Anonymous Referee #1

This paper presents a nice example of combining theory based models and machine learning to efficiently identify parameters of an ecosystem model, exploiting observation data recorded at multiple sites. The approach is valid and the results are interesting. However, the documentation of data and methods is currently deficient on a level that makes it hard to grasp the main messages and interpret the results. Section 2 of the paper does in my yes require a thorough revision, including new explanatory figures, restructuring and replacement of text blocks. For this reason I recommend a major revision or rejection with an invitation to resubmit.

Thank you for your evaluation!

1. Major comments

1. I assume a key point of the developed framework is that it enables to directly backpropagate from the outputs through the model equations to the neural networks. This is not clear from the paper at all. Much of the framework description seems like you feed NN predictions of parameters through a black box physics-based model, which is a standard approach. I suggest a dedicated subsection, possibly including a figure, to clarify this detail.

Yes, the differentiability which supports gradient-based optimization is the soul of the proposed work. We have discussed this in the paper (Abstract: "*programmatically differentiable (meaning gradients of outputs to variables used in the model can be obtained efficiently and accurately)…*", lines 146 "*In order to train the physical equations and neural networks together using gradient descent, the above equations were implemented on differentiable platforms to support backpropagation*"). To further emphasize it, we will add a paragraph at the beginning of section 2.1 (General overview) which explains Figure 1 to emphasize it. Also Figure 1 will be modified to represent both the forward run (blue arrows) and the backpropagation (black arrows) and thus better represent the framework (shown below).

"*Our general framework trains connected neural networks to provide parameters (and later process representations) to process-based models (PBM), in this case the photosynthesis module in FATES, on all the training data points simultaneously (Figure 1a). The neural networks maps from some raw inputs to some tuneable physical parameters (θ) (later extensible to processes) required for the PBM. The predicted physical parameters are then fed into the differentiable PBM along with other required forcing variables (F) and untuned constant attributes (θ_c) to compute the simulated target variable (y_{sim}) which is compared with observations to compute a loss function. The forward run starts from the neural networks and ends at the loss function (blue arrows in Figure 1a). We then backpropagate the errors (shown by black arrows in Figure 1a) through the PBM equations back to the neural networks to train them. To support gradient-based training, the entire framework must be differentiable [Shen et al., 2023] and neither the neural network nor the process-based model is a black box --- they both allow explicit inspection and modification of the internal structures. We had to reimplement the photosynthesis module of FACETS on differentiable platforms.*"

