in silico Plants, 2025, 7, diaf018
https://doi.org/10.1093/insilicoplants/diaf018
Advance access publication 22 September
Technical Advance

Technical Advance

Virtual world coupling with photosynthesis evaluation for synthetic data production

Dirk N. Baker[1,2,*], Mona Giraud[3], Jens Henrik Göbbert[4], Hanno Scharr[5], Morris Riedel[2,4], Ebba Þóra Hvannberg[2], Andrea Schnepf[3]

[1]Research Data Management, Helmholtz-Centre for Environmental Research - UFZ, Permoser Straße 15, Leipzig 04318, Germany
[2]School of Engineering and Natural Sciences, University of Iceland, Sæmundargata 2, Reykjavík 101, Iceland
[3]Institute of Bio- and Geosciences 3, Forschungszentrum Jülich GmbH, Jülich 52425, Germany
[4]Jülich Supercomputing Centre, Forschungszentrum Jülich GmbH, Jülich 52425, Germany
[5]Institute for Advanced Simulation 8, Forschungszentrum Jülich GmbH, Jülich 52425, Germany

*Corresponding author. E-mail: dnh2@hi.is

Handling Editor: Dr. Xin-Guang Zhu

Abstract. In this work, we couple the functional–structural plant model CPlantBox to the Unreal Engine by exploiting the implemented ray-tracing pipeline to evaluate light influx on the plant surface. There are many approaches for photosynthesis computation and light evaluation, though they typically are limited by versatility, compute speed, or operate on much coarser resolutions. This work specifically addresses the concern that data generation pipelines tend to be unresponsive and do not include model-based knowledge as part of the generation pipeline. Using established photosynthesis solvers, we model the interaction between the Unreal Engine and the FSPM to measure physical properties in the virtual world. This is successful if we are able to reproduce experimental results using an in silico model. As part of the pipeline, we generate a surface geometry and utilize material shaders that are designed to establish a baseline surface model for light interception and transmission, based on simple parameter sets that can be calibrated. Using a Selhausen field experiment as baseline, we reproduce the photosynthesis effectiveness of the plants in the 2016 winter wheat experiments. Our pipeline is deeply intertwined with data generation and has been proven to perform well at scale. In this work, we build on our previous work by showcasing both a simulation study of a light evaluation as well as quantifying how well our system performs on high-performance computing systems.

Keywords: functional–structural plant model; photosynthesis; model coupling; synthetic data generation; high-performance computing

1. INTRODUCTION

Simulating the real world is a pathway to deeper understanding of its mechanisms. This is especially true in plant science, where many individually distinct processes jointly form emergent behaviour of a plant (Thornley 2011, Walia et al. 2024). Here, studying the influences of the environment, or of plant properties, becomes crucial to fully explore crop behaviour as agriculture becomes more sustainable and resilient. Having access to a model of the real world, particularly in cases for which reference experimental data are available but sparse, can assist with filling gaps in the data. A use-case for this is to map the experimental results onto synthetic data for Deep Learning (DL) models. Particularly in plant sciences, where image analysis is a critical tool and the primary bottleneck (Taiz 2013, Minervini et al. 2015, Tsaftaris et al. 2016), using model-based data sources can uncover hidden mechanisms. A faithful version of an experiment would include a calibrated structure of a plant, producing both images and dense training labels. The potential to uncover systematic image features that plants with a certain trait exhibit, is the first and most important step towards training models that detect such traits remotely. Embedding models into the data generation pipeline is especially impactful as a common critique of data production pipelines is their unresponsiveness and one-directional character, e.g. as reported by Li et al. (2024). In this work, we showcase a method to embed and communicate with the virtual world in a way that does not only produce more useful data labels for a concrete problem such as photosynthesis but also explore the expressiveness of our virtual world model to accommodate these models.

The category of models that describe both structural aspects, i.e. leaf shape, and the function, i.e. photosynthesis, of a plant in a coupled simulation is called Functional–Structural Plant Models (FSPMs). FSPMs, such as CPlantBox (Giraud et al. 2023) seldomly represent only one functional process and are instead a coupling of different models (Sievänen et al. 2014). The explicitly defined 3D structure is a vessel to define and

communicate functional properties, such as water uptake (Mai et al. 2019), and is the baseline structure that allows the simulation of these processes. In all FSPMs, the structure of the plant affects one or several functions (e.g. plant total assimilation rate). FSPMs can be used to gain insights into plant-scale mechanisms (e.g. link between plant light competition and plant growth) or to yield input parameters for larger-scale models in the form of a heuristic such as average light influx per plant. One aspect of plant models, and other model domains such as soil (Zha et al. 2019), is that they often operate on discretizations of the real world, particularly regarding the organ structure of the plant. The structural component of a plant simulation often encodes the plant as a graph (G), which is a collection of vertices V (points in space) and edges E (connections between points). A baseline algorithm to generate a spatial structure is a Lindenmayer system (L-system) (Chuai-Aree et al. 2005), which is a rule-based generation method for graph structures.

Photosynthesis is the process by which a plant converts electromagnetic radiation (sunlight) into chemical energy by using carbon dioxide (co_2) and water (h_2o) as educts. An important aspect of the photosynthetic process is the role of the stomata on the leaf surface, which are pores that regulate the gas exchange between the leaf and the atmosphere. Photosynthetic simulations with stomatal regulation generally solve a fixed-point iteration (Von Caemerer 2013) and the reactions involved in the uptake of co_2 are solved on the basis of a steady reaction flow limited by availability of resources (Giraud et al. 2023) or stomatal metabolism (Yin et al. 2004). The CPlantBox module for photosynthesis, developed by Giraud et al. (2023), combines different models (Farquhar et al. 1980, Leuning 1995, Tuzet et al. 2003) into a whole-plant framework that couples the water and carbon dynamics from root to leaf. Although the model can use specific input data for each leaf section, Giraud et al. (2023) only used mean atmospheric data.

Models that calculate the actual light influx in a plant involve Radiative Transfer Modelling (RTM), which describes the light propagation and eventual impact onto the surface of the plant. RTM has standardized concepts (Nicodemus et al. 1977, Martonchik et al. 2000) and is implemented in a variety of models, particularly DART (Gastellu-Etchegorry et al. 2004). For the retrieval of functional properties from remote sensing data, empirical models such as LESS developed by Qi et al. (2019) and extended e.g. by Zhao et al. (2024), have been successfully implemented. Another example of a high-quality model for radiative transfer available as open-source is `prospect`, developed by Féret and Boissieu (2024). Worth noting is that similar concepts and models have been implemented when a rendering technique called *physics-based rendering* was established, and the similarity between the two fields regarding this specific use-case is of particular note, as described by Salesin et al. (2024).

In experiments to parameterize FSPMs, particularly the measurement of the structural portion can yield much information on the performance of the crop. Diversity of structure, e.g. in root systems, has been identified to be one of the primary indicators of adaptation of a crop to its environment by Yu et al. (2024). As such, many data generation pipelines make use of a plant's structure to accommodate variability (Ward et al. 2018, Bailey 2019, Baker et al. 2023). The key factor for the embedding of an FSPM for data production is a reasonable reproduction of the target space, which is RGB (red, blue, and green channel) images for many applications. In our pipeline, we are using the Unreal Engine (UE) to compute the virtual world. UE is a multimedia 3D graphics engine, which has seen use across industries, including its origin in gaming, simulation science (Agarwal et al. 2023), plant science (Li et al. 2024), and virtual reality (Krüger et al. 2024). Our own recent work brought forth the Synavis framework, a library designed to dynamically setup, steer, and measure within the UE virtual world (Baker et al. 2023, 2024).

An embedding recently implemented in the HELIOS pipeline by Lei et al. (2024) shows the potential for pipelines to accommodate functional information. We believe that the inclusion of an FSPM that can simulate functions in the whole plant (roots and shoot) and at a sub-organ level provides a valuable extension to this by allowing the models to communicate between domains (soil–plant–atmosphere) and scales (plant section–field). Thus, in this work, we extend the Synavis framework to work together with the photosynthesis module of CPlantBox, which until recently used light influx averages. Our implementation of this pipeline uses the physics-based rendering employed in UE to simulate light exposure to achieve an accurate light exposure computation for CPlantBox, which we validate by modelling a digital twin of an experiment at the Field Minirhizotron Facility in Selhausen. The data-based pipeline that is enabled in Synavis is extended by using a model-view which takes physical properties to allow radiative transfer modelling and a data view that is able to map functional photosynthesis data onto a synthetic data pipeline, which we showcase by the example of light exposure efficiency. In our replication, we make use of experimental data provided by the TERENO platform (Bogena 2016, Reichenau et al. 2020) to recreate a field experiment while also retaining a high level of structural and functional information through the use of a full-scale plant model. Lastly, our pipeline is compatible with High-Performance Computing (HPC) systems, with a focus on ensuring that parallelization is feasible, and we include full scaling experiments with this manuscript.

