Searcher-Shoot: a Reinforcement Learning approach to understand climbing plant behaviour

Lucia Nasti[1], Giacomo Vecchiato[1], Tom Hattermann[2], Patrick Heuret[2], Nicholas P. Rowe[2], Michele Palladino[1,3], and Pierangelo Marcati[1]

[1]Gran Sasso Science Institute, L'Aquila, Italy
[2]AMAP, Univ Montpellier, CIRAD, CNRS, INRAe, IRD, Montpellier, France
[3]DISIM, Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, Via Vetoio - 67100 L'Aquila, Italy

Abstract

Plants' structure is the result of constant evolution towards the adaptation to the surrounding environment. From this perspective, our goal is to investigate the mass and radius distribution of a peculiar plant organ, namely the *searcher shoot*, by providing a Reinforcement Learning (RL) environment, that we call Searcher-Shoot, which considers the mechanics due to the mass of the shoot and leaves. We uphold the theory that plants can maximize their length, avoiding a maximal stress threshold. To do this, we explore whether the mass distribution along the stem is efficient, formulating this hypothesis as a Markov Decision Process (MDP). By exploiting this strategy, we are able to mimic and thus understand the plant's behavior, finding that shoots decrease their diameters smoothly in order to efficiently distribute the mass. The strong agreement between our results and the experimental data allows us to remark on the strength of our approach in the analysis of biological systems traits.

1 Introduction

Plants are living organisms coordinating a complex network of internal, e.g., nutrient concentration, and external signals, e.g., light and soil resources. As a result, plant growth is a delicate balance among different factors involving environmental and physiological conditions [1].

Despite their sessile life, plants can move and react to external stimuli to look for nutrients, avoiding obstacles and dangerous conditions [2]. Differently from the animal kingdom, plants do not perform these movements only through "active" reversible actions, but mainly by expanding their organs [3]. This is a fundamental issue about plants: they move by growth, so how do they grow? What characterises their shape? To investigate in this direction, we need to

place plants in their context. Plants actually leave in a complex environment with a limited amount of resources. These resources are shared between plants of the same species as well as plants of different species. In this context, the efficient use of such resources can be crucial for plant subsistence. The concept of efficiency may be the key for understanding how plants interact with the environment and develop their organs. This gives a different perspective to plant modelling. Indeed, this point of view enriches a model by adding to the mathematical description of the biological system a possible interpretation. Concerning the shape of the plants and the subject matter of this work, we consider the mechanical aspect of a plant organ. The possibility of mathematically studying the mechanics of a structure and understanding the extent of its physical limits has fascinated scientists since the time of Galileo (see in particular the work "two new sciences"). Specifically on plants, there are studies on critical lengths (see for instance [4, 5, 6]), on the distribution of roots and branches [7] or on how a root penetrates the soil [8]. The optimisation paradigm can be applied effectively in biological contexts [9, 10] and is what also guides this work.

In this work, we focus on a particular climbing plant species, the *Condylocarpon Guianense* Desf.. This plant species is a liana widely found in the flora of French Guyana, which twines around the branches and the trunks of its hosts in order to reach the canopy. Several studies on its structure, see [11, 12, 13] for instance, have revealed that in different growth stages, it changes the thickness and the nature of the layers that form its stem and consequently it changes its stiffness. More specifically, the plant is stiffer when and where is developing a self-supporting state, while it displays a less dense material and a thicker compliant cortex when attached to a support. Such a wide adaptability of the *C. guianense* to the surrounding environment suggests that it is following a paradigm of efficiency, making it a suited subject for our study. In particular, we want to support the thesis that the self-supporting organs of the *C. guianense*, called searcher shoots, are trying to maximize their length, avoiding a maximal stress threshold. This idea is motivated by the fact that the searcher shoots are organs whose purpose is to search for external support and attach to it. Hence, they need to be as long as possible, in order to explore the surrounding environment, but at the same time, they have to sustain their own weight. To investigate this specific behavior and prove our conjecture, we combine mechanical modelization and Reinforcement Learning. Specifically, to prove the thesis that climbing plants optimize the mass distribution in their self-supporting organs, we developed a Reinforcement Learning environment, which we called Searcher-Shoot, to study the radius along the climbing plant shoot. At the base of this environment, we considered two planar mechanical models: (ML) and (MM), which give us information about the development of the curvature and the mass along the shoot, considering the leaves or not, respectively.

We compare the simulated radius with the experimental data and observe that the optimal policy is able to reproduce the decreasing behavior that characterizes the radius of the sample in exam. This result suggests that, at least for

the *C. guianense*, the mass distribution along the searcher shoot is optimized in function of the length.

The application of Artificial Intelligence (AI) to many biological problems is increasing rapidly in the analysis of plant morphology, growth, and development, or the understanding of their changing environment in conjunction with agriculture [14]. In particular, Machine Learning (ML) is playing a conspicuous role in developing predictive models in complex plant biological systems [15] whenever possible the integration and the analysis of multidimensional omics data [16]. Moreover, as described in [17], with inadequate data, AI/ML applications perform poorly. When Supervised and Unsupervised methods are not able to generate a direct linear or non-linear mapping among the raw data, Reinforcement Learning stands up for being a valid alternative method [18, 19] and, as underlined in [20], it is emerging as a robust and reliable tool to face out real-world problems concerning biological systems. Example of this application are available in synthetic biology [21], metabolic engineering [22], chemical reaction network [23], and plant biology [24].

2 Results

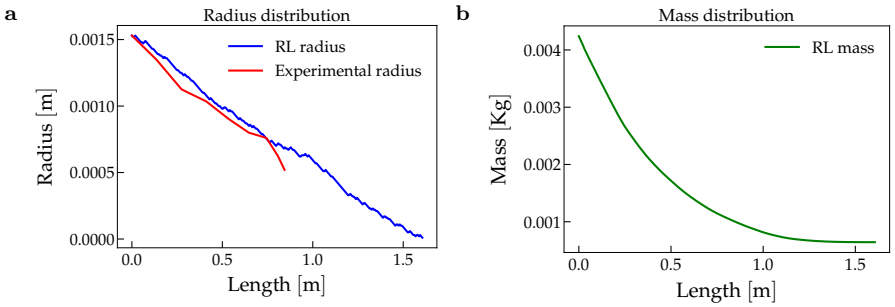
{sec:results}

We develop the Searcher-Shoot environment in `Python`. Precisely, we employ the `OpenAIsGym` [25] and `Stable-