taken to correspond to an observed intersection would have properties sampled from the expected distributions, without the need to construct those distributions explicitly. However, the computational time required to generate such a large number of fractures for each intersection would be prohibitive, meaning that this approach would not be practicable.

The approach presented in this report makes this brute-force method more practical by creating a library, containing an empirical distribution of fractures that might potentially intersect the engineered openings, and then adding fractures to the final model by sampling from that distribution. It further explores the effects of conditioning using connectivity and flow observations associated with the intersections being conditioned. Ultimately it is the flow and fracture size that are the important characteristics of the fracture network in the context of repository safety.

Because the library is only generated once, much less computing power is required than in the brute force approach outlined above, and it is this that makes the approach practicable.

The library is generated by creating many realisations of the fractures in the region of interest by sampling from the underlying distributions of fracture properties. Each fracture in each realisation that intersects the engineered openings is added to the library with the geometric and flow properties of the intersection. Once a library has been generated, it can be used to condition any similar DFN model (with the same fracture statistics and whose engineered openings have the same or an equivalent geometry and an equivalent flow model), as many times as required.

The starting point for conditioning, once the library is in place, is an unconditioned realisation of the fractures in the region of interest. Each realisation is then conditioned to replace those fractures that intersect the engineered opening with fractures that match the observed intersections. This is a two stage process: first, those fractures that intersect the engineered openings are removed. Second, for each observed intersection, the library is searched to find a fracture that gives an intersection that is considered an acceptable match to the observed intersection. The library fracture is then included in the realisation.

When finding a matching fracture, there are various techniques that allow the search of the library to be very efficient, meaning that it is not necessary to search through every fracture in the library for every intersection observed, but only a much smaller subset. This also provides a significant performance improvement compared with the brute force approach.

This approach is applicable whatever the underlying distributions of fracture properties, provided that fracture distributions within a given domain are stationary (see Subsection 2.1.3) and that fracture properties are independent of one another. A useful validation test is to verify that the underlying distributions are maintained through the conditioning process: conditioning should not alter the *a priori* distribution of fracture properties for the synthetic case where the DFN recipe is known.

The approach has been implemented in ConnectFlow and extensive testing of the approach has been carried out. Initial work was released in version 11.1 of ConnectFlow, further work was released in version 11.3 (Amec Foster Wheeler 2015), and it is anticipated that ongoing improvement of the method will be included in subsequent versions.

1.5 Glossary

The following is a glossary of terminology used in this report.

Deposition tunnel	A side tunnel from the main tunnel, under which the deposition holes are located.
Deposition hole	A short vertical shaft that may be used to dispose of spent fuel canisters.
DFN Recipe	A conceptual model of fracture properties and their statistical distributions
Domain	A subset of the library and/or observed data within which fracture distributions can be assumed to be stationary.
Engineered opening	A constructed underground structure, such as a tunnel, deposition hole, shaft, pilot hole or borehole.
F	The flow-related transport resistance measured for a particle released from a deposition hole in the model. See Joyce et al. (2010) for details of calculation.
Fracture	A natural discontinuity in the rock formed by brittle failure that is modelled as a planar feature
Geometric conditioning	Conditioning taking account of geometric data only, i.e. without using flow measurements.
Hydraulic conditioning	Conditioning taking account of injection test data.
HypoSite	Hypothetical site for use as a synthetic reality to provide observed data. A suite of benchmarks is conceived; the benchmark used here is BM-1b. See also Subsection 1.3.1 and Section 3.1.
Inflow	The measured flow from a fracture through a given intersection between a fracture and a model surface or pilot hole that is empty of water and at atmospheric pressure.
Inflow conditioning	Conditioning taking account of inflow data.
Injection test	A test in which a section of pilot hole is sealed off and water is injected into it to maintain a given pressure above that of the surrounding fractures; the flow-rate of water injected is measured.
Intersection (with engineered opening)	The physical intersection that may be observed between a fracture and any engineered opening.
Large-fracture library	A library containing a subset of the distribution of fractures in the model; this library contains large and very large fractures only, see Section 3.2.
Library	List of fractures intersecting the engineered openings from a large number of realisations of a stochastic DFN model and associated intersection and (if applicable) flow information.
Main tunnel	The main access tunnel.
Medium-fracture library	A library containing a subset of the distribution of fractures in the model; this library contains mid-sized and large and very large fractures, see Section 3.2.
Observed data	The collection of data recorded and input to the modelling as a set of observations to be matched. This may be observed on site or based on simulated data.
P_{21}	The total length of the intersections on a specified surface (such as the external surface of a tunnel or deposition hole), per unit area of the same surface.
P_{32}	The total area of the fractures in a given volume, per unit volume.
Pilot hole	A cored borehole, modelled as a scan line, drilled from within a tunnel along the line planned for a new or extended tunnel or deposition hole. In this work, pilot holes are modelled as 1-dimensional scan lines with point intersections.
Pilot hole for deposition hole	A pilot hole along the line of a planned deposition hole.
Pilot hole for deposition tunnel	A pilot hole along the line of a planned deposition tunnel.
Pole (of fracture)	The direction of the vector perpendicular to the plane of a fracture.
Radius (of fracture)	Measure of the size of a fracture. As fractures used in this work are all square, the equivalent radius is the radius of a circle of the same area as the fracture, i.e. $r = l/\sqrt{\pi}$, where r is radius and l is fracture side length.

Realisation	One instance of the specific stochastic fracture network that results from generating fractures according to the DFN recipe.
Set (fractures)	A discrete category of fractures that can be distinguished from other categories. In general, different sets will be associated with different statistical distributions in the DFN recipe.
Simulated observed data	A set of observed data that has been generated by DFN simulation, rather than being observed at a real site. This is normally one realisation of the model. In this work, all observed data has been simulated.
Surface (of engineered opening)	The surface of an engineered opening, such as the wall, ceiling and floor of a tunnel or the walls and base of a deposition hole. Modelled surfaces of tunnels and deposition holes are generally made up of planar sections, and intersections are observed on them as lines. In this work, pilot holes are modelled as scan lines, and are thus assumed not to have modelled “surfaces” per se.
Tunnel	Any significant volume, modelled as a void, whose floor is approximately horizontal; this includes the main tunnel and deposition tunnels.
U	The average tangential flow-rate per unit length measured for a given particle released from a deposition hole in the model. See Subsection 2.8.7 for a description of how this is calculated.
Unconditioned model/ realisation	A model with no conditioning applied, i.e. a purely stochastic realisation generated according to the DFN recipe.
Very large-fracture library	A library containing a subset of the distribution of fractures in the model; this library contains very large fractures only, see Section 3.2.

1.6 Structure of this report

The structure of the report is as follows. In Chapter 2, the approach for conditioning in fracture intersections with surfaces or pilot holes is described in detail. In Chapter 3, verification results and tests of geometric calibration and consistency are demonstrated. In Chapter 4, results of conditioning on flow are demonstrated, including the prediction of performance measures, U (average flow-rate per unit length) and F (flow-related transport resistance). Chapter 5 provides conclusions.

2 Methodology

The following chapter describes the conditioning methodology as implemented in ConnectFlow. As shown in Figure 2-1, the method starts by generating a library of fractures that intersect the engineered openings. The fractures that intersect the engineered openings in an unconditioned realisation are replaced with fractures chosen from the library as approximate matches for the intersections observed, and this conditioned realisation is then used as a basis for flow and transport calculation.

2.1 Overview

2.1.1 Initial considerations

The starting point for the approach is a DFN recipe for the fractures in the region of interest. This model will have been specified on the basis of analysis of data such as fracture data from boreholes, surface outcrops, lineaments and geophysical interpretations. Statistics related to intersections on the walls of engineered openings, such as those to be used for conditioning, may also have been taken into account. The model may also have been constrained by calibration against other information, such as statistics of inflows to boreholes. Considerable effort (which lies outside the discussions in this report) may have been expended in the specification of a suitable DFN recipe. See, for example, Follin et al. (2013) for a discussion of the specification of a DFN recipe for the fractures at Forsmark.

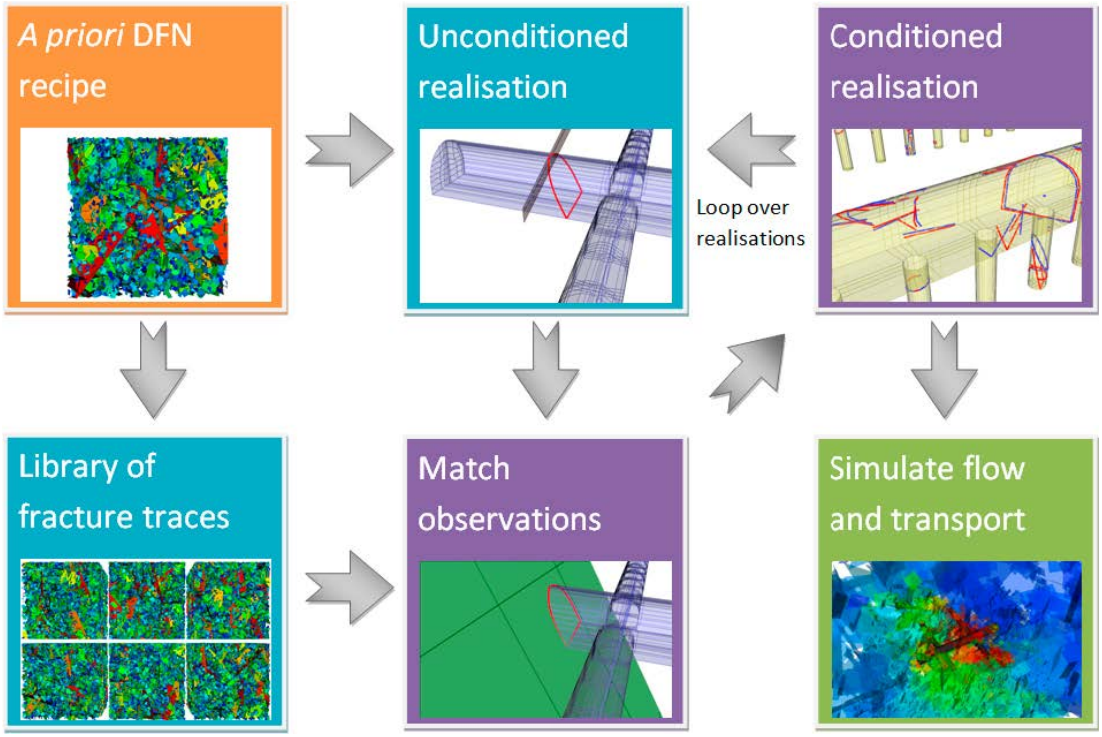

Figure 2-1. *Flow chart showing the process of conditioning in ConnectFlow. The DFN recipe is used to generate a library of fracture intersections with the engineered openings and a number of unconditioned realisations. The fractures intersecting the engineered openings in the unconditioned realisations are removed and replaced with fractures from the library that approximately match the observed data. Flow and transport is then simulated for each conditioned realisation.*

In a practical application of the approach, before attempting to generate realisations conditioned on the observed intersections, it is good practice to first check the consistency between the specified DFN recipe and the observed data, for example by creating a number of realisations of the unconditioned model and confirming that the observed data is within the range of possibilities predicted by those realisations (allowing for the fact that the observed data may not fully represent all possibilities implied by the recipe). This is because significant computational effort is involved in generating libraries and conditioned realisations, and it would not be sensible to expend this effort if the model being used is inappropriate or can reasonably be improved. The importance of this will depend on how different the models are from those that would create the observations: it might be very difficult or time-consuming to condition realisations of an inappropriate model on observed intersections, and the final results are less likely to reflect the variation expected in the observed fractures.

If the DFN model is reasonably consistent with the observed data, but differs in terms of fracture intensity, the conditioned result from geometric conditioning may not be significantly worse than when conditioning using a DFN that is fully consistent with the observed data (though it may be more difficult to correctly match flows). An example of conditioning using an unconditioned DFN model that was not entirely consistent with observed data is given in Section 3.5.

2.1.2 Method description

As indicated in Section 1.4, the first step in the approach to conditioning on observed intersections is to create a library of fractures intersecting the engineered openings of interest. As will be discussed in detail in Section 2.3, the library is created from many realisations of the fractures in the region of interest, generated by sampling the distributions in the underlying DFN model. Then, for each fracture in each realisation, it is determined whether or not the fracture intersects the engineered openings. If it does, the details of the fracture and its associated intersection (or intersections), including all the information needed to recreate the fracture, the details of the geometry of the intersections, and whatever flow measures have been simulated, are added to the library. Once complete, the library is stored for later use.

To create a conditioned realisation of the model, first, an unconditioned realisation of the same DFN model as the library is created. Next, the fractures that intersect the engineered openings are identified and removed (see Figure 2-2 and Figure 2-3). Note, however, that if all intersections between a fracture and the engineered openings are so short that they would not have been recorded, that fracture is retained in the model. Then, for each observed intersection, the library is searched to find candidate fractures that have an acceptably-matching intersection. As will be described in Section 2.4, the data structures used for the library search have been optimised to provide a very efficient search. A fracture is selected from the candidate list by a weighted random selection, translated into place (either along the engineered opening or from opening to opening) and added to the model (see Figure 2-4). This approach ensures that the properties of the fractures that correspond to the observed intersections are effectively sampled on the basis of the appropriate distributions, given the identified DFN recipe.

2.1.3 Benefit of taking the fracture distributions to be stationary

The conditioning methodology described here assumes that fracture distributions are stationary within a domain (see Subsection 2.1.4 for details on domains). This has the significant benefit that a much smaller library can be used than would otherwise be the case, leading to a much lower computational cost for generating the library.

The reason that the library can be smaller is that the distribution of fractures leading to a particular intersection does not depend on the position of the intersection on the engineered opening. Therefore, in attempting to match an observed intersection, it is not necessary to match the position of the observed intersection in the opening. Once an acceptable match has been found, the library intersection and its associated fracture can be translated an appropriate distance along the opening, or between openings, so that the overall distance of the intersection along the opening matches that of the observed intersection. The fracture distributions obtained in this way are the same as those that would be obtained by using a much larger library and matching the distance along the engineered opening of an observed intersection as well as its other parameters.

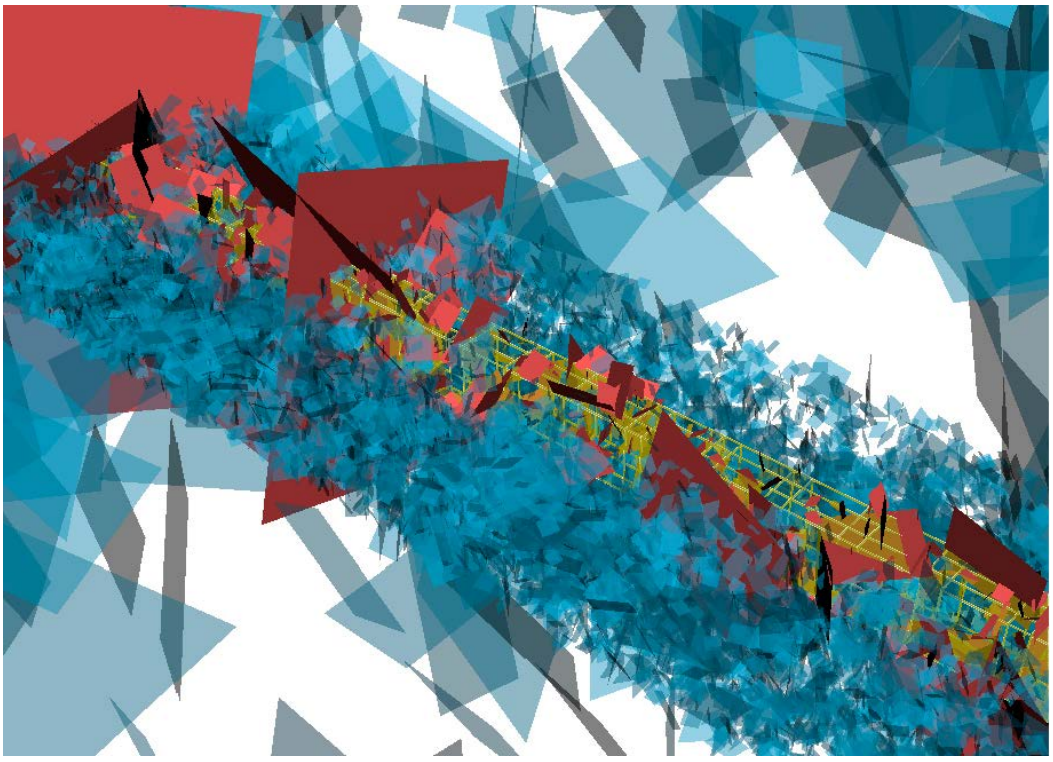

Figure 2-2. *An unconditioned realisation of the fractures in a region around a tunnel, with the fractures that intersect the tunnel highlighted (in red). The intersections from these fractures will not match the observed intersections. For ease of visualisation, some fractures are removed from view (around the top of the tunnel).*

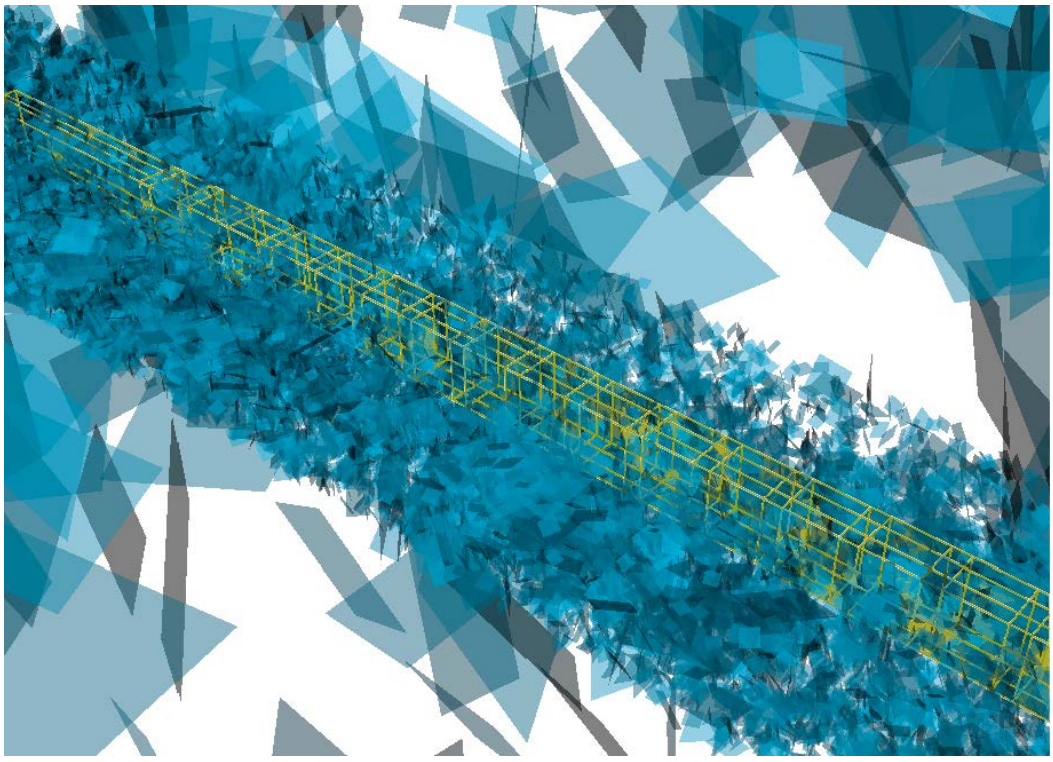

Figure 2-3. *The same unconditioned realisation of the fractures in a region around a tunnel as in Figure 2-2, with the fractures that intersect the tunnel removed.*

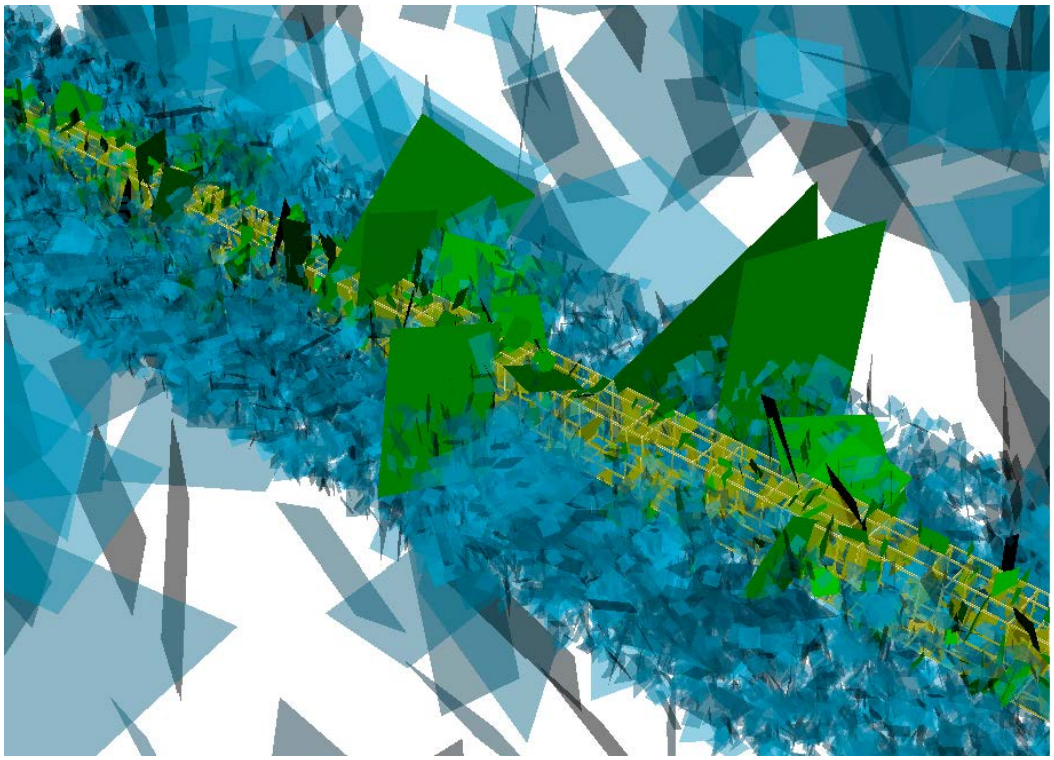

Figure 2-4. *The same realisation as in Figure 2-2 after conditioning, with the added fractures whose intersections acceptably match the observed intersections highlighted in green.*

Stating that fracture distributions are assumed stationary within a domain is thus considered equivalent to stating that it is assumed that it is possible to move fractures intersecting the engineered openings from place to place within a domain without compromising the statistical distributions of fracture properties defined by the user. Note that this assumption has no effect on other aspects of the calculation such as the closeness-of-fit and weighted selection in Subsection 2.4.2, and no assumption is made about the distributions of fractures that do not intersect the engineered openings.

Note that fracture distributions only have to be stationary in the directions in which fractures are liable to be moved. For example, if all the engineered openings are at (approximately) the same elevation, there is nothing to prevent the user from specifying a depth-dependent fracture distribution, as this depth dependence will not be affected by moving fractures at constant elevation. In principle, if both an observed intersection and its potential match lie completely within a single flat surface of an engineered opening, it may also be possible to move at right angles to the opening as well as along the opening. However, this additional flexibility has not been exploited as it does not apply to larger fractures or longer intersections (of the scale of the diameter of the opening) that are harder to match.

In addition, it is not assumed that fracture intensity will be stationary. Any difference in fracture intensity along an opening is likely to have been inferred from a difference in the intensity of intersections between the fractures and that opening. In conditioning, fractures are added intersection-by-intersection, meaning that in (for example) a region with fewer intersections, the number of conditioned fractures generated will be lower and hence the final fracture intensity in the conditioned model will be correspondingly smaller.

It would be possible to remove the assumption of stationarity using a minor modification of the approach described above. It would simply be necessary to take the distance along the engineered opening into account in attempting to find a match to an observed intersection. As indicated, it would then be necessary to use a much larger library, so that the intersections in the library provide adequate coverage of the values of the distance along the engineered opening. It would be expected that, for efficient searching of the library, bins would be defined for the distance along the intersection and combined with the bins for the other intersection and flow parameters as described in Section 2.4.

This might mean that the costs of searches in the library do not differ greatly from those for the case in which fracture distributions are taken to be stationary. However, as indicated, the computational cost of setting up a sufficiently large library would be much larger.

It is also possible to limit the assumption of stationarity to specific locations or fracture types using the concept of domains described in Subsection 2.1.4.

2.1.4 Division of the model into fracture domains

As noted in Subsection 2.1.3, stationarity is only assumed locally within a given domain. This makes the division of the model into domains a fundamental initial step in setting up a conditioning calculation, as it allows the user to use far more complicated model structure or fracture distribution than would otherwise be the case.

The aim of creating domains is to divide the library into smaller sub-libraries, where each sub-library contains only fractures that are within the same stationary distribution, and hence could in theory be moved into place as described in Subsection 2.1.3. This allows a closer match to properties in a given region of a model than would otherwise be possible, and it also prevents fractures from being moved in ways that are inappropriate (for example, it prevents a fracture found on a pilot hole from being used to match an intersection observed on a tunnel wall).

However, for all the reasons described in Subsection 2.1.3, it is desirable that the assumption of stationarity be available where possible. The more domains a given library is divided into, the fewer fractures are available per domain for a given number of fractures overall, and hence the more fractures are likely to be needed to create an appropriately-sized library.

As described in Section 1.5, a domain is defined in the context of conditioning as a region or division of the model in which fracture properties are assumed to be stationary. Thus in order to determine where it is necessary to use multiple domains it is necessary to consider the circumstances in which a fracture distribution might not be considered stationary.

The simplest such reason is a genuine difference in the fracture property distributions in different regions of the modelled rock volume, beyond a simple change in fracture intensity. This may be due to, for example, different lithology on either side of a deformation zone. If fracture properties are different in different regions, it might be more appropriate to sample from fractures generated for the specific distribution required in the library than to sample from the full set of fractures. In this case, the surfaces and pilot holes in each region should be put in a different domain. If the distribution is more complex – where fracture property distributions vary significantly over small areas, or where there is a gradual change in properties – it may be more accurate to create multiple domains that are approximately stationary, than to assume stationarity across the model.

If the only difference between the regions is a change in fracture intensity, it is not necessary to distinguish the two sides using domains because the difference will already be reflected in the different numbers and sizes of intersections observed in different parts of the engineered openings (from which distinction between the regions with different fracture properties was likely inferred).

There are other instances where stationarity may not be appropriate. It cannot be assumed that the same fracture – unrotated – will create equivalent intersections on two different engineered openings that are not parallel, do not have the same cross-section or do not have the same radius or size. For example, it cannot be assumed that a fracture generated on a deposition tunnel will, when moved into place, make the same or a similar intersection on a deposition hole whose axis is perpendicular to the tunnel or whose radius and cross-section are significantly different. Thus where there are substantial discrepancies in shape, size or orientation between different engineered openings, the features will have to be put into different domains according to those characteristics. The intersections shown in Figure 2-5 are divided into domains in this way.

Similarly, when dealing with more than two engineered openings, the spacing between the openings should ideally be consistent, as in the case of deposition tunnels. This will prevent the model from attempting to link two openings with a fracture that does not reach, because in the library in which the fracture was created the spacing between two openings it intersected was smaller.

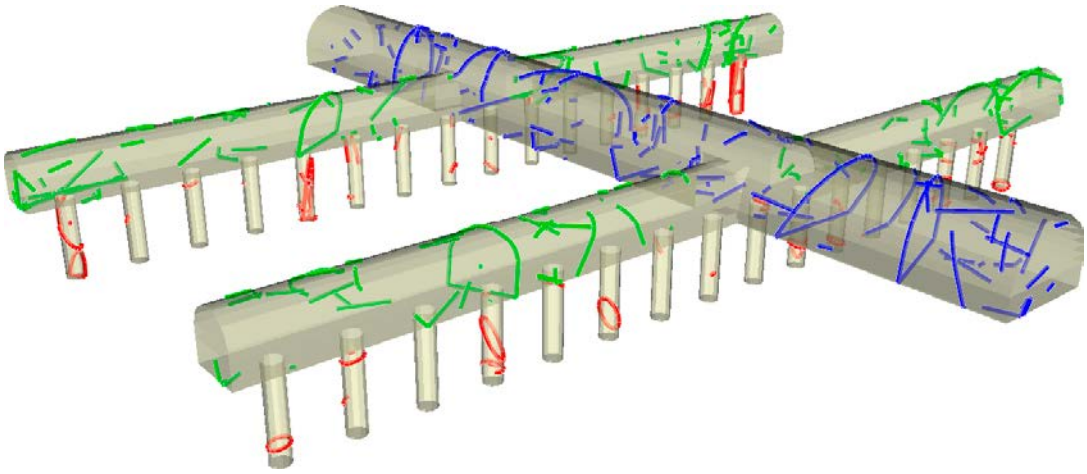

Figure 2-5. *Intersections found in a realisation of HypoSite (see Subsection 1.3.1 and Section 3.1), coloured by domain. The main tunnel, the deposition tunnels and deposition holes are each a separate domain.*

Other cases where stationarity may be considered inappropriate may be determined according to the choices of the geologists mapping the engineered openings or by the modeller. For example, if the precision to which the openings are mapped varies (so that, e.g. short intersections are mapped in some parts of the opening but not in others), it will be necessary to maintain a division between these areas in order to ensure that the specified fracture intensities are maintained. Alternatively, if certain intersections have been identified as associated with a specific deformation zone, the user might create a separate domain for fractures in that deformation zone (defined according to the fracture set) to ensure the deformation zone fractures are not used in other parts of the model.

The mechanics of splitting the model are relatively simple. Each panel, and therefore each intersection in the library and observed data, is assigned a domain index. Library fractures are only allowed to match observed intersections where the domain indices match.

When domains are input into the model, there is a check that openings in the same domain are parallel (to within 5°) and apply the same physical concept (i.e. point intersections on scan lines or line intersections on surfaces of engineered openings). However, there is no other attempt made to confirm that fractures within a domain are stationary, and the further from the ideal situation the model domains are, the more likely it is that errors will be introduced because stationarity has been assumed where it does not apply. For this reason it is a good idea to visualise a sample set of intersections by domain to ensure that the domain structure chosen is correct.

It is possible that some observed intersections will cross the dividing line between the domains, or that fractures with multiple intersections will include intersections on different domains. These will have to be matched to library fractures that have a similar arrangement of fractures. Measures are taken to prevent fractures from being moved into locations where they create unwanted connections between domains (e.g. at junctions between non-parallel features) or between separate engineered openings in the same domain.

2.2 Types of observations considered

The types of observations that can be considered by the conditioning method differ between observations in pilot holes (i.e. openings modelled as scan lines) and observations mapped on surfaces of tunnels and holes (i.e. openings modelled as voids in three dimensions) As noted in Section 1.3, when conditioning is used with real-world data, observations will almost certainly have to be converted into a form that is suitable for modelling, but this process is not within the scope of this report.

When the intersection is on a modelled panel, the only *essential* information is:

- Start and end points of each fracture intersection segment on each surface panel.
- An identification number for each intersection.
- The domain that the intersection is associated with (where applicable).

Similarly, when modelled as an intersection with a pilot hole, the only required information is:

- Location of each intersection between a fracture and the pilot hole.
- An identification number for each intersection.
- The domain that the intersection is associated with (where applicable).

It is useful to have additional information, if it is available, as this allows the algorithm to more accurately match the observed data. If only the intersection location is available – particularly where this is modelled as a point intersection – there is no information about the extent of the underlying fracture, nor of its orientation or flow potential, and thus no reason why the algorithm should guess these things accurately.

Additional pieces of information that can be used in both cases are:

- Dip angle (or plunge) and dip direction (or strike, or trend).

  In the case of fractures intersecting pilot holes, it is anticipated that it will be possible to determine a dip angle and dip direction from each fracture intersection. For fractures intersecting surfaces, a dip angle and dip direction can be inferred for those fractures that intersect more than one planar section of the surface, as these provide three non-colinear points on the fracture plane.

- A measure of the flow through the fracture.

  There are three forms of flow measure that can be provided. Firstly, it may be a simple connectivity indicator. A connected fracture is a fracture that provides a flow passage into the engineered openings, even if the flow associated with it is negligible: this may be detected for example through observation of flows or a visibly open aperture, by the presence of minerals such as calcite that indicate hydraulic activity, or by staining on the wall of the opening. In modelling terms, a connected fracture is a fracture that is hydraulically connected (either directly or indirectly) to an external model boundary, providing a flow route from that model boundary to the engineered openings; a connected fracture is flagged 1, and an unconnected fracture is flagged 0. This requires no flow calculation in the library generation phase (and thus may be quicker than other methods) but in practice connectivity is likely to be difficult to determine in the field (because a connected fracture may have negligible flow in practice) and likely to give less satisfactory results than conditioning using flow directly.

  Secondly, it may take the form of a flow category. A flow category would be mapped according to a number of qualitative assessments, for example, “dry”, “moist”, “dripping”, and “flowing”. There can be any number of categories, each assigned an identification number starting from zero, increasing according to increasing flow. However, using this semi-quantitative method means that the flow through each DFN realisation used to build the library will have to be calculated, and there will need to be some means of determining what quantity of flow (in cubic metres per second or litres per minute) implies each category.

  Third, the user can associate a quantitative measurement of the flow to each fracture intersection. The flow measurements used must be consistent within a given domain, but beyond that may result from any simulation that can be performed using ConnectFlow. For example, this may include the result of an injection test, or an inflow measurement. As with flow categories, a flow through the model must be calculated for every realisation used to build the library. Note that the units of the calculation in the library may need to be transformed if they are different from those used in the observed data.

  Note that in practice all of these measures are likely to be based on steady-state flows that can be characterised by a single number. The use of transient data is discussed in Section 4.5.

- Linking intersections in multiple locations.

  While it is allowable not to interpret links between intersections where a single fracture intersects the engineered openings in more than one place, conditioning results will be better if these links have been properly evaluated. This is because the links between the intersections provide valuable information about the size of a fracture.

  If the user specifies that a fracture intersects multiple engineered openings, or the same engineered opening in multiple domains, the algorithm requires that those connections be respected in the final realisation. This is important because these are generally the largest fractures, i.e. those that are most likely to conduct flow and that will play a significant role in safety assessment. Honouring these connections forces a fracture to be chosen that is large enough to fulfil this role in the conditioned realisation.

  However, there is no equivalent restriction if observed intersections are not linked. Fractures that have intersections in multiple engineered openings will be considered alongside those without any link, provided that the additional intersections created by the candidate fracture correspond to other observed intersections or are shorter than the relevant detection threshold for observed intersections. The effect of this is that, if the user is not able to link intersections, the conditioning algorithm may do it anyway, but this is not guaranteed, so it is preferable that these links be explicitly specified.

In general, where injection tests are taken into account in conditioning, this is called “hydraulic conditioning”; conditioning where inflow measurements are taken into account is “inflow conditioning” and conditioning where flow measurements are not taken into account (i.e. where all the data used is geometric), this is geometric conditioning.

2.3 Generating the library

The library is generated from a number of realisations of the fractures in a region containing the engineered openings, using a DFN recipe consistent with site data. The number of realisations that need to be created depends on the quality of the match to observed intersections that is desired and is influenced by factors such as the nature and complexity of the underlying DFN model and the layout of the engineered openings. As an initial recommendation, it is suggested that several hundred to several thousand realisations should be used, more if the observed data contains fractures that are likely to be difficult to match. More realisations are likely to give a better match to the observed intersections, but the computational cost of creating the library is proportional to the number of realisations used.

A suitable choice has to be made for the region in which library fractures are generated. This needs to be sufficiently large that it can contain the centres of the largest fractures that may intersect the engineered openings. If the library is to contain information about flow, the model in which flows are calculated (which may be different in size to the fracture generation region) needs to be large enough and have enough detail to allow accurate modelling of the flow conditions in the observed fractures. However, both the generation region and the calculation region need to be small enough to ensure that it is practical to generate the number of realisations required in a reasonable amount of time. To improve the efficiency of the generation process, one might choose to generate small fractures only in a sub-region around the engineered openings.

For each realisation, each fracture is checked to see whether it intersects the walls of the relevant engineered openings. If so, the fracture and its intersection are added to the library. Sufficient information is included in the library for each intersection/fracture combination to completely characterise the fracture and to enable the fracture to be regenerated and inserted into a conditioned model.

There are some factors that are taken into account when generating the information required for a library:

- In ConnectFlow fractures may be tessellated, i.e. divided into a number of sub-fractures, which are effectively treated as fractures in their own right. It is important that the library refers to the

original untessellated fracture rather than to one or more sub-fractures. Then, in the process of conditioning, the original fracture can be added to a realisation, and subsequently tessellated.

- Where intersections are modelled on the surfaces of an engineered opening, depending on the shape of the fractures and the cross-section of the modelled opening, a single fracture may lead to several unconnected intersections on the surfaces of that opening. For example, for an opening with rectangular cross-section, a planar convex fracture may lead to 0, 1, 2, 3 or 4 unconnected intersections because it is oriented differently from the cross-section of that opening. It is important to ensure that the link between a fracture leading to multiple unconnected intersections and the intersections themselves is recorded in the library. It is then possible to check whether several intersections correspond to the same underlying fracture.
- An intersection mapped on a series of surface panels may form a continuous ring around an engineered opening. In this case, the distance along the opening from a given reference location to each point on the intersection is measured, and the start and end points of the intersection are taken to be the points with smallest and largest distances respectively.

In ConnectFlow, the library is stored in a data file, whose format is detailed in Appendix D.
A description of the input commands and keywords used is given in Appendix C.

2.3.1 Use of multiple libraries

If so desired, the user may create multiple library files, potentially containing different fracture size ranges. For example, the user may choose to generate one library containing a small number of realisations including all of the fractures in the model, and a second library containing a large number of realisations but that excludes the smallest fractures in the model. By doing this the user saves computing time in library generation while ensuring that a reasonable sampling of the largest fractures in the model is available.

The same method may also be used to effectively parallelise the generation of large libraries (if identical fracture size ranges are used). This will significantly reduce the time needed to generate large or complex libraries.

When generating these libraries, it is important to be aware that the flow field in the model may be affected by the absence of some fracture size ranges. If the larger fractures – the main flow conduits – are not included in a particular model, the difference in the flow field may be very significant. Thus, if flow or connectivity attributes are to be used in conditioning it is good practice to truncate from the small end of the size range only, so that the largest fractures are included in every realisation of every library. As they are also likely to be in the tail of the distribution, and may intersect the engineered openings in a significant number of locations (thus constraining their use), it is likely that the modeller will want to retain as many of the largest fractures as possible anyway, in which case, the change to the flow field is not a major concern.

As noted in Appendix B, the use of libraries with different size distributions will have an effect on the distribution of fractures available. To avoid biasing the fracture size distribution in the final result, one must thus add a weighting to the selection of fractures from the candidate list, based on the size of the fracture and the number of library realisations available in each size range. For this reason, the user must in this case specify the size range of the fractures in the library and the number of realisations used in each library.

2.4 Selecting a fracture

In order to include a fracture from the library, the library is first searched for a list of suitable candidates. This is completed through an efficient binning algorithm to allow the closest matches to be found quickly without need to search the entire library. Once a list of at least 10 suitable candidates has been found (or the entire library has been searched), a candidate is chosen from the list by random selection, weighted based on a closeness-of-fit function that measures the similarity between the (implied) observed fracture and the candidate fracture. Finally, the selected candidate is added to the model in the appropriate location.

2.4.1 Searching the library

The use of a library already provides a significant improvement in efficiency compared to the brute force approach outlined in Chapter 1. It means that it is only necessary to generate a large number of fractures once and to determine their intersections on engineered openings once. Effectively, the intersections (with their associated fractures) in the library are reused for comparisons with all the observed intersections in a given conditioning calculation.

Further efficiency in the approach arises because of the algorithm used to search the library to try to find a match to an observed intersection. This algorithm means that it is not necessary to compare the intersection for every fracture in the library with each observed intersection. Rather, comparisons are only made for a subset of the library.

This is achieved through a process of binning. A hyperspace of sub-ranges (bins) is created according to six key quantities for intersections on surfaces of engineered openings, or four for intersections on pilot holes modelled as scan lines. Then, the value of each quantity for each fracture in the library is calculated and the fracture is added to a list associated with a single combined bin.

When conditioning on intersections made on surfaces of engineered openings, the key quantities used in the search algorithm are (see Appendix A for definitions):

- Length of the intersection.
- Azimuthal angle of the start point.
- Azimuthal angle of the end point.
- Apparent angle.
- Connectivity/flow category.
- Fracture domain.

For pilot holes modelled as scan lines, they are:

- Strike angle of the fracture.
- Dip angle of the fracture.
- Connectivity/flow category.
- Fracture domain.

When conditioning, these same quantities are calculated for the observed intersection in the model, and the bin associated with it is found. The library fractures associated with this bin are then taken as an initial candidate list of fractures that may be used to match the observed intersection. If an observed fracture intersects the engineered openings more than once, one intersection is chosen for searching.

Provided that the library is sufficiently large and the bins are suitably defined, it would be expected that, for most observed intersections, it would be possible to find sufficient appropriate matches in this initial candidate list to allow an appropriate match to the fracture. However, it is also likely that, if a large number of observed intersections are considered, there will be a few intersections that are in some sense statistically rare. Therefore there will be some cases in which the initial candidate list is empty or does not contain enough fractures that acceptably match the observed intersection to sample from. In this case, bins associated with quantity ranges that are adjacent to the bin initially identified are searched and fractures may be added to the initial candidate list from these bins. If there are still too few candidates, the algorithm proceeds outwards until the initial candidate list has been suitably populated (with at least ten fractures) or the entire library has been searched, as illustrated in Figure 2-6. This provides an efficient means of finding an appropriate number of fractures for the initial candidate list while not requiring a full search of the library unless necessary.