2. METHODS

Our work (described as video in Baker (2025)) consists of three main components that are connected using the Synavis framework (Baker et al. 2023). First, we run the FSPM CPlantBox (Giraud et al. 2023) to simulate the plant structure and its functional properties on a field scale. We embed the structure of this plant into a virtual world to measure light influx in Unreal Engine (UE). (Unreal Engine, Epic Games, Inc., Cary, NC, USA, https://www.unrealengine.com/en-US/) using the Synavis framework. Lastly, we use the measured light influx to compute the photosynthesis of the plant, which is then used to produce synthetic data. This section first describes the virtual world generation in UE, then the embedding of the FSPM into the virtual world, and finally how we replicate experiments and the domain partitioning required to scale to field experiments.

2.1 Virtual world generation in synavis

We used the Synavis framework to create a virtual world in UE. An in-depth description of the Synavis framework can be found

in our recent paper (Baker et al. 2023). Synavis is a framework for the coupling of simulations with virtual environments, allowing the sampling and measuring of values through the framework for the purposes of training neural networks. In this work, we extended this functionality by allowing the FSPM simulation to access data from the virtual environment directly, allowing the superimposition of plant function onto the virtual environment and the subsequent measurement physical properties in it. Within UE, we used the base rendering framework as well as the entity-component system implemented in UE. We developed a framework to measure light intensity, a pipeline to embed model data into UE to generate synthetic data based on FSPM simulation results, as well as a pipeline to setup and manage distributed virtual scene handling through Synavis. Since the implementation in this work is inherently bi-directional, we refer to the placement of the FSPM into the virtual world as *embedding*.

The coupling of the FSPM with the virtual environment, both in terms of the simulation and the data production, is shown in Fig. 1. Our setup has three distinct aspects: the CPlantBox simulation (Fig. 1A), the UE model scene (Fig. 1B), and the data production (Fig. 1C). The two virtual scenes need to be separated, as the light and surface properties in these are fundamentally different, yielding two 'views' on the plant structure, one as a model scene and one as a data scene. These aspects of the setup can be separated and communicated asynchronously, but in our setups we typically assume that each application is running concurrently (Baker et al. 2024). The field setup configures and runs the CPlantBox simulations. The FSPM simulation has photosynthesis and growth model components, along with a visualizer that produces a geometry, linking the simulation structure with the virtual world geometry. This information is synchronous between the simulation and the virtual world, meaning that the point in time is the same for both applications. An additional set of parameters, the environmental measures (Fig. 1B), is taken directly from measurements, such as the weather data (Reichenau et al. 2020) from the TERENO platform (Bogena 2016). The environmental measures, the light meters (Fig. 1B), and the runtime geometry are time dependent, and this time-variant coupling is visualized by interlocking rings. The model scene informs the photosynthesis module in the form of edge-based light influx measurements. The data generation (Fig. 1C) is not coupled to the model scene, for the simple reason that the light influx measurement is not useful on its own, without biological interpretation through the simulation model. As such, the visualizer also produces plant geometries for the data scene, which are the same base geometries as for the model scene, but the textures are different. The annotation also directly accesses data from CPlantBox, as described below in Section 3.1.

The visualizer builds semantic correspondence that can be used to map functional properties. Through Synavis, plant geometries are inserted into the virtual scene once every time step, keeping it synchronous as indicated in Fig. 1. A texture containing surface properties is being generated per organ because of how the texture wraps around the plant, see Fig. 2 and Section 2.2. Light influx measurement is done after the radiation calibration, submitted as point measurement command to the `ALightMeter` class in UE, which carries out the measurement. Necessary parameters are meter resolution, measurement delay to accommodate scene update and measurement duration to accommodate path tracing. The size of the measurement area is furthermore defined within the light meter. The light meter has a USceneCaptureComponent which is aimed at the measurement spot in the minimum distance required to avoid near-plane culling. As indicated by the coupling symbol in Fig. 1, the light meter is specifically requested for sections of the plant when the photosynthesis module requires it. Path tracing is a UE term that refers to the physics-based raytracing of image pixels without approximated sampling. UE additionally has an accelerated approximation for the raytracing called Lumen, which uses a reduced-detail scene to complete the light information and might not correctly reflect the physical properties of the surfaces, as reported by Agarwal et al. (2023) in a remote sensing context. The measurement is passed back through Synavis as influx per segment, which is passed to the Photosynthesis module. The parametrization is entirely on the Python side, which retains all functional parameters and values. Mapping of the photosynthesis output is done exactly like the mapping of functional properties, but this is a one-way coupling, where no information is fed back and the data generation scene is only used to produce synthetic data.

2.2 CPlantBox model embedding

CPlantBox produces the structure of a plant, using a graph $(G_1 := (V, E, P))$. G_1 is made of vertices (V) connected by edges (E) and stores a set of properties (P) defined on either the edges or the vertices. In this work, we use G_1 to refer to the 1D *model structure*, meaning the discretization produced by the plant model, while the term G_3 refers to the 3D *geometry* that is produced by the geometrization of the model into a plant geometry. For the purpose of reproducibility, we will specifically highlight how these structures are linked, particularly regarding the fact that in this work, we are transmitting information back and forth. This structure can be seen outlined in Fig. 2A. Our embedding typically uses an inferred 3D structure, called G_3, which contains shape information informed by G_1. G_3 is a meshing based on shape representations (cylinder or quad) that is implicitly generated from vertex positions and organ type, as shown in Fig. 2B. We make certain assumptions to define the output geometry, e.g. the leaf blades are perpendicular to the branching direction, and remain perpendicular to successive vertex differences. This can be calibrated empirically through the CPlantBox visualizer (Baker et al. 2023). This implementation ensures that the property mapping always directly corresponds to the edge in G_1, as the centreline is used as the coordinate for the leaf. We interpolate properties to vertex level and implicitly further by encoding properties into textures, seen in Fig. 2C, which can be accessed through tangent information like Fig. 2D.

Game engine rendering has evolved over time, particularly regarding considerations of realism and physical accuracy, which can impact user experience. An important step, particularly in UE, is the implementation of physics-based rendering (Schraml 2019), which implements light and surface properties for a physically plausible rendering path. UE uses a bi-directional radiative transfer function (BRDF), which yields the outgoing light intensity for the angle according to the incident angle. Additional effects are possible, such as distance to the image plane, but the surface properties are mostly related

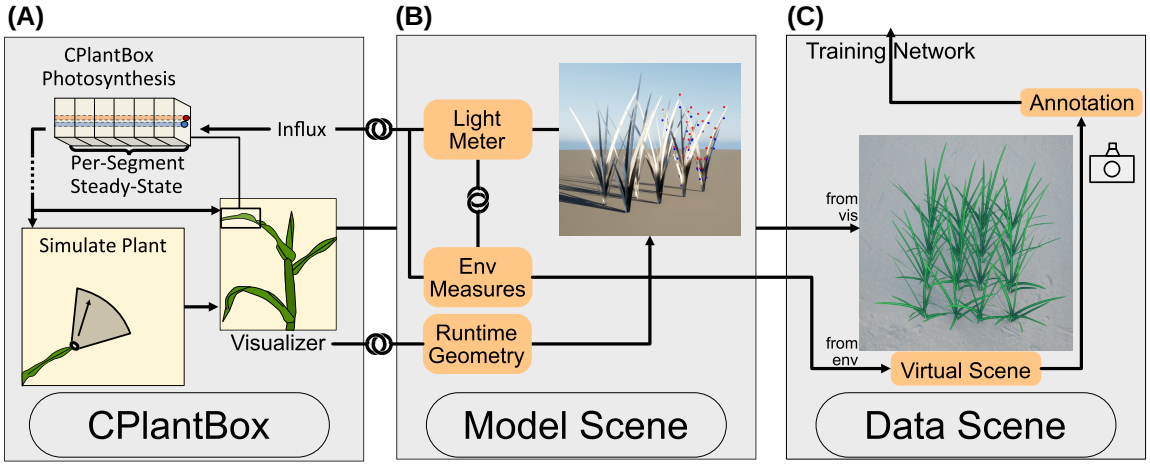

Figure 1. Data flow of the coupline, showing information flow with lines and time-variant synchonization with links A) CPlantBox simulation, B) UE virtual scene for photosynthetic measurement, C) UE virtual scene for synthetic data output.

