

# AquaCrop.jl: A Process-Based Model of Crop Growth

Gabriel Díaz Iturry<sup>1,2</sup>, Marco Matthies<sup>1,2</sup>, Guy Pe'er<sup>1,2</sup>, and Daniel Vedder<sup>1,2,3</sup>

<sup>1</sup> Helmholtz Centre for Environmental Research - UFZ <sup>2</sup> German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig <sup>3</sup> Friedrich-Schiller-Universität Jena ¶ Corresponding author

DOI: 10.xxxxxx/draft

## Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Open Journals](#)

## Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#))

## Summary

All agriculture is dependent on the growth of plants. Crop plants provide food for humans, fodder for domestic animals, and fibre and other resources for our manufacturing economy. Therefore, understanding how plants grow under different conditions is important not just for farmers themselves, but also for the rest of society. Crop models based on physical and physiological processes use information about environmental parameters (e.g. temperature, rainfall, soil quality) and knowledge of plant biology to simulate how crop plants grow over time and estimate the resulting yield. Such models can be used to optimise farm management, forecast regional or national yields, or study climate change impacts. Here we present an expanded reimplementations of the AquaCrop model in Julia, in order to make it accessible to a wider group of researchers.

## Statement of need

AquaCrop.jl is an independent Julia translation of the AquaCrop model (version 7.2), originally developed by the FAO (Steduto et al., 2009). This is a well-established crop growth model that has been used to model numerous crops worldwide (Mialyk et al., 2024), and is known to produce reliable estimates of crop phenology and yield (Kostková et al., 2021).

AquaCrop is already available in multiple languages. First implemented in Delphi, it was later open-sourced in a Fortran version (de Roos et al., 2021; Remote Sensing and Data Assimilation Research Group, 2021/2024). There are also versions available in Matlab, Python, and R, although these are not up-to-date with the most recent version of the original model (Camargo Rodriguez & Ober, 2019; Foster et al., 2017; Kelly & Foster, 2021). With AquaCrop.jl, we want to expand this portfolio to make the model more easily accessible to the growing number of environmental modellers working with Julia.

Beyond just adding another language, our purpose is also to provide a package that can be readily integrated into other scientific software. Recent research has emphasised the need for the creation of interdisciplinary models that consider the multiple processes inherent in global challenges such as climate change or biodiversity loss (Cabral et al., 2023). This will require the use of model coupling, and the adaptation of existing models to be usable as components in integrated models (Vedder, Fischer, et al., 2024).

Specifically, we developed the package to use it as a component within Persefone.jl, a process-based model of agricultural ecosystems (Vedder, Matthies, et al., 2024). The aim of this model is to study the impact that agricultural management and policy has on biodiversity, for which the growth of crop plants is an important mediating factor.

## Comparison to original implementation

The core code of `AquaCrop.jl` closely follows the FAO's Fortran implementation, which allows us to quickly integrate changes and updates to the original AquaCrop code. The code was translated verbatim as much as possible, leaving all function and variable names as they are. To maintain compatibility, we also support the original (highly idiosyncratic) input file formats. The correctness of our code is tested by comparing its output with that of the original software, to ensure that it produces the same results.

Whereas the original software was primarily designed as a stand-alone model, our purpose is to use `AquaCrop.jl` as an integrated component. Therefore, we developed a wrapper layer with an API that improves the package's interoperability. First, we added support for standardised input and output file formats (TOML and CSV), and for loading input data from memory rather than disk (for example using output from a coupled model). Second, we bundled all state variables for a simulation in one struct (`AquaCropField`), thereby eliminating global state and allowing multiple simulations to be carried out in parallel (as well as making serialisation and data transfer easier). Third, and most importantly, we enabled the model to be run interactively. Rather than simply being executed as a batch job, the simulation can now be advanced one day at a time, and state variable inspected and changed on the go. This makes it possible to use the package for dynamic, bidirectional model coupling.

