SoftwareX 19 (2022) 101208

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

pyfao56: FAO-56 evapotranspiration in Python

Kelly R. Thorp

USDA-ARS, Arid Land Agricultural Research Center, 21881 N Cardon Ln., Maricopa, AZ 85138, United States of America

A R T I C L E   I N F O

Article history:
Received 10 February 2022
Received in revised form 10 May 2022
Accepted 12 September 2022

Keywords:
Agriculture
Crop
Evapotranspiration
Irrigation
Management
Water

A B S T R A C T

The pyfao56 software package is a Python-based implementation of (1) the American Society of Civil Engineers (ASCE) Standardized Reference Evapotranspiration Equation and (2) the Food and Agricultural Organization of the United Nations (FAO) Irrigation and Drainage Paper No. 56 (FAO-56) dual crop coefficient methodology. The software was initially developed to support crop water use estimation and irrigation scheduling for field research at the Maricopa Agricultural Center in Arizona. Recent efforts to generalize and modularize the software design have increased its applicability and relevance for broader scientific studies on crop evapotranspiration and irrigation management worldwide.

Code metadata

Current code version	*v1.0.9*
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-22-00038
Permanent link to reproducible capsule	*N/A*
Legal code license	*Creative Commons Zero (CC0)*
Code versioning system used	*git*
Software code languages, tools and services used	*python*
Compilation requirements, operating environments and dependencies	*Python: charset-normalizer-2.0.11, idna-3.3, numpy-1.21.5, pandas-1.3.5, python-dateutil-2.8.2, pytz-2021.3, requests-2.27.1, six-1.16.0, urllib3-1.26.8*
If available, link to developer documentation/manual	https://github.com/kthorp/pyfao56/blob/main/README.md
Support email for questions	kelly.thorp@usda.gov

1. Motivation and significance

Evapotranspiration (ET) is a hydrologic process involving the physical movement of water from land and plant surfaces (evaporation) and the biophysical movement of water through plant material (transpiration) to the atmosphere. Standardized calculations for ET estimation have resulted from many decades of scientific research, primarily related to water used for production of agricultural crops [1]. Field studies have focused on quantifying reference ET for two reference surfaces, including well-watered 0.12 m grass (i.e., the "short crop") and 0.47 m alfalfa (i.e., the "tall crop"). In 2005, the Task Committee on Standardization of Reference Evapotranspiration, a subcommittee within the Environmental and Water Resources Institute (EWRI) of the American Society of Civil Engineers (ASCE), published a document outlining equations for standardized computation of short crop reference ET (ET_{os}) and tall crop reference ET (ET_{rs}) [2]. Using this standardized approach, daily estimates of ET_{os} and ET_{rs} can be obtained from meteorological measurements, including daily incoming solar radiation, maximum and minimum air temperature, and daily average wind speed and dew point temperature. This methodology has now become a widely accepted approach for characterizing and comparing the evaporative demand of the atmosphere among different environments.

Due to the diversity of agricultural crops and cropping practices, the characteristics of the soil–plant–atmosphere interface can vary widely, leading to different amounts of crop ET as compared to ET from the two reference surfaces. To estimate daily ET for different crop types, the Food and Agriculture Organization of the United Nations (FAO) has published the Irrigation and Drainage Paper No. 56 (FAO-56), which details crop coefficient methodologies for adjusting reference ET to crop ET [3]. Specifically, the dual crop coefficient methodology partitions ET calculations into evaporation and transpiration components:

$$ET_a = (K_{cb}K_s + K_e)ET_{os} \tag{1}$$

E-mail address: kelly.thorp@usda.gov.

https://doi.org/10.1016/j.softx.2022.101208

where ET_a is the ET for the crop of interest, K_{cb} is the basal crop coefficient, K_s is the water stress coefficient, K_e is the evaporation coefficient, and ET_{os} is, again, the standardized short crop reference ET. FAO-56 tabulates data for computing trapezoidal K_{cb} time series for different crop types, which defines daily transpiration under non-stressed conditions. FAO-56 also defines equations for daily computation of K_e based on water depletion in the surface soil layer and K_s based on the soil water content and soil water holding properties of the root zone. The FAO-56 publication has been widely adopted for crop ET estimation and irrigation scheduling worldwide. According to Google Scholar, it has presently achieved more than 30,000 citations.

