Analysis Report for Preparation of the 2020 Culebra Potentiometric Surface Contour Map

Task Number: 4.4.2.3.1

Report Date: 5/6/2021

Author:	*Signature on file*	5/6/21
	Amelia M. Hayes, 8882	Date
	Repository Performance	
Technical Review:	*Signature on file*	5/6/21
	Patricia Johnson, 8882	Date
	Repository Performance	
QA Review:	Steve Davis	5/6/21
	Steve Davis, 8880	Date
	Carlsbad Programs Group	
Management Review:		5/6/21
	Antonio Triventi, 8882	Date
	Manager, Repository Performance	

WIPP: 4.4.2.3.1 TD: QA-L: RECERT: 549085

Information Only

Table of Contents

1 Introduction 3
2 Scientific Approach 3
2.1 Observed Data 4
2.2 Modeling Overview 6
2.3 Creation of the Averaged MODFLOW Simulation 8
2.4 Boundary Conditions 8
2.5 PEST Calibration of Averaged MODFLOW Model to Observations 9
2.6 Particle Track and Figure Generation from Averaged MODFLOW Model 10
3 2020 Results 11
3.1 2020 Equivalent Freshwater Head Contours 11
3.2 2020 Particle Track 14
3.3 2020 Measured vs. Modeled Fit 14
4 Conclusions 19
5 References 20
6 Run Control Narrative 22
6.1 Input Files 28
6.2 Output Files 29

Information Only

1. Introduction

This report documents the preparation of the 2020 potentiometric contour map and associated particle track for the Culebra Member of the Rustler Formation in the vicinity of the Waste Isolation Pilot Plant (WIPP). The Stipulated Final Order sent to the New Mexico Environment Department (NMED) on May 28, 2009 (Moody, 2009) requires, or is the driver for, this analysis. This Analysis Report follows Sandia National Laboratories' Specific Procedure (SP) 9-9 (Kuhlman, 2009), which was prepared in response to this NMED driver. This report uses December 2020 data obtained from the WIPP Management & Operations Contractor for contour mapping.

The contour mapping analysis utilizes a modified version of the WIPP Performance Assessment (PA) calibrated Culebra groundwater flow model "T-fields" (a broad term used to describe parameters including transmissivity, recharge, and horizontal anisotropy [Hart et al., 2009]). Simulated equivalent Culebra freshwater heads were created by averaging the ensemble of 100 calibrated PA MODFLOW "T-fields". The averaged MODFLOW model was then adjusted using PEST to improve the match between the model-predicted heads and the December 2020 observed freshwater heads at Culebra well locations. The result is the averaged, PEST-calibrated MODFLOW model used to compute the contour lines for the potentiometric surface map and the annual advective particle track forward from the WIPP waste-handling shaft to the Land Withdrawal Boundary (LWB).

The averaged MODFLOW model captures the bulk steady-state Culebra flow behavior. The process of adjusting the boundary conditions of the averaged steady-state MODFLOW model outlined in SP 9-9 (Kuhlman, 2009) cannot match the effects of transient processes such as well maintenance, pumping events, etc. (Hayes, 2018). Because transient processes cannot be replicated with a steady-state model, the differences between the measured and modeled freshwater heads are presented and the causes of discrepancies are discussed in this report.

2. Scientific Approach

The potentiometric surface map for a given time period is generated using the following steps from SP 9-9:

- Examine hydrographs to identify the time period having the largest number of Culebra water levels available with the fewest wells affected by pumping or other anthropogenic events. (Section 2.1)
- Convert water levels from target time period to equivalent freshwater heads using fluid densities appropriate to the date. (Section 2.1)
- Fit trend surface through freshwater heads. (Section 2.3)
- Extrapolate the trend surface to the boundaries of the model domain used in PA and define initial fixed-head boundary conditions based on the trend surface. (Section 2.4)

Information Only

- Using the ensemble-average Culebra "T-fields" from the PA, optimize the model boundary heads using optimization software interactively with the groundwater flow model to improve the fit of the model to the freshwater heads. (Section 2.5)
- Run the groundwater flow model with optimal boundary conditions. (Section 2.5)
- Contour groundwater flow model head results on WIPP site map. (Section 2.6)
- Compute particle path and travel time from the waste handling shaft to the LWB. (Section 2.6)

2.1 Observed data

The observed data used in this analysis was supplied by the monthly water level data from Regulatory Environmental Services (RES) (Thompson, 2021). The data supplied included the depth to water, water level above mean sea level, Culebra midpoint elevation, water density, height of the water column above the midpoint, and calculated freshwater head. Table 1 lists the freshwater heads for December 2020 and the calculated densities from 2019 that were supplied by RES. The month of December was chosen because several wells were still recovering in early 2020 from the drawdown caused by the Mills Ranch well pumping that ended in 2018. In addition, choosing a different month would have resulted in more wells being excluded from the analysis because of well maintenance activities, such as the drilling and purging of wells H-5bR and WIPP-11R. Well entries have one group code indicating general geographic location: north, south, central, or excluded from mapping. Previous years included a second code that could be applied to wells affected by pumping, but this is no longer included, because pumping at the Mills Ranch well (well C-2492) ceased in January 2018 (Bowman and Hayes, 2019) and the wells in the monitoring network have recovered from drawdown that the pumping caused. The observed values are weighted based on location and distance from the LWB to improve the fit of the area of interest in the PEST calibration (see section 2.5).

Table 1. Freshwater head calibration targets used in PEST. Measurement dates, freshwater head, and specific gravity values were taken from Thompson 2021. Green text indicates wells not used because of long-term recovery or recovery from development activities. Code: N=north, S=south, C=central, X=excluded.

Well	Code	Observation Weight	Measurement Date	Freshwater Head (ft AMSL)	Calculated Density (g/cm³)
AEC-7R	N	0.4	12/07/20	3058.79	1.065
C-2737	S	2.5	12/09/20	3009.67	1.024
ERDA-9	C	2.5	12/09/20	3023.90	1.073
H-4bR	S	2.5	12/08/20	2995.98	1.020
H-5bR*	X	-	-	-	-
H-6bR	N	2.5	12/08/20	3061.76	1.039
H-9bR	S	0.4	12/07/20	2970.72	1.005
H-10cR	S	0.4	12/07/20	3019.92	1.087
H-11b4R	S	2.5	12/08/20	2995.66	1.078
H-12R	S	0.4	12/07/20	3004.26	1.104

Information Only

Well	Code	Observation Weight	Measurement Date	Freshwater Head (ft AMSL)	Calculated Density (g/cm³)
H-15R	S	2.5	12/09/20	3009.33	1.125
H-16	C	2.5	12/09/20	3038.02	1.032
H-19b0	S	2.5	12/08/20	3002.46	1.066
IMC-461	C	0.4	12/08/20	3032.45	1.000
SNL-1	N	0.4	12/08/20	3073.45	1.033
SNL-2	N	1.0	12/08/20	3060.79	1.010
SNL-3	N	2.5	12/08/20	3070.98	1.029
SNL-5	N	1.0	12/08/20	3062.35	1.007
SNL-6†	X	-	12/07/20	3435.61	1.245
SNL-8	N	1.0	12/07/20	3051.76	1.091
SNL-9	C	1.0	12/08/20	3045.40	1.018
SNL-10	C	2.5	12/08/20	3045.93	1.011
SNL-12	S	0.4	12/07/20	2989.31	1.010
SNL-13	S	1.0	12/07/20	3000.38	1.024
SNL-14	S	1.0	12/08/20	2992.35	1.047
SNL-15†	X	-	12/07/20	3100.73	1.231
SNL-16	S	0.4	12/07/20	3006.26	1.017
SNL-17	S	1.0	12/07/20	3000.77	1.014
SNL-18	N	0.4	12/08/20	3064.53	1.011
SNL-19	N	0.4	12/08/20	3060.94	1.006
WIPP-11R	N	1.0	12/08/20	3072.23	1.036
WIPP-13	N	0.4	12/08/20	3068.14	1.035
WQSP-1	N	0.4	12/09/20	3063.99	1.041
WQSP-2	N	0.4	12/09/20	3067.12	1.033
WQSP-3	N	0.4	12/09/20	3062.04	1.141
WQSP-4	S	0.4	12/09/20	3004.50	1.078
WQSP-5	S	0.4	12/08/20	3002.03	1.029
WQSP-6	S	0.4	12/08/20	3014.21	1.016

*December 2020 data not supplied for well H-5bR because well is under development
†Wells are in long-term-recovery from development

Information Only

2.2 Modeling Overview

Table 2 contains a summary of the software used in this analysis and the quality assurance (QA) status for each. Steady-state groundwater flow simulations were carried out using software similar to that used for the WIPP Compliance Recertification Application 2009 Performance Assessment Baseline Calculation (CRA-2009 PABC), as presented in Hart et al. 2009, and in CRA-2014 (DOE, 2014). The sets of 100 Culebra parameter fields for transmissivity (*T*), anisotropy (*A*) and Recharge (*R*) from the original PA MODFLOW model were averaged to create a single ensemble averaged groundwater flow model field. This calibrated field is referred to as the "averaged MODFLOW model" to distinguish it from the PA MODFLOW model. The averaged MODFLOW model was used to predict regional Culebra groundwater flow across the WIPP site. Section 2.3 describes the creation of the averaged MODFLOW model.

