MATATAUIRA: A GROWING SOFTWARE PACKAGE FOR NUMERICAL GEOTHERMAL RESERVOIR SIMULATIONS

Peter Franz[1]

[1]Mercury NZ Ltd, 283 Vaughan Road, Rotorua, NZ

peter.franz@mercury.co.nz

Keywords: *coupled reservoir simulator, graphical user interface, streamline analysis*

ABSTRACT

Numerical reservoir simulations are a very useful tool in geothermal reservoir engineering. However existing simulators were often developed for scientific rather than engineering purposes. This limits their usability to operators within the geothermal community – operators require easy-to-use, reliable tools which can provide quick, accurate answers to a multitude of questions.

At Mercury we have spent significant time to develop a user interface and bring features from different simulation packages together to form a flexible, easy to use tool for the geothermal reservoir engineer. MataTauira is a combined pre- and post-processing tool to run reservoir simulations either under a modified TOUGH2 code or using the in-house development Tauira. It features gridding of the model, population of the grid from a 2D grid-independent conceptual editor, visual run-control and post-processing either in 3D or along 1D welltracks. More complex post-processing and automatization can be achieved using customized Python scripts on broadly supported file systems (XML, HDF5). Important field data can be stored inside the model for comparison and calibration. Automated report generation is supported.

Geothermal wells can be easily implemented and are automatically coupled to the TOUGH2/Tauira reservoir simulator using our in-house wellbore simulator Paiwera. Pressures and flow through complex surface networks can be modelled and automatically adjusted to changing demand targets. Makeup wells can be brought online when existing wells can't satisfy the targets.

MataTauira has been a great success at Mercury to transfer our existing TOUGH2 models into one coherent system and has already been used to develop a new full-field model. We intend to continue building more functionality into the system with streamline analysis, reactive transport and geomechanic simulations envisaged as next steps.

1. INTRODUCTION

TOUGH2 (Pruess, Oldenburg, & Moridis, 1999) has been the industry standard tool for running geothermal reservoir simulations for a long time. Despite its wide acceptance in the industry the basic simulator lacks many features which are useful for geothermal operators. In particular its input/output files are hard to set up and analyze; since they do not contain the full geometric information of the underlying mesh 3D reconstructions of the output always require additional information.

There are a small number of preprocessing tools available for TOUGH2, for example PetraSim and Leapfrog. However since these are closed-code it is not possible to create additional features.

PyTough (Croucher, 2013) is a very useful scripting tool based on the Python programming language and enables script-based pre- and post-processing of TOUGH2 models. We have found PyTough very useful for working with TOUGH2 based models but the diversity of TOUGH2 variations in combination with different pre-processing tools made it very hard to write scripts which can be ported between different reservoir models.

From a geothermal operator's point of view the most important features missing in TOUGH2 are the lack of proper wellbore and surface network simulators. Geothermal wells and surface equipment represent capital worth many million dollars, and operators are very interested to see how individual wells will perform in future scenarios. Typical questions put to the modeler are often the number of makeup wells required and their timing, or how changes in plant equipment or surface network constraints limit the capability of a geothermal field.

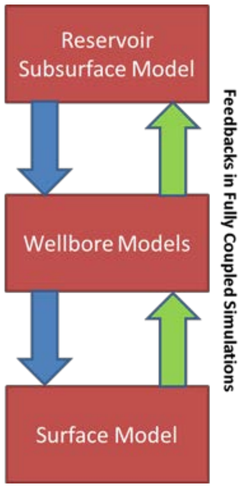

Figure 1: Logical flow of information between reservoir, wellbore and surface model. Traditionally the feedback mechanisms were largely ignored.

The logical flow of information in a model for a geothermal operation is shown in figure 1. Traditionally most emphasis has been given to the reservoir subsurface model. It was usually run through a calibration period where "typical" mass flow rates were assigned to the feedzones of the wells. Depending on the level of sophistication these "typical" rates were just guesses – for example a certain fraction of the total observed rate at the wellhead – or were manually checked with a wellbore simulator, which is very labor intensive. The final state after the calibration period was then used to run a production scenario, usually making additional simplifications.

