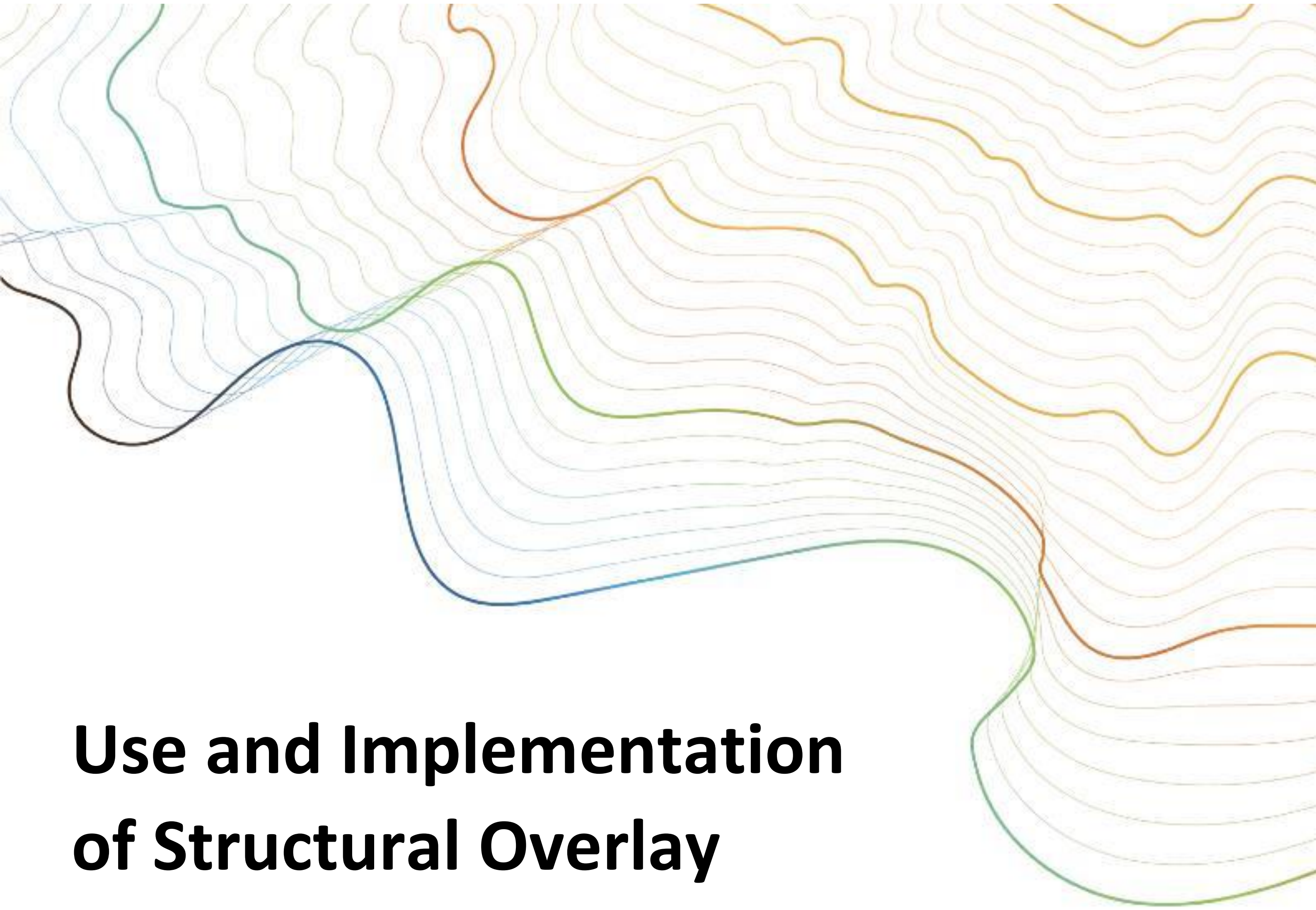


Use and Implementation of Structural Overlay Parameters

A GMDSI worked example report

gmdsi
Groundwater Modelling
Decision Support Initiative

BHP
Flinders UNIVERSITY
NATIONAL CENTRE FOR GROUNDWATER RESEARCH AND TRAINING
RioTinto

PUBLISHED BY

The National Centre for Groundwater Research and Training
C/O Flinders University
GPO Box 2100
Adelaide SA 5001
+61 8 8201 2193

DISCLAIMER

CITATION

For bibliographic purposes this report may be cited as: Minchin, W. and Doherty, J., (2023). Use and Implementation of Structural Overlay Parameters. A GMDSI Worked Example Report. National Centre for Groundwater Research and Training, Flinders University, South Australia.

AFFILIATIONS

Will Minchin: Watershed HydroGeo
John Doherty: GMDSI Project, Flinders University and Watermark Numerical Computing

ISBN:

DOI:

DOI as a weblink:

COPYRIGHT

PREFACE

The Groundwater Modelling Decision Support Initiative (GMDSI) is an industry-funded and industry-aligned project focused on improving the role that groundwater modelling plays in supporting environmental management and decision-making. Over the life of the project, it will document a number of examples of decision-support groundwater modelling. These documented worked examples will attempt to demonstrate that by following the scientific method, and by employing modern, computer-based approaches to data assimilation, the uncertainties associated with groundwater model predictions can be both quantified and reduced. With realistic confidence intervals associated with predictions of management interest, the risks associated with different courses of management action can be properly assessed before critical decisions are made.

GMDSI worked example reports, one of which you are now reading, are deliberately different from other modelling reports. They do not describe all of the nuances of a particular study site. They do not provide every construction and deployment detail of a particular model. In fact, they are not written for modelling specialists at all. Instead, a GMDSI worked example report is written with a broader audience in mind. Its intention is to convey concepts, rather than to record details of model construction. In doing so, it attempts to raise its readers' awareness of modelling and data-assimilation possibilities that may prove useful in their own groundwater management contexts.

The decision-support challenges that are addressed by various GMDSI worked examples include the following:

- assessing the reliability of a public water supply;
- protection of a groundwater resource from contamination;
- estimation of mine dewatering requirements;
- assessing the environmental impacts of mining and civil infrastructure; and
- management of an aquifer threatened by salt water intrusion.

In all cases the approach is the same. Management-salient model predictions are identified. Ways in which model-based data assimilation can be employed to quantify and reduce the uncertainties associated with these predictions are reported. Model design choices are explained in a way that modellers and non-modellers can understand.

The authors of GMDSI worked example reports make no claim that the modelling work which they document cannot be improved. As all modellers know, time and resources available for modelling are always limited. The quality of data on which a model relies is always suspect. Modelling choices are always subjective, and would often made differently with the benefit of hindsight.

What we do claim, however, is that the modelling work which we report has attempted to implement the scientific method to address challenges that are typical of those encountered on a day-to-day basis in groundwater management worldwide.

As stated above, a worked example report purposefully omits many implementation details of the modelling and data assimilation processes that it describes. Its purpose is to demonstrate what can be done, rather than to explain how it is done. Those who are interested in technical details are referred to GMDSI modelling tutorials. A suite of these tutorials has been developed specifically to assist modellers in implementing workflows such as those that are described herein.

We thank and acknowledge our collaborators, and GMDSI project funders, for making these reports possible.

GLOSSARY

Anisotropy

A condition whereby the properties of a system (such as hydraulic conductivity) are likely to show greater continuity in one direction than in another. At a smaller scale it describes a medium whose properties depend on direction.

Application Programming Interface (API)

A set of functions that allows the user of a software package to control the inner workings of that package from his/her own program.

Bayesian analysis

Methods that implement history-matching according to Bayes equation. These methods support calculation of the posterior probability distribution of one or many random variables from their prior probability distributions and a so-called “likelihood function” – a function that increases with goodness of model-to-measurement fit.

Boundary condition

The conditions within, or at the edge of, a model domain that allow water or solutes to enter or leave a simulated system.

Boundary conductance

The constant of proportionality that governs the rate of water movement across a model boundary in response to a head gradient imposed across it.

Capture zone

The three-dimensional volumetric portion of a groundwater flow field that discharges water to a well.

Connected linear network (CLN) package

This package is supported by the MODFLOW-USG simulator. Water flows through a series of one-dimensional features, each of which can be linked to another such feature, or to a cell within a two or three-dimensional groundwater model domain.

Contributing area

The two-dimensional areal extent of that portion of a capture zone that intersects the water table and surface water features where water entering the groundwater flow system is discharged by a well. (This is also referred to as the *area contributing recharge.*)

Covariance matrix

A matrix is a two-dimensional array of numbers. A covariance matrix is a matrix that specifies the statistical properties of a collection of random variables - that is, the statistical properties of a random vector. The diagonal elements of a covariance matrix record the variances (i.e. squares of standard deviations) of individual variables. Off-diagonal matrix elements record covariances between pairs of variables. The term “covariance” refers to the degree of statistical inter-relatedness between a pair of random variables.

Ensemble

A collection of realisations of random parameters.

Drain (DRN) package

A one-way Cauchy boundary condition implemented by MODFLOW. Water can flow out of a model domain, but cannot enter a model domain through a DRN boundary condition.

Evapotranspiration (EVT) package

MODFLOW's simulation of plant evapotranspiration and direct evaporation from a groundwater system. The extraction rate increases, up to a user-specified maximum rate, as the water table approaches a user-prescribed elevation from below. If the water table falls below a user-specified extinction depth, evapotranspiration ceases.

General head boundary (GHB) package

This is MODFLOW parlance for a Cauchy boundary condition. Water flows into or out of a model domain in proportion to the difference between the head ascribed to the boundary and that calculated for neighbouring cells. The rate of water movement through the boundary in response to this head differential is governed by the conductance assigned to the boundary.

Hydraulic conductivity

The greater is the hydraulic conductivity of a porous medium, the greater is the amount of water that can flow through that medium in response to a head gradient.

Jacobian matrix

A matrix of partial derivatives (i.e. sensitivities) of model outputs (generally those that are matched with field measurements) with respect to model parameters.

Managed aquifer recharge (MAR)

Human-managed storage of water in the subsurface. Storage can be effected using recharge basins or through direct injection of water through boreholes.

Matrix

A two-dimensional array of numbers indexed by row and column.

MODFLOW

A family of public-domain, finite-difference groundwater flow simulators developed by the United States Geological Survey (USGS).

MODFLOW-USG

A version of MODFLOW which employs an unstructured grid. This was developed by Sorab Panday in conjunction with the United States Geological Survey (USGS).

MODFLOW package

An item of simulation functionality that describes one aspect of the operation of a groundwater system, for example recharge or a boundary condition. The word "package" describes the computer code that implements this functionality in a modular fashion, as well as its input and output file protocols.

MODPATH

A family of MODFLOW-suite post-processors that undertake particle-tracking in MODFLOW-calculated flow fields.

Mod-PATH3DU

A particle tracking program that can accommodate unstructured grids. It can evaluate particle tracks in flow fields computed by programs of the MODFLOW suite, and by MODFLOW-USG.

Null space

In the parameter estimation context, this refers to combinations of parameters that have no effect on model outputs that are matched to field observations. These combinations of parameters are thus inestimable through the history-matching process.

Objective function

A measure of model-to-measurement misfit whose value is lowered as the fit between model outputs and field measurements improves. In many parameter estimation contexts the objective function is calculated as the sum of squared weighted residuals.

Parameter

In its most general sense, this is any model input that is adjusted in order to promote a better fit between model outputs and corresponding field measurements. Often, but not always, these inputs represent physical or chemical properties of the system that a model simulates. However there is no reason why they cannot also represent water or contaminant source/sink strengths and locations.

Phreatic surface

The water table.

Pilot point

A type of spatial parameterisation device. A modeller, or a model-driver package such as PEST or PEST++, assigns values to a set of points which are distributed in two- or three-dimensional space. A model pre-processor then undertakes spatial interpolation from these points to cells comprising the model grid or mesh. This allows parameter estimation software to ascribe hydraulic property values to a model on a pilot-point-by-pilot-point basis, while a model can accept these values on a model-cell-by-model-cell basis. The number of pilot points used to parameterise a model is generally far fewer than the number of model cells.

Pit dewatering

Extraction of water from a groundwater system in order to reduce the elevation of the water table to below the elevation of the bottom of a mining pit.

Prior probability

The pre-history-matching probability distribution of random variables (model parameters in the present context). Prior probability distributions are informed by expert knowledge, as well as by data gathered during site characterisation.

Posterior probability

The post-history-matching probability distribution of random variables (model parameters in the present context). These probability distributions are informed by expert knowledge, site characterisation studies, and measurements of the historical behaviour of a system.

Probability density function

A function that describes how likely it is that a random variable adopts different ranges of values.

Probability distribution

This term is often used interchangeably with “probability density function”.

Quadtree mesh refinement

This term refers to a means of creating fine rectilinear model cells from coarse rectilinear model cells by dividing them into four. Each of the subdivided cells can then be further subdivided into another four cells. However it is a design specification of a quadtree-refined grid that no cell within the domain of a model be connected to more than two neighbouring cells along any one of its edges.

Realisation

A random set of parameters.

Regularisation

The means through which a unique solution is sought to an ill-posed inverse problem. Regularisation methodologies fall into three broad categories, namely manual, Tikhonov and singular value decomposition.

Residual

The difference between a model output and a corresponding field measurement.

River (RIV) package

A MODFLOW package which provides basic simulation of the interaction between groundwater and a surface water body. Flow between the two regimes is driven by the head difference between them. Through definition of the elevation of the bottom of the river, the driving head difference can be limited.

Singular value decomposition (SVD)

A matrix operation that creates orthogonal sets of vectors that span the input and output spaces of a matrix. When undertaken on a Jacobian matrix, SVD can subdivide parameter space into complementary, orthogonal subspaces; these are often referred to as the solution and null subspaces. Each of these subspaces is spanned by a set of orthogonal vectors. The null space of a Jacobian matrix is composed of combinations of parameters that have no effect on model outputs that are used in its calibration, and hence are inestimable.

Solution space

The orthogonal complement of the null space. This is defined by undertaking singular value decomposition of a Jacobian matrix.

Specific storage

The amount of water that is stored elastically in a cubic metre of porous medium when the head of water in which that medium is immersed rises by 1 metre.

Specific yield

The amount of accessible (“drainable”) water that is stored in the pores of a porous medium per volume of that medium.

Stochastic variable

A stochastic variable is a random variable.

Stress

In the context of groundwater modelling, this term generally refers to those aspects of a groundwater model that cause water to move. They generally pertain to boundary conditions. User-specified heads along one side of a model domain, extraction from a well, and pervasive groundwater recharge, are all examples of groundwater stresses.

Stress period

The MODFLOW family of models employs this terminology to describe each member of a series of contiguous time intervals that collectively comprise the simulation time of a model. Stresses applied to a groundwater system may be different in each stress period.

Tikhonov regularisation

An ill-posed inverse problem achieves uniqueness by finding the set of parameters that departs least from a user-specified parameter condition, often one of parameter equality and hence spatial homogeneity.

Time-variant specified head (CHD) package

This is a Dirichlet (i.e. “fixed head”) boundary condition implemented by MODFLOW in which the head can vary with time on a stress-period-by-stress-period basis.

Vector

A collection of numbers arranged in a column and indexed by their position in the column.

Well (WEL) package

A MODFLOW package that simulates withdrawal of water from a groundwater system.

EXECUTIVE SUMMARY

Geological structural features can have a significant impact on groundwater flow. They can reduce permeability in one direction at the same time as they can enhance permeability in orthogonal directions.

In this GMDSI worked example report, we consider the influence of structural features on groundwater flow induced by tunnelling. These features are assumed to be hydraulically conductive in both the horizontal and vertical directions. They do not offset any of the stratigraphic layers that they intersect.

These structures intersect an alternating sequence of sandstones and claystones. The hydraulic properties of these stratigraphic units, and of the structures that intersect them, are represented stochastically. Pilot points provide the primary parameterisation device. The hydraulic properties of the structural features are overlain on those of the background sedimentary sequence using an automated parameterisation script that is implemented using the PLPROC parameter pre-processor. This allows PLPROC to run as part of a model that is subjected to history-matching using the PESTPP-IES ensemble smoother.

The report begins by providing a brief overview of the groundwater model. The discussion then turns to how values of horizontal hydraulic conductivity are awarded to all model cells, taking into account both the stratigraphic and structural origins of this hydraulic property. (Values for other hydraulic properties are calculated in a similar manner.) Use of PLPROC's structural overlay parameterisation functionality is described in detail. A PLPROC script is provided.

