EXPERIENCES IN DEVELOPING A DUAL POROSITY MODEL OF THE LEYTE GEOTHERMAL PRODUCTION FIELD

Jericho Omagbon[1/2], Michael O'Sullivan[2], John O'Sullivan[2] and Cameron Walker[2]

[1] Energy Development Corporation, One Corporate Centre Building, Ortigas Center, Pasig City, Philippines

[2] Department of Engineering Science, The University of Auckland, 70 Symonds Street, Auckland, New Zealand

omagbon.jb@energy.com.ph

Keywords: *Leyte geothermal production field, PEST, PyTOUGH, dual porosity, MINC, inverse modeling, SGEMS, pilot points.*

ABSTRACT

The Leyte geothermal production field is among the largest developed geothermal system in the world. It has a total installed capacity of 700MWe and more than 180 production and injection wells have been drilled over its more than 30 years of production life. Several numerical models have been developed in the past for use in managing the production in the field. The recent numerical model has been updated with significant enhancements of previous models. The model now uses a more efficient grid system with a finer grid resolution. It also includes the implementation of dual porosity and automated model calibration.

The model is presently in the calibration stage. Significant progress in calibration has been accomplished in the past months and the model is now in a satisfactory state for use in prediction of future reservoir response. Nonetheless, calibration effort is still being continued to get a still better match of model response to the measured data. Therefore, this paper is a work-in-progress report of the latest numerical simulation effort on the Leyte geothermal production field. The purpose of this paper is to convey the challenges encountered during the development and automated calibration of the model and how some of those challenges were handled.

1. INTRODUCTION

The Leyte geothermal field is a large liquid dominated geothermal system located in central Philippines. It is composed of two independent hydrothermal systems, namely: (1) Tongonan and (2) Mahanagdong. Commercial operation in the field started in 1983 with the commissioning of the 112.5 MWe Tongonan 1 power plant. The most recent power plants built, Mahanagdong A and B, were commissioned in 1997. At present, the field has a total installed capacity of 700MWe with more than 180 wells drilled.

The Tongonan system within the Leyte geothermal production field is a high temperature geothermal system. Temperature greater than 300ºC have been measured in wells near the upflow area of the field. There was also a natural two-phase zone present in the shallow depths around the upflow region. The large scale production that started in 1997 triggered intensive boiling, a result of a massive pressure drawdown. Cooling from injection returns soon followed, most notably in wells close to the injection area. Other processes affecting the steam availability in the field include casing erosion (brought about by increase in enthalpy of some wells combined with high level of suspended solids), wellbore blockage (due to mineral scaling), brine and condensate return, enhanced groundwater inflow and feedzone sharing.

In the Mahanagdong system, temperatures higher than 300 ºC were also measured in some wells. The production zone in Mahanagdong was generally deeper than in Tongonan. Signs of boiling were observed after the start of commercial operation but development of a two-phase zone was slowed down by cooler groundwater inflow and brine returns from the northern and southern injection area.

2. DESCRIPTION OF THE MODEL

In 2010, a 3D numerical model of the Leyte geothermal production field was developed by Ciriaco et al. This was the first model that is large enough to cover both the Tongonan and Mahanagdong reservoirs. The model was used to investigate the future steam flow trends in the field under different production scenarios, estimate the make-up well requirements and come up with a long-term drilling schedule. The present model is based on the 2010 model but modifications were introduced to address some of the identified shortcomings of the 2010 model.

2.1 Model Grid

The model is set to cover a large area of 1,800 km^2 (40km x 45km). The production area is confined to a smaller 56 km^2 (4km x 14km) region around the center of the model. One future improvement that the previous modeling team recommended was the use of smaller grid blocks in the production area, finer than the 500m x 500m that they used. It was thought that a finer grid would provide improved accuracy and would be better to handle the dense spacing of the wells in the field.

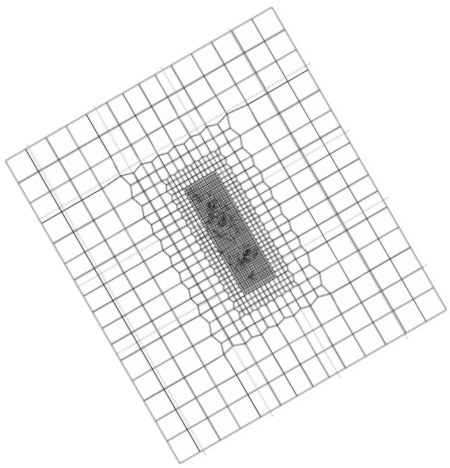


Figure 1: Grid structure for the Leyte Geothermal Production field numerical model

The grid structure for the new model of Leyte geothermal field is shown in Figure 1. The PyTOUGH library was used to construct this grid. The block size in the production region for this model is 300m x 300m. The present model has a total of 32,835 active blocks. This is only about 20% more than the previous model which has 27,509 grid blocks. The 5-sided Voronoi grid refinement technique suggested by Croucher and O'Sullivan (2013) was used to achieve this improved resolution without a significant increase in the total number of grid blocks.