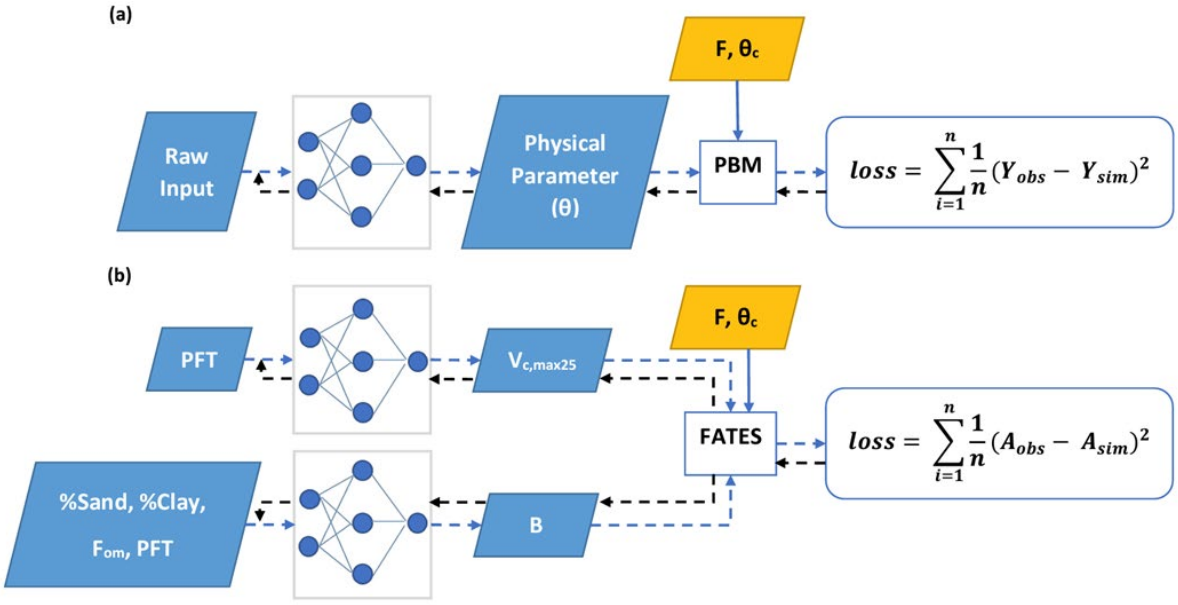


Figure 1. Diagram showing the differentiable parameter learning (dPL) framework which is a hybrid of neural networks and the photosynthesis module in the FATES ecosystem model written on a differentiable platform. (a) The generic workflow: Some raw information is mapped into physical parameters via a neural network. These parameters are sent into a process-based model (PBM), which then outputs variable Y that is compared with observations. Direct supervision for the physical parameters is not required -- we do not need ground truth for these parameters. The loss function is "global" in that it involves all training data points, rather than being computed site-by-site as done in traditional calibration. (b) The workflow for the computational example described in this work. We estimate either $V_{c,max25}$ or the parameter B using neural networks, or both of them at the same time. When they were not estimated from data, default values from the literature were used. Blue arrows show running the neural networks with the PBM in a forward mode, while black arrows indicate backpropagation from the loss function back through the differentiable model equations to the neural networks to update their weights.

2. The datasets used for training and testing are not properly documented. We don't know how many datapoints are included over which time periods. The random holdout suddenly appears in the results, and in general we don't know how training/validation/testing splits are defined.

This paragraph will be added to section **(Synthetic data and real data experiments)** to explain more about the temporal and the random holdout tests as well as data splitting.

"*For training and testing our candidate models, two different tests were performed with respect to data splitting: random holdout test and temporal holdout test, the latter of which stresses the models' ability to project into the future. In the temporal holdout test, for each PFT in each location, the available dates of measurements were counted where data points measured at the older 80% of these dates were used for training and the other more recent 20% were used for testing. Due to the randomness of dates of measurements available at each location (as mentioned previously in section 2.4.1), the temporal periods for the training and testing datasets vary by location. The temporal holdout test was used for both synthetic and real data experiments. For the random holdout test, as the name implies, 80% of the datapoints were randomly selected for training from the available PFT measurements in each location while the rest were used for testing. This test was run only for the real case experiments.*"
--- we will in fact change the train:test ratio to 80:20 and run a cross validation (actually--- this is really easy and we already did it, see results later).

CLM4.5 standard parameters play a central role in the results, but we know nothing about where they come from / how they are defined and if, for example, all or a subset of values are used for comparison.

https://opensky.ucar.edu/islandora/object/technotes%3A515/datastream/PDF/view
CM4.5 documentation presents the standard values of the parameters and the equations that we used in this study as a benchmark. In fact we compared to many other models in Table 3 and provided references (previously in the text and will be in the table itself). We will also add subsection (shown below) to section **(Input and observation datasets)** to better clarify.

"***CLM4.5 default parameters"***
CLM4.5 documentation provided reference values and equations for both parameters $V_{c,max25}$ and B. For $V_{c,max25}$, the values corresponding to each PFT are well documented in CLM4.5 (section 8; table 8.1) (Oleson et al., 2013) and are shown in Table 3 in the manuscript. The same applies for parameter B with the default equations shown in Appendix A. CLM4.5 was also used to provide other photosynthetic parameters such as the soil matric potentials for closed stomata ψ_c and open stomata ψ_o (see Equation 9), and the plant root distribution parameters (see Equation 8) required for β_t computations where all these parameters are considered as PFT-dependent."