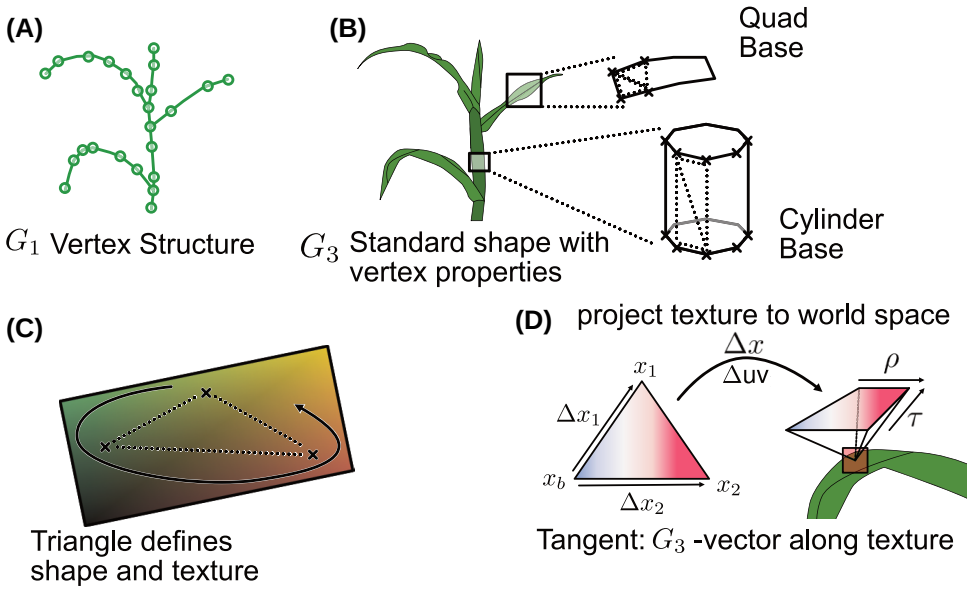

Figure 2. Relationship between functional properties on the texture level, according to the mapping employed by Synavis (Baker et al. 2023). A) We denote G_1 to be the graph structure produced by the FSPM, including diameters, organ shape parameters, as well as functional aspects. B) Geometrization is done through shape extrusion, taking uniform shapes as template and placing them in a way that they encase the plant. Triangulization is the connection between the shapes that are used to build the geometry. C) On top of a triangle, the texture map is further defined on each vertex. D) To be able to fully map the texture onto the geometry, the tangents need to be computed, which indicate for the texture is wrapped locally around the geometry, being as long as one pixel is wide on that vertex.

to how the measurement enters and exists the surface. The function is defined either through geometric or texture properties encoded in the Material shader that dictates how the surface reacts to light. A surface can also emit light, and the final colour at the point where the pixel ray intersects the surface will include both direct emission, direct reflection, and indirect reflection. The reflectance of a surface is represented semi-empirically via a single factor, representing implicitly the interaction of the light with the different layers of an object. As we are measuring the brightness, we are reversing certain aspects of the light simulation—e.g. a surface that would have absorbed a band of the light spectrum would now be brighter instead. This was implemented as such mostly for convenience, allowing the values to be representing of the actual absorption rather than its inverse. The BRDF is sampled across a hemispherical space on top of the initial point where the measurement ray intersects the surface.

The resulting brightness of the surface is also scaled by the surface normal. The surface normal is interpolated from its vertex definition much like other properties (cf. Fig. 2C). Indirect (diffuse) light is sampled by the path tracer through repeated measurements of the same pixel. This measurement needs to be performed in all directions, which means that a faithful measurement requires extensive sampling of the scene. While there are certain optimizations implemented in Lumen, there are no intrinsic tricks that can be utilized to speed up the sampling, as

it is entirely dependent on the geometric content of the scene. Texture properties rely on a function that assigns each vertex a 2D coordinate corresponding to the texture. This is illustrated on triangles of G_3 in Fig. 2C. Figure 2C also illustrates that the texture and the vertices might not be aligned, as there are additional coordinates that are being used to correspond the vertices to the texture. For most purposes, it yields less texture tearing or stretching if the vertices and the texture is as well aligned as possible. Since the CPlantBox visualizer described in Baker et al. (2023) uses a texture map for each organ separately, the information is much more fine-tuned to the organ level, yielding a more expedient mapping. The correspondence between vertex and texture is still important to consider for the replication, and the full description on what value would be retrieved at a specific point is given in Appendix A. As described in Baker et al. (2023), the texture mapping of the FSPM structure fills the space of the texture map, thus resulting in a stretching of the properties contributing to the transfer function. Since the texture mapping is done on vertex level, but the rendering will densely prompt for surface properties at every location, the evaluation of a specific point relies on the texture map. Figure 2D shows how the texture (and thus functional) information is wrapped around the geometry. How the texture is locally warped is determined by the tangents, which are computed from the vertex positions and the texture coordinates. This estimation can be done before rendering (thus saving the tangents at vertex level) and allows for seamless stitching of a texture covering unconnected surfaces by explicit tangent declaration. The computation of a value from the functional properties at a specific surface location is described in Fig. 8. The primary surface properties (https://dev.epicgames.com/documentation/en-us/unreal-engine/physically-based-materials-in-unreal-engine, Accessed 17 September 2025) that are implemented in UE that are of interest are diffuse, which is set to grey for the purpose of measurement, specular, roughness, and auxiliary properties such as subsurface, translucency, or masking. Our implementation aims to mimick the surface properties relating to photosynthetic active radiation (PAR), a subspectrum of sunlight. However, this is not a full-spectrum model, and we rely on experimental measurements to calibrate the surface properties, along with experimental verification of the photosynthesis model. We break down the most relevant contributions to photosynthesis and light absorption to material properties. The diffuse property (set to $(0.5, 0.5, 0.5)^{\mathsf{T}}$) is a measurement tool that does not contain any physical relevance. However, although we are using a single channel, one could separate three channels for the transfer model. The bulk of the empirical information will be envoded in the specular and roughness properties as well as the subsurface scattering. In this work, we are not using the emissive property, as there is no data basis for the calibration of the emissive light. However, its contribution is discussed in related work (Miao et al. 2018), and as such, we include a characterization of our framework's usability for emissive light in Appendix B.

2.3 Domain partition and HPC

Since we are aiming to simulate field-scale setups, the actual workload of simulating and measuring in the virtual environment needs to be distributed. For this purpose, we utilize field partitioning by allocating instances of Unreal Engine dynamically through Synavis. This use-case has also been implemented for field reconstruction in Baker et al. (2024), and we adapt it to accommodate the use-case of light evaluation. Field distribution is based on direct partitioning. Because CPlantBox uses stochastic structures that are random-number-based, we assign a specific starting seed to the field in total, varying the individual plants deterministically. This ensures that the specific realization of a plant organ does not need to be communicated between nodes, as all nodes have implicit knowledge about the neighbouring structures and could access their specific realization by simulating the corresponding field seed plus plant position. All simulations that are being run have unique plant structures, but these structures can be inferred from the starting time of the cluster job.

We implement our field distribution for the JURECA-DC (Thörnig 2021) cluster, equipped with four NVIDIA A100 GPUs in addition to two AMD EPYC 7742 CPUs. These nodes are connected with InfiniBand Interfaces (NVIDIA Mellanox Connect-X6). Within each node, the field consists of a realization of a grid of plants, growing in a separate CPlantBox configuration. It is entirely optional, and independent of the virtual scene, whether the boundaries of the individual FSPM instances are being coupled, or not. In fact, as we are measuring the light influx on the geometry in the scene, the competition and individual exposure of individual plants can be estimated without running the FSPM in a continuous system. Nevertheless, for a full view of the plant field, a continuous system of plants, particularly regarding their soil properties, would be necessary.

The field-scale embedding further needs to be distributed over nodes to ensure efficient computation of plant instances. This is the case for all field-scale setups, irrespective of the soil-related simulation setup, since the amount of individual segments that need to be measured exceeds the memory of common GPUs (Baker et al. 2024). Our scaling experiment, results of which are shown in Section 3.3, consists of a partitioned field using a total number of plants N^2 arranged in a square. Based on total amount and MPI world size W, this square is subdivided into sets of squares of N/W plants per side. The seed numbers act as IDs for the plants, with the added randomness introduced by the starting time. The simulation can use one coherent global starting seed that is concatenated with the local ID of the plants. The grid of plants is scaled with the world embedding scale (which can be calibrated if needed for numerical precision) and the plant density. However, custom placements of plants are generally encouraged when producing light simulations, as our partitioning setup uses simple idioms to distribute the simulation as a showcase for HPC partitioning.

