



CLASSIM: A relational database driven crop model interface



Dennis Timlin^{a,*}, David Fleisher^a, Maura Tokay^{a,b}, Kirsten Paff^a, Wenguang Sun^c, Sahila Beegum^c, Sanai Li^d, Zhuangji Wang^e, Vangimalla Reddy^a

^a Adaptive Cropping Systems Laboratory, USDA-ARS, Bldg 001, Rm 342, 10300 Baltimore Ave, Beltsville, MD 20705, USA

^b Science Systems and Applications, Inc., SSAI, USA

^c University of Nebraska, Lincoln, NE, USA

^d U.S. Forest Service, USA

^e University of Maryland, College Park, MD, USA

ARTICLE INFO

Editor: Stephen Symons

Keywords:

GUI
Graphical user interface
Python
Crop
Simulation

ABSTRACT

Crop models are valuable tools for examining the interactions of cultivar characteristics, environment, and management practices, and how they affect crop growth and development. The difficulty in finding all the data needed to set up a simulation can often deter potential users from utilizing a crop model. Model interfaces are necessary to make these complex tools accessible to end-users who may lack the expertise needed to work with the models directly, but who would benefit from the information generated by the models. As crop models vary in terms of input and output structures, there is no one universally compatible interface, so different crop model suites require their own interface. CLASSIM is a graphical user interface (GUI) for a suite of models developed by USDA-ARS. The interface provides a simplified tool to assemble input data for a model simulation. Web access to online databases for soil and weather data simplifies finding site-related data. The user is guided step by step to build the necessary input files. A database structure facilitates data entry, storage, and retrieval. The use of Structured Query Language (SQL) for data retrieval allows for advanced analysis and visualization outside of the interface. We describe the features of the GUI and provide examples of various operations. Model outputs are provided in tables and graphs to allow for visualization and analysis. Two-dimensional contour plots of soil processes are provided to visualize output from the two-dimensional finite element soil model. CLASSIM also allows users to set up single- and multiple-season runs for maize, cotton, soybean, potato, and fallow treatments, and is expected to be expanded to more crops in the future. A user-friendly GUI is essential for making simulation tools accessible to the broad range of researchers and practitioners working to solve the agricultural system challenges.

1. Introduction

Crop models are powerful tools that can be used to simulate crop development, water and nutrient use, yields, and environmental impacts under varying growing conditions, management practices, and cultivar characteristics. These models allow users to perform system level analyses of the effects of climate change and management practices on agriculture [2,5]. As crop models become more sophisticated, they also require an increasingly detailed level of expertise to use directly, making them cumbersome for non-modelers to use and reducing their potential as decision support tools or for usage in interdisciplinary research [20, 32]. Therefore, easy to use graphical user interfaces (GUIs) are needed to make crop models accessible to a broader audience.

There are four major benefits to using GUIs: (1) Simpler data input, (2) Reduced need to manage or modify model source code, (3) Improved data management, and (4) Built in data analysis tools. Most crop models require input data to be in a specific format, but the required units and format are not always clear to a novice user. Finding all the required input data and formating the input files can also be cumbersome. In a GUI, the user is walked through the data input process and is given clear written instructions, rather than having to interpret often confusing variable names. Since a GUI prevents the user from directly interacting with the source code and often has built in error checking mechanisms that prevent the user from inputting out of bounds data, the risk of users accidentally tampering with the model is decreased.

Many GUIs include tools for graphing output data or running

* Corresponding author.

E-mail address: Dennis.Timlin@usda.gov (D. Timlin).

statistics, so the user does not have to write separate code for processing outputs [3,32]. These are useful for simple views and analysis of the simulation results. Most projects that use models have research objectives, however, so the user may require detailed analysis techniques specific to the project. Crop models produce varied and large data sets which often require external tools for more complicated analysis as it is difficult to create a GUI to handle all analysis methods. For spatial analysis, the outputs can be ported to applications such as a Geographic Information System (GIS). In addition, crop model specific script-based tools using the R-System can be used to manage and analyze output data, such as those developed for the Decision Support System for Agrotechnology Transfer (DSSAT) [14] models [3]. Alderman [3] also provides an in-depth discussion of other scripting approaches to managing data for the DSSAT models. Another potential external analysis tool is through a Microsoft Excel based interface [22] as was developed for the APEX model [34].

There are several existing GUIs for crop models and crop modeling platforms, including the Graphical User Interface for Crop Simulations (GUICS) [1], DSSAT [13], and the Agricultural Production Systems Simulator (APSIM) [12]. GUICS is a generic user interface that was not created for any specific crop model or modeling platform. However, crop model scripts must be written so that input data for new models can be managed by GUICS. This requires some programming experience. GUICS also has historically needed to be expanded upon whenever a model had new formating types [7]. This limits GUICS useability for non-programmers and novice modelers if new models need to be added. DSSAT is a modeling platform and GUI that supports over 42 crop models within the DSSAT family. DSSAT has a modular structure and all DSSAT crop models are linked to the same one-dimensional capacitance-based soil model [13]. While the DSSAT GUI works well for DSSAT models, it is not compatible with models outside of the platform. The necessary data files are stored in local folder structures. The large number of input files needed for DSSAT and storage method complicates file management. APSIM stores model outputs and observed data in databases that can be queried and filtered using Structured Query Language (SQL). Much like DSSAT, the APSIM interface is designed for use with APSIM models only [12]. While both APSIM and DSSAT have example datasets available, they do not have built in tools to access online weather and soil databases, so if users want to run simulations at a new location, they also need to independently find and input all the corresponding environmental data.

Many existing crop model GUIs utilize ASCII files stored in a hierarchy on a user's file system. Both input and output files are stored using a flat file structure, often using multiple files for a simulation. The user needs to manually edit the various input files for soil, weather, management, etc. as text files or write a separate program to generate the files. Because there is often no relational structure in a hierarchical file system, sorting and searching all the data by desired properties or values is difficult. The use of a relational database is an alternative and more efficient approach to manage hierarchical data. Using a relational database for data output, for example, can also facilitate sharing of simulation output. One possible reason databases are not used more often in crop models and interfaces is that many of these models are written in FORTRAN, which does not have libraries for database connections. There have been previous efforts to develop relational databases that can be used with agricultural models. The AgroDB database, for example, has been interfaced with the DSSAT family of models [21]. Also, the APSIM interface [12] utilizes a relational database.

Crop simulation models developed by different groups need a GUI with data management capabilities to be usable for a wider audience than just developers and scientists with programming experience. This includes models developed by the United States Department of Agriculture (USDA) Agricultural Research Service's (ARS) Adaptive Cropping Systems Laboratory (ACSL). We developed an interface called Crop, Land, and Soil Simulation (CLASSIM) as a GUI for the crop models developed by the ACSL. The crop models currently supported by

Table 1

Tab names and their functions with respect to model input and output data.

Tab Name	Function
Input	
Welcome	Informational screen
Site	Planting site data management
Soil	Soil data management
Weather	Weather data management
Cultivar	Cultivar data management
Management	Create and edit management scenarios
Seasonal Run	Simulate the crop for one season. Allows user to select site, weather, soil and management data to carry out a simulation for one growing season
Rotation Builder	Simulate a rotation over one or more years. Allows user to input similar information as in Seasonal Run, but also can select multiple crops along with fallow
Output	
Seasonal Output	Graphs and tables of results from single season runs
Rotation Output	Graphs and tables of results from rotation (multiple season) runs

CLASSIM are MAIZSIM [18] for maize, SPUDSIM [8,9,11] for potato, GOSSYM [25] for cotton, and GLYCIM [28] for soybean. CLASSIM also supports the 2DSOIL model [29], a two-dimensional modular finite element soil process model, which is coupled with all of the crop models and can be run independently to simulate fallow periods. The interface is not intended for large scale simulations that require hundreds or even thousands of runs, such as geospatial crop modeling studies, but can be used to evaluate single- or multi-year rotation studies at single field sites. The CLASSIM interface is designed to make the ACSL's crop models accessible to novice and intermediate level crop modelers and students who wish to apply our models for their own research. A further goal of this project was to develop an interface that has the capability to use online sources for soil and weather data to minimize the work needed to assemble data for crop simulations. The objective of this paper is to describe the operation of the CLASSIM interface and how it interacts with the user and database to build input files for simulations. We will provide several examples of this interaction along with model output.

2. Interface structure

The CLASSIM interface is written in Python version 3.10. [4]. Python was used for the interface because the various packages are available as user supported software. Furthermore, Python programs can be run on a wide range of operating systems without the need for system specific compilers or a purchased license. Python's use of high-level built-in data structures makes it ideal for managing large amounts of data. The user interface utilizes the PyQt [26] programming framework that is a set of Python modules for GUI programming and can call model executables written in FORTRAN and C++.

CLASSIM uses a 'tabbed' windows structure to organize access to the various model functions, which include input data entry and input/output data management. The code for each tab is encapsulated in a single python file. Tabs are used for both input and output of model information (Table 1). The tabs provide panes of content that can be organized into independent entities. Each tab provides an entry to informational screens that are discretely different from each other. Navigation between tabs can only be done by selecting the tabs themselves via user-initiated mouse clicks.

3. Data management: relational database approach

When the crop models supported by CLASSIM are run independent of an interface, the user needs to manually generate the various input files for soil, weather, management, etc. as text files. CLASSIM generates these files for the user, simplifying this process and minimizing the chances of user error. The files are only generated and saved to a storage

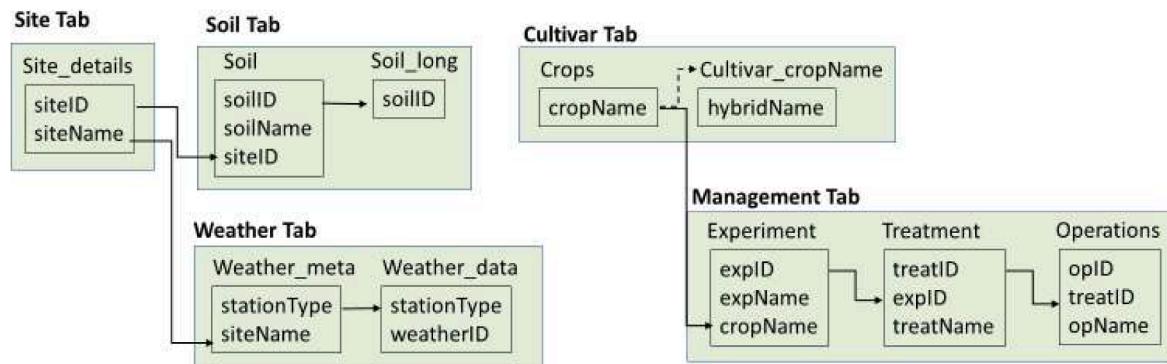


Fig. 1. Simplified structure of the main tables in the relational database used for managing input data and their relationships. The categories of the data are in bold and correspond to the tab in the interface. The table name *Cultivar_cropName* is a generalization, as each crop has its own table, e.g., *Cultivar_Cotton* for the cotton cultivars.

medium when a simulation is executed in the interface, not as the selections are made in the dialog boxes. This minimizes the possibility of saving incomplete data. The input files are generated for each seasonal (or rotation) simulation of plant growth and yield and stored in a separate folder for each simulation. This allows for use outside of the interface if desired. All the input information for CLASSIM entered by a user in the dialog boxes, however, is stored in a relational database (Fig. 1), regardless of whether it has been used in a simulation. The outputs of the simulation runs are stored in a separate database. Unlike APSIM, CLASSIM does not store observed crop data, such as observed grain yields, in a database [12]. The databases are installed alongside the CLASSIM program which allows the inclusion of example data sets and completed simulations for users to familiarize themselves with the GUI and models. The soil and weather tables can also be updated from a

server using code that is included in the CLASSIM installation.