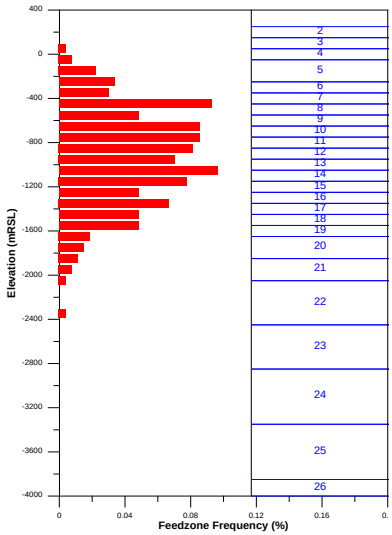


Figure 2: Feedzone location histogram and layer structure of the model.

Both the old and new model has 24 active layers. The layers structure is shown in Figure 2. Layers with a high feedzone density were set to have a thickness of 100m.

2.2 Dual Porosity Grid

A dual porosity representation was used so that the model can capture the thermal breakthrough observed in the field more accurately. The latest PyTOUGH library contains a function to generate MINC blocks from a single porosity model. The use of the PyTOUGH library was preferred over TOUGH2's built-in MINC generator in converting the present model to dual porosity because of its flexibility. In the built-in MINC generator in TOUGH2, MINC is implemented to all blocks in the model. With PyTOUGH, we were able to apply MINC to selected grid blocks only, thereby avoiding an unnecessary increase in the total number of blocks. The MINC function in PyTOUGH also allows one to customize the fracture and matrix blocks naming convention. This feature eliminated the problems with duplication of matrix block names normally encountered in models with many grid blocks when using TOUGH2's MINC function. Lastly, the PyTOUGH library provided an easy-to-use utility for generating a dual porosity initial condition file from a single porosity INCON or SAVE file (the importance of this feature is explained later in this paper).

The dual porosity model of the Leyte geothermal field used 3 interacting continua (1 fracture and 2 matrix blocks) with corresponding volume fractions of 2%-10%-88%. The permeability for the matrix blocks were set to 0.001E-15 m^2. Fracture spacing was set to 150m and the fracture porosity was set to 90%. At the time of writing, only the fracture permeabilities were included as parameters to be estimated during calibration. MINC was applied only to the grid blocks within the production area. The dual porosity model has a total of 59,185 grid blocks (about twice the number of blocks in the single porosity model).

2.3 Permeability distribution and visualization

At present, assigning rock-types into the model is carried out using the commercially available software Petrasim. Petrasim is a graphical user interface for TOUGH2. Two of the most useful features offered by Petrasim include (1) the windows interface for easy assignment of rock-types to the model blocks and (2) the ability to overlay figures like conceptual model, geophysical boundary and geologic maps on top of the model grid (which can be used as guide in assigning rock-types). In the future, we wish to utilize more powerful tools like Leapfrog for this task.

One problem with Petrasim is that it cannot read a model grid generated by other utilities like PyTOUGH. Moreover, it is also difficult (if not impossible) to replicate the desired grid structure for the Leyte model in Petrasim. Fortunately, Petrasim is capable of exporting a TOUGH2 input file and a there is a utility in PyTOUGH that can read the grid information from this exported input file then transfer them to a PyTOUGH generated TOUGH2 input file.

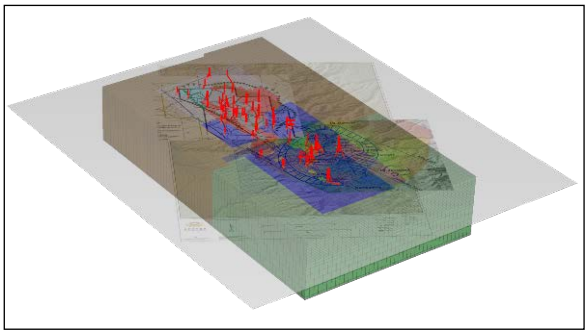


Figure 3: Petrasim model used for assigning rock-types into the working model.

