SWANHILD: NUMERICAL PRESSURE TRANSIENT ANALYSIS USING THE VOLSUNG GEOTHERMAL RESERVOIR SIMULATION PACKAGE

Katie McLean[1], Peter Franz[2] and Jonathon Clearwater[2]

[1] Contact Energy Ltd, Private Bag 2001, Taupo 3352, New Zealand

[2] Flow State Solutions Ltd, 67 Hamon Place, Rotorua 3015, New Zealand

katie.mclean@contactenergy.co.nz

Keywords: *Swanhild, Volsung, well testing, pressure transient analysis, numerical*

ABSTRACT

The need for numerical models for geothermal pressure transient analysis (PTA) has been long recognised. Until recently the only tools/software available for PTA in the geothermal industry have been based on analytical models, which are too simplistic to represent geothermal wells. The lack of numerical tools for PTA led to the creation of the numerical PTA framework (McLean and Zarrouk, 2017), which is a set of guidelines on how to set up and run numerical PTA models using TOUGH2 and Python. Analysis of the model results and comparison to field data required skills in writing auxiliary computer scripts by the individual reservoir engineer. Despite assistance from the many functions built into Python, numerical PTA was still relatively time-consuming and cumbersome to perform. Specialized software was necessary to bridge this gap, and hence Swanhild was developed, based on the numerical PTA framework guidelines.

This study presents testing of the new Swanhild software application. Swanhild forms part of the Volsung geothermal reservoir simulation package and is specifically designed to run numerical PTA simulations using the standardised approach as specified by the numerical PTA framework. The application makes numerical PTA straightforward, including model setup and inversion with field data. The model matching can be performed manually or automated using non-linear regression. A comparison between Swanhild and the TOUGH2/Python implementation of the numerical PTA framework across a range of model parameters reveals no practical difference in the model results, thus validating Swanhild for use in numerical PTA. Software licenses for Swanhild are available for free as part of an ongoing effort to enable and promote quality PTA in geothermal wells.

1. INTRODUCTION

Pressure transient analysis (PTA) has been largely disregarded by the geothermal industry for geothermal wells. This overlooked technique is potentially one of the most useful tools available to a reservoir engineer. It can inform on well condition and also the nature of the wider reservoir including boundaries.

Some of the problems with geothermal PTA have been practical issues around collection of field datasets (pressure and flow rate, during flow changes). Most of these issues are relatively easy to avoid or minimise with appropriate design of the well testing program (Zarrouk and McLean, 2017). The bigger issue to overcome has been a lack of appropriate tools for analysis of geothermal datasets. Conventional tools (plots and software) have been based on analytical models which are too simplistic to represent geothermal reservoirs, due fundamentally to their non-isothermal and potential two-phase nature. The requirement for numerical tools has been recognised for a long time, however the specialised knowledge required to set up and run numerical models using appropriate reservoir simulators has been an impediment.

The numerical PTA framework was created in order to assist with this issue. It is a set of guidelines on how to set up and run numerical models for PTA, using TOUGH2 (Pruess et al., 1999) and Python, though the guidelines are generally applicable to all reservoir simulators. While the framework does enable numerical PTA to a much greater extent than before, it does require knowledge of coding in Python. For rapid and effective PTA of field data with no knowledge of numerical simulators or coding required, the software application Swanhild has been developed using the Volsung reservoir simulation package (Clearwater and Franz, 2019; Franz et al., 2019), and tested against the TOUGH2/Python implementation of the numerical PTA framework.

2. BACKGROUND

2.1 Pressure transient analysis

Pressure transient analysis refers to the analysis of pressure transients which are induced by flow rate changes in a well. Historically these have been simple step changes in flow rate, such as: cessation of production inducing a buildup pressure response, or cessation of injection inducing a pressure falloff response. Modern methods involving time superposition allow any step change in flow rate to be analysed by PTA, including between non-zero flow rates (Zarrouk and McLean, 2019). In fact, pressure transients between non-zero flow rates are recommended testing practice (Figure 1) in order to avoid/minimise a range of practical issues which can arise with data collection in geothermal wells (Zarrouk and McLean, 2019).

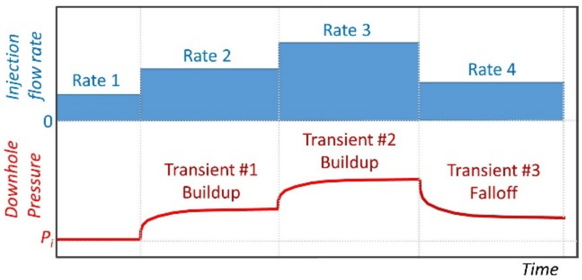

Figure 1: Example of a 4 flow-rate injection test design with 3 pressure transients, all measured between non-zero flow rates.

The Bourdet pressure derivative plot is a log-log plot of pressure and pressure derivative (Bourdet, 2002). It is widely regarded as the most significant advance in the history of PTA (Houzé et al., 2012). This is because the pressure

Proceedings 42nd New Zealand Geothermal Workshop
24-26 November 2020
Waitangi, New Zealand
ISSN 2703-4275

derivative has characteristic shapes for many different flow regimes and reservoir/boundary behaviours (Figure 2), and is therefore the key to diagnosing which model may be a good match to a set of field data.