Fig. 1 shows the relationships among the main data tables in the CLASSIM database. The *Site_details* table is linked to the *Soil* table through the *siteID* field in the *Site_details* table and *siteID* field in the *Soil* table. The *Soil* table contains meta information for each soil. The details such as layers and soil texture are stored in the *Soil_long* table. The *soilID* field in the *Soil* table references the associated rows in the *Soil_long* table through the *soilID* field. The weather data are encapsulated in two tables called *Weather_meta* and *Weather_data*. The *Weather_meta* table contains additional available information related to the site associated with the weather data, such as relative humidity and wind. Solar radiation, air temperatures and rainfall are always required for the *Weather_data* table. The weather data can be stored as daily (max-min values for temperature and summation for solar radiation) or as hourly values. CLASSIM

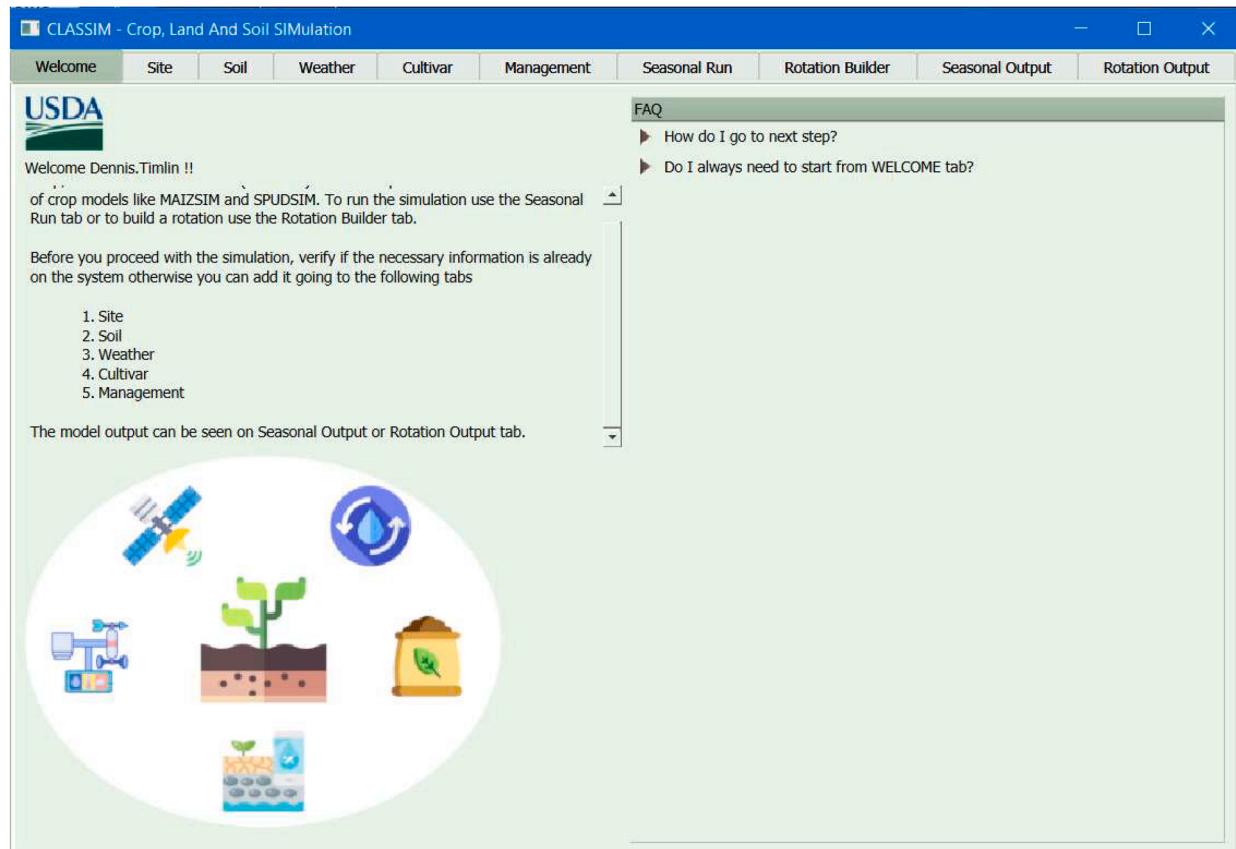


Fig. 2. CLASSIM Welcome Tab.

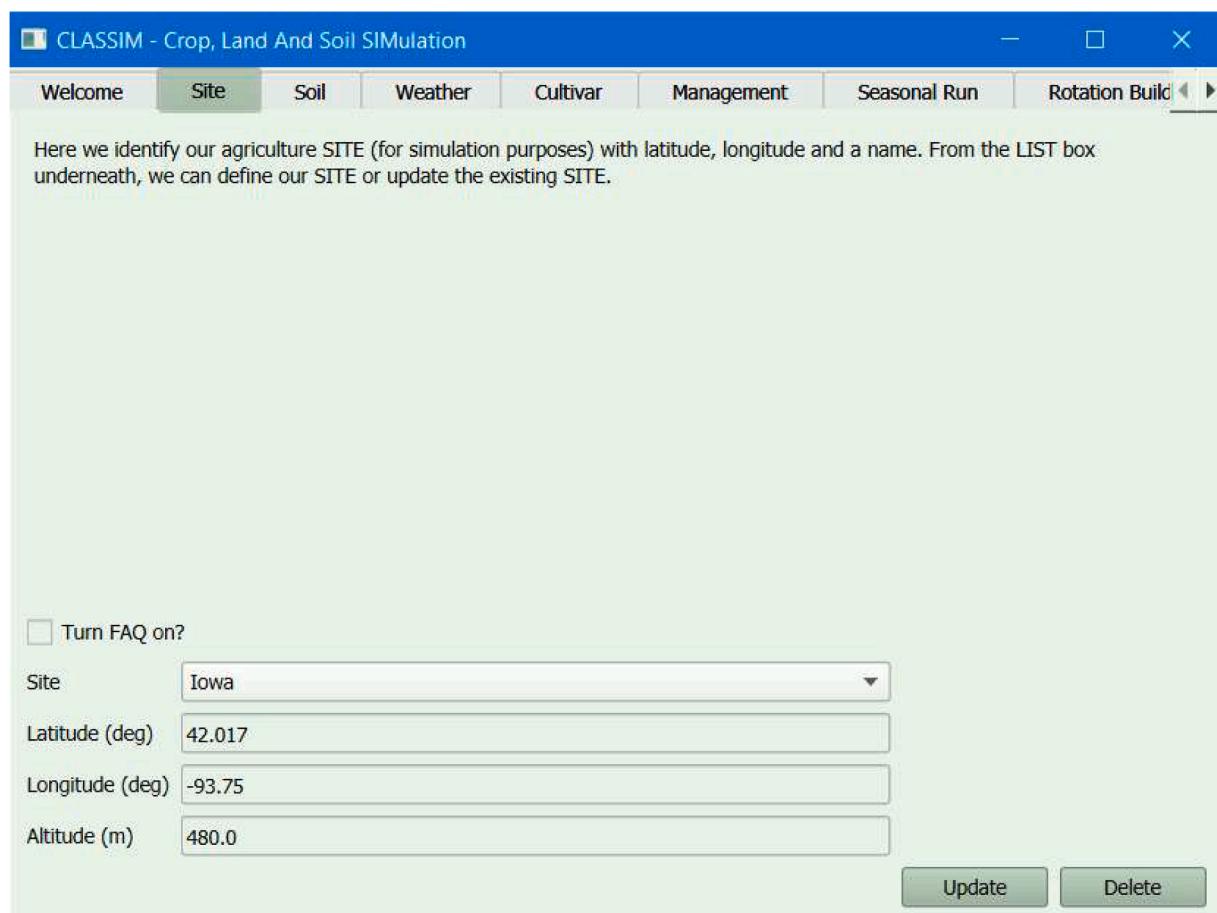


Fig. 3. Site tab screen. This is the dialog that is presented if the user selects to edit an existing site or add a new site. On this tab the user can input the site name, latitude, longitude, and elevation.

can detect if data are hourly or daily. The weather data are linked to the *Weather_meta* table through the *stationType* field. The *weatherID* field indicates different groups of weather data for the same site (*stationType*). For example, if one wants to have a simulation with additional rainfall but the same temperatures and radiation, a different *weatherID* would be assigned to the rows with the varied rainfall. The *cropName* field in the *Crops* table is used to link the crop name (i.e., maize, cotton, etc.) to the crop used in the experiment table and to the cultivar parameters in the *Cultivar_croname* table.

The tables used in the *Management* tab store details of a simulation that a grower would have control over. The highest-level grouping of management is an *Experiment*. An *Experiment* can be considered a set of related simulations for one crop species (i.e., maize, cotton, soybean, etc.) that vary with management. *Experiments* are further subdivided into *Treatments* that contain variations in the management options such as fertilizer applications, planting dates, row spacing, cultivar, etc. The *Experiments* are referenced by *expID* that provides the relationship between *Treatments* and *Experiments*. The various operations in *Treatment* (i.e., fertilizer application, planting date, etc.) are linked to the *Treatment* using the *treatID*. Names of the items are included to make it easier for the user to make selections in the various dialog boxes.

The benefit of the database approach is that the user only needs to enter the data once into the GUI. The information is stored in the database and can then be retrieved by the user or the program when the user decides to run a simulation. Input data reusability ensures that data retrieval is quick, and data are easy to manage using Structured Query Language (SQL). The database approach in CLASSIM allows users to easily manage information through the GUI and delete and modify the information. This reduces the likelihood of formating errors in input files

and encourages the use of the interface to organize and manage input data. This approach also facilitates accessing the information through structured database queries rather than searching and managing a large number of flat files in a hierarchical file system. Input validation is used throughout the interface which minimizes the risk that the user will input wrong or incomplete information.

The database approach also allows developers to easily share example inputs with users without having to share many individual input files. As the crop models used by CLASSIM are designed to have constant cultivar coefficients across locations, the database feature would be particularly helpful for sharing cultivar parameters. The one drawback to the database system is that, at this time, the database requires custom coding to prevent custom user-generated simulations from being deleted when the general database is updated by the developers. Users will be notified of database updates that will occur with new version of the program.

