Field Crops Research 328 (2025) 109912

Contents lists available at ScienceDirect

Field Crops Research

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

Knowledge-guided machine learning with multivariate sparse data for crop growth modelling

Jingye Han [a,b], Liangsheng Shi [a,*], Qi Yang [a], Jin Yu [c], Ioannis N. Athanasiadis [b]

[a] *State Key Laboratory of Water Resources and Hydropower Engineering Sciences, Wuhan University, China*
[b] *Wageningen University and Research, Wageningen, the Netherlands*
[c] *School of Civil Engineering, Yantai University, Yantai, China*

ARTICLE INFO

Keywords:
Deep learning
Knowledge-guided machine learning
LSTM
Crop model

ABSTRACT

Context: Process-based crop models are widely used to simulate the crop growth process. However, these models face limitations due to the simplified process representation and challenges in parameter estimation. Machine learning methods, as an emerging paradigm, have shown potential in circumventing these limitations, but they are criticized for their black-box nature that does not necessarily encompass known crop growth mechanisms, and their demand for big data that may be not available in most agricultural applications.
Objective: This research aims to propose a deep learning architecture that can leverage agronomic knowledge and sparse observational data for crop multivariable simulation, thereby establishing a novel paradigm for crop growth modeling.
Methods: We propose a Deep learning Crop Growth Model (DeepCGM) with a mass-conserving architecture that adheres to the principles of crop growth. Two additional knowledge-guided constraints regarding crop physiology and model convergence are designed to train the model with sparse datasets. An observational dataset from a two-year rice experiment of 105 plots is used to evaluate the DeepCGM against a process-based crop model (ORYZA2000) and two classical deep learning models, also employing augmentation methods. To demonstrate the validity and generalizability of the proposed model, we also conducted a replication case study of a three-year rice experiment totaling 122 plots.
Results: The DeepCGM architecture produces physically plausible crop growth curves for all simulated variables, while the classical machine learning models may make unreasonable predictions that violate the law of mass conservation. Furthermore, DeepCGM simulates more accurately the observed growth process when compared with the traditional process-based model, with overall accuracy (weighted normalized mean square error) across all variables improves by 8.3 % (2019) and 16.9 % (2018).
Conclusions: Knowledge-guided deep learning can integrate the principal mechanisms of crop growth process with deep learning. It addresses the issue of data scarcity, and thereby facilitating data-driven crop growth modelling with multivariable sparse datasets.
Implications: OR SIGNIFICANCE: This study highlights the potential of knowledge-guided deep learning to overcome structural error due to the simplification in conventional crop models and reduce the data requirements of data-driven models. The capacity to autonomously identify multivariable dynamic patterns in crop growth from sparse data suggests a new generation of crop growth models.

1. Introduction

Process-based (PB) crop models conceptualize the interactions between crops and the environment in the form of theoretical and empirical formula, and simulate processes involving multivariable temporal dynamics. They have extensively been used to estimate crop growth and optimize crop management (Ewert et al., 2015; Jones et al., 2017; Keating et al., 2003). Studies over the past decades have developed crop models such as WOFOST (De Wit et al., 2019; Van Diepen et al., 1989), SWAP (Van Dam et al., 1997), APSIM (Holzworth et al., 2014; Keating et al., 2003) and ORYZA (Bouman, 2001) based on different process conceptualizations. Despite their extensive use, crop

* Corresponding author.
E-mail address: liangshs@whu.edu.cn (L. Shi).

https://doi.org/10.1016/j.fcr.2025.109912
Received 8 October 2024; Received in revised form 31 March 2025; Accepted 6 April 2025
Available online 23 April 2025

models are limited by the given presentation of the underlying processes and the challenge of estimating parameters (Wallach et al., 2021). PB models could be improved by refining the process description or model structure (Donatelli et al., 2017; Li et al., 2017; Rosenzweig et al., 2013). However, such adjustments are labor-intensive and may increase the model complexity (Yin et al., 2021).

Machine learning methods, due to their non-dependence on mechanistic processes and the absence of parameters defined based on assumptions, have the potential to circumvent the issues encountered by mechanistic models (Droutsas et al., 2022). As a data-driven method, its capability for feature extraction and capturing nonlinear dynamics (Kashinath et al., 2021) endows it with the power of unraveling complex patterns of crop growth from data. However, the application of purely data-driven methods for crop growth simulation is limited by data scarcity (Droutsas et al., 2022) and the black-box nature. These limitations can lead to overfitting problems, which are comparable to the challenges of parameter estimation in crop models, and may also result in conflicts with fundamental physical principles. The main challenges of data scarcity include: 1) data sparsity caused by the inability to conduct continuous monitoring (Pylianidis et al., 2022) or missing observations on partial variables of interest, and 2) the limited representativeness of datasets due to the inability to collect data covering all possible Genetics × Environment × Management combinations. A straightforward solution is to augment sparse data by generating synthetic labels through statistical interpolation (Moon et al., 2023), or using crop growth models to produce synthetic data under varied inputs, providing a more diverse and complete training dataset (Han et al., 2023; Jia et al., 2019; Pylianidis et al., 2024; Read et al., 2019). These strategies can increase data size and diversity, but may introduce new sources of error. For example, synthetic data generated from crop growth models is inherently prone to model structural errors (Kallenberg et al., 2023).

Although the aforementioned methods primarily enhance the applicability of machine learning in crop modeling by enriching training data, they do not reduce the black-box nature to fundamentally improve ML models. In contrast, conventional crop growth models can be effectively calibrated with small datasets and gradually refined. This efficiency is attributed to the detailed representation of sub-processes (Brisson et al., 2003; Pasley et al., 2023) and the incorporation of extensive prior knowledge, which simplifies the model structure while adhering to fundamental physical laws, such as the Law of Mass Conservation. Given this context, integrating domain knowledge into ML models offers a promising solution. Embedding established principles not only reduces the model's black-box nature but also mitigates overfitting by adding knowledge-based constraints, making it a superior approach to data augmentation.

Knowledge-guided machine learning (KGML) represents a new paradigm that integrates machine learning with domain knowledge. This approach has been applied across various disciplines, investigating how domain knowledge can be incorporated into machine learning models, to ensure better regularization and generalization. Some comprehensive reviews of KGML are available (Karpatne et al., 2017; Von Rueden et al., 2021; Willard et al., 2022). According to (Willard et al., 2022), KGML can be implemented through various methods: 1) Knowledge-guided loss function to instruct consistency with physical laws (Read et al., 2019; Wang et al., 2023); 2) Knowledge-guided initialization, using PB model simulations for ML model pre-training (Jia et al., 2019; Liu et al., 2022; Read et al., 2019); 3) Knowledge-guided structure, designing model structure based on real-world variable relationships and interactions; 4) Residual modeling to address model structural errors through ML (Zhang et al., 2019); and 5) Other hybrid approaches that couple PB and machine learning models. A notable application of KGML is demonstrated by Read et al. (2019), where a neural network, pre-trained on a synthetic dataset generated by a physical model, accurately predicted lake water temperatures under various conditions. In the agricultural area, Liu et al. (2022) developed a KGML model to estimate N_2O emissions. They showed that the KGML model consistently outperformed both PB models and traditional ML models in predicting N_2O fluxes, particularly in capturing complex temporal dynamics and emission peaks. This suggests that KGML models have the potential to surpass PB models in simulating time series states.

In agricultural studies, the predominant approach of coupling knowledge and data-driven methods primarily can be divided into mechanism-centric approach and data-centric approach. Mechanism-centric approach usually leverage data-driven algorithms to 1) optimize or estimate parameters of crop model (Guo et al., 2019; Kawakita et al., 2024), 2) estimate variables that can be used as crop model input (Droutsas et al., 2022; Fan et al., 2015). The mechanism-centric approach leverages crop model as a hard downstream structure to constrain the output of data-driven model. It improves usability and accuracy, but the capability of hybrid models is still limited by crop models. Data-centric approaches often utilize crop models as feature engineering tools to extract features from raw data (Everingham et al., 2016; Feng et al., 2020; Kaneko et al., 2022; Paudel et al., 2021) or as the basis to calculate the residual error (Fan et al., 2015; Paudel et al., 2021). This approach integrates the outputs of crop models as the input features of data-driven models, allowing data-driven models to indirectly leverage the crop model knowledge embedded in the features. Although these features provide some constraints from crop growth mechanisms on top of the raw data, they cannot guarantee that data-driven models obey all the mechanisms that have been discovered. The estimated crop state thus may violate crop growth principles. Moreover, the original goal of crop models is to describe the detailed crop growth process (Hornberger and Spear, 1981) and enhances the interpretability of ML models, while most feature engineering-based hybrid models focus on final yield prediction and neglect the intermediate process of crop growth.

In addition to hybrid modelling, some studies investigate to use data-driven models to simulate crop or plant growth process. For example, Liu et al. (2024) trained a LSTM model to predict the dynamic of green chromatic coordinate of forest and grassland. Moon et al. (2023) built a transformer-based model for tomato growth simulation. Although these studies proved that machine learning has the ability to simulate crop growth, their black-box nature may challenge trust in the models and lead to results that do not align with physical intuition (Pearl, 2019). PB models are interpretable and can be improved because they describe intermediate crop physiological processes and the interaction, while conventional deep learning (DL) models often suffer from opacity in their end-to-end processes. Although explainable AI technologies could partially explain these black-box models (Moon et al., 2023; Paudel et al., 2023), the way forward is to design models that are inherently interpretable (Rudin, 2019). Some recent research has similar insight that incorporating more fundamental or domain knowledge can improve data-driven crop model robustness (Han et al., 2023; Moon et al., 2023).

To embed crop growth mechanisms into machine learning models, thereby guiding the models to learn correct-by-design crop growth mechanisms from multivariate sparse data, we propose a knowledge-guided machine learning model, Deep learning Crop Growth Model (DeepCGM). This model incorporates knowledge-based constraints in its structural design, loss function design, and input weight optimization. We compared DeepCGM with the PB crop growth model ORYZA2000 and two classical DL models (LSTM and mass-conserving LSTM) in a two-year experiment where multivariate sparse observational data were available. We additionally investigate the effect of knowledge-guided structure and constraints on mitigating the model dependence on large datasets; and provides an open-access case study and dataset that can serve as a benchmark for future deep learning-based crop growth modelling studies.

2. Data and methodology

This section first introduces the dataset collected from a two-year experiment. We then concisely summarize three existing methods: a purely knowledge-based model (ORYZA2000); a purely data-driven model (Long Short-Term Memory model, LSTM, and a mass-conserving variant of LSTM (Mass-Conserving LSTM, MC-LSTM). Subsequently, we present the DeepCGM model and introduce three key loss functions and constraints. As illustrated in Fig. 1, the DeepCGM model was trained using a dataset comprising time-series inputs and sparse observations, with constraints derived from plant physiology and general knowledge. The trained model is capable of simulating daily crop states during the testing or application phase. Its performance was evaluated by comparing it with the other three models. The code and dataset will be available at GitHub.

2.1. Dataset

A two-year late-season rice experiment was conducted in Binyang County (23°5′52″~23°7′23″ N, 108°57′7″~108°58′34″ E), Guangxi, China (Fig. 1). Meteorological data, including solar radiation, air temperature, relative humidity, atmospheric pressure, and precipitation, were obtained from NASA POWER (https://power.larc.nasa.gov/). The study included 65 observed paddy plots in 2018 and 40 plots in 2019. Management records, detailing seeding, transplanting, fertilization, and irrigation dates and amounts were acquired through the survey. The soil conditions across all plots were assumed to be homogeneous, as they were not measured during the experiment. Observational data on crop states, including phenology (DVS, development stage, -), leaf (WLV, kg/ha), stem (WST, kg/ha), storage organ (WSO, kg/ha) biomasses, and above-ground biomass (WAGT, kg/ha), were collected through destructive sampling (the definitions of these variables refer to Appendix A). A total of 10 and 12 observation rounds were conducted in 2018 and 2019, respectively. Due to the large number of samples in 2018, organ segmentation for rice leaves, stems, and storage organs was performed at selected times and plots, further contributing to data sparsity. The number of observations for each variable is presented in Figs. S1–2, which shows the lack of organ biomass observations during the early growth stages in 2018. More details on the dataset can be found in **Supplementary S1**. To assess whether generating daily time-series labels can help address the challenge of training model with sparse data, we applied an augmentation procedure to the sparse dataset. The details of this augmentation strategy are provided in **Supplementary S1**.

2.2. ORYZA2000, LSTM and MC-LSTM models

The ORYZA2000 model (REF) was employed as a PB model in this study. It intricately describes rice growth, capturing the complex interactions among numerous intermediate variables and environmental factors. The model calculates the gross daily growth rate using the following equation (Bouman, 2001):

$$G_p = (A_d \times \frac{30}{44} - R_m + R_t) \Big/ Q\# \tag{1}$$

where A_d is the daily rate of gross CO_2 assimilation, R_m is the maintenance respiration costs, R_t is the amount of available stem reserves for growth, and Q is the assimilate requirement for biomass production.

This model function fundamentally represents carbohydrate assimilation through photosynthesis, with a portion consumed by maintenance respiration (R_m). The remaining carbohydrates contribute to biomass synthesis, during which a fraction is used for growth respiration (Q). The residual biomass is then allocated to various organs based on specific partitioning coefficients.

The **LSTM** model was served as the baseline for the purely data-driven approach due to its superiority in handling time-series data. LSTM enforces causal ordering in time series by imposing structural constraints, ensuring that each iterative step is influenced only by the state of the previous day and any newly introduced information. This sequential processing structure shares similarities with conventional process-based crop models for time-series forecasting. A key distinction arises in the way LSTM handles information: the processes of adding, forgetting, and outputting information are autonomously learned by the model without direct control.

MC-LSTM, a variant of the standard LSTM, is specifically designed to enforce mass conservation within a given system during computation. This is achieved by integrating the law of mass conservation directly into the model's architecture. The structure of MC-LSTM enables controlled processes for information input, storage, and removal. Previous research on MC-LSTM has primarily focused on hydrological simulations (Bertels and Willems, 2023; Frame et al., 2023, 2022; Hoedt et al., 2021). In this study, we explore the application of MC-LSTM in crop growth simulation for comparative analysis.

The inputs and outputs of the LSTM and MC-LSTM models are the same as those of the DeepCGM model. After initialization, the ML model produces daily crop states (PAI, WLV, WST, WSO, WAGT, YIELD) based on daily input data, including temperature, radiation, nitrogen and DVS. Since sowing dates vary across plots, all data are aligned based on days

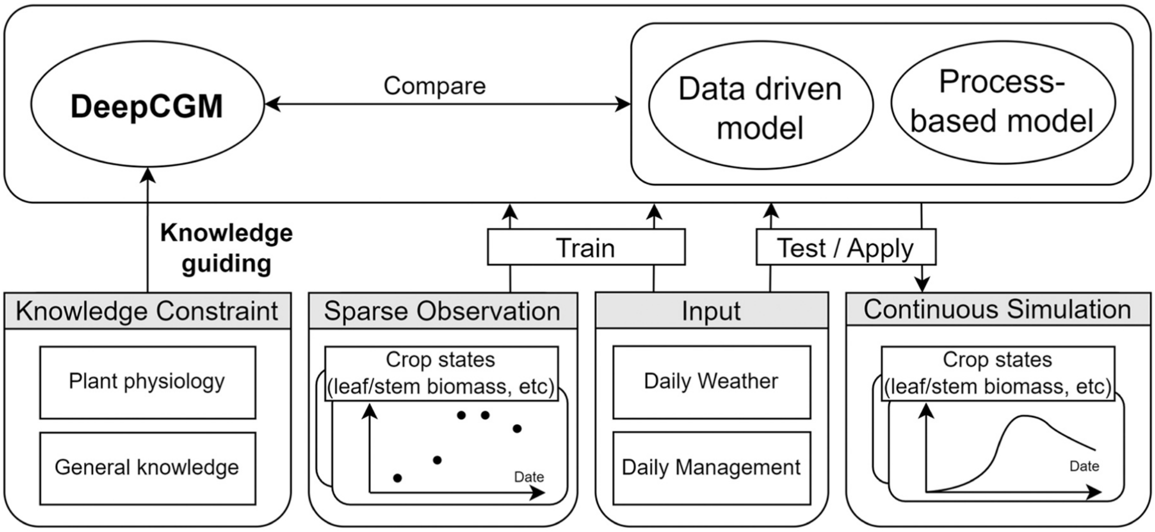


Fig. 1. Framework for training the DeepCGM model and assessing the performance.

after sowing (DAS).

2.3. DeepCGM model

The DeepCGM model is designed to offer inherent interpretability and similar capabilities as PB crop models. To achieve these objectives, two essential principles are required: 1) using model states with physical meaning rather hidden states, and 2) constructing and coupling modules to simulate intermediate crop physiological processes rather than employing an end-to-end approach. Inspired by MC-LSTM (Hoedt et al., 2021), we employed a mass-conserving vector to store model states and assigned physical meaning to the states (e.g., biomass), allowing the hidden states to directly correspond to the crop states. We further aligned mass-conserving gates, which regulate the model states dynamic, with the physiological processes of crops (e.g., respiratory consumption). This conceptualization enhances both the comprehension and manipulation of the model states and functions, enabling the incorporation of domain-specific knowledge in crop. For instance, it allows the model structure to be designed based on plant physiological processes. The key distinction between DeepCGM and PB models lies in their construction requirements. DeepCGM requires only a predefined structure and can automatically learn details from observations, whereas PB models rely on meticulously designed modules and formulas to explicitly describe intermediate processes, such as the allocation, storage, and consumption of assimilated carbon throughout the growth season.