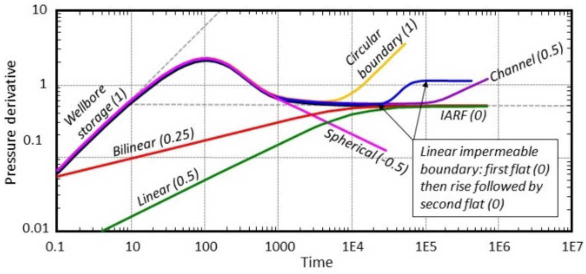

Figure 2: Examples of characteristic pressure derivative shapes (from Zarrouk and McLean, 2019).

2.2 The need for numerical PTA for geothermal wells

Conventional PTA involves matching models to field data, where the model response is calculated analytically. There are a variety of limitations to analytical PTA models in geothermal reservoirs, as many assumptions of the pressure diffusivity equation (which underlies all analytical models) are violated in geothermal reservoirs. In particular, geothermal reservoirs are non-isothermal and have non-uniform fluid properties (McLean and Zarrouk, 2017).

A strongly non-isothermal scenario is created during completion testing by injecting ambient temperature water into geothermal reservoirs, which can be over 300°C. This is a particularly obvious example of the inapplicability of analytical models, which require the assumption of a single fluid temperature. Failure to account for the non-isothermal scenario can lead to massive overestimates in both reservoir permeability and skin factor (McLean and Zarrouk, 2015; Guerra and O'Sullivan, 2018).

O'Sullivan et al. (2005) also recognised this gap, and created the first software for numerical PTA, AWTAS, which generates the model response using the simulator TOUGH2. AWTAS had a range of model options and built-in inverse modelling capability. Unfortunately AWTAS was developed for a private client and was never widely available. Also the graphical user interface was written in a programming code which is now obsolete (Zarrouk and McLean, 2019).

2.3 The numerical PTA framework

The numerical PTA framework is a set of guidelines developed to guide in the implementation of numerical models for geothermal PTA. While McLean and Zarrouk (2017) implemented the framework using Python to control the numerical simulator TOUGH2, the guidelines are applicable to other reservoir simulators. The guidelines cover details of the radial grid setup, which includes a well block, skin zone and reservoir zone (Figure 3), and also time stepping and other reservoir parameters.

The objective of the numerical PTA framework is to aid the reservoir engineer with the multiple decisions necessary in the setup of such models, and at the same time promote comparability of results (McLean and Zarrouk, 2017).

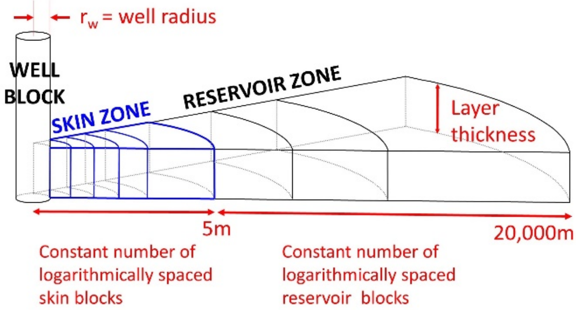

Figure 3: Schematic of radial model grid domains: not to scale and does not show all blocks (modified from McLean and Zarrouk, 2017).

3. THE SWANHILD GUI

The Swanhild graphical user interface (GUI) is an addition to the Volsung geothermal reservoir modelling software package. It has been specifically designed for numerical pressure transient analysis. PTA requires substantially less functionality than what is offered by the main Brynhild user interface in Volsung; Swanhild therefore utilises only the limited functionality required and facilitates setting up simulations using commonly used concepts in PTA.

A logarithmically spaced grid is used for PTA, based on the guidelines of the numerical PTA framework. The user can edit key features like well volume, wellbore and skin radii and the extent of the reservoir boundary. Non-infinite boundaries like channels and fixed boundaries to one side are supported. The fractional dimension model is implemented, so the grid can be linear, radial, spherical or any non-integer dimension in-between.

The model is split into three different domains as per the numerical PTA framework: the wellbore, the skin zone and the reservoir zone (Figure 3). For each domain the user can define the initial state and rock properties (including MINC - Multiple INteracting Continua; Pruess and Narasimhan, 1985). Permeability is set as a scalar for the reservoir. The permeability in the skin zone k_s is calculated via the skin factor. The permeability of the well block is set at 1000 times the reservoir permeability.

Injection/production is defined as a table of mass rate steps. Each step in the table is interpreted as a transient period, i.e. later analysis can be performed for each of these periods.

Time settings include the start time and duration of the test. Time steps are logarithmically spaced, growing from a minimum step size by a constant factor up to a maximum step size set by the user. At each flow rate change the time step is reduced again to the minimum time step.

Swanhild further has the capability to import and display pressure field data to which the simulation data can be compared.

