ERDC/CERL TN-22-1
November 2022

Automation of Gridded HEC-HMS Model Development Using Python

Initial Condition Testing and Calibration Applications

By Sean A. Matus and Daniel R. Gambill

ABSTRACT: The US Army Corps of Engineers's (USACE) Hydrologic Engineering Center-Hydrologic Modeling System (HEC-HMS) rainfall-runoff model is widely used within the research community to develop both event-based and continuous rainfall-runoff models. The soil moisture accounting (SMA) algorithm is commonly used for long-term simulations. Depending on the final model setup, 12 to 18 parameters are needed to characterize the modeled watershed's canopy, surface, soil, and routing processes, all of which are potential calibration parameters. HEC-HMS includes optimization tools to facilitate model calibration, but only initial conditions (ICs) can be calibrated when using the gridded SMA algorithm. Calibrating a continuous SMA HEC-HMS model is an iterative process that can require hundreds of simulations, a time intensive process requiring automation. HEC-HMS is written in Java and is predominantly run through a graphical user interface (GUI). As such, conducting a long-term gridded SMA calibration is infeasible using the GUI. USACE Construction Engineering Research Laboratory (CERL) has written a workflow that utilizes the existing Jython application programming interface (API) to batch run HEC-HMS simulations with Python. The workflow allows for gridded SMA HEC-HMS model sensitivity and calibration analyses to be conducted in a timely manner.

BACKGROUND: Flooding is consistently identified as one of the deadliest natural hazards in the United States by the National Weather Service (NWS). In 2017 alone, there were 116 flood-related deaths in the US, and more than $60 billion in flood damages (NOAA 2020). Floods are naturally occurring events whose intensity and duration are dependent on rainfall amounts and rates but also land characteristics, such as topography, land use, soil types, and antecedent soil moisture conditions (Funk 2006). There are three main flood categories: flash flooding, river flooding, and coastal flooding (French and Holt 1989). Flash and river flooding can be caused by a combination of saturation (i.e., soil is waterlogged) and infiltration (i.e., precipitation rate is greater than hydraulic conductivity) excess runoff, highlighting the importance of a watershed's characteristics in shaping the hydrologic response (Margulis 2017).

Rainfall-runoff models are used to simulate floods, providing water managers with physically based evidence to improve flood protection and awareness. The US Army Corps of Engineers's (USACE) Hydrologic Engineering Center (HEC) Hydrologic Modeling System (HMS) rainfall-runoff model is a physically based watershed model that has been used to study hydrologic responses across the globe (USACE 2020; Knebl et al. 2005; Chu and Steinman 2009; Cydzik and Hogue 2009; Halwatura and Najim 2013; Matus et al. 2020; Gambill et al. 2022). Although typically regarded as a simple model when compared to other hydrologic models (e.g., Gridded Surface Subsurface Hydrologic Analysis [GSSHA] and Weather Research and Forecasting [WRF]-Hydro), a variety of configurations exist within HMS that vary in complexity. HMS simulates physical watershed processes either through a lumped or distributed parameterization (USACE 2020). A lumped model represents entire subbasins as a discrete element, while a distributed model is parameterized on a grid, adding spatial heterogeneity to model parameters. HMS can also be configured as an event-based or continuous rainfall-runoff model. When switching from event-based to continuous rainfall-runoff modeling, temporal evolutions of evapotranspiration and soil storage cannot be ignored, thus adding several layers of complexity to the model (Bashar 2012; Singh and Jain 2015).

HMS includes the soil moisture accounting (SMA) module, which uses five reservoirs within three layers to represent dynamic moisture content within the soil profile, land surface, and vegetation canopy (USACE 2020). Figure 1 shows the five storage reservoirs SMA uses to simulate the different aspects of the rainfall-runoff process; all five require use-assigned initial condition values. The SMA module is forced by time series of precipitation and potential evapotranspiration (ET) (USACE 2000). Ten soil parameters are needed to model infiltration, percolation, soil storage, and groundwater storage (Fleming and Neary 2004). An additional three canopy and surface parameters are needed to model interception, surface depression storage, and actual evapotranspiration (Table 1). Certain transform and routing element methods in HMS introduce additional calibration parameters. For example, the ModClark transform method is quantified by time of concentration and a storage coefficient, while the Muskingum-Cunge routing method involves Manning’s *n*.