This traditional calibration process is seriously flawed since it does not fully account for the feedback mechanisms. Changes in the thermodynamics in the reservoir change the behavior of the wells, in particular their feedzone allocations vary over time. Using fixed/guessed mass rate allocations during the calibration period can hence lead to a very different reservoir state at the end of the calibration period versus the correct state which would consider these variations. We have found in several models in the past that the reservoir state found with this traditional approach would not even support a well to flow – this of course makes accurate prediction about future makeup wells very difficult. We have also found that including fully coupled wellbore simulations during the calibration period can actually make the calibration process easier since the manual allocations in the past introduced artificial behavior into the model.

Also the feedback mechanism from the surface network was often treated too simplistic. Scenarios were often run over multiple decades using a constant production flow rate. This does not take changes in enthalpy into effect - changes in enthalpy can have large effects on the mass rates required to run a geothermal power station. Also changes in well characteristics may require a shift in production/injection locations and use additional makeup wells. A "smart" surface network simulator hence must be able to autonomously adapt the flow from the wells to meet production targets during a scenario run.

To overcome these problems Mercury has made a strong effort into developing new software tools and to improve existing ones. The result is the MataTauira software package, which features:

- fully coupled reservoir/wellbore/surface simulations
- graphical user interfaces to easily set up models
- data analysis and visualization in 3D
- auxiliary applications

The surface network simulator "Mata" and the wellbore simulator "Paiwera" (Franz, 2015) were fully developed in-house. The reservoir simulator TOUGH2 was modified to enable coupled simulations and some additional features were added. The setup of the reservoir model part is handled nearly simulator independently, i.e. it is very easy to switch to another reservoir simulator provided it has similar features to TOUGH2. A new reservoir simulator "Tauira" is still under development in-house and will hopefully soon replace TOUGH2 in the software package. It may also be possible to use the reservoir simulator currently developed by the University of Auckland and GNS once it has been released.

2. MODEL DEVELOPMENT

2.1. Gridding

The development of a geothermal model in MataTauira typically starts with creating the grid mesh. The user has the choice between a recti-linear, triangular, radial or Central-Voronoi meshes. Already existing meshes can be imported from a VTK[1] unstructured grid. It is also possible to have an undulated surface layer.

Currently MataTauira only supports vertical layers in parallel planes; it was found that this restriction allows for easier navigation in the 3D viewers. However it is a minor restriction and could be removed in the future if the need arose.

The mesh processor establishes all geometric and topological information, like block volumes and connection parameters. All 3D information about the mesh is stored in a hierarchical data file (HDF[2]) from where it is easily accessible.

2.2. Material Properties

The definition of materials in MataTauira is similar to TOUGH2 but has additional features.

In MataTauira each rock type is linked individually to capillary pressure and relative permeability functions. Absolute permeabilities are defined as proper tensors by specifying the 3 values of the tensor diagonal and supplying 3 rotation angles. Each connection in the mesh evaluates its individual permeability from this tensor using its own direction. Since this is not a feature supported by TOUGH2 one of the alterations to the code was to allow TOUGH2 to read two more parameters (k1, k2) for each connection and use these values when calculating fluxes.

TOUGH2 assigns a rock type to each computational element. MataTauira does so as well but introduces the concept of "compounds". While internally still each computational element is assigned to a rock type it is the compounds which are assigned to the geometric blocks in the mesh. Each compound can contain one or more rock types and describes their internally layered structure using the MINC approach. For example a compound could contain 2 layers (fracture and matrix); the user specifies which rock types to assign to each layer, what volume fraction each layer has and (x/y/z) fracture spacing. Since blocks are filled with compounds they will at run time generate the required number of computational elements for the layers and connect them up properly, both internally and externally to the neighboring blocks.

2.3. Conceptual Editor

The main workbench for creating a reservoir model is the conceptual model editor. It was found to be too hard from a user's perspective to edit shapes for a model in 3D. Hence the decision was made to use a 2D editor working in the (x,y) plane. To aid the modeler the editor can display the model grid and welltracks projected to the (x,y) plane. Pictures can serve as overlays.

The conceptual model is edited via creating polynomial shapes; these 2D shapes can be transformed to 3D by extrusion along the z-axis and can be rotated around an origin and 3 axes. However the rotation feature is rarely ever needed. Each conceptual shape can be used in one or more ways to set properties of the model. A hierarchical top-down model is used to resolve conflicts between shapes.

[1] Visualization Tool Kit, http://www.vtk.org/
[2] The HDF Group, https://www.hdfgroup.org/HDF5/

Shapes can be filled with compounds, which is their primary use. But they can also be used to set the state of a block (i.e. enabled or fixed state), or to modify the permeability of individual connections on a sub-grid level (barriers between certain rock types, and faults). They can act as primary variable setters for the regions covered by them and can be used for defining sources and sinks.

