A Modeling Methodology for Crop Representation in Digital Twins for Smart Farming

Pascal Archambault
DIRO, Université of Montréal
Montréal, CA
pascal.archambault@umontreal.ca

Houari Sahraoui
DIRO, Université of Montréal
Montréal, CA
sahraoui@iro.umontreal.ca

Eugene Syriani
DIRO, Université of Montréal
Montréal, CA
syriani@iro.umontreal.ca

ABSTRACT

Digital twins of complex systems are operated by stakeholders from different domains, who typically do not work in the same language. This problem is exacerbated in digital twins where domain-specific representations are required to convey actionable results, such as in cyber-biophysical systems. Particularly, in controlled environment agriculture, agronomists devise seasonal production plans and run simulations to optimize the system in terms of crop phenology while growers maintain crops and ensure their optimal growth by assessing crop morphology. To breach this gap, we consider an optimization problem to reconcile the different users' points of views. We propose a modeling methodology that bridges the gap between crop phenology and morphology, generating visual representations of crops based on simulated phenological characteristics. To demonstrate the validity of our proposed methodology for digital twins in smart farming, we apply our approach to two case studies: a strawberry vertical farm and a smart canola field.

CCS CONCEPTS

• **Computing methodologies** → **Modeling methodologies**; *Scientific visualization*; *Simulation tools*; Continuous simulation; Discrete-event simulation.

KEYWORDS

cyber-biophysical systems, digital twins, multi-paradigm modeling, controlled environment agriculture, smart farming

ACM Reference Format:
Pascal Archambault, Houari Sahraoui, and Eugene Syriani. 2024. A Modeling Methodology for Crop Representation in Digital Twins for Smart Farming. In *ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems (MODELS Companion '24), September 22–27, 2024, Linz, Austria.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3652620.3688247

1 INTRODUCTION

Advances in computing technologies, combined with the digitization of various domains such as manufacturing [33], avionics [36], and logistics [16], have increased the prevalence of cyber-physical systems. In agriculture, digitization has the potential to make a significant impact through smart farming technologies [2], such as vertical farming. Vertical farming places crops in a fully controlled environment, known as controlled environment agriculture (CEA), to optimize yields, thereby creating a cyber-biophysical system (CBPS) [11]. The operation of such systems typically relies on the expertise of multiple domain specialists, each responsible for different aspects of the system. For example, in CEA, a grower ensures the crops grow healthily under the given environmental conditions, while an agronomist oversees the overall production schedule and environmental planning.

Integrating digital twin capabilities into CBPS allows operators of CEA systems to utilize services like what-if analyses, predictions, and visualizations to optimize production in ways that are both economically viable and aligned with business goals [11]. However, it is crucial for the CBPS to present key performance indicators (KPIs) to its stakeholders at various levels of abstraction to provide actionable insights. The digital twin must accurately represent the crop's phenology and morphology—the developmental and architectural growth dynamics—while remaining consistent with the farm's conditions and providing valuable information to both agronomists and growers.

The goal of this work is to enable the prescriptive capabilities of digital twins via simulation of future states to improve the stakeholders' decision making process. We propose a model transformation chain to visualize the crop morphology via its phenological state by employing domain-specific languages to refine an abstract morphological state into a concrete morphological 3D representation. By exploring the design space of possible crop morphologies and optimizing a range of morphological parameters, we generate a morphological representation consistent with its phenological development, enabling CEA stakeholders to assess the impact of environmental factors on morphology. The main contributions of this paper are: (i) a domain-specific language defining crop morphology using non-destructive measurements; (ii) an optimization-based model transformation to refine the output of a crop simulator into a consistent visual representation of the crop morphology; (iii) a methodology to accommodate visualizations for the different points of view of the digital twin stakeholders We validate our approach on two case studies: a strawberry vertical farm and a smart canola field. The results show that our methodology is applicable and yields actionable results for smart farming stakeholders.

In Section 2, we provide the background information and introduce our running example. In Section 3, we outline our approach and explain how we derive the abstract morphological state. In Section 4, we detail the process of refining the abstract state into a

MODELS Companion '24, September 22–27, 2024, Linz, Austria

ACM ISBN 979-8-4007-0622-6/24/09
https://doi.org/10.1145/3652620.3688247

concrete morphological state and describe how to generate its visualization. In Section 5, we validate our approach through two case studies. We then discuss related works in Section 6 and conclude in Section 7.

2 BACKGROUND

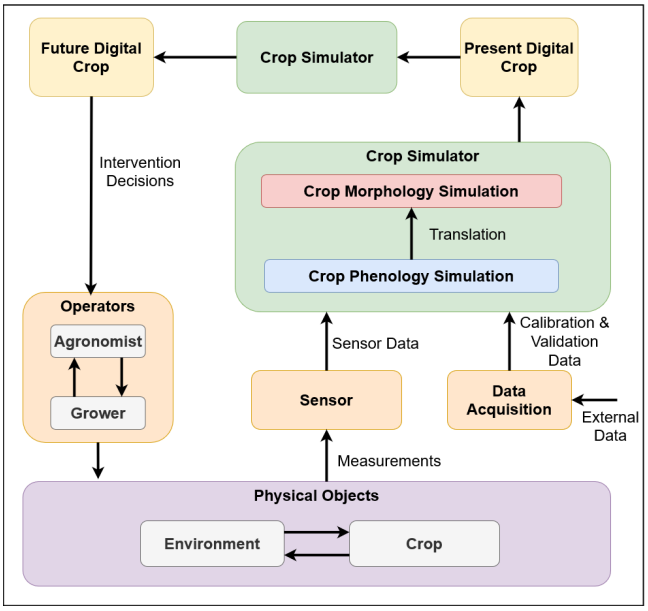

Figure 1: Prescriptive digital twin conceptual framework, adapted from [38]

In this section, we present a running example of the challenges encountered when operating a CBPS in a CEA context. We then define key concepts and techniques that are used in the remaining sections.

2.1 Running example: a digital twin for CEA

As a running example, let us consider the CEA of tomatoes with two stakeholders: an agronomist and a grower. CEA systems are complex to operate and involve the use of multiple cyber-physical systems to regulate ventilation, irrigation, radiation, chemical distribution, and pressure within the environment [32]. By maintaining direct control on the environment, agronomists and growers can adapt the growth conditions of crops to satisfy specific business needs. For instance, a tomato nursery aims to nurture and sell tomato seedlings, whereas a tomato farm wants to grow the most marketable crop yield.

To operate the CEA system, the *agronomist* plans *environmental recipes* which define a production calendar detailing the environmental conditions necessary to achieve the desired yield outcome. The agronomist chooses the recipes based on their tacit knowledge and the pheonological data available for specific tomato cultivars, such as the density of roots or chlorophyll content of leaves. Most of this data needs to be collected in a lab by destroying crops, which reduces potential production and removes longitudinal data points from the production cycle. Recipes typically do not exploit the full

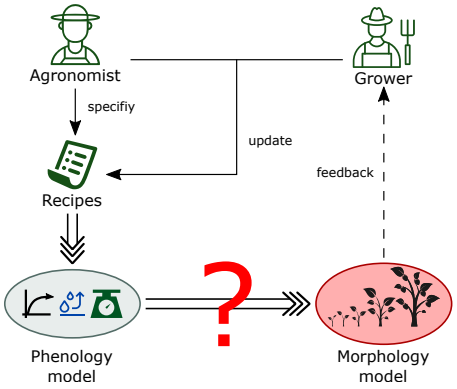

Figure 2: Knowledge gap between agronomist and grower

range of possible environmental conditions of the CEA system, as uncertainty brought by experimental recipes may lead to large discrepancies between the expected and produced yield.