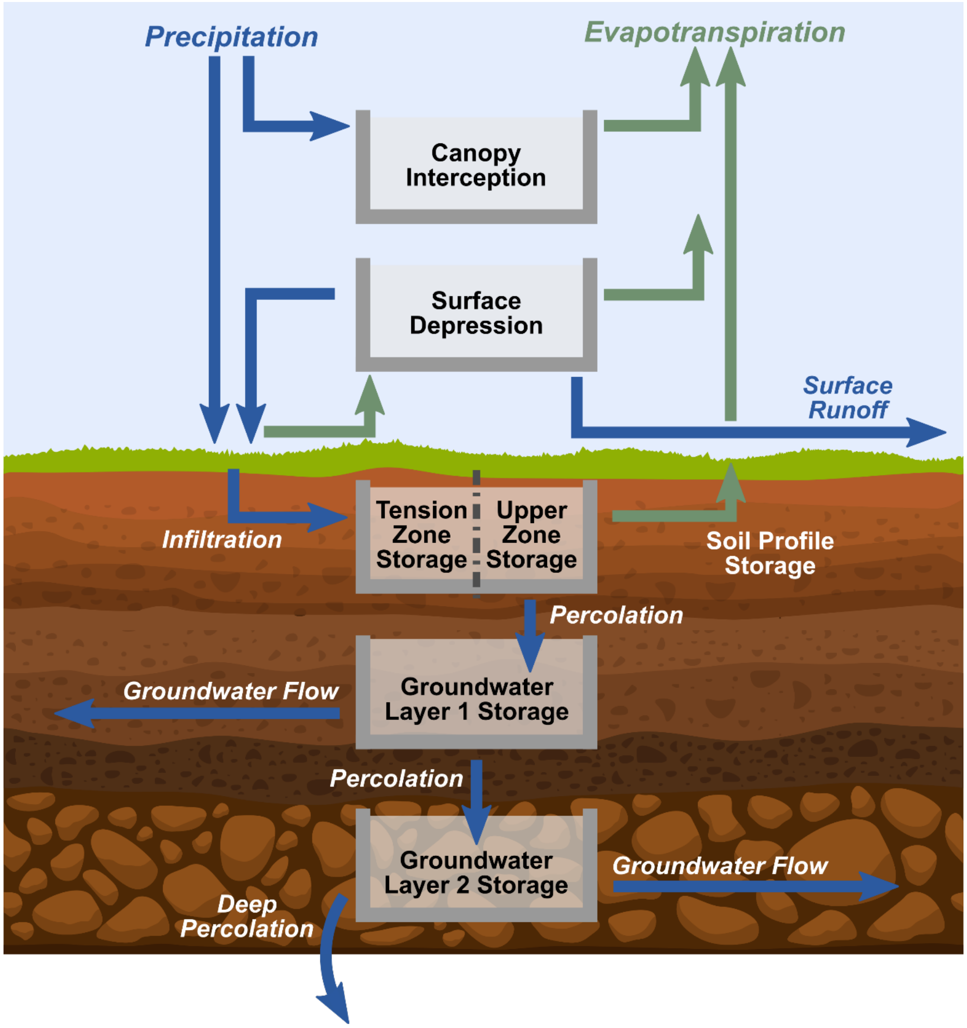

Figure 1.Conceptual schematic of the continuous soil moisture accounting (SMA) module within HEC-HMS based on Bennett (1998). SMA uses five reservoirs within three layers to represent dynamic moisture content within the soil profile, land surface, and vegetation canopy.

Regardless of the model configuration, uncertainty exists within the parameterization of all rainfall-runoff models. Some parameters are physically based and taken directly from field measures, but some are estimated (i.e., lookup tables) from literature (Fleming and Neary 2004; USACE 2020). Model calibration accounts for parameter uncertainty by matching the simulated streamflow to historical observations, while keeping the model bounded to physically feasible parameters.

Table 1. Initial conditions and parameters for a gridded SMA HMS model setup. Initial conditions and parameters are shaded *green* for those defined in the BASIN file and *blue* for the GRID file.

HMS Element	Hydrologic Method	Parameter	Initial Condition	Distributed/Lumped
Gridded Loss	Soil Moisture Accounting (SMA)	Max Infiltration		D
		Soil Percolation		D
		Soil Storage	✓	D
		Tension Storage		D
		GW 1 Storage	✓	D
		GW 1 Percolation		D
		GW 1 Coefficient		D
		GW 2 Storage	✓	D
		GW 2 Percolation		D
		GW 2 Coefficient		D
	Simple Canopy	Canopy Storage	✓	D
		Crop Coefficient		D
	Simple Surface	Surface Storage	✓	D
Transform	ModClark	Time of Concentration		L
		Storage Coefficient		L
Routing	Muskingum-Cunge	Manning's *n*		L