A very important aspect when using shapes to define sources/sinks is to correctly calculate the interaction between the shape the reservoir grid, i.e. what fraction of the shape is located in which reservoir block. Being able to properly calculate the interaction makes it possible to have grid-independent conceptual models – if a new mesh is calculated the conceptual model does not have to change at all, only the interactions are recalculated. It was found that calculating these interactions accurately and efficiently can be very hard to achieve; while very good and stable results were achieved for 0D, 1D and 2D shapes a restriction had to be made to use only convex shapes in 3D. However most sources/sinks can be set in 2D.

Commonly used boundary conditions like an atmosphere at the top or a hot plate at the bottom of a model can be set separately from the conceptual editor since they do not require a visual representation.

2.4. Surface Network and Wells

The steps described above are sufficient to set up a model to simulate the subsurface flow of fluid and heat.

Wells and surface kit objects are created and connected up to each other using a symbolic editor. Available objects are different types of wells, unions, splitters, pipes with friction losses, valves separators and diverse power plants.

Wells are defined using a table containing the (x,y,z) coordinates of the welltrack. An intersection algorithm accurately identifies the intersection points between the 1D welltrack and the faces of the 3D grid cells. Feedzone locations are determined and the block containing the feedzone is identified. Vertical locations are automatically shifted in the model to match the vertical coordinate of the grid block; this avoids having to estimate a pressure gradient between the actual location and the block centre.

There exist different well types for various levels of wellbore model sophistication. The simplest types use fixed mass fraction allocations, the most advanced types make use of the Paiwera wellbore simulator to model productivity or injectivity of the well. To fully wellbore-model a well only a well completion and the feedzone productive indices need to be specified; the thermodynamic conditions in the feedzone are determined during the simulation. Well completions and feedzone indices can change over time to reflect modifications to the well or scaling.

During a simulation the system can automatically run through three periods: natural state, calibration and scenario. The behavior of the surface network can be changed during these periods. Typically there will be no wellbore simulations during the natural state run to save computational time. During calibration time the goal may be to reproduce observed well data; the surface network can be kept passive during this time. In scenario mode the behavior of the surface network is determined by a power plant scenario and potential makeup wells; the system may try to autonomously allocate fluid take from the wells to achieve a set plant target.

3. MODEL RUN

With TOUGH2 chosen as reservoir simulator the system first generates a TOUGH2 compatible input file then runs the appropriate equation of state executable over the input files. All ROCK, ELEM, CONNE, GENER blocks etc in the TOUGH2 input file are automatically generated; there is no need for any manual intervention by the user. A unique naming convention is used whereby each ELEME statement can be linked back to the block and MINC layer it was generated from, making it possible to locate elements which generated difficulties in TOUGH2.

If Tauira is chosen as the reservoir simulator there is no need to generate another input file.

At the beginning of each time step of the simulation the reservoir simulator passes the thermodynamic conditions at the feedzones to the wellbore simulator. The wellbore simulator then calculates the deliverability/injectivity curves of the producers and injectors. These form the input parameters for the network simulator, which then determines the individual well fluid take via interpolation of these curves. Once this operating point of the wells is established the wellbore simulator determines the individual feedzone takes. The mass rate and – for the injectors – fluid composition at the feedzones is passed back to the reservoir simulator which will use them as sources/sinks during the time step.

Output files are generated at user defined times. The key output files are a file saving the current primary variables of the reservoir simulator in an HDF format, an XML file containing the well and surface network data and an HDF file containing all reservoir data together with a copy of the mesh data.

4. VISUALIZATION AND ANALYSIS

After the simulation has run the results are loaded back into the graphical user interface. There are several standard views present which quickly enable the modeler to visualize the results in commonly used ways or to do further analysis on the simulation data.

4.1. 1D-Views

One dimensional views are typically line plots along welltracks. This view is dominantly used for displaying reservoir properties, in particular natural state temperature and pressures since the data for these were collected along the wellbore during or after drilling.

The individual sections of the wellbore within the grid are accurately calculated by computing the intersection points of the welltrack with the grid faces. A VTK probe filter is then applied to sample the 3D reservoir data along the wellbore sections. It is possible to either sample the data – resulting in a “sawtooth” type of plot – or to interpolate the data. Interpolation is often more useful when comparing pressure data.