While the agronomist handles the CEA production plan, the *grower* ensures that the tomato crops receive the proper care throughout production. For example, the grower ensures that irrigation remains adequate, that the tomato crops are in good health, and prunes organs contributing biomass that negatively impacts crop yield. Due to the inherent nature of crop development and the added system inertia caused by mass crop production, the grower must adjust the growing conditions daily, slightly deviating from the recipes provided by the agronomist. The grower relies on their tacit knowledge to visually inspect the crop and act upon it.

A digital twin for this farm would allow it to improve its efficiency in terms of yield (fruit production) and energy consumption (cost). Consider the components of the prescriptive digital twin (inspired from [38]) presented in Figure 1. In this paper, we focus on the crop simulator (in green) to provide agronomists and growers a tool to evaluate the impact of environmental configurations on the phenological and morphological development of the crop based on the analysis of simulation results.

The operational complexity of CEA systems makes the interactions between the agronomist and grower crucial to achieving production goals. Both experts must work together to iterate on a production plan that meets business expectations and ensures healthy crop development. To achieve this, those domain experts rely on simulation-based decision support systems [4] to explore the state space of the CEA system and provide recommendations for its operations. Typically, such systems provide support for the agronomist via the simulation of phenological crop characteristics, such as photosynthesis and evapotranspiration, but fail to support the grower with a model of the tomato morphology. While tomato morphological models exist, their construction and usage is often decoupled from the underlying phenological phenomena that characterize crop growth [14]. By bridging the gap between the crop phenology and its morphology, we aim to provide a consistent multi-view representation of the crop to support the management decisions of smart agriculture system operators. Figure 2 highlights the knowledge gap our methodology aims to bridge between domain experts.

Our methodology aims to provide the visualization service necessary for the engineering of a smart farming digital twin. Borrowing definitions from a recent conceptual framework on the topic [38], we aim to engineer a prescriptive digital twin to provide agronomists and growers a tool to evaluate the impact of environmental configurations on the phenological and morphological development of the crop based on the analysis of simulation results.

2.2 Crop Phenology

Definition 1. Crop phenology characterizes the underlying functions regulating crop development in regards to their environment. For example, crop photosynthetic activity and biomass allocation are phenological characteristics with dynamics that highly vary depending on crop genetics and growth environment. Agronomists often rely on specific crop phenological characteristics to categorize crop development into vegetative and reproductive stages. Measuring crop phenological phenomena involves laborious, expensive, and often destructive methods [9].

Models of crop phenology capture the underlying dynamics of crop growth in relation to its environment. Typically, such models comprise modules for the computation photosynthetic activity, soil nutrient exchange, evapotranspiration, and farming operations affecting the crop. These models are used by agronomists to predict the impact of environmental recipes on phenological crop attributes such as the weight of organs, the distribution of chemicals within the crop, and water evapotranspired. Since the dynamics of the crop continuously evolve through time, phenology is described through systems of differential equations. Due to the complexity of their constituent parts, there are no analytical solutions that can model the entirety of crop phenology and its factors, and only a few general solutions exist for very specific characteristics [41]. Thus, simulation is often employed to approximate model results. For instance, the `TOMGRO` simulation model [21] involves 21 constants to characterize specific tomato cultivars. Typically, most model parameters must be calibrated through destructive measurements, killing the crop.

By describing the evolution of underlying crop functions across time, these models present actionable results to the agronomist via graphs and provide a basis for optimizing production plans. The agronomist can simulate different recipes and assess the simulated system traces to find optimal growth conditions. However, phenological models fail to provide actionable insights to the grower since their domain expertise lies on the visual assessment of physical crop characteristics, not on the understanding of underlying developmental phenomena. Thus, while models of phenology provide actionable results to the agronomist, there is a need to model the morphology to provide actionable results to the grower.

2.3 Crop Morphology

Definition 2. Crop morphology defines the architectural and structural development of the crop, and includes functions such as branching patterns, and organ size and shape. Typically, most morphological characteristics do not necessitate destructive measurements, as it is possible to count the number of organs and measure their dimensions. It is the most appropriate level of abstraction for non-agronomists.

Models of crop morphology aim to capture the topology and relations between crop organs. They include cultivar-specific parameters, such as branch angle or reproduction thresholds to capture organogenesis and branching patterns. Also known as functional-structural plant models [39], they represent the physical characteristics of the crop and are useful for growers and agronomists to assess the evolution of the crop within their domain of expertise. Compared to crop phenology models that simulate underlying crop functions, morphological models simulate measurable and non-destructive crop attributes, which can be observed to characterize crop genotypes [7]. Following our running example, morphological models of tomatoes have been developed to infer the development of the crop structure via greenhouse images [27].

2.4 L-System

Definition 3. The L-system formalism [26] models the growth processes of biological organisms, with an emphasis on crop development. It provides a framework for describing complex development patterns through the use of formal grammars and iterative, parallel rewriting rules and is a popular simulation formalism for crop morphology. Many variations of the formalism exist, but we will focus on parametric L-systems.

A parametric L-system is described as a tuple $G = \langle \Sigma, \omega, P, I \rangle$ where Σ contains the symbol of an alphabet, ω is the axiom, P is a set of parameterized production rules and I a set of interpretation rules. Starting from the axiom ω, the evolution of the system is simulated by repeatedly computing the new parameters and applying the production rules on the previous iteration string. Interpretation rules define how to interpret the derived L-System string according to the rules of turtle graphics [31].

2.5 Turtle Graphics

Turtle graphics [3] is a method for visually interpreting and representing strings generated by L-systems. The method sets a cursor, or so-called turtle, on a blank canvas that interprets each symbol as specified by the L-system interpretation rules to draw the appropriate shapes. The turtle reads the string iteratively, and when a separate context is found in the interpreted string, a new turtle starts drawing in parallel. This results in the efficient interpretation and rendering of L-System strings, bridging the gap between grammar and graphics.

3 MODEL TRANSFORMATION CHAIN

We first present the overall architecture of our transformation chain, then describe how to produce a phenological state from a model of crop phenology. Finally, we propose an abstract morphological state and describe how to obtain it from a phenological state.