Section 2.4, Boundary Conditions, describes the types of boundary conditions taken from the PA MODFLOW model and the calibration of the averaged MODFLOW model boundary conditions using PEST. The calibration process is analogous to the PEST calibration of the PA MODFLOW model described in Hart et al. 2009; however, a simplified process was used in this analysis. PEST was used to modify a subset of the averaged MODFLOW model boundary conditions (see red boundaries in Figure 1) rather than adjusting the T-fields *T*, *A*, and *R* individually. This choice was made to avoid results of multiple contour maps, which would be difficult to present. The PEST calibration targets for the averaged MODFLOW model represent the December 2020 measured annual freshwater heads at Culebra monitoring wells.

Section 2.5, PEST Calibration of Averaged MODFLOW Model to Observations, describes the fitting of the averaged MODFLOW model to the observed Culebra well data using PEST. Contours are drawn over an area encompassing and surrounding the WIPP site from the results of the fit averaged MODFLOW model using Python (a commercial off the shelf software). The contour lines are extrapolated to the PA model domain boundaries, which are labeled as the Active MODFLOW Boundaries in Figure 1. A subset of the contour map is used (green rectangle surrounding the WIPP LWB in Figure 1) for computing the path of a conservative (i.e. non-dispersive and non-reactive) particle taken through the Culebra from the waste handling shaft to the LWB. The particle track is computed from the averaged MODFLOW model flow field using DTRKMF. MODFLOW, PEST, DTRKMF, and the Bash and Python input files and scripts written for this work were executed on the WIPP PA Solaris cluster (santana.sandia.gov) running SunOs 5.11. The creation of figures was done using Python scripts on an Intel-Xeon-equipped Dell desktop computer running Windows 10 Enterprise.

Information Only

Table 2. Status of key software used in analysis

Software	Version	Description	Platform	Software QA Status
MODFLOW-2000	1.07	Groundwater flow model	SunOS PA cluster	Acquired; qualified under NP 19-1 (Harbaugh et al., 2000)
PEST	9.12	Automatic parameter estimation code	SunOS PA cluster	Developed; qualified under NP 19-1 (Doherty, 2002)
DTRKMF	1.01	Particle tracker	SunOS PA cluster	Developed; qualified under NP 19-1 (Treadway, 2014)
Python	2.7.9	Scripting language (file manipulation)	SunOS PA cluster	Commercial off the shelf
Python	2.7.11	Scripting language (plotting)	Windows 10 desktop	Commercial off the shelf
Bash	4.3.33	Scripting language (file manipulation)	SunOS PA cluster	Commercial off the shelf

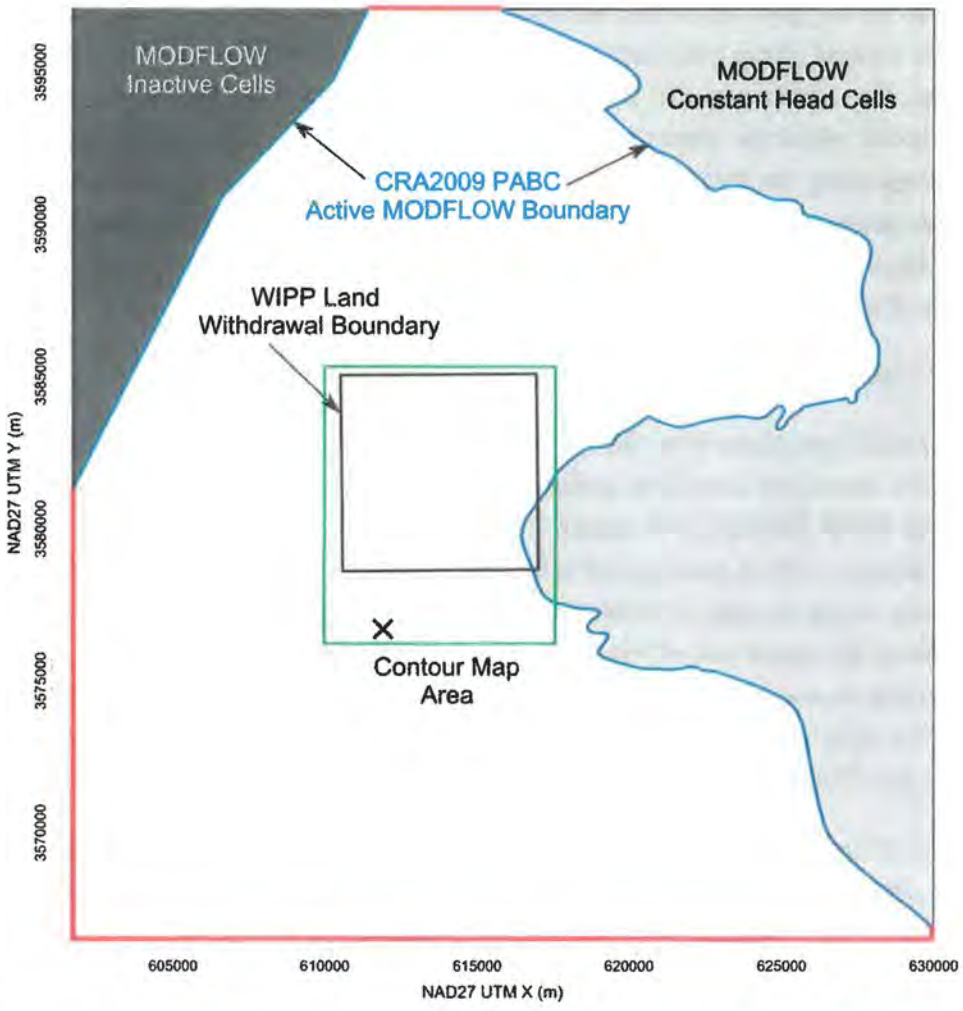

Figure 1. MODFLOW model domain, adjusted boundary conditions shown in red, contour map area outlined in green. Mills Ranch well location indicated with "×" (from Kuhlman, 2017)

Information Only

2.3 *Creation of the Averaged MODFLOW Simulation*

An averaged MODFLOW model is used to compute the equivalent freshwater head and cell-by-cell flow solution. The computed heads are contoured, and the flow solution is used to compute particle tracks. The MODFLOW model grid is a single 7.5-m thick layer, comprising 307 rows and 284 columns; each model cell is a 100-meters square. The modeling area spans 601,700 to 630,000 meters in the east-west direction, and 3,566,500 to 3,597,100 meters in the north-south direction, both in Universal Transverse Mercator (UTM) North American Datum 1927 (NAD27) coordinates, zone 13 north.

The calibrated T, A, and R parameter fields from the PA MODFLOW model were checked out of the PA version control repository using the checkout_average_run_modflow.sh script (scripts are listed in Section 6; input and output files are available from the WIPP version control system in the repository /nfs/data/CVSLIB/Analyses/SP9_9). Model inputs can be divided into two groups. The first group includes model inputs that are common across all 100 calibrated realizations; these include the model grid definition, the boundary conditions, and the model solver parameters. The second group includes the T, A, and R fields, which are different for each of the 100 realizations. The constant model inputs in the first group are used directly in the averaged MODFLOW model, while the inputs in the second group were averaged across all 100 calibrated model realizations using the Python script average_realizations.py. All three averaged parameters were geometrically averaged (i.e., the arithmetic average was computed from the $\log_{10}$ values, which was then exponentiated to give the resulting value), since they vary over multiple orders of magnitude.

2.4 *Boundary Conditions*

The boundary conditions taken from the PA MODFLOW model are used as the initial conditions from which PEST boundary condition calibrations proceed. There are two types of boundary conditions in the WIPP MODFLOW model. The first type of condition includes geologic or hydrologic boundaries, which correspond to known physical features in the flow domain. The no-flow boundary along the axis of Nash Draw is a hydrologic boundary (the boundary along the dark gray region in the upper left of Figure 1). The constant-head boundary along the halite margin corresponds to a geologic boundary (the eastern irregular boundary adjoining the light gray region in the right of Figure 1). Physical boundaries are believed to be well known and are not adjusted in this PEST calibration.

The second type of boundary condition includes the constant-head cells along the rest of the model domain. This type of boundary includes the straight-line southern, southwestern, and northern boundaries that coincide with the primary compass directions and the rectangular frame surrounding the model domain (shown as heavy red lines in Figure 1). The value of specified head assigned in boundary cells corresponds with this second boundary type and is adjusted in the PEST calibration process.

Information Only

The Python script boundary_types.py is used to distinguish between the two different types of specified head boundary conditions based on the specified head value used in the PA MODFLOW model. All constant-head cells (specified by a value of -1 in the MODFLOW IBOUND array from the PA MODFLOW model) with a starting head value greater than 3280 feet above mean sea level (AMSL) are left fixed and not adjusted in the PEST optimization, because they correspond to the no-flow constant head region to the east of WIPP. The remaining constant-head cells are distinguished by setting their IBOUND array value to -2 (which is still interpreted as a constant-head value by MODFLOW, but allows simpler discrimination between boundary conditions in Python scripts elsewhere in this analysis).

Using output from boundary_types.py, the Python script surface_02_extrapolate.py computes initial head at active model cells (IBOUND=1) and the specified constant-head at adjustable boundary condition cells (IBOUND=-2), given parameter values for the surface to extrapolate.

2.5 PEST Calibration of Averaged MODFLOW Model to Observations

There are three major types of inputs to PEST. The first input class is the "forward model", which includes the entire MODFLOW model setup derived from the PA MODFLOW model described in the previous section, along with any pre- or post-processing scripts or programs needed. These files comprise the forward model PEST runs repeatedly to estimate sensitivities of model outputs to model inputs. The second input type includes the PEST configuration files, which list parameter and observation groups, observation weights, and indicate which parameters in the MODFLOW model are adjusted in the inverse simulation. The third input class includes the set of observed freshwater head values. Freshwater head values from December 2020 used as targets for the PEST calibration are listed in Table 1, which are specified along with weights in the PEST configuration files. In contrast to Table 1, well names used in figures do not include leading zeros or hyphens and are capitalized.