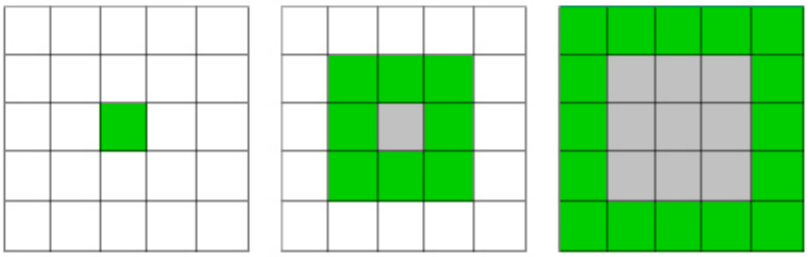

Figure 2-6. *Illustration of the expansion of the range of combined bins searched (in 2D). The green bins are those being searched, the grey bins are those already searched, and the white bins are those not yet searched. The central bin is the bin that matches the parameters of the observed trace.*

However, note that only – at least initially (see Section 2.7) – adjacent bins that are associated with geometry are searched. Only bins associated with the same flow as the target value are allowed, partly because the flow may be a discrete variable (one would not want to look for dry fractures to match an observed intersection through which water flows freely), and partly because the flow is dependent on the background fracture network and it is thus desirable to reduce uncertainty as to whether a correctly-flowing fracture has been chosen from the library. In the case of models with multiple domains, only bins associated with the same domain as the observed intersection are allowed because allowing fractures from other domains into the list would defeat the purpose of dividing the model into domains in the first place. There are a number of factors that might make it less likely that a suitable number of library fractures will be available in a given bin. Factors reducing the number of fractures in the initial candidate list with a given set of criteria include the statistical rarity of the fracture, the size of the library and the number of bins:

- If an intersection implies a fracture that is statistically rare given the DFN recipe being used, either because its properties are in an outlying part of a distribution or because it is well constrained (e.g. if it is large and intersects multiple engineered openings), it is likely that the number of fractures with similar properties in the library will be relatively small.
- A larger number of bins used to divide the library will result in a smaller number of fractures in each bin. In this instance there may be many fractures that are well-matched to the intersection in question in bins other than the first, and those bins may need to be searched to create a long enough initial candidate list. This is not a cause for concern.
- A small library is also likely to have fewer fractures in each bin than a large library.

The fact that multiple bins are searched for some fractures is not a cause for significant concern on its own. It is not implausible that some rarities will occur given a large number of intersections. However, if a large number of observed intersections are statistically rare, that might suggest that the DFN recipe inferred is not suited to the observed data. Alternatively, and particularly if significant numbers of bins have to be searched in order to generate initial candidate lists for statistically common intersections, or if there are no matches at all in the library for some fractures, it is likely that a larger library would be of benefit.

ConnectFlow has an option that gives details of the progress of the search algorithm, including the bins being searched and the number of fractures in the initial candidate list, and also allows the user to print the number of fractures in each range for each key quantity in order to assist in determining whether a library is large enough for a given model. This functionality may also assist in finding mistakes in the conditioning input.

2.4.2 Selecting a candidate

When an initial candidate list for an observed intersection has been compiled, it would be possible simply to select at random one of the intersections and its associated fracture from the candidate list and take this to match the observed intersection. This would provide a crude match to the observations, which could well be adequate. However, it is better to weight the selection based on the quality of the match between each fracture in the initial candidate list and the observed intersection when selecting a suitable match.

The candidate is selected by a weighted random selection based on a measure of closeness-of-fit, described in detail in Appendix B. In broad terms, the closeness-of-fit brings together a selection of measures of similarity between the trace observed and the trace that would be generated by a given library fracture after it is moved into position. The measures included (defined in more detail in Appendix B) are:

1. Distances between the end points of the intersections.
2. Difference in length between the intersections.
3. Difference in azimuthal angles of the end points of the intersections.
4. Difference in apparent angles of the intersections.
5. Relative distances between the end points of the intersections.
6. Relative difference in length between the intersections.
7. Difference in flow found in the intersections.
8. Angle between the fractures implied by the intersections (or, if not available, between the intersection lines).
9. Component along the tunnel of the distances between the end points of the intersections.
10. Distance between the mid points of the intersections.

In each case a small value indicates a good match and a large value a poor match. The user may choose how to weight the different measures, and to set limits on the allowed variation for each measure individually: this allows the user to prioritise those aspects of the calculation that are considered most important, and to set hard limits on what degree of difference between the initial observation and the final fracture is permissible. If one of these limits is exceeded, the fracture will not be considered.

It is obvious that many of these measures are closely related and it is likely in practice that some measures will be zero-weighted (thus, removed from the closeness-of-fit entirely); this applies particularly in the case of intersections with pilot holes modelled as scan lines as the first six measures are trivial in this case.

The weighted measures are then combined to create a single value, an overall closeness-of-fit where a small value indicates a good match and a large value indicates a poor match. The user will also set a hard limit on the maximum value of the combined closeness-of-fit measure to ensure that each fracture added model is of a minimum acceptable quality. The overall closeness-of-fit for each candidate fracture is then converted to a weight and one fracture is selected from the list randomly.

Note that there is nothing to prevent the same fracture from being chosen from the library to represent multiple different intersections in the observed data, other than the probabilistic nature of the choice, given the size of the library and the fact that the stopping criterion for the search for the initial candidate list is 10 fractures. Because fractures are not necessarily created in the same place in the observed data, choosing the same library fracture to represent two similar observed intersections does not necessarily cause unbalanced or inappropriate models. However, if this happens too often the fractures in the model may become unrealistically uniform in terms of distribution of fracture size, orientation and flow potential; this may indicate that the library needs to be enlarged.

2.5 Considerations with multiple engineered openings

Although the above description of the method applies generally, there are specific considerations to be taken into account when the approach is applied to multiple engineered openings as opposed to a single opening. This may occur in the case of a single domain of parallel openings, or in the case of a network of non-parallel openings, potentially also including pilot holes modelled as scan lines. The method allows all of these features to be considered in the same conditioning run.

A first key consideration is the question of the domains. As described in Subsection 2.1.4, domains are sub-divisions of data where the fractures and intersections are considered to have the same

statistical properties. An observed intersection given a particular domain index can only be matched using a fracture from the library that has the same domain index, thus allowing the user to define multiple different stationary distributions in the same model.

When matching a fracture on multiple engineered openings, the domain on every single intersection has to be matched individually. In some instances this may significantly hinder the application of the assumption of stationarity: if it is known that a fracture intersection is at the join between two or more perpendicular openings, it is very likely that the matching fracture will need to have detected a similar arrangement of intersections in the library and thus will have to have been generated in the same place in the library.

A second key consideration is the issue of fractures that intersect multiple engineered openings. The knowledge that a fracture intersects multiple openings implies significant information about the size of that fracture, which makes it harder to match but also makes any match found more likely to be accurate. In the field these links may be determined using, for example, tunnel seismic data, radar data, the mise à la masse method or hydraulic interference tests.

Ideally, it would always be possible to determine whether a fracture intersects multiple engineered openings, and where this is in fact possible, the conditioning will take full advantage of this information. If the user has recorded that an observed fracture intersects a certain set of openings, then the fracture chosen to be match to it in the model must intersect at least the same set of openings. If the openings are divided into domains, the domain at each intersection must also be respected. However, if a fracture is found that intersects multiple openings, correctly predicting (within limits of closeness-of-fit) the locations of intersections that have not been explicitly marked as belonging to the same fracture, it will still be accepted as a potential match and may be selected. If a fracture is found that intersects multiple openings, creating intersections where there are none found in the observed data, that fracture is discarded unless the intersections in question are shorter than the minimum detectable intersection length (if any) specified by the geologists.

In practice, this means that, if it is not clear whether fractures intersect multiple openings, links may be inferred some of the time, but it is not possible to rely on every link between openings being found, and it is not certain that when links are found that they will reflect real links on the site or in the data. It is likely that the size distribution of fractures may be biased somewhat as smaller fractures are chosen that do not link when they should, particularly if the orientation and other properties of the fractures found at each end do not match. Ultimately, knowledge of the links between engineered openings removes a source of uncertainty and thus increases confidence in the final conditioned result, and hence also increases confidence in any calculations (such as measures of repository performance) arising from that result.

In multiple engineered openings, as in a single opening, the closeness-of-fit for a fracture that creates multiple intersections is considered to be the largest value of all the intersections concerned. It may also arise that a given candidate fracture, when moved to its proposed location, could plausibly match (within limits of closeness-of-fit) a number of observed intersections; in this case the observed intersection with the best closeness-of-fit is taken as the potential match.

It is possible for the user to create situations that simply cannot be modelled. For example, if the fracture described by the user in the observed data is not approximately planar, then it will not be possible to model it using a single planar fracture. While some tolerance is allowed (that depends on the closeness-of-fit function defined by the user), ultimately, for particularly problematic intersections, it may be necessary to split the observations up manually or as described in Section 2.7 – in which case there is a risk that the sections will not connect – or to model it deterministically.

2.6 Considerations for flow

There are certain considerations that have to be taken into account when conditioning on flow. In ConnectFlow conditioning, flow is treated similarly to other quantities defined at an intersection. The user may choose to exclude it from consideration, to categorise it depending on a qualitative assessment of the flow-rate, or to include a quantitative flow for each fracture, as described in Section 2.2. The quantitative flow may be determined as the natural inflow to the engineered openings,

or it may be the result of pumping tests, but it is required that flow be described on an intersection-by-intersection basis, and clearly the observed data and library must consistently represent the same types of measurement in the same places.

It is likely that not all fractures will have flow, as some will be isolated from the wider fracture network. It is also likely that some fractures will have a flow arising from a connection to the fracture network, but that that flow will be too small to be practicably detected. When determining how to treat flow in the model, the user may specify a detection limit, where flow below the detection limit is treated as no flow at all.

It is likely that the user in a real-world scenario will have flow data in some places and not others, or may have flow data from different sources in different places. This does not cause a problem so long as different kinds of flow data are not mixed in the same domain.

In order to correctly calculate flow for a library, each realisation of the model used to generate the library must include a flow simulation that reflects the measurements observed, and thus must include enough fractures and appropriate boundary conditions to ensure that the statistical distribution of flows that occur in the observed data, along with any variation in measurement between domains, is reasonably matched. If flow is not simulated, it is possible to skip calculation of intersections between fractures to save time in generating the realisations for the library; this option is not appropriate when conditioning with flow. If observed flow data is available in some domains and not others, it may be necessary to post-process the library to provide appropriate flows in those domains where flow data is not available. For example, all flows in domains where no data is available might be set to zero in both library and observed data. Note however, that all flows calculated for intersections in the same domain must be comparable: if different techniques or calculations are mixed in a domain, it will be impossible to distinguish the different techniques, potentially resulting in a bias in the results.

As with other quantities, flow is included in the closeness-of-fit, but unlike other quantities, the flow category or bin assigned to a given fracture must be respected unless an observation is made for which no suitable fracture can be found (see Section 2.7). This effectively means that there is a different library of fractures for each level of flow, though a given fracture may have different flow in different places. Because transmissivity, and hence flow, are likely to be correlated with fracture size, the size distributions of fractures in each of these subsections of the library are likely to be significantly different. This does not mean that quantitative measures of flow are not useful, as the difference in between flows through the intersection in the library and the intersection observed is still included in closeness-of-fit and hence used as a weighting (see Section 2.4.2).

Even given these additional requirements, it is not certain that the flow will be correctly matched in the final conditioned realisation because the flow depends critically on the distribution and connectivity of fractures away from the engineered openings, which cannot be inferred with certainty from observed data (unless there is additional data such as high-resolution tunnel wall seismic data). For example, if flow in a small observed fracture relies on a large transmissive fracture parallel to the engineered opening, the algorithm may pick a similar small conditioned fracture and fail to register the appropriate flow because there is no equivalent to the large transmissive fracture in the conditioned model. This may even apply if the large transmissive fracture itself intersects other engineered openings (and hence is added through the conditioning) as the intersections found will not fully define the fracture. A fracture may be added to represent the large transmissive fracture that is suitably large and transmissive but that extends in the wrong direction to make the necessary flow connections, for example.

As will be shown in Chapter 4, it is nonetheless useful to condition on flow, because of the relationship between flow and fracture size. It is often assumed that transmissivity scales with size for mechanical reasons (see Klimczak et al. 2010). Even if this assumption is relaxed, flow focuses toward large fractures even of moderate transmissivity. A highly-flowing fracture is thus likely to be larger both in the observed data and in the library, and consequently the flow indirectly adds information about the observed fractures at a distance from the engineered openings.

It is also possible, with any method of conditioning, to compare the results when flow is calculated in the conditioned realisations with the flows observed and determine which realisations give the best match to the observations. These in turn are likely to be better matched to the wider fracture network and thus may provide better predictions of the quantities of interest. In order for this to work, a large number of conditioned realisations need to be calculated for the same observed data (but using different combinations of the unconditioned far-field fracture network and conditioned fractures), and a means has to be devised of determining which of the realisations give a sufficient match to the observed flows. This is not a trivial point: different selection methods are possible that may give a different choice of realisations and hence a different set of results to work with. In this work, a selection method is chosen that selects ten realisations from a pool of thirty in a way that emphasises accurate measure of the large flows. This method is exemplified in Section 4.3. However, a larger pool would be better as it would provide more options, and it is likely that a variety of methods should be considered in a real situation.

2.6.1 Considerations for hydraulic injection testing

One method of obtaining flow data is through hydraulic injection testing, as performed in Hjerne et al. (2016), whereby each pilot hole for a deposition hole is pumped, one at a time. In order to model injection testing, the flow injection is treated as a boundary condition, and it is necessary to simulate the flow for a given realisation separately for each pilot hole, which may involve several simulations for each realisation.

As this is necessary for an individual realisation, so it is also necessary for a library containing a large number of realisations, a process that is likely to prove exceedingly time-consuming, particularly if (as was the case before this work was completed) the entire system of equations has to be regenerated from scratch for every individual pilot hole simulation. Where the only difference between the simulations is in the boundary conditions, most of the work involved is redundant as the equations are identical until boundary conditions are applied.

As part of this work therefore, the flow simulation has been optimised to allow the user to run repeated flow calculations on the same model with different boundary conditions, while retaining those parts of the equations that do not change from simulation to simulation. This significantly reduces the time necessary to simulate the injection testing, and also provides a more convenient input command structure for this type of calculation.

To make use of this, the user first determines which boundary conditions will vary from solution to solution (for example, the injection tests), and which will remain constant for all solutions (for example, the external boundaries). Those that will vary are each assigned to a “group” using an index number; each group denotes a different calculation (i.e. a different injection test). The equations for the model as a whole are assembled once, including all the constant boundary conditions but ignoring those in groups, and stored. For each injection test, the associated boundary conditions are added to this stored matrix of equations and a solution calculated.

If there are no intersections associated with the boundary conditions in the group (for example, because an injection test is carried out on a pilot hole that no fracture intersects), no flow simulation will be carried out.

Figure 2-8 shows the time taken to assemble the equations for a model with a number of boreholes that ran a series of injection tests using this functionality before and after the performance improvements were implemented with different numbers of boundary condition groups. It is clear that the performance improvement is significant; in the example given, the equation assembly time was approximately 60 seconds per injection test calculation without the use of boundary condition groups, but with boundary condition groups the equation assembly time was approximately 60 seconds for the first injection test and approximately 8 seconds for each additional injection test. Figure 2-7 shows a model with two boreholes pumped in turn, demonstrating the differing flow between the two groups of boundary conditions associated with the two boreholes.

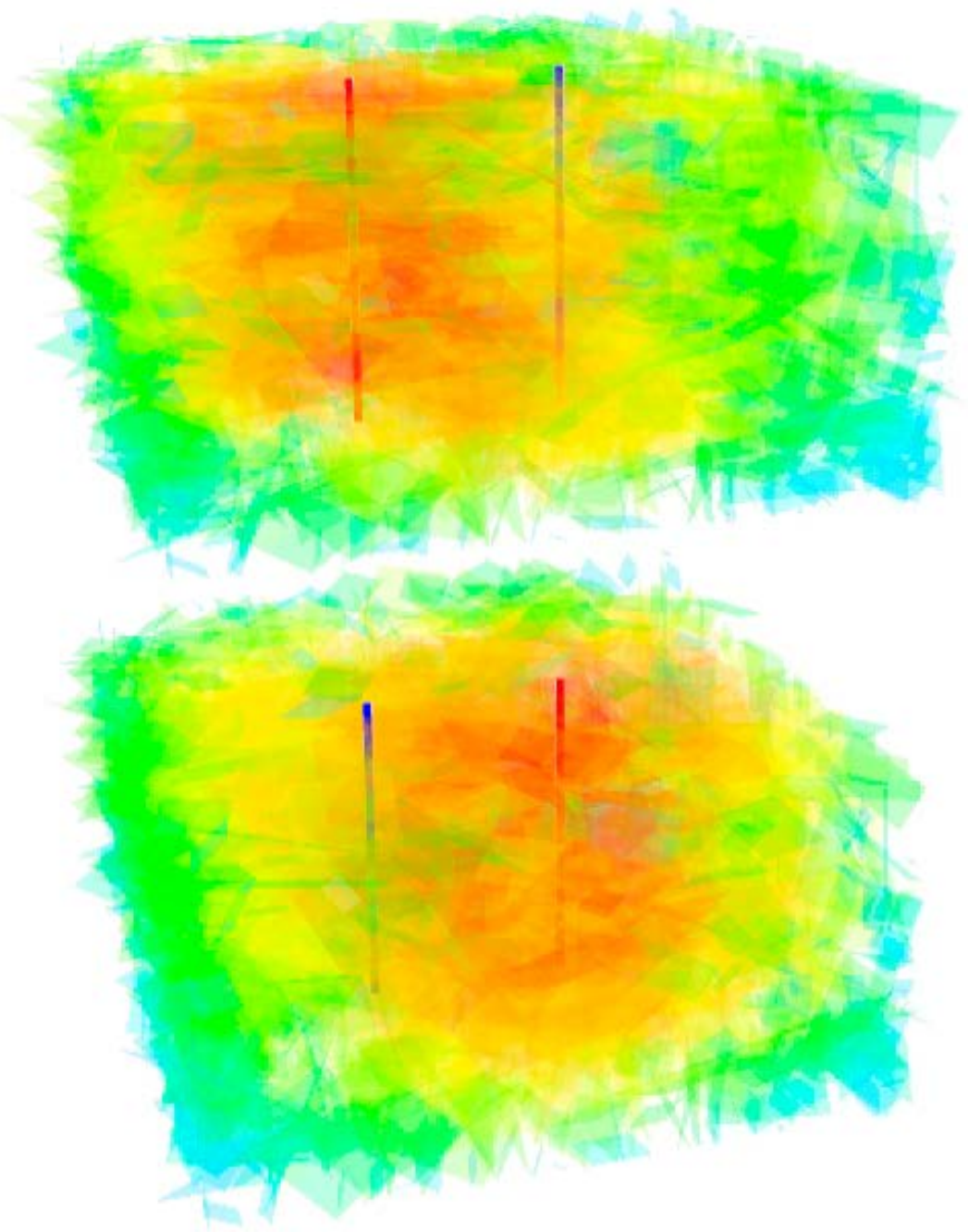

Figure 2-7. *Image of two boreholes in a random fracture network. Each borehole is pumped in turn (the left one in the top figure, the right one in the bottom), and the flow calculated. In this case, the blue areas have lower drawdown and the red areas have higher drawdown. Fractures have been made partially transparent to give a general view of the internal location of the drawdown.*

2.7 Case where no suitable fracture is found

In some circumstances, the range of bins searched might extend to cover the whole library, without an acceptable match being found. The candidate list at the end of the search would thus be left empty and so no suitable match for the intersection could be included in the model.

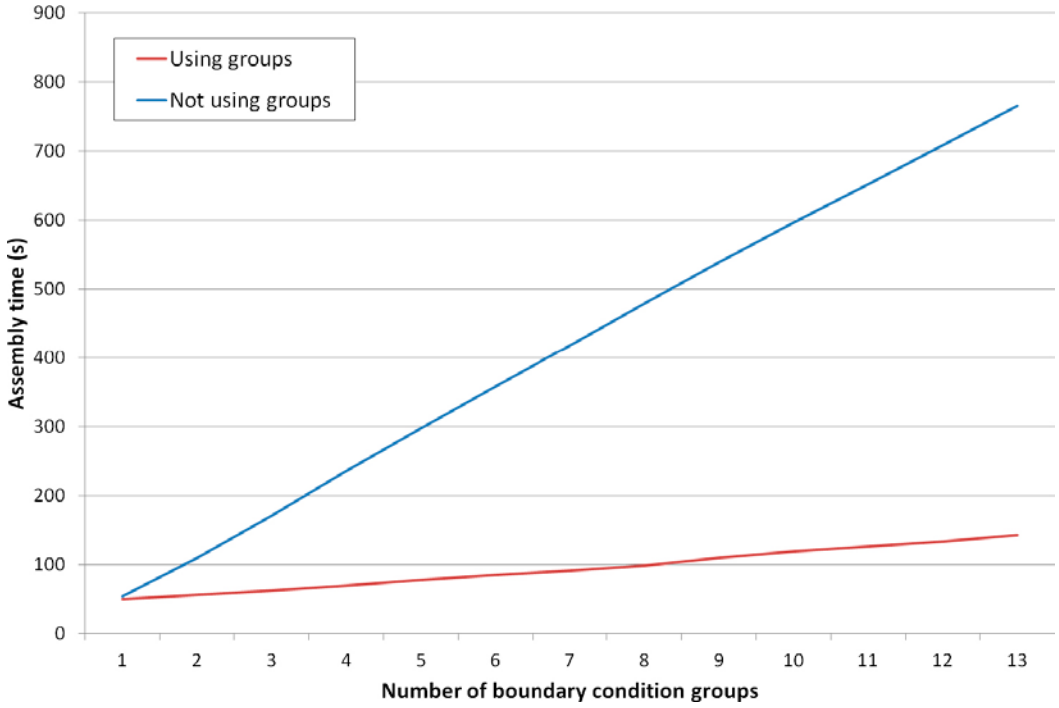

Figure 2-8. *Total time taken to assemble the equations in a series of ConnectFlow calculations for a model with 25 boreholes, with (red) and without (blue) performance improvements from using boundary condition groups.*

In general, a suitable candidate for an observed fracture is likely to be found most easily if the library is large, and the fracture properties cannot easily be inferred from the intersection or intersections associated with it. Clearly, a large library will contain a greater variety of fractures than a small one, and thus the list of suitable candidates will be larger. However, it is also possible for an observed fracture to be highly constrained in size and orientation by intersections on multiple engineered openings, and this constraint must be respected by every fracture in the candidate list from the library. This applies particularly at joins between engineered openings, where the placement of an observed intersection can effectively negate the benefits of assuming stationarity. The number of matches may also be reduced if the user defines the closeness-of-fit function and limits to require higher quality matches; although the average closeness-of-fit may be smaller, the number of candidate matches will also be smaller and may be zero.

If the entire library is searched for a given intersection and no suitable fracture is found, the conditioning process will warn the user and proceed to the next observed intersection in the list. This is likely (if the observed fracture in question creates multiple intersections) to be associated with the same fracture as the first intersection; if so, the effect that this change has on measure number 10 of the closeness-of-fit (see Appendix B) may be sufficient to allow a fracture to be found in a new search that was not found in the first search.

However, there remains no guarantee that a match will be found. Because the fractures that are least likely to be matched are those with larger numbers of intersections, and therefore are also likely to be important for determining the flow in the model, it is better that at least a partial match be found. As such, the method allows the requirements of matching to be relaxed in these circumstances to make it more likely that a fracture will be found. Once the first attempt to match fractures to all observed intersections is complete, if there are intersections that have not been matched, the algorithm will make further attempts, relaxing the requirements each time, until all fractures are matched or until all of the allowed methods for relaxing the requirements are exhausted.

There are five methods by which the requirements described are relaxed, which are applied in the following order:

1. It expands the search through the flow bins. Although (as described in Section 2.4) it is desirable to match the flow bin, if a fracture is present in the library that is an acceptable geometric match then the flow match can be relaxed. As discussed in Section 2.6, the surrounding realisation in the conditioned model will create a different flow field to that in the realisation in which the library fracture is generated and so this will not necessarily mean that the flow through the fracture is incorrect. Suitable fractures that are not identified in the initial flow conditioning are often found at this stage.
2. It allows short additional intersections to be added to surfaces (not pilot holes modelled as scan lines) that are not observed. These will be no longer than 5 % of the length of the longest surface intersection observed for the fracture. This may allow additional fractures to be found that were previously not considered because of these small additional intersections.
3. It ignores the shortest intersections with surfaces created by the observed fracture. This reflects the uncertainty in the assignment of multiple intersections on different surfaces to a single observed fracture.
4. It allows some pilot hole (scan line) intersections associated with a single observed fracture not to be matched, starting with the furthest holes from the common centre of mass of the pilot holes.
5. It treats all intersections associated with a single fracture as individual unconnected intersections. This is not entirely satisfactory, but it is not necessary in the majority of cases.

Where required, these additional steps will try to honour the user's specification for an acceptable closeness of fit.

It is emphasised that all of these measures are compromises on the requirements of the conditioning process, allowed because it is better to have a relatively inexact match than no match at all. Therefore, should any of these additional steps be required, ConnectFlow will warn the user in its output file. It is good practice before going further to check those matches that have produced warnings visually to determine their acceptability. On inspecting the visualisation, the user may decide that the choice of fractures is in fact acceptable despite not meeting all the normal requirements. If not, the visualisation may make it more obvious why problems arose. For example, if observed intersections have been projected on to places where they should not appear, or if a domain for one area is accidentally being used in a different area, the warnings might be easily resolved by correcting the observed data file. However, if fractures are not being matched that should be, the user may wish to consider the suitability of their library (whether it contains enough fractures – particularly large fractures – and whether it has the correct distributions of fractures) and closeness-of-fit measure.

In terms of the library, if the number of large fractures in particular is a problem, it is possible as discussed in Subsection 2.3.1 to include additional realisations of large fractures only, and this may assist in resolving the problem. In terms of closeness-of-fit, note that requirements that are too strict may rule out fractures in the library that, in fact, will correctly reflect the flow to a given area with an acceptable accuracy. Relaxing the requirements may therefore improve the matches found overall.

If, after all of these steps, a fracture still cannot be found for a particular intersection, a further warning will be issued, which would strongly suggest either that the library is not suitable for the model (too small or a distribution that produces intersections with properties that are inconsistent with the observed data), that there is something about the observed fracture that makes it extraordinarily rare or difficult to match, or that the closeness-of-fit requirements are too strict. Again, it is good practice to visualise the problematic intersection or intersections to determine whether the factors that made them difficult to match can be easily resolved before attempting more time-consuming fixes such as increasing the size of the library.

2.8 Calibration and testing methods

In order to provide a general means of verifying the conditioning algorithms developed and consequences for performance assessment, a set of seven calibration tests, consistency checks and flow testing methods is detailed in Appendix E.

In summary, the measures are split into three types:

- Calibration tests are checks that the conditioning method is working effectively, i.e. how well a measure that the conditioning method is actively trying to match is matched. The tests used are P_{21}, fracture location and fracture orientation.
- Consistency checks are tests that the conditioned model is consistent with the defined distributions of fracture properties. The checks used here are the P_{32} and size distribution.
- Flow tests are quantified tests to determine whether the method improves predictions of flow-related measures. The tests used here are injection tests, inflow comparisons and post-closure flow tests.

2.8.1 Fracture length per unit area (P_{21})

The first calibration target is that the model should accurately reflect the P_{21} observed. The P_{21} is the total length of intersections on a surface per unit area of surface.

Whereas P_{21} is observed on a tunnel wall, the P_{21} of the observed data input into the model is likely to be different from that actually observed on site, as it will have been idealised for modelling. For example, observed intersections may be projected from the rough walls found on real engineered openings on to flat walls used in modelling as in Baxter et al. (2016b).

While it is theoretically possible to generate an exact match to the observed P_{21} (as specified, i.e. after any projection), this was not deemed necessary as enforcing it may bias other aspects of the fracture network such as by altering the fracture orientations and sizes. Given that as the modelled intersections will only be an approximation to the real intersections, even an exact match for a post-projection model will remain an approximate match to the real intersections.

Thus, the calibration target is that the P_{21} should be within 5 % of the equivalent value in the observed data provided, or within 20 % of the variation of P_{21} between unconditioned realisations. These values are chosen so as to be large enough to not require an exact match, but small enough to demonstrate that conditioning is otherwise successfully matching the observed data to a reasonable degree. It is also possible to define P_{21} targets for sub-groups of fractures, such as those that create long intersections, providing there are sufficient fractures within the sub-group to provide statistically meaningful comparisons; this may be appropriate to reduce the effects from censoring during the mapping of short intersections, for example.

P_{21} is not relevant to pilot holes modelled as scan lines and so this calibration target is not applied. The equivalent would be the intersection count per unit length (P_{10}).

In ConnectFlow, both P_{21} and P_{10} may be reported for different types of surface (chosen by the modeller) and at different scales when writing out the intersections using the format described in Appendix D. Though there is no requirement to generate these files after conditioning, it is a useful final step as it puts the conditioned output in the same format as the observations, allowing a visual comparison as in Figure 2-9.

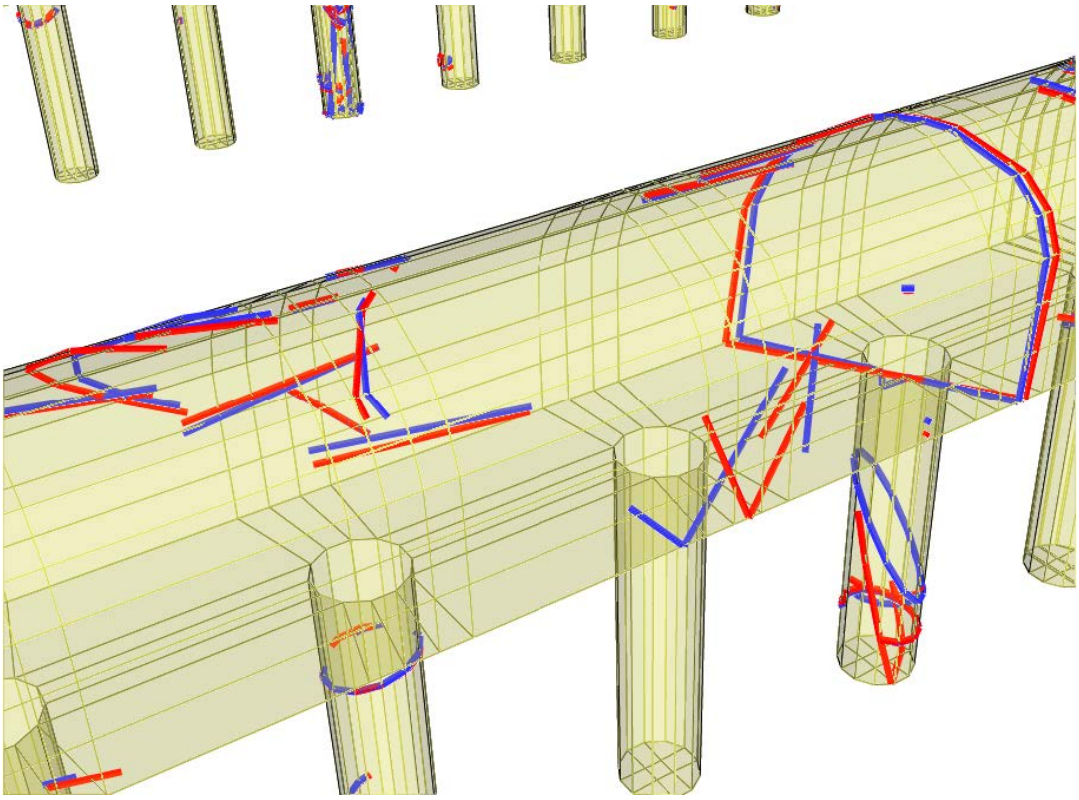

Figure 2-9. *Example of observed (blue) and conditioned (red) intersections on a length of tunnel with deposition holes. The length of each of intersection contributes to P_{21}.*

2.8.2 Fracture location and orientation

While calibration targets that apply to the whole model are useful, it is also desirable to have a measure that determines the match specifically between individual observed fractures and the corresponding conditioned fractures. In principle, it would be possible to get a perfect P_{21} without successfully matching any single fracture. Thus, the fracture location calibration target is intended to test the geometrical match between individual fractures.

There are two measures that form part of this target. They are the difference in fracture locations and the difference in fracture poles.

The first target is that the distance between the midpoints of an observed intersection and the corresponding conditioned intersection should rarely be greater than the diameter of a deposition hole; this limit was chosen as a rough indication of the point at which the error becomes significant (as a larger difference may cause a fracture to be the wrong side of a deposition hole). The second target is that the dihedral angle between an observed fracture and the corresponding conditioned fracture should rarely be greater than $\tan^{-1}\left(\frac{d_h}{l_h}\right)$, where d_h is the diameter of a deposition hole and l_h is the length of a deposition hole; if this limit is exceeded for a vertical fracture, the matching fracture will always intersect out of the side of a deposition hole instead of the bottom. An example length of tunnel with fractures shown is given in Figure 2-10.

If requested, the result from this measure in ConnectFlow is output from the conditioning process, using limits chosen by the modeller.

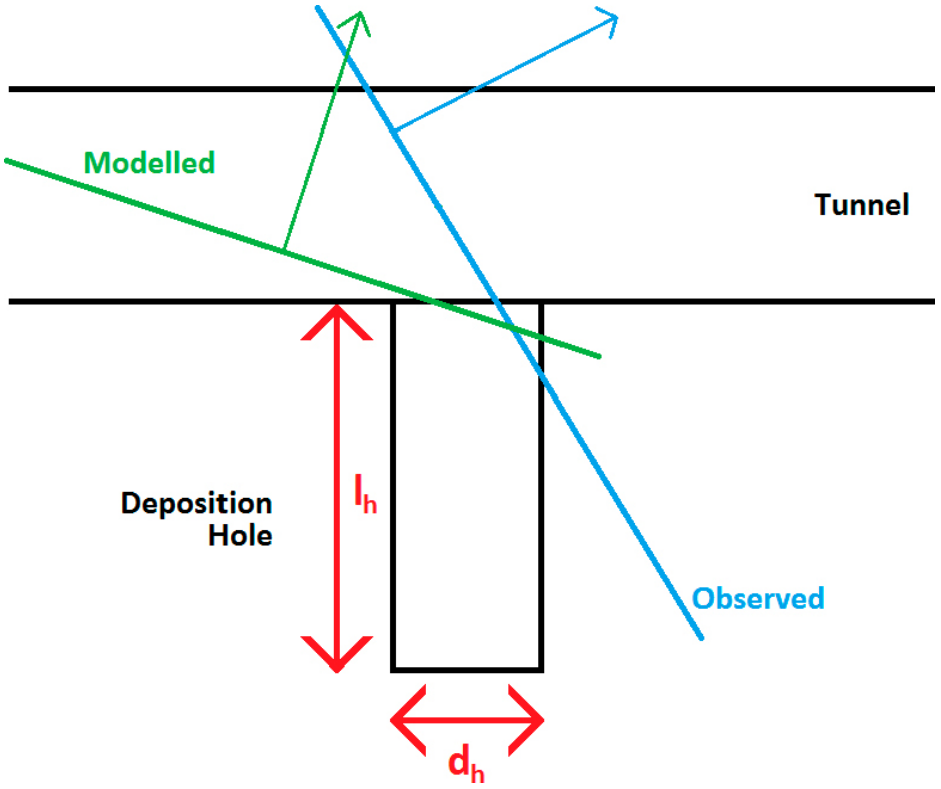

Figure 2-10. *A side-on view of a length of tunnel with a deposition hole. A modelled and an observed fracture are shown, with the directions of the fracture poles demonstrated. The lengths l_h and d_h are shown.*

2.8.3 Fracture area per unit volume (P_{32})

The aim of consistency checks is to confirm internal consistency of the conditioning process, i.e. to confirm that, given a set of library, observed and unconditioned fracture data that all share the same fracture recipe, the conditioned fracture data preserves the intended recipe in a statistical sense. Consistency checks are unlikely to be appropriate for use with real data because the quantities they calculate are not be directly measurable in the field.

The first consistency check is to compare the P_{32} in a representative sample of conditioned realisations, with the P_{32} generated from a large number of realisations using the same DFN recipe (such as those used to generate a library).

In the field, it is not possible to measure the P_{32} directly, only to infer it from borehole P_{10}, surface P_{21} and simulation. As the nature of consistency checks is to check internal consistency of the model, it is not desirable to introduce external variables or potential sources of error that might obscure errors in the model, or that might make errors appear where none exist. For this reason, it is preferable to use a number of individual modelled realisations of the DFN recipe – generated on exactly the same basis as the library and the unconditioned data – as the observed input for this check.

The target for the check is, within a small local volume around the engineered openings (the smallest cuboid containing all rock volume within 10 metres of an engineered opening, as shown in Figure 2-11), that the median P_{32} from a representative number of conditioned realisations should be within 5 % of the median of a large number of unconditioned realisations, and that no more than 15 % of P_{32} values should fall outside the range from the fifth percentile to the ninety-fifth percentile of the values from the unconditioned realisations. These targets were chosen so as to demonstrate a reasonably close statistical match, without requiring a comparison for an individual realisation and acknowledging the statistical nature of the comparison.

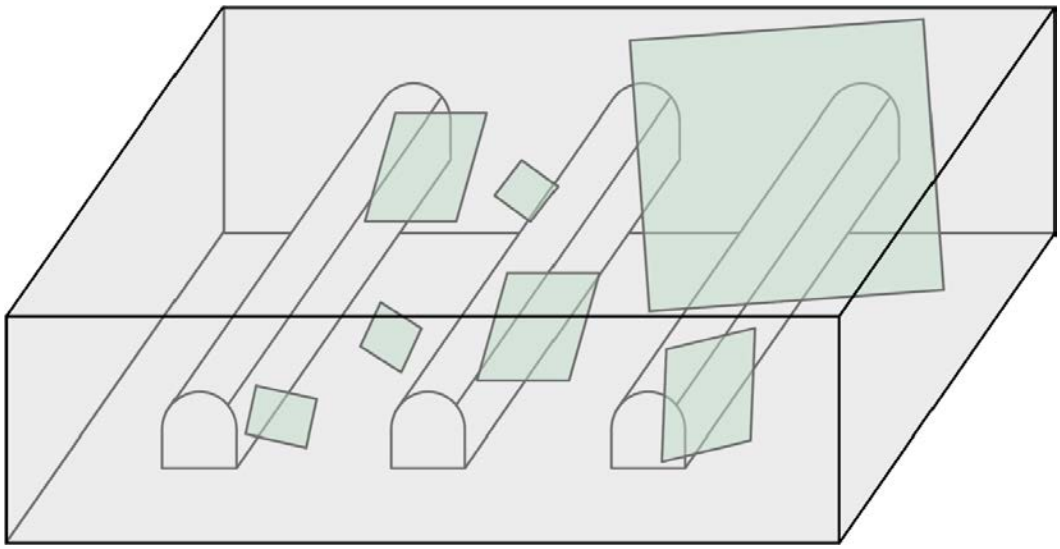

Figure 2-11. *Schematic of the region used to calculate P_{32} in the vicinity of the engineered openings. A local P_{32} is chosen to ensure that variation in P_{32} is sensitive to any changes in fracture statistics caused by conditioning.*

A local P_{32}, rather than a global P_{32} across the entire model, was chosen to ensure that the fractures intersecting the engineered openings made up a significant proportion of the fracture area within the test volume. This provides a more sensitive test of the effects of conditioning on P_{32}.

In ConnectFlow, the P_{32} may be output from the conditioning model when writing out the model intersections to a file (e.g. after calculating conditioning or when generating a library).

2.8.4 Size distribution

The second consistency check is based on the size distribution of the fractures. As the flow through the model is primarily dependent on the large fractures in the model, it is important that the distribution of fractures by size should be correctly represented.

As with P_{32}, this is a check of the internal consistency of the model. The conditioned realisation should preserve (statistically speaking) the underlying DFN recipe used by the unconditioned realisations, library data and simulated observed data. Consequently, for the same reasons, this check should also be run based on realisations modelled using the DFN recipe instead of real or simulated real data.

The target is that the size distribution of fractures found along scan lines placed in specific locations in conditioned realisations (one metre above, below and either side of those engineered openings modelled as three-dimensional voids, and along the centre-lines of pilot holes modelled as scan lines as illustrated in Figure 2-12) should be very similar to those found in the same places in unconditioned realisations (in this case, those used to produce the library). The reason for using scan lines instead of simply comparing fractures intersected by the engineered openings is to simplify the distribution somewhat; otherwise the gradient of the size distribution of fractures that intersect the surfaces will vary depending on whether the cross-sections of the openings are large or small compared with the fracture.

In principle, given a relatively simple system, an analytic distribution should be calculated for this check to use as a comparison. However, in practice it is sufficient to compare against the results from the library or a similar large number of realisations, which provides a reasonable empirical equivalent to the analytic distribution and is more practical in the general case.

In ConnectFlow, the size of each fracture intersecting each scan line is available when writing out the model intersections to a file (e.g. after calculating conditioning or when generating a library). However, in the case of the library a very large amount of data is created and so a process of scripting is needed to create usable results.