3. The explanation of the ecosystem model suffers from a clear struggle between trying not to include the entire set of equations in the paper, while providing sufficient detail. For me the level of detail provided in the paper was actually confusing, because it required constant looking up in the appendix to understand the context, distracting from the main messages. I think a way out could be to include a figure that summarizes the main blocks of the model (including what parts correspond to f1 and f2), include only the changed equations in the paper, and otherwise keep the full model description in the appendix. On a sidenote: is f2 not the same as an observation equation, that is commonly used in state space models?

We plan to add the figure below which show the block equations corresponding to f1 and f2 equations respectively. Yes, f_2 is the observation equation. f_1 and f_2 may share common components but they are mathematically different: f1 is a system constraint while f2 is a "observation equation". In this example, f1 is solved for the unknown c_i while f2 connects c_i to the observation A_n.

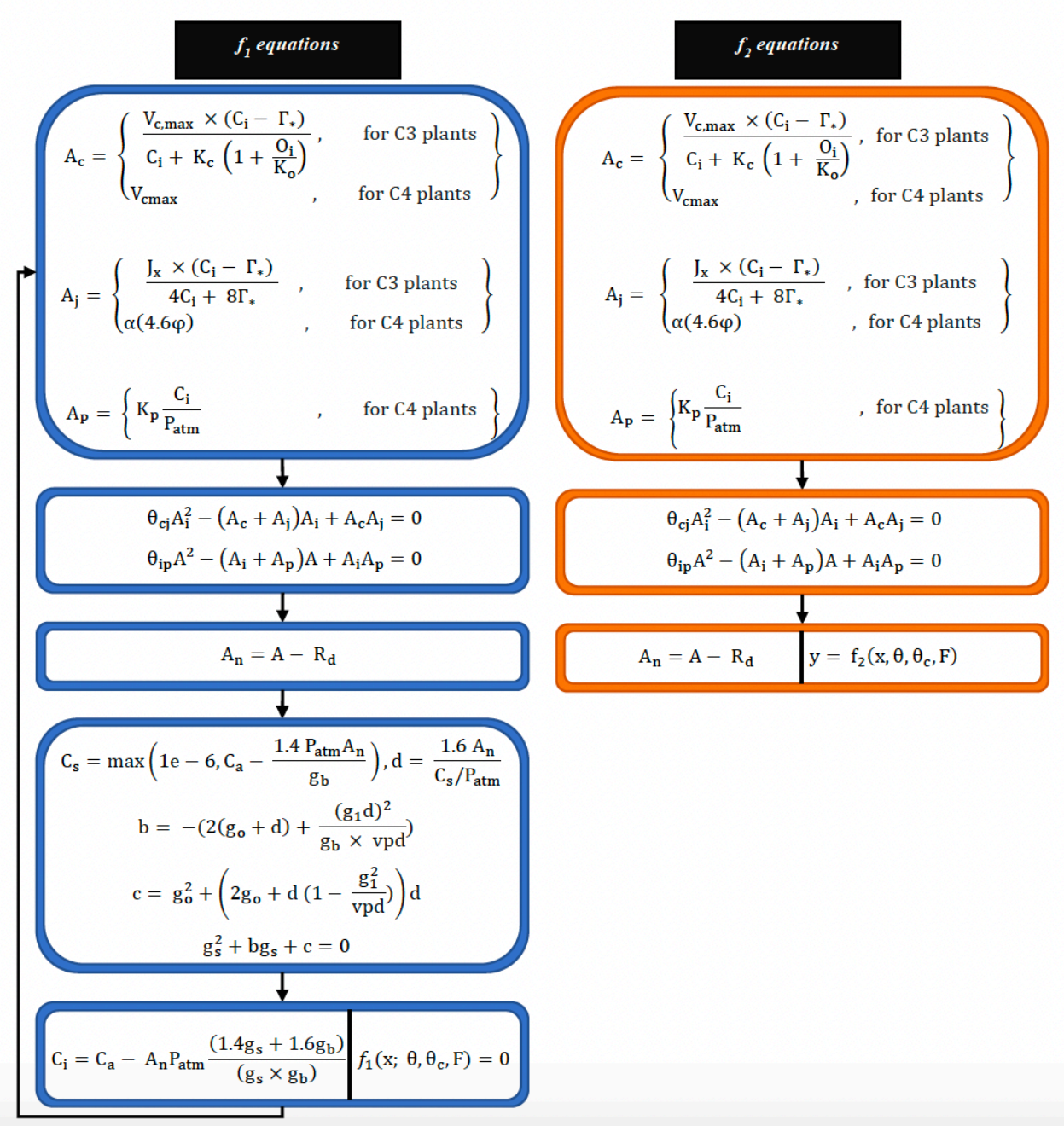


Figure 2. showing the model block of equations corresponding to f1 and f2. Blue boxes refer to equations corresponding to f1. Orange boxes refer to equations corresponding to f2. Further details about the variables and parameters in these equations will be given in a separate table. Once we get the solution for Ci (intercellular leaf CO2 pressure) from f1 equations (nonlinear system), we can run f2 equations to get An (net photosynthesis rate)

4. Details on hyperparameters (neural network # of layers, activation functions, learning rates etc.) are not provided at all. Some key information should be provided in the paper, and a reference to supporting information or the code should be provided for details.

This paragraph will be added to section **(Synthetic data and real data experiments)** which states some details about the hyperparameters