Time-dependent calibration data can be assigned to each well, making it very easy to plot calibration and simulation data in the same plot.

4.2. 2D-Views

Currently only one type of two-dimensional plot is implemented in MataTauira – the display of gravity in the (x,y) plane.

Prior to the simulation run the modeler can enter station coordinates and – if available – field data of a microgravity survey. Gravity factors between the station locations and the grid blocks can be calculated either via the $1/r^2$ method or by refining the grid blocks into tetrahedrons (Franz, 2013). The resulting change in microgravity is calculated during the simulation and can be plotted using colored 2D-maps, contour lines or colored station points.

4.3. 3D-Views

The heart of post-processing is the 3D viewer which enables the modeler to plot all thermodynamic data of the reservoir.

Block-based data can be colored by any thermophysical property present in the HDF output file. The block-based view can be sliced with a cutting plane to look into the interior of the reservoir.

Selecting an individual block can bring up a quick view window, which gives the numeric values for all thermophysical properties of the block at the current time. It also includes quick plots, showing the temporal evolution of the thermophsyical properties, its sources/sinks and all connections to other elements. This is a very useful feature to quickly glance at computational elements which create problems in a simulation.

Vectors can be added to visualize the flow of mass through the model. These vectors can be colored by block entities, i.e. to highlight the temperature of the fluid. Also the velocity of the fluid (gas, liquid, or total) can be displayed as a vector.

Connection parameters – primarily the permeability – can be displayed as network of colored line segments. This view can be very informative since it eliminates the rock/compound categories and fully shows where the reservoir is permeable, down to the last individual connection.

Feedzones and other sources/sinks can be colored by source/sink information. Welltracks can be colored by calibration data, making it easy to spot areas in a model which are too hot or too cold.

Further to the above the 3D view can be exported as a portable network graphics, as VTK files or as an animated movie.

4.4. Surface Model

Output from the surface model features different plots showing the mass rate and fluid properties at the output ports of the individual objects. Wellbore modeled wells can display additional data, e.g. the deliverability/injectivity curves over time, feedzone data and borehole conditions.

4.5. Python Scripts and Report Generation

Plots which are not implemented as standard feature in MataTauira can be generated via simple Python scripts. Some object classes are included in MataTauira which enable efficient data extraction from the simulation data; these can be used by the modeler to generate additional plots.

Most of the standard plots in MataTauira feature a snapshot mode in which the plot settings are captured. After a simulation these snapshots can be used to automatically generate plots and put these plots into a LaTeX script, resulting in a PDF document.

4.5. Streamline Analysis

Output from the MataTauira model can be loaded into a simple streamline simulator called Makenu. Makenu can simulate the transport of virtual particles using the 3D phase velocities as calculated by MataTauira. It simulates a short burst of a tracer (stable, radioactive or thermally decaying) and follows the virtual particles through the reservoir until they are absorbed by a sink or leave the model domain. Output generated by Makenu can be used to form time-of-flight histograms and compare these to actual tracer breakthrough data.

5. EXAMPLE

A simple demonstration model is used to illustrate the most important capabilities of the software. The example model is loosely based on the commonly-known five-spot model. Figure 2 shows a cross section through the reservoir, highlighting the cold area around the injector.

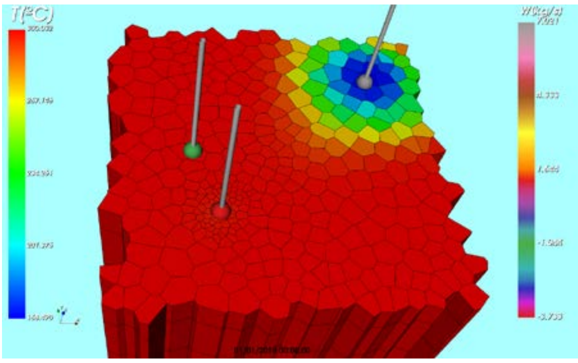

Figure 2: The demonstration model. The injector is located at the upper right hand corner. The producing well is located in a refined mesh at the lower left. The left well is the makeup well. Cells are colored by temperature, feedzones are represented as spheres

The corresponding surface model setup is show in figure 3. The flows from the producer and one initially inactive makeup well pass through valves which are required for equalizing the pressures. A union combines the flows and a pipe is used to simulate pressure friction losses. A separator, set at 15bar, splits the phases. The gas phase travels through a power plant which is set to operate at fixed electrical output. The brine flows through another valve into a splitter. The split ratio is set to 90% reinjection. The brine recombines with the condensate and is sent into the injection well.