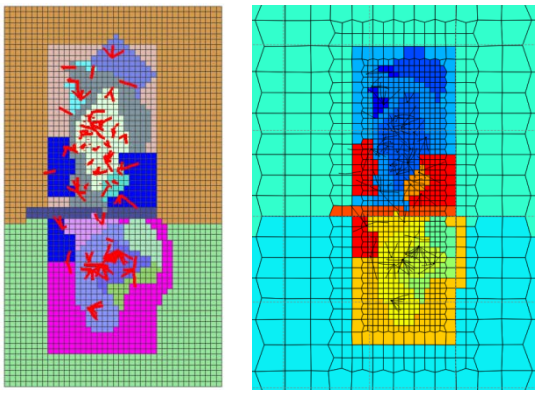


Figure 4: Structure and rock-type distribution of the Petrasim model (left) and the resulting rock-type distribution in the working model (right).

Figure 3 shows the Petrasim model used for assigning rock-types. It also shows a conceptual model map overlaid on the grid which served as guide in assigning the rock-types. The Petrasim model was made in such a way that the grid sizes in the production region is the same as that with the working model. After setting/modifying the rock-type distribution in the Petrasim model, a TOUGH2 input file is generated using Petrasim's export function. A python script then reads this

exported file and copies the rock-types assigned in each blocks into the working model.

2.4 Boundary conditions

The top boundary of the model was set at the approximate elevation of the top of the water table and is connected to a large atmospheric block. Estimating the top of the water table was not a straightforward task because the elevation to be assigned for all the columns in the model is not easy to interpret from the well data. It was assumed that the elevation of the water table varies proportionally with the topographic elevation. These topographic elevations were then multiplier by a factor and fitted into the columns of the model using PyTOUGH's *fit_surface()* command. This process was repeated many times until the pressure in the model matched those that were measured in the well.

The upflow in the model is represented in the model by injecting water at a specified temperature at the bottom-most layer. Springs were defined by blocks with high vertical and horizontal permeability from the top of the model down to the main reservoir. Mass input at the bottom which represents the upflow in the field and the springs at the surface of the model were assigned in Petrasim and transferred into the working model in the same manner as the rock-types. Allocating upflows and springs in Petrasim was convenient because their locations were already indicated in many conceptual model maps of Leyte. By overlaying these maps on the Petrasim grid, it was easy to locate which blocks in the model corresponds to these upflow and outflow regions.

3. CALIBRATION

The model underwent the usual natural state calibration and production history matching. Data used to calibrate the model is composed of interpreted downhole temperatures from 140 wells, PCP pressure from 76 wells, pressure vs. time data from 57 wells and enthalpy data from 119 wells. Calibration of the model was carried out using a mix of manual calibration and automated parameter estimation.

The automatic parameter estimation was carried out using the BeoPEST software and a 32-core server from Amazon. Tikhonov regularization and singular value decomposition were both employed in the inversion process. In theory, the inversion difficulties resulting from parameters with low sensitivity should have been taken care of by Tikhonov regularization and singular value decomposition. Experience from calibrating this model, however, showed that the regularization settings that were used were not very effective. Manually identifying the non-influential parameters and then fixing them at their preferred values ended up giving better results.

3.1 Parameterization

Parameters estimated in the model include permeability, porosity, upflow mass input and upflow enthalpy. The elevation of the top boundary was also adjusted during the early stages of the model development to match the depth of the water table that was inferred from the measured pressures in the wells. In the present model, there are 36 rock-types defined. Permeability parameters included the horizontal and vertical permeability of each of these rock-types.

A pilot point parameterization scheme was employed after obtaining a set of rock-type permeability values that result in a reasonably good match between the modeled results and observed data. This was done to allow the model to exhibit heterogeneity in the permeability in hope that this would give an improved match to the measured data. The pilot point parameters were set as multipliers to the rock-type permeability. To implement this, two models were needed, namely: a primary model which has the 36 rock-types and a secondary model in which each block is given its own rock-type (to allow individual blocks to have their own permeability values).

A total of 576 pilot points were scattered around the production area in the model. Multiplier values were assigned to each of the pilot points. These were set equal to 1.0 prior to the pilot point calibration so that the permeability distribution of the secondary model was exactly the same as that of the primary model. Ordinary kriging interpolation was used to obtain the multiplier for each of the blocks in the model. This kriging interpolation from the pilot points to the TOUGH2 model grid was done autonomously through the use of the geostatistical software SGEMS and some python scripts. Once the multiplier for a particular block is obtained, a python script then searches the rock-type that is assigned to it in the primary model, obtains the permeability value of that rock-type, applies the multiplier to that permeability value and write the new permeability value to the secondary model. An example of the permeability distribution obtained after the inclusion of the pilot point parameters in the model calibration process is shown in Figure 5.