Programs of the Groundwater Utility Suite that support visualisation and model-GIS data exchange are also briefly discussed.

CONTENTS

1. INTRODUCTION

This GMDSI worked example report focusses on parameterization of a complex, transient, three-dimensional groundwater model. The model calculates inflow into a tunnel, and the effect of this inflow on the surrounding groundwater system.

The tunnel will be excavated through a layered sedimentary system of alternating sandstones and claystones. This stratigraphic sequence is represented using 20 model layers. Part of the uppermost model layer also represents surficial swamps that are the outcomes of recent geomorphological processes. Along its length, the tunnel intersects 14 model layers. The hydraulic conductivities of these layers are likely to vary laterally, as they result from small scale fracturing of brittle rocks. Patterns and magnitudes of hydraulic conductivity heterogeneity cannot be back-calculated from an existing set of measurements that comprise the model's history-matching dataset; hence they are represented stochastically, while being constrained by this dataset.

A number of lineaments and faults have been mapped in the study area. The effects on groundwater flow of the structural features that they evince is unknown. Predictive conservativism requires that their possible significance be accommodated when assessing the range of possible tunnel inflows, together with regional drawdowns that are induced by these inflows. Hence they are presumed to potentially possess high horizontal and vertical hydraulic conductivities compared with the material that they intersect. They are presumed to intersect all model layers except for those parts of the uppermost model layer that represent recent and unconsolidated swamp deposits. The hydraulic properties of the structural features can thus be considered to be overlain on those of the regional geology.

This report describes how pervasive structural features such as these can be defined and parameterised using the PLPROC parameter pre-processor. Automation of their parameterisation is essential, as their properties may vary from model run to model run as parameters are adjusted, or as their stochasticity is expressed.

The contents of this short report are as follows.

First a brief description of the site and the groundwater model is provided in Sections 2 and 3, respectively. The manner in which PLPROC is used to define and assign values to "structural overlay parameters" is described in Section 4; some details of the PLPROC script are presented along with this description. Some outcomes of the stochastic parameterisation process are also provided in this section.

2. THE SITE

2.1 General

The Great Western Highway is the key east-west road freight and transport route between Sydney and Central West New South Wales (Australia). At the time of writing, plans are being made to upgrade this highway. The focus of the model that is described herein is on construction of a twin roadway tunnel through part of the Blue Mountains west of Sydney. The tunnel will be about 11 km in length. Its maximum depth will be about 180 m. See Figure 2.1.

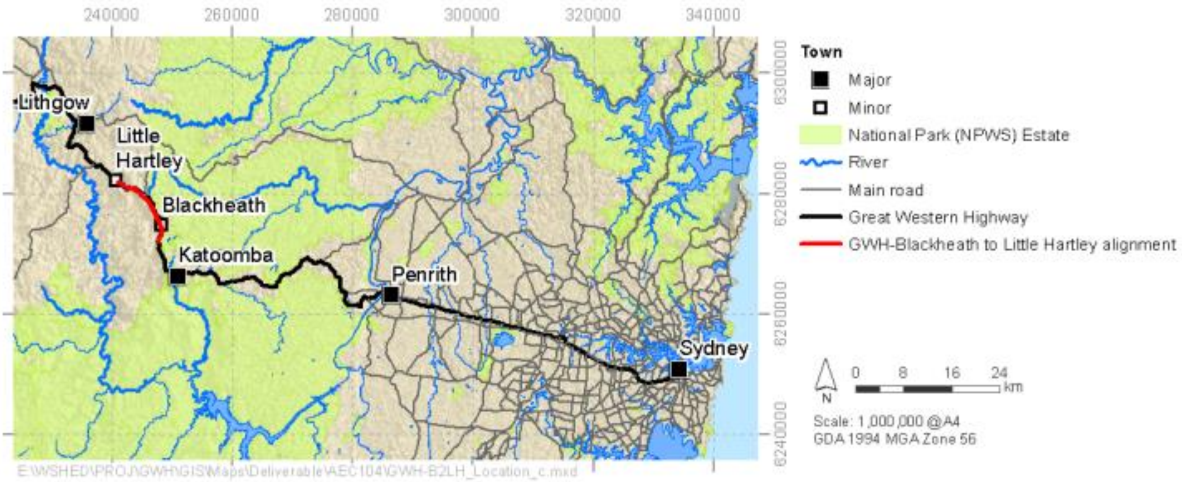


Figure 2.1. Great Western Highway and location of proposed tunnel. (From Watershed HydroGeo, 2022).

The segment of the present highway that the tunnel will replace follows a ridgeline that is bounded on both sides by steeply sided valleys. To the north and west of this, the highway descends down an escarpment to a valley floor. Hanging swamps are not uncommon on the many escarpments that characterise this part of the Blue Mountains. These mark locations of groundwater seepage, often along planes where a relatively permeable stratigraphic unit is underlain by a less permeable unit (see below).

The World Heritage listed Blue Mountains National Park lies to the east of the study area.

2.2 Hydrogeology

The tunnel will be excavated through a sequence of sandstones and claystones comprising the Lower Triassic Narrabeen Group and the Upper Permian Illawarra Coal Measures. For all but the Upper Banks Wall Sandstone (which is the uppermost stratigraphic unit at the site) the conceptual model characterises groundwater flow as occurring almost exclusively through a fracture system composed of:

- joints that are predominantly close to vertical;
- horizontal bedding partings that may be sheared;
- fracture zones associated with faulting; and
- thin zones (aureoles) of highly to extremely weathered rocks around these features.

Sandstones tend to be more permeable than claystones. Being more brittle, they are more intensely fractured.

The hydraulic gradient is predominantly vertical. Recharging waters seep through the Banks Wall Sandstone. Where hydraulic conductivity allows, these waters move vertically through joints, and then laterally through bedding partings which connect to other vertical joints and eventually to the surface on valley sides and cliff faces. Estimates of average annual recharge vary between 65 mm/yr and 130 mm/yr, this being about 5 -10% of average rainfall.

Within the study area, a number of significant lineaments and faults have been mapped. These may represent near-vertical, planar zones of jointing- or faulting-enhanced horizontal and vertical permeability. As is described below, these zones are represented in the model.

2.3 The Tunnel

It is proposed that two parallel tunnels of approximately 11 km in length be excavated between the towns of Blackheath and Little Hartley. Each tunnel will contain a carriageway of approximately 11 m width, this including two traffic lanes plus shoulders. Each tunnel will have a diameter of about 15 m; each will be separated from the other by approximately 11 m. Short cut-and-cover sections will be excavated at the eastern and western tunnel entrances. See Figure 2.2.

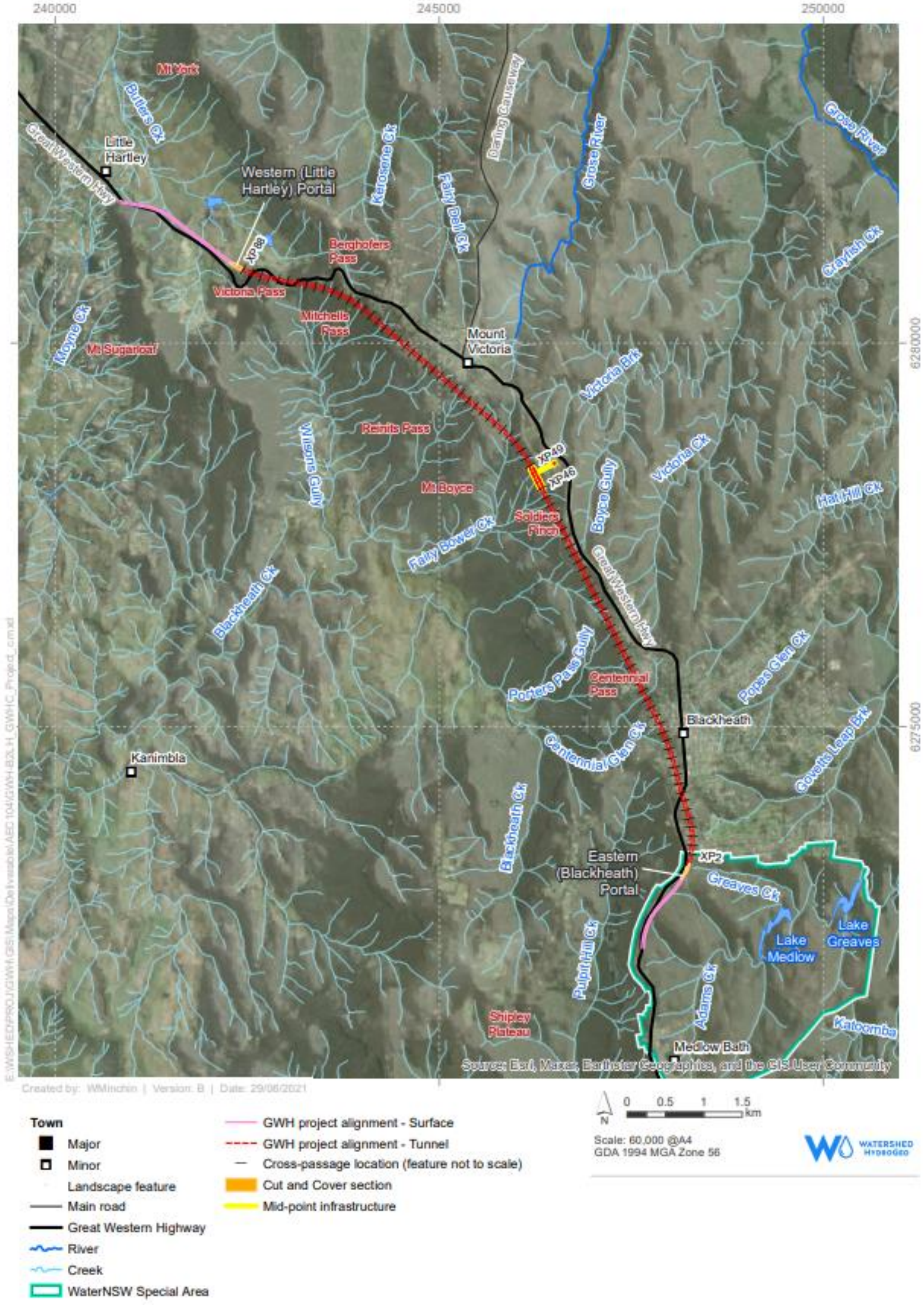


Figure 2.2. Tunnel layout. (From Watershed HydroGeo, 2022).

Tunnel boring machines (TBMs) will be used for tunnel construction. The construction process is expected to induce minimal groundwater inflow as lining will be installed immediately behind the cutting face; the cutting face will be pressurised with slurry where required.

After excavation and lining of the dual tunnels, cross-passages between them will be installed at intervals of approximately 120 m. These will house plant items such as substations. They will also allow emergency egress between tunnels if an incident occurs in one of them. Their excavation and construction will be such that each will be open to groundwater inflow for a period of about 3 months.

During construction of the tunnel, a vertical access shaft of about 100 m depth, together with a 300 m long sub-horizontal adit will be excavated near the tunnel midpoint. These will enable access to tunnel workings, including an enlarged cavern at the midpoint that allows for TBM maintenance. All these features (shaft, adit and cavern) may drain groundwater during their construction. After completion of tunnelling, the adit and shaft will be backfilled.

3. MODELLING

3.1 Modelling Objectives

Modelling was undertaken in order to calculate a number of quantities that are of engineering and environmental significance. These include the following:

- Inflow to the tunnel during its construction. (Post construction inflow will be negligible because the tunnel, and most of the ancillary excavations such as cross-passages, will be lined);
- Long-term inflow to cut-and-cover tunnel approaches and the midpoint cavern;
- The effects of these inflows on regional groundwater levels, surficial swamps, hanging swamps and valley streams.

3.2 Modelling Platform

The model was built using the Groundwater Vistas (GV) graphical user interface. GV supports grid and boundary condition construction, as well as parameterisation and history-matching design. However introduction of structural overlay parameters to the model was undertaken outside the GV platform, as GV does not support it at the time of writing.

3.3 Simulator and Grid

Flow of groundwater is simulated using MODFLOW-USG-Transport v1.9.0 (Panday, 2022; Panday, 2021; Panday et al, 2013). This is referred to as “MFUSG” herein.

“USG” stands for “unstructured grid”. Features of the model that rely on an unstructured grid include the following.

- Cells of very small vertical dimension are eliminated where they disappear from the stratigraphic sequence; meanwhile, overlying and underlying cells are connected to each other through missing cells.

- Inactive model cells are eliminated from the model, this reducing memory requirements and increasing model execution speed.

- The model grid is quadtree-refined at three places within its domain, namely at the cut-and-cover approaches to the tunnel, and at the midpoint of the tunnel where cavern, shaft and adit construction will take place.

Cells that are not subject to quadtree refinement have lateral dimensions of 200 m × 200 m. The smallest cells in refined areas have lateral dimensions of 25 m × 25 m; this is similar to the width of some of the excavations.

Layering of the numerical model is based on stratigraphy. The model possesses 20 layers. In general, neighbouring layers can be classed as aquifers or aquitards; stratigraphic boundaries that are likely to promote horizontal flow of groundwater are therefore respected.

Part of the uppermost model layer represents surficial swamps.

Figure 3.1 shows the lateral disposition of active cells in model layers 1, 2, 5, 10, 15 and 20. The tunnel and ancillary excavations (eastern/western cut-and-cover segments, as well as the midpoint shaft and adit) are also shown in each of these pictures.

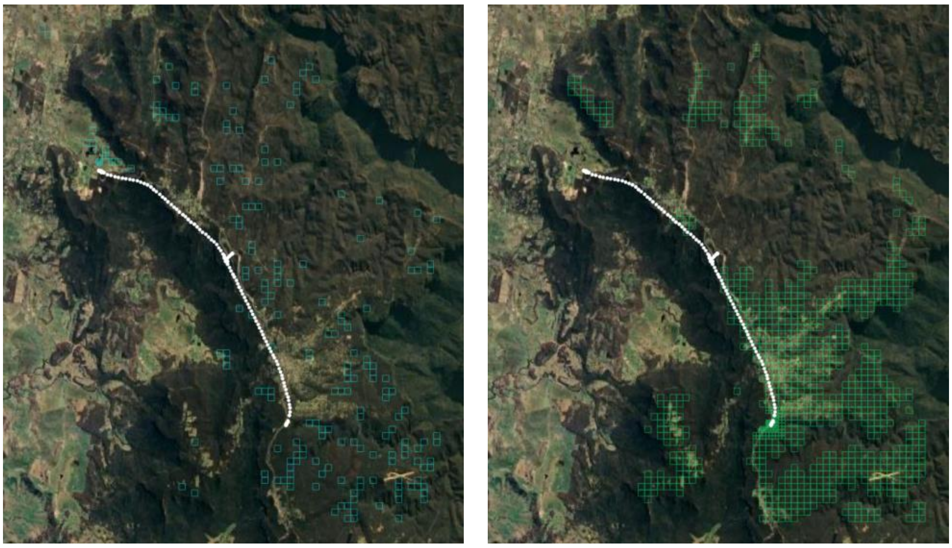


Figure 3.1a. Cells in model layers 1 and 2.

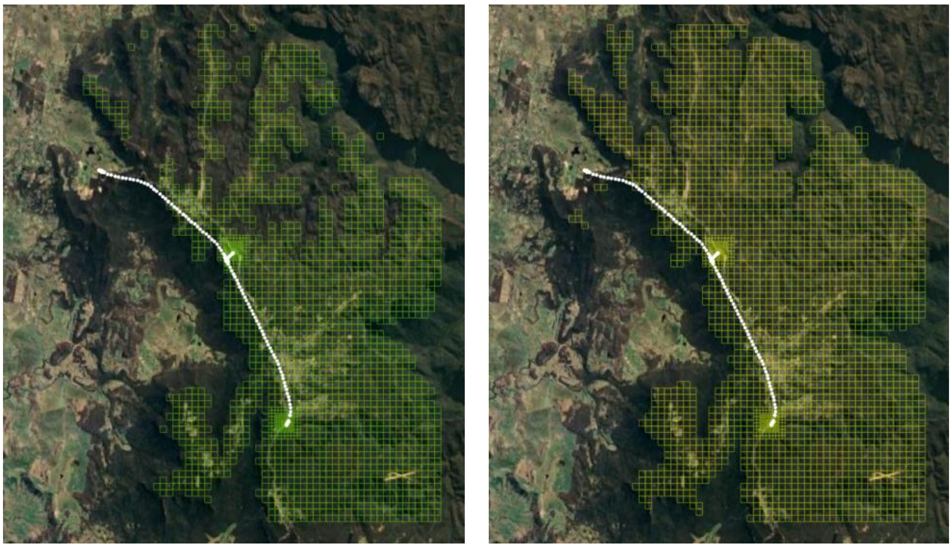


Figure 3.1b. Cells in model layers 5 and 10.