The ASCE Standardized Reference ET Equation and FAO-56 were prepared as textual documents to describe the equations and procedures for standardized ET calculations. Subsequent efforts have codified the procedures within software tools. For example, the Ref-ET software was developed to calculate ET_{os} and ET_{rs}, among other ET definitions [4]. Also, Hunsaker et al. described a spreadsheet tool that incorporated FAO-56 equations for cotton irrigation scheduling [5]. The FAO also provides software for calculating ET_{os} [6] and irrigation requirements [7]. While important and useful, these software tools are not completely open-source and are being made obsolete by advances in software development paradigms and software requirements for modern data science applications. The goal of the present software, named "pyfao56", was to codify the ASCE Standardized Reference ET Equation and the FAO-56 dual crop coefficient methodology in Python. Hosted on Github, conda-forge, and the Python Package Index (PyPI), pyfao56 will make the standardized ET algorithms available to a new generation of scientists with aspiration for modern software accessibility. Furthermore, the Python routines will permit more diverse uses of the standardized ET methods, from irrigation scheduling apps on desktops or smartphones to more advanced ET computations on high-performance computers. Recent examples of pyfao56 implementation for crop ET estimation and irrigation scheduling in agricultural field research are provided by Thorp et al. [8,9].

2. Software description

The pyfao56 package codifies the ASCE Standardized Reference ET Equation and the FAO-56 dual crop coefficient methodology in Python. The software was designed to facilitate access to these algorithms from the Python command line or from a user's customized Python script. Users can import the pyfao56 package to apply the FAO-56 methodology within their individual workflows. Efforts were taken to generalize and modularize the software design to promote broader software applicability and relevance.

2.1. Software architecture

As shown in the Unified Modeling Language (UML) class diagram (Fig. 1), the pyfao56 package uses an object-oriented software design to modularize the components of the FAO-56 dual crop coefficient methodology. Classes are established for handling and storing data related to model parameterization ("Parameters"), irrigation schedules ("Irrigation"), and daily weather ("Weather"). A separate class ("Model") is used for implementation of FAO-56 equations and storage of output data. Within the Model class, the "ModelState" class stores all information for defining the model state at a given timestep, and an internal method ("_advance") implements the equations for advancing the model state by one daily time step. A utility class ("refet") provides functionality for computing daily ASCE Standardized Reference ET based on weather input data. Alternatively, users can provide their own reference ET data through the Weather class. Another class ("Update") is used for providing data to update model state variables. Current functionality permits state updating of the basal crop coefficient (K_{cb}), fractional crop cover (f_c), and rooting depth (Z_r) via direct insertion, where model states are simply overwritten with the provided data for a given timestep. For classes that store time series data, a DataFrame object based on Python's "pandas" package is used, and row indices are defined from a text string that incorporates the year and day of year. Data can be input or output from the data class objects using their "loadfile" and "savefile" methods to read and write data to and from text files having specific format specifications. However, use of text files is not required for data input. Required model parameter data and irrigation schedules can also be incorporated into user scripts, and the Weather and Update classes provide "customload" methods that users can override with customized approaches for populating the data frames. Users can create custom classes that access the "customload" method through inheritance with the Weather or Update classes. For example, in the "custom" sub-package of pyfao56, the "AzmetMaricopa" class demonstrates an approach for populating the weather data frame by downloading data from the website for the Arizona Meteorological Network (AZMET; https://ag.arizona.edu/azmet/) station in Maricopa, Arizona. Furthermore, the "Forecast" class demonstrates an approach to augment historical weather data from AZMET with weather forecast information from the National Digital Forecast Database (https://graphical.weather.gov/xml/rest.php). This approach is most useful for irrigation scheduling applications that require estimates of future weather. Users can follow these examples to customize use of the pyfao56 package for specific applications at their own locale, and users are welcome to work with the developer to incorporate their approaches into the "custom" subpackage.

