Appendix 1—Input Instructions

In this appendix, the general strategy for input instructions is described. The input is arranged in a file called <casename>.bgp, which is made up of input blocks, as discussed below. Following a discussion of more detail of the general input protocols, subsections are presented in which specific input blocks are discussed, including variables and data that can be inserted.

General Structure of Input

The general input structure is designed on a subset of the JUPITER protocol (Banta and others, 2006). The advantage of this protocol over XML or the previous input format for PEST is that annotations that are easily read by humans are part of the input protocol. The full JUPITER protocol, however, has memory and computational overhead that can become a problem for large and complicated data sets. The protocol used here, therefore, is simplified but should be easily recognizable to users of other JUPITER-compatible programs.

The strategy for input is designed to use BLOCKS that are made up of either KEYWORDS for individual variables or TABLES for a series of data. The specification of whether a given block uses KEYWORDS or TABLES is predetermined and the input blocks defined below indicate which is required.

Blocks

Input blocks are allowed to take one of two forms: either KEYWORDS or TABLES. All input blocks are delineated by the words BEGIN and END. The header line also includes the name and type of the block and the final line contains the name of the block. For example,

```
BEGIN prior_mean_cv KEYWORDS
prior_betas=1
beta_cov_form = 0
END prior_mean_cv
```

Keywords

Keyword variables correspond to single values identified with an "=" sign. Multiple KEYWORDS can be entered on each line in an input file, but no spaces are allowed in KEYWORDS names or variable values. An example is: prior_betas=1.

Tables

Table variables are used for tabular data series that have multiple values in categories. Tables are identified by listing the number of rows (nrow), and number of columns (ncol), and by providing the keyword columnlabels. This is followed by nrow rows of data, with values arranged in ncol columns, corresponding to the same order as the columnlabels and delimited by one or more spaces. For example,

```
BEGIN Q_compression_cv TABLE
nrow=2 ncol=5 columnlabels
BetaAssoc Toep_flag Nrow Ncol Nlay
1 0 21 21 1
2 1 21 21 1
END Q_compression_cv
```

Files

A user may want to shorten the length of the main input file by reading certain input from external text files. This can be done by signaling an input block with the word FILES, to read a file containing the entire set

of information for the block. Regardless of whether the external text file contains a KEYWORDS or TABLE block, a block definition must be in place directing the program to the external file. For example,

```
BEGIN Q_compression_cv FILE
compression.txt
END Q_compression_cv
```

In this example, the contents of the file compression.txt would be

```
BEGIN Q_compression_cv TABLE
nrow=2 ncol=5 columnlabels
BetaAssoc Toep_flag Nrow Ncol Nlay
1 0 21 21 1
2 1 21 21 1
END Q_compression_cv
```

bgaPEST Input Blocks

The specific input blocks used in bgaPEST are discussed, in order of appearance in the <casename>.bgp file. It is important to maintain the order of the blocks in the same order as discussed in this report. For each block, data types are identified either as *float*, *integer*, or *string*. Values entered as *float* can include scientific/engineering notation, but in all cases should contain a "." even if no fractional detail is included. Conversely, *integers* must not contain ".". Variables identified as *string* may not include spaces because whitespace is used as the delimiter for rows in tables and separating keywords.

Each block is also defined with a suffix of "cv" for "control variables," or "data" for data. Control variables are those that govern the behavior of the algorithm as a whole as opposed to data points (such as parameter values, structural parameter values, and so forth).

A note on default variable values

In the input instructions below, some variables list a default value. Part of the design strategy of this software was to not burden users with determining appropriate values for each and every variable that controls the algorithm. As a result, default values are provided for some variables. In those cases, input by the user in the .bgp file is optional. If no value is provided by the user, the default value will be used by bgaPEST. If a variable not listed with a default value in these input instructions is omitted by a user, bgaPEST will return with an error indicating that the variable is not present.

Algorithmic Control Variables algorithmic_cv KEYWORDS

The following KEYWORDS variables are in the algorithmic_cv block.

structural_conv *float,default=0.001* Convergence criterion for structural parameter convergence. If positive, convergence is based on the absolute difference in structural parameter objective function over consecutive iterations. If negative, convergence is based on the norm of the difference between consecutive structural parameter values. Used only if at least one structural parameter is to be estimated.

phi_conv *float, default=0.001* Convergence criterion for objective function inner iterations.

bga_conv *float, default*=10×phi_conv Convergence criterion for objective function outer iterations.

it_max_structural *integer, default=10* Total number of iterations allowed in structural parameter optimization.

it_max_phi *integer, default=10* Total number of iterations allowed in each quasi-linear estimation optimization.

`it_max_bga` *integer, default=10* Total number of outer iterations allowed for the entire algorithm.

`linesearch` *integer, default=0* Flag to determine whether a line search should be conducted. [0] = do not use line search, [1] = use line search.

`it_max_linesearch` *integer, default=4* Total number of outer iterations allowed for the line search. Used only if `linesearch` = 1.

`theta_cov_form` *integer, default=0* Form of the theta covariance matrix. [0] = none, [1] = diagonal, [2] = full matrix. [0] means no prior covariance on theta provided and it is assumed to be unknown. Used only if at least one structural parameter is to be estimated.

`Q_compression_flag` *integer, default=0* Flag to determine how to calculate $\mathbf{Q}_{ss}$ [0] = no compression—calculate full $\mathbf{Q}_{ss}$ matrix, [1] = Calculate separate $\mathbf{Q}_{ss}$ matrix for each beta association.

`par_anisotropy` *integer, default=0* Flag to determine whether parameter anisotropy should be considered when making the $\mathbf{Q}_{ss}$ matrix. [0] = do not consider anisotropy, [1]=consider anisotropy. If anisotropy is considered, a `parameter_anisotropy` block should be included, as defined below.

`deriv_mode` *integer, default=0* Flag to determine whether sensitivities are calculated by using an external call to PEST or using a user-supplied program (such as adjoint state). [0] = use PEST, [1] = use external program identified below in the `model_command_lines` block, [4] = use external derivatives in parallel (see appendix 4 for details).

`posterior_cov_flag` *integer, default=0* Flag to determine whether posterior covariance matrix should be calculated. [0] = do not calculate posterior covariance matrix, [1] = calculate posterior covariance matrix. If `Q_compression_flag = 1`, only the diagonal of the posterior covariance matrix is calculated.

`jacobian_file` *string, default="scratch.jco"* Name of the file generated by an external program if `deriv_mode = 1`. If `deriv_mode = 0`, this value is ignored and left at its default value.

`jacobian_format` *string, default="binary"* Format of the file indicated in `jacobian_file`. [binary] indicates a binary file formatted as a JCO file from PEST, [ascii] indicates a file of a standard PEST matrix format, discussed below in this documentation. If `deriv_mode = 0`, this value is ignored and left at its default value.

Prior Mean Control Variables `prior_mean_cv` KEYWORDS

The following `KEYWORDS` variables are in the `prior_mean_cv` block.

`prior_betas` *integer* Flag indicating whether information about prior mean (β) will be supplied. [0] = no, [1] = yes.

`beta_cov_form` *integer, default=0* Form of the prior mean (β) covariance matrix $\left(\mathbf{Q}_{\beta\beta}\right)$. [0] = none, [1] = diagonal, [2] = full matrix. This value is used only if `prior_betas = 1`.

Beta Association Data `prior_mean_data` TABLE

This table must contain the same number of rows as there are beta associations to be defined. The rows must be in ascending order of beta association numbers. This is also the block where beta associations are defined, even if prior means are not defined. Parameter transformations are also defined in this table. Details about transformation options are in appendix 3.

`BetaAssoc` *integer* Identifier of each beta association (one per row). These should be sequential integers.

`Partrans` *string* Transformation indicator determining whether β values will be in physical or estimation space. Acceptable values are `log`, `power`, and `none`.

`alpha_trans` *float, default = 50.0* Exponent of the power transformation if `Partrans = power`.

`beta_0` *float* Value of prior mean (β_0) for the row's beta association. This value is used only if `prior_betas = 1`.

`beta_cov_#` *float* The number of values provided is based on the value of `beta_cov_form` specified above:

If `beta_cov_form = 1`, one value is provided.

If `beta_cov_form = 2`, `nrow` values are provided, corresponding to the current row of the beta covariance matrix $(\mathbf{Q}_{\beta\beta})$.
This value is used only if `prior_betas = 1`.

Structural Parameter Control Variables `structural_parameter_cv` TABLE

This table must contain the same number of rows as there are beta associations to be defined. The rows must be in ascending order of beta association numbers.

`BetaAssoc` *integer* Identified for each beta association (one per row). These should be sequential integers.

`prior_cov_mode` *integer, default = 1* Flag to indicate whether prior covariance of parameters ($\mathbf{Q}_{ss}$) is supplied or calculated. This is reserved for future use. Currently, $\mathbf{Q}_{ss}$ is always calculated, so this value is ignored if present.

`var_type` *integer, default=1* This is a flag to indicate which variogram type is used to express the prior covariance ($\mathbf{Q}_{ss}$). Acceptable choices are [0] = pure nugget, [1] = linear, [2] = exponential.

`struct_par_opt` *integer, default=1* Flag for whether structural parameters are meant to be optimized or not. This can be chosen for each structural parameter individually: [0] = do not optimize (hold at initial value), [1] = optimize by using a marginal distribution.

`trans_theta` *integer, default=0* Flag for whether a power transformation should be applied to the structural parameters in the current row. [0] = do not transform, [1] = transform. This value is used only if `struct_par_opt = 1`.

`alpha_trans` *float, default = 50* Exponent of the power transformation, used only if `trans_theta = 1`. Details of the power transformation are in appendix 3.

Structural Parameter Data `structural_parameter_data` TABLE

This table must contain the same number of rows as there are beta associations to be defined. The rows must be in ascending order of beta association numbers.

`BetaAssoc` *integer* Identifier of each beta association (one per row). These should be sequential integers.

`theta_0_1` *float* Initial value of $\theta_{1,0}$, which is the starting value of the first structural parameter for prior covariance.

`theta_0_2` *float* Initial value of $\theta_{2,0}$, which is the starting value of the second structural parameter for prior covariance. If a linear or nugget variogram is used, an arbitrary negative value should be entered here indicating that the value will be ignored. For an exponential variogram, this parameter is the correlation length.

Structural Parameter Covariance Data `structural_parameter_cov` TABLE

The only covariance model currently supported is diagonal, so there must be one covariance value for each θ parameter. This block is read only if `theta_cov_form` is not zero.

`theta_cov_1` *float* Variance of the current row's θ parameter. If an exponential variogram is used, then a single beta association will have two structural parameters. To handle this possibility, the variance values should be listed, one per line, in the order of beta associations, then in order θ_1 then θ_2. Even structural parameters that will not be estimated (for example, that are held at their initial values, as indicated by `struct_par_opt` above) must have a placeholder value entered here to maintain the order—the placeholder value is arbitrary and will be ignored.

Epistemic Error Term `epistemic_error_term` KEYWORDS

`sig_0` *float* Initial value of the epistemic uncertainty variance $\left(\sigma_{R_0}^2\right)$.

`sig_opt` *integer* Flag indicating whether epistemic uncertainty variance should be optimized for or not. [0] = do not optimize, [1] = optimize. If `sig_opt = 0`, then the value of `sig_0` is used throughout the inversion.

`sig_p_var` *float, default=0* Prior variance on $\sigma_{R_0}^2$. `sig_p_var = 0` means no prior variance on epistemic error is provided and it is assumed totally unknown. This value is used only if `sig_opt = 1`.

`trans_sig` *integer, default=0* Flag for whether a power transformation should be applied to the epistemic error. [0] = do not transform, [1] = transform. This value is used only if `sig_opt = 1`.

`alpha_trans` *float, default = 50* Exponent of the power transformation, used only if `trans_sig = 1`. Details of the power transformation are in appendix 3.

Parameter Control Variables `parameter_cv` KEYWORDS

`ndim` *integer* Number of dimensions over which the estimated parameters span.

Prior Covariance Compression Control Variables `Q_compression_cv` TABLE

This table must contain the same number of rows as there are beta associations to be defined. The rows must be in ascending order of beta association numbers. This block is read only if `Q_compression_flag` = 1.

`BetaAssoc` *integer* Identified of each beta association (one per row). These typically are sequential integers.

`Toep_flag` *integer* This is a flag to determine whether a Toeplitz transformation should be applied to the prior covariance matrix $(\mathbf{Q}_{\mathbf{ss}})$. [0] = do not use Toeplitz transformation, [1] = use Toeplitz transformation.

`Nrow` *integer* Number of rows in the current beta association (read only if `Toep_flag = 1`).

`Ncol` *integer* Number of columns in the current beta association (read only if `Toep_flag = 1`).

`Nlay` *integer* Number of layers in the current beta association (read only if `Toep_flag = 1`).

Parameter Groups `parameter_groups` TABLE

Each row of this table corresponds to one of the parameter groups. These groups are used to group together parameters by type and are not the same as beta associations.

`groupname` *string* Name of the group in the current row. Note that these cannot contain spaces.

`grouptype` *integer* Integer identifying which type of parameter the group corresponds to. This is used to ensure that beta associations do not span parameter types (for example, hydraulic conductivity parameters should not be in the same group type as recharge parameters). The specific values are arbitrary, but a distinct value should be assigned to each parameter group type.

`derinc` *float* The derivative increment for calculating external derivates if using external derivatives calculation (`deriv_mode = 0 or deriv_mode = 4`).

Parameter Data `parameter_data` TABLE

Each row of this table provides information for one parameter.

`ParamName` *string* Name for the parameter.

`StartValue` *float* Starting parameter value.

`GroupName` *string* Name of the group to which the parameter belongs. This name must be defined in the `parameter_groups` block.

`BetaAssoc` *integer* Beta association to which this parameter belongs.

`SenMethod` *integer* Sensitivity method used for this parameter type. This parameter may now be arbitrary—it is reserved for future use and currently is ignored.

`x1` *float* Location in the first dimension.

`x2` *float* Location in the second dimension. Read only if `ndim >= 2`.

`x3` *float* Location in the third dimension. Read only if `ndim = 3`.

Observation Groups `observation_groups` TABLE

Each row of this table corresponds to one of the observation groups. These groups are used to group together observations by type and are used to report portions of the objective function.

`groupname` *string* Name of the group in the current row. Note that these cannot contain spaces.

Observation Data `observation_data` TABLE

One observation is presented on each line.

`ObsName` *string* Name of an observation.

`ObsValue` *float* Value of the observation.

`GroupName` *string* Name of the group to which the observation belongs. This name must be defined in the `observation_groups` block.

`Weight` *float* A relative weight that gets applied to the epistemic error.

Model Command Lines `model_command_lines` KEYWORDS

Currently, a single forward model command and an optional derivative model command can be supplied here. These string keywords can include path information if the command line batch files or shell scripts are not located in the current working directory, but spaces are not allowed.

`Command` *string* This is the batch file or shell script that runs the forward model.

`DerivCommand` *string* This is the optional batch file or shell script that is used to calculate derivatives. This is used only if `deriv_method = 1` in the `algorithmic_cv` block.

Model Input Files `model_input_files` TABLE

Each row of this table includes a matched template file and model input file. This allows the program to create the correct input files for the model.

`TemplateFile` *string* Name of a template file for making model input. Must end in `.tpl`.

`ModInFile` *string* Name of the model input file corresponding to the TemplateFile identified on the same row.

Model Output Files `model_output_files` TABLE

Each row of this table includes a matched instruction file and model output file. This allows the program to read the results of model runs correctly.

`InstructionFile` *string* Name of an instruction file for reading model output. Must end in `.ins`.

`ModOutFile` *string* Name of the model output file corresponding to the InstructionFile identified on the same row.

Parameter Anisotropy `parameter_anisotropy` TABLE

Each row of this table contains information for parameter anisotropy for a beta association. This block is read only if the variable `par_anisotropy` =1 in the `algorithmic_cv` block.

`BetaAssoc` *integer* Identifier of a beta association.

