Bringing service to life

serco

NAPSAC Technical Summary

Release 9.6

Our Reference: SA/ENV/CONNECTFLOW/12

Date: August 2008

SERCO IN CONFIDENCE

This page intentionally left blank

Title	NAPSAC Technical Summary
Confidentiality, copyright and reproduction	This document has been prepared by Serco Limited in connection with a contract to supply goods and/or services and is submitted only on the basis of strict confidentiality. The contents must not be disclosed to third parties other than in accordance with the terms of the contract.
File reference	connectflow/doc/user_manuals/napsac_technical.doc
Report Number	SA/ENV/CONNECTFLOW/12
Report Status	Version 9.6

Technical & Assurance Services
Building 150 Harwell IBC
Didcot
Oxfordshire OX11 0QB

Telephone: 01635 280300
Facsimile: 01635 280305
Email: gw.support@sercoassurance.com

Technical & Assurance Services is part of Serco Defence, Science and Technology, a division of Serco Ltd.

In the areas where this work was performed, Technical & Assurance Services is certified to ISO 9001 (2000) and ISO 14001.

This page intentionally left blank

Abstract

NAPSAC is a software package to model flow and transport through fractured rock. The models are based on a direct representation of the discrete fractures making up the flow-conducting network. NAPSAC uses a stochastic approach to generate networks of planes that have the same statistical properties as those that are measured for fractures in field experiments. The software is based on a very efficient finite-element method that allows the flow through many thousands of fractures to be calculated accurately.

This Technical Summary Document provides a list of the current capabilities of the program and a description of the numerical methods used.

COPYRIGHT AND OWNERSHIP OF NAPSAC

Funding provided by United Kingdom Nirex Limited towards production of the documents is acknowledged.

NAPSAC also makes use of the freely available LAPACK linear algebra library.

This page intentionally left blank

Contents

This page intentionally left blank

Preface

NAPSAC is a software package to model flow and transport through fractured rock. The models are based on a direct representation of the discrete fractures making up the flow-conducting network. NAPSAC uses a stochastic approach to generate networks of planes that have the same statistical properties as those that are measured for fractures in field experiments. The software is based on a very efficient finite-element method that allows the flow through many thousands of fractures to be calculated accurately.

The following documentation is available for NAPSAC:

- NAPSAC Technical Summary Document;
- NAPSAC On-line Help;
- NAPSAC Verification Document;
- NAPSAC Installation and Running Guide.

NAPSAC Capabilities

Simulation of fluid flow and transport in fractured rock is an essential tool for the study of water resources, oil and gas reservoir management, assessment of underground waste disposal facilities, evaluation of hot dry rock reservoirs, and for the characterisation and remediation of contaminated land. It can be used to interpret field and laboratory data, to validate conceptual models, to make quantitative predictions, and to develop practical solutions for a range of environmental, reservoir engineering, and civil engineering problems.

NAPSAC is a finite-element software package for modelling fluid flow and transport in fractured rock. A discrete fracture network (DFN) approach is used to model fluid flow and transport of tracers and contaminants through the fractured rock. NAPSAC incorporates fracture generation, flow simulation, upscaling, transport and 3D graphics capabilities using GeoVisage. The Graphical User Interface (GUI) to NAPSAC allows models to be generated and analysed quickly. A job submission ('batch') facility is included in the GUI that allows additional options not yet implemented in the GUI to be accessed - these features are indicated below[†].

NAPSAC has been developed over a 15-year period and includes a number of sophisticated capabilities, such as:
Geological modelling:

- NAPSAC has the flexibility to model a variety of scales varying from well/borehole scale to regional/reservoir scales. Detail can be included to model heterogeneity of a single heterogeneous fracture as well as models with many tens to millions of fractures at a regional or reservoir scale.
- the DFN approach allows users to compare aspects of their conceptual geologic models and field observations with simulated models. This comparison includes fracture orientation, size, transmissivity and flow distribution. An examination of the simulated network can be performed using hypothetical cores, stereonets, fracture maps and connectivity analysis.
- generation of regular and irregular meshes and structural grids (e.g. ZMap, VIP, FEMGEN and a CAD format);
- inclusion of deterministic fractures specified within the NAPSAC data or by importing a fracture file (e.g. GOCAD Vset, GOCAD Tsurf, Seisworks pointsets and other international formats). Deterministic Faults (or structures) can be used to control populations of stochastic fractures (i.e. proximity or 'Damage Zone' models). For example, NAPSAC allows the clustering of fractures around parent fractures, random points or surfaces. NAPSAC allows spatially varying fracture densities based on 3D maps of fracture drivers;
- variable distribution laws for stochastic fracture parameters. NAPSAC can generate stochastic fractures from a wide variety of probability distribution functions. Not just Fractal!
- coupling of distributions/parameters for the same feature i.e. length-aperture relationships.
- areal / volumetric distribution of stochastic fractures can be imported from external map or grid data e.g. bed thickness, curvature ('strain'), lithological (mechanical) variation.
- dynamic behaviour of 'production fractures' and present-day stress can be incorporated.
- all scale ranges, from core observation to seismic scale, can be simulated and integrated into the final model.
- high permeability 'matrix streaks' may be incorporated into models as extra flow conduits.
- flow in the matrix can be represented by additional flow channels

NAPSAC is able to:

- simulate steady-state or transient flow in a fracture-network;
- enable steady-state calculations to be performed on very large networks, because it uses an efficient finite-element scheme;
- calculate the full equivalent continuum permeability tensor including off-diagonals, principal values and principal directions. This is automated to sample flows in several different directions. This can be used for upscaling, analysis of scale dependencies and determination of the representative elementary volume (REV);
- calculate porosity and inter-fracture matrix block size;
- identify connected fracture clusters around wells;
- predict transient pressures and drawdowns at well bores for various types of pump tests;
- calculate steady-state and transient inflows to tunnels and shafts;
- calculate the effects of hydro-mechanical coupling. The hydraulic aperture is coupled to a stress distribution based on an analytical description of the stress field due to either rock overburden or a radial stress around a tunnel;
- simulate tracer transport through a network using a stochastic particle tracking method. Output[†] includes plots of breakthrough curves for many thousands of particles, particle tracks, swarms of particles at specified times or the points of arrival on the surfaces of the model. This can be used to calculate dispersion of a solute transported by the groundwater;
- simulate mass transport for a variable density fluid. This can be used to model coupled groundwater flow and salt transport[†];
- simulate unsaturated flow in fractured rocks;
- [†]analysis of percolation between surfaces.

NAPSAC can be used for the following applications:

- simulation of a range of hydrogeological tests (hydraulic borehole/well tests, including constant, head, pressure and flow tests);
- site and regional scale modelling to determine the effects of various forms of fracture flow on pressure distributions, flows and travel times to discharge points under natural conditions;
- to understand and simulate the behaviour of fracture-influenced sites/reservoirs by being able to parameterise and justify heterogeneous continuum models. For example, the estimation of equivalent parameters for input to conventional dual-porosity simulators;
- simulation of solute transport (tracer) experiments;
- simulation of the hydraulic impact of a tunnel or shaft construction;
- simulation of various simple hydromechanical models for the purpose of estimating the impact of rock overburden and in-situ stress.

NAPSAC has been used in the following industries:

- water resources for the purpose of hydraulic test and tracer simulation, fractured reservoir estimation and parameterisation, and remediation studies (such as the estimation of fracture flow in dual-porosity systems). In addition, it can be used to model saline intrusion and unsaturated flow;
- deep radioactive waste disposal, as both a tool useful for site-characterisation and safety assessments (simulation of hydrogeological tests and estimation of flow distributions and travel times to the biosphere);
- oil and gas industry to aid well planning, simulation of various well tests (PBU etc), and the parameterisation of oil simulation software by calculation of up-scaled equivalent continuum parameters (permeability, porosity and matrix block-size and distribution);

- hot-dry rock studies to estimate connectivity and parameters (permeabilities), to help analyse the effectiveness of the fractured reservoir;
- civil engineering projects concerned with construction or groundwater remediation in fractured rock. This includes the estimation of water ingress due to excavation of tunnels, studies of underground oil caverns, dam construction in fractured rocks, remediation or containment of contaminated fractured sites.

1 Introduction

The NAPSAC fracture-network modelling software was developed by Serco Limited, to simulate flow and mass transport through fractured rock.

1.1 Report Structure

The following Sections present the detailed technical specification of NAPSAC. A brief discussion of the Discrete Fracture Network (DFN) approach is given in Subsection 1.2.

The technical description is split into three phases:

- model generation (Section 2);
- calculations (Section 3);
- and output (Section 4).

These correspond to the major steps in numerical modelling. A description of the complementary GeoVisage for NAPSAC software package for 3D visualisation of NAPSAC results is given in Section 5.

Finally, the Quality Assurance programme employed for development and maintenance of NAPSAC is described briefly in Section 6.

1.2 The Discrete Fracture Network Approach

In many geological formations, the primary flow is through a connected network of discrete fractures. This provides a very heterogeneous system, and the fracture-network geometry can lead to dispersion of any solute being transported through the formation. It is often necessary to show sufficient understanding of the flow system to give confidence that predictions of the large-scale properties of the flow system can be made from the results of field-scale investigations. In order to build confidence, it is important to show a very detailed understanding of field experiments that are generally on scales at which the influence of the fracture-network geometry is significant. The geometry and connectivity of the fracture system and the possibility of hydraulically important pathways through the network can play an important role in determining the scale dependence of the effective properties of the system. Indeed, one of the early motivations for the development of the DFN approach was to develop an understanding of the scale dependence of the effective dispersion parameters for radionuclide transport through fractured rock, which had been inferred from field data ([2] for example).

In the DFN approach, the geometry of the fracture-network is accounted for explicitly. The approach is needed to describe or predict aspects of the performance of the fractured system where the geometry of the fracture-network plays a significant role. Some examples of such circumstances are:

- representations of any flow experiments where the fracture connectivity is important, which in practice means almost all interpretations of field experiments where a detailed understanding is needed;
- prediction of the effective flow properties of the fracture-network system and of the scale dependence of effective properties;
- prediction of the effect of the fracture-network geometry on the effective dispersion for solute transport;
- prediction of the effect of the fracture-network geometry on the effective hydraulic diffusivity of the pressure field in response to a pressure change and the inferred radius of influence of pressure tests.

From the above list, it can be seen that an understanding of the role of the fracture geometry can be important in almost all aspects of an investigation of a fractured rock system. The two main reasons that such discrete models are not more commonly used are the complexity of the models and the fact that stochastic models inevitably require uncertainty to be addressed formally.

The complexity means that many data are required to characterise fracture systems adequately. Whilst there are still issues to be resolved in the experimental characterisation of fracture-network flow geometry, a number of research projects for the radioactive waste industry have demonstrated the feasibility of collecting suitable basic input data [3, 4, 5, 6]. Understanding fracture channelling and the extent of the flow wetted surface of the fracture are still research tasks, but simple assumptions can be made and the other data interpreted consistently so that the resulting fracture-network geometry reproduces key features of the physical network. In many cases, however, there will be a balance between the benefits of a more detailed representation of the system, and the increased cost of collecting data for which there may be significant uncertainty.

The second reason why the discrete fracture-network approach is not more widely used is the need to treat predictions in a probabilistic framework and consider the uncertainty due to the details of the fracture geometry directly. Fracture-network models are necessarily stochastic since it is not possible to determine the location and extent of each flow-conducting or mechanical break in the rock. Instead a stochastic approach is used, in which the statistics of the fracture system are determined and realisations of the fracture-network geometry that exhibit the same statistics as the physical system are generated and used for simulation. This means that a discrete fracture-network approach does not predict the result of a given experiment. Instead, it predicts a probability distribution of equally likely results given the stochastic description of the fracture geometry and properties. This realisation-dependent uncertainty corresponds to a lack of knowledge of the precise fracture geometry. In many respects this is an advantage of the approach over deterministic models since the uncertainty is real and unavoidable. Conventional approaches often make single-valued predictions, however this is simply not facing up to the reality of uncertainty.

1.3 Applications

Applications of NAPSAC include:

- interpreting site characterisation data;
- modelling of flow and transport in regional fracture-network systems;
- obtaining effective properties as data input to large-scale effective porous medium models.

In site characterisation programmes, NAPSAC has been used to validate the fracture network approach by comparing data from hydrogeological experiments in fractured rock (e.g. well tests) against model predictions. As part of the assessment of post-closure performance of potential deep repositories, NAPSAC fracture-network models have been used to predict the groundwater pathways by which radionuclides released from a repository might return to the environment. Effective properties have been obtained using NAPSAC for input into large-scale 3-D porous medium models or reservoir simulators (for example [7]).

2 NAPSAC Model Generation

The first phase in a NAPSAC calculation involves the:

- definition of the **model domain**;
- **fracture generation**;
- inclusion of **engineered features** (boreholes, tunnels or shafts);
- calculation of **fracture intersections**.
- provision of **boundary conditions**;

These processes are described in this Section.

2.1 Model Domain

The model is defined within a domain formed from the union of a number of (possibly irregular) hexahedra or `region elements'. The region elements are defined by supplying a list of the coordinates of the vertices belonging to each element. Where the faces of two region elements are joined, the four corners of the adjacent sides must be coincident. The faces of the region elements need not be planar. In general, they form bilinear surfaces. An example of a complex flow domain is shown in Figure 1.

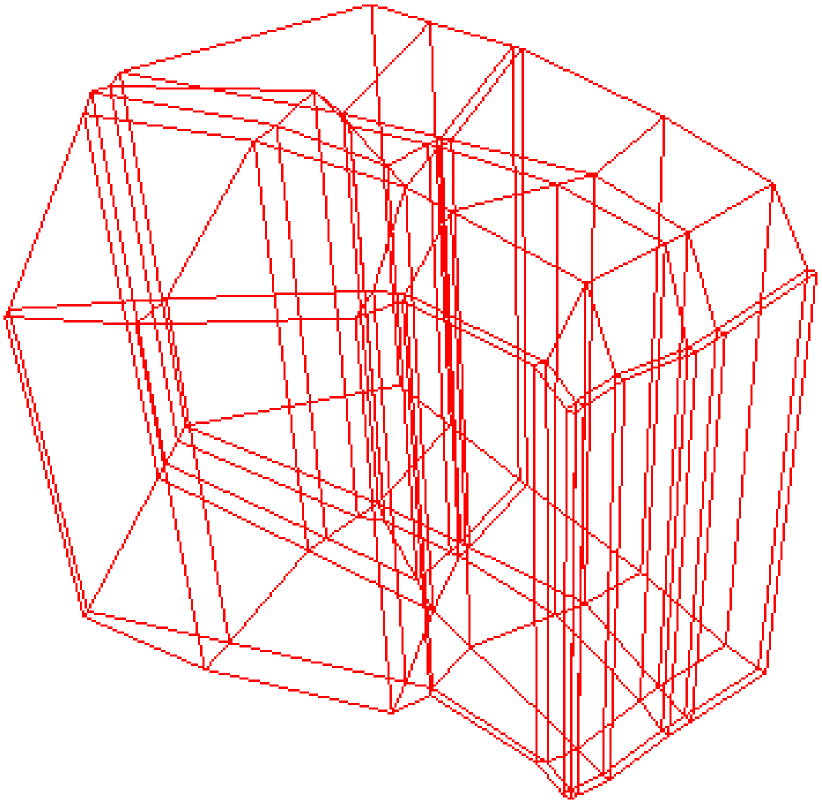


Figure 1 An example of a complex flow domain built from 37 irregular hexahedra.

2.2 Fracture Generation

An individual planar fracture is completely defined by:

- the location of its centre;
- three orientation angles (dip angle, *ψ*, dip direction, *α*, and orientation, *ω*);
- the lengths of each side (or in the case of square fractures, a single length);
- an effective hydraulic aperture or transmissivity (and possibly the variation of aperture within the fracture).

The definition of the orientation angles (*ψ*, *α*, *ω)* relative to the Cartesian coordinate system (*x, y, z*) is shown in Figure 2. It is usual to orient the axes such that *x* is east, *y* is north, and *z* is vertically upwards.

Fractures can either be `known'  (deterministic) in which case the above properties are specified explicitly, or `random' in which case the fracture properties can be sampled from a wide range of statistical distributions.

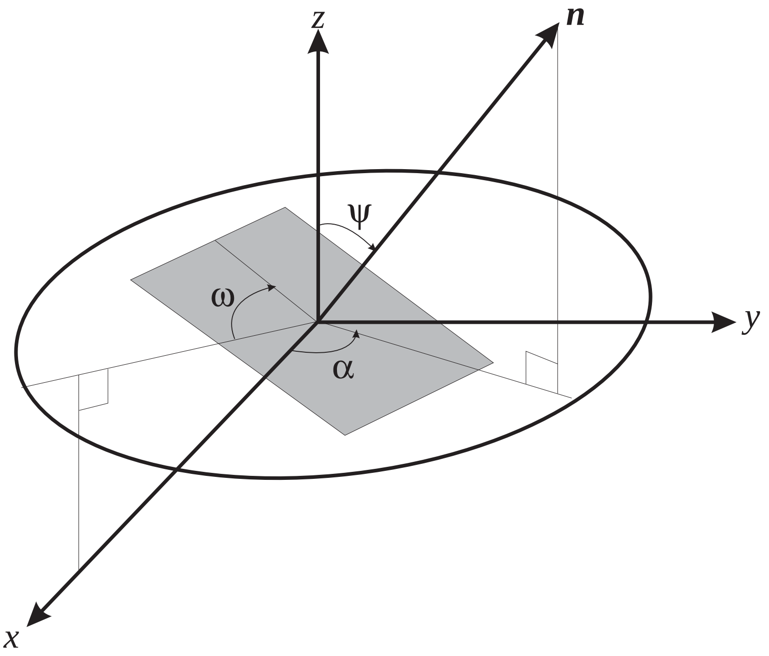


Figure 2 The angles describing the orientation of a fracture, in relation to the coordinate axes and the normal to the plane, n. *ψ*: the dip angle, *α*: the dip direction, *ω*: the orientation angle.

2.3 Fracture Network Characterisation

This section describes the main methods for inferring fracture-network geometries from field measurements of the fracture-network properties [8,9]. This is the first major task the user faces in three-dimensional simulations. Analogous methods are used in the derivation of appropriate two-dimensional equivalent networks.

The key parameters used to characterise a fracture-network are:

- identification of independent fracture sets;
- the distribution of fracture orientations;
- the statistical process for generating the fracture locations in space;
- the fracture density;
- the distribution of fracture lengths;
- and the distribution of fracture transmissivities.

When characterising the fracture orientation distribution, it is generally found that the fractures can be divided into a number of distinct fracture sets. These sets of fractures comprise fractures that can be characterised by common distributions of parameters, and which have a common origin and history.