In the soil-plant-atmosphere continuum, most existing crop growth models consider water, nitrogen, and carbon as the primary conserved variables. To streamline the model, this initial version of DeepCGM focuses exclusively on carbon cycle, while disregarding water and nitrogen cycles. The nitrogen process is simplified by incorporating the cumulative nitrogen application time series as an input. The water process is not involved because the experimental area received sufficient precipitation during the study season, ensuring that growth variability was not influenced by water availability. As shown in Fig. 3, a conceptual model of DeepCGM was designed based on the ORYZA2000 model

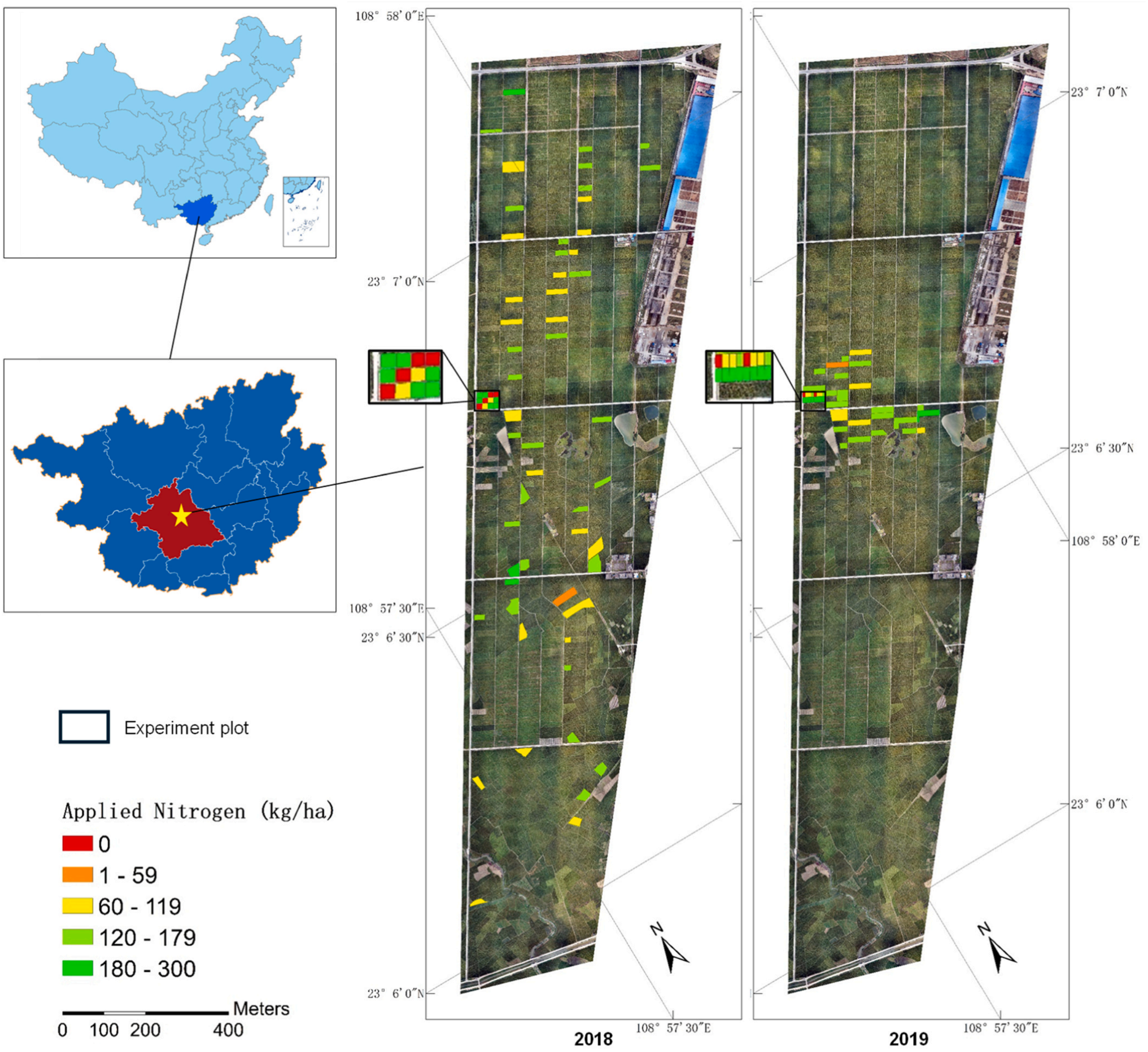


Fig. 2. The location and total applied nitrogen of plots in 2018 and 2019. The experiment plots (the zoom-in area in both years) were managed by us, the other plots were managed by local farmers.

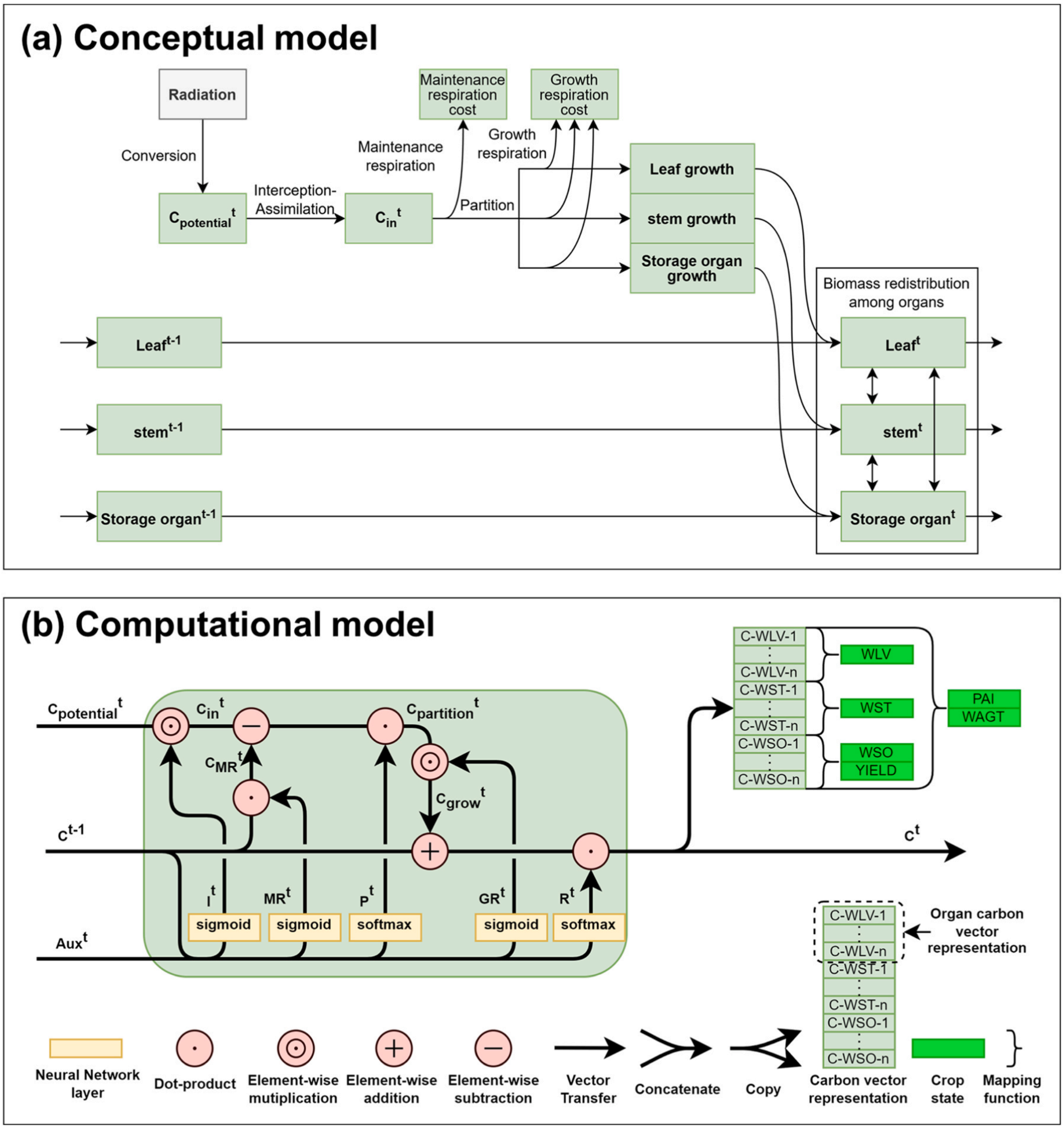


Fig. 3. Schematic of the main components in DeepCGM. (a) the conceptual model. (b) the computational model. The definition of variables can refer to Appendix A. The meanings of other symbols in (b) are as explained in the legend. A more detailed diagram is available in Appendix C.

architecture. The dynamics of carbon state were generalized into five interconnected processes: light interception and carbon assimilation, maintenance respiration, partitioning, growth respiration, and redistribution. Following the this conceptual framework (Fig. 3a), we developed DeepCGM, incorporating a mass-conserving vector to store the carbon state. Additionally, five mass-conserving gates regulate carbon flow among the interconnected processes (Fig. 3b). Appendix A lists the involved variables, including the input variables (driving factors), output variables (simulated crop state), intermediate variables, and gate variables (controlling units). The calculations in the model are divided into two stages, data preprocessing and iterative loop. During the preprocessing stage, raw data are transformed into auxiliary inputs (Aux^{t}) and mass-conserving input ($C_{potential}$), which are organized as follows:

$$DVS^{t} = ORYZA2000_{DVS}(T_{\min}^{t}, T_{\max}^{t})\# \tag{2}$$

$$Aux^{t} = \left[DVS^{t}, R^{t}, T_{\min}^{t}, T_{\max}^{t}, N_{cum}^{t},\right]\# \tag{3}$$

$$C_{potential}^{t} = Rad^{t} \cdot ORYZA2000_{photosynthesis}(T_{\min}^{t}, T_{\max}^{t})\# \tag{4}$$

where variable definitions can be found in Appendix A; $ORYZA2000_{DVS}$ is the phenology module of ORYZA2000; $ORYZA2000_{photosynthesis}$ is the photosynthesis module of ORYZA2000.

In the iterative loop, each step transitions carbon from its state on the previous day to its state on the next day. The main components of each iteration are described as follows (the formulas for the six steps refer to Appendix B):

1) Gate Calculation: At the beginning of each iteration, the model calculates gate variable values (I^{t}, MR^{t}, P^{t}, GR^{t}, R^{t}) based on the gate input (X^{t}). The gate input consists of the previous day's carbon (C^{t-1}) and the current day's auxiliary data (Aux^{t}).
2) Light interception and carbon assimilation : Due to the limited leaf area and chlorophyll content, only a portion of radiation can be intercepted and assimilated as carbon by the crop. Therefore, the

interception-assimilation gate is used to estimate the proportion of radiation that is intercepted and assimilated as carbon. The day's carbon input (C_{in}^t) is determined as the residual carbon after the potential carbon (the maximum amount of carbon that can be synthesized given the current photosynthetic efficiency and available radiation) passes through the interception-assimilation gate (I^t).

3) Maintenance respiration costs: The intercepted carbon is first utilized for respiratory consumption (C_M^t), which is determined by maintenance respiration gate (MR^t) and carbon state from the previous day (C^{t-1});
4) Carbon partitioning: After accounting for maintenance respiration, the remaining carbon ($C_{in}^t - C_M^t$) is allocated for synthesis of different plant tissues through the carbon partitioning gate (P^t), resulting in partitioned carbon ($C_{patition}^t$).
5) Growth respiration costs: A portion (GR^t) of the partitioned carbon allocated for tissue synthesis is consumed by growth respiration, while the remaining amount (C_{grow}^t) is incorporated into the carbon vector as net growth.
6) Redistribution: The carbon stored in the carbon vector ($C^{t-1} + C_{grow}^t$) is redistributed among vector elements, as controlled by the redistribution gate (R^tThis facilitates carbon redistribution across different plant components. The redistributed carbon vector (C^t) represents the final output of this step, from which the crop states can be derived.

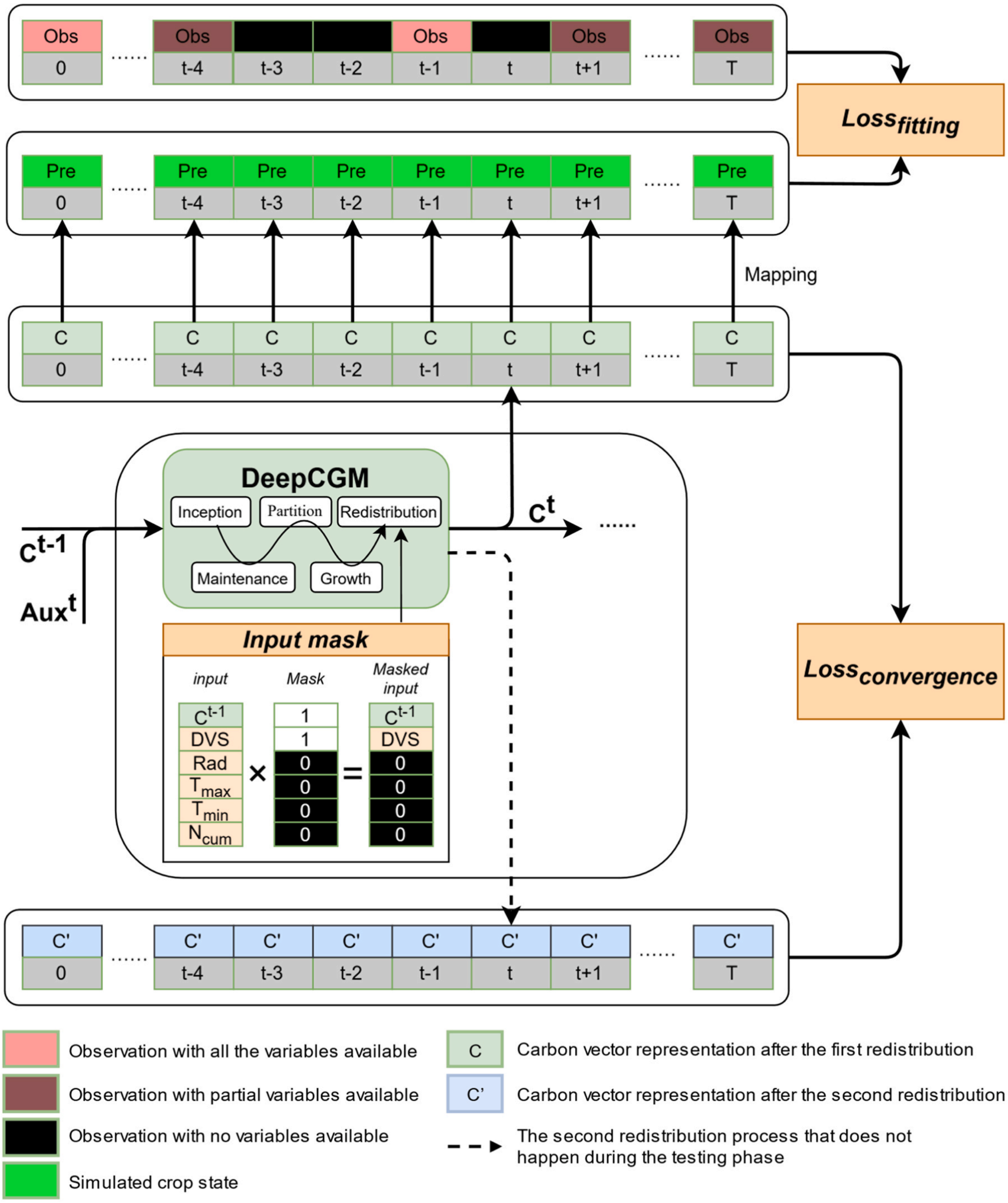

Obs
0
t-4
t-3
t-2
t-1
t
t+1
T
Loss fitting
Pre
Mapping
C
DeepCGM
Inception
Partition
Redistribution
Maintenance
Growth
C t-1
Aux t
C t
Input mask
input
Mask
Masked input
DVS
Rad
T max
T min
N cum
1
0
×
=
Loss convergence
C'
......
Observation with all the variables available
Observation with partial variables available
Observation with no variables available
Simulated crop state
Carbon vector representation after the first redistribution
Carbon vector representation after the second redistribution
The second redistribution process that does not happen during the testing phase

Fig. 4. The training process of DeepCGM model using designed fitting loss, convergence loss and input mask.

2.4. Fitting loss and Knowledge-guided constraints

During the training process, model parameters are optimized using gradient backpropagation. In this study, we employed three constraints to guide the model training:

The first constraint is the fitting loss ($Loss_{fitting}$, Fig. 4), which is a standard supervised loss rather than a knowledge-guided constraint. It enables the model to learn the mapping between input and output data of training data, but it cannot incorporate the prior knowledge that aligns with the crop physiological processes. Additionally, to ensure that different variables are weighted similarly during model training, we applied variable-specific normalizing factors and weights (Table 1). The final fitting loss is a weighted, normalized mean square error, which is defined as follows:

$$Loss_{fitting} = \sum_{k=1}^{K} \frac{\sum_{n=1}^{N_k} \left(y_{k,n}^{pre} - y_{k,n}^{obs} \right)^2}{N_k} \times w_k \# \tag{5}$$

where K is the number of variables (K=6 in this study), and k = 1,2,3,4,5,6 correspond to the six estimated variables: PAI, WLV, WST, WSO, WAGT, YIELD). The weight for each variable is denoted by w_k, which is empirically assigned balance their of different state variables within the loss function. N_k is the number of observations for variable K, while $y_{k,n}^{pre}$ and $y_{k,n}^{obs}$ are the model's simulated and the observed values, respectively.

The second constraint is a knowledge-guided loss, termed 'convergence loss' ($Loss_{convergence}$, illustrated in Fig. 4). This loss aims to mitigate overfitting caused by data sparsity. Due to their strong nonlinear fitting capabilities, DL models can adopt distorted curves to fit sparse data, often leading to unrealistic results that contradict real-world processes. In other words, DL model may overfit the observed data points in training dataset, failing to achieve convergence during daily iterations. To address this issue, an unsupervised loss is needed constrain the DL models, ensuring the model is regulated even on dates without observations. The concept of convergence loss is inspired by the iterative calculation methods used in numerical models, such as soil water simulation in the Hydrus-1D model (Simunek et al., 2005). In Hydrus-1D, dynamic changes in soil water across different layers require multiple iterations to approximate real conditions. Convergence is reached when the differences between successive iterations fall below a specified threshold. Similarly, in DeepCGM, the redistribution process is executed twice during training. By minimizing the discrepancy between the results of these two iterations, the model approaches a convergent state, improving its stability and accuracy under sparse observational data conditions. The convergence loss is defined as:

$$Loss_{convergence} = \sum_{t=1}^{T} \frac{\sum_{i=1}^{N} \left(C_{i,t}^{pre} - C_{i,t}^{pre'} \right)^2}{N} \times \frac{1}{T} \# \tag{6}$$

where T is the length of input series (number of days to simulate); N is the size of the carbon vector (N = 3 n); $C_{i,t}^{pre}$ and $C_{i,t}^{pre'}$ represent the carbon vector state after the first and second redistribution, respectively.