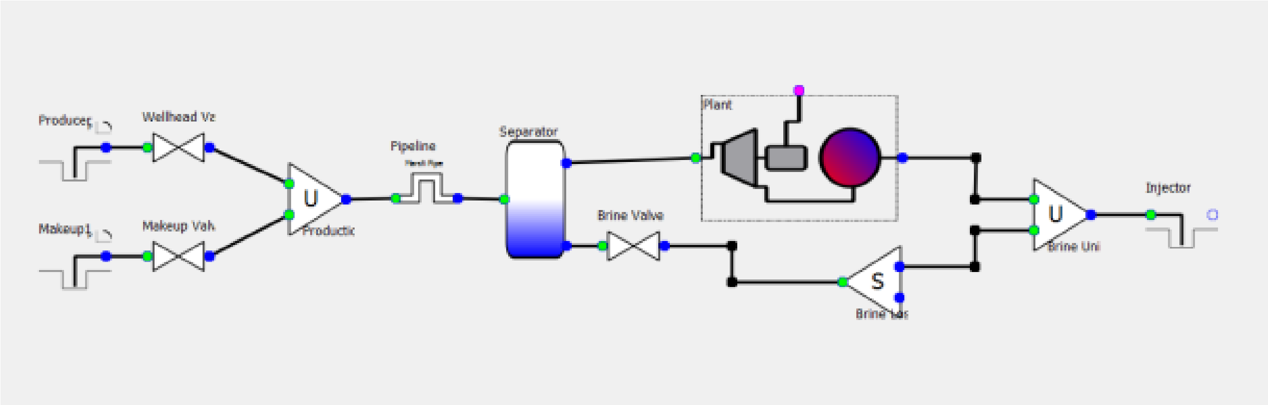

Figure 3: Surface setup for the example model.

The model is run over a 10 year period. Mostly due to the 10% losses during reinjection the pressure in the reservoir decreases, leading to a decreasing productivity of the producing well. Figure 4 shows this change in productivity; figure 5 shows the pressure profiles in the wellbore during the simulation. After ~4 years of operation the wellhead pressure of the well quickly decreases, showing the effects of an aging well. After ~8 years the pressure would have gone below the 15bar separator pressure. However at this stage the system has brought the makeup well online, relieving the stress on the well shown. It can hence operate at a slightly reduced flow rate and maintain pressures above the separator pressure.

Figure 6 (left) shows the change in liquid saturation in the model. Loss of mass due to the 10% void fraction during reinjection leads to the reservoir drying out from the top. This results in a change in microgravity which is shown in figure 6 (right).

To demonstrate the streamline analysis a simple tracer test is shown in figure 7. 10,000 virtual particles are injected and their trajectories through the reservoir are modeled using the Makenu simulator. Figure 7 (right) shows the time-of-flight histogram of the particles arriving in the producing well. The tracer arrives in several peaks, the most pronounced being the initial breakthrough at ~300 days. This demonstrates transport along preferential pathways in the model due to its permeability structure (which is not shown here). Tests on an isotropic five-spot model showed the expected single peak with elongated tail.