"*The used MLPs were very simple with only three layers; input layer, one hidden layer, and an output layer. To ensure an output value between 0 to 1 for both $V_{c,max25}$ and B parameterizations, sigmoid activation functions were used for both hidden and output layers. The quantity of available data posed a limitation and did not permit an extensive hyperparameter tuning experiment with a train/validation/test split. Hence, we employed a lazy trial and error with hyperparameters (learning rates and hidden size) using 70% of the randomly selected data as training data and 30% as a validation set, just to ensure we had a roughly performing hyperparameter set. We selected a learning rate of 0.01 and a hidden size that is equal to the number of inputs (9 for the NN_V and 8 for the NN_B). We kept the same hyperparameters in the reporting, where we ran 5-fold cross validation. In addition, we found that moderately perturbing the hyperparameters resulted in very little change in the performance. This design considered the practical limits of available data, even this study already represents a large-sample study in the domain of ecosystem modeling.*"

You can see, from the table below, moderate changes to the hiddensize does not matter too much. Thus, due to data limitation, we did not tune hyperparameter extensively. We simply use a hiddensize that is equal to the number of inputs. Should there be more data, we can certainly use a train/validation/test split and run more hyperparameter tuning.

	Corr		**RMSE (µmol m-2 s-1)**		**Bias (µmol m-2 s-1)**		**NSE**		
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	
V+B	0.7994	0.7478	4.3002	4.4255	0.0476	0.3618	0.6379	0.5313	NN_B[8,6,1]
V+B	0.7984	0.7473	4.3105	4.4281	0.0416	0.3569	0.6362	0.5308	NN_B [8,7,1]
V+B	0.7994	0.7479	4.3003	4.4232	0.0376	0.3467	0.6379	0.5318	NN_B [8,8,1]
V+B	0.7972	0.7445	4.3211	4.4358	0.0251	0.3001	0.6344	0.5291	NN_B [8,9,1]
V+B	0.7989	0.7474	4.3053	4.4320	0.0420	0.3601	0.6371	0.5299	NN_B [8,8,8,1]

Detailed comments

line 61: nonuniqueness is also going to be a problem if we employ newer frameworks like PINNs or dPL