4. CLASSIM features

CLASSIM opens to a *Welcome* tab that provides basic introductory information and FAQs about the program and tabs (Fig. 2). All other aspects of CLASSIM are reached via the tabs at the top of the interface. All tabs have a box labeled "Turn FAQ on?" that the user can check if they wish for more detailed information about how to use that particular tab. An overview of the tabs is given in Table 1. The tabs are organized to guide the user through the sequence of actions needed to compile data for the models. The first tab is Site, which is the basic requirement of a crop simulation (Fig. 3). Site information includes latitude, longitude, and elevation. The latitude and longitude entered into the *Site* tab can be

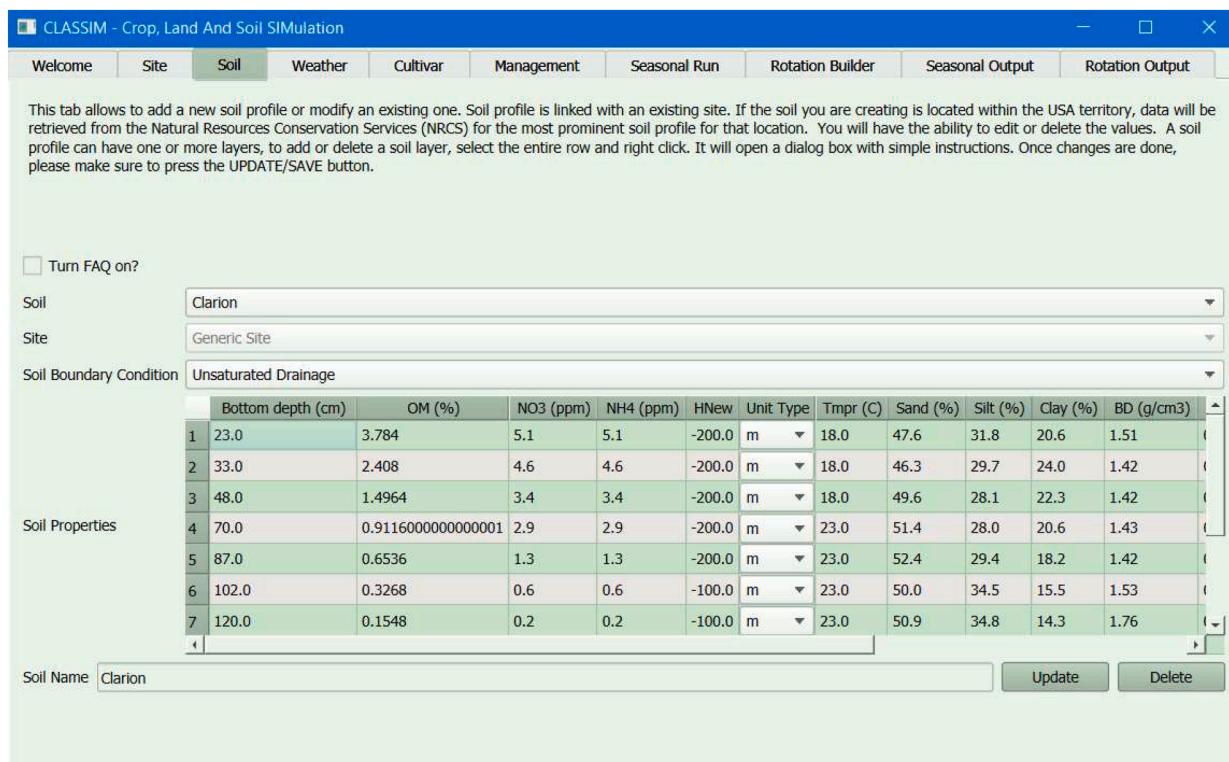


Fig. 4. CLASSIM Soil Tab. The FAQ is turned off for clarity. Because of space limitations columns for the van Genuchten parameters are truncated. Soil layers, bulk density and texture values for each layer can be manually entered or CLASSIM can pull existing data from the NRCS SSURGO database. Values for the drained upper limit (TH33), wilting point (TH1500) and van Genuchten parameters can be generated by a pedotransfer function if not known (when a value of -1 is given as input, not shown in the figure).

used by the interface to query online databases for both soil (Fig. 4) and weather data (Fig. 5) which are the next two input tabs. The user is next led to the *Cultivar* (Fig. 6) and *Management* (Fig. 7) tabs. The dialog boxes in the *Cultivar* tab allow the user to manage the parameters for the different crop varieties. The crops supported by the interface include soybean, maize, potato, and cotton. The user can add new cultivars or edit existing ones if necessary. The *Management* tab allows the user to enter management information for a particular simulation. The management information that can be added includes cultivar, row spacing, plant population, planting and harvesting dates, fertilization with nitrogen (N), seed depth and tillage. The hierarchy of *Experiments* and *Treatments* allows the user to organize simulations on a common theme. Once the user assembles the required data, the *Seasonal Run* and *Rotation Builder* tabs can be used to simulate either a single run or a rotation using input data gathered by the user. Finally, the results are available in the *Seasonal Output* and *Rotation Output* tabs.

5. Demonstration of the interface

5.1. Inputs

We will be using example datasets and simulations to help illustrate the functions of CLASSIM with respect to data input. The *Welcome* tab is shown in Fig. 2 and is the first screen the user sees after launching the program.

The *Site* tab (Fig. 3) allows users to input new field locations or view or alter existing ones. When the user selects “Add New Site” from the drop-down menu (not shown) they must enter the latitude, longitude, elevation (m), and site name. Latitude and longitude must be put in as decimal values. The site used in this example (Fig. 3) is near Colfax, Iowa, USA (42.017 N, -93.75 E, elevation 480 m) The field data from this site have previously been used for an Agricultural Model

Intercomparison and Improvement Project (AgMIP) study [19]. The site name entered in CLASSIM is Iowa.

The *Soil* tab (Fig. 4) is used to create new soils or view or alter existing ones. If the user selects “Add New Soil”, they are first asked to provide the site name from the drop-down menu of previously defined sites. Once a site has been selected new options appear. The user must select the soil boundary conditions, which define the interface at the bottom of the soil profile (e.g. water table, seepage, or unsaturated drainage), and type in a soil name. The following information is required for each soil layer: bottom depth (cm), soil organic matter (%), soil nitrate (ppm), soil ammonium (ppm), initial water content, unit type, which indicates if the initial water content is expressed as soil matric potential (-cm) or water content ($\text{cm}^3 \text{cm}^{-3}$), soil temperature ($^{\circ}\text{C}$), sand (%), silt (%), clay (%), bulk density (g cm^{-3}), drained upper limit of water availability ($\text{cm}^3 \text{cm}^{-3}$), wilting point ($\text{cm}^3 \text{cm}^{-3}$), and the standard van Genuchten [31] water retention and hydraulic conductivity curve parameters. These are the residual water and saturated water contents (thr, ths, $\text{cm}^3 \text{cm}^{-3}$), slope parameters (Alfa, n), and saturated hydraulic conductivity (K_s, cm^{-d}). The other parameters are additional van Genuchten parameters that modify the water retention curve (tha, thm, and thk, $\text{cm}^3 \text{cm}^{-3}$) and hydraulic conductivity values (Kk) to minimize convergence problems in the water movement equations [33]. If the van Genuchten parameters, drained upper limit, or wilting point are unknown, they can be set to -1. This triggers CLASSIM to use the ROSETTA program to calculate the parameters based on soil texture [27]. New soil layers can be added by right clicking an existing soil layer and selecting “Duplicate this layer above” or “Duplicate this layer below”. CLASSIM can also pull soil data for site locations within the contiguous USA from the online Soil Survey Geographic Database (SSURGO) [24] using the latitude and longitude of the experimental site. In the case of multiple soils being listed for the same site, the interface automatically selects the most prominent soil. The soil shown in Fig. 4 is

weather_id	date		srad		wind		rh		rain		temperature	
	min	max	min	max	min	max	min	max	min	max	min	max
IowaN	2006-01-01	2013-12-31	0.0	4.59367	0.513381	50.9744	0.0	100.0	0.0	57.583300	-34.49	37.93
IowaR1	2006-01-01	2013-12-31	0.0	4.59367	0.513381	50.9744	0.0	100.0	0.0	46.066640	-34.49	37.93
IowaT1	2006-01-01	2013-12-31	0.0	4.59367	0.513381	50.9744	0.0	100.0	0.0	46.066640	-30.49	41.93
iowaR2	2006-01-01	2013-12-31	0.0	4.59367	0.513381	50.9744	0.0	100.0	0.0	55.279968	-30.49	41.93

Fig. 5. Weather tab. The user can choose a station name and then edit or add new information. If there are different permutations of the weather data, they will be identified by different values of weatherID.

a Clarion soil (mesic Typic Hapludoll). The data were obtained from the NRCS Soil Survey Geographic Database.

The *Weather* tab (Fig. 5) is used to input weather data. Weather data can be imported as a .csv file or CLASSIM can pull hourly weather data for US sites from the five years prior to the access date from the North American Land Data Assimilation System (NLDAS) [35] and Multi-Radar/Multi-Sensor System (MRMS) [37,38] online databases using the latitude and longitude. To create new weather records from a .csv file, the user needs to input the site name, average wind speed (km h^{-1}), average rain rate (cm h^{-1} or cm d^{-1}), average atmospheric carbon dioxide concentration (CO_2) (ppm), nitrogen (N) content in rainfall (kg ha^{-1}), and a climate station name. Multiple station names can be associated with the same site. This allows for varying CO_2 , average wind speed, average rain rate or N content in the rainfall for the same site. Multiple permutations for the same weather station (i.e., future weather along with historical weather or a variation of historical weather) at the same site can also be used (Fig. 5). Variation of historical data for example can include rows with the same date but different precipitation amounts. The input .csv file must contain an ID field named ‘WeatherID’, which includes unique names for each permutation. This allows for weather data from disjointed years or simulated future weather data to be available for the same location.

The weather data in the .csv file should include daily or hourly solar radiation (W m^{-2}), maximum and minimum temperature ($^{\circ}\text{C}$) for daily data, or hourly temperature ($^{\circ}\text{C}$), rainfall (mm), wind speed (km h^{-1}), and relative humidity (%). If wind speed and relative humidity data are unavailable, the missing data are estimated during model executions

using psychrometric relationships in the 2DSOIL model as per Timlin et al. [30]. ACSL crop models function on an hourly time-step, and thus hourly weather data are estimated from daily weather data using celestial mechanic calculations which are also conducted during model execution in 2DSOIL [30]. The interface will notify the user if there are missing records.

The *Cultivar* tab accounts for intra-specific genetic variation (Fig. 6). The user can input variety specific parameter values regarding the sensitivity of individual cultivars of each crop to climatic factors (e.g. temperature and photoperiod), and water and nitrogen deficiencies, as well as developmental parameters (e.g. stay-green values, thermal time relationships of specific developmental stages, etc.). The cultivar parameters differ by crop. The FAQ section provides explanations for each parameter as well as a range of suggested values. Depending on the parameter, the value can either be obtained directly from measured field data and information provided by breeders, or through calibration to fit the model to measured field data. The components that make up the models used in CLASSIM, such as leaf growth [18,23] and photosynthesis [10,16,17], have all been independently tested and evaluated, so the species wide parameters established in these studies are reasonably stable across location, time, and cultivar. Default values were previously obtained for several widely grown cultivars of each crop and incorporated in the database for any new CLASSIM installation.