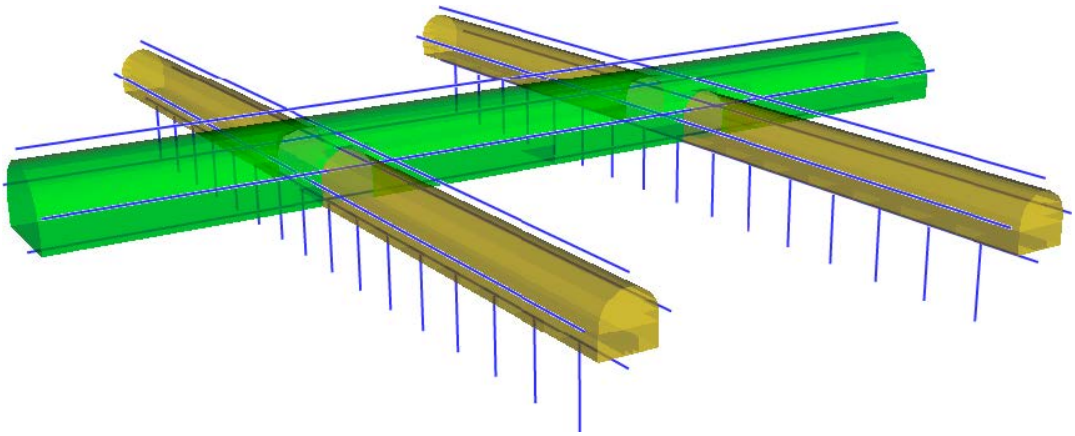

Figure 2-12. *Scan lines used to determine which fractures should have their lengths included for the size distribution consistency check in a model with deposition tunnels and pilot holes for deposition holes. If a fracture intersects more than one scan line, it is counted separately for each.*

2.8.5 Injection testing

The first flow test is based on injecting water into each pilot hole and then measuring the flow into each pilot hole to maintain a constant pressure (see Hjerne et al. 2016). This requires that each pilot hole is sealed, for example, in HypoSite each pilot hole is taken to be an isolated section starting one metre below the deposition tunnel floors. The injection tests are run to steady-state.

In principle, a successfully conditioned model will correctly predict the injected flow through each fracture into the engineered openings. In practice, as discussed in Section 2.6, the flow in a given pilot hole depends to a large degree on the details of the fracture network away from the engineered openings, of which little can be deduced based on the observed intersections and inflows. In consequence, it is highly unlikely that such a perfect solution will be realised.

The test is that the calculated flow due to injecting tests from specified parts of the conditioned model (such as pilot holes for deposition holes) should be closer to the observation than in unconditioned models. The calculated flows can then be used to screen conditioned realisations for the most acceptable matches to flow prior to calculation of (unmeasurable) post-closure flow and transport simulations.

In ConnectFlow, provided that a flow simulation has been completed using the conditioned model, the flow between the fracture network and each modelled opening is printed to the output file when writing intersections to file (the type of flow depending on the nature of the flow simulation calculated). In addition, the file itself contains the calculated flows through each intersection with each engineered opening allowing results to be processed very easily.

2.8.6 Inflow

The second flow test is very similar to the first, except that instead of using hydraulic injection tests in deposition holes, the inflow to the engineered openings through each fracture is measured under open repository conditions. The inflow, in this case, is driven by the head difference between the engineered openings and the outer boundary of the model, illustrated in Figure 2-13.

As with injection testing and as discussed in Section 2.6, the results of the inflow testing will depend to a significant degree on the fracture network in the far-field, whose geometry cannot be deduced based on the observed intersections and inflows, and so it is unlikely that the solution will perfectly match the observed inflows. However, results may be screened by comparing conditioned results with observed results, so that the best or most acceptable matches are chosen for calculation of (unmeasurable) post-closure data.

In ConnectFlow, the inflow testing results can be output in the same way as the injection testing results.

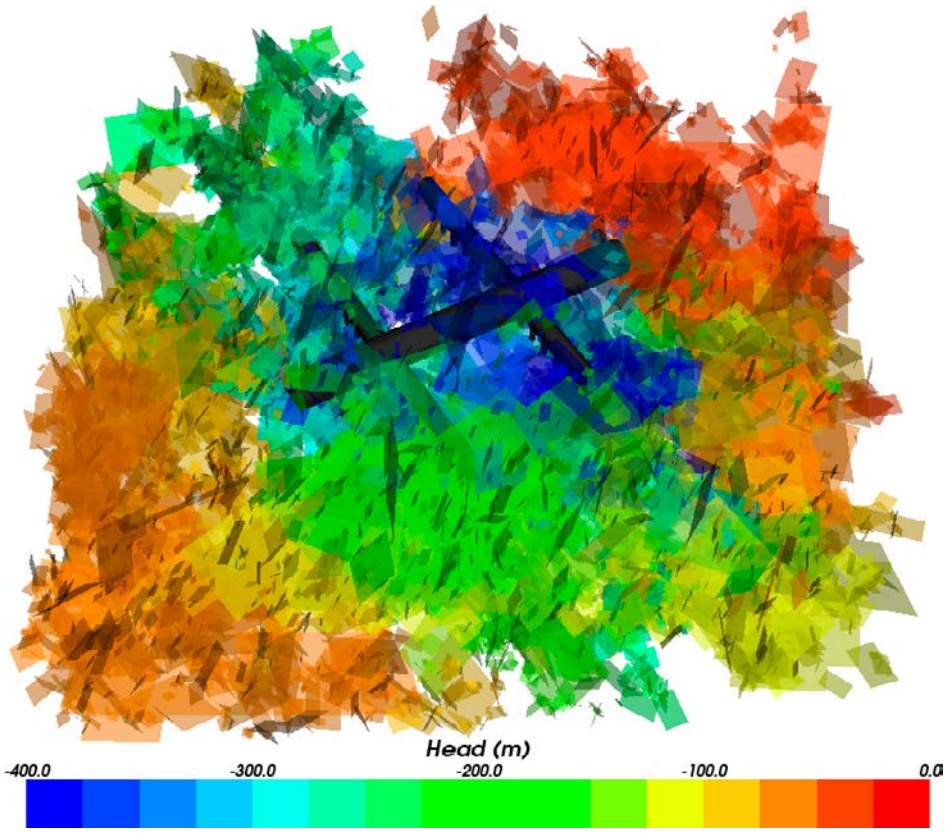

Figure 2-13. *Head in the fractures near the engineered openings in a portion of the HypoSite model. Blue areas are areas with lower head and red areas have higher head. The boundary conditions on the model are 0 metres head on all external boundaries and -400 metres head at the engineered openings.*

2.8.7 Post-closure flow

It is ultimately the flow predicted by the model representing repository post-closure conditions that is most relevant to a safety assessment. It is not possible to directly measure actual post-closure flows for a real repository, but pilot hole injection rates and pressures, inflows, and other proxies can be measured. Given a synthetic reality such as HypoSite, however, it is possible to simulate post-closure conditions in order to check how successfully the conditioned realisations predict those simulated conditions.

The third flow test is thus to use conditioned models to predict results for values such as average tangential flow-rate per unit length (U) around a deposition hole and flow-related transport resistance (F) for a model representing backfilled engineered openings, and determine the success of those predictions. In order to achieve this, a model is created whereby the HypoSite tunnels and deposition holes are represented as a continuous porous medium with properties consistent with the backfill and buffer materials, respectively. This is surrounded by the DFN realisations created by conditioning on the first two flow tests (based on inflow and injection tests) in a combined DFN/CPM model with boundary conditions that give continuity of pressure and conservation of mass at the interface between the DFN and CPM domains. Post-closure conditions are simulated assuming a linear pressure gradient on external boundaries.

For each realisation, U is calculated as the average flow-rate per unit length tangential to the repository structures. In ConnectFlow, this is determined using the Cordes-Kinzelbach (CK) mass-conserving method (Cordes and Kinzelbach 1992, Amec Foster Wheeler 2016). As is standard in ConnectFlow, each fracture (or discretised sub-fracture) is divided into rectangles, each made up of four triangular finite elements. Any intersection is mapped on to the finite element edges (though where properties such as intersection length are used, these are based on the intersection before it is mapped). The CK method further subdivides each triangular finite element into four CK sub-triangles as shown in Figure 2-14.

To calculate U, the vector transport velocity in each sub-triangle that shares a point with the mapped intersection between discretised surface element and sub-fracture (i.e. those with arrows in Figure 2-14) is calculated. These are then combined as follows:

$$U_i = \frac{\hat{\vec{l}_i}}{n} \cdot \sum_s \left(e_{t_s} \vec{v_s} \right) \tag{2-1}$$

$$U_f = \sum_i \left(l_i e_{t_i} U_i \right) \Big/ \sum_i \left(l_i e_{t_i} \right) \tag{2-2}$$

$$U = \sum_f U_f \tag{2-3}$$

where:

- $\vec{v_s}$ is the transport velocity in a CK sub-triangle [m/yr],
- e_{t_s} is the transport aperture of the CK sub-triangle [m],
- n is the number of CK sub-triangles on which flow is defined [-],
- $\hat{\vec{l}_i}$ is a unit vector along the original intersection (between sub-fracture and surface element) [-],
- l_i is the length of the original intersection (as opposed to the mapped intersection) [m],
- U_i is the flow-rate per unit length associated with the original intersection [m^2/yr],
- e_{t_i} is the mean transport aperture of the sub-fracture [m],
- U_f is the weighted average flow-rate per unit length associated with the fracture as a whole [m^2/yr],
- U is the average flow-rate per unit length associated with the deposition hole as a whole [m^2/yr].

The velocities in the CK sub-triangles adjacent to the mapped intersection (i.e. the blue triangles in Figure 2-14) are averaged, and the component of the average vector tangential to the original intersection (e.g. in the direction of the arrows in Figure 2-14) is calculated (Equation 2-1). To scale up to intersection between fracture as a whole and the deposition hole, an average of the sub-fracture/surface element intersections is taken, weighted by the contact area between the fracture and deposition hole, i.e. the product of the length of the intersection and transport aperture (Equation 2-2). Finally, the U for the deposition hole is the sum of those for each fracture intersecting the hole (Equation 2-3).

In this work, transport aperture is assumed constant over a sub-fracture (and thus equal to the mean transport aperture e_{t_i} in all CK sub-triangles). As such, Equation 2-1 could in this case be rewritten as:

$$U_i = \frac{e_{t_i} \hat{\vec{l}_i}}{n} \cdot \sum_s \vec{v_s} \tag{2-4}$$

Values for F are calculated (as described in Joyce et al. 2010) based on particle pathways originating at each deposition hole, as illustrated in Figure 2-15. The U and F values for each hole for each realisation are then compared with the results without conditioning, and with only geometric conditioning.

Significantly, in the case where the conditioning results for a stage of construction before the deposition holes are built are available, this becomes a verification test. For instance, conditioning is carried out for models based on pilot hole data, but the performance measures are calculated for back-filled deposition holes. This leaves open the possibility that deposition holes may intersect fractures that are not intersected by the pilot holes due to their larger volume. However, it is of value to be able to reject deposition holes before they are drilled, based on the measurements and predictions from the pilot holes that precede them.

In order to complete this test in ConnectFlow, conditioned fractures are read into the combined DFN/CPM model, steady-state groundwater flow is calculated and particle tracking carried out, starting particles at locations where fractures intersect deposition holes.

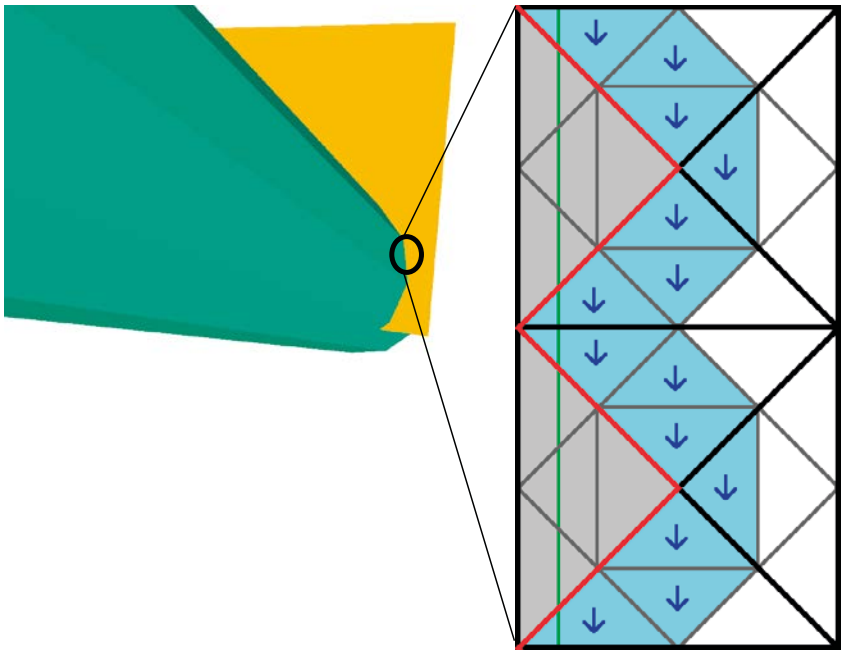

Figure 2-14. *Depiction of the measurement of U based on the flow in a small section of the intersection between a fracture and deposition hole. Left: a fracture (yellow) intersects a deposition hole (green). Right: the fracture is discretised as a regular array of rectangular blocks, each is divided into four triangular finite elements for the flow solve (bounded by black and red lines), and each of them is then subdivided into four sub-triangles for calculating the flow field and particle tracking (bounded by dark grey lines). Two rectangular blocks are depicted here. The green line indicates the physical intersection between the finite elements and the deposition hole, the red line indicates how it is mapped to the finite element edges. Grey areas are areas of the fracture that are ignored because they are within the deposition hole. Blue arrows show the tangential component of the flow vectors used to calculate U (tangential to the original intersection, not to the mapped intersection) in the blue triangles.*

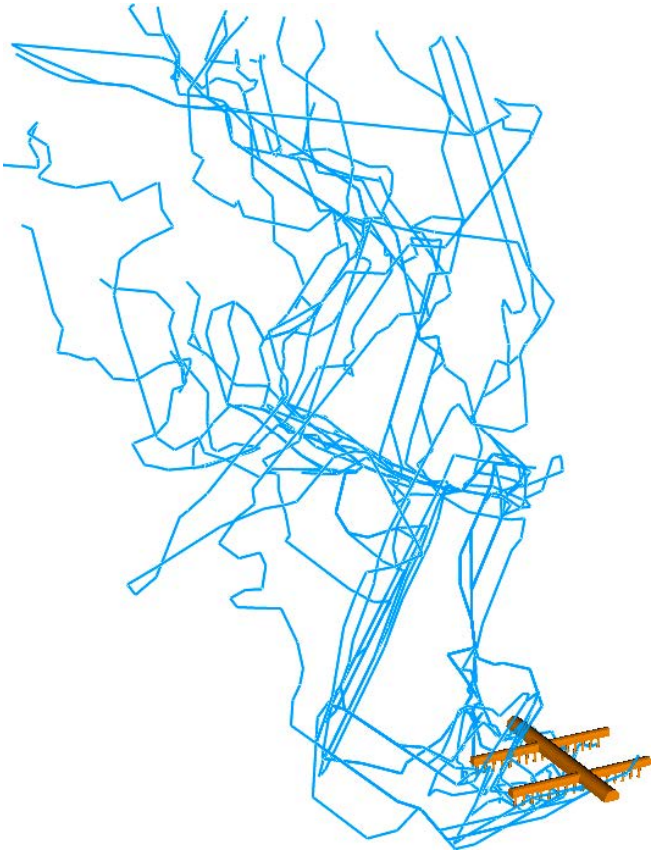

Figure 2-15. *Particle pathways originating from the deposition holes in the model. From these pathways, the F and advective travel time can be calculated for conditioned models and for the synthetic reality.*

3 Verification tests

The approach described in Chapter 2 is implemented in ConnectFlow and various tests of the approach are carried out. The tests are based on HypoSite, a model generated by SKB to reflect a synthetic reality, described in Section 3.1. All testing reported here is based either on HypoSite or on models based on the same DFN recipe as HypoSite, with the same arrangement of engineered openings.

In addition to the realisations of HypoSite provided by SKB, it is also useful to generate additional realisations based on the HypoSite model in order to provide a good statistical basis for the checks and in order to demonstrate that the model remains internally consistent after conditioning, as described in Subsection 2.8.3.

Flow-related applications of the DFN conditioning method are described in Chapter 4.

3.1 HypoSite model

As described in Subsection 1.3.1, HypoSite is a hypothetical fracture network that was generated by SKB in order to provide a synthetic reality that can be used to compare and test DFN modelling techniques. Whereas a number of versions of HypoSite of varying complexity are in development, the principal version of HypoSite used in this work is version BM-1b.

The HypoSite model region is a cube of side length 800 metres. The engineered openings within the model, depicted in Figure 3-1, are a horizontal main tunnel in the centre of the model and two horizontal deposition tunnels that cross the main tunnel at a right angle. Each deposition tunnel has 16 equally-spaced vertical deposition holes. The main tunnel is located at the centre of the model and has a flat floor but an arched roof. It is of length 100 metres, height 7 metres (to the top of the arch) and width 10 metres (at the base of the arch). The deposition tunnels are also 100 metres long and have arched roofs, but are 5 metres tall and 5 metres wide. The deposition holes are 7.833 metres long (except in models for hydraulic tests, see Section 4.2), and their cross-sections are circles of diameter 1.7 metres. In practice, in these tests, curved parts of tunnels and deposition holes are modelled as multiple planar sections.

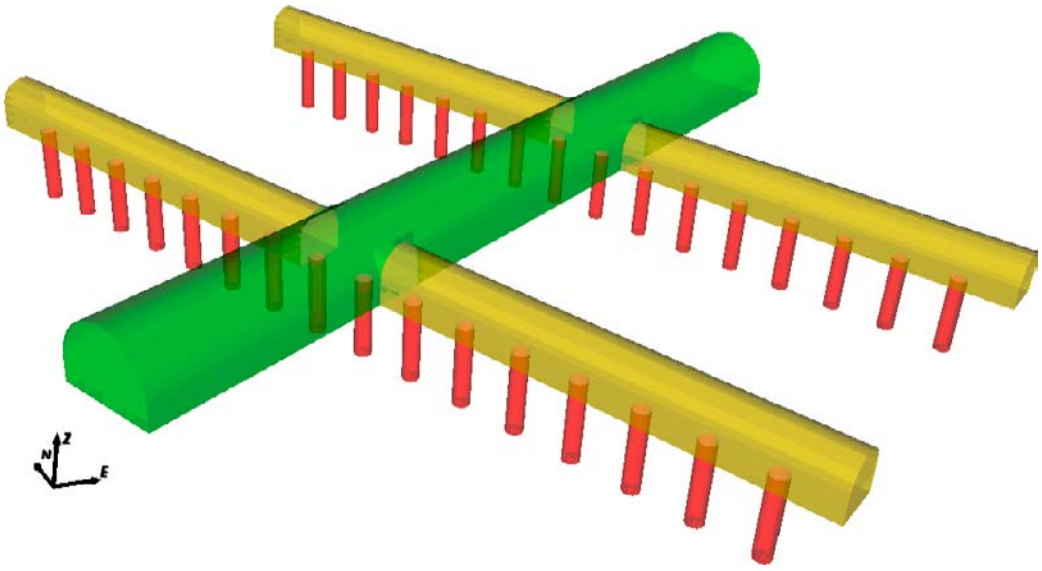

Figure 3-1. *Image of the tunnels and deposition holes in HypoSite as modelled in ConnectFlow. The main tunnel is in green, the deposition tunnels are in yellow and the deposition holes are in red. This is the same image as is found in Figure 1-1, repeated here for convenience.*

The two deposition tunnels are named NNW (the western tunnel) and NNWB (the eastern tunnel). The deposition holes are numbered from 1 to 16 from south-east to north-west, and prefixed by the name of the tunnel; for example, NNW:6 is the sixth deposition hole from the south-eastern end of NNW.

Models can be generated to represent various stages of construction, specifically:

1. Including the main tunnel only.
2. Including the main tunnel and pilot holes for the deposition tunnels.
3. Including the main tunnel and deposition tunnels.
4. Including the main tunnel, deposition tunnels and pilot holes for the deposition holes.
5. Including the main tunnel, deposition tunnels, and deposition holes.

For simplicity, these models will be referred to in this report according to the last stage of construction to be completed in the model, so for example the model including deposition tunnels is the third stage. Where pilot holes are used, they are placed at the centre-line of each tunnel or hole; the pilot holes for the deposition tunnels are placed half way between the floor and the top of the arch.

The fractures provided for HypoSite are regular dodecagons (i.e. 12-sided polygons), truncated at the model edges. They were generated using Golder's FracMan software (Golder 2012). They exist in two distinct fracture domains; the statistical fracture distributions are identical in the two domains, except that the dependence of fracture intensity on depth is different in each domain. Fractures are provided in four sizes, with smaller fractures generated in sub-regions of the model to provide additional detail in areas where it is likely to be significant to the flow paths:

- Site-scale fractures have a radius of between 10 metres and 250 metres and are generated throughout the model.
- Panel-scale fractures have a radius of between 1 metre and 10 metres and are generated in a cube of side length 200 metres at the centre of the model region (i.e. around the engineered openings).
- Tunnel-scale fractures have a radius of between 0.1 metres and 1 metre and are generated in a cuboid around the engineered openings extending 100 metres in the x-direction, 120 metres in the y-direction and 20 metres in the z-direction.
- Pilot hole-scale fractures have a radius of between 0.038 metres and 0.1 metres and are generated in the vicinity of pilot holes.

However, note that flow calculations generated as part of HypoSite in Follin (2015) ignore the contribution of any fracture other than site- or panel-scale fractures.

Ten realisations of the HypoSite model, numbered 1–10, have been provided by SKB and all of them are used for one or more parts of the assessment of conditioning reported here. Realisation 4 is identified as the base realisation for all tests because it has the largest number of pilot holes for deposition holes that are connected to the external boundary of the model by the fracture network. This realisation contains approximately 165 000 site- and panel-scale fractures, of which approximately 71 000 connect either to the edges of the model or to the engineered openings.

3.1.1 HypoSite recipe

Fractures in HypoSite are divided into three sets, each with its own intensity, size and orientation distribution. The size distribution for fractures in each set is a power law distribution, truncated at the size limits applicable to the scale of fractures being generated. The orientations are distributed according to a univariate Fisher distribution. Parameters are given in Table 3-1.

Table 3-1. Fracture size and orientation distributions in HypoSite.

Set	k_r	r_0 [m]	Trend [°]	Plunge [°]	Fisher Concentration	Total P_{32} [m^2/m^3] ($r_0 \leq r \leq \infty$)
1	2.5	0.038	45	0	10	0.8
2	2.6	0.038	0	90	10	0.65
3	2.7	0.038	30	0	10	0.5

Fractures have a transmissivity calculated based on Zimmerman and Bodvarsson (1994), which is:

$$T = \frac{1}{12}\frac{1}{\vartheta} b^3 \left(1 - \frac{1.5\sigma_b{}^2}{b^2}\right)(1 - 2c), \quad (3\text{-}1)$$

where:

- $b = \frac{1}{4}\alpha\pi\sqrt{2}\sqrt{r}$, the aperture based on Klimczak et al. (2010),
- α is a random number from a normal distribution of mean 2×10^{-4} and standard deviation 2×10^{-5},
- r is fracture radius,
- $\sigma_b = \frac{b}{10} = \frac{\alpha\pi\sqrt{2}\sqrt{r}}{40}$ (the standard deviation of b for a given r),
- $c = 0.05$;
- $\vartheta = 0.00178$.

Substituting these values into Equation 3-1 gives:

$$T = 41.5b^3 = 41.5\left(\alpha\pi\sqrt{\frac{r}{8}}\right)^3 \quad (3\text{-}2)$$

The model is divided into two domains at a plane that bisects the tunnels at the centre of the model. The difference between the two domains only affects the depth dependence of the fracture intensity, so the size and orientation distributions are unchanged. The depth dependence is implemented in fifty 16-metre depth zones such that the P_{32} in a given set s, domain d and depth zone i is calculated as:

$$(P_{32})_{d,i,s} = (P_{32})_s \cdot \frac{n \cdot (h_{i,d})^{-0.4}}{\sum_i (h_{i,1})^{-0.4} + \sum_i (h_{i,2})^{-0.4}} \quad (3\text{-}3)$$

$$h_{i,d} = \frac{dz}{2}\left(i - \frac{1}{2}\right) \quad (3\text{-}4)$$

Where n is the number of depth zones; z is the vertical size of a depth zone in metres and $(P_{32})_s$ is the overall P_{32} for set s. The difference in intensity between the two domains is illustrated in Figure 3-2.

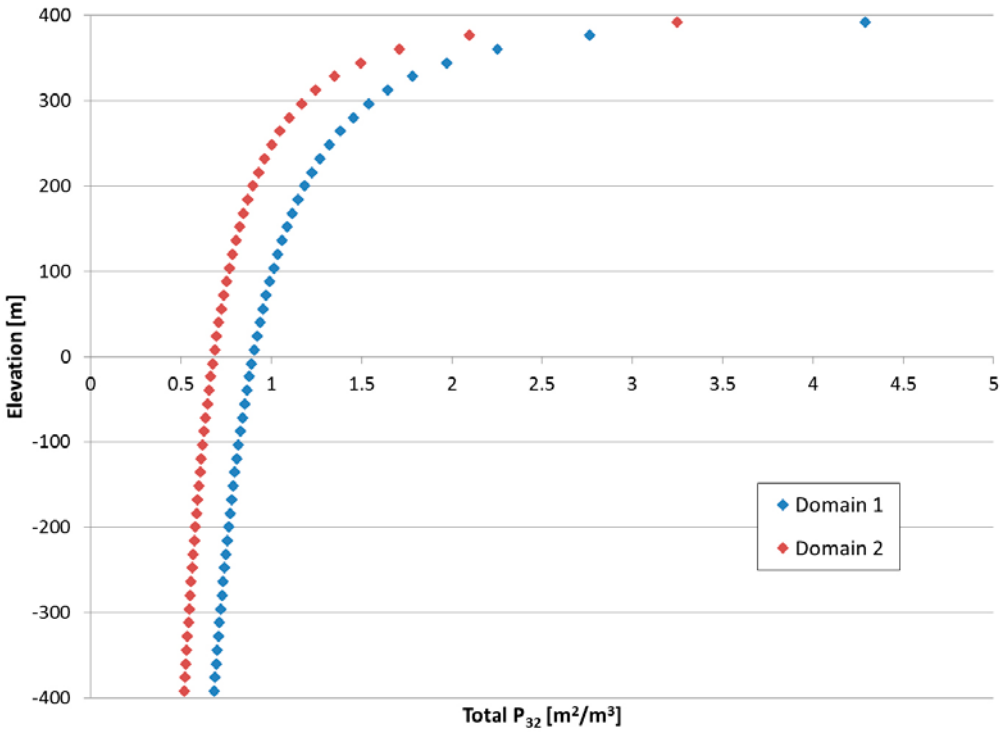

Figure 3-2. *Relationship between P_{32} [m^2/m^3] and elevation for the two fracture domains in HypoSite. The elevation is the Z-coordinate in the HypoSite model; note that the engineered openings are located at zero elevation. The P_{32} given is the sum over all three sets of the values calculated in Equation 3-3 for a given domain and depth zone, for the entire size range from r_0 (0.038 metres) to infinity. Note that for modelling, only fractures of radius greater than 1 metre were used and that the P_{32} was thus correspondingly smaller.*

3.1.2 Implementation of HypoSite in ConnectFlow

For reasons of simplicity, all models and library realisations generated for this work are based on site- and panel-scale fractures generated in the same regions as in HypoSite, i.e. site-scale fractures in a cube of side length 800 metres and panel-scale fractures in a cube of side length 200 metres around the tunnels. Smaller-scale fractures are not included because they are relatively simple to condition and for this sparse network generally do not connect.

As the only difference between the fracture domains is in intensity distribution, and as the absent panel-scale fractures outside the 200-metre cube would be too small to intersect the engineered openings, the HypoSite model is stationary according to the requirements described in Subsection 2.1.3.

Due to the computational cost of handling dodecagonal fractures in ConnectFlow, the models used for this work replace the dodecagonal fractures with randomly-oriented squares of equivalent area. ConnectFlow is designed to use fractures that are fundamentally collections of right-angled triangles or rectangles. In this scheme, while it is possible to model dodecagonal fractures, each one must be subdivided into at least 20 triangular sub-fractures, as opposed to a single sub-fracture required to model a square fracture, as depicted in Figure 3-3. As a result, using this shape considerably increases the complexity of the model simulation; memory usage, run time and model file size are all significantly increased compared with square fractures, as is the risk of poor numerical stability in the calculation. However, the disadvantage of replacing the dodecagonal fractures with square fractures is that this may in some cases mean that connections between fractures, or between fractures and the engineered openings, are created where they do not exist in HypoSite or are not created where they do exist in HypoSite.

To ensure that the flows generated using the alternative fracture geometries are similar, a comparison is made in ConnectFlow between a model using dodecagonal fractures and a model using square fractures, both replicating a simulation performed as part of the delivery of flow information for HypoSite (Follin 2015). It is found that, whereas the results of both tests are closely correlated with the flows produced for HypoSite, the flows in ConnectFlow with dodecagons are approximately 25 % of those found in Follin (2015) and the flows with equivalent squares are approximately 60 % of those found in Follin (2015). It is speculated that this change may result from a difference in boundary conditions or a difference in model domain between the models used, but a verification exercise to more conclusively determine the cause of the difference between the ConnectFlow calculations and the calculations performed in Follin (2015) is considered outside the scope of conditioning, and, particularly given the significant performance improvements generated, the use of equivalent squares is felt to be an acceptable choice. As a consequence, all simulated observations are re-created using ConnectFlow models with square fractures for the work reported here.

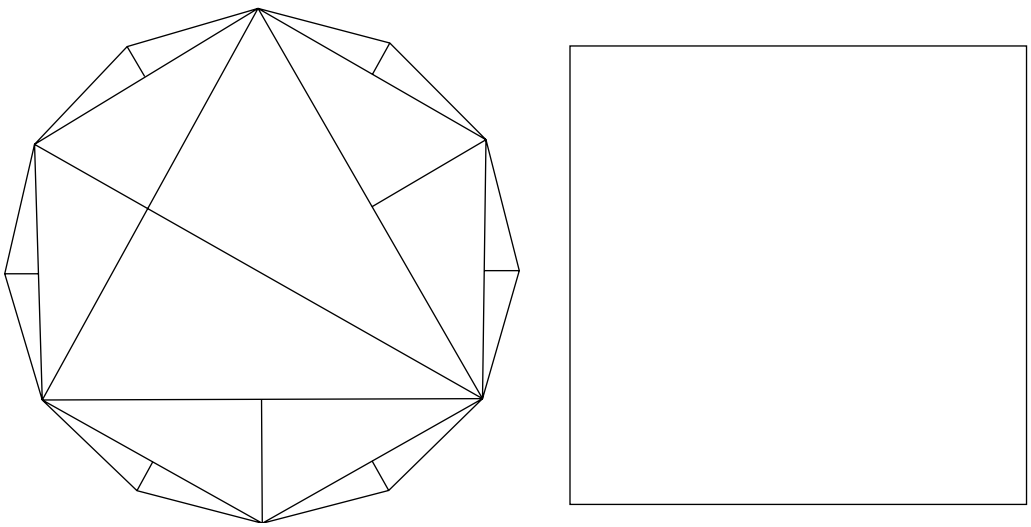

Figure 3-3. *A convex twelve-sided fracture (left) cannot generally be represented in ConnectFlow using fewer than twenty sub-fractures, whereas a square fracture (right) of equivalent area can be represented using a single sub-fracture. The large number of sub-fractures required to represent each twelve-sided fracture significantly increases the complexity of any calculation involving such fractures.*

3.2 Model implementation

For models at each stage of construction, three sizes of library are generated:

- A library containing 5,000 realisations based on the recipe for site- and panel-scale fractures (the "medium-fracture library").
- A library containing 10 000 realisations based on the recipe for site-scale fractures (the "large-fracture library").
- A library containing 30 000 realisations based on the recipe for site-scale fractures of radius larger than 25 metres (the "very large-fracture library").

Each of these size distributions is a subset of the last, i.e. any fracture in the large-fracture library could also have been generated in the medium-fracture library and any fracture in the very large-fracture library could have been created in the large-fracture library or in the medium-fracture library. The fractures themselves, however, are different in each library.

As described in Subsection 2.8.3, for some tests, particularly when checking the internal consistency of the method, it is of benefit to compare against additional realisations generated by ConnectFlow, using the HypoSite DFN recipe described in Subsection 3.1.1; this allows the internal consistency of the model to be checked, as well as providing a better statistical basis for some of the tests.

Conditioning models are run using the default settings for closeness-of-fit, except that for fracture dihedral angle (measure 8, see Appendix B) a maximum difference of 15° is imposed. This limit is used as it is felt that a larger value would give a poor visual fit to the fracture and may significantly affect connectivity far from the fracture. The maximum allowable total closeness-of-fit is otherwise 5.

3.3 Preliminary testing

Initial testing focussed on simple tests to confirm that fundamental aspects of the conditioning method work as expected.

3.3.1 Using observed data as the library

A key initial test of the model is to demonstrate that, if the intersections that make up the observed data are present in the library, the associated fractures are in fact chosen. While in theory it should be possible for other fractures to be chosen, the weight function (described in Appendix B) is such that, in practice, the probability of any other fracture being chosen is negligible. The test is passed if the mean closeness-of-fit of fractures chosen is very small.

This test is completed in two forms using the model with pilot holes for deposition holes. In the first, a direct copy of the observed data is used as the contents of the library. In the second, before being used, the observed data is altered to remove information about fracture connections between pilot holes and between tunnels and pilot holes. In this second instance, the conditioning process needs to establish whether and where these connections exist.

The results for both tests are identical. The average closeness-of-fit over the 290 intersections is 1.08×10^{-4}. This is equivalent to a location difference of approximately 0.1 millimetres, or a difference in dihedral angle of 0.0016 degrees. The largest individual closeness-of-fit for any fracture found is 5.67×10^{-3}, which is due to a 5.67-millimetre difference in the calculated centre line intersection points for the observed and library fractures (see Appendix B). It is thus judged that both tests were passed.

3.3.2 Use of multiple libraries

As described in Subsection 2.3.1, the aim of using multiple libraries with different fracture sizes is to ensure that there are a sufficient number of relatively large fractures available as conditioning candidates, and in particular a sufficiently large number of fractures that intersect multiple engineered openings. It is thus useful to know how many fractures in the library intersect a given number of engineered openings.

ConnectFlow can output this data for any conditioning run, and it may be useful for the user to review it to ensure that the library has the full range of fractures required to achieve satisfactory results.

In this test, the number of fractures intersecting a given number of engineered openings is calculated for each of the three libraries (described in Section 3.2). It is intended to demonstrate that the libraries containing larger fractures only are more likely to include fractures that intersect greater numbers of engineered openings.

Figure 3-4 shows the relative proportions of fractures that intersect different numbers of engineered openings arising from each library. An engineered opening in this case includes the main tunnel, four deposition tunnels (as each of the two in the model is divided for modelling at the main tunnel) and the 32 deposition holes to give a total of 37 engineered openings in the model. It is clear that the large- and very large-fracture libraries have a higher proportion of fractures that intersect multiple engineered openings compared to the medium fracture library. Because the time taken to generate a realisation depends on the number of fractures that need to be generated, it is possible to generate these fractures far more quickly than if it were necessary to create an equivalent number of large fractures in the medium-fracture library.

Figure 3-5 shows that the vast majority of fractures still intersect only one engineered opening and come from the medium-fracture library: these will be the easiest to find when conditioning and any given observed fracture is likely to match a significant number of candidates. The hardest to find will be fractures intersecting very large numbers of engineered openings. As the number of engineered openings intersected increases, the constraints placed on a matching fracture become harder to fulfil and so the number of candidate fractures reduces.

Knowledge of the intersections in the observed data can thus provide insight into the likelihood of sufficient candidate fractures being available in the conditioning libraries. If, for example, the user is aware of a fracture in the observed data that intersects 20 engineered openings, they might feel that more than the 24 fractures from the three libraries combined are needed to provide an adequate statistical distribution of the possible fractures – particularly as not all of those fractures may allow the correct 20 openings to be chosen. On the other hand, if no fracture intersects more than four openings, they may feel that the 12 034 fractures available that intersect 4 openings are sufficient and that no further realisations are needed. For use in this work, no observed fracture intersects more than seven engineered openings, so the fact that the number of library fractures that intersect much larger numbers of openings is relatively small is not of significant concern.

3.3.3 Visual Checks

It is useful to verify whether there is a good visual match between the traces from conditioned fractures and the observed fracture traces.

Figure 3-6 and Figure 3-7 compare the observed and conditioned fractures added for two different realisations of HypoSite BM-1b. These show that, while they are not identical, the fractures added through conditioning are generally similar to those observed.

In performing visual checks, aspects such as the P_{21} and geometric measures given below can be taken into account, but it is also possible to quantify this using the closeness-of-fit measure, which is defined in detail in Appendix B. Figure 3-8 shows the mean closeness-of-fit values over the intersections after conditioning each of the ten realisations of a HypoSite model including pilot holes for deposition holes. This shows an average closeness-of-fit of less than one. As a closeness-of-fit of one for any individual measure is considered a poor match, and the closeness-of-fit reported is that for the aggregate of all the measures, this suggests that the matches being made are reasonable.