Agree that nonuniqueness will still remain an issue and will need to be tested/controlled, but it should be better with dPL than with previous site-by-site calibration approach, because one neural network is constrained by all data points. There is an implicit spatial constraint. This effect was demonstrated in fine details in Tsai et al., 2021. As shown in that paper, as we turn parameter calibration into parameter learning, the framework can generalize better in space and in uncalibrated variables. It's obviously a tricky issue between the available data we have, the amount of structure we specify, and the tradeoff between variance and bias. What we hope to achieve is to maximally leverage the available information.

line 110: it might be worthwhile to start with a reference to figure 1 and a down to earth explanation of the objective of your work, i.e. to calibrate model parameters across many sites, to capture the variation of parameters using neural networks, and to employ differentiable programming to speed up the identification process

This paragraph will be added to section **(General overview)**

As replied earlier, we added the new first paragraph in Section 2.1, General overview, about the overall framework and citing Figure 1.
The original first paragraph will be modified:
"*In this case, the process-based model is related to the photosynthesis module in FATES, which can be written as a nonlinear system of equations and its solution is implicit. The system can be written as:"*

line 118: please explain PFT again in this section

PFT will be replaced with the full description plant functional type and the whole text will be modified to:
"*Some of the tunable parameters are typically formulated as being Plant Functional Type (PFT)-dependent (e.g., the maximum carboxylation rate) where each PFT include group of plant species that share similar physical and phenological characteristics leading to similar interaction with the environment*"

line 140: If you preserve eq. 4 and 5 in the paper, I think they should be presented in reverse order (f1 first, f2 second)

Both equations will be reversed

line 146-164: please include only methodological descriptions that are relevant for the results. of the julia implementation was not used, then it should not be described and discussed

Thanks for the point and we do understand where the reviewer is coming from. While Julia was not the main tool for production here, we thought it might be useful to mention it because the SciML toolset, co-developed by two of the coauthors, may be valuable to ecosystem modelers. Moreover, it is formulated very differently in a novel symbolic format which is in fact quite interesting and could potentially lead to a different path, and the package is evolving rapidly. Hence we think preserving it has some value. Removing it will also mean removing some coauthors, which we do not want to do.

line 183: you don't describe anywhere in your data how many PFTs you consider. it is therefore here also not clear how many dummy variables this model receives as input.

Our dataset included 9 different PFTs categories, a paragraph with more details about Lin15 dataset will be added to subsection **(Forcing and Photosynthesis rates)** stating the number of PFTs considered plus the name of each PFT.

"*We refer to this dataset as Lin15 throughout the rest of this work with 43 sites chosen whose dates and times of measurements were available. Lin15 covered nine different PFT categories including the following: rainfed crop "Crop R", Broadleaf Evergreen Tree Tropical "BET Tropical", Broadleaf Evergreen Tree Temperate "BET Temperate", C3 grass, C4 grass, Needleleaf Evergreen Tree Boreal "NET Boreal", Needleleaf Evergreen Tree Temperate "NET Temperate", Broadleaf Deciduous Tree Temperate "BDT Temperate", and Broadleaf Deciduous Shrub Temperate "BDS Temperate". Measurements were taken on sub-hourly scale but not necessarily on a continuous daily interval. That's why for almost all the sites, data were available on some random days (not necessarily continuous) in one or a few years. Lin15 also contained meteorological forcing variables, including air temperature, atmospheric pressure, relative humidity, and radiation. Moreover, we used ERA5 to fill in for any missing forcing variables in Lin15.*"

line 190-205: I think this information is not needed to understand the main message

We think this information is important because we refer to it in different parts of the paper and they show brief description on how the soil water stress function (β_t) is calculated:

Line 190 – 195: show equation 7 which we later refer to as the equation to be replaced with equation 10 in the model changes section. Thus, we need to mention the old and the proposed equations.