The GUI has three main charts for viewing the data:

1. A chronological chart is used for displaying data traces (pressure and flow rate) in comparison to real time and is good for getting an overview of the entire dataset, which can include multiple transient periods.

2. Pressure history plot: a linear chart plots the selected transient period as delta pressure versus delta time, where the offsets for both pressure and time are determined at the beginning of each transient period.
3. Pressure derivative plot: a log-log plot displaying pressure and pressure derivative as shown in Figure 4. It is traditionally the main plot used for test diagnosis and model matching.

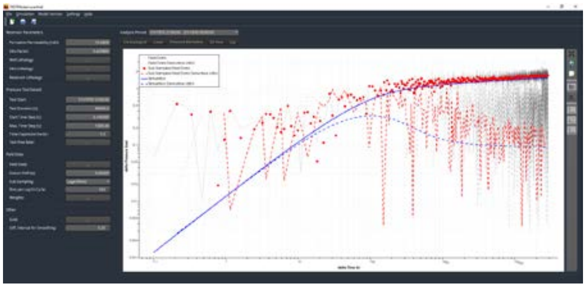

Figure 4: Example pressure derivative plot from Swanhild GUI.

In addition to the above, the GUI also contains a simple 3D viewer (Figure 5). The radial model is mathematically one-dimensional, and while features such as linear impermeable boundaries and fractional dimension reservoirs can be implicitly represented via modifications to the radial model grid, they cannot be directly displayed in 3D. However, the simple 3D viewer displays the radial grid as a narrow wedge, which can be zoomed and cells individually selected. This can be used to investigate the thermophysical properties of each cell of the simulation, and the extent of pressure disturbance into the reservoir from the well testing (radius of detection). There is also a text log detailing the progress and issues of a simulation run.

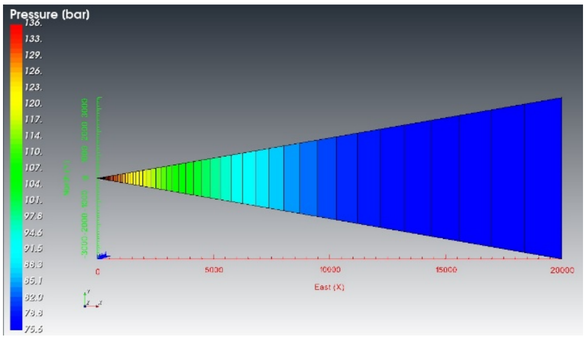

Figure 5: Simple 3D grid viewer example: a radial model grid appears as a wedge, however the block volumes and connection areas correspond to a radial model.

4. COMPARISON OF SWANHILD AND TOUGH2/PYTHON IMPLEMENTATION

The numerical PTA framework was implemented in TOUGH2 and Python by McLean and Zarrouk (2017); however the guidelines are generally applicable to any reservoir simulator. The implementation of the numerical PTA framework in Swanhild utilises the Volsung reservoir simulator rather than TOUGH2. Advantages of Volsung over TOUGH2 include:

- Better control of time stepping: not limited to 104 non-uniform time steps.
- Full numerical precision in both pressure and time output; this issue is very important when dealing with derivatives. In TOUGH2 the simulation had to be restarted with every flow rate change due to this issue.
- Model input is automatically generated by the GUI.
- Field data can be directly included in the model and analyzed in the same manner as simulation output for true one-on-one comparisons.

In order to validate the Swanhild implementation, Swanhild model results are compared to those from TOUGH2/Python, for a base model and then across a range of variable model parameters.

4.1 Base model (A1)

The base model radial grid was set up following all the guidelines of the numerical PTA framework, including the following parameters which remain constant for this investigation:

- 50 blocks in the skin zone (recommended default).
- 100 blocks in the reservoir zone (recommended default).
- Porosity of the well block = 0.9 (recommended default).
- Permeability of the well block = 1000 times permeability of reservoir (recommended default).
- Layer thickness of 1000 m, being in the typical range for open-hole sections in geothermal wells.

The parameter values for the base model are summarised in Table 1. The simulation is of a pressure buildup from a single injection flow step. The base model is isothermal as the injectate temperature is the same as the initial reservoir temperature (Table 1).

Table 1: Model parameter values for base model (A1)

Model ID	**A1**	
Description	Base model	
Parameter	**Units**	**Value**
Well radius r_w	m	0.11
Well volume V	m^3	100
Well compressibility C	Pa^{-1}	8.00e-08
Transient duration	s	14,400
Injection flow rate	kg/s	20
Initial reservoir pressure P_i	bara	110
Initial reservoir temperature T_i	°C	250
Injectate temperature T_{inj}	°C	250
Fractional dimension n	-	2.0
Reservoir permeability k	m^2	1.0e-14
Skin factor s	-	0
Differentiation interval	-	0.2

The Swanhild base model results are indistinguishable from an equivalent base model with the same parameters using TOUGH2/Python, in both the pressure history plot (Figure 6) and pressure derivative plot (Figure 7). Note in these plots the TOUGH2/Python data equals the sub-sampled data; a difference between these is only visible in noisy field data sets.