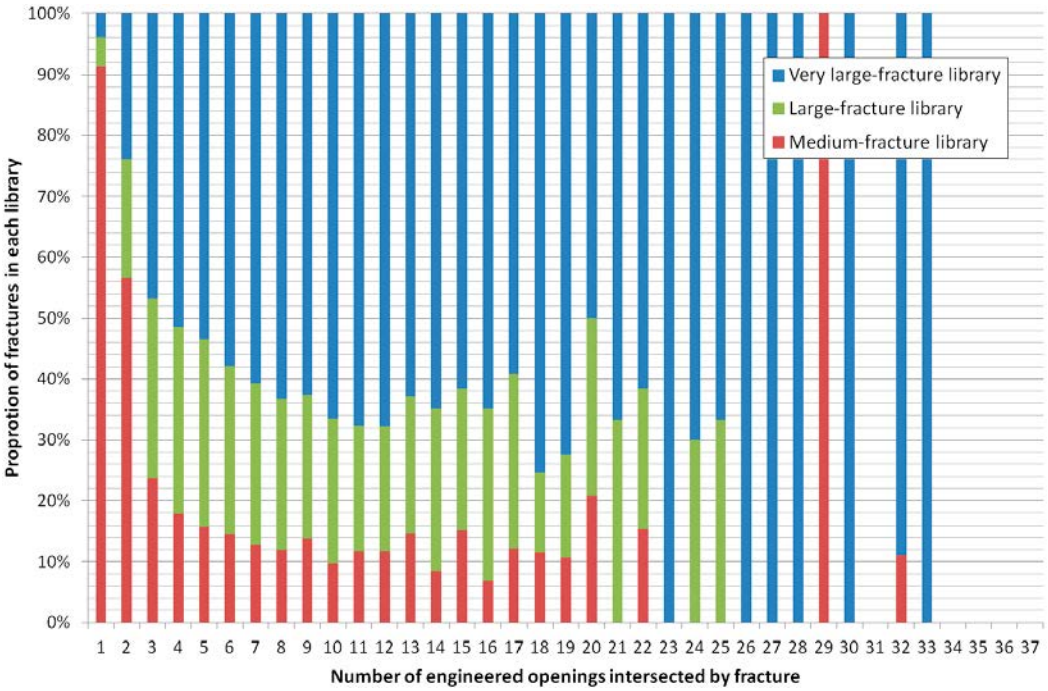

Figure 3-4. *Graph showing the proportion of fractures intersecting each number of engineered openings in the model arising from each library. The red parts of the line are for the medium-fracture library, the green parts are for the large-fracture library and the blue parts are for the very-large fracture library (see Section 3.2 for details).*

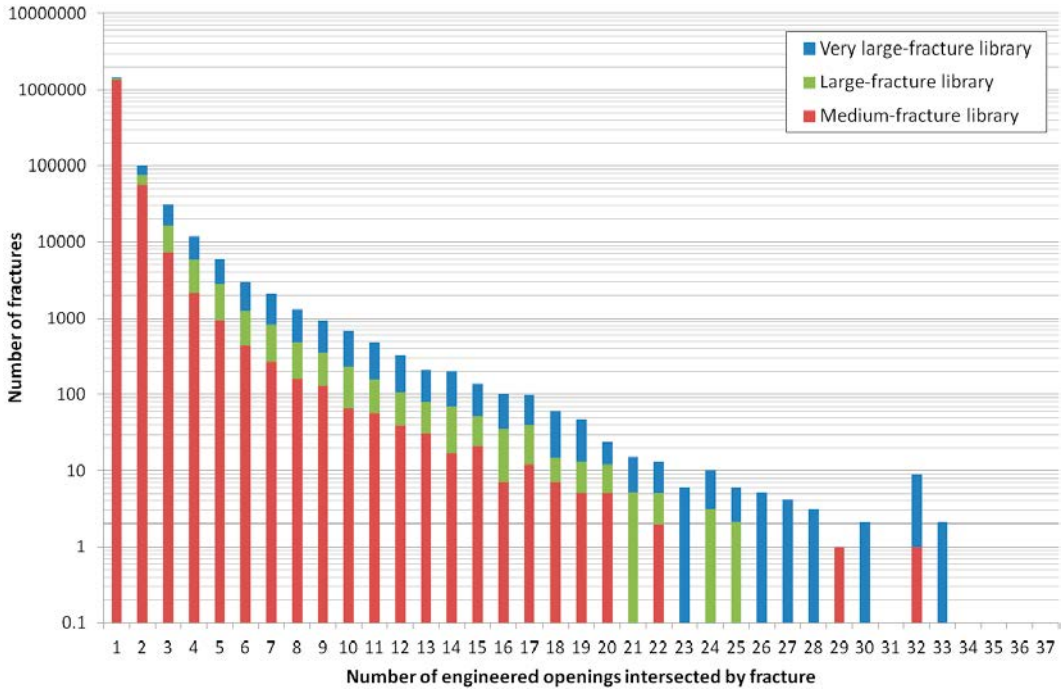

Figure 3-5. *Graph showing the number of fractures that intersect each number of engineered openings in the model. The red parts of the line are for the medium-fracture library, the green parts are for the large-fracture library and the blue parts are for the very-large fracture library (see Section 3.2 for details). Note that the scale on the Y-axis is logarithmic.*

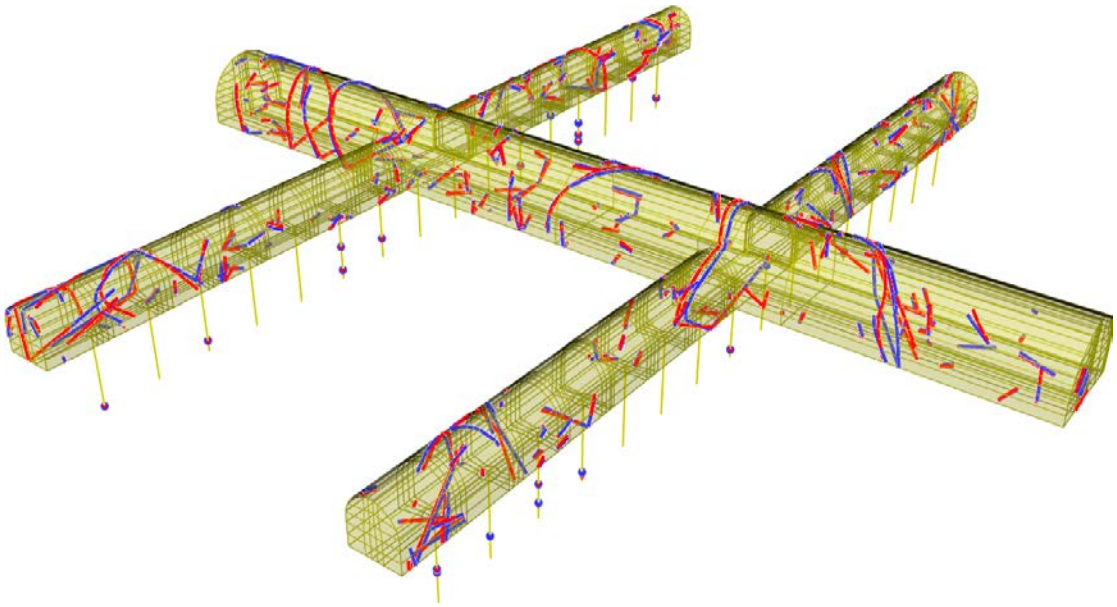

Figure 3-6. *Comparison between fractures observed (blue) and added through conditioning (red) for a model conditioned without flow for Realisation 4 of HypoSite BM-1b. Where there is a fracture with no obvious counterpart, this is generally because the two traces are in similar places and overlaid.*

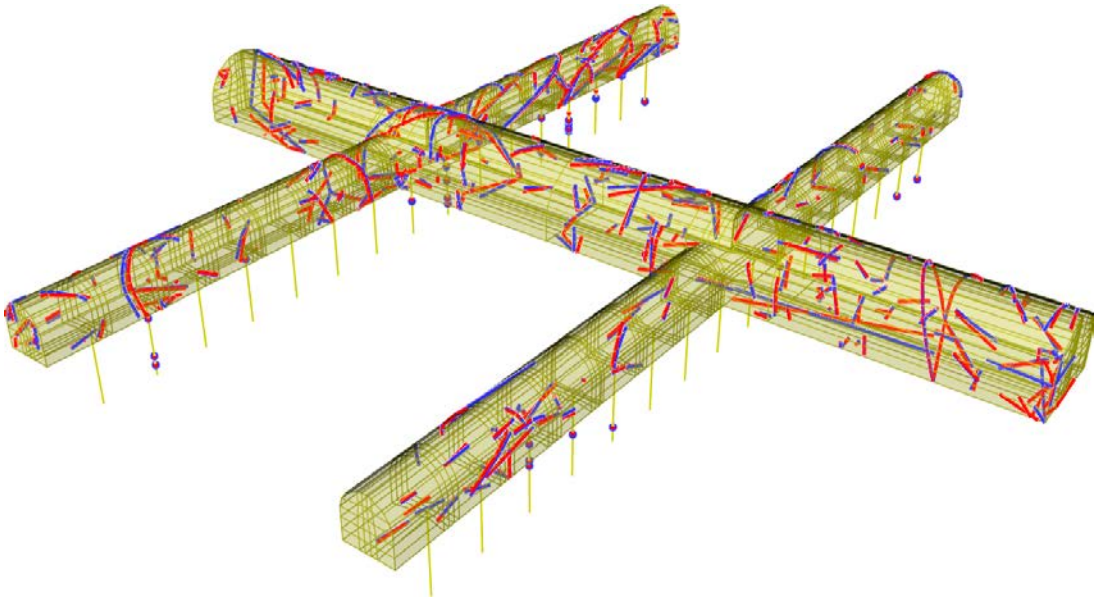

Figure 3-7. *Comparison between fractures observed (blue) and added through conditioning (red) for a model conditioned without flow for Realisation 6 of HypoSite BM-1b. Where there is a fracture with no obvious counterpart, this is generally because the two traces are in the similar places and overlaid.*

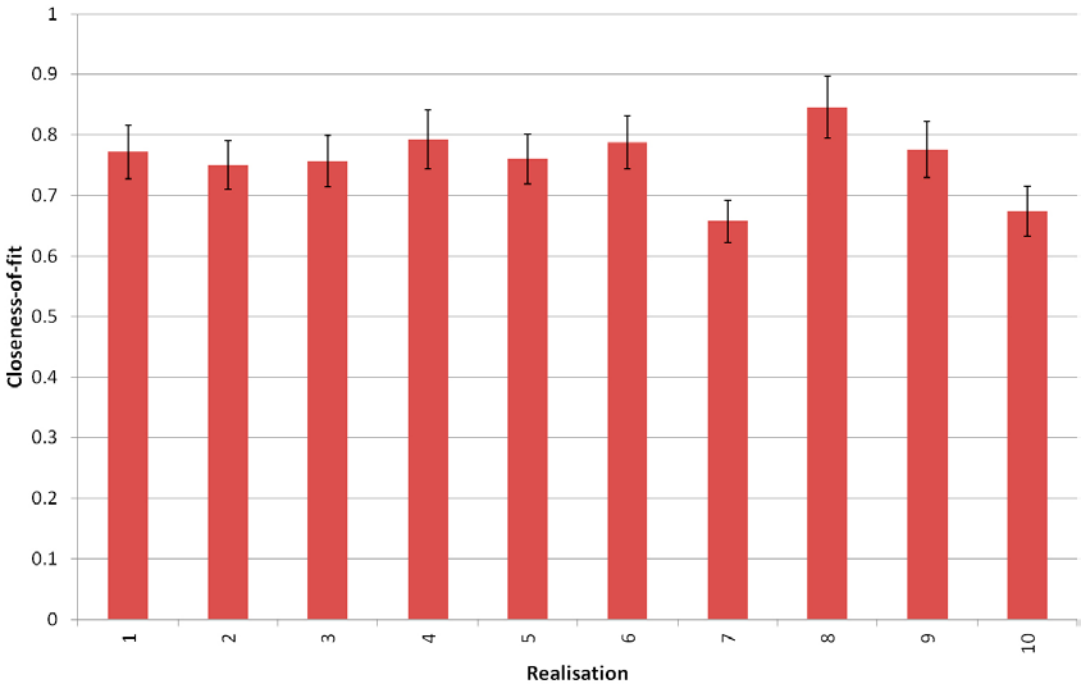

Figure 3-8. *Mean and standard error in closeness-of-fit (see Appendix B) over the individual intersections for a single conditioned realisation of each realisation of HypoSite, conditioned using pilot holes for deposition holes.*

3.4 Calibration targets and consistency checks in HypoSite BM-1b

The calibration targets and consistency checks described in Section 2.8 are evaluated for the HypoSite BM-1b model described in Section 3.1, conditioning on pilot holes for deposition holes. Other models are also presented to demonstrate generality. The flow tests are discussed separately in Chapter 4.

3.4.1 P_{21} tests

Figure 3-9 shows the variation in P_{21} after conditioning on 10 realisations of HypoSite using intersections on the main tunnel, deposition tunnels and pilot holes for deposition holes. Note that as pilot holes are modelled as 1-dimensional scan lines, their intersections have of zero length and there is no meaningful way in which those intersections can be included in this measure. Thus, whereas the conditioning included the pilot holes the P_{21} values take no account of them.

The figure shows that in all but three realisations, the difference between conditioned and observed results is less than 5 %. Of the three remaining realisations, Realisation 4 gives a difference of 5.2 %, Realisation 2 a difference of 7.9 % and Realisation 5 a difference of 12.2 %.

A second suggested target from Subsection 2.8.1 is that the P_{21} be within 20 % of the variation between unconditioned realisations. In this case, the extreme values in the unconditioned data give a difference in P_{21} of 0.057, and 20 % of this is 0.011. The test is passed for eight of the ten realisations, the exceptions being Realisations 2 and 5.

Note that in each case only one realisation of conditioning is used in calculating this measure. As conditioning relies on random selection, it is likely that if different initial settings are chosen, different results will be obtained.

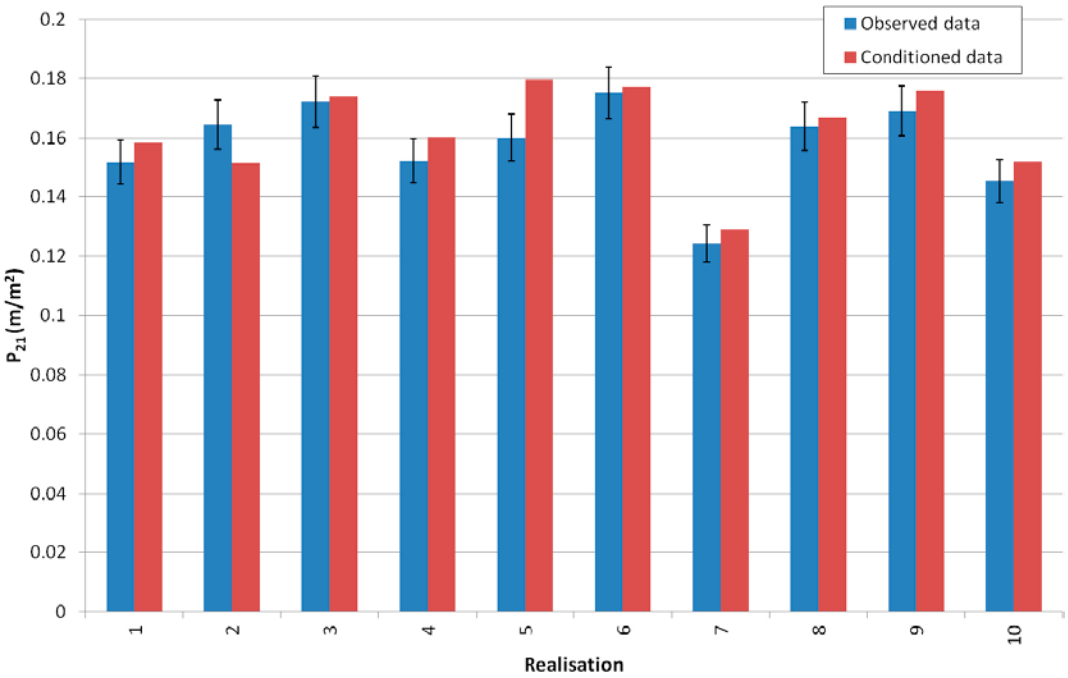

Figure 3-9. *P_{21} for conditioning of ten realisations of HypoSite, comparing observed and conditioned data. The error bars on the observed data are 5 % of the total P_{21} observed in the realisation, and are shown to provide an indication of what is considered an acceptable range of conditioning outcomes.*

3.4.2 Fracture location tests

The two tests for fracture location and orientation are based on the proportion of conditioned fractures that demonstrate a difference from the observed traces greater than a given tolerance in each of two measures: the dihedral angle between the conditioned and observed fractures (the difference in fracture pole) and the position of the conditioned and observed traces.

The acceptance limit for the distance test is 1.7 metres and for the dihedral angle test is 12.25°. Observed traces for which there is no conditioned fracture are treated as being outside the acceptance limit.

The match rate is calculated for differences in positions and poles in one realisation of conditioning for each realisation of HypoSite. The success rate for position varies from 96 % (302 traces of out of 313 meet the target) in Realisation 5 to just less than 100 % (285 out of 286 meet the target) in Realisation 9. The success rate for poles varies from 97 % (268 out of 276 traces successfully matched within the limit) in Realisation 8, to a 100 % match in four Realisations (numbers 3, 6, 9 and 10). The distributions of these results are shown in Figure 3-10.

3.4.3 P_{32} tests

As noted in Subsection 2.8.3 above, the aim of the test of P_{32} is to check the internal consistency of the model, and as such, the test is based on thirty internally-generated realisations rather than the HypoSite realisations provided. The test is based on local P_{32}, i.e. P_{32} in the vicinity of the engineered openings, in order to increase the sensitivity to changes in P_{32} caused by conditioning.

Each realisation has a different set of simulated observed data and a different unconditioned realisation of fractures.

Results are calculated for each of the five stages of construction of the engineered openings, but the results presented here are for the model following construction of pilot holes for deposition holes. This is an important model as it is the model used as a basis for the hydraulic tests.

Figure 3-11 shows the P_{32} for 30 realisations of the model with pilot holes for deposition holes. Compared against the benchmarks proposed, it is noted that the median value of the realisations in the library is 0.241 and in the conditioned realisations it is 0.251, a difference of approximately 4 %. Three of the thirty realisations (10 %) are outside the range between the 5th and 95th percentiles of the library (all in this case above the 95th percentile), which is lower than the target limit of 15 % and so the test is passed.

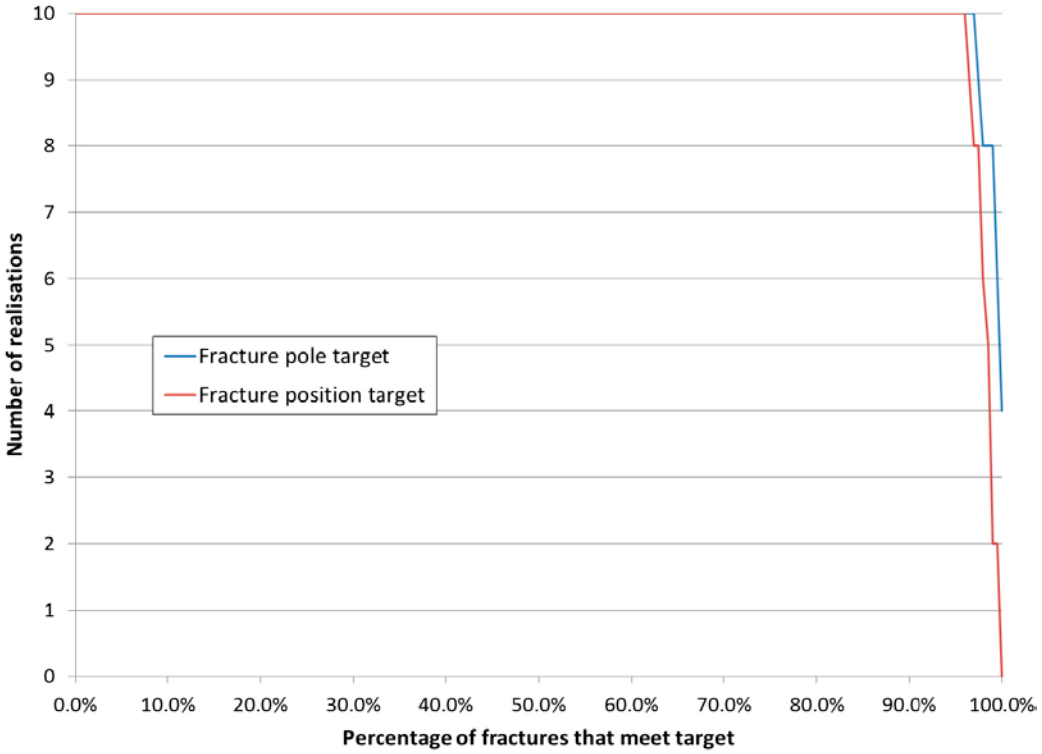

Figure 3-10. *The number of realisations (out of 10) in which a given proportion of fractures meet the target for the tests based on the fracture pole (blue) and fracture position (red). The graph shows that the vast majority of intersections successfully pass both tests.*

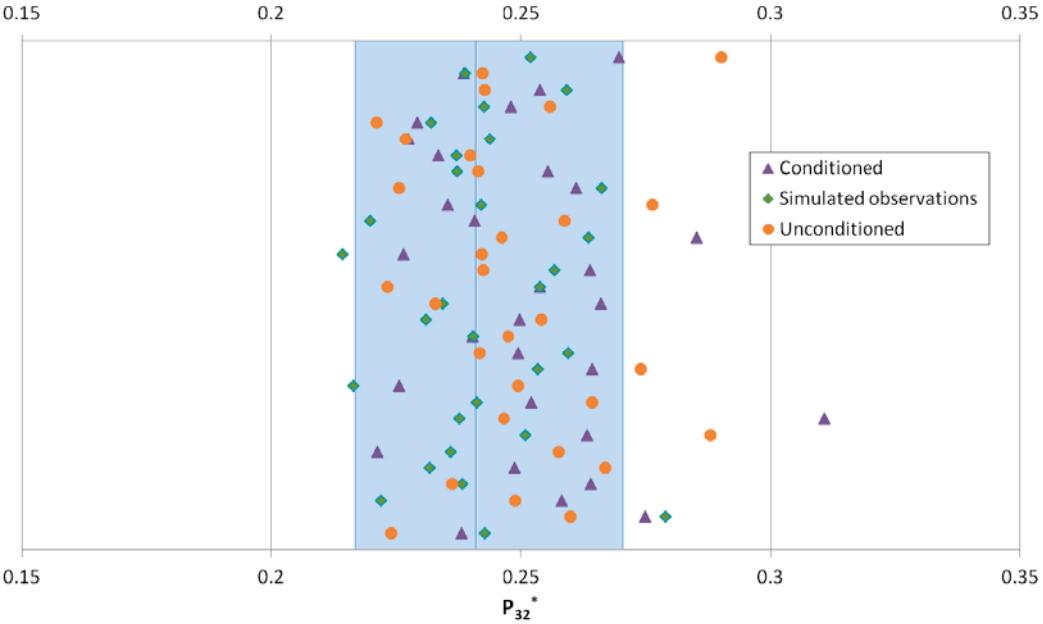

Figure 3-11. *Graph of the P_{32} [m^2/m^3] results for thirty realisations. The Y-axis divides the model by realisation for presentational purposes. The blue band signifies the range of P_{32} in the library between the 5th and 95th percentile values, and the line down the middle of the band is the median value. The orange dots are the P_{32} of each realisation before conditioning, and the purple triangles are the P_{32} of each realisation after conditioning. The green diamonds are the P_{32} of each set of simulated observed data.*

While it is possible to compare P_{32} for individual realisations using this graph, and the conditioning process does attempt to match P_{32} indirectly, it is not possible to reliably recreate the P_{32} on the level of an individual observed realisation because the observed data input is not sufficient to determine the precise size of each fracture. Methods for inferring the size and extent of fractures away from the engineered openings are discussed in Sections 4.3 and 4.5, but these have not been applied in this case.

It is, however, useful to compare the variation found in the conditioned data with that found in the unconditioned and simulated observed data because the unconditioned and simulated observed data are generated from the same distributions as the library data. In this case, the unconditioned data set has more realisations than the conditioned data set with a P_{32} higher than the 95th percentile from the library, and neither conditioned nor unconditioned data sets have any below the 5th percentile.

3.4.4 Size distribution tests

The second consistency check compares the distribution of fracture size based on their equivalent radius. As with P_{32} it is important to ensure that relevant fractures are counted to provide a sensitive measure, and that the anticipated size distribution of fractures found does not vary with fracture size (as it would if the fractures selected were those that intersected the engineered openings). Thus, the test considers the radii of fractures intersecting scan lines. If a fracture intersects two scan lines, it is counted for both.

It is important to note that the distribution of fracture radii may be different for fractures intersecting scan lines in different directions. It is likely that a number of different fracture sets will be in a model, with different fracture size and orientation distributions, and that a scan line in a given direction will be more likely to intersect fractures from some sets than those from others. The details of the three sets in HypoSite are given in Subsection 3.1.1.

As noted in Subsection 3.1.2, the fractures used for modelling in this work are squares and not circles; the equivalent radius is thus defined as the radius of a circle with equivalent area to the square, i.e. $r = \frac{l}{\sqrt{\pi}}$, where r is the fracture radius and l the fracture side length.

Figure 3-12, Figure 3-13 and Figure 3-14 show the equivalent radius distribution of fractures found along the scan lines after conditioning the model at each phase of construction, from the same thirty realisations as were used for the P_{32} test above. The scan lines are in directions parallel to the main tunnel centre line, parallel to the deposition tunnel centre lines, and parallel to the deposition hole centre lines (i.e. vertical). Not all directions are applicable to all models, so only those that are relevant are shown.

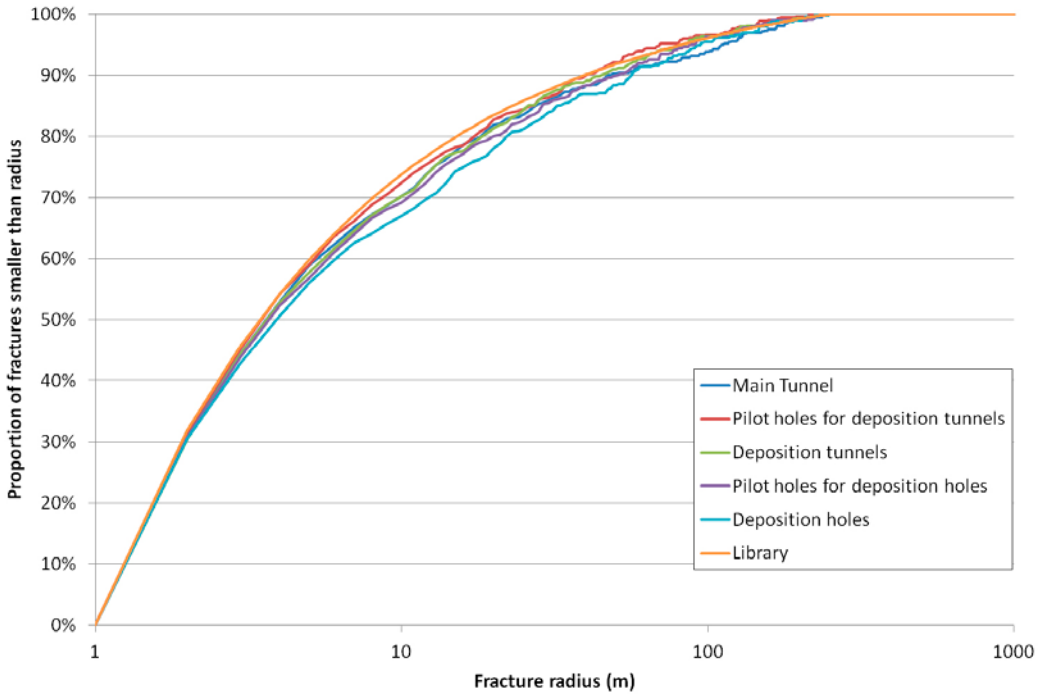

Figure 3-12. *Cumulative distribution of fracture radii along scan lines parallel to the main tunnel. The orange line is based on data from the library, considered a reasonable empirical equivalent to the analytical distribution, and the other five lines are from the results of conditioning after each of the different construction stages.*

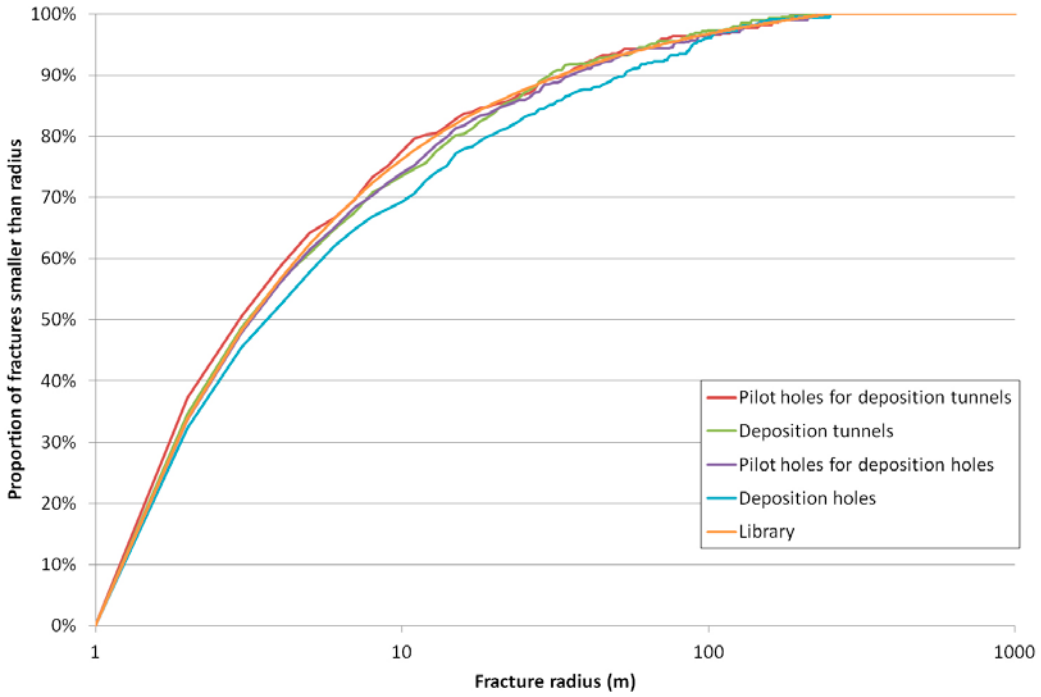

Figure 3-13. *Cumulative distribution of fracture radii along horizontal scan lines perpendicular to the main tunnel. The orange line is based on data from the library, considered a reasonable empirical equivalent to the analytical distribution, and the other four lines are from the results of conditioning after each of the different construction stages.*

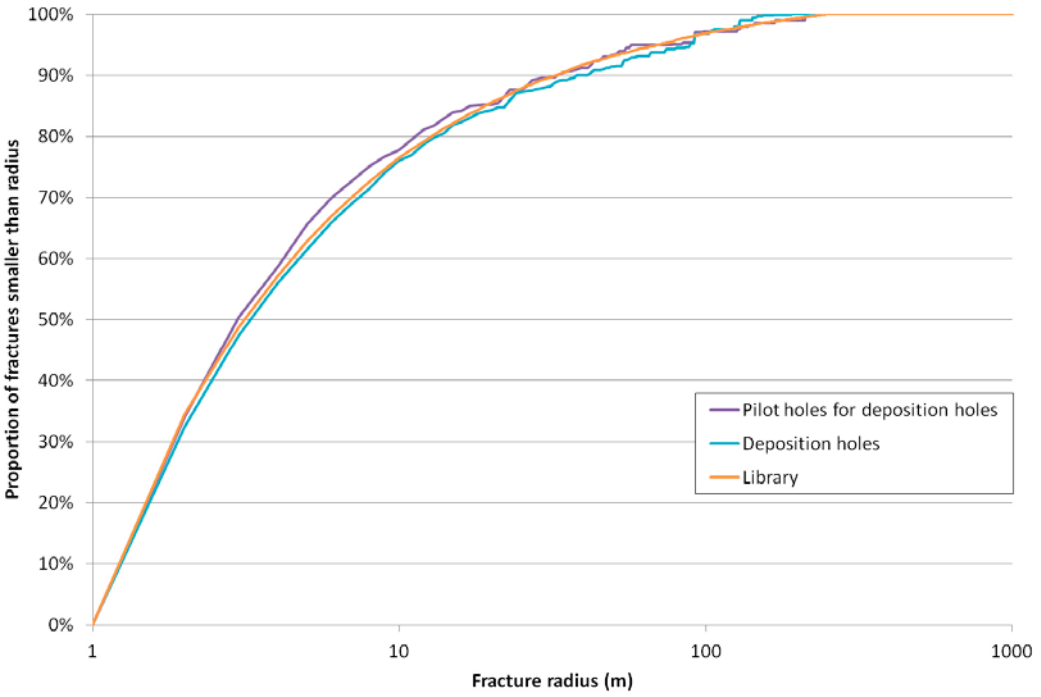

Figure 3-14. *Cumulative distribution of fracture radii along vertical scan lines. The orange line is based on data from the library, considered a reasonable empirical equivalent to the analytical distribution, and the other two lines are from the results of conditioning after each of the different construction stages.*

These distributions are compared against the equivalent results from the library. As the library is generated by generating a very large number of fractures sampled from the analytical size distributions required, it is reasonable to assume that the library result represents an empirical equivalent to the analytical solution.

3.5 Geometric testing in HypoSite BM-1_Hydro-test-II

HypoSite BM-1_Hydro-test-II is an early version of HypoSite to be used to test conditioning methods. However, it emerged that an error had been made in the depth dependence of the fracture intensity, meaning that fractures at repository depth were far more numerous than would be expected from the intended DFN recipe.

While not used for the main study, this nonetheless provides a useful test of the conditioning method in the situation where the fracture model is based on fracture statistics that are not consistent with the observed data. Note, however, that the HypoSite BM-1_Hydro-test-II model only differs from HypoSite BM-1b in terms of fracture intensity, not in terms of other factors such as fracture orientation or fracture size distribution. Significant differences in fracture property distributions between the observed data and the model used for conditioning may adversely affect the effectiveness of the conditioning.

Figure 3-15 shows the P_{21} observed and calculated for each of the five stages of construction using HypoSite BM-1_Hydro-test-II. Note that only one realisation of the model is available and that results were calculated before the extra searches described in Section 2.7 were made available.

The conditioning method successfully matches the P_{21} for all five of these cases. But note that these P_{21} values are at the very highest end of the values in the libraries used for conditioning. In the case where it is calculated on deposition tunnels, 4999 of the 5000 realisations in the library have a P_{21} lower than that observed in HypoSite BM-1_Hydro-test-II. The conditioning is still able to add appropriate fractures despite the incorrect intensity, though as the wider fracture network may be too sparse, flow calculations are likely to under-predict the observed flows.

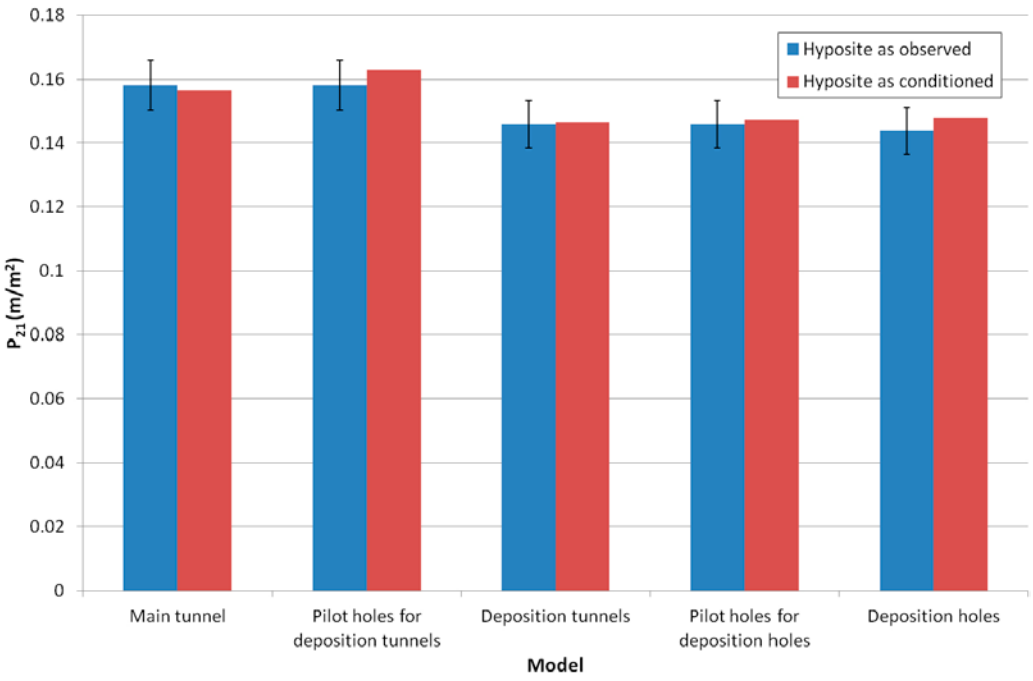

Figure 3-15. *Observed and conditioned P_{21} for HypoSite BM1-Hydro-test-II for five stages of construction, using libraries and unconditioned models based on a fracture recipe equivalent to HypoSite BM-1b. Error bars show acceptance limits described in Subsection 2.8.1.*

4 Application of conditioning for HypoSite BM-1b

While there are benefits in having a model that respects the geometry of the observed data, a key benefit of using the conditioning method is its ability to predict, given intersection and flow data from a pilot hole or deposition hole, the local post-closure flow conditions in that specific deposition hole. This then allows less suitable holes (based on deposition criteria) to be screened out.

This chapter demonstrates two methods for doing this, the first based on inflows into fully-drilled deposition holes, and the second based on injection tests carried out in pilot holes for the same deposition holes.

In each case, ten realisations of conditioning have been run, with a single realisation of HypoSite chosen as a synthetic reality (Realisation 4, chosen because it has the most deposition holes with connections to the external boundary via the DFN; see Section 3.1 for a description of HypoSite). Each realisation of conditioning can potentially include different fractures chosen by weighted random selection during conditioning. This does not guarantee that different fractures will be chosen in every case, as it may be that one match is sufficiently better than the alternatives that it is easily the most likely to be chosen. In addition, a different background (unconditioned) realisation surrounding the conditioned fractures has been used with each realisation of conditioning, meaning that the fracture connectivity and flow field will be different in each case.

Inflow measurements and hydraulic tests can be carried out in the field, but the average flow-rate per unit length (U) around the deposition holes and flow-related transport resistance (F) for particles originating in each deposition hole are not directly measurable. However, U and F can be calculated for the HypoSite synthetic reality and so will enable the effectiveness of conditioned models in predicting these quantities to be assessed.

Note that holes NNW:8 and NNWB:8 are located directly underneath the main tunnel. It is unlikely, in reality, that such holes will be considered and it is relatively difficult to find matches for fractures in that part of the model (as the combination of multiple nearby tunnels effectively prevents the method from exploiting the assumption of stationarity). Consequently, these holes have not been included in the results in this chapter. Results for holes NNW:9 and NNWB:9 are retained even though they protrude into the main tunnel, as their centre lines are underneath deposition tunnels and they do not create significant problems in matching intersections.

4.1 Deposition holes

Following construction of the deposition holes it is assumed that a measurement is taken of each of the inflows into each open deposition hole after construction, as well as into the open tunnels. It is, for the purposes of this demonstration, assumed that it is technically possible to measure these inflows for each intersection. The lower measurement limit (below which flows are indistinguishable from non-flowing fractures) is taken to be 10^{-5} L/min (1.667×10^{-10} m^3/s).

4.1.1 Inflow tests

Inflows are simulated given a head of 0 metres at all external boundaries and −400 metres at all engineered openings. The predicted inflows are compared with the synthetic reality in the case where no conditioning has been applied in Figure 4-1, where geometric conditioning has been used in Figure 4-2, and where inflow conditioning has been used in Figure 4-3.

As these figures show, predicted inflows after conditioning using flow data are frequently closer to the observed data than predicted inflows after conditioning using geometric data only. This is particularly noticeable for deposition holes such as NNWB:5 and NNW:12 (where geometric conditioning incorrectly predicts no inflow or underpredicts the inflow).

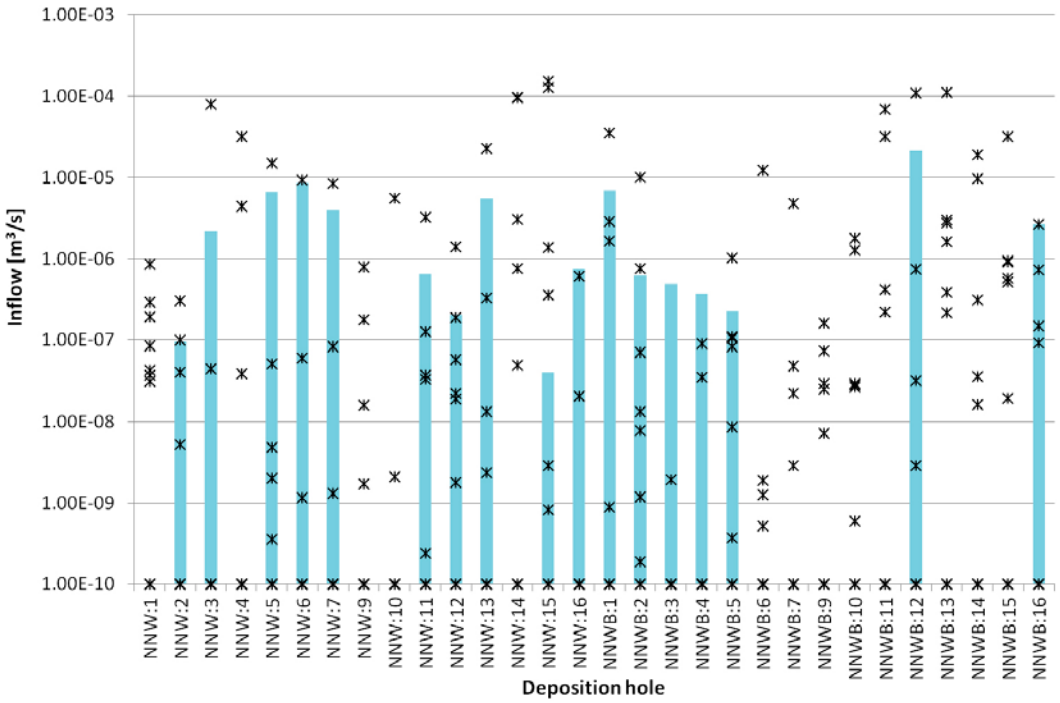

Figure 4-1. *Inflow into deposition holes for each of ten realisations of the HypoSite model with no conditioning. The blue lines are the total inflow into each deposition hole of the synthetic reality; each star is the total inflow value into one hole for one unconditioned realisation. Inflows at 10^{-10} m³/s are those where no flow was found, or where flow was negligible.*

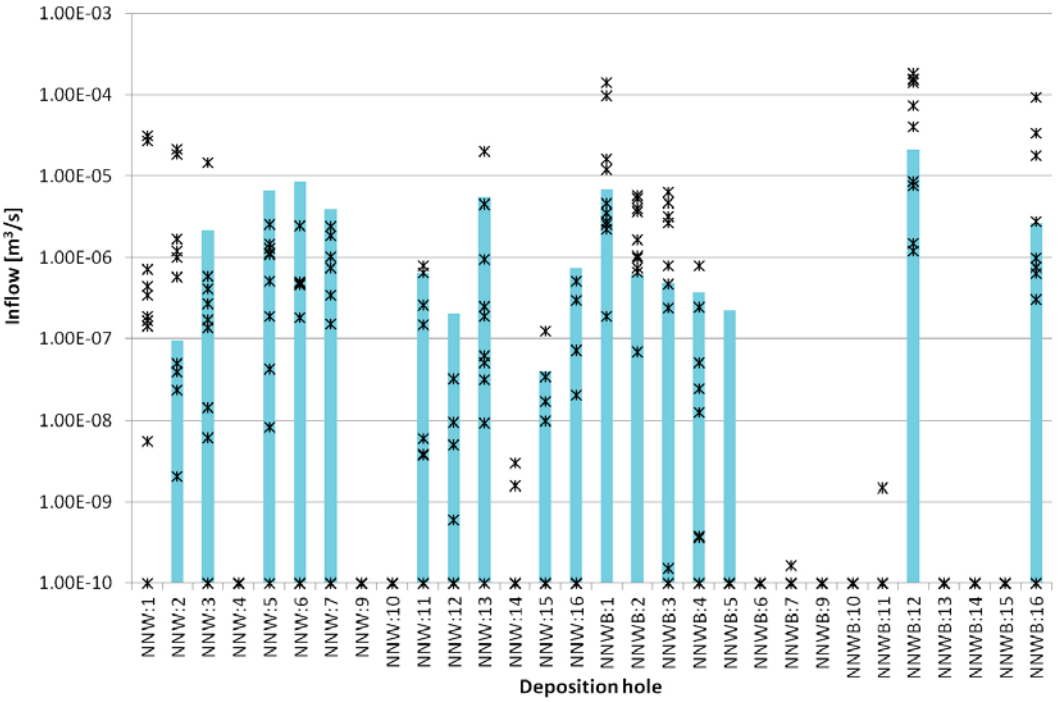

Figure 4-2. *Inflow into deposition holes for each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are the total inflow into each deposition hole of the synthetic reality; each star is the total inflow value into one hole for one conditioned realisation. Inflows at 10^{-10} m³/s are those where no flow was found, or where flow was negligible.*

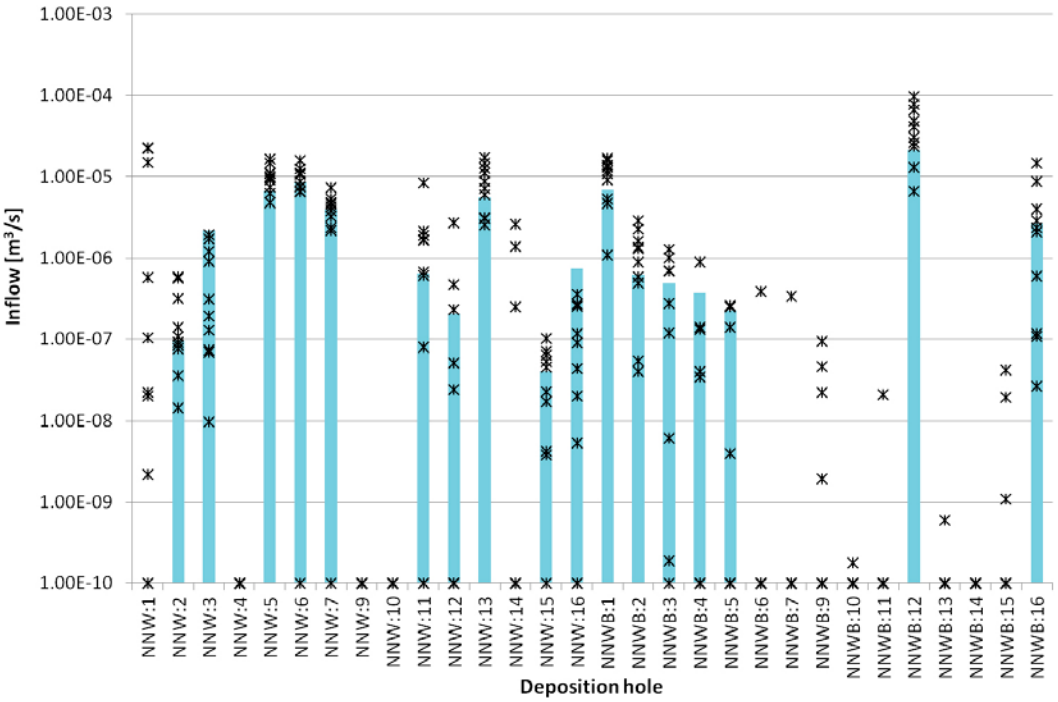

Figure 4-3. *Inflow into deposition holes for each of ten realisations of the HypoSite model after conditioning using observed inflows. The blue lines are the total inflow into each deposition hole of the synthetic reality; each star is the total inflow value into one hole for one conditioned realisation. Inflows at 10^{-10} m^3/s are those where no flow was found, or where flow was negligible.*

Comparing both sets of conditioned inflows against those obtained using unconditioned models, it is clear that even geometric conditioning adds a significant degree of predictive power for specific holes. For unconditioned stochastic models it is only possible to make general predictions based on statistical probability, i.e. the proportion of holes that will be screened out can be predicted, but not which holes these will be.