These fracture sets are often defined in terms of their orientation distributions that tend to be clustered around preferred orientations of the normals to each fracture plane projected on to a lower hemisphere. This definition of the characteristic orientation is best achieved by using conventional statistical methods to identify distinct clusters. The fractures can then be separated into their distinct sets and further parameters inferred for each set independently.

The distribution of fractures has commonly been assumed to be uniform in space with just a single fracture density being used to specify how many fractures to generate. An equivalent approach to using a fracture number density is to generate fractures up to a specified area density of fracture surfaces per unit volume. Sampling the various distributions of the parameters generates the fractures. The positions of the fracture centres are generated assuming a Poisson process. Care must be taken to avoid edge effects, and this is usually accomplished by generating the fracture-network in a larger region than that to be simulated.

The fracture density may be obtained from the spacing of fractures along a scan line on a mapped exposure, or from a fracture log along a borehole or core. Each distinct set of fractures has its own characteristic distributions of properties, and the density of each of these fracture sets is usually determined independently. For a given fracture set, the number density, ρ, is given in terms of the mean spacing of intersections along a straight line, $\bar{s}$, by:

$$\bar{s} = (\rho \bar{X})^{-1}, \qquad (2.1)$$

where $\bar{X}$ is the mean projected area of the fractures onto a plane perpendicular to the measurement line.

The fracture set length distribution is one of the more difficult parameters to infer since we have only one- or two-dimensional data from which to infer a length distribution that will only be fully determined by a three-dimensional description. A number of assumptions need to be made at this stage. First, it is difficult to characterise the shape of the transmissive area of the fracture plane. It is generally assumed that this surface has a simple geometry. In NAPSAC, it is assumed to be rectangular.

Once the fracture shape has been fixed, then one can use analytical results giving the relationship between the distribution of fracture lengths to the distribution of fracture trace lengths as measured on a large two-dimensional trace plane intersecting the network. For example, for square fractures of side length distribution, *L*, the moments of the length distribution, L_i, are related to the moments of the corresponding distribution, *t*, of fracture traces measured on a large trace mapping plane by:

$$t_1 = \frac{\pi}{4}\frac{L_2}{L_1}, \qquad (2.2a)$$

$$t_2 = \left[\ln\left(1+\sqrt{2}\right) - \frac{\sqrt{2}-1}{3} \right] \frac{L_3}{L_1} \qquad (2.2b)$$

where L_i are the *i*-th moments of the length distribution and t_i are the *i*-th moments of the trace length distribution. Similar formulae can be obtained for higher moments. A common approach is to make an assumption as to the mathematical form of the distribution of fracture lengths and then either use these simple formulae between the means and second moments of the distribution, or to simply calibrate against statistics from a specific trace map. In fact the trace length to fracture length relationship is quite insensitive to the precise shape assumed for the fractures and there is relatively little difference between the results for circular or square fractures. A more significant assumption is the choice of the mathematical form of the fracture length distribution. Typically, log-normal or power law distributions are used. Although these often result in a good fit between the main parts of the simulated and measured trace length distributions, the goodness-of-fit of the tails of the two distributions is often less good. A poor match in the tail of the distribution may result in the existence of extreme, unphysical fractures with very long traces. These are quite unimportant to many of the statistics used to infer parameters but may have a much more important role in the network flow.

Finally, the hydraulic properties of the fractures need to be defined. The usual assumption is that some form of the parallel plate law for plane fracture flow applies, but rather than measure a distribution of apertures directly, a more reliable approach is to infer a distribution of fracture transmissivities. This too, generally relies on an assumption as to the form of the probability distribution of fracture transmissivities. Generally the log-normal distribution is used. With this distribution and a specified fracture spacing, then the mean and standard deviation of fracture transmissivities can be related to the mean and standard deviation of short interval packer tests in boreholes so long as it is assumed that the transmissivities of fractures intersecting the test section add to give the transmissivity of the test section. Strictly, fracture connectivity away from the borehole will affect the packer test results, but for short tests, the radius of influence of the test will be small and the measurements can be taken to correspond to the summation of local transmissivities. The fitting process involves typically using maximum likelihood estimators and in general will require numerical evaluation of the best estimates. Again, the results of the fracture property interpretation should be checked by simulation of the measurement process and it may be appropriate to infer the parameters of the distribution by calibrating directly against the experimental data (see [9] for more detail).

An alternative approach to generating the fracture-network, which is often used, is to generate fractures using an initial approximation and test the resulting network by simulating the experimental measurement procedures. Then the network is modified to improve the correspondence between, for example, the numerically simulated log and the physical log. This calibration procedure is particularly appropriate when simulations are made based on more complex statistical descriptions of the fracture properties and spatial densities. Such simulated measurements should in any case be used to check the validity of the interpretation of the network parameters.

2.4 Known Fractures

For known (or deterministic) fractures, all properties are specified either within the NAPSAC data or imported from a formatted file. There are several instances when the use of known fractures is appropriate. Firstly, when large-scale fracture zones are defined. The appropriate transmissivity for such zones may be obtained from hydraulic tests or by calculating an effective transmissivity based on a local scale stochastic model of the fracturing around the zone.

Secondly, where the network has been well characterised, the stochastic network may be replaced by the set of fractures that have been measured. For example, a stochastic network of fractures may be generated based on a statistical analysis of the data from several boreholes.

The random fractures around each borehole can then be replaced by the observed fractures at the borehole. This is a simple method of conditioning random simulations.

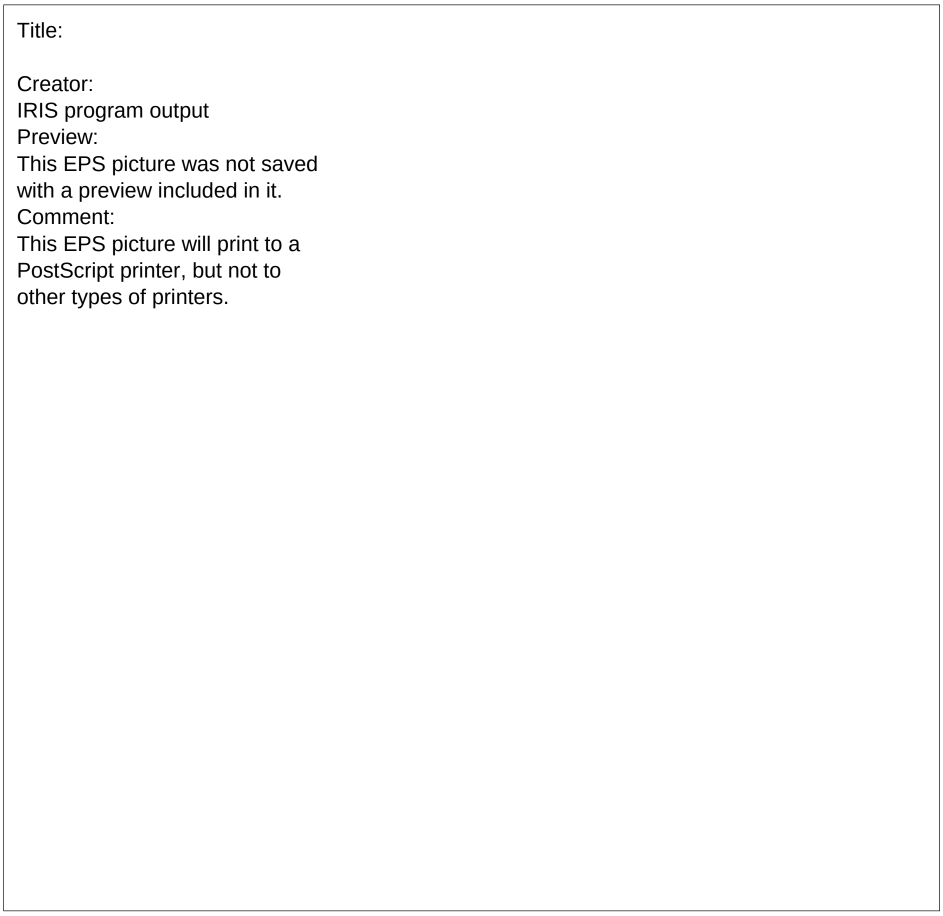
2.5 Random Fractures

Up to 50 separately parameterised sets of random fractures can be defined in a NAPSAC model. The locations of the centres of the fractures are distributed uniformly within a cuboidal region whose boundaries are set by the user. This region should be sufficiently larger than the flow domain, bearing in mind the expected size of the fractures, so that there is no reduced density of fractures near the edge of the flow domain. For each of the other fracture properties, the user specifies the distribution type and its parameters.

In NAPSAC the following distributions are available:

- constant;
- uniform;
- normal;
- log-normal;
- two parameter negative exponential;
- triangular;
- log-triangular;
- univariate Fisher (for dip angles and dip directions only);
- truncated log-normal;
- power-law (for fracture lengths only).

Given this information, fractures are generated randomly up to a user-prescribed density. A typical NAPSAC generated fracture-network is shown in Figure 3.



Title:

Creator:
IRIS program output
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

Figure 3. An example of a fracture-network generated within a cuboid flow domain. The fractures are coloured according to the logarithm of transmissivity: red for high transmissivity, blue for low.

2.5.1 Variable Apertures on Fractures

As well as random fractures, each with a uniform aperture, NAPSAC can represent the random variations of aperture within a given fracture. This option may be specified for some or all of the sets of random fractures, and also on a known fracture. The local values of the aperture are generated from a log-normal distribution with standard deviation prescribed by the user. With a known fracture, the mean value of the distribution is simply the aperture given by the user.

In the case of a random fracture, a value is first randomly sampled in the same way as for a uniform fracture, but this value is then used as the mean aperture about which the local aperture distribution on the fracture is generated. Note that the standard deviation of the aperture distribution for a single fracture need not be the same as the standard deviation of the mean apertures of the fracture set.

In reality, the apertures at nearby points on the same fracture may well be correlated to some extent. To represent this, NAPSAC provides the user with the facility to specify a correlation length scale and one of 3 correlation functions (note that these are the distribution of the logarithm of the aperture):

$$
\begin{array}{ll}
\rho_1(\xi) = 1 - \dfrac{2\xi}{2l} - \dfrac{\xi^3}{2l^3} & \xi < l, \\
\quad\quad = 0 & \xi \geq l; \\
\rho_2(\xi) = \exp(-\xi / l); & \\
\rho_2(\xi) = 1 - (2\xi l - \xi^2)^{1/2} / l & \xi < l, \\
\quad\quad = 0 & \xi \geq l;
\end{array}
$$

where ξ is separation and *l* is the correlation length scale. If uncorrelated apertures are required, this may be achieved by using either correlation function 1 or 3, with a zero correlation length.

2.5.2 Fracture Sub-division (`Tessellation')

A simpler method of generating a variable transmissivity on each fracture (tessellated fractures) is to sub-divide the fractures into smaller fractures (sub-fractures) according to an approximate correlation length, and generate the transmissivity on each sub-fracture independently. In this way fractures are generated according to a specified length distribution, but then sub-divided such that no sub-fracture is longer than the correlation length. Hence, the number of fractures increases but the fracture area density is maintained. This method is more appropriate for large random networks.

Another reason for sub dividing fractures is one of discretisation. The number of finite elements used to discretise each fracture is similar on every fracture, irrespective of the fracture length. If fracture lengths vary by orders of magnitude then large fractures may be under-refined, and small ones over-refined. Hence, tessellation can be used to split large fractures into a network of sub-fractures of a more uniform size, and consequently a more uniform discretisation. In this case the transmissivity of the sub-fractures is inherited from the transmissivity of the tessellated-fracture from which it was created.

2.6 Engineered Features

Since field experiments in fractured rock usually involve boreholes, a model feature (`borehole') is provided to facilitate their incorporation in simulations. There is a further feature (`shaft') to represent engineered features of larger radii such as tunnels or shafts. Both types of feature are specified by the coordinates of the two ends and a radius. Both models add extra flow connections to the network where the engineered features intersect fractures. Generally, if the radius of the engineered feature is small compared to the length of fractures then the borehole submodel is adequate, otherwise greater accuracy is gained by using the shaft model. For a borehole only the fractures intersecting a line joining the two ends are hydraulically joined to the borehole. For a shaft all fractures intersecting a cylinder with the specified radius and axis are hydraulically joined to the shaft. The axial hydraulic conductance (units of $m^3 s^{-1}$) of the feature is calculated from the radius or from a specified permeability. The properties of a skin layer due to grouting or skin effects can be specified to limit radial flow. Section 3.5 gives more details on the simulation of flow around engineered features.

Any two engineered features may be joined hydraulically. Hence, curved boreholes can be represented by several joined boreholes of varying inclination. Figure 4 gives an example of a complex model with many sections of engineered feature joined together to form a spiral tunnel. Two vertical shafts are also shown.

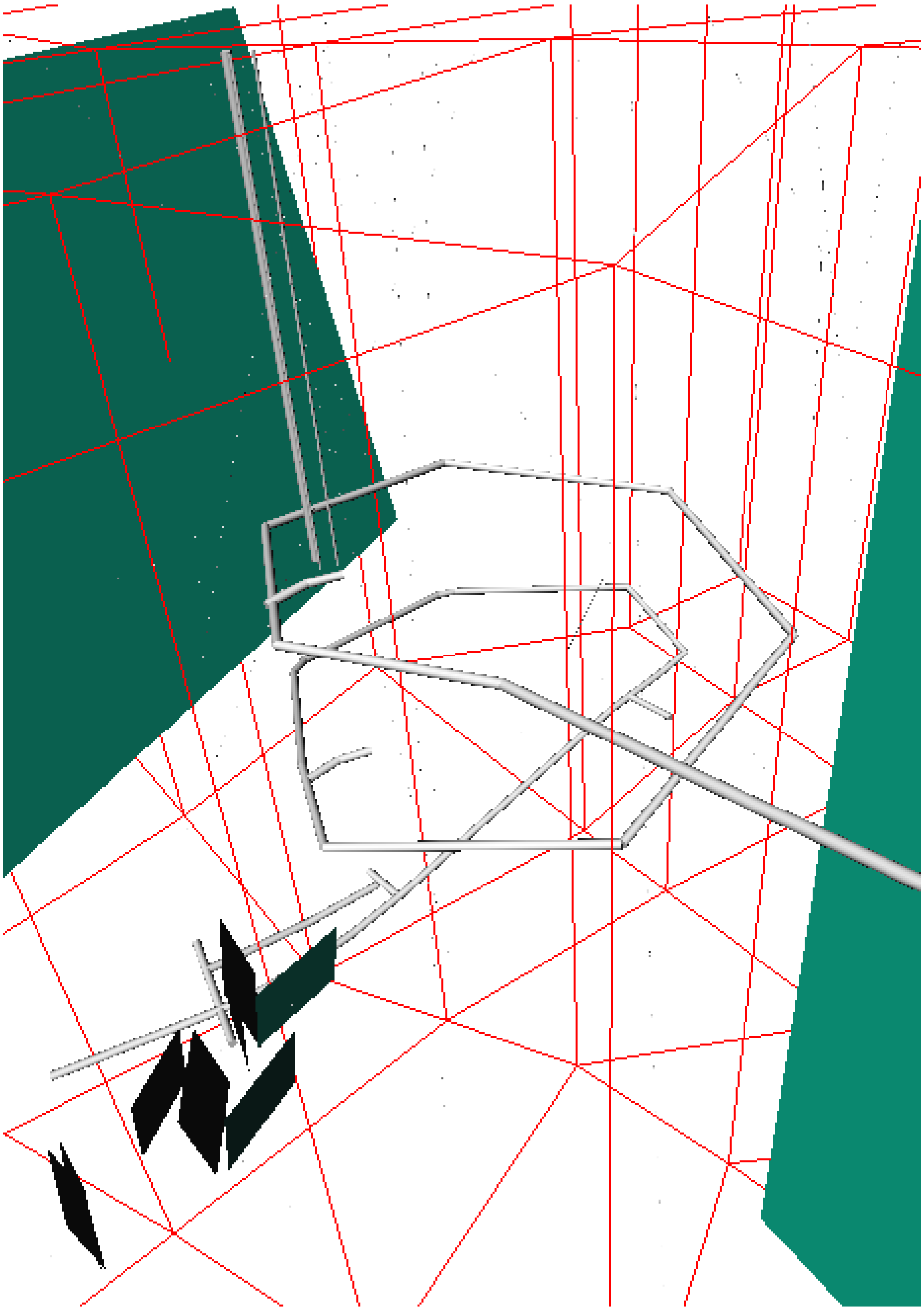


Figure 4 An example representation of a complicated system of tunnels (spiral) and shafts (vertical). Some region element edges, two large deterministic fractures, and a few random fractures are also shown.

2.7 Fracture Intersections

Once the fracture-network has been generated, the next step is to calculate all the fracture intersections. This allows an interpretation of the fracture-network connectivity. Intersections between the planes and the boundaries of the flow domain are calculated, and part or all of a fracture falling outside the flow domain is discarded.

The intersections are determined by solving the equation for the intersection of the two fracture planes using elementary geometry. In order that large networks can be handled the search for intersections is optimised by dividing the overall region into subregions and determining the planes wholly or partly within each subregion, and then only testing planes in the same subregion for intersections. In this way the asymptotic cost of the calculation of the intersections is proportional to the number of planes rather than the square of the number of planes. The flow field is discretised by assigning a number of nodes, referred to as the global flow nodes, to each intersection.

2.8 Boundary Conditions

Boundary conditions are set on the boundaries of the flow domain. By default, NAPSAC treats any boundary for which no condition is set as impermeable. For a permeable boundary, either a pressure distribution or a fluid mass flux can be specified. A pressure distribution can be defined in three ways:

- a constant value may be set over the whole surface;
- a linear pressure variation may be specified;
- the pressure can be interpolated (bilinearly) from a set of pressure values at the region element vertices. These pressure values can be specified in the input data file or interpolated from a regular mesh that is read from a file.

For flux boundary conditions a uniform fluid mass flux in units of $kgm^{-2}s^{-1}$ is specified over a boundary surface. A mass flux enters each fracture that intersects the surface. The amount of flux entering a particular fracture is weighted according to the length of the fracture’s trace, such that the total mass flux entering the surface equals the mass flux value specified multiplied by the area of the surface.

In addition to the boundary conditions set on the flow domain boundaries, the user may specify the pressure or flux on individual engineered features.

For mass transport the salinity can be specified on selected surfaces. The default boundary condition for this case is zero dispersive flux (an outflow condition).

3 NAPSAC Calculations

This Section details the various types of calculations that can be performed using NAPSAC.

3.1 Geometric Analysis

Prior to a flow calculation useful information can be gained by analysis of the fracture intersections. One of the main characteristics of a fracture-network that controls the behaviour of the flow is the connectivity of the network.