To minimize the number of estimated parameters, and to ensure a degree of smoothness in the specified constant-head boundary condition values, a parametric surface is used to extrapolate the heads to the estimable boundary conditions. The surface is of the same form described in the Analysis Report for Task 7 of AP-114. The parametric surface is given by the following equation:

$$h(x,y) = A + B(y + D\,\mathrm{sign}(y)\,\mathrm{abs}(y)^{\alpha}) + C(Ex^3 + Fx^2 - x) \qquad (1)$$

where $\mathrm{abs}(y)$ is absolute value and $\mathrm{sign}(y)$ is the function returning 1 for $y > 0$, -1 for $y < 0$ and 0 for $y = 0$ and x and y are coordinates scaled to the range $-1 \le \{x,y\} \le 1$. In Hart et al. (2009), the values $A = 928$, $B = 8$, $C = 1.2$, $D = 1$, $E = 1$, $F = -1$ and $\alpha = 0.5$ are used with the above equation to assign the boundary conditions in the PA MODFLOW model.

Information Only

PEST was used to estimate the values of parameters *A, B, C, D, E, F*, and α given the observed heads in Table 2. The Python script surface_02_extrapolate.py was used to compute the MODFLOW starting head input file (which is also used to specify the constant-head values) from the parameters *A-F* and α. Each forward run of the model corresponded to a call to the Bash script run_02_model. This script called the surface_02_extrapolate.py script, the MODFLOW-2000 executable, and the PEST utility mod2obs, which is used to extract and interpolate model-predicted heads from the MODFLOW output files at observation well locations.

The PEST-specific input files were generated from the observed heads using the Python script create_pest_02_input.py. The PEST input files include the instruction file (how to read the MODFLOW output), the template files (how to write the MODFLOW input), and the PEST control file (listing the ranges and initial values for the estimable parameters and the values and weights associated with observations). Weights were compiled using scientific judgement and trial-and-error (different combinations of weights were tried). Higher observation weights (2.5) were assigned to wells inside the LWB, lower observation weights (0.4) were assigned to wells distant from the WIPP site, wells near the WIPP LWB were assigned an intermediate weight (1.0) (see Table 1). Additional observations representing the average heads north of the LWB and south of the LWB were used to help prevent over-smoothing of the estimated results across the LWB. The additional observations and weights were assigned to improve the fit in the area of interest (inside the WIPP LWB), possibly at the expense of a somewhat poorer fit far from the WIPP LWB and closer to the boundary conditions.

2.6 Particle Track and Figure Generation from Averaged MODFLOW Model

The MODFLOW model is run predictively using the averaged MODFLOW model parameters, along with the PEST-calibrated boundary conditions. The resulting cell-by-cell flow budget is then used by DTRKMF to compute a particle track from the waste-handling shaft to the WIPP LWB. Particle tracking stops when the particle crosses the WIPP LWB. The Python script convert_dtrkmf_output_for_surfer.py converts the MODFLOW cell-indexed results of DTRKMF into a UTM *x* and *y* coordinate system, saving the results in the Surfer blanking file format to facilitate plotting results. The heads in the binary MODFLOW output file are converted to an ASCII matrix file format using the Python script head_bin2ascii.py.

The resulting particle track and contours of the model-predicted head are plotted using a matplotlib Python script for an area including the WIPP LWB, corresponding to the region shown in previous versions of the Annual Site Environmental Report (ASER) (e.g., see Figure 6.11 in DOE, 2008, specifically the green box in Figure 1). The modeled heads extracted from the MODFLOW output by mod2obs are then merged into a common file for plotting using the Python script merge_observed_modeled_heads.py.

Information Only

3. 2020 Results

3.1 2020 Equivalent Freshwater Head Contours

The model-generated freshwater head contours are given in Figure 2 and Figure 3. There is a roughly east-west trending band of steeper gradients, corresponding to lower Culebra transmissivity. The uncontoured region to the right of the purple line in the eastern part of the figures corresponds to the portion of the Culebra that is located stratigraphically between halite and other members of the Rustler Formation (Tamarisk Member above and Los Medaños Member below). This region east of the "halite margin" has a high freshwater head but extremely low transmissivity, essentially serving as a no-flow boundary in this area.

Information Only

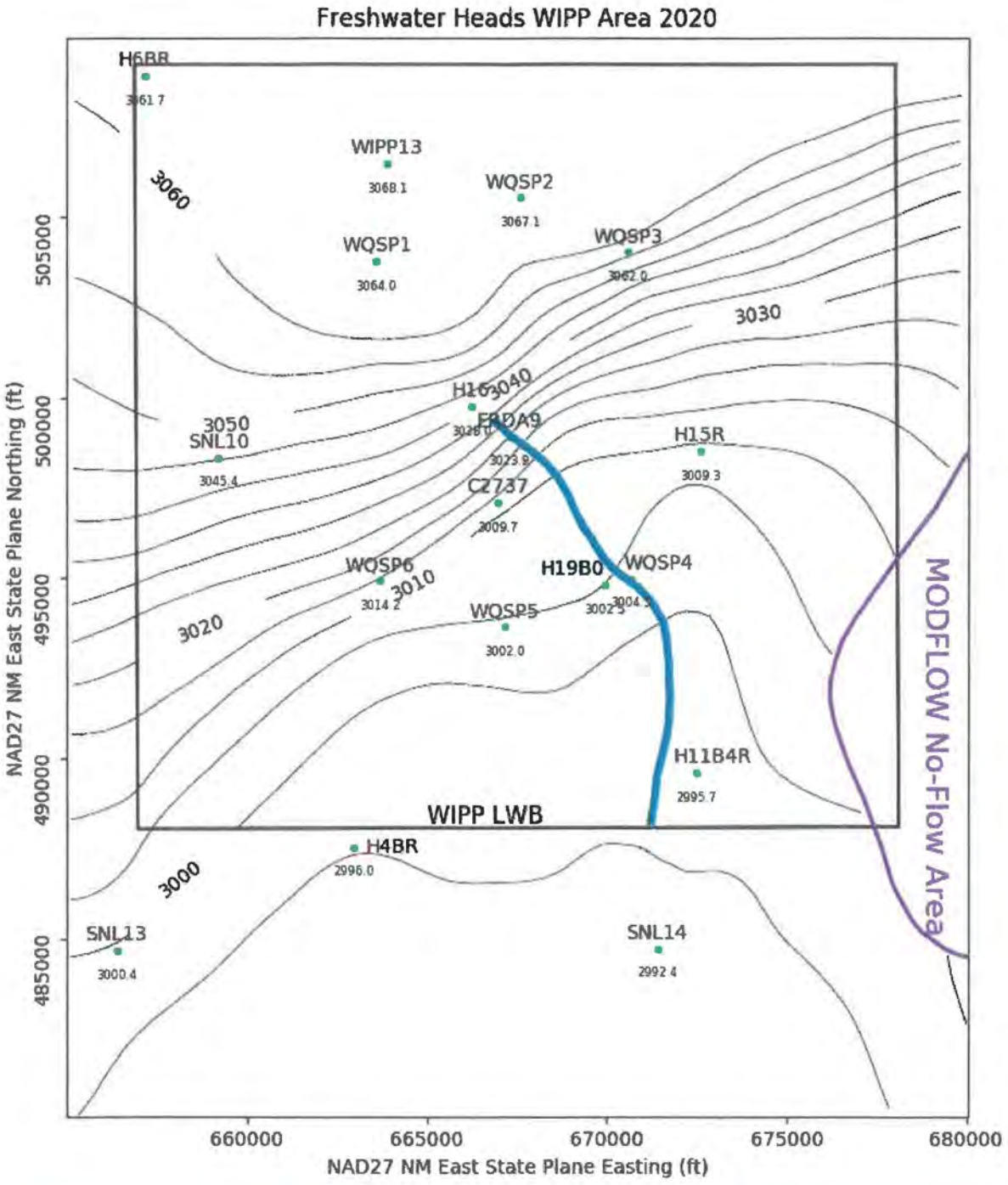

Figure 2. Model-generated December 2020 freshwater head contours with observed heads (ft) listed at each well (5-foot contour intervals) with blue line particle track from waste handling shaft to WIPP LWB. Purple line represents Rustler halite margin.

