ACME (Agile Crop Model Ensemble): A package to generate and run large virtual experiments with ensembles of crop simulation models

Michel Giner, Francois Affholder, Gatien N. Falconnier, Vimbayi Grace Petrova Chimonyo, Louise Leroux, Sandrine Auzoux, Myriam Adam, Patrice Kouakou, Romain Loison, Cyrille A. Midingoyi, et al.

SoftwareX 27 (2024) 101777

Contents lists available at ScienceDirect

SoftwareX

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

ACME (Agile Crop Model Ensemble): A package to generate and run large virtual experiments with ensembles of crop simulation models

Michel GINER [a,b,*], Francois AFFHOLDER [b,c,d], Gatien N. FALCONNIER [b,e,f,g], Vimbayi Grace Petrova CHIMONYO [f,h], Louise LEROUX [b,i,j], Sandrine AUZOUX [b,k], Myriam ADAM [l,m,n], Patrice KOUAKOU [b,o], Romain LOISON [a,p], Cyrille A. MIDINGOYI [a,b], Erik CHAVEZ [q]

[a] CIRAD, UPR AIDA, F-34398 Montpellier, France
[b] AIDA, Univ Montpellier, CIRAD, Montpellier, France
[c] CIRAD, UPR AIDA, Maputo, Mozambique
[d] Faculdade de Agronomia e Engenharia Florestal, Univ Eduardo Mondlane, Maputo, Mozambique
[e] CIRAD, UPR AIDA, Harare, Zimbabwe
[f] International Maize and Wheat Improvement Centre (CIMMYT), 12.5km Peg, Mazowe Rd, Harare, Zimbabwe
[g] Department of Plant Production Sciences and Technologies, University of Zimbabwe, Harare, Zimbabwe
[h] Centre for Transformative Agriculture and Food Systems, University of KwaZulu-Natal, Scottsville, Pietermaritzburg, South Africa
[i] CIRAD, UPR AIDA, Nairobi, Kenya
[j] IITA, Nairobi, Kenya
[k] CIRAD/AIDA Bâtiment D, Bureau 031 Station de La Bretagne, 40 chemin de Grand Canal, CS 12014 97743 Saint-Denis Cedex 9 Réunion
[l] Faculty of Agriculture and Food Processing, National University of Battambang, Battambang, Cambodia
[m] CIRAD, UMR AGAP Institut, F-34398 Montpellier, France
[n] AGAP Institute, Univ. Montpellier, INRAE, CIRAD, Inst. Agro. F-34398 Montpellier, France
[o] CIRAD/AIDA INERA, GRN/SP Farakoba, 01 BP 910 Bobo-Dioulasso 01 Burkina Faso
[p] CIRAD, UPR AIDA, Institut de Recherche sur le Coton (IRC) Cotonou, Bénin
[q] Centre for Environmental Policy, Imperial College London, South Kensington Campus, London, UK

ARTICLE INFO

Keywords:
Database
Multi-modelling
Batch computing

ABSTRACT

The objective of the ACME (Agile Crop Model Ensemble) software is to facilitate the simulation of large virtual experiments using an ensemble of crop models on personal computers running Windows OS. These simulations can either be point-based or spatially explicit. The package includes an ensemble of three crop models (executable files and input and output data) and two databases: 'MasterInputOutput' which contains the input and output variables of the models, and 'ModelDictionary', which contains the connections and required transformations between variables of the different models. In addition, an executable code 'DataMill' applies these transformations and manages the data flow between the MasterInputOutput database and the file system of each model, before (input parameters) and after (output parameters) running the virtual experiment.