`horiz_angle` *float* Angle, in degrees of the principal direction of anisotropy in a horizontal plane. See figure 3.1 for details.

`horiz_ratio` *float* Ratio of maximum to minimum principal property values in the horizontal plane. See figure 3.1 for details.

`vertical_ratio` *float* Ratio of maximum to minimum principal property values in the vertical direction. See figure 3.1 for details. This value is read only if `ndim=3`.

PEST Matrix Formats for Jacobian and Posterior Covariance

On two occasions in bgaPEST, a matrix text file format from PEST is used to store matrices: when posterior covariance output from bgaPEST is specified as a full matrix; and when Jacobian sensitivity matrix information is exchanged from an external code with bgaPEST run through, for example, a Python script.

```
   3   4   2
3.4423    23.323    2.3232   1.3232
5.4231    3.3124    4.4331   3.4442
7.4233    5.4432    7.5362   8.4232
* row names
apar1
apar2
apar3
* column names
aobs1
aobs2
aobs3
aobs4
```

Figure 1.1. Example of a standard PEST matrix file, adapted from Doherty (2010b).

The posterior covariance matrix may take two forms: a full matrix or a diagonal matrix. These options are discussed below. Two options are available for Jacobian sensitivity matrices (**H**) to be read by bgaPEST. If `deriv_mode=0`, PEST is used, external to bgaPEST, to calculate the Jacobian matrix resulting in a binary file with the extension `.jco`. If `deriv_mode=1` then an external program is used to calculate **H**, and a file, written by the external program, must be communicated to bgaPEST. This file can either be a `.jco` file or a `.jac` file which is an ASCII file following the format of a standard matrix file used by PEST, as described in Doherty (2010b), section 4.4.3. An example and description of the format of a standard PEST matrix follow, quoting from Doherty (2010b).

Figure 1.1 depicts an example matrix file holding a matrix with three rows and four columns.

The first line of a matrix file contains 3 integers. The first two indicate the number of rows (`NROW`) and number of columns (`NCOL`) in the following matrix. The next integer (named `ICODE`) is a code, the role of which will be discussed shortly. Following the header line is the matrix itself, in which entries are space-separated and wrapped to the next line if appropriate. The maximum line length is 500 characters, so wrapping to the next line must occur within 500 characters. It is recommended to wrap lines after 8 values and to maintain maximum possible precision.

In use for Jacobian matrices by bgaPEST, `ICODE` is set to 2, so the string "`* row names`" is printed next, followed by `NROW` names (of 20 characters or less in length), containing the names associated with rows of the matrix. `NCOL` column names follow in a similar format, following the string "`* column names`".

Other options for `ICODE` are described in Doherty (2010b) and are used in bgaPEST for output of the posterior covariance matrix. The two options for posterior covariance output both refer to square matrices that have the same names of columns and rows. As a result, only one list of names follows the data following the string "`* row and column names`".

If compression is used in the prior covariance matrix, bgaPEST outputs only the diagonal elements of the posterior covariance. In this case, `ICODE=-1` and only the diagonal entries are listed, one per line, after the header line. If compression is not used, the entire posterior covariance matrix is printed using `ICODE=1` with 8 values per line.

References Cited

Banta, E.R., Poeter, E.P., Doherty, J.E., and Hill, M.C., 2006, JUPITER: Joint Universal Parameter IdenTification and Evaluation of Reliability—An application programming interface (API) for model analysis: U.S. Geological Survey Techniques and Methods, book 6, chap. E1, 268 p.

Doherty, J., 2010b, PEST, Model-independent parameter estimation—Addendum to user manual (5th ed.): Brisbane, Australia, Watermark Numerical Computing.

Appendix 2—Quick Start Instructions

One advantage of using block input and keywords, as discussed in appendix 1, is that default values are supplied within bgaPEST so they can be skipped by a user. The values supplied as defaults have general applicability and will all be reported in the <casename>.bpr file. In this section, then, the bare minimum level of input is described to get a project running.

Forward Model

The forward model must exist and have the ability—either inherently or through pre- and post-processing—to receive input and provide output using text (ASCII) files. For bgaPEST to be able to run the model, template files (.TPL) and instruction files (.INS) must be provided, corresponding with model input and output, respectively. Details of the construction of these files are in Doherty (2010, chap. 3). The template and instruction files are detailed in the model_input_files and model_output_files blocks, respectively. The model_command_lines block also must be included with an entry for either a batch file or shell script in the command keyword that runs the model.

Observations

The observation_groups block must be completed. All observations can belong to the same group if desired. Groups are reported in output to assist in interpretation of results. The observation_data block also must be completed.

Beta Associations

Beta associations are first defined in the prior_mean_data block. If no prior information about mean values and their covariance is to be supplied, the only information necessary is a row for each beta association in the prior_mean_data block and a decision about whether to transform the value with a logarithmic or power transform. Note that beta associations indicate regions and groups that will have the same mean value estimated regardless of whether prior information about the mean is provided.

Structural Parameters

Each beta association must have a variogram specified for it, defined by structural parameters. Therefore, the structural_parameters_cv and structural_parameter_data blocks must be completed. Whether to optimize for structural parameter values and whether to provide prior information about the values are optional.

Parameters

The parameter_groups block must be completed and, like with observations, it is acceptable for all parameters to be in a single group, and groups do not need to correspond with beta associations. The parameter_cv keyword ndim must be provided, and the parameter_data block must be completed.

Algorithmic Control Variables

The algorithmic_cv block contains variables that all have default values; however, bgaPEST must find the algorithmic_cv block—even if it is empty. If the algorithmic_cv block is empty, all default values will be used.

Example .bgp Input File

The following text shows a bgaPEST input file. Two dependent files for parameters and observations are shown in abbreviated form to indicate their format.

Example1.bgp

```
BEGIN algorithmic_cv KEYWORDS
structural_conv=0.004 phi_conv=0.004
bga_conv=1.0e-2 it_max_structural=10
it_max_phi=15 it_max_bga=15
linesearch=1 it_max_linesearch=3
theta_cov_form=1 Q_compression_flag=1 deriv_mode = 1
jacobian_format = ascii jacobian_file = S1_1.jac
posterior_cov_flag = 1 par_anisotropy=0
END algorithmic_cv
BEGIN prior_mean_cv KEYWORDS
prior_betas= 1 beta_cov_form=1
END prior_mean_cv
BEGIN prior_mean_data TABLE
nrow=1 ncol=7 columnlabels
BetaAssoc Partrans beta_0 beta_cov_1 beta_cov_2 beta_cov_3 beta_cov_4
1 log -7.6 5e-7 65.  3.1 4.1
END prior_mean_data
BEGIN structural_parameter_cv TABLE
nrow=1 ncol=6 columnlabels
BetaAssoc prior_cov_mode var_type struct_par_opt trans_theta alpha_trans
1 2 1 1 1 20
END structural_parameter_cv
BEGIN structural_parameters_data TABLE
nrow=1 ncol=3 columnlabels
BetaAssoc theta_0_1 theta_0_2
1 1.0e-007 -0.1
END structural_parameters_data
BEGIN structural_parameters_cov TABLE
nrow=1 ncol=1 columnlabels
theta_cov_1
11.1
END structural_parameters_cov
BEGIN epistemic_error_term KEYWORDS
sig_0 = 1.000e-000 sig_opt = 0 sig_p_var=0.00001
END epistemic_error_term
BEGIN parameter_cv KEYWORDS
ndim=3
END parameter_cv
BEGIN Q_compression_cv TABLE
nrow=1 ncol=5 columnlabels
BetaAssoc Toep_flag Nrow Ncol Nlay
1 1 21 21 1
END Q_compression_cv
BEGIN parameter_groups TABLE
```

```
nrow=1 ncol=2 columnlabels
groupname grouptype
pargp_uno 1
END parameter_groups
BEGIN parameter_data FILES
PARAMETERS.txt
END parameter_data
BEGIN observation_groups TABLE
nrow=1 ncol=1 columnlabels
groupname obsgp_oden
END observation_groups
BEGIN observation_data FILES
obs12.txt
END observation_data
BEGIN model_command_lines KEYWORDS
Command = modflow.bat
DerivCommand = modflow_adj.bat
END model_command_lines
BEGIN model_input_files TABLE
nrow=1 ncol=2 columnlabels
TemplateFile ModInFile
S1_mul.tpl S1_.mul
END model_input_files
BEGIN model_output_files TABLE
nrow=1 ncol=2 columnlabels
InstructionFile ModOutFile
S1_1_hbs.ins S1_1.hbs
END model_output_files
BEGIN parameter_anisotropy TABLE
nrow = 1 ncol = 4 columnlabels
BetaAssoc horiz_angle horiz_ratio vertical_ratio
1 45 10 10
END parameter_anisotropy
```

PARAMETERS.txt

```
BEGIN parameter_data
TABLE nrow=441 ncol=8 columnlabels
ParamName StartValue GroupName BetaAssoc SenMethod x1 x2 x3
P1 2.000000000000000E-04 pargp_uno 1 1 1.0000E+00 1.000E+000.00E+00
P2 2.000000000000000E-04 pargp_uno 1 1 2.0000E+00 1.000E+000.00E+00
P3 2.000000000000000E-04 pargp_uno 1 1 3.0000E+00 1.000E+000.00E+00
...
P440 2.000000000000000E-04 pargp_uno 1 1 2.0000E+01 2.100E+010.00E+00
P441 2.000000000000000E-04 pargp_uno 1 1 2.1000E+01 2.100E+010.00E+00
END parameter_data
```

obs12.txt

```
BEGIN observation_data TABLE
nrow=12 ncol=4 columnlabels ObsName ObsValue GroupName Weight
P001T0000 33.8154 obsgp_oden 1.0
```

```
P002T0000 29.9383 obsgp_oden 1.0
P003T0000 28.4674 obsgp_oden 1.0
P004T0000 30.9286 obsgp_oden 1.0
P005T0000 24.7332 obsgp_oden 1.0
P006T0000 31.5769 obsgp_oden 1.0
P007T0000 27.3057 obsgp_oden 1.0
P008T0000 29.3834 obsgp_oden 1.0
P009T0000 27.8658 obsgp_oden 1.0
P010T0000 30.4177 obsgp_oden 1.0
P011T0000 28.5865 obsgp_oden 1.0
P012T0000 27.4403 obsgp_oden 1.0
END observation_data
```

Tables 2.1 and 2.2 summarize the input blocks and variables names, types, and default values.

Reference Cited

Doherty, J., 2010, PEST, Model-independent parameter estimation—User manual (5th ed., with slight additions): Brisbane, Australia, Watermark Numerical Computing.

Table 2.1. Summary of input blocks with variables identified.

algorithmic_cv	Algorithmic Control Variables: KEYWORDS	***This block is optional if all default values are used***	
Variable type	*Variable name*	*Default*	*Description*
double precision	structural_conv	0.001	Structural parameter convergence values
double precision	phi_conv	0.001	Objective function convergence value
double precision	bga_conv	0.001	BGA outer loop convergence value
integer	it_max_structural	10	Max number of iterations for struct parameters
integer	it_max_phi	10	Max number of iterations for objective function
integer	it_max_bga	10	Max number of iterations for BGA
integer	linesearch	0	Linesearch procedure flag: [0] not perform [1] perform
integer	it_max_linesearch	10	Max number of iterations for linesearch procedure
integer	theta_cov_form	0	Form of theta covariance: [0] none, [1] diag, [2] full matrix
integer	deriv_mode	0	derivatives (Jacobian) calculation method: [0] make PEST files internally, [1] use secondary command line argumern (typically adjoint state)
integer	posterior_cov_flag	0	[0] do not calculate posterior covariance, [1] calculate posterior covariance.
character, len = 6	jacobian_format	"binary"	Two options for how the Jacobian matrix calculated by an external code is communicated to bgaPEST: binary means a jco file, ascii means a text file
character(len=100)	jacobian_file	scratch.jco	Jacobian File
integer	par_anisotropy	0	Anisotropy flag: [0] no anistropy, [1] anisotropy
integer	Q_compression_flag	0	[0] none - calculate full Qss, [1] Calculate Qss for each beta separately and if nugget store just 1, if toep_flag store just a vector
prior_mean_cv	Prior Mean Control Variables: KEYWORDS		
Variable type	*Variable name*	*Default*	*Description*
integer	prior_betas	0	Have or not prior informations about mean? [0] No - [1] Yes
integer	beta_cov_form	0	Form of Beta covariance: [0] none, [1] diag, [2] full matrix
prior_mean_data	Beta Association Data: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
character(len=100)	Partrans	-	Vector of parameter transformation : [NONE], [POWER], or[LOG]. (Not case sensitive)
double precision	beta_0	-	Prior beta values
double precision	beta_cov_i i = 1, p	-	Covariance of beta
structural_parameter_cv	Structural Parameter Control Variables: TABLE		
Variable type	*Variable name*	*Default*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
integer	prior_cov_mode	1	Supplied matrix [0] or calculated [1].
integer	var_type	1	Type of variogram [0] pure nugget, [1] linear, [2] exponential
integer	struct_par_opt	1	Structural parameters optimization: [0] No optimization, [1] Optimization
integer	trans_theta	1	Transformation of structural parameters in the estimation space (power transform): [0] No, [1] Yes
double precision	alpha_trans	50	Exponent of power transformation in case of trans_theta
structural_parameter_data	Structural Parameter Data: TABLE		
Variable type	*Variable name*	*Default*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
double precision	theta_0_1	-	Initial value of theta 1 value
double precision	theta_0_2	-	Initial value of theta 2 value -- negative if not used
structural_parameter_cov	Structural Parameter Data: TABLE		
Variable type	*Variable name*	*Default*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
double precision	theta_cov_i i=1,…,max(num_thata_type)	-	Theta covariance matrix
epistemic_error_term	Epistemic Error Term: KEYWORDS		
Variable type	*Variable name*	*Defaults*	*Description*
double precision	sig_0	-	Initial value of sigma (epistemic uncertainty parameter)
integer	sig_opt	-	Optimization for sigma: [0] No, [1] Yes
double precision	sig_p_var	0	Prior variance on sigma
integer	trans_sig	0	Transformation of epistemic error in the estimation space (power transform): [0] No, [1] Yes
double precision	alpha_trans_sig	50	Exponent of power transformation in case of trans_sig

Table 2.2. Summary of input blocks with variables identified (continued).

parameter_cv	Parameter Control Variables: KEYWORDS		
Variable type	*Variable name*	*Defaults*	*Description*
integer	ndim	-	Spatial dimensions for parameters (1 if temporal only)
Q_compression_cv	Prior Covariance Compression Control Variables: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
integer	Toep_flag	-	Using Toeplitz matrix for Qss. [0] No, [1] Yes
integer	Nrow	-	Number of model rows
integer	Ncol	-	Number of model columns
integer	Nlay	-	Number of model layers
parameter_groups	Parameter Groups: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character (len=50)	groupname	-	Name of the parameter groups
integer	grouptype	-	Identifier to segregate groups of different types
double precision	derinc	-	Derivative increment for external Jacobian.
parameter_data	Parameter Data: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character (len=50)	GroupName	-	Name of group
double precision	StartValue	-	Starting values of parameters
character (len=50)	ParamName	-	Name of parameter
double precision	x1	-	Location in first dimension (time if a time series)
double precision	x2	-	Location in second dimension (read if ndim >= 2)
double precision	x3	-	Location in third dimension (read if ndim >= 3)
integer	SenMethod	-	Sensitivity calculation method
integer	BetaAssoc	-	Beta association
observation_groups	Observation Groups: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character (len=50)	groupname	-	Name of the observation groups
observation_data	Observation Data: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character (len=50)	GroupName	-	Name of groups
double precision	ObsValue	-	Vector of observations
character (len=50)	ObsName	-	names of observations
double precision	Weight	-	Weight for R matrix
model_command_lines	Model Command Lines: KEYWORDS		
Variable type	*Variable name*	*Defaults*	*Description*
character (len=50)	Command	-	Command line
character (len=50)	DerivCommand	-	derivative Command line
model_input_files	Model Input Files: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character(len=100)	TemplateFile	-	Template file
character(len=100)	ModInFile	-	Input file
model_output_files	Model Input Files: TABLE		
Variable type	*Variable name*	*Defaults*	*Description*
character(len=100)	InstructionFile	-	Instruction file
character(len=100)	ModOutFile	-	Output file
parameter_anisotropy	Parameter Anisotropy: TABLE	***This block is optional if parameter anisotropy is not used***	
Variable type	*Variable name*	*Defaults*	*Description*
integer	BetaAssoc	-	Integer identifiers of beta associations
double precision	horiz_angle	-	angle, in degrees, of principal anisotropy direction
double precision	horiz_ratio	-	Ratio of maximum to minimum principal property values in the horizontal plane
double precision	vertical_ratio	-	Ratio of maximum to minimum principal property values in the vertical direction (read only if ndim=3)

Appendix 3—Details of the Method

The Bayesian geostatistical approach is described in detail by Kitanidis and Vomvoris (1983), Hoeksema and Kitanidis (1984), Kitanidis (1995), and Nowak and Cirpka (2004) among others. The mathematics are reviewed here.