3.2 Percolation Analysis

The most basic measure of connectivity is whether the fracture-network has a connection across the region or not. This depends on the fracture density and the change from unconnected to connected networks is predicted by the percolation threshold. The percolation threshold gives the density at which the size of connected clusters of fractures suddenly increases from a relatively small typical cluster size to the existence of a percolating cluster that spans the region. The percolation threshold is quite a sharp transition: a small increase in fracture densities will change the network from one for which no realisations have connections across the model region to one for which all realisations are well-connected [10]. This percolation threshold depends upon the statistical properties of the network, but for random networks there is a much more significant dependence on the dimension of the network geometry. Three-dimensional networks become well connected at much lower fracture densities than two-dimensional networks.

The NAPSAC percolation option builds the network one fracture at a time. As each individual fracture is included, a list of fracture clusters is maintained and updated. If the fracture intersects any fracture belonging to an existing cluster it is added to the appropriate cluster list. If it intersects with two or more disjoint clusters, then the cluster lists are combined. If it is isolated, then a new cluster list is started. When a single cluster connects all the relevant boundary surfaces, percolation has occurred, and any remaining fractures in the original network may be discarded if required.

3.3 Steady-State Constant Density Groundwater Flow

Steady-state constant-density groundwater flow in a fracture network can be modelled in the current version of NAPSAC. The basic approach is very simple. Groundwater flow in each fracture is modelled numerically. Then the flow in the overall network is obtained by combining the flows in the different fractures, using the conditions that

(i) the groundwater pressure is continuous between two intersecting fractures;

(ii) groundwater is conserved at an intersection, so that groundwater which flows out of one fracture flows into the other.

NAPSAC uses a Galerkin finite-element approach to modelling. The Galerkin approach starts from the weak form of the governing equation, which is derived by multiplying the governing equation by an arbitrary test function in a suitable function space, and integrating over the domain, integrating by parts terms involving high-order derivatives (see Equation (3.4) below). The benefit of this manoeuvre is that the weak form is equivalent to the original equation for sufficiently smooth functions, but it is also applicable to functions that are not as smooth, such as the functions derived by finite-element discretisation.

In the finite-element method, the domain is discretised into 'finite elements' of simple shape. On each element, a quantity of interest is approximated by a simple function, such as a polynomial, determined by the values at a small number of points, or nodes on the element. This is equivalent to approximating the quantity of interest as a linear combination of certain basis functions that are associated with the nodes; the basis function associated with a node taking the value 1 at the node and 0 at all the other nodes. The discretised equations are obtained by taking the test functions in the weak form to be the set of basis functions for nodes where the value of the quantity is not specified by a Dirichlet boundary condition. These equations are supplemented by the Dirichlet boundary conditions.

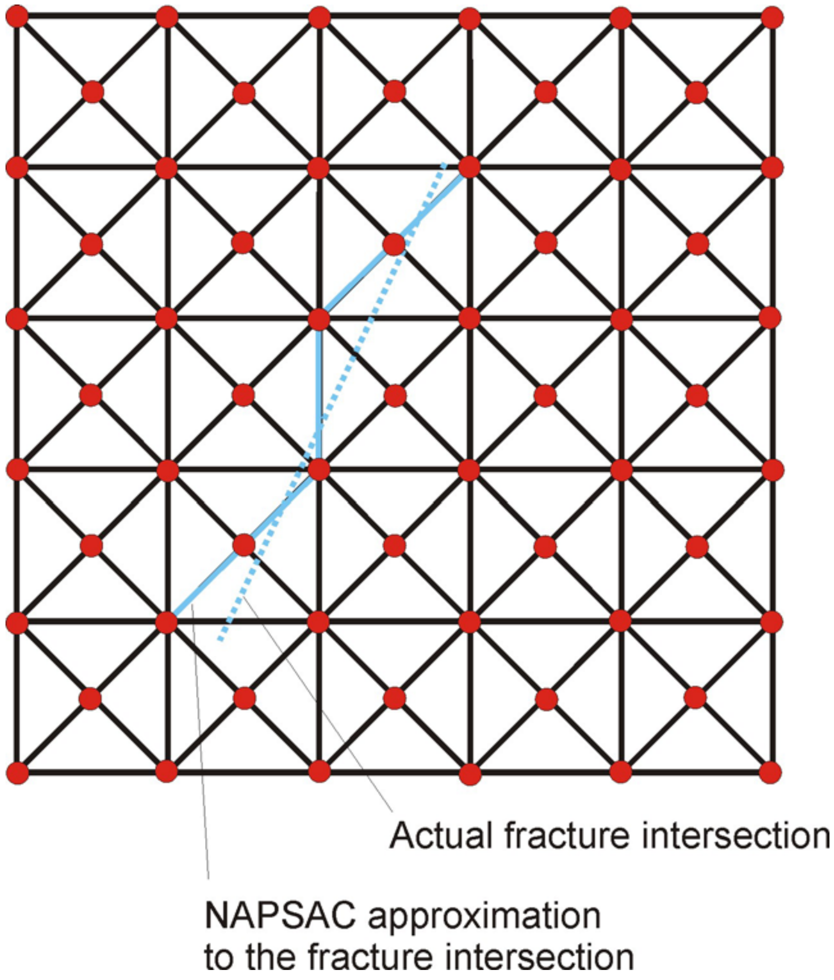


Figure 5 The finite element discretisation of a fracture in NAPSAC and the approximation of intersections with other fractures by lines along finite-element boundaries.

In NAPSAC, the finite-element method is applied on two levels: individual fractures and fracture intersections. Individual fractures are discretised into triangular elements as shown in Figure 5. On each element, the residual pressure

$$P^R = P + \rho_0 g(z - z_0) \qquad (3.1)$$

is approximated as a linear function. (Here

P is the groundwater pressure;

ρ_0 is a reference value of the groundwater density;

z is the elevation;

z_0 is a reference elevation.)

As noted above, this is equivalent to approximating the residual pressure on the fracture as a linear combination of the 'local basis functions'.

On the scale of the overall network, the residual pressure is characterised by its values at certain 'global nodes' associated with the fracture intersections, which are approximated by lines along the boundaries of the elements representing the fracture (see Figure 5). On a fracture, the global basis function Ψ_I corresponding to a global node I on one of the intersections with other fractures is taken to be the finite-element solution for steady-state groundwater flow on the fracture in the case in which the residual pressure is specified to be 1 at global node I and 0 at all the other global nodes on the fracture.
The steady-state groundwater-flow equation in the fracture is

$$-\nabla \cdot \left(T \nabla P^R \right) = 0 \tag{3.2}$$

where

ρ is the groundwater density;

μ is the groundwater viscosity;

T is the transmissivity of the fracture, which is related to the fracture aperture;

∇ denotes the two-dimensional gradient operator in the fracture.

(The possibility that the fracture aperture, and hence fracture transmissivity, may vary over the fracture surface is allowed.) The transmissivity is generally taken to be related to the fracture aperture e by the cubic law

$$T = \frac{\rho g e^3}{12\mu} \quad . \tag{3.3}$$

on the basis of flow between parallel plates [11].

The weak form of Equation (3.2) is

$$\int_F \nabla w \cdot T \nabla P^R + \int_B w \mathbf{n} \cdot \mathbf{f} = 0 \quad , \tag{3.4}$$

where

F is the fracture;

B is that part of the boundary of the fracture on which the flux is specified. Note that B includes both sides of intersections with other fractures;

$\mathbf{n}$ is the unit normal to the boundary directed out of the domain;

$\mathbf{f}$ is the groundwater flux density on B;

w is a test function.

The approach used in NAPSAC allows considerable flexibility in the number and location of the global nodes. This allows highly refined models to be used for accuracy or coarser models to be used in order to keep computational costs down as appropriate. However, the details of the method are quite complicated in the most general case. Therefore, the basis of the approach is first presented in the case in which the global nodes are identical to the local nodes on the intersections (see Figure 6), and then the modifications for more complicated cases are indicated. In the simplest case, the finite-element equations that characterise the global basis function for node I (denoted ψ_I in this case) are

$$\int_F \nabla\phi_n \cdot T\nabla\psi_I = 0 \text{ , for local nodes } n \text{ not on intersections,} \qquad (3.5)$$

supplemented by the boundary conditions

$$\psi_I = \begin{cases} 1 \;\; for \;\; j = local\ node\ i\ corresponding\ to\ I \\ 0 \;\; else \end{cases} \text{ , for local nodes } j \text{ on the intersections.} (3.6)$$

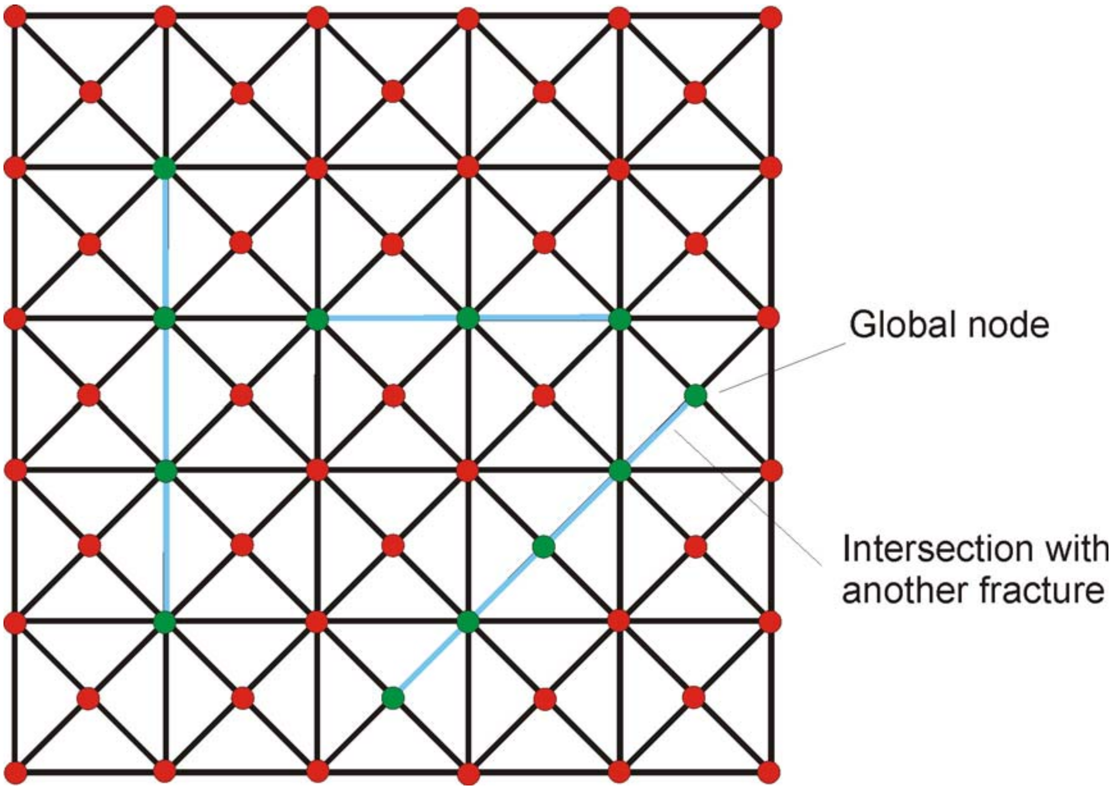


Figure 6. Intersections on a fracture in the case in which they lie along finite-element boundaries and the global nodes correspond to the local nodes on the intersections.

These equations can be readily solved using a suitable numerical scheme. In NAPSAC, a direct solver, which employs a variant of Gaussian elimination, is used. The solver is purpose-built to exploit the structure of the numerical equations resulting from the regular form of the finite-element grid used in NAPSAC. Essentially, the equations for the nodes at the centre of each rectangular block (see Figure 5) are eliminated first, to give a matrix with the same form as that for a regular rectangular discretisation. This matrix is then solved used straightforward Gaussian elimination. This is a very efficient approach.

It can easily be seen that the global basis function for global node I is equal to the local basis function for the local node i corresponding to I plus a linear combination of local basis functions for local nodes n not on intersections, that is

$$\psi_I = \phi_i + \sum_n b_{In}\phi_n \quad , \tag{3.7}$$

for certain constant coefficients b_{In} (which depend on the geometry of the fracture and its intersections).

Because of the linearity of the flow equation, the finite-element solution for steady-state flow on a fracture in the case in which the residual pressure has values P_I^R at the global nodes is given by a linear combination of the global basis function:

$$\hat{P}^R = \sum_I P_I^R \psi_I \quad . \tag{3.8}$$

It is easy to see that this ensures that the residual pressure is continuous between intersecting fractures (condition (i) above), because on each fracture the residual pressure on the intersection is given by the same interpolation between the global nodes on the intersection. In accord with the basic Galerkin approach, condition (ii) above is imposed in a weak way:

$$\sum_{\text{intersecting fractures}} \int \psi_I(\alpha) Q_{fracture}(\alpha) d\alpha = 0 \quad , \tag{3.9}$$

where

α is a coordinate along the intersection;

$Q_{fracture}(\alpha)$ is the flux to the intersection on the fracture.

A key issue is how the flux to a fracture intersection is calculated. In fact, in order to impose the constraint of Equation (3.9) it is not necessary to calculate the flux itself, but only the integrals that appear in equation (3.9). These are calculated from the following quantities

$$Q_i = -\int_F \nabla\phi_i \cdot T\nabla\hat{P}^R \quad . \tag{3.10}$$

As can be seen from Equation (3.4), if, rather than specifying a Dirichlet boundary condition on an intersection, the flux to the intersection were specified to be Q, then the finite element equation for a local node i on the intersection would be

$$-\int_F \nabla\phi_i \cdot \frac{\rho T}{\mu}\nabla\hat{P}^R = \int_{\text{Intersection}} \phi_i Q \quad . \tag{3.11}$$

The left hand-side of this equation is just Q_i so that it can be seen that it is natural to call this the flux to node i on the intersection.

Now, from Equation (3.6),

$$\psi_I = \phi_i \quad \text{on the intersection containing } i \tag{3.12}$$

where i is the local node corresponding to global node I .
(This is because all the other local basis functions appearing in Equation (3.5) are zero on the intersections.) Thus, the integral that appears on the right-hand side of Equation (3.9) can also be written as

$$\int_{\text{Intersection}} \psi_I Q \quad . \tag{3.13}$$

Therefore, the quantities Q_i give the natural way to evaluate the integrals appearing in Equation (3.9). Further, it should be noted that Q_i can be expressed as

$$Q_i = -\int_F \nabla \psi_I \cdot T\nabla \hat{P}^R \quad . \tag{3.14}$$

because

$$-\int_F \nabla \psi_I \cdot T\nabla \hat{P}^R = -\int_F \nabla\left(\phi_i + \sum_n b_{In}\phi_n \right)\cdot T\nabla \hat{P}^R = -\int_F \nabla \phi_i \cdot T\nabla \hat{P}^R \quad . \tag{3.15}$$

using Equation (3.5).

The flux on a fracture to a node on an intersection calculated as described above is a linear combination of the residual pressures at the global nodes on the fracture. The equations for the conservation of groundwater at a fracture intersection therefore give linear relations between the residual pressures at the global nodes on the intersecting fractures. These equations, together with any boundary conditions on the fracture-network model provide the overall set of discretised equations for the residual pressures at the global nodes. In NAPSAC, these equations can be solved using either a direct solver or a preconditioned conjugate gradient method.

It is worth noting that the approach described above leads to the equations that would be obtained were the finite-element approach to be applied directly to all the fractures together, with the basis functions for a node on an intersection being taken to be defined on both intersecting fractures in the obvious way. That is, on each fracture, the basis function would have the form of the basis function for the appropriate local node. This result is almost trivial because the equation for the conservation of groundwater for a global node on the intersection would then just be the sum of the contributions to the finite-element equation for the basis functions for the corresponding local node on each fracture, which is exactly what is obtained for the corresponding overall basis function defined as above. This is exactly the same as Equation (3.9). Thus the approach is a very natural one.

In the discussion above, the approach used in NAPSAC has been presented for the simplest case in which global nodes are identical to local nodes on fracture intersections. However, NAPSAC allows considerable flexibility about the number and position of global nodes on intersections. (This means that highly refined models can be used for accuracy or coarser models can be used in order to keep computational costs down, as appropriate.) The flexibility about the global nodes complicates the analysis slightly.

First, there may be fewer global nodes along an intersection than local nodes. This can readily be handled by a minor extension of the approach previously described. It is simply necessary to take the global basis functions to be the appropriate linear combinations of the global basis functions described above. In fact

$$\Psi_I(\alpha) = \sum_j \Psi_I(\alpha_j)\psi_j(\alpha) \quad . \tag{3.16}$$

This is illustrated in Figure 7.

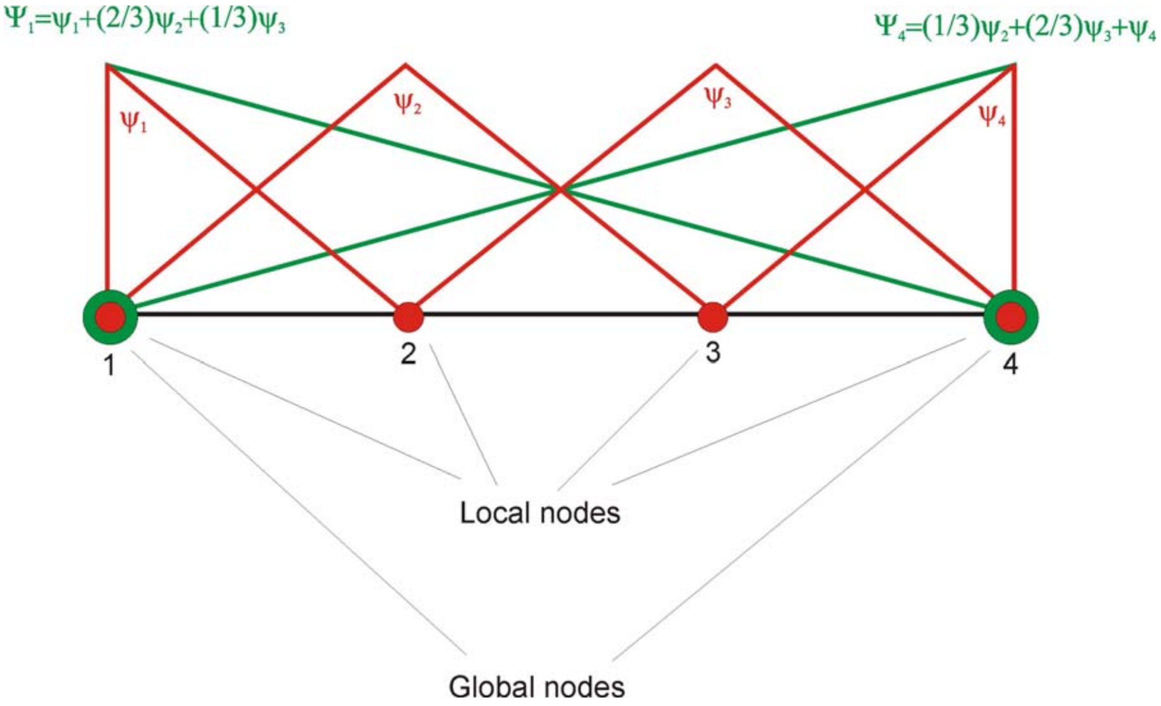


Figure 7. The relation of local and global basis functions on an intersection in a case in which there are fewer global nodes than local nodes on the intersection

Second, it is possible that two (or more) fracture intersections may themselves intersect. (Although the probability of more than two intersections intersecting is vanishingly small for the physical fractures, it is quite possible that this may occur in the NAPSAC model with fractures approximated to lie along element boundaries.) Provided that there are global nodes at the 'multiple intersection points' where two (or more) fracture intersections, then the discussion above applies unchanged. However, if global nodes are not present at multiple intersection points, then the residual pressure and the groundwater flux are effectively double counted at the multiple intersections, because there are separate contributions to the residual pressure (or flux) from each intersection. In order to avoid this double counting, the global basis functions are modified by reducing the contributions form the local node associated with a multiple intersection point. The contribution to each affected global basis function is divided by the number of fracture intersections crossing the local node, that is

$$\Psi_I(\alpha) = \sum_j \frac{1}{N_j} \Psi_I(\alpha_j)\psi_j(\alpha) \quad , \qquad (3.17)$$

where N_j is 1 if there is a global node at local node j, and the number of fracture intersections intersecting at node j otherwise. This is illustrated in Figure 8. The approach introduces an approximation, which was tested during the development of NAPSAC and found to be acceptable.