Overall, our aim was to leave the scientific core of the model unchanged, but to make it as easy as possible for environmental modellers using Julia to integrate into their own software. One side benefit of this is that our package can of course be complemented with other libraries from the Julia ecosystem, giving access to other high-quality research software such as the `EcoJulia` collection, and utility packages such as `Makie.jl`, `StatsModels.jl`, or `Optimisers.jl`. Another benefit is that modellers who wish to expand or otherwise modify the scientific core of `AquaCrop` can now do so in Julia rather than Fortran, if they are unfamiliar with the latter.

## Examples

The following tutorials are provided in the documentation:

- [Basic run](#) shows how to set up a model run using the original AquaCrop configuration file format, simulating a complete vegetation period in one go.
- [Intermediate run](#) shows how to configure the model with TOML and CSV files, how to advance the simulation one day at a time, and how to access crop variables at run time.
- [Advanced run](#) shows how to configure the model via the API (i.e. using Julia variables), and how to change variables at run time.

A simple demonstration of a basic run using the data from the `AquaCrop.jl/test/testcase` directory is shown here:

```
using AquaCrop
using CairoMakie
using Unitful
```

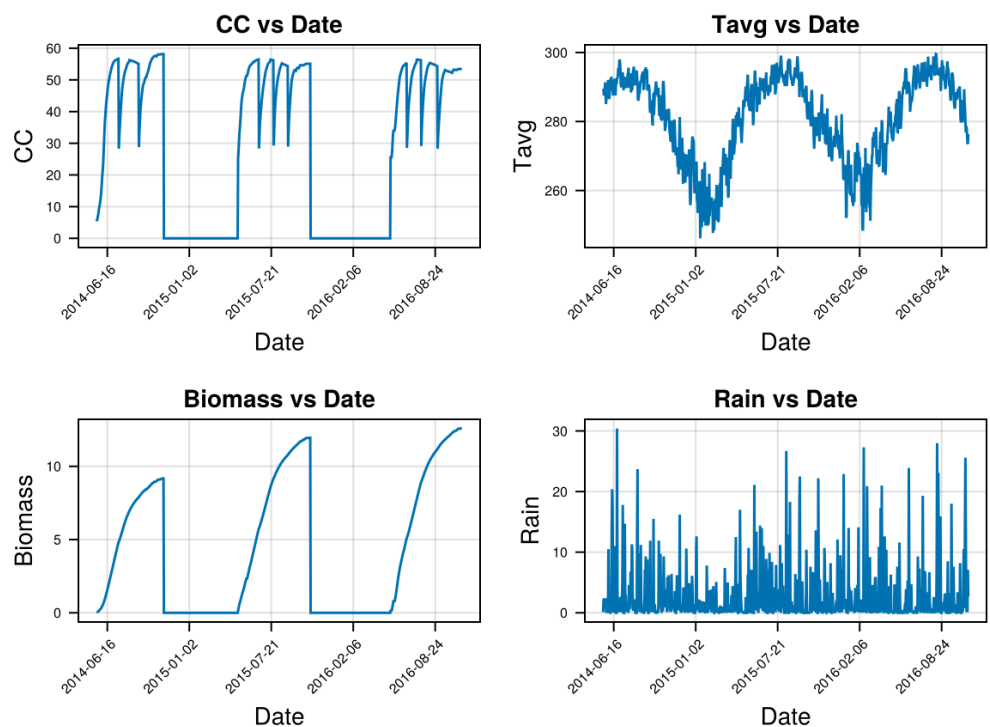
```
# First, we specify the input file format:
# NormalFileRun(): use the original AquaCrop file format
# TomlFileRun(): use TOML and CSV formatting
# NoFileRun(): provide input data via the API
runtype = NormalFileRun();
```

```
# Then specify the directory containing the necessary input files
parentdir = AquaCrop.test_dir; # "../AquaCrop.jl/test/testcase"
```

```
# Now we can do a simulation run and plot the results
outputs = basic_run(; runtype=runtype, parentdir=parentdir);

function plot_basic_out(cropfield, cols)
    x = cropfield[!, "Date"]
    aux_sz = round{Int, sqrt(length(cols))}
    f = Figure()
    for (i, coli) in enumerate(cols)
        ii, jj = divrem(i-1, aux_sz)
        ax = Axis(f[ii, jj],
            title = coli*" vs Date",
            xlabel = "Date",
            ylabel = coli
        )
        lines!(ax, x, ustrip.(cropfield[!, coli]))
        ax.xticksize = 8
        ax.yticksizelabelsize = 8
    end
    return f
end

f = plot_basic_out(outputs[:dayout], ["CC", "Tavg", "Biomass", "Rain"])
```