The *Management* tab (Fig. 7) allows users to input fertilizer, irrigation, planting, harvesting, and tillage information. These are organized by crop in which *Experiments* are nested. Each *Experiment* may be named after a site for organizational purposes, but the management practices of

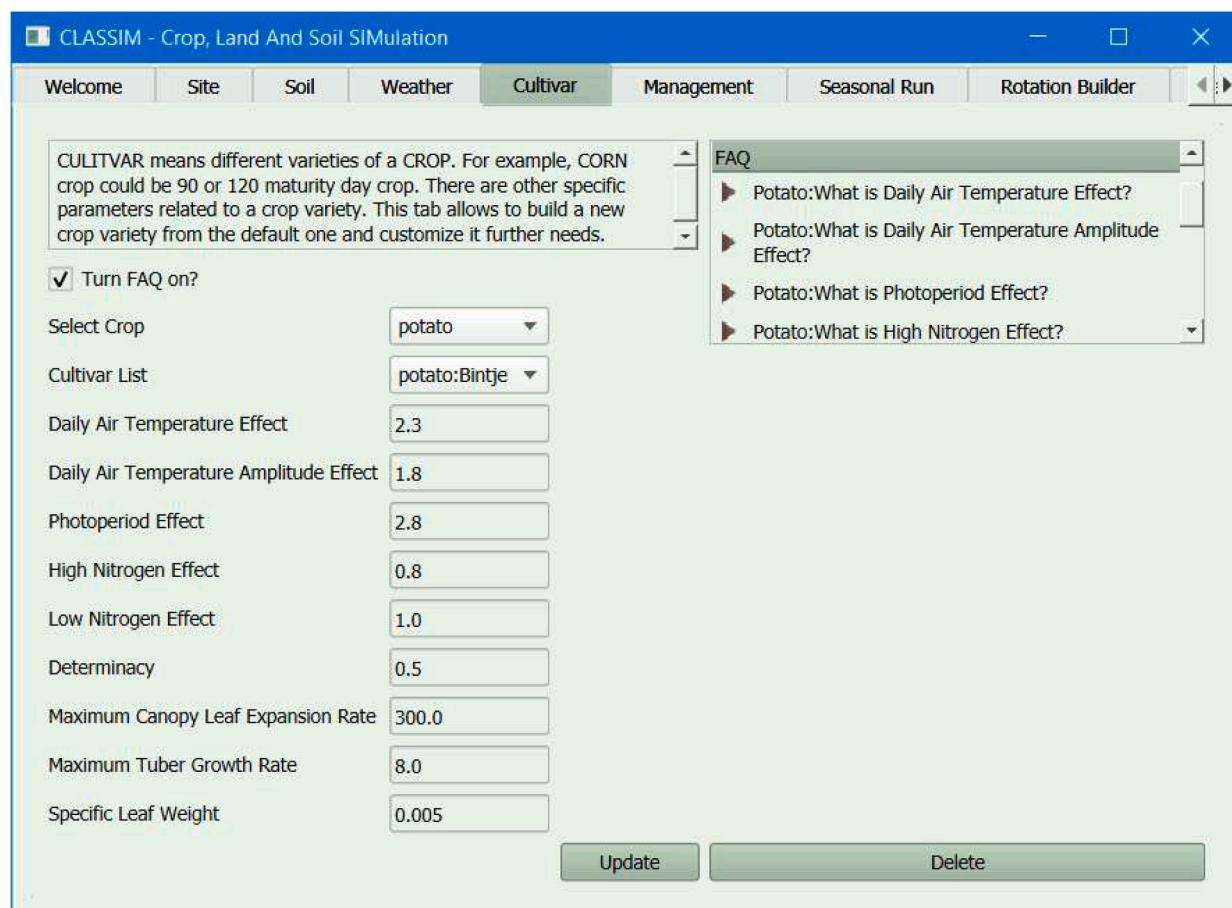


Fig. 6. Cultivar Tab allows the user to add or modify the properties of individual crop cultivars. This dialog box shows the properties of a potato cultivar called Bintje. The FAQ on the right provides information on the properties.

an *Experiment* can be applied to any site. There can also be multiple *Experiments* for the same site. An *Experiment* can have multiple treatments and to simplify using multiple treatments with only small differences, such as a fertilizer application rate study, CLASSIM has the *Copy To* button, which allows users to create a copy of an existing treatment and give it a new name. The tillage options are no tillage, moldboard plow, chisel plow, and vertical tillage. Once a tillage option is selected, the user can enter the tillage date. Under *Simulation Start*, the user can input the simulation start date (which can be set before or at sowing), the cultivar (from the list of cultivars defined in the *Cultivar* tab described above), the plant density (number of plants m^{-2}), the seed depth (cm), row spacing (cm), location of planting grid (which describes if the plant is at the left or middle of the two-dimensional simulated soil slab), and whether auto irrigation is turned on. Some crops, such as potato, have further inputs, such as emergence date and seed piece mass (g). The *Sowing*, *Emergence* (if used), and *Harvest* sections cover the dates of planting, crop emergence, and harvest (if the crop has not already matured), which ends the plant simulation. The *Simulation Start* and *Simulation End* dates start and terminate the simulation of associated soil processes. The *Simulation End* date is set for a date after the harvest. Since the soil model can simulate soil processes before planting and after harvest, it has its own start and termination dates. The *Fertilizer* section includes the date, depth of application (cm), fertilizer type (surface residue, synthetic N, litter, or manure), quantity of N applied ($kg\ N\ ha^{-1}$), and in the cases of litter and manure, the quantity of carbon applied ($kg\ C\ ha^{-1}$). Multiple fertilizer treatments can be added using the *Add New Treatment* option.

5.2. Simulation modes

CLASSIM has two modes of simulation, Seasonal Run and Rotation Builder. Seasonal Run is for simulating a single crop for a single growing season. Rotation Builder allows the user to grow different crops sequentially, though not simultaneously, as well as to add fallow periods between sequential crop runs. The outputs for both the seasonal and rotation runs can be given on an hourly or daily basis.

The *Seasonal Run* tab (Figs. 8, 9) requires the user to select the following: *Site*, *Soil*, *Station Name*, *Weather*, *Crop*, *Experiment/Treatment* (as defined in the Management tab), and whether the simulation should be conducted with a focus on potential yield (i.e., water stress and nitrogen stress removed, potato only at this time). Once the site, soil, and station name have been selected, the interface queries the databases for associated weather. When the crop is selected, the interface presents only treatments that are applicable for the weather dates and crop selected. After the selections have been completed, the user can run the model with a single button click (Fig. 9).

The same inputs are needed for the *Rotation Builder* tab (Fig. 10), though *Site*, *Station Name*, *Soil*, and *Weather* are constant for the entire rotation cycle, while *Crop*, *Experiment/Treatment*, and the simulation of stresses can be changed for each step of the rotation. The *Rotation Builder* tab is shown in Fig. 11 filled out with a maize-fallow-maize rotation. Unlike the *Seasonal Run* tab, where multiple runs can be set up at the same time before running, the *Rotation Builder* tab only allows users to set up and run one rotation at a time. More rotation steps can be added by right clicking and selecting *Insert Row Below*. After the user selects the site, the interface will filter the station name and weather according to availability of stations for the site and weather data for the station.

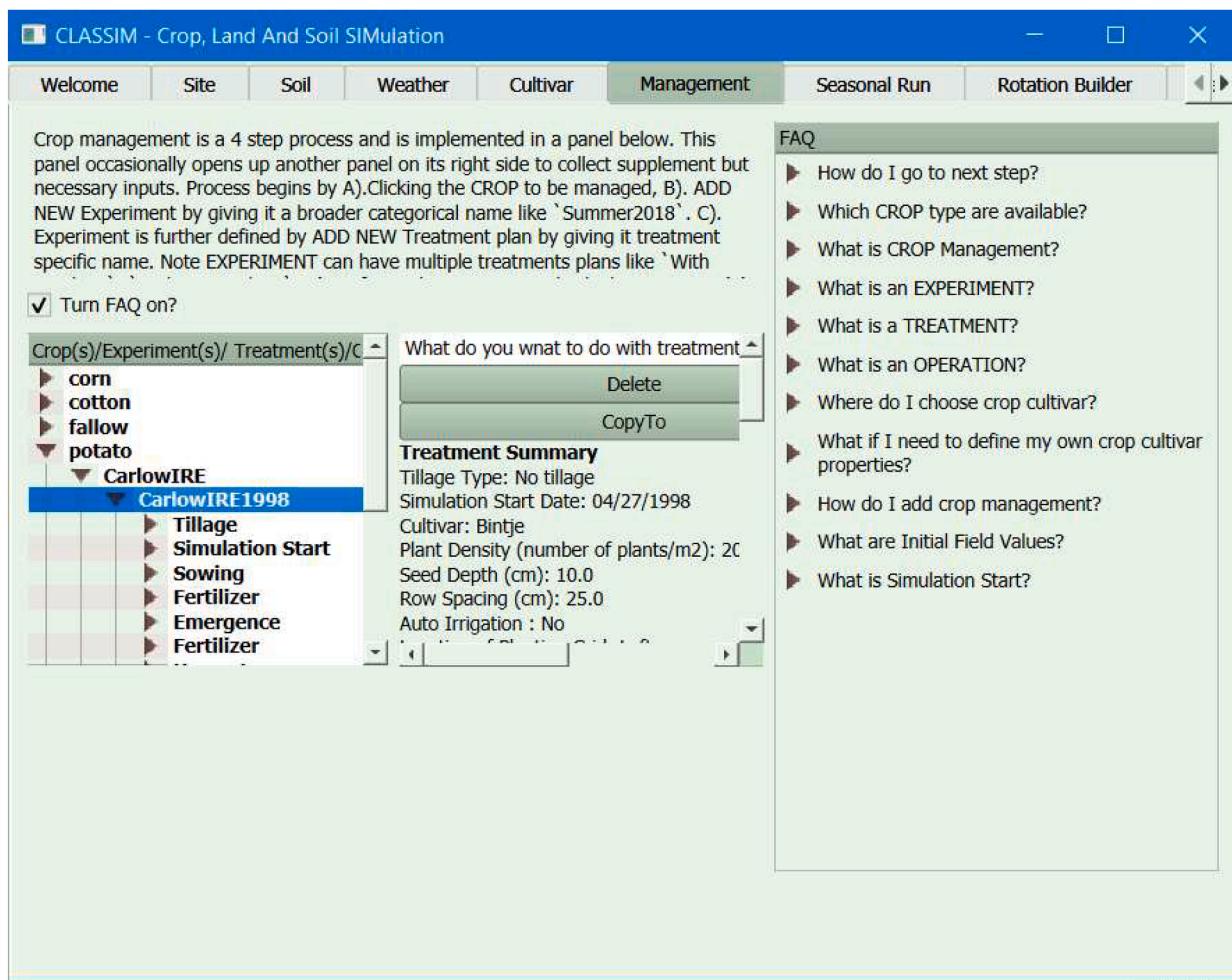


Fig. 7. Management tab dialog box. This dialog box allows the user to add or edit management options. The hierarchy is organized by crop, then by Experiment and then by treatment. Selecting a treatment will show a summary of the options. The example here is for a potato experiment.

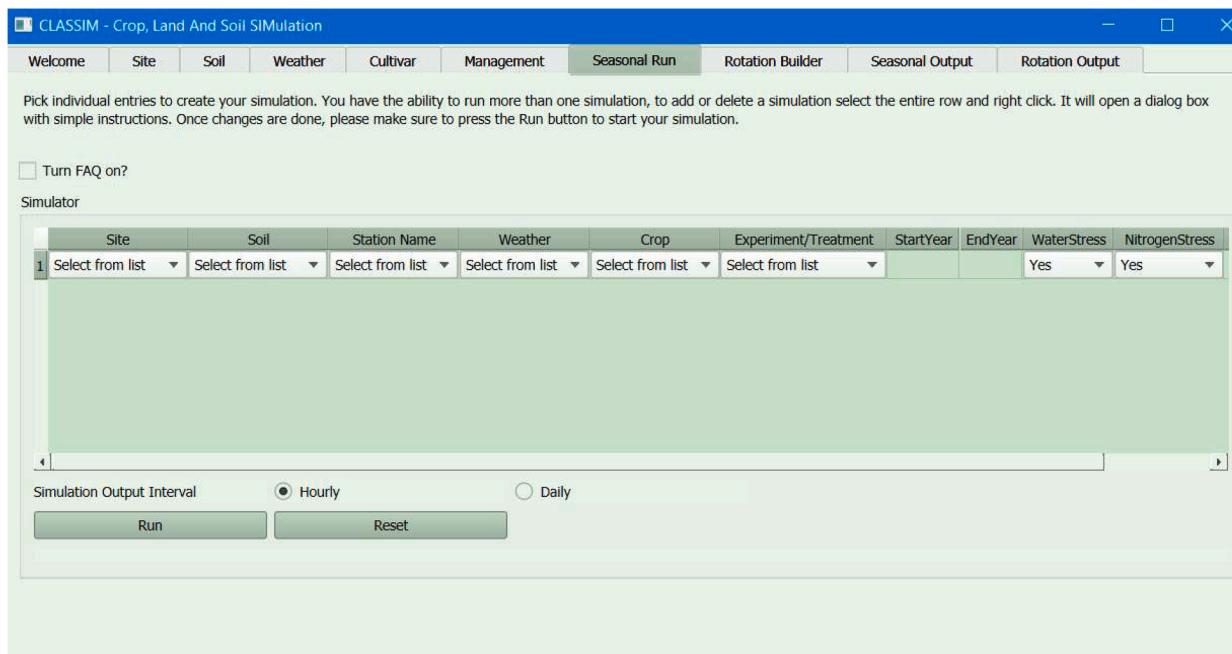


Fig. 8. Seasonal Run tab. This dialog box allows a user to select data sets to use to run a single crop for one growing season. Additional seasonal runs can be added for each row and multiple rows can be run at the same time.