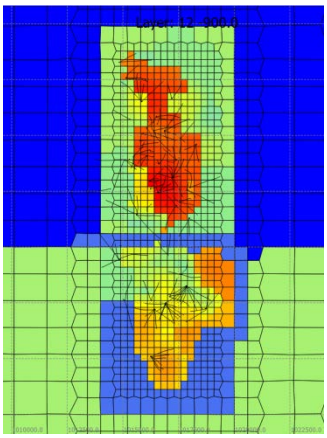
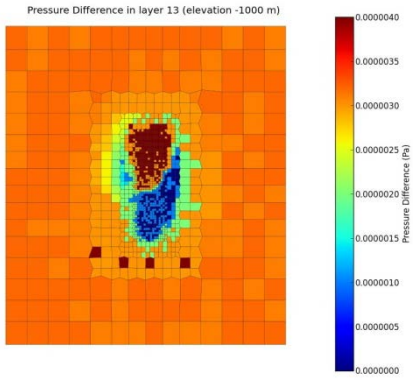


Figure 5: Permeability distribution obtained after application of pilot point parameterization.

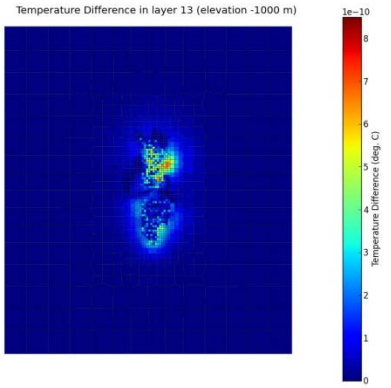
3.2 Use of a single porosity model for natural state calibration

For the natural state model, the single porosity model was observed to produce a very similar temperature and pressure values compared to the dual porosity model. This is illustrated in Figure 6 where the difference in temperature and pressure between the block from the single porosity model and the corresponding fracture block in the dual porosity model at one layer in the model is shown to be very small (<9E-10 ^{o}C for temperature and <4E-6 Pa for pressure). In generating this plot, the single porosity model was set to run for a total simulation time of ~30 million years. Using a python script, the resulting SAVE file from this model was then converted into an INCON file that is compatible with the dual porosity model. The dual porosity model is then allowed to run for ~30 million years. Given this finding, it was concluded that running the much bigger dual porosity model to natural state is unnecessary and can

be skipped totally, thus cutting the run-time for the natural state model by several hours.



(a)



(b)

Figure 6: Pressure (a) and temperature (b) difference for a single porosity and dual porosity natural state models.

3.3 Suppressing boiling for faster convergence in TOUGH2

In TOUGH2, models with single phase fluid generally run faster than models with two-phase fluid. In the early stage of the natural state and production history calibration, the atmospheric block was set at a significantly higher pressure than atmospheric pressure in order to prevent the fluid in the model from boiling. This allowed the model to run in a relatively shorter time.

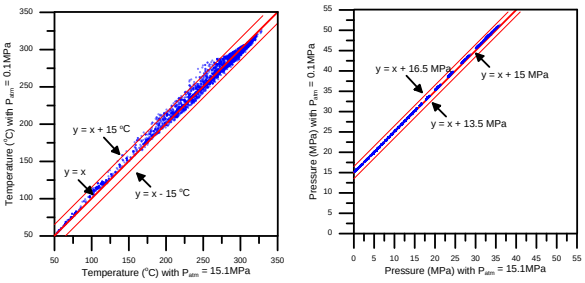


Figure 7: Natural state temperature (left) and pressure (right) results for two models with the same permeability distribution. Boiling is suppressed (y-axis) in one and allowed (x-axis) in the other.

Figure 7 shows a plot comparing the modeled natural state temperature and pressure in the wells from two models with the same permeability distribution but with boiling suppressed in one model (by using high atmospheric pressure) while allowed in the other. Boiling was suppressed by increasing the atmospheric block pressure by 15MPa. In the pressure plot, it is seen that the pressure is basically just shifted by 15MPa. The temperature plot suggests that the temperatures resulting from the model with high atmospheric pressure are almost the same as the temperatures obtained from the model with the correct atmospheric block pressure. There is however a difference of about 15ºC in some temperatures. Nonetheless the speed-up obtained using a non-boiling model benefited the early stages of the natural state calibration where the temperature mismatches were still very large.

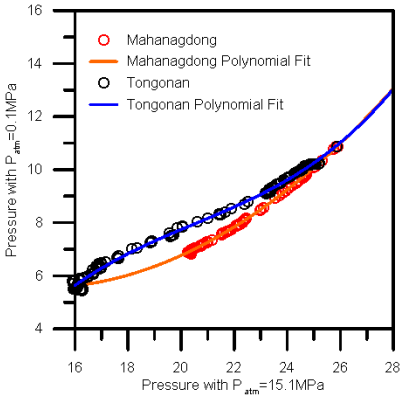


Figure 8: Comparison of production pressures for two models with the same permeability distribution, but with boiling suppressed (x-axis) in one and allowed (y-axis) in the other.