Our partitioning and scaling experiments are based on a plant grid that is being set and defined remotely. There are many hyperparameters that contribute to performance and results. In our tests, the total virtual screen resolution, i.e. number of light meters, is a factor in overall performance. For the hard scaling, we tested the progression using three steps of meter count: $50 \mapsto 3200$Pixels, $100 \mapsto 6400$Pixels, and $200 \mapsto 12\,800$Pixels. Virtual resolution values outside of these yield either decreased render or measurement performance. We increase the number of nodes, resulting in an increased field size or an increased distribution of the field size. In terms of UE rendering configuration,

we do not utilize culling and will keep the reference point for render optimization (Occlusion Culling Description) close to the target for measurement. When path tracing (Path Tracing Description) is employed, lightmap settings do not matter, and we are not utilizing virtual textures (Virtual Texturing Description), though it would not impact the measurement if this was added.

2.4 Embedding of experimental data

Our reference data have been taken from the Field Minirhizotron Facility in Selhausen (50.865°N, 6.447°E, 203m a.s.l. in 2024) (Lärm et al. 2023, Nguyen et al. 2024). The facility is an experimental site that includes field data as well as extensive sensory data, including a minirhizotron setup that includes tubing to measure root distribution (Bauer et al. 2022). These data sets are hosted on the TERENO platform (Bogena 2016), which is an open data platform that enables us to access sensory as well as experimental data across parts of central Europe.

Weather has three functions in this work. It defines the atmospheric conditions of the photosynthetic reaction, as each time step of the simulation draws on the corresponding weather data. Furthermore, it provides the baseline sun radiation for the light measurement, where the radiation measurements from a nearby measurement station are used to calibrate the UE light sources. Lastly, it provides the visual calibration for data production. From top to bottom, air pressure and humidity, temperature, and radiation are being derived from Selhausen measurements. These values are partially resolved at the 10-minute rate. Using the calibration in Fig. 8, we can linearly scale the light measurement using a clearsky measurement value as reference. This value can stem from singular measurements done as reference, or a densely resolved radiation flux measurement as present in the Selhausen above-ground data. From a data generation standpoint, PAR radiation can be directly mapped onto the pixel space by using the relative radiance at each point without needing to consult measurement data. An illustration of this approach is shown in Fig. 1. We vary the directional light brightness based on the `RadiationGlobal` $[W/m^2]$ data collected in the Selhausen facility, as illustrated in Fig. 1. Our site data was partially extracted from the TERENO data hub (Bogena 2016) for the climate station Selhausen 3 (Schmidt 2024). The soil characterization was done in Lärm et al. (2023), which features matric potential measurements at different depths. The soil parameters were implemented as 1D model for simplicity, while the root structure was simulated explicitly. We chose not to infer air co_2 molar fraction from the experimental measurements as the sensors we would have available are not representative of the chamber measurements in Nguyen et al. (2024). Nevertheless, the full weather model can be referenced to in the simulation script, see Section 5.4.

For the 2016 measurement, winter wheat was sown on 26th October 2015, while first emergence was logged at 1st November 2015. Nguyen et al. (2024) describe the gas chamber measurement in the field regarding the photosynthetic activity of the plants. The measurements were done at multiple time of days, and the PAR was always measured concurrently, along with other indicators. Our comparison focuses on the PAR adsorption and the co_2 uptake. We assumed the start of the growing season to be 5th March 2016 and simulate the plant growth up to the measurement, at which time we run the photosynthesis module in combination with the UE measurement of light influx. We are using a pre-flowering gas chamber measurement for the photosynthesis activity, including a sunlit wheat plant. We measure the light photon influx per segment on the leaf surface. This value has been relayed to CPlantBox, following a photosynthesis evaluation with simplistic soil coupling. We assumed no limitation by soil water content. The leaf surface parameters are kept simple, not including subsurface scattering or fluorescence. These values are included in the model, but need to be parameterized, and our TERENO-based parametrization is illustrated in Fig. 1. Both the virtual world models are steered through Synavis and implemented in UE, while CPlantBox runs as a precursor to the data scene, needing the model scene as light influx reference. In UE, the measurement was done over three frames following a two frame waiting period after the measurement point has been registered. This is to ensure that the virtual scene is fully updated before the measurement begins.

3. RESULTS

Model coupling is largely based on the combination of the virtual world embedding (Baker et al. 2023) and the photosynthesis calculation developed by Giraud et al. (2023). Here, we highlight our results regarding the primary evaluation of our coupling's main contributions. This section focuses on the accuracy of the framework, the scalability of its implementation, as well as the production of synthetic light exposure training data from FSPM simulations.

3.1 Synthetic light exposure data

Mapping the functional data onto a virtual measurement space ensures the increased usefulness of the photosynthesis data. It allows the extension of the synthetic data pipeline to uncover mechanistic insights into plant development in a pipeline otherwise restricted to other information that is being rendered in the engine, such as implemented by Mousavi et al. (2020).

Our UE implementation measures exactly around the submitted points, an array which can be mapped onto the geometry. As the rendering needs to switch modes for this task, the workload needs to be separated from the measurement or performed subsequently. The photosynthesis module of CPlantBox, if not restricted through other mechanisms, imposes a limitation on carbon uptake A_g, which is $A_g = \min(F_j, F_c)$ (Giraud et al. 2023), where F_j is the electron transport rate and F_c is the carboxylation rate, which is largely restricted by Nitrogen uptake. These rates are output by the simulation, particularly the photosynthesis module, when solving for the photosynthesis reactions. They are given by segment, which we resampled to organ level for the visualization in Fig. 3. Different populations, one large and one rather small are shown in Fig. 3, showcasing how this kind of scaling would function in the Synavis pipeline. An output of this simulation is the fact that we can scale such an otherwise very mechanical

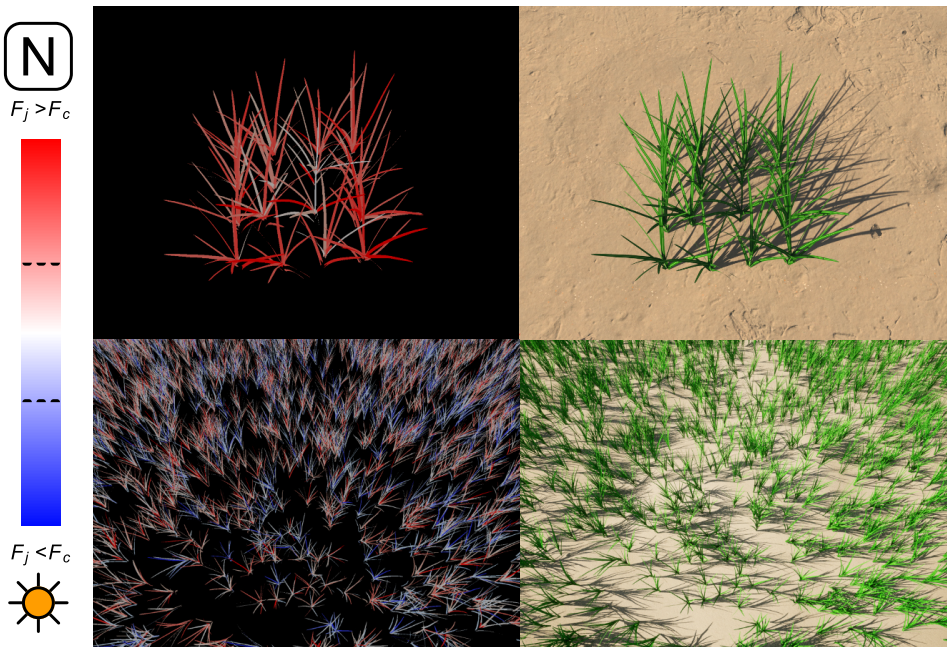

Figure 3. Mapping of functional properties, in this case light assimilation effectiveness, as single value per leaf onto the FSPM visualizer surface. Value scale is divergent to distinguish the cases, with the red/blue distinction describing the flipping point between nitrogen limitation $F_j > F_c$ and photoelectric limitation $F_j < F_c$.

value to a more clearer result, showing a potential limitation by F_c which could indicate the need for more nitrogen fertilization.