2.2. Software functionalities

As demonstrated in the documentation and in the examples provided in the test suite, the functionality of the software can be realized with a few simple commands. After importing the package to their Python environment, users must instantiate instances of the Parameters, Weather, Irrigation, and optionally the Update classes. Multiple instances can be created if needed. For example, a field study with well-watered and water-limited irrigation treatments would require two instances of the Irrigation class to represent the two irrigation schedules. Users must then populate the class objects with appropriate data. The simplest way is to use the "loadfile" methods to read data from text files, but more advanced users have other options as discussed previously. To run the dual crop coefficient model, users must instantiate the Model class, providing inputs to define the starting and ending dates for the simulation and passing the instances of the Parameters, Weather, Irrigation, and optionally the Update classes. Calling the "run" method will cause the software to iterate through the daily computations of crop water use and soil water balance while populating the "odata" data frame with model output data. The output data can subsequently be printed or written to a file. The definitions for model input and output data are fully commented in the code. Most often, the starting date for the simulation can be specified as the crop planting date. Also, the ending date can be specified at either a past or future time, although the latter presumes the user has an approach to estimate future weather.

3. Illustrative examples

The pyfao56 test suite provides a script named "refet_testB.py" which computes ET_{os} and ET_{rs} using 18 years of daily

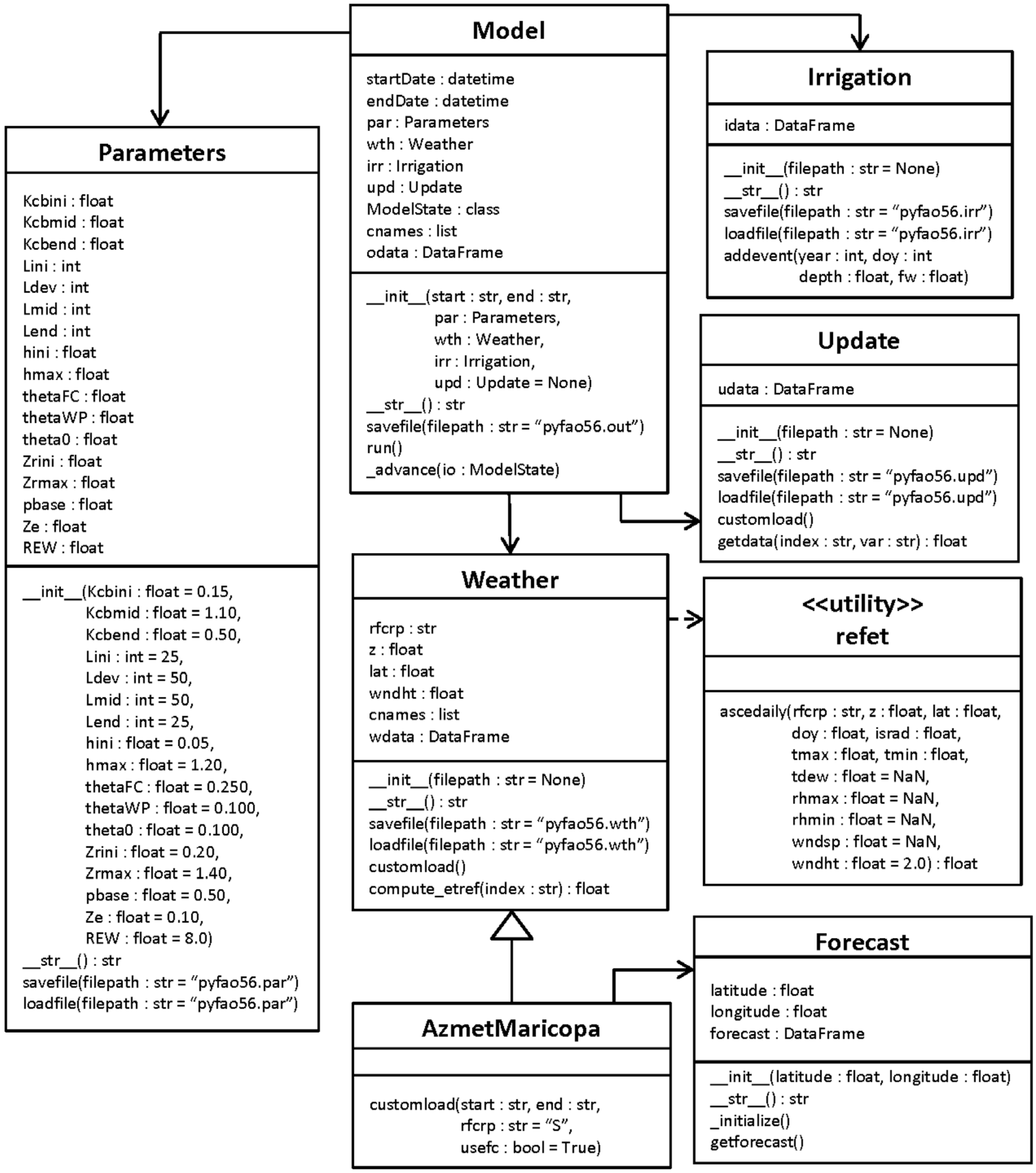

Fig. 1. A Unified Modeling Language (UML) class diagram for the object-oriented "pyfao56" software package.

meteorological data (2003 through 2020) from the AZMET station at Maricopa, Arizona. The same weather data was imported to Ref-ET software [4] for separate estimates of ET_{os} and ET_{rs}. Results indicated that the root mean squared differences (RMSD) between ET_{os} and ET_{rs} as computed by the two software programs were 0.070 and 0.071 mm, respectively (Fig. 2). Thus, pyfao56 can compute standardized reference ET to within a tenth of a mm as compared to Ref-ET software, the latter of which is a known authoritative source for accurate reference ET calculations. Because Ref-ET software is closed source, it was not possible to assess the specific reasons for the small deviations between the two software tools.