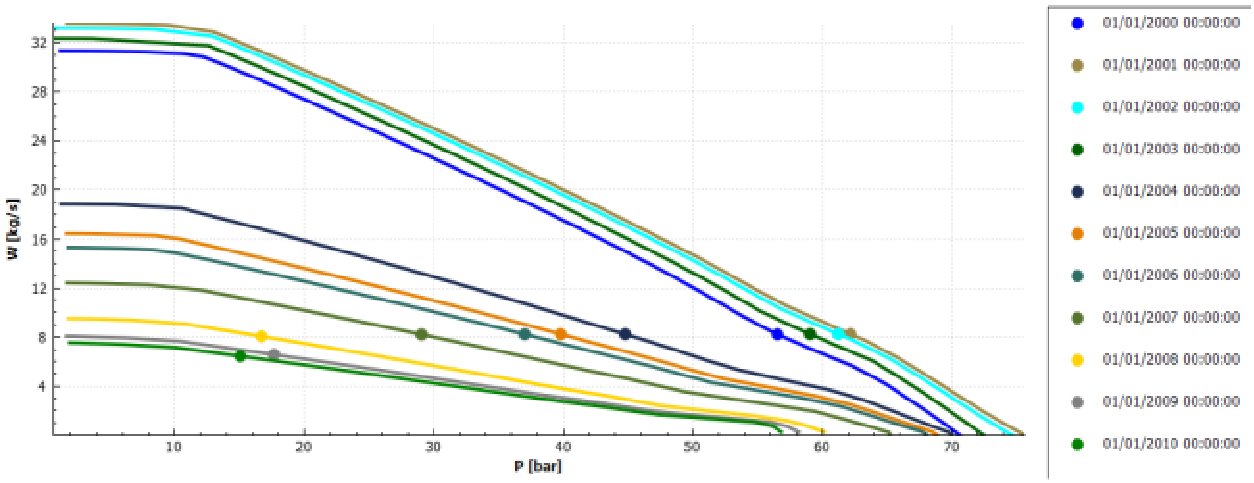

Figure 4: Changing productivity of the producing well over time. The disc symbolize the operating point of the well. The system ensures that the wellhead pressure stays above the 15bar threshold set by the separator.

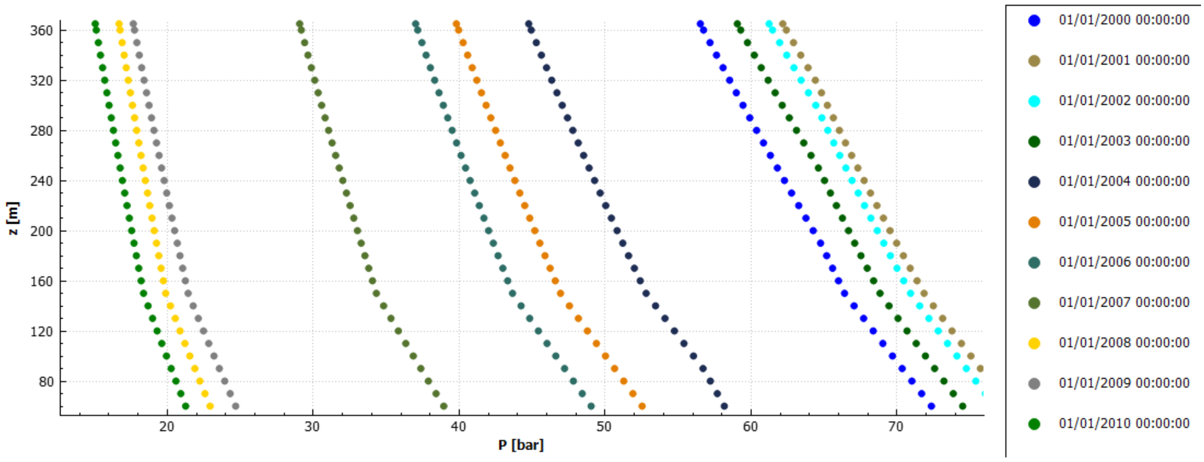

Figure 5: Wellbore modeled pressure profiles in the producing well. The wellhead pressures correspond to the operating points shown in figure 4.

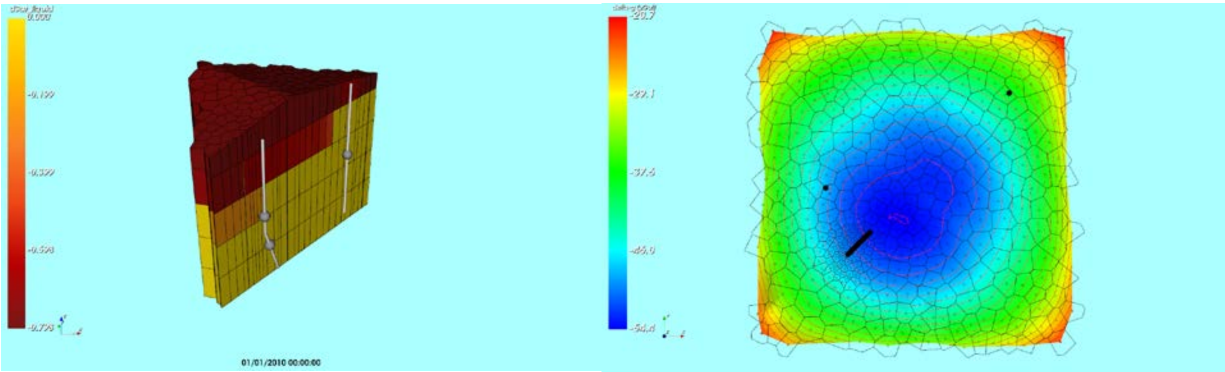

Figure 6: The left view shows a cross section with the change in liquid saturation of the model after 10 years. The right view shows the resulting change in gravity at the surface.