Our implementation for this work furthermore allows a full export of the FSPM geometry for multiple file formats, including VTK-compatible files, or Wavefront Object files that can be used with software such as DART (Gastellu-Etchegorry et al. 2004). The higher compatibility also allows users to employ more coupling techniques that can be used with other RT models, including lightweight models that run on CPU-only devices. The coupling is furthermore entirely steerable within Python, which also contains the simulation information provided by CPlantBox.

3.2 Light influx accuracy

We compare the experimentally measured values for light influx and carbon uptake to our virtual world model for the 20th April 2016. This date was chosen primarily due to the growth stage of the plant being well reproducible and the weather being relatively stable. As baseline for the expressiveness of the UE light model, this is sufficient, as the primary goal is to show that extrapolation from weather data to light influx on the plant is possible using our method. Our comparison is shown in Fig. 4, which highlights both light influx as well as co_2 uptake per area. Our calibration yielded a close match to the experimental data, though we will note that multiple parameters could yield similar results. The calibration of the individual parameters used for the light evaluation can be referenced to in the resources that accompany this manuscript, see Section C. The plant light influx variation that is shown in Fig. 4 is segment based, meaning that the light influx is measured at different positions on all leafs present in the scene.

For our simulation study, we sub-sampled the 10-minute intervals to 30-minute intervals to reduce simulation time to allow for a better progression of the FSPM simulation. This is mostly a reduction of the computational workload to replicate the data, while still maintaining similar expressiveness. The base experimental time resolution of the experiment is non-uniform, with the measurements taking place in specific time slots with multiple measurements. We averaged measurements that were taken in the same interval, though on different plants labelled the same, resulting in the variance seen in Fig. 4. Similarly, we highlight that while the measurements that were deemed *sunlit* in the experiments described in Nguyen et al. (2024). The light absorption measured in both the experiment and in our simulation study yielded high variance. This is likely caused by differences in measurement positions on the plants. The increase in the mornings was steeper than the decrease in the evening. This is also present in the raw data we used for the modelling (Schmidt 2024). Similarly, the co_2 uptake increased in such a fashion, though in the evening both radiation and uptake values are outside the experimental range. Our simulated co_2 uptake at around 11 ó clock was lower than the average experimental value, though within the variance. Our light influx variance was peculiarly low in the range of 7 to 8 ó clock, which we could attribute to the scattering values that were used in the virtual environment, in combination with the fact that overall illumination was low.

3.3 Scaling and performance

We have investigated the scalability of the coupling to determine the performance of the method on HPC systems. The simulation setup uses a field of 500 (Placement uses nearest larger N for non-quadratic input sizes.) plants, which in our case

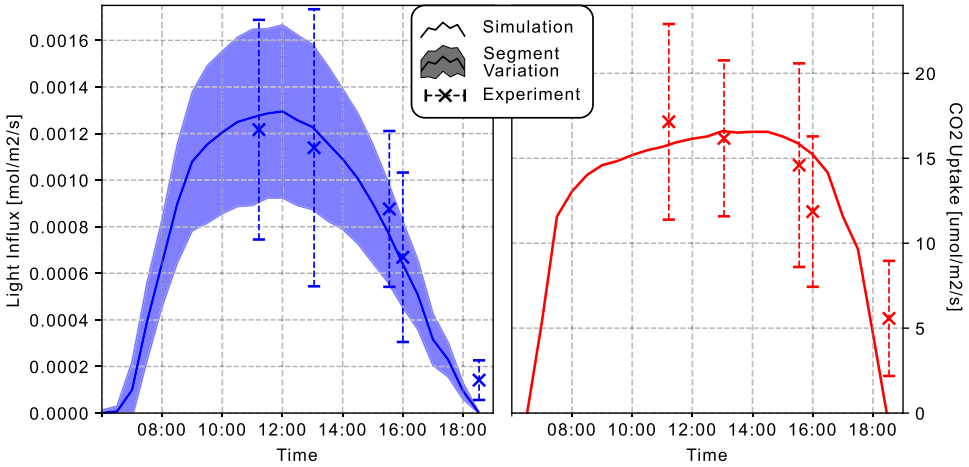

Figure 4. Absorbed PAR in blue/left, and net co_2 uptake in red/right. The light flux standard deviation is segment based, i.e. depending on where on the plant the light is measured, the light measurement has different results, resulting in a high standard variation. The experimental data are marked with an x and a dotted line for measured variance.

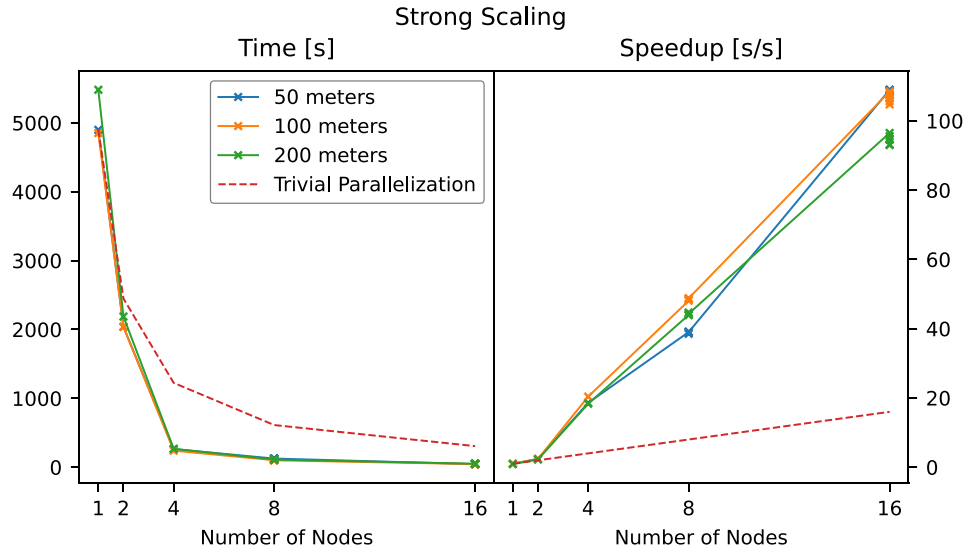

Figure 5. Scaling of number of nodes while not scaling total problem size. A) Time of execution for the photosynthesis pipeline for a field of 500 plants, measuring the performance of the light evaluation only, from submission to reception of the exposure data. B) Derived speedup values for the parallel performance. We compare our results against the 'trivial parallelization' baseline, representing a speedup directly proportional to the number of nodes used. Our memory-bound applications has a superscaling behaviour, resulting in an overall faster computation by means of parallelization and reduction of wait cycles for the computation.

was the maximum number of plants that could be simulated in a single node. We show both timings and resulting speedup in Fig. 5. In it, we observe what appears to be super-scaling behaviour, which requires closer investigation. First, the light evaluation being distributed across nodes will make the measurement faster than on a single node, with N/P measurement points as opposed to N. We note that there *is* overhead in the shape of the surrounding plants that get rendered as to not distort the boundaries between the compartments of the field, even though these plants act as geometric inclusion only. However, because the overall amount of triangles in the virtual scene is significantly lower, the engine also performs better *per render pass*, which results in an increased performance of a single evaluation. Thus, the parallel performance appears to be so steeply increasing with the number of nodes. This is commonly referred to as memory-bound performance (see e.g. Allande et al. 2014).