Metadata

Nr	Code metadata description	*Please fill in this column*
C1	Current code version	*2.0*
C2	Permanent link to code/repository used for this code version	https://github.com/GinerM/ACME-Datamill/tree/main/repo
C3	Permanent link to reproducible capsule	https://github.com/GinerM/ACME-Datamill/tree/main/repo/ACME_EspaceDeTravail
C4	Legal code license	*BSD Licence.*
C5	Code versioning system used	*Git (Gitlab of Cirad)*

(*continued on next column*)

(*continued*)

(*continued on next page*)

* Corresponding author.
E-mail address: michel.giner@cirad.fr (M. GINER).

https://doi.org/10.1016/j.softx.2024.101777
Received 2 February 2024; Received in revised form 21 May 2024; Accepted 26 May 2024
Available online 3 June 2024

(*continued*)

C6	Software code languages, tools and services used	*Visual Basic .NET*
C7	Compilation requirements, operating environments and dependencies	*.NET framework 4.0*
C8	If available, link to developer documentation/manual	*https://github.com/GinerM/ACME-Datamill/blob/main/repo/DataMill_Manual.docx*
C9	Support email for questions	Michel.giner@cirad.fr or francois.affholder@cirad.fr

1. Motivation and significance

Cropping systems are complex systems in which plants interact with climate and soils, influenced by farmers' management. Crop models describe processes of crop growth and development as a function of weather and soil conditions, and management. Typically, such models predict the biomass of crop components (e.g., leaves, stems, roots and harvestable products) as they change over time and similarly, changes in soil moisture and nutrient status. "Virtual experiments" run by crop simulation models can be used as an effective means to explore the impact of changes in management and the environment on crops to help design more sustainable cropping systems. Numerous crop models exist with similar objectives and predictive capacities but with varying levels of detail across the components of the cropping system. The simulated data is subject to uncertainties related to model parametrisation and the structure of the model itself [1–3], which occasionally might be substantial enough to compromise the validity of the simulations (e.g. [4]). "Multi-model ensembles", i.e. sets of different simulation models, are increasingly being used to ensure better robustness of the conclusions drawn from virtual crop simulation experiments (see e.g. https://agmip.org/).

However, the use of a multi-model ensemble amplifies the technical challenges with the use of crop models and increases the time needed to prepare model input files, to run the virtual experiment and to extract and analyse model output data. A crop model typically has more than 400 state variables, consists of more than 10 000 lines of computer code, and takes 0.3 to 1 s on a personal computer to run a 'simulation unit' (a crop growth simulation over one growing season). A basic simulation experiment to identify the best sustainable cropping practices in a given environment will require between 10^3 and 10^5 simulation units. This translates to a total simulation time ranging from a few minutes to over a day on a standard PC computer. Most crop models rely on a complex file system containing the necessary model inputs and the intended model outputs, without a standardized format for easily transferring inputs between different models [5,6]. Updating the file system of a single model with the data necessary for a virtual simulation experiment demands between half a day to a week for a scientist with expertise in the use of that model. This time depends on the size of the experiment and on how raw input data are structured. Furthermore, extracting and adapting model output data for systematic analysis and visualization may also take a significant amount of time from experts. The data to be extracted depends largely on the purpose of the virtual experiment.

The objective of the software chain, 'Agile Crop Model Ensemble' (ACME), presented here was to reduce these technical challenges and hence the time needed for scientists and engineers to prepare and analyse large virtual experiments run by an ensemble of crop models. The software chain presented in this paper was developed to run on a PC under the current version of Windows OS (Windows 10 Professional 22H2 Build 19045.4170 and later)

The software chain enables large virtual simulation experiments across soils and climates (including climate change scenarios) and spatialization over large geographic areas, using a set of crop models. These virtual experiments will contribute to answering critical research questions regarding how model uncertainty impacts recommendations for sustainable crop management, also aiding climate change adaptation and mitigation.

2. Software description

2.1. Software architecture

The general concept of ACME is described in Fig. 1.

The ACME software package consists of i) an ensemble of crop models, ii) a database enabling the management of input and output data of these crop models ('MasterInputOutput'), iii) a database containing tables that link the variables in the MasterInputOutput to the variables in the different crop models ('ModelDictionary'), and iv) the software that generates the data flows between these databases and runs the crop models – named 'DataMill'. These components are described in the next paragraphs.

The crop model ensemble comprises a number of dynamic crop models that simulate at a daily time step key variables of the cropping system. Each crop model is an executable file containing mathematical equations translated into a computer program, along with the commands handling the inputs and outputs from and to a file system specific to each model. This file system is hereafter referred to as the '*native files*' of the crop model. Besides, each crop model has its set of input parameters and variables and its simulated output variables, hereafter named 'inputs and outputs' of the model. Currently, ACME incorporates the three following crop models: Dssat [7], Stics [8], and Celsius [9]. These are dynamic soil-crop process-based models that simulate potential crop growth, together with water and nitrogen stress. The models differ in the formalisms implemented to account for these stresses [10]. Any other crop model compatible with standalone execution could be included in ACME.