Information Only

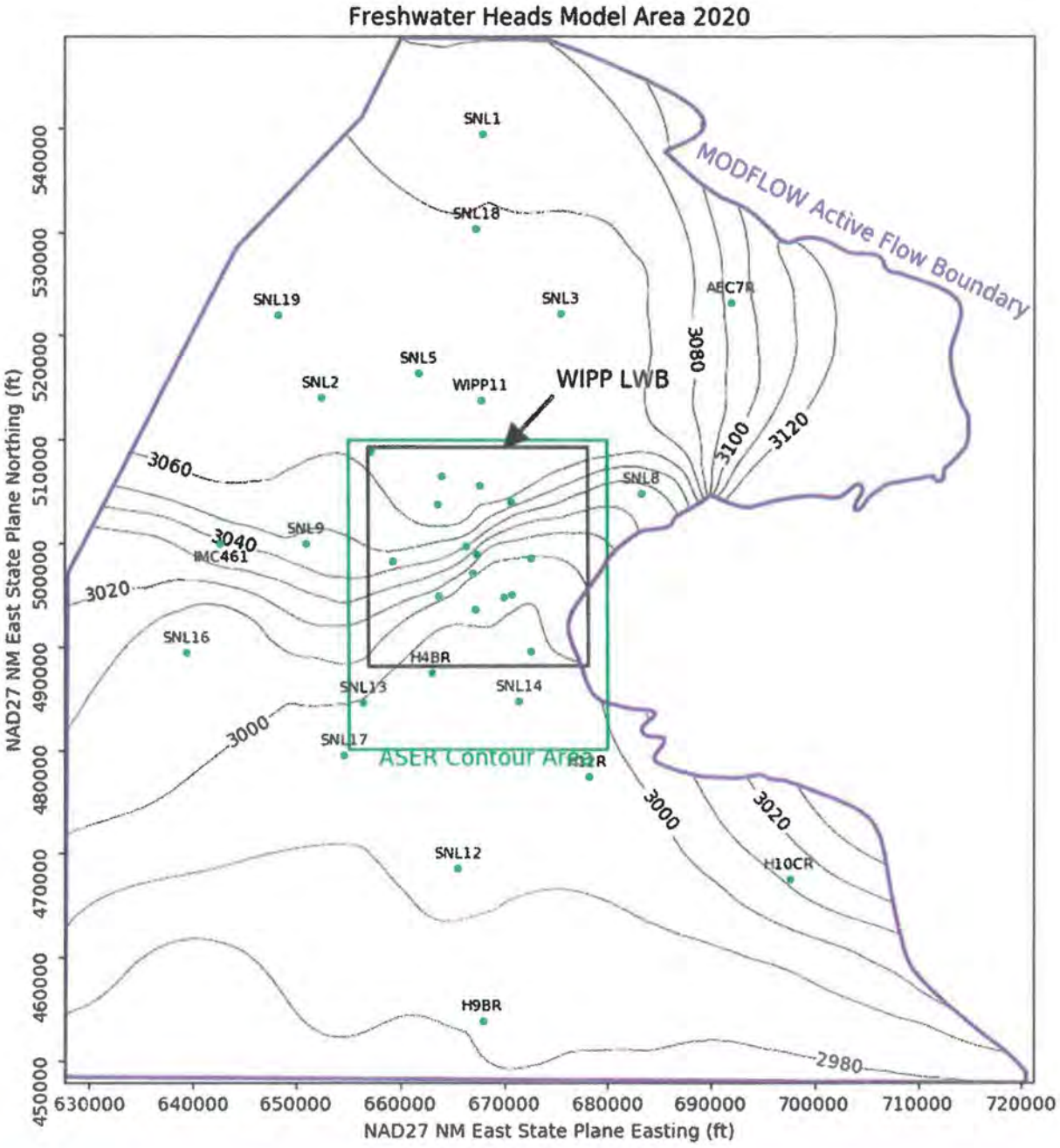

Figure 3. MODFLOW-modeled December 2020 heads for entire model domain (10-foot contour interval). Green rectangle indicates region contoured in Figure 2, inner black square is WIPP LWB. Wells within the LWB are not labeled due to the high density of wells. Refer to Figure 2 for identification of wells within the LWB.

Information Only

3.2 2020 Particle Track

The blue line in Figure 2 represents the DTRKMF-calculated path a water particle would track through the Culebra from the coordinates corresponding to the WIPP waste handling shaft to the LWB (a path length of 4098 m [13445 ft]). The DTRKMF output is adjusted to reflect the assumptions that the transmissive portion of the Culebra is 4 m (13 ft) thick (as opposed to the entire thickness of the aquifer at 7.5 m [25 ft]) and has a constant porosity of 16% based on the geologic properties of the Culebra dolomite [Holt, 1997]. The resulting travel time to the WIPP LWB from the waste handling shaft is 5934 years. This is an average velocity of 0.69 m/yr.

3.3 2020 Measured vs. Modeled Fit

The scatter plot in Figure 4 shows measured and modeled freshwater heads at the observation locations used in the PEST calibration. The observations are divided into three groups, based on proximity to the WIPP site. Wells within the LWB are represented by red crosses, wells outside but within 3 km of the LWB are represented with green '×'s, and other wells within the MODFLOW model domain but distant from the WIPP site are indicated with blue stars. Additional observations representing the average heads north of the LWB and south of the LWB were used to help prevent over-smoothing of the estimated results across the LWB. This allowed PEST to improve the fit of the model to observed heads inside the area contoured in Figure 2, at the expense of fitting wells closer to the boundary conditions (i.e., wells not shown in Figure 2).

Information Only

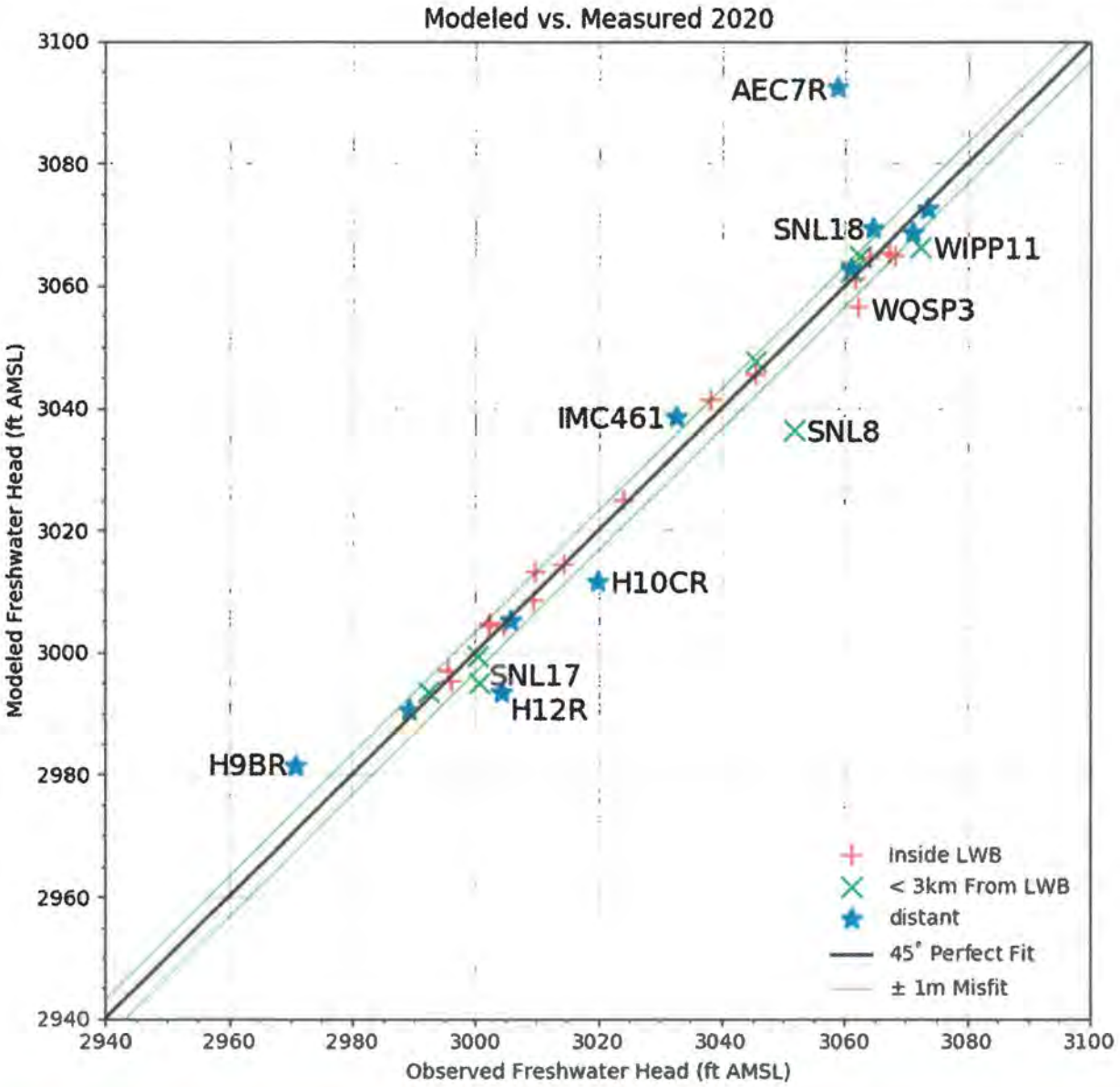

Figure 4. Measured vs. modeled scatter plot for averaged MODFLOW model generated heads and December 2020 observed freshwater heads.

The central black diagonal line in Figure 4 represents a perfect model fit (1:1 or 45-degree slope); the two green lines on either side of this represent a 1-m misfit above or below the perfect fit. Wells more than 1.5 m (4.9 ft) from the 1:1 line are labeled.

The squared correlation coefficient (R^2) of the measured versus modeled data is listed in Table 3. Figure 5 and Figure 6 show the distribution of errors resulting from the PEST-adjusted fit to observed data. The wells within the WIPP LWB have a weighted R^2 of 94 % and wells within 3 km of the WIPP LWB have a weighted R^2 of greater than 97%. Figure 6 shows the individual residuals for each well grouped by location. Well AEC-7R is a notable outlier in the measured

Information Only

versus modeled head values. This discrepancy is caused by the proximity of AEC-7R to the MODFLOW active flow boundary in the northeast coupled with the well location being in a low transmissivity portion of the Culebra.

This report presents a contour map (in Figures 2 and 3) which represents the Culebra freshwater piezometric surface as it would possibly look if it were unperturbed by any outside influence. Figure 7 shows the difference between the unperturbed freshwater head and the observed freshwater head. Previous potentiometric maps had a large discrepancy between the unperturbed freshwater head and the observed freshwater head due to pumping at the commercial well C-2492 (known as the Mills Ranch well). For example, the difference between the unperturbed freshwater head and the observed freshwater head in the analysis for the 2017 data, when there was active pumping at C-2492, was over 30 ft for five wells within the LWB (Hayes, 2017). As seen in Figure 7, this discrepancy has been greatly reduced with the largest discrepancy within the LWB at well WQSP-3 (5.6 ft) indicating that water levels have recovered from the pumping.