Another script within the pyfao56 test suite, named "cotton2018.py", sets up and runs two treatments (well-watered and water-limited) from a 2018 cotton experiment at Maricopa, Arizona [10]. In this study, the well-watered and water-limited treatments received 1282 and 856 mm of irrigation, respectively, between 1 March 2018 (day of year (DOY) 121) and 31 October 2018 (DOY 304). Rainfall and ET_{os} during this period amounted to 179 and 1617 mm, respectively. Cotton was planted on 18 April (DOY 108) and harvested on 30 October (DOY 303). Uniform, biweekly irrigation for crop emergence was initiated on 20 April (DOY 110). Irrigation treatments were initiated on 30 May (DOY 150), and irrigation was terminated on 7 September (DOY 250). Weekly measurements of soil water content with a nuclear gauge were used with a soil water balance methodology to estimate ET_a during the field study. Measured and pyfao56-simulated ET_a from 4 May (DOY 124) to 23 September (DOY 266) were 732 and 770 mm for the water-limited treatment (+5.3% error) and 1072 and 1011 mm for the well-watered treatment (−5.6% error), respectively. The pyfao56 calculations of K_s, the stress-adjusted crop coefficient ($K_c = K_{cb} \times K_s + K_e$ from Eq. (1)) and the root-zone soil water depletion (D_r) demonstrated the model's responses to the input data for these irrigation treatments (Fig. 3). As expected from FAO-56 methodology, early-season K_c spikes due to irrigation events were appropriately computed. Also, the K_c and D_r for the well-watered and water-limited treatments began to diverge after initiating the irrigation treatments on DOY 150. Throughout the summer (until DOY 225), K_c for the well-watered treatment showed little signs of water stress, and irrigation events reduced D_r to zero. However, K_c for the water-limited treatment indicated water stress effects, and irrigation amounts were not sufficient to reduce D_r to zero. After irrigation termination on DOY 250, D_r for both treatments continued to increase, except for responses to a few rainfall events in late September and early October (DOY 262–286). Overall, the K_s, K_c, and D_r time series demonstrated expected responses to watering events and to water limitation, which suggests that pyfao56 is correctly formulated according to the methodology described in FAO-56.