The fluxes to the global nodes are calculated from

$$Q_I = -\int_F \nabla\Psi_I \cdot T\nabla\hat{P}^R = -\sum_J \int_F \nabla\Psi_I \cdot T\nabla\Psi_J \hat{P}_J^R \quad , \qquad (3.18)$$

which is a straightforward extension of Equation (3.14). These fluxes are used, for example, in the approximate particle tracking calculations.

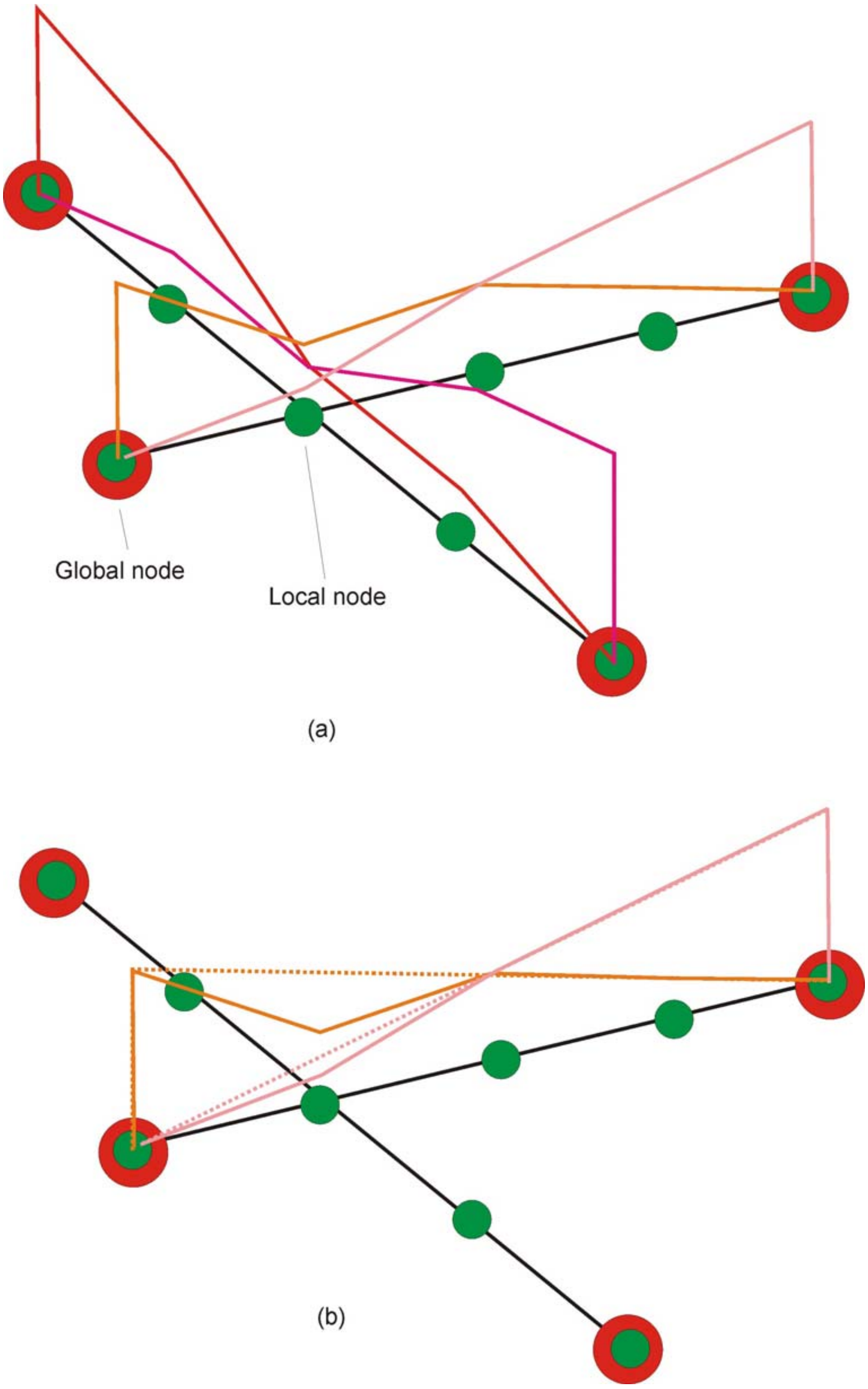

Global node
Local node
(a)
(b)

Figure 8 **The global basis functions for a case in which fracture intersections intersect. (a) the basis functions for the global nodes (b) the construction of the basis functions on fracture intersection from the basis functions for local nodes**

The calc
shows a
random ent
scale fo

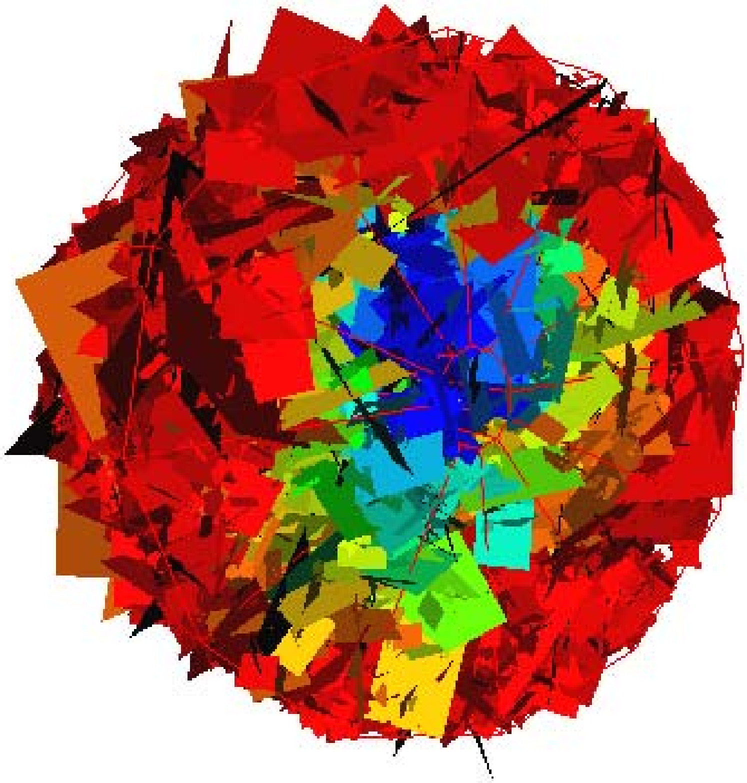


Figure 9 **The pressure distribution on the network scale. Each fracture is coloured according to the mean pressure on the plane. Red indicates high pressure, blue low pressure. The flow domain is an annular region built from 12 hexahedra. Flow is from the external surface to the internal surface to model inflow to a tunnel.**

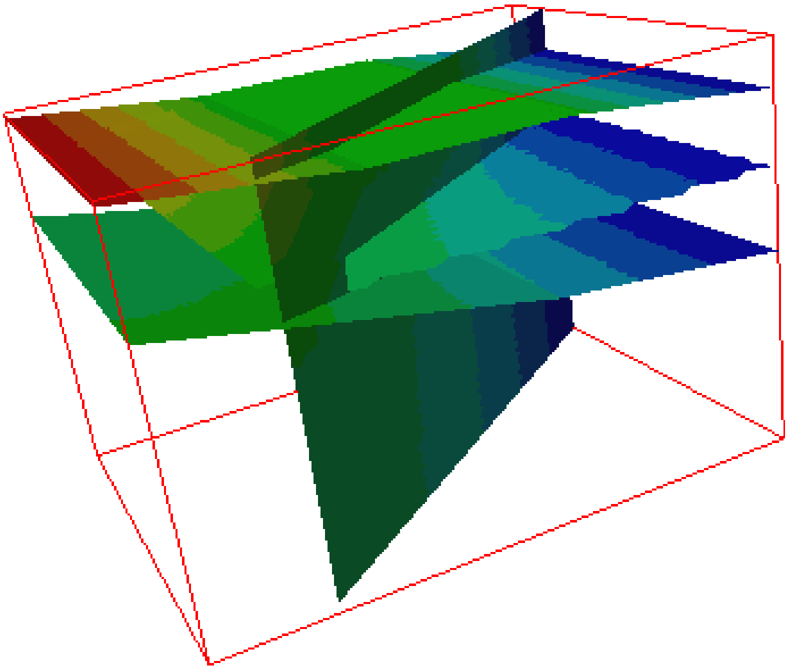


Figure 10 The pressure distribution on the scale of the finite-elements for a network of 8 fractures. Red indicates high pressure, blue low pressure.

3.4 Full Permeability Tensor

The approach adopted to the calculation of the full permeability tensor, that is, all six independent components, k_{xx}, k_{yy}, k_{zz}, k_{xy}, k_{yz}, k_{zx} is as follows. Consider a rectangular block of a continuum porous medium with an anisotropic permeability tensor with components k_{ij}. For a given head gradient with components G_i, the specific discharge would be

$$q_i = -\sum_i k_{ij} G_j \quad . \tag{3.19}$$

The fluxes through the faces of the block are given by

$$Q_\alpha = A_\alpha \sum_i n_{\alpha i} q_i = A_\alpha \sum n_{\alpha i} k_{ij} G_j \ , \tag{3.20}$$

where $n_{\alpha i}$ are the components of the normal to face α with area A_α. These fluxes vary in a simple way with the imposed head gradient.

Now consider a similar block composed of fractures. Using NAPSAC, the fluxes through the faces of the block can be calculated for a specified head gradient imposed as a boundary condition on the block. These fluxes will vary with the imposed head gradient. The variation is unlikely to be as simple as the variation of the fluxes through the faces of a block composed of an anisotropic continuum porous medium. However, one can look for the best fit to the variation of the fluxes through the block composed of fractures in terms of the variation of the fluxes through an anisotropic continuum porous medium. The corresponding permeability tensor provides, in an average sense, the effective permeability tensor for the block composed of fractures.

In practice, rather than fitting to the variation of the fluxes through the faces of the block as continuous functions of the imposed head gradient, the fit is made to the fluxes for a modest number of directions of the imposed head gradient. In the current implementation in NAPSAC, the head gradients may be specified by the user, or their directions may be chosen automatically in the following manner, which is designed to avoid directional bias, as far as possible. A number of randomly oriented regular icosahedra centred on the origin are chosen. Then the directions of the head gradients are taken to be along the lines from the centre to the mid-points of the sides of each icosahedron. (There are 15 such lines for each icosahedron.) This gives a uniform coverage of direction.
Thus the components of the effective permeability tensor are obtained by minimising

$$f = \sum_{\beta,\alpha}\left(q_{\beta\alpha} + A_\alpha \sum_{i,j} n_{\alpha i} k_{ij} G_{\beta j} \right)^2 , \qquad (3.21)$$

where $Q_{\beta\alpha}$ is the flux through face α for imposed gradient $G_{\beta\alpha}$. (Here β indexes the imposed head gradients.)

This leads to the `normal equations' (see for example Reference [12])

$$\frac{\partial f}{\partial k_{ii}} = 0 = 2\sum_{\beta,\alpha}\left(Q_{\beta\alpha} + A_\alpha \sum_{k,l} n_{\alpha k} k_{kl} G_{\beta l} \right) A_\alpha n_{\alpha i} G_{\beta i} \text{ , i=1, 2, 3;} \qquad (3.22)$$

$$\frac{\partial f}{\partial k_{ii}} = 0 = 2\sum_{\beta,\alpha}\left(Q_{\beta\alpha} + A_\alpha \sum_{k,l} n_{\alpha k} k_{kl} G_{\beta l} \right) \left(A_\alpha n_{\alpha i} G_{\beta j} + A_\alpha n_{\alpha j} G_{\beta i} \right) \text{ , } i\text{<j and } i\text{=1,2,3.} \qquad (3.23)$$

(It should be noted that only six components of the permeability tensor are independent, namely k_{xx}, k_{yy}, k_{zz}, $k_{xy} = k_{yx}$, $k_{yz} = k_{zy}$, $k_{zx} = k_{zx}$)

Equations (3.22) and (3.23), which are a system of linear equations, are solved by Gaussian elimination. The quantities $A_\alpha n_{\alpha i} G_{\beta i}$ and $(A_\alpha n_{\alpha i} G_{\beta j} + A_\alpha n_{\alpha j} G_{\beta i})$ are called the basis functions. The inverse of the matrix for the system is called the covariance matrix and is closely related to the uncertainties in the parameter estimates obtained by the least-squares fitting.

3.5 Efficient Implementation

It is important to make the effective permeability calculations as efficient as possible.
In fact, the calculation of the full permeability tensor is implemented in such a way that its computational cost is little more than that of computing the flow through the block for a single set of boundary conditions.
The reason for this is as follows. The discretised flow equations for the network can be written in matrix notation as

$$\mathbf{Ax} = \mathbf{b} , \qquad (3.24)$$

where

$\mathbf{x}$ is the vector of unknowns, that is the residual pressures (or effectively heads) at the internal nodes of the model;
$\mathbf{b}$ is the right-hand side vector of specified residual pressures (heads) on the boundary.

NAPSAC uses a direct Gaussian algorithm to solve these equations. The method is equivalent to making a decomposition of $\mathbf{A}$ into the product $\mathbf{LU}$ of a lower triangular matrix $\mathbf{L}$ and an upper triangular matrix $\mathbf{U}$ followed by successively solving

$$\mathbf{Ly} = \mathbf{b}, \qquad (3.25)$$

$$\mathbf{Ux} = \mathbf{y}. \qquad (3.26)$$

The computationally expensive step in this procedure is the determination of the LU-decomposition. Relatively speaking, equations (3.25) and (3.26) are very cheap to solve. Once the LU-decomposition has been made, it is therefore possible to solve the matrix equations for a number of different right hand sides very cheaply. This is exactly what is required to calculate the flows for the different imposed head gradients. The various gradients are defined by specifying the corresponding distribution of head around the boundary of the block. In this way, a very efficient method for calculating the full effective permeability tensor is obtained.

3.6 Calculation of Effective Permeabilities for Many Blocks

One application of this facility is the calculation of effective permeabilities in a study of upscaling. Such studies require the calculation of the distribution and correlation structure of the effective permeabilities. An option is therefore available to generate a realization of a fracture-network in a specified region, and then automatically calculate effective permeability tensors for each block in a subdivision of the region. It is then possible to analyse the statistics of the data obtained using this option in order to estimate the correlation structure of the effective permeability.

3.7 Effective Permeability of an Internal Block

The facility to calculate the permeability for many blocks provides a very powerful tool. However, the issue of boundary short-circuits needs to be considered. As discussed above, the algorithm is based on calculating the flows through the block of interest for a number of different imposed head gradients, which are specified in terms of imposed heads on the boundaries of the block. However, it is possible that there are one, or more, highly transmissive fractures within the block directly connecting an inlet face with an adjacent outlet face (see Figure 11)

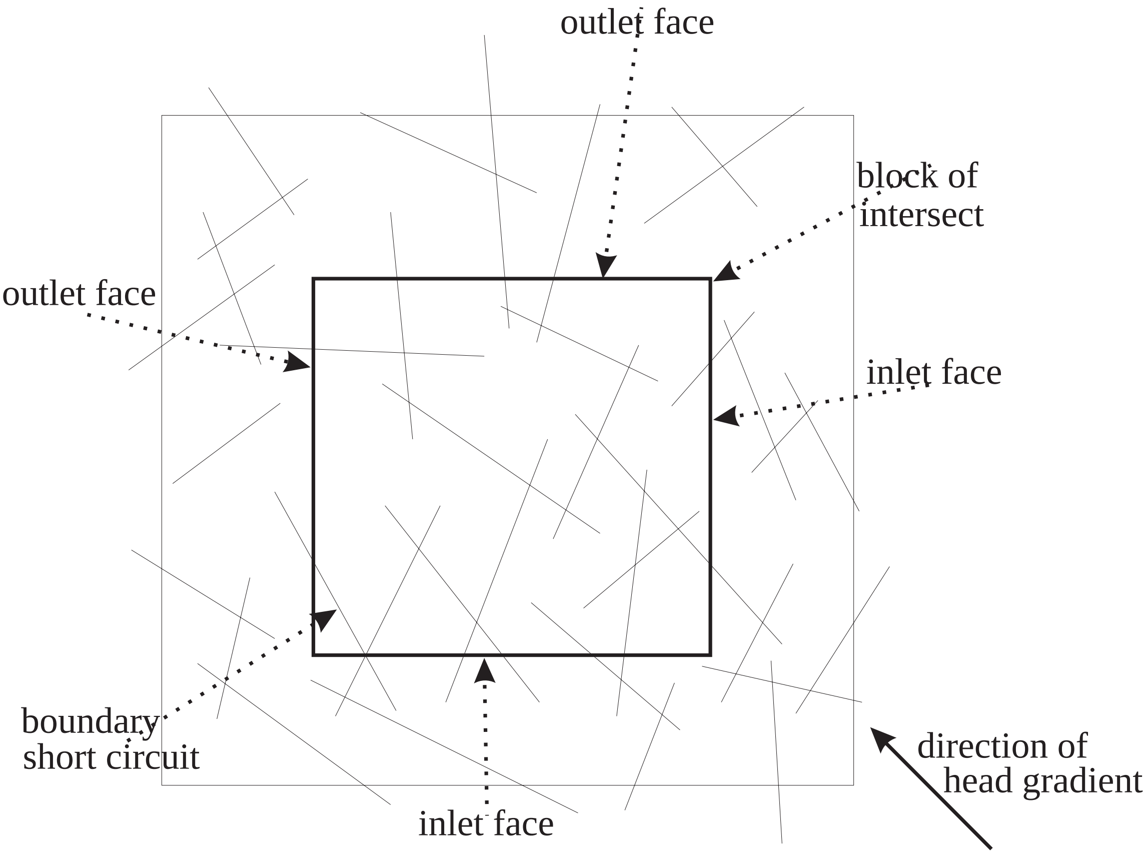


Figure 11 Schematic of short circuits in a two-dimensional case.