The Bayesian Geostatistical Approach

In the Bayesian geostatistical approach, the posterior pdf is calculated as

$$p(\mathbf{s}|\mathbf{y}) \propto \underbrace{\exp\left(-\frac{1}{2}(\mathbf{y}-\mathbf{h}(\mathbf{s}))^T \mathbf{R}^{-1}(\mathbf{y}-\mathbf{h}(\mathbf{s}))\right)}_{\mathrm{L}(\mathbf{y}|\mathbf{s})} \underbrace{\exp\left(-\frac{1}{2}(\mathbf{s}-\mathbf{X}\beta^*)^T \mathbf{G}_{\mathbf{ss}}^{-1}(\mathbf{s}-\mathbf{X}\beta^*)\right)}_{\mathrm{p}(\mathbf{s})} \tag{3.1}$$

where $\mathbf{s}$ is the $m \times 1$ vector of parameter values at distributed spatial locations in the model, $\mathbf{X}\beta^*$ is the prior mean, $\mathbf{G}_{\mathbf{ss}}$ is the $m \times m$ prior covariance of $(\mathbf{s}-\mathbf{X}\beta^*)$, $\mathbf{h}(\mathbf{s})$ is the $n \times 1$ vector of modeled forecasts colocated with observations $(\mathbf{y})$, and $\mathbf{R}$ is the $n \times n$ epistemic uncertainty covariance, modeled as $\sigma_R^2\mathbf{W}$ where σ_R^2 represents epistemic uncertainty, and $\mathbf{W}$ is an $n \times n$ diagonal matrix containing the observation weights. In general terms, the likelihood function $(L(\mathbf{y}|\mathbf{s}))$ characterizes the misfit between model forecasts and observations whereas the prior pdf $(p(\mathbf{s}))$ defines a characteristic (such as smoothness or continuity) that is assumed to apply to the parameter field. The prior pdf also serves the role of regularization. The best estimate of $\mathbf{s}$ maximizes the posterior pdf. A computationally efficient method to find the best estimates of $\mathbf{s}$ and β $\left(\hat{\mathbf{s}} \text{ and } \hat{\beta}, \text{ respectively}\right)$ is through

$$\hat{\mathbf{s}} = \mathbf{X}\hat{\beta} + \mathbf{Q}_{\mathbf{ss}}\mathbf{H}^{\mathbf{T}}\xi \tag{3.2}$$

which is the superposition of the prior mean (first term) and an innovation term that accounts for deviations of the model outputs from the observations (second term). In this context, β in the first term is not the prior mean, but is the best estimate of the mean (mapped onto the parameter field through the $m \times p$ $\mathbf{X}$ matrix) whereas the second term is fluctuations about the estimated mean. $\mathbf{H}$ in the second term (often referred to as the $n \times m$ Jacobian, sensitivity, or susceptibility matrix) is the sensitivity of observation values to parameter values where $H_{ij} = \frac{\partial \mathbf{h}(\mathbf{s})_i}{\partial \mathbf{s}_j}$, which can be calculated by using either finite difference or adjoint-state methods. In bgaPEST, finite-difference calculations for $\mathbf{H}$ are made by using PEST whereas adjoint-state calculations depend on the specific model being used and must be made by using an external program.

The values for $\hat{\beta}$ and ξ are found by solving the $(n+p) \times (n+p)$ linear system of cokriging equations

$$\begin{bmatrix} \mathbf{Q}_{\mathbf{yy}} & \mathbf{HX} \\ \mathbf{X}^T\mathbf{H}^T & -\mathbf{Q}_{\beta\beta}^{-1} \end{bmatrix} \begin{bmatrix} \xi \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ -\mathbf{Q}_{\beta\beta}^{-1}\beta^* \end{bmatrix} \tag{3.3}$$

where $\mathbf{Q}_{\mathbf{yy}}$ is the $n \times n$ auto-covariance matrix of the observations, defined as $\mathbf{HQ}_{\mathbf{ss}}\mathbf{H}^{\mathbf{T}} + \mathbf{R}$, n is the number of observations, and p is the number of beta associations. In hydrogeologic applications, the numerical forward model is typically nonlinear. Further nonlinearity can be induced by using a logarithmic or power transformation, which is a convenient way to enforce non-negativity on parameters.

Provided that the nonlinearities introduced are not too extreme, a solution can be obtained through successive linearizations following the quasi-linear extension (Kitanidis, 1995). The forward model, $\mathbf{h}(\mathbf{s})$ is expanded about the current best estimate of the parameters $\tilde{\mathbf{s}}$

$$\mathbf{h}(\mathbf{s}) \approx \mathbf{h}(\tilde{\mathbf{s}}) + \tilde{\mathbf{H}}(\mathbf{s}-\tilde{\mathbf{s}}) \tag{3.4}$$

where $\tilde{\mathbf{H}}$, as a function of $\tilde{\mathbf{s}}$, is evaluated at each linearization. We assign the subscript $_k$ to indicate iteration

number, and correct the measurements for the kth linearization as

$$\mathbf{y}_k^{'} = \mathbf{y} - \mathbf{h}(\tilde{\mathbf{s}}_k) + \tilde{\mathbf{H}}_k \tilde{\mathbf{s}}_k. \tag{3.5}$$

Then the cokriging equations (equation 3.3) are updated

$$\begin{bmatrix} \tilde{\mathbf{Q}}_{\mathbf{yy},k} & \tilde{\mathbf{H}}_k \mathbf{X} \\ \mathbf{X}^T \tilde{\mathbf{H}}_k^T & -\mathbf{Q}_{\beta\beta}^{-\mathbf{1}} \end{bmatrix} \begin{bmatrix} \xi_k \\ \hat{\beta}_k \end{bmatrix} = \begin{bmatrix} \mathbf{y}_k^{'} \\ -\mathbf{Q}_{\beta\beta}^{-\mathbf{1}} \beta^* \end{bmatrix} \tag{3.6}$$

where $\tilde{\mathbf{Q}}_{\mathbf{yy},k} = \tilde{\mathbf{H}}_k \mathbf{Q}_{\mathbf{ss}} \tilde{\mathbf{H}}_k^T + \mathbf{R}$. From this set of equations, the next estimate of $\mathbf{s}$ is

$$\tilde{\mathbf{s}}_{k+1} = \mathbf{X}\hat{\beta}_k + \mathbf{Q}_{\mathbf{ss}} \tilde{\mathbf{H}}_k^T \xi_k. \tag{3.7}$$

This process can be iterated until there is minimal difference in the parameter estimates or minimal further improvement in the objective function. The objective function, which we seek to minimize, is $-\ln p(\mathbf{s}|\mathbf{y})$ which is equivalent to maximizing equation 3.1

$$\Phi_T = -\frac{1}{2} \ln p(\mathbf{s}|\mathbf{y}) = (\mathbf{s} - \mathbf{X}\beta^*)^T \mathbf{G}_{\mathbf{ss}}^{-\mathbf{1}} (\mathbf{s} - \mathbf{X}\beta^*) - \frac{1}{2} (\mathbf{y} - \mathbf{h}(\mathbf{s}))^T \mathbf{R}^{-\mathbf{1}} (\mathbf{y} - \mathbf{h}(\mathbf{s})) \tag{3.8}$$

Line Search

In some cases, numerical instability makes convergence difficult. A line search is implemented in which a linear search is performed between the most recent best estimate of the parameters ($\hat{\mathbf{s}}$) and the current linearization of the parameters ($\widetilde{\mathbf{s}}$), seeking a parameter value that minimizes an objective function.

The line search optimizes a single parameter, ρ, along a linear dimension between $\hat{\mathbf{s}}$ and $\widetilde{\mathbf{s}}$ as

$$\mathbf{s}_{opt} = \hat{\mathbf{s}}\rho + \widetilde{\mathbf{s}}(1-\rho) \tag{3.9}$$

where $\mathbf{s}_{opt}$ minimizes the objective function, Φ_T, using a Nelder-Mead simplex (see, for example, Press and others, 1992), which guarantees monotonic decrease in Φ_T over successive iterations. It is recommended to limit the number of line-search iterations to a relatively low number, because the goal of handling weak linearity is balanced against the computations required to perform the line search. The greatest advantage is likely achieved in the first few (less than five) iterations. The role of the line search is not to find a minimum value of Φ_T because the nonlinearity of the overall problem prevents it. Rather, the line search is meant to be a correction of search direction for stability.

Parameter Field Anisotropy

In distributed parameter fields, such as hydraulic conductivity in groundwater models, it is common to encounter anisotropy along an axis that may or not be aligned with the coordinate (x, y, z) axes. bgaPEST allows the definition of anisotropy in a horizontal plane at any angle from the x-axis and also in the vertical direction. The general layout of horizontal anisotropy is illustrated in figure 3.1. The angle from the x-axis (specified in degrees) is defined by `horiz_angle`, and the amount of anisotropy is defined by `horiz_ratio`. The designation `p_max` refers to the direction with maximum parameter values and `p_min` refers to the direction of minimum parameter values. The ratio is used to adjust the effective distance (and thereby the covariance values) along that principal direction. The user supplies values for `horiz_angle` and `horiz_ratio` for each beta association. If some beta associations are not meant to exhibit anisotropy, the user may simply set `horiz_ratio=1.0`. If none of the beta associations exhibit anisotropy, the entire `parameter_anisotropy`

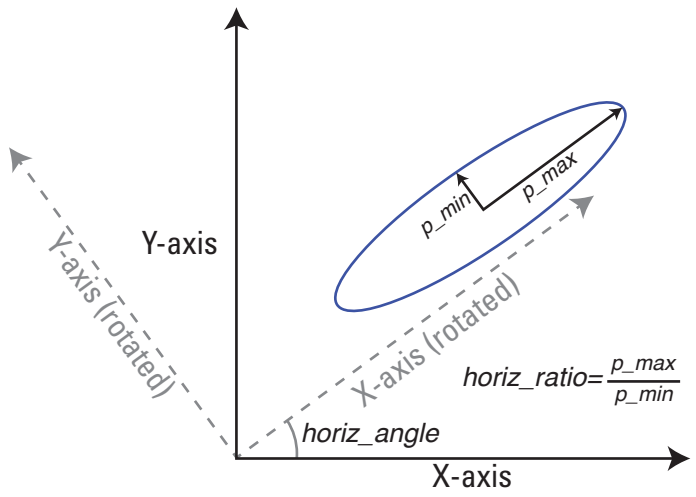

Figure 3.1. Schematic diagram of conventions defining horizontal anisotropy in bgaPEST .

block can be eliminated by setting the algorithmic control variable `par_anisotropy=0`, which means the block (if present) is ignored.

Anisotropy is introduced in the calculation of distances that are then used in the calculation of the prior covariance matrix $\mathbf{Q}_{ss}$ discussed below. Every pair of points must first be rotated into the principal direction orientation. This is accomplished by means of a rotation matrix:

$$\begin{bmatrix} x_{rot,i} \\ y_{rot,i} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{3.10}$$

where i indicates the ith point of the pair $(i = 1,2)$, θ is the angle (in degrees) specified by `horiz_angle`, x and y constitute the point coordinates in the original coordinate system, and x_{rot} and y_{rot} constitute the location projected into the coordinate system corresponding to the orientation of horizontal anisotropy. Once this projection is made, the horizontal distance is calculated as

$$distance = \sqrt{(x_{rot,1} - x_{rot,2})^2 + \texttt{horiz\_ratio} \times (y_{rot,1} - y_{rot,2})^2} \tag{3.11}$$

For three-dimensional parameter fields, a second anisotropy ratio may be specified as `vertical_ratio`. In the vertical direction, no angle is specified, so the rotation step is not required and distance is calculated as

$$distance = \sqrt{\begin{array}{c} (x_{rot,1} - x_{rot,2})^2 \\ +\texttt{horiz\_ratio} \times (y_{rot,1} - y_{rot,2})^2 \\ +\texttt{vertical\_ratio} \times (z_{rot,1} - z_{rot,2})^2 \end{array}} \tag{3.12}$$

Prior Probability Density Function

The prior pdf of $\mathbf{s}$ can be characterized as multi-Gaussian through its mean and covariance. The $(m \times 1)$ unknown parameter vector, $\mathbf{s}$, is modeled as a random process with mean

$$E[\mathbf{s}] = \mathbf{X}\beta \tag{3.13}$$

where $E[\cdot]$ indicates expected value, m is the number of parameters, β is a $(p \times 1)$ vector of drift coefficients, and $\mathbf{X}$ is an $(m \times p)$ matrix of base functions. In the absence of prior drift, the β constitute the beta association mean values, and $\mathbf{X}$ is a selection matrix mapping each value in $\mathbf{s}$ and β into their appropriate beta association. $\mathbf{X}$ contains all zeros except for $\mathbf{X}_{ij}^{th}$ element, which maps the ith parameter to the jth beta association, which contains the value of 1. Subdivision into beta associations within distributed parameter domains has been critical to success in hydrogeologic settings that include strong contrasts in parameter values indicative of geologic contacts (see, for example, Fienen and others, 2004 and Fienen and others, 2008). Prior drift is accounted for in $\mathbf{X}$ through trends expressed in the nonzero terms, although this is not currently implemented in bgaPEST. The prior covariance ($\mathbf{Q_{ss}}$) of $\mathbf{s}$ for a known β is

$$\mathbf{Q_{ss}}(\theta) = E[(\mathbf{s}-\mathbf{X}\beta)(\mathbf{s}-\mathbf{X}\beta)^T] \tag{3.14}$$

where $\mathbf{Q_{ss}}$ is a covariance function with structural parameters θ. In bgaPEST, allowable covariance functions include

1. the nugget

$$\mathbf{Q}_{ss}(\mathbf{d}) = \sigma^2 \tag{3.15}$$

and

2. the exponential covariance function

$$\mathbf{R}_{ss}(\mathbf{d}) = \sigma^2 \exp(-\frac{|\mathbf{d}|}{\ell}) \tag{3.16}$$

where $|\mathbf{d}|$ is separation distance, σ^2 is variance, and ℓ is integral scale. If the integral scale is set such that $\ell > \max(|\mathbf{d}|)$ we can substitute $\sigma^2 = \theta\ell$ and restate equation 3.16 as

$$\mathbf{Q_{ss}}(\mathbf{h}, \sigma^2) = \sigma^2 \ell \exp(-\frac{|\mathbf{d}|}{\ell}). \tag{3.17}$$

We can also set $\ell = 10 \times \max(|\mathbf{d}|)$ so that the behavior of the covariance function will be as a linear variogram (Fienen and others, 2008) which enforces continuity at a scale determined by the single free structural parameter σ^2. The motivation for this covariance function choice is to impart minimal assumptions about parameter structure onto the solution.

The appropriate values of θ (the vector of structural parameters including σ_R^2 and σ^2)are calculated through restricted maximum likelihood. For the remainder of this derivation, θ is assumed to be known. In bgaPEST, as discussed below in the input instructions, either the exponential or linear variogram models may be used.