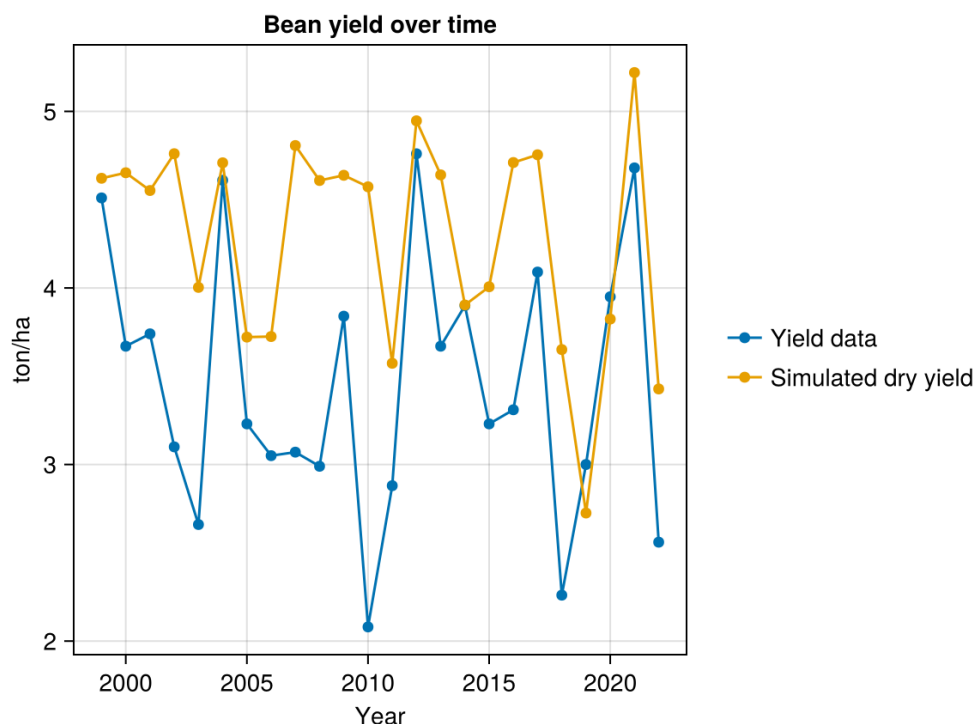


**Figure 1:** Simulated Canopy Cover (CC) and Biomass of crops over time in a generic simulation run. We also show the average temperature (Tavg) and rain data

74 The resulting graph is shown in Figure 1. Biomass increases over the growing season, is then

75 reset to 0 with the harvest, and then sown again the following spring.

76 **Figure 2** displays the results a more realistic use case. Here, AquaCrop.jl was used to simulate  
77 the growth of beans (*Vicia faba*) based on environmental data from Thuringia, Germany, with  
78 historical yield data shown for comparison. This showcases the high predictive accuracy that  
79 can be achieved with AquaCrop.



**Figure 2:** Simulated yield of beans (*Vicia faba*) compared to observed yields in Thuringia, Germany.

## Conclusion

81 We present AquaCrop.jl as an up-to-date addition to the portfolio of AquaCrop implemen-  
82 tations. To our knowledge, this is the first process-based crop growth model available in  
83 Julia. By focussing on improving interoperability, we aim to contribute to the creation of  
84 interdisciplinary, integrated models in the environmental sciences.

## Acknowledgements

86 GDI, MM, and DV are funded through the project CAP4GI by the Federal Ministry of Education  
87 and Research (BMBF), within the framework of the Strategy, Research for Sustainability  
88 (FONA, [www.fona.de/en](http://www.fona.de/en)) as part of its Social-Ecological Research funding priority, funding  
89 no. 01UT2102A. Responsibility for the content of this publication lies with the authors. MM,  
90 GP, and DV gratefully acknowledge the support of iDiv, funded by the German Research  
91 Foundation (DFG-FZT 118, 202548816).