4.1.2 Post-closure performance measures

In order to calculate post-closure performance measures, models were developed that have the tunnels and deposition holes filled with low permeability backfill and buffer, respectively, represented as a CPM (continuous porous medium). The surrounding fractured bedrock is represented by realisations of the conditioned HypoSite DFN. The model boundary conditions are specified such that there is a low flow towards the north-eastern corner of the top surface of the model. Particles are tracked from each of the deposition holes, and their paths are used to calculate U and F as discussed in Subsection 2.8.7.

Results are shown first for U, then for F.

Average flow-rate per unit length (U)

Figure 4-4, Figure 4-5 and Figure 4-6, compare predictions of U made using the conditioning methods against inflow values simulated in the synthetic reality (i.e. HypoSite). There is, in general, a better prediction when conditioning on inflows than when conditioning on geometry alone. Without conditioning, it is clear that no sensible predictions on a per-hole basis are possible. Note that a very high U (over 0.1 m^2/yr) occurs in five of the ten unconditioned realisations, i.e. it is not just one anomalous realisation.

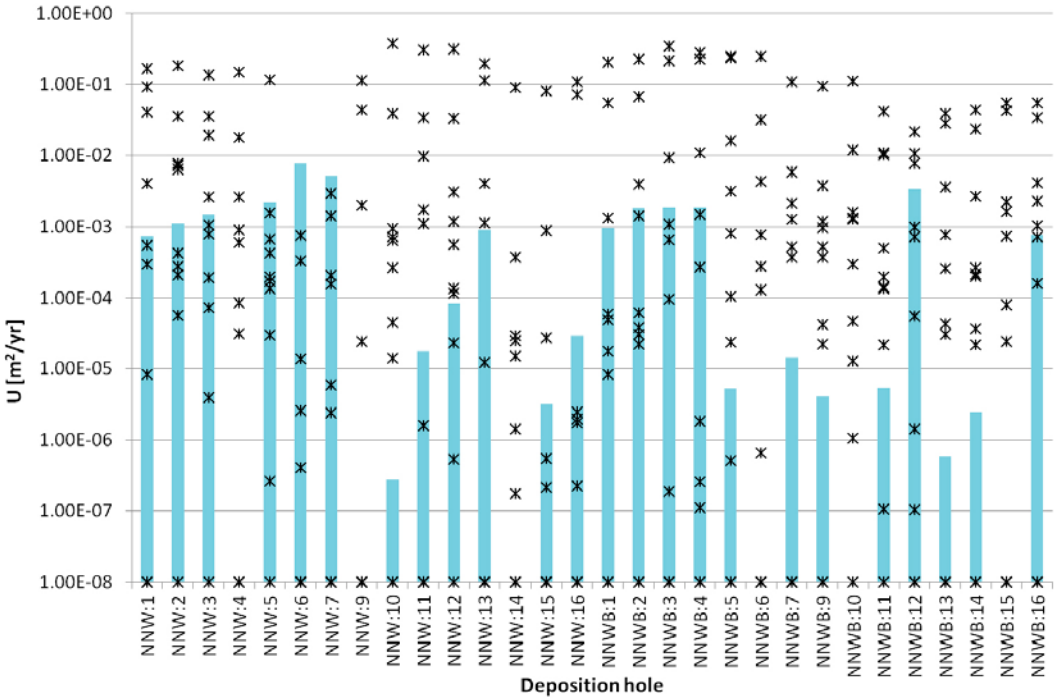

Figure 4-4. *U in each of ten realisations of the HypoSite model with no conditioning done. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one unconditioned realisation. Values less than or equal to 10^{-8} m^2/yr are marked as 10^{-8} m^2/yr.*

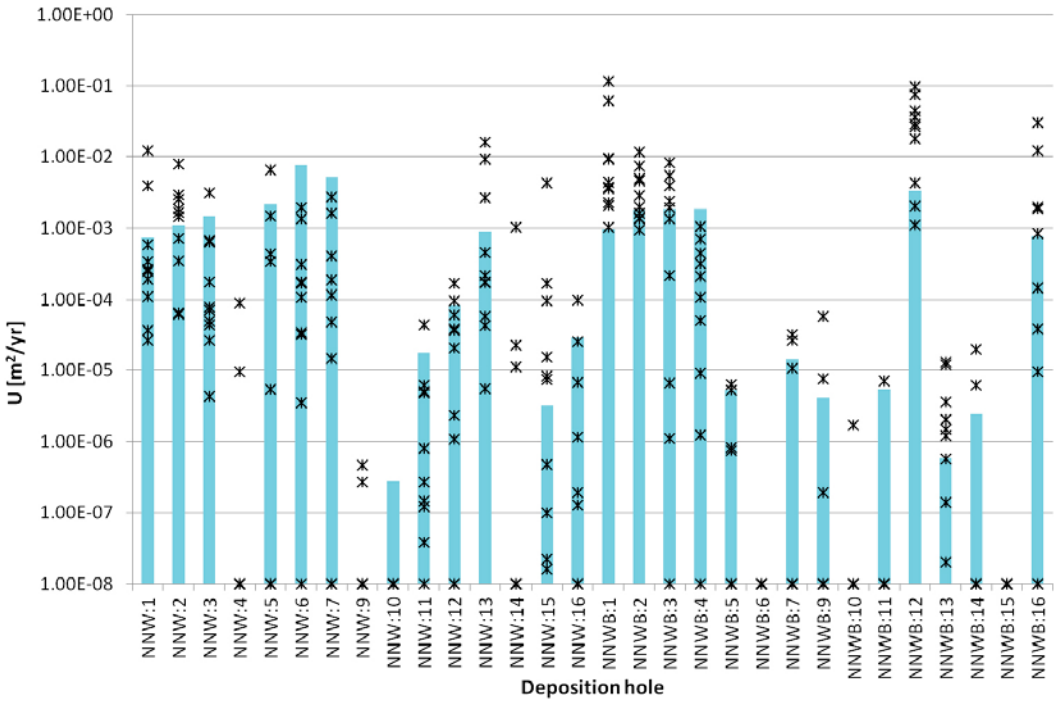

Figure 4-5. *U in each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one conditioned realisation. Values less than or equal to 10^{-8} m^2/yr are marked as 10^{-8} m^2/yr.*

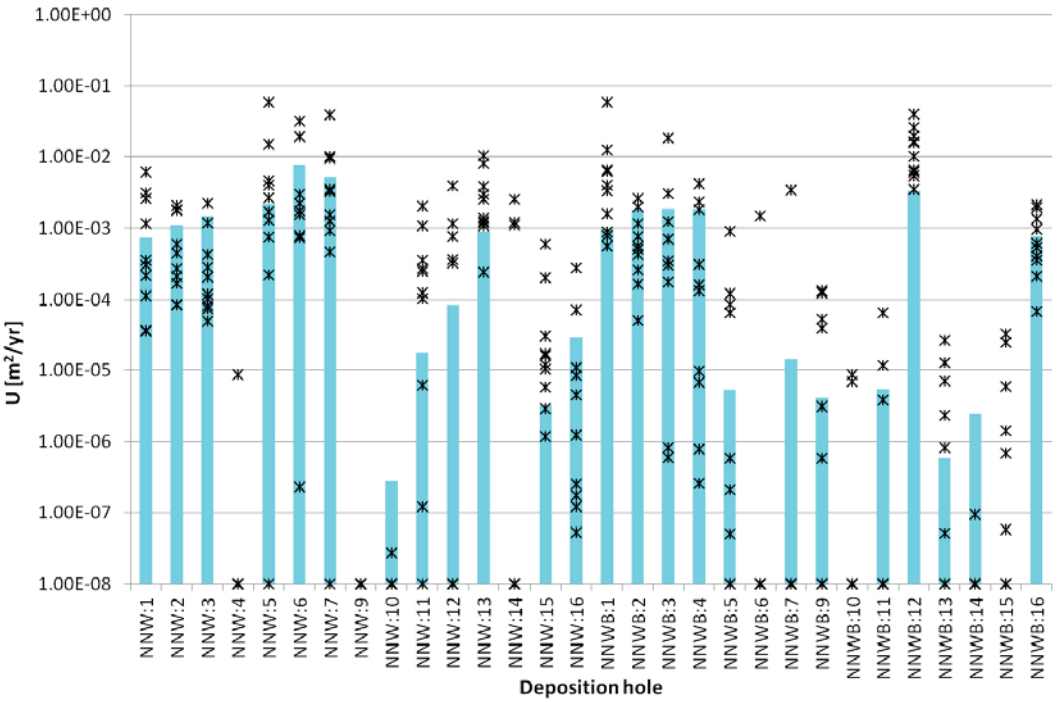

Figure 4-6. *U in each of ten realisations of the HypoSite model after conditioning using observed inflows. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one conditioned realisation. Values less than or equal to 10^{-8} m²/yr are marked as 10^{-8} m²/yr.*

Flow-related transport resistance (F)

As with *U*, it is clear from Figure 4-7, Figure 4-8 and Figure 4-9 that there is some predictive capability of the method for calculating *F* in individual holes, and that this is better when conditioning on flow data. There is little predictive capability for individual holes if conditioning is not done.

The graphs suggest that *F* is somewhat over-predicted by the conditioning method in a few cases. This is likely to arise because the realisation of HypoSite chosen to represent the synthetic reality was not selected randomly, but was selected on the basis of a high number of deposition holes connected to the external boundary, i.e. this realisation is atypical. These connections are caused by certain large fractures that are close to the pilot holes but do not intersect them. These fractures reduce the flow-related transport resistance in the observed realisation in a way that cannot be predicted based on the data used for conditioning, particularly when flow is not taken into account.

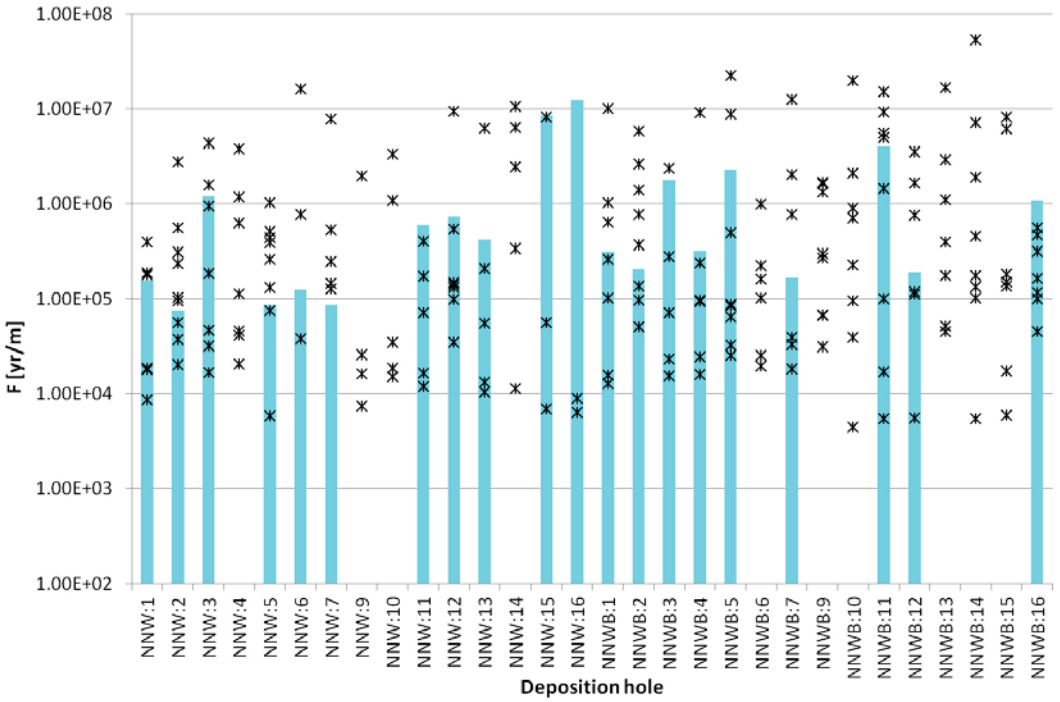

Figure 4-7. *F in each of ten realisations of the HypoSite model with no conditioning done. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one unconditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

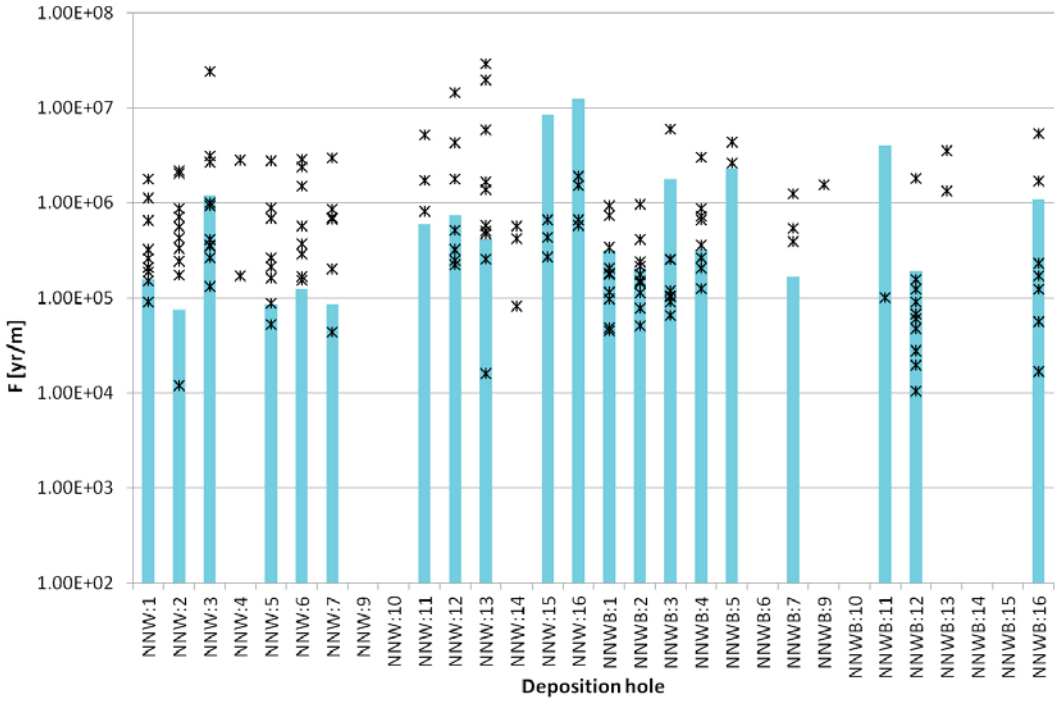

Figure 4-8. *F in each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one conditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

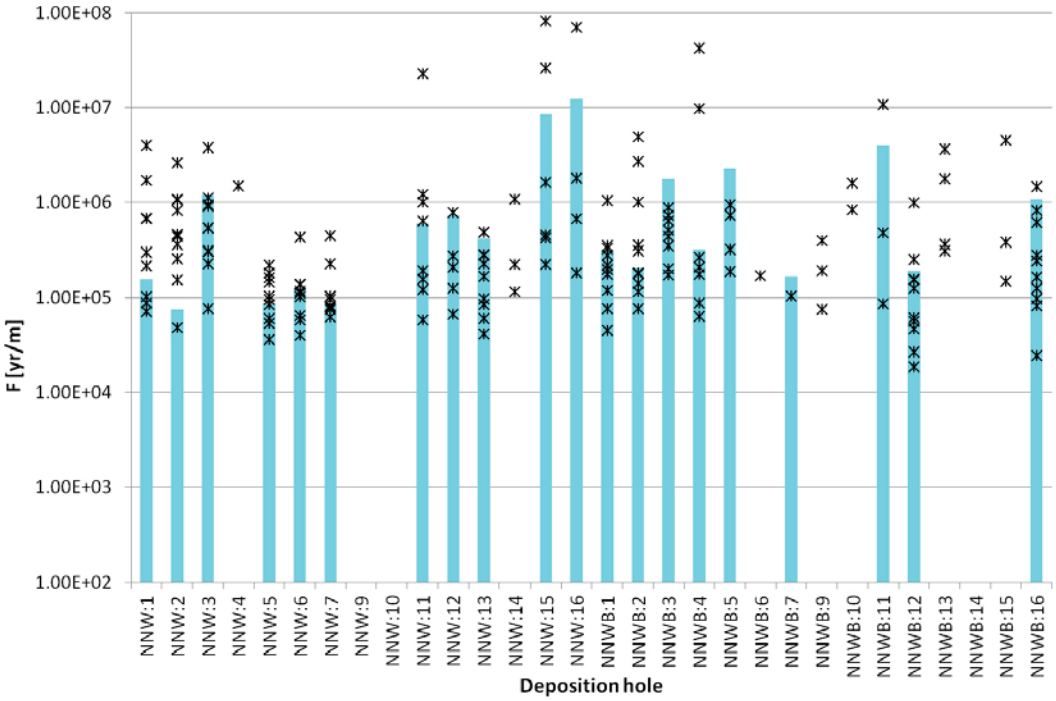

Figure 4-9. *F in each of ten realisations of the HypoSite model after conditioning using observed inflows. The blue lines are observations from the synthetic reality in each deposition hole; each star is the result in one hole for one conditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

4.2 Pilot holes for deposition holes

This application considers injection tests in pilot holes for deposition holes. Water is injected into each pilot hole in turn with an over-pressure of 20 metres of head, and the flows in the pilot holes are measured once steady-state is achieved. This method increases the sensitivity for identifying water conducting fractures. The steady-state inflows in the tunnels (undisturbed by injection tests) are also measured as before.

The pilot holes used are modified slightly from those described in Section 3.1. Instead of 7.833-metre holes located directly underneath the deposition tunnels, the pilot holes are 7 metres long and start 1 metre below the tunnel floor, i.e. the upper section is assumed to be sealed.

The objective is to use models conditioned on pilot hole injection tests to predict post-closure flow results in deposition holes. Note that there may be predictive errors in identifying the fractures that will intersect a deposition hole based on those that intersect a pilot hole due to the difference in hole diameters.

4.2.1 Injection tests

The modelled specific capacities in the pilot holes are compared with the synthetic reality in the case where conditioning is not used in Figure 4-10, with geometric conditioning only in Figure 4-11, and after conditioning using injection tests in Figure 4-12. The conclusions are similar to those found with inflow tests, i.e. conditioning with flow data gave a closer match on specific capacity, with less spread, than conditioning with geometric data only, and there was no evidence that calculations without conditioning could be used to make predictions for a given pilot hole.

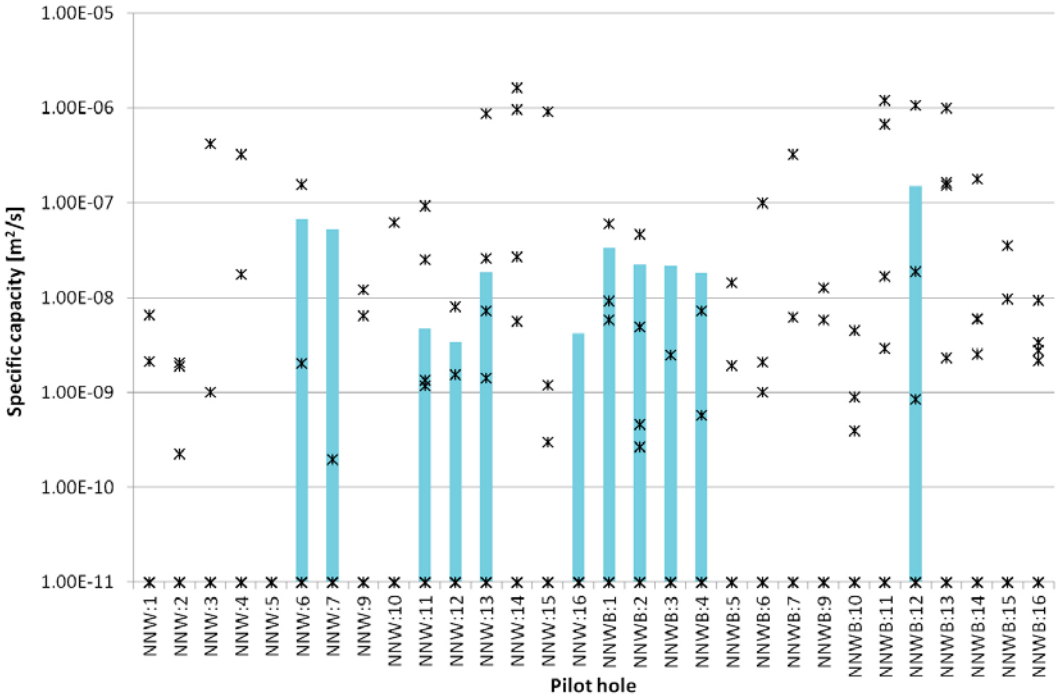

Figure 4-10. *Specific capacity in each of ten realisations of the HypoSite model without conditioning. The blue lines are the specific capacity in each pilot hole of the synthetic reality; each star is the total value into one hole for one unconditioned realisation. Values at 10^{-11} m^2/s are those where no flow was found, or where flow was negligible.*

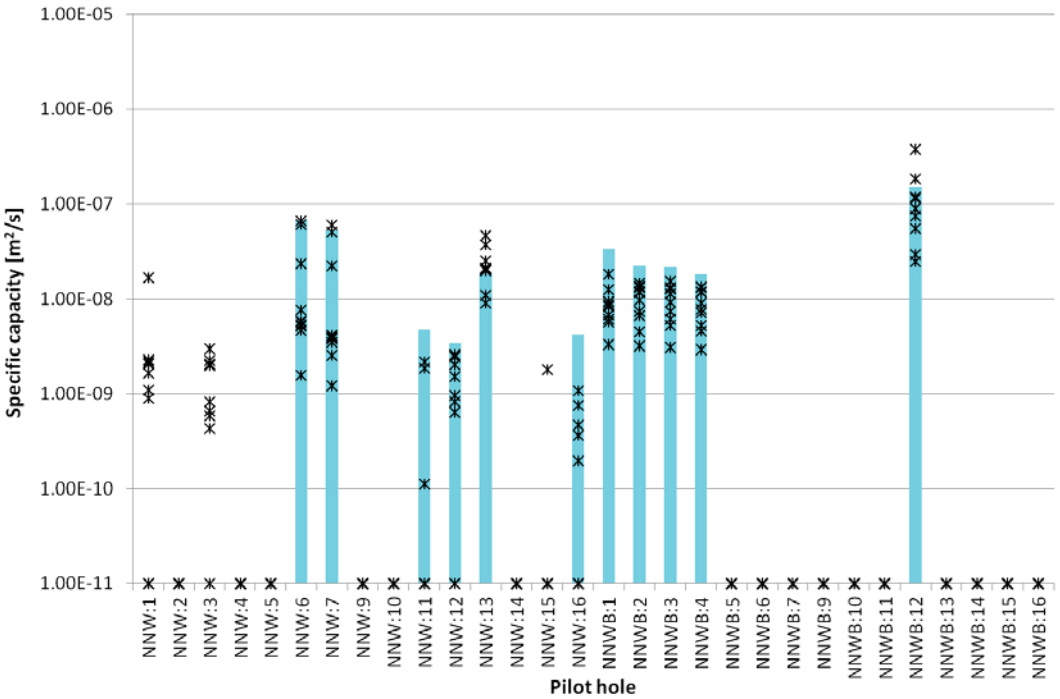

Figure 4-11. *Specific capacity in each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are the specific capacity in each pilot hole of the synthetic reality; each star is the total value into one hole for one conditioned realisation. Values at 10^{-11} m^2/s are those where no flow was found, or where flow was negligible.*

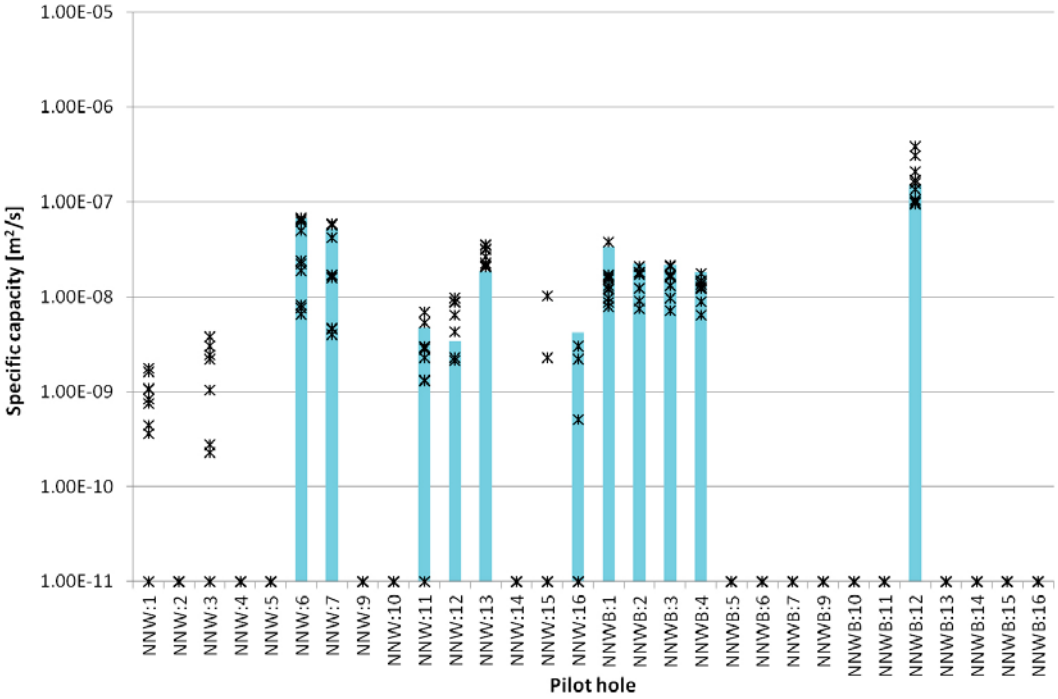

Figure 4-12. *Specific capacity in each of ten realisations of the HypoSite model after conditioning using hydraulic data. The blue lines are the specific capacity in each pilot hole of the synthetic reality; each star is the total value into one hole for one conditioned realisation. Values at 10^{-11} m^2/s are those where no flow was found, or where flow was negligible.*

4.2.2 Post-closure performance measures

The process for predicting post-closure performance measures based on conditioning using injection tests is as described in Subsection 4.1.2. The tunnels and deposition holes are filled with low permeability backfill and buffer, respectively, that is represented as a CPM surrounded with a conditioned realisation of the HypoSite DFN to represent the fractured bedrock. A boundary condition is applied to give a low flow towards the north-eastern corner of the top surface of the model. Particles are tracked from each of the deposition holes, and their paths are used to calculate U and F.

Note that, as mentioned in Subsection 2.8.7, there may be additional intersections and flows present in deposition holes that were not observed in their pilot holes due to the difference in the diameters of the holes.

Average flow-rate per unit length (U)

As with the inflow conditioning, the data in Figure 4-13, Figure 4-14 and Figure 4-15, shows how effectively U can be predicted by the conditioning methods. There is, in general, a better prediction when conditioning based on hydraulic data from injection tests. Without conditioning, it is clear that no sensible predictions on a per-hole basis can be drawn.

Flow-related transport resistance (F)

As with U, it is clear from Figure 4-16, Figure 4-17 and Figure 4-18, that there is some predictive capability of the method in calculating F for individual holes, and that this is better when conditioning using hydraulic data. There is little predictive capability for individual holes if conditioning is not done. Note also that there is some over-prediction of F present in some cases, as when conditioning on inflows and for the same reasons.

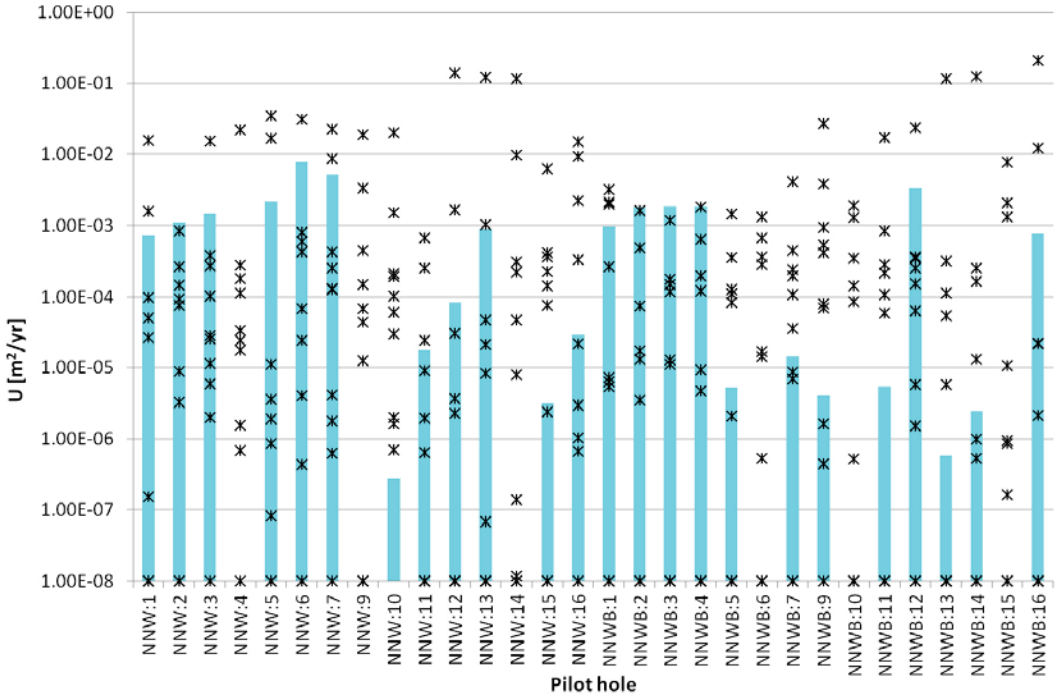

Figure 4-13. *U in each of ten realisations of the HypoSite model with no conditioning done. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one realisation. Values less than or equal to 10^{-8} m²/yr are marked as 10^{-8} m²/yr.*

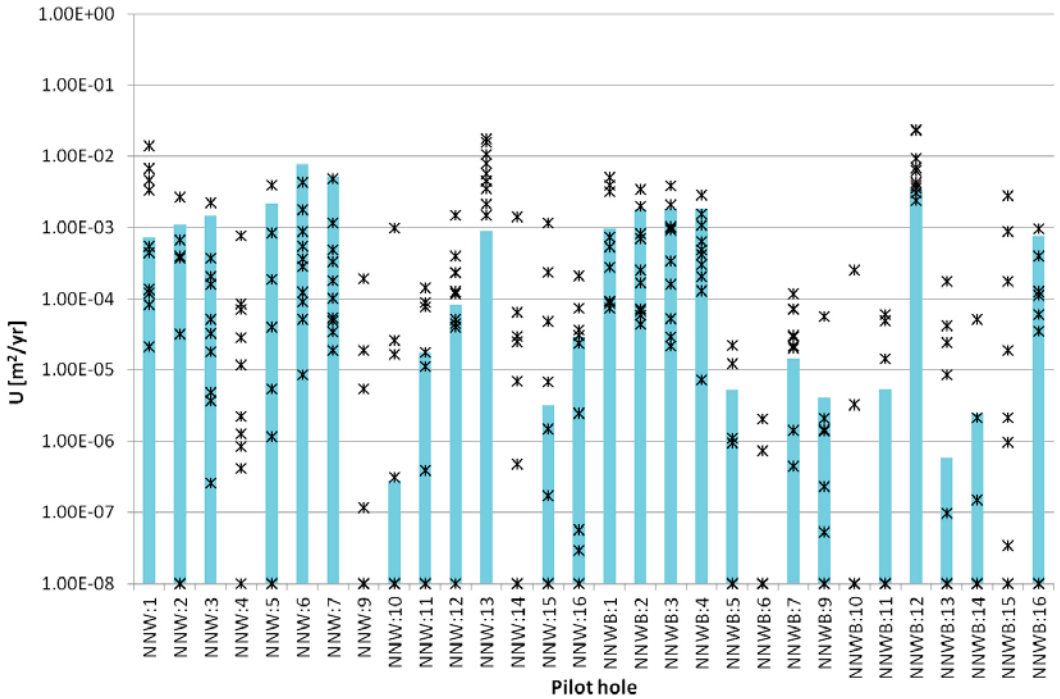

Figure 4-14. *U in each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one conditioned realisation. Values less than or equal to 10^{-8} m²/yr are marked as 10^{-8} m²/yr.*

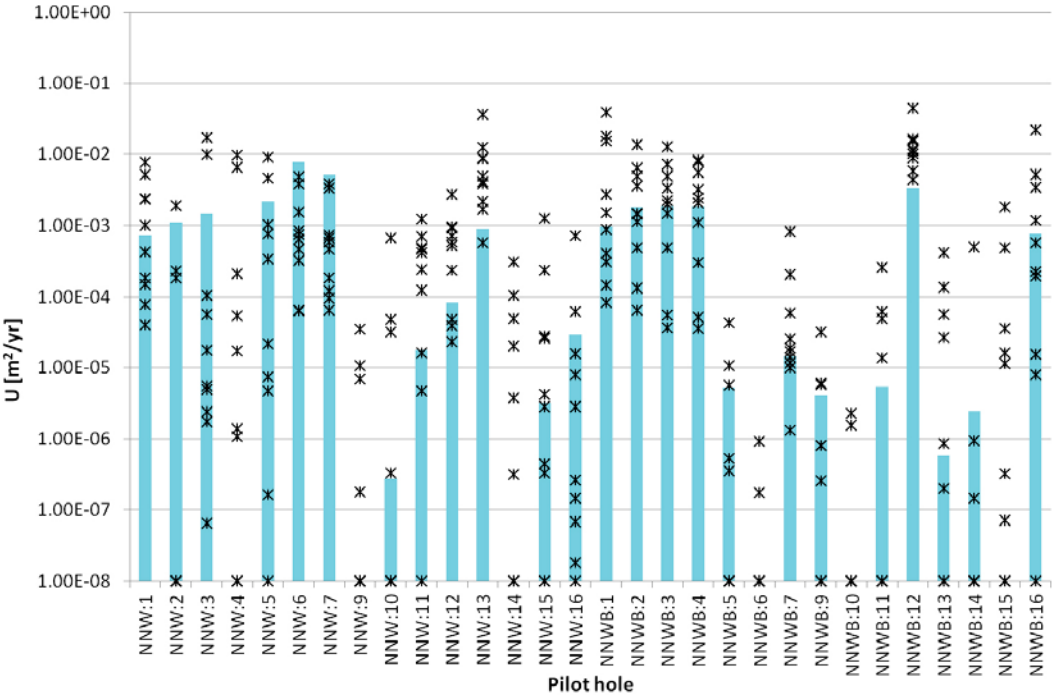

Figure 4-15. *U in each of ten realisations of the HypoSite model after conditioning using observed hydraulic data. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one conditioned realisation. Values less than or equal to 10^{-8} m^2/yr are marked as 10^{-8} m^2/yr.*

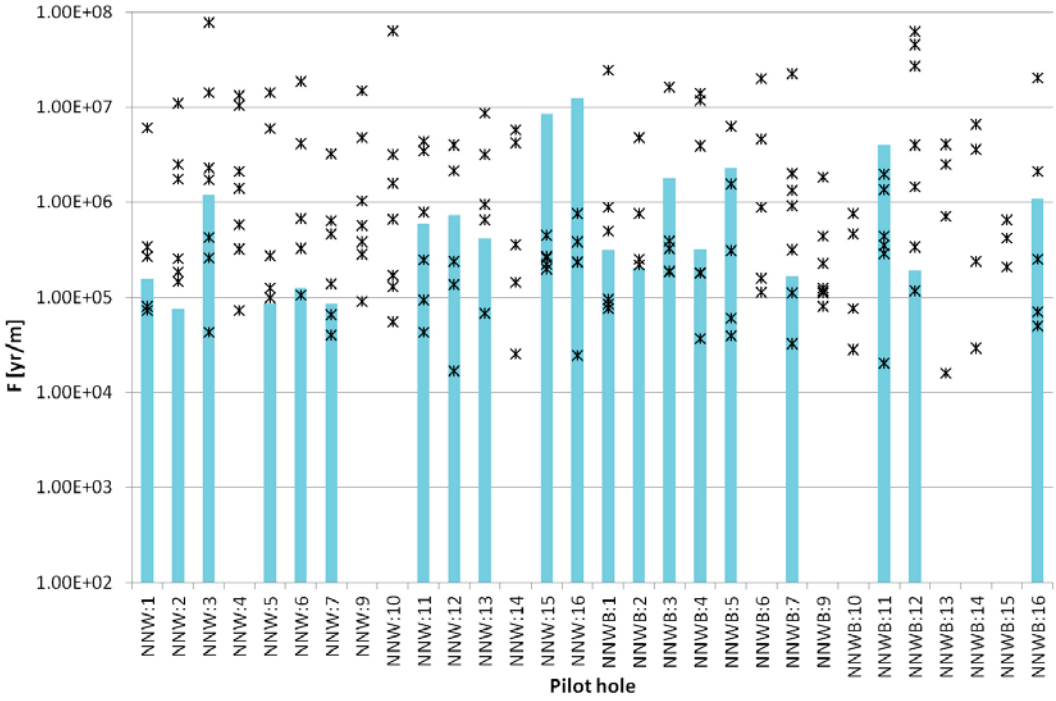

Figure 4-16. *F in each of ten realisations of the HypoSite model with no conditioning done. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one unconditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

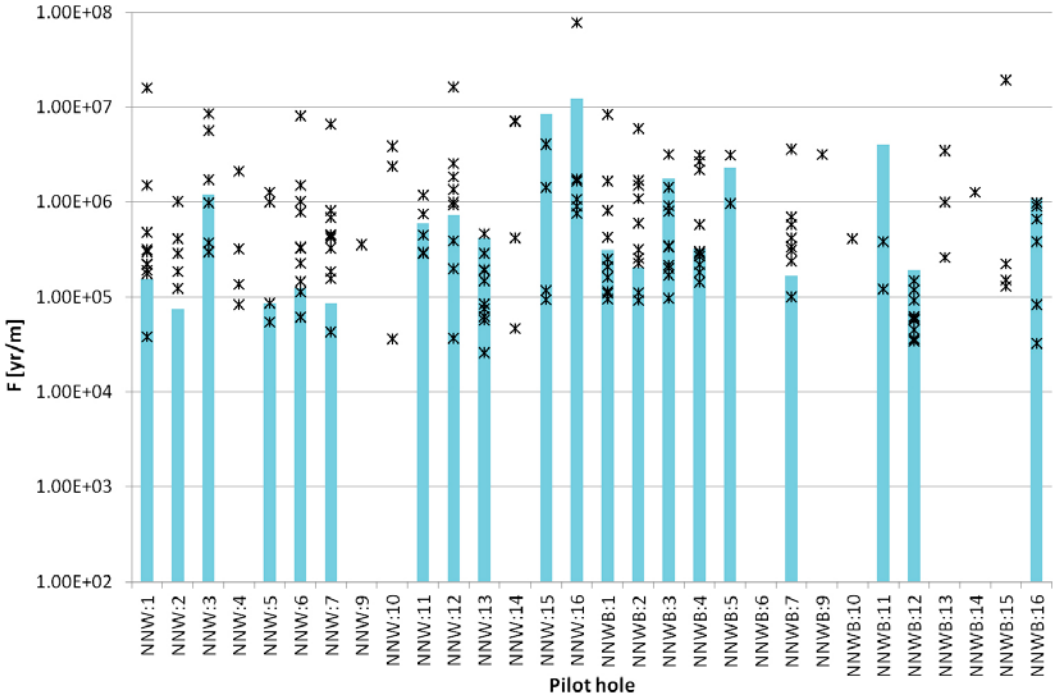

Figure 4-17. *F in each of ten realisations of the HypoSite model after conditioning using geometric data only. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one conditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

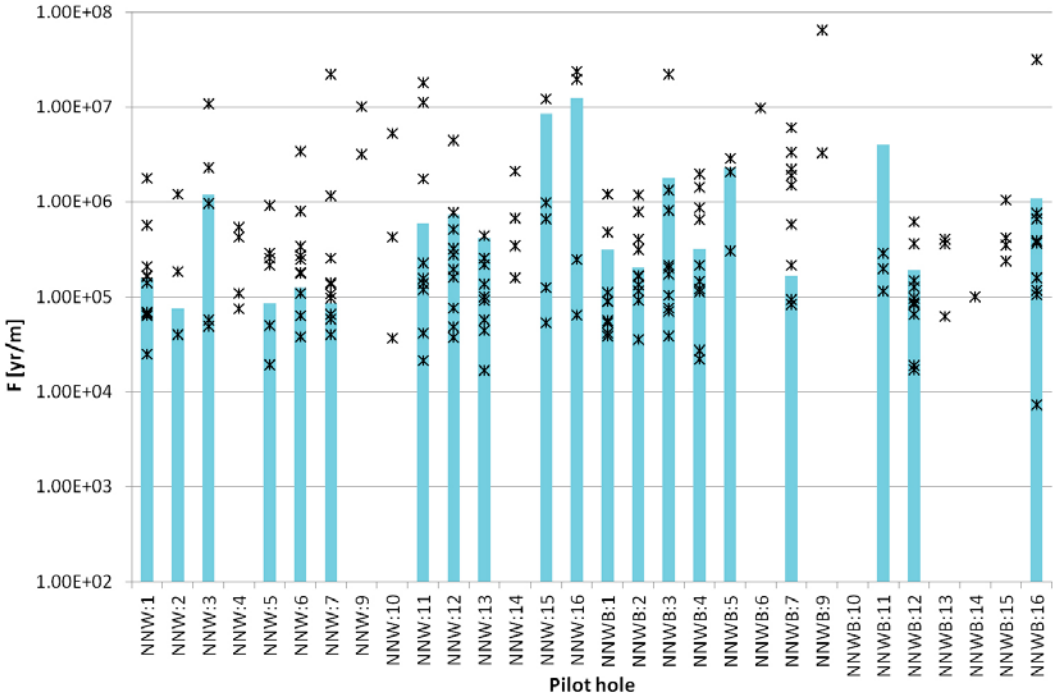

Figure 4-18. *F in each of ten realisations of the HypoSite model after conditioning using observed hydraulic data. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one conditioned realisation. Values greater than or equal to 10^8 yr/m are excluded.*

4.3 Screening realisations

While it is clear that the conditioned realisations are able to predict the U and F to a degree, it is also clear that these predictions still show significant variability, and some interpretation would be required to use them to make a decision on whether to proceed with a deposition hole. It is worth noting, however, that the realisations displayed in each case above are ten random realisations. While they have been conditioned to the injection test data (in Section 4.2) or inflow data (Section 4.1), and are compared against that data in Subsections 4.1.1 and 4.2.1, the results given for U and F are displayed for well-matched results and poorly-matched results alike.