Line 195 – 205: show the two final equations for calculating the soil water stress function (β_t) and the plant wilting factor (w_i), which we later refer to as part of the equations used in the synthetic and real data experiments after retrieving or estimating the parameter B

eq. 10: why is psi_max replaced by psi_0? (missing explanation)

Line (216 – 218), we stated the actual equations that we used in for computing ψ_i (in which ψ_{sat} was replaced with ψ_o).

In Appendix A, we kept all the original equations the same whether those related to FATES or to computing the soil water stress function (β_t).

Actual equation used in this study (Line 216 – 218)	Original equation (Appendix A)
$\Psi_i = \Psi_o \times S_i^{-B_i} \geq \Psi_c$	$\Psi_i = \Psi_{sat,i} \times S_i^{-B_i} \geq \Psi_c$

Reasons for this replacement:

In the original CLM4.5 equations, ψ_{sat} is based on empirical functions, percentage of sand *(%sand)*, and fraction of organic matter (F_{om}) (Equations A17 – A18). Using the original Equation 7 for computing ψ_i results in a plant wilting factor w_i equals to one for more than 90% of the data points across different soil layers.

To give the model more flexibility in the computation of ψ_i and thus allow more variability in w_i values, ψ_{sat} was replaced with ψ_o. However, to ensure having w_i values less than or equal 1 as in

the original equation 9, we tried to create equation 10 in a way that satisfies this condition using ψ_o. For parameter B (outputted from NN_B), it was restricted to be within the range 0 and 1 to satisfy the same condition as well. Applying those changes, we were able to get ψ_i values within the range of ψ_o and ψ_c while showing more variability in the computed w_i.

Also, we plan to add this paragraph to the **(Model changes)** section for clarification:

"*These changes were implemented to give more flexibility in the computation of the soil matric potential ψ_i. Using the original Equation 7 for computing ψ_i results in a plant wilting factor w_i equals to one for more than 90% of the datapoints across different soil layers. Thus, changing Equation 7 to the form shown in Equation 10 helped to express more variability in w_i and eventually in the computed soil water stress function (β_t).*"

Here, the point is to calculate photosynthesis. We can see clearly the modified model works very well for photosynthesis. The differentiable modeling approach was specifically designed to enable inspection of various modules and assumptions in the model to improve model performance. It is possible that alternative formulations can also perform well and we do not preclude that here, as this is not a main point of concern for this paper.

eq. 11: what is F_om?

F_om is the fraction of organic matter and this is mentioned here after equation 7 *"where ψ_{sat} is the saturated soil matric potential and S is the soil wetness, both defined for a specific soil layer. Different soil attributes such as percentages of sand (%sand) and clay (%clay), fraction of organic matter (F_{om}), and soil moisture (θ_{liq}) are used in computing ψ_{sat}, S, and B (Appendix A)."*

line 232: the CLM4.5 data points should be documented in a dedicated data section. In general, I suggest they you separate the description of data and experiments

https://opensky.ucar.edu/islandora/object/technotes%3A515/datastream/PDF/view

CM4.5 documentation clearly presents the standard values of the parameters and the equations that we used in this study. A subsection (shown below) will also be added to section **(Input and observation datasets)** to better clarify

"***CLM4.5 default parameters"***

CLM4.5 documentation played an important role in the results of this study by providing reference values and equations for both target parameters $V_{c,max25}$ and B. For $V_{c,max25}$, the values corresponding to each PFT are well documented in CLM4.5 (Oleson et al., 2013) and are shown in Table 3. The same applies for parameter B with the default equations shown in Appendix A. CLM4.5 was also used to provide other photosynthetic parameters such as the soil matric potentials for closed stomata ψ_c and open stomata ψ_o (see Equation 9), and the plant root distribution parameters (see Equation 8) required for βt computations where all these parameters are considered as PFT-dependent."

The description of data and experiments are already located in different subsections. In the new submission, we can move the data subsection prior to the experiments' description subsection for better clarification.

line 239: were all calculations performed only for the topsoil layer in all experiments?