Table 1. 2020 Measured vs modeled correlation coefficients

	Dataset	Measured vs. Modeled R^2
Uncalibrated	wells inside WIPP LWB	0.9447
	wells <3km from WIPP LWB	0.9792
	all wells	0.9947
Calibrated	wells inside WIPP LWB	0.9443
	wells <3km from WIPP LWB	0.9804
	all wells	0.9946

Information Only

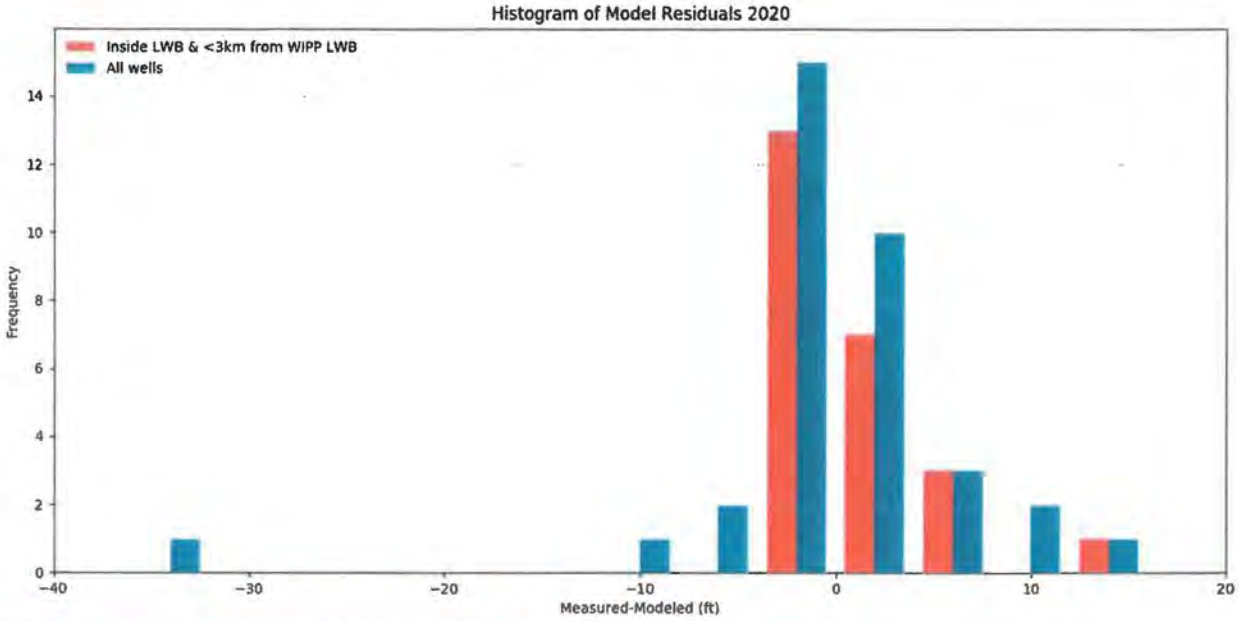

Figure 5. Histogram of measured-modeled errors for 2020

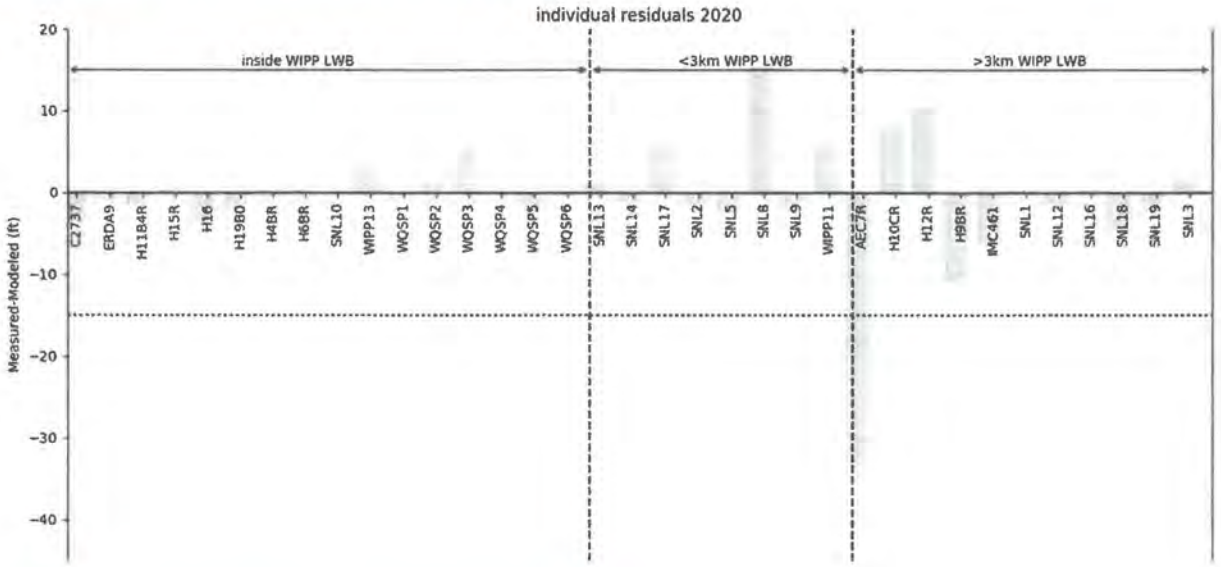

Figure 6. Measured-modeled errors at each well location for 2020

Information Only

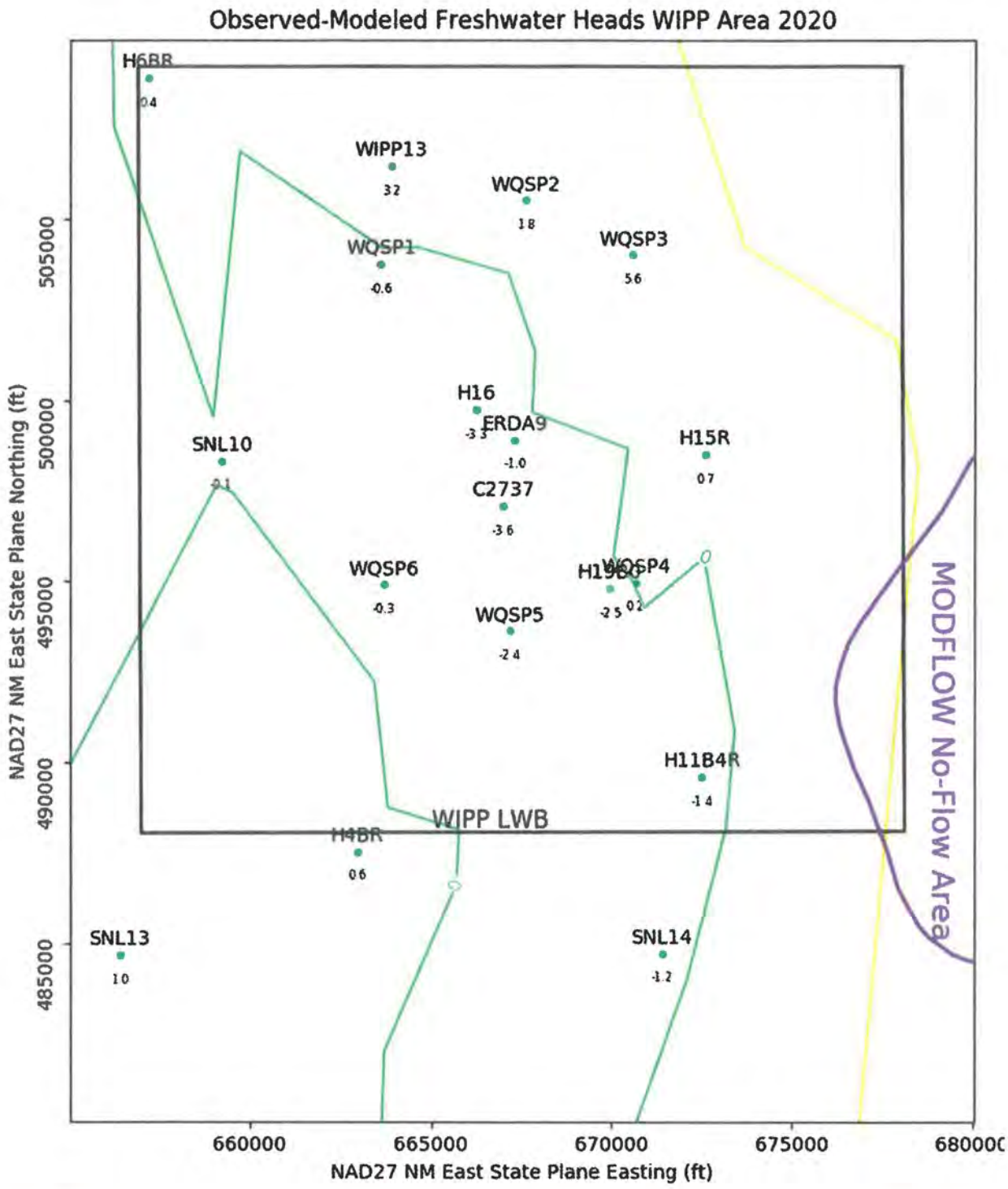

Figure 7. Triangulated contours for observed minus model-predicted freshwater head. The contour interval is 10 ft. The two green 0 ft contours present show mark the areas where the model calculated a higher head value than measured versus a lower head value than measured.