Assembling the mean and covariance, the prior pdf is

$$p(\mathbf{s}|\beta) \propto \exp\left[-\frac{1}{2}(\mathbf{s}-\mathbf{X}\beta)^{\mathrm{T}}\mathbf{Q_{ss}^{-1}}(\mathbf{s}-\mathbf{X}\beta)\right]. \tag{3.18}$$

In the case of no knowledge about the prior mean, the prior pdf of β can be modeled as uniform over all space as $p(\beta) \propto \mathbf{1}$ with both $\mathbf{s}$ and β being estimated together, so that the conditional distribution in equation 3.18 is replaced by a joint distribution

$$p(\mathbf{s}, \beta) \propto \exp\left[-\frac{1}{2}(\mathbf{s}-\mathbf{X}\beta)^{\mathrm{T}}\mathbf{Q_{ss}^{-1}}(\mathbf{s}-\mathbf{X}\beta)\right]. \tag{3.19}$$

Frequently, at least diffuse knowledge about the prior mean is available and can be modeled as multi-Gaussian

with mean β^* and covariance $\mathbf{Q}_{\beta\beta}$. Typically, $\mathbf{Q}_{\beta\beta}$ is modeled as a diagonal matrix with variance values on the diagonal indicating independence among the β^*. Incorporating the prior information yields a prior pdf for $\mathbf{s}$

$$p(\mathbf{s}) \propto \exp\left[-\frac{1}{2}(\mathbf{s}-\mathbf{X}\beta^*)^{\mathrm{T}}\mathbf{G}_{\mathbf{ss}}^{-\mathbf{1}}(\mathbf{s}-\mathbf{X}\beta^*)\right] \tag{3.20}$$

where $\mathbf{X}\beta^*$ is the prior mean, and $\mathbf{G}_{\mathbf{ss}} = \mathbf{Q}_{\mathbf{ss}} + \mathbf{X}\mathbf{Q}_{\beta\beta}\mathbf{X}^{\mathbf{T}}$ is the prior covariance (Nowak and Cirpka, 2004). The incorporation of prior mean information, even assuming very high variance values in $\mathbf{Q}_{\beta\beta}$, can provide numerical stability without overly biasing the results. In the original formulations of Kitanidis and Vomvoris (1983); Hoeksema and Kitanidis (1984); Kitanidis (1995), $\mathbf{G}_{\mathbf{ss}} = \mathbf{Q}_{\mathbf{ss}}$ and no prior covariance is supplied on the values for β. This behavior can be duplicated in bgaPEST by specifying `prior_betas=0` in the input block for Prior Mean Control Variables.

Prior Covariance Matrix Storage Issues

In underdetermined problems suitable for bgaPEST, the number of parameters can be very large. The prior covariance matrix discussed above can, therefore, grow to such large dimensions that it cannot be practically stored in computer memory. However, two techniques are provided to alleviate some of this storage stress: compression and Toeplitz transformation.

Compression takes advantage of the fact that values in the $\mathbf{G}_{\mathbf{ss}}$ matrix relating parameters in different beta associations, by definition, have the value of zero. As a result, a general $\mathbf{G}_{\mathbf{ss}}$ matrix can be viewed as a partitioned matrix of nonzero blocks $\left(\mathbf{G}_{\mathbf{ss},\beta i}\right)$ and zero blocks

$$\mathbf{G}_{\mathbf{ss}} = \begin{bmatrix} \mathbf{G}_{\mathbf{ss},\beta 1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{\mathbf{ss},\beta 2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{G}_{\mathbf{ss},\beta p} \end{bmatrix}. \tag{3.21}$$

There is no need to store the zero elements provided that accommodations are made to avoid multiplications that involve the zeros. These accommodations have been made in bgaPEST and compression is, therefore, allowed.

In cases where spacing of model cells or nodes is constant in respective directions, $\mathbf{G}_{\mathbf{ss},\beta i}$ is a block Toeplitz matrix (Gray, 2005). A square, symmetric $j \times j$ matrix is Toeplitz in form if it has diagonals that all have the same value as in this example

$$\mathbf{T} = \begin{bmatrix} t_0 & t_1 & \dots & t_{j-2} & t_{j-1} \\ t_1 & t_0 & t_1 & \ddots & t_{j-2} \\ \vdots & t_1 & t_0 & \ddots & \vdots \\ t_{j-2} & \ddots & \ddots & \ddots & t_1 \\ t_{j-1} & t_{j-2} & \dots & t_1 & t_0 \end{bmatrix}. \tag{3.22}$$

This matrix has the properties that there are only j unique values, and these values occur in a regular order such that only a vector of length j needs to be stored from which individual rows can be constructed to perform matrix multiplication operations. The spacing, as indicated above, must be constant. For example, in a spatial model such as properties in a groundwater model, Δx, Δy, and Δz must be constant, but these values do not need to be equal to each other. This condition is restrictive in the sense that it implies a regular grid that may not correspond to geometry in the field; however, the regular grid required to take advantage of Toeplitz storage

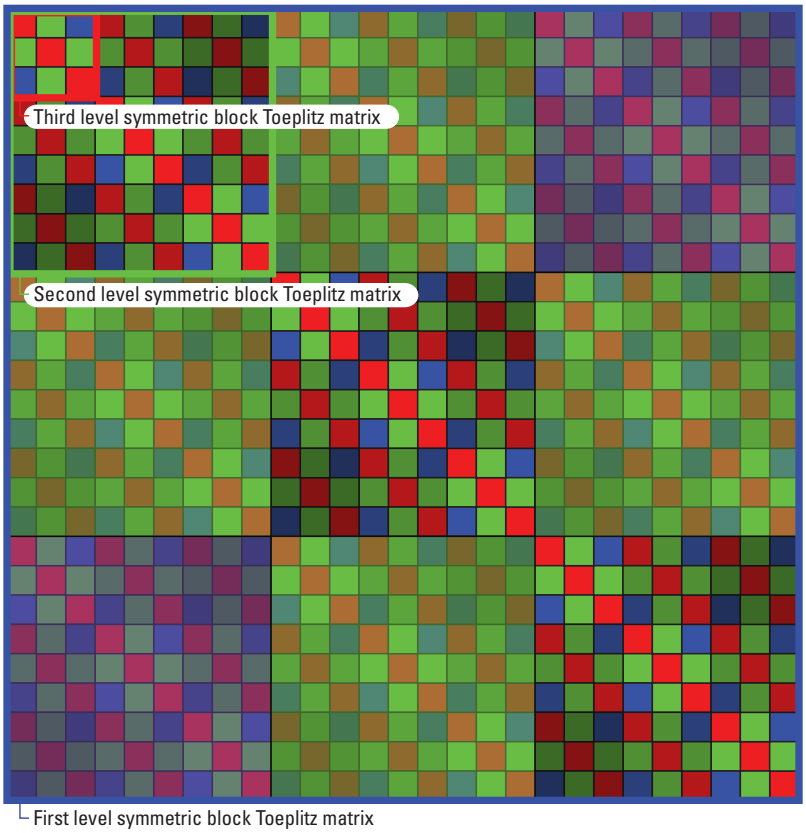

Figure 3.2. Schematic graphical visualization of a three-level embedded set of Toeplitz blocks in a covariance matrix (modified from D'Oria (2010)). The smallest squares represent individual matrix entries (values), and the colors correspond to distinct values. In this synthetic example, it is assumed that there are three rows, three columns, and three layers in the underlying model.

and operations can be assigned to one beta association with a surrounding, irregular grid put in another beta association with fewer parameter values.

In order to use Toeplitz structure in three dimensions, there must be a three-level embedding of Toeplitz matrices (D'Oria, 2010). The first level corresponds to the model layers, the second level corresponds to model rows, and the third level corresponds to model columns. Inspection of the schematic example in figure 3.2 shows that every distinct value represented in the entire matrix is found in the first (leftmost) column. Cycling of rows or columns, relative to the single stored vector, can be used to reconstruct any row or column of the main matrix to be used in multiplication operations. In bgaPEST, a combination of Toeplitz and complete blocks can make up the $\mathbf{G}_{ss}$ matrix in Compressed form, as discussed above.

For even larger problems (more parameters) specialized Fourier transform-based functions may be useful to speed up the computations made with compressed matrices (Nowak and others, 2003; Nowak and Cirpka, 2004). The restriction of regular grid spacing can also be relaxed by using a Karhunen-Loève transform (Li and Cirpka, 2006). Both of these advances may be considered as future improvements to bgaPEST but are not currently implemented.

Detailed input instructions for bgaPEST are presented in appendix 1. It is important to note, however, that if Toeplitz compression is invoked, parameters must be listed in the `.bgp` input file in order, sorted first by layer, then by column, and finally by row.

Likelihood Function

The parameters **s** are related to observations **y** through a measurement equation

$$\mathbf{y} = \mathbf{h}(\mathbf{s}) + \mathbf{v} \tag{3.23}$$

where **y** is an $(n \times 1)$ vector of observations, such as hydraulic heads or solute concentrations, $\mathbf{h}(\mathbf{s})$ is a transfer function or numerical model that calculates predictions which are colocated spatially and temporally with the observation values, and **v** is an $(n \times 1)$ vector of epistemic uncertainty terms, modeled as a random process with zero mean and covariance matrix **R**. Epistemic uncertainty is the result of imperfect or sparse measurements and an incomplete or inappropriate conceptual model (Rubin, 2003, p. 4). The epistemic uncertainty terms are assumed to be independent and uncorrelated so

$$\mathbf{R} = \sigma_{\mathbf{R}}^{\mathbf{2}}\mathbf{W} \tag{3.24}$$

where σ_R^2 is the epistemic uncertainty parameter and **W** is an $(n \times n)$ diagonal weight matrix in which each element is $\mathbf{W}_{ii} = \frac{1}{\omega_i^2}$ where ω_i is the ith weight, specified by the user. The purpose of the values of ω is to allow for different confidence in different individual observations or groups of observations. In reality, the component of epistemic uncertainty due to measurement error is likely uncorrelated, but the component due to modeling and conceptual uncertainty is likely systematic and correlated (Gaganis and Smith, 2001). A significant portion of this uncertainty may be reduced by not lumping parameters into homogeneous zones (Gallagher and Doherty, 2007), and the means to characterize the structure of **R** are rarely available. If information about **R** is available, however, it could be included and equation 3.24 replaced by a more complicated matrix. This option is currently not available in bgaPEST, however. Proceeding with equation 3.24, the likelihood function, assumed to be multi-Gaussian, is

$$L(\mathbf{y}|\mathbf{s}) \propto \exp\left[-\frac{1}{2}\left(\mathbf{y}-\mathbf{h}(\mathbf{s})\right)^{\mathrm{T}}\mathbf{R}^{-\mathbf{1}}\left(\mathbf{y}-\mathbf{h}(\mathbf{s})\right)\right]. \tag{3.25}$$

The structural parameter for the likelihood function is σ_R^2 and is calculated along with θ by using restricted maximum likelihood.

Posterior Probability Density Function

Applying Bayes' theorem with the product of equations 3.20 and 3.25 yields the posterior pdf

$$p(\mathbf{s}|\mathbf{y}) \propto \exp\left[-\frac{1}{2}(\mathbf{s}-\mathbf{X}\beta^*)^{\mathrm{T}}\mathbf{G}_{\mathbf{ss}}^{-\mathbf{1}}(\mathbf{s}-\mathbf{X}\beta^*)-\frac{1}{2}\left(\mathbf{y}-\mathbf{h}(\mathbf{s})\right)^{\mathrm{T}}\mathbf{R}^{-\mathbf{1}}\left(\mathbf{y}-\mathbf{h}(\mathbf{s})\right)\right] \tag{3.26}$$

The best estimate of **s** maximizes the posterior pdf. A computationally efficient method to find the best estimates of **s** and β $\left(\hat{\mathbf{s}} \text{ and } \hat{\beta}, \text{ respectively}\right)$ is through

$$\hat{\mathbf{s}} = \mathbf{X}\hat{\beta} + \mathbf{Q}_{\mathbf{ss}}\mathbf{H}^{\mathbf{T}}\xi \tag{3.27}$$

which is the superposition of the prior mean (first term) and an innovation term that accounts for deviations of the model outputs from the observations (second term). **H** in the second term (often referred to as the Jacobian, sensitivity, or susceptibility matrix) is the sensitivity of observation values to parameter values where $H_{ij} = \frac{\partial \mathbf{h}(\mathbf{s})_i}{\partial \mathbf{s}_j}$ is calculated by using either finite-difference or adjoint-state methods. The values for $\hat{\beta}$ and

ξ are found by solving the $(n+p)\times(n+p)$ linear system of cokriging equations

$$\begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ \mathbf{X}^T\mathbf{H}^T & -\mathbf{Q}_{\beta\beta}^{-1} \end{bmatrix} \begin{bmatrix} \xi \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ -\mathbf{Q}_{\beta\beta}^{-1}\beta^* \end{bmatrix} \tag{3.28}$$

where $\mathbf{Q_{yy}}$ is the auto-covariance matrix of the observations, defined as $\mathbf{HQ_{ss}H^T}+\mathbf{R}$.

Quasi-Linear Extension

As discussed by Kitanidis (1995), we must adjust calculations of the posterior pdf to account for nonlinearity. To do this, we expand the solution in a first-order Taylor expansion, resulting in an updated set of cokriging equations from equation 3.28

$$\begin{bmatrix} \mathbf{Q_{yy}} & \mathbf{HX} \\ \mathbf{X}^T\mathbf{H}^T & -\mathbf{Q}_{\beta\beta}^{-1} \end{bmatrix} \begin{bmatrix} \xi \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{y}-\mathbf{h}(\tilde{\mathbf{s}})+\mathbf{H}\tilde{\mathbf{s}} \\ -\mathbf{Q}_{\beta\beta}^{-1}\beta^* \end{bmatrix} \tag{3.29}$$

At each iteration (later referred to as inner iterations), the system in equation 3.29 is solved, resulting in an updated estimate of $\hat{\mathbf{s}}$ calculated through equation 3.27. At each iteration, the objective function, based on minimizing the negative logarithm of the posterior pdf (equation 3.26) is evaluated by using the current value of $\hat{\mathbf{s}}$: this is equivalent to finding the values of $\mathbf{s}$ that *maximize* the posterior probability. Switching to a minimization problem and taking the logarithm has computational advantages.

The objective function, then, is

$$\Phi_T = \Phi_M + \Phi_R \tag{3.30}$$

where Φ_T is the total objective function, Φ_M is the misfit objective function (also corresponding to the likelihood function) and Φ_R is the regularization objective function (also corresponding to the prior pdf). The components of equation 3.30 are

$$\Phi_M = \frac{1}{2}(\mathbf{y}-\mathbf{h}(\hat{\mathbf{s}}))^T\mathbf{R}^{-1}(\mathbf{y}-\mathbf{h}(\hat{\mathbf{s}})) \tag{3.31}$$

and

$$\Phi_R = \frac{1}{2}(\hat{\mathbf{s}}-\mathbf{X}\beta^*)^T\mathbf{G}_{\mathbf{ss}}^{-1}(\hat{\mathbf{s}}-\mathbf{X}\beta^*) \tag{3.32}$$

where both the negative signs and exponentiation are obviated by taking the negative logarithm of $p(\mathbf{s}|\mathbf{y})$.

Implementation of Partitions into Beta Associations

The concept of beta associations is discussed above and details of their implementation are given here. First, the prior covariance matrix $\mathbf{Q_{ss}}$ is censored by assigning a value of zero to each element that characterizes covariance between cells of different regions or parameter types, as defined by beta associations. It is not required that the covariance model be the same for each beta association. If different covariance models are used for different zones, this is reflected in the appropriate parts of $\mathbf{Q_{ss}}$. Furthermore, in some applications, a single structural parameter, θ, may be estimated and applied to all of $\mathbf{Q_{ss}}$. In other cases, and necessarily if the covariance model differs in various beta association, multiple elements of θ are estimated.

A distinct prior mean parameter β^* is assigned for each beta association, and the matrix $\mathbf{X}$ (equation 3.13) is determined as explained above. In cases where the mean of each zone is completely unknown, no values for

β^* are provided, but the $\mathbf{X}$ matrix is constructed nonetheless and in both cases a value of $\hat{\beta}$ is calculated for each beta association.

