dpest: Streamlining Creation of PEST input files for DSSAT Crop Model Calibration

Luis Vargas-Rojas [1] and **Diane R. Wang** [1]

1 Department of Agronomy, Purdue University, West Lafayette, IN, United States

Summary

Process-based crop models simulate plant growth and can support crop improvement research by enacting different "what-if" scenarios. Decision Support System for Agrotechnology Transfer (DSSAT) is one of the most commonly used crop modeling platforms, containing modules that can be used to model different species (Jones et al., 2003). To simulate different crop varieties, DSSAT calibration is needed. This involves adjusting model parameters, which are values stored in a model input file, to determine which sets of values give rise to simulations that most closely match measured data from field experiments; these data can include single timepoint measurements as well as time-series information.

While the DSSAT installation includes Graphical User Interface (GUI) tools specifically designed for calibration, they are not amenable to time-series data. In contrast, the PEST software suite for parameter estimation and uncertainty analysis (Doherty, 2015), a command-line interface (CLI) tool, is a model-independent tool that can calibrate DSSAT models using not only the end-of-season crop data but also measurements collected within-season. To carry out model calibration, PEST requires three types of input files: template files (.TPL) that specify the parameters to calibrate, instruction files (.INS) that contain the guidelines to extract the model outputs, and the control file (.PST) that includes the specifications to manage all the calibration settings. Generation of these input files for PEST-based DSSAT calibration, however, can be a complex and time-consuming process, making it difficult to scale, for example, across many cultivars or strains.

To address these challenges, we developed *dpest*, a Python library that streamlines the generation of PEST input files for calibrating DSSAT; the current version supports DSSAT's wheat models. Importantly, using dpest enables researchers to script the entire calibration process, thereby enhancing the efficiency of the workflow to scale easily across multiple varieties.

Statement of Need

With the adoption of new data collection technologies in agriculture, plant researchers have been developing methodologies to integrate remote sensing data and crop models to simulate crop performance and development (Kasampalis et al., 2018). Remote sensing data have many advantages over traditional data collection methods in crop science. For instance, they can be collected repeatedly over time without disturbing the plants and at a low cost. Using these kinds of time-series data can improve the accuracy of crop model calibration. However, the calibration tools included in the DSSAT installation do not support the use of time-series data, which limits the potential of remote sensing for model calibration. Researchers can use PEST-based calibrations for the DSSAT models to address this limitation, but manual preparation of PEST input files requires expertise in both the PEST software and DSSAT models, which makes it a complex and time-consuming task. Previous efforts to streamline this process include an R-based DSSAT-PEST script distributed as supplementary material with the paper (Ma et al., 2020), which automates PEST file generation but requires manual setup of configuration files. While libraries like pyEMU (White et al., 2016) enable general-purpose

PEST control file construction, they rely on users to provide all model parameter definitions, bounds, groupings, and observation data inputs that, for DSSAT models, must be extracted and formatted from input and output files. The *dpest* Python library helps overcome these challenges, allowing crop researchers to integrate time-series data from either remote sensing or direct measurements to improve PEST-based DSSAT model calibration. Moreover, streamlining the calibration process using *dpest* can facilitate the application of DSSAT models to large populations of crop varieties. Finally, *dpest* includes model-agnostic utilities for targeted modification of PEST control files (.PST), reducing the need for full-file reconstruction and facilitating iterative calibration workflows.

Functionality

The *dpest* package includes the following modules, each with detailed usage instructions and examples available in the [documentation](#):

- *cul()*: Creates PEST template files (.TPL) for DSSAT cultivar parameters. The file is used for cultivar calibration.
- *eco()*: Creates PEST template files (.TPL) for DSSAT ecotype parameters. The file is used for ecotype calibration.
- *overview()*: Creates PEST instruction files (.INS) for reading observed (measured) values of key end-of-season crop performance metrics and key phenological observations from the OVERVIEW.OUT file. The instruction file tells PEST how to extract model-generated observations from the OVERVIEW.OUT file, compare them with the observations from the DSSAT A file, and adjust model parameters.
- *plantgro()*: Creates PEST instruction files (.INS) for reading simulated plant growth values from the DSSAT PlantGro.OUT file. The .INS file guides PEST in comparing those simulated values with the time-series data measured and provided in the DSSAT T file.
- *pst()*: Generates the main PEST control file (.PST) to guide the entire calibration process. It integrates the template (.TPL) and instruction (.INS) files, defines calibration parameters, observation groups, weights, PEST control variables and model run command.
- *uplantgro()*: modifies the DSSAT output file (PlantGro.OUT) to prevent PEST errors when simulated crop maturity occurs before the final measured observation. This ensures PEST can compare all available time-series data, even when the model predicts maturity earlier than observed in the field.
- *utils*: Provides a set of functions for updating target variables on PEST control file (.PST) without regenerating the full file. This preserves the existing adjustments made in the file. The *utils* functions can be used with any model supported by PEST.

Use Cases and Applications

dpest has been used to calibrate DSSAT using data collected from a research experiment carried out at the International Maize and Wheat Improvement Center (CIMMYT) facilities, where 14 wheat varieties were grown in three different environmental conditions (irrigation, heat and draught) over two growing seasons. It has enabled the integration of remote sensing data and other time-series measurements to calibrate the DSSAT CERES wheat model ([Vargas-Rojas et al., 2024](#)).

Acknowledgements

References

Doherty, J. (2015). *Calibration and uncertainty analysis for complex environmental models*. Watermark Numerical Computing. ISBN: 978-0-9943786-0-6

Jones, J. W., Hoogenboom, G., Porter, C. H., Boote, K. J., Batchelor, W. D., Hunt, L. A., Wilkens, P. W., Singh, U., Gijsman, A. J., & Ritchie, J. T. (2003). The DSSAT cropping system model. *European Journal of Agronomy*, *18*(3), 235–265. https://doi.org/10.1016/S1161-0301(02)00107-7

Kasampalis, D. A., Alexandridis, T. K., Deva, C., Challinor, A., Moshou, D., & Zalidis, G. (2018). Contribution of remote sensing on crop models: A review. *Journal of Imaging*, *4*(4). https://doi.org/10.3390/jimaging4040052

Ma, H., Malone, R. W., Jiang, T., Yao, N., Chen, S., Song, L., Feng, H., Yu, Q., & He, J. (2020). Estimating crop genetic parameters for DSSAT with modified PEST software. *European Journal of Agronomy*, *115*, 126017. https://doi.org/10.1016/j.eja.2020.126017

Vargas-Rojas, L., Wang, D., & Reynolds, M. (2024). How can we streamline wheat model calibration with remote sensing? In *ASA, CSSA, SSSA International Annual Meeting*. https://scisoc.confex.com/scisoc/2024am/meetingapp.cgi/Paper/161215

White, J. T., Fienen, M. N., & Doherty, J. E. (2016). A python framework for environmental model uncertainty analysis. *Environmental Modelling & Software*, *85*, 217–228. https://doi.org/10.1016/j.envsoft.2016.08.017