It is thus possible to add an additional step to the process. Instead of using just ten conditioned realisations, it is possible to create a large number of conditioned realisations and then compare the results (specific capacities or inflows) against the observed hydraulic or inflow data, and choose only the realisations that match these most closely. The realisations that have a good match to the observed data are likely to be those that most closely resemble the fracture network away from the engineered openings. For example, if there is a fracture that conducts water from a series of small fractures that intersect pilot holes, the realisations with the more accurately-predicted flows are more likely to approximately represent this fracture than those with less accurately-predicted flows.

In order to get the best possible improvement in the final predictions using this method, the number of conditioned realisations initially generated should be as large as is practicable. The more realisations that are generated, the more likely it is that some of them will have flow data that closely resembles that observed. Once these have been generated, when determining which realisations pass through the screening, the user will need to choose enough realisations to allow an assessment of the deposition hole acceptance criteria, without picking realisations whose calculated flows are significantly different from those observed.

This section demonstrates the outcome of using such a screening method based on the injection test conditioning described in Section 4.2.

4.3.1 Realisation selection

There are a variety of criteria that might be used to determine what constitutes a "good" conditioned realisation, and it is not immediately obvious which criteria are most likely to lead to the best predictions for U and F in a given situation. A visual inspection of observed and conditioned hydraulic data in graphs is a useful means of determining the effectiveness of the screening method. For this demonstration, the screening method chosen was to select the realisations with the lowest mean absolute difference between specific capacities in the observed and conditioned data. In both cases, where a pilot hole had no flow registered, this was taken to be a flow of 10^{-11} m^3/s.

An initial pool of thirty conditioned realisations was generated, and the specific capacities in each pilot hole were calculated, as shown in Figure 4-19. Of these, ten realisations were selected; the specific capacities from these realisations are shown in Figure 4-20.

The screening method chosen leads to conditioned realisations that have a close visual match between observed and conditioned specific capacity on graphs such as those in Figure 4-19 and Figure 4-20, focussing particularly on high-flow pilot holes. When comparing Figure 4-20 with Figure 4-12 (the same plot with ten arbitrary realisations), the results are visually more clustered around the observed flows, and there are fewer results where conditioned results give flows that are not present in the observed data. Quantitatively, the average of the mean absolute differences for the ten random realisations was 6.11×10^{-9}, but with the screened realisations it was reduced to 3.44×10^{-9}.

4.3.2 Post-closure flow tests

A comparison between Figure 4-21 and Figure 4-15 (see page 74), and between Figure 4-22 and Figure 4-18 (see page 76), shows that the predicted U and F values for the conditioned realisations tend to move closer to the simulated observations when screening of realisations is used for some holes.

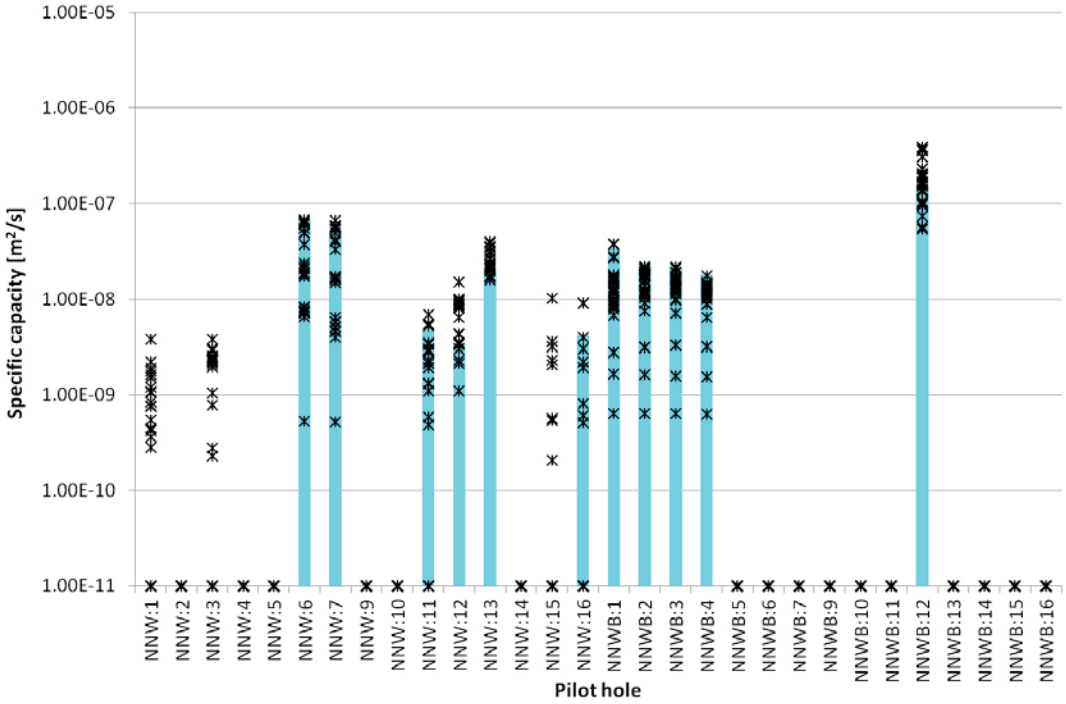

Figure 4-19. *Results of hydraulic tests in each of thirty realisations of the HypoSite model after conditioning using hydraulic data. The blue lines are the specific capacity in each pilot hole of the synthetic reality; each star is the total value into one hole for one conditioned realisation. Values at 10^{-11} m^2/s are those where no flow was found, or where flow was negligible.*

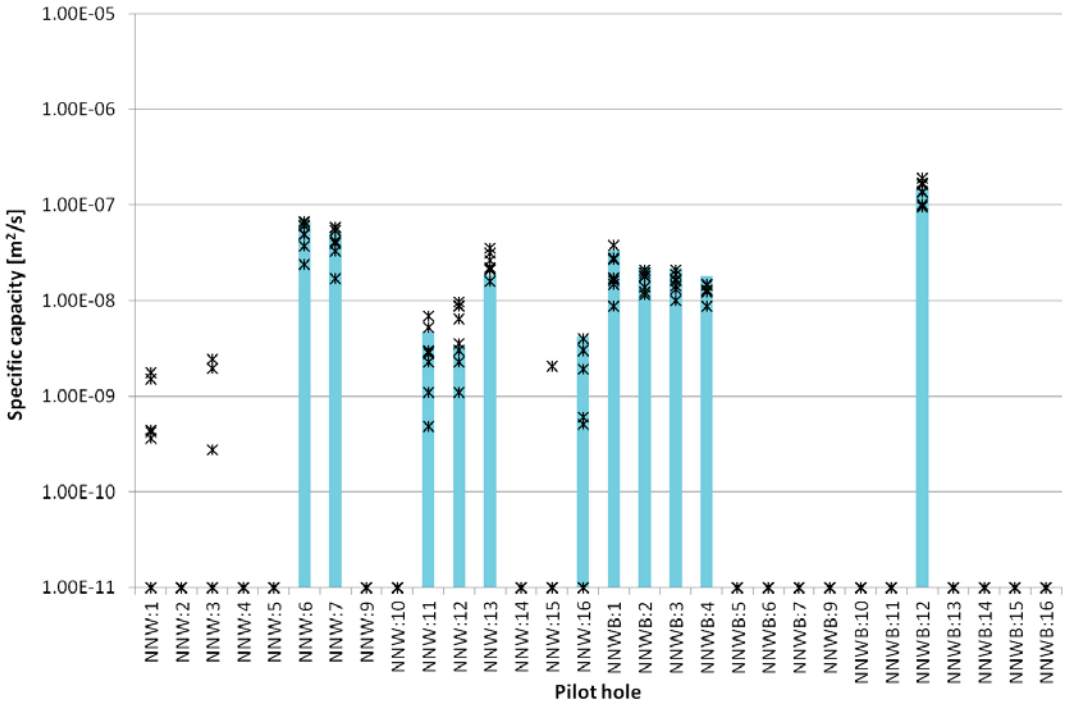

Figure 4-20. *Results of hydraulic tests in each of the ten selected realisations of the HypoSite model after conditioning using hydraulic data. The blue lines are the specific capacity in each pilot hole of the synthetic reality; each star is the total value into one hole for one selected conditioned realisation. Values at 10^{-11} m^2/s are those where no flow was found, or where flow was negligible.*

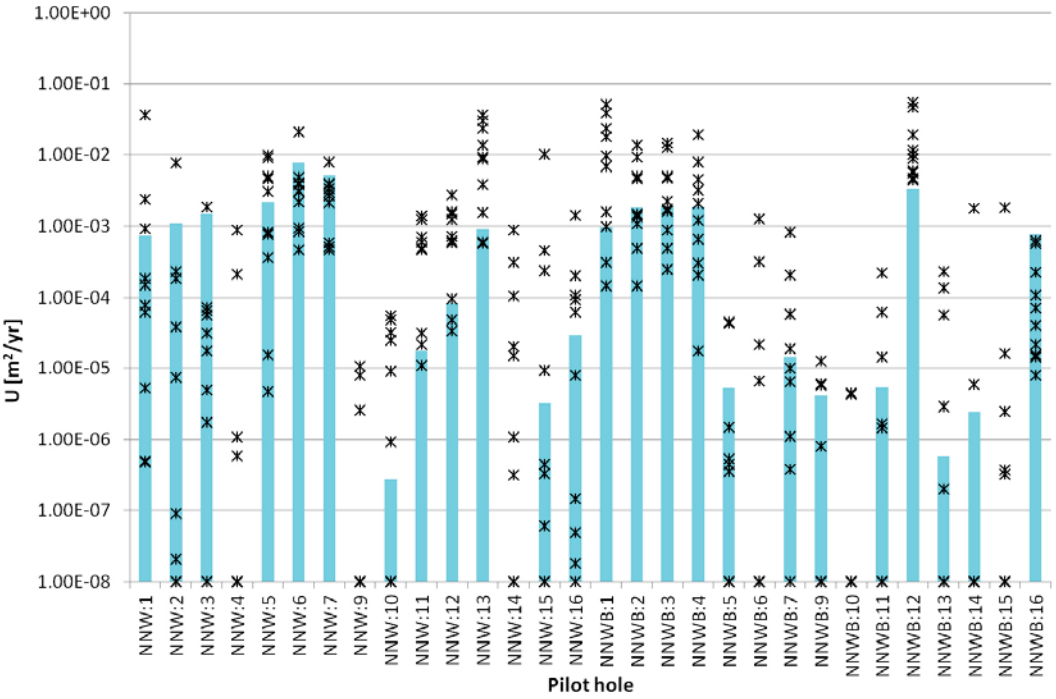

Figure 4-21. *U in each of the ten selected realisations of the HypoSite model after conditioning using observed hydraulic data. The blue lines are observations from the synthetic reality in each pilot hole; each star is the result in one hole for one screened conditioned realisation. Values less than or equal to 10^{-8} m²/yr are marked as that value.*

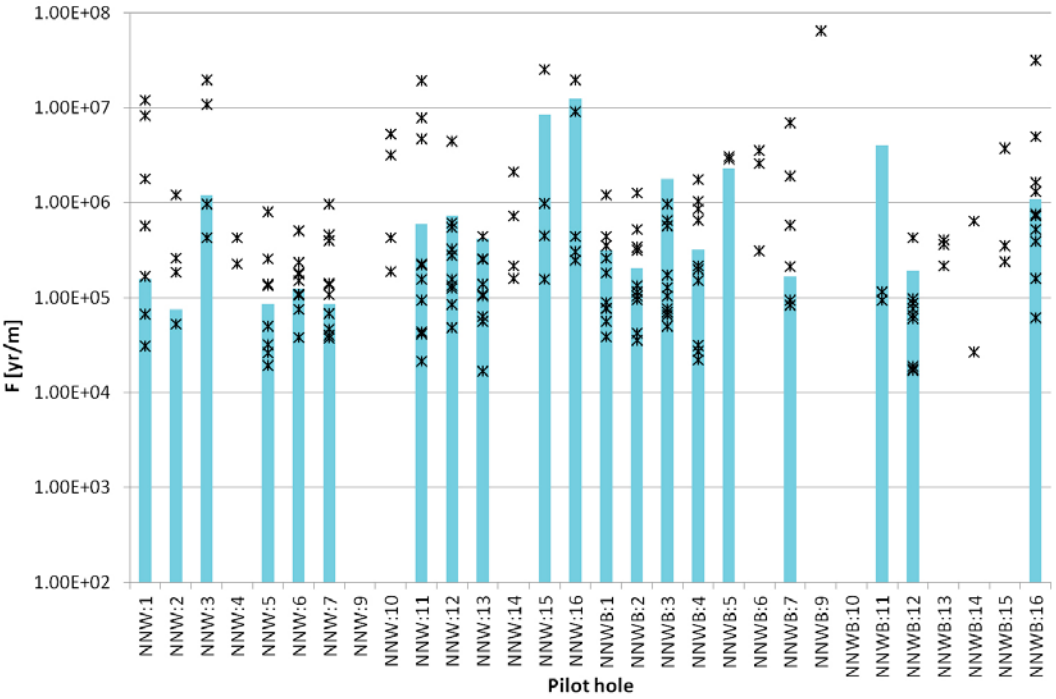

Figure 4-22. *F in each of the ten selected realisations of the HypoSite model after conditioning using observed hydraulic data. The blue lines are the observations from the synthetic reality in each pilot hole; each star is the result in one hole for one screened conditioned realisation. Values greater than or equal to 10^{8} yr/m are excluded.*

In this instance, one third of the conditioned realisations have been selected. It may be that this is too large a proportion, i.e. when selecting ten screened realisations from thirty it is necessary to include realisations that do not match the observed data closely enough to give a significant improvement. If there were a hundred realisations from which ten were chosen, it is likely that some of those hundred would be better than the ten selected here.

4.4 Quantitative comparison of results

While it is possible to see how well the conditioned realisations predict the *U* and *F* from the figures in Sections 4.1.2, 4.2.2 and 4.3.2, it is useful to be able to quantify these differences. Figure 4-23 and Figure 4-24 thus show how close the *U* and *F* from each conditioned realisation was to the corresponding value for the HypoSite synthetic reality.

The data shown on the graphs is the mean absolute difference in *Log(U)* and *Log(F)* between conditioned realisations and the synthetic reality, averaged over deposition holes and over conditioned realisations. The lower the result, the closer the conditioned realisation is to the synthetic reality. In each comparison between a conditioned realisation and the synthetic reality, a maximum mean absolute difference of 2 has been imposed. Imposing a maximum value in this way means that, if the hole has no flow in a given conditioned realisation, but has flow in the synthetic reality, this discrepancy can be accounted for in some sense for both *U* and for *F*. It also means that the maximum possible difference in *Log(U)* or *Log(F)* in these plots is 2. On the other hand, if a deposition hole is observed as not having significant flow, it is clear without the need for conditioning that it is unlikely to start flowing significantly when the hole is backfilled; thus holes where *U* for the synthetic reality is less than 10^{-4} metres per second are not included in this calculation.

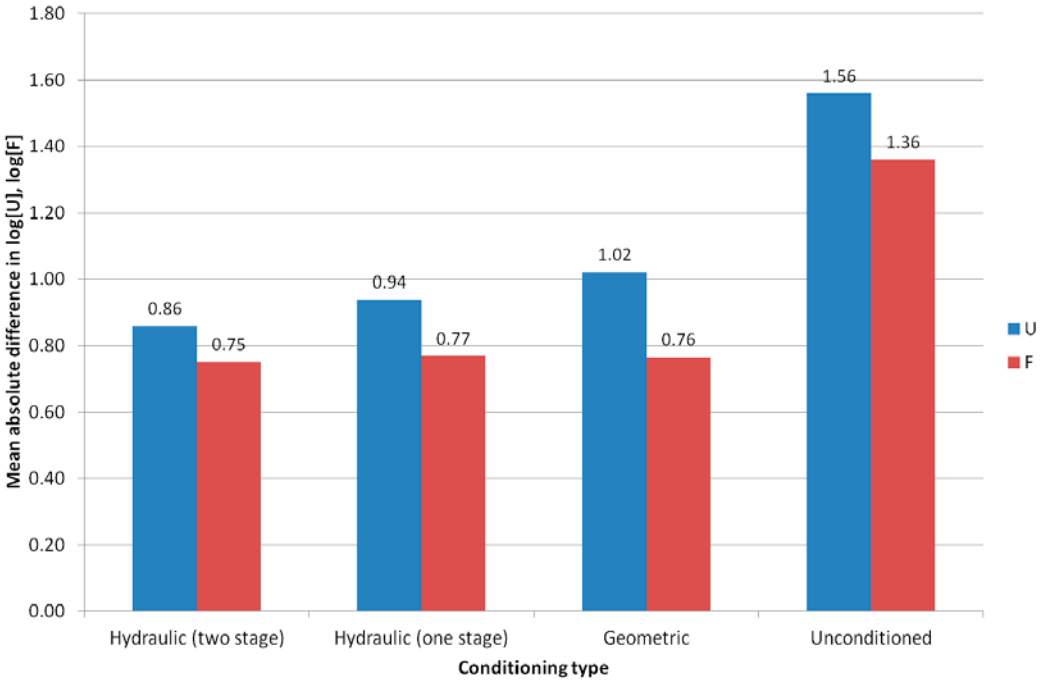

Figure 4-23. *Comparison between predictions of U and F for pilot holes for deposition holes for each scenario: "Hydraulic" results consider injection tests, "Geometric" results consider geometric data only and "Unconditioned" results do not include conditioning at all. The "two stage" process screens realisations, as described in Section 4.3. Other results are as described in Section 4.2.*

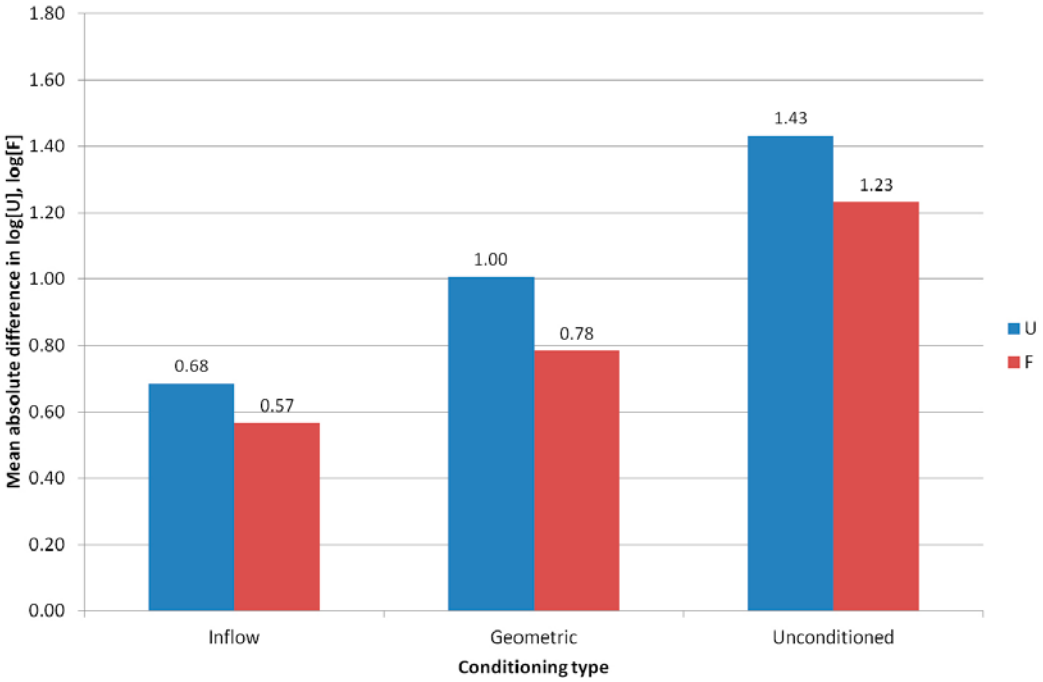

Figure 4-24. *Comparison between predictions of U and F for deposition holes for each scenario: "Inflow" results consider inflow tests, "Geometric" results consider geometric data only and "Unconditioned" results do not include conditioning at all. Results are described in Section 4.1.*

The graphs show that, as one would expect, the more data is made available for each intersection to be conditioned, the more accurate the resulting predictions become. The most improvement is gained simply by using a conditioned model as opposed to an unconditioned model. Unconditioned models are not good at predicting performance measures for individual holes. Adding additional types of observation to the conditioning gives improvements in the performance measure predictions that can be relatively small in several cases.

Conditioning on inflows to deposition holes gives the best predictions of *U* and *F*. However, there may be technical difficulties in obtaining these measurements in practice. A hypothetical set of hydraulic injection tests in deposition holes may also produce relatively close results, potentially closer than those using inflows, but it is felt unlikely that this will be possible in practice and so this situation has not been simulated.

4.5 Use of transient data

The flow data used thus far has been entirely based on results obtained or calculated at steady-state. This provides a single value for each observation that is convenient for use in the conditioning methodology and provides useful information about the water-conducting fracture network close to the observation point. However, transient data has the potential to provide additional information about the fracture network away from the observation point based on the variation in the flow rate over time (due to the time taken for the pressure change to reach free-flowing areas and choke points). This information can be used as additional screening criteria for the method described in Section 4.3.

The user is free to decide how this transient data can be applied as screening criteria and several approaches are available. This section, however, considers only one approach.

As when conditioning on steady-state flows (as described in Subsection 3.1.2), to ensure that the transient flows generated were consistent with other HypoSite data, an attempt was made to replicate a flow simulation performed as part of the delivery of flow information for HypoSite (Follin 2015). While there are differences between the FracMan results and the ConnectFlow results, the results are considered similar enough that ConnectFlow could be used to produce acceptable simulated observed transient data.

Once this was complete, transient flow data from an injection test in each pilot hole for a deposition hole in the HypoSite model was calculated individually using ConnectFlow. As with the steady-state simulations, the over-pressure in each pilot hole was 20 metres of head.

Transient parameters were set as follows:

- The storativity was set to:

$$S = (7 \cdot 10^{-4})T^{0.35} \quad (4\text{-}1)$$

where T is transmissivity, as in Follin (2015).

- Time step size varied from 1 second to 5 days, with the time step interval increasing gradually over time.
- Fractures intersecting pilot holes were discretised very finely near the pilot holes, such that each sub-fracture had a side length of 20 centimetres. This is needed to resolve the detailed pressure profile and flows around the pilot holes.

These simulations allowed the calculation of the time evolution of specific capacity as simulated observations. In line with Follin (2015), the reciprocal of specific capacity, and the derivative of the same reciprocal, were also calculated. Equivalent values were also calculated for each of the thirty realisations conditioned as part of Section 4.3.

Figure 4-25 shows the transient evolution of flow in each pilot hole that intersects one or more fractures in the synthetic reality (HypoSite realisation 4). Pilot holes with no flow are not included on the graph and those that are not connected to the wider fracture network are coloured red. It is noted that, particularly in cases where the fractures connected to a pilot hole formed an isolated cluster (i.e. red on Figure 4-25), numerical issues prevented calculation of particularly small flows. This in some cases led to results with odd kinks or that did not tail off quite as would normally be expected.

From this graph, it is clear that there are three visually distinct categories of pilot hole:

- Isolated pilot holes, not shown on Figure 4-25 because it is known that there will be no flow.
- Pilot holes connected to isolated clusters. These are indicated with red lines in Figure 4-25.
- Pilot holes connected to the wider fracture network. These are indicated with green and blue lines in Figure 4-25.

However, these categories are already distinguishable based on results from steady-state simulations (Figure 4-12, Subsection 4.2.1):

- In isolated pilot holes, it is known from the observed data and conditioning output that no intersections exist.
- In pilot holes connected to isolated clusters, it is known that intersections exist but that the (steady-state) specific capacity in Figure 4-12 is zero.
- In pilot holes connected to the wider fracture network, the specific capacity in Figure 4-12 is non-zero.

Based on these results, there is thus not an obvious benefit from transient tests if steady-state results are available. However, there is too little data to rule out the possibility that benefits might arise in the general case and it is not demonstrated that other correlations in the transient data will not prove valuable.

In the field, obtaining steady-state flow results for injection tests may be prohibitive to operations. A five-day injection test would certainly delay operations, and during this time period no other injection test can be run (in case they interfere). Even though the isolated fracture clusters connected in the HypoSite example are known to be relatively small, it is noted from Figure 4-25 that it is not necessarily possible to distinguish them from low-flow connected clusters in the early part of the test. This work thus confirms the conclusion of Follin (2015) that a significant period of time (greater than 20 minutes) is needed to corroborate whether a fracture connected to a pilot hole is connected to the wider fracture network or not. It appears in this realisation, however, that results have effectively reached their steady-state after five days (see Figure 4-26), and in fact about 3 hours would be sufficient in most cases.

It is possible to create graphs similar to Figure 4-25 for the conditioned realisations. These could, in principle, be used to screen realisations as in Section 4.3 – but this is only relevant if it is possible to usefully distinguish different situations from the data in Figure 4-25. Two such realisations are presented here as Figure 4-28 and Figure 4-29, both chosen as part of the screening process in Section 4.3. Each pilot hole is given the same colour as in Figure 4-25 (i.e. based on the status of the fracture connectivity in the synthetic reality, not the conditioned realisation).

In Figure 4-28, it is clear that the isolated fracture clusters are not correctly predicted in this realisation. Two of the red lines (isolated clusters in the synthetic reality (as shown in Figure 4-25) are connected to the wider fracture network in the conditioned realisation and one of the green lines (connected clusters associated with deposition tunnel NNWB) behaves like an isolated cluster. It is clear from the steady-state results that this analysis is correct: the isolated clusters come from NNW:3 and NNWB:8. In Figure 4-29, the correct three holes are connected to isolated fracture clusters, but it is noticeable that NNW:11 and NNWB:8 (the highest 5-day values for NNW clusters and NNWB clusters respectively) still have significant gradients even at 5 days. Closer inspection of the region in question (as shown in Figure 4-27) reveals that the flow route out of NNW:11, in particular, is somewhat convoluted, requiring flow through a series of low-transmissivity fractures, i.e. there is a choking effect. Although this effect is not observed in the synthetic reality, it may be an additional category of transient flow data that could be used generally.

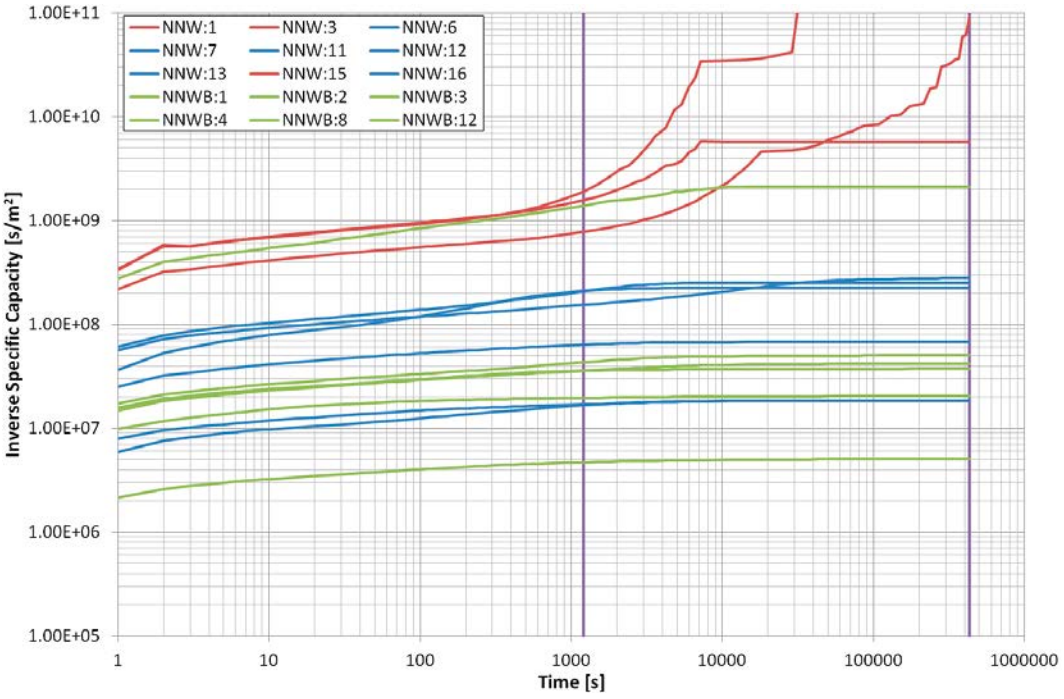

Figure 4-25. *Variation in inverse specific capacity over time for HypoSite realisation 4. Red lines show injection into pilot holes that are only connected to isolated clusters of fractures (all associated with tunnel NNW); blue lines show injection into pilot holes associated with tunnel NNW and green lines show injection into pilot holes associated with tunnel NNWB. Vertical purple lines cross at times 20 minutes and 5 days.*

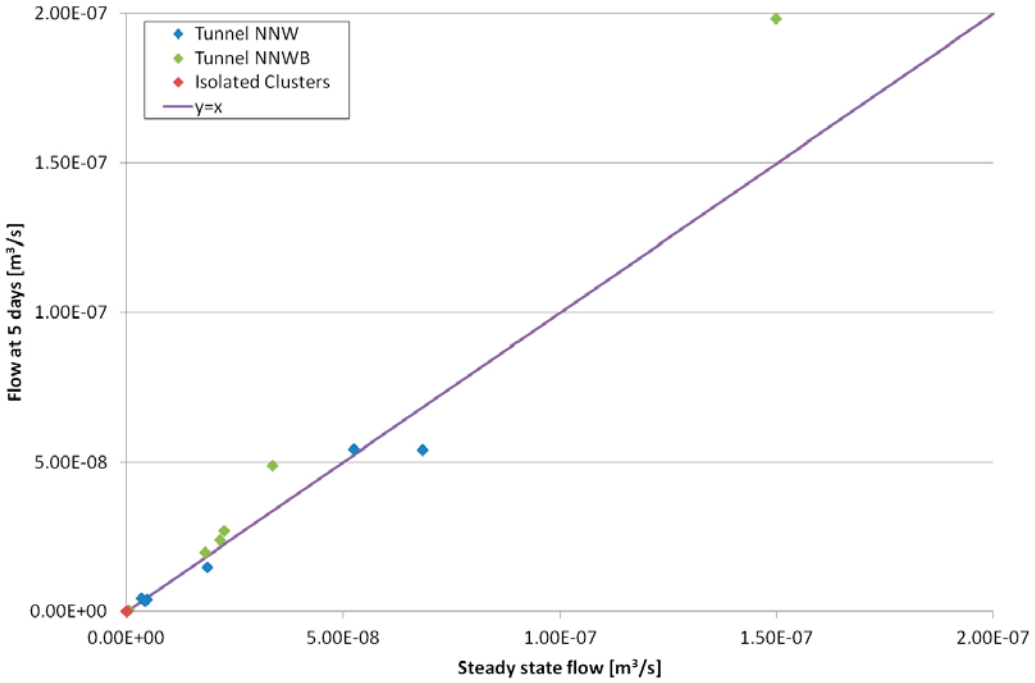

Figure 4-26. *Comparison between steady-state flow and transient flow after 5 days. Blue dots are in pilot holes associated with tunnel NNW and green dots are in pilot holes associated with tunnel NNWB. Pilot holes connected to isolated clusters are marked at zero flow with a red dot. The purple line indicates the line where the two values would be the same.*

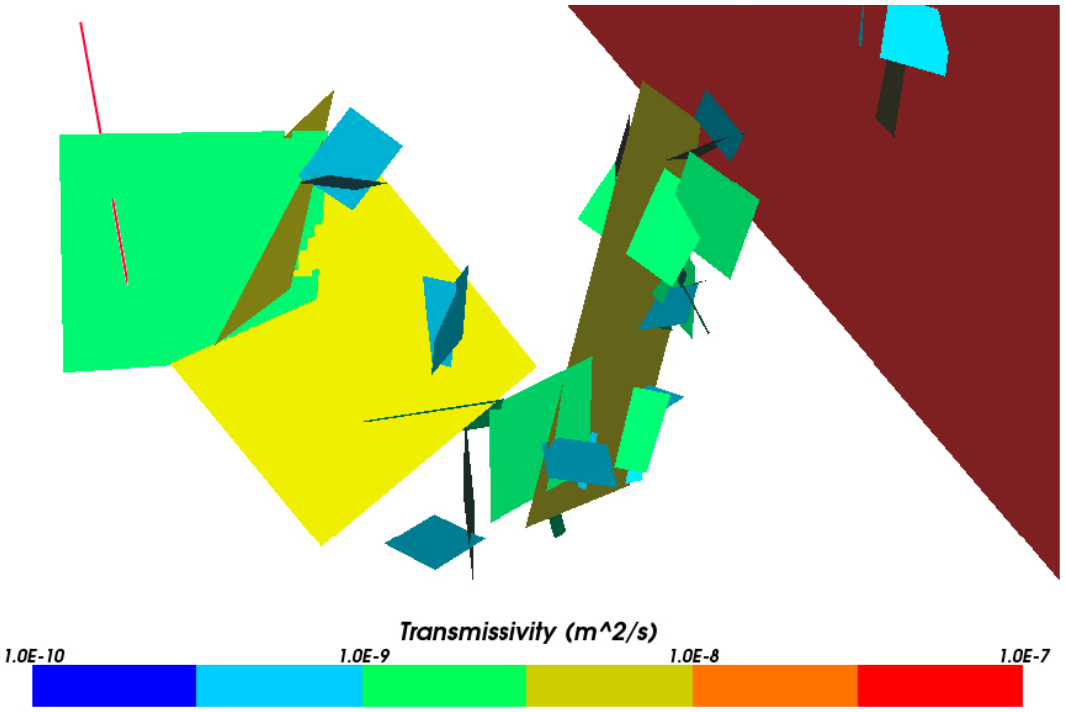

Figure 4-27. *Depiction of the flow route from pilot hole NNW:11 (red line to the left) to the network (through the large red fracture to the right) in conditioned realisation 28. Fractures are coloured by transmissivity. It is clear that the only route to the network traverses a number of fractures coloured green and blue, i.e. with relatively low transmissivity, implying that, while a flow route exists, it may be choked. Fractures not contributing to this flow route have been removed from the image for ease of viewing.*

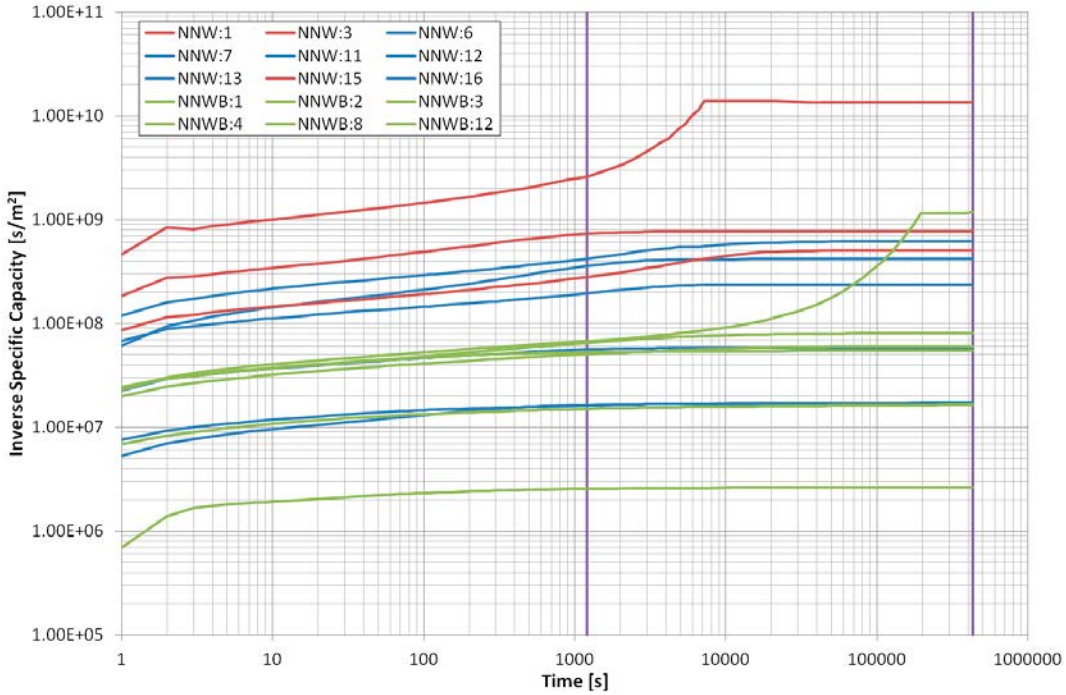

Figure 4-28. *Variation in inverse specific capacity over time for conditioned realisation 23. Colours are as in Figure 4-25: red lines show injection into pilot holes that are only connected to isolated clusters in the synthetic reality; blue lines show injection into pilot holes associated with tunnel NNW and green lines show injection into pilot holes associated with tunnel NNWB. Vertical purple lines cross at 20 minutes and 5 days.*

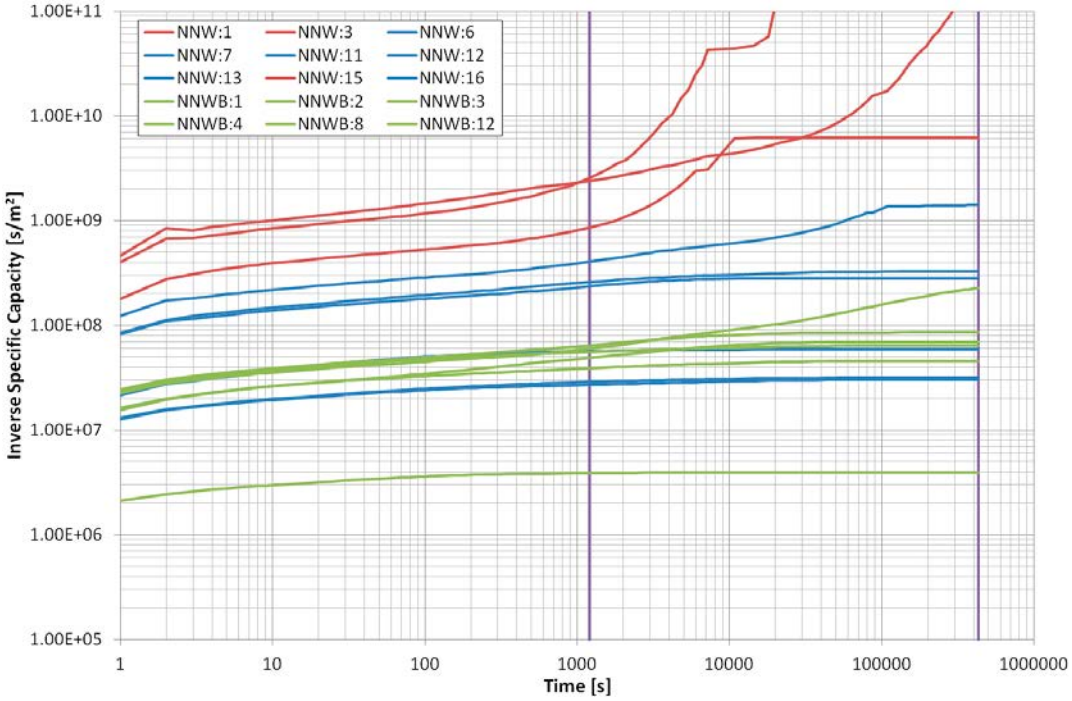

Figure 4-29. *Variation in inverse specific capacity over time for conditioned realisation 28. Colours are as in Figure 4-25: red lines show injection into pilot holes that are connected to isolated clusters in HypoSite realisation 4; blue lines show injection into pilot holes associated with tunnel NNW and green lines show injection into pilot holes associated with tunnel NNWB. Vertical purple lines cross at 20 minutes and 5 days.*

Figure 4-30 shows the transient flow in each conditioned realisation in NNW:15, which is connected to an isolated fracture cluster in the synthetic reality and in 22 of the 30 conditioned realisations. Figure 4-31 shows the equivalent plot for NNW:11, which is connected to the wider fracture network in the synthetic reality and in 22 of the 30 conditioned realisations. While there are several pilot holes that are connected to the wider fracture network in every conditioned realisation, this one was chosen because there are no fractures observed in it that also intersect other engineered openings.