3.1 Overall architecture

The prescriptive digital twin of a smart farming system, as shown on Figure 1, must represent the crop and simulate its possible future states to provide intervention decisions to operators. Models of phenology enable the digital twin to predict the impact of the environment on the phenological characteristics of the crop but fail to provide information on the crop structure. In contrast, morphological models fail to capture the impact of environmental conditions

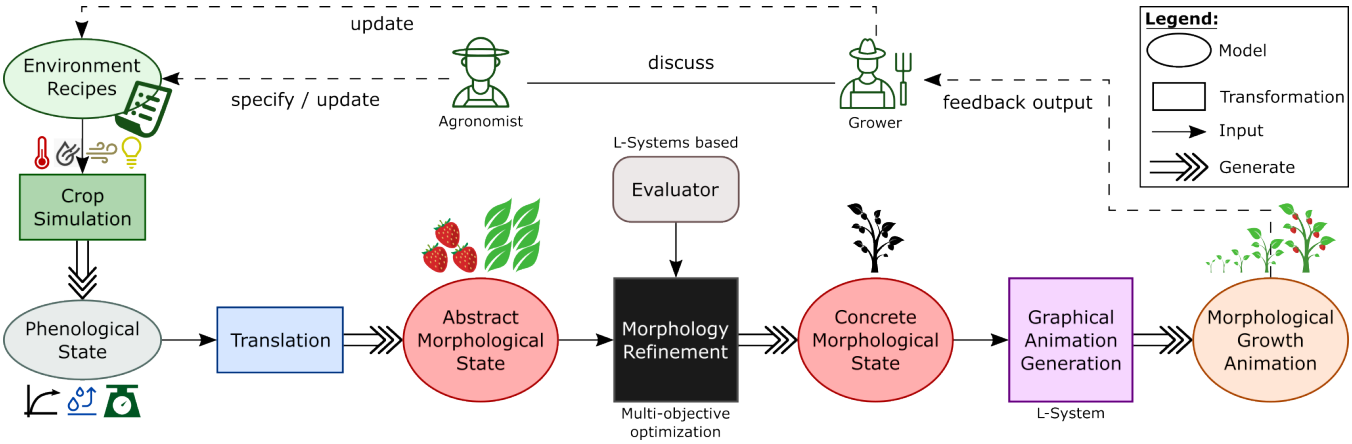

Figure 3: Methodology for the generation of crop phenology from environmental recipes

on the crop. Thus, we propose a methodology that bridges the gap between different aspects of crop growth.

To bridge the gap between crop representations, our architecture sequentially transforms models of different formalisms to achieve the graphical representation of a crop from a given set of growing conditions. The conditions become the input of a crop simulator, which outputs the phenological state of the crop. This state is then mapped to an abstract morphological state using a model transformation (i.e., translation), which gets refined into a concrete morphological state through a multi-object optimization-based transformation. From the optimal concrete state, we generate an animation of crop growth using L-Systems to convey the graphical results. Figure 3 summarizes this transformation chain.

The phenological and morphological models capture specific, actionable KPIs for the agronomist and grower, which would include predicted yield and branching patterns in the case of a tomato crop. However, the different models typically do not work jointly to provide a consistent multi-view of the system. As such, the models may conflict in their results by projecting different organogenesis patterns or by providing mismatching estimates of the same KPI, such as leaf area index [8]. By coupling the morphological model to its phenological counterpart through an abstraction of crop morphology, we can present consistent results at multiple levels of abstraction to the grower and agronomist while producing specific KPIs that guide the farming system operation. In the case of a tomato greenhouse, this would include information about the evapotranspiration of the crop, its weight, and its height.

3.2 Producing the phenological state

Phenology is typically modeled as a dynamical system through differential equations, which describe the system continuously values. However, we want to capture the state of development of the crop at discrete intervals to generate results at specific points in time. This can either be done through deriving an analytical solution to the system, which is unfeasible for most biological systems due to their complexity, or by using simulation to approximate the solution using numerical methods [17]. The simulator outputs a trace of model variables that are used to represent the phenological state of the crop at specific points in time.

3.2.1 Crop simulator. Simulation models of crop phenology simulate the continuous growth of the crop based on its environment. Such models incorporate modules that compute photosynthesis [42], water and chemical absorption [29], and the effects of temperature on crop development [25], and output crop growth phenological characteristics such as crop dry weight, evapotranspiration, and relative growth rate [22]. Since crop simulation models vary in complexity and implementation, the representations of phenological states vary. For the interested readers, we refer to a recent survey on crop yield models [13].

3.2.2 Phenological state representation. The phenological state encapsulates the results of the crop simulator, which can take many forms, such as an object or a time series of properties. We export the simulator output to a CSV file to integrate the simulation results with our optimization algorithm. This is done automatically by the crop simulator, but as simulation models vary in implementation, the file may need to be filled out manually by the agronomist. Each row of the file represents the phenology of the crop at every simulation time step, typically a day, and contains information about the crop. For the tomato example, the file might contain the age and weight of organs. The translation then uses the file to generate the abstract morphological state.

3.3 Generating the abstract morphological state

As crop simulation models typically aim to assess final crop yield, they often do not accurately capture the organogenesis patterns of every organ and their counts and are often unable to characterize the crop morphology. Thus, to bridge the gap between phenological and morphological characteristics, we generate an abstract morphological state from the simulated crop characteristics. To do this, we employ a user-defined translation heuristic that maps the phenological state to a concrete abstract morphological state based on the available phenological state representations output by the crop simulator. Certain phenological models directly compute the counts of organs of interest, which are directly translatable to an

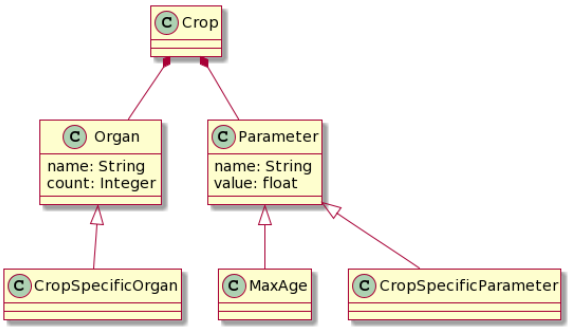

Figure 4: Abstract Morphological State Metamodel

abstract morphological state. However, this is not the case for every phenological model, as relations between phenological attributes and observable morphological attributes must be defined, such as a mapping from organ weight to organ count.

3.3.1 Abstract morphological state. The abstract morphological state is an intermediary representation between the phenology and morphology of the crop that bridges the gap between models relying on destructive measurements such as crop weight and non-destructive measurements of crop organ counts. If we take the tomato example, the abstract morphological state describes the leaf count, tomato fruits, length of the main stem, and its age. By introducing this abstract state between models relying on destructive and non-destructive measurements, we aim to characterize different crop genotypes by calibration of the translation heuristic.

The abstract state is implemented with a domain-specific language that captures the crop-specific parameters used for the morphology refinement. The language relies on user-defined crop modules that characterize the metamodel of the abstract morphological state, comprised of organ types and simulation parameters as shown on Figure 4. The metamodel of the abstract morphological state relies on crop-specific organs and parameters that must be defined for individual crop species within a Java object loaded by the language. We provide such examples in the case studies. The abstract state is then refined through our optimization-based transformation to generate the appropriate concrete morphological state.

3.3.2 Translation. The translation heuristic maps the output of the crop simulator to the abstract morphological state and is derived by the agronomist based on their knowledge and experience with specific crop cultivars. As crop growth is highly sensitive to its environment, and as developmental characteristics depend on the crop cultivar and species, this heuristic function must be adjusted to each farming operation. To do this, agronomists must experiment and derive their parameters by collecting destructive crop measurements, i.e., measurements that kill the crop, such as vegetative dry weight, to correlate organ weight to organ count. If available, the agronomist can use a simulator to estimate destructive crop measurements. Once this mapping between organ weight and count is defined, we translate the phenological state to an abstract morphological state. The translation operation can be automated via an interface linking the outputs of the crop simulator with the abstract morphological state language. As agronomists are not computer

```
S+[S[SF][SL]]-[S]
```

Figure 5: Example tomato L-System interpretation String. S denotes a stalk, +/- a rotation, F a fruit, and L a leaf.

experts, they can manually apply the translation using spreadsheet tools such as Microsoft Excel, from which they can manually input the abstract state using our domain-specific language.