## References

- 92
- 93 Cabral, J. S., Mendoza-Ponce, A., da Silva, A. P., Oberpriller, J., Mimet, A., Kieslinger, J.,  
 94 Berger, T., Blechschmidt, J., Brönnner, M., Classen, A., Fallert, S., Hartig, F., Hof, C.,  
 95 Hoffmann, M., Knoke, T., Krause, A., Lewerentz, A., Pohle, P., Raeder, U., ... Zurell, D.  
 96 (2023). The road to integrate climate change projections with regional land-use-biodiversity  
 97 models. *People and Nature*, n/a(n/a). <https://doi.org/10.1002/pan3.10472>
- 98 Camargo Rodriguez, A. V., & Ober, E. S. (2019). AquaCropR: Crop Growth Model for R.  
 99 *Agronomy*, 9(7, 7), 378. <https://doi.org/10.3390/agronomy9070378>
- 100 de Roos, S., De Lannoy, G. J. M., & Raes, D. (2021). Performance analysis of regional  
 101 AquaCrop (v6.1) biomass and surface soil moisture simulations using satellite and in situ  
 102 observations. *Geoscientific Model Development*, 14(12), 7309–7328. <https://doi.org/10.5194/gmd-14-7309-2021>  
 103
- 104 Foster, T., Brozović, N., Butler, A. P., Neale, C. M. U., Raes, D., Steduto, P., Fereres,  
 105 E., & Hsiao, T. C. (2017). AquaCrop-OS: An open source version of FAO's crop water  
 106 productivity model. *Agricultural Water Management*, 181, 18–22. <https://doi.org/10.1016/j.agwat.2016.11.015>  
 107
- 108 Kelly, T. D., & Foster, T. (2021). AquaCrop-OSPy: Bridging the gap between research  
 109 and practice in crop-water modeling. *Agricultural Water Management*, 254, 106976.  
 110 <https://doi.org/10.1016/j.agwat.2021.106976>
- 111 Kostková, M., Hlavinka, P., Pohanková, E., Kersebaum, K. C., Nendel, C., Gobin, A., Olesen,  
 112 J. E., Ferrise, R., Dibari, C., Takáč, J., Topaj, A., Medvedev, S., Hoffmann, M. P., Stella,  
 113 T., Balek, J., Ruiz-Ramos, M., Rodríguez, A., Hoogenboom, G., Shelia, V., ... Trnka,  
 114 M. (2021). Performance of 13 crop simulation models and their ensemble for simulating  
 115 four field crops in Central Europe. *The Journal of Agricultural Science*, 159(1-2), 69–89.  
 116 <https://doi.org/10.1017/S0021859621000216>
- 117 Mialyk, O., Schyns, J. F., Booij, M. J., Su, H., Hogeboom, R. J., & Berger, M. (2024). Water  
 118 footprints and crop water use of 175 individual crops for 1990–2019 simulated with a global  
 119 crop model. *Scientific Data*, 11(1), 206. <https://doi.org/10.1038/s41597-024-03051-3>
- 120 Remote Sensing and Data Assimilation Research Group. (2024). *AquaCrop repository* (Version  
 121 7.2). KU Leuven. <https://github.com/KUL-RSDA/AquaCrop> (Original work published  
 122 2021)
- 123 Steduto, P., Hsiao, T. C., Raes, D., & Fereres, E. (2009). AquaCrop—The FAO Crop Model  
 124 to Simulate Yield Response to Water: I. Concepts and Underlying Principles. *Agronomy*  
 125 *Journal*, 101(3), 426–437. <https://doi.org/10.2134/agronj2008.0139s>
- 126 Vedder, D., Fischer, S. M., Wiegand, K., & Pe'er, G. (2024). Developing multidisciplinary  
 127 mechanistic models: Challenges and approaches. *Socio-Environmental Systems Modelling*,  
 128 6, 18701. <https://doi.org/10.18174/sesmo.18701>
- 129 Vedder, D., Matthies, M., & Pe'er, G. (2024). *Persefone.jl* (Version 0.6.0). German Centre of  
 130 Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig. <https://persefone-model.eu/>