Figure 6 shows the weak scaling of our system. The speedup appears to be slightly increasing, though this is constraint by the controlling node that handles the same area. From measurement, the added speedup even though the problem size remains constant might have different reasons, as investigated in Section 4.1. The measurement is taken through the framework itself, and as such it does not account for effects due to MPI setup and the initialization phase of the photosynthesis simulation. In our evaluation, we found that there was not much difference in the performance regarding the total (virtual) resolution of the light measurement, based on the top three performing

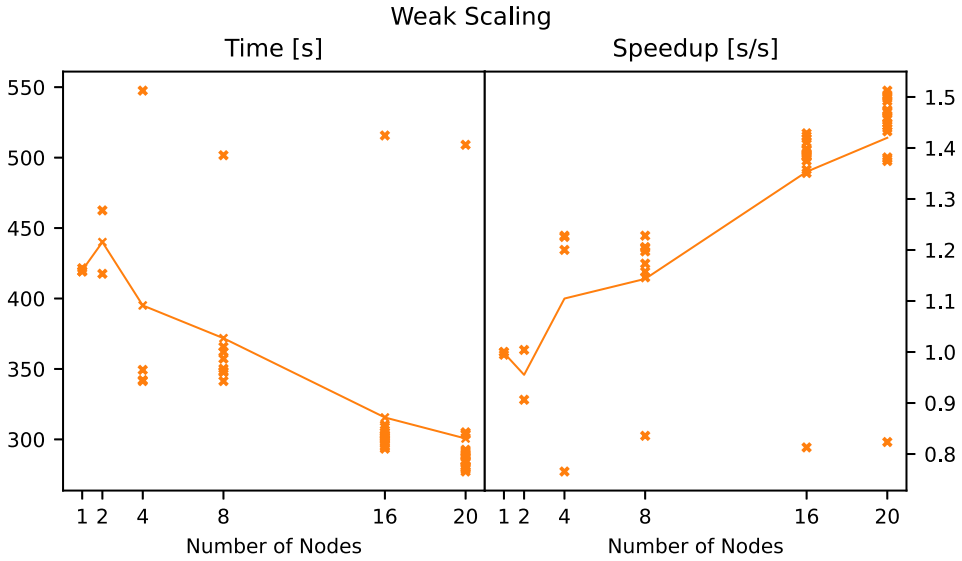

Figure 6. Scaling of both problem size and number of nodes. A) Time of execution if we increase the data in the same way as the number of nodes. B) Derived speedup values for the parallel performance, again calculated as $T(1)/T(N)$.

settings in this context. An increased resolution, i.e. an increased number of light meters, will result in a decreased frame performance, resulting in a relatively similar performance overall. From the hard scaling comparison in Fig. 5, we found that 100 light meters yields best performance, resulting in a total resolution of $8^2 \cdot 100 = 6400$ measurement points.

4. DISCUSSION

This work has been an investigation into the validity of using the Synavis framework for simulated embeddings, lifting the data production pipeline to a true embedding of simulated data. The contribution on a technical level is the coupling to the FSPM as well as the implementation of a functional embedding into UE such that it can be used together with other models beyond plant models.

4.1 Distributed simulation of FSPM embeddings

It is generally no issue to distribute plant simulations across nodes in cases where the simulation is entirely at organ scale. Scale conversions, such as between cells and plant level, always require a form of heuristics, as it would be cumbersome to simulate a plant on cell level entirely. As such, the data scale conversion is an important tool to not only use varying resolutions of experimental data but also to distribute the plant simulation. We use this to estimate the size of the buffer region in our tests. Typically, with our parameters regarding the size and density of the field, two rows of plants around our measurement area is sufficient on average. During midday, and depending on geographic location, the buffer plants might even be omitted, which is also the case for cloudy days. In the first case, this is because the shadows are very close to the plant and might not significantly touch neighbouring plants, and in the second case, most light arrives in so many scattered pathways that shadows have decreased contrast.

The scaling experiments in Figs. 5 and 6 show a clear speedup through HPC use. We note, however, that we do not believe the apparent trend in Fig. 6 regarding the speedup between 8 and 20 GPU nodes has a systematic cause. There are components regarding the order of operations, particularly the spawning of plants and the initialization of the measurement, that cause a greater variance in the sampling. Particularly we note that the main compute node did not speedup, but rather remained somewhat constant regarding its sampling and the collection of light influxes. Uncovering the hyperparameters that contribute to the eventual parallel efficiency is beyond the scope of this work, but would be the next step to complete the HPC integration of the Synavis framework, particularly in combination with AI workloads. To accommodate DNN pipelines and to reduce over-fitting, the number of visualizers should be lower than the number of learners, as the learning algorithms needs to use image-based augmentations in addition to domain-based augmentations to fully make use of the synthetic data, as image-based augmentations have uses beyond adding more pictures to the data space.

Distributed performance using the Unreal Engine also would have some limitations regarding inter-module connectivity, as scaling up the virtual world rendering on a dedicated visualization module might impact the latency. Users of the framework would need to be cognizant of the local properties of the HPC systems they operate on, as it could introduce topology or module-depended differences in communication times. Our communication tests typically yielded latencies between modules (through Ethernet, e.g. between JURECA-DC and other clusters on-site) of around 200 ms. We do highlight the usefulness enabling graphics on user-exclusive GPU nodes as opposed to offloading the capability of rendering onto interactive nodes that cannot tolerate high workloads.

4.2 In-silico recreation of a field experiment

The gas chamber measurements can be referred to in Nguyen et al. (2024), and we will note that it is not an isolated chamber, and thus the concentrations are susceptible to wind. We are comparing distribution-free measurements, as the reported

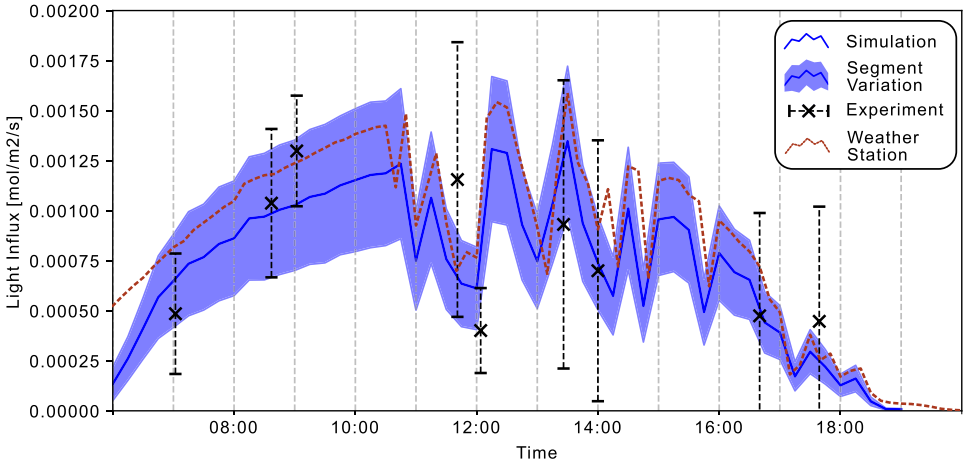

Figure 7. Post hoc analysis of accuracy of the light reproduction in the presence of averse effects such as clouds and low sun angles. Experimental conditions (vertical) vary significantly from the measurements of the same day in the field campaign. Our simulation (shaded) is able to reproduce the weather data (dotted) quite well, being calibrated particularly for it, yet the combination of this with the plant model yields a lower measured light influx on the leaf surface than was measured in the field.

photosynthesis measurements are provided per area and time. Our pipeline is able to replicate the experiments of this specific configuration well. While the pipeline can match observed measurements, we caution that multiple parameter combinations, some potentially biophysically unrealistic, can yield similar outputs. This highlights the need for physiological validation when applying such synthetic systems to model fitting or remote sensing. The overfitting of simulation models has been previously reported in other domains, such as radiotherapy in silico simulation (van der Schaaf et al. 2012). The calibration of the light influx is not only dependent on the actual sunlight intensity, but also on the atmospheric scattering of specific wavelengths, as well as leaf surface scattering. To truly provide an embedding for more different FSPM simulations, a full study, including experimental measurements of leaf surfaces, would be required, along with a study on the mapping of wavelengths into the virtual world. We have fitted and highlighted that we can represent experimental data, but the full parametrization and analysis of the virtual world is beyond the scope of this article, particularly regarding its use as data generation pipeline. The soil parametrization regarding light propagation is an important factor that contributes to total illumination.

The factors that contribute to indirect illumination are partially omitted from our simulation, as there is need for a dedicated experiment to evaluate these attributes. However, due to the comparatively low accuracy in the evening, based on our above replication study, we specifically investigated the occurrence of averse effects on the light influx measurement on the leaf surface. We picked a later date, the 23rd June 2016, which included more diverse conditions throughout the day, as shown in Fig. 7. We compared the light influx on the leaf surface to the measured radiation in the field. We additionally included the weather station data that we obtained to parametrize the virtual world. A full list of the station data can be seen in Reichenau et al. (2020). Our simulation is on average lower than the measured radiation, which is intended and to be expected from ground-level measurements. Our dynamic does not quite match the measured radiation in the morning, but as here the parametrization of the evening effects is fitting rather well, this is most likely a ground-level effect as well. The 'calibration meter' which is included in the scene is usually measuring about the same radiation as the weather station, but of course, there is no full match, as a single point in the field cannot represent the entire field. The experimental data also show larger variances here and does not necessarily convey the same trends. In-depth analysis of the data basis does not yield definitive answers to this, but we note that the weather data used does not originate from the exact same location as the chamber data. Thus, local effects, such as clouds or the lack thereof, might have influenced the measurements.