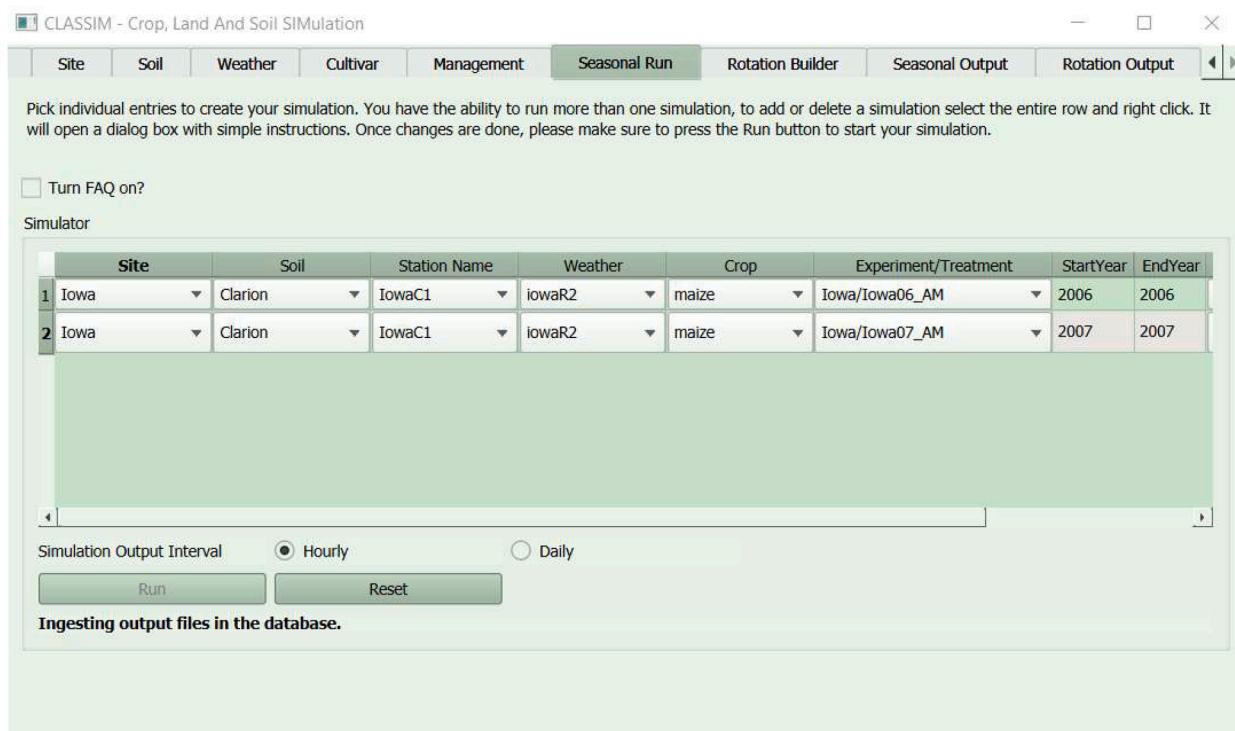


Fig. 9. Setup for two Seasonal Runs for the example simulation. These runs vary with respect to the Experiment/Treatment input. Messages at the bottom of the page inform the user of the program's progress.

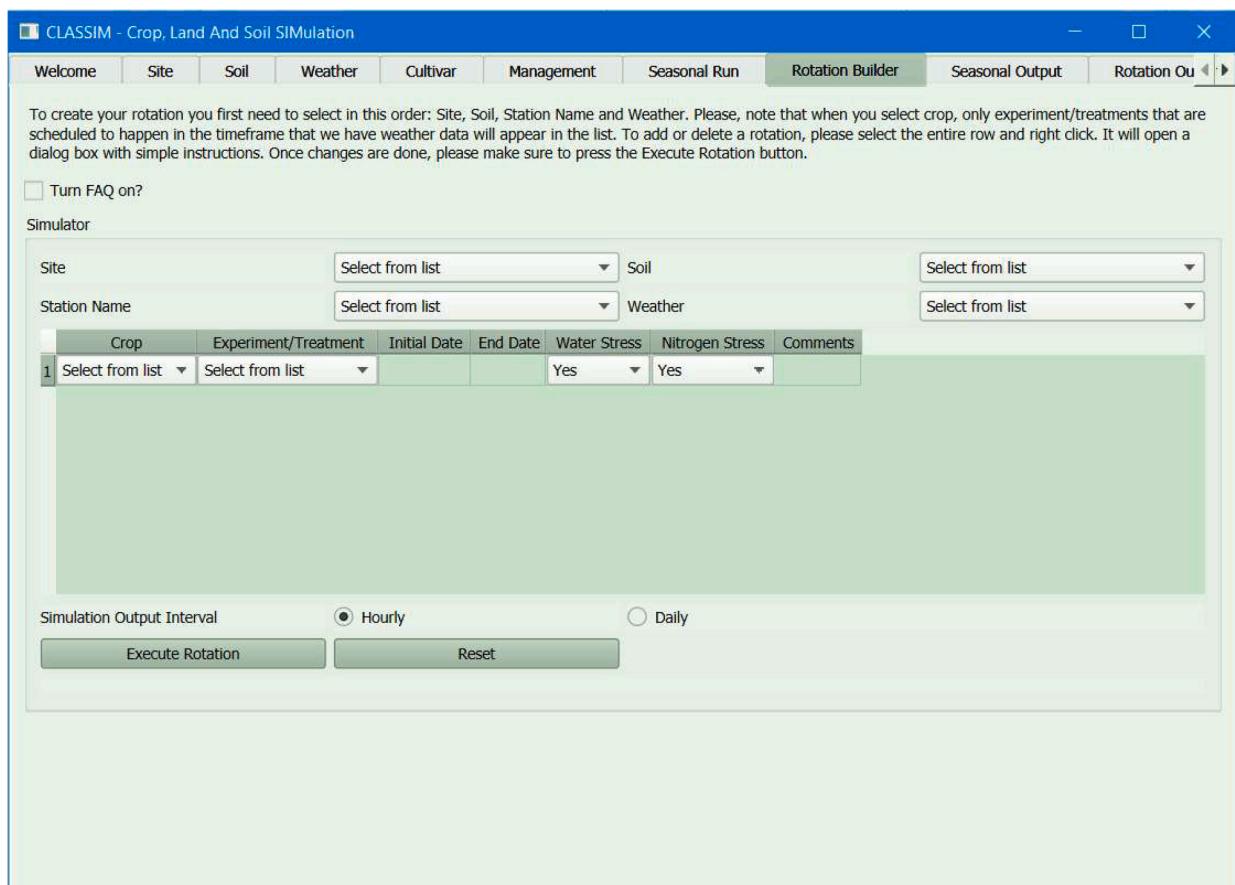


Fig. 10. Rotation Builder Tab. This tab is used to select data sets for a rotation. Each row is a different crop and time slot for the sequence.

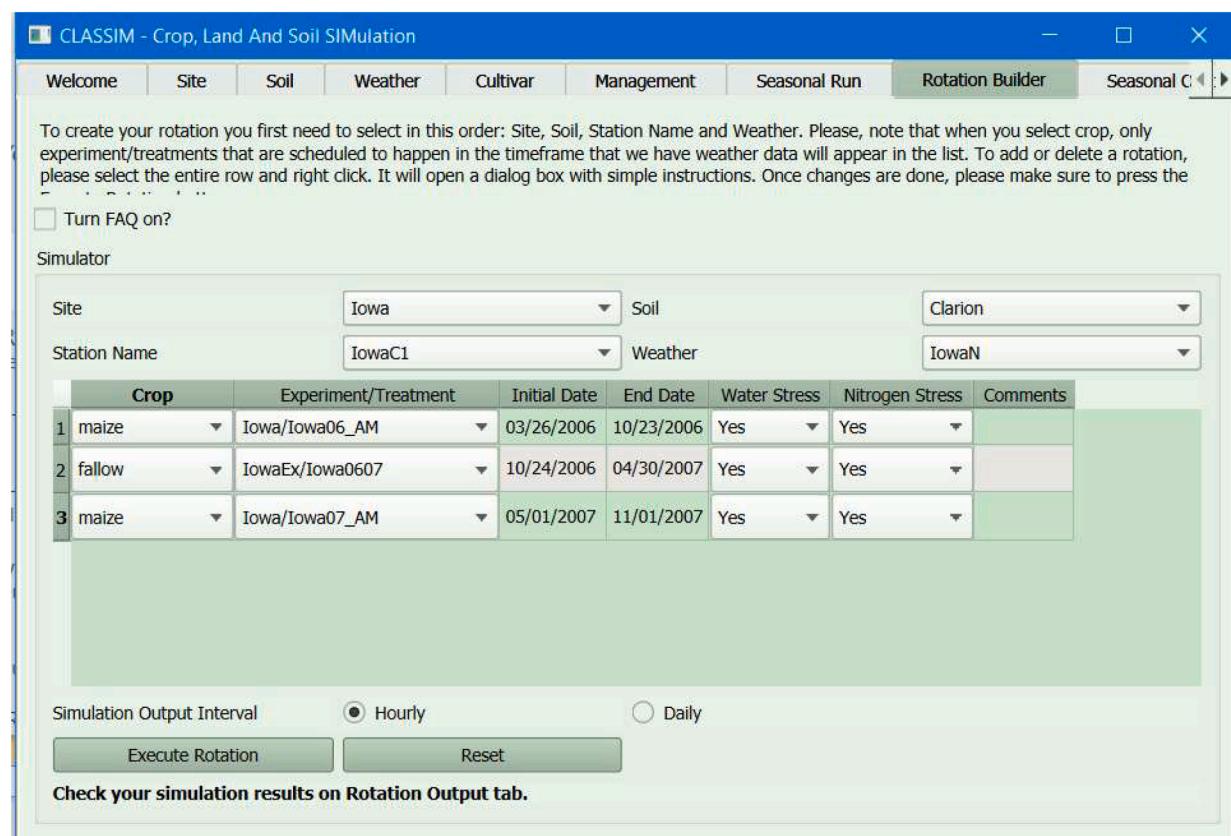


Fig. 11. Rotation Builder dialog box showing selected parameters to simulate a maize-fallow-maize rotation for the Iowa site. Messages about the simulation are output at the bottom of the window. The initial and end dates define the start and end dates of the simulation (soil processes can be started before the crop models and continue execution after the plant matures).

Similar filtering will be done after the user selects the crop and moves to the *Experiment/Treatment* dialog. For example, only *Experiment/Treatment* rows that use the chosen crop will be shown in the drop-down menu. The initial and end dates are selected from the treatment data which will provide the user confirmation of the date range they would like to simulate. If the dates ranges have a gap, the interface will notify the users. Once all the inputs have been selected, the rotation is run by clicking the *Run* button. The *Reset* button clears all the inputs.