MasterInputOutput is the database containing selected input variables and output variables for running ACME. In its current state, this set of variables, named ACME inputs and ACME outputs, with their names and units, was identified by a team of agronomists as the minimum set of variables required to evaluate the impact of climatic risks on cropping systems. It can be modified to include any selection of variables from the set of input and output variables formed by the union of all the sets of variables of each specific model.

As a result, the total number of input (or output) variables of a specific model is greater or equal to the number of ACME inputs (or outputs). Certain input variables of a given model may match ACME input variables or input variables of other models of the ensemble. Among them, some may share identical names and units, while others may differ in one or both of these aspects. Certain input variables may be model-specific, and while there may be conceptual similarities, mathematical transformation can link these input variables. ACME input variables are organized in tables corresponding to each key component of the simulated cropping system (e.g. soil; weather; crop; and management).

ModelDictionnary is the database containing: i) the list of ACME input and output variables with their description and type (e.g. real or integer number, text, Boolean), ii) the list of crop model inputs specific to the model of the ensemble, iii) the data defining the link between ACME inputs and model-specific inputs for each crop model of the ensemble, including the mathematical transformation applied to ensure the match between model-specific inputs and ACME inputs, whenever applicable, iv) default fixed values of model-specific inputs that are not related to any ACME input. For these inputs, model experts need to be consulted to ensure that the default values are appropriate for the study that is to be undertaken.

Some crop models may have different assumptions and formalisms to simulate a given process (e.g. light utilization using the 'radiation use efficiency approach' or 'gross photosynthesis minus respiration', soil water dynamics using the 'tipping bucket capacity approach' or

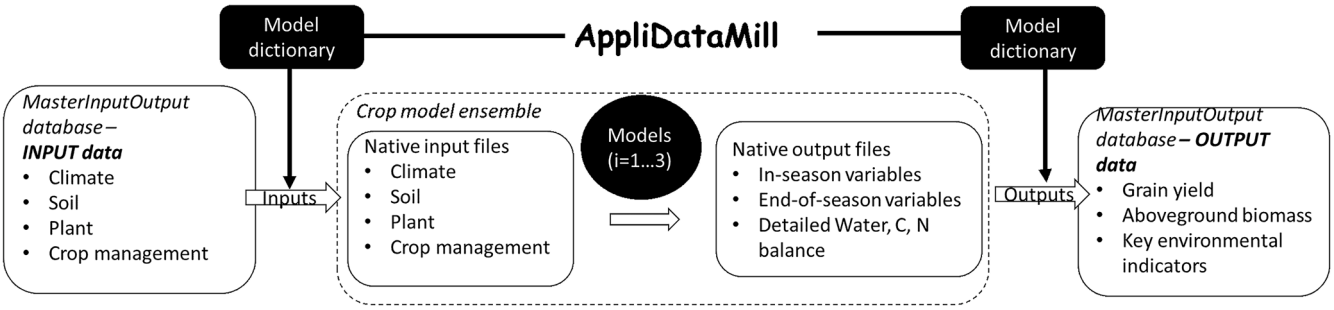

Fig. 1. Conceptual diagram presenting the structure and functioning of the ACME software package.

Richards equations, see [11]). For these types of parameters, the ACME software will not help to harmonize the parameter values across models and calibration of model-specific parameters by model experts is still required.

DataMill is the executable code written in Visual Basic (VB.net). It reads ACME input variables from the MasterInputOutput database. For each model-specific input, Datamill assigns a value based on the link defined in the ModelDictionnary, i.e. either using the default value or applying the mathematical transformation. Then, each model-specific input is written in the model's native file system, and DataMill launches the model simulations and distributes the computing tasks across the processor cores. Finally, the code reads the selected output variables from the native file system of each crop model and writes them into the MasterInputOuptut database. The main input table in the MasterInputOuptut database, containing the list of simulation units, is indexed with a unique identifier, which is used for naming the files in the native file system of the models. This ensures immediate and secure retrieval of simulation outputs and inputs for comparison and data analysis.