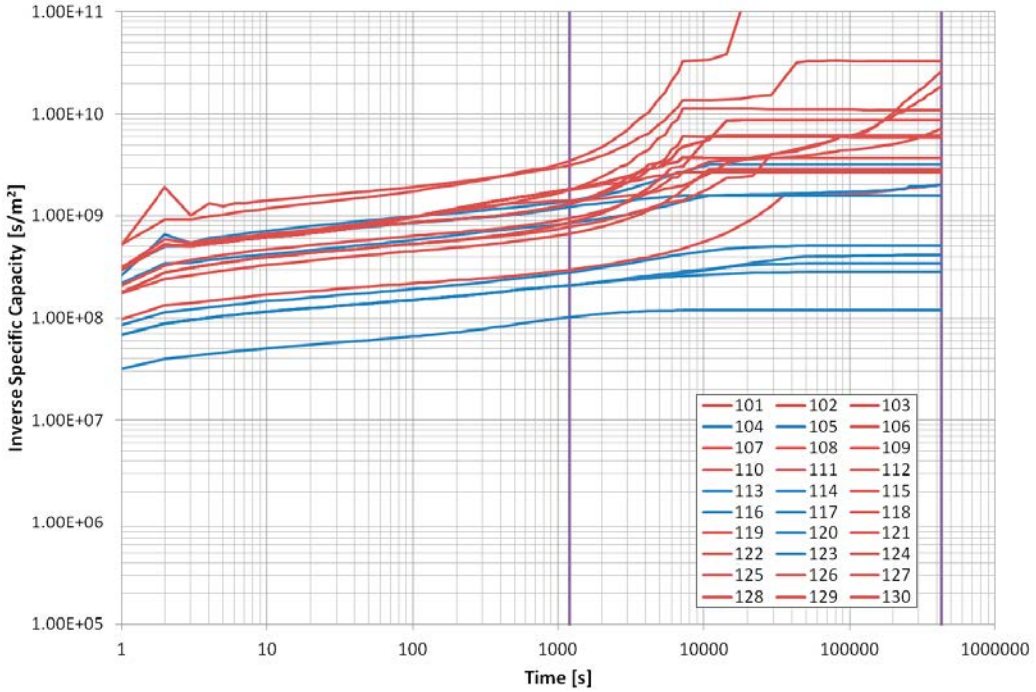

Figure 4-30. *Transient evolution of inverse specific capacity for borehole NNW:15 for each of thirty conditioned realisations. Red lines are for fracture clusters that appear isolated in steady-state simulations; blue lines indicate fractures that are connected to the wider fracture network.*

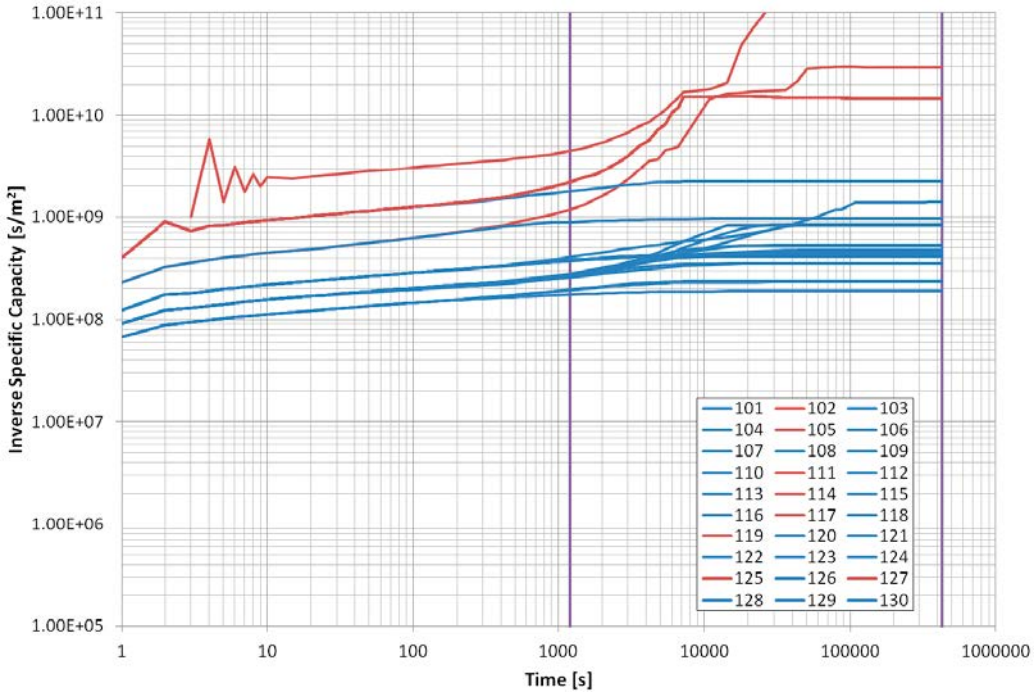

Figure 4-31. *Transient evolution of inverse specific capacity for borehole NNW:11 for each of thirty conditioned realisations. Red lines are for fracture clusters that appear isolated in steady-state simulations; blue lines indicate fractures that are connected to the wider fracture network.*

In the initial period in Figure 4-31, the thirty realisations divide themselves into six distinct groups of lines. Each of these likely corresponds to a different candidate fracture that has been chosen from the library (by weighted random selection, so this will be different in different realisations), as the injected water has not yet reached the wider fracture network. It is possible to divide these into three groups based on how the lines divide between isolated clusters and clusters connected to the wider fracture network in the period between 5 minutes and 20 minutes (this is presumably the point at which the water either reaches the intersection with the next fracture in the network, or fails to do so). One choice of candidate fracture is always isolated. Three choices of candidate fracture are always connected to the wider network. The other two choices are fractures that may be isolated or may be connected to the wider network, depending on the surrounding fractures.

4.6 Issues for further consideration

The preceding analysis demonstrates possible methods for predicting the performance measures U and F for a series of conditioned realisations, and how they might compare with the actual U and F in a synthetic reality (HypoSite BM-1b realisation 4).

However, it is noted that the analysis of the results raises two related and unresolved questions as to how the methodology might be used in the future. It is perhaps useful to ask these questions, even though it is not within the scope of this project to answer them, so as to identify issues that might be considered in more detail at a more appropriate time.

Firstly, a question remains as to how one might use the data to determine whether to drill and use a deposition hole based on the results found. Conditioning provides useful input to this decision, but the distributions provided do not provide a clear-cut yes or no answer. Secondly, if the screening method described in Section 4.3 is used, a question remains as to how to select realisations that give the most accurate or most useful input into that determination.

While a method for selecting realisations has been provided and demonstrated in Section 4.3, it is not the only option available. The screening method may not select the best realisations to give the clearest and most reliable answer to the question as to whether to use a deposition hole or not. In particular, while selection involving transient data was not attempted in Section 4.5 because it was not clear that it would add anything in this case, information gained from transient data may nonetheless be a useful tool in other cases. It would be of benefit to use a larger model or larger range of realisations to determine how useful it might prove to be.

The second question clearly feeds into the first. If a selection method were found, for example, that did not achieve accurate quantitative measures but was successful in predicting in a binary manner whether the U or F was higher or lower than a given threshold, for example, the deposition hole acceptance rules would likely be different to those for a selection method that gave a more probabilistic answer.

Both of these issues would be useful avenues of research in conditioning, and a tool such as HypoSite, in which the U and F can be calculated, is invaluable for this in that it allows the user to judge exactly how well the methods have performed compared with the synthetic reality. The methods may then be altered and repeated to optimise the outcome.

5 Summary

The conditioning method presented in this report is intended to allow the creation of structural models that closely reflect measured data in tunnels and pilot holes, thereby giving the user a far better understanding of the fracture network around a repository and of the risk associated with disposal in each deposition hole. If rejection criteria for deposition holes can be applied successfully to constrain repository performance measures, the overall risk associated with the repository may be significantly reduced.

The ConnectFlow method for conditioning is fundamentally to create an empirical distribution of fractures in a library, and choose fractures from the library to match the fractures associated with the observed data. As the statistical recipe used to create the fractures in the library is consistent with the observed data (at least to the best of our understanding and given modelling constraints) and with the unconditioned fracture network model, the conditioned model should not have significantly different statistical properties from the observed data. It should, however, more accurately represent the local conditions around the repository than a purely stochastic realisation. Chapter 2 of this report discusses the methodology of the model in detail.

While this is theoretically sound, it is important to ensure that the model works correctly in practice as well. To test the data more fully, a series of calibration tests and consistency checks were defined and have been put to use to test the model (Chapter 3). While not perfect in every case, the results demonstrated that the methodology does work, produces conditioned models that are consistent with the prescribed DFN recipe and produces reasonable and accurate results in general. It is believed that these results may have been even better for a model representing a full repository panel of long parallel tunnels with more deposition holes, as this would have allowed the assumption of stationarity used by this method to be exploited more fully. The geometry of large fractures intersecting several tunnels that can have a strong influence on the panel-scale flow distribution would also have been more certain.

The objective of the conditioning methodology is ultimately to provide information about the groundwater flow in fractured bedrock, and specifically the likely post-closure flows around deposition holes. Given that the observed data being used in conditioned models is specific to each location in the tunnel or pilot hole, it is unsurprising that when flow data was introduced to measure performance in Chapter 4, it was possible to predict observed flows (either inflows to deposition holes or injection tests in pilot holes for deposition holes) far more accurately than when no conditioning was used and hence no location-specific information was available. Based on this result, a method was thus demonstrated for predicting, probabilistically, the post-closure average flow-rate per unit length (U) near a deposition hole and flow-related transport resistance (F) in a given model using observed flow data. This was completed firstly with ten random conditioned realisations, and secondly with ten selected from thirty random conditioned realisations. In each case, the predictions with flow taken into account during conditioning were better than those using only geometric data, but both were far more accurate than the predictions made without any conditioning at all.

References

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

Almeida J, Barbosa S, 2008. 3D stochastic simulation of fracture networks conditioned both to field observations and a linear fracture density. In Proceedings of the 8th International Geostatistics Congress, Santiago de Chile, Chile, 1–5 December 2008.

Amec Foster Wheeler, 2015. ConnectFlow, version 11.3. Aberdeen: Amec Foster Wheeler.

Amec Foster Wheeler, 2016. ConnectFlow Technical Summary, release 11.4. Aberdeen: Amec Foster Wheeler.

Baxter S, Hartley L, Witterick W, Fox A, 2016a. Database design and prototyping for Olkiluoto Discrete Fracture Network (DFN) in support of site descriptive modelling 2016. Posiva Working Report 2016-13, Posiva Oy, Finland.

Baxter S, Appleyard P, Hartley L, Hoek J, Williams T, 2016b. Exploring conditioned simulations of discrete fracture networks in support of hydraulic acceptance of deposition holes – application to ONKALO Demonstration Area. Posiva SKB Report 07, Posiva Oy, Svensk Kärnbränslehantering AB.

Cordes C, Kinzelbach W, 1992. Continuous groundwater velocity fields and path lines in linear, bilinear and trilinear finite elements. Water Resources Research 28, 2903–2911.

Follin S, 2015. Numerical Simulation of Hydrotests close to repository tunnels. Report 1451240024, Golder Associates AB.

Follin S, Hartley L, Rhén I, Jackson P, Joyce S, Roberts D, Swift B, 2013. A methodology to constrain the parameters of a hydrogeological discrete fracture network model for sparsely fractured crystalline rock, exemplified by data from the proposed high-level nuclear waste repository site at Forsmark, Sweden. Hydrogeology Journal 22, 313–331.

Golder, 2012. FracMan, version 7.4. Golder Associates.

Hartley L, Hoek J, Swan D, Appleyard P, Baxter S, Roberts D, Simpson T, 2013. Hydrogeological modelling for assessment of radionuclide release scenarios for the repository system 2012. Posiva Working Report 2012-42, Posiva Oy, Finland.

Hestir K, Martel S J, Yang J, Evans J P, Long J C S, D'Onfro P, Rizer W D, 2001. Use of conditional simulation, mechanical theory, and field observations to characterize the structure of faults and fracture networks. In Evans D D, Nicholson T J, Rasmussen T C (eds). Flow and transport through unsaturated fractured rock. 2nd ed. Washington, DC: American Geophysical Union, 61–72.

Hjerne C, Komulainen J, Aro S, Winberg A, 2016. Development of hydraulic test strategies in support of acceptance criteria for deposition hole positions – Results of hydraulic injection tests in ONKALO DT2 pilot holes for experimental deposition holes. Posiva Working Report 2016-06, Posiva Oy, Finland.

Joyce S, Simpson T, Hartley L, Applegate D, Hoek J, Jackson P, Roberts D, Swan D, Gylling B, Marsic N, Rhén I, 2009. Groundwater flow modelling of periods with temperate climate conditions – Laxemar. SKB R-09-24, Svensk Kärnbränslehantering AB.

Joyce S, Simpson T, Hartley L, Applegate D, Hoek J, Jackson P, Swan D, Marsic N, Follin S, 2010. Groundwater flow modelling of periods with temperate climate conditions – Forsmark. SKB R-09-20, Svensk Kärnbränslehantering AB.

Klimczak C, Schultz R A, Parashar R, Reeves D M, 2010. Cubic law with aperture-length correlation: implications for network scale fluid flow. Hydrogeology Journal 18, 851–862.

Tran N H, Rahman S S, 2006. Modelling discrete fracture networks using neuro-fractal-stochastic simulation. Journal of Engineering and Applied Sciences 1, 154–160.

Zimmerman R W, Bodvarsson G S, 1994. Hydraulic conductivity of rock fractures. LBL-35976, Lawrence Berkeley Laboratory, University of California.

Appendix A

Key Quantities

A1 Tunnels

On a given tunnel, the geometry of a fracture trace is effectively characterised by the following quantities:

- The distance of the start point of the trace in the direction along the axis of the tunnel, from some reference location.
- The distance of the end point of the trace in the direction along the axis of the tunnel, from some reference location.
- The azimuthal angle of the start point measured from the vertical direction around the central axis of the tunnel.
- The azimuthal angle of the end point measured from the vertical direction around the central axis of the tunnel.
- The dip angle of the fracture that caused the trace measured between 0 and 180 degrees (allowing an indication of the dip direction), called the "apparent angle" to distinguish it from a conventional dip angle.

Because of the assumption that the fracture distribution is stationary along the tunnel and between tunnels in the same domain, the first two of these points are unimportant to the matching process. A fracture from any part of the tunnel can be moved along the axis of the tunnel to any other part of the tunnel, provided both are in the same domain. However, it remains useful to categorise the fracture by its length, which is defined based on the sum of the straight-line segments that make up the trace.

The start point and end point are taken to be such that the end point is clockwise from the start point in terms of azimuthal angle as shown in Figure A-1. If the tunnel is vertical (i.e. a shaft), the "vertical" direction on the cross-section is taken to be north. These measures allow an idea of where on the sides of the tunnel the fracture is located; note that fractures cannot be moved in directions perpendicular to the tunnel centre line, other than between equivalent points on different tunnels. Thus, in principle it is possible for distributions to be non-stationary in these directions. For example, it is possible, without breaking the assumption of stationarity, to use a depth dependence that is significant across the *height* of the tunnel, provided that it is not also significant along the *length* of the tunnel (e.g. because the tunnel is horizontal).

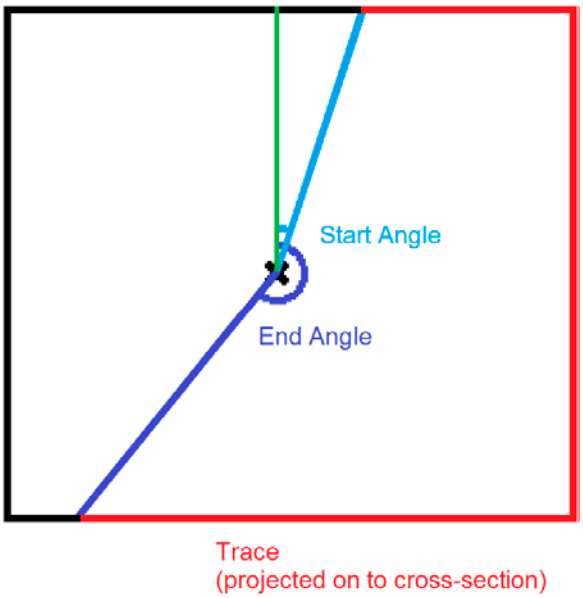

Figure A-1. *Diagram of start and end angle, and trace length (the sum of the lengths of the red trace segments) of a trace in a tunnel.*

The apparent angle is a dip angle measured between 0 and 180 degrees, allowing the dip direction also to be considered. If the fracture dips in the direction of the tunnel axis, the apparent angle is measured between 0° to 90°. If it dips in the opposite direction, the apparent angle is measured between 90° and 180° (as $\alpha = 180 - \delta$, where α is the apparent angle and δ the dip angle). If the fracture dips at a right angle to the tunnel axis, an angle from 0° to 90° is used where the fracture dips to the right, and an angle from 90° to 180° is used where the fracture dips to the left, according to an observer looking along the tunnel axis. This is illustrated in Figure A-2.

While this creates a continuous function from 0° to 180° it is also desirable to consider fractures at apparent angle 0° and 180° to be equivalent. As such, wherever apparent angles are compared, the comparison is made such that the difference between e.g. 5° and 175° is taken to be 10 degrees (a relatively small difference) and not 170 degrees (a relatively large difference). The maximum difference between two apparent angles is thus 90 degrees.

The remaining quantities considered are:

- The flow through the trace.
- The domain on which the fracture is located.

The flow may be categorised in one of three different ways:

- As an index that is zero or one depending on whether the facture is connected to the wider fracture network or not.
- Using a qualitative scale that can be translated into a series of flow categories (0, 1, 2, 3, 4...). This may originate as categories such as “dry”, “moist”, “dripping” and “flowing”.
- Using a quantitative value of flow through the trace, under conditions that are known to the user (and can be modelled using ConnectFlow). This may include, for example, injection tests or inflow measurements.

The domain is a number used to connect areas over which stationarity is assumed. This allows the user to restrict the selection of fractures to only those in the library that arise from an appropriate part of the distribution. In addition to allowing multiple fracture domains on the same tunnel, this allows the user to include tunnels that may not be parallel, or the same size or shape, and also allows the user to mix tunnels and pilot holes in the same model.

The precise domain boundary depends on the discretisation of the surfaces in the model. Each conditioned surface panel must be in exactly one domain, which can be difficult to achieve in practice. ConnectFlow thus includes a concept “domain priority” which determines which domain is to be used if a surface panel is defined on more than one domain.

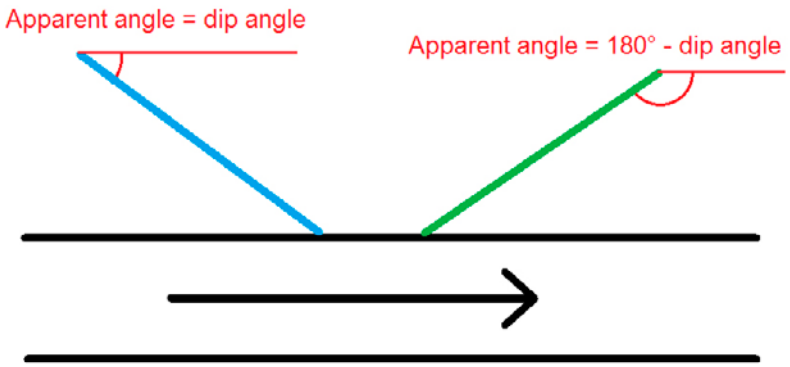

Figure A-2. *Illustration of the apparent angle as used in conditioning. The key point in this diagram is that the apparent angle is greater than 90° for fractures that dip in the opposite direction to the tunnel axis (the green fracture) and less than 90° for fractures that dip in the direction of the tunnel axis (the blue fracture). Apparent angle is a continuous measure from 0° to 180°.*

A2 Pilot holes

For a pilot hole, several of the quantities for tunnels are trivial or irrelevant, as a pilot hole intersection is effectively a point. The relevant quantities are:

- The position of the fracture intersection.
- The strike angle of the corresponding fracture.
- The dip angle of the corresponding fracture.

As with tunnels, position along the pilot hole is not needed for matching because stationarity allows any appropriate fracture to be used for any intersection within a given domain. However, unlike in the case of tunnels, it is assumed that any fracture that intersects a pilot hole will intersect the entire pilot hole and thus that the strike and dip angles can be measured.

Matching based on flow and domain is identical to that used in tunnels, however it is important to note that the definition of domains relies on the discretisation of the pilot holes; if a single pilot hole is to have more than one domain, the user must treat it as multiple connected pilot holes.

Appendix B

Fracture Selection

This appendix defines the closeness-of-fit measure and the weighted random selection.

B1 Closeness-of-fit

The closeness-of-fit is a measure that describes quantitatively how well a library intersection matches an observed intersection. Small values correspond to a good match. In selecting a fracture from the candidate list, the closeness-of-fit is calculated for each candidate and one candidate is selected randomly from the list with a probability weighted by the closeness-of-fit.

There is a wide range of possible closeness-of-fit measures and weightings that could be used. Ten measures have been identified to be used as components of closeness-of-fit in this work:

M_1 The square root of the sum of the squares of the distances between the end points of the observed intersection and the end points of the library intersection. [m]

M_2 The modulus of the difference in the lengths of the observed intersection and the library intersection. [m]

M_3 The sum of the difference in azimuthal angles of the start and end points of the observed intersection and the library intersection. [°]

M_4 The modulus of the difference in apparent angle for the observed intersection and the library intersection. [°]

$M_5 = \frac{M_1}{l}$ where l is the length of the observed intersection. [-]

$M_6 = \frac{M_2}{l_{min}}$ where l_{min} is the length of the shorter intersection. [-]

M_7 The logarithm of the difference between the flows into the engineered opening through the observed intersection and the library intersection. (The logarithm is taken because flows typically range over several orders of magnitude.) [log[m^3/s]]

M_8 The angle between the observed fracture and the library fracture (dihedral angle or angle between the lines of the intersections, see below). [°]

M_9 The distance along the engineered opening between the start point of the library intersection and the start point of the observed intersection plus the distance along the engineered opening between the end point of the library intersection and the end point of the observed intersection. [m]

M_{10} The distance between the observed intersection and library intersection (see below). [m]

These measure aspects of the match between the two intersections that might be of interest. In pilot holes modelled as scan lines, only the last four measures are meaningful as the relevant intersection has no length and apparent angle is entirely redundant to strike and dip angle. The first six measures are thus excluded from the calculation of closeness-of-fit for these intersections.

In the case of measure 8 (the angle between the observed fracture and the library fracture), ideally the dihedral angle would be used. However, in practice it is anticipated that the normal vector of a fracture associated with an observed intersection that intersects a single planar surface may not be known and thus it will be impossible to calculate a dihedral angle. In this case, the absolute angle between the observed intersection and the library intersection is used instead. Information about the fracture dip is still provided through the apparent angle.

In the case of measure 10 (the distance between the observed intersection and the library intersection), the precise calculation varies depending on how it is being used. The calculation of closeness-of-fit finds fractures for each observed intersection in turn, even if an observed fracture creates multiple intersections. This means that, any time that a potential candidate fracture is found that will create multiple intersections, it will have been found because of a match between only one of those

observed intersections and an intersection for the potential candidate fracture. The other intersections for the potential candidate will then have to be matched to other observed intersections (either on the same engineered opening or a different engineered opening). Measure 10 of closeness-of-fit distinguishes these two situations as follows:

1. When matching the first intersection, the distance used is the distance between the physical midpoints of the observed intersection and the candidate intersection. Though this distance is measured after the library fracture is moved into position, these will still not generally be in the same place, as library fractures can only be moved along the engineered opening. This is illustrated at point 1 in Figure B-1.
2. When matching the other intersections made by the candidate fracture (if any), the distance measured is based on where the candidate and observed fractures will (if infinitely extended) intersect the centre line of the engineered opening associated with the observed intersection, see point 2 in Figure B-1.

To combine the measures, a general measure for intersections mapped on surfaces is defined in Equation B-1.

$$M = \left(\sum_{i=1}^{10} (a_i M_i)^{\alpha} \right)^{1/\alpha} \quad \text{(B-1)}$$

where a_i and α are parameters. For intersections mapped on pilot holes modelled as scan lines, the general measure excludes the first six measures as in Equation B-2.

$$M = \left(\sum_{i=7}^{10} (a_i M_i)^{\alpha} \right)^{1/\alpha} \quad \text{(B-2)}$$

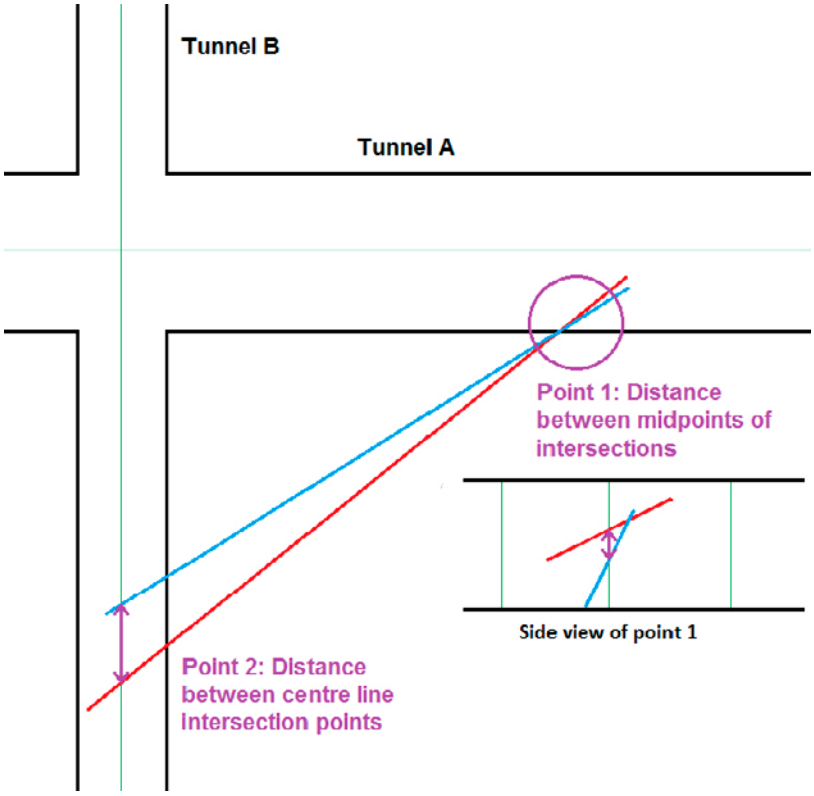

Figure B-1. *Distances used in calculation of Measure 10 of closeness-of-fit: at point 1, the measure used is the distance between the midpoints of the two intersections (i.e. based on the lateral distance between them, see the side view). At point 2, the measure used is the distance between the points where the two fractures intersect (or would intersect if extended) the centre line of the engineered opening.*

The default values of the parameters are:

$\alpha = 2$

$a_1 = 0$

$a_2 = 0$

$a_3 = 1/30$

$a_4 = 1/15$

$a_5 = 0$

$a_6 = 2$

$a_7 = 4$

$a_8 = 1/15$

$a_9 = 0$

$a_{10} = 1$

The non-zero values were chosen such that a value greater than one for a particular measure indicates a poor match. For example, a value of one corresponds to:

- The intersection lengths differing by a factor of 1.5.
- The angle between the intersections being 15°.
- The difference in apparent angle being 15°.
- The sum of the differences in start and end angles being 30°.
- The flows from the two intersections differing by a quarter of an order of magnitude (approximately a factor of 1.77), or
- the centres of the two intersections being 1 metre apart.

If more than one measure gives a non-zero answer (which is likely given that the measures are not necessarily independent), then each will contribute to the closeness-of-fit. For example, if intersection lengths differ by a factor 1.4 and the angle between fractures is 9°, and all other values are zero, then the total closeness-of-fit will be one. It is possible to get a reasonable match with a total closeness-of-fit greater than one, although an individual measure greater than one may indicate a poor match.

Zero coefficients are used to exclude measures from consideration, but they are also excluded if they are not relevant; for example measure 7 (flow) is not included if flow conditioning is not used. Because measures 1–6 are not relevant for fractures intersecting pilot holes modelled as scan lines, the total closeness-of-fit calculated for intersections on those pilot holes tends to be significantly smaller than that calculated for intersections on surfaces. Consequently the two measures are not strictly comparable, though this is unlikely to be significant in practice.

Where a fracture intersects the engineered openings in more than one place (whether it is on different panels of the same opening or on different openings), the largest single value of closeness-of-fit between individual intersections is taken as the closeness-of-fit for the entire fracture. Further issues related to conditioning with multiple intersections are discussed in Section 2.5.

It is worth noting that five of the six key quantities used for binning (discussed in Section 2.4) are included in the closeness-of-fit – the sixth being domain, which is considered binary (a fracture not in the correct domain should never be chosen). This means that in general, the closest matches to the observed intersection are most likely to be found in the first bin or bins searched early in the calculation.

In ConnectFlow, the mean and standard deviation of the closeness-of-fit for the fractures actually chosen to match the observed intersections are calculated and reported for each conditioned realisation. These provide an overall measure of how successful conditioning has been.

B2 Weighted random selection

Once the closeness-of-fit for each fracture in the initial candidate list has been calculated, the list is screened to exclude fractures deemed not close enough. The user may choose the limit at which a fracture is considered inappropriate, and limits may be applied both for the closeness-of-fit overall and for any or all individual measures. Note that the library search described above counts only acceptable matches: if a search finds fewer than ten suitable matches, it will continue until more suitable matches are found or until the library is exhausted.

As indicated above, the intersection (and associated fracture) that is taken to match an observed intersection is determined by weighted random selection from the final (screened) candidate list. The weight is taken to be a function of the closeness-of-fit measure, thus it takes into account how well the intersection matches the observed intersection. The weight function used is given in Equation B-3.

$$W = \frac{1}{N_R\left(M^2 + \frac{M_{min}}{10} + \varepsilon\right)} \tag{B-3}$$

Where:

N_R is the number of library realisations that could have generated this fracture (see below).

M is the closeness-of-fit calculated in Equation B-1 or Equation B-2.

M_{min} is the minimum actual value of M for intersections in the relevant bin(s).

ε is 10^{-8}, chosen as a very small number.

The form was chosen for the following reasons. For large values of M, the weight is inversely proportional to the square of M. This means that a fracture with a large value of M (i.e. a poor match to the observed intersection) is given a very low weight. M_{min} effectively defines the scale on which the closeness-of-fit measure is small i.e. the fit to the observed intersection is good. The ε generally has negligible effect. It is included to prevent numerical error in the case in which the library actually includes an exact match to the fracture that led to the observed intersection (hence $M = M_{min} = 0$). While this case is extremely unlikely to occur in a real scenario it is a useful test of the conditioning algorithm to check that if the library includes the realisation that produced a set of simulated observed intersections, then the algorithm reproduces the simulated intersections.

The term N_R arises from the possibility of generating more than one library for the model (as discussed in Subsection 2.3.1). In order to avoid biasing the results in favour of those fracture sizes for which more realisations have been generated in the library, the calculated weight must be divided by the total number of realisations that could have created the fracture in question in any library. If the user has chosen to only use a single library, or libraries with a uniform size range, then this term is redundant (as it will be constant for all fractures).

Appendix C

ConnectFlow input

There are two parts to the input to ConnectFlow relating to conditioning: that relating to library generation and that relating to conditioning using a library.

Note that this section only provides an overview of the input options to ConnectFlow. The ConnectFlow documentation provides more detail on alternative inputs. It is also possible to create conditioning datasets using the ConnectFlow graphical user interface, which may be more convenient in practice.

C1 Library generation

The ConnectFlow input for library generation depends on the type of library required and whether the user wants to employ any of the refinements that make conditioning using a library easier.

All library files are written using the new command `>> WRITE FRACTURE TRACE DATA` which is available under both `>> DATA INPUT` and `>> OUTPUT DATA`. This command, when invoked, will write out the trace data from the intersections between fractures and the selected surfaces to a file specified by the user, in the library file format. As it is expected that many realisations will be included in the same file, the command will automatically concatenate new traces on to the end of an existing file without rewriting the header line for the file (see Appendix D1 for an explanation of the file format).

If this command is selected under `>> DATA INPUT`, it must be placed after the fracture generation section.

The main keyword under this command is `WRITE FRACTURE TRACE DATA TO UNIT`, which provides the unit number that the library file should be saved on. The unit number is mapped to a file specified by the user. The pilot hole or tunnel surfaces are selected using the same structure of commands and keywords as used when specifying boundary conditions. Tunnel definitions are provided using a `TUNNEL DIMENSIONS` table.

The normal ConnectFlow model file is only needed if `>> WRITE FRACTURE TRACE DATA` is called from `>> OUTPUT DATA`. If it is called from `>> DATA INPUT`, the data is taken directly from the generated model without the need for an intermediate model file. For library generation it is likely that the model file for thousands of realisations would be very large, so it is desirable not to generate it.

To assist in avoiding creating a model file, a new keyword `DO NOT WRITE MODEL FILE` can be used in place of `WRITE MODEL FILE TO UNIT`. This can be invoked from the beginning of any phase of the job and will suppress all model file generation from that phase even if a model file is later explicitly requested or required for later commands.

C2 Library generation with flow

To generate a library with flow information, the user must use the library-writing command under `>> OUTPUT DATA` as calculated flow information is required. This may simply involve adding a calculation phase to an existing run in the normal way. However, this is still likely to produce very large model files and the keyword `DO NOT WRITE MODEL FILE` cannot be used (as a model file is required to pass the model information to the calculation).

To address this, an outer layer of looping has been added in ConnectFlow that will allow the run to be carried out in the order Model, Solve, Output, Model, Solve, Output instead of in the order Model, Model, Solve, Solve, Output, Output. The advantage of this is that while the model file will still be generated, it will be overwritten with each successive realisation, meaning that it is unlikely to be large enough to cause a problem unless the model itself is extremely large.

The keywords controlling this looping are:

- NUMBER OF MODELS (the number of runs to be created).
- RANDOM SEED MULTIPLIER (the number to multiply the model index by to generate a random number seed).

These keywords are placed at a very high level, under the command >> NAPSAC, and if they are invoked then the number of models for each phase must be set to one. The phases cannot be split into separate ConnectFlow data sets: the model, solve and output phases must all be present for the flow calculation to succeed and results to be written to the library file.

The random number seeds actually used will necessarily be exact multiples of the specified multiplier. In cases where the user simulates parallelisation of library generation by creating multiple library files for the same conditioning model (particularly where the same range of fracture sizes are used), it is important to ensure that different sets of random number seeds are used and that no realisation will be included twice. For example if the multiplier in library file A is set to 2, and the multiplier in file B is set to 4, the first realisation in file B will be identical to the second realisation in file A (both will have a random number seed of 4), thus defeating the purpose of creating a large number of realisations.

As well as the outer looping, in order to pass the individual flow values from the calculation phase to the output phase, it is necessary to generate various small internal files that contain this information. As these files are only used for conditioning, it is undesirable to force their creation in every case, which means that there must be some explicit indication that the intention is to create a conditioning library. For this reason, the command >> SAVE FRACTURE FLOWS FILES has been added under command >> STEADY STATE.

C3 Conditioning fractures

All conditioning takes place in the >> DATA INPUT phase, but information regarding the accuracy of flow conditioning is not printed unless a flow calculation is carried out.

To condition fractures, there is a new command under >> GENERATE FRACTURE SYSTEM, namely >> CONDITION ON FRACTURE TRACE DATA. There are a number of keywords and tables under >> CONDITION ON FRACTURE TRACE DATA.

To read the observed data:

- READ FRACTURE TRACE DATA FROM UNIT

To read the library, either:

- READ FRACTURE TRACE LIBRARY FROM UNIT, or
- WEIGHTED LIBRARIES table

The latter option allows more than one file to be read in, and for each file to have a different range of fracture sizes and a different number of realisations.

For information about tunnels, the most flexible option is:

- TUNNEL DIMENSIONS table

This allows domain, height, start and end point to be set for each tunnel. It is only used for tunnels, rather than boreholes.

For details of binning:

- NUMBER OF LENGTH BINS
- LENGTH BIN SIZE
- NUMBER OF ANGLE BINS
- NUMBER OF FLOW BINS

- UPPER FLOW LIMIT
- LOWER FLOW LIMIT
- FLOW LIMITS IN CUBIC METRES PER SECOND
- FLOW LIMITS IN LITRES PER MINUTE

The number of apparent angle bins (for tunnels) will be half the number of angle bins and the number of dip angle bins (for pilot holes) will be a quarter of the number of angle bins specified; this means that the size of each angle bin will be the same. If the number of angle bins specified does not divide evenly as required, then the bin size is preserved and the last bin will just be smaller.

The UPPER FLOW LIMIT and LOWER FLOW LIMIT specify the limits of the range for flow bins. The LOWER FLOW LIMIT is considered the detection limit: anything with less flow than this will be considered non-flowing. Any flow higher than the UPPER FLOW LIMIT will be placed in the bin with highest flows.

For customising the closeness-of-fit:

- END POINTS CLOSENESS COEFFICIENT (measure 1)
- LENGTH CLOSENESS COEFFICIENT (measure 2)
- WALL ANGLE CLOSENESS COEFFICIENT (measure 3)
- APPARENT ANGLE CLOSENESS COEFFICIENT (measure 4)
- RELATIVE END POINTS CLOSENESS COEFFICIENT (measure 5)
- RELATIVE LENGTH CLOSENESS COEFFICIENT (measure 6)
- FLOW CLOSENESS COEFFICIENT (measure 7)
- FRACTURE ANGLE CLOSENESS COEFFICIENT (measure 8)
- TUNNEL POSITION CLOSENESS COEFFICIENT (measure 9)
- TRACE DISTANCE CLOSENESS COEFFICIENT (measure 10)
- CLOSENESS EXPONENT

The first ten of these keywords define the coefficients a_i in Equations B-1 and B-2 (see Appendix B), where i is the measure index specified. The CLOSENESS EXPONENT is the value α in the same equations.

To set user-defined limits:

- ACCEPTABLE CLOSENESS OF FIT (final measure)
- END POINTS CLOSENESS MAXIMUM (measure 1)
- LENGTH CLOSENESS MAXIMUM (measure 2)
- WALL ANGLE CLOSENESS MAXIMUM (measure 3)
- APPARENT ANGLE CLOSENESS MAXIMUM (measure 4)
- RELATIVE END POINTS CLOSENESS MAXIMUM (measure 5)
- RELATIVE LENGTH CLOSENESS MAXIMUM (measure 6)
- FLOW CLOSENESS MAXIMUM (measure 7)
- FRACTURE ANGLE CLOSENESS MAXIMUM (measure 8)
- TUNNEL POSITION CLOSENESS MAXIMUM (measure 9)
- TRACE DISTANCE CLOSENESS MAXIMUM (measure 10)

The ACCEPTABLE CLOSENESS OF FIT is the maximum overall closeness-of-fit (i.e. the maximum value of M in Equations B-1 and B-2, see Appendix B), allowed before a fracture is rejected outright. The remaining ten keywords define the maximum allowed values of M_i in Equations B-1

and B-2, where *i* is the measure index specified. If any of these maximums are exceeded, the fracture is rejected. Values for the maximums for individual measures can be inferred from the overall maximum and coefficients as in Equation C-1:

$$M_{i_{max}} = \frac{M_{max}}{a_i} \quad \text{(C-1)}$$

Where:

$M_{i_{max}}$ is the implied maximum allowed measure for measure *i*,

M_{max} is the `ACCEPTABLE CLOSENESS OF FIT` (default 5),

a_i is the coefficient associated with measure *i*.