4 MORPHOLOGY REFINEMENT AND GROWTH REPRESENTATION

Given the large state space of the morphological model, which results from the numerous possible varying parameters, it is essential to explore this space efficiently to refine the abstract state into its concrete form. To achieve this, we model the space exploration as a multi-objective optimization problem and solve it using a genetic algorithm. Through this algorithm, we derive the parameters of the concrete morphological crop model, leading to the creation of a concrete morphological state, enabling the visualization of the crop morphology. Multi-objective genetic algorithms are particularly effective because they can identify Pareto-optimal solutions. From these solutions, we select the optimal one using a user-defined selection heuristic. This heuristic acts as a tie-breaker among the solutions, prioritizing specific crop organs.

4.1 Morphology Refinement

To refine the abstract morphological state, we employ an optimization-based model transformation that uses a crop-specific L-System model to simulate possible morphological configurations of the crop and select which parameters lead to an optimal representation of the crop morphology. The optimal parameters of our morphological model are selected with a user-defined heuristic that prioritizes correct organ counts for specific organs.

4.1.1 Concrete Morphological state. The concrete morphological state of the crop is the result of a parametric L-system model, which characterizes the structural and topological development of the crop by applying parallel string rewriting operations on a derivative string. This derivative string is then transformed to an interpretation string, as shown on Figure 5, and interpreted by the turtle to visualize the crop. Our optimization-based transformation derives the parameters that generate a concrete morphological state closest to the specified abstract morphological state.

4.1.2 Crop Specification. L-system models for different crops have different nomenclatures for organ names and types. To define the abstract morphological state of any crop, the DSL relies on crop-specific modules that define a mapping between the interpreted L-System string and the crop organs. The mapping pairs an organ with a regular expression used to extract crop organ counts from the string, which we use to optimize the L-system model parameters. For instance, the interpreted string of a tomato might contain the character `F` to represent fruits, which we count using a regular expression.

4.1.3 Optimization Problem. To find the optimal concrete morphological state that characterizes the abstract state, we pose a multi-objective optimization problem where we minimize the difference between the generated organ count and the target specified by the DSL. That is, given a tomato crop with organs $o_i \in O$ such as o_{leaf} and o_{fruit} and their respective count n_i, and $C_i(S_j)$ a function evaluating the count of organ o_i for the candidate solution S_j for parameters $x_{veg}, x_{rep} \in X$, we define objective functions F_i counting the distance between the target organ count and candidate solution:

$$F_i(x_j) = |\lambda_i C_i(S_j) - n_i| \tag{1}$$

As crop morphological models may not simulate precise organ counts but rather the branching structure, we scale the count of organs i for a solution with parameter λ_i. This way, λ can be adjusted to minimize the impact of an over or under-estimation of organ counts by the morphological model on the family of solutions.

We pose the following multi-objective optimization problem for each objective F_i:

$$\forall i \in o_i \min_{x_j \in X} F_i(x_j) \tag{2}$$

Our goal is to minimize the distance between the count of each generated organ and the target specified by the abstract state. To solve our multi-objective optimization problem, we resort to genetic algorithms.

4.1.4 Genetic Algorithm. We use the NSGA-III algorithm to solve our optimization problem [12]. This choice is motivated by the ability of NSGA-III to tackle multiple objectives with large amounts of parameters. We promptly illustrate the NSGA-III algorithm using the tomato example.

The tomato crop module provided to the DSL specifies which parameters need to be optimized, along with their constraints, if any. For example, we may wish to optimize parameters x_{veg} and x_{rep}, the vegetative and reproductive plastochron, to obtain a count for each organ $o_i \in \{fruit, leaf\}$.

The algorithm first encodes our parameters into random candidate vectors $\langle x_{veg}, x_{rep} \rangle \in X$. To bind the space of possible vector values, each parameter is discretized and encoded as a binary string. Then, the solution space is divided p times for each of our M objectives to create $H = \binom{M+p-1}{p}$ reference points on an M-dimensional normalized hyperplane. The algorithm then normalizes our solutions into the normalized hyperplane space. We refer the reader to [12] for details on this normalization and hyperplane construction. To maintain a diverse set of solutions, we set the number of divisions to $p = 15$.

Next, the algorithm selects k points closest to the H reference points. When a point is closest to multiple reference points, e.g., $k < H$, random candidates are randomly generated to be in the neighborhood of the most referenced point until a population P_1 is formed. From this population, an offspring population Q_1 is created by applying a two-point crossover operation and a mutation operator, each with a 50% chance. The mutation randomly increases or decreases the value of the encoded binary string with a 50% chance.

Once the population $P_1 \cup Q_1$ is obtained, N points are selected using a fitness function to form the next population P_2. To do this, the morphology of every candidate is simulated using the morphological model, and evaluated by summing each objective function F_i, which we minimize. To converge to an optimal solution, the algorithm iterates on the population h times, starting again from the construction of the hyperplane. The output of the genetic algorithm is a set of Pareto-optimal solutions S, containing optimal solution vectors $\langle x_{veg}, x_{rep} \rangle_j$ to our optimization problem, which we reduce to a single optimal solution using a selection heuristic.

4.1.5 Selection Heuristic. To prune our solution set, we apply a crop-specific heuristic function that prioritizes certain organs over others. Continuing with our tomato example, we may select a solution vector $\langle x_{veg}, x_{rep} \rangle$ that produces an optimal number of fruits over an optimal number of leaves. The agronomist typically chooses this heuristic function according to his knowledge of the crop cultivar. Formally, the function applies some preference over certain organ counts, that is, for organs $k_1, k_2 \in O$ where k_1 is preferred over k_2:

$$\min_{x^* \in X} F_{k_1}(x^*) \leq \min_{x^* \in X} F_{k_2}(x^*) \tag{3}$$

After applying our selection heuristic, we obtain the optimal set of morphological parameters from which we generate the visualization.

4.1.6 Implementation. The abstract morphological state is refined to a concrete state through an XText domain-specific language that interfaces with the L-System simulator VLab [23] through a Java API. The DSL represents the abstract morphological state via a grammar comprising the crop age and organ keywords specified by the `CropSpecificParameters` (see Figure 4). Once the language input file is defined, we refine the abstract state using our transformation, which uses the NSGA-III algorithm provided by the Java MOEA framework [18] to explore the design space.

To interface with VLab, we implemented an interface that spawns processes with the necessary context to call the L-System simulator programmatically. Every time VLab is called, a file containing morphological parameters is created, and when the simulation ends, we save either the resultant derivation or interpretation string to a file, depending on the specified objectives. The objectives define pairs of organ names and regular expressions that are used to extract specific organ counts from the generated string. The choice to extract organs from either the derivation or interpretation string is provided since the concrete morphological states are arbitrarily complex, and defining regular expressions for one string type may be easier.

4.2 Crop Vizualisation

When an optimal concrete morphological state has been found, we present it graphically in two possible ways: either by visualizing the final resulting crop and its evolution as a timeline of simulation snapshots, from which we generate an animation. The default interpretation of the L-System string generates the final state of the crop by interpreting the entire string using turtle graphics. Depending on the implementation of the morphological model, snapshots of the morphological states can be replayed to a specific threshold and displayed side-by-side to assess the evolution of the crop. The

strings can be re-interpreted by the turtle step-by-step to generate an animation of the evolution of the crop. It is possible to view the entire animation at once or step through it.