The experimental replication furthermore yields a type of bottleneck that correlates with the production of synthetic data. In future work, less light influx settings, i.e. day, time, and weather, should be computed to speedup evaluation, in addition to only computing a representative subset of the field. This would speedup the total evaluation significantly, and more time can be allocated for the actual data generation, after deriving a heuristic of the mapping between the parameter space and the light influx or carbon uptake limitation on the plant. For the domain composition, the challenge poses itself that a camera either needs to be facing downwards, as this is how the light evaluation is parallelized, or the render scene needs to be setup by reducing the decomposed scene into one.

4.3 Functional synthetic data using UE

One factor contributing to the increased use of UE in certain domains, specifically in plant science, is the ability to very quickly prototype scenes. Users can visually achieve a sufficiently high visual quality for many applications very quickly. This significantly lowers the barrier for testing simulation-based hypotheses or generating synthetic datasets. There are additional toolsets that partially assist or alleviate texture and geometry modelling for scenes such as the data scene, and within the scope of the model scene used in this work, the effort is comparable to other approaches. Examples of this are Helios or

DART with subsequent rendering, or any alternative to UE for synthetic data implementation. These models also require calibration of surface properties, even if respective rendering pipeline use multi-spectral or physics-based models.

There are certain aspects of the pipeline that required adjustment seeing as UE is historically a game engine, which goes beyond terminology. Certain parts of our workflow are implemented on top of basic optimizations we explicitly turn off, or circumvent using procedural geometry. Similarly, certain features might not be as fleshed out as others, particularly regarding the implementation of simulation models, requiring some fine-tuning in cases that extend beyond what a game engine typically delivers. In Section 3.2, we showcase the connection between the simulation and the eventual training output that could be used for inference in field settings. There are caveats, but generally, our method adds to the virtual scene in a graphics engine information that is otherwise not available, by employing simulation models to act as data source.

While this work focuses on the central validity and applicability of the workflow for photosynthesis problems, this pipeline is a generic data production pipeline. More functional information within the virtual world would enable a more informed embedding and ultimately strengthen the ability of deep neural networks to estimate plant health through remote sensing techniques. Camera footage together with distance sensors are comparatively low-cost and easily reproducible in experiments, and they are also always computed in UE, as pixel shaders depend on pixel depth and relative velocity information. Camera settings, particularly within the data scene, but also for the real world, are extremely important. In Fig. 3, we showcase a functional mapping and visualize the plants in a way that makes them easily distinguishable for this manuscript, but the scene is technically overlit, as two pixels yielding different light absorption information might have the same scene colour value. This is an issue that is a vital component of any functional analysis that depends on consumer camera images. Unless the exposure is configured to take only leaf surfaces into account, there will be some loss of information due to the restricted brightness range of the images. This is the case even in UE, which internally uses a high dynamic range system but is flattened into pictures to produce synthetic data. We have previously reported the dependency of threshold-based relative leaf area on image capturing conditions in Baker et al. (2023).

5. CONCLUSION

This article presents a coupling approach that is directly embedded into a data generation framework. Recent advances in data generation, particularly for agricultural science, have highlighted the need for embedded models in these frameworks. Here, we proposed a coupling with an FSPM model, highlighting the usefulness of its functional simulation for the enrichment of the data generation. A new implementation of radiative transfer representation in UE was developed to couple the FSPM CPlantBox to a light influx evaluation for photosynthesis evaluation. We show that we can embed functional properties into the virtual scene accurately, scalable, and usable for synthetic data.

We have included a replication of an experiment, a data production pipeline for light exposure efficiency and the carbon uptake limitation in plants, as well as the scalability of our method on HPC systems. Exploring how to best utilize the tools available, especially when implementing the simulations into a wider workflow, is essential.

ACKNOWLEDGMENTS

The authors thank Thuy Huu Nguyen for his advise on data interpretation and use. Furthermore, the authors acknowledge compute time granted to the JURECA-DC Supercomputer in the project visforai.

AUTHOR CONTRIBUTIONS

This work was written, revised, and edited by all authors. D.B. is primary author and conducted the implementation, data analysis, visualization, and drafted this work. D.B., M.G., and J.H.G. adapted the Synavis/UE software for this work. M.R. and J.H.G. provided access to supercomputing resources. M.G., H.S., and A.S. provided domain expertise and guided the implementation of the plant simulation and experimental replication. E.H. and A.S. guided the data analysis and research aims of this work. M.R. and E.H. are the primary supervisors for the University of Iceland. A.S. is primary supervisor of this work.

SUPPLEMENTARY DATA

Supplementary data is available at *in silico Plants* online.

Conflict of interest: None declared.

FUNDING

This work has partly been funded by the EUROCC2 project funded by the European High-Performance Computing Joint Undertaking (JU) and EU/EEA states under grant agreement No 101101903. This work has partly been funded by the German Research Foundation under Germany's Excellence Strategy, EXC-2070 - 390732324 - PhenoRob and by the German Federal Ministry of Education and Research (BMBF) in the framework of the funding initiative 'Plant roots and soil ecosystems, significance of the rhizosphere for the bio-economy' (Rhizo4Bio), subproject CROP (ref. FKZ 031B0909A). The authors would like to acknowledge funding provided by the BMBF to the Gauss Centre for Supercomputing via the InHPC-DE project (01–H17001).

DATA AVAILABILITY

The Selhausen data sets are described in Lärm et al. (2023) and Nguyen et al. (2024), and all relevant download information can be seen in those articles. Our framework, Synavis, is Open Source and the relevant codes can be seen in dhelmrich/synavis. CPlantBox is open source and available at Plant-Root-Soil-

Interactions-Modelling/CPlantBox. A video description is available at Baker (2025). The simulation code that is associated with this manuscript is available in the `feature/experiment` branch and is scheduled to be merged.

REFERENCES

Agarwal D, Kucukpinar T, Fraser J, *et al.* Simulating city-scale aerial data collection using unreal engine. *2023 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), St. Louis, MO, USA* 2023:1–9. https://doi.org/10.1109/AIPR60534.2023.10440697

Allande C, Jorba J, Sikora A, *et al.* A performance model for openmp memory bound applications in multisocket systems. *Procedia Comput Sci* 2014;**29**:2208–2218. https://doi.org/10.1016/j.procs.2014.05.206

Bailey BN. Helios: a scalable 3D plant and environmental biophysical modeling framework. *Front Plant Sci* 2019;**10**:1–17. https://doi.org/10.3389/fpls.2019.01185

Baker DN. Video to: Virtual world coupling with photosynthesis evaluation for synthetic data production. *FigShare* 2025, https://doi.org/10.6084/m9.figshare.28280780

Baker DN, Bauer FM, Giraud M, *et al.* A scalable pipeline to create synthetic datasets from functional-structural plant models for deep learning. *In Silico Plants* 2023;**6**:diad022. https://doi.org/10.1093/insilicoplants/diad022

Baker DN, Bauer FM, Schnepf A, *et al.* Adapting agricultural virtual environments in game engines to improve HPC accessibility. In: *Communications in Computer and Information Science*. Nordic e-Infrastructure Collaboration Conference, Tallinn (Estonia), 27 May 2024–29 May 2024. Springer, 2024, 1–15, https://doi.org/20.500.11815/4936

Bauer FM, Lärm L, Morandage S, *et al.* Development and validation of a deep learning based automated minirhizotron image analysis pipeline. *Plant Phenomics* 2022;**2022**:9758532. https://doi.org/10.34133/2022/9758532

Bogena HR. TERENO: German network of terrestrial environmental observatories. *J Large-Scale Res Facil JLSRF* 2016;**2**. https://doi.org/10.17815/jlsrf-2-98

Chuai-Aree S, Jäger W, Bock HG, *et al.* Simulation and visualization of plant growth using Lindenmayer systems. In: Bock HG, Phu HX, Kostina E, Rannacher R (eds.) *Modeling, Simulation and Optimization of Complex Processes*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, 115–126, https://doi.org/10.1007/3-540-27170-8_9

Farquhar GD, von Caemmerer S, Berry JA. A biochemical model of photosynthetic co2 assimilation in leaves of C_3 species. *Planta* 1980;**149**:78–90. https://doi.org/10.1007/BF00386231

Féret J-B, de Boissieu F. prospect: an R package to link leaf optical properties with their chemical and structural properties with the leaf model prospect. *J Open Source Softw* 2024;**9**:6027. https://doi.org/10.21105/joss.06027