It is thus not necessary to define maximums for individual measures unless they are lower than the implied maximums.

The pilot holes and tunnel surfaces must be selected using the same structure of commands and keywords as used when specifying boundary conditions. However, for a multi-domain model, each domain must be specified as well. To specify these, the command `>> SET DOMAIN` and keyword `DOMAIN NUMBER` are used above the surface selection commands.

Most of the keywords are optional and have default values. The only strict requirements are:

- Observed data must be provided.
- At least one library must be provided.
- Conditioning surfaces and pilot holes must be defined.
- Tunnel end points must be explicitly defined (i.e. it is insufficient to only give the surfaces).
- Domain numbering must be consistent (e.g. non-parallel tunnels may not have the same domain number).

Appendix D

File formats

The format of the library and observed data files are very similar, allowing the same file reader in ConnectFlow to be used for both files. However, there are some differences in details between the formats, which reflect the different information required for the observed trace data and for the library of fractures.

There are also slightly different formats depending on whether the file contains only pilot hole intersections, only surface traces, or both. This reflects the fact that some of the parameters relevant to conditioning for a 3D tunnel are not applicable for a 1D pilot hole.

For both kinds of file, the standard file name extension is *.trc*. ConnectFlow allows 3D visualisation of the traces in each file.

D1 Library file

The library file is an ASCII file that has up to 35 columns of numerical data. The first row of the file contains labels identifying the columns, and the following lines give details for each fracture trace in turn, with one line per trace. Columns may be included in any order, but each trace must include something for every column. Whereas some columns contain information that is indispensable, others contain information that is desirable, but can be omitted if appropriate, or that provide alternative means of calculating a given quantity. Still other columns contain information that is not going to be used at all but that may provide useful information for the user.

Files of this format are written by ConnectFlow where library generation is specified.

The allowed column names are provided in Table D-1.

Table D-1. Column names allowed for use in ConnectFlow trace files, noting whether they are required, optional, unused or not allowed.

	Surface trace files	Pilot hole intersection files	Mixed-concept files
START_X	Required	Not allowed	Required
START_Y	Required	Not allowed	Required
START_Z	Required	Not allowed	Required
END_X	Required	Not allowed	Required
END_Y	Required	Not allowed	Required
END_Z	Required	Not allowed	Required
INTERSECTION_X	Not allowed	Required	Not allowed
INTERSECTION_Y	Not allowed	Required	Not allowed
INTERSECTION_Z	Not allowed	Required	Not allowed
SET	Optional	Optional	Optional
TOP_STRUCTURE TRACE_NUMBER TRACE_INDEX	Required	Required	Required
MACROFRACTURE	Unused	Unused	Unused
MF_CENTRE_X	Required	Required	Required
MF_CENTRE_Y	Required	Required	Required
MF_CENTRE_Z	Required	Required	Required
STRIKE_X	May be required	May be required	May be required
STRIKE_Y	May be required	May be required	May be required
STRIKE_Z	May be required	May be required	May be required
DIP_X	May be required	May be required	May be required
DIP_Y	May be required	May be required	May be required

	Surface trace files	Pilot hole intersection files	Mixed-concept files
DIP_Z	May be required	May be required	May be required
STRIKE_ANGLE	May be required	May be required	May be required
DIP_ANGLE	May be required	May be required	May be required
ORIENTATION	May be required	May be required	May be required
STRIKE_LENGTH	May be required	May be required	May be required
DIP_LENGTH	May be required	May be required	May be required
HYDRAULIC_AP	May be required	May be required	May be required
TRANSMISSIVITY	May be required	May be required	May be required
TRANSPORT_AP	Optional	Optional	Optional
TUN_TRACE_LEN	Optional	Optional	Optional
START_ANGLE	Optional	Optional	Optional
END_ANGLE	Optional	Optional	Optional
APPARENT_ANGLE	Optional	Optional	Optional
CONNECTED	Optional	Optional	Optional
FLOW_CATEGORY	Optional	Optional	Optional
FLOW	Optional	Optional	Optional
DOMAIN	May be required	May be required	May be required
CONCEPT	Not allowed	Not allowed	Required

A file of a given type must contain those columns marked “required” in Table D-1, and may contain any of the columns marked “optional” or “unused” (the former may be used for calculation, the latter are ignored). They must not contain any of the columns marked “not allowed”.

Those columns marked “may be required” are needed in some circumstances:

- The file must define the fracture’s strike and dip vectors in some form, using either of the combinations:
  - STRIKE_X, STRIKE_Y, STRIKE_Z, DIP_X, DIP_Y and DIP_Z
  - STRIKE_ANGLE, DIP_ANGLE, ORIENTATION, STRIKE_LENGTH and DIP_LENGTH
- The file must contain either HYDRAULIC_AP or TRANSMISSIVITY
- The file must contain DOMAIN unless the default domains are to be used.
  - In single-concept files, the default is that only one domain will be used.
  - In mixed-concept files, the default is that all tunnel surfaces are in domain 1 and all pilot holes in domain 2.

Other considerations:

- For pilot holes in a mixed-concept file, the intersection coordinates specified using END_X, END_Y and END_Z are the same as those specified using START_X, START_Y and START_Z. In files only containing pilot holes intersections the columns INTERSECTION_X, INTERSECTION_Y and INTERSECTION_Z are used instead.
- Column titles TOP_STRUCTURE, TRACE_NUMBER and TRACE_INDEX are equivalent. Exactly one of them must be included in every file.

The data to be included in the columns is listed in Table D-2.

Table D-2. Descriptions of data to be included in each column in a ConnectFlow trace file.

START_X	X coordinate of the first end of the trace
START_Y	Y coordinate of the first end of the trace
START_Z	Z coordinate of the first end of the trace
END_X	X coordinate of the second end of the trace
END_Y	Y coordinate of the second end of the trace
END_Z	Z coordinate of the second end of the trace
INTERSECTION_X	X coordinate of the fracture intersection with the pilot hole
INTERSECTION_X	Y coordinate of the fracture intersection with the pilot hole
INTERSECTION_Z	Z coordinate of the fracture intersection with the pilot hole
SET	The number of the set that generated the trace
TOP_STRUCTURE TRACE_NUMBER TRACE_INDEX	The reference number of the fracture that generated the trace
MACROFRACTURE	The fracture index of the trace
MF_CENTRE_X	X coordinate of the centre of the fracture that generated the trace
MF_CENTRE_Y	Y coordinate of the centre of the fracture that generated the trace
MF_CENTRE_Z	Z coordinate of the centre of the fracture that generated the trace
STRIKE_X	X component of the half vector along the side of the fracture that generated the trace in the strike direction.
STRIKE_Y	Y component of the half vector along the side of the fracture that generated the trace in the strike direction.
STRIKE_Z	Z component of the half vector along the side of the fracture that generated the trace in the strike direction.
DIP_X	X component of the half vector along the side of the fracture that generated the trace in the dip direction.
DIP_Y	Y component of the half vector along the side of the fracture that generated the trace in the dip direction.
DIP_Z	Z component of the half vector along the side of the fracture that generated the trace in the dip direction.
STRIKE_ANGLE	The strike angle of the fracture that generated the trace
DIP_ANGLE	The dip angle of the fracture that generated the trace
ORIENTATION	The orientation angle of the fracture that generated the trace
STRIKE_LENGTH	The length of the fracture that generated the trace in the direction of strike
DIP_LENGTH	The length of the fracture that generated the trace in the direction of dip
HYDRAULIC_AP	The hydraulic aperture of the fracture that generated the trace
TRANSMISSIVITY	The transmissivity of the trace
TRANSPORT_AP	The transport aperture of the trace
TUN_TRACE_LEN	The length of the trace
START_ANGLE	The angle of the start point of the trace
END_ANGLE	The angle of the end point of the trace
APPARENT_ANGLE	The apparent angle of the fracture that generated the trace
CONNECTED	Whether or not the trace is connected to the wider fracture network (0 if not, 1 if so)
FLOW_CATEGORY	The flow category of the trace (discretised integer value)
FLOW	The flow between the fracture generating the trace and the engineered opening
DOMAIN	The domain number of the fracture
CONCEPT	“Tunnel” for a surface trace, “PilotHole” (with no space) or “Borehole” for a pilot hole intersection

It does not matter which end of the trace is considered the start and which is considered the end, and it is anticipated that a new trace will be added to the file at any point where a single physical trace meets a sharp corner in the tunnel wall. These traces must have the same number in the "TOP_STRUCTURE" column, and will be placed next to one another in the file. Fracture definitions will be the same for each.

Flow properties of fractures can either be specified in terms of the flow through the fracture itself ("FLOW"), a category corresponding to a range of flow ("FLOW_CATEGORY"), or in terms of whether there is a connected network of fractures linking the fracture to a surface at a distance from the tunnel ("CONNECTED"). ConnectFlow's library generation will automatically create a library with "FLOW" included if the flow through the model has been calculated, and "CONNECTED" if not.

The five column headings TUN_TRACE_LEN, START_ANGLE, END_ANGLE, APPARENT_ANGLE, FLOW_CATEGORY and DOMAIN correspond to the parameters that are binned, and are used unmodified if present.

D2 Observed data file

The data required for the observed data file is a subset of the data required for the library file, allowing single-realisation library files to be used to simulate observed data if desired. However, whereas the library file must include certain properties required for defining the fracture (such as the fracture's centre location); the data file needs only to contain the data actually required to identify the trace.

This means that an observed data file can contain significantly less information than the library and remain valid.

The following columns, describing properties of the fracture that may not be evident when identifying a trace, are required or optional when writing the library, but are not used when processing the observed data (and can thus be left out):

- SET
- MF_CENTRE_X
- MF_CENTRE_Y
- MF_CENTRE_Z
- STRIKE_LENGTH
- DIP_LENGTH
- HYDRAULIC_AP
- TRANSPORT_AP
- TRANSMISSIVITY

If conditioning using apparent angle on tunnel surfaces, or using strike and dip angle on pilot hole intersections, the user must include in the observed data file either:

- All of STRIKE_X, STRIKE_Y, STRIKE_Z, DIP_X, DIP_Y and DIP_Z (in this case the magnitude of the vectors is not used and thus arbitrary);
- All of STRIKE_ANGLE, DIP_ANGLE and ORIENTATION, or
- APPARENT_ANGLE (for apparent angle on tunnel surfaces only; this data will not be used for pilot holes)

If this condition is not met, the algorithm will continue without using the parameters that it cannot calculate.

It may be that this information is available for some traces and not others. In observed data files (but not library files), columns STRIKE_X, STRIKE_Y, STRIKE_Z, DIP_X, DIP_Y and DIP_Z may contain a non-numeric value (e.g. "null"), in which case they will be ignored for the trace in question.

If conditioning on flow or connectivity, the user must include either:

- CONNECTIVITY
- FLOW_CATEGORY
- FLOW

If this condition is not met, the algorithm will ignore flow.

If conditioning in multiple domains is required, the user must include DOMAIN. If DOMAIN is not included, all fractures are assumed to be in domain 1. If using a mixed-concept file and DOMAIN is not included, surface traces are assumed to be in domain 1 and pilot hole intersections in domain 2.

Appendix E

Calibration targets

The aim of the conditioning process is to generate model realisations that honour the local conditions at a site by taking account of the locations and properties of fractures observed on tunnel walls and in boreholes within the site, and in doing so to reduce the uncertainty in predictions made using the models. The aim of the calibration targets is to confirm that conditioning honours both the observations and the wider site model, while performance measures quantify how uncertainty is reduced in predicting local structural and flow conditions by making use of underground data.

To this end, we have chosen to present three different types of calibration and performance measure target:

- Confirming that the model does what it is designed to do (calibration targets).
- Quantifying the spread of results, to ensure that the resulting fracture model is internally consistent (consistency checks).
- Demonstrating improvement in the quantities (primarily related to flow) that we are interested in (performance measures).

Two calibration targets or performance measures are proposed in each case.

E1 Confirming that the model honours the data

The measures proposed in this section are calibration targets. They look at the characteristics of the model that can be measured by the user and are intended to confirm that the model honours the data. The nature of these checks means that it should be possible to confirm them against e.g. the models used for HypoSite.

E1.1 P_{21}

It is proposed that we verify that local P_{21} is within a reasonable range of the observed value.

Where there is tunnel surface mapping, P_{21} is, in principle, a measured value from the observed data, accepting artefacts of mapping on real excavated surfaces that can be handled by, for example, projecting actual traces on to idealised tunnel surfaces. As such, P_{21} is one of the attributes that the conditioning process is actively trying to match, and thus is a useful calibration measure in ensuring that the conditioning is doing what it is intended to do.

It is theoretically possible to create an exact match to this projected P_{21}. However, it is worth considering whether an exact P_{21} match is desirable if this were to require the rotation, translation or cropping of fractures that might have biased the distribution of fracture properties. Figure E-1 shows an example of traces that are similar, but not exactly equivalent. There is also a question of whether it is most important to ensure a good match in the P_{21} of all fractures, or to put more emphasis on assuring a good match to the P_{21} of larger fractures, either full perimeter intersects or those with a trace length greater than, for example, the diameter of the tunnel.

Thus it is proposed that, total P_{21} (P_{21T}) and P_{21} of those traces that are longer than the diameter of the tunnel (P_{21D}) be calibration targets. To establish what degree of matching should be acceptable it is proposed that the P_{21} in the conditioned realisation should be within 5 % of the equivalent value in the observed realisation (an alternative could be to say the discrepancy must be within c. 20 % of the variation in P_{21} between unconditioned realisations). Figure E-2 shows an example graph of P_{21} from a number of realisations, demonstrating the differing P_{21T}; in this example, 26 of the 30 conditioned realisations pass this test.

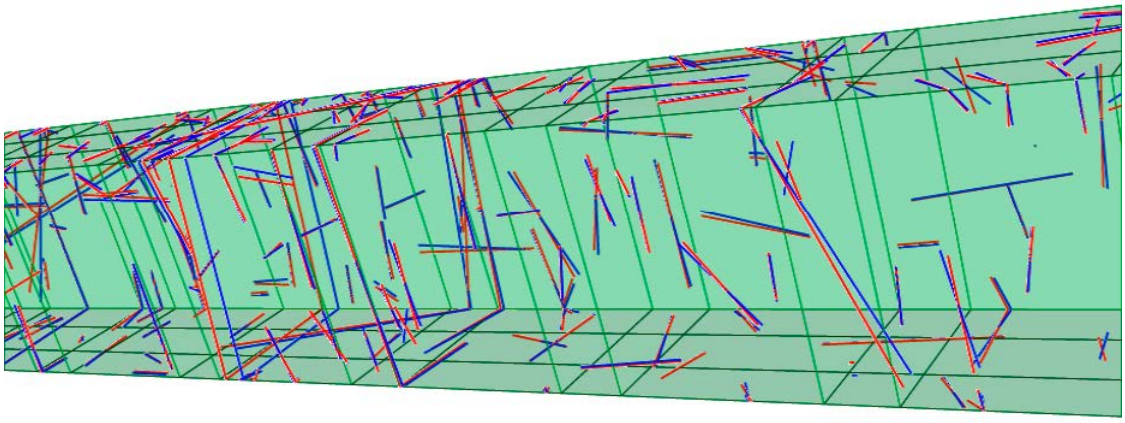

Figure E-1. *A length of tunnel containing example traces from which P_{21} could be measured. Blue lines are observed traces, red lines are conditioned traces.*

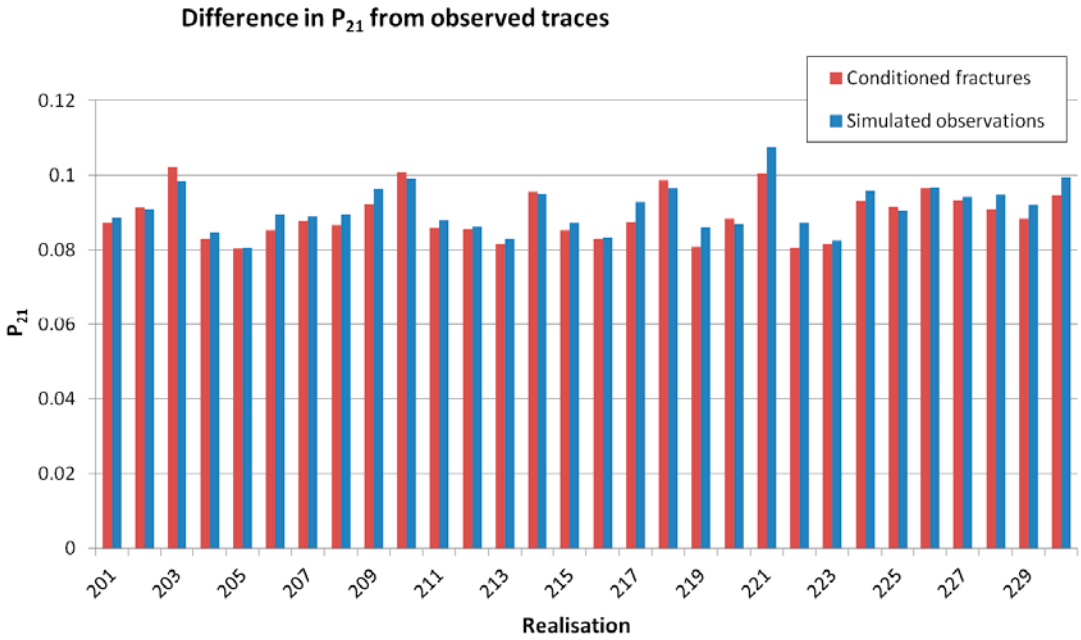

Figure E-2. *Values of P_{21} for thirty realisations, comparing conditioned fractures against simulated observations.*

The mapped areas on which P_{21} is to be measured will depend on the fracture intensity of the model in question. Smaller lengths of tunnel would be desirable in order to ensure that P_{21} is not just statistically correct across the entire tunnel but also locally correct; 10 metres is suggested as this is the approximate deposition hole spacing. However, if segments contain relatively small numbers of fractures, this may not actually provide a reliable measure. It is anticipated that segments used for measurement will not be amalgamated over several tunnels, and will be specific to individual fracture domains seen within a tunnel (as relevant to, for example, HypoSite BM-1).

E1.2 Geometry of large fractures

While for short tunnel lengths the P_{21} provides some assurance that the fractures are correct locally as well as on a statistical level, it is also useful to check that the geometries of each fracture match to ensure that the fractures generated are sufficiently similar geometrically to the fractures observed. In particular, it is desirable to reduce uncertainty in which deposition holes will see large, and possibly connected, fractures.

It is thus proposed that the distance between the centre of the observed trace and the centre of the conditioned trace be compared with the diameter of a deposition hole. This will ensure that the conditioned fracture creates approximately the correct trace to match the observed data.

It is also proposed that the fracture poles be compared (between conditioned and observed fractures), and that the angle between the poles should be less than $\tan^{-1}(\frac{d_h}{l_h})$, where d_h is the diameter of a deposition hole and l_h the length of a deposition hole. This will ensure that the conditioned fracture is approximately correctly oriented and predicted to intersect the correct deposition hole.

According to this calibration target, the proportion of conditioned large fractures (with trace length greater than the diameter of the tunnel) that exceed each limit should be small and ideally should be zero. An image illustrating example fractures is provided in Figure E-3; the fractures illustrated may allow the model to pass on other measures, but would fail on this calibration measure.

When calculating this measure for a borehole, size cannot be determined, but the important fractures can be identified as those with flow or specific capacity greater than 10^{-9} m^2/s. The same location and orientation metrics can be applied to fractures with above this specific capacity in boreholes, providing flow measurements have been made.

For the angle metric, it may be necessary to exclude cases where a fracture does not intersect multiple tunnel sides as the fracture pole cannot be easily calculated for an observed fracture in this case.

It is possible that in some cases, there will be unmatched fractures. This could occur where a modelled fracture intersects tunnels that it should not intersect, or in cases where no match is possible for an observed fracture and thus no conditioned fracture is included. In either case, both measures should be treated as failing (i.e. exceeding the limit).

In ConnectFlow, the workflow for determining the metric is relatively simple: each measure can be relatively easily calculated when a fracture is chosen and then tested against the necessary limits. In the case of unmatched observed fractures, if any exist, they are reported at the end of the calculation.

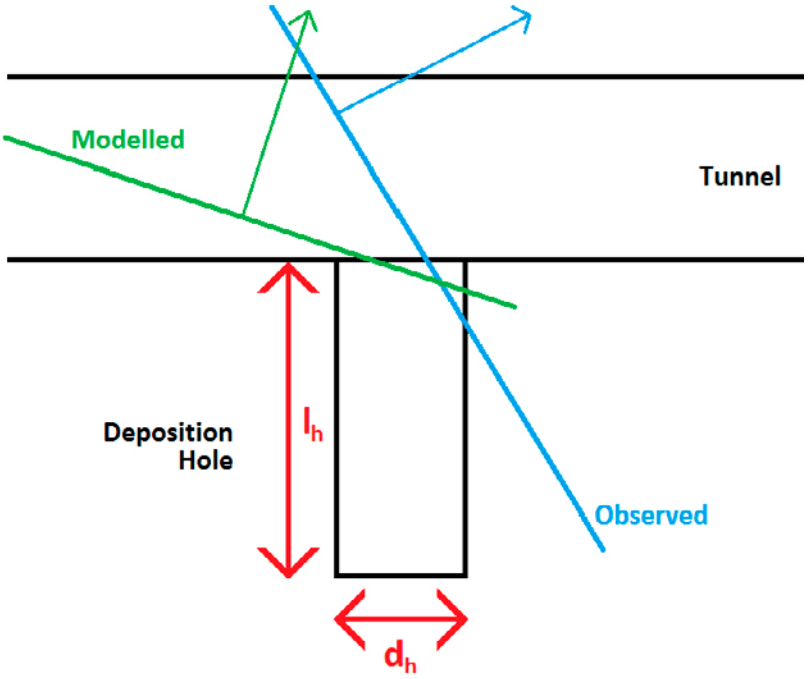

Figure E-3. *A figure demonstrating a section of tunnel with deposition hole, and a modelled and observed fracture (green and blue) with the directions of their fracture poles. This example would fail on this calibration measure.*

E2 Assuring self-consistency of the conditioning methodology

The measures proposed in this section are consistency checks. They analyse the characteristics of the fracture system that are modelled to ensure that the outcome of conditioning is consistent with the statistical recipe input by the user. It is undesirable for conditioning to create realisations that are not consistent with the statistical model that the user has determined is appropriate for the surrounding bedrock, and an incorrect statistical model may adversely affect the modelled flow to the tunnels. This requirement is only practical in the case of a synthetic benchmark test, and not real data. However, it is a necessary assurance step in developing the methodologies.

The aim is to demonstrate internal consistency for benchmark cases where the "observed" data is generated synthetically from a known recipe. Statistical characteristics of the fracture recipe used as the basis for conditioning should not be changed by the conditioning process – even if that recipe creates a model that is not itself entirely consistent with a given version of HypoSite or other external benchmarks. Because of this, these targets are best evaluated against the models as interpreted by the software that is performing the conditioning, rather than a direct comparison against HypoSite data.

E2.1 P_{32}

It is proposed that P_{32} in a volume around the mapped tunnel be verified as being within a reasonable range of the value implied by the recipe.

The reason to test P_{32} is to ensure that the conditioned model honours the statistical fracture recipe that defines the DFN model. It is important that this remains internally consistent, i.e. this recipe must be respected and not adversely affected by the model, otherwise this may have a significant impact on other results.

When considering P_{32}, it is possible to calculate the global P_{32} (for the entire model) or the local P_{32} (in the vicinity of the engineered structures). It is desirable that, in both cases, the values generated correctly honour the fracture recipe provided by the user as a basis for the model (to within a reasonable degree of statistical variance). However, in practice, it seems unlikely that the global P_{32} will give a significantly different result from that originally generated, particularly in a large model, unless there is a relatively extreme error in the conditioning process. The fractures actually changed through the conditioning process are likely to be a very small proportion of the fractures in the global model, and thus are likely to comprise a very small proportion of the P_{32}. Such a test is thus unlikely to provide a sensitive test of the conditioning process.

On the other hand, local P_{32} in the vicinity of the engineered structures is likely to be far more sensitive to changes caused by the conditioning process, and consequently more useful as a calibration target. It is also likely, in general, that most changes to the global P_{32} will result from changes in the local P_{32}. It is thus proposed that P_{32} be calculated based on a suitable volume surrounding the engineered openings to be used for conditioning, taken to be the smallest cuboid that includes all rock volume within 10 metres of any conditioned feature. A schematic example (not to scale) is given in Figure E-4.

The distance 10 metres was chosen as the approximate length and spacing of deposition holes; it is also far enough away from the conditioned features to judge whether the model is strongly affected by the conditioning, while not being so far as to render that influence irrelevant due to the high number of unaltered fractures in a large model.

In practice, it is not reasonable to expect that the P_{32} used in the DFN recipe will match exactly to the conditioned realisation, as there will be some stochastic variability. Consequently, it will be necessarily to conduct a comparison between statistical distributions based on a significant number of conditioned and unconditioned realisations.

In ConnectFlow, the comparison of P_{32} would be conducted by comparing the unconditioned distribution of P_{32} determined through library generation with several conditioned realisations (e.g. thirty) using different unconditioned realisations and observed data for each model. An example graph generated using a similar metric is presented as Figure E-5, with a set of results from unconditioned realisations also presented to demonstrate the variability of the results.

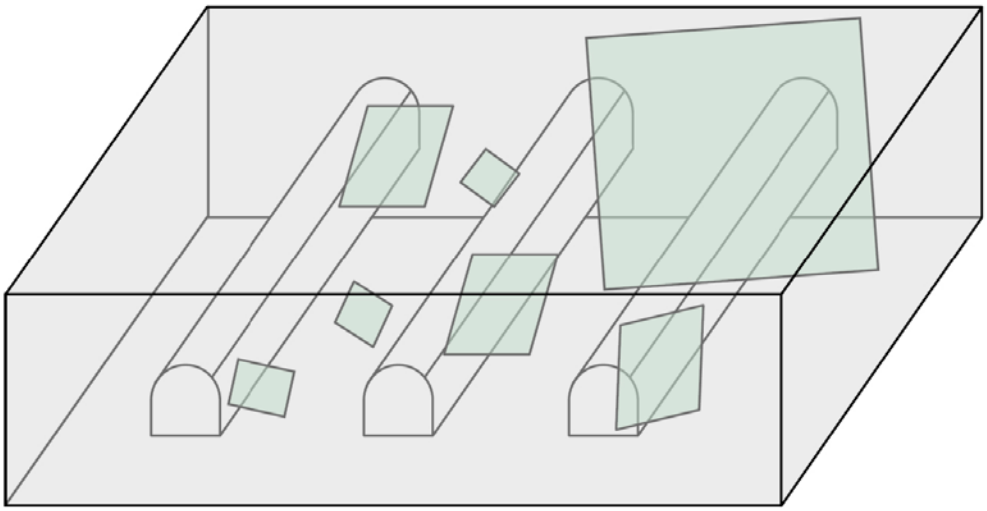

Figure E-4. *Schematic (not-to-scale) image of the box to be used for P_{32} calculations around the tunnels.*

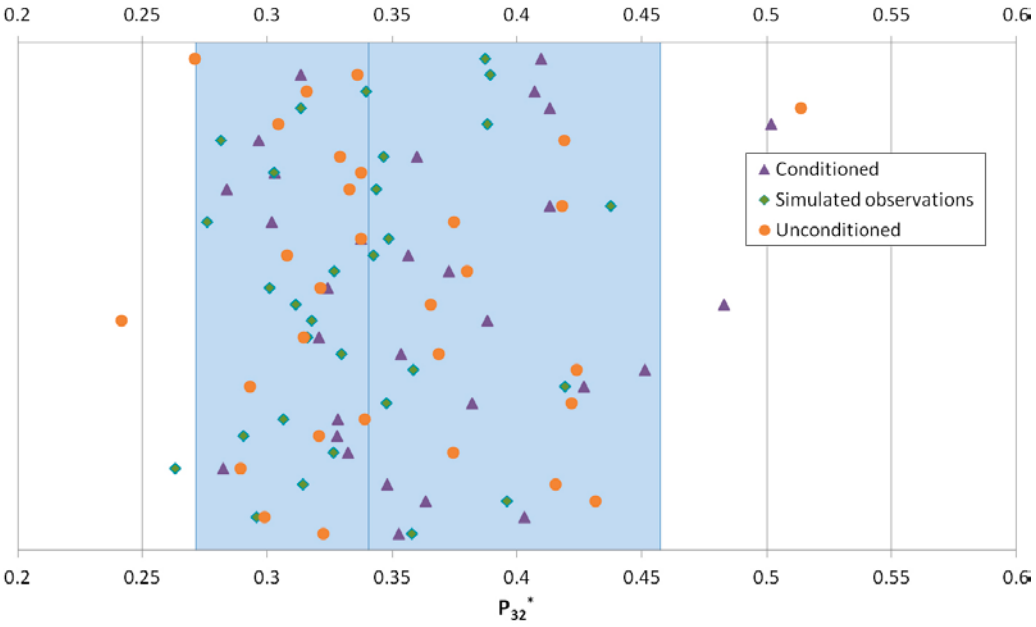

Figure E-5. *Example results from calculation of P_{32}.*

As measures of success, it is proposed that the median P_{32} of the conditioned realisations is within 5 % of the median of the unconditioned realisations, and that no more than 15 % of the P_{32} values for the conditioned realisations fall outside the 5th and 95th percentile of P_{32} values for unconditioned realisations. It is recognised that this comparison needs several tens of unconditioned realisations and conditioning on several tens of different synthetic realisations of observed data. Still, it is considered appropriate to provide assurance the methodology does not bias or corrupt the network characteristics on at least one benchmark example.

In Figure E-5, the blue band is the range between the fifth and ninety-fifth percentiles of P_{32} in a large number of unconditioned realisations; the vertical line in the middle of this range is the median value. The purple triangles are the results from conditioned realisations, the green diamonds are the results from the observed data and the orange dots the results from the unconditioned realisations prior to conditioning.

E2.2 Size distribution

It is proposed that it be checked that the distribution of the size of fractures in the vicinity of the tunnel is also not biased by the conditioning process.

As with P_{32}, the reason to test the fracture size distribution is to ensure that the conditioned model is consistent with the fracture recipe originally input by the user. This is a very important consideration because the bulk of the flow to the tunnels and deposition holes is likely to arise through a few large fractures in the tail of the size distribution. As these fractures are likely to deliver a large proportion of the flow, a systematic error in size distribution causing these fractures to be over- or underrepresented could significantly alter the modelled flow in the vicinity of the tunnels.

It is proposed that the size distribution of the fractures be determined based on the number of fractures of a given size that intersect scan lines that run parallel to each tunnel centre line. Two of these scan lines would be vertically above and below the tunnel centre line, such that they are within the rock, consistently one metre from the tunnel ceiling and floor. The other two would be horizontally displaced such that they are within the rock, consistently one metre from the tunnel side walls. For pilot holes or boreholes modelled as one-dimensional lines, the scan lines should run along the central axis of each pilot hole. The one-metre distance was chosen such that the scan lines would generally intersect fractures that also intersect the tunnel; however, as these are one-dimensional structures the predicted distribution of fractures by size is far simpler than for a three-dimensional structure. An image of potential scan lines is provided in Figure E-6.

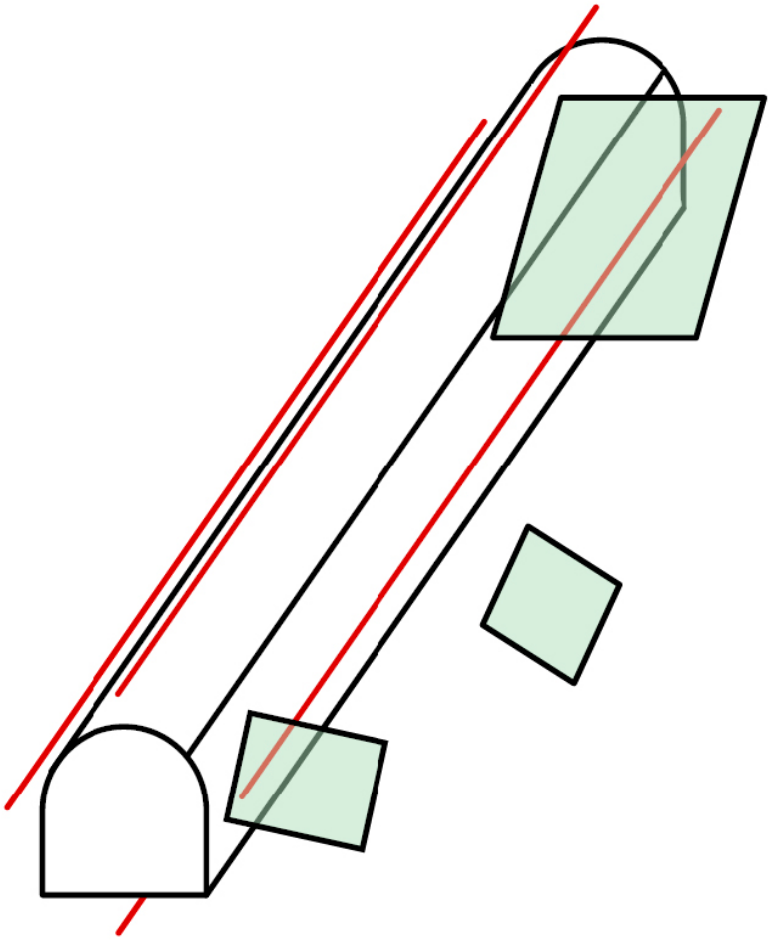

Figure E-6. *Locations of the four scan lines with respect to a single tunnel, with some fractures included.*

As with P_{32}, the size distribution of fractures in a real world situation is unknown, but the size distribution of fractures in the fracture recipe used to generate the model will be available. For a simple model with a constant fracture size distribution (i.e. without depth dependence or varying fracture domains) it will be possible to use an analytical distribution. For more complex systems, it would be preferable to generate an empirical distribution based on a significant number of realisations of the DFN. In such a distribution, care would have to be taken to ensure that there are sufficient large fractures considered for a comparison to be statistically valid.

In ConnectFlow, as with P_{32}, it is anticipated that these distributions will be generated as part of the fracture library generation. It will then be possible to visualise the size distributions based on a binning structure or as a cumulative distribution function and compare them, conditioned versus unconditioned realisations. The size distributions should be very similar.

E3 Demonstrating improvement in model performance

The measures proposed in this section are intended to demonstrate the outcome of the conditioning process in terms of model performance, i.e. they are performance measures. In this case, it is understood that the models will not necessarily perform perfectly, but it is intended that conditioning should reduce uncertainties in making model predictions of quantities important to safety assessments or acceptance criteria for deposition areas.

E3.1 Flow to an open feature

It is proposed that the performance improvement be determined by measuring the flow into the tunnels.

The flow to an open tunnel is a relatively simple test as to whether the conditioning is providing better agreement with observations or not. The flow can be relatively easily measured in the field, for an entire tunnel or for sections of a tunnel. It is desirable for conditioning to reduce uncertainty in reproducing such measurements.

It is likely that there will be some significant difference in flow, because flow depends on properties of the surrounding fracture network as well as the properties of fractures intersecting the tunnels. That is, the only fractures likely to change as a result of conditioning are fractures that intersect the tunnel, and the flow to the tunnel is likely to be highly dependent on the surrounding network of fractures, which is unlikely to be identical or similar to the unknown real situation in the rock or observed data. As a result, flow paths giving rise to the observed data are unlikely to be present in the same places as in the conditioned model, and vice versa.

Another such issue is the non-uniqueness of the conditioning solution. While it is possible to determine some aspects of a fracture intersecting a wall from its trace characteristics, those characteristics do not fully describe the fracture. For example, it is not clear from the trace alone whether the fracture is small or large. While it is possible to compensate for this to some degree (for example, by including flow as part of the conditioning calculation, or where fractures have been mapped in several tunnels) such allowance will be imperfect.

It is thus proposed that the first performance measure be based on the degree of improvement between the unconditioned and conditioned inflow into an open tunnel and to each 10-metre subsection of a tunnel compared to the observed data.

E3.2 Injection tests

In addition to inflow simply to an open tunnel, it is also desirable to further test the effectiveness of conditioning by using injection tests, whereby water is injected into the pilot hole for each deposition hole in turn. The injection tests allow injected specific capacity (outflow divided by injection head) to be measured, which is related to fracture transmissivity. The injection tests also give an indication of how fractures connect to the wider fracture network. It is thus proposed that these injection tests be used as a second performance measure.

The results of this hydraulic testing will be quantified as the difference between the simulated injected specific capacity and the observed injected specific capacity for each pilot hole or deposition hole. The success of the conditioning will be evaluated by the reduction in the difference between simulated and observed values for conditioned models relative to unconditioned models. Deposition holes and pilot holes will be assumed to be sealed from the deposition tunnels, e.g. by ignoring flows through the top metre of the pilot hole. Injected specific capacity may be evaluated at a particular transient time or at steady-state.

E3.3 Post-closure flow

In order to measure the performance of the model, it is ultimately desirable to determine the post-closure flow and transport conditions around potential deposition volumes. As with measures such as P_{32} and fracture size distribution, this is not a measure that can be practically determined using the data likely to be available at the site and that must instead be determined based on the calculations of synthetically generated "observed" data.

It is proposed that the post-closure values of *U* and *F* for deposition holes be determined for conditioned realisations and compared with the equivalent values for the synthetic observed data and for unconditioned realisations. In order to test this, it is envisaged that several tens of synthetic realisations will be required and several conditioned realisations will have to be generated for each set of synthetic observations. It is proposed that for each synthetic realisation, the *U* and *F* distribution is calculated across at least 10 conditioned realisations and compared with the *U* and *F* in the synthetic realisation. Given an acceptance criterion in *U* and *F*, each deposition hole should be screened out (due to high *U* or low *F*) in a majority of realisations if it would be screened out in the synthetic observed data, and not screened out in a majority of realisations if it would not be screened out in the synthetic observed data.

In ConnectFlow, the post-closure *U* and *F* are calculated (Joyce et al. 2010) as part of particle tracking. A particle is released in each deposition hole (e.g. as shown in Figure E-7) and allowed to move according to the calculated flow (post-closure). The particle start locations are within the deposition holes.

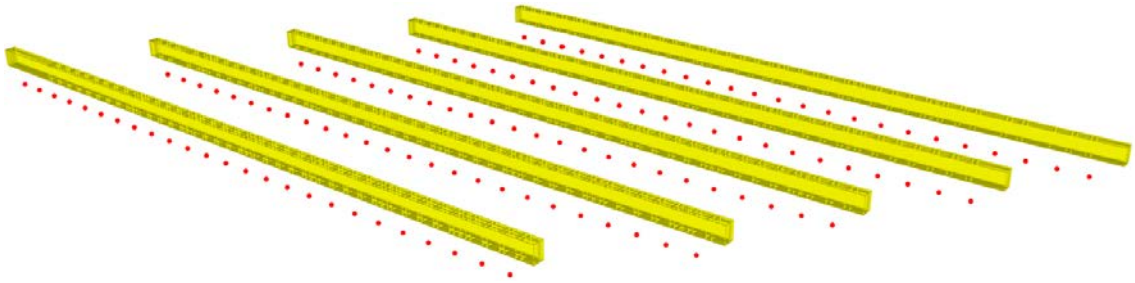

Figure E-7. *Start points (red dots) for particle tracking used to calculate U and F. The yellow structures are tunnels.*

E4 Summary table

Table E-1 is provided as a simple summary of the above proposals. For ease of reference, each proposal is given a reference number, CT for calibration targets, CC for consistency checks and PM for performance measures.

Table E-1. Summary of proposed calibration targets, consistency checks and performance measures; d_h is diameter of deposition hole, l_h is length of deposition hole.

Metric	Ref No.	Method	Scale	Success criterion
P_{21}	CT1a	Comparison between observed and modelled	Characteristic length of tunnel	– Difference < 5 %, or – Discrepancy < 20 % of the variation in P_{21} between unconditioned realisations
	CT1b	Comparison between observed and modelled	Characteristic length of tunnel; large traces only	– Difference < 5 %, or – Discrepancy < 20 % of the variation in P_{21} between unconditioned realisations
Geometric criteria	CT2a	Distance between observed and modelled traces	All tunnels; large fractures only	Number of traces with distance $>d_h$ should be small
	CT2b	Angle between fracture poles	All tunnels; large fractures only	Number of traces with angle less than $\tan^{-1}(\frac{d_h}{l_h})$ should be small
P_{32}	CC1	Difference between distributions of observed and modelled values	Cuboid immediately surrounding engineered openings	– Median within 5 % of observed data, and – <15 % of data outside 5th and 95th percentiles
Size distribution	CC2	Difference between distributions of observed and modelled values	Scan lines immediately surrounding engineered openings	Distributions should be very similar.
Flow	PM1	Comparison between observed and modelled	As for P_{21}	Reduction in difference compared with comparison between observed and unconditioned
Injection tests	PM3	Comparison between observed and modelled	Pilot holes for deposition holes	Reduction in difference compared with comparison between observed and unconditioned
U and *F*	PM2	Comparison between observed and modelled	Deposition holes below tunnels	For each deposition hole, a majority of realisations should detect whether it is above or below a given acceptance criterion.

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