The inputs for both seasonal runs and rotations are grouped to allow flexible run designs (Figs. 8–11). The user is guided through the process where the selections presented at each stage are narrowed according to previous choices. For example, a site can have multiple station names to represent different levels of CO₂ or average wind speeds. Once a user selects a site, only stations at that site are available for selection. Each station has multiple *weatherIDs*, which can be used to represent both historical and hypothetical future weather data sets for the selected station, such as historical weather with a 20% increase in rainfall or weather generated by a climate model. Once the station is selected, only weather data having those station parameters are available. This allows users to analyze the effects of rainfall, temperature, and CO₂ in isolation and in combination. Each crop species can have multiple *Experiments* and each *Experiment* can have multiple *Treatments*. The *Experiment* names predominantly exist for organizational purposes, allowing users to group the *Treatments* into categories. The *Treatments* are for selecting the management information. Though a site may be selected for a soil in the *Soil* tab, when setting up a run, any soil can be paired with any site. The availability of latitude and longitude for a site allows the program to search for soils data from the NRCS web site available at that location [24] and associated weather data.

6. Output visualization and analysis

The crop models in CLASSIM produce large quantities of daily or hourly data which are not easy to visualize or interpret without further processing, especially for novice modelers. Therefore, CLASSIM has built-in output analysis tools that create plots and statistics for a selected run. CLASSIM's built-in graphing tools make it easier for non-coders to use crop models, as users do not have to write a separate graphing code to interpret the data. Additional options for modifying the plot and exporting the graph are available within CLASSIM by right clicking on the graph. Available modifications include changing the axis ranges, changing the x and y axes to log scale, adding in an average line if multiple variables are plotted simultaneously, and selecting the opacity of the x and y background grid.

In the next sections we will give examples of a seasonal simulation and a rotation simulation and demonstrate the output capabilities.

6.1. Data source for examples

The weather, soil, and management data for the examples come from a study comparing evapotranspiration estimates of 29 models for multiple locations [19]. This paper compared several models used to simulate maize growth and yield under rainfed conditions in Iowa, USA over the period of 2006 to 2013. The maize cultivar used for the case study was from Golden Harvest (GH-9014). The sowing dates are in Kimball et al. [19]. The data for the examples were taken from the years 2006 and 2007. The simulation start dates were set as two weeks before planting. The initial soil water content was assumed to be at the drained upper limit for both years, as Iowa has fairly regular rainfall during the winter and spring to recharge the soil water. Fertilizer was applied as a split application of 100 kg N ha⁻¹ at planting and 200 kg N ha⁻¹ at 30

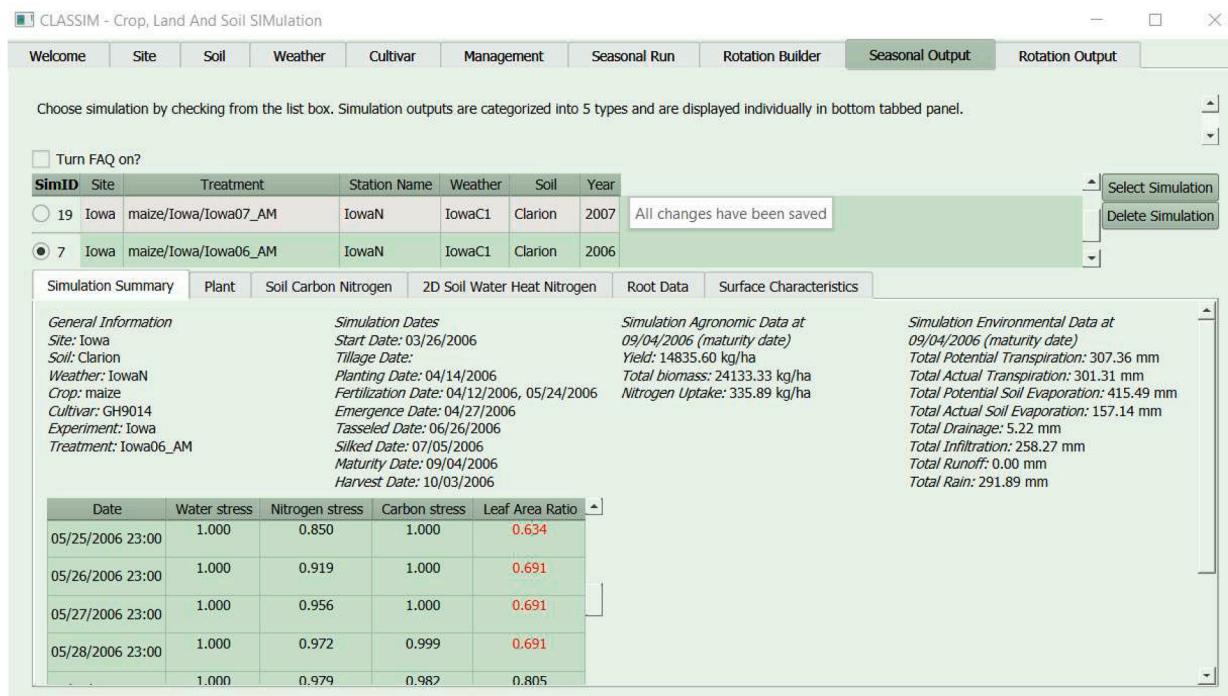


Fig. 12. Seasonal Output Tab. Here the user selects a completed simulation and summary information is provided as shown. A table indicating the date and time of occurrence of an abiotic stress and its magnitude from 0 to 1 is generated. Stress=1 indicates no stress. When stress values are less than 0.75, they are flagged in red. The user can select the sub-tabs (Plant, 2D Soil Water Heat Nitrogen, etc.) to view graphics and additional time-series information.



Fig. 13. Graphics example from a seasonal run showing gross photosynthesis and respiration. The simulation results are from the Iowa 2006 simulation.

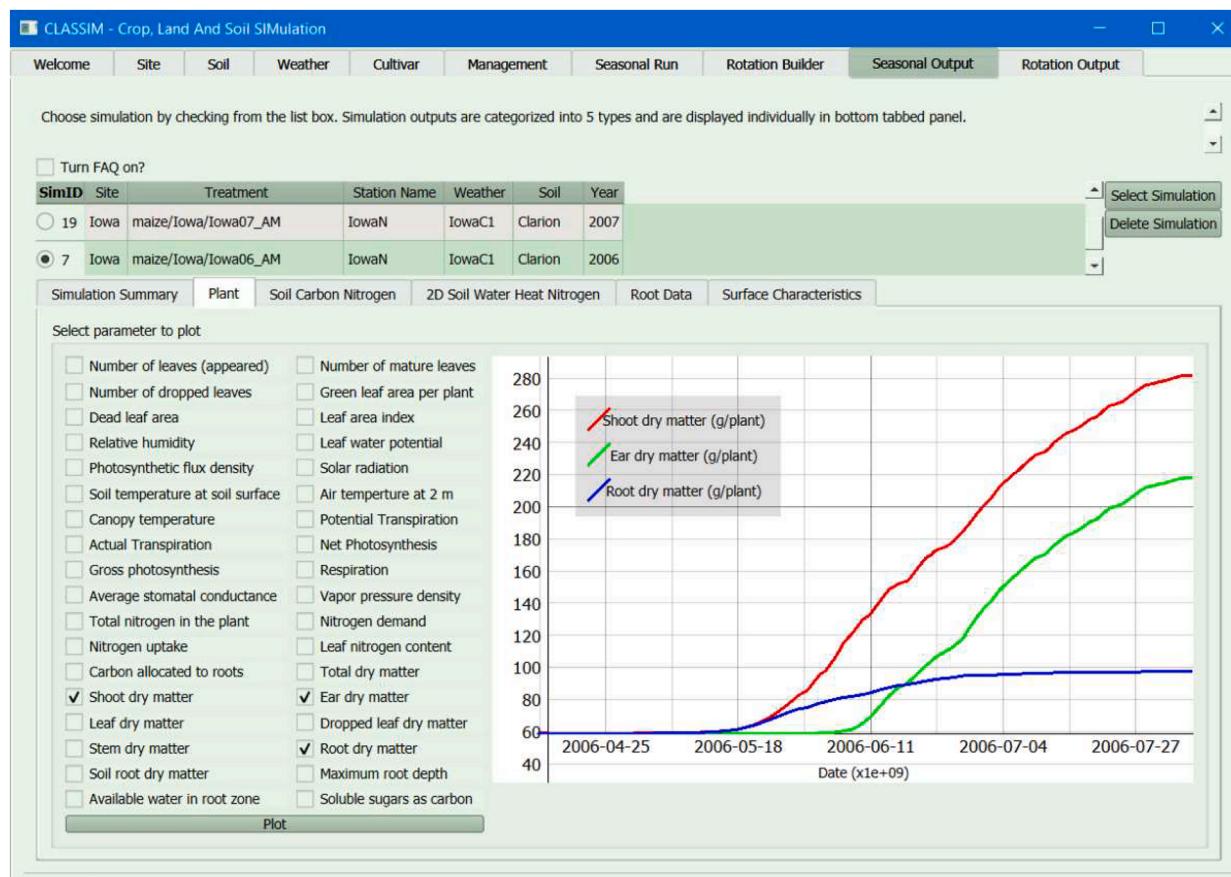


Fig. 14. Graph of shoot dry matter and ear dry matter available from the seasonal output tab. This is for the same simulation results as in Figs. 12 and 13.

days after planting (DAP). The case study was rainfed.

6.2. Seasonal output tab

For individual seasonal runs, the outputs can be viewed in the *Seasonal Output* tab (Fig. 12). The user must select a simulation and click *Select Simulation* to view the results. Simulations can also be removed using the *Delete Simulation* button. The *Simulation Summary* sub-tab provides general information on the selected run, such as the site and soil names used for the experiment, the significant dates for the simulation, including the simulation start and end dates, planting date, and the major phenology dates, the simulated agronomic data at maturity, and the simulated environmental data at maturity.

The *Simulation Summary* tab also contains a table of daily water, nitrogen, and carbon stress factors ($1 = \text{no stress}$). When stress factors are less than 0.75, they are highlighted in red to alert the user. The *Plant* sub-tab allows users to graph plant and weather variables, such as yield, leaf area index, leaf water potential, and solar radiation over time. Fig. 13 is an example graph generated from the *Plant* sub-tab that displays gross photosynthesis and respiration for maize as simulated by the Farquhar Ball Berry [6,36] model of photosynthesis and stomatal conductance implemented in MAIZSIM. Graphs have time on the x-axis as default. Fig. 14 shows the dry matter values for roots, ears, and shoots. Multiple variables can be plotted on the same graph for comparison.

The *Soil Water Heat Nitrogen* sub-tab generates contour plots of soil matric potential (cm suction), soil water content ($\text{cm}^3 \text{cm}^{-3}$), soil nitrogen concentration (mg L^{-1}), and soil temperature ($^{\circ}\text{C}$) across the 2D soil profile for user selected dates. The *Root Data* sub-tab generates daily contour plots of total root density (g cm^{-3}) and total root mass (g) across the 2D soil profile. Having an easy to use, clickable interface to generate graphs is particularly important when dealing with 2D data, as it makes

the interpretation and visualization of complex data much easier for users. The *Surface Characteristics* sub-tab allows the user to plot above ground environmental effects such as actual soil evaporation and runoff over the season. Multiple parameters can be graphed in the same plot for comparisons.