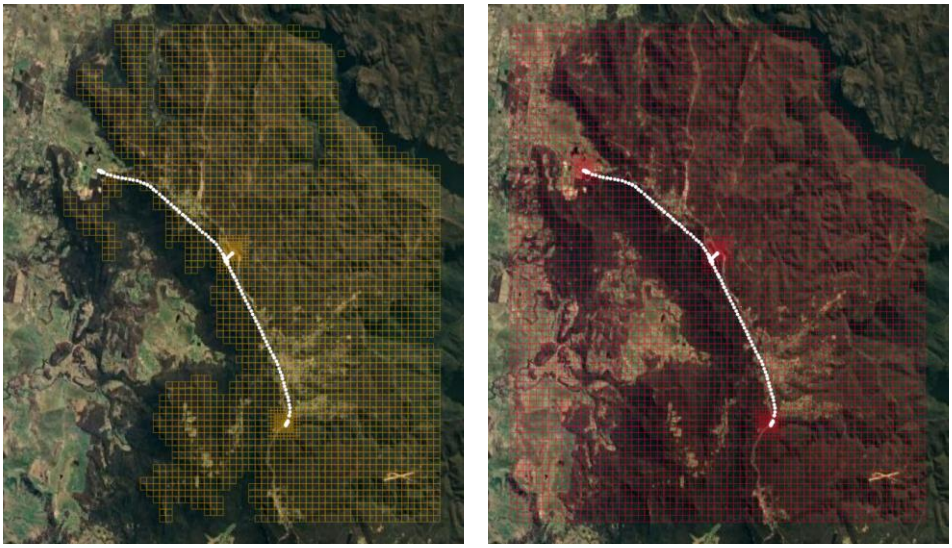


Figure 3.1c. Cells in model layers 15 and 20.

Figure 3.2a show a 3D view of the model domain, looking towards the east. The vertical exaggeration is 10 to 1. Cells are coloured according to elevation (see the legend in Figure 3.2b). Figure 3.2b shows a view towards the north. Note that in these (and all other 3D views of the model domain that are presented herein), model cells are endowed with horizontal tops and bottoms. This makes viewing of their relative dispositions clearer than if gradational cell boundaries were employed. The apparent disconnectedness of some cells in the south-eastern corner of the model domain is illusory. They are connected in the model despite their vertical offsets in the figures.

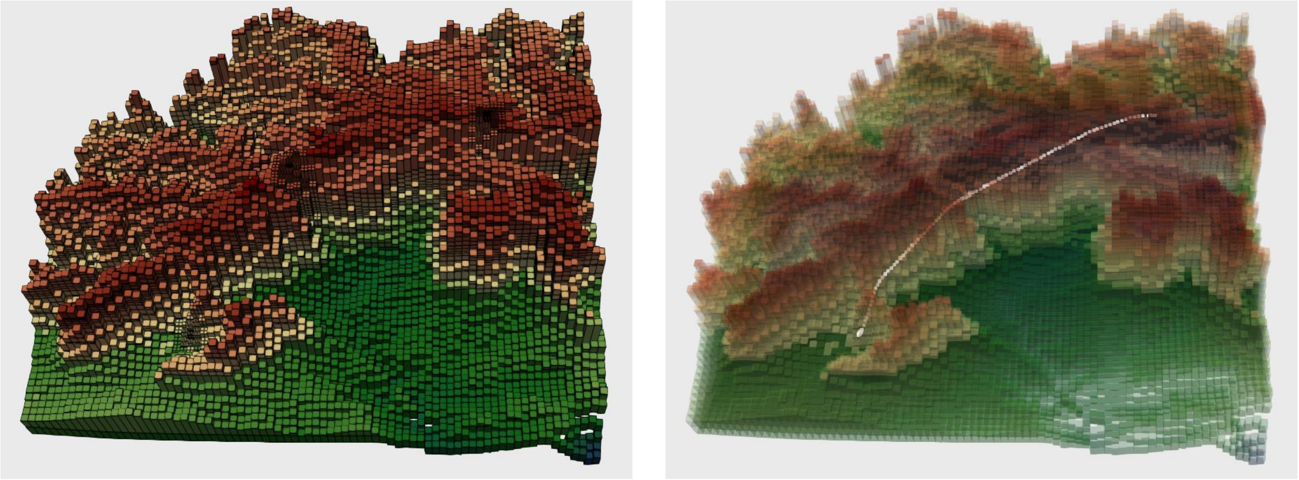


Figure 3.2a. View of the model towards the east. Cells are coloured according to elevation. Cells are transparent in the right picture, where the tunnel is depicted in white.

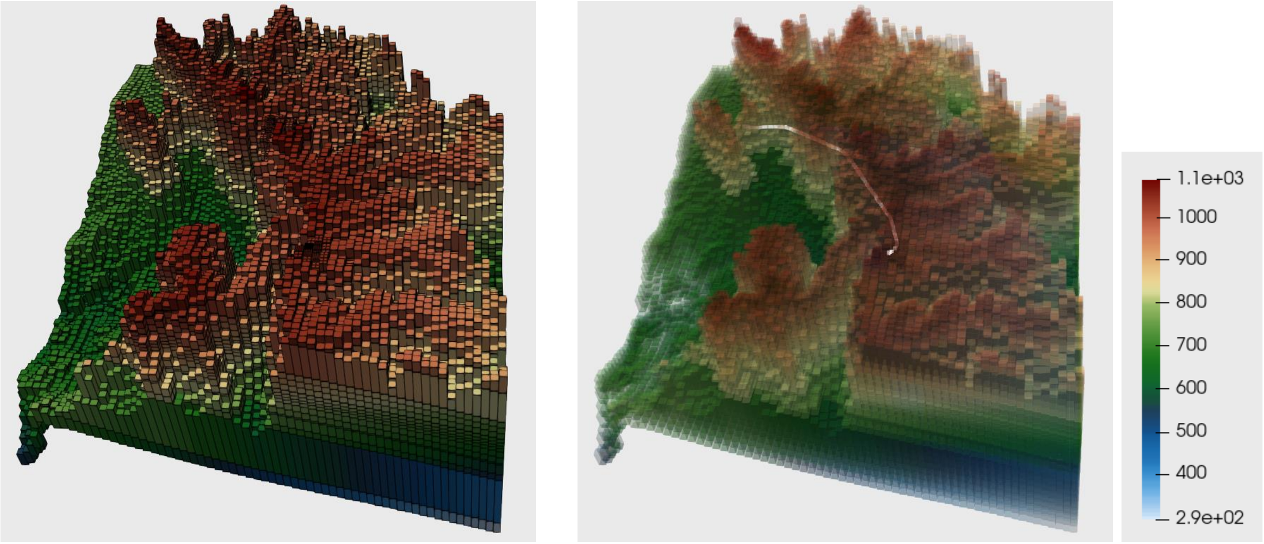


Figure 3.2b. View of the model towards the north. Cells are coloured according to elevation (m). Cells are transparent in the right picture, where the tunnel is depicted in white.

Figure 3.3 displays a series of north-south sections through the model domain that illustrate the trajectory of the tunnel and its relationship to model layering. The view is towards the east. Juxtaposed model layers are coloured in such a way as to render layer boundaries clearly visible.

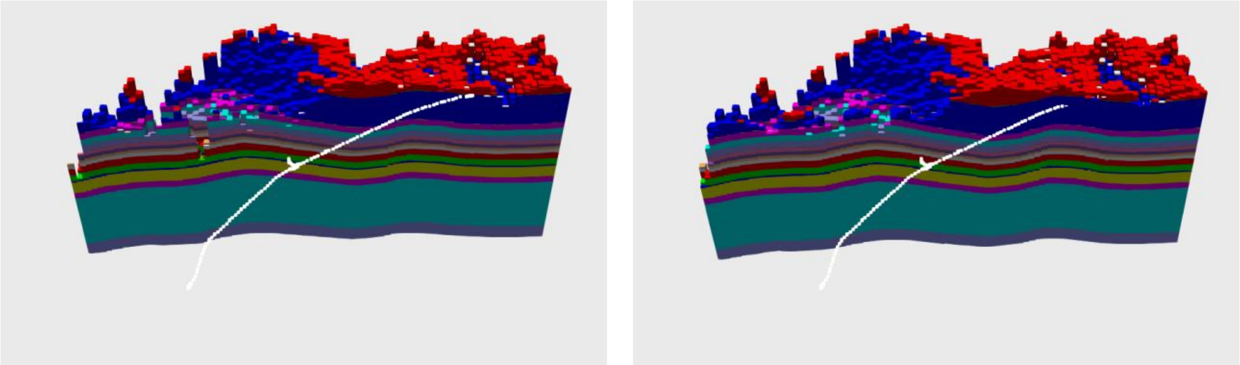


Figure 3.3a. N-S section through the model showing path of tunnel and ancillary excavations.

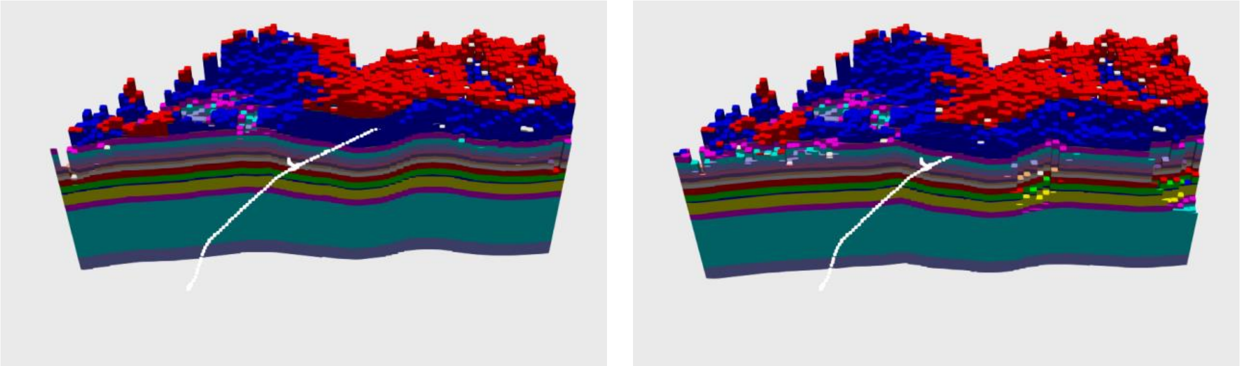


Figure 3.3b. N-S section through the model showing path of tunnel and ancillary excavations.

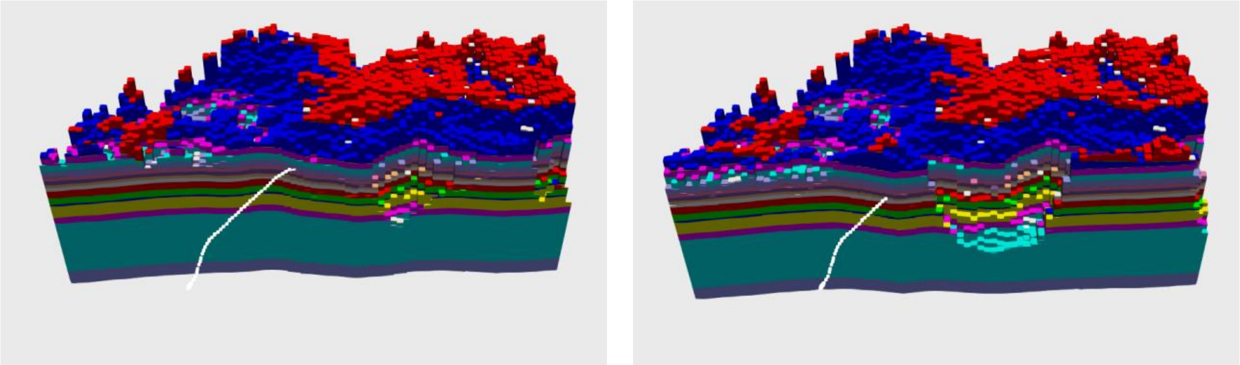


Figure 3.3c. N-S section through the model showing path of tunnel and ancillary excavations.

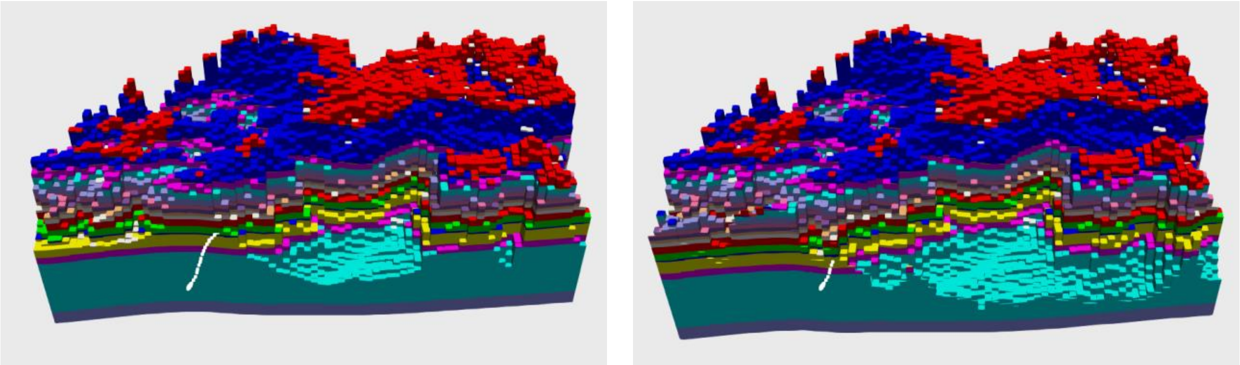


Figure 3.3d. N-S section through the model showing path of tunnel and ancillary excavations.

3.4 Boundary Conditions

The model receives recharge through surficial cells; the layer to which a surficial cell belongs varies with location as it depends on topography. Recharge is simulated using the MFUSG RCH package. Water can also be lost through surficial cells; this is simulated using the MFUSG EVT package.

Water escapes from the model through MFUSG RIV (i.e. river) cells. The locations of cells which contain RIV boundary conditions are shown in Figure 3.4. Note the following.

- For all RIV boundary conditions, the bottom elevation of the “river bed” is set equal to its top elevation. Under these circumstances, RIV boundary conditions perform identically to DRN (i.e. drain) boundary conditions.
- The elevation of the “river bed” is set equal to the lowest topographic elevation in each affected cell. This is not apparent in Figure 3.4. In many of the cells that are depicted in this figure, water is lost through the lower parts of the cells; it is not lost over a cell’s entire vertical length as suggested by the picture.
- RIV boundary conditions are used to simulate escape of water from hanging and surficial swamps, as well as from streams, springs and valley seeps. RIV boundary conditions which perform these different functions are assigned to different zones. Zonal RIV conductances were adjusted during model history-matching.