4. Impact

The pyfao56 software was originally conceived as an FAO-56 irrigation scheduling tool for agricultural research studies at the Maricopa Agricultural Center (MAC) in Arizona. Due to the usual lack of rainfall at the MAC, field research cannot proceed successfully without a strategy for irrigation management decisions. The tool continues to be used in this capacity. At least three field scientists at the MAC have integrated the software into their personal workflows to schedule irrigation for their field experiments.

In a recent study [9], pyfao56 was implemented as the core irrigation scheduling algorithm for a two-year cotton field study that tested different methodologies for conducting precision irrigation management. In the control treatment, pyfao56 was used as a stand-alone irrigation scheduling algorithm with standard trapezoidal crop coefficient data. For the other treatments, the software was combined with (1) site-specific soil water holding data, (2) basal crop coefficient and fractional crop cover data estimated from weekly images collected using a small unoccupied aircraft system (sUAS), and (3) commercial software for site-specific irrigation applications with an overhead lateral-move irrigation sprinkler (Zimmatic, Lindsay Corp., Omaha, Neb.). Demonstrating the interoperability of pyfao56 algorithms with several other irrigation management technologies was a notable outcome of this study.

The software was also used to quantify water productivity for a cotton field trial that tested eight cotton cultivars, two planting dates, and four irrigation rates [8]. Basal crop coefficients and fractional crop cover were estimated from multispectral images collected via an sUAS. These data were incorporated with pyfao56 to estimate seasonal crop water use among experimental treatments. This study was foundational for development of the pyfao56 state variable updating methodology based on inputs from remote sensing data.

The FAO-56 dual crop coefficient methodology has commonly been implemented through incorporation of the model equations into spreadsheet software, such as Microsoft Excel. While this approach has worked well for basic irrigation scheduling computations, it has limited flexibility for activities that require multiple iterative FAO-56 calculations, use of FAO-56 methods with Linux operating systems (which is common for most high-performance computers), and integration of FAO-56 methods with other technologies, software, or data. Python has now become a widely popular programming language, particularly for modern data science and machine learning applications. The pyfao56 package