Information Only

4. Conclusions

The model fit to the December 2020 freshwater head observations shows improvements from past years. The modeled contours reflect that the water levels within the LWB have recovered from the historical pumping activities at C-2492, which ceased in January 2018. The travel time for a particle to the WIPP LWB from the waste handling shaft is 5934 years with a path length of 4098 m, an average velocity of 0.69 m/yr. The travel time calculated in this analysis was 45 years shorter than the travel time calculated for the 2018 potentiometric surface map (the first analysis after pumping ceased at C-2492), but the path length was 28 m longer (Hayes, 2019). The travel time calculated in this analysis was 345 years longer than the travel time calculated for the 2019 Annual Site Environmental Report but has and 23 m shorter path length (DOE, 2020). The 2020 particle track length is 28 meters longer than the 2018 particle track.

Information Only

5. References

Bowman, D.O., and Hayes, A. 2019. *WIPP Milestone Report: 2018 Culebra Monitoring Network Groundwater Level Fluctuations*. Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 571630

Doherty, J. 2002. *PEST: Model Independent Parameter Estimation*. Watermark Numerical Computing, Brisbane, Australia.

U.S. Department of Energy (DOE). 2008. *Waste Isolation Pilot Plant Annual Site Environmental Report for 2007*. DOE: Carlsbad New Mexico. DOE/WIPP-08-2225

DOE, 2014. *Compliance Recertification Application 2014 for the Waste Isolation Pilot Plant*. DOE: Carlsbad, New Mexico. DOE/WIPP-14-3503

DOE, 2020. *Waste Isolation Pilot Plant Annual Site Environmental Report for 2019*. DOE: Carlsbad, NM. DOE/WIPP-20-3591

Harbaugh, A.W., E.R. Banta, M.C. Hill, and M.G. McDonald. 2000. *MODFLOW-2000, the U.S. Geological Survey modular ground-water model – User guide to modularization concepts and the Ground-Water Flow Process*. U.S. Geological Survey Open-File Report 00-92.

Hart, D.B., S.A. McKenna, and R.L. Beauheim. 2009. *Analysis Report for Task 7 of AP-114: Calibration of Culebra Transmissivity Fields*. Sandia National Laboratories: Carlsbad, NM, ERMS 552391.

Holt, R.M. 1997. *Conceptual Model for Transport Processes in the Culebra Dolomite Member of the Rustler Formation*. Sandia National Laboratories: Albuquerque, NM. SAND97-0194.

Hayes, A.M. 2018. *Analysis Report for Preparation of the 2017 Culebra Potentiometric Surface Contour Map*. Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 570574

Hayes, A. M. 2019. *Analysis Report for Preparation of the 2018 Culebra Potentiometric Surface Contour Map*. Sandia National Laboratories: Carlsbad, NM. WIPP Records Center. ERMS 571792

Kuhlman, K.L. 2009. *Procedure SP 9-9, Revision 0, Preparation of Culebra Potentiometric Surface Contour Maps*. Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 552306

Kuhlman, K.L. 2017. *Analysis Report for Preparation of 2016 Culebra Potentiometric Surface Contour Map*. Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 552306

Information Only

Moody, D.C. 2009. *Stipulated Final Order for Notice of Violation for Detection Monitoring Program*, Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 551713.

Thompson, Hayden. 2021. *WIPP Groundwater level Measurements for December 2020* (provided by Regulatory Environmental Services). Sandia National Laboratories: Carlsbad, NM.

Treadway, A. 2014. *DTRKMF Version 1.01.* Sandia National Laboratories: Carlsbad, NM. WIPP Records Center, ERMS 562205

Information Only

6. Run Control Narrative

This section is a narrative describing the calculation process mentioned in the text, which produced the provided figures.

Figure 8 gives an overview of the driver script checkout_average_run_modflow.sh (§A-4.1); this Bash shell script (listed in Section 6.1) first exports the 4 parameter fields used in the TFIELDS calibration process (transmissivity (*T*), anisotropy (*A*), and recharge (*R*), and storativity (*S*)) from CVS version control for each of the 100 realizations of MODFLOW, listed in the file keepers (see lines 17-26 of script). In the steady-state simulations done for this report, the *S* field is not used. Some of the realizations are inside the Update or Update2 subdirectories in CVS, which complicates the directory structure. An equivalent list included in keepers_short is made from keepers, and the directories are moved to match the flat directory structure (lines 31-53). At this point, the directory structure has been modified but the MODFLOW input files checked out from CVS are unchanged.

Python script average_realizations.py (§A-4.2) is called, which first reads in the keepers_short list, then reads in each of the 400 input files and computes the geometric average at each cell across the 100 realizations. The 400 input files are each saved as flattened matrices, in row-major order. The average result is saved into 4 parameter files, each with the extension.avg instead of .mod. A single value from each file, corresponding to either the cell in the southeast corner of the domain (input file row 87188 = model row 307, model column 284 for *K* [hydraulic conductivity – what is actually used by MODFLOW] and *A*) or on the west edge of the domain (input file row 45157 = model row 161, model column 1 for *R* and *S*) is saved in the text file parameter_representative_values.txt to allow checking the calculation in Excel, comparing the results to the value given at the same row of the .avg file. The value in the right column of Table 4 can be found by taking the geometric average of the values in the text file, which are the values from the indicated line of each of the 100 realizations.

The input files used by this analysis, the output files from this analysis (including the plotting scripts) are checked into the WIPP version control system (CVS) under the repository $CVSLIB/Analyses/SP9_9.

Information Only

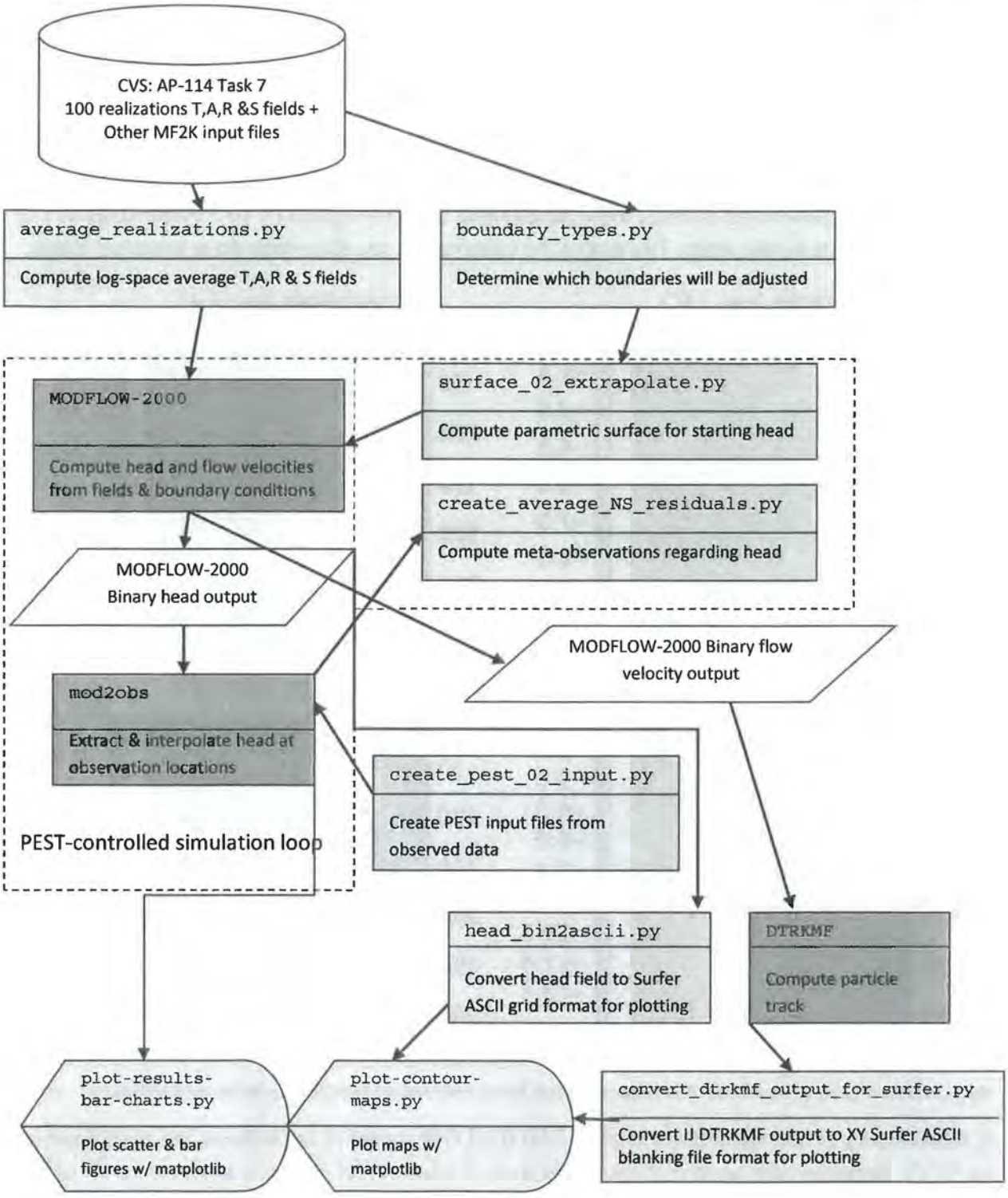

Figure 8. Process flowchart; dark gray indicates qualified programs, light gray are scripts written for this analysis

Information Only

Table 2. Averaged values for representative model cells

Field	Input file row	Model row	Model column	Geometric average
K	87188	307	284	9.2583577E-09
A	87188	307	284	9.6317478E-01
R	45157	161	1	1.4970689E-19
S	45157	161	1	4.0388352E-03

Figure 9 shows plots of the average $\log_{10}$ parameters; inactive regions ($< 10^{-15}$) were reset to 1 to improve the plotted color scale. The rest of the calculations are done with these averaged fields.