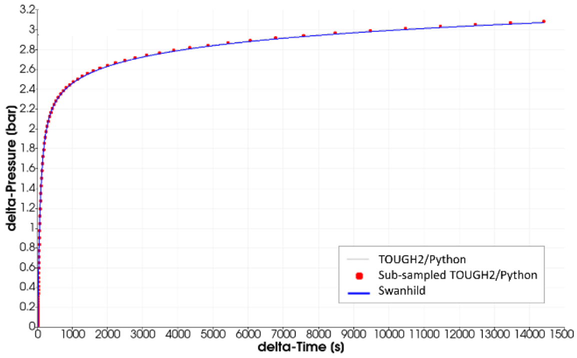

Figure 6: Pressure history plot comparing base model A1 results from Swanhild and TOUGH2/Python implementations.

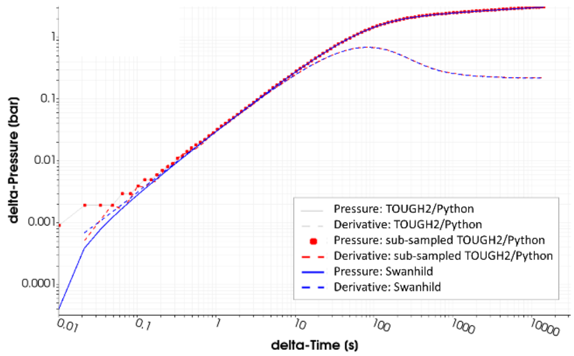

Figure 7: Pressure derivative plot comparing base model results from Swanhild and TOUGH2/Python implementations.

4.2 Test models phase 1

Twenty-one test models (models A2-A22) were created to assess whether the success of Swanhild in matching the TOUGH2/Python implementation remains the case across a range of variable model parameters. The changes from the base model parameters (Table 1) for the test models are summarised in (Table 2). In most cases only a single parameter is changed for each model.

For most of the test models there is negligible difference in the model results, as was the case for the base model A1 (Figure 6 and Figure 7). However there is a minor difference in the results of three test models:

- Non-isothermal models A5 and A6: shown in Figure 8 (pressure history plot) and Figure 9 (pressure derivative plot) for model A5, the more strongly non-isothermal case of the two.
- Hotter and deeper model A16: shown in Figure 10 (pressure history plot) and Figure 11 (pressure derivative plot).

These minor differences in the A5, A6 and A16 results are likely due to differences in the calculation of thermodynamic fluid properties between the simulators Volsung and TOUGH2 - Volsung uses IAPWS-97IF while TOUGH2 uses IFC-67. These are the only test models involving differences in pressure and temperature, which control thermodynamic properties. The differences are of the same order of magnitude as the difference observed by McLean and Zarrouk (2017) between two different versions of TOUGH2. This is not an issue provided that models to be directly compared are generated using the same simulator (McLean and Zarrouk, 2017).

Table 2: Summary of test models (A2-A22) parameter changes from base model values.

Model ID	Description	Change to base model parameter
A2	Well compressibility increase	C = 8.00e-07 Pa^{-1}
A3	Well compressibility decrease	C = 8.00e-09 Pa^{-1}
A4	Longer transient duration	1,440,000 s
A5	Non-isothermal #1	T_{inj} = 40 °C
A6	Non-isothermal #2	T_{inj} = 70 °C
A7	Reservoir perm. increase	k = 1.0e-13 m^2
A8	Reservoir perm. decrease	k = 1.0e-15 m^2
A9	Positive skin factor #1	s = 5
A10	Positive skin factor #2	s = 10
A11	Negative skin factor #1	s = -2
A12	Negative skin factor #2	s = -3
A13	Bigger well radius	$r_{w=}$ 0.2 m
A14	Bigger well volume	V = 150 m^3
A15	Smaller well volume	V = 70 m^3
A16	Deeper and hotter well	P_i = 150 bara T_i = 300 °C T_{inj} = 300 °C
A17	Lower flow rate	rate = 5 kg/s
A18	Higher flow rate	rate = 50 kg/s
A19	Longer differentiation interval	0.4
A20	Shorter differentiation interval	0.1
A21	Higher fractional dimension	n = 2.5
A22	Lower fractional dimension	n = 1.5

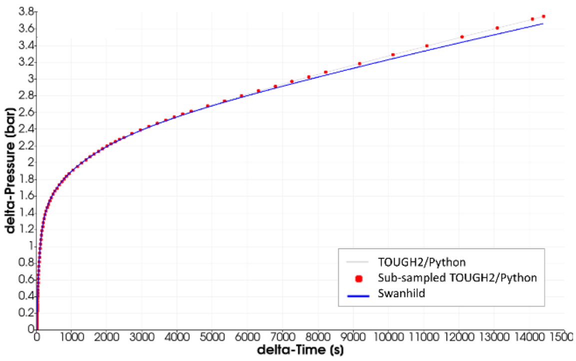

Figure 8: Pressure history plot comparing test model A5 (non-isothermal) results from Swanhild and TOUGH2/Python implementations.