5 CASE STUDIES

We applied our methodology to two different crops, strawberry and canola. We first present a typical scenario that motivates our methodology in a controlled environment agriculture context, then evaluate the results demonstrating the validity of our approach.

5.1 Application to strawberry

5.1.1 Motivation. Strawberry crops typically grown in colder outdoor climates benefit from controlled environment agriculture to control florescence patterns that optimize fruit yield [34]. Such patterns, known as thyrse, are not found in other crops [24] and require precise control of the environment through simulation to optimize development [11]. Using our methodology, a grower can assess the thyrse architecture of simulated crops.

5.1.2 Crop phenological model. We simulated the phenology of a strawberry of the Albion cultivar using a Python simulator developed for the digital twin of our industrial partner [5]. The model outputs the plant age in days α, fruit yield $yield_{fruit}$, and organ counts for η_{fruit} fruits and η_{leaf} leaves, but fails to characterize the thyrse structure. The experiment was conducted over a growing season of 8 weeks.

The operation of the controlled environment system depends on measurable attributes such as leaf and fruit counts. By optimizing the thyrse pattern, growers can effectively produce an adequate ratio of leaves to fruits that lead to the least energy waste since too many leaves hamper fruit yields, and too many fruits hamper the health of the crop. Thus, the phenological model must be augmented with a morphological model to provide branching information to our stakeholders.

5.1.3 Crop morphological model. We simulate the morphological development of the crop using the VLab model developed by Lembinen et al. [24], which models the dynamics of thyrse architectures of woodland strawberries. By varying the model parameters, it is possible to capture different strawberry genotypes that differ in fruit and leaf organogenesis.

In total, we vary eight parameters to characterize the growth of thyrses, with 10^12 possible combinations of parameters in our search space. Figure 6 showcases the parameter file used by VLab at every evaluation of our genetic algorithm.

5.1.4 Translation to abstract state. The phenological model outputs the average number of fruits and leaves for the strawberry crop, so we did not need to use a translation heuristic to derive our abstract morphological state from the simulated mass. To distinguish the number of leaves between fully mature leaves (i.e., tri-lobe) and growing leaves (i.e., single-lobe), we assumed that growing conditions were adequate to support 66% of leaves to maturity. We obtain the following translation from the phenological to the

```
#define MAX_AGE 8
#define VEG_INIT 0.61
#define VEG_HALF_LIFE 0.25
#define TH_MONO 0.11
#define TH_SYM 0.8
#define TH_DIFF -0.05
#define PLASTOCHRON 0.42
#define BRACT_TH_1 0.5
#define BRACT_TH_2 0.5
```

Figure 6: Example VLab strawberry parameter file, only optimization parameters are shown

abstract morphological state:

$$
\begin{aligned}
n_{fruit} &= \eta_{fruit} \\
n_{single} &= \lceil 0.33 \times \eta_{leaf} \rceil \\
n_{tri} &= \lfloor 0.66 \times \eta_{leaf} \rfloor \\
age &= \alpha
\end{aligned}
$$

5.1.5 Abstract state metamodel. The abstract state for the strawberry captures the observable measurements of interest in a vertical farming context, in our case, the number of single and triple-lobed leaves and the fruit count. The state also defines the age of the crop which we want to generate. In conformity with the metamodel, we manually apply the translation to the crop simulation results to generate the abstract morphological states for our strawberry cultivars:

Listing 1: Abstract morphological states of the simulated Albion cultivar

```
Strawberry(
  SingleLobe: 3
  TriLobe: 6
  Fruits: 8
  Age: 8
)
```

5.1.6 Validation. To validate our approach, we generated the concrete morphological states of the Albion strawberry cultivar. The final simulated phenological state was exported to text and then translated into an abstract morphological state using our translation heuristic. Next, we refined the model using our optimization-based transformation to generate the concrete morphological state.

Genetic algorithms are stochastic and, at best, provide solutions lying on the Pareto front, which may be too computationally expensive to explore due to its size and variance. As such, we evaluate aggregate results from parallel executions by taking the mean absolute distance between the generated crop and the objective and the average hypervolume, as shown on Table 1. We compare the NSGA-III algorithm to random search to assess the applicability of our approach.

We parameterized the reference point set of NSGA-III by applying 15 divisions of the normalized hyperplane to preserve solution diversity. As our search space contains 10^{12} possible configurations,

and as the evaluation of each solution can be computationally expensive, we kept a population size of 100 solutions evolved over 100 generations. To compare our results to random, we sampled the n best solutions generated by random, where n is the size of the NSGA-III solution set.

We observe that our approach outperforms simple random search of morphological configurations. Every execution of the genetic algorithm yielded optimal representations in terms of fruits and tri-lobe leaves, while random sometimes fails to generate such solutions. As the random algorithm still found at least one optimal configuration in almost every execution, the hypervolumes of both algorithms are nearly identical. Since our solution sets lack diversity, the hypervolume is not a good indicator to evaluate our approach; instead, we pay attention to convergence time. The results in Figure 7 show that, on average, NSGA-III converges to an optimum in 50 generations, while random sometimes fails to converge. Convergence is an important aspect of our methodology, and we expand on that aspect in the discussion. An example of optimal strawberry representation is shown on Figure 8.

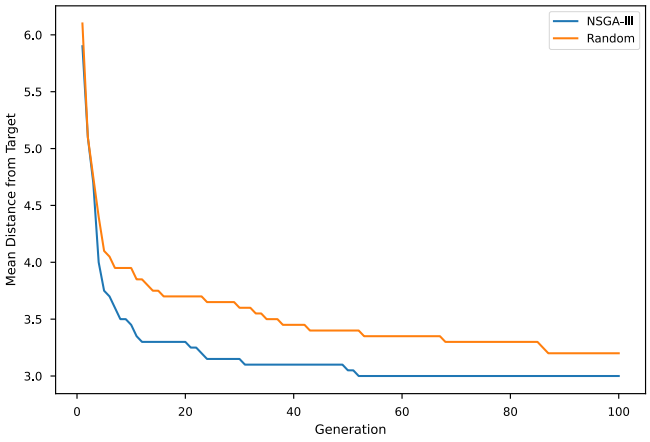

Figure 7: Population Comparison between NSGA-III and Random

Figure 8: Optimal morphology of the strawberry, missing 3 single-lobed leaves. Results are shown for weeks 2 to 8.

5.2 Application to canola

5.2.1 Motivation. Canola crops can hardly be grown indoors due to their harvesting method and exhibit beneficial interactions with their planted soil microbiome [28]. However, growers typically fertilize their soils with nitrogen to increase their yield, which may negatively impact the soil quality [6]. The growth of canola follows a well-defined sequence of vegetative and reproductive phases, the lengths of which depend on the growing conditions. The two phases are further split into seven stages [1]: germination, leaf production, stem elongation, flower production, flowering, pod development, and seed development. Most of the canola harvestable yield starts to form in the pod development stage, where $40 - 55\%$ of flowers transform into pods. A particularity of canola is that the pods contribute to the photosynthesis process, thus the growth dynamics of the crop are directly affected by the number of flowers. By controlling the crop to produce a specific amount of flowers and using specific fertilizer treatments that benefit the production of pods, it is possible to increase the total harvestable crop yield.