The third constraint is a knowledge-guided constraint, termed the 'input mask' (Fig. 4). It is used to filter out irrelevant variables from input for specific subprocesses, thereby making it easier to capture data characteristics and mapping relationships compared to purely ML model. A predefined vector, set based on prior agronomic knowledge, determines the mask. In this study, the input mask was applied only to the biomass redistribution process. Building on insights from ORYZA2000, certain variables such as radiation, temperature, and cumulative fertilizer application do not directly influence carbon redistribution. Instead, their impact on crop growth is indirectly mediated through other mechanisms, such as water stress and nitrogen stress (Bouman et al., 2001a). Consequently, as shown in Fig. 4, the input mask assigns a weight of zero to Rad, T_{max}, T_{min} and N_{cum} in the redistribution gate, while maintaining a weight of one for the other relevant variables.

The training schematic in Fig. 4 demonstrates that the MC-LSTM and DeepCGM models can simultaneously incorporate all three constraints. In contrast, the ORYZA2000 and LSTM models, due to their structural limitations, can only utilize the fitting loss for their calibrating/training.

Table 1
The normalization parameters and weights in fitting loss.

	PAI	WLV	WST	WSO	WAGT	YIELD
	m^2/m^2	kg/ha	kg/ha	kg/ha	kg/ha	kg/ha
Maximum value of observation	7.51	3830	9553	9710	17770	9226
Scaling factor used for normalization	8	20000	20000	20000	20000	20000
Maximum normalized value of observation	0.94	0.19	0.48	0.49	0.89	0.46
Weights in fitting loss	1	5	2	2	1	2
Maximum normalized value of observation ×Weights in fitting loss	**0.94**	**0.96**	**0.96**	**0.97**	**0.89**	**0.92**

2.5. Model configuration and case setup

The ORYZA2000 model was calibrated separately using data from 2018 and 2019. For each year, all raw sparse observations, management records, and weather data from all plots were used for calibration in one year and for testing in the other. The calibration procedure consists of two steps: first, three phenology-related parameters were adjusted using phenology observations; then, sixteen biomass-related parameters were calibrated based on PAI, organ biomass, and yield observations. Loss_fitting was used as the objective function to balance the weights of different variables during the calibration of biomass-related parameters. Calibration was conducted using Particle Swarm Optimization (Kennedy and Eberhart, 1995), as implemented in the Pymoo package (Blank and Deb, 2020). The default and calibrated parameters are provided in Supplementary S2 and S3. The performance of the calibrated ORYZA2000 in both the calibration and testing years is presented in Appendix D.

The ADAM (Kingma and Ba, 2017) optimizer was used for parameter optimization in the deep learning models. The learning rates were set to 0.005 for LSTM, and 0.1 for both MC-LSTM and DeepCGM. The hidden state size of LSTM model was 64. The carbon vector size of MC-LSTM and DeepCGM were 24 (8*3). All models were trained for 700 epochs, and the model that performed best on the training set was selected for testing (early stopping). No learning rate decay was applied during model training. The loss function for both MC-LSTM and DeepCGM was defined as:

$$\text{Loss} = Loss_{fitting} + Loss_{convergence} * alpha \# \tag{7}$$

where *alpha* was set to 100,000. This study employs fitting loss (function 5) as the evaluation metric for overall accuracy, and additionally uses RMSE to assess the accuracy of each state:

$$RMSE_k = \sqrt{\frac{\sum_{n=1}^{N_k} \left(y_{k,n}^{pre} - y_{k,n}^{obs} \right)^2}{N_k} \#} \tag{8}$$

where k = 1,2,3,4,5,6 correspond to the six state variables (PAI, WLV, WST, WSO, WAGT, YIELD). N_k is the number of observations for each variable in the test dataset. Due to the limited number of observations in our dataset relative to the number of parameters in deep learning models, the training process was extremely sensitive to initial random seeds. To facilitate comparison in the radar chart, RMSE was further converted into a normalized index (NI).

$$NI_{k,j} = 1 - \frac{RMSE_{k,j}}{RMSE_{k,\max}} \# \quad (9)$$

where k = 1,2,3,4,5,6 correspond to the six state variables (PAI, WLV, WST, WSO, WAGT, and YIELD); *j* denoted the model id; $RMSE_{k,\max}$ represents the maximum RMSE for variable *k* across all models. To mitigate the effects of random initialization, each training and testing iteration was conducted 25 times using random seeds ranging from 0 to 24. The final evaluation metrics were derived from the statistical aggregation of these 25 repetitions.

Several computational experiments were designed to assess the contribution of data availability, model design, and knowledge constraints in simulating crop growth using DeepCGM. For data availability, a two-year dataset was divided so that one year was used for training and the other for testing, with the option to swap the training and testing sets to assess model performance across different years. Additionally, we incorporated cases using augmented and partial-removed datasets to investigate the impact of data availability. For models and constraints, a PB model (ORYZA2000) and three data-driven models (LSTM, MC-LSTM and DeepCGM) were trained or calibrated using various strategies to investigate the effects of different losses and constraints. Table 2 presents seventeen designed cases and their corresponding computational experiments:

(**E1**, Cases 1–3 and 4) Different model trained/calibrated with fitting loss were compared to assess the effect of model structure.
(**E2**, Cases 7–10) DeepCGM was trained using various strategies to evaluate the effects of the input mask and convergence loss.
(**E3**, Cases 2, 7, 10, 13 and 16–17) Conventional research employs augmented data to address the issue of sparse observations. In these cases, models trained with augmented data were compared against those trained with sparse data to assess the effect of constraints and the contribution of data augmentation in improving model performance.
(**E4**, Cases 2 and 10–12) Given that real-world datasets often lack full coverage of the growing season, early-season observations (first 50 days in 2019) were removed to simulate temporally uneven data. Comparing models trained on complete and partially removed datasets enabled an investigation of the effects of data gaps.
(**E5,** Cases 1–10 and 13–17) All models trained on sparse dataset and augmented were compared to determine the optimal performance achievable from our dataset.

Table 2
Model, training strategy and dataset for different cases.

Case Num	Computational Experiments					Model	Training strategy			Dataset	
	E1	E2	E3	E4	E5		Fitting loss	Input mask	CG loss	Training	Testing
1	Model structure				The best accuracy	ORYZA2000	√			2018 2019 Sparse	2019 2018 Sparse
2			■	■		LSTM	√				
3						MC-LSTM	√				
4							√	√			
5							√		√		
6							√	√	√		
7	■	Training strategy	■			DeepCGM	√				
8							√	√			
9							√		√		
10			■	Data and constraints			√	√	√		
11							√	√	√	2019 Sparse, Removed	2018 Sparse
12						LSTM	√				
13			■		■	LSTM	√			2018 2019 Augmented	2019 2018 Sparse
14						MC-LSTM	√				
15							√	√	√		
16			Constraints and augmentation			DeepCGM	√				
17							√	√	√		

3. Results

The results section is divided into two main parts. The first part examines model training strategies, highlighting the performance of different cases during training. The second part focuses on model performance, presenting the simulation results of DeepCGM on the test set and comparing its accuracy with other models. To demonstrate the generalization capacity of the proposed approach we offer a replication study in **Supplementary S4**.

3.1. Training model with different structures, strategies, and datasets

3.1.1. Model performance with different structures

The simulations depicted in Fig. 5 (a1-f3), which compare the results from ORYZA2000 (case 1), LSTM (case 2), MC-LSTM (case 3), and DeepCGM (case 7) trained with fitting loss, demonstrate the effect of model structure (**E1**). Specifically, the result curves generated by the calibrated ORYZA2000 model closely aligned with observed data and effectively captured the actual crop growth process. For example, the WAGT initially exhibited slow growth, followed by a gradual acceleration, and eventually slowed down, stabilizing without further growth (Fig. 5 e1). This pattern closely resembles a logistic growth curve, which is commonly used to describe crop growth dynamics (Yin, 2003). Given the absence of continuous real-world growth observations, the ORYZA2000 simulations, which reflect typical rice growth patterns, serve as a valuable reference for comparison with other models.

Compared to ORYZA2000, the LSTM model occasionally approximates the observed labels well but produces growth curves that deviate from typical crop growth patterns (Fig. 5 a1-f1). For WLV, WST, WSO, and PAI (Fig. 5 a1-d1), these deviations were characterized by: (1) noticeable fluctuations in the early development stages across all variables; (2) reduced fluctuations in the middle and later development stages, with simulation results inconsistent with observation or ORYZA2000 outcomes. The primary cause of these early-stage fluctuations was the absence of training data during the early development stages. In the middle and later stages, the intrinsic long-term memory capabilities of the LSTM model contributed to more stable simulations (Fig. 5 a1-d1). However, minor fluctuations persisted, likely due to the sparsity and noise in the training data, which prevented the model from learning universal growth patterns and led to inaccuracies during data-scarce periods. For WAGT, simulations in early development stage were more accurate (Fig. 5 e1), possibly due to more early development stage WAGT observations in the training set (Figs. S1–2). Regarding the YIELD, the LSTM model's predictions completely contradicted the trend reflected by ORYZA2000 during the early development stage (Fig. 5 f1). This issue arose because yield observations were available only twice throughout the entire growth period—once at planting (initial value) and once at harvest (final value). As a result, the LSTM model struggled to effectively reproduce the temporal evolution of YIELD.

The simulation results of MC-LSTM and DeepCGM, as shown in Fig. 5 (a2-f2) and (a3-f3), exhibited better consistency with ORYZA2000 compared to the LSTM model particularly for WAGT (Fig. 5 e2 and e3). Notably, both MC-LSTM and DeepCGM showed less fluctuation in early-stage simulations than LSTM. Furthermore, the WAGT simulated by DeepCGM demonstrated a slower initial growth followed by a gradual acceleration, aligning more closely with ORYZA2000 and real crop growth patterns. This improvement can be attributed to the structure of DeepCGM, which was designed based on ORYZA2000, ensuring better alignment with known physiological processes. However, for organ biomass (WLV, WST, and WSO) and their derived states (PAI and YIELD), while the trends were similar to ORYZA2000, both models still exhibited significant fluctuations (Fig. 5 a2-d2 and f2). This issue arises from the mass-conserving structure, which ensures the temporal stability of total biomass by regulating growth and consumption but does not impose restrictions on redistribution among individual organs. As a result, fluctuations occur in the biomass of individual organs. Despite fluctuations in YIELD, its simulated values in the early and middle development stages for both MC-LSTM and DeepCGM remained close to zero, which is consistent with real crop growth. This behavior can be attributed to the model structure, which was designed based on biomass-related variable relationships (as formulas A12 to A17). These relationships enable DeepCGM to infer YIELD from WSO rather than simply fitting to sparse YIELD observations.

Consequently, the following summaries were drawn **E1** cases: (1) ORYZA2000 demonstrated the closest match to observations among all models, indicating that relying solely on a mass conservation structure is insufficient for a data-driven model to outperform conventional crop models; (2) WAGT simulated by MC-LSTM and DeepCGM aligned more closely with real crop growth processes compared to the LSTM model, demonstrating that the mass conservation principle enables models to capture aspects of crop growth; (3) WAGT simulated by DeepCGM was superior to that of MC-LSTM, suggesting that structures designed based on plant physiological knowledge can improve model performance; (4) YIELD simulations from MC-LSTM and DeepCGM better reflected real crop growth patterns, indicating that relationship structure constraints can help guide model learning from sparse observations.

3.1.2. DeepCGM model trained with different strategies

To further reduce fluctuations in the temporal simulation results of organs biomass, we introduced input mask and convergence loss to constrain calculations within the redistribution process, as illustrated in the **E2** cases. The application of input mask (Fig. 5 a4-f4) and convergence loss (Fig. 5 a5-f5) significantly reduced fluctuations. This improvement can be attributed to: (1) input mask selectively filtering out variables irrelevant to the redistribution process; (2) convergence loss establishing convergence criteria, effectively preventing under-allocation and over-allocation during, which caused severe fluctuations (Fig. 5 a1-f1). Since input mask and convergence loss regulate the redistribution from different perspectives, their combined application resulted in an additive effect, further smoothing all simulation curves (Fig. 5 a6-f6) and aligning them closely with the growth trends simulated by ORYZA2000. Moreover, as shown in Fig. 5 a5-f5, the fluctuation was less pronounced compared to Fig. 5 a4-f4, suggesting that convergence loss might be more effective than input mask in reducing fluctuations. In conclusion, the E2 results demonstrated that by constraining the redistribution process, convergence loss and input mask effectively improved the simulation performance of organ biomass.

3.1.3. Training deep learning models on augmented dataset

To compare with traditional methods for handling data sparsity, we trained LSTM and DeepCGM models using both augmented dataset and a raw sparse dataset. Compared to models trained on sparse data, those trained on the augmented dataset produced results that were more closely aligned with the ORYZA2000 model (e.g., Fig. 6 a4), specifically: (1) the early-stage growth rates of all variables showed better alignment (e.g., Fig. 6 a4) compared to case 10 (Fig. 6 a2); (2) the simulated PAI, WLV, WST, WSO, and YIELD were noticeably smoother (Figs. S1–4 a4-f4, a5-f5) compared to those in case 2 and case 7 (Figs. S1–4 a1-f1, a2-f2).

The abovementioned improvement can be attributed to data augmentation, which fills in the gaps in sparse data. However, using ORYZA2000 for augmentation introduces its inherent structural errors into the augmented data, leading to similar errors in models trained on this data. For example, WLV simulated by ORYZA2000 always remained constant in the later stages, whereas the deep learning models trained on this augmented data also exhibited similar constant trends, deviating from the observed data, where WLV gradually declined (Fig. 6 a3 and a4). Additionally, using augmented data does not guarantee adherence to physiological processes, as the fitting loss merely directs the model to fit observations rather than ensuring it follows physiological constraints. For example, in the LSTM model trained on augmented data, YIELD exhibited minor fluctuations during the early and mid-growth stages

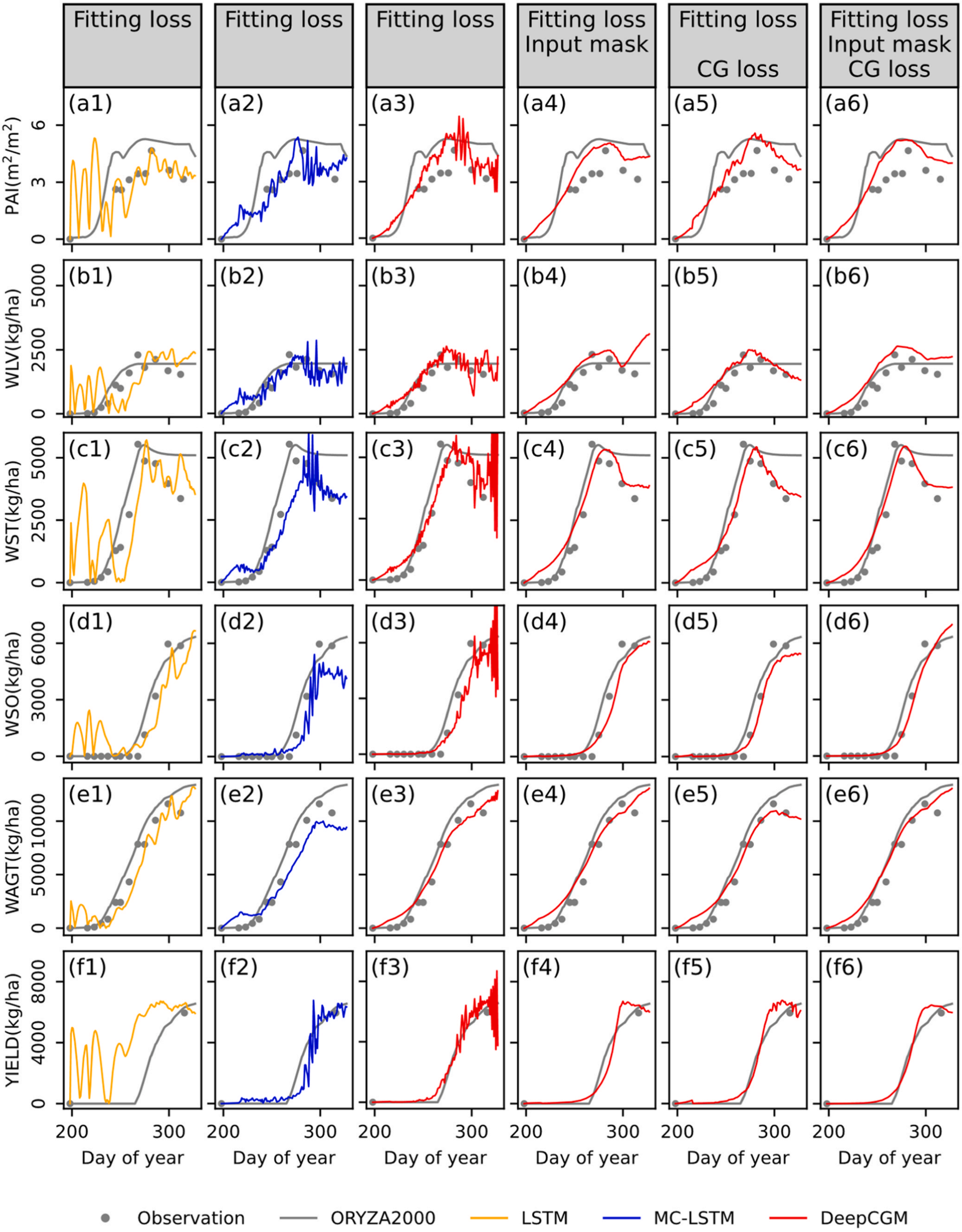


Fig. 5. Crop growth simulation results ORYZA2000, LSTM, MC-LSTM, and DeepCGM. Each model was trained or calibrated on the 2018 data using a random seed of 0, and the displayed results were the testing outcomes for a randomly selected plot from 2019.