Structural Parameters and Restricted Maximum Likelihood

A vital element to the method outlined above is proper selection of the structural parameters. Structural parameters—also called hyperparameters—or nuisance parameters, are the parameters that characterize the covariance structure of both the epistemic uncertainty related to the observations, and the inherent variability of the parameters. In bgaPEST, structural parameters may include the epistemic uncertainty term in equation 3.24 $\left(\sigma_R^2\right)$ and the prior pdf variogram parameters in equation 3.17 (θ). These parameters are estimated by using restricted maximum likelihood consistent with the approaches of Kitanidis and Vomvoris (1983), Kitanidis (1995) and Li and others (2007).

Applying Bayes' theorem to the structural parameters, given the measurements, we calculate

$$p\left(\theta|\mathbf{y}_\mathbf{k}^{'}\right) \propto L\left(\mathbf{y}_k^{'}|\theta\right) p\left(\theta\right) \tag{3.33}$$

The likelihood function evaluates how closely the observations and predictions match, given the current linearization and the current set of structural parameters

$$L\left(\mathbf{y}_k^{'}|\theta\right) \propto \det\left(\mathbf{G}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right)^T \mathbf{G}_{\mathbf{yy}}^{-\mathbf{1}}\left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right)\right] \tag{3.34}$$

where $\mathbf{G}_{\mathbf{yy}}$ is the measurement autocovariance defined as

$$\mathbf{G}_{\mathbf{yy}} = \mathbf{Q}_{\mathbf{yy}} + \mathbf{HXQ}_{\beta\beta}\mathbf{X}^T\mathbf{H}^T. \tag{3.35}$$

Note that $\mathbf{Q}_{\mathbf{yy}}$ is intrinsically dependent upon the values of θ.

Prior information about the structural parameters may also be included, with prior mean θ^* and covariance matrix $\mathbf{Q}_{\theta\theta}$:

$$p\left(\theta\right) \propto \det\left(\mathbf{Q}_{\theta\theta}\right)^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(\theta-\theta^*\right)^T \mathbf{Q}_{\theta\theta}^{-1}\left(\theta-\theta^*\right)\right] \tag{3.36}$$

The posterior pdf is the product of equations 3.36 and 3.34

$$p\left(\theta|\mathbf{y}_\mathbf{k}^{'}\right) \propto \det\left(\mathbf{Q}_{\theta\theta}\right)^{-\frac{1}{2}} \det\left(\mathbf{G}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \exp\left[\begin{array}{c} -\frac{1}{2}\left(\theta-\theta^*\right)^T \mathbf{Q}_{\theta\theta}^{-1}\left(\theta-\theta^*\right) \\ -\frac{1}{2}\left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right)^T \mathbf{G}_{\mathbf{yy}}^{-\mathbf{1}}\left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right) \end{array}\right]. \tag{3.37}$$

To find the most likely values for θ we minimize $-\ln\left(p\left(\theta|\mathbf{y}_\mathbf{k}^{'}\right)\right)$ resulting in the objective function

$$\Phi_S = \frac{1}{2}\ln\left(\det\left(\mathbf{G}_{\mathbf{yy}}\right)\right) + \frac{1}{2}\left[\left(\theta-\theta^*\right)^T \mathbf{Q}_{\theta\theta}^{-1}\left(\theta-\theta^*\right) + \left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right)^T \mathbf{G}_{\mathbf{yy}}^{-\mathbf{1}}\left(\mathbf{y}_k^{'}-\mathbf{HX}\beta^*\right)\right] \tag{3.38}$$

where unchanging quantities are absorbed into the constant of proportionality including $\det\left(\mathbf{Q}_{\theta\theta}\right)^{-\frac{1}{2}}$. The optimal values for θ are found byusing the Nelder-Mead simplex algorithm (for example, Press and others, 1992, p. 408-410). Non-negativity in the θ parameters can be enforced by using a power transformation (Box and Cox, 1964) discussed below. As indicated by Kitanidis (1995), nonlinearity requires that structural parameters to be estimated iteratively with the estimation of model parameters. This is accomplished through a

sequence of coupled inversion as follows.

1. Initialize model parameters as ($\mathbf{s_0}$) and structural parameters ($\theta_\mathbf{0}$).
2. Solve for a new estimate of model parameters ($\hat{\mathbf{s}}$) holding θ constant.
3. Solve for a new estimate of structural parameters $\left(\hat{\theta}\right)$ holding $\mathbf{s}$ constant.
4. Repeat steps 2 and 3 until the change in θ in two consecutive outer iterations of steps 2 and 3 decreases below a specified tolerance.

Logarithmic and Power Transformations

In some cases, structural parameters and model parameters are best estimated in transformed space. A common reason is to enforce non-negativity. For model parameters either a logarithmic (base `e`) or a power transformation may be used. For structural parameters a power transformation is the only option.

The power transformation (Box and Cox, 1964; Fienen and others, 2004) is defined as:

$$\mathbf{s} = \alpha\left(\mathbf{p}^{\frac{1}{\alpha}} - 1\right) \tag{3.39}$$

where $\mathbf{s}$ is the vector of transformed parameters, $\mathbf{p}$ is the vector of non-transformed parameters, and α is a tuning variable that controls the strength of the transformation. The back-transformation is:

$$\mathbf{p} = \left(\frac{\mathbf{s}+\alpha}{\alpha}\right)^{\alpha} \tag{3.40}$$

At the limit, as α increases to infinity, the transformation and back-transformation converge on the natural logarithm and exponential function, respectively:

$$\mathbf{s} = \lim_{\alpha\to\infty} = \alpha\left(\mathbf{p}^{\frac{1}{\alpha}} - 1\right) = \ln(\mathbf{p}) \tag{3.41}$$

and

$$\mathbf{p} = \lim_{\alpha\to\infty} = \left(\frac{\mathbf{s}+\alpha}{\alpha}\right)^{\alpha} = \exp(\mathbf{s}). \tag{3.42}$$

Posterior Covariance

Calculation of the posterior covariance can be based on the inverse of the Hessian of the objective function (for example, Nowak and Cirpka, 2004). In closed form, the equation for the full posterior covariance matrix is:

$$\mathbf{V} = \mathbf{G_{ss}} - \mathbf{G_{sy}}\mathbf{G_{yy}}^{-1}\mathbf{G_{sy}}^{T} \tag{3.43}$$

where $\mathbf{G_{sy}} = \mathbf{G_{ss}}\mathbf{H}^T$ and $\mathbf{G_{yy}} = \mathbf{H}\mathbf{G_{ss}}\mathbf{H}^T + \mathbf{R}$. In the case where compression of $\mathbf{Q}_{ss}$ is not used, the full matrix $\mathbf{V}$ is calculated and reported. Where compression of $\mathbf{Q}_{ss}$ is used, however, the diagonal of $\mathbf{V}$ is returned as a vector of variances on parameters. This information is reported in a separate file but is also used to calculated posterior 95 percent confidence intervals. The full matrix, when reported, can be used to calculate conditional realizations (Kitanidis, 1995, 1996).

References Cited

Box, G., and Cox, D. R., 1964, An analysis of transformations: Journal of the Royal Statistical Society, series B (Methodolodical), v. 26, no. 2, p. 211–252.

D'Oria, M., 2010, Characterization of aquifer hydraulic parameters—From Theis to hydraulic tomography: Università degli Studi di Parma, Ph.D. dissertation, 153 p.

Fienen, M., Kitanidis, P., Watson, D., and Jardine, P., 2004, An application of Bayesian inverse methods to vertical deconvolution of hydraulic conductivity in a heterogeneous aquifer at Oak Ridge National Laboratory: Mathematical Geology, v. 36, no. 1, p. 101–126, doi:10.1023/B:MATG.0000016232.71993.bd.

Fienen, M.N., Clemo, T.M., and Kitanidis, P.K., 2008, An interactive Bayesian geostatistical inverse protocol for hydraulic tomography: Water Resources Research, v. 44, W00B01, doi:10.1029/2007WR006730.

Gaganis, P., and Smith, L., 2001, A Bayesian approach to the quantification of the effect of model error on the predictions of groundwater models: Water Resources Research, v. 37, no. 9, p. 2309–2322, doi:10.1029/2000WR000001.

Gallagher, M.R., and Doherty, J., 2007, Parameter interdependence and uncertainty induced by lumping in a hydrologic model: Water Resources Research, v. 43, no. 5, W05421, doi:10.1029/2006wr005347.

Gray, R., 2005, Toeplitz and circulant matrices—A review: Delft, The Netherlands, Now Publishers, 90 p.

Hoeksema, R.J., and Kitanidis, P.K., 1984, An application of the geostatistical approach to the inverse problem in two-dimensional groundwater modeling: Water Resources Research, v. 20, no. 7, p. 1003–1020, doi:10.1029/WR020i007p01003.

Kitanidis, P.K., 1995, Quasi-linear geostatistical theory for inversing: Water Resources Research, v. 31, no. 10, p. 2411–2419, doi:10.1029/95WR01945.

Kitanidis, P.K., 1996, Analytical expressions of conditional mean, covariance, and sample functions in geostatistics: Stochastic Hydrology and Hydraulics, v. 10, no. 4, p. 279–294, doi:10.1007/bf01581870.

Kitanidis, P.K., and Vomvoris, E.G., 1983, A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations: Water Resources Research, v. 19, no. 3, p. 677–690, doi:10.1029/WR019i003p00677.

Li, W., and Cirpka, O.A., 2006, Efficient geostatistical inverse methods for structured and unstructured grids: Water Resources Research, v. 42, no. 6, W06402, doi:10.1029/2005WR004668.

Li, W., Englert, A., Cirpka, O.A., Vanderborght, J., and Vereecken, H., 2007, Two-dimensional characterization of hydraulic heterogeneity by multiple pumping tests: Water Resources Research, v. 43, no. 4, W04433, doi:10.1029/2006WR005333.

Nowak, W., and Cirpka, O.A., 2004, A modified Levenberg-Marquardt algorithm for quasi-linear geostatistical inversing: Advances in Water Resources, v. 27, no. 7, p. 737–750, doi:10.1016/j.advwatres.2004.03.004.

Nowak, W., Tenkleve, S., and Cirpka, O.A., 2003, Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities: Mathematical Geology, v. 35, no. 1, p. 53–66.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.O., 1992, Numerical recipes in C—The art of scientific computing (2d ed.): Cambridge, UK; New York; Cambridge University Press, 994 p.

Rubin, Y., 2003, Applied stochastic hydrogeology: Oxford, UK; New York; Oxford University Press, 391 p.

Appendix 4—Parallel Implementation of Jacobian Calculations

Introduction

A substantial computational burden in quasi-linear parameter estimation is the repeated calculation of the Jacobian matrix (**H**). Calculation of this matrix is "embarassingly" or "pleasingly" parallel—the number of runs required is the same as the number of parameters, but the individual model runs do not need to interact in any way. Given this property of Jacobian matrix calculation, a cluster of individual computers can be used to make the computations in parallel. Parallel implementation can be performed using a problem-specific implementation coded into the software (for example, Schreüder, 2009), compiling the software against parallelization libraries (for example, Muffels and others, 2012), or using a parallel management system outside the software (for example, Condor Team, 2012).

As an initial step toward full support of parallel Jacobian calculations, this appendix documents an implementation using the Condor package (Condor Team, 2012) and Python scripts. This implementation currently calculates the Jacobian matrix using forward finite-differences with the increment set by the user in the `derinc` variable in the `parameter_groups` table (appendix 1).

Requirements

Several requirements must be met to use this parallel implementation for Jacobian calculations.

1. Condor and the related program DAGman must be installed and active on both the master node where bgaPEST is to run and at least one worker node. Details for installation are available from Condor Team (2012).

2. Python, including the module numpy, must be installed on all compute nodes that will be used by Condor, including the master node where bgaPEST is running.

3. In the folder where bgaPEST is running, the following files must be present

   - Condor_ATC.py
   - parallel_condor_Jacobian.py
   - jacobian_pre.py
   - jacobian_pre.bat (or jacobian_pre.sh on non-Windows operating systems)
   - jacobian_post.py
   - jacobian_post.bat (or jacobian_post.sh on non-Windows operating systems)
   - condor_jacobian.sub.orig
   - unzip.exe
   - zip_results.py

4. A folder called "data" must be present in the same folder where bgaPEST is running on the master node. The "data" folder must contain all forward-model related files, the `.bgp` file, and all `.tpl` and `.ins` files used by bgaPEST. Additionally, the following files must be present

   - tempchek.exe (included in example files or available from `http://www.pesthomepage.org`)
   - inschek.exe (included in example files or available from `http://www.pesthomepage.org`)
   - condor_single_run.py
   - parallel_condor_Jacobian.py

5. Set the variable `deriv_mode=4` in the `algorithmic_cv` Keywords input block (appendix 1). This instructs bgaPEST to write necessary data transfer files and to execute the code necessary to run the Condor-based parallel Jacobian calculation.

File modifications

The file `condor_jacobian.sub.orig` must be customized to contain instructions specific to the Condor network on which bgaPEST is running for the `requirements` and the `request_memory` variables. Details of these variables are described by Condor Team (2012). All other variables should remain unchanged from the files provided with this documentation.

Description of the Method

This parallel implementation of bgaPEST using Condor is a scripted approach using a combination of existing PEST utilities and custom Python code. The general approach consists of the following steps:

1. Files are written by bgaPEST to communicate to the parallel codes information specific to the current run including derivative increments (specified by the user in the `derinc` variable).
2. At the time of each Jacobian calculation, the current parameter values are written to a temporary file
3. The current parameter values are combined with all other necessary model files in the "data" subfolder, which is compressed into a zipfile.
4. The Condor submit file is updated to specify the correct number of model runs required to calculate the Jacobian matrix.
5. A single-node directed acyclic graph (DAG) is initiated using the Condor utility `DAGman`. The need for the DAG is to monitor the set of Condor jobs corresponding to the specific model runs. `DAGman` monitors the progress of the DAG and returns control to bgaPEST ocne the entire Jacobian is calculated.
6. On each worker node, the following tasks are performed:
    - Model input files are written by using the `tempchek.exe` utility
    - The model is run once using the model command line information provided by bgaPEST
    - The model output files are read by using the `inschek.exe` utility. The version of `inschek.exe` provided with bgaPEST has been modified to carry maximum numerical precision.
7. After the model is run for each perturbation and a base case, the derivatives are calculated using forward differences and conveyed back to bgaPEST by using a text file.

References Cited

Condor Team, 2012, Condor Version 7.6.6 Manual: Madison, Wisconsin, University of Wisconsin—Madison.

Muffels, C., Schreüder, W., Doherty, J., Karanovic, M., Tonkin, M., Hunt, R., and Welter, D., 2012, Approaches in highly parameterized inversion—GENIE, a general model-independent TCP/IP run manager, U.S. Geological Survey Techniques and Methods, book 7, chap. C6, 26 p.

Schreüder, W., 2009, Running BeoPEST, *in* Proceedings, PEST Conference 2009, Potomac, Md., November 1–3, 2009: Bethesda, Md., S.S. Papadopulos and Associates, p. 228–240.

Appendix 5—Single-Layer Example Application

The first example application presented is a single-layer groundwater model. The forward model is a steady-state MODFLOW-2005 model with 21 rows and 21 columns and with constant row and column spacing of 1 m. The hydraulic conductivity field is heterogeneous, varying from 6.068×10^{-5} to 0.048 meters per day. The true hydraulic conductivity field is depicted in figure 5.1. Observations of head at the locations shown in figure 5.4 were used for parameter estimation. To generate observations representing what would be field measurements in a non-synthetic case, the model was run forward using the true synthetic hydraulic conductivity values and the resulting head values were perturbed with normally distributed noise with mean of zero and standard deviation of 0.01 m.

The boundary conditions are constant head, highest at the northwest corner and linearly decreasing to the southeast, as depicted in figure 5.2. There is a well at row 13, column 6, extracting water at a constant rate of 0.231 liters per minute. No recharge is simulated in this case.