A similar technique was also applied for production history pressure matching. Figure 8 shows a plot of the pressure in the pressure monitoring blocks obtained from a non-boiling model against a boiling model. Unlike the results from the natural state models, the relationship is not linear but a polynomial curve was able to fit the data. It was also observed that this relation changes when permeability values change. Hence, this curve needs to be updated when a significant change in the permeability is made. In this plot, the pressure from the Tongonan wells were fitted using a cubic polynomial, while the pressure from the Mahanagdong wells were fitted using a quadratic polynomial. These polynomial curves enabled us to run the much faster non-boiling model then calculate what the pressure would be if the boiling model was used, with reasonable accuracy, and saving a significant amount of model development time.

3.4 Enthalpy matching

Upon reaching a reasonable match for the natural state temperature, PCP pressure and production pressure trends, the atmospheric pressure now needs to be set at the correct value in preparation for flowing enthalpy matching. Permeability and porosity were again adjusted in an attempt to replicate the enthalpy measured from the wellhead of each well. Because the model was now allowed to boil, the run-time increased significantly making this process slower. Significant progress was attained after several attempts of matching the enthalpy. Nonetheless, many wells still showed enthalpies that are different from the measured ones especially those wells that have feedzones in the two-phase region of the reservoir.

Another factor that may affect the flowing enthalpy from a well is the proportion of flow from different feedzones for wells with more than one feedzone. All of the wells in the

field produce from multiple feedzones. This proved to be a challenge because these wells do not all have flowing surveys available to infer the proportion of the mass flow to be assigned to each feedzone. Flowing surveys for a few wells do exist but most of them were carried out at well head conditions that were different from normal operating conditions (e.g. at highly throttled condition). Hence, the mass flows assigned to each feed zone during the production history runs are highly approximate.

One approach that we employed for matching the enthalpy in the Leyte model was adjusting the flow from individual feedzones in the well. Again, this was an iterative process that required several tries. Fortunately, the flowing enthalpy at the wellhead resulting from changing the feedzone distribution can be estimated rather quickly without having to run the model again in TOUGH2. Upon completion of a TOUGH2 production history run, the individual feedzone enthalpy is available from the output file. Assuming a two feedzone well with feedzone enthalpies (as obtained from a TOUGH2 run) equal to h_1 and h_2, the new enthalpy value (H_{new}) after modifying the feedzone proportion to x_{1new} and x_{2new} ($x_{1new} + x_{2new} = 1$) may be calculated using the simple energy balance equation:

$$H_{new} = x_{1new}\, h_1 + x_{2new}\, h_2$$

It is important to note that changing the mass distribution for the feedzone in the well will have an effect on the enthalpy from the individual block. Our experience however showed that the effect is not very large. Regardless, it is still advisable to run a new model that reflects this new feedzone flow proportion once the desired matches in the well head enthalpies are attained.

3.5 Calibration results

Most of the measured data used for calibrating the model is composed of the natural state temperature and production history enthalpy and pressure. The current states of the matches for these data are presented in Figure 9 to Figure 12. From Figure 9, it can be seen that temperature matches obtained from the current model are similar to those for the old 2010 model. Enthalpy matches (which are reflected by the matches in the steam flow) are seen to be better for the present model than the old one (see Figure 10). Production pressure matches for Tongonan reservoir in the present model are about the same as for the old model. In the Mahanagdong area, the pressure matches are better in the new model than for the 2010 model.

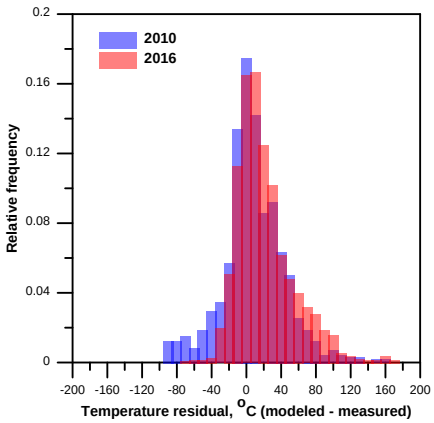


Figure 9: Histogram of natural state temperatures showing results for the 2010 and 2016 model.