Because of the imposed head boundary conditions there are large flows through such fractures. This leads to an estimate of the effective permeability that is biased towards high values. The problem arises because of the interaction of such short circuits with the imposed boundary conditions. Consider the behaviour of the block in question in the context of the surrounding network. It is unlikely that the imposed head would be present across such a short circuit.

Rather, because of its high transmissivity, the head drop across the short circuit would be small, with most of the head drop being taken up by other fractures. One limiting case, which is shown in Figure 11 is that of a short circuit that is not connected beyond the block in question. In the case shown in Figure 11 when the surrounding network is taken into account the head in the short circuit would actually be constant, and there would be no flow in the short circuit, whereas if the block was considered in isolation, the short circuit might well completely dominate the flows through the block and hence the calculated effective permeability. In order to address this problem it is necessary to modify the procedure for calculation of the effective permeability tensor. This modification introduces a `guard zone' around the block of interest. This allows the effective permeability of the block to be calculated in the context of the surrounding network, so the effect of short circuits is restricted to the guard zone. In practice, it is necessary to choose a guard zone of a sensible size, which depends on the distributions of the fracture properties.

3.8 Transient Flow Modelling

As the field experiments from which the data to generate fracture-networks are derived usually involve transient flows, a transient flow modelling capability has been developed in NAPSAC. An approach consistent with the steady-state approach is adopted. This ensures that the code is applicable to the complex networks that the steady-state code is able to handle. The equation describing transient flow through a fracture-network is

$$\frac{S}{\rho g}\frac{\partial P}{\partial t} = \frac{T}{\rho g}\nabla^2 P \ , \tag{3.27}$$

where *S* is the fracture storativity which is dependent both on fluid and rock compressibility. The choice of a suitable model for fracture storativity is important for an accurate transient flow model. Three models for fracture storativity are available in NAPSAC:

$$S = \rho g A e; \tag{3.28a}$$

$$S = \rho g[1/RKN + eC_f]; \tag{3.28b}$$

$$S = \alpha T^b. \tag{3.28c}$$

Here *A, RKN, C_f, α* and *β* are constants that can be specified by the user.

The fracture-network is discretised in the way described in Subsection 3.2 and a forward difference is used to approximate the time derivative. The finite-element equation to be solved for the pressure values at the global flow nodes becomes

$$\sum_J \left(\frac{\rho g S}{\Delta t} \int_\Omega \Psi_I \Psi_J P_J^{n+1} + T \int_\Omega \nabla \Psi_I \nabla \Psi_J P_J^{n+1} \right) = \sum_J \frac{\rho g S}{\Delta t} \int_\Omega \Psi_I \Psi_J P_J^n \ , \tag{3.29}$$

This equation is solved for a fixed timestep Δt to give P^n the pressure solution at time $t = n\Delta t$. The second term on the left-hand side of this equation is simply the flux term,

$$\sum_J F_{IJ} P_J^{n+1}, \tag{3.30}$$

that appears in Equation (3.18), and the first term on the left-hand side of this equation will be referred to as the storativity term

$$\sum_J S_{IJ} P_J^{n+1}, \tag{3.31}$$

The second step of a transient groundwater flow calculation is analogous to that in the steady-state calculation. The contributions from the individual planes to the global matrix are calculated. As before, the contributions to the flux term from each plane are evaluated by solving the mass conservation equation on each plane subject to a number of different boundary conditions. In addition, the contributions to the storage term are evaluated for each fracture plane.

These contributions are then assembled into the global matrix, ready to start timestepping. In order to simplify generating the right-hand sides of equation (3.30) for each timestep, two global matrices are assembled, one containing the flux term, F_{IJ} , and the other containing the storativity term, S_{IJ} . It is assumed that the boundary conditions are fixed with respect to time, and therefore can be imposed by deleting terms from the storativity matrix and changing the flux matrix in the same way as for the steady-state model. A boundary condition vector is constructed at this point. The two global matrices are added together, component by component, to form one global matrix. The resulting global matrix and the boundary condition vector are unchanged for all timesteps of the same size, since they depend on time only through the timestep size Δt.

The third stage of the transient groundwater flow calculation is to solve the matrix equations for each timestep. The right-hand side of equation~(3.29), for the first timestep, is evaluated by multiplying the storativity matrix by an initial solution which is prescribed by the user. This initial solution may be a constant pressure, a linearly varying pressure or an existing solution previously found by NAPSAC. The contribution from the boundary conditions is added to this right-hand side and the matrix equation (3.29) is solved using the same direct frontal solver as for a steady-state calculation to produce the pressure solution at the first timestep. This solution is used to evaluate the next right-hand side and the timestepping loop is repeated producing a sequence of solutions P^{n+1} which define the flow field at the (n+1)-th timestep.

As with steady-state problems, this method allows quite coarse meshes to be used on very large systems. However more detailed refinement might be required for smaller networks, or near sources and sinks. To deal with this the transient model permits optional local refinement for significant fractures. This option involves solving the transient mass conservation equation on the fracture and then adding this contribution to the refined matrix. At each timestep the local pressure solution is saved on the finite-element mesh of the refined plane and this solution is used to calculate the next solution to the transient mass conservation equation on the refined plane.

3.9 Engineered Features

Close to a point where an engineered feature intersects a fracture, the pressure field behaves like that within a parallel plane having a single sink:

$$P = P_w \frac{\rho g Q}{2\pi T} \ln\left(\frac{r}{r_w}\right), \qquad (3.32)$$

where P_w is the wall pressure, *r* is the distance from the engineered feature, r_w is the engineered feature radius and *Q* is the volumetric flow rate from the engineered feature into the fracture.

This logarithmic behaviour is poorly approximated by a regular linear finite-element discretisation, and so a correction is applied to the pressure calculated by the finite-element method at the engineered feature, P_{PE} , using the analytical solution in equation (3.32), to obtain the wall pressure, P_w

$$P_w = P_{PE} + \frac{Q}{\Gamma} \qquad (3.33)$$

where Γ is a productivity index, which is dependent on the size of the finite element mesh, the feature radius and the transmissivity of the fracture. Although this model is based on a steady-state analytical solution, tests have shown that the correction factor to be reasonable when modelling transient flow.

3.10 Two-dimensional Networks

For formations with very high aspect ratios, a simplification to a two-dimensional network may be justified. Alternatively when approaches to complex physics are being developed it may be necessary to simplify the geometry by approximating a three-dimensional network by a two-dimensional one. A two-dimensional version of NAPSAC was developed. This uses much of the existing three-dimensional code. The flow model and the approach used to solve the equations are analogous to the three-dimensional version. The network can be considered as a slice through a three-dimensional network in the *x-z* coordinate plane, with constant flow in the *y*-direction, and mass conservation governing flow through the fracture. The fractures are represented as straight-line segments, and are defined by an orientation angle, a length and an effective aperture. As in three-dimensional simulations, the parameters describing each fracture may be sampled from statistical distributions. A more physical way of generating a two-dimensional network, and the one usually adopted, is to generate a three-dimensional fracture-network and to map traces onto a plane.

The flow solution in the two-dimensions is less complicated than in three-dimensions. The flow field is discretised by assigning one global flow node to each intersection between fractures. Linear basis functions, which are zero outside the fractures, are defined at each node *J* by

$$\Psi_J = 1 \qquad \text{at node } J, \tag{3.34a}$$

$$\Psi_J = 0 \qquad \text{at node } I \neq J, \tag{3.34b}$$

and are defined by linear interpolation along fractures between nodes. The contributions to the matrix equations for each plane are calculated directly without needing to calculate the response functions on each plane. Boundary conditions are imposed and the resulting matrix equation is solved using the direct frontal solver.

3.11 Modelling the Effect of Stress on the Fracture Network

NAPSAC is designed to deal with complex fracture-networks, and as a consequence it is only practical to incorporate relatively simple models for the effect of stress on flow (hydro-mechanical coupling). The effect on the flow of a change in stress caused by disturbing the surrounding fracture-network, for example by drilling a repository tunnel, can be calculated. The network is assumed initially to be in hydro-mechanical equilibrium, with the apertures of the fractures being those that apply to the in-situ network under the specified equilibrium stress field. NAPSAC does not calculate the stress field directly. An analytical solution may be used to determine the changed stress field, or the results of field experiments can be used to obtain an empirical specification of the changed stress field. Thus the normal stress acting at any point on a fracture may be calculated.

Having calculated the change in normal stress, a stress-aperture coupling is used to change the fracture aperture. In three-dimensions, the change in aperture for each finite-element is computed from the value of the normal stress at the centre of the finite-element.
In two-dimensions, the change in normal stress acting on the fracture is calculated at the centre point of each section of fracture between intersections.

There are several stress-aperture couplings available. Three that have been used in NAPSAC modelling [13] are

$$T = cons\tan t, \tag{3.35a}$$

$$T / T_0 = \left(\sigma_n / \sigma_{n0}\right)^{-\alpha} \tag{3.35b}$$

$$e = \max(e_0 - (\sigma_n - \sigma_{n0}) / RKN, e_{\min}). \tag{3.35c}$$

In the first coupling, the transmissivity, *T*, of each fracture is assumed to be unchanged by the excavation of the tunnel. NAPSAC directly converts transmissivities into apertures using the parallel-plate law, as defined by equation (3.3), and so the apertures remain constant. The second coupling is a compliance law relating the change in fracture transmissivity, *T*, to changes in normal stress, σ_n through a power law with exponent α . The value of α is obtained from laboratory tests carried out on the rock. For fractured rock α typically has values between~0.1 and~1.0. The third law relates the change in aperture, $e - e_0$ directly to the change in stress and the fracture normal stiffness, RKN. It is necessary to define a minimum aperture, $e_{\min}$ to ensure that all apertures remain positive, and to reflect the physical reality that fractures can only be compressed so far. Again, the results of laboratory tests are used to determine the value of RKN.

3.12 Tracer Transport

The tracer transport option in NAPSAC is designed to calculate the migration and dispersal of tracer through a discrete fracture-network. Within the groundwater, it is assumed that tracer transport is dominated by advection, so that molecular diffusion can be ignored, and the major cause of dispersion is due to the existence of a number of different paths through the fracture network. It is also assumed that the fracture apertures are small enough that the tracer diffuses quickly across the aperture.

The transport calculations are based upon a particle-tracking algorithm. The problem is split up into the calculation of single fracture responses followed by the calculation of the transport of a particle swarm through the network. For each fracture plane a representative number of pathlines between the intersections on the plane are calculated. Intersections are discretised by transport nodes and pathlines are calculated from each transport node. There are two algorithms available for calculating these pathlines, ‘exact particle tracking’ (with standard and mass-conserving methods) and ‘approximate particle tracking’. Each algorithm is described in the following sections.

3.12.1 Exact Particle Tracking (Standard Method)

For each fracture, the flow field is discretised in terms of linear triangular finite-elements. The flow is determined by the pressure field, and since the pressure varies linearly over each triangle, the groundwater velocity,

$$v = \frac{q}{e} = -\frac{e^2}{12\mu}\nabla P, \qquad (3.36)$$

is constant on each element. The pathlines are calculated on each fracture by stepping the path across the mesh, one element at a time. On reaching a fracture intersection, the path is complete. Once the pathlines from the transport nodes on each fracture plane have been calculated, the possible connections for that node are determined. A list of possible destinations, travel-times, distances and relative probabilities for a particle leaving each node are calculated. In this way, a library of paths is created for every transport node in the network. The model relies on the calculation of a very accurate flow solution. If a low accuracy solution is used, then problems with local flow sinks on fractures may occur, resulting in the loss of a significant fraction of the particle swarm.

3.12.2 Exact Particle Tracking (Mass-conserving Method)

The standard method of exact particle tracking using the finite-elements does not guarantee conservation of mass in the flows between elements. Effectively there are sinks or sources at finite-element boundaries and nodes. This can cause problems for particle tracking calculations because particles can become ‘trapped’ or ‘lost’ at the sinks. However, the mass-conserving method (introduced with NAPSAC version 9.3 as the default) does conserve mass between elements. It does this by using a method proposed by Cordes and Kinzelbach [22]. This method takes each NAPSAC triangular finite-element and sub-divides it into four sub-triangles. The velocity within each sub-triangle is then calculated in such a way as to conserve mass between elements and sub-triangles. The set of sub-triangles around each finite-element node is termed a patch. The velocity calculations are carried out patch by patch, with the calculations for each patch independent of those of the others. The velocities for the middle sub-triangles that are not part of any patch can be calculated from the finite-element pressures. This keeps the scale of the calculations small and velocities only need to be calculated for those patches or middle sub-triangles that particles enter.

Figure 12 shows the sub-triangle patch for a non-boundary, non-intersection node. The black lines are the element edges and the red lines are the sub-triangle edges. The purple arrows are the fluxes across the sub-triangle edges and the green arrows are the velocities for the sub-triangles.

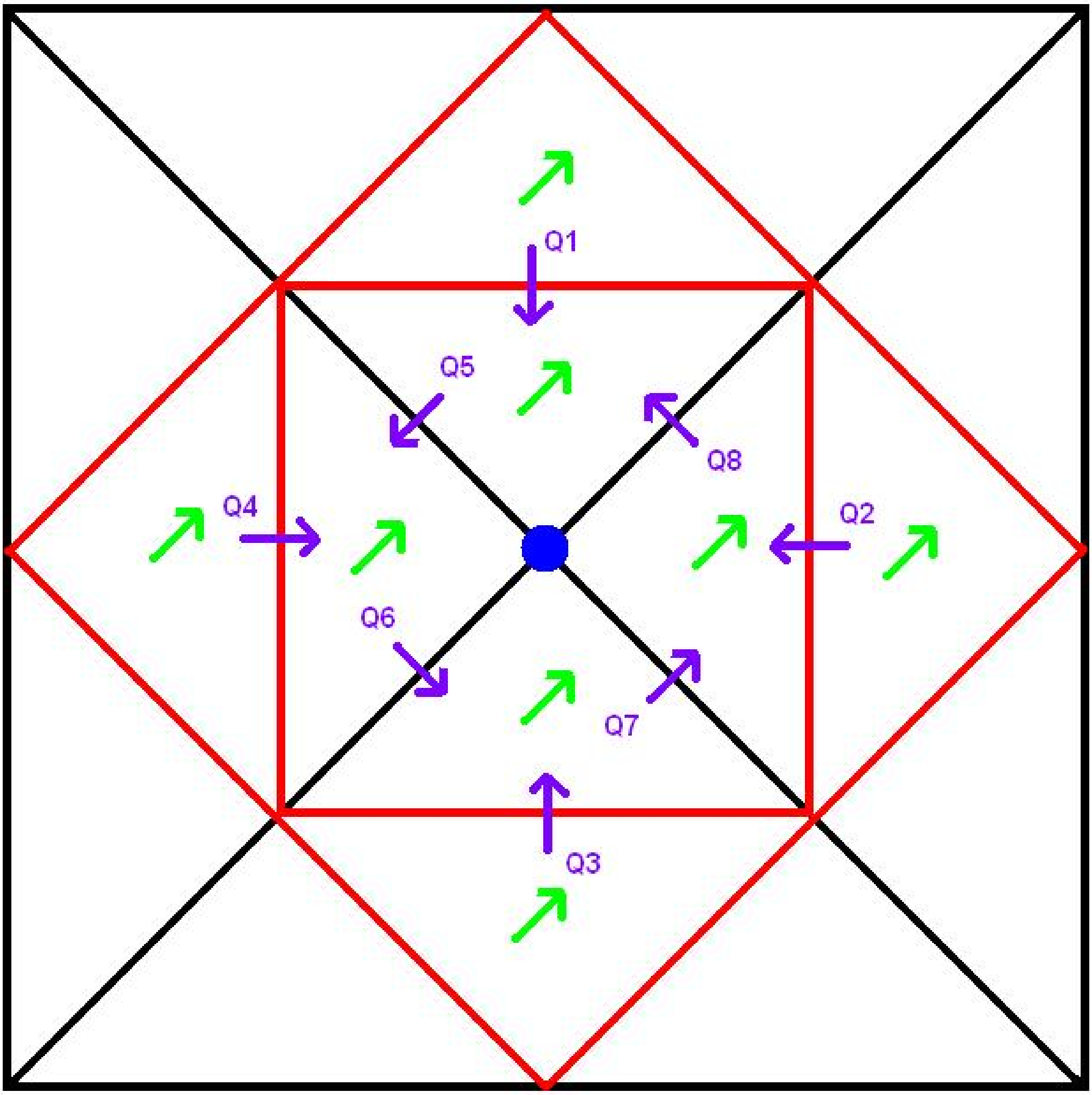


Figure 12 A sub-triangle patch for a non-boundary, non-intersection node.

The fluxes entering or leaving the patch (Q1 to Q4) can be calculated directly from the finite-element pressures. The fluxes through the edges connected to the node can be calculated by applying the following constraints:

a) There is no net flux into or out of a sub-triangle, e.g. Q5 = Q1 + Q8;

b) The fluxes entering or leaving the patch sum to zero, i.e. Q1 + Q2 + Q3 + Q4 = 0;

c) The integral of the head gradient around the node is zero (the irrotationality constraint).

The velocity in each sub-triangle can be calculated directly from the fluxes entering or leaving it. Note that for the middle sub-triangles (those not connected to an element node), the fluxes, and hence velocities, can be calculated directly from the finite-element pressures.

Figure 13 shows a boundary node on the edge of a NAPSAC fracture. There is no flow across the boundary and so fluxes Q5 and Q9 will be zero. Fluxes Q1 to Q4 can be calculated from the finite-element pressures. Since flux is preserved between sub-triangles, this enables fluxes Q6 to Q8 to be calculated and hence the sub-triangle velocities.