2.2. Software functionalities

2.2.1. Facilitate the design of large virtual experiments

First, the user has to inform a database with tables containing the relevant information defining the virtual simulation experiment. Queries that compute the cartesian product of these tables generate the combinatory space needed for exploring how variations in input factors influence the outputs of the different models. These tables need to contain the factor levels for the virtual experiment (one factor per table), enabling the crossing of factors and their levels using simple queries. Additional queries are used to place the resulting data (derived from the queries computing the cartesian product of the initial tables) into the relevant input tables of the MasterInputOutput database.

2.2.2. Automatically generates the numerous input files required for large virtual experiments with several crop models from a relational data base

Subsequently, the user can launch the DataMill application. Through the user interface, the user can set the path to the MasterInputOutput database in his computer file system and generate the input files of the ensemble's models in their native file system. These native input files are stored in a subdirectory of the Datamill working directory, with individual subdirectories for each model. The user needs to provide the native plant file with plant parameters for each model.

2.2.3. Launches simulations and distributes workload across processors

The user can adjust the number of processor cores used for the simulation. Launching simulations with the models of the ensemble is facilitated via model-specific buttons. The development of DataMill in VB.Net and the MS-DOS subsystem allow relevant parallelization on a single personal computer for batch computing when batch elements are independent as it is the case with the ACME virtual experiments.

Once the simulations have been executed, the user can launch the summary function, compiling the main simulation outputs, in a table of the MasterInputOutput database, named SummaryOutput, referenced by the identifier of each simulation and the name of the model used.

3. Illustrative examples

The approach was first developed and tested for field locations in Tanzania that represent contrasting climates and a diversity of eco-regions where crop production is prone to climate risks. We chose maize, as a typical crop sensitive to inter-annual variations of rainfall and temperature.

We used ACME to generate, run, and analyse a virtual simulation experiment crossing key factors determining the agronomic and environmental performance of the maize-based cropping systems:

- 31 sowing dates covering, with a 10 days step, a large range of sowing dates starting from the earliest observed onset of the rainy season in the region of Tanzania under study,
- 4 soil types that are identical in terms of fertility (same organic carbon and nitrogen on a mass basis) but contrasting in their capacity to store moisture,
- 2 maize cultivars with contrasting crop cycle duration, thus with a contrasting potential to intercept solar radiation, but also with contrasting exposure to water and nutrient stresses
- 20 years of daily weather data (1980–2000),
- 6 locations with corresponding climate and thus with a specific series of weather data,
- 3 production situations: i) without stress to simulate 'radiation and temperature determined potential yield', ii) only water stress to simulate 'water-limited yield' and iii) nitrogen and water stresses. These production situations are achieved by de-activating the simulation of water and/or nitrogen stresses in the three models.

This virtual experiment resulted in $31 \times 4 \times 2 \times 20 \times 6 \times 3 = 89{,}280$ simulation units for each of the three models in ACME, meaning a total of 267 840 simulations.

The climate data (6 localities x 20 years x 365.25 days per year = 43,830 daily observation of rainfall, solar radiation, temperature, and potential evapotranspiration) was imported from global climate data servers (Chirps and AgERA5 data, https://data.apps.fao.org/catalog/dataset/global-weather-for-agriculture-agera5, https://www.chc.ucsb.edu/data/chirps) into the relevant daily climate data table of MasterInputOutput 'RAClimateD'. The list of the six locations with their coordinates was entered into the table 'Coordinates' linked one to many with RAclimateD. A list of the 31 sowing dates was created in the table 'IncrementSD', in the form of a list of values (0, 15, 30… to 30×15) that were added to the first sowing date considered in the analysis. A table 'YearList' lists the 20 simulated years. The four soils were added to tables 'Soil' linked one-to-many to 'SoilLayers', and their 29 parameters were typed manually. The three production situations were specified in the table 'SimulationOptions'.

The tables containing the lists of factor levels for the virtual

Table 1
Time (and size of inputs/outputs) required to perform the virtual experiment on sowing dates in Tanzania with parallel processing on a single computer.