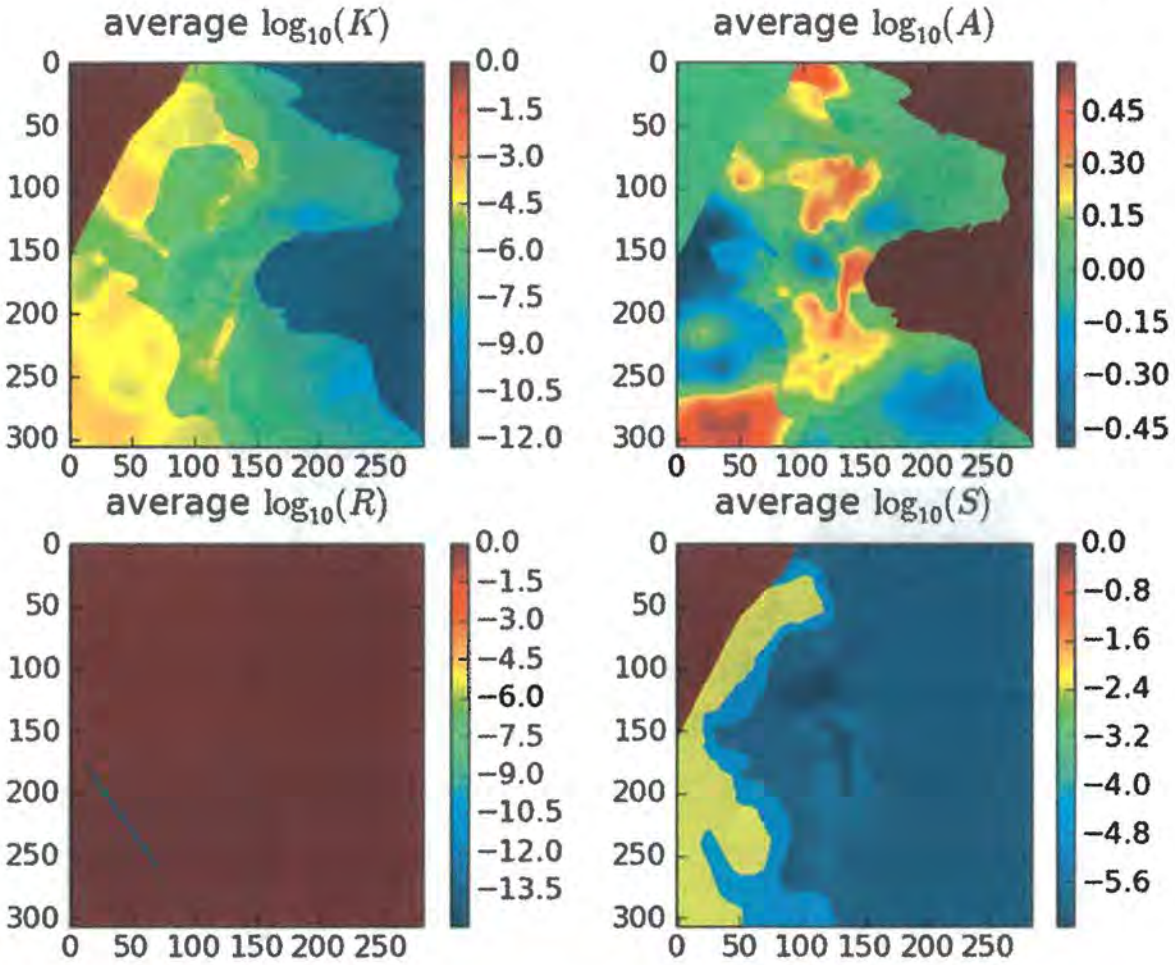

Figure 9. Plots of base-10 logarithms of average parameter fields; rows and columns are labeled on edges of figures.

A subdirectory is created, and the averaged MODFLOW model is run without any modifications by PEST. Subsequently, another directory is created where PEST is run to improve the fit of the model to observed heads at well locations.

The next portion of the driving script checkout_average_run_modflow.sh links copies the input files needed to run MODFLOW-2000 and DTRKMF into the original_average run directory. Then MODFLOW-2000 is run with the name file mf2k_head.nam, producing binary head (modeled_head.bin) and binary cell-by-cell flow budget (modeled_flow.bud) files, as well as a

Information Only

text listing file (modeled_head.lst). DTRKMF is then run with the input files dtrkmf.in and wippctrl.inp, which utilize the cell-by-cell budget file written by MODFLOW to generate a particle track output file, dtrk.out. The input file wippctrl.inp specifies the starting location of the particle in DTRKMF face-centered cell coordinates, the porosity of the aquifer (here 16%), and the coordinates of the corners of the WIPP LWB, since the calculation stops when the particle reaches the LWB.

The Python script head_bin2ascii.py (§A-4.7) converts the MODFLOW binary head file, which includes the steady-state head at every element in the flow model domain (307 rows × 284 columns) into a Surfer ASCII grid file format. This file is simply contoured in Python using matplotlib, no interpolation or gridding is needed. The Python script convert_dtrkmf_output_for_surfer.py (§A-4.9) reads the DTRKMF output file dtrk.out and does two things. First it converts the row, column format of this output file to an x, y format suitable for plotting, and second it converts the effective thickness of the Culebra from 7.75 m to 4 m. The following table shows the first 10 lines of the dtrk.out and the corresponding output of the Python script dtrk_output_original_average.bln. The first three columns of dtrk.out (top half of Table 5) after the header are cumulative time (red), column (blue), and row (green). The three columns in the blanking file (second half of Table 5) after the header are UTM NAD27 X (blue), UTM NAD27 Y (green), and adjusted cumulative time (red, which is faster than the original cumulative travel time by the factor 7.75/4=1.9375). The conversion from row, column to x, y is

$$\begin{aligned} X &= 601700 + 100 * column \\ Y &= 3597100 - 100 * row \end{aligned},$$

since the I,J origin is the northwest corner of the model domain (601700, 3597100), while the X,Y origin is the southwest corner of the domain. The blanking file is plotted directly in Python using matplotlib, since it now has the same coordinates as the ASCII head file.

Table 3. Comparison of first 10 lines of DTRKMF output and converted Surfer blanking file for `original_average`

1	159							
0.00000000E+00	118.79	150.21	1.18790000E+04	1.50210000E+04	0.00000000E+00	1.85168267E-01	1.59999996E-01	1.00000000E+00
5.53946616E+01	118.86	150.29	1.18859872E+04	1.50285080E+04	1.02562574E+01	1.85130032E-01	1.59999996E-01	1.00000000E+00
1.10787323E+02	118.93	150.36	1.18929942E+04	1.50359947E+04	2.05104788E+01	1.85094756E-01	1.59999996E-01	1.00000000E+00
1.66037959E+02	119.00	150.43	1.19000000E+04	1.50434379E+04	3.07321029E+01	1.85062532E-01	1.59999996E-01	1.00000000E+00
3.27560609E+02	119.21	150.62	1.19206651E+04	1.50624751E+04	5.88294962E+01	1.73534671E-01	1.59999996E-01	1.00000000E+00
4.89963060E+02	119.42	150.81	1.19415109E+04	1.50813473E+04	8.69490492E+01	1.73684593E-01	1.59999996E-01	1.00000000E+00
6.51458155E+02	119.62	151.00	1.19624759E+04	1.51000000E+04	1.15010608E+02	1.73860152E-01	1.59999996E-01	1.00000000E+00
7.40581455E+02	119.75	151.10	1.19749757E+04	1.51102419E+04	1.31170520E+02	1.81333000E-01	1.59999996E-01	1.00000000E+00
8.29712755E+02	119.87	151.20	1.19874963E+04	1.51204665E+04	1.47335525E+02	1.81390626E-01	1.59999996E-01	1.00000000E+00
159,1								

Information Only

```
613579.0,3582079.0,0.00000000e+00
613586.0,3582071.0,2.85907931e+01
613593.0,3582064.0,5.71815861e+01
613600.0,3582057.0,8.56866885e+01
613621.0,3582038.0,1.69285424e+02
613642.0,3582019.0,2.52884160e+02
613662.0,3582000.0,3.36232338e+02
613675.0,3581990.0,3.82235598e+02
613687.0,3581980.0,4.28238841e+02
```

The PEST utility script mod2obs is run to extract and interpolate the model-predicted heads at observation locations. The input files for mod2obs.exe were taken from AP-114 Task 7 in CVS. The observed head file has the wells and freshwater heads, but is otherwise the same as that used in the model calibration in AP-114 (Hart et al., 2009). The Python script merge_observed_modeled_heads.py (§A-4.9) simply puts the results from mod2obs and the original observed heads in a single file together for easier plotting and later analysis.

A similar process is carried out in a new directory called pest_02 (beginning line 146 of the driver script). The PEST calibration is carried out there, to keep it separate from the original_average simulation. The Python script boundary_types.py (§A-4.3) is also run to create a new MODFLOW IBOUND array, where the two different types of boundary conditions are differentiated. This Python script uses the MODFLOW IBOUND array (init_bnds_orig.inf first ⅓ of Table 6) and the initial head array (init_head_orig.mod middle ⅓ of Table 6) as inputs, and writes a new MODFLOW IBOUND array (init_bnds.inf bottom ⅓ of Table 6) with constant-head nodes indicated in red in Figure 1 marked as -2 and other constant-head nodes remaining as -1 as output. The script differentiates between these two types of boundary conditions by checking if the starting head is <1000m. Starting heads >1000m are associated with the constant-head areas to the east of the halite margins (lighter gray areas in Figure 1).