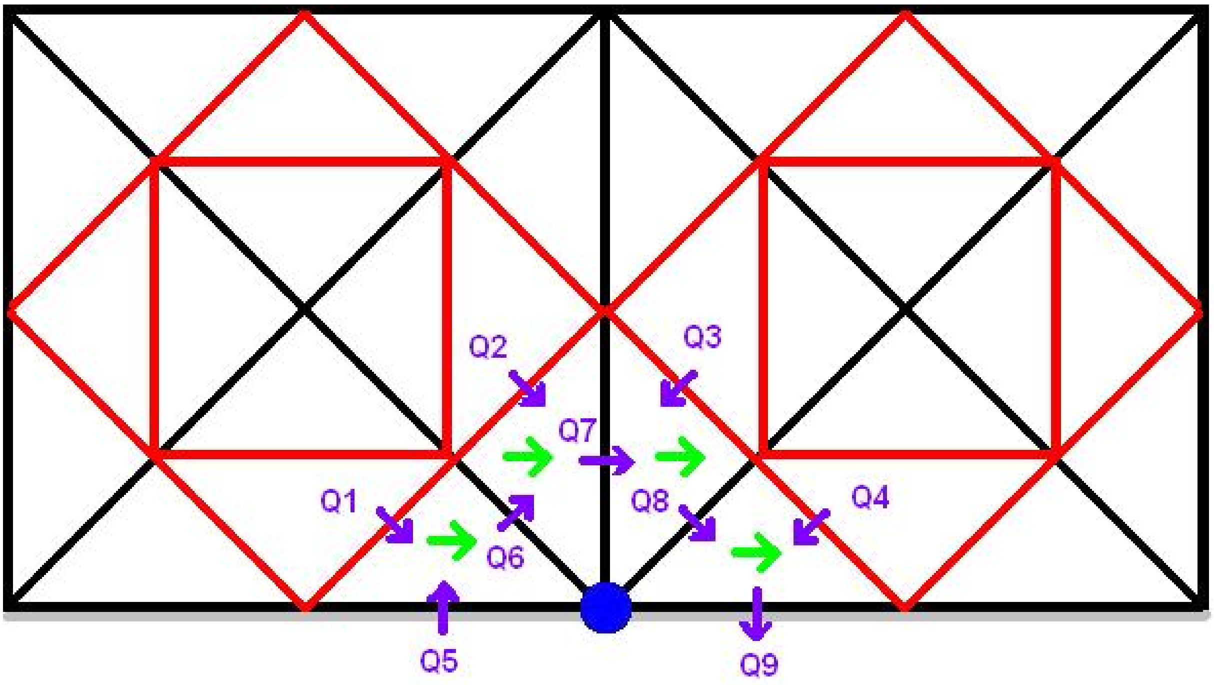


Figure 13 A sub-triangle patch for a boundary node.

Figure 14 shows a node on a fracture intersection, with the intersection edges highlighted in blue.

Because flow can be entering or leaving the fracture through the intersection then constraints b and c above are no longer true. In this case, the patch is split into segments, with each segment being the sub-triangles between a pair of intersection edges. Then each segment is treated separately since the fluxes on each side of the intersection are not necessarily equal, e.g. Q7 may not equal Q8. For each segment, the sum of the fluxes entering or leaving the patch through sub-triangle edges, e.g. Q1 + Q4, must equal the sum of the fluxes entering or leaving the intersection, e.g. Q5 + Q7. The fluxes entering the patch through the sub-triangle edges can be calculated from the finite-element pressures and it is possible to assign this flux to the intersection edges. This enables the remaining fluxes, and hence the velocities, to be calculated. A similar approach is taken for very short, i.e. point, intersections and for boreholes, but in this case the flux is assigned to the node itself and the radial flow component is taken into account.

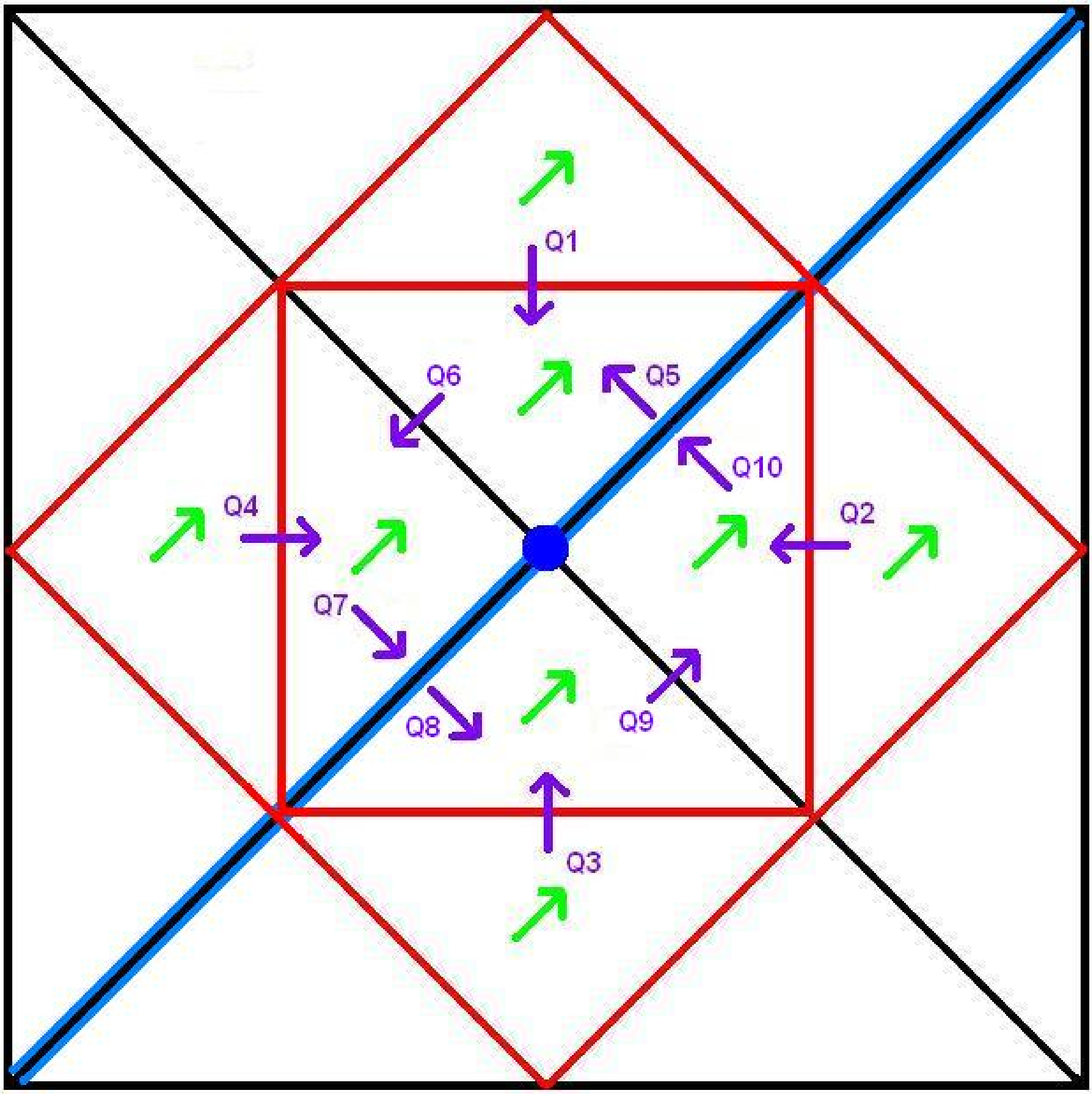


Figure 14 A sub-triangle patch for an intersection node.

Once the velocities have been calculated, the particle paths between the transport nodes can be calculated using a stepping method, where each step is between a sub-triangle edge. In the case of a point intersection or borehole node, a radial flow component is present in the sub-triangles around the node, which requires a time-stepping approach within those sub-triangles. Information on the paths is stored in the same way as for the standard method.

3.12.3 Approximate particle tracking

NAPSAC is able to create a database that records the net flux between all the intersections for a flow solution. This network of flux connections links the centre of every intersection on a given fracture with every other intersection centre on the fracture. A transport option has been developed that is based on this flux database in which particles are transported between intersections. This is a more robust method than exact particle tracking as it does not need a highly accurate flow solution, just a good flow balance at the network intersections. It is also more computationally efficient. One disadvantage is that this model cannot accurately model

dispersion on a single fracture. However, where dispersion is dominated by the different paths
th
m
ca
e
a
h

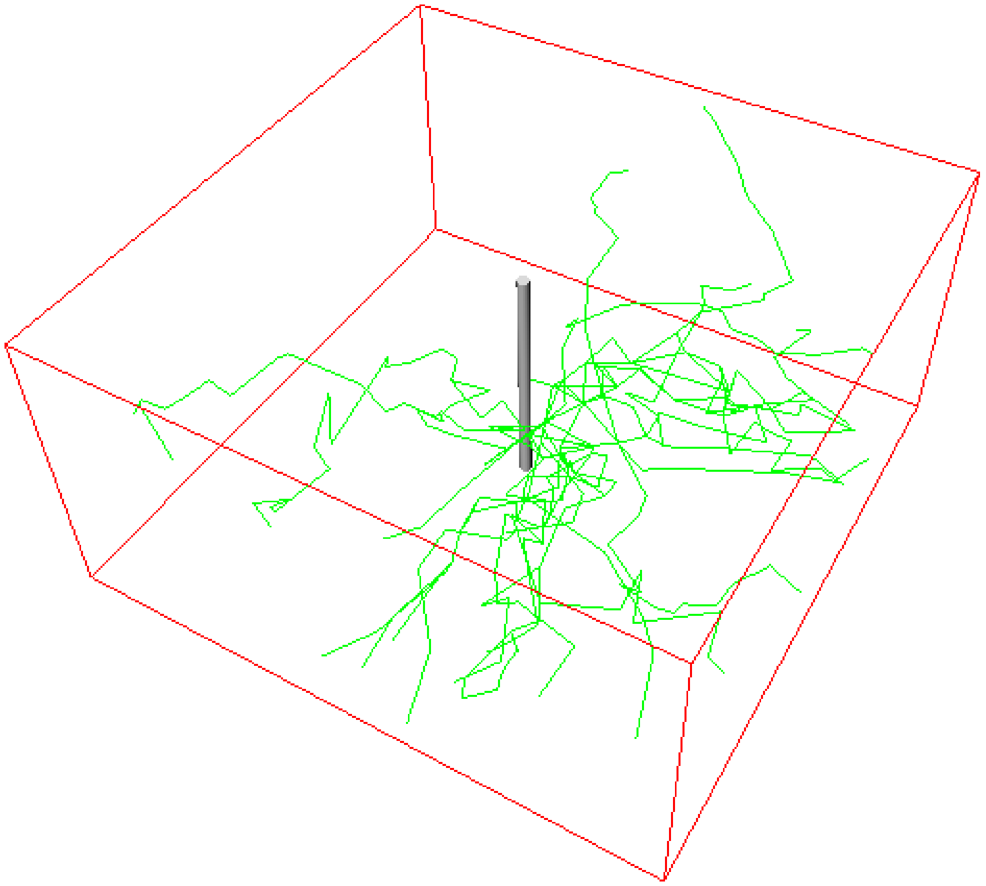


Figure 15 An example of a particle tracking calculation for 30 particles starting at the same injection borehole interval (located centrally and vertical) and dispersing outward toward the vertical boundaries. Flow is radially outward from the injection borehole.

The next step in the transport calculation consists of following a large swarm of particles across the network. Particles can be started on any surface of the fracture-network region where there is an inflow or from any number of engineered features. Particles are tracked through the network from node to node, building up the path taken by each of the particles using the information calculated in the first step. Figure 16 is an example of a breakthrough curve for particle tracking across a simple cuboid region.

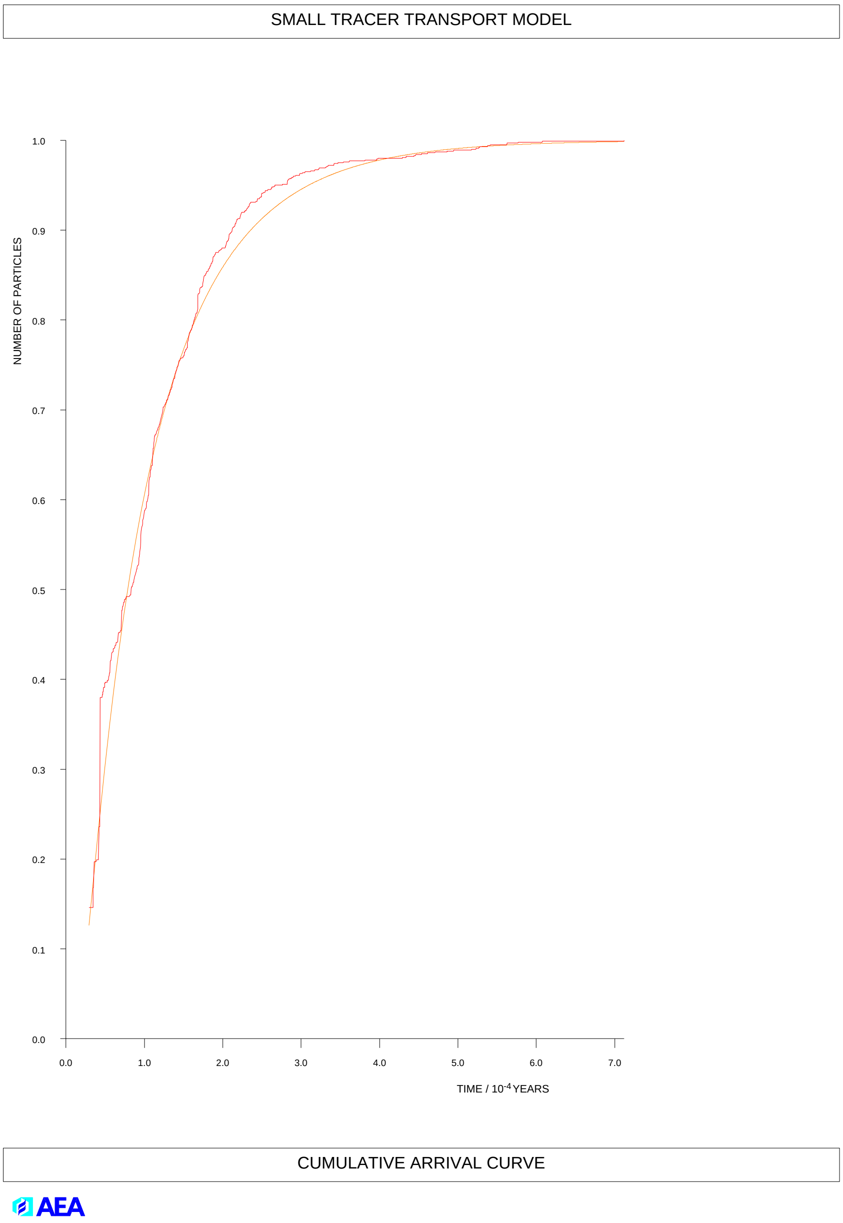


Figure 16 An example of a breakthrough curve based on 1000 particle tracks through a small random network. The jagged line is the cumulative curve based on the particle tracks. The smooth curve is a fit to the data based on a two-parameter advection-dispersion curve.

3.13 Coupled Groundwater Flow and Salt Transport

The philosophy of decomposing a problem into separate calculations on each fracture plane has also been taken in developing the variable density flow algorithm for NAPSAC. The overall structure of the approach involves the following four stages:

1. calculate a steady-state solution for constant density flow in a three-dimensional fracture-network in the normal way;
2. define an equivalent network of one-dimensional conduits (or pipes) connecting fracture intersections. The physical parameters associated with these conduits are derived on the basis of the full network solution (stage (1));
3. for each time-step of the variable density flow calculation:

    - solve for the pressures at the ends of the conduits, based on the current salinity distribution;
    - update the salinity distribution by moving the salt distribution in each conduit on the basis of the velocity field derived from the computed pressures and the current density distribution;

4. after the final time-step, interpolate the pressure and salinity distribution back on to the full fracture-network.

Subsection 3.9.1 describes how the flow across each fracture plane is converted into an equivalent set of one-dimensional conduits connecting intersections (stage (2)). The coupled flow and salinity transport equations are solved by a simple time-stepping approach in which the flow and transport equations are solved alternately [14]. This de-coupling of the flow and salinity transport on the level of each time-step is necessary if calculations are to be performed for realistic networks. Although the approach taken in some continuum codes, such as NAMMU [15], in which the equations are solved as a fully coupled system, is more accurate, it would not be tractable for large fracture-networks.

The advantages of the approach used here are that it is efficient, robust and accurate, providing parameters such as time-step size are chosen sensibly. The approach needs to be robust because large property contrasts and a complicated heterogeneous flow geometry are inevitable when simulating a fracture-network by stochastic methods. Subsections 3.9.2 and 3.9.3 describe the flow and salinity transport calculations in the solution algorithm (stage (3)). Subsection 3.9.4 discusses the applicability of the algorithm and its limitations. Stage (4) will be implemented in a later development.

3.14 Simplification to a Network of One-Dimensional Conduits

The approach taken to solving the variable density flow problem is to represent the network of planar fractures by an equivalent network of conduits, where flow within each conduit is one-dimensional. This is also the approach taken in one of the tracer transport options within NAPSAC, which has been shown to give very similar results to more detailed calculations based on the solution of finite-element equations for the two-dimensional flow field on each fracture plane [16,17]. Indeed, one may regard conventional approaches to the discretisation of continuum equations as essentially assigning properties to connections between a number of discrete nodes. Thus the important step is to ensure that the choice of connections and properties can give an accurate description of the system.

In the NAPSAC model of saline groundwater flow, the connections are defined between nodes centred on the intersections of the fracture-network. At the start of the salinity transport calculation, a list of all possible pairs of nodes on each fracture plane is assembled. This will be a very large list for highly connected networks. However, in networks where there are a large number of intersections on each plane, there will be many pairs of intersections that are not hydraulically connected; for example if there are intervening intersections that must be crossed (see Figure 17). In order to avoid defining conduits along which there is little or no flow, the flow on the fracture is considered when assigning properties to the conduits as well as the fracture apertures and intersection geometry. Four principal properties are defined for each conduit linking the nodes *i* and *j* on a given fracture plane. These are:

L_{ij} - the conduit length [m], defined as the distance between the nodes;

Ω_{ij} - is the volumetric flux between fracture intersections *i* and *j* [m^3s^{-1}]as calculated on the full fracture-network;

A_{ij} - is the area for the conduit between intersections *i* and *j* [m^2]. This area is a function of the geometrical area of the quadrilateral formed by the end points of the two intersections, Π, and the flux between the intersections Ω_{ij} ;

t_{ij} - is the average time [s] for the flowing pore water to travel between the intersections in the constant density calculation.

The coefficients in the equations describing transient flow, transport of salinity, and mass conservation in the conduit network are derived from these four quantities.