		Full virtual experiment (89,280 Simulations)			*1 simulation*		
Task/information	Unit	DSSAT	STICS	Celsius	DSSAT	STICS	Celsius
Build model inputs	(hh:mm)	03:56	07:43	00:03:10			
	(*sec*)	14,160	27,780	190	0.1586	0.3112	0.0021
Size before simulation	Go (full virtual experiment) / Ko (1 simulation)	4.66	6.05		54.7308	71.0561	
Simulation	(hh:mm)	03:41	11:04	03:03			
	(*sec*)	13,260	39,840	10,980	0.1485	0.4462	0.1230
Size after simulation	Go (full virtual experiment) / Ko (1 simulation)	37	52.3	0.75	434.5577	614.2532	8.8086
Harmonize outputs	(hh:mm)	00:05:45	00:03:10	00:01:20			
	(*sec*)	345	190	80	0.0039	0.0021	0.0009
Total time all steps	(hh:mm)	07:42:45	18:50:10	03:07:30	0.3110	0.7595	0.1260
	(*sec*)	27,765	67,810	11,250	0.3110	0.7595	0.1260

experiment were manipulated using SQL queries to generate the full set of 44,640 combinations of these levels, in the two following steps (where a field containing "id" as a prefix is a unique record identifier in a table):

Step 1: generate records in the table 'CropManagement' from a single record typed into that table, to copy identical crop management parameters except the sowing date that varies with the values in the field 'Increment' in the table 'IncrementSD':

INSERT INTO CropManagement ([list of all the fields except that of the sowing date])
SELECT (a text identifier combining the identifier of CropManagement with the identifier of IncrementSD) AS id, CropManagement. [sowingdate] + IncrementSD. [increment] AS SD, [list of all the other fields with values kept identical across all CropManagement records] FROM CropManagement, IncrementSD

Step 2: generating the 44,360 simulation units in the table SimUnitList designed for this purpose, linked one-to-one (e.g. CropManagement, coordinates, soil) or one-to-many (e.g. RaClimateD) with all the other tables containing model parameters. This was done by crossing CropManagement with YearList, Coordinates, Soils and SimulationOptions in a SQL query:

INSERT INTO SimUnitList (list of all fields) SELECT (a composite text identifier combining the identifiers of all the tables) as Id.SimUnitList, SimulationOptions.IdOptions, Coordinates. idPoint, CropManagement.idMangt, Soil.IdSoil, YearList.year AS andeb, …, FROM CropManagement, Soil, SimulationOptions, Coordinates

Stics, Celsius and Dssat were run one after the other. Table 1 shows the computing time per model, using a standard PC laptop.

Fig. 2 shows typical simulation results, as plotted with a plotting software using as input simple queries applied to the SummaryOutputs table built by dataMill into MasterInputOuput at the end of the simulations of each model.

4. Impact

Virtual experiments with crop models pursue the general objective of exploring and assessing the sustainability of cropping systems and their management across a range of environments. The focus of these virtual experiments is increasingly on climate-related risk and the impact of climate change. They are powerful tools for the design of innovation in crop production systems.

However, the computing time for running the models and the time needed for processing input and output data strongly constrain the potential of quantitative systemic approaches. The assessment or design of cropping systems using models is usually undertaken by large teams that combine agronomists, data scientists and computer engineers. While such teams are common in developed countries, the participation of scientists from low- and middle-income nations in sustainability assessments has been limited due to a lack of resources and skills. Our package considerably alleviates the constraints of preparing crop model simulations and output processing. Consequently, it allows to substantially expand the size of virtual experiments that are manageable by scientists using a PC. We expect this will be especially useful to small, emerging teams of low- and middle-income nations. These teams often address region-specific research questions that are also critical to inform global sustainability issues, but need quantitative crop modelling to answer them. These research questions that can be further explored thanks to our software revolve around sustainable intensification of cropping systems, climate change adaptation and mitigation in the context of smallholder farmers. What are the relevant plot-level technical options that contribute to sustainable intensification and adaptation to climate change? Current studies addressing this type of question often rely on the use of one crop model (e.g. [11,12] and neglect the