This is valid for the synthetic case only whose purpose was just to test the whole framework, while for the real case all the five soil layers (mentioned in Static attributes subsection) were used to estimate the parameter B.

Table 1: missing symbol explanations for means and standard deviations

For clarification, this line will be added to the bottom of table 1
"*σ refers to the standard deviation, OBS refers to the mean of observations, SIM refers to the mean of simulations*"

line 383: please include time series for observations and model predictions

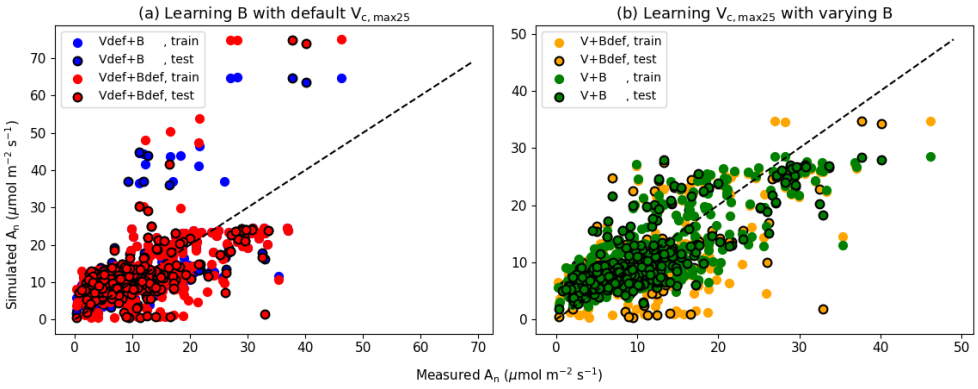
We did not provide time series for several reasons. Measurements in Lin15 dataset were taken on sub-hourly scale but not necessarily on a continuous daily interval. for almost all the sites, data were available on some random days (not necessarily continuous) in one or a few years. In fact, many of the measurement days are far from each other and we can barely find consecutive days for producing sensible time series. Second, this model was not posed as a time-continuous problem. In other words, there is no accumulated memory between different dates. Hence, we think time series plot may even be somewhat misleading.

fig. 5: symbols in legend cannot be distinguished. are results shown for the test dataset?

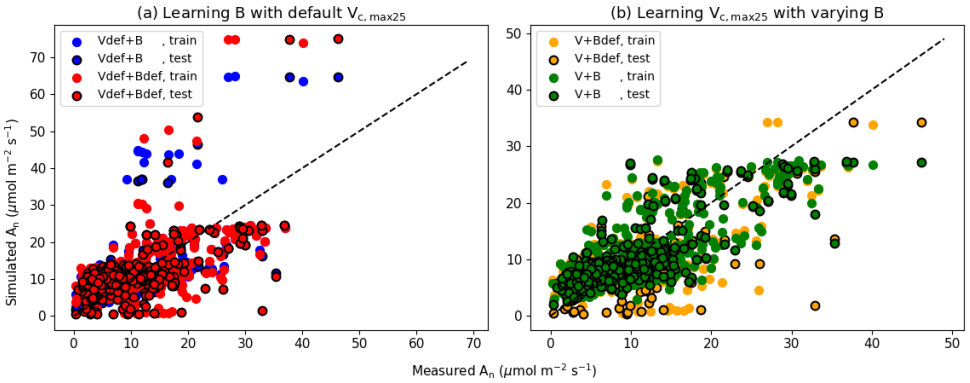
These points belong to both training and testing datasets. We previously have a version that distinguish train and test, as pasted below. As you can see, there are no visual differences between two types of points and such symbology does not really bring in new information. Later, we wanted to use symbols to indicate PFTs, which seems more informative. So, to avoid overcomplicating the figure, we removed the train/test differences. We also remind the reviewer that we will provide cross validation results in the revised manuscript, which shows similar statistics as the random holdout.