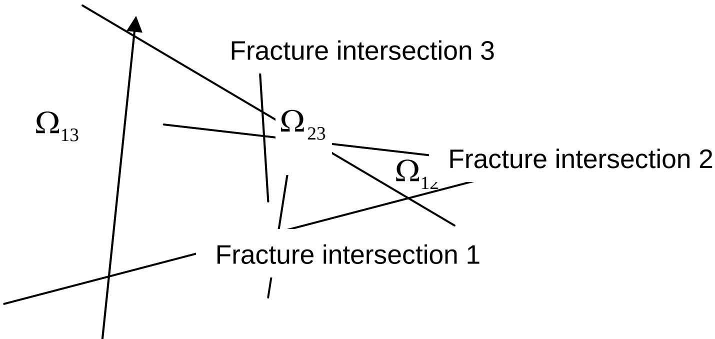


Figure 17 A schematic of flows associated with three intersections on a fracture plane.

The conduit area, A_{ij}, is straightforward to evaluate and, when there are no intervening intersections, is a meaningful flowing area for the conduit. However, the conduit areas will not (in general) sum to the area of the fracture plane. For instance, where there are a number of intersections on the fracture plane, there may be flow between each pair of intersections, and the areas formed by pairs of intersections will often have a large overlap. This situation is illustrated in Figure 17. The area available for flow between intersections 1 and 3 will be relatively much smaller than the geometrical area between these intersections. This is reflected in the groundwater flux between these two intersections, Ω_{13}. Ω_{13}. will correspond to the flux that bypasses the intervening intersection 2, and thus has to flow over the outer area of the fracture plane. Taking into account the fluxes leads to a better approximation of the flowing area of the conduit that connects intersections 1 and 3. The conduit areas are set to the quadrilateral area between the two intersections, weighted by the flux between the intersections on the fracture plane, and then scaled to sum to the fracture area:

$$A_{ij} = \frac{\Pi_{ij}\Omega_{ij}}{\sum_{ij}\Pi_{ij}\Omega_{ij}} \times \text{fracture plane area.} \qquad (3.37)$$

This definition of the conduit area means that the conduit network provides a good approximate description of flow on a fracture plane. This is because the properties of the conduits defined above reflect the partition of the flux through the fracture plane into separate fluxes between

pairs of intersections: the conduits corresponding to the connections between large A_{ij} intersections will be assigned a greater proportion of the fracture transmissivity. Local errors are not important in a stochastic simulation, so long as they do not consistently bias the solution. This approach has been tested for tracer transport calculations, where the results were found to be very close to those from more direct methods (which are much more computationally intensive) [16].
The pore water travel time, t_{ij}, along the conduit is defined as

$$t_{ij} = A_{ij} \times \frac{e}{\Omega_{ij}}, \tag{3.38}$$

where *e* is the fracture aperture [m]. The time t_{ij} corresponds to the conduit volume divided by the flux, or equivalently the conduit length divided by the mean pore water velocity. This parameter is only used to define the relationship between travel time along conduit and the mean pore water velocity. In a variable-density flow evolving in time, the pore-water velocity and hence the transit time between intersections will change. However, these two quantities are linked by a linear relationship dependent on geometric factors that are assumed to remain constant. Any transient variations in the fracture geometry can be modelled through the fracture storativity, as discussed in Subection 3.9.1. Hence, at any instant in time, t_{ij}, is used to determine the appropriate travel time along a conduit given the groundwater flux in the conduit.

A number of other dependent properties are calculated for each conduit. First, the conduit flow-width, w_{ij}, [m] is evaluated from the conduit length and flowing area:

$$w_{ij} = \frac{A_{ij}}{L_{ij}}. \tag{3.39}$$

Next, the conduit transmissivity, T_{ij}, [m^{2}s^{-1}] is calculated from the constant density flow solution flux between the intersections, the pressures at the intersection nodes, and the conduit

$$T_{ij} = \rho g \Omega_{ij} \frac{L_{ij}}{w_{ij} \delta P_{ij}}. \tag{3.40}$$

Here δP_{ij} is the change in pressure along the conduit [kgm^{-1}s^{-2}]. Finally, to ensure consistency, the conduit flow aperture, e_{ij}, is recalculated from the conduit flow-width and the transmissivity using the parallel-plate law.

In summary, the fracture-network solution for constant density flow has been used to define an equivalent set of flow conduits characterised by connectivity, length, area, and transmissivity. Although these properties depend most strongly on the fracture properties and network geometry, they also depend on the flow solution. A future development of the algorithm will allow the conduit network to be updated based upon a variable density flow solution. This planned refinement to the algorithm is expected to result in only minor changes to calculated results.

3.15 Variable Density Flow Solution

The solution of the distribution of pressure and salinity at each time-step is divided into two stages. Firstly, the transient pressure field in the conduit network is calculated given the current variable distribution of density. Pressure is calculated at a set of nodes located at either end of the conduits.

Then the new salinity distribution is calculated given the updated flow solution. Salinity is calculated in a finite number of cells which discretise the distribution of salinity in each conduit. Hence, in terms of the fracture-network, pressure is calculated at each fracture intersection, and the distribution of salinity is calculated along flow connections joining pairs of intersections on the same fracture plane. The distribution of density is then inferred from the salinity distribution to be used in setting the equation for pressure at the next time-step. This procedure is repeated at each time-step. A more detailed description is given below.

The mass conservation equation for flow in a continuum is

$$\frac{\partial}{\partial t}(\rho\phi) = -\nabla.(\rho\mathbf{q}) , \tag{3.41}$$

where

$$q = \frac{k}{\mu}(\nabla P - \rho\mathbf{g}) , \tag{3.42}$$

ρ is the fluid density [kgm^{-3}], ϕ is the total porosity, **q** is the specific discharge [ms^{-1}], **k** is the permeability [m^2], and **g** is the acceleration due to gravity [ms^{-2}].

When the corresponding equations are discretised for the conduit network, the dependent variable is the residual pressure defined at the nodes corresponding to the junctions of the conduit network (that is, at the centres of the fracture intersections of the fracture-network). In order to discretise equations (3.41) and (3.42) for the node *i* one must consider flow and transients in the set of conduits *j*. Integrating over the conduits *j*, gives

$$V_i \frac{\partial}{\partial t}(\rho\phi) = \sum_j \rho q_{ij} , \tag{3.43}$$

where V_i is the volume [m^3] of the sub-domain associated with node *i*. V_i is the sum of the flowing volume (the product of the conduit area and the aperture $A_{ij}e_{ij}$) for each conduit adjoining the node. q_{ij} is the volumetric flux through the conduit *j* and is determined from equation (3.42). Therefore, it depends on the density, which can vary along the conduit. If the flux is constant along the conduit (that is the change in storage, defined by the time derivative term, is considered to be localised at the node) then we can integrate the flux along the conduit to give q_{ij} in terms of the residual pressures at the nodes *i* and *j* (the node at the other end of conduit *j*) only

$$q_{ij}\int_0^{L_{ij}} \frac{\mu(s)}{k_{ij}} ds = -A_{ij}\left\{P_j - P_i - g.n\int_0^{L_{ij}} (\rho(s) - \rho_0) ds\right\} \tag{3.44}$$

where the distance along the conduit is measured by *s* [m] and P_i is the residual pressure at node *i* in the discretised system.

Since the distribution of salinity, and hence density and viscosity, is discretised into a number of cells, the integrals become sums of the appropriate quantities in each cell. In equation (3.44) the conduit permeability parameter k_{ij} is calculated from the transmissivity of the conduit T_{ij} by $k_{ij} = T_{ij} / \rho g$. The permeability is used in preference to the transmissivity since it is a property of the rock, whereas the transmissivity involves the fluid properties. The contribution to the rock permeability due to a fracture is related to the aperture of the conduit, e_{ij} by $k_{ij} = e_{ij} /12$ [11]. The code has been developed to allow the viscosity $\mu(s)$ to depend upon the distance along the conduit and thereby on the variation in salinity of the fluid within the conduit. However, currently $\mu(s)$ can only be a constant. q_{ij} is thus given by

$$q_{ij} = -A_{ij}(P_j - P_i - \Gamma_{ij}) / M_{ij}, \tag{3.45}$$

where Γ_{ij} and M_{ij} are the gravity and viscosity integrals of equation (3.44).
The left hand side of equation (3.28) is expanded using the chain rule

$$\frac{\partial(\rho\phi)}{\partial t} = \frac{\partial(\rho\phi)}{\partial P}\frac{\partial P}{\partial t} + \phi\frac{\partial \rho}{\partial C}\frac{\partial C}{\partial t}, \tag{3.46}$$

where *C* is the dimensionless relative concentration of saline water. *C*=1 corresponds to saturated brine of density ρ_s and *C*=0 corresponds to fresh water of density ρ_0. Both a power law and a mass fraction constitutive relationship for density have been implemented power law:

$$\rho(C) = \rho_0 \left(\frac{\rho_s}{\rho_0} \right)^C \tag{3.47}$$

mass fraction:

$$\rho(C) = \left(\frac{1-C}{\rho_0} + \frac{C}{\rho_s} \right)^{-1} \tag{3.48}$$

Replacing the pressure derivative with the conventional storage term from NAPSAC, we can write equation (3.46) as

$$\frac{\partial(\rho\phi)}{\partial t} = \frac{S}{ge} + \phi\frac{\partial \rho}{\partial C}\frac{\partial C}{\partial t}. \tag{3.49}$$

Here, *S* is the dimensionless fracture storativity. Typically an empirical model is used to relate fracture storativity to transmissivity, *S=S(T).* The discrete form of equation (3.39) integrated over the volume associated with node *i* is

$$V_i \frac{\partial(\rho\phi)_i}{\partial t} = \sum_j \left(\frac{A_{ij}S_{ij}}{g}\frac{\partial P_j}{\partial t} + w_{ij}e_{ij}\int_0^{L_{ij}} \frac{\partial \rho(s)}{\partial C}\frac{\partial C(s)}{\partial t} ds \right). \tag{3.50}$$

Equations (3.45) and (3.50) can now be substituted into (3.43) to give the spatially discretised equations:

$$\sum_j \left(\frac{A_{ij}S_{ij}}{g}\frac{\partial P_j}{\partial t} + w_{ij}e_{ij}\Lambda_{ij} \right) = -\sum_j \rho_{ij}A_{ij}(P_j - P_i - \Gamma_{ij}) / M_{ij}. \tag{3.51}$$

Here, ρ_{ij} is the arithmetic mean density along each conduit, and Λ_{ij} is the right-hand integral term in equation (3.44).

The time discretisation remains to be addressed. In principle, either an implicit or an explicit scheme could be used. However, an implicit scheme would involve fully coupling the solution to the pressure field and the concentration field; this would require an iterative calculation at each time-step and would be computationally very expensive. To avoid this, the concentration and density terms are treated explicitly, and the degree of implicitness for the pressure terms is set by the user. The derivative of *C* in the cells along each conduit is evaluated during the concentration solve using a backwards-difference approximation to the time derivative, and assumes full mixing of water in each cell (the evaluation of concentration in each cell is described in more detail in Subsection 3.9.3).

The time discretisation is then standard, and leads to a set of linear equations for the pressure values at the nodes of the conduit network.

The use of concentration and density values from the previous time-step leads to some error in the solution. However, if the time-step is sufficiently small to capture the evolution of concentration on each conduit, then this error should be acceptable. The set of linear equations for the pressure values at the nodes of the conduit network is solved using a direct frontal matrix solver. The Harwell Subroutine Library (HSL) bandwidth reduction program MC43 [18] is used to minimise the front width using the Sloan algorithm [19]. The equations are then solved using the HSL matrix inversion program MA42 [18], which uses a highly optimised implementation of the frontal method [20].

3.16 Advective Transport of Salinity

Once the distribution of pressure has been calculated for a time-step, salt is advected along the conduits in accordance with the resulting flows. Values of the concentration of salinity are stored at a number of cells along each conduit. Typically 10-30 cells are used to discretise each conduit, but this number can be chosen by the user to give the appropriate level of refinement.

The pore water velocity along each conduit is evaluated using the volumetric flux given in equation (3.32), together with the conduit flowing width and aperture. This velocity is used to determine the number of cells that salinity in each cell should be moved along each conduit during a time-step. In general, this is not an exact number of cells, and so the salinity (assumed to be distributed uniformly within a given cell) is advected into the two cells that overlap the finishing location. The salinity variable is stored as a single value for each cell, and thus this approach leads to some numerical dispersion (in common with most numerical algorithms for this problem). Where a front is being advected through a conduit, the front is spread by up to one cell at each time-step.

If the salinity would be advected beyond the end of the conduit, then it is assigned to a temporary array associated with the links downstream of the end node of the conduit.

This salinity is divided amongst the out-flowing conduits, in proportion to flux, so as to correspond to perfect mixing at the node. A second pass of the algorithm through the conduits takes the salinity stored in the temporary array, and advects it into the new conduits. If either the flow is large or the conduits are short, then some of the salinity may cross several conduits during a single time-step, and several passes may be required to advect all the salt. For a good choice of time-step only a few passes will be required, and most conduits will have their salinity advected to its final location in just two passes.

The new values of concentration of salinity are now stored for each cell, ready to evaluate the Γ_{ij} , M_{ij} , Λ_{ij} integrals of equations (3.32) and (3.37).

The algorithm has been designed so that numerical dispersion and other discretisation errors can be arbitrarily reduced by increasing the refinement (subject to the limits of computer performance).

The final stage of the algorithm for each time-step is to evaluate the concentration and the time derivative of concentration at each node ready for the pressure solve at the next time-step.

3.17 Discussion

The algorithm has been designed to allow the user to refine the numerical discretisation both in time and in space. However, there are some approximations inherent in the approach.

The main approximation is the decoupling of the pressure and salt equations that occurs by choosing an explicit representation of concentration and density terms. This is unavoidable given the requirements for a robust and efficient algorithm.

The algorithm assumes that the time derivative terms are constant over the sub-domains associated with each node, and that the discharge is constant along the connections. This issue essentially relates to the discretisation of the network into conduits. The effects of this approximation are expected to be most significant when the spacing between fracture intersections is large compared to the thickness of the saline transition zone, and hence will be problem specific. The significance of this approximation could be quantified by sub-dividing each fracture into a number of sub-fractures with the same geometry and properties.

The equations solved are for the advective transport of saline water through the fracture network using Darcy's law. This means that a number of physical processes are not represented. First, dispersion within the conduits is neglected. This corresponds to neglecting dispersion on individual fractures.

Secondly, the use of Darcy's law neglects some of the terms from the Navier-Stokes equation. In particular, buoyancy driven flows within each conduit are approximated by the one-dimensional treatment, and advective transport of salinity takes place only in the flowing porosity of fractures. Any interaction with groundwater in the non-flowing porosity of fractures or dead-end fractures is neglected. Such interactions will take place through flows of buoyant water in the non-flowing porosity or by the diffusion of salinity into dead ends. Although these effects will not alter the final steady-state pressure and salinity distributions, they will act to retard the transient evolution of variable density flows. Hence, neglecting the non-flowing fracture porosity tends to over-predict the rate at which transients develop. Similarly, neglecting the retardation of transport due to diffusion into the rock matrix can be expected to give an over-prediction of transport rates.

The algorithm was designed to model advection of salt through a fracture-network, and to investigate the impact of network geometry on this transport process. However, the algorithm can be straightforwardly extended to incorporate dispersion and diffusion within the conduits. There are also a number of possible approaches that can be used to approximate the effects of rock-matrix diffusion.

One of the simplest is to include a total porosity on each plane to model retardation by diffusion into the matrix. The total porosity could be defined as an arbitrary function of fracture properties and geometry. Such an approach will be reasonable for predicting the retardation of transport as salt diffuses into a matrix containing freshwater. However, it will perform less well at modelling processes such as salt saturation of the matrix or out-diffusion from the matrix. To model these processes it is necessary to use an algorithm that calculates the distribution of salinity in the matrix as well as in the fractures.

The algorithm calculates the variable density flow solution and predicts the distribution of salinity. If this flow solution is to be used for further calculations by NAPSAC, for example, tracer transport, then the results will need to be interpolated back from the conduit network onto the full fracture-network. The approach to this back interpolation is also likely to introduce some numerical dispersion.

Finally, the definition of the equivalent conduit network is based on flux weighted areas calculated from the steady-state, constant density flow solution in the full fracture network. As salt moves through the fracture-network, the flow directions within fractures will change and the weighting scheme will be less relevant. The weighting scheme is not unique, and was selected empirically for use in a tracer transport algorithm available within NAPSAC [16].

Figure 18 shows a network of 626 fractures used in demonstration of a variable density flow calculation. The boundary conditions are a specified salinity and pressure on the base, and no-flow on the vertical sides. The pressure on the base is chosen so that at steady-state the box is half full of salt water. Two cases were considered: zero variation in the fracture transmissivity (see Figure 16); high variation (variability over about 6 orders of magnitude) in the fracture transmissivity (see Figure 17). The figures show contours of salinity by colouring the cells along each conduit. Red is a high salinity, light blue is a low salinity. Although there is the same total amount of salinity in the domain for both cases, there is much enhanced mixing of salinity due to dispersion in the high variability case.

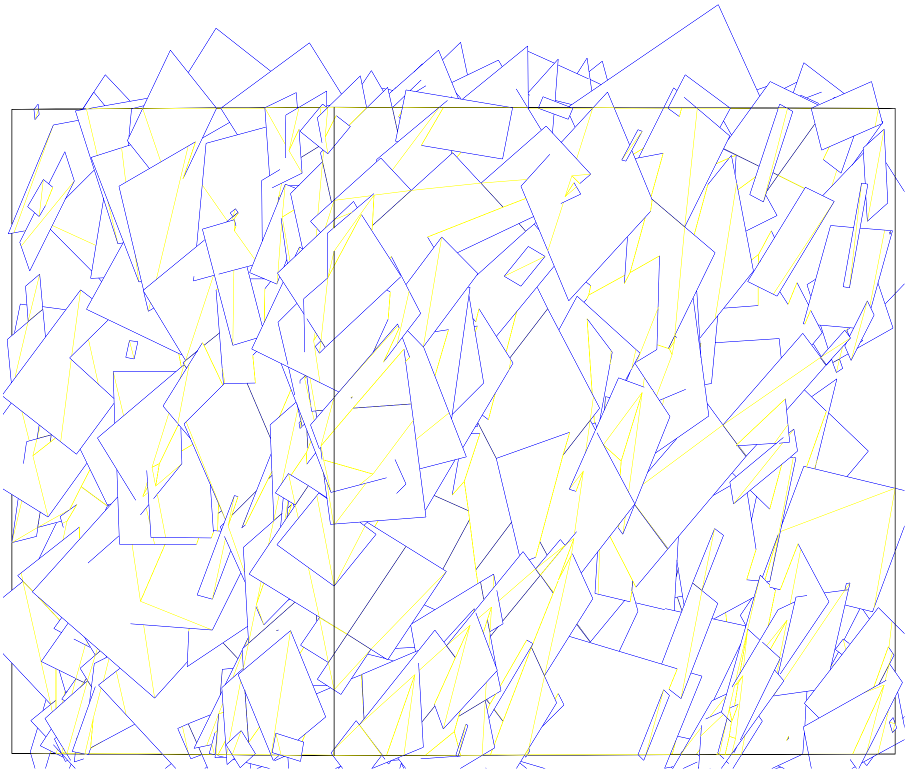


Figure 18 Random network used in the variable density flow.