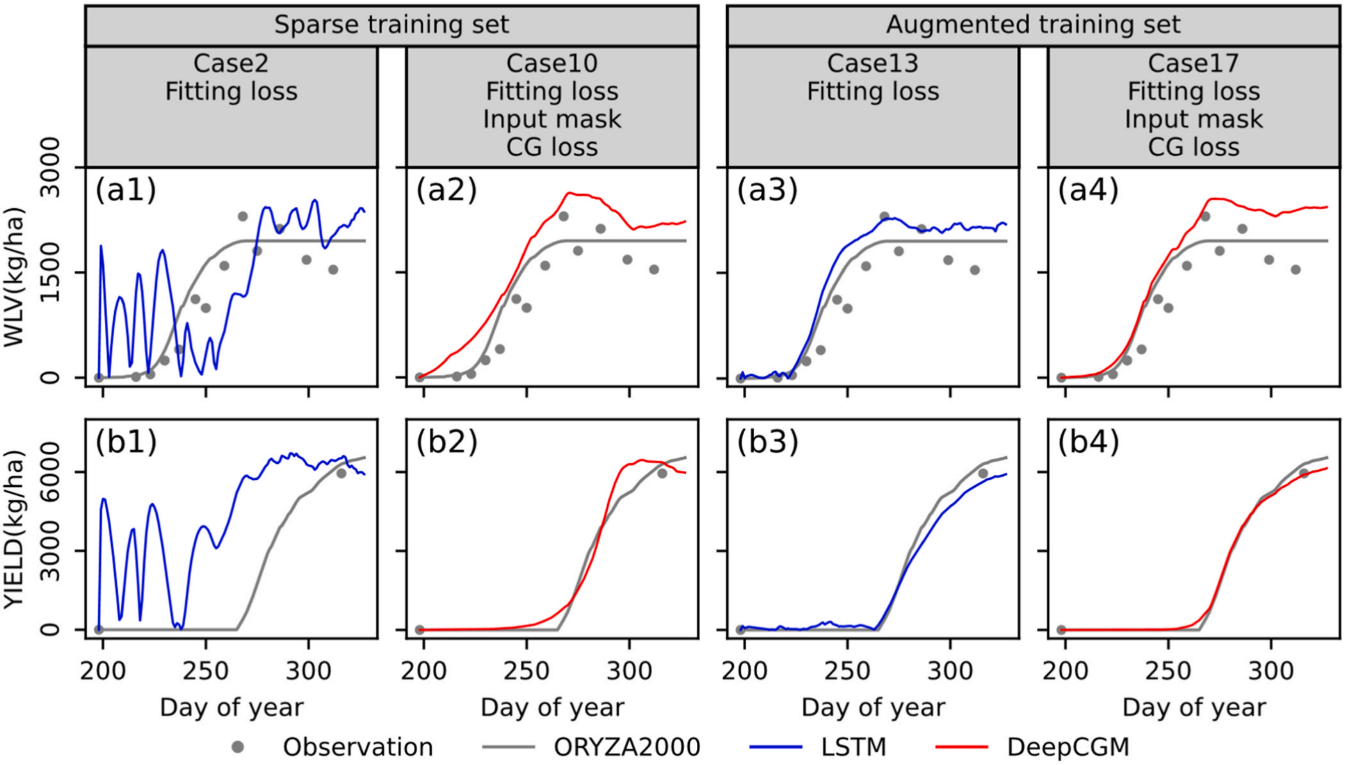


Fig. 6. Crop growth simulated by LSTM and DeepCGM on sparse and augmented dataset (all the models were trained/calibrated on data from 2018 with random seed of 0). This figure only displays WLV and YIELD, and detailed results refer to Figs. S1–4.

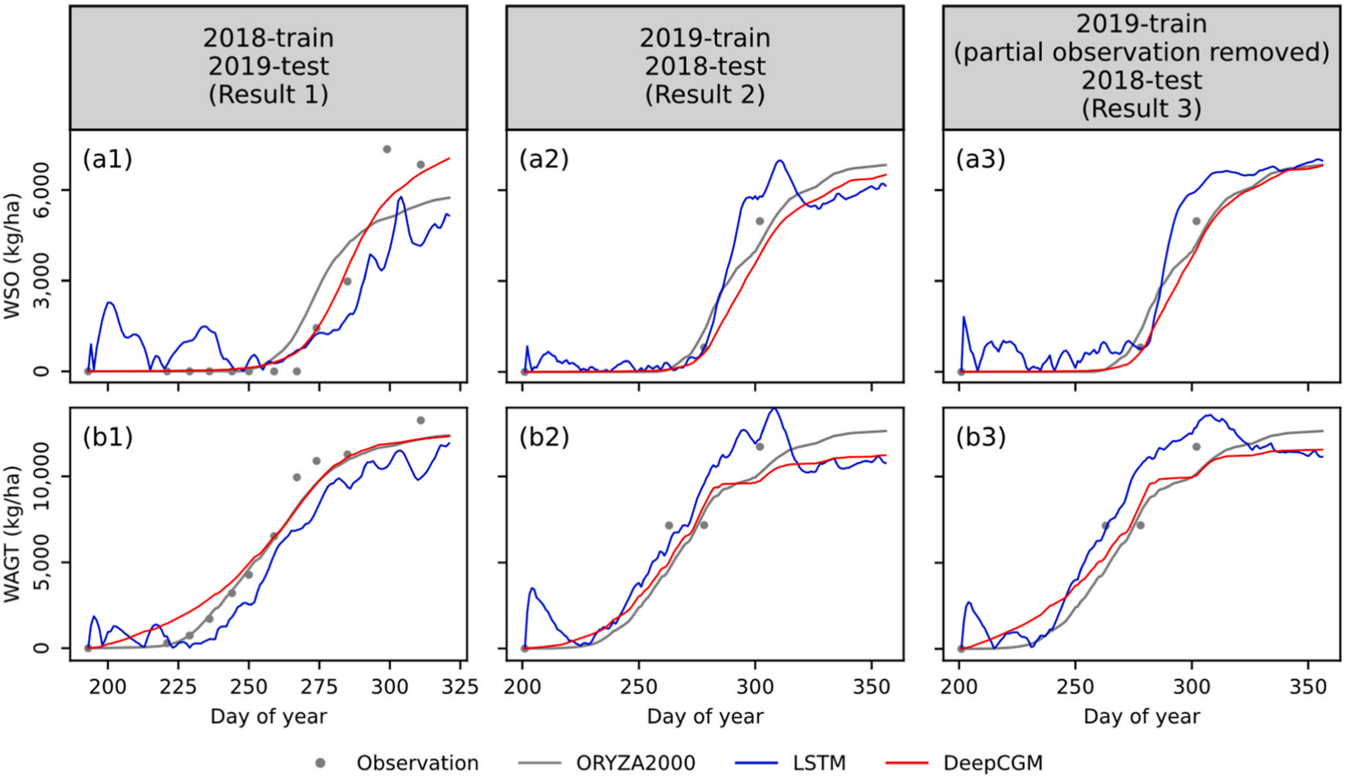


Fig. 7. Result examples of model trained on the 2018 dataset, 2019 dataset, and the 2019 dataset with partial observation removed. Each model was trained with the random seed of 1, and the displayed results are the testing outcomes for a random selected plot from both test datasets. This figure only displays WSO and WAGT, and detailed results refer to Figs. S1–5.

(Fig. 6 b3), even though YIELD should consistently remain zero at that stage, as rice filling had not yet begun. After incorporating input mask and convergence loss, fluctuations in the simulated YIELD were eliminated, regardless of the use of augmented data (Fig. 6 b2 and b4). Therefore, the **E3** results indicate that while data augmentation helps compensate for sparse data by providing additional labels, it also introduces structural errors into the model. Furthermore, relying solely on data without integrating fundamental constraints cannot ensure that the model adheres to physiological processes.

3.1.4. Training model with partial observation removed dataset

This section presents the results from **E4**, highlighting the impact of missing observational data on model performance. We trained LSTM and DeepCGM models using three datasets: the 2018 dataset, the 2019 dataset, and the 2019 dataset with early-stage observations removed. For the LSTM model, early-stage rice growth simulations in Result 2 (Fig. 7 a2 and b2) showed significantly less fluctuation than in Result 1 (Fig. 7 a1 and b1), which can be attributed to more early observations in the 2019 dataset (Figs. S1–2). Therefore, removing early-stage observations from the 2019 training set resulted in worse early-stage simulations in Result 3 (Fig. 7 a3 and b3), producing fluctuations similar to those observed in Result 1. For the DeepCGM model, simulations remained smooth and accurate across all results, without the fluctuations observed in LSTM simulations (e.g., Fig. 7 a2 and b2). Although removing early-stage observations led to a slightly higher biomass growth rate compared to both observed data and ORYZA2000-predicted

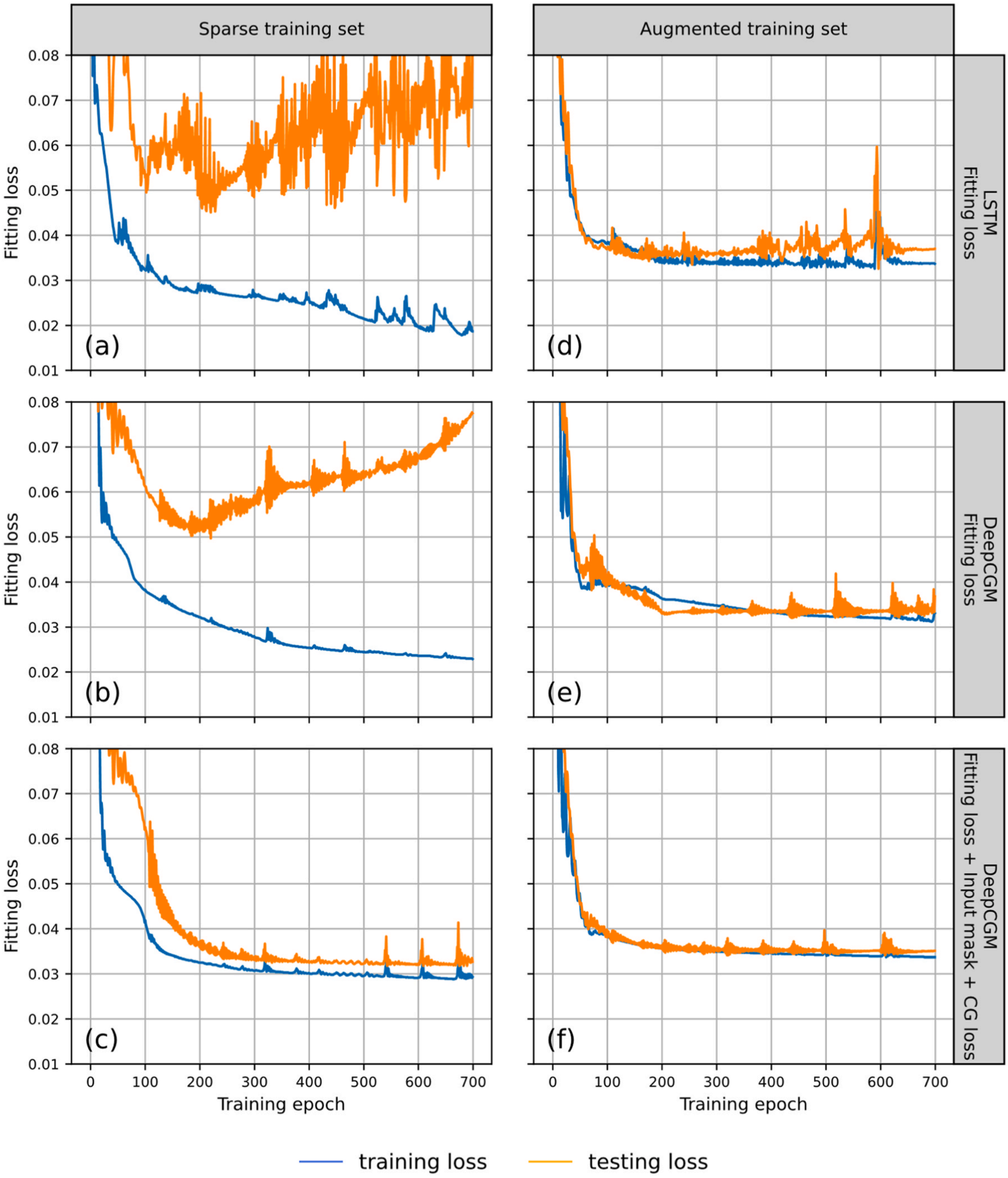


Fig. 8. Fitting loss of training and test set of the LSTM and DeepCGM models trained with different strategies and dataset over the training process (random seed = 0, and the results from other seeds refer to Figs. S1–6).

rates, the deterioration was negligible compared to the LSTM model (Fig. 7 b3 and Figs. S1–5 a3-c3).

These results demonstrate that a lack of training data indeed weakens the model's simulation ability. However, compared to the unconstrained LSTM model, the performance decline of DeepCGM was less pronounced, even after some observations were removed. This suggests that the constraints applied in DeepCGM, including structure, input mask, and convergence loss, help mitigate the negative impact of data scarcity and partially compensate for performance deterioration.

3.1.5. Robustness during training process

This section presents the loss curves of training and test set losses over epochs, with the aim of discussing the impact of train strategies (**E3**) and comparing the robustness of different models to random initialization.

When the models were trained on sparse datasets with fitting loss, the LSTM model exhibited poor stability, with significant fluctuations in its loss curve (Fig. 8a), whereas the DeepCGM model showed a more stable curve (Fig. 8c). Similar patterns were observed in the loss curves generated from other random seeds (Figs. S1–6 a and c). Additionally, the decreasing loss on the training set, alongside the increasing loss on the test set indicated overfitting (Fig. 8a and c). The introduction of knowledge-guided constraints significantly reduced overfitting (Fig. 8e), and smoothed the loss curve, demonstrating the effectiveness of these constraints in improving model training and mitigating overfitting. When the models were trained on augmented datasets, all training processes became more stable (Fig. 8b, d, and f): (1) fitting loss on the training set did not continuously decrease; (2) fitting loss on the test set showed less fluctuations. Moreover, the loss curves of all models trained on augmented datasets were similar (Fig. 8b, d, and f), indicating that when observations are abundant, model training relies more on the dataset rather than the model architecture or structure. Among all the results in Fig. 8b, d, f and e, the test set loss of DeepCGM trained on a sparse dataset with all constraints (Fig. 8e) was lower than that of other models. This demonstrates that although augmentation methods can improve training stability, it does not necessarily enhance model performance, as augmentation may introduce structural errors into the data.

Figs. S1–6 presents the fitting losses from models trained using 25 different random seeds. For models trained on sparse datasets using only fitting loss, both LSTM (Figs. S1–6 a) and DeepCGM (Figs. S1–6c) exhibited substantial variability across the 25 training processes, indicating that parameter initialization had a significant impact on model performance. In contrast, DeepCGM trained with all constraints showed minimal variation across 25 training runs. For models trained on augmented datasets (Figs. S1–6 b, d, and f), each had slight differences among their 25 loss curves. These results suggest that both knowledge-guided constraints and data augmentation can effectively reduce the impact of parameter initialization on the training process, leading to more stable model performance.

3.2. Model performance comparison

3.2.1. Simulating crop growth under different nitrogen levels using DeepCGM

The temporal simulation of DeepCGM for three selected plots with varying fertilization levels were compared with those of ORYZA2000 (Fig. 9 and Figs. S1–7). Although the ORYZA2000 model successfully captured the overall growth trends for most variables, noticeable discrepancies remained between its simulations and the observations. For example, the simulated WLV remained constant during the middle and later growth stages, whereas the observed values showed a declining trend (Fig. 9 c1-c3). Similarly, WST experienced a substantial decline in the later growth stages, while the ORYZA2000 simulations showed only a slight decrease (Fig. 9 d1-d3). These deviations may be attributed to potential structural errors in the ORYZA2000 model.

For DeepCGM, the overall trend of the simulated variables was reasonable and well aligned with both ORYZA2000 and observations. In some cases, DeepCGM even outperformed ORYZA2000, as demonstrated by the following: (1) In zero and moderate fertilization cases, the peak PAI occurred at similar dates in both models (Fig. 9 b1 and b2, and Figs. S1–7 b4 and b5). In high fertilization cases, the peak PAI simulated by DeepCGM occurred later than in ORYZA2000, aligning closer to the observations (Fig. 9 b3 and Figs. S1–7 b6); (2) WLV decreased in the middle and later stages (e.g., Fig. 9 c1-c3, although c1 increased again after decreasing), while ORYZA2000 maintained a constant WLV, deviating from observations; (3) WST decreased in the later stages, with a more pronounced decline at higher fertilization levels (e.g., Fig. 9 d1-d3 and Figs. S1–7 d4-d5). This was an improvement over ORYZA2000, as the WST decline in ORYZA2000 did not vary with fertilization levels, whereas observations showed a greater decline in WST under higher fertilization (e.g., Fig. 9 d1-d3). However, in some cases, DeepCGM deviated from actual crop processes and performed worse than ORYZA2000: (1) PAI, WLV, WST, and WAGT showed higher growth rates than both ORYZA2000 and observations, particularly in the 2018-training-2019-testing case (Fig. 9 b1-f3); (2) In 2018, WLV in the low-fertilization plot showed an unexpected increase in the later stages (Fig. 9 c1); (3) In 2018, YIELD under moderate and high fertilization showed a decline in the later stages (Fig. 9 g2 and g3), which should not occur in reality.

Furthermore, it was observed that the crop growth simulated by DeepCGM exhibited a positive correlation with the amount of fertilization. Specifically, higher fertilizer levels led to increased biomass production (e.g., Fig. 9 f1-f3), which aligns with real crop growth patterns. However, compared to ORYZA2000, DeepCGM demonstrated lower sensitivity to fertilization. For example, the variation in WAGT in response to fertilization changes was less pronounced in DeepCGM (Fig. 9 f1-f3). In summary, the DeepCGM model effectively simulated crop growth patterns, producing results that were comparable to or even better than those of the knowledge-based ORYZA2000 model.

Due to the impact of random initialization, the results across 25 random seeds showed some variability. To assess model performance more consistently, we divided the growth period into four intervals and calculated the average RMSE along with the corresponding standard deviation. During the emergence to panicle initiation stage, the RMSE of DeepCGM was higher than that of ORYZA2000 (Fig. 10). In the subsequent development stages, DeepCGM outperformed ORYZA2000 in terms of PAI, WLV, WST, and WSO, though it was slightly inferior for WAGT (Fig. 10). This discrepancy may be because the WAGT process is simpler compared to other variables. As a result, ORYZA2000 is sufficient to capture WAGT dynamics but struggles with more complex processes related to other biomass components. Additionally, in DeepCGM, the need to balance components within $Loss_{fitting}$ may have led to improved accuracy in other variables at the cost of slight deterioration after flowering. For YIELD, since observations were only available after the maturity stage, the results presented correspond to harvest YIELD. Notably, the YIELD RMSE for DeepCGM was significantly lower than that of ORYZA2000 (Fig. 10). In summary, these results suggest that DeepCGM outperformed ORYZA2000 during most of the growth period. However, the larger simulation errors in the early development stages may be due to the lack of early-stage observations for training. Introducing additional knowledge constraints during the early growth stages may further enhance DeepCGM's performance.