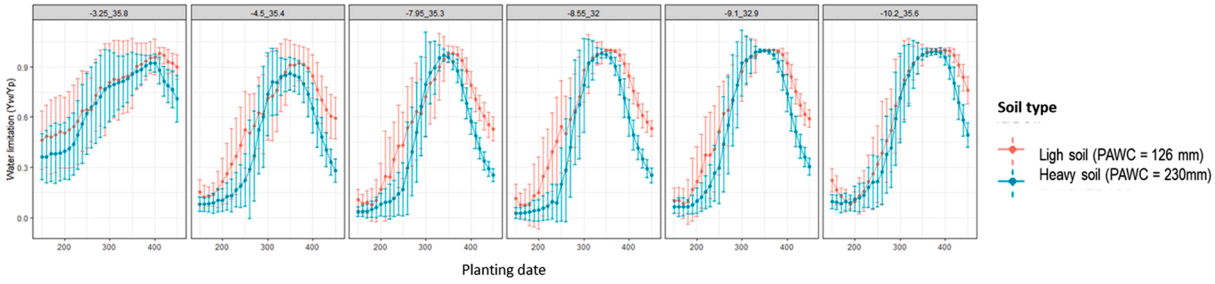

Fig. 2. Average simulated (20 years, model ensemble only) ratio of water-limited yield to potential yield (Yw/Yp) for six locations (the title of the panel gives the latitude and longitude) and 31 sowing dates in Tanzania. Vertical bars are standard deviations attributed to the different simulated years. Model simulations start 150 days before planting so that the first planting date is on the day of the year 150. The locations are in the southern hemisphere with a rainy season expanding from one year to the following one. Sowing dates are expressed in the number of days since January 1st of the first year of the rainy season. The last planting date has a value of 450 which correspond to model initialisation on the day of the year 300 and a planting date on day 85 of the second year of the season when the first year is not a leap year.

uncertainty that arises from the use of different models. When the studies include large model ensemble, they are usually limited in scope due to the time required to coordinate modelling teams scattered around the world. For example, [10] explored model uncertainty when simulating maize response to individual climate variables, but the study could not include the exploration of climate change projections. We believe that our software can help individual users and small teams to compare crop models and improve their structure and calibration, thus increasing the relevance of the study and our understanding of sustainability issues.

So far, the package has been used mostly by those who conceived and developed it. The software has been used so far for two virtual experiments that are the basis of three publications that are in the process of submission, one on the risk related to maize intensification in central Senegal, another on the risk related to delayed planting in Mozambique, and the third one on comparing existing crop calendars with optimal sowing dates determined with ACME in Zimbabwe and Zambia. One of the authors of this paper was not involved in software conception and development but successfully mastered the package after a few remote meetings with the developers. Subsequently, she designed and successfully performed a large virtual experiment, and contributed to improvements of the software.

5. Conclusions

The use of databases facilitates the exploitation of the results as well as the archiving and reuse of virtual experiments. The ACME software package helps analyse models input and output variables at low cost and facilitates their understanding. The software package eliminates the need for tedious data entry, helps agronomists to focus on their research objective when designing innovative cropping systems, and allows rapid refinement of virtual experiments to test specific hypotheses. Our current efforts beyond the current version of the package aim at i) adapting the package to High Performance Computing under Unix/Linux operating systems, to allow large virtual experiments and simulations spatialized over large geographic areas such as the African continent, ii) including new models in the list of models managed by ACME, iii) increase the number of parameters handled by MasterInputOutput (and thus decrease the number of those fixed at a default value in the native file system of each model as generated by DataMill). For example, the share of stable organic matter is left at its default value in the different models. There is therefore an opportunity to expand the variables handled by ACME and build the connections between model-specific inputs and ACME inputs with the regard to the simulation of soil organic matter mineralization.

CRediT authorship contribution statement

Michel GINER: Writing – review & editing, Writing – original draft, Software, Methodology, Formal analysis, Conceptualization. **Francois AFFHOLDER:** Writing – review & editing, Writing – original draft, Supervision, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Gatien N. FALCONNIER:** Writing – review & editing, Data curation. **Vimbayi Grace Petrova CHIMONYO:** Writing – review & editing, Data curation. **Louise LEROUX:** Writing – review & editing. **Sandrine AUZOUX:** Writing – review & editing, Formal analysis. **Myriam ADAM:** Writing – review & editing. **Patrice KOUAKOU:** Writing – review & editing, Validation. **Romain LOISON:** Validation. **Cyrille A. MIDINGOYI:** Software. **Erik CHAVEZ:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data can be found here https://www.github.com/GinerM/ACME-Datamill/tree/main/repo/ACME_EspaceDeTravail.