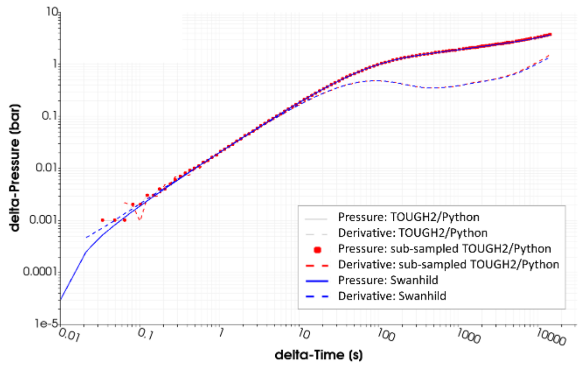

Figure 9: Pressure derivative plot comparing test model A5 (non-isothermal) results from Swanhild and TOUGH2/Python implementations.

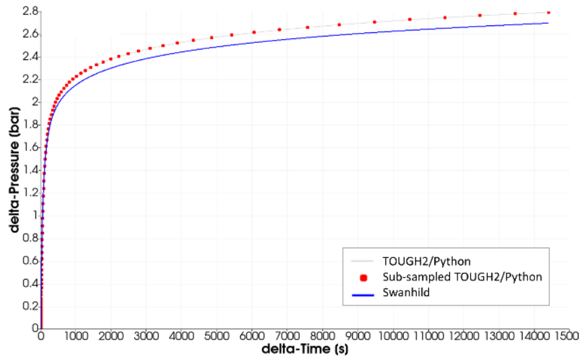

Figure 10: Pressure history plot comparing test model A16 (hotter and deeper) results from Swanhild and TOUGH2/Python implementations.

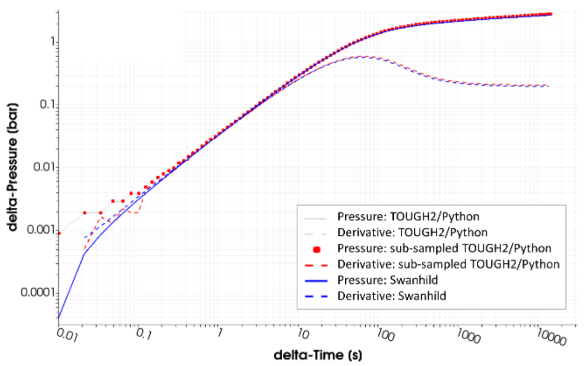

Figure 11: Pressure derivative plot comparing test model A16 (hotter and deeper) results from Swanhild and TOUGH2/Python implementations.

It should be noted that the major change in shape of the pressure derivative in Figure 9 as compared to Figure 7 (base model) is purely the result of non-isothermal effects from injecting cold water into a hot reservoir with no prior cooling. It is not reflective of any change in the nature of the reservoir. Distortion of the characteristic shapes of the pressure derivative by these non-isothermal effects is a recognised phenomenon (McLean, 2020). It is necessary to include the reservoir cooling induced by drilling injection in the simulation to avoid this distortion and to have a chance of matching real-life field data.

One further difference between the TOUGH2/Python and Swanhild implementation was found in the treatment of the change in porosity due to compressibility/expansivity. Both simulators calculate the change in porosity in the same way but differ in the way they treat the rock volume fraction. TOUGH2 simply keeps the rock volume fraction constant to its initial value. This implicitly changes the block volume since the rock volume fraction plus the porosity no longer add up to unity. In the Swanhild implementation an increase in porosity reduces the rock volume fraction; the energy balance is preserved by adjusting the rock grain density. Both implementations give equivalent results under most circumstances. A downside of the TOUGH2 approach is that it does not honor the grid geometry. Swanhild fixes this by ensuring the porosity plus rock volume fraction equal unity; however, under artificial conditions in which the rock volume fraction approaches zero the pressure can spike. In practice this behaviour is of little importance and can be adjusted by changing the well block volume and its porosity value.

4.3 Test models phase 2

Due to the fluid thermodynamic property-related difference in the model results from phase 1 (strongest in test model A16, Figure 10), a subset of test models was created in order to further investigate this difference. The parameters of these additional test models are specified in Table 3.

The results in Table 3 show the difference in the pressure at the finish time of the simulation between TOUGH2 and Swanhild. A clear trend in temperature is visible, i.e. the pressure difference (TOUGH2 minus Swanhild) increases with temperature. The overall difference however is small. To further analyze this behaviour we would need to make a detailed comparison between the TOUGH2 and Swanhild implementations of the relevant quantities, i.e. viscosity, density and compressibility. However this is beyond the scope of this work.

Table 3: Summary of test models (A16 B to L) parameter changes from base model. The relative difference is the observed difference between the TOUGH2 and Swanhild model runs at the finish time.