3.2.2. Accuracy of different models

The overall accuracy of models and their simulation capabilities for six variables were evaluated using fitting loss (Fig. 11) and normalized index (Fig. 12) to identify the best-performing model (**E5**). The source data for both figures were from Appendix F. The performance of ORYZA2000 in these figures represents the optimal performance achievable by conventional crop models, serving as a benchmark for comparison.

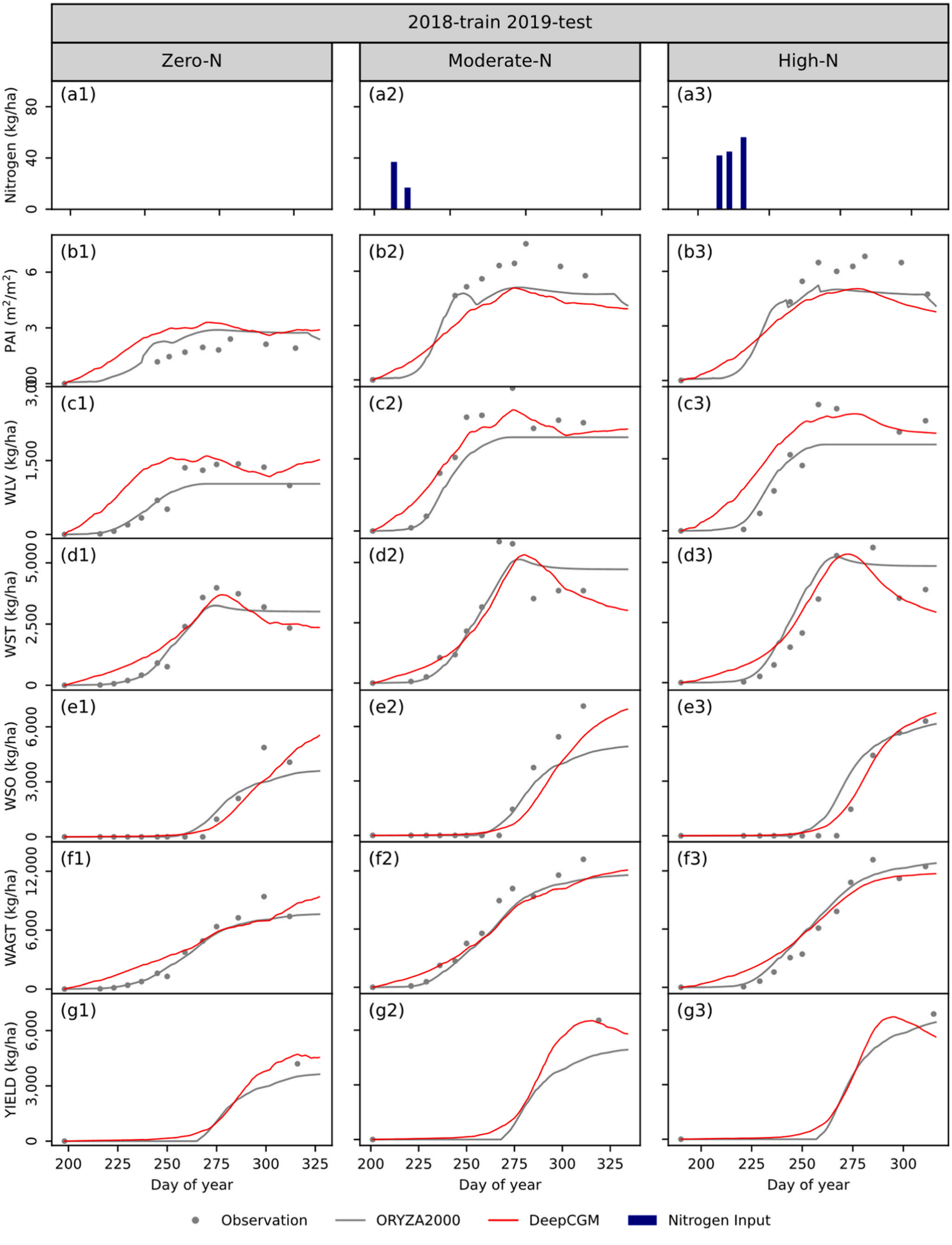


Fig. 9. Crop growth processes simulated by DeepCGM (random seed = 1) and ORYZA2000 for plots with varying fertilization levels.

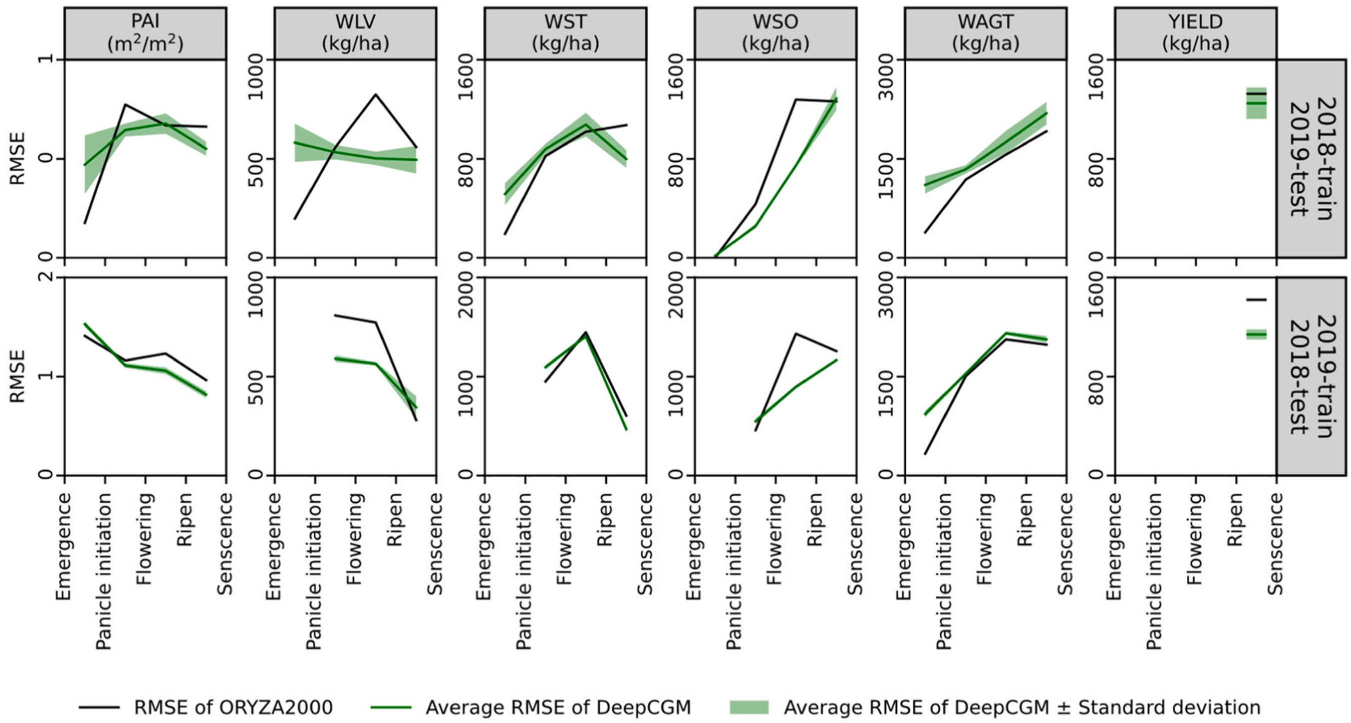


Fig. 10. The average RMSE and the corresponding standard deviation of DeepCGM and RMSE of ORYZA2000 at different growth stages.

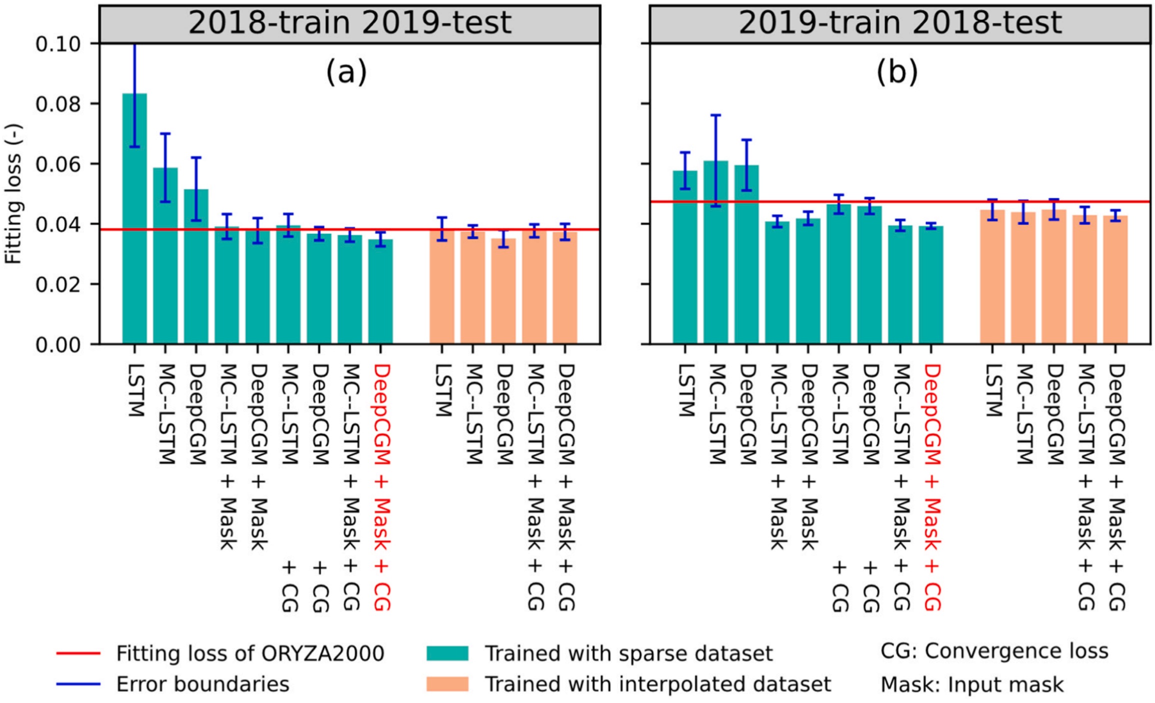


Fig. 11. Overall accuracy of different models trained with different strategies on sparse and augmented datasets. The model corresponding to the red x-axis label demotes the optimal model of this study.

As demonstrated on Fig. 11 a, when models were trained on sparse datasets using only fitting loss, their performance was consistently lower than that of the ORYZA2000 model. This suggests that deep learning models trained on small datasets, without additional constraints, could not outperform PB models. However, the accuracy of both MC-LSTM and DeepCGM improved significantly after incorporating input mask or convergence loss, with DeepCGM trained with convergence loss achieving superior performance over ORYZA2000 (Fig. 11 a). When both input mask and convergence loss were applied simultaneously, MC-LSTM and DeepCGM outperformed ORYZA2000 in both years, with DeepCGM demonstrating slightly better performance than MC-LSTM (Fig. 11 a). Compared to ORYZA2000, the overall accuracy of DeepCGM improves by 8.3 % (2019) and 16.9 % (2018). These results indicated that both input mask and convergence loss were beneficial for

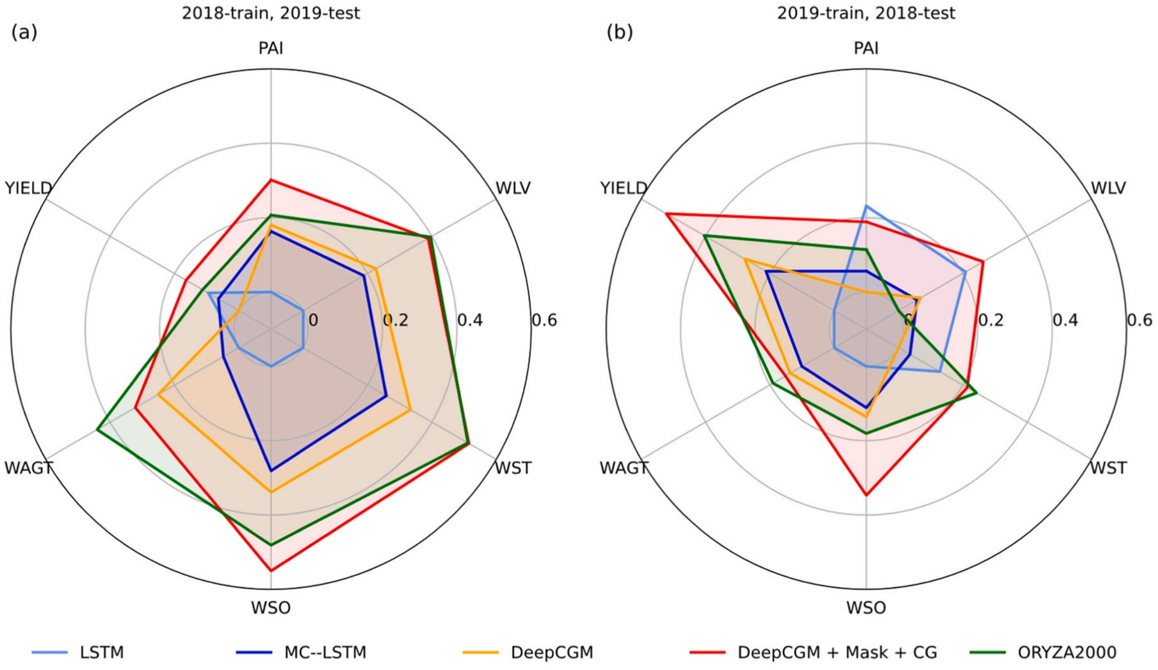


Fig. 12. The normalized index of different models trained by different strategies on sparse dataset. More detailed results of cases in E5 refer to Appendix F.

the model, with convergence loss being more effective. Results also demonstrated the structural superiority of DeepCGM over MC-LSTM. Furthermore, these results suggest the potential to outperform conventional PB crop models when training machine learning models with knowledge-guided constraints on sparse data.

When the models trained on augmented data, the overall accuracy of all models matched or outperformed that of the ORYZA2000 (Fig. 11 b). This improvement occurred because the training data were augmented based on ORYZA2000 simulations, allowing the models to simultaneously align with both the prior knowledge embedded in ORYZA2000 and the actual processes observed in the data. However, the pattern of the augmented data was influenced by ORYZA2000, which restricted the model's ability to fully learn from observations. This limitation resulted in lower accuracy compared to the knowledge-guided DeepCGM model. These results suggest that data augmentation remains an effective strategy for training models on small datasets, but it is less effective than the knowledge-guided approach proposed in this study.

The accuracy of six state variables is presented in Fig. 12. For the DeepCGM model trained with input mask and convergence loss, significant improvements over ORYZA2000 were observed in PAI, WLV, WSO, and YIELD, while the WAGT simulation accuracy for all models was lower than that of ORYZA2000. Additionally, the number of state variables in each model that outperformed ORYZA2000 was counted. As shown in Appendix F, among the twelve RMSE values (six variables × two years), eight variables in Cases 6, 10, and 16 demonstrated superior performance compared to ORYZA2000. Among these, Case 10 had the highest number (three) of best-performing variables. This confirms that, for simulating individual state variables, the knowledge-guided DeepCGM model remains the most effective. Appendix E provides scatter plots comparing the best model simulations with observed values, allowing for a visual comparison with the ORYZA2000 model's scatter plots in Appendix D.

Based on the overall performance metrics and individual state variable accuracy, DeepCGM with convergence loss and input mask should be considered the best model (E5). It achieved the lowest fitting loss and successfully simulated the greatest number of state variables with accuracy surpassing that of ORYZA2000.

3.3. Replication study

To demonstrate the validity and generalizability of the proposed model, we also conducted a replication case study of a three-year rice experiment totaling 122 plots (more details refer to Supplementary S4). The results demonstrated that DeepCGM can achieve better accuracy in most variables than classical ML models and ORYZA2000 model. The results also showed that DeepCGM is capable to infer the unobserved processes under the knowledge constraints.

4. Discussion

4.1. The complementarity of constraints and data

Training a deep learning model on a limited dataset is a significant challenge. Owing to the scarcity of training data, the learned complex relationships might actually arise from sampling noise (Srivastava et al., 2014). Various regularization techniques and constraints have proven effective in mitigating data scarcity issues in fields such as image recognition and natural language processing (Sari et al., 2019; Srivastava et al., 2014; Vidaurre et al., 2013). In natural science research, the distinct causal relationships and adherence to physical principles in these fields allow for the incorporation of domain-specific constraints (Han et al., 2023; Jia et al., 2019; Liu et al., 2022). However, applying constraints to a model should be based on a well-founded rationale rather than being arbitrary (Liu et al., 2022). In this study, the sparse observational dataset failed to provide effective constraints across the entire growth period, leading to poor model performance during time intervals without observations. Therefore, the knowledge of crop growth process was employed to constrain the model in terms of structure design, input selection, and loss function design. The utilization of constraints to mitigate data sparsity issue is primarily considered from the following perspectives:

First, given that crop growth is a temporal sequence, adjacent time steps are interdependent and mutually constraining. Therefore, it is essential to impose constraints on the temporal dimension to ensure that the biomass of each organ does not increase or decrease unreasonably (e.g., Fig. 5 b1-e1). The mass conservation structure of MC-LSTM effectively fulfills this requirement. Furthermore, considering crop physiological processes, we designed the DeepCGM structure to better align with crop biomass dynamics. Progressing from LSTM to MC-LSTM, and further to DeepCGM, the model exhibits an incremental integration of temporal dependencies within its structural design. As a result, its ability to capture real-world processes improves as more domain knowledge is integrated, particularly when trained on sparse data (Fig. 5 a1-f1, a2-f2 and a3-f3).

Second, the performance of the simple DeepCGM architecture remained inferior compared to ORYZA2000. Analyzing the simulated results revealed that while the existing structural constraints effectively ensured a smooth temporal simulation of total above-ground biomass, the biomass redistribution process required additional constraints for improved accuracy. To address this, we introduced convergence loss to enhance model stability. Additionally, an input mask was applied to the redistribution gate, guided by crop physiological knowledge. This approach excluded inputs unrelated to redistribution, reducing noise interference and further improving the model's performance.