Acknowledgements

The development of the software was supported by the European Institute of Innovation & Technology (EIT), a body of the European Union, under the Climate-Kic project Agriculture Resilient Inclusive and Sustainable Enterprise (ARISE), task EIT_2.2.28_200275_P383_1B.

References

[1] Affholder F, Tittonell P, Corbeels M, Roux S, Motisi N, Tixier P, et al. Ad hoc modeling in agronomy: what have we learned in the last 15 years? Agron J 2012; 104:735–48. https://doi.org/10.2134/agronj2011.0376.
[2] Reynolds JF, Acock B. Predicting the response of plants to increasing carbon dioxide: a critique of plant growth models. Ecol Modell 1985;29:107–29. https://doi.org/10.1016/0304-3800(85)90049-3.
[3] Tao F, Rötter RP, Palosuo T, Díaz-Ambrona CGH, Mínguez MI, Semenov MA, et al. Contribution of crop model structure, parameters and climate projections to uncertainty in climate change impact assessments. Glob Chang Biol 2018;24: 1291–307. https://doi.org/10.1111/gcb.14019.
[4] Li L, Wang B, Feng P, Jägermeyr J, Asseng S, Müller C, Macadam I, et al. The optimization of model ensemble composition and size can enhance the robustness of crop yield projections. Commun Earth Environ 2023;4:1–11. https://doi.org/10.1038/s43247-023-01016-9.
[5] Rosenzweig C, Jones JW, Hatfield JL, Ruane AC, Boote KJ, Thorburn P, et al. The Agricultural Model Intercomparison and Improvement Project (AgMIP): protocols and pilot studies. Agric For Meteorol 2013;170:166–82. https://doi.org/10.1016/j.agrformet.2012.09.011.
[6] Rötter RP, Carter TR, Olesen JE, Porter JR. Crop–climate models need an overhaul. Nat Clim Chang 2011;1:175.
[7] Jones JW, Hoogenboom G, Porter CH, Boote KJ, Batchelor WD, Hunt LA, Wilkens PW, et al. The DSSAT cropping system model. Eur J Agron 2003;18: 235–65. https://doi.org/10.1016/S1161-0301(02)00107-7.
[8] Brisson N, Gary C, Justes E, Roche R, Mary B, Ripoche D, et al. An overview of the crop model STICS. Eur J Agron 2003;18:309–32. https://doi.org/10.1016/S1161-0301(02)00110-7.
[9] Ricome A, Affholder F, Gérard F, Muller B, Poeydebat C, Quirion P, et al. Are subsidies to weather-index insurance the best use of public funds? A bio-economic farm model applied to the Senegalese groundnut basin. Agric Syst 2017;156: 149–76. https://doi.org/10.1016/j.agsy.2017.05.015.
[10] Falconnier GN, Corbeels M, Boote KJ, Affholder F, Adam M, MacCarthy DS, et al. Modelling climate change impacts on maize yields under low nitrogen input conditions in sub-Saharan Africa. Glob Chang Biol 2020;26:5942–64. https://doi.org/10.1111/gcb.15261.
[11] Ranaivoson L, Falconnier GN, Affholder F, Leroux L, Autfray P, Muller B, et al. Can green manure contribute to sustainable intensification of rainfed rice production in Madagascar? Field Crops Res. 2022;289:108711. https://doi.org/10.1016/j.fcr.2022.108711.
[12] Traore A, Falconnier GN, Couëdel A, Sultan B, Chimonyo VGP, Adam M, et al. Sustainable intensification of sorghum-based cropping systems in semi-arid sub-Saharan Africa: the role of improved varieties, mineral fertilizer, and legume integration. Field Crops Res. 2023;304:109180. https://doi.org/10.1016/j.fcr.2023.109180.