Model ID	P_i [bara]	T_i [°C]	T_{inj} [°C]	Relative difference TOUGH2 - Swanhild
A16-B	100	100	100	-0.08 bar
A16-C	100	200	200	-0.03 bar
A16-D	100	300	300	+0.11 bar
A16-E	125	100	100	-0.09 bar
A16-F	125	200	200	-0.03 bar
A16-G	125	300	300	+0.09 bar
A16-H	150	100	100	-0.09 bar
A16-I	150	200	200	-0.03 bar
A16	150	300	300	+0.09 bar
A16-J	200	100	100	-0.10 bar
A16-K	200	200	200	-0.03 bar
A16-L	200	300	300	+0.08 bar
A1 base model	110	250	250	+0.01 bar

5. INVERSE MODELLING CAPABILITIES

5.1 Description

The typical problem in PTA analysis is to determine reservoir properties from the pressure transient field data. The pressure derivative plot is a very useful part of this process, though it utilises superposition time in the calculation of the pressure derivative, in order to account for the entire flow history. Superposition time strictly only applies to mathematically linear systems, with radial flow (fractional dimension of n = 2.0). However despite this, the pressure derivative plot remains the key diagnostic tool to identify which model is likely to match the field data, based on characteristic shapes (McLean, 2020).

Reservoir properties are not determined directly from features of the pressure derivative plot, as they have been with historic graphical methods (for example using the slope of a straight line in a semilog plot to calculate the reservoir permeability). The reservoir properties are determined by inverse modelling, which has the advantage that it can deal with non-ideal, complicated models. Its disadvantage is that it is computationally expensive and may require a starting point which is already close to the final solution.

Swanhild uses PEST (Doherty, 2018) for parameter estimation purposes. PEST is available as free software and has been extensively used for inverse modelling purposes due to its flexible interface. However setting up an inverse modelling problem with PEST is still time consuming and daunting for a beginner. Also, due its broad range of functionality, new PEST users have to learn about a large number of control parameters and figure out their applicability and default values.

Swanhild makes this process easier by fully automating the setup and running of the inverse simulation. An advanced user could still edit most of the PEST control parameters; however they are already setup with sensible default values and most of the time the user doesn't need to alter them at all. Hence the user only requires some basic knowledge about the inverse modelling process.

The inverse modelling can be performed in parallel with BEOPEST which can speed up the simulation significantly. Typical runtimes for an inverse model with multiple parameters are in the order of several minutes when using a modern multi-core CPU.

The inverse modelling process starts with the user entering field data and performing manual data quality control. An important parameter which needs to be determined at this stage is the datum shift, i.e. the time lag between change in flow rate and the response seen by the pressure sensor.

Next the subsampling type needs to be selected. If the user opts for no subsampling then only basic operations like datum shift and removal of data points outside of the test period are performed. The default logarithmic subsampling type employs a user defined number of bins per log cycle. Logarithmically spaced time intervals are chosen, starting with a small value at each flow rate change. For each interval the average pressure and average time is found and used as subsampled data. This process is of importance for the inverse modelling since field pressure data is typically sampled at constant intervals. If no subsampling is applied then only a few data points will be influential for the early-time parameters like wellbore volume and skin, while the later data dominates the objective function.

Another way of modifying the importance of some data is by giving it a different weight in the objective function. For this optional purpose the user can enter a table of weights. For example it is possible to eliminate all weights of early-time data so the inverse modelling process focuses on later data.

After the preparation steps the user selects one or more free parameters; currently about a dozen different choices are available. The most important ones are the wellbore volume, the skin factor and the formation permeability. When using real-world data it is also very important to include the initial pressure as a free parameter so that the inverse model zeroes in on the same mean pressure; neglecting to include this parameter can lead to skewed results or failure of PEST to find a solution.

Running of the PEST inverse model including generation of all auxiliary files is fully automatic. On completion all auxiliary information such as the run control log or parameter sensitivity output are presented. On user acceptance the best parameters found are set and the model is run again to display the final match to the field data.

5.2 Inverse modelling example

To demonstrate the inverse modelling capabilities in Swanhild we generated a simple test model using three consecutive flow rate steps of 10, 20 and 30 kg/s spaced 8 hours apart. We ran this model forward and generated output at high frequency, i.e. at least one data point every 10 seconds. From this data we generated two data sets. The first contains the exact pressure over time from the model run.

For the second data set we added significant noise to the pressure data. The noise was generated using a normal distribution with a standard deviation of 0.2bar. For comparison, the model was run at a pressure of ~200bar, with the response to the flow rate being ~5bar. So while the synthetic instrument error is only about 0.2/200=0.1% the actual noise to signal ratio 0.2/5=4% is quite high and typical for a pressure test. Figure 12 shows this dataset, along with subsampled data and the final simulation results.

For the inverse modelling process we chose as free parameters the initial pressure, the formation permeability, skin factor and well volume. Logarithmic subsampling was used with 100 bins per log cycle. The inverse modelling process was started using an initial set of parameters which was quite far from the final solution with the exception of the initial pressure. Experience has shown that this particular parameter needs to be close to the real value otherwise PEST can struggle to find a valid solution.