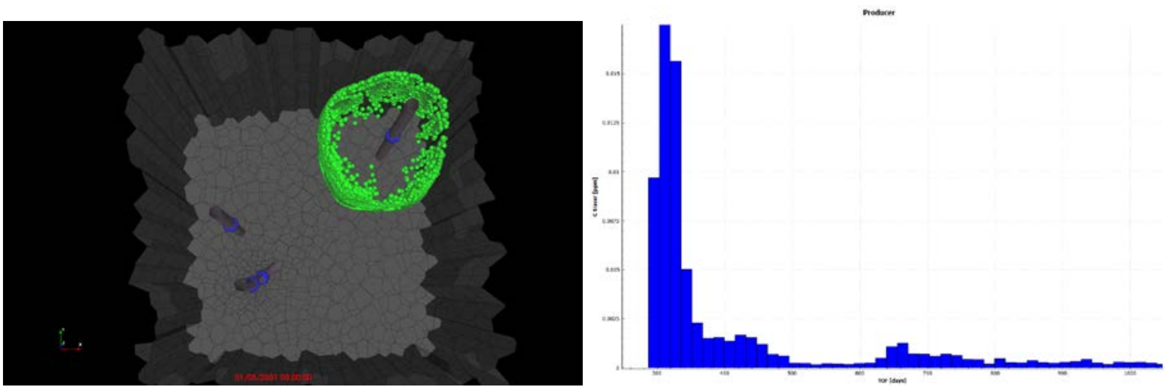

Figure 7: Tracer injection test using 10,000 virtual particles. On the left green spheres symbolize particle locations at a given time. The plot on the right shows the time-of-flight of the particles reaching the producing well.

6. DISCUSSION AND OUTLOOK

The MataTauira software package has been a great success within Mercury's reservoir engineering team. We have started porting our full field models to the system and developing further models with it. Its major benefits are its portability, ease of use and its coupled reservoir/wellbore/surface features.

Portability is a very important factor when engineers work on a model as a team. Using multiple pre- and post-processing scripts made collaborations very hard in the past, since it was near to impossible to replicate all ancillary settings on another team member's computer. Now models can be easily shared and can run on different platforms (Windows 32/64bit, Linux).

The ease of use has led to an increased productivity of the modelers. Model output is now very easy to inspect and it has become much simpler to figure out where a model needs to be adjusted to yield a better calibration.

The greatest benefit from an operator's point of view are the coupled reservoir/wellbore/surface simulations, which are now a widely used feature within our company. It is now possible to achieve a realistic calibration of all the wells in a reservoir simulation and use this calibration to predict both the field and individual well behavior during realistic scenarios with much more confidence than before. This greatly reduces the risk of geothermal operations.

The next focus is on improving the in-house developed reservoir simulator Tauira. This simulator has shown great promise in reliability and accuracy but still needs to be tuned in order to run faster than TOUGH2. However its biggest advantage is that it can serve as a backbone for future extension, like reactive transport or geomechanical models

ACKNOWLEDGEMENTS

We would like to acknowledge the great help the reservoir engineering team has received from John Burnell (GNS) during the development of MataTauira. John has written the key modifications in our TOUGH2 version without which the development of the coupled simulators would not have been possible.

REFERENCES

Croucher, A. (2013). *PyTOUGH User' s guide*.

Franz, P. (2013). Calculation of Micro-Gravity using a Refined Tetrahedral Scheme for Use with Reservoir Simulators. In *35th New Zealand Geothermal Workshop 2013 Proceedings*.

Franz, P. (2015). PAIWERA – A ROBUST WELLBORE SIMULATOR FOR GEOTHERMAL APPLICATIONS. In *New Zealand Geothermal Workshop 2015 Proceedings*.

Pruess, K., Oldenburg, C., & Moridis, G. (1999). *TOUGH2 User's Guide, Version 2.0*. *LBNL-43134*. LBNL-43134.