6.3. Rotation output tab

The same sub-tabs and options exist for the *Rotation Output* tab (Fig. 15) as for the *Seasonal Output* tab. Outputs for rotation simulations are graphed as a continuous variable over time and are not broken up by cropping season. A maize-fallow-maize rotation in Iowa was simulated to showcase CLASSIM's rotation capabilities. The menu presented to the user to set up the simulation is shown in Figs. 15, and Fig. 16 shows the results. The example demonstrates how multi-year simulations with a range of weather and management data can be carried out. Once the scenarios are added to the database, it is easy to run simulations and view output results. The trends for cumulative actual and potential soil evaporation and transpiration along with rainfall for the entire rotation are shown in Fig. 16.

The tables in Fig. 15 and graph in Fig. 16 show the results of the maize and fallow simulations in Iowa for 2006 and 2007. The effects of a fallow period on water availability for a following crop is often of interest. The fallow period (24 Oct 2006 to 30 April 2007) was during the winter and spring. Total rainfall was only about 10 mm less than during the growing season (Fig. 16.). Most of this rainfall was during the spring (data not shown). Actual soil evaporation was less than potential (202 vs. 422 mm). This is because the surface soil dries resulting in a soil mulch that resists further evaporation. Actual soil evaporation during fallow was less than infiltration indicating recharge of about 70 mm of water in the soil profile. There was no drainage and a small amount of

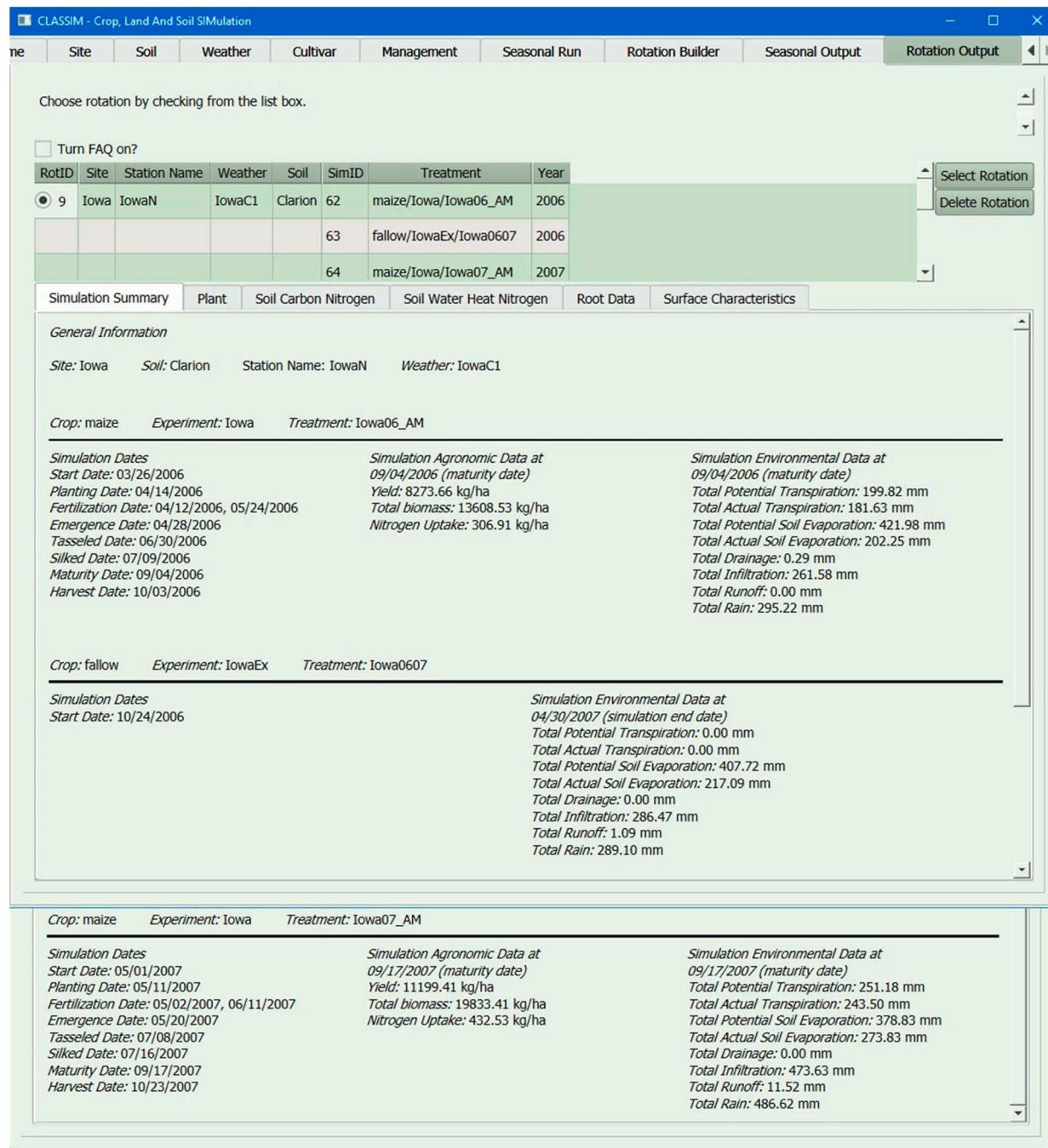


Fig. 15. Rotation Output tab. Once a rotation is selected a summary of the simulation is presented as well as tabs for further graphics. Summary results are shown in the main window, these include management, yield, and water related results.

runoff. At this point the models do not simulate snowfall or frozen soils. It is assumed that all rainfall that is not evaporated or runoff will infiltrate. When the maize model started simulation on 11 May of the second year, the soil water content at planting was affected by the previous fallow period.

There was more water stress in 2006 than in 2007 resulting in yield differences. The simulation summary shows the differences in potential and actual transpiration between the two seasons (200 vs 181 mm for 2006 and 251 vs 244 mm for 2007). The greater water stress in 2006 resulted in higher root growth relative to 2007.

The soil model used for the crop models is a two-dimensional finite element model (2DSOIL). CLASSIM provides two-dimensional contour plots of the average soil state for any day during the simulation. The date

is chosen from a drop-down menu (Fig. 17). The available variables are root mass, root density, water potential, water content, nitrogen concentration and temperature. Fig. 17 shows contour plots of these variables for a date that is about 10 days before tassel emergence. This provides a picture of the state of the soil just before the reproductive phase is reached.

The nitrogen (N) content in the soil in 2007 (Fig. 17b) is higher than in 2006 (Fig. 17a). This is partially due to carry over of N from the 2006 growing season. The effect of the roots on nitrogen uptake can be seen where the N content directly under the plant on the left side of the grid near the surface is lower than at the right side of the grid, which is farthest away from the plant and contains fewer roots (Fig. 17). In 2007 (Fig. 17b), the surface soil to about 8 to 10 cm is cooler than the soil

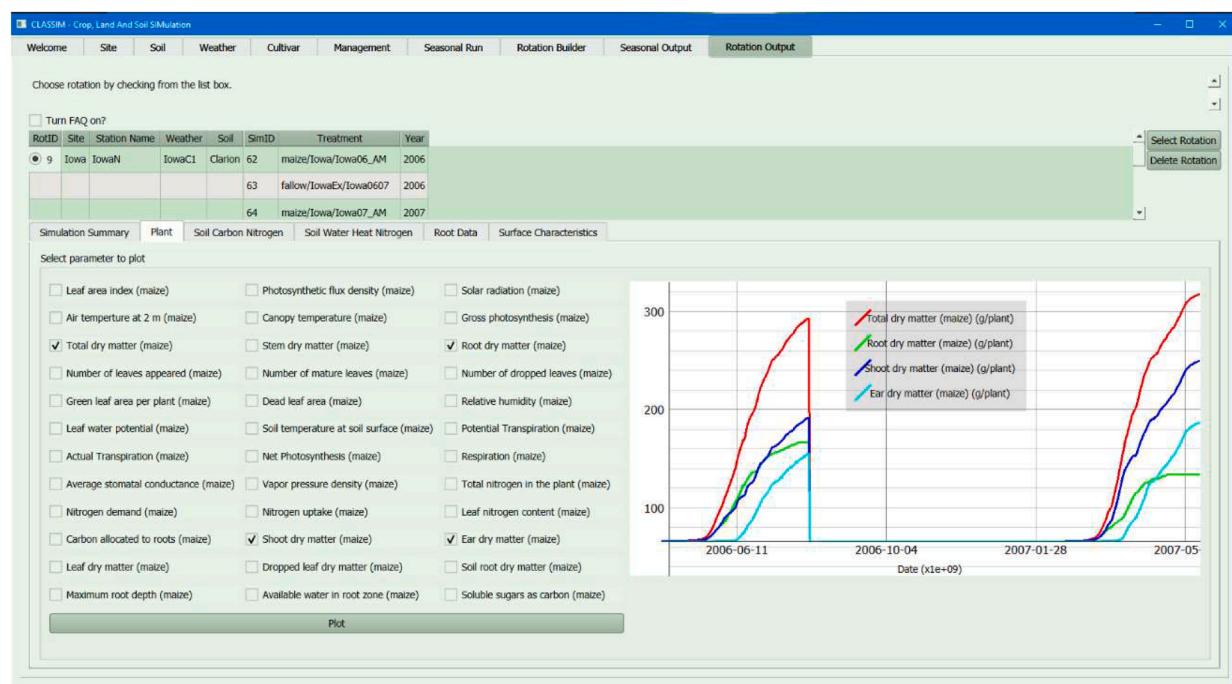


Fig. 16. Plant graphics dialog and example plot showing ear, root, shoot, and total dry matter for maize in a maize-fallow-maize rotation in Iowa, 2006 to 2007.

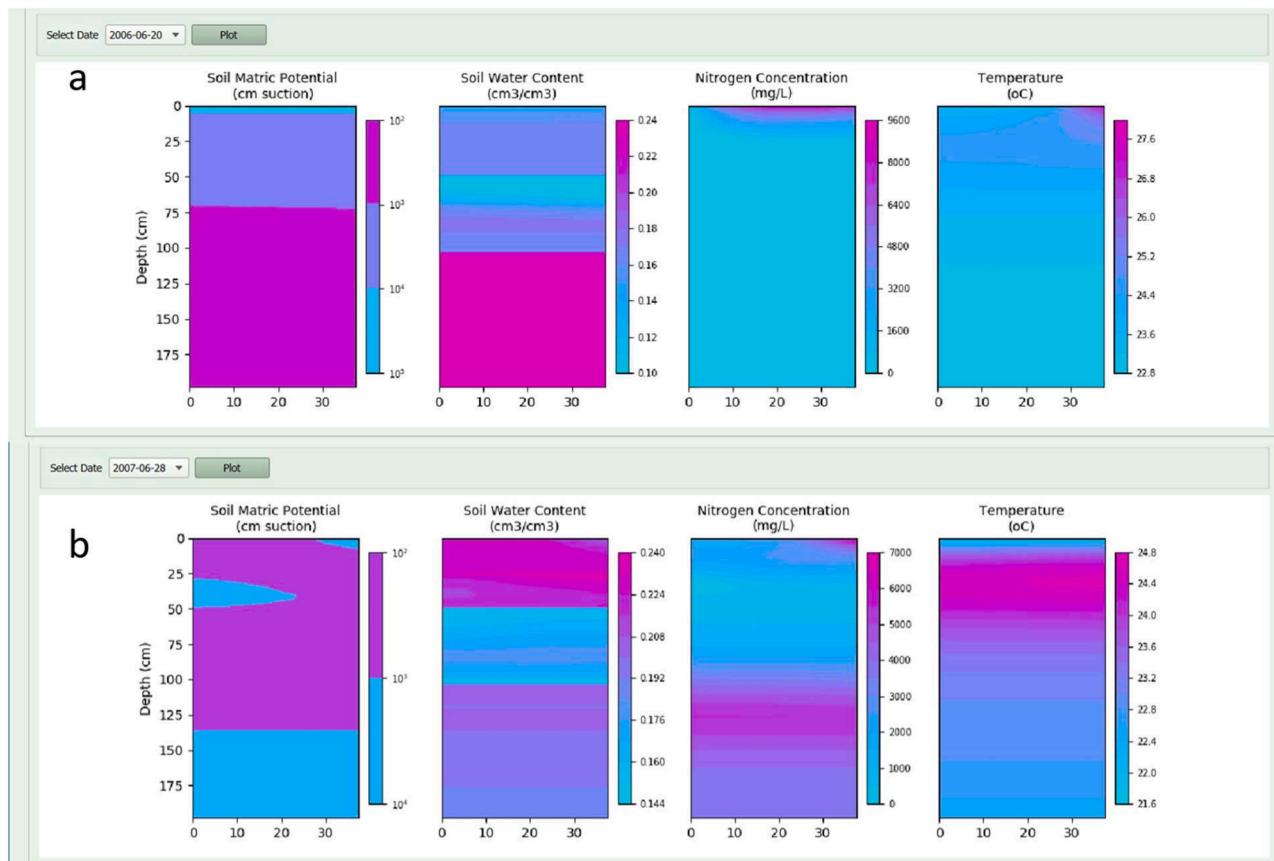


Fig. 17. Two-dimensional plots of soil properties for the maize years of a maize-fallow-maize rotation. The time is a date about 10 days before tassel emergence for 2006 (a) and 2007 (b). The plant is on the left of the contour plot (0,0). The top right of the contour plot (37.5,0) is half the interrow distance (75 cm rows).

below. The water contents are also high at the surface from a recent rainfall. This contributes to a cooling effect on the soil.