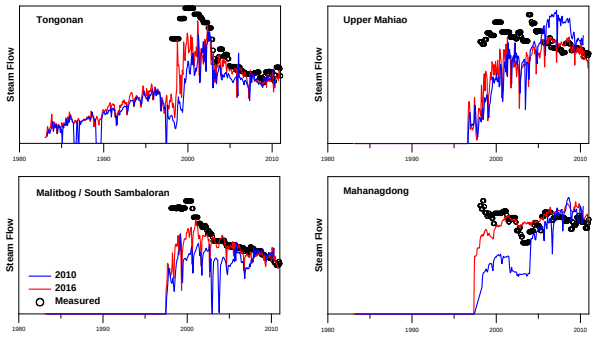


Figure 10: Measured vs modeled steamflow for the 2010 (red) and 2016 (blue) model

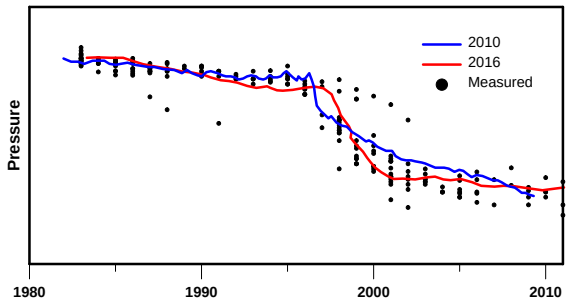


Figure 11: Representative pressure match for the Tongonan reservoir

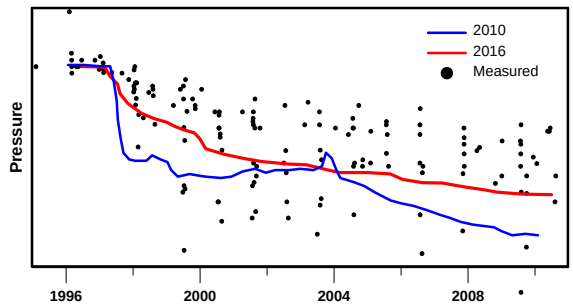


Figure 12: Representative pressure match for the Mahanagdong reservoir

4. CHALLENGES AND ISSUES WITH MODEL CALIBRATION

There were several issues and challenges encountered throughout the course of model calibration. Some of these issues significantly slowed down the progress of the calibration. Some of these issues and the work-around used to overcome these challenges are discussed here.

Convergence problem: TOUGH2 employs adaptive time-stepping allowing it to reduce the time step (by a factor given by REDLT) if convergence is not attained after a certain number (given by NOITE) of Newton-Raphson (NR) iterations or increase it by a factor of 2 if convergence occurs in less than a certain number (given by MOP(16)) of NR iterations. By design, TOUGH2 does not update the primary variables when convergence is achieved after one NR iteration. This behavior was found to cause the simulation to stall for the present model when difficulty occurred with the NR iteration. This stalling problem is illustrated in Figure 13 and was encountered on several occasions during natural state model calibration.

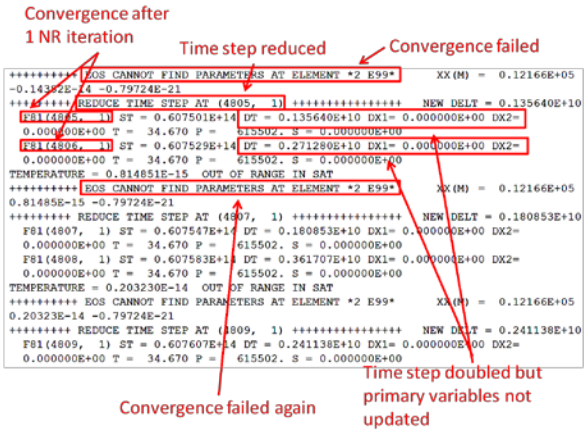


Figure 13: Screenshot of a model run in which TOUGH2 got stuck in a problematic time step.

In this model, a bad time step (in which NR convergence was not attained) was encountered causing TOUGH2 to reduce such time step to a smaller value. When the reduced time step is used, convergence is attained after one NR iteration. This causes the time step to double without updating the primary variables (as indicated by both DX1=0 and DX2 = 0 in Figure 13). The time step kept increasing but at each increase, convergence is still attained with one NR iteration until the time step becomes equal or greater than the bad time step. Because no changes in the primary variables happened in the previous iterations, TOUGH2 is back in the same situation where the linear solve failed again repeating the cycle until the maximum number of time step (MCYC) is reached.

One solution to this problem is to use an appropriate value for REDLT. For example, setting REDLT equal to 1.1 will cause TOUGH2 to reduce the time step slightly below the bad time step after non-convergence of the linear solve and hopefully after a few time step reductions, TOUGH2 will find a time step in which convergence is achieved after two or more NR iterations (we want to avoid convergence with one NR iteration so that update in the primary variables happens). Note that there is no guarantee that this solution will work because it is difficult to know in advance what time step will result in convergence after 2 or more NR iterations. In general, REDLT equal to 2, 4, 8, and so on are bad choices and likely to result in this problem.