Two options exist to calculate sensitivity (Jacobian) matrices: (1) an experimental adjoint-state version of MODFLOW or (2) finite difference calculations using the Python linkage and PEST, as capable in the released version of bgaPEST. Two scenarios were tested as well: (1) a case in which the epistemic error term $\left(\sigma_R^2\right)$ is estimated and (2) one where σ_R^2 is fixed at a value approximately 2 orders of magnitude higher than the artificial noise used to corrupt the synthetic "true" values previously generated by the model—1.0 m. This level of epistemic uncertainty is intended to be unrealistically high, but it encompasses both measurement and modeling error and was thus used to demonstrate a case where overfitting would be avoided at all costs.

Figures 5.3 and 5.4 show the estimated parameter field and the corresponding squared residuals, respectively, for the case in which σ_R^2 is fixed at 1.0 m. In this case, the relatively high value ascribed to epistemic uncertainty certainly prevents overfitting. The solution, in fact, is smooth, and the linear variogram slope parameter θ was estimated to be 3.66×10^{-1}. Appropriately, the most structure expressed in the parameter field and the correspondingly lowest residuals are found near the pumping well where stress is the greatest (and therefore the amount of information is the greatest) is found. This phenomenon is discussed in further detail by Fienen and others (2008).

The smoothness of the parameter field is as expected from the bgaPEST algorithm and consistent with the maximum entropy property of the algorithm. In other words, the true hydraulic conductivity field is rough but the algorithm estimates parameters that smooth over the rough areas. The "smearing" of higher values across a larger area than in reality is also consistent with both maximum entropy and information content—an anomalous region of high hydraulic conductivity emanates from the area of the well in response to the focus of stress (and therefore information) being near the pumping well.

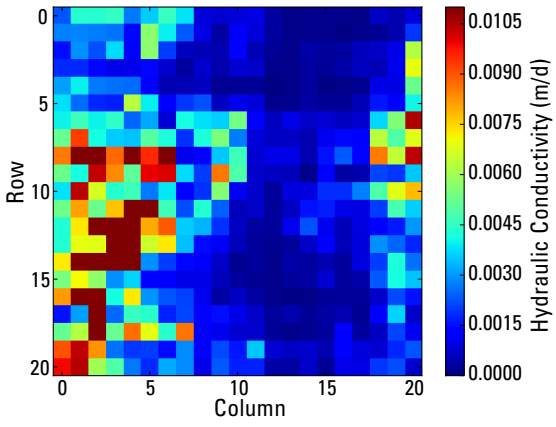

Figure 5.1. True hydraulic conductivity field for the single-layer example application.

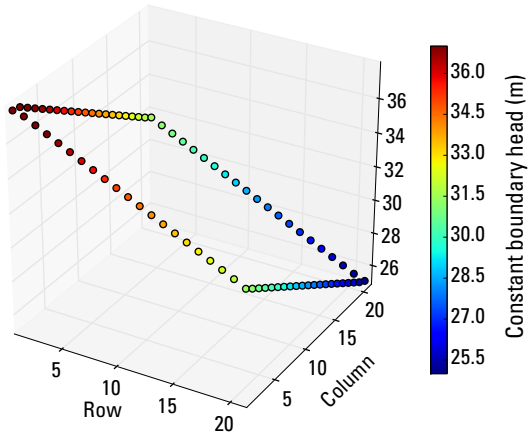

Figure 5.2. Boundary condition (constant head) for the single layer example application.

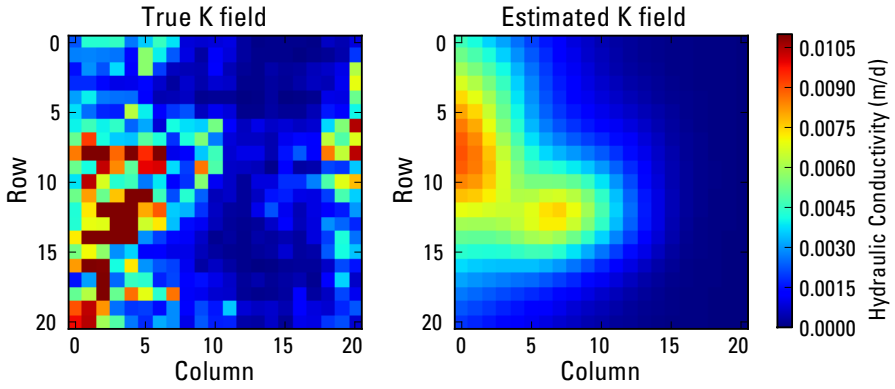

Figure 5.3. Parameters (hydraulic conductivity field) estimated for the single-layer example application using bgaPEST with σ_R^2 fixed at 1.0 m.

Figures 5.5 and 5.6 show the estimated parameter field and the corresponding squared residuals, respectively, for the case in which σ_R^2 is estimated by the bgaPEST algorithm. The estimated value for epistemic uncertainty was 1.007×10^{-1} and the variogram slope (θ) was 2.836×10^{-1}. These results are similar to the case in which σ_R^2 was held constant. The parameter field shows a similar shape and smoothness level, although figure 5.5 shows a bit more structure (roughness). Correspondingly, the residuals are generally lower. A key point here, however, is that the pattern of the residuals is similar and, again, reflects the general information content of the stress induced on the system.

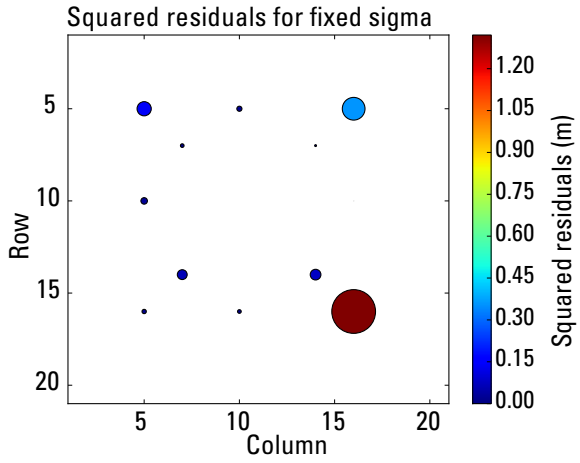

Figure 5.4. Squared residuals, plotted at their locations in the model, for the 1 layer example application using bgaPEST with σ_R^2 fixed at 1.0 m.

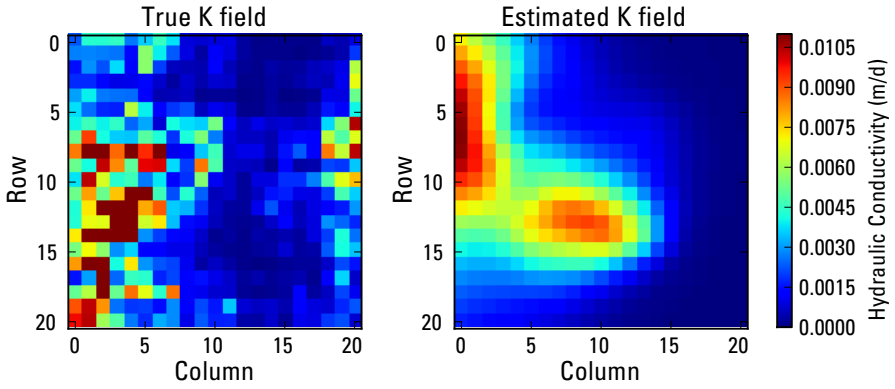

Figure 5.5. Parameters (hydraulic conductivity field) estimated for the single-layer example application using bgaPEST with σ_R^2 estimated by the bgaPEST algorithm.

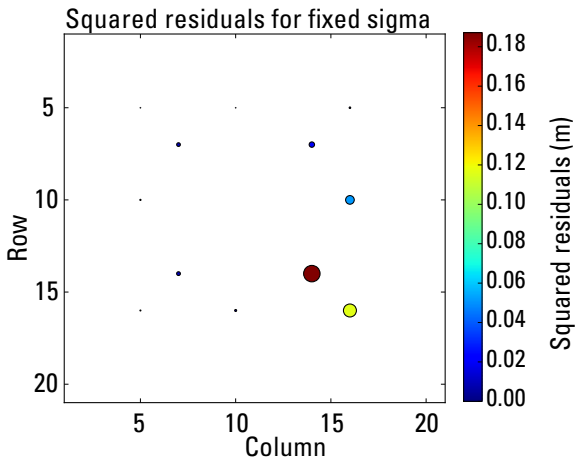

Figure 5.6. Squared residuals, plotted at their locations in the model, for the single-layer example application using bgaPEST with σ_R^2 estimated.

Reference Cited

Fienen, M.N., Clemo, T.M., and Kitanidis, P.K., 2008, An interactive Bayesian geostatistical inverse protocol for hydraulic tomography: Water Resources Research, v. 44, W00B01, doi:10.1029/2007WR006730.

Appendix 6—Three-Layer Example Application

A three-layer groundwater model is presented to explore the use of multiple beta associations, anisotropy, and a larger number of parameters. In this case, the model is 40 rows by 35 columns across 3 layers. The row spacing is 2.0 meters (m) while the column spacing is 1.5 m. The layers, from shallowest to deepest, are 1.8 m, 1.4 m, and 1.8 m in thickness, respectively. The disparate row, column, and layer spacing was used to test the Toeplitz compression option. The model has constant head boundaries on all sides (set at the same elevation—60 m) and a single well at row 18, column 17, extracting at 0.01 liters per minute from each layer. This low flow rate is not meant to represent typical field conditions, but rather highlights what can be learned with even a very small stress on the system.

The true parameter field, shown in figure 6.1, varies from 0.01 to 0.075 meter per day.

Five cases are illustrated here, as summarized in table 6.1. In all cases, each layer is treated as a separate beta association. In each of these layers, the initial value for the prior structural parameter (the linear variogram slope, θ) is 1.0×10^{-5}.

Cases 1 and 2 illustrate how the level of fit (and, therefore, the degree of roughness of the solution) can be influenced by adjusting the epistemic uncertainty term $\left(\sigma_R^2\right)$, so in these cases, σ_R^2 is set at a fixed value. In case 3, σ_R^2 is set low $\left(1.0 \times 10^{-5}\right)$ and the restricted maximum likelihood algorithm is given the freedom to estimate it. This setup illustrates the best achievable fit that one might achieve given the specific observation set provided without regard for overfitting. In cases 5 and 6, estimates are made with specification of

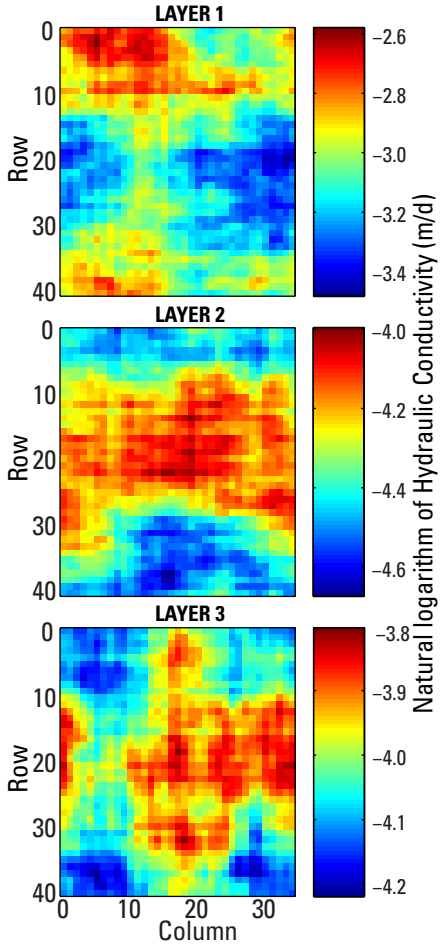

Figure 6.1. True synthetic hydraulic conductivity field for each layer in the three layer example application. Values are shown in natural logarithm space to make the differences more visible.

Table 6.1. Summary of the five cases investigated. The table shows which structural parameters were estimated and fixed, and also indicates anisotropy when used.

	Scenario	Case 1	Case 2	Case 3	Case 4	Case 5
	Initial σ_R^2	1.00E-01	1.00E-02	1.00E-05	1.00E-04	1.00E-01
	Estimated σ_R^2	-	-	7.79E-08	1.18E-05	-
Prior Parameters						
Beta Association 1	*Initial* θ	1.00E-05	1.00E-05	1.00E-05	1.00E-05	1.00E-05
	Estimated θ	2.46E-03	1.55E-02	1.25E-02	5.54E-03	3.61E-03
Beta Association 2	*Initial* θ	1.00E-05	1.00E-05	1.00E-05	1.00E-05	1.00E-05
	Estimated θ	6.16E-03	2.47E-02	1.34E-02	3.19E-03	7.97E-03
Beta Association 3	*Initial* θ	1.00E-05	1.00E-05	1.00E-05	1.00E-05	1.00E-05
	Estimated θ	2.46E-03	1.55E-02	1.21E-02	2.51E-03	7.73E-05
Anisotropy Parameters						
Beta Association 1	*horiz_angle*	-	-	-	0.0	0.0
	horiz_ratio	-	-	-	100.0	100.0
	verical_ratio	-	-	-	1.0	1.0
Beta Association 2	*horiz_angle*	-	-	-	0.0	0.0
	horiz_ratio	-	-	-	100.0	100.0
	verical_ratio	-	-	-	1.0	1.0
Beta Association 3	*horiz_angle*	-	-	-	0.0	0.0
	horiz_ratio	-	-	-	100.0	100.0
	verical_ratio	-	-	-	1.0	1.0

anisotropy in the prior covariance. Inspection of the true parameter field in figure 6.1 suggests a possible correlation along the horizontal axis, indicative of a channel feature. In cases 5 and 6, therefore, an arbitrarily chosen ratio of 100 is applied with a rotation angle of zero. In case 4, like in case 3, σ_R^2 is estimated to achieve the best possible fit, whereas in Case 5, σ_R^2 is held constant at 1.0×10^{-1}.

Figures 6.2 and 6.3 show the estimated hydraulic conductivity field and squared differences between measured and observed head values, respectively, for case 1. In this case, meant to be conservative with respect to overfitting, the squared differences are smaller in magnitude than the specified value of σ_R^2 $\left(1.0 \times 10^{-1}\right)$ and very little roughness in the solution is required to achieve the level of fit desired.

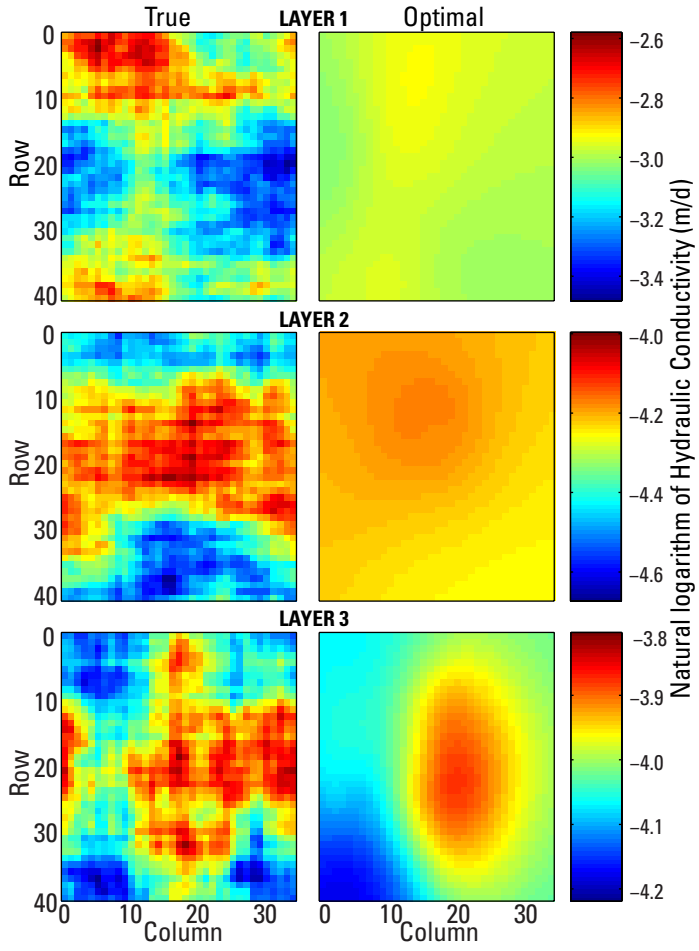

Figure 6.2. Case 1: Hydraulic conductivity fields estimated by using bgaPEST compared to the true, synthetic hydraulic conductivity field. σ_R^2 is held constant at 1.00×10^{-1}.