We already ran the requested cross validation (5-fold). The figures below show train/test points from some of the random holdout folds:

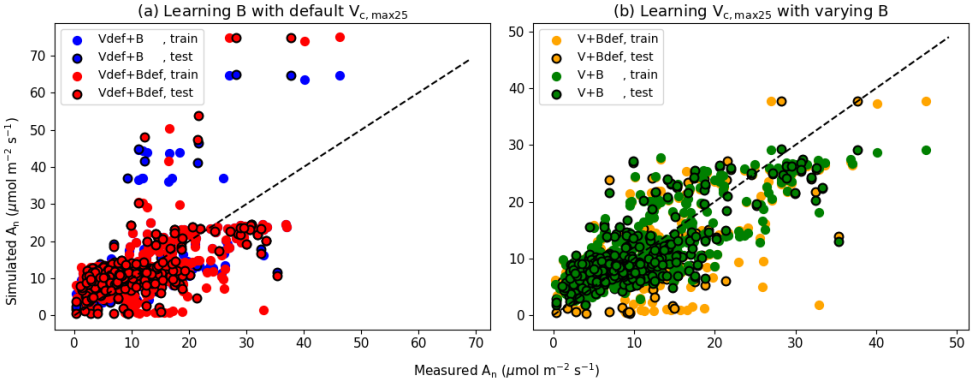
Fold 1:



Fold 2:

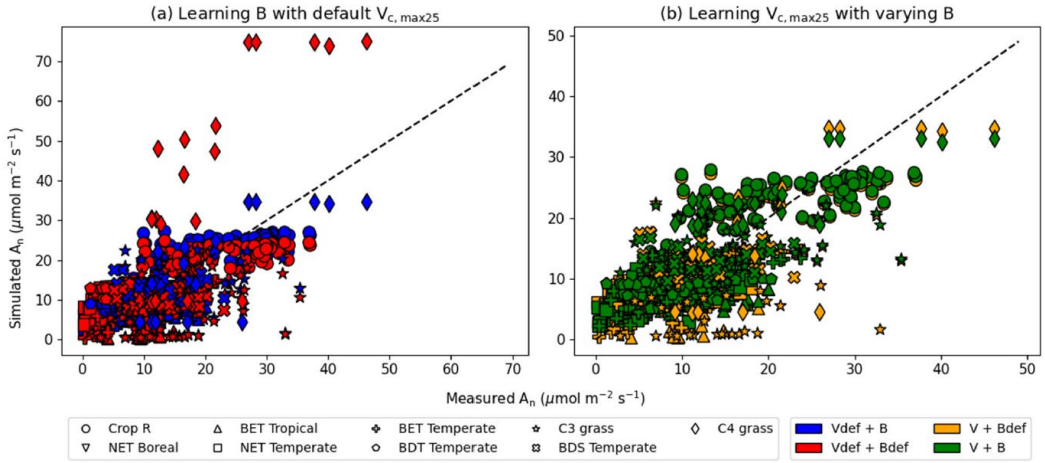


Fold 3:



We skipped the other folds.

The one used in the paper:



It seems the one with different PFT type, used in the original paper, delivers more useful information. We can mention in the revision that the train test figures show that the test points were correctly captured.

line 426: i would add that you have identified parameter values that are optimized for the considered set of model equations and forcings. both of these have limitations. Equations may be wrong, ERA5 is rather uncertain, and measurement principles can vary between stations. This is both a limitation and a strength of your framework. Parameter values will not be transferable to other inputs. On the other hand you can obtain optimized predictions for the given set of forcings.

Good point. Just like any other model, the performance may be impacted when you change the forcing datasets because these datasets may have certain biases. If the model is trained on a global scale, we hope the various different kinds of forcings to be encountered can serve to limit overfitting. We will add the following sentences.

"*Such parameterizations are suitable to the target and forcing dataset used in training (still the most representative dataset we have access to) and are related to the process-based model employed. The dataset may have limitations related to the consistency in the measurement approach, the forcing data contain errors, while the model structure can be improved. The model performance may vary based on different forcing data, too.*"