Calibrating continuous rainfall-runoff models is an iterative process that requires numerous simulations. HMS provides both deterministic and stochastic search algorithms to optimize a selected objective function by adjusting parameter values (USACE 2020). The HMS algorithms search for optimum values of all required initial conditions as well as the baseflow, transform, and routing parameters. Initial conditions and lumped parameters can be optimized using the existing HMS search algorithms. However, when using the more complex, gridded SMA module, the HMS optimization (as of version 4.10) cannot optimize any of the gridded SMA parameters. The problem is compounded by the gridded SMA parameters being stored in the native HEC database file format, HEC-DSS. There is currently not functionality (as of version 4.10) within the HMS optimization algorithms to search parameters input to HMS as DSS files.

OBJECTIVES: The goal of this study is to expand the capability of HEC-HMS to run multiple simulations without needing to use the graphical user interface (GUI). We accomplish this by putting a wrapper around HEC-HMS using Python within an Anaconda environment. The wrapper allows users to preprocess, run HMS, and postprocess multiple simulations from the same Anaconda environment. By eliminating the need to use the GUI, previously infeasible analyses that require numerous simulations (i.e., 100s) with a gridded SMA HEC-HMS model (e.g., ensemble streamflow) can be conducted. We demonstrate three examples of the expanded capability using a gridded SMA model of Cowhouse Creek at Fort Hood, TX, with (1) an initial condition sensitivity analysis, (2) single parameter calibration, and (3) multiparameter model calibration.

METHODS: The following section describes the HMS model components, how the Anaconda/Python framework wraps HMS, and how Python interacts with the different HMS model components. An example hydrologic model of Cowhouse Creek at Fort Hood, TX, is shown in Section "Case Study Site Description," and three analyses using the automated version of HMS are detailed in Sections "HMS Initial Conditions Sensitivity" and "HMS Parameter Calibration."

The workflow begins with the user needing to delineate a watershed and starting an HMS project in the GUI. When an HMS project is created, the GUI automatically creates and manages a set of configuration files, which are stored in the project directory (USACE 2020). These configuration files are standardized ASCII files detailing every aspect of the model. A full list and description of the HMS configuration files can be found in the *HMS User Manual* (USACE 2020). Table 2 provides a list of the HMS files utilized by the workflow. It is imperative that the user sets the parameterization schemes (e.g., gridded SMA loss) in the GUI and the uncalibrated parameter values of the model so that the ASCII files are populated with information that can later be modified. For distributed models, the user will need to set these parameters to default gridded DSS files. The assignment is done in the GUI so that the user can later modify the default directory path in the GRID file without needing to modify the BASIN file. In other words, it is easiest for the user to set up the model in the GUI to the point that it runs to completion once and then switch to modifying and running the model from an Anaconda environment.

Table 2. File extensions and descriptions associated with a HEC-HMS hydrologic model.

File Extension	File Description
hms	Project definition, including lists of basin models, meteorologic models, and control specifications
basin	Configuration of element properties, channel network, zones, and map layers
control	Configuration for start time, end time, and time interval. One per control specification
grid	List of all grid data with the corresponding paths pointing to the grid DSS files
dss	Project DSS file containing manual entry time-series gages and paired data
state	Saved state information for all the hydrologic elements in a specific basin model

Anaconda (docs.anaconda.com) was used to house the HMS wrapper. Anaconda is an open-source software. Users can build independent environments, and Anaconda has its own package dependency handler, making it easier for the user compared to building from source code. Additionally, Anaconda can support the user working with HMS either directly from command line or within an integrated development environment (IDE), such as Spyder or Jupyter Notebook. The YML file for our Anaconda environment can be made available upon request.

The Anaconda/Python wrapper allows the user to modify each of the ASCII files in Table 2. The wrapper will interact with a range of the files depending on what the user is trying to accomplish. Figure 2 outlines the capabilities of the Anaconda/Python wrapper for a typical hydrologic simulation. They are discussed in detail in order of importance:

> **Running HMS outside GUI:** The simplest, but most important, capability of the wrapper is computing an HMS simulation run without needing the GUI. We use existing HEC literature (USACE 2000; 2020), which steps the user through installing Jython and modifying the correct environment variables to run HMS from a BAT file (Juneau et al. 2010). Our expansion of this is (1) directing the BAT file to a Python-generated file containing the correct simulation metadata and (2) calling the BAT file from within Python with the *subprocess* module.
>
> **Reading HMS output:** Postprocessing HMS output without the need of the GUI is a significant time-saver for the user. By combining the Python modules *xarray* with *pydsstools* (www.github.com/gyanz/pydsstools), the user can read the output DSS file of a simulation and save it into a more convenient data structure (e.g., netCDF).
>
> **"Hot-Start":** Simulating streamflow is considered an initial condition problem. Gridded SMA HMS simulations can be started with the five SMA reservoirs assumed to be dry, but this will typically underestimate model results. "Hot starting" the model simulation with the output from a long, continuous simulation as the first timestep of the current run minimizes the initial condition uncertainty. The wrapper accomplishes this by reading a previous run's STATE file. The five storages (canopy, surface, soil, groundwater layer 1, and 2) for each subbasin are then overwritten into the current run's BASIN file as the initial storage percentages.
>
> **Modifying simulation start and end:** The wrapper allows the user to change the start and end times specified within the CONTROL file for a given run. Such capability is useful for studying the required time for the model storages to spin-up or for when the user has multiple precipitation sources with differing temporal coverages.
>
> **Modifying gridded parameters:** As stated earlier, the wrapper interacts with the paths of "dummy" datasets within the GRID file so that the user can switch in and out different gridded parameters or forcings without HMS needing any additional modifications (i.e., to the BASIN file). This capability is quintessential for model calibration as well as when running the hydrologic model with probabilistic precipitation data.
>
> **Modifying lumped parameters:** The wrapper interacts with basin and reach parameters within the BASIN file, so that the user can manipulate lumped parameter values. This capability is quintessential for model calibration.

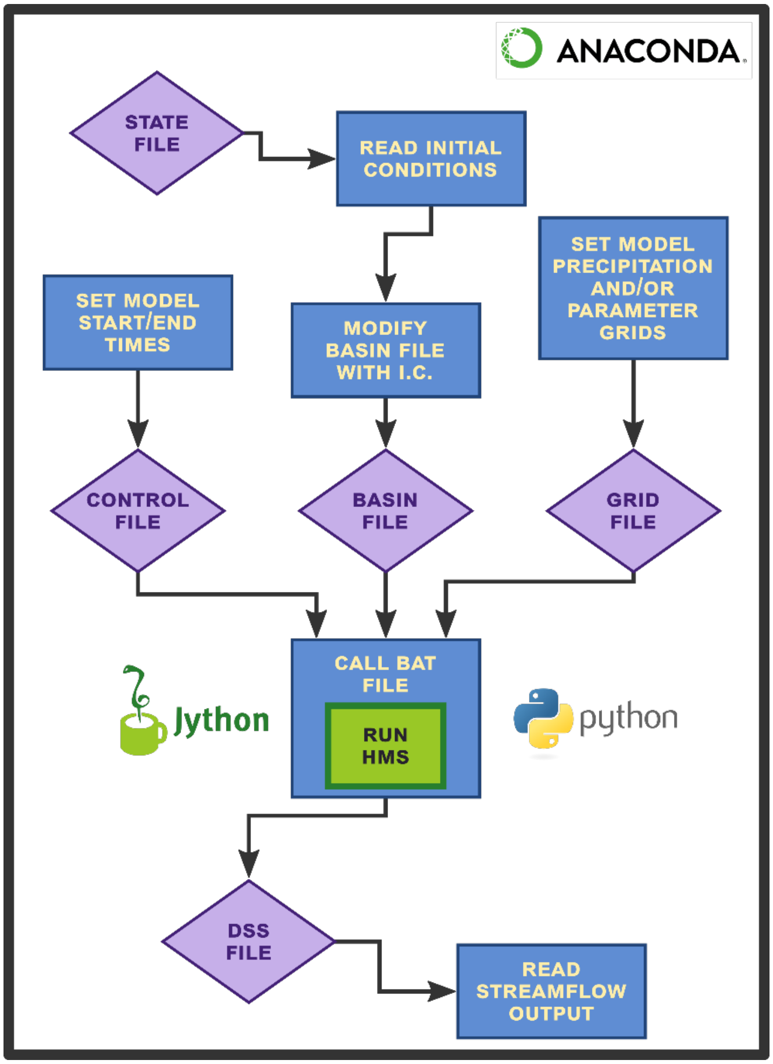

Figure 2. A flowchart of how the Anaconda/Python wrapper interacts with the different components of an HMS model. HMS model ASCII files (*purple diamonds*) are manipulated by Python scripts (*blue rectangles*, *yellow text*) within the Anaconda environment (*outer black outline*). For specifically executing HMS, a Python script wraps around an existing Jython wrapper (*green box*, *black text*).