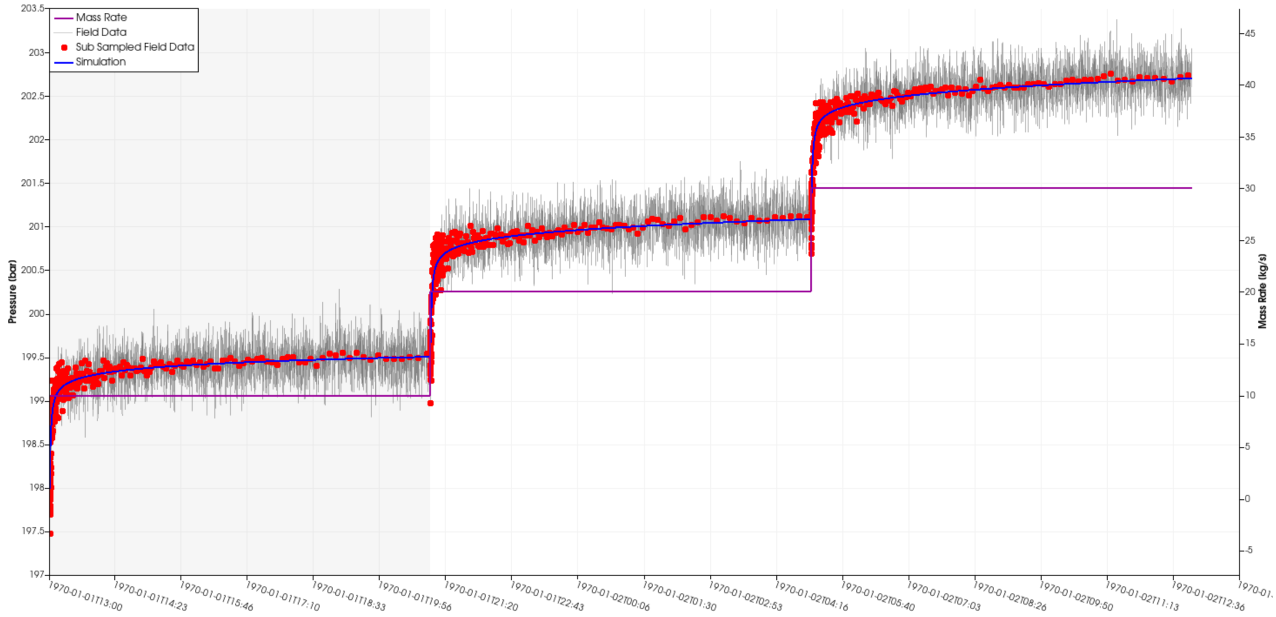

Figure 12: Chronological plot for 3-flow-step test, displaying mass flow rate, field pressure data (red and grey) and modelled pressure (blue line). First transient period is highlighted in light grey.

Table 4: Estimated parameters from inverse modelling example: initial values are the starting point for inverse modelling.

Parameter	Initial Value	True Value	Final Value – Dataset without Noise	Final Value – Noisy Dataset
Initial Pressure P_i	197.75 bara	198.00 bara	198.00 bara	197.99 bara
Skin Factor s	2.00	0.00	0.06	0.43
Formation Permeability k	50.00 mD	10.00 mD	10.07 mD	10.59 mD
Well Volume V	100.00 m^3	150.00 m^3	150.59 m^3	148.44 m^3

