

Dealing with Model Uncertainty and Deficiencies when Assimilating Thermal Breakthrough Observations

*Elvar Karl Bjarkason¹, Anna Suzuki¹

1. Tohoku University

Thermal breakthrough models are simplified representations of thermal and fluid transport processes occurring within geothermal systems which consist of hot, fractured porous rocks. Since simple thermal breakthrough models only require modest computational resources and are easy to use, they are widely adopted in practice to describe observed temperature changes in geothermal fields. However, those models have various model deficiencies (model errors), which can severely impact uncertainty estimates and model predictions. This study considers methods to enhance data assimilation of temperature changes in geothermal fields. To account for or soften the effects of model errors, we use a total observation error covariance matrix. This total covariance matrix is used to account for both model errors as well as observation errors, which, like the model errors, might be underestimated initially by the model developers. For estimating model parameter uncertainty, we apply an iterative ensemble smoother and update the total covariance matrix iteratively based on the posterior ensemble of observation residuals given by the ensemble smoother. We applied the total covariance approach to geothermal field data from two different sites. The results suggest that the total covariance approach can provide better thermal breakthrough predictions for real geothermal field studies. The total covariance scheme gives wider predictive temperature intervals than those found when neglecting model errors. Consequently, the total covariance approach gives a better portrayal of potential future temperature changes. In conjunction with the total covariance approach, we, additionally, discuss the importance of refining imperfect thermal breakthrough models to improve model predictions under uncertainty.

Keywords: Data assimilation, Model error, Model uncertainty, Geothermal, Ensemble smoother