The major advantage of this methodology is containing every step of the modeling process within a single Anaconda environment. We acknowledge that multiple pieces of this framework have already been developed. The novelty of this method is housing it all within a single environment, making not only running simulations but their accompanying input/output processing analyses simpler and faster.

Case Study Site Description. To demonstrate the Anaconda/Python wrapper, we show the results from a watershed HMS model included in an Environmental Security Technology Certification Program (ESTCP) demonstration and validation study (RC19-D3-5082). The Cowhouse Creek watershed HMS model was developed to forecast and investigate flash flooding impacts. The model was designed for continuous simulations, but frequent hot starts were required to demonstrate an operational capability when working with precipitation forecasts. The model used the gridded SMA algorithm due to the use of gridded National Centers for Environmental Prediction Environmental Modeling Center (NCEP/EMC) Stage IV precipitation data as the precipitation model input (Du 2011). The ModClark transform method and the Muskingum-Cunge routing method completed the HMS model configuration. The use of the gridded SMA loss method precluded automated parameter optimization of all gridded SMA parameters, including the six groundwater parameters. Prior to this work, calibration of the gridded SMA parameters would have required the user to manually update the corresponding DSS file in the HMS GUI and rerun the model numerous times. The process was time consuming. As a result, the wrapper was developed to automate gridded SMA parameter sensitivity and calibration analyses. The wrapper design also includes functionality to manipulate the lumped ModClark and Muskingum-Cunge parameters; therefore, any parameter in the model could be calibrated.

CASE STUDY RESULTS

HMS Initial Conditions Sensitivity. Determining an appropriate model spin-up duration for the gridded SMA reservoirs is a task that requires rerunning the model many times. The wrapper enables the process to be automated, simplifying a previously tedious and time-consuming task. To conduct this analysis, only the CONTROL file needs to be modified (Figure 2). The model is initialized with the five reservoirs dry; therefore, the BASIN file does not need to be modified.

The wrapper allows the user to loop through iterations of modifying the start time in the CONTROL file, running HMS, and processing the output storage time series from the DSS file. Comparatively, if the user were to conduct the same analysis in the GUI, those three steps would have to be manually completed for each simulation. Automating the analysis with the capability of the wrapper decreases the time necessary while also minimizing the possibility of human error.

The results of an initial condition sensitivity analysis with the Cowhouse Creek model are shown in Figure 3 for the soil storage reservoir of the gridded SMA scheme. Twelve simulations were run with the same end time (01 January 2016), but each run had a different start time corresponding to the firsts of each month (e.g., 01 January 2015 and 01 February 2015). The results are summarized across all Cowhouse Creek subbasins by reporting the median and interquartile range.

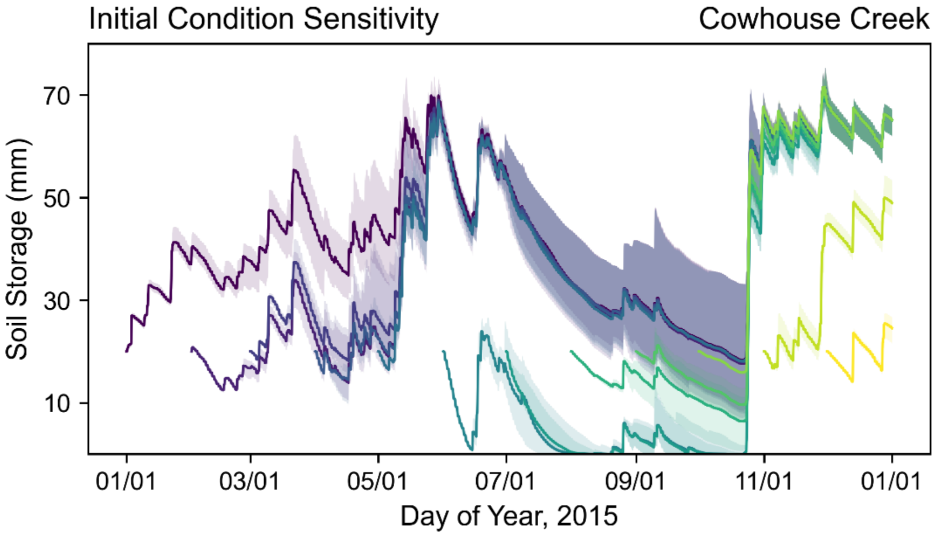

Figure 3. Time varying soil storages for the subbasins of the Cowhouse Creek HMS model for 12 different simulations. Each colored line represents the median soil storage of all the subbasins for 1 of the 12 simulations (with the interquartile range banded). Each of the 12 simulations has a different start time corresponding to the 12 firsts of the months in 2015. The final timestep soil storages converge for all simulations initialized before October, suggesting the appropriate spin-up duration.