Figures 6.4 and 6.5 show the estimated hydraulic conductivity field and squared differences between measured and observed head values, respectively, for case 2. In this case, the specified value of σ_R^2 $\left(1.0 \times 10^{-2}\right)$ is lower than in case 1 and, accordingly, the squared head differences are lower, and more structure (roughness) is observed in the parameters, as expected. Note that, in this case, even with very low residuals, the parameter fields estimated are a smoothed representation of the "truth."

Figures 6.6 and 6.7 show the estimated hydraulic conductivity field and squared differences between measured and observed head values, respectively, for case 3. In this case, the value of σ_R^2 is estimated by the restricted maximum likelihood value algorithm. The head values match perfectly to machine precision, and the roughness of the field is the greatest of cases 1 through 3, as expected. The major features of the "true" hydraulic conductivity field are reproduced by this solution although they are smoothed, somewhat, as expected. Importantly, although the highest hydraulic conductivity values in layer 2 are slightly offset to the west, no artifacts are introduced that would be considered spurious in this solution.

Figures 6.8 and 6.9 show the estimated hydraulic conductivity field and squared differences between measured and observed head values, respectively, for sase 5. In this case, the value of σ_R^2 is set very low $\left(1.0 \times 10^{-4}\right)$ to attempt to achieve excellent fit while introducing anisotropy with the principal direction aligned with the horizontal axis. In layer 1, a somewhat spurious artifact is visible in the form of a high hydraulic conductivity zone near the middle of the field. The head targets almost match within machine

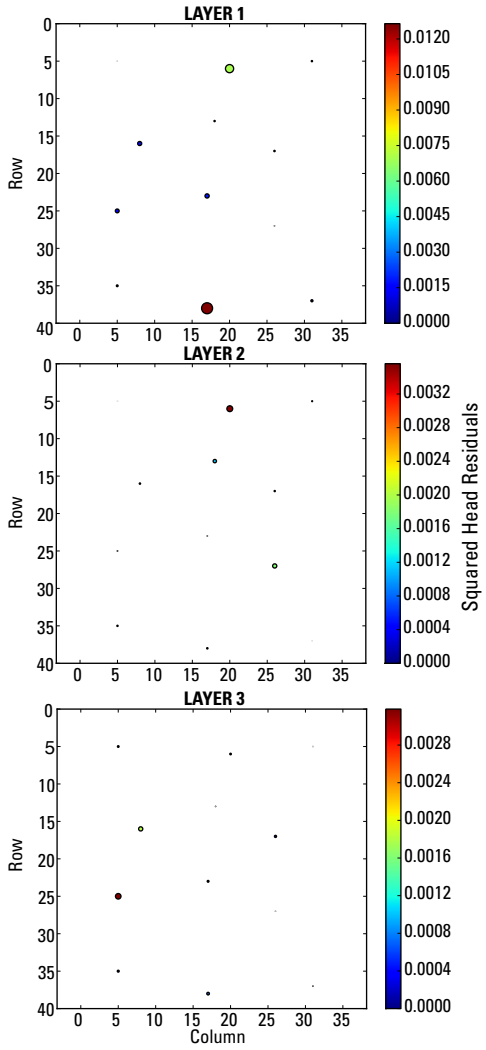

Figure 6.3. Case 1: Squared differences between modeled and "true" head values. Symbol size qualitatively indicates magnitude, and color scale quantifies magnitude. Locations of the circles indicate observation location in the model domain in plan view. σ_R^2 is held constant at 1.00×10^{-1}.

precision, however, and all other features are reasonable. This highlights the fact that, within a single beta association, if anisotropy is used, *all* features estimated will roughly correspond to that framework so, in a Bayesian sense, the answer is *conditional* on the prior assumption that the anisotropy is an appropriate general characteristic shape of the parameter field. Such assumptions must be made cautiously.

Figures 6.10 and 6.11 show the estimated hydraulic conductivity field and squared differences between measured and observed head values, respectively, for case 5. In this case, the value of σ_R^2 is set at a the same value as case 1 $\left(1.0 \times 10^{-1}\right)$ to compare a solution with and without anisotropy assumed. Because anisotropy is a reasonable characteristic of the "true" field in this case, better fits are achieved (nearly an order of magnitude lower residuals) and the general pattern of the parameter field is better in case 5 with anisotropy than in case 1 without anisotropy. This highlights the power that anisotropy can bring to a parameter estimation problem when it is appropriate even when σ_R^2 is set conservatively to avoid overfitting. As discussed above, however, this anisotropy will, in a sense, force the solution to conform to such a shape, so it's use should be approached with caution.

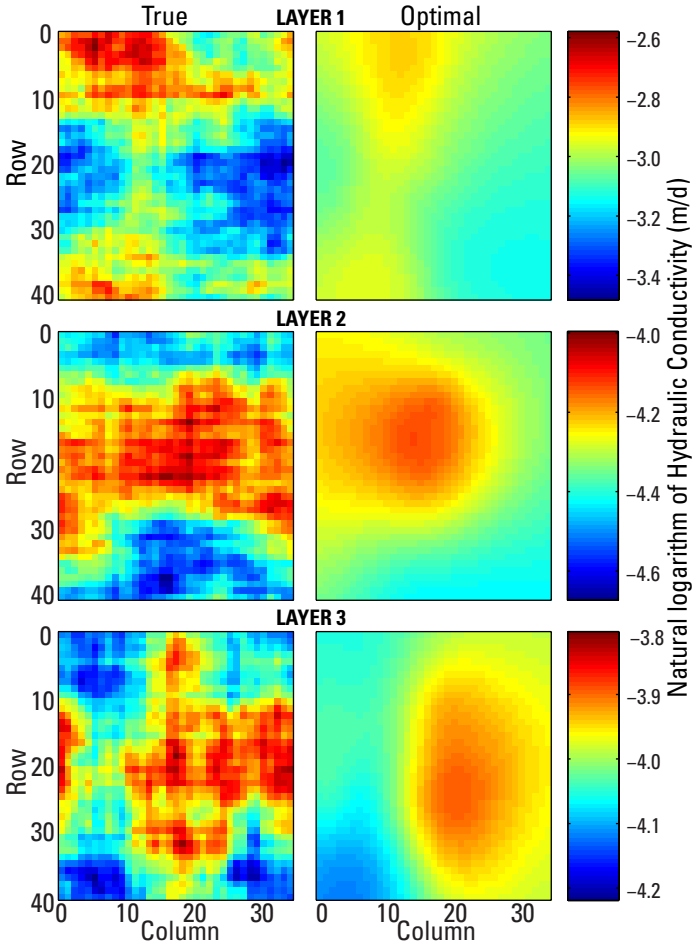

Figure 6.4. Case 2: Hydraulic conductivity fields estimated by using bgaPEST compared to the true, synthetic hydraulic conductivity field. σ_R^2 is held constant at 1.00×10^{-2}.

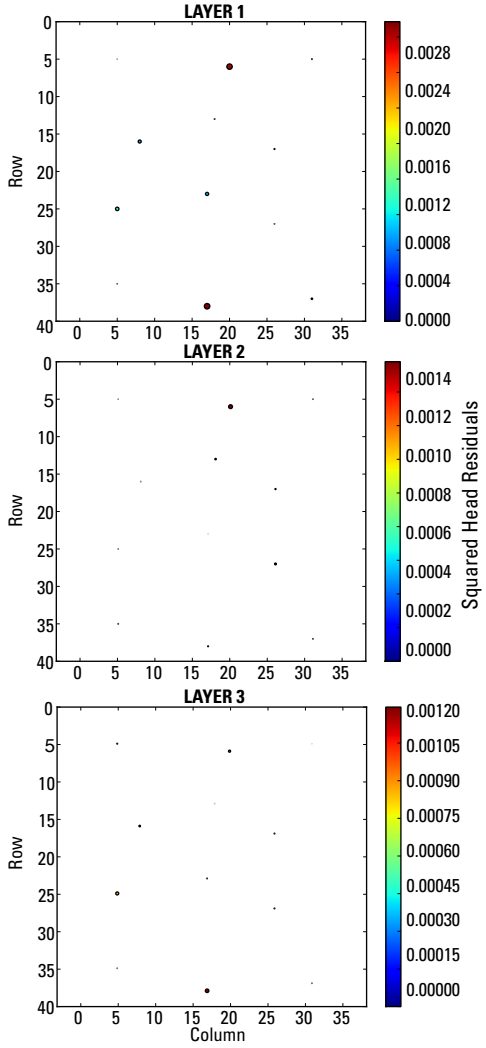

Figure 6.5. Case 2: Squared differences between modeled and "true" head values. Symbol size qualitatively indicates magnitude, and color scale quantifies magnitude. Locations of the circles indicate observation location in the model domain in plan view. σ_R^2 is held constant at 1.00×10^{-2}.

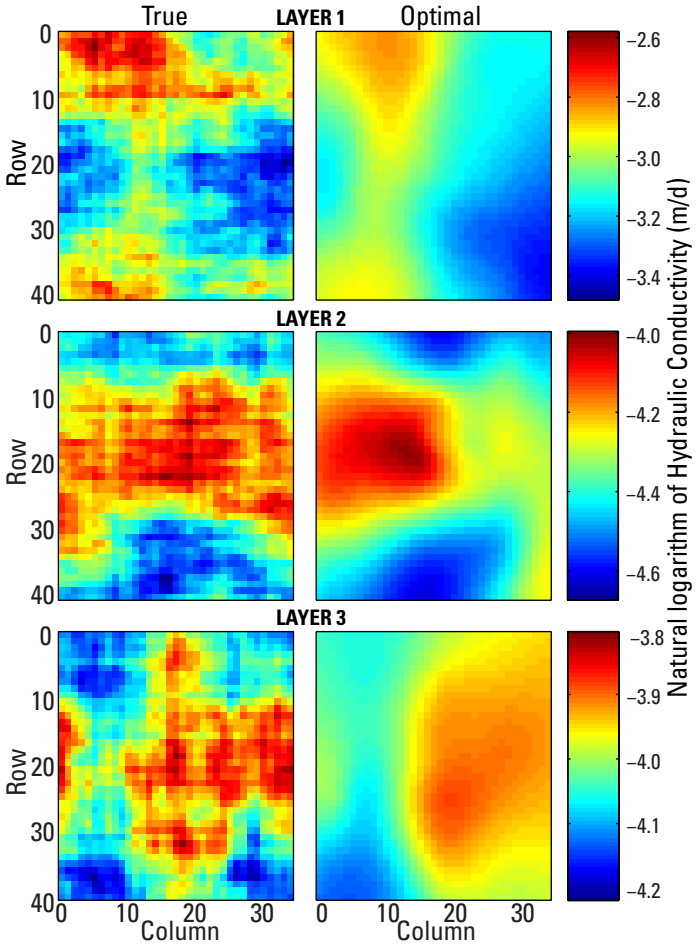

Figure 6.6. Case 3: Hydraulic conductivity fields estimated by using bgaPEST compared to the true, synthetic hydraulic conductivity field. σ_R^2 is initially 1.00×10^{-5} and estimated by bgaPEST at an optimal value of 7.79×10^{-8}.

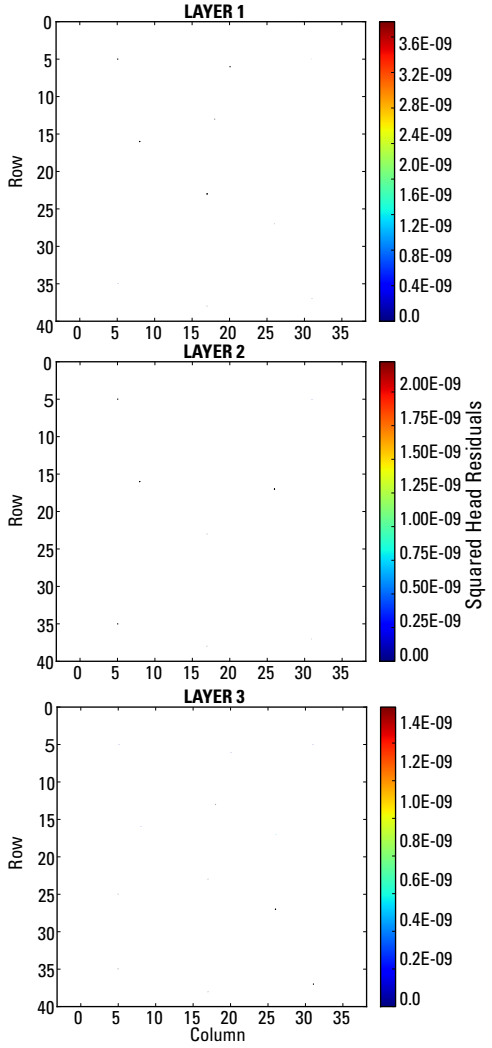

Figure 6.7. Case 3: Squared differences between modeled and "true" head values. Symbol size qualitatively indicates magnitude, and color scale quantifies magnitude. Locations of the circles indicate observation location in the model domain in plan view. σ_R^2 is initially 1.00×10^{-5} and estimated by bgaPEST at an optimal value of 7.79×10^{-8}.

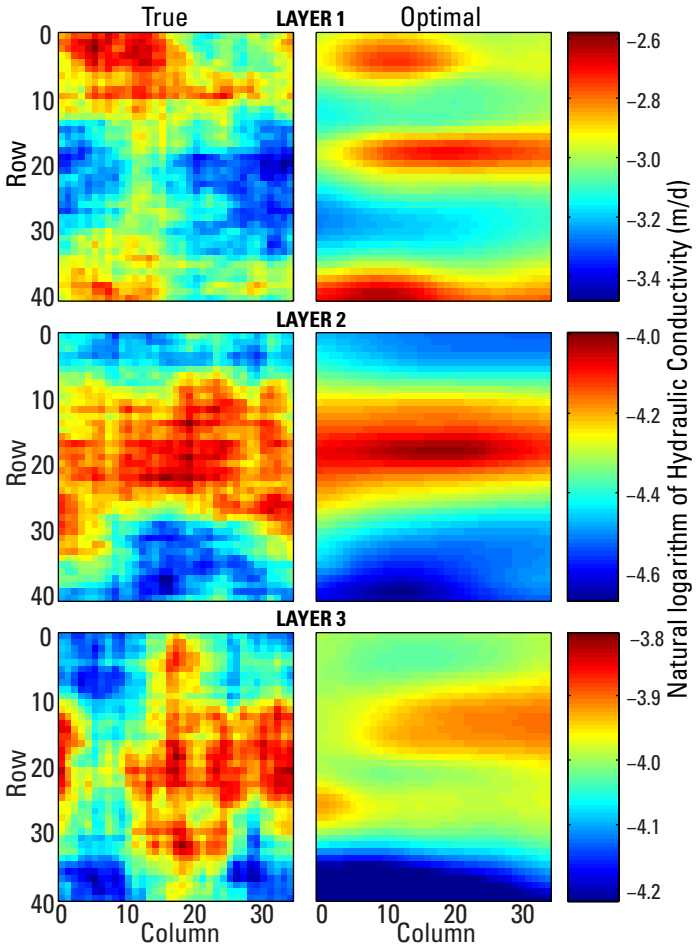

Figure 6.8. Case 4: Hydraulic conductivity fields estimated by using bgaPEST compared to the true, synthetic hydraulic conductivity field. σ_R^2 is initially 1.00×10^{-4} and estimated by bgaPEST at an optimal value of 1.18×10^{-5}. Parameter anisotropy also invoked as described in Table 6.1.