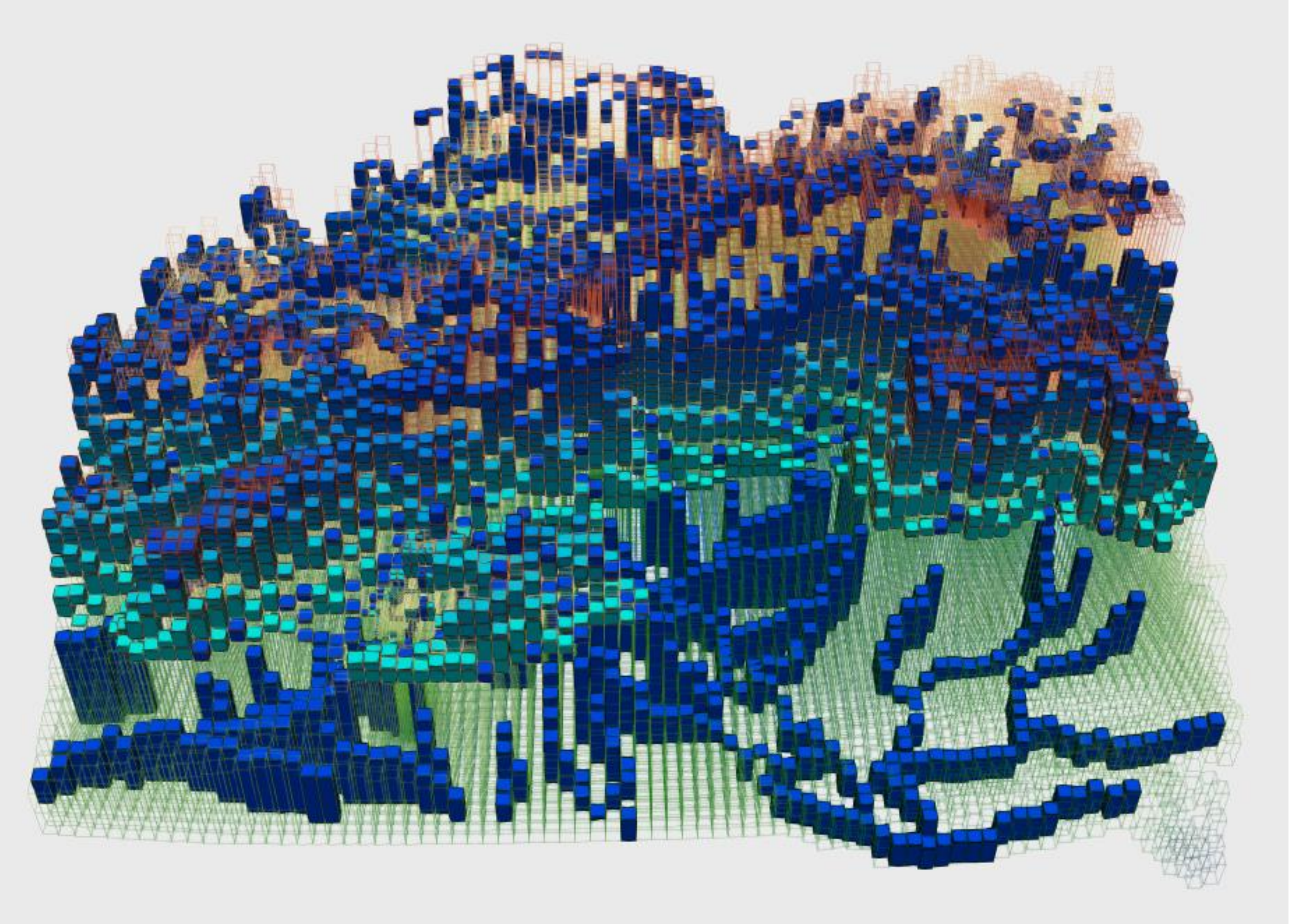


Figure 3.4. Cells in which RIV boundary conditions are emplaced.

A small number of MFUSG GHB (i.e. general head) boundary conditions are used to simulate connection to the broader groundwater system, the area of which (and importance to predictions) is relatively limited, given the topographic relief and the position of the tunnel. See Figure 3.5.

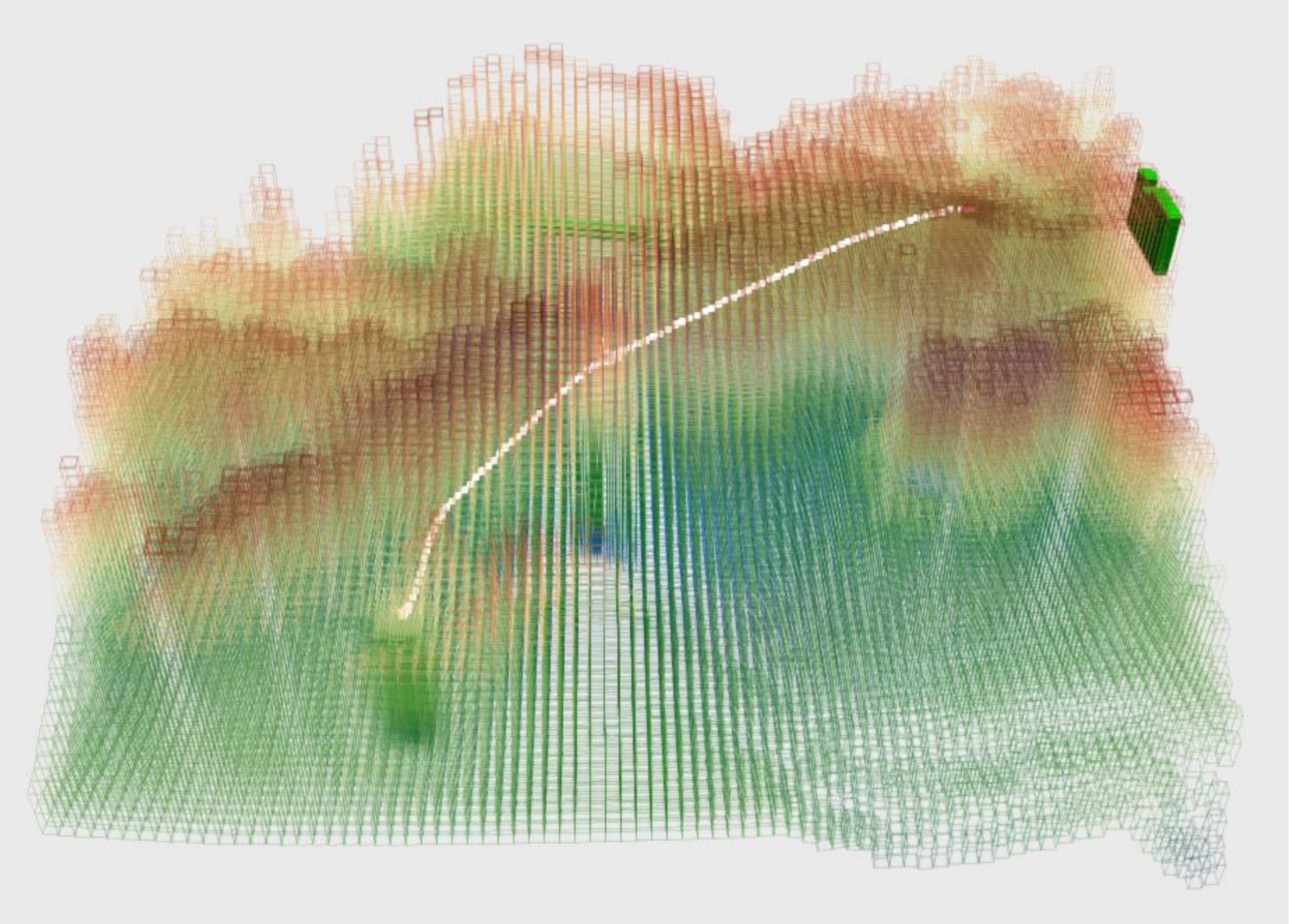


Figure 3.5. Cells in which GHB boundary conditions are emplaced.

Flow of water into the tunnel and ancillary excavations is simulated using the MFUSG DRN (i.e. drain) package. For each DRN boundary condition, the elevation of the DRN is set at the excavation invert. The DRN conductance is assigned a value that accounts for passage of water within the cell to the drainage feature that it simulates; see Zaidel et al (2010). When running the model, the disposition of DRN boundary conditions changes from stress period to stress period to reflect the timing of tunnel excavation.

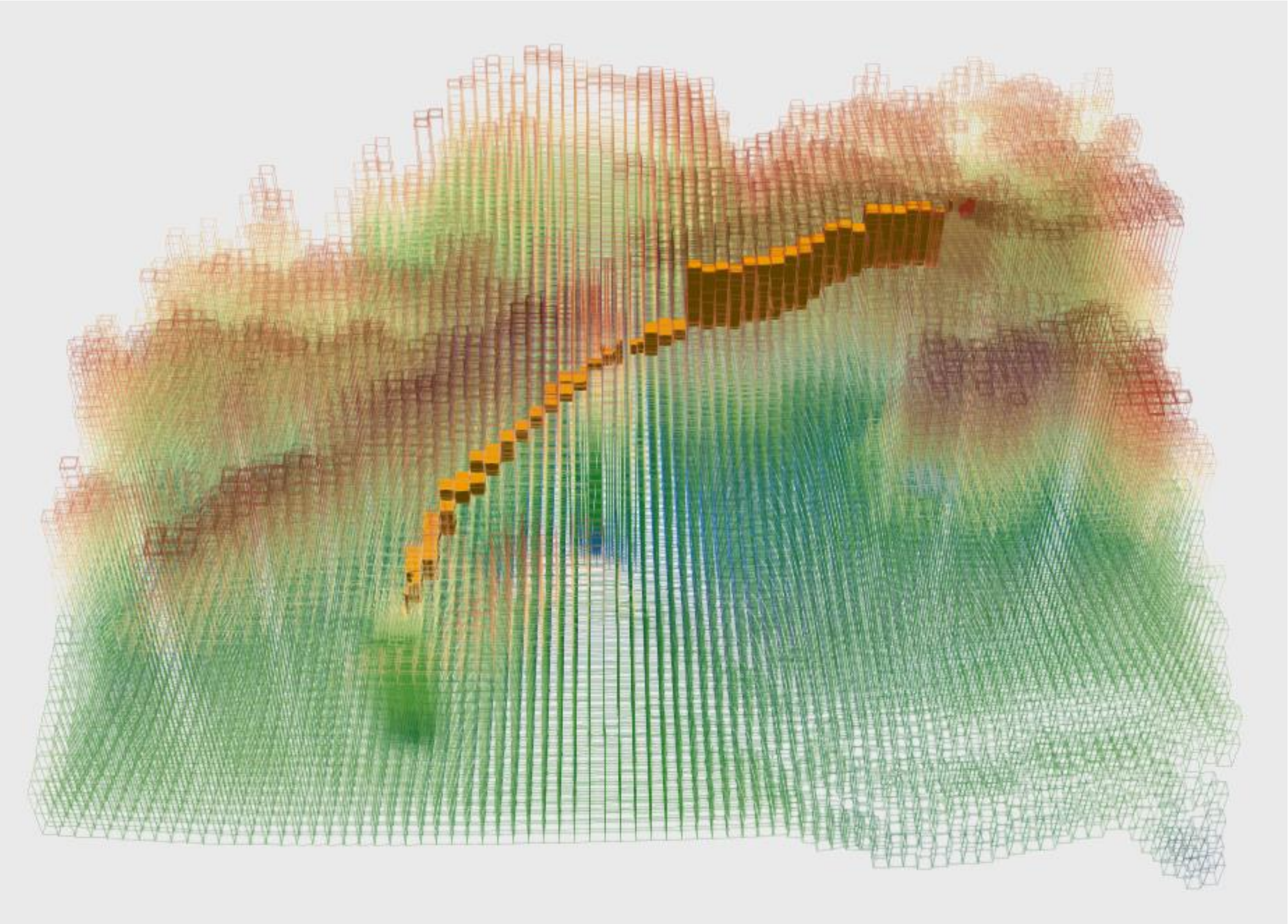


Figure 3.6. Cells in which DRN boundary conditions are emplaced.

3.5 Stress Periods

The model simulates groundwater conditions over the period 2008 to 2130. This is discretised into 121 stress periods. The historical component of this simulation time is subdivided into 38 stress periods; these cover the period January 2008 to June 2022. The first stress period is steady state; this is used for head initialisation.

The predictive period extends from July 2022 until 2130. This covers the period of tunnel construction (which extends until 2030). The model then calculates heads over the 100 year design-life of the tunnel. Over this time, the eastern cut-and-cover tunnel approach and the mid-tunnel cavern may act as groundwater sinks.

3.6 Model Deployment

As is discussed in the next section of this document, the parameterisation complexity of the model complements its structural complexity. Model parameters were awarded a joint prior probability distribution. This distribution was then sampled 300 times. Using the PESTPP-IES ensemble smoother (White, 2018), these prior parameter realisations were adjusted until model outputs matched field observations that comprise the model's history-matching dataset. In doing so, they became samples of the posterior parameter probability distribution.

Field observations of historical head and (in a limited number of cases) historical vertical head differences, are available from observation bores that are depicted in Figure 3.7.

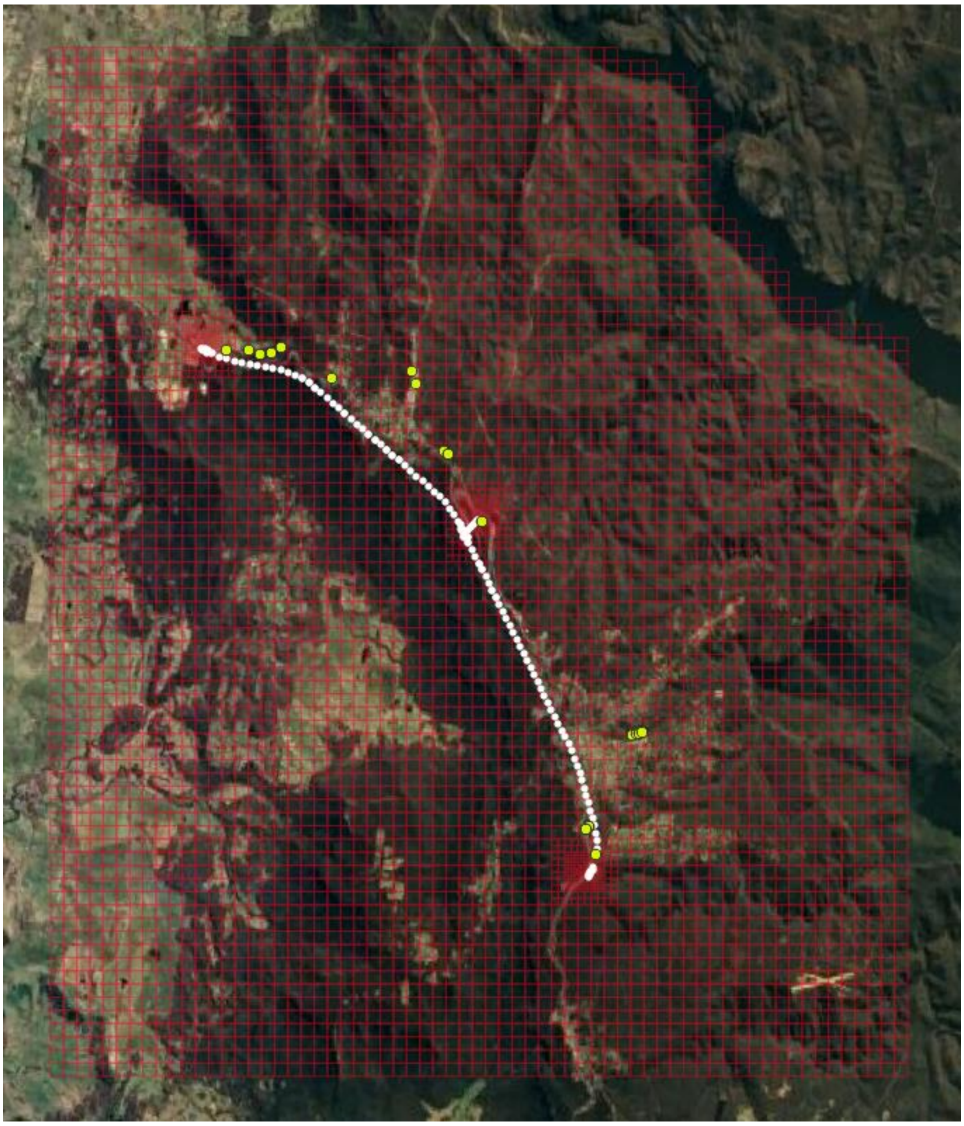


Figure 3.7. Boreholes from which head measurements are available for history-matching.

The history-matching dataset was expanded with a number of qualitative observations of water levels that reflect the existence of seepage features such as hanging swamps.

History-match-constrained parameter fields were used to make predictions of tunnel inflow and inflow-induced regional drawdown.

4. PARAMETERISATION

4.1 General

Only some aspects of the model's parameterisation are discussed in detail in this chapter, namely those that pertain to stratigraphic layering, and those that pertain to structural features that vertically intersect stratigraphic layering. Furthermore, the discussion focuses only on parameterisation of horizontal hydraulic conductivity. Parameterization of other spatially varying hydraulic properties follows a similar protocol. Fragments of a PLPROC script that implements this parameterisation are depicted. The entire script is provided in the Appendix.