Another way to prevent convergence after one NR iteration is by setting small values for the convergence criteria (RE1 and RE2). However, this may result in longer run-times and is again not guaranteed to work. Those who have access to the TOUGH2 code (and are capable of modifying and recompiling it) may instead opt to modify it and force it to always make at least two NR iteration. This modification was already present in iTOUGH2 and can be activated by setting MOP2(1) variable equal to 2. The latest version of AUTOUGH2 also has this modification implemented. In the present model, this convergence problem was no longer encountered after the modification was made in AUTOUGH2.

Densely spaced wells: Many wells in the field are closely spaced causing these wells to share blocks with adjacent wells in the model. Furthermore, some wells show temperature data that is different from their neighboring wells with which they share a common block. An example of this problem is shown in Figure 14. In this figure, the measured temperature for the well is shown by the red line. Temperature data from the six other wells (which are listed in the plot) are shown as green circles. This type of data was found to cause difficulty with automatic calibration in PEST because of the fact that matching one well will result in a mismatch with one or more of the others.

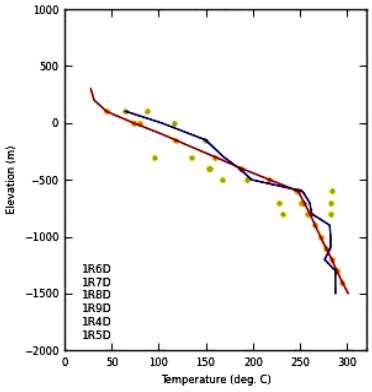


Figure 14: An example of a well sharing a block with adjacent wells. The red line is the temperature of the well, green circles are temperatures from other wells sharing the same block and the blue line is the modeled temperature.

The use of finer blocks in the model has partially solved this problem. Nevertheless, multiple wells sharing blocks with other wells are still unavoidable especially at shallow depths. At present, the solution employed was to evaluate each of the blocks shared by multiple wells, decide which well has the most reliable value (or best reflects the average temperature) and then remove the data from the other wells (or reduce their weights by significant amount). With this fix, PEST only sees one set of data to match. However in assessing the results, plots similar to that shown in Figure 14 are generated for all wells to serve as a reminder of the presence of adjacent wells in the same blocks and their corresponding temperature data.

Production history data: The flow measurements in the wells were, at best, done once every 3 months. In the model, well flows that are used for production history matching are provided with in a monthly period. For production wells, monthly mass flows are easily estimated using the well output curves (which are updated after every flow measurements) and the daily well head pressure data. This method of estimating flows from the production wells is relatively robust and is able to capture even the daily changes in well utilization. However for injection wells, estimation of flows is very crude. It normally involved simple interpolation between succeeding flow tests. Since flow tests were typically 3 months apart, changes in the injection well utilization in between successive flow tests is very easy to overlook.

Sometimes, simple checks can be made to detect potential problems with the data used in the model. The plot in Figure 15 shows the total mass extraction (black circle) and injection (blue circle) from all the wells in one region of the field. Except for the apparent noise in data, there is nothing unusual that can be inferred. However when we subtract the mass injection from the mass extraction and compare the results with the estimated steam flow delivered to the power plant (red circles vs black line), an inconsistency is now more evident. As can be seen from the plot, the steam flow

calculated from mass extraction and injection data is lower than the steam flow delivered to the power plant.

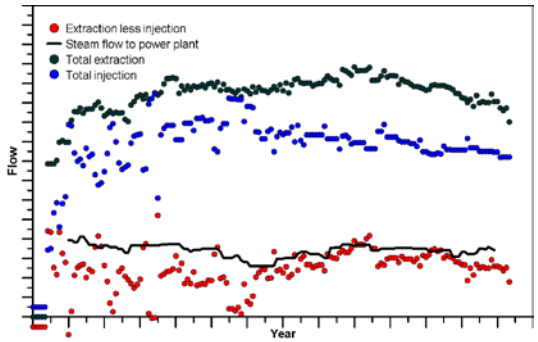


Figure 15: Comparison of the steam flow delivered to the power station and the calculated steam from mass extraction and injection (excluding condensate) data.

Investigation later revealed that the main reason for the inconsistency is the overestimation of the water flow in the injection wells. Some of this overestimation is a result of data obtained when injection capacities of the wells were measured rather than the injection load. In these cases, the flow to the other wells is diverted to the well of interest to determine how much it can accept. Once the measurement is completed, the wells are then put back to their usual injection loading. The recorded flow is however higher than the normal loading resulting in an overestimation of flow in the calculation of the monthly injection rate. This small detail tended to get lost as data were processed for use in the model. Correction was applied to the data resulting to the plot in Figure 16 which shows more consistent steam flow values. Even so, proper flow measurement is still the best solution to ensure the reliability of the data that is being used in the model.