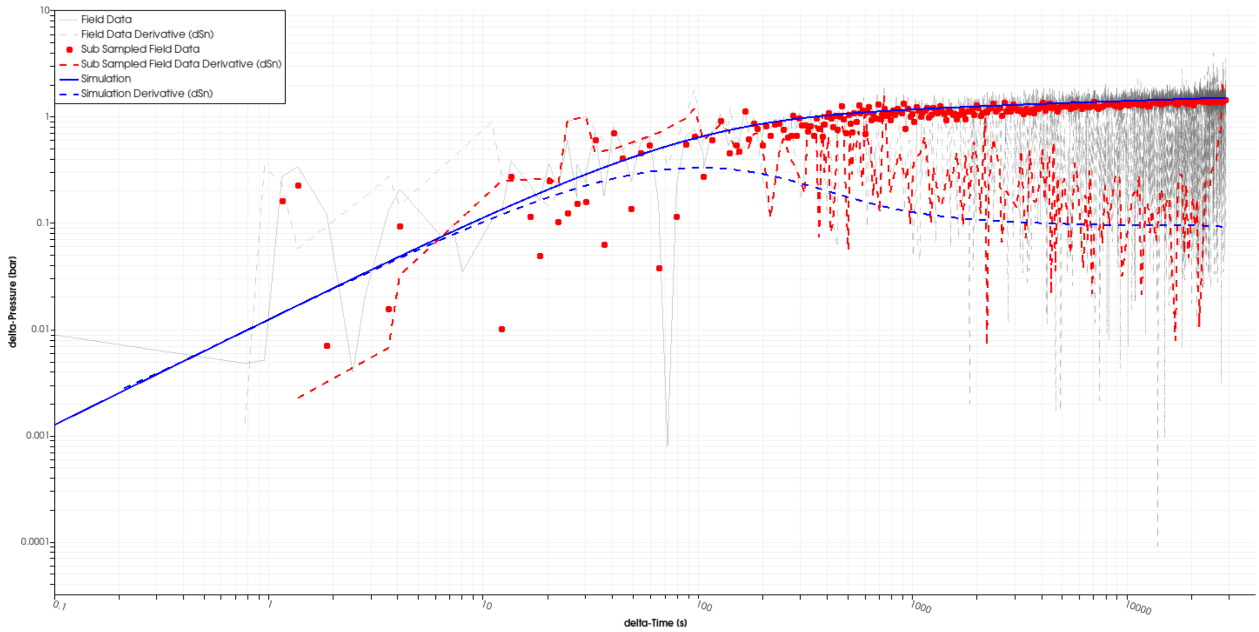

Figure 13: Pressure derivative plot for first transient period of Figure 12, showing pressure and derivative for the field data and the model. Grey dashed lines are unfiltered field data. Red points are logarithmically subsampled field data. Derivatives are shown in dashed blue lines for the model and dashed red lines for the subsampled field data.

The results from the inverse modelling process for both models are shown in Table 4, together with the true and initial parameter choices. The results demonstrate an excellent capability of PEST to reconstruct the values for the dataset with no noise. For the dataset with noise the results are still very good; the exception seems to be the skin factor which would probably require more or better mid-times data for its precise evaluation.

Figure 13 shows the pressure derivative plot for the first transient in this example for the noisy dataset using a smoothing interval of 0.2 (differentiation interval). Traditionally the formation permeability would be determined using the final slope in this plot. Judging by the amount of noise, even after smoothing, it would be quite difficult to determine a meaningful value for the permeability in this fashion. However repetitive inverse modelling runs using different initial parameters have shown that PEST strongly zooms in onto the true value for the permeability. This demonstrates that inverse modelling has great potential in use for PTA.

6. CONCLUSIONS

- Swanhild is an effective implementation of the numerical PTA framework, enabling rapid analysis of geothermal PTA datasets without any requirement for specialised knowledge of numerical simulators or coding.
- The Swanhild results which use the Volsung reservoir simulator have been validated by comparison with the results of a TOUGH2/Python implementation of the numerical PTA framework.
- Across a range of model parameter values there is negligible difference in model results between the two simulators.
- The only exception to this is a minor difference in some models related to different calculations of thermodynamic properties within the two simulators, for different temperature conditions. This difference is not significant in the PTA context and is of the order of magnitude to be expected between different simulators. It does not affect the shape of the pressure derivative.
- The use of PEST for inverse modelling has been successfully demonstrated on a model with both clean and noisy data. In both cases Swanhild/PEST was able to find parameters close to their true values.

ACKNOWLEDGEMENTS

This work grew from a PhD project which was financially supported by the Todd Foundation Postgraduate Scholarship in Energy Research, and Contact Energy.

REFERENCES

Bourdet, D. (2002): *Well Test Analysis: The Use of Advanced Interpretation Models*. Volume 3 of Handbook of Petroleum Exploration and Production, Elsevier.

Doherty, J. (2018): *PEST Model-Independent Parameter Estimation User Manual Part I*. Watermark Numerical Computing <www.pesthomepage.org/downloads>.

Clearwater, J., & Franz, P. (2019). *Introducing the Volsung Geothermal Simulator: Features and Applications.* Proceedings 41st New Zealand Geothermal Workshop

Franz, P., Clearwater, J., & Burnell, J. (2019). *Introducing the Volsung Geothermal Simulator : Benchmarking and Performance*. Proceedings 41st New Zealand Geothermal Workshop

Guerra, R. J. A. and O'Sullivan, J. (2018): *Investigating the effects of non-isothermal reservoir conditions on pressure transient analysis of an injection/fall-off test using numerical modelling*. Proceedings 40th New Zealand Geothermal Workshop, Taupo, New Zealand, 14-16 November.

Houzé, O., Viturat, D. and Fjaere, O. (2012): *Dynamic data analysis: the theory and practice of pressure transient, production analysis, well-performance analysis, production logging and the use of permanent downhole gauge data*. KAPPA, v4.12.03 edition.

McLean, K. and Zarrouk, S. J. (2015): *Impact of cold water injection on geothermal pressure transient analysis: a reservoir modelling assessment*. Proceedings 37th New Zealand Geothermal Workshop, Taupo, New Zealand, 18-20 November.

McLean, K. and Zarrouk, S. J. (2017): *Pressure transient analysis of geothermal wells: a framework for numerical modelling*. Renewable Energy 101, 737-746.

McLean, K. (2020): *Pressure transient analysis of geothermal wells: a numerical modelling approach*. University of Auckland doctoral thesis.

Pruess, K. and Narasimhan, T. N. (1985) *A practical method for modeling fluid and heat flow in fractured porous media*. Society of Petroleum Engineers Journal, 25, (1), 14-26, February.

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

Zarrouk, S. J. and McLean, K. (2019): *Geothermal well test analysis: fundamentals, applications and advanced techniques*. 1st edition, Elsevier.