4.2 PLPROC: A Few Words

PLPROC is a model pre-processor. It enables flexible and adjustable parameterisation of a groundwater model of arbitrary complexity. The model's grid can be structured or unstructured.

PLPROC defines and manipulates entities called LISTs. A CLIST is a set of two- or three-dimensional coordinates. One or a number of PLISTs, and one or a number of SLISTs, can be associated with a parent CLIST. A PLIST houses real numbers (normally parameter values) while an SLIST houses integers; the latter can be used for definition of zones or other quantities on which selection of sublists can be based.

CLISTs may represent:

- the cell centres of a model grid, or of a single layer of a model grid;
- the locations of pilot points.

A SEGLIST is a list of polylinear segments. These are used for representation of one-dimensional topographic features such as streams, of structural features such as faults, joints and shear-zones. A hydraulic property can be awarded to each segment, or to either end of each segment; in the latter case the hydraulic property undergoes linear interpolation along the length of the segment to points between its ends. Hydraulic properties assigned to segments can be transferred to nearby cells of a model grid using various types of line-to-point extrapolation; the value of the extrapolated property depends on the perpendicular distance between the grid point to which extrapolation takes place and the nearest point on a segment.

PLPROC functionality is accessible to a user through high level function calls. Processed PLISTs can be written to model input files using templates of those files which facilitate compliance with model input protocols. These templates can include embedded PLPROC functions.

4.3 Background Parameters

4.3.1 Description

Each layer of the model that was described in the previous section is awarded a set of pilot points. These support parameterisation of Kh, Kv, Ss and Sy (i.e. horizontal and vertical hydraulic conductivity, specific storage and specific yield). The distribution of pilot points reflects the disposition of active cells within each model layer. Pilot points associated with layers 5 and 20 are shown in plan view in Figure 4.1. (In retrospect, more pilot points could have been added to some parts of the model domain, even though these would have had limited impact on tunnel inflow predictions; these parts of the model domain include the south-western corner of layer 20. Extra parameters incur zero cost where history-matching is implemented using an ensemble smoother such as PESTPP-IES.)

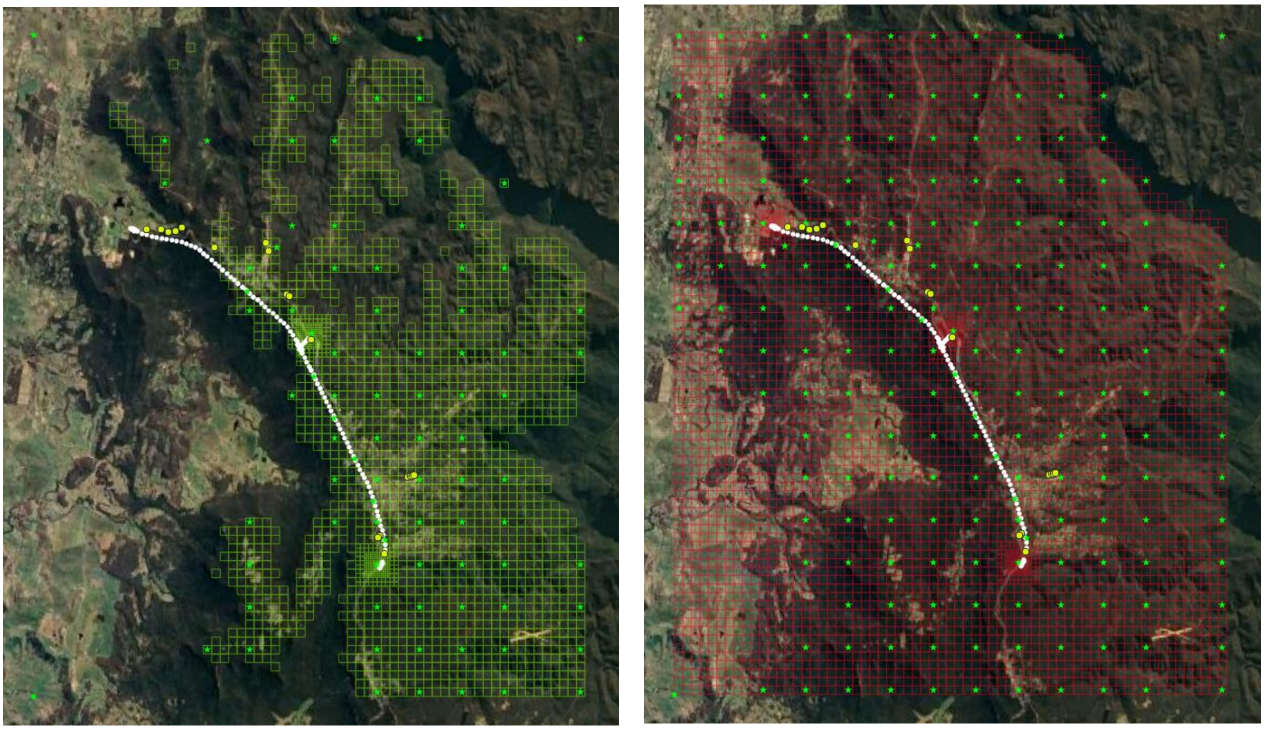


Figure 4.1. Disposition of pilot points in model layers 5 and 20. The tunnel, and groundwater head observation points, are also shown.

Pilot points associated with all model layers are depicted in three dimensions in Figure 4.2.

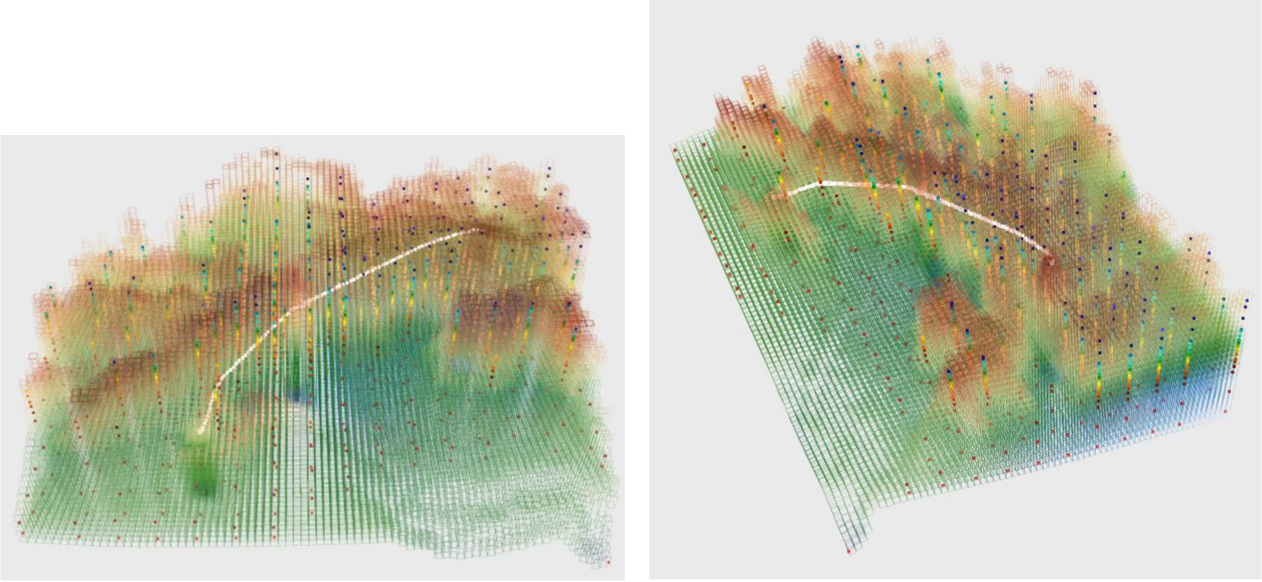


Figure 4.2. Pilot points used for parameterisation of layer hydraulic properties. Points are coloured according to the model layer in which they lie, from blue (layer 1) to red (layer 20). Views are towards the east (left) and north-east (right).

As stated above, parameterization of the model is stochastic. A prior mean parameter value is ascribed to each pilot point; this is a function of its depth and lithology. (Empirical functions which characterise depth-dependence of hydraulic properties for different rock types were derived as part of the modelling study.)

Layer-specific covariance matrices are employed to characterise intra-layer spatial variability of hydraulic property values about their mean values. Correlation lengths of about 3100 m were assumed for all layers.

4.3.2 Implementation

Pilot point specifications are usually provided to PLPROC in tabular data files. Columns within these files associate the following quantities with each pilot point:

- name,
- easting,
- northing,
- zone and/or layer number,
- hydraulic property #1,
- hydraulic property #2,
- etc

The names of PEST parameters are usually derived from the names of pilot points through an appropriate suffix or prefix. A PEST-compatible template file is normally made from a pilot points file so that programs of the PEST and PEST++ suites can award hydraulic properties to pilot points. A function call such as the following tells PLPROC to read the file.

```
cl_pp_kx = read_list_file(dimensions=2,                          &
                          slist=”kx_pp_zones”;column=4,          &
                          plist=kx;column=5,                     &
                          id_type=indexed,                       &
                          file=”points.dat”,                     &
                          skiplines=0)
```

The above call to function *read_list_file()* creates a pilot-point-based CLIST named *cl_pp_kx*. Two ancillary LISTs are also created. One of these is a PLIST which associates hydraulic conductivity values with all pilot points, while the other is an SLIST which associates zone numbers with all pilot points; zones coincide with model layers for all but the first model layer.

PLPROC must also be informed of the coordinates of all model grid cells so that it can interpolate from pilot points to the centres of these cells. For the tunnel model discussed herein, it obtains this information from a MFUSG grid specification file. This is a complex file (written by Groundwater Vistas) that provides not only the coordinates of grid cell centres; it also provides the coordinates of cell vertices. For an unstructured grid, the coordinates of cell vertices cannot be back-calculated from those of its centres; hence they must be explicitly provided in a grid specification file so that pictures of the model grid can be drawn, and so that grid details can be imported into programs such as SURFER, PARAVIEW and GIS platforms.

PLPROC reads a MFUSG grid specification file using the *read_mf_usg_grid_specs()* function. A typical call is as follows.

```
musg_grid = read_mf_usg_grid_specs(file=”GWHv4TR011.gsf”)
```

The above function call creates a CLIST named *musg_grid* which houses the coordinates of all model cell centres. The layer number of each model cell is automatically assigned to an SLIST named *layer*.

PLPROC provides several mechanisms for interpolation from pilot points to a model grid. Interpolation can be two-dimensional or three-dimensional. It can be based on inverse-power-of-distance, radial basis functions or simple/ordinary kriging. Pilot points can host native parameters, or multiplier/adder parameters that alter an existing model-based hydraulic property field.

For the model that is the focus of the current report, interpolation is two-dimensional because pilot points, and the parameters that are associated with them, are layer-specific. Interpolation is undertaken using ordinary kriging. This is a two-step process. First, layer-specific sets of interpolation factors are calculated; this needs to happen only once. Spatial interpolation from values ascribed to pilot points to the model grid can then be carried out using these interpolation factors. Interpolation factors do not change as the values ascribed to pilot points change.

The following PLPROC function call instructs PLPROC to calculate interpolation factors to cells in model layer 20 from pilot points assigned to zone 20. Interpolation factors are stored in file *factors20.dat*.

```
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==20),    &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==20),          &
                        file=factors20.dat,                                      &
                        kriging=ordinary)
```

Kriging (based on these factors) is implemented using the *krige_using_file()* function.

```
kx_model=kx.krige_using_file(file=”factors20.dat”,transform=log)
```

Through the above function, all cells of the model grid which belong to layer 20 are supplied with a value that is obtained through interpolation from pilot points that are associated with this layer. Calls to the above two functions are repeated for every model layer.

4.4 Polylinear Features of Hydrogeological Interest

Figure 4.3 depicts lineaments that have been identified within and near the study area. These may (or may not) indicate near-vertical features of enhanced horizontal and vertical hydraulic conductivity. Their presence may therefore impact tunnel inflow, as well as the distribution of inflow-induced drawdown.

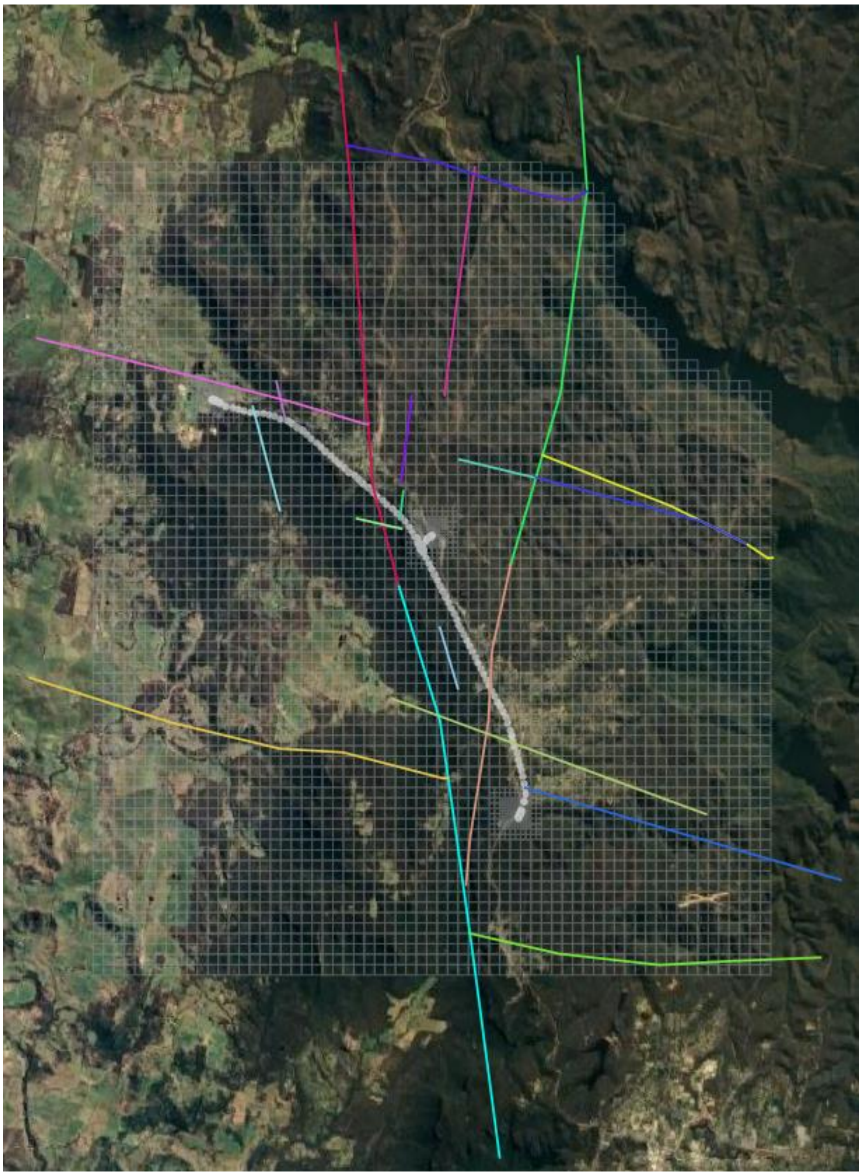


Figure 4.3. Lineaments which may indicate enhanced horizontal and vertical hydraulic conductivity.

Note that two of the roughly NS-trending lineaments that are featured in Figure 4.3 have been subdivided into two sublineaments for parameterisation purposes.

Two-dimensional polylinear features such as these can be represented in PLPROC using SEGLISTs. Each SEGLIST is comprise of one or a number of linear segments. Each segment is uniquely coloured in Figure 4.3. SEGLIST specifications can be read from a SEGFILE. Part of a SEGFILE is shown in Figure 4.4.

```
START SEGMENT
SEGMENT_ID  30_1
244649.045053779 6288265.31701487
244935.346847125 6285187.57273637
245235.586093823 6281079.51599046
245364.799537148 6279389.96142106
245865.827675504 6277457.42431596
END SEGMENT

START SEGMENT
SEGMENT_ID  30_2
245865.827675504 6277457.42431596
246653.157607208 6274880.70817582
246939.459400557 6272876.59562238
247297.336642239 6270299.87948225
247798.364780596 6266506.38072038
END SEGMENT

START SEGMENT
SEGMENT_ID  31_1
249301.449195656 6287621.13797983
249474.555998557 6285040.05620894
248969.99810029 6281177.85291499
248012.331854198 6277880.46464757
END SEGMENT
etc
```

Figure 4.4. Part of a SEGFILE.