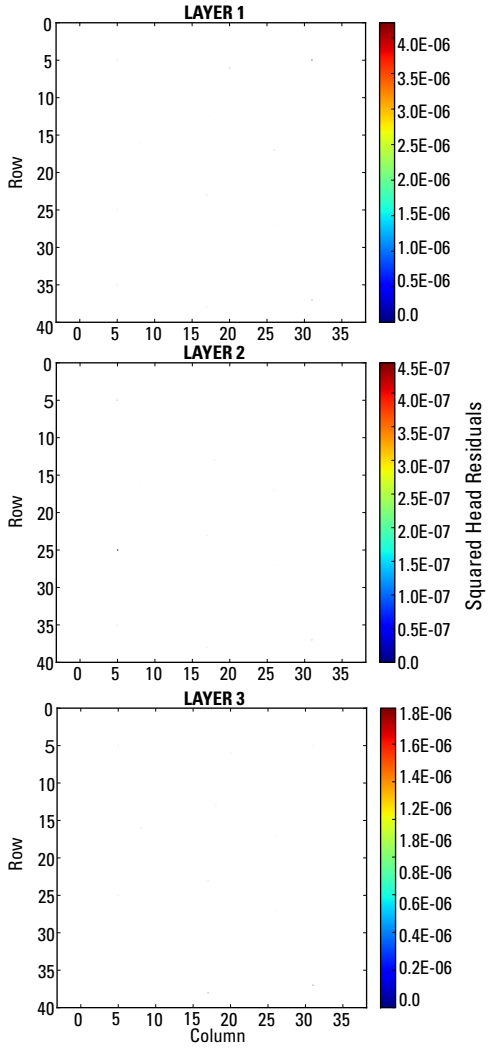

Figure 6.9. Case 4: Squared differences between modeled and "true" head values. Symbol size qualitatively indicates magnitude, and color scale quantifies magnitude. Locations of the circles indicate observation location in the model domain in plan view. σ_R^2 is initially 1.00×10^{-4} and estimated by bgaPEST at an optimal value of 1.18×10^{-5}. Parameter anisotropy also invoked as described in Table 6.1.

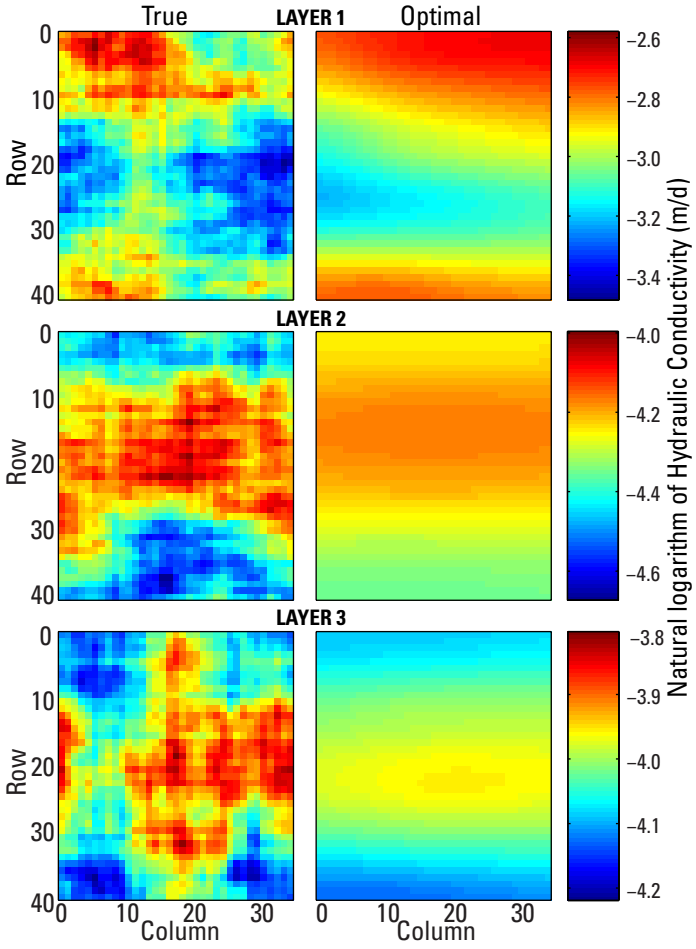

Figure 6.10. Case 5: Hydraulic conductivity fields estimated by using bgaPEST compared to the true, synthetic hydraulic conductivity field. σ_R^2 is held constant at 1.00×10^{-1}. Parameter anisotropy also invoked as described in Table 6.1.

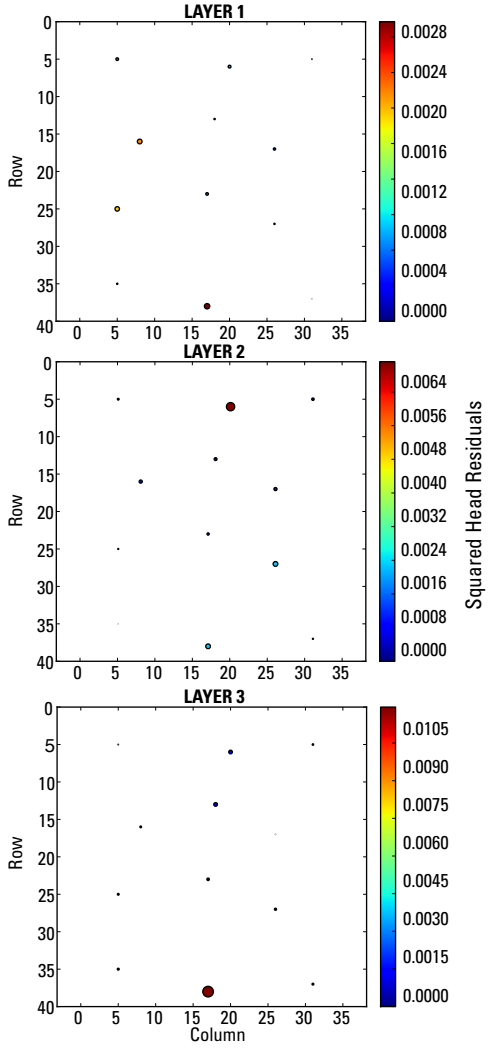

Figure 6.11. Case 5: Squared differences between modeled and "true" head values. Symbol size qualitatively indicates magnitude, and color scale quantifies magnitude. Locations of the circles indicate observation location in the model domain in plan view. σ_R^2 is held constant at 1.00×10^{-1}. Parameter anisotropy also invoked as described in Table 6.1.

Appendix 7—Reverse Flood Routing Example Application

An example application of reverse flood routing in open channels is presented in this section (D'Oria and Tanda, 2012). Reverse flood routing is useful to obtain hydrographs at upstream ungaged stations by means of information available at downstream gaged sites. The considered channel was prismatic and 20 kilometers long; the cross sections (spaced by 100 meters (m)) were trapezoidal in shape with bottom width of 10 m and side slope of 2/1. A longitudinal channel slope of 0.001 and a unitless Manning coefficient of 0.033 were adopted. The Manning coefficient is expressed as unitless when all other quantities are in SI units (as is the case here). A correction factor would be required if English units were used for the other quantities. The upstream and downstream boundary conditions were a streamflow time series and the uniform flow condition, respectively. The initial condition was set consistent with the steady state of a constant flow rate equal to the first value of the upstream hydrograph. The BASEChain module of BASEMENT (Faeh and others, 2011) that solves the De Saint Venant equations for unsteady one dimensional flow was adopted as forward model. A flood wave with time to peak of 2.5 hours, peak flow of 164 cubic meters per second (m^3/s) and base flow of 25 m^3/s was considered to obtain the corresponding downstream outflow subsequently corrupted with multiplicative random errors and used in the inverse procedure. The simulation time was equal to 15 hours; the input and output hydrograph time discretization was constant and equal to 5 minutes resulting in 181 values. The initial flow condition and the downstream streamflow time series (181 observations) were then used to estimate the inflow hydrograph (181 parameters). The initial parameter values were set to the mean value of the observations. The sensitivity (Jacobian) matrix was evaluated by means of a finite-difference calculations using the Python linkage and PEST, as capable in the released version of bgaPEST. The epistemic error term σ_R^2 and the linear variogram slope parameter θ were estimated by the restricted maximum likelihood value algorithm.

In figure 7.1 the actual input hydrograph, the actual downstream hydrograph, assessed by applying the forward model, and the error corrupted one used for the inversion are reported along with the reproduced inflow and the corresponding outflow. Table 7.1 summarizes the estimated structural parameters. In the first case the downstream hydrograph was corrupted with a 1 percent multiplicative random error (figure 7.1a), in the second case a 10 percent multiplicative random error was used (figure 7.1b). In both the inversions, there is a close match between the estimated input hydrograph and the actual one; the peak flow and time are properly reproduced. The estimated epistemic error variance increases in the second case taking account of the higher erroneous observations (table 7.1).

References Cited

D'Oria, M., and Tanda, M.G., 2012, Reverse flow routing in open channels—A Bayesian geostatistical approach: Journal of Hydrology, doi: 10.106/j.hydrol.2012

Faeh, R., Mueller, R., Rousselot, P., Vetsch, D., Volz, C., Vonwiller, L.R.V., and Farshi, D., 2011, System manuals of BASEMENT, version 2.1: Zurich, Switzerland, ETH Zurich Laboratory of Hydraulics, Glaciology and Hydrology (VAW).

Table 7.1. Reverse routing: estimated structural parameters .

Random errors	1%	10%
θ [m^6s^{-3}]	2.0×10^{-2}	1.0×10^{-2}
σ_R^2 [m^6s^{-2}]	1.7×10^{-1}	9.3

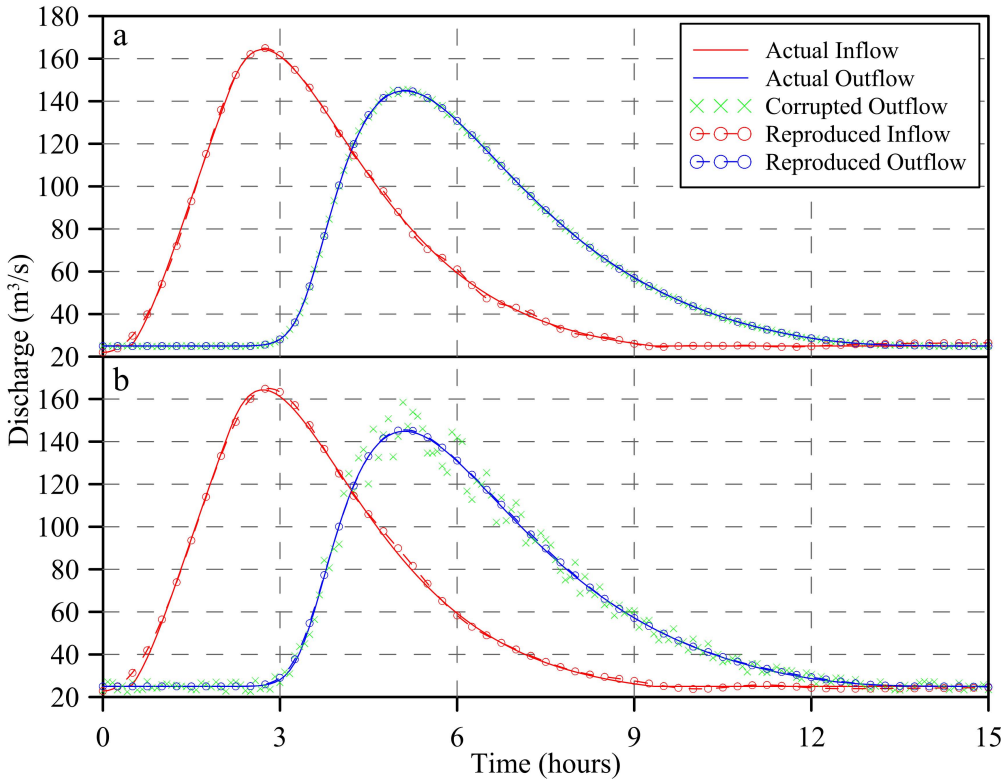

Figure 7.1. Reverse routing: inflow and outflow hydrographs for the prismatic channel. (a) the observations were corrupted with a 1 percent random error; (b) a 10 percent random error was used.

Glossary

Mathematical Symbols and Variables

$\boldsymbol{m}$ — Number of parameters

$\boldsymbol{n}$ — Number of observations

$\boldsymbol{p}$ — Number of beta associations

$\mathbf{s}$ — Vector of parameters

$\mathbf{s}_0$ — Starting values of parameters

$\mathbf{y}$ — Vector of observations

$\mathbf{y}'_k$ — Observations corrected for the k^{th} linearization: $\mathbf{y}'_k = \mathbf{y} - \mathbf{h}\tilde{\mathbf{s}}_k + \tilde{\mathbf{H}}_k\tilde{\mathbf{s}}_k$

$\mathbf{h}(\mathbf{s})$ — Modeled results of $\mathbf{s}$

$\mathbf{H}$ — Jacobian matrix of parameter sensitivities

$\mathbf{W}$ — Diagonal weight matrix

σ_R^2 — Epistemic uncertainty

σ^2 — Variogram variance or slope parameter

$\mathbf{R}$ — Epistemic uncertainty covariance

$p(\mathbf{s})$ — Prior probability of parameter values

$L(\mathbf{y}|\mathbf{s})$ — Likelihood function

$p(\mathbf{s}|\mathbf{y})$ — Posterior probability of parameter values

$\mathbf{X}$ — A selection matrix that can include drift of the prior mean

β — Prior mean values for each beta association

β^* — Initial prior mean values for each beta association used in the prior PDF

β_0 — Starting prior mean values for each beta association

θ — Vector of the structural parameters

$\hat{\theta}$ — Best estimate of the vector of the structural parameters

θ^* — Initial value of structural parameters (used in the prior PDF of θ)

θ_0 — Starting value of the structural parameters

$p(\theta)$ — Prior probability of structural parameter values

$L\left(\mathbf{y}_k^{'}|\theta\right)$ — Likelihood function in structural parameter optimization

$p\left(\theta|\mathbf{y}_\mathbf{k}^{'}\right)$ — Posterior probability of sturctural parameters given the current data

$\mathbf{G}_{ss}$ — Prior covariance of $(\mathbf{s}-\mathbf{X}\beta^*)$

$\mathbf{G}_{sy}$ — Cross covariance between observations and parameters

$\mathbf{G}_{yy}$ — Auto-covariance of the observations when covariance of the prior mean is included

$\mathbf{Q}_{ss}$ — Prior covariance of the parameters

$\mathbf{Q}_{yy}$ — Auto-covariance of the observations

$\mathbf{Q}_{\beta\beta}$ — Prior covariance of the prior mean values

$\mathbf{Q}_{\theta\theta}$ — Prior covariance of the structural parameter values

$\tilde{\mathbf{s}}$ — Current best estimate of $\mathbf{s}$ during quasi-linear runs

$\tilde{\mathbf{H}}$ — Current sensitivity matrix, a function of $\tilde{\mathbf{s}}$ during quasi-linear runs

ξ — Interpolation weights for the innovation in solving for $\hat{\mathbf{s}}$

$\hat{\mathbf{s}}$ — Best estimate of $\mathbf{s}$

$\hat{\beta}$ — Best estimate of β

Φ_T — Total objective function

Φ_M — Measurement component of objective function

Φ_R — Regularization component of the objective function

Φ_S — Structural parameters objective function

$\mathbf{s_{opt}}$ — Optimal value of parameters in the linesearch

ρ — Line search parameter

$E[\cdot]$ — Expected value

$\mathbf{d}$ — Separation distance in variogram calculation

ℓ — Integral scale in variogram calculation

$\mathbf{v}$ — Vector of epistemic uncertainty terms

$\mathbf{W}$ — Observation weight matrix

$\mathbf{T}$ — An arbitrary block Toeplitz matrix

ω — Observation weight

$\mathbf{p}$ — Untransformed parameter values in the context of power transformation

α — Exponent for power transformation

$\mathbf{V}$ — Posterior covariance matrix of the parameters