5.2.2 Crop phenological model. We simulated the development of two canola harvests, one with no soil fertilizer treatment (0N) and one with fertilizer treatment of $100 kgha^-1$ nitrogen (100N), based on the experiments used to develop the CSM-CROPGRO-Canola model [20] provided by DSSAT. The experiments were conducted in 2014, where canola was planted in late May and grown over 104 days.

The model of the crop incorporates field management activities such as fertilizer application and constant irrigation and simulates various properties related to soil and crop development. In terms of crop phenology, we are interested in the model results for the number of pods per meter squared $\eta_{pods} \cdot m^{-2}$ with a crop density d of 62 crops/m^2, which is how we evaluate yield. While the model is able to simulate individual seed numbers, that level of granularity is too specific for our methodology. However, the model is unable to simulate the count of flowers, which is an important morphological property of the crop.

5.2.3 Crop morphological model. We simulate the morphological development of the canola crop using the VLab model developed by Cieslak et al. [10], which models the crop architecture to generate images for automatic data labeling. While intended for image-based phenomics, we repurposed the model to graphically represent the underlying canola phenological phenomena. A main limiting factor of this model, as highlighted by its authors, is the overestimation of organs, mostly flowers, due to the self-repeating structure of branch developments. To palliate this, we adjusted the λ_{flower} parameter to be 0.01, as the VLab model outputs flower counts at maturity in the thousands. In contrast, a mature canola crop typically produces around 50 flowers [1]. By varying the model parameters shown on Figure 2 we alter the counts of pods and flowers to obtain the concrete morphology.

5.2.4 Translation to abstract state. We define a translation heuristic to convert the number of pods produced by the DSSAT model to the number of flowers to generate the abstract morphological state of both canola treatments. This proves challenging as we do not have access to canola inflorescence count. Ideally, we could infer ratios specific to certain canola genotypes or treatments with more

Algorithm	Avg. solution fruit	Avg. solution single-Lobe	Avg. solution tri-Lobe	Avg. hypervolume
NSGA-III	0 +/- 0	3 +/- 0	0 +/- 0	0.25 +/- 0
Random	0.11 +/- 0.25	3 +/- 0	0.86 +/- 0.42	0.24 +/- 0.014

Table 1: Strawberry optimization algorithm results

Listing 2: Example VLab canola parameter file, only optimization parameters are shown

```
#define MaxPlantAge 104
#define VegPhytomers 20
#define MaxPhytomers 100
#define PoddingAge 90
#define VegPlastochron 0.18942437
#define RepPlastochron 0.12848748
```

agronomic data. For our translation heuristic, we assumed that half of the canola flowers turn to pods based on an agronomic rule of thumb [1] and posed that there are twice as many flowers as there are pods:

$$n_{pods} = \eta_{pods}$$
$$n_{flowers} = \lceil 1.5 \times \eta_{pods} \rceil$$
$$age = \alpha$$

5.2.5 Abstract state metamodel. The abstract state for the canola represents the variables of interest, in our case, the number of flowers and pods, and parameters for the age of the crop. By applying our translation to the simulation results, we obtain the following abstract morphological states for both canola treatments:

```
Canola(
  Flowers: 45
  Pods: 30
  Age: 104
)
```

(a) 0N Canola

```
Canola(
  Flowers: 111
  Pods: 74
  Age: 104
)
```

(b) 100N Canola

Figure 9: Abstract morphological states of canola with no fertilizer (a) and fertilizer treatment (b)

5.2.6 Validation. To validate our approach in a smart agriculture canola growing system, we generated the concrete morphological states of two canola crops exposed to different fertilizer treatments using the CSM-CROPGRO-Canola model [20] packaged with DSSAT. The first canola crop was simulated without any fertilizer treatment, while the other was simulated with the application of a fertilizer with a nitrogen concentration of 100 kg.ha^{-1}. As canola flowers transform into pods, and as pods get replaced throughout the canola growth cycle, we assumed that the phenological state at the end of the growing period characterizes the final crop pre-harvest, ignoring missing pods. We exported the phenological state of the crop to a comma-separated value file and then translated it manually into an abstract morphological state using our translation heuristic. We then refined the model using our optimization-based transformation to generate the concrete morphological state.

Similarly to the strawberry use-case, we executed our algorithm parallel on 20 machines. The search space is smaller than for the strawberry case study as there are 10^9 possible configurations. Thus, we kept a population size of 100 evolved over 30 generations. We use the same comparison methodology as the previous case to compare our results, shown on Table 2.

Our results show that the solution set of the NSGA-III algorithm is very homogeneous, while the random algorithm is highly varied. Both algorithms fail to generate a canola crop with pods, which limits the relevancy of the generated representation. While random produces solutions with large variances, it is still able to find at least one pareto-optimal configuration and achieve the same hypervolume as NSGA-III. As shown on Figure 10, NSGA-III fails to converge to a better solution than random in the case of unfertilized canola, while it is slightly better for fertilized canola. Figure 11 shows an example of optimal canola representation found using NSGA-III.

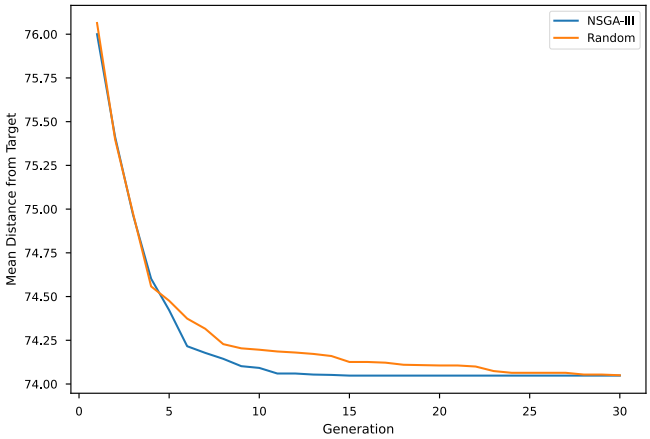

Figure 10: Population Comparison between NSGA-III and Random

5.3 Discussion

Our approach is able to generate near-optimal representations of strawberry crop morphology based on a model of their phenology, bridging the gap between operators of the vertical farm. Results for canola crops were mixed, and further work on the calibration of various components of the methodology is required to achieve actionable representations. While LSystem models enforce the well-formedness of crop structure, they may use abstractions to simplify computing costs, leading to inaccurate organ counts. For instance, the self-repeating structure of the canola crop branching patterns recursively creates branches with fixed amounts of flowers, getting smaller in size until getting visually unnoticeable, which leads

Cultivar	Algorithm	Avg. Pods	Avg. Flower	Avg. Hypervolume
0N	NSGA-III	30 +/- 0	0.398 +/- 0	0.599 +/- 0.001
0N	Random	30 +/- 0	6.701 +/- 3.971	0.6 +/- 0
100N	NSGA-III	74 +/- 0	0.048 +/- 0	0.0133 +/- 0
100N	Random	74 +/- 0	3.54 +/- 2.09	0.0133 +/- 0

Table 2: Canola optimization algorithm results

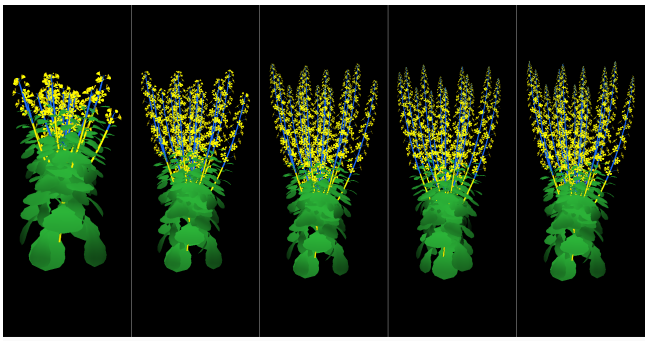

Figure 11: Optimal concrete morphology of the 100N canola. Results are shown for every 20 days until 100.

to an overestimate in flower counts between a real and virtual canola crop. This lack of fidelity between crop models and their real counterpart is mitigated by the grower manually validating the proposed solutions but would require further calibration of our translation heuristic.