Gastellu-Etchegorry JP, Martin E, Gascon F. DART: a 3D model for simulating satellite images and studying surface radiation budget. *Int J Remote Sens* 2004;**25**:73–96. https://doi.org/10.1080/0143116031000115166

Giraud M, Le Gall S, Harings M, *et al.* Cplantbox: a fully coupled modelling platform for the water and carbon fluxes in the soil-plant-atmosphere continuum. *In Silico Plants* 2023;**5**:diad009. https://doi.org/10.1093/insilicoplants/diad009

Krüger M, Gilbert D, Kuhlen TW, *et al.* Game engines for immersive visualization: using unreal engine beyond entertainment. *PRESENCE: Virtual Augmented Real* 2024;**33**:31–55. https://doi.org/10.1162/pres_a_00416

Lärm L, Bauer FM, Hermes N, *et al.* Multi-year belowground data of minirhizotron facilities in Selhausen. *Nat Sci Data* 2023;**10**:672. https://doi.org/10.1038/s41597-023-02570-9

Lei T, Graefe J, Mayanja IK, *et al.* Simulation of automatically annotated visible and multi-/hyperspectral images using the helios 3D plant and radiative transfer modeling framework. *Plant Phenomics* 2024;**6**:1–16. https://doi.org/10.34133/plantphenomics.0189

Leuning R. A critical appraisal of a combined stomatal-photosynthesis model for C_3 plants. *Plant Cell Environ* 1995;**18**:339–355. https://doi.org/10.1111/pce.1995.18.issue-4

Li X, Park J, Reberg-Horton C, *et al.* Photorealistic arm robot simulation for 3d plant reconstruction and automatic annotation using unreal engine 5. In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seattle, WA, USA. 2024, 5480–5488, https://doi.org/10.1109/CVPRW63382.2024.00557

Mai TH, Schnepf A, Vereecken H, *et al.* Continuum multiscale model of root water and nutrient uptake from soil with explicit consideration of the 3D root architecture and the rhizosphere gradients. *Plant Soil* 2019;**439**:273–292. https://doi.org/10.1007/s11104-018-3890-4

Martonchik JV, Bruegge CJ, Strahler AH. A review of reflectance nomenclature used in remote sensing. *Remote Sens Rev* 2000;**19**:9–20. https://doi.org/10.1080/02757250009532407

Miao G, Guan K, Yang X, *et al.* Sun-induced chlorophyll fluorescence, photosynthesis, and light use efficiency of a soybean field from seasonally continuous measurements. *J Geophys Res Biogeosci* 2018;**123**:610–623. https://doi.org/10.1002/jgrg.v123.2

Minervini M, Scharr H, Tsaftaris SA. Image analysis: the new bottleneck in plant phenotyping. *IEEE Signal Process Mag* 2015;**32**:126–131. https://doi.org/10.1109/MSP.2015.2405111

Mousavi M, Khanal A, Estrada R. AI playground: Unreal Engine-based data ablation tool for deep learning. In: Bebis G, Yin Z, Kim E, Bender J, Subr K, Kwon BC, Zhao J (eds.) *Advances in Visual Computing*. Cham: Springer International Publishing, 2020, 518–532, https://doi.org/10.1007/978-3-030-64559-5_41

Nguyen TH, Lopez G, Seidel SJ, *et al.* Multi-year aboveground data of minirhizotron facilities in Selhausen. *Nat Sci Data* 2024;**11**:674. https://doi.org/10.1038/s41597-024-03535-2

Nicodemus FE, Richmond JC, Hsia JJ, *et al.* Geometrical considerations and nomenclature for reflectance. In: *NBS Monograph*. Washington, DC, USA: National Bureau of Standards, 1977, https://doi.org/10.6028/NBS.MONO.160

Qi J, Xie D, Yin T, *et al.* Less: large-scale remote sensing data and image simulation framework over heterogeneous 3D scenes. *Remote Sens Environ* 2019;**221**:695–706. https://doi.org/10.1016/j.rse.2018.11.036

Reichenau TG, Korres W, Schmidt M, *et al.* A comprehensive dataset of vegetation states, fluxes of matter and energy, weather, agricultural management, and soil properties from intensively monitored crop sites in western Germany. *Earth Syst Sci Data* 2020;**12**:2333–2364. https://doi.org/10.5194/essd-12-2333-2020

Salesin K, Knobelspiesse KD, Chowdhary J, *et al.* Unifying radiative transfer models in computer graphics and remote sensing, Part I: A survey. *J Quant Spectrosc Radiat Transf* 2024;**314**:108847. https://doi.org/10.1016/j.jqsrt.2023.108847

Schmidt M. Ec/climate station Selhausen 3, 2024, https://doi.org/20.500.11952/TERENO/1056009555

Schraml D. Physically based synthetic image generation for machine learning: a review of pertinent literature. In: Rosenberger M, Dittrich P-G, Zagar B (eds.) *Photonics and Education in Measurement Science 2019*, Vol. **11144**. International Society for Optics and Photonics, SPIE, 2019, 111440J, https://doi.org/10.1117/12.2533485

Sievänen R, Godin C, DeJong TM, *et al.* Functional-structural plant models: a growing paradigm for plant studies. *Ann Bot* 2014;**114**:599–603. https://doi.org/10.1093/aob/mcu175

Taiz L. Agriculture, plant physiology, and human population growth: past, present, and future. *Theor Exp Plant Physiol* 2013;**25**:167–181. https://doi.org/10.1590/S2197-00252013000300001

Thörnig P. JURECA: data centric and booster modules implementing the modular supercomputing architecture at Jülich supercomputing

centre. *J Large-Scale Res Facil JLSRF* 2021;**7**:1–9. https://doi.org/10.17815/jlsrf-7-182

Thornley JHM. Plant growth and respiration re-visited: maintenance respiration defined – it is an emergent property of, not a separate process within, the system – and why the respiration:photosynthesis ratio is conservative. *Ann Bot* 2011;**108**:1365–1380. https://doi.org/10.1093/aob/mcr238

Tsaftaris SA, Minervini M, Scharr H. Machine learning for plant phenotyping needs image processing. *Trends Plant Sci* 2016;**21**:989–991. https://doi.org/10.1016/j.tplants.2016.10.002

Tuzet A, Perrier A, Leuning R. A coupled model of stomatal conductance, photosynthesis and transpiration. *Plant Cell Environ* 2003;**26**:1097–1116. https://doi.org/10.1046/j.1365-3040.2003.01035.x

van der Schaaf A, Xu C-J, van Luijk P, *et al.* Multivariate modeling of complications with data driven variable selection: guarding against overfitting and effects of data set size. *Radiother Oncol* 2012;**105**:115–121. https://doi.org/10.1016/j.radonc.2011.12.006

Von Caemerer S. Steady-state models of photosynthesis. *Plant Cell Environ* 2013;**36**:1617–1630. https://doi.org/10.1111/pce.2013.36.issue-9

Walia A, Carter R, Wightman R, *et al.* Differential growth is an emergent property of mechanochemical feedback mechanisms in curved plant organs. *Dev Cell* 2024;**59**:3245–3258.e3. https://doi.org/10.1016/j.devcel.2024.09.021

Ward D, Moghadam P, Hudson N. Deep leaf segmentation using synthetic data. arXiv, arXiv:1807.10931, 2018, preprint: not peer reviewed. https://doi.org/10.48550/arXiv.1807.10931

Yin X, van Oijen M, Schapendonk A. Extension of a biochemical model for the generalized stoichiometry of electron transport limited C_3 photosynthesis. *Plant Cell Environ* 2004;**27**:1211–1222. https://doi.org/10.1111/pce.2004.27.issue-10

Yu P, Li C, Li M, *et al.* Seedling root system adaptation to water availability during maize domestication and global expansion. *Nat Genet* 2024;**56**:1245–1256. https://doi.org/10.1038/s41588-024-01761-3

Zha Y, Yang J, Zeng J, *et al.* Review of numerical solution of Richardson-Richards equation for variably saturated flow in soils. *WIREs Water* 2019;**6**:e1364. https://doi.org/10.1002/wat2.v6.5

Zhao X, Qi J, Jiang J *et al.* Fine-scale retrieval of leaf chlorophyll content using a semi-empirically accelerated 3d radiative transfer model. *Int J Appl Earth Obs Geoinf* 2024;**135**:104285. https://doi.org/10.1016/j.jag.2024.104285