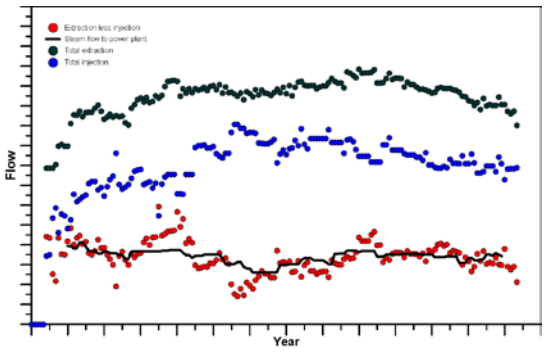


Figure 16: Comparison of steam flow to the power station and the calculated steam from mass extraction and injection data after applying the correction.

6. SUMMARY AND FUTURE WORKS

We have successfully developed a new numerical model for the Leyte geothermal production field in which we achieved improved resolution and better matches with some of the measured data compared to the old model. We have also successfully upgraded the model from single porosity in 2010 to dual porosity in the present model. Application of the work flow described in this paper and the use of modern tools like PyTOUGH, Petrasim, PEST and SGEMS enabled us to speed up the model development and calibration. The model has now achieved satisfactory matches to the data and we expect the model to produce reliable predictions. Even so, calibration effort is still being made and more progress is being accomplished.

The work presented here is part of a research project on uncertainty quantification for geothermal models. Ultimately, we want to use the present calibrated model for application of uncertainty quantification techniques.

ACKNOWLEDGEMENTS

The authors appreciate the support of Energy Development Corporation, the New Zealand Ministry of Business, Innovation and Employment, and the Energy Education Trust of New Zealand for this work. We would also like to acknowledge the contributions of Minh Pham, Angus Yeh, Adrian Croucher, Kristine Eia Antonio and Elvar Bjarkason whose ideas helped in developing the workflow described in this paper and in resolving the issues encountered throughout the model development.

REFERENCES

Angcoy, E.: Geochemical modelling of the high-temperature Mahanagdong geothermal field, Leyte, Philippines. University of Iceland, MSc thesis, UNU-GTP, report 1, 79 pp. (2010).

Ciriaco, A.E., Marquez, S.L., Omagbon, J.B., Yglopaz, D.M., Sta. Ana, F.X.M.: Leyte geothermal production field reservoir modeling updata using TOUGH2. Unpublished. Energy Development Corporation Internal Report. (2012).

Croucher, A.E. and O'Sullivan M.J.: Approaches to local grid refinement in TOUGH2 models. *Proc. 35th New Zealand Geothermal Workshop*, Rotorua, New Zealand. (2013).

Croucher, A.E.: PyTOUGH: a Python scripting library for automating TOUGH2 simulations. *Proc. 33rd New Zealand Geothermal Workshop*, Auckland, New Zealand. (2011).

Croucher, A.E.: Recent developments in the PyTOUGH scripting library for TOUGH2 simulations. Proc. *37th New Zealand Geothermal Workshop*, Taupo, New Zealand. (2015).

Dacillo, D.B., Uribe, M.H.C., Andrino, R.P. Jr. and Alcober, E.H., Sta. Ana, F.X.M., and Malate, R.C.M.: Tongonan geothermal field: conquering the challenges of 25 years of production. *Proc. World Geothermal Congress 2010*, Bali, Indonesia. (2010).

Ogena, M.S.: Experience in energy development in the Philippines: The EDC Leyte geothermal field. Presentation for the IEA-GIA Workshop. (2013).

Pruess, K., Oldenburg C. and Moridis, G.: Tough2 user's guide, version 2.0. Earth Sciences Division, Lawrence Berkeley National Laboratory. (1999).

Remy, N., Boucher, A. and Wu, J.: Applied Geostatistics with SGeMS - A User's Guide. Cambridge University Press. (2009).

Sta. Ana, F.X.M., Siega, C.S., and Andrino, R.P.: Mahanagdong geothermal sector, Greater Tongonan Field, Philippines: Reservoir evaluation and modelling update. *Proc. Twenty-Seventh Workshop on Geothermal Reservoir Engineering*, Stanford University, Stanford, California. (2002).

Uribe, M.H.C., Dacillo, D.B., Dacoag, L.M., Andrino, R.P. Jr., and Alcober, E.H.: 30 years of Tongonan-1 (Leyte, Philippines) sustained production. *Proc. World Geothermal Congress 2015*, Melbourne, Australia. (2015).

Yamamoto, H.: PetraSim: A Graphical User Interface for the TOUGH2 Family of Multiphase Flow and Transport Codes. Ground Water, 46: 525–528. doi:10.1111/j.1745-6584.2008.00462.x. (2008).