7. User applications

CLASSIM is intended for students and researchers who are new to crop modeling and/or the crop models supported by CLASSIM. The interface is distributed as an installer and is available on GitHub (link provided at the end of this paper).

The models supported by CLASSIM are based on the 2D finite element soil process simulation model 2DSOIL [29] and tend to be more mechanistic and modular than other crop and soil models [17,18]. This allows for less location specific site calibration [15], which saves time and allows for a clearer understanding of the effects of model error, as frequent parameter calibration can compensate for underlying errors without actually resolving the root of the problem [9]. CLASSIM is an evolving platform and ACSL has several further additions under development. Future versions of CLASSIM will include a cover crop rye model, a rice model, and the ability to add observed development and growth data into the SQL databases. The ability to compare simulations will also be added. The addition of the cover crop model in combination with the existing 2DSoil model and rotation simulation abilities, will make CLASSIM a powerful tool for examining the long-term effects of cover crops on soil organic matter, soil water availability, and cash crop yields. This set up could also be used to study carbon sequestration in cover cropping systems.

8. Summary and conclusions

CLASSIM is the database driven interface for the mechanistic crop models GOSSYM, GLYCIM, SPUDSIM and MAIZSIM. The interface facilitates the generation of input files and visualization of results. CLASSIM can run both individual, single crop growing seasons and multi-season crop rotations at a specific field location identified by a user-specified latitude, longitude, and elevation. Each site is associated with input data regarding soil physical and hydraulic characteristics, a climate station and associated hourly or daily weather data, and treatments which describe the simulations to be conducted at those locations. These treatments include inputs for management and crop selection. Users can then assemble desired site, weather, soil, and treatment combinations to conduct either single- or multiple-season runs, including rotation studies. CLASSIM provides above- and below-ground time-series and end-of-season outputs regarding agronomic and physiological responses of interest to growers and scientists.

CLASSIM (source code and installer) can be obtained from:

Release CLASSIM V 2.2 · USDA-ARS-ACSL/CLASSIM ([github.com](https://github.com/USDA-ARS-ACSL/CLASSIM))

Declaration of Competing Interest

Authors have no competing interests.

Data availability

<https://github.com/USDA-ARS-ACSL/CLASSIM>

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at [doi:10.1016/j.atech.2023.100281](https://doi.org/10.1016/j.atech.2023.100281).

References

- [1] B. Acock, Y. Pachepsky, E. Mironenko, F. Whisler, V. Reddy, GUICS: a generic user interface for on-farm crop simulations, *Agron. J.* 91 (4) (1999) 657–665.
- [2] L. Ahuja, L. Ma, D. Timlin, Trans-disciplinary soil physics research critical to synthesis and modeling of agricultural systems, *Soil Sci. Soc. Am. J.* 70 (2) (2006) 311–326.
- [3] P.D. Alderman, A comprehensive R interface for the DSSAT cropping systems model, *Comput. Electron. Agric.* 172 (2020), 105325.
- [4] Anaconda Software Distribution, Anaconda Documentation, Anaconda Incorporated, 2020. Retrieved from, <https://docs.canaconda.com>.
- [5] S. Asseng, *Wheat Crop Systems: A Simulation Analysis*, CSIRO, Melbourne, Australia, 2004.
- [6] J.T. Ball, I.E. Woodrow, J.E. Berry, A model predicting stomatal conductance and its contribution to the control of photosynthesis under different environmental conditions, *Prog. Photosynth. Res.* 4 (1987) 221–224.
- [7] W. Bauerle, D. Timlin, Y. Pachepsky, S. Anantharamu, Adaptation of the biological simulation model MAESTRA for use in a generic user interface, *Agron. J.* 98 (1) (2006) 220–228.
- [8] A. Dathe, D. Fleisher, D. Timlin, J. Fisher, V. Reddy, Modeling potato root growth and water uptake under water stress conditions, *Agric. For. Meteorol.* 194 (2014) 37–49.
- [9] D. Fleisher, K. Haynes, D. Timlin, Cultivar coefficient stability and effects on yield projections in the SPUDSIM model, *Agron. J.* 112 (2) (2020) 828–843.
- [10] D. Fleisher, R. Shillito, D. Timlin, S. Kim, V. Reddy, Approaches to modeling potato leaf appearance rate, *Agron. J.* 98 (3) (2006) 522–528.
- [11] D.H. Fleisher, et al., Effects of CO₂ and temperature on crops: lessons from SPAR growth chambers, Editors, in: J.L. Hatfield, D. Fleisher (Eds.), *Improving Modeling Tools to Assess Climate Change Effects Oncrop Response*, ASA, CSA, and SSSA, Madison, WI, 2010, pp. 55–86.
- [12] D. Holtzworth, et al., APSIM next generation: overcoming challenges in modernising a farming systems model, *Environ. Model. Softw.* 103 (2018) 43–51.
- [13] J. Jones, et al., The DSSAT cropping system model, *Eur. J. Agron.* 18 (3–4) (2003) 235–265.
- [14] J.W. Jones, et al., The DSSAT cropping system model, *Eur. J. Agron.* 18 (3) (2003) 235–265.
- [15] Kim, S.H. et al., 2012. Modeling temperature responses of leaf growth, development, and biomass in Maize with MAIZSIM. 104(6): 1523–1537.
- [16] S. Kim, J. Lieth, A coupled model of photosynthesis, stomatal conductance and transpiration for a rose leaf (*Rosa hybrida* L.), *Ann. Bot.* 91 (7) (2003) 771–781.
- [17] S. Kim, et al., Canopy photosynthesis, evapotranspiration, leaf nitrogen, and transcription profiles of maize in response to CO₂ enrichment, *Glob. Chang. Biol.* 12 (3) (2006) 588–600.
- [18] S. Kim, et al., Modeling temperature responses of leaf growth, development, and biomass in maize with MAIZSIM, *Agron. J.* 104 (6) (2012) 1523–1537.
- [19] B.A. Kimball, et al., Simulation of maize evapotranspiration: an inter-comparison among 29 maize models, *Agric. For. Meteorol.* 271 (2019) 264–284.
- [20] P. Krishnan, et al., Web-based crop model: web InfoCrop - wheat to simulate the growth and yield of wheat, *Comput. Electron. Agric.* 127 (2016) 324–335.
- [21] Lazzaretti, A.T., Fernandes, J.M.C., Pavan, W., Toebbe, J. and Wiest, R., 2016. AgroDB – integration of database management systems with crop models.
- [22] J.M.O. Leyton, APEXeditor: a spreadsheet-based tool for editing APEX model input and output files, *J. Softw. Eng. Appl.* 12 (10) (2019) 15.
- [23] J. Lizaso, W. Batchelor, M. Westgate, A leaf area model to simulate cultivar-specific expansion and senescence of maize leaves, *Field Crops Res.* 80 (1) (2003) 1–17.
- [24] NRCS, Soil Survey Geographic (SSURGO) Database, Natural Resources Conservation Service United States Department of Agriculture, 2010.
- [25] K.R. Reddy, V.G. Kakani, J.M. McKinion, D.N. Baker, Applications of a cotton simulation model, GOSSYM, for crop management, economic, and policy decisions, Editors, in: L.R. Ahuja, L. Ma, T.A. Howell (Eds.), *Agricultural System Models in Field Research and Technology Transfer*, CRC Press, Boca Raton, 2002.
- [26] RiverBank Computing, PyQt5 Documentation, RiverBank Computing, 2022. Retrieved from, <https://www.riverbankcomputing.com/software/pyqt/>.
- [27] M. Schaap, F. Leij, M. van Genuchten, ROSETTA: a computer program for estimating soil hydraulic parameters with hierarchical pedotransfer functions, *J. Hydrol.* 251 (3–4) (2001) 163–176.
- [28] W. Sun, et al., Effects of elevated CO₂ and temperature on soybean growth and gas exchange rates: a modified GLYCIM model, *Agric. For. Meteorol.* 312 (2022), 108700, <https://doi.org/10.1016/j.agrformet.2021.108700>.
- [29] D. Timlin, Y. Pachepsky, B. Acock, A design for a modular, generic soil simulator to interface with plant models, *Agron. J.* 88 (2) (1996) 162–169.
- [30] D. Timlin, et al., Error analysis of soil temperature simulations using measured and estimated hourly weather data with 2DSOIL, *Agric. Syst.* 72 (3) (2002) 215–239.
- [31] M. van Genuchten, A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Sci. Soc. Am. J.* 44 (5) (1980) 892–898.
- [32] N. Villoria, et al., Rapid aggregation of global gridded crop model outputs to facilitate cross-disciplinary analysis of climate change impacts in agriculture, *Environ. Model. Softw.* 75 (2016) 193–201.
- [33] T. Vogel, M.T. van Genuchten, M. Cislerova, Effect of the shape of the soil hydraulic functions near saturation on variably-saturated flow predictions, *Adv. Water Resour.* 24 (2) (2000) 133–144.
- [34] J. Williams, R. Izaurralde, The APEX Model, Blackland Res. Center, Texas A&M Univ., Temple, 2005. BRC Rep. 2005-02.
- [35] Y. Xia, et al., Continental-scale water and energy flux analysis and validation for the North American Land Data Assimilation System project phase 2 (NLDAS-2): 1. Intercomparison and application of model products, *J. Geophys. Res. Atmos.* 117 (2012).

- [36] Y. Yang, et al., Simulating canopy transpiration and photosynthesis of corn plants under contrasting water regimes using a coupled model, *Trans. ASABE* 52 (3) (2009) 1011–1024.
- [37] J. Zhang, K. Howard, J. Gourley, Constructing three-dimensional multiple-radar reflectivity mosaics: examples of convective storms and stratiform rain echoes, *J. Atmos. Ocean. Technol.* 22 (1) (2005) 30–42.
- [38] J. Zhang, et al., National mosaic and multi-sensor QPE (NMQ) system: description, results, and future plans, *Bull. Am. Meteorol. Soc.* 92 (2011) 1321–1338.