As is apparent from Figure 4.4, a SEGFILE has an easily-understood protocol. It can be readily built from a SURFER BLN file; the latter can be constructed using SURFER digitisation functionality. A SEGFILE can be read using the PLPROC *read_segfile()* function as shown below.

```
sl_struc = read_segfile(file=”structures_seglist.dat”, protocol=block)
```

The above function creates a SEGLIST named *sl_struct* from segments that are defined in file *structures_seglist.dat*.

4.5 Parameterization of Polylinear Features

PLPROC provides a number of options for parameterisation of structural features that intersect many model layers. One option is to assign each of them to a three-dimensional zone that extends across the many layers that it intersects. A vertical, two-dimensional array of pilot points can then be assigned to the zone.

A simpler option was adopted in the present case as uncertainties in the properties of structural features are high. Hence their possible effects on groundwater flow can be represented using a simplistic parameterisation scheme that acknowledges their presence but does not require introduction of two-dimensional heterogeneity across their vertical planes. Vertical homogeneity of structure-pertinent hydraulic properties is therefore assumed. However their hydraulic properties can vary linearly along their horizontal lengths.

To enable linear variation of hydraulic properties along each SEGLIST segment, a pilot point can be assigned to each of its ends. PLPROC ensures that interpolation from pilot points that are linked to SEGLIST segments in this manner is linear with respect to distance along the segment, regardless of the whether the segment is composed of one or multiple subsegments, and regardless of the angles between these subsegments. The locations of pilot points that are linked to segment endpoints should coincide (at least roughly) with the ends of each segment so that PLPROC can create appropriate linkages. This is easily ensured by copying segment endpoint coordinates from the SEGFILE which defines these segments when creating a SEGLIST-related pilot points file. Hydraulic properties can be ascribed to pilot points through this same pilot points file; a template of this file enables PEST to adjust the values of pilot-point-associated hydraulic properties.

The following fragment of a PLPROC script employs the *read_list_file()* function to read a pilot points file named *structures_ppoints.dat*. A pilot points CLIST named *cl_pp_struc* is thereby created. Two values are associated with each pilot point, thereby creating two ancillary PLISTS. One of these PLISTS houses values for horizontal hydraulic conductivity while the other houses values for vertical hydraulic conductivity.

```
cl_pp_struc = read_list_file(skiplines=1,dimensions=2,         &
                             plist=kx_struc_pp;column=4,       &
                             plist=kz_struc_pp;column=5,       &
                             id_type=character,                &
                             file="structures_ppoints.dat")
```

Linkage of a pilot points CLIST to segments of a SEGLIST is implemented using the *link_seglist_to_clist()* function. This function locates the pilot point that is closest to the end of each SEGLIST segment. If the ends of two segments are close enough (or are joined), they can each be linked to the same pilot point. Thus a set of pilot points can collectively form a basis for polylinear interpolation over a set of complex polylinear segments. Once pilot-point-to-segment linkages have been created in this fashion, any left over pilot points are considered to indicate an error condition; this is reported to the user by *link_seglist_to_clist()*.

```
link_seglist_to_clist(seglist=sl_struc,                &
                      clist=cl_pp_struc,               &
                      linkage=endpoints,               &
                      max_dist=1.0e-3)
```

Figure 4.5 shows pilot points that are linked to the structural segment that are depicted in Figure 4.3. Two of these points are common to joined segments of the larger N-S oriented structures.

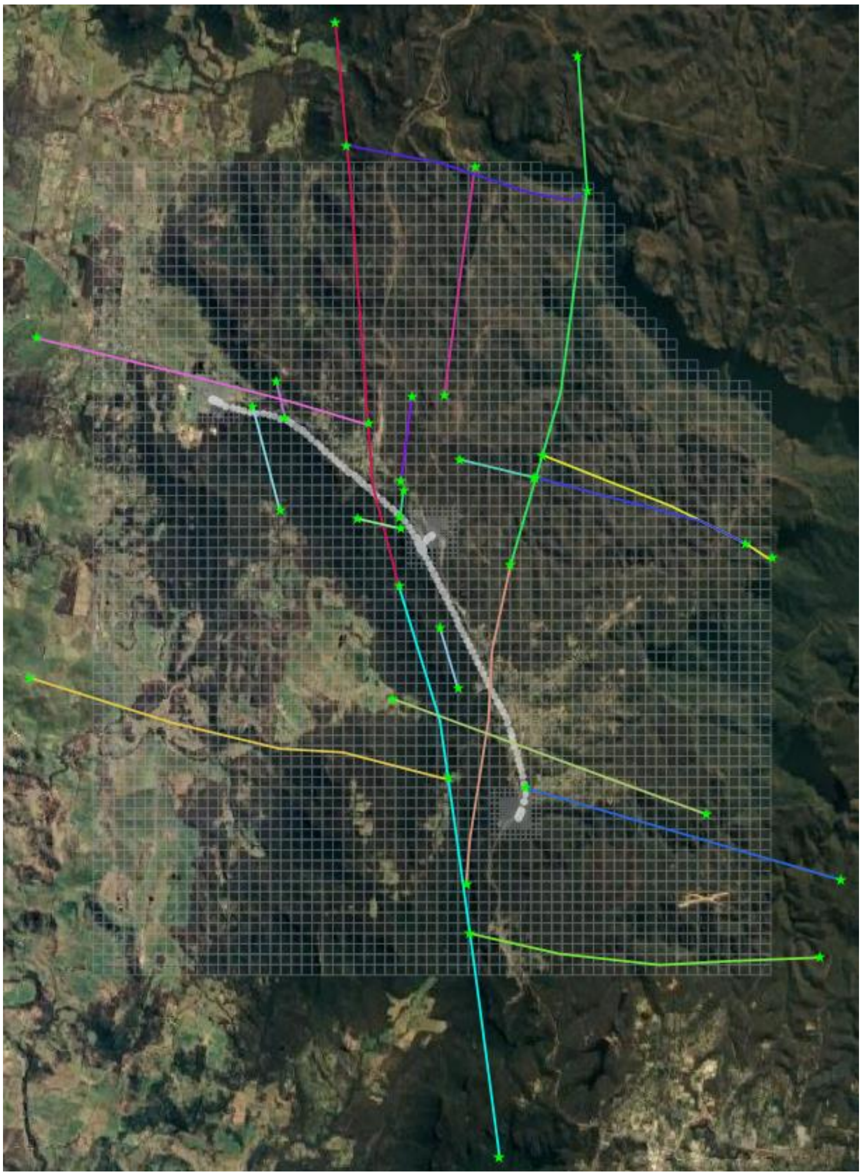


Figure 4.5. Structural segments together with pilot points that are linked to them.

4.6 Extrapolation from Segments to a Model Grid

As is described above, hydraulic property values are linearly interpolated along the length of each SEGLIST segment from pilot points that are linked to either end of the segment. However it is ultimately the model, and not the segment, to which hydraulic property values must be assigned. Furthermore, these must be assigned to individual cells of the model grid.

PLPROC employs a Gaussian function to transfer values from a polylinear segment to nearby cells of a model grid. If a model grid cell centre lies directly beneath a segment, then the value that is assigned to that cell is exactly equal to the value of the linearly-interpolated hydraulic property at the coinciding segment point. Hydraulic property values (or their logarithms) assigned to cell centres at greater distances from the segment decay to zero with increasing distance according to a Gaussian decay function. (Actually, as will be shown below, it is a “mixing factor” value rather than a hydraulic property value that is extrapolated in this manner.) The formulas are:

$$f = g \qquad \text{if } d \leq w/2 \qquad (4.1a)$$

$$f = g\ exp\left(-\left[\frac{d-w/2}{a}\right]^2\right) \qquad \text{if } d > w/2 \qquad (4.1b)$$

In the above equations:

d	is the perpendicular distance from a model cell centre to a SEGLIST segment;
g	is the value of the hydraulic property at the nearest point of a segment to the model cell;
f	is the value that is assigned to the model cell; and
a, w	are supplied by the user to govern decay rate of hydraulic property extrapolation with distance.

For the tunnel model that is describe herein, values of 100 m and 150 m are provided for *a* and *w*. The segment-to-grid extrapolation function is depicted in Figure 4.6.

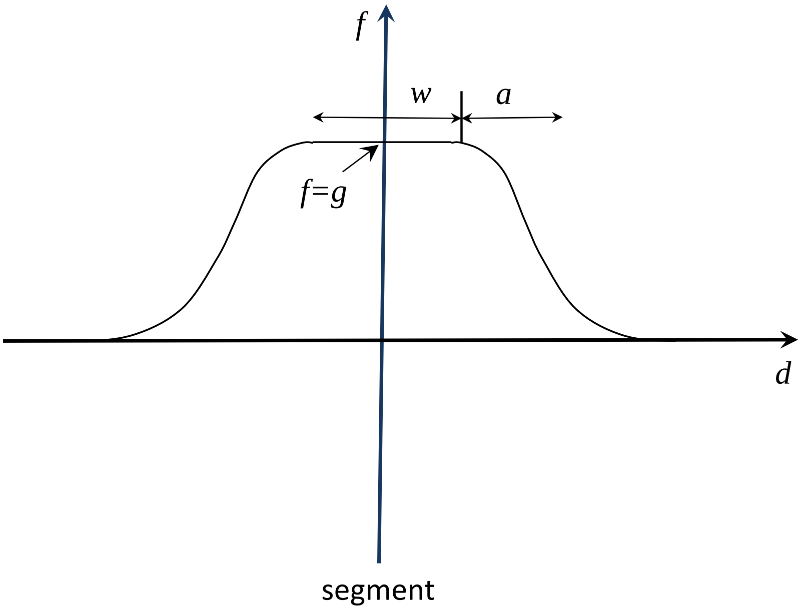


Figure 4.6. Extrapolation function from a SEGLIST segment to cells comprising a model grid.

Linear interpolation along SEGLIST segments from pilot point endpoints, and Gaussian extrapolation from segments to cells of a model grid, are undertaken using the PLPROC *calc_linear_gauss_interp_factors()* function. For the model that is the focus of this report, the function call is:

```
calc_linear_gauss_interp_factors(target_clist= musg_grid;select=(layer==20),   &
                                 source_clist=cl_pp_struc,                      &
                                 file=sfactors.dat,                             &
                                 dimensions=2,                                  &
                                 conwidth=150,                                  &
                                 a=100)
```

Notice that interpolation factors are computed only for layer 20. The reason for this will become clear in a moment. These factors are stored in file *sfactors.dat*.

Extrapolation factors that are calculated using this function are used to extrapolate two quantities from SEGLIST-linked pilot points to the model grid. These are:

- the logarithms of hydraulic conductivity values; and

- “mixing factors”.

Extrapolated values of the above quantities are assigned to dedicated model-based PLISTS. One more step is required in order to alter hydraulic properties that are used by the model; these are assigned to model-grid-associated PLISTs.

We now turn our attention to mixing factors. A PLPROC function call is used to assign a mixing factor value of 1.0 to each SEGLIST-linked pilot point that appears in Figure 4.5. Hence mixing factor values that are extrapolated to model cells all lie between 0.0 and 1.0, with a value of 1.0 being assigned to cells whose centres directly underlie SEGLIST segments. Function calls for mixing factor assignment, followed by distance-based extrapolation to model cells, are as follows.

```
shape_struc_pp=new_plist(reference_clist=cl_pp_struc,value=1.0)
shape_struc=shape_struc_pp.interp_using_file(file='sfactors.dat',       &
                                              transform=none,            &
                                              upper_limit=1.0,           &
                                              lower_limit=0.0)
```

Model-cell-extrapolated mixing factors are coloured according to value in Figure 4.7; mixing factor values of zero are denoted by transparent cells. Notice that the extrapolation distance from SEGLIST segments to the model grid is wide enough to ensure continuity of groundwater flow along structural features. However it is small enough to ensure that the effects of these structures fades rapidly with distance from them.

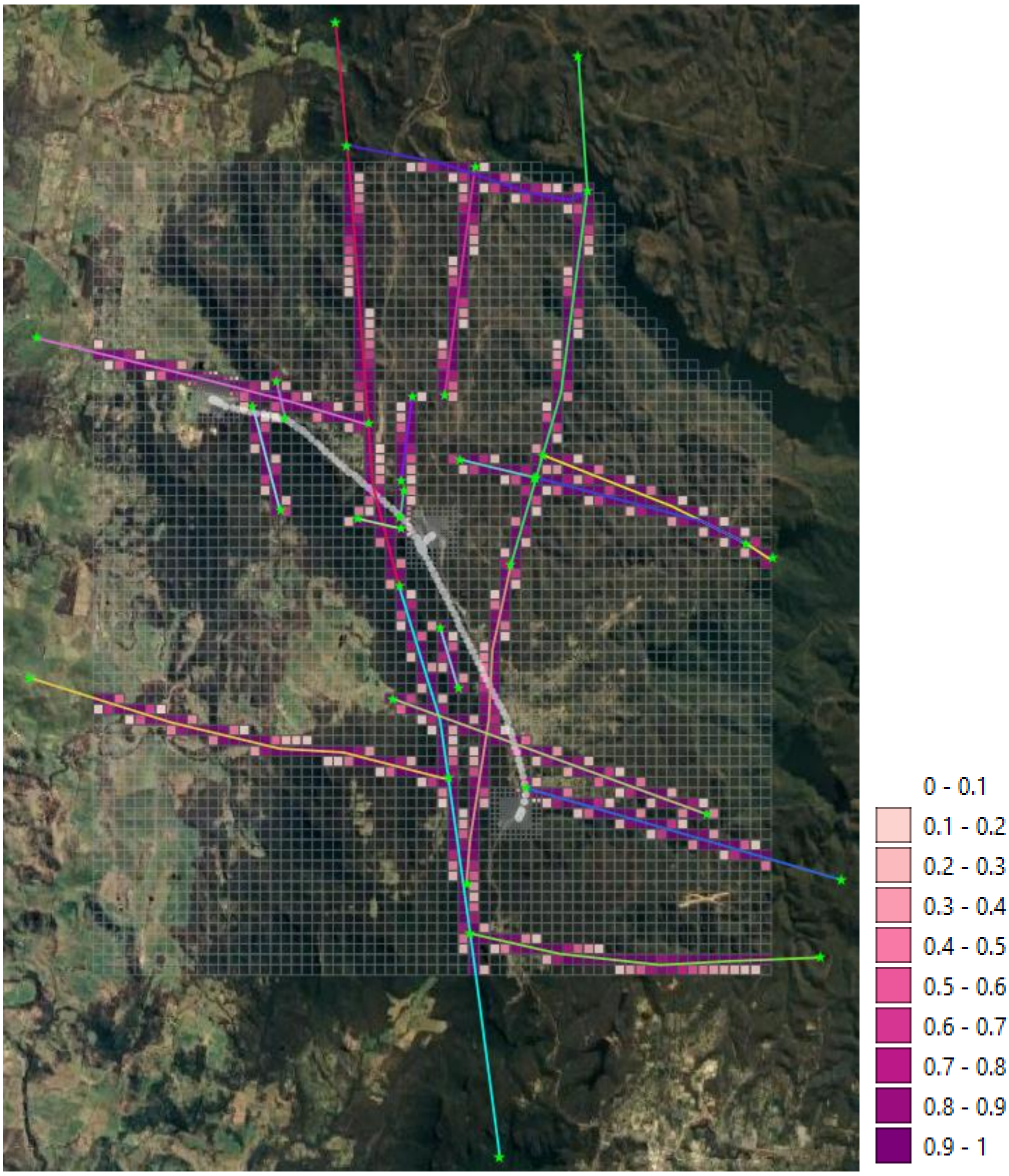


Figure 4.7. Mixing factors extrapolated from SEGLIST segments to cells of the model grid.

4.7 Structural Overlay

So far, hydraulic property and mixing factor extrapolation to the model grid has taken place only to layer 20 of this grid. Two problems remain. They are:

1. "blending" of structure-inherited hydraulic properties with stratigraphic background properties in layer 20.
2. transferal of this blending process upwards to all other layers of the model except for those parts of layer 1 that represent swamps, and that therefore postdate structural features.