Figure 3 shows that the soil storage converges for 10 of the 12 simulations. The shortest 2 simulations (starting on 01 November 2015 and 01 December 2015) are not long enough, and the final soil storage is underestimated compared to the other 10 simulations. If the end state from either of these 2 simulations was used to hot start a model simulating January 2016, then one would expect an underestimation of hydrologic responses involving saturation excess runoff. In other words, because the soil did not have enough time to spin-up, the soil in the hot-started simulation is drier than it should be, resulting in greater available storage for precipitation to infiltrate instead of being transformed into runoff due to soil saturation.

It is worth noting that soil storage is one of the five reservoirs within the gridded SMA scheme. To confidently spin-up the model, all five reservoirs should be investigated for convergence. A complete initial condition sensitivity analysis is easily accomplished with the wrapper by reading all five storage time series from the DSS file after each HMS run. Additionally, for this example, it is evident that a large precipitation event occurred in mid-October 2015, causing the soil to saturate. There is not necessarily a standard for how long the spin-up period should be. Figure 3 also shows the two simulations initialized on 01 January and 01 February converging over five months later in June. Having a wrapper around HMS allows the user to easily minimize a major source of uncertainty in the initial storage conditions with little effort compared to manually conducting the same analysis.

HMS Parameter Calibration. A more complicated application of the Anaconda/Python wrapper is aiding with the calibration of a gridded SMA model. Calibration is one of the most time-consuming aspects of building a gridded HMS model. The gridded SMA scheme requires more than 10 parameters, which are not all physically based. The problem is compounded by HMS optimization being incompatible with the SMA module due to an inability to search gridded parameters stored in DSS files. To use the wrapper in a calibration analysis, the workflow in Figure 2 is utilized with slight changes. The CONTROL file remains constant across the simulations, while the BASIN and GRID files are updated with new parameter values for each simulation.

Here, we demonstrate automated HMS capabilities with the Cowhouse Creek model for both single parameter (Figure 4) and multiparameter calibration (Figure 5). To conduct the calibration, the model is run several times, looping through a set of different multiplication factors applied to a parameter. Figure 4 depicts this analysis for a single parameter, with 14 different streamflow time series (*yellow/blue lines*) each corresponding to a different multiplication factor applied to the surface storage parameter. With a gridded SMA HMS model, there are 16 potential calibration parameters. By using this method, a sensitivity analysis can be applied to each of those 16 parameters. The method then determines which of those parameters are first-order drivers of the model physics (i.e., the most important during calibration). Reducing the total number of calibration parameters reduces the dimensionality of the calibration, drastically cutting down on the calibration computational load.

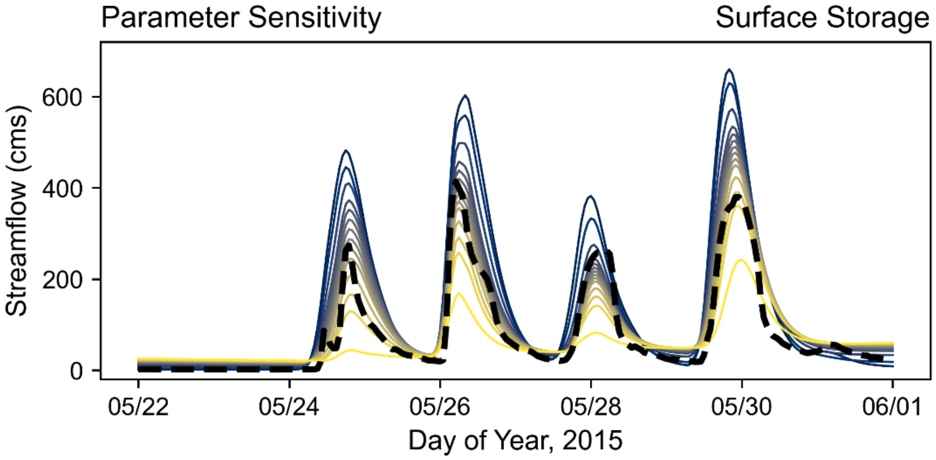

Figure 4. An example of HMS single-parameter calibration. Each colored line represents a hot-started simulation using the Cowhouse Creek model with a different grid of surface storage magnitudes. The *black dashed line* is the observed streamflow taken from a USGS gauge on Cowhouse Creek.