The generated morphologies may vary wildly in appearance and organogenesis patterns, and it is possible that no generated morphology represents the morphology of the crop in the real system. Due to this variance and the stochasticity of our genetic algorithm, we had to generate a lot data points to have an accurate representation of possible morphologies. As digital twins continuously evolve in time, the state of the virtual system needs to be updated repeatedly. As such, the convergence speed of the optimization algorithm must be fast enough to accurately depict the system. In its current state, our methodology is fast enough to support what-if analyses but is limited in its real-time usage due to the computational cost of the L-System model evaluation.

As morphologies are not derived from sequential morphological states but rather from the initial state of the crop, our methodology is best suited to what-if analyses, where consistent and sequential morphological representations are unnecessary. The agronomist and grower can assess the desired structure of the crop and its phenological attributes to search for optimal configurations of the farming system through exploring the possible system state space, which is an integral part of prescriptive digital twins for smart farming [38].

6 RELATED WORK

Digital twins of cyber-biophysical systems have been applied to drug discovery [35] and livestock farming [30]. In smart agriculture, digital twin prototypes have been developed for the identification of diseases from images [37] or the harvesting of potatoes [15]. Such approaches either do not integrate the collaboration of multiple stakeholders for the operation of the system, or do not model different abstractions of the physical entity. Prior work on the digital twins of a strawberry vertical farming has identified key challenges and learned lessons on the topic [11], where authors highlight the importance of domain-specific tools the digital twin operators.

Recent approaches in crop modeling have focused on the image-based phenomics calibration of models [10] and the extraction of morphological attributes using multiple detection methods [43]. Image-based phenomics for the inference of crop maturity stages have been applied to wheat [19] and rice [40], but the inferred phenological attributes are not sufficient to operate a vertical farm. Thus, models of phenology and morphology must be coupled to capture the behavior of crops within smart farming digital twins.

7 CONCLUSION

In this paper, we present a methodology to generate 3D visualizations of crop morphology from a model of crop phenology for the operation of a smart agriculture prescriptive digital twin. The methodology aims to convey domain-specific representations of a crop to agronomists and growers for the operation and analysis of the vertical farm. Our proposed transformation chain enables the design-space exploration of possible concrete crop morphological states on the basis of their abstract morphology, which is inferred from a phenological model. This enables the execution of what-if analyses within the digital twin, able to prescribe desirable crop states according to the agronomist and grower who operates the system.

We applied our methodology to two case studies: a strawberry crop within a vertical farming system and a field-grown canola crop. Our approach can infer possible crop morphologies based on crop phenological models with mixed results. In the case of strawberry crops, we show that our model is able to converge to an optimal morphology, while that is not the case for canola. Further work on the calibration of heuristic transformations between models of phenology and abstract morphology and of morphological models is required to apply our methodology to different smart farming environments.

The main limitation of our methodology for digital twins is the generation of crop morphologies, which always start from initial planting. To palliate this, we aim to extend our model refinement method to take into account prior morphological states of the crop to generate results that are morphologically consistent through time. These intermediate concrete morphological states would enable the extension of our methodology to real-time monitoring digital twins, and is a step towards an autonomous digital twin.

REFERENCES

[1] 2024. Canola growth stages | Canola Encyclopedia. https://www.canolacouncil.org/canola-encyclopedia/growth-stages/

[2] Rabiya Abbasi, Pablo Martinez, and Rafiq Ahmad. 2022. The digitization of agricultural industry–a systematic literature review on agriculture 4.0. *Smart Agricultural Technology* 2 (2022), 100042.

[3] Harold Abelson and Andrea DiSessa. 1986. *Turtle geometry: The computer as a medium for exploring mathematics.* MIT press.

[4] Susannah Amundson, Dennis E Deyton, Dean A Kopsell, Walt Hitch, Ann Moore, and Carl E Sams. 2012. Optimizing plant density and production systems to maximize yield of greenhouse-grown 'Trust'Tomatoes. *HortTechnology* 22, 1 (2012), 44–48.

[5] Pascal Archambault, Istvan David, Eugene Syriani, and Houari Sahraoui. 2023. Co-Simulation For Controlled Environment Agriculture. In *2023 Annual Modeling and Simulation Conference (ANNSIM)*. SCS.

[6] Frederick Becker, Chloe MacLaren, Casparus Johannes Brink, Karin Jacobs, Marcellous Remarque le Roux, and Pieter Andreas Swanepoel. 2020. High nitrogen rates do not increase canola yield and may affect soil bacterial functioning. *Agronomy Journal* 112, 1 (2020), 523–536.

[7] Olivera Bicikliski, Fidanka Trajkova, and Ljupco Mihajlov. 2018. Morphological characteristics of some pepper genotypes (Capsicum annuum L.) grown in conventional and organic agricultural systems: comparative analysis. *Annual Research & Review in Biology* 28, 3 (2018), 1–11.

[8] Nathalie JJ Bréda. 2003. Ground-based measurements of leaf area index: a review of methods, instruments and current controversies. *Journal of experimental botany* 54, 392 (2003), 2403–2417.

[9] Frank-M Chmielewski. 2003. *Phenology and agriculture.* Springer.

[10] Mikolaj Cieslak, Nazifa Khan, Pascal Ferraro, Raju Soolanayakanahally, Stephen J Robinson, Isobel Parkin, Ian McQuillan, and Przemyslaw Prusinkiewicz. 2022. L-system models for image-based phenomics: case studies of maize and canola. *in silico Plants* 4, 1 (2022), diab039.

[11] Istvan David, Pascal Archambault, Quentin Wolak, Cong Vinh Vu, Timothé Lalonde, Kashif Riaz, Eugene Syriani, and Houari Sahraoui. 2023. Digital twins for cyber-biophysical systems: challenges and lessons learned. In *2023 ACM/IEEE 26th International Conference on Model Driven Engineering Languages and Systems (MODELS)*. IEEE, 1–12.

[12] Kalyanmoy Deb and Himanshu Jain. 2013. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE transactions on evolutionary computation* 18, 4 (2013), 577–601.

[13] Arianna Di Paola, Riccardo Valentini, and Monia Santini. 2016. An overview of available crop growth and yield models for studies and assessments in agriculture. *Journal of the Science of Food and Agriculture* 96, 3 (2016), 709–714.

[14] J Anja Dieleman, Pieter HB De Visser, Esther Meinen, Janneke G Grit, and Tom A Dueck. 2019. Integrating morphological and physiological responses of tomato plants to light quality to the crop level by 3D modeling. *Frontiers in plant science* 10 (2019), 434037.

[15] Kampker et. al. 2019. Business models for industrial smart services–the example of a digital twin for a product-service-system for potato harvesting. *Procedia Cirp* 83 (2019), 534–540.