A problem arises in linking cells in one layer of an unstructured model grid to cells in overlying layers. The node numbers of directly overlying cells are not obtained by sequentially subtracting the number of cells per layer from the node number of a cell in the lowest model layer, for this number is different from layer to layer; furthermore the node number of a cell may be unrelated to a cell's position within a layer. PLPROC provides a function through which these linkages can be made where overlying cells of an unstructured model grid share the same *x* and *y* coordinates. The function is *build_relational_slist()*. An example of its use follows.

```
sl_relate = build_relational_slist(target_clist=musg_grid,          &
            source_clist=musg_grid;select=(layer==20),              &
            no_relation_value=-999,                                 &
            epsilon=1.0)
```

In this call to function *build_relational_slist()* a model-based SLIST (i.e. an integer array with an element devoted to each model cell) is created in which the value assigned to each SLIST element is the node number of a layer 20 cell that directly underlies it.

Armed with this relational SLIST, the blending operation can now take place. This is done using function *graduated_replace()*. This function performs the blending operation between Kh values attributed to geological structures on the one hand, and background Kh values that pertain to layer stratigraphy on the other hand. Function *graduated_replace()* calculates new values of Kh for all cells in all layers of the model. In doing this, it combines structure-inherited Kh values ascribed to layer 20 with stratigraphy-derived Kh values that already exist for these layers, using mixing factors ascribed to layer 20. Because these mixing factors decay rapidly to zero with distance from each SEGLIST segment, cell Kh values are altered only if they lie close to geological structures.

The PLPROC function call is:

```
kx_model=graduated_replace(source_plist=kx_struc,                          &
                     mixing_plist=shape_struc,                             &
                     relational_slist=sl_relate;no_relation_value=-999,    &
                     transform=log)
```

Model parameterization of horizontal hydraulic conductivity is now almost complete. Recall that model layer 1 contains two zones, one of which pertains to recent swamp deposits whose hydraulic properties are unaffected by structure. Horizontal interpolation from pertinent pilot points to cells within this zone can now take place. The outcomes of this interpolation process overwrite the SEGLIST extrapolation process discussed above.

That part of the PLPROC script which pertains to Kh parameterisation of the entire model is reproduced in its entirety in the Appendix of this report.

4.8 Could This Have Been Made Easier?

This process could have been made somewhat easier through use of higher level function calls than those which PLPROC presently provides. A function named *mix_using_file()* could replace *interp_using_file()*. This would allow SEGLIST interpolation/extrapolation and mixing of SEGLIST-interpolated hydraulic properties with those already ascribed to model cells through horizontal, stratigraphy-based interpolation to be undertaken in a single operation. Extrapolation factors could then be calculated separately for each layer of the model grid. This would dispense with the need to construct a relational SLIST which links each cell in layer 20 to all model cells which overly it.

This function will be added to PLPROC in the near future.

It is worth noting that use of a fully unstructured (e.g. Voronoi-style) model mesh could have comprised an alternative means of representing the structural features shown in Figure 4.3. This would have allowed more explicit representation of their geometries (to the extent that these can be known or inferred). A disadvantage of such explicit representation, however, is that these features are then ‘baked in’ to the model. In contrast, their representation using PLPROC structural overlay functionality is independent of the model grid. Being flexible, it is readily altered if new structural features are introduced, or if grid specifications are altered.

4.9 Some Pictures

As has already been discussed, history-matching of the tunnel model was performed using the PESTPP-IES iterative ensemble smoother. An iterative ensemble smoother adjusts an ensemble of realisations of prior parameter values until they become samples of the posterior parameter probability distribution. They do this when pertinent model outputs match corresponding field measurements of system behaviour. Once they achieve this status, the ensemble of parameter realisations can be used to make probabilistic predictions of future system behaviour.

This report does not discuss predictions made by the tunnel model. So we finish this chapter with a series of pictures of realisations of posterior Kh. Three sets of pictures are provided, each featuring four realisations of posterior Kh. One of these realisations is the so-called “base realisation”. PESTPP-IES derives realisations other than the base realisation by adjusting random samples of the prior parameter probability distribution. However the base realisation is obtained by adjusting prior parameter mean values.

The colour scale for Kh fields is provided in Figure 4.8. All views are towards the east. The view from the underside of the model clearly shows the importance of structural features in its stochastic parameterisation.

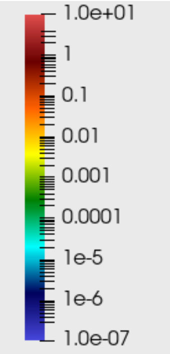


Figure 4.8. Legend used for specification of horizontal hydraulic conductivity (Kh) in Figures 4.9 to 4.11. Units of hydraulic conductivity are m/day.

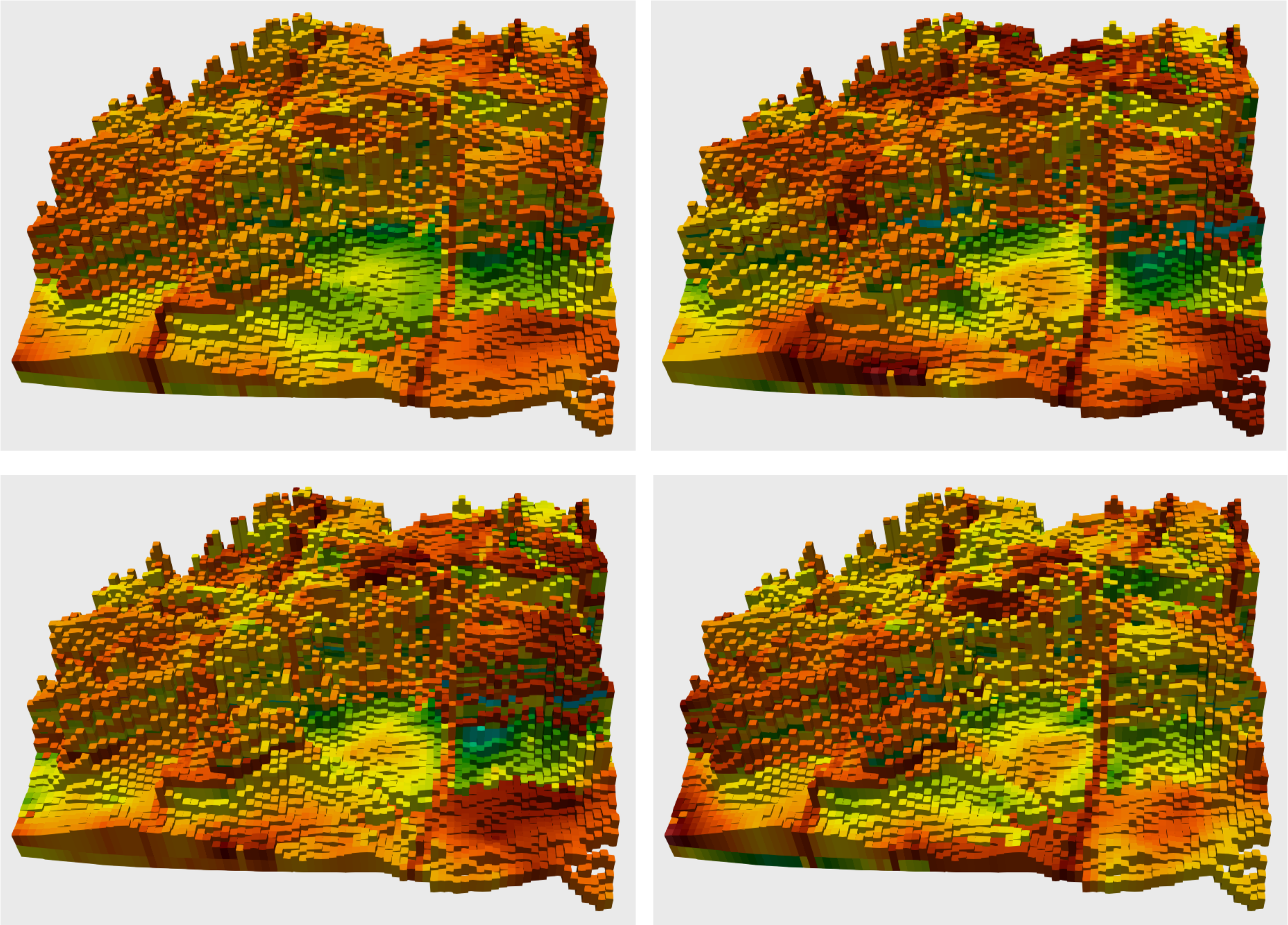


Figure 4.9. Four realisations of posterior log Kh. The first is the base realisation. View is towards the east.

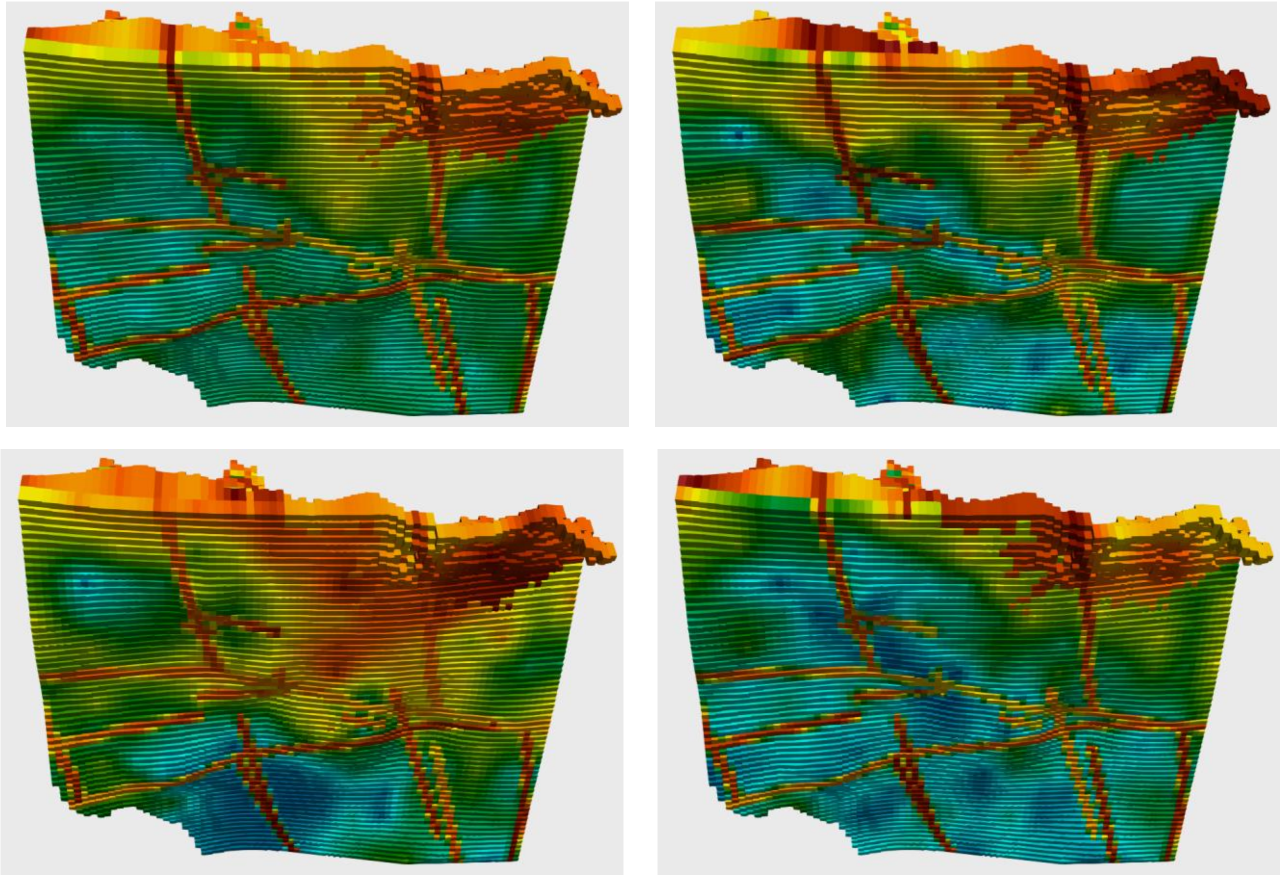


Figure 4.10. Four realisations of posterior log Kh. The first is the base realisation. View is of the underside of the model, looking east.

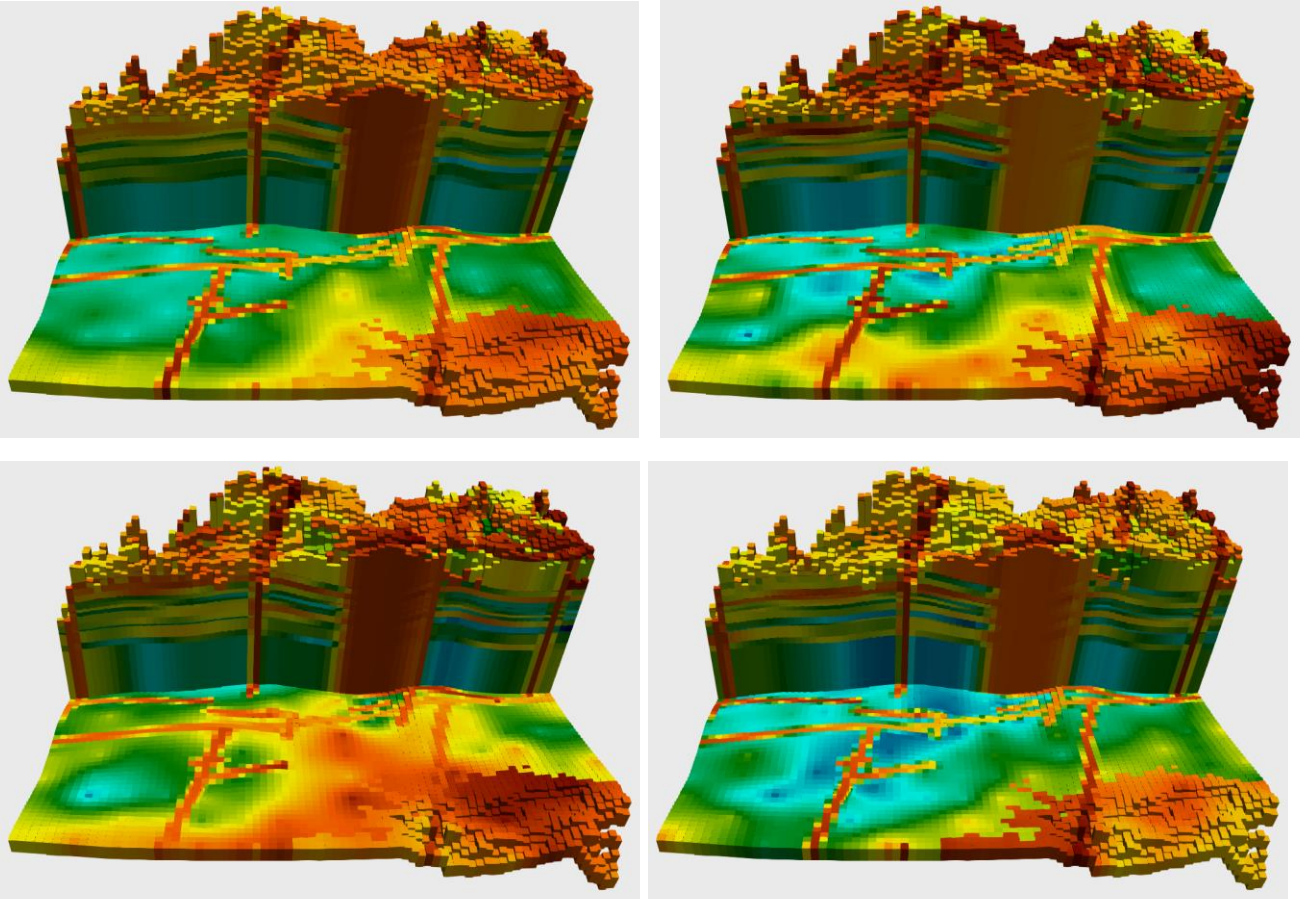


Figure 4.11. Four realisations of posterior log Kh. The first is the base realisation. View is towards the east.

5. CONCLUSIONS

This short GMDSI worked example report focusses on one aspect of a complex, multi-layer model. This is the design of a parameterisation scheme that reflects the presence of structural features of enhanced hydraulic conductivity. These planar features may direct flow of groundwater to excavations that will emplace a tunnel into a layered sequence of sedimentary rocks. They may also determine the distribution of drawdown that is induced by this inflow.

