Workflow for DREAMzs Calibration of RHESSys

1. **Evaluate parameter correlations and select parameters to calibrate.**
   The RHESSysParamSelection.R script does the following:

   Select the Parameters to Calibrate
   The top 10% of the Elementary Effects (EE) for each of the 6 metrics are gathered for each hillslope (9 and 10). All parameters whose 95th percentile EE estimates were greater than the 10% threshold value were selected for calibration.

2. **Define priors for the parameters to be calibrated**
   Each parameter to be calibrated is assumed to have a uniform prior distribution, with bounds based on literature values or physical laws. The same priors were used as the SA, except for impervious landuse (set spatially from land cover dataset for l4 and l3 parameters), septic water load lower bound was decreased to allow for 0 septic load, and Ksat parameters' lower bounds were decreased slightly to allow for constraints to be met.

3. **Select MCMC Chain Starting Locations**
   The AnalyzeLikelihoods.R script was used to select the chain starting locations using a Latin Hypercube Sample (improvedLHS function in R lhs package) from the RHESSys parameter ranges defined by the bounds in the previous step. 4 sets of 10 chains, each with a different random seed, were used to evaluate random seed variability in the MCMC search. Files with the chain starting locations are named BaismanChainStarts_LHS10.txt.

   All parameters that were not calibrated were fixed at the same values for all chains. Most fixed values were the same as were used as defaults in the sensitivity analysis. The new values are in modified GIS2RHESSys files vegCollection_modified_Cal.csv, lulcCollectionEC_Cal.csv, and soilCollection_Cal.csv.

4. **Check and adjust parameter values to meet constraints**
   Parameter values were checked for satisfying the constraints, and adjusted if necessary using the DREAM_ParameterBoundChecks_ChainStarts.py script. This script is a modified MorrisSampling.py script to account for only the parameters that are being calibrated. The RunChainStarts.sh script is used to run this file. The following commands are needed:

   - working directory (e.g., RHESSysRuns)
   - initial random seed for the chain
   - directory of def files for calibration
   - round tolerance (<= 10)
   - problem file name with extension (e.g., BaismanCalibrationParameterProblemFile.csv)
   - chain parameter sample text file name without extension (e.g., 'BaismanChainStarts', 'BaismanChain_1' where 1 is chain iteration)

5. **Check that R packages for MCMC calibration are installed**
   The following packages are needed: BayesianTools, parallel, foreach, iterators, doParallel, and rlist. Note that the BayesianTools package was edited for this work and is provided in an associated data release.

6. **Setup files on Rivanna for calibration**
   The following directories (and files) should be made (placed) inside of a main folder (e.g., Baisman30mDREAMzs-10Ch):

   RHESSysRuns folder:

   - output folder – empty
   - RunChainStartsCheck.sh
   - DREAM_ParameterBoundChecks.py, DREAM_ParameterBoundChecks_ChainStarts.py
   - BaismanChainStarts_LHS10.txt
   - BaismanCalibrationParameterProblemFile.csv
   - MakeDefs_fn_Chains.py
   - ModifyVeg.py
   - RHESSysDREAM_NoG2W_rr_arg

   RHESSys_Baisman30m_g74 folder:

   - defs
     - all def files, with constants at their assumed values.
   - tecfiles
     - tec_daily_cal.txt
   - output - empty
   - clim
     - Cal_Feb2020Revised.tmax
     - Cal_Feb2020Revised.tmin
     - Cal_Feb2020Revised.rain
     - Cal_Feb2020Revised.base
   - flows
     - subflow.txt

surfflow.txt

worldfiles

worldfile.csv

worldfile.hdr – change the climate prefix name if needed

GIS2RHESSys folder

LikelihoodFun folder:

likelihood.py

Flow_MLEfits_Cal.py

obs folder:

The code will trim these two files to the specified start and end dates:

BaismanStreamflow_Feb2020Revised_Cal.txt

RunRHESSysDREAMCal-10Ch_rr_arg.sh

RunJoinDREAM.sh

JoinDREAM.R

Finally, one directory higher than the main folder, place the MoveDREAMFiles_10Ch.sh script.

7. Run DREAM MCMC Calibration

The RunRHESSysDREAMCal-10Ch_rr_arg.sh script is used to run the RHESSysDREAM_NoG2W_rr_arg.R script that sets up and runs the DREAM MCMC calibration. The following edits may be needed to run the shell script:

- SLURM commands (directories). Number of cores should be number of chains + 1

The rest of the commands are explained in the shell script. Some additional details are provided here:

- #2 – R random seed should be different when DREAM is restart.
- #4 - Observed streamflow record
- #10 -16 - WRTDS interpolation tables
- #17 – Full path to the file that describes the parameter names and bounds.
- #18 - File with chain starting locations (.csv file) OR the output from a previous chain run (.RData file)
- #20 – iterations = (Desired iterations per chain)*(number of chains)
- #21 - #30 – additional DREAMzs R function parameters.
- #34 - File that describes the parameter names and bounds. Just the name, not the full path.

- #40 – 42 - Initial random seed values. These same values are used even upon restart of DREAM.
- #43 – 44 - path to the processed streamflow observation .txt files. Processing happens within this R script, and these files are written in this script.
- #45 – 46 - Number of initial locations for the multi-start MLE solver

Note: 100 chain steps with 4 sets of 10 chains = about 111,000 files and 60 GB space (not RAM) in 3.5 days of wall clock time

8. **Join the output files into fewer files**
   The RunJoinDREAM.sh script is used to run the JoinDREAM.R script to join files and summarize useful output information. The following edits to the shell script may be needed:
   - SLURM commands (e.g., directories, email)
   - System argument #6: The starting value for the chain step index will change for each successive DREAM run.

9. **Move files to permanent storage**
   Run the MoveDREAMFiles_10Ch.sh script. The following edits may be needed:
   - Directories (SLURM, permanent storage directory, directory where files to be moved are located)
   - The chain step indicator: (e.g., s100). Do a find and replace for _s###. The ### should be the last step in the chain from the completed DREAM run.

10. **Analyze MCMC Output**
    The PlotRHESSysDREAM.R script is used to visualize the MCMC output results and to compute summary convergence metrics.