The workflow is the same for multiparameter calibration as it is for single parameter, except the model loops through unique combinations of the variables at different multiplication factors (e.g., 2 variables with 10 multiplication factors would be 100 runs). For each of the runs, the simulated streamflow is read from the DSS file and saved to a netCDF file. The output netCDF file can then be used to conduct postprocessing, which for this example was computing the Nash-Sutcliffe efficiency coefficient (NSE) with the simulated and observed streamflow.

An example of the multiparameter calibration is presented in Figure 5. We use two parameters here only for the ease of visualization, but the analysis does not have to be constrained to only two parameters. Figure 5(*a*) shows a heatmap of NSE scores for different combinations of the ModClark time of concentration and ModClark storage coefficient parameters. NSE scores closest to 1 are the most skillful, while negative scores are indicative of poorer performance compared to the observational mean. It is interesting to note that the uncalibrated model version (both parameters having a multiplication factor of 1) has a negative NSE, highlighting the importance of calibration. There is a maximum NSE for the combination of 1.5 factor for time of concentration and 2.0 for the storage coefficient.

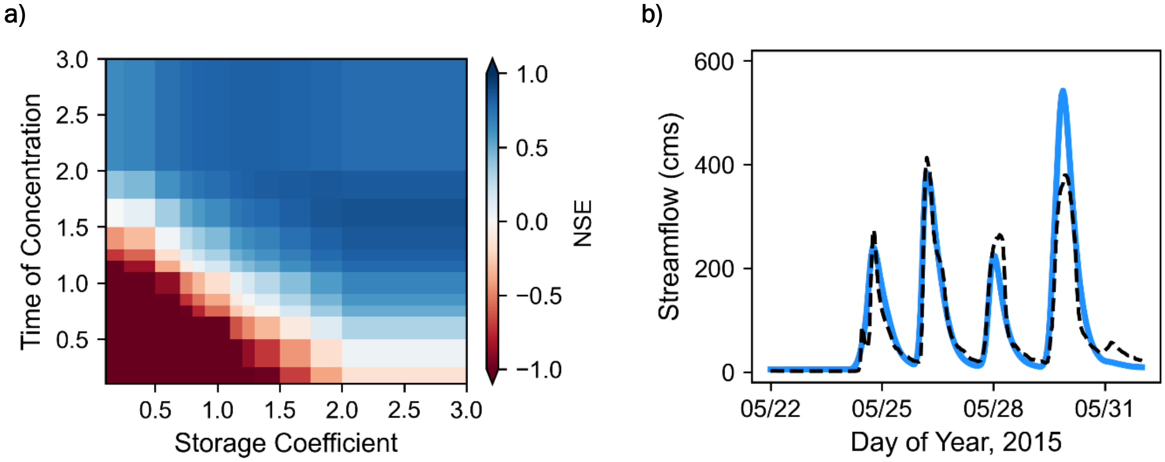

Figure 5. Calibration results for two parameters, time of concentration and transform storage coefficient, from a 10-day simulation with the Cowhouse Creek HMS model. (*a*) The Nash-Sutcliffe model efficiency coefficient plotted for variations of the two parameters, where variations are applied as a multiplication factor of the control parameter found from literature (i.e., factor of 2 on either axis is a 100% increase). (*b*) Observed streamflow (*black dashed line*) and simulated streamflow (*blue solid line*) using the parameters corresponding to the maximum NSE in (*a*).

Streamflow simulated using the calibrated Cowhouse Creek model with the new parameter values is shown in Figure 5(*b*). The simulated streamflow matches the observations well except for the last of the four hydrologic responses, which are being overestimated. Further investigation of the model output provided insight. In the simulation, the last event was being driven by primarily groundwater excess, causing the overestimation. As such, the next step in this calibration example would be to run additional simulations with the new parameters shown in Figure 5 but variable parameters pertaining to the groundwater storage in the SMA module.

SUMMARY: This study expands the capability of HEC-HMS through the development of an open-source Anaconda/Python wrapper. The wrapper centralizes the entire workflow to one workspace. The time saved from running HMS outside of the GUI allows for expanded analyses to be conducted that HEC-HMS currently does not support. Examples of these analyses are provided using an HMS model for the Cowhouse Creek watershed in Texas. Analyses of initial condition sensitivity and multiparameter calibration demonstrate the useability of the wrapper while highlighting its streamlining of model development, making previous manual methods obsolete.

ADDITIONAL INFORMATION: This work is supported by the Department of Defense Environmental Security Technology Certification Program RC19-D3-5082, Hydrologic Risk Forecaster (Hydro RF): Demonstration and Validation of Forecasting Model for Flooding and Infrastructure Risk on Military Lands. The Python codes used within the Anaconda environment are stored on a public-facing GitLab repository and can also be made available upon request.