The introduction of constraints can enforce the simulation results to adhere to physical laws, preventing model parameters from producing erroneous outputs. This approach helps the model learn crop growth patterns more effectively from data. However, the model's ability to learn correct patterns still depends on sufficient data support, as the quality of the output is ultimately determined by the quality of the input—otherwise, it results in a "garbage in, garbage out" scenario. For example, although MC-LSTM, aided by mass conservation and convergence loss constraints, successfully reduces fluctuations, it fails to capture the pattern of slow biomass accumulation during the early development stage (Fig. 5 a2-f2). If sufficient observational data were available, the model could learn this pattern directly from the data, as evidenced by models trained with augmented data (Fig. 6). Conversely, an abundance of data can significantly enhance model capability, particularly in settings such as controlled environments or high-throughput phenotyping platforms. However, given the complexity of real-world agricultural systems, which are influenced by numerous factors, solely relying on data to accurately reproduce crop growth processes is impractical. Moreover, even with large datasets, there is no guarantee that the model will strictly adhere to physical laws. For instance, fluctuations in organ biomass can still occur when training the model on augmented datasets (e.g., Fig. 6 a3 and b3). Therefore, the optimal strategy involves training models under knowledge-guided constraints while leveraging as much data as possible. This approach balances data-driven learning with physical realism, leading to more robust and accurate crop growth simulations.

4.2. Lessons learned for DeepCGM development

There have been numerous PB crop models, and several review articles have provided guidance on building and improving conventional crop models (Bouman, 2001; Pasley et al., 2023; Yin et al., 2021). However, apart from a recently published model based on the attention mechanism (Moon et al., 2023) and our previous research (Han et al., 2023), there are few machine learning-based crop growth models available for reference. Through this research, we have gained several key insights, particularly by comparing our model with conventional crop models and study of Moon et al. (2023).

Firstly, it is advisable to select variables for information memory between adjacent time steps in the following priority order: vector representations with physical meaning (e.g., the carbon vector in this study), variables with physical meaning (e.g., leaf biomass), and hidden states. In natural language processing and computer vision, features are often abstract and difficult to represent. A common way is to use hidden states as representations (Pasley et al., 2023). However, in the Soil-Plant-Atmosphere Continuum (SPAC) systems, system states can be described using physically meaningful variables, such as biomass or soil moisture content, which store the majority of system information. Although these physically meaningful variables may not fully capture all system details, they offer two major advantages over hidden states:(1) These variables help the model discard insignificant or low-value information, allowing it to focus on relevant data (e.g., ignoring the previous day's radiation level while retaining the biomass value at the end of the day); (2) They allow for direct adjustments within the model, such as applying redistribution loss in this study. In up-to-date practices, both PB model (e.g., ORYZA2000) and the attention-based model (Moon et al., 2023) utilize historical crop states as the system memory, reinforcing the effectiveness of this approach. However, using predefined physical variables limits the representational capacity of deep learning models and hinders further improvement. Because the number of such variables is inherently limited, increasing model capacity requires additional effort to define new variables, often necessitating extra assumptions. For example, ORYZA2000 uses both stem biomass and stem reserve biomass, whereas WOFOST only uses stem biomass (Bouman, 2001; Van Diepen et al., 1989). This two-variable approach allows ORYZA2000 to better model stem dynamics (Bouman, 2001), but it requires more experiments and equations to describe stem reserve biomass interactions. If these interactions are not properly defined, the additional variable may not contribute to model accuracy. A more fundamental and comprehensive approach is to use vector representations with physical meaning for information storage. These vectors do not correspond to specific crop state variables but can be used to derive them. This method retains the advantages of physically meaningful variables (e.g. the mass-conserving structure and the input mask) while eliminating the need to explicitly define equations, as deep learning models can autonomously learn these relationships. However, this perspective does not imply eliminating hidden states entirely, as some system states cannot yet be described by physical vectors due to an incomplete understanding of their mechanisms. For example, spikelet sterility due to extreme temperature cannot yet be fully defined (Bouman, 2001). Removing hidden states completely would limit the model's ability to learn unknown mechanisms from large datasets. Therefore, we recommend empirically evaluating hidden states in scenarios where data availability is limited.

Secondly, it is beneficial to modularize the model rather than amalgamating the computations of all processes. While deep learning parameters are often difficult to interpret, model structure can still be designed based on crop physiological knowledge. Structural constraints serve as an effective regularization technique, significantly enhancing model performance. In this study, the knowledge-based design of DeepCGM led to the highest accuracy among tested models. Similar benefits of structural design have been observed in other studies, such as those estimating N_2O emissions (Liu et al., 2022). Even deep learning models benefit from good modeling practices, as a modular architecture enables independent control and management of different subprocesses. For example, implementing an input mask allowed precise regulation of inputs to the redistribution module. This modular philosophy is also widely used in conventional crop models, where separating subprocesses helps minimize uncertainty in each (Pasley et al., 2023).

Lastly, the model should be conceptualized based on existing knowledge and given objectives. Before construction, a model should be conceptualized by selecting the core architecture (e.g., Recurrent Neural Network or Transformer), defining input-output configurations, identifying sub-processes and sub-modules, and considering model simplifications based on research objectives. For instance, this study utilized a Recurrent Neural Network based architecture to capture the strong temporal causality in crop growth. In contrast, Moon et al. (2023) used a Transformer model, which ignores the sequential dependency of crop growth states. When selecting input variables, only the necessary driving variables should be included based on the research objectives, rather than incorporating all available variables. For example, the objective of this study is to simulate carbon cycle based on mass conservation, thus excluding the simulation of water and nitrogen cycles. Consequently, temperature, radiation, cumulative nitrogen application, and DVS (simulated using ORYZA2000) were selected as driving factors, while water-related driving factors (such as precipitation, wind speed, and saturated vapor pressure) were excluded. Careful input selection also helps reduce redundant inputs, thereby minimizing overfitting risks, especially when training on sparse datasets (Moon et al., 2023). Thus, comprehensive conceptualization is essential before model construction. It aims to balance the research objectives with the available resource (data and knowledge), rather than simply choosing the most advanced structure or including the largest number of input features.

4.3. Limitations and future study

The development of crop models has evolved from initially considering only potential growth conditions to progressively incorporating water limitations, nitrogen limitations, and additional environmental factors (Bouman et al., 1996). At the same time, models have transitioned from generic frameworks to species- and cultivar-specific models (Wang et al., 2019). A similar progression can be applied to deep learning-based crop growth models. The current version of DeepCGM is designed to address the challenge of training a deep learning model using sparse observation dataset. As a result, it focuses on basic biomass simulation and does not yet account for water availability, cultivar differences, or soil texture. Consequently, the minimum data requirements for model training include weather data (radiation, temperature), management data(sowing date and nitrogen application recording) and crop labels (development stage, biomass and plant area index observations). Restricting this work only to biomass observation was due to the lack of availability of observational data. However, previous research has demonstrated that designing multiple parallel branches to handle different conserved variables (e.g., nitrogen and water-related variables) is an effective strategy (Bertels and Willems, 2023). A similar modular architecture could be developed in future research to incorporate water and nitrogen cycles. Additionally, LSTM models could serve as auxiliary models to simulate and represent non-mass-conserving variables, such as phenology, which MC-LSTM cannot effectively handle. Limited by its current structure, DeepCGM cannot yet be transferred to other locations or cultivars. A promising direction is to integrate DeepCGM with a static property recognizer (Tsai et al., 2021), to account for soil, cultivar, and species characteristics. Building on this, by further combining the soft weight sharing of hypernetworks (Ha et al., 2016) with the task-wise fine-tuning strategy of LoRA (Hu et al., 2021), it becomes possible to calibrate the model for specific cultivars, soils, or even particular subprocesses within the model.

The current DeepCGM model is constrained by data limitations, leading to inaccuracies during unobserved periods. These include the late-stage decline in yield, and the rapid growth of biomass in the early stages. Without sufficient observational data, the model struggles to identify correct physiological processes, and random factors may introduce unrealistic results. This occurs despite the model adhering to mass conservation principles. To address this limitation, two strategies can be considered: supplementing observational data with crop model simulations (without solely relying on them, as discussed in Section 3.1.3) and introducing additional knowledge-based constraints, such as penalizing the loss function for unrealistic late-stage yield declines. Despite the incorporating mass-conserving structure, convergence loss, and input mask, the utilization of existing knowledge in this study remains inadequate. Therefore, integrating a substantial amount of plant physiology-based knowledge is necessary. To improve accuracy, greater integration of plant physiological knowledge is necessary. Currently, knowledge integration in this model relies on manual design, whereas crop models already contain a wealth of domain knowledge. Therefore, further research is needed to efficiently assimilate knowledge from crop models into deep learning frameworks.

This study demonstrates that KGML models can outperform PB models in crop growth simulation. In addition to this, knowledge-guided machine learning offers new opportunities for agricultural model development. One advantage of KGML is its ability to integrate both data and knowledge, surpassing the capabilities of traditional crop models. With the ongoing advancements in phenotyping technologies, a data explosion is expected (Jin et al., 2020). In contrast, the hypothesis-validation approach used in PB models often requires lengthy iterations and is susceptible to biases from the developers' personal knowledge backgrounds (Shen et al., 2023). In the face of massive data, KGML offers a faster and bias-free alternative for knowledge extraction. Another advantage of KGML over purely data-driven models is its stronger extrapolation capabilities and improved interpretability. While data-driven models can directly extract knowledge from data, their findings are often difficult to interpret, despite aligning with the objectives of plant physiologists (Yin et al., 2021). On the other hand, PB models excel in extrapolation and interpretability but lack flexibility. By combining the strengths of both approaches, KGML establishes a balanced framework, positioning itself between data-driven and PB models, thereby advancing the development of agriculture-related modeling. Additionally, KGML can be easily extended to applications such as model coupling and data assimilation. Many PB models, including SWAP, WOFOST, and DSSAT, encapsulate extensive domain knowledge. However, integrating knowledge across these models remains challenging due to differences in programming languages, model assumptions, and limited human resources (Midingoyi et al., 2021). The KGML approach provides a promising solution by not only integrating existing crop models but also deriving new, more accurate knowledge from them. Recent studies further confirm that KGML can be effectively applied to crop growth modeling and data assimilation (Yang et al., 2023), reinforcing its potential for future advancements in agricultural sciences.

5. Conclusions

In this study, we propose a Deep learning Crop Growth Model (DeepCGM) with a mass-conserving architecture that adheres to the mechanism of crop growth. Knowledge-guided constrains were employed to train the model with sparse datasets, in the form of input mask, and convergence loss. This study makes two major contributions: 1) The proposal and open-sourcing of a deep learning crop growth model along with the corresponding dataset, which can serve as a benchmark for future research; 2) The first attempt to incorporate crop growth mechanisms into a deep learning model for modeling the multivariate crop growth process from a small dataset. By comparing DeepCGM with the traditional ORYZA2000 model and classic machine learning models on a two-year dataset, DeepCGM showed superior in overall accuracy and achieved lower RMSE in the time series simulation of plant area index, leaf biomass, stem biomass, grain biomass, and yield. A replication case study was also conducted to demonstrate the validity and generalizability of the proposed model. The results demonstrated that the proposed mass-conserving structure, convergence loss, and input mask significantly contributed to improving accuracy and aligning with real growth processes. Additionally, experiments using an augmented dataset revealed that, although data augmentation helps mitigate the impact of data sparsity, it also introduces structural errors, leading to lower accuracy compared to models trained with knowledge-based constraints. Tests on datasets with removed observations further demonstrated the complementary relationship between constraints and data availability. In summary, this study demonstrated that knowledge-guided machine learning can overcome structural errors due to the simplification in conventional crop models while retaining key crop growth mechanisms. The integration of biological mechanisms within deep learning frameworks provides valuable insights for modeling complex multivariate systems under sparse data conditions.

CRediT authorship contribution statement

Yu Jin: Data curation. **Athanasiadis Ioannis N.:** Writing – review &

editing, Supervision, Funding acquisition. **Shi Liangsheng:** Writing – review & editing, Supervision, Funding acquisition. **Yang Qi:** Data curation. **Han Jingye:** Writing – original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization.

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.

Acknowledgment

This study was supported by the National Natural Science Foundation of China Grant 52425901; the Open Research Fund of Guangxi Key Laboratory of Water Engineering Materials and Structures, Guangxi Institute of Water Resources Research, under grant NO: GXHRI-WEMS-2020-06, the European Union Horizon Europe Research and Innovation program (Grant 101094587, PHENET), the Digital Europe Programme under Grant agreement AgrifoodTEF - Test and Experiment Facilities for the Agri-Food Domain (ID 101100622), and the China Scholarship Council for 1.5 year-long research visit at the Wageningen University & Research.

Appendix A. Abbreviations

	Abbreviations	Definition
Model Input Variables	Rad	Daily Radiation
	T_{min}	Daily minimum temperature
	T_{max}	Daily maximum temperature
	N_{cum}	Cumulative fertilization
	DVS	Development stage (Simulated by ORYZA2000)
Model Output and Observation Variables	PAI	Plant area index
	WLV	Leaf biomass
	WST	Stem biomass
	WSO	Storage organ biomass
	WAGT	Above ground biomass
	YIELD	Yield
Intermediate Variables of DeepCGM	Aux	Auxiliary input, consisting of **Model Input Variables**
	X	Input for gates, consisting of Aux and C
	C	Carbon vector representation
	C_i	Carbon representation element, i is the index
	$C_{potential}$	Potential carbon input
	C_{in}	Actual carbon input
	C_{MR}	Maintenance respiration costs
	$C^t_{patition}$	Partitioned carbon
	C_{grow}	Gross daily carbon growth
Gate Variables of DeepCGM	I	Light interception and carbon assimilation gate, denoting the intercepted and assimilated proportion of penitential carbon
	MR	Maintenance respiration gate, denoting the proportion of maintenance respiration to the cumulative carbon.
	P	Partition gate, denoting the proportion of carbon allocated for the growth of different parts.
	GR	Growth respiration gate, denoting the proportion of carbon retained after growth respiration
	R	redistribution gate, denoting the allocation ratio of carbon flow among carbon vector components during redistribution
Others	CG	Convergence loss
	DeepCGM	Deep learning Crop Growth Model
	KGML	Knowledge guided machine learning
	LSTM	Long Short-Term Memory model
	MC-LSTM	Mass-Conserving LSTM
	ML	Machine learning
	NI	Normalized index
	PB	Process based

Note: Carbon can be converted to biomass using a fixed coefficient in the final mapping step.

Appendix B. : Formulas of DeepCGM

$$X^t = \left[C^{t-1}, Aux^t\right]\# \tag{A1}$$

$$I^t = sigmoid(w_I X^t + b_i)\# \tag{A2}$$

$$MR^t = sigmoid(w_{MR} X^t + b_{MR})\# \tag{A3}$$

$$P^t = softmax(w_P X^t + b_P)\# \tag{A4}$$

$$GR^t = sigmoid(w_{GR}X^t + b_{GR})\# \quad (A5)$$

$$R^t = softmax(w_R X^t + b_R)\# \quad (A6)$$

$$C_{in}^t = C_{potenmtial}^t \odot I^t \# \quad (A7)$$

$$C_M^t = MR^t \cdot C^{t-1} \# \quad (A8)$$

$$C_{patition}^t = P^t \cdot (C_{in}^t - C_M^t)\# \quad (A9)$$

$$C_{grow}^t = C_{patition}^t \odot GR^t \# \quad (A10)$$

$$C^t = R^t \cdot (C^{t-1} + C_{grow}^t)\# \quad (A11)$$

$$WLV^t = k_{WLV} \sum_{i=1}^{n} C_i^t \# \quad (A12)$$

$$WST^t = k_{WST} \sum_{i=n+1}^{2n} C_i^t \# \quad (A13)$$

$$WSO^t = k_{WSO} \sum_{i=2n+1}^{3n} C_i^t \# \quad (A14)$$

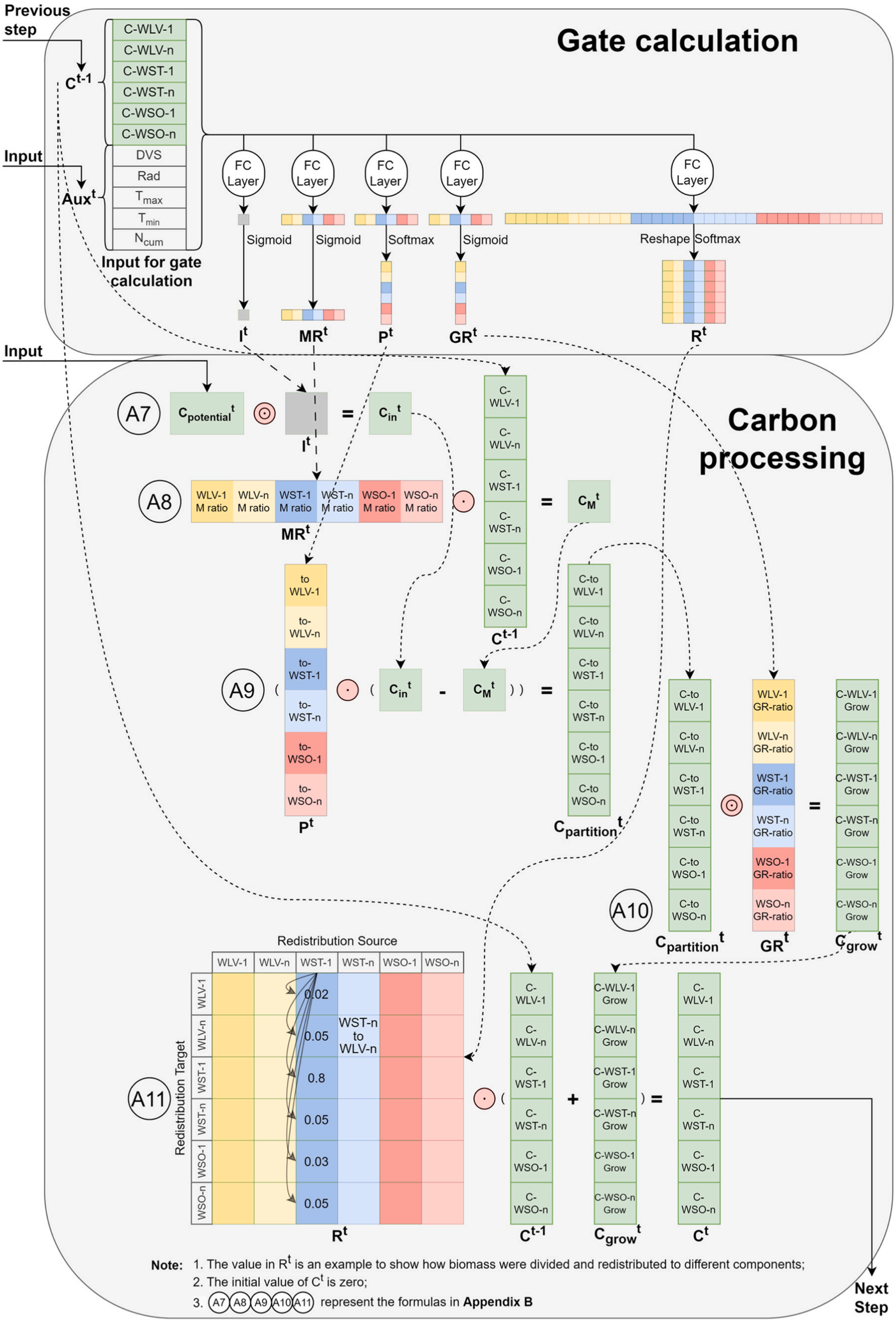
$$WAGT^t = WLV^t + \quad WST^t + \quad WSO^t \# \quad (A15)$$