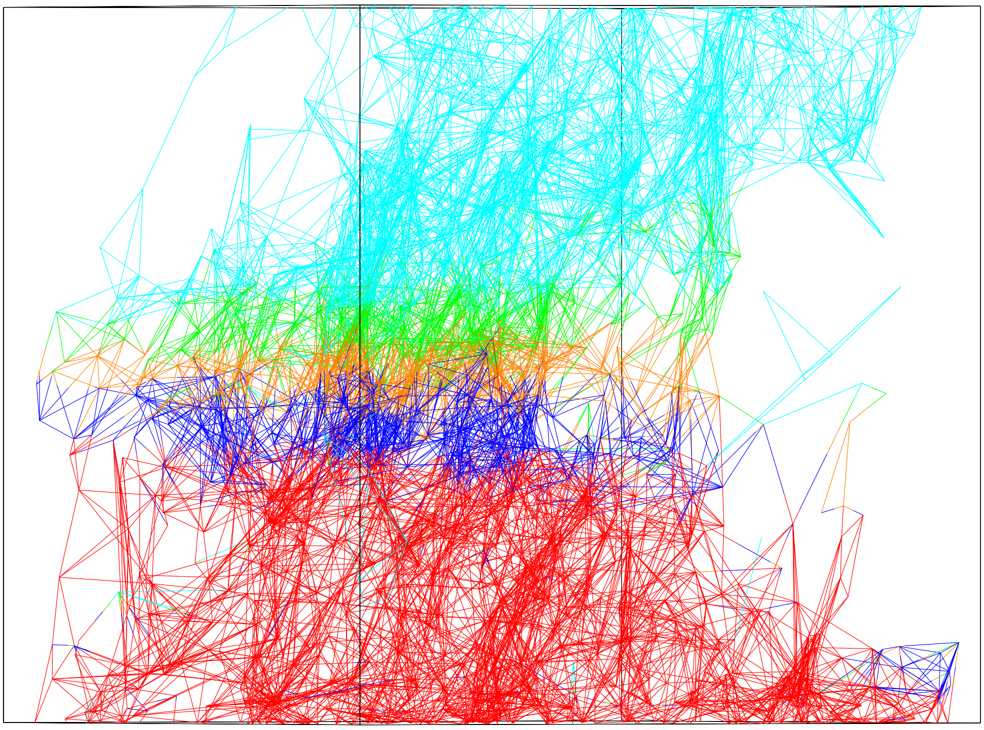


Figure 19 Profile of salinity for the case with no variability in fracture transmissivity.

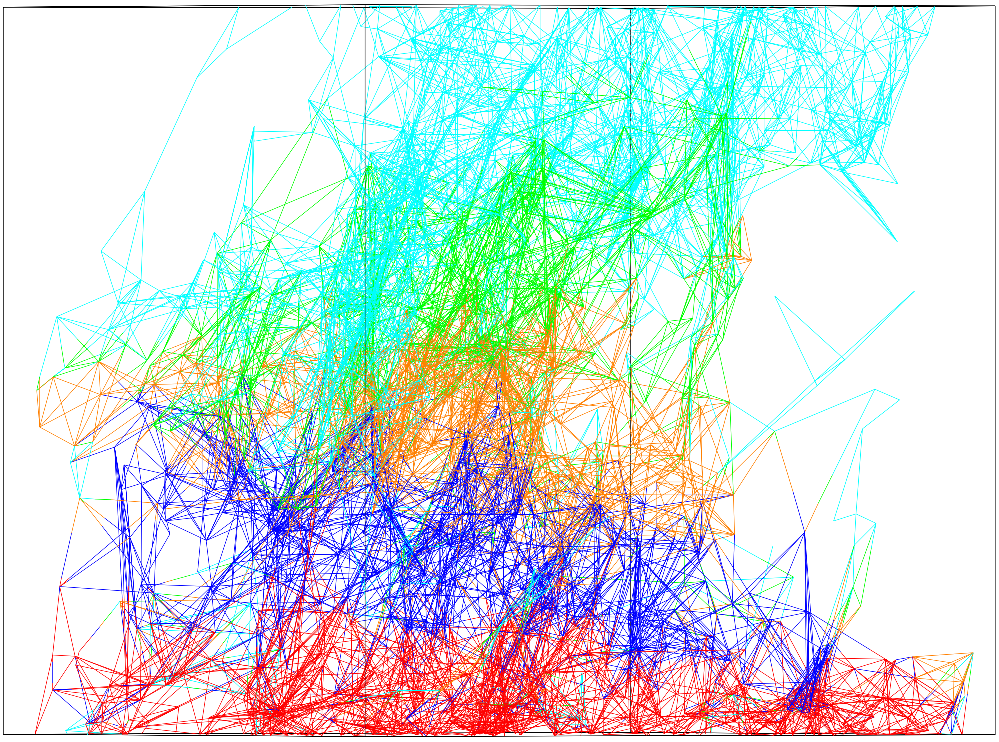


Figure 20 Profile of salinity for the case with high variability in fracture transmissivity.

4 NAPSAC Output

The problems modelled by NAPSAC can be highly complex, so it is useful to have a variety of ways of displaying output from the model, in order to facilitate the interpretation of results.

4.1 Standard Output File

During the model generation and calculation phases of a NAPSAC run, text based summary output is written to the ASCII standard output file (.out file). This contains information such as a summary of the statistical properties of the fractures generated, groundwater fluxes through surfaces, pressures and fluxes to engineered features. Some output options generate further text based output, such as the option to sample fractures along hypothetical cores.

4.2 Graphical Output

Most output options produce graphical output via an internal graphics package that creates flat plot (2D) images in PostScript format.

Some options draw perspective diagrams illustrating views of the network in three dimensions. The user has complete flexibility to specify the point from which the network is observed, but may instead choose from a set of standard view options. For clarity, lines hidden from the observation point by other parts of the network are not shown in the picture.

4.3 Inspecting the Network

There are several options that allow detailed inspection of the fracture-network, including one that simply draws a perspective diagram of either the whole system or a specified subset of fractures. Another option (`trace mapping') can be used to examine any (plane) cross-section through the network. As well as producing a diagram showing where fractures intersect the cross-section, it can scan along selected lines in the plane, reporting information on all the intersections encountered.

This information consists of the distance from the start of the scan line, and the angle at which the fracture was crossed, and the length of the intersection of the fracture with the cross-section plane.

There is also a facility for probing the network with line segments similar to boreholes. The `core logging' option produces both geometrical and hydraulic information for all fractures crossed by the line segment. This includes the distance from the start of the segment, the dip direction and dip angle of the fracture, the angle relative to the core, the fracture set number, the aperture and transmissivity of the fracture.

Another option is useful when simulating the local variation of the aperture over individual fractures. This produces maps of the aperture width on one or more fractures, with contours at equal intervals of the logarithm of the aperture. In addition, one can request the printing of data on the two-point correlation of apertures.

4.4 Examining the Pressure and Flow Solutions

There are many options available to display the flow solution. The user may request a plot of the fracture-network showing pressure contours or flux vectors, represented by arrows, on fractures. Plots of pressure on individual fractures may also be requested. Pressure profiles in which graphs of pressure against distance along a line segment through the network are plotted can be drawn for steady-state and transient flow options. Plots of pressure at a point against time are also available. When locally-varying apertures are specified, histograms of fluxes across a line within a fracture can be plotted.

The `pipe' model is a tool for analysing the solution of the flow problem by representing each of the fractures in the network by a collection of pipes. Every pair of intersections in a fracture is connected by a set of pipes, which contains a pipe joining every node on one intersection with every node on the other. It should be understood that the fluxes in the pipes only represent an approximation to the flow field in the fracture. They are computed from exact solutions of the pressure at the nodes and the global response matrices. The pipe model can be used to estimate the flux that crosses a series of parallel plane rectangular surfaces.

4.5 Tracer Transport

Several types of diagram are available for displaying the results of tracer transport simulations. To look at the results of a large swarm of particles sent through the network, graphs showing the proportion of particles leaving the network as a function of time or path length are available. The number of particles is scaled by the total number released to show a `breakthrough curve'. A cumulative arrival curve is plotted by integrating the number of particles arrive as a function of time.

This plot also includes a curve derived by fitting an analytical solution of the advection-dispersion equation to the times by which 25% and 75% of the particles have left the system.

A three-dimensional view of the network showing the swarm of particles at user specified times or a diagram of a boundary surface showing the arrival of particles can also be selected. Individual particle tracks can be investigated by producing a three-dimensional plot showing the tracks or by plotting graphs showing aperture against time or distance for individual particle tracks.

4.6 3D Visualisation

The most powerful way to understand and interpret the results of a 3D network simulation is naturally to visualise the results in 3D. A complementary 3D visualisation software package has been developed specifically for NAPSAC. This is described in the next section.

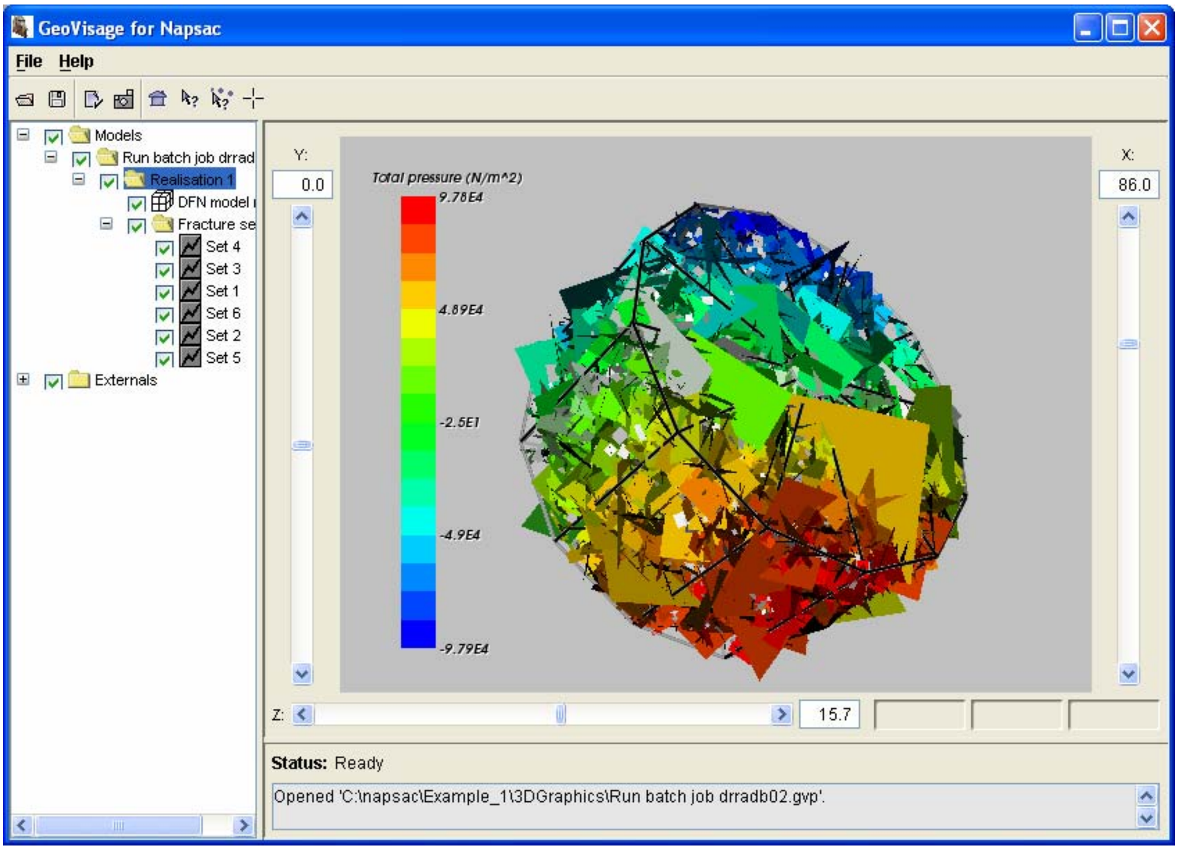
4.7 GeoVisage for NAPSAC

GeoVisage is a 3D visualisation software package designed to display the output from NAPSAC, NAMMU and ConnectFlow. GeoVisage displays in 3D all the features in a NAPSAC model, including the model region, fractures, engineered features and upscaling and their associated scalar and vector data. It is also able to carry out calculations on the model, including clustering, pathlines, and flux through a plane. In addition to displaying the NAPSAC model features it is also possible to overlay external data, such as maps.

The capabilities of GeoVisage for NAPSAC include the interactive 3D visualisation of the following:

- the fracture-network with each fracture coloured according to transient or steady-state properties such as aperture or pressure;
- a reduced network with removal of fractures according to location or properties;
- engineered features;
- upscaling;
- pathlines generated from multiple user-defined sources;
- slices through the model;
- clustering around engineered features or points.

A snapshot of any screen can be saved as a 3D image using VRML (Virtual Reality Modelling Language) format. The saved image can be viewed and manipulated in 3D using a web browser (for example Netscape). Alternatively, the screen can be saved in a 2D format such as PNG, JPEG, or PostScript. 2D images can be combined to create animated movies.



Status: Ready

Opened 'C:\napsac\Example_1\3DGraphics\Run batch job drradb02.gvp'.

Figure 21 A NAPSAC model coloured by total pressure in GeoVisage.

5 Quality Assurance

A Quality Assurance (QA) programme defines a set of procedures for carrying out a particular type of work in such a way as to maintain the quality of the work. A well designed QA programme plays an important role in computer program maintenance by ensuring that high standards of coding are adhered to. There are procedures for reporting and fixing program errors and that there is a system for testing and issuing new releases of the program which ensures that the new program gives the correct results for a standard set of test cases.

NAPSAC is maintained and developed under an appropriate QA programme [21] by the Environmental Management Department of Serco Limited. The QA programme conforms to the international standard ISO 9000 and TickIT. The Concurrent Versions System (CVS) is used to store all source code, documentation and test data for NAPSAC. This automatically logs the author and date of each change to the system, and enables previous versions of the code to be accessed and recreated if necessary. All changes are thoroughly tested, and must be approved by the Software Manager before they are accepted. A comprehensive set of approximately 40 test cases is used to test each new release. Through the NAPSAC QA programme, Serco Limited seeks to continually improve the quality and reliability of NAPSAC.

6 References

1. E. Anderson, Z. Bai, C. Bischof, J. Demmel, J Dongarra, J. DuCroz, A. Greenbaum, S. Hammerling, A McKenney, S Ostrouchov and D. Sorenson. *LAPACK Users' Guide, Second Edition* SIAM, Philadelphia, PA 1995.

2. L. Smith and F. Schwartz, *An Analysis of the Influence of Fracture Geometry on Mass Transport in Fractured Rock*. Water Resour. Res., **20(9)**, 1241, 1984.

3. J.E. Bolt, P. Bourke, N. L. Jefferies, R. Kingdon, D. Pascoe, and V. Watkins. *The Application of Fracture Network Modelling to the Prediction of Groundwater Flow Through Highly Fractured Rock*. Science Report NSS/R281, Nirex, 1995.

4. J. Geier, C. Axelsson, L. Hassler, and A. Benabderahmane. *Discrete Fracture Modelling of the Finnsjön Rock Mass*. Technical Report 92-07, SKB, 1992.

5. Herbert, J. Gale, G. Lanyon, and R. MacLeod. *Modelling for the Stripa Site Characterization and Validation Drift Inflow: Prediction of Flow through Fractured Rock*. Technical Report 91-35, Stripa Project, 1991.

6. O. Olsson,and J. Gale. *Site Assessment and Characterisation for High-Level Nuclear Wast Disposal: Results from the Stripa Project Sweden*. Q. J. Eng. Geol. **28**, S17, 1995.

7. R. A. Scaefer, J. Gale and A. Herbert. *3-D Discrete Fracture Flow Simulation Using Monterey Formation Fracture Data*. SPE 29135, p. 415, 1995.

8. W. S. Dershowitz.. *Rock Joint Systems*. PhD Thesis, MIT, 1984.

9. Herbert and B. Splawski. *Prediction of Inflow into the D-Holes at Stripa Mine*. Technical Report 90-14, Stripa Project, 1990.

10. P.C. Robinson, Connectivity, Flow and Transport in Network Models of Fractured Rock. D.Phil, thesis, Oxford University, 1984.

11. D.T. Snow. *Rock Fracture Spacings, Openings , and Porosities*. J.Soil Mech. Found. Div, Proc. Amer. Civil Engrs., 94, 73, 1968.

12. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, Numerical Recipes. *The Art of Scientific Computing*. Cambridge University Press, 1986.

13. P.M. Wilcock, *Generic Study of Coupled T-H-M Processes in the near-field (BMT3) in Coupled Thermo-Hydro-Mechanical Processes of Fractured Media*, Developments in Geotechnical Engineering 79, S. Stephanson and L. Jing (Eds.), Elsevier, 1996.

14. P. Bogorinski, C.P. Jackson and J.D. Porter, *INTRAVAL Test Case 13: Simulation of an Experimental Study of Brine Transport in Porous Media*. GRS-A-1984, 1984.

15. *NAMMU User Guide*. SA/ENV/CONNECTFLOW/10, Serco Assurance, Harwell.

16. A.W. Herbert and G.W.Lanyon, *Modelling Tracer Transport in Fractured Rock at Stripa*. Stripa Project Technical Report 91-01, SKB, Stockholm, 1991.

17. F.W. Schwartz and G. Lee, *Cross-Verification testing of Fracture Flow and Mass Transport Codes*. Stripa Project Technical Report 91-29, SKB, Stockholm, 1991.

18. HSL, *Harwell Subroutine Library Specifications (Release 12).* Volume 1, AEA Technology Report, Harwell Laboratory, Oxfordshire, 1995.

19. S.W. Sloan, *An Algorithm for Profile and Wavefront Reduction of Sparse Matrices*. Inter. J. Numer. Meth. Engng., 23, 239, 1986.

20. P. Hood, *Frontal Solution Program for Unsymmetric Matrices*. Int. J. Numer. Meth. Engng., 10, 379, 1976.

21. S.T. Morris and G. Webster*, Quality Sub-Programme: Computer Packages Based on TGSL.* AEA Technology Report, WEG/QAP/TGSL, 1996

22. C. Cordes and W. Kinzelbach, "Continuous Groundwater Velocity Fields and Path Lines in Linear, Bilinear, and Trilinear Finite Elements", Water Resources Research, Vol. 28, No. 11, Pages 2903-2911, November 1992.