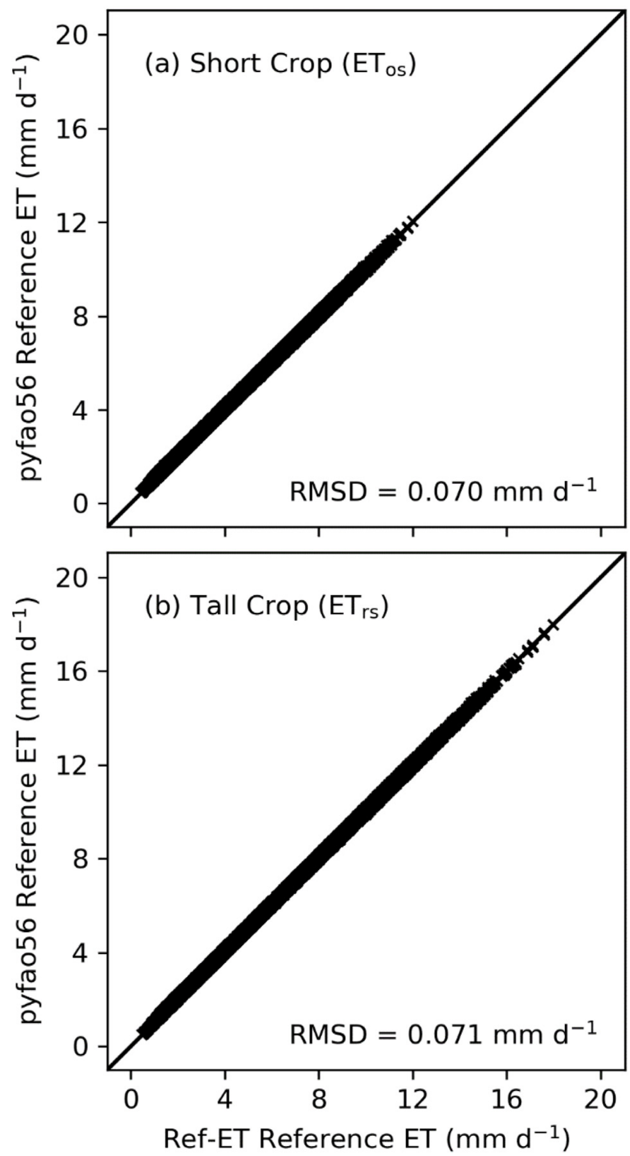

Fig. 2. Comparisons of (a) short crop reference evapotranspiration (ET_{os}) and (b) tall crop reference evapotranspiration (ET_{rs}) from pyfao56 and Ref-ET software, based on 2003–2020 daily meteorological measurements from the Arizona Meteorological Network (AZMET) station at Maricopa. The pyfao56 methodology can compute ET_{os} and ET_{rs} with root mean squared differences (RMSD) less than a tenth of a mm, as compared to Ref-ET software.

modernizes the implementation of FAO-56 using Python and makes FAO-56 methods available and accessible to a wider scientific audience. While pyfao56 has been utilized mainly for local studies at Maricopa, it is new generalized and modular design now gives it potential for much broader application.

5. Conclusions

The pyfao56 software package is a Python-based implementation of the ASCE Standardized Reference ET Equation and the FAO-56 dual crop coefficient methodology. It explicitly follows the methodologies described in prominent ET literature [2,3]. The software has been under development since 2016 and used primarily to estimate ET and schedule irrigation for field studies at the MAC in central Arizona. There are now many examples of pyfao56 testing and implementation for conditions at the MAC.