$$YIELD^t = \sum_{i=2n+1}^{3n} (w_{YIELD} C_i^t)\# \quad (A16)$$

$$PAI^t = \sum_{i=1}^{3n} (w_{PAI} C_i^t)\# \quad (A17)$$

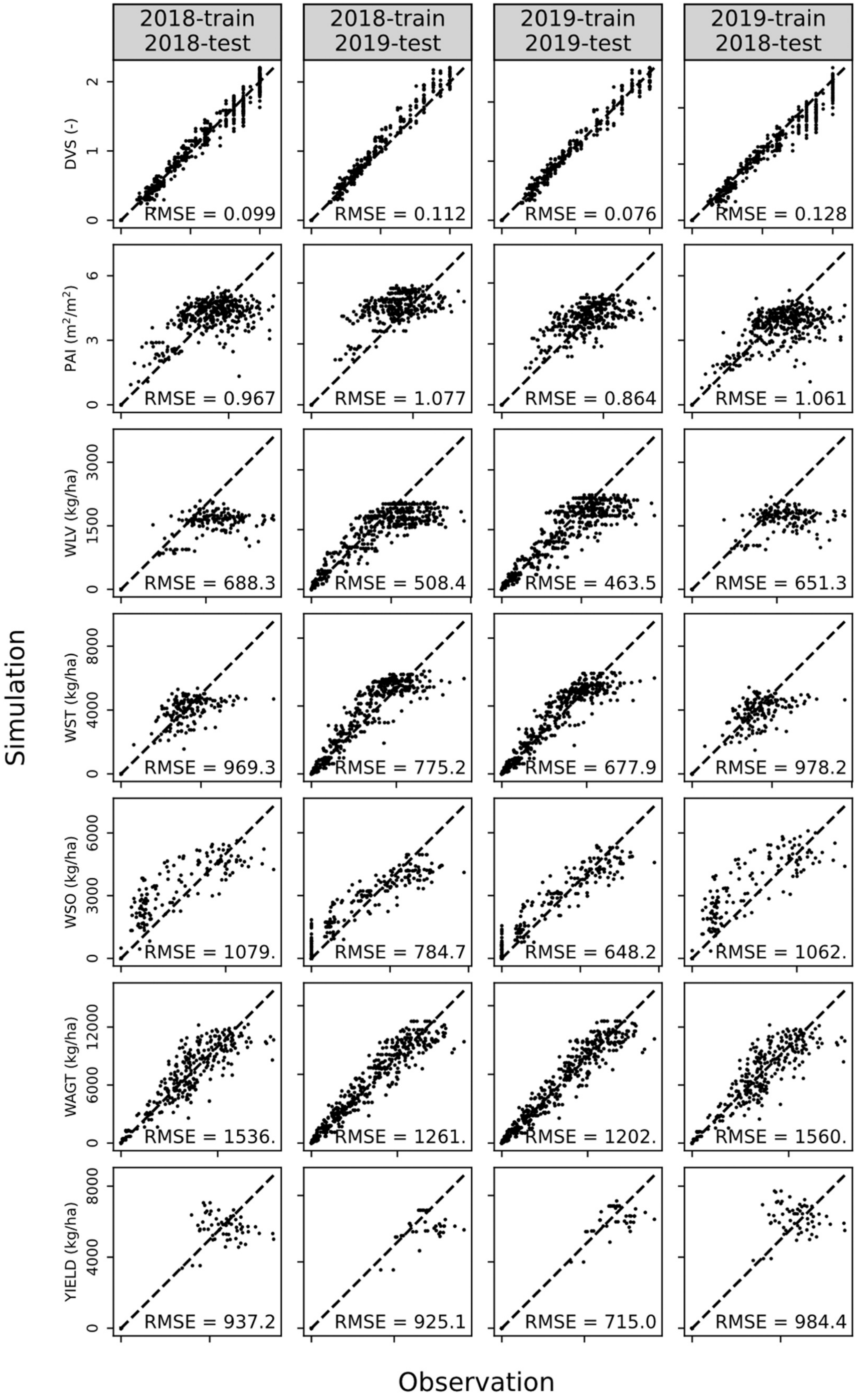
The variables and operators in the formulas are defined in Fig. 3 and **Appendix A**, where W and b represent the trainable model weights and bias parameters, respectively. Formula (A1) constructs the gate input; formulas (A2) to (A6) correspond to the calculations for various gates as described in Step 1. Specifically, the interception-assimilation gate, maintenance respiration gate, and growth respiration gate (I^t、 MR^t and GR^t) use the sigmoid function, as their outputs are scalar values between 0 and 1, representing the proportion of intercepted-assimilated and consumed carbon. This ensures that the intercepted-assimilated carbon is less than the input carbon and the consumed carbon is less than the stored carbon, thus achieving mass conservation. The partitioning gate and redistribution gate (P^t and R^t) use the softmax function, as their outputs are vectors with elements that sum to 1, representing the proportions of (re)distributed carbon to each elements, thereby maintaining mass conservation before and after (re)distribution. Formula (A7) represents the light interception and carbon assimilation process corresponding to Step 2; formula (A8) represents the maintenance respiration process corresponding to Step 3; formula (A9) represents the carbon partitioning process corresponding to Step 4; formula (A10) represents the growth respiration process corresponding to Step 5; formula (A11) represents the redistribution process corresponding to Step 6. Formulas (A12) to (A17) represent the mapping function that translates the carbon vector representation into the crop state. k denotes the mass fraction carbon in biomass of different organ (kg carbon kg^{-1} biomass). t denotes the time step. Leaf, stem and storage organ biomass is internally represented in a vector representation of size n.

Appendix C. The detail process of DeepCGM. The carbon vector from previous step were concatenated with the auxiliary driven factor as input of the all the gates. The calculated gates were then used to control the carbon process, including light interception and carbon assimilation (A7), maintenance respiration (A8), partition (A9), growth respiration (A10) and redistribution (A11)

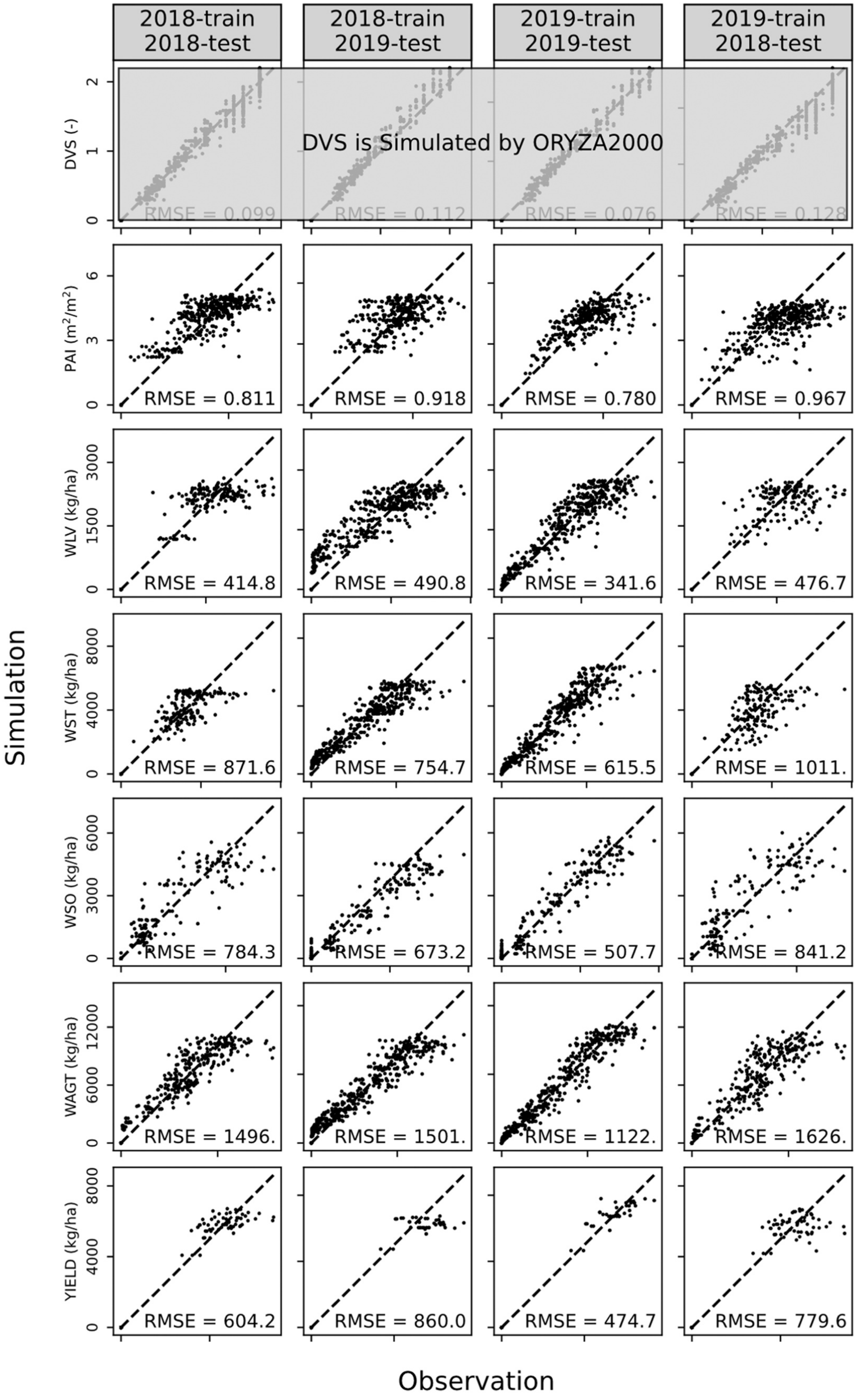


Note: 1. The value in R^t is an example to show how biomass were divided and redistributed to different components;
2. The initial value of C^t is zero;
3. A7 A8 A9 A10 A11 represent the formulas in **Appendix B**

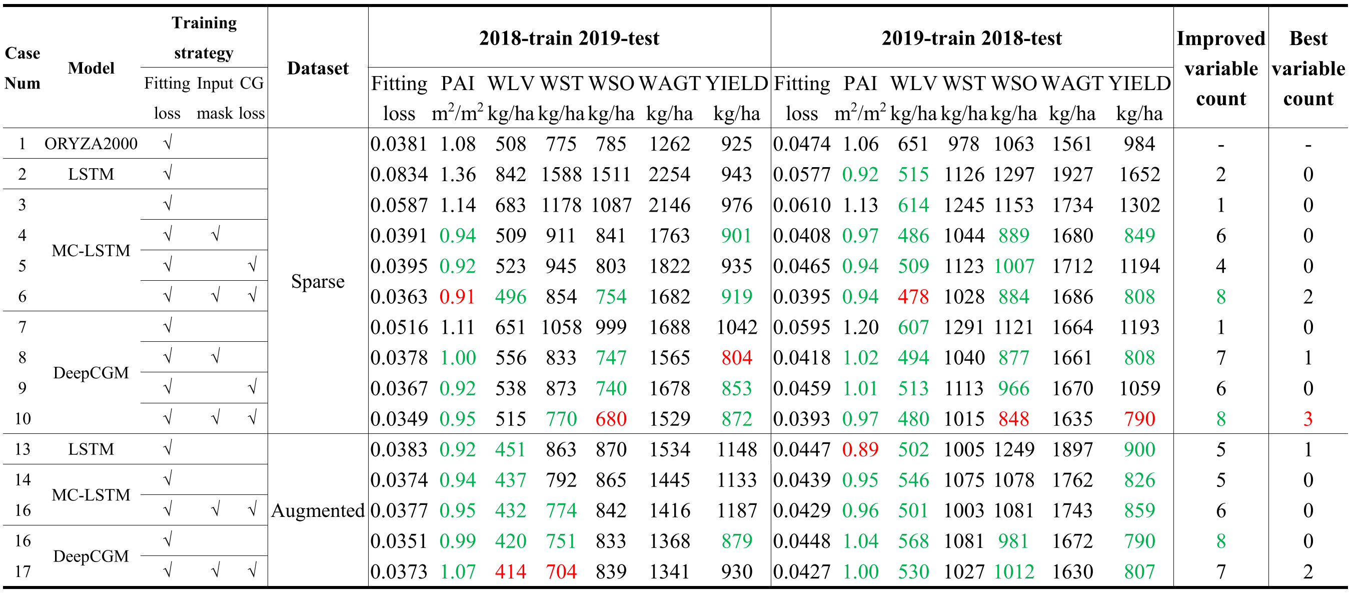
Appendix D. The scatter plot of observation and simulation by calibrated ORYZA2000



Appendix E. The scatter plot of observation and simulation by calibrated DeepCGM trained with all constraints (average result)



Appendix F. The fitting loss and the RMSE of six simulated variables of different cases



Case Num	Model	Training strategy			Dataset	2018-train 2019-test							2019-train 2018-test							Improved variable count	Best variable count
		Fitting loss	Input mask	CG loss		Fitting loss	PAI m^2/m^2	WLV kg/ha	WST kg/ha	WSO kg/ha	WAGT kg/ha	YIELD kg/ha	Fitting loss	PAI m^2/m^2	WLV kg/ha	WST kg/ha	WSO kg/ha	WAGT kg/ha	YIELD kg/ha		
1	ORYZA2000	√			Sparse	0.0381	1.08	508	775	785	1262	925	0.0474	1.06	651	978	1063	1561	984	-	-
2	LSTM	√				0.0834	1.36	842	1588	1511	2254	943	0.0577	0.92	515	1126	1297	1927	1652	2	0
3	MC-LSTM	√				0.0587	1.14	683	1178	1087	2146	976	0.0610	1.13	614	1245	1153	1734	1302	1	0
4		√	√			0.0391	0.94	509	911	841	1763	901	0.0408	0.97	486	1044	889	1680	849	6	0
5		√		√		0.0395	0.92	523	945	803	1822	935	0.0465	0.94	509	1123	1007	1712	1194	4	0
6		√	√	√		0.0363	0.91	496	854	754	1682	919	0.0395	0.94	478	1028	884	1686	808	8	2
7	DeepCGM	√				0.0516	1.11	651	1058	999	1688	1042	0.0595	1.20	607	1291	1121	1664	1193	1	0
8		√	√			0.0378	1.00	556	833	747	1565	804	0.0418	1.02	494	1040	877	1661	808	7	1
9		√		√		0.0367	0.92	538	873	740	1678	853	0.0459	1.01	513	1113	966	1670	1059	6	0
10		√	√	√		0.0349	0.95	515	770	680	1529	872	0.0393	0.97	480	1015	848	1635	790	8	3
13	LSTM	√			Augmented	0.0383	0.92	451	863	870	1534	1148	0.0447	0.89	502	1005	1249	1897	900	5	1
14	MC-LSTM	√				0.0374	0.94	437	792	865	1445	1133	0.0439	0.95	546	1075	1078	1762	826	5	0
16		√	√	√		0.0377	0.95	432	774	842	1416	1187	0.0429	0.96	501	1003	1081	1743	859	6	0
16	DeepCGM	√				0.0351	0.99	420	751	833	1368	879	0.0448	1.04	568	1081	981	1672	790	8	0
17		√	√	√		0.0373	1.07	414	704	839	1341	930	0.0427	1.00	530	1027	1012	1630	807	7	2

Appendix G. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.fcr.2025.109912.

Data Availability

I have shared the link of my data in the manuscript

References

Bertels, D., Willems, P., 2023. Physics-informed machine learning method for modelling transport of a conservative pollutant in surface water systems. J. Hydrol. 619, 129354.

Blank, J., Deb, K., 2020. Pymoo: Multi-Objective Optimization in Python. IEEE Access 8, 89497–89509. https://doi.org/10.1109/ACCESS.2020.2990567.

Bouman, B.A.M., 2001. ORYZA2000: modeling lowland rice. IRRI, Manila, Philippines.

Bouman, B., Van Keulen, H., Van Laar, H., Rabbinge, R., 1996. The 'School of de Wit' crop growth simulation models: a pedigree and historical overview. Agric. Syst. 52, 171–198.

Brisson, N., Gary, C., Justes, E., Roche, R., Mary, B., Ripoche, D., Zimmer, D., Sierra, J., Bertuzzi, P., Burger, P., Bussière, F., Cabidoche, Y.M., Cellier, P., Debaeke, P., Gaudillère, J.P., Hénault, C., Maraux, F., Seguin, B., Sinoquet, H., 2003. An overview of the crop model stics. Eur. J. Agron. 18, 309–332. https://doi.org/10.1016/S1161-0301(02)00110-7.

De Wit, A., Boogaard, H., Fumagalli, D., Janssen, S., Knapen, R., Van Kraalingen, D., Supit, I., Van Der Wijngaart, R., Van Diepen, K., 2019. 25 years of the WOFOST cropping systems model. Agric. Syst. 168, 154–167. https://doi.org/10.1016/j.agsy.2018.06.018.

Donatelli, M., Magarey, R.D., Bregaglio, S., Willocquet, L., Whish, J.P.M., Savary, S., 2017. Modelling the impacts of pests and diseases on agricultural systems. Agric. Syst. 155, 213–224. https://doi.org/10.1016/j.agsy.2017.01.019.

Droutsas, I., Challinor, A.J., Deva, C.R., Wang, E., 2022. Integration of machine learning into process-based modelling to improve simulation of complex crop responses. silico Plants 4, diac017. https://doi.org/10.1093/insilicoplants/diac017.

Everingham, Y., Sexton, J., Skocaj, D., Inman-Bamber, G., 2016. Accurate prediction of sugarcane yield using a random forest algorithm. Agron. Sustain. Dev. 36, 27. https://doi.org/10.1007/s13593-016-0364-z.