[16] Markus Fruth and Frank Teuteberg. 2017. Digitization in maritime logistics—What is there and what is missing? *Cogent Business & Management* 4, 1 (2017), 1411066.

[17] Cláudio Gomes, Casper Thule, David Broman, Peter Gorm Larsen, and Hans Vangheluwe. 2017. Co-simulation: State of the art. *arXiv preprint arXiv:1702.00686* (2017).

[18] David Hadka et al. 2012. MOEA framework: a free and open source java framework for multiobjective optimization. *MOEAFramework-2.1-ManualFixed. pdf* (2012).

[19] Abbas Haghshenas and Yahya Emam. 2019. Image-based tracking of ripening in wheat cultivar mixtures: A quantifying approach parallel to the conventional phenology. *Computers and Electronics in Agriculture* 156 (2019), 318–333.

[20] Qi Jing, Jiali Shang, Budong Qian, Gerrit Hoogenboom, Ted Huffman, Jiangui Liu, Bao-Luo Ma, Xiaoyuan Geng, Xianfeng Jiao, John Kovacs, et al. 2016. Evaluation of the CSM-CROPGRO-Canola Model for simulating canola growth and yield at West Nipissing in eastern Canada. *Agronomy Journal* 108, 2 (2016), 575–584.

[21] James W Jones, Ehud Dayan, LH Allen, Herman Van Keulen, and Hugo Challa. 1991. A dynamic tomato growth and yield model (TOMGRO). *Transactions of the ASAE* 34, 2 (1991), 663–0672.

[22] James W Jones, Gerrit Hoogenboom, Cheryl H Porter, Ken J Boote, William D Batchelor, LA Hunt, Paul W Wilkens, Upendra Singh, Arjan J Gijsman, and Joe T Ritchie. 2003. The DSSAT cropping system model. *European journal of agronomy* 18, 3-4 (2003), 235–265.

[23] Radoslaw Karwowski, Przemyslaw Prusinkiewicz, et al. 2004. The L-system-based plant-modeling environment L-studio 4.0. In *Proceedings of the 4th international workshop on functional-structural plant models.* UMR AMAP Montpellier, France, 403–405.

[24] Sergei Lembinen, Mikolaj Cieslak, Teng Zhang, Kathryn Mackenzie, Paula Elomaa, Przemyslaw Prusinkiewicz, and Timo Hytönen. 2023. Diversity of woodland strawberry inflorescences arises from heterochrony regulated by TERMINAL FLOWER 1 and FLOWERING LOCUS T. *The Plant Cell* 35, 6 (2023), 2079–2094.

[25] Zhen-qi LIAO, Jing ZHENG, Jun-liang FAN, Sheng-zhao PEI, Yu-long DAI, Fucang ZHANG, and Zhi-jun LI. 2023. Novel models for simulating maize growth based on thermal time and photothermal units: Applications under various mulching practices. *Journal of Integrative Agriculture* 22, 5 (2023), 1381–1395.

[26] Aristid Lindenmayer. 1968. Mathematical models for cellular interactions in development I. Filaments with one-sided inputs. *Journal of theoretical biology* 18, 3 (1968), 280–299.

[27] Chenyao Lu, Li Deng, and Minrui Fei. 2015. An improved visualization modelling method of greenhouse tomato plants based on L-system. In *2015 Chinese Automation Congress (CAC)*. 480–485. https://doi.org/10.1109/CAC.2015.7382548

[28] Newton Z Lupwayi, K Neil Harker, John T O'Donovan, T Kelly Turkington, Robert E Blackshaw, Linda M Hall, Christian J Willenborg, Yantai Gan, Guy P Lafond, William E May, et al. 2015. Relating soil microbial properties to yields of no-till canola on the Canadian prairies. *European Journal of Agronomy* 62 (2015), 110–119.

[29] J-P Matteau, SJ Gumiere, J Gallichand, G Létourneau, L Khiari, M-O Gasser, and A Michaud. 2019. Coupling of a nitrate production model with HYDRUS to predict nitrate leaching. *Agricultural Water Management* 213 (2019), 616–626.

[30] Suresh Neethirajan and Bas Kemp. 2021. Digital twins in livestock farming. *Animals* 11, 4 (2021), 1008.

[31] Przemyslaw Prusinkiewicz. 1986. Graphical applications of L-systems. In *Proceedings of graphics interface*, Vol. 86. 247–253.

[32] S Ragaveena, A Shirly Edward, and U Surendran. 2021. Smart controlled environment agriculture methods: A holistic review. *Reviews in Environmental Science and Bio/Technology* 20, 4 (2021), 887–913.

[33] Andreas Schumacher, Wilfried Sihn, and Selim Erol. 2016. Automation, digitization and digitalization and their implications for manufacturing processes. In *Innovation and Sustainability Conference Bukarest*. Elsevier Amsterdam, The Netherlands, 1–5.

[34] Anita Sønsteby, Nina Opstad, and Ola M Heide. 2013. Environmental manipulation for establishing high yield potential of strawberry forcing plants. *Scientia Horticulturae* 157 (2013), 65–73.

[35] Kalyanasundaram Subramanian. 2020. Digital twin for drug discovery and development—the virtual liver. *Journal of the Indian Institute of Science* 100, 4 (2020), 653–662.

[36] Rosa Arnaldo Valdés, Víctor Fernando Gómez Comendador, Alvaro Rodriguez Sanz, and Javier Perez Castán. 2018. Aviation 4.0: more safety through automation and digitization. In *Aircraft technology*. IntechOpen.

[37] CN Verdouw and Jan Willem Kruize. 2017. Digital twins in farm management: illustrations from the FIWARE accelerators SmartAgriFood and Fractals. In *Proceedings of the 7th Asian-Australasian Conference on Precision Agriculture Digital, Hamilton, New Zealand*. 16–18.

[38] Cor Verdouw, Bedir Tekinerdogan, Adrie Beulens, and Sjaak Wolfert. 2021. Digital twins in smart farming. *Agricultural Systems* 189 (2021), 103046.

[39] J Vos, Jochem B Evers, G Hh Buck-Sorlin, Bruno Andrieu, Michaël Chelle, and Pieter HB De Visser. 2010. Functional–structural plant modelling: a new versatile tool in crop science. *Journal of experimental Botany* 61, 8 (2010), 2101–2115.

[40] Qi Yang, Liangsheng Shi, Jingye Han, Jin Yu, and Kai Huang. 2020. A near real-time deep learning approach for detecting rice phenology based on UAV images. *Agricultural and Forest Meteorology* 287 (2020), 107938.

[41] Xinyou Yin, Paul C Struik, and Jan Goudriaan. 2021. On the needs for combining physiological principles and mathematics to improve crop models. *Field Crops Research* 271 (2021), 108254.

[42] H Zekki, C Gary, A Gosselin, and L Gauthier. 1999. Validation of a Photosynthesis Model through the Use of the CO 2 Balance of a Greenhouse Tomato Canopy. *Annals of Botany* 84, 5 (1999), 591–598.

[43] Yu Zhang, Poching Teng, Mitsuko Aono, Yo Shimizu, Fumiki Hosoi, and Kenji Omasa. 2018. 3D monitoring for plant growth parameters in field with a single camera by multi-view approach. *Journal of agricultural meteorology* 74, 4 (2018), 129–139.