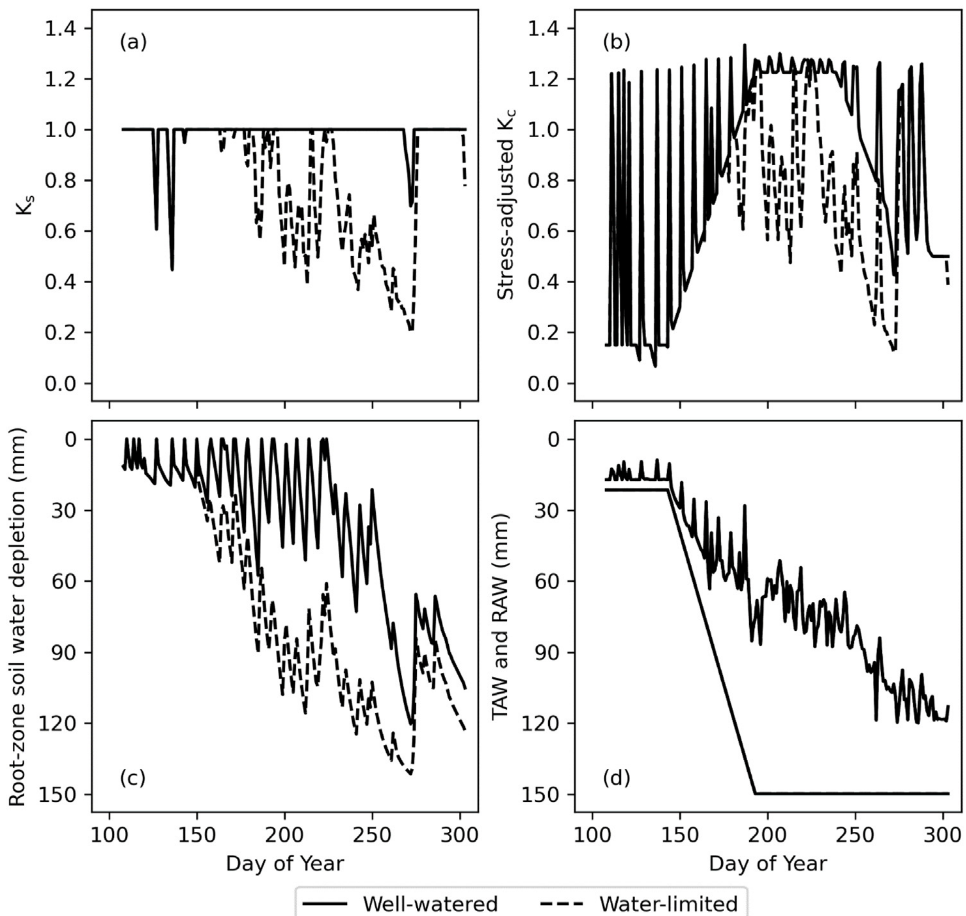

Fig. 3. A subset of pyfao56 simulation output for a 2018 cotton study with well-watered and water-limited irrigation treatments at Maricopa, Arizona. Daily time series outputs are provided for (a) the water stress coefficient (K_s), (b) the stress-adjusted crop coefficient ($K_c = K_{cb} \times K_s + K_e$), (c) the root-zone soil water depletion, and (d) the total available water (TAW) and readily available water (RAW) in the root zone, which were identical among irrigation treatments.

Recent efforts have focused on generalizing and modularizing the software design, so it can be more applicable and relevant to a wider audience. The software was thus formulated as a Python package and released on Github, conda-forge, and PyPI. The developer welcomes collaborations to further develop and test the software for diverse agricultural research activities.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Kelly R. Thorp reports financial support was provided by Cotton Inc.

Acknowledgments

The author acknowledges the scientists who have devoted their careers to development and demonstration of the FAO-56 methodology for crop ET estimation and irrigation scheduling, including Rick Allen and Doug Hunsaker among others. The research was funded by the USDA Agricultural Research Service (2020-13660-008-000-D) and Cotton Incorporated (17-642).

References

[1] ASCE task committee on revision of manual 70 (Jensen, M. E., Allen, R. G., Howell, T. A., Martin, D. L., Snyder, R., Walter, I. A). In: Evaporation, evapotranspiration, and irrigation water requirements (asce manuals and reports on engineering practice no. 70). 2nd ed.. Reston, VA: American Society of Civil Engineers; 2016.

[2] ASCE task committee on standardization of reference evapotranspiration (Walter, I. A., Allen, R. G., Elliott, R., Itenfisu, D., Brown, P., Jensen, M. E., Mecham, B., Howell, T. A., Snyder, R., Eching, S., Spofford, T., Hattendorf, M., Martin, D., Cuenca, R. H., Wright, J. L). In: The ASCE standardized reference evapotranspiration equation. Reston, VA: Environmental and Water Resources Institute of the American Society of Civil Engineers; 2005.

[3] Allen RG, Pereira LS, Raes D, Smith M. Crop evapotranspiration: guidelines for computing crop water requirements (fao irrigation and drainage paper no. 56). Rome, Italy: Food and Agricultural Organization of the United Nations; 1998.

[4] Allen RG. Ref-ET: reference evapotranspiration standardized calculators, version 3.1.16. Moscow, ID: University of Idaho; 2012.

[5] Hunsaker DJ, Barnes EM, Clarke TR, Fitzgerald GJ, Pinter Jr PJ. Cotton irrigation scheduling using remotely sensed and FAO-56 basal crop coefficients. Trans ASAE 2005;48(4):1395–407. http://dx.doi.org/10.13031/2013.19197.

[6] FAO. ET_o *calculator, version 3.1*. 2009, Available at https://www.fao.org/land-water/databases-and-software/eto-calculator/en/ (Accessed 2 February 2022).

[7] Smith M. CROPWAT: A computer program for irrigation planning and management. In: (FAO irrigation and drainage paper no. 46). Rome, Italy: Food and Agricultural Organization of the United Nations; 1992.

[8] Thorp KR, Thompson AL, Harders SJ, French AN, Ward RW. High-throughput phenotyping of crop water use efficiency via multispectral drone imagery and a daily soil water balance model. Remote Sens 2018;10(1682). http://dx.doi.org/10.3390/rs10111682.

[9] Thorp KR, Calleja S, Pauli D, Thompson AL, Elshikha DE. Agronomic outcomes of precision irrigation management technologies with varying complexity. J ASABE 2022;65(1):135–50. http://dx.doi.org/10.13031/ja.14950.

[10] Thorp KR, Thompson AL, Bronson KF. Irrigation rate and timing effects on arizona cotton yield, water productivity, and fiber quality. Agricult Water Manag 2020;234:106146. http://dx.doi.org/10.1016/j.agwat.2020.106146.