Parameterisation of the model is stochastic; it is based on pilot points. Subsets of pilot points represent hydraulic properties of different model layers. Another subset is dedicated to representation of the hydraulic properties of the vertical planar features that permeate stratigraphic layering. Parameters that are associated with all pilot points are history-match adjustable.

Pilot points parameterisation is implemented using the PLPROC parameter pre-processor that complements the PEST suite of programs. PLPROC can accommodate both structured and unstructured model grids. The tunnel model that is discussed herein employs an unstructured grid; simulation of groundwater flow is performed using MODFLOW-USG.

Production of the pictures that are shown in this report was accomplished using programs of the PEST Groundwater Utility suite. The programs that were used, and a brief description of their functions, are listed in Table 5.1. Two dimensional plots that are presented in this report were drawn in QGIS. Three-dimensional plots were produced by PARAVIEW.

Program	**Function**
PT2VTK	Reads coordinates of points, and data pertaining to points, from a tabular data file. Writes a VTK file for visualization of point data.
USG2VTK1	Creates a VTK file which displays cells of a MODFLOW-USG model, together with data that are associated with these cells.
USGGRIDLAY	Reads data of any type pertaining to a single layer of a MODFLOW-USG model. Records these data as a MIF/MID file pair for importation into a GIS.
USGMODGSF	Reads a MODFLOW-USG grid specification file. Writes another such file in which cell tops and bottoms are flat.
USGPTINGRID	Locates MODFLOW-USG cells within which user-supplied points lie.

Table 5.1. Programs of the Groundwater Utility Suite which were used to produce pictures that are featured in this report.

6. REFERENCES

Panday, S., 2022. MODFLOW-USG-Transport (v1.9.0). Techniques and Methods.

Panday, S., 2021. USG-Transport Version 1.8.0: Block-centred Transport (BCT) Process for MODFLOW-USG, v. 1.8.0. ed, Techniques and Methods, Book 6, Chapter A45. GSI Environmental.

Panday, S., Langevin, C.D., Niswonger, R.G., Ibaraki, M., Hughes, J.D., 2013. MODFLOW–USG Version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation. Techniques and Methods, Book 6, Chapter A45. U.S. Geological Survey.

Watershed Hydrogeo, 2020. Great Western Highway – Blackheath to Little Hartley. Groundwater Modelling report for Environmental Impact Statement.

White, J.T., 2018. A model-independent iterative ensemble smoother for efficient history-matching and uncertainty quantification in very high dimensions. *Environ Model Softw.*, 109, 191-201.

Zaidel, J., Markham, B., Bleiker, D., 2010. Simulating Seepage into Mine Shafts and Tunnels with MODFLOW. *Ground Water* 48, 390–400. https://doi.org/10.1111/j.1745- 6584.2009.00659.

APPENDIX

The contents of a PLPROC input file that implements functionality that is described in the body of this report follows. This is a heavily abridged version of the PLPROC file that was used in conjunction with the original tunnel model. It omits:

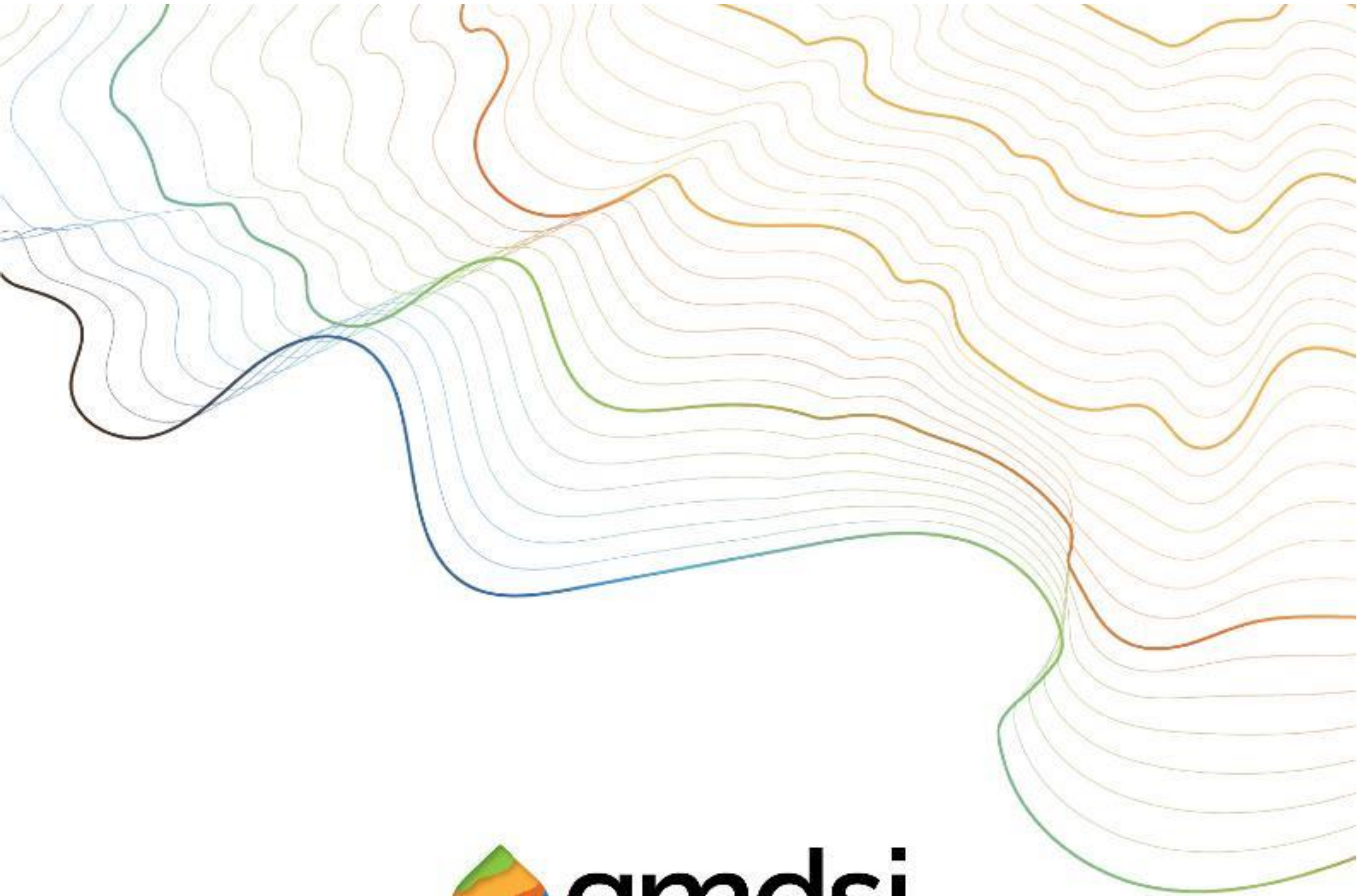
- parameterisation of vertical hydraulic conductivity and storage;
- some details pertaining to parameterisation of the surficial model layer.

```
# -- This demonstration PLPROC file only cites Kx parameters. It is an abridgement of a
#    longer file that was originally used for parameterisation of Kh, Kz, Ss, and Sy.
#    Note that it also omits some of the complexities of parameterisation of the top model
#    layer that were included in the original PLPROC file.

# -- Read a MODFLOW-USG grid specification file. This was produced by Groundwater Vistas
#    This produces a CLIST named m

musg_grid = read_mf_usg_grid_specs(file=GWHv4TR011.gsf)

# -- First handle the structures.
#    Create a SEGLIST for structures. This is named sl_struc.

sl_struc = read_segfile(file="structures_seglist.dat", protocol=block)

# -- Read a CLIST and associated PLISTS for structure pilot points.
#   (Note that, for convenience, CLIST coordinates correspond to segment endpoints.)

cl_pp_struc = read_list_file(skiplines=1,dimensions=2,         &
                             plist='kx_struc_pp';column=4,     &
                             id_type='character',              &
                             file='structures_ppoints.dat')

# -- Now link these points to the SEGLIST

link_seglist_to_clist(seglist=sl_struc,               &
                      clist=cl_pp_struc,              &
                      linkage='endpoints',            &
                      max_dist=1.0e-3)

# -- In following functions we will interpolate to the bottom model layer as it has its
#    full complement of cells. So we create linkages from this layer to all overlying layers.
#    An SLIST, whose parent CLIST is musg_grid (i.e. model cells) is built.

sl_relate = build_relational_slist(target_clist=musg_grid,          &
            source_clist=musg_grid;select=(layer==20),              &
            no_relation_value=-999,                                 &
            epsilon=1.0)

# -- Calculate extrapolation factors from SEGLIST segments to layer 20 of the grid.
#    These are stored in a file named sfactors.dat. This function call can be
#    commented out once this file has been created.

calc_linear_gauss_interp_factors(target_clist= musg_grid;select=(layer==20),   &
                                 source_clist=cl_pp_struc,                     &
                                 file=sfactors.dat,                            &
                                 dimensions=2,                                 &
                                 conwidth=150,                                 &
                                 a=100)

# -- Create a "mixing PLIST" for the graduated replacement function.
#    First assign all SEGLIST-related pilot points a value or 1.0.

shape_struc_pp=new_plist(reference_clist=cl_pp_struc,value=1.0)

# -- Now interpolate to layer 20 of a dedicated model-based PLIST.

# -- Now extrapolate to dedicated PLISTs.
#    First we create a model-based CLIST named shape_struct which holds the shaping function

shape_struc=new_plist(reference_clist=musg_grid,value=0.0)
```

```
shape_struc=shape_struc_pp.interp_using_file(file='sfactors.dat',       &
                                             transform=none,            &
                                             upper_limit=1.0,           &
                                             lower_limit=0.0)

# -- Now extrapolate kx values from the SEGLIST to layer 20 of a model-based PIST named kx_struc.

kx_struc=new_plist(reference_clist=musg_grid,value=1.0)
kx_struc=kx_struc_pp.interp_using_file(file='sfactors.dat',transform=log)

# -- We next turn our attention to pilot points parameterisation of model layers
#    Recall that model layers are based on stratigraphic layers.

# -- Read Kx values associated with pilot points for all layers.
#    These files also contain layer numbers. This creates a pilot-point-based
#    CLIST named cl_pp_kx and ancillary PLISTS.

cl_pp_kx = read_list_file(dimensions=2,                          &
                          slist='kx_pp_zones';column=4,          &
                          plist=kx;column=5,                     &
                          id_type=indexed,                       &
                          file='points.dat',                     &
                          skiplines=0)

# -- Read a file in which cells of the model are allocated zones. Mostly these zones correspond
#    to model layers. However the first layer of the model requires 2 zones as one of these
#    zones represent swamps.

read_list_file(reference_clist=musg_grid,slist='kx_node_zones';column=2,              &
               file=kx_zones.dat,skiplines=0)

# -- Calculate interpolation factors from pilot points to model grid cells in each
#    zone. Function calls which create interpolation factor files can be
#    commented out once these files have been built.

calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==2),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==2),             &
                        file=factors2.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==3),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==3),             &
                        file=factors3.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==4),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==4),             &
                        file=factors4.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==5),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==5),             &
                        file=factors5.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==6),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==6),             &
                        file=factors6.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==7),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==7),             &
                        file=factors7.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==8),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==8),             &
                        file=factors8.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==9),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==9),             &
                        file=factors9.dat,                                         &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==10),    &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==10),            &
                        file=factors10.dat,                                        &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==11),    &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==11),            &
                        file=factors11.dat,                                        &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==12),    &
```

```
                        source_clist=cl_pp_kx;select=(kx_pp_zones==12),             &
                        file=factors12.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==13),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==13),             &
                        file=factors13.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==14),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==14),             &
                        file=factors14.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==15),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==15),             &
                        file=factors15.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==16),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==16),             &
                        file=factors16.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==17),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==17),             &
                        file=factors17.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==18),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==18),             &
                        file=factors18.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==19),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==19),             &
                        file=factors19.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==20),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==20),             &
                        file=factors20.dat,                                          &
                        kriging=ordinary)
calc_kriging_factors_auto_2d(target_clist=musg_grid;select=(kx_node_zones==21),     &
                        source_clist=cl_pp_kx;select=(kx_pp_zones==21),             &
                        file=factors21.dat,                                          &
                        kriging=ordinary)

# -- Now interpolate from pilot points to layers of the model grid.

kx_model=new_plist(reference_clist=musg_grid,value=1.0)
kx_model=kx.krige_using_file(file='factors2.dat',transform=log)
kx_model=kx.krige_using_file(file='factors3.dat',transform=log)
kx_model=kx.krige_using_file(file='factors4.dat',transform=log)
kx_model=kx.krige_using_file(file='factors5.dat',transform=log)
kx_model=kx.krige_using_file(file='factors6.dat',transform=log)
kx_model=kx.krige_using_file(file='factors7.dat',transform=log)
kx_model=kx.krige_using_file(file='factors8.dat',transform=log)
kx_model=kx.krige_using_file(file='factors9.dat',transform=log)
kx_model=kx.krige_using_file(file='factors10.dat',transform=log)
kx_model=kx.krige_using_file(file='factors11.dat',transform=log)
kx_model=kx.krige_using_file(file='factors12.dat',transform=log)
kx_model=kx.krige_using_file(file='factors13.dat',transform=log)
kx_model=kx.krige_using_file(file='factors14.dat',transform=log)
kx_model=kx.krige_using_file(file='factors15.dat',transform=log)
kx_model=kx.krige_using_file(file='factors16.dat',transform=log)
kx_model=kx.krige_using_file(file='factors17.dat',transform=log)
kx_model=kx.krige_using_file(file='factors18.dat',transform=log)
kx_model=kx.krige_using_file(file='factors19.dat',transform=log)
kx_model=kx.krige_using_file(file='factors20.dat',transform=log)

# -- Overwrite with structure.

kx_model=graduated_replace(source_plist=kx_struc,                             &
                     mixing_plist=shape_struc,                                &
                     relational_slist=sl_relate;no_relation_value=-999,       &
                     transform=log)

# -- Overwrite with swamp properties. (Swamps are younger than regolith and structures)

kx_model=kx.krige_using_file(file='factors21.dat',transform=log)

# -- Now write MODFLOW-USG model input files. One file is expected for each layer.
```

```
write_model_input_file(template_file=kx1.tpl, model_input_file=GWHv4TR0111._kx)
write_model_input_file(template_file=kx2.tpl, model_input_file=GWHv4TR0112._kx)
write_model_input_file(template_file=kx3.tpl, model_input_file=GWHv4TR0113._kx)
write_model_input_file(template_file=kx4.tpl, model_input_file=GWHv4TR0114._kx)
write_model_input_file(template_file=kx5.tpl, model_input_file=GWHv4TR0115._kx)
write_model_input_file(template_file=kx6.tpl, model_input_file=GWHv4TR0116._kx)
write_model_input_file(template_file=kx7.tpl, model_input_file=GWHv4TR0117._kx)
write_model_input_file(template_file=kx8.tpl, model_input_file=GWHv4TR0118._kx)
write_model_input_file(template_file=kx9.tpl, model_input_file=GWHv4TR0119._kx)
write_model_input_file(template_file=kx10.tpl,model_input_file=GWHv4TR01110._kx)
write_model_input_file(template_file=kx11.tpl,model_input_file=GWHv4TR01111._kx)
write_model_input_file(template_file=kx12.tpl,model_input_file=GWHv4TR01112._kx)
write_model_input_file(template_file=kx13.tpl,model_input_file=GWHv4TR01113._kx)
write_model_input_file(template_file=kx14.tpl,model_input_file=GWHv4TR01114._kx)
write_model_input_file(template_file=kx15.tpl,model_input_file=GWHv4TR01115._kx)
write_model_input_file(template_file=kx16.tpl,model_input_file=GWHv4TR01116._kx)
write_model_input_file(template_file=kx17.tpl,model_input_file=GWHv4TR01117._kx)
write_model_input_file(template_file=kx18.tpl,model_input_file=GWHv4TR01118._kx)
write_model_input_file(template_file=kx19.tpl,model_input_file=GWHv4TR01119._kx)
write_model_input_file(template_file=kx20.tpl,model_input_file=GWHv4TR01120._kx)
```



gmdsi

Groundwater Modelling
Decision Support Initiative

gmdsi.org

CRICOS NO 00114A

BHP

Flinders
UNIVERSITY

NATIONAL CENTRE FOR
GROUNDWATER
RESEARCH AND TRAINING

RioTinto