Ewert, F., Rötter, R.P., Bindi, M., Webber, H., Trnka, M., Kersebaum, K.C., Olesen, J.E., Van Ittersum, M.K., Janssen, S., Rivington, M., Semenov, M.A., Wallach, D., Porter, J.R., Stewart, D., Verhagen, J., Gaiser, T., Palosuo, T., Tao, F., Nendel, C., Roggero, P.P., Bartošová, L., Asseng, S., 2015. Crop modelling for integrated assessment of risk to food production from climate change. Environ. Model. Softw. 72, 287–303. https://doi.org/10.1016/j.envsoft.2014.12.003.

Fan, X.-R., Kang, M.-Z., Heuvelink, E., De Reffye, P., Hu, B.-G., 2015. A knowledge-and-data-driven modeling approach for simulating plant growth: A case study on tomato growth. Ecol. Model. 312, 363–373. https://doi.org/10.1016/j.ecolmodel.2015.06.006.

Feng, P., Wang, B., Liu, D.L., Waters, C., Xiao, D., Shi, L., Yu, Q., 2020. Dynamic wheat yield forecasts are improved by a hybrid approach using a biophysical model and machine learning technique. Agric. For. Meteorol. 285–286, 107922. https://doi.org/10.1016/j.agrformet.2020.107922.

Frame, J.M., Kratzert, F., Gupta, H.V., Ullrich, P., Nearing, G.S., 2023. On strictly enforced mass conservation constraints for modelling the Rainfall-Runoff process. Hydrol. Process. 37, e14847.

Frame, J.M., Kratzert, F., Klotz, D., Gauch, M., Shalev, G., Gilon, O., Qualls, L.M., Gupta, H.V., Nearing, G.S., 2022. Deep learning rainfall–runoff predictions of extreme events. Hydrol. Earth Syst. Sci. 26, 3377–3392.

Guo, C., Tang, Y., Lu, J., Zhu, Y., Cao, W., Cheng, T., Zhang, L., Tian, Y., 2019. Predicting wheat productivity: Integrating time series of vegetation indices into crop modeling via sequential assimilation. Agric. For. Meteorol. 272–273, 69–80. https://doi.org/10.1016/j.agrformet.2019.01.023.

Ha, D., Dai, A., Le, Q.V., 2016. HyperNetworks. https://doi.org/10.48550/arXiv.1609.09106.

Han, J., Shi, L., Pylianidis, C., Yang, Q., Athanasiadis, I.N., 2023. DeepOryza: A knowledge-guided machine learning model for rice growth simulation.

Hoedt, P.-J., Kratzert, F., Klotz, D., Halmich, C., Holzleitner, M., Nearing, G.S., Hochreiter, S., Klambauer, G., 2021. Mc-lstm: Mass-conserving lstm, in: International Conference on Machine Learning. PMLR, pp. 4275–4286.

Holzworth, D.P., Huth, N.I., deVoil, P.G., Zurcher, E.J., Herrmann, N.I., McLean, G., Chenu, K., Van Oosterom, E.J., Snow, V., Murphy, C., Moore, A.D., Brown, H., Whish, J.P.M., Verrall, S., Fainges, J., Bell, L.W., Peake, A.S., Poulton, P.L., Hochman, Z., Thorburn, P.J., Gaydon, D.S., Dalgliesh, N.P., Rodriguez, D., Cox, H., Chapman, S., Doherty, A., Teixeira, E., Sharp, J., Cichota, R., Vogeler, I., Li, F.Y., Wang, E., Hammer, G.L., Robertson, M.J., Dimes, J.P., Whitbread, A.M., Hunt, J., Van Rees, H., McClelland, T., Carberry, P.S., Hargreaves, J.N.G., MacLeod, N., McDonald, C., Harsdorf, J., Wedgwood, S., Keating, B.A., 2014. APSIM – Evolution towards a new generation of agricultural systems simulation. Environ. Model. Softw. 62, 327–350. https://doi.org/10.1016/j.envsoft.2014.07.009.

Hornberger, G.M., Spear, R.C., 1981. Approach to the preliminary analysis of environmental systems. J. Environ. Mgmt 12, 7–18.

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., 2021. LoRA: Low-Rank Adaptation of Large Language Models. https://doi.org/10.48550/arXiv.2106.09685.

Jia, X., Willard, J., Karpatne, A., Read, J., Zwart, J., Steinbach, M., Kumar, V., 2019. Physics guided RNNs for modeling dynamical systems: A case study in simulating lake temperature profiles, in: Proceedings of the 2019 SIAM International Conference on Data Mining. Presented at the Society for Industrial and Applied Mathematics, pp. 558–566.

Jin, X., Zarco-Tejada, P.J., Schmidhalter, U., Reynolds, M.P., Hawkesford, M.J., Varshney, R.K., Yang, T., Nie, C., Li, Z., Ming, B., et al., 2020. High-throughput estimation of crop traits: A review of ground and aerial phenotyping platforms. IEEE Geosci. Remote Sens. Mag. 9, 200–231.

Jones, J.W., Antle, J.M., Basso, B., Boote, K.J., Conant, R.T., Foster, I., Godfray, H.C.J., Herrero, M., Howitt, R.E., Janssen, S., Keating, B.A., Munoz-Carpena, R., Porter, C. H., Rosenzweig, C., Wheeler, T.R., 2017. Brief history of agricultural systems modeling. Agric. Syst. 155, 240–254. https://doi.org/10.1016/j.agsy.2016.05.014.

Kallenberg, M.G.J., Maestrini, B., Bree, R. van, Ravensbergen, P., Pylianidis, C., Evert, F. van, Athanasiadis, I.N., 2023. Integrating processed-based models and machine learning for crop yield prediction. https://doi.org/10.48550/arXiv.2307.13466.

Kaneko, T., Nomura, K., Yasutake, D., Iwao, T., Okayasu, T., Ozaki, Y., Mori, M., Hirota, T., Kitano, M., 2022. A canopy photosynthesis model based on a highly generalizable artificial neural network incorporated with a mechanistic understanding of single-leaf photosynthesis. Agric. For. Meteorol. 323, 109036. https://doi.org/10.1016/j.agrformet.2022.109036.

Karpatne, A., Atluri, G., Faghmous, J., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N., Kumar, V., 2017. Theory-guided Data Science: A New Paradigm for Scientific Discovery from Data. IEEE Trans. Knowl. Data Eng. 29, 2318–2331. https://doi.org/10.1109/TKDE.2017.2720168.

Kashinath, K., Mustafa, M., Albert, A., Wu, J.-L., Jiang, C., Esmaeilzadeh, S., Azizzadenesheli, K., Wang, R., Chattopadhyay, A., Singh, A., Manepalli, A., Chirila, D., Yu, R., Walters, R., White, B., Xiao, H., Tchelepi, H.A., Marcus, P., Anandkumar, A., Hassanzadeh, P., Prabhat, 2021. Physics-informed machine learning: case studies for weather and climate modelling. Philos. Trans. R. Soc. A. 379, 20200093. https://doi.org/10.1098/rsta.2020.0093.

Kawakita, S., Yamasaki, M., Teratani, R., Yabe, S., Kajiya-Kanegae, H., Yoshida, H., Fushimi, E., Nakagawa, H., 2024. Dual ensemble approach to predict rice heading date by integrating multiple rice phenology models and machine learning-based genetic parameter regression models. Agric. For. Meteorol. 344, 109821. https://doi.org/10.1016/j.agrformet.2023.109821.

Keating, B.A., Carberry, P.S., Hammer, G.L., Probert, M.E., Robertson, M.J., Holzworth, D., Huth, N.I., Hargreaves, J.N.G., Meinke, H., Hochman, Z., McLean, G., Verburg, K., Snow, V., Dimes, J.P., Silburn, M., Wang, E., Brown, S., Bristow, K.L., Asseng, S., Chapman, S., McCown, R.L., Freebairn, D.M., Smith, C.J., 2003. An overview of APSIM, a model designed for farming systems simulation. Eur. J. Agron. 18, 267–288. https://doi.org/10.1016/S1161-0301(02)00108-9.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. IEEE, Perth, WA, Australia, pp. 1942–1948. https://doi.org/10.1109/ICNN.1995.488968.

Kingma, D.P., Ba, J., 2017. Adam: A Method for Stochastic Optimization. https://doi.org/10.48550/arXiv.1412.6980.

Li, T., Angeles, O., Marcaida, M., Manalo, E., Manalili, M.P., Radanielson, A., Mohanty, S., 2017. From ORYZA2000 to ORYZA (v3): An improved simulation model for rice in drought and nitrogen-deficient environments. Agric. For. Meteorol. 237–238, 246–256. https://doi.org/10.1016/j.agrformet.2017.02.025.

Liu, G., Migliavacca, M., Reimers, C., Kraft, B., Reichstein, M., Richardson, A.D., Wingate, L., Delpierre, N., Yang, H., Winkler, A.J., 2024. DeepPhenoMem V1.0: deep learning modelling of canopy greenness dynamics accounting for multi-variate meteorological memory effects on vegetation phenology. Geosci. Model Dev. 17, 6683–6701. https://doi.org/10.5194/gmd-17-6683-2024.

Liu, L., Xu, S., Tang, J., Guan, K., Griffis, T.J., Erickson, M.D., Frie, A.L., Jia, X., Kim, T., Miller, L.T., Peng, B., Wu, S., Yang, Y., Zhou, W., Kumar, V., Jin, Z., 2022. KGML-ag: a modeling framework of knowledge-guided machine learning to simulate agroecosystems: a case study of estimating N<sub>2</sub>O emission using data from mesocosm experiments. Geosci. Model Dev. 15, 2839–2858. https://doi.org/10.5194/gmd-15-2839-2022.

Midingoyi, C.A., Pradal, C., Enders, A., Fumagalli, D., Raynal, H., Donatelli, M., Athanasiadis, I.N., Porter, C., Hoogenboom, G., Holzworth, D., Garcia, F., Thorburn, P., Martre, P., 2021. Crop2ML: An open-source multi-language modeling framework for the exchange and reuse of crop model components. Environ. Model. Softw. 142, 105055. https://doi.org/10.1016/j.envsoft.2021.105055.

Moon, T., Kim, D., Kwon, S., Son, J.E., 2023. Process-Based Crop Modeling for High Applicability with Attention Mechanism and Multitask Decoders. Plant Phenomics 5, 0035. https://doi.org/10.34133/plantphenomics.0035.

Pasley, H., Brown, H., Holzworth, D., Whish, J., Bell, L., Huth, N., 2023. How to build a crop model. A review. Agron. Sustain. Dev. 43, 2. https://doi.org/10.1007/s13593-022-00854-9.

Paudel, D., Boogaard, H., De Wit, A., Janssen, S., Osinga, S., Pylianidis, C., Athanasiadis, I.N., 2021. Machine learning for large-scale crop yield forecasting. Agric. Syst. 187, 103016. https://doi.org/10.1016/j.agsy.2020.103016.

Paudel, D., De Wit, A., Boogaard, H., Marcos, D., Osinga, S., Athanasiadis, I.N., 2023. Interpretability of deep learning models for crop yield forecasting. Comput. Electron. Agric. 206, 107663.

Pearl, J., 2019. The limitations of opaque learning machines. Possib-.-. minds 25, 13–19.

Pylianidis, C., Kallenberg, M.G.J., Athanasiadis, I.N., 2024. Domain adaptation with transfer learning for pasture digital twins. Environ. Data Science 3, e8. https://doi.org/10.1017/eds.2024.6.

Pylianidis, C., Snow, V., Overweg, H., Osinga, S., Kean, J., Athanasiadis, I.N., 2022. Simulation-assisted machine learning for operational digital twins. Environ. Model. Softw. 148, 105274. https://doi.org/10.1016/j.envsoft.2021.105274.

Read, J.S., Jia, X., Willard, J., Appling, A.P., Zwart, J.A., Oliver, S.K., Karpatne, A., Hansen, G.J.A., Hanson, P.C., Watkins, W., Steinbach, M., Kumar, V., 2019. Process-Guided Deep Learning Predictions of Lake Water Temperature. Water Resour. Res. 55, 9173–9190. https://doi.org/10.1029/2019WR024922.

Rosenzweig, C., Jones, J.W., Hatfield, J.L., Ruane, A.C., Boote, K.J., Thorburn, P., Antle, J.M., Nelson, G.C., Porter, C., Janssen, S., Asseng, S., Basso, B., Ewert, F., Wallach, D., Baigorria, G., Winter, J.M., 2013. The Agricultural Model Intercomparison and Improvement Project (AgMIP): Protocols and pilot studies. Agric. For. Meteorol. 170, 166–182. https://doi.org/10.1016/j.agrformet.2012.09.011.

Rudin, C., 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. 1, 206–215.

Sari, E., Belbahri, M., Nia, V.P., 2019. How does batch normalization help binary training? arXiv preprint arXiv:1909.09139.

Shen, C., Appling, A.P., Gentine, P., Bandai, T., Gupta, H., Tartakovsky, A., Baity-Jesi, M., Fenicia, F., Kifer, D., Li, L., Liu, X., Ren, W., Zheng, Y., Harman, C.J., Clark, M., Farthing, M., Feng, D., Kumar, P., Aboelyazeed, D., Rahmani, F., Song, Y., Beck, H.E., Bindas, T., Dwivedi, D., Fang, K., Höge, M., Rackauckas, C., Mohanty, B., Roy, T., Xu, C., Lawson, K., 2023. Differentiable modelling to unify machine learning and physical models for geosciences. Nat. Rev. Earth Environ. 4, 552–567. https://doi.org/10.1038/s43017-023-00450-9.

Simunek, J., Van Genuchten, M.T., Sejna, M., 2005. The HYDRUS-1D software package for simulating the one-dimensional movement of water, heat, and multiple solutes in variably-saturated media. Univ. Calif. -Riverside Res. Rep. 3, 1–240.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958.

Tsai, W.-P., Feng, D., Pan, M., Beck, H., Lawson, K., Yang, Y., Liu, J., Shen, C., 2021. From calibration to parameter learning: Harnessing the scaling effects of big data in geoscientific modeling. Nat. Commun. 12, 5988. https://doi.org/10.1038/s41467-021-26107-z.

Van Dam, J.C., Huygen, J., Wesseling, J., Feddes, R., Kabat, P., Van Walsum, P., Groenendijk, P., Van Diepen, C., 1997. Theory of SWAP version 2.0; Simulation of water flow, solute transport and plant growth in the soil-water-atmosphere-plant environment. DLO Winand Star. Cent.

Van Diepen, C.A., Wolf, J., Van Keulen, H., Rappoldt, C., 1989. WOFOST: a simulation model of crop production. Soil Use Manag. 5, 16–24. https://doi.org/10.1111/j.1475-2743.1989.tb00755.x.

Vidaurre, D., Bielza, C., Larranaga, P., 2013. A survey of L1 regression. Int. Stat. Rev. 81, 361–387.

Von Rueden, L., Mayer, S., Beckh, K., Georgiev, B., Giesselbach, S., Heese, R., Kirsch, B., Walczak, M., Pfrommer, J., Pick, A., Ramamurthy, R., Garcke, J., Bauckhage, C., Schuecker, J., 2021. Informed Machine Learning - A Taxonomy and Survey of Integrating Prior Knowledge into Learning Systems, 1–1 IEEE Trans. Knowl. Data Eng.. https://doi.org/10.1109/TKDE.2021.3079836.

Wallach, D., Palosuo, T., Thorburn, P., Hochman, Z., Gourdain, E., Andrianasolo, F., Asseng, S., Basso, B., Buis, S., Crout, N., Dibari, C., Dumont, B., Ferrise, R., Gaiser, T., Garcia, C., Gayler, S., Ghahramani, A., Hiremath, S., Hoek, S., Horan, H., Hoogenboom, G., Huang, M., Jabloun, M., Jansson, P.-E., Jing, Q., Justes, E., Kersebaum, K.C., Klosterhalfen, A., Launay, M., Lewan, E., Luo, Q., Maestrini, B., Mielenz, H., Moriondo, M., Nariman Zadeh, H., Padovan, G., Olesen, J.E., Poyda, A., Priesack, E., Pullens, J.W.M., Qian, B., Schütze, N., Shelia, V., Souissi, A., Specka, X., Srivastava, A.K., Stella, T., Streck, T., Trombi, G., Wallor, E., Wang, J., Weber, T.K. D., Weihermüller, L., De Wit, A., Wöhling, T., Xiao, L., Zhao, C., Zhu, Y., Seidel, S.J., 2021. The chaos in calibrating crop models: Lessons learned from a multi-model calibration exercise. Environ. Model. Softw. 145, 105206. https://doi.org/10.1016/j.envsoft.2021.105206.

Wang, E., Brown, H.E., Rebetzke, G.J., Zhao, Z., Zheng, B., Chapman, S.C., 2019. Improving process-based crop models to better capture genotype×environment×management interactions. J. Exp. Bot. 70, 2389–2401. https://doi.org/10.1093/jxb/erz092.

Wang, Y., Shi, L., Hu, X., Song, W., Wang, L., 2023. Multiphysics-Informed Neural Networks for Coupled Soil Hydrothermal Modeling. e2022WR031960 Water Resour. Res. 59. https://doi.org/10.1029/2022WR031960.

Willard, J., Jia, X., Xu, S., Steinbach, M., Kumar, V., 2022. Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems.

Yang, Q., Liu, L., Zhou, J., Ghosh, R., Peng, B., Guan, K., Tang, J., Zhou, W., Kumar, V., Jin, Z., 2023. A flexible and efficient knowledge-guided machine learning data assimilation (KGML-DA) framework for agroecosystem prediction in the US Midwest. Remote Sens. Environ. 299, 113880. https://doi.org/10.1016/j.rse.2023.113880.

Yin, X., 2003. A Flexible Sigmoid Function of Determinate Growth. Ann. Bot. 91, 361–371. https://doi.org/10.1093/aob/mcg029.

Yin, X., Struik, P.C., Goudriaan, J., 2021. On the needs for combining physiological principles and mathematics to improve crop models. Field Crops Res. 271, 108254. https://doi.org/10.1016/j.fcr.2021.108254.

Zhang, Q., Shi, L., Holzman, M., Ye, M., Wang, Y., Carmona, F., Zha, Y., 2019. A dynamic data-driven method for dealing with model structural error in soil moisture data assimilation. Adv. Water Resour. 132, 103407. https://doi.org/10.1016/j.advwatres.2019.103407.