Table 4. Input IBOUND, starting head, and output IBOUND array data corresponding to first row of MODFLOW model

```
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1
```

```
943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 943 944
944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944
944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944
```

Information Only

```
944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944
944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944 944
944 944 944 944 944 944 944 944 944 944 944 1087 1085 1085 1082 1081 1080 1079 1078 1078 1077 1077 1077
1077 1077 1077 1078 1078 1077 1078 1079 1081 1083 1084 1086 1086 1088 1090 1092 1095 1096 1098 1099
1099 1100 1100 1100 1101 1101 1101 1102 1104 1104 1104 1104 1104 1104 1105 1105 1105 1105 1106 1106
1106 1107 1107 1109 1110 1110 1112 1113 1113 1114 1114 1115 1115 1115 1116 1116 1116 1117 1118 1119
1120 1121 1122 1123 1124 1125 1126 1127 1127 1129 1129 1130 1131 1132 1132 1133 1133 1134 1135 1136
1137 1137 1138 1139 1139 1140 1140 1140 1141 1142 1142 1142 1142 1142 1143 1143 1144 1144 1144 1145
1145 1146 1147 1147 1147 1147 1148 1148 1147 1146 1144 1143 1143 1145 1147 1148 1149 1150 1149 1149
1148 1149 1149 1151 1151 1151 1150 1152 1153 1154 1155
```

```
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 -2 -2 -2 -2 -2 -2 -2
-2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2 -2
-2 -2 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1
```

Table 6 shows the data corresponding to the northernmost row of the MODFLOW model domain (284 entries long) for the two input files and one output file. In the top IBOUND array, the values are either 0 or -1, indicating either inactive (the region northwest of the no-flow area shown in dark gray in Figure 1) or constant head (both red and light gray cells in Figure 1). The first 284 values from the initial head file (reformatted from scientific notation to integers to facilitate printing) show a jump from approximately 944 (in blue) to >1000 (in red). These same cells are colored in the output, showing how the initial head value is used to distinguish between the two types of constant-head boundaries. MODFLOW treats any cells that have an IBOUND entry < 0 as constant head, so both -2 and -1 are the same to MODFLOW, but allow distinguishing between them in the Python script which extrapolates the heads to the boundaries.

The required PEST input files are created by the Python script create_pest_02_input.py (§A-4.4). This script writes **1)** the PEST instruction file (modeled_head.ins), which shows PEST how to extract the model-predicted heads from the mod2obs.exe output; **2)** the PEST template file (surface_par_params.ptf), which shows PEST how to write the input file for the surface extrapolation script; **3)** the PEST parameter file (surface_par_params.par), which lists the starting parameter values to use when checking the PEST input; **4)** the PEST control file (bc_adjust_2016ASER.pst), which has PEST-related parameters, definitions of extrapolation surface parameters, and the observations and weights that PEST is adjusting the model inputs to fit. The observed heads are read as an input file in the PEST borehole sample file format (meas_head_2016ASER.smp), and the weights are read in from the input file (obs_loc_2016ASER.dat).

PEST runs the "forward model" many times, adjusting inputs and reading the resulting outputs using the instruction and template files created above. The forward model actually consists of a

Information Only

Bash shell script (run_02_model) that simply calls a pre-processing Python script surface_02_extrapolate.py (§A-4.5), the MODFLOW-2000 executable, the Python script create_average_NS_residuals.py, and the PEST utility mod2obs.exe as a post-processing step. The script redirects the output of each step to /dev/null to minimize screen output while running PEST, since PEST runs the forward model many dozens of times.

The Python script create_average_NS_residuals.py takes the output from the PEST utility mod2obs and creates a meta-observation that consists of the average residual between measured and model-prediction, only averaged across the northern or southern WIPP wells (the wells in the center of the WIPP site are not included in either group). This was done to minimize cancelation of the errors north (where the model tended to underestimate heads) and south (where the model tended to overestimate heads) of the WIPP. The results of this script are read directly by PEST and incorporated as four additional observations (mean and median errors, both north and south of WIPP).

The pre-processing Python script surface_02_extrapolate.py reads the new IBOUND array created in a previous step (with -2 now indicating which constant-head boundaries should be modified), the initial head file used in AP-114 Task 7 (init_head_orig.mod), two files listing the relative X and Y coordinates of the model cells (rel_{x,y}_coord.dat), and an input file listing the coefficients of the parametric equation used to define the initial head surface. This script then cycles over the elements in the domain, writing the original starting head value if the IBOUND value is -1 or 0, and writing the value corresponding to the parametric equation if the IBOUND value is -2 or 1. Using the parameters corresponding to those used in AP-114 Task 7, the output starting head file should be identical to that used in AP-114 Task 7.

After PEST has converged to the optimum solution for the given observed heads and weights, it runs the forward model one more time with the optimum parameters. The post-processing Python scripts for creating the Surfer ASCII grid file and Surfer blanking file from the MODFLOW and DTRKMF output are run and the results are plotted using additional Python scripts that utilize the plotting and map coordinate projection functionality of the matplotlib library.

These two plotting scripts (plot-contour-maps.py and plot-results-bar-charts.py) are included in the Section 6.2 for completeness, but only draw the figures included in this report, which are passed on to the Site Management & Operations Contractor for the ASER.

6.1 Input Files

Bytes	File Type	Description	File Name
1.5K	Python script	average 100 realizations	average_realizations.py
2.1K	Python script	distinguish different BC types	boundary_types.py

Information Only

Bytes	File Type	Description	File Name
6.2K	Bash script	main routine: checkout files, run MODFLOW-2000, run PEST, call Python scripts	checkout_average_run_modflow.sh
624	Python script	convert DTRKMF IJ output to Surfer X, Y blanking format	convert_dtrkmf_output_for_surfer.py
2.8K	Python script	create meta observations of avg heat	create_average_NS_residuals.py
3.1K	Python script	create PEST input files from observed data	create_pest_02_input.py
48	input listing	responses to DTRKMF prompts	dtrkmf.in
4.0K	Python script	convert MODFLOW-2000 binary output to Surfer ASCII grid format	head_bin2ascii.py
1.1K	input	listing of 100 realizations from CVS	keepers
1.4K	input	observed December 2020 heads in mod2obs bore sample file format	meas_head_2020ASER.smp
968	Python script	paste observed head and model-generated heads into one file	merge_observed_modeled_heads.py
76	file listing	files needed to run mod2obs	mod2obs-files.dat
139	input listing	responses to mod2obs prompts	mod2obs_head.in
372	file listing	files needed to run MODFLOW-2000	modflow_files.dat
393	input	listing of wells and geographic groupings	obs_loc_2020ASER.dat
215	file listing	files needed to run PEST	pest_02_files.dat
2.3M	input	relative coordinate $-1 \le x \le 1$	rel_x_coord.dat
2.3M	input	relative coordinate $-1 \le y \le 1$	rel_y_coord.dat
490	Bash script	PEST model: execute MODFLOW-2000 and do pre- and post-processing	run_02_model
26	input	mod2obs input file	settings.fig
47	input	mod2obs input file	spec_domain.spc
1.8K	input	mod2obs input file	spec_wells.crd
2.4K	Python script	compute starting head from parameter and coordinate inputs	surface_02_extrapolate.py
506	input	DTRKMF input file	wippctrl.inp

6.2 Output Files

Bytes	File Type	Description	File Name
19K	DTRKMF output	particle track results	dtrk.out
16K	DTRKMF output	particle track debug	drtk.dbg
1.9K	script output	heads at well locations	modeled_vs_observed_head_pest_02.txt
1.1M	script output	formatted MODFLOW-2000 heads	modeled_head-pest_02.grd

Information Only

Bytes	File Type	Description	File Name
5.3K	script output	formatted DTRKMF particle	dtrk_output_pest_02.bln
12K	PEST output	matrix condition numbers	bc_adjust_2020ASER.cnd
2.7K	PEST output	binary intermediate file	bc_adjust_2020ASER.drf
7.3K	PEST output	binary intermediate file	bc_adjust_2020ASER.jac
7.4K	PEST output	binary intermediate file	bc_adjust_2020ASER.jco
9.8K	PEST output	binary intermediate file	bc_adjust_2020ASER.jst
3.8K	PEST output	parameter statistical matrices	bc_adjust_2020ASER.mtt
477	PEST output	parameter file	bc_adjust_2020ASER.par
56K	PEST output	optimization record	bc_adjust_2020ASER.rec
4.5K	PEST output	model outputs for last iteration	bc_adjust_2020ASER.rei
8.2K	PEST output	summary of residuals	bc_adjust_2020ASER.res
28	PEST output	binary restart file	bc_adjust_2020ASER.rst
22K	PEST output	relative parameter sensitivities	bc_adjust_2020ASER.sen
3.9K	PEST output	absolute parameter sensitivities	bc_adjust_2020ASER.seo
214K	png image	matplotlib plot (Fig 2)	aser-area-contour-map2020.png
226K	png image	matplotlib plot (Fig 3)	large-area-contour-map2020.png
161K	png image	matplotlib plot (Fig 7)	aser-area-modobs-contour-map2020.png
34K	png image	matplotlib plot (Fig 5)	model-error-histogram2020.png
25K	png image	matplotlib plot (Fig 6)	model-error-residuals2020.png
117K	png image	matplotlib plot (Fig 4)	scatter_pest_02_2020.png

Information Only