REFERENCES

Anaconda. 2020. “Anaconda Distribution.” Anaconda Documentation. https://docs.anaconda.com/.

Bashar, K. E. 2012. “Comparative Performance of Soil Moisture Accounting Approach in Continuous Hydrologic Simulation of the Blue Nile.” *Water Science and Engineering* 5 (2): 1–10.

Bennett, T. H. 1998. “Development and Application of a Continuous Soil Moisture Accounting Algorithm for the Hydrologic Engineering Center Hydrologic Modeling System (HEC-HMS).” Master’s Thesis, University of California, Davis.

Chu, X., and A. Steinman. 2009. “Event and Continuous Hydrologic Modeling with HEC-HMS.” *Journal of Irrigation and Drainage Engineering* 135 (1): 119–124.

Cydzik, K., and T. S. Hogue. 2009. “Modeling Postfire Response and Recovery Using The Hydrologic Engineering Center Hydrologic Modeling System (HEC-HMS).” *Journal of the American Water Resources Association* 45 (3): 702–714.

Du, J. 2011. “NCEP/EMC 4KM Gridded Data (GRIB) Stage IV Data.” Version 1.0. UCAR/NCAR—Earth Observing Laboratory. https://doi.org/10.5065/D6PG1QDD.

Fleming, M., and V. Neary. 2004. “Continuous Hydrologic Modeling Study with the Hydrologic Modeling System.” *Journal of Hydrologic Engineering* 9 (3): 175–183.

French, J. G., and K. W. Holt. 1989. “Floods.” In *The Public Health Consequences of Disasters*, edited by M. B. Greg, 69–78. Atlanta, GA: US Department of Health and Human Services.

Funk, T. 2006. “Heavy Convective Rainfall Forecasting: a Look at Elevated Convection, Propagation, and Precipitation Efficiency.” Proceedings 10th Severe Storm and Doppler Radar Conference. Des Moines, IA, National Weather Association.

Gambill, D. R., M. M. Stoklosa, S. A. Matus, H. R. Howard, and G. R. Feezor. 2022. *White Sands Missile Range Thurgood Canyon Watershed: Analysis of Range Road 7 for Development of Best Management Practices and Recommendations*. ERDC/CERL TR-22-11. Champaign, IL: Engineering Research and Development Center, Construction Engineering Research Laboratory.

Halwatura, D., and M. M. M. Najim. 2013. “Application of the HEC-HMS Model for Runoff Simulation in a Tropical Catchment.” *Environmental Modelling & Software* 46:155–162.

Juneau, J., J. Baker, F. Wierzbicki, L. S. Muoz, V. Ng, A. Ng, and D. L. Baker. 2010. *The Definitive Guide to Jython: Python for the Java Platform*. New York: Apress.

Knebl, M. R., Z. L. Yang, K. Hutchison, and D. R. Maidment. 2005. Regional Scale Flood Modeling Using NEXRAD Rainfall, GIS, and HEC-HMS/RAS: a Case Study for the San Antonio River Basin Summer 2002 Storm Event. *Journal of Environmental Management* 75 (4): 325–336.

Margulis, S. 2017. *Introduction to Hydrology*. Self-published, iBooks.

Matus, S. A., F. Dominguez, D. R. Gambill, and H. R. Howard. 2020. “Embracing Uncertainty: Using Probabilistic Weather Forecasts to Make Ensemble Hydraulic Predictions at Remote Low-Water Crossings.” *Journal of Hydrometeorology* 21 (5): 953–969.

NOAA. 2020. "Weather Related Fatality and Injury Statistics." National Weather Service. https://www.weather.gov/hazstat/.

Singh, W.R., and M.K. Jain. 2015. "Continuous Hydrological Modeling Using Soil Moisture Accounting Algorithm in Vamsadhara River Basin, India." *Journal of Water Resource and Hydraulic Engineering* 4 (4): 398–408.

US Army Corps of Engineers (USACE). 2000. *Hydrologic Modeling System HEC-HMS Technical Reference Manual*. CPD-74B. Davis, CA: USACE, Hydrologic Engineering Center.

US Army Corps of Engineers (USACE). 2009. *HEC Data Storage System Visual Utility Engine: User's Manual*. CPD-79. Davis, CA: USACE, Hydrologic Engineering Center.

US Army Corps of Engineers (USACE). 2020. *Hydrologic Modeling System HEC-HMS User's Manual*. CPD-74A. Davis, CA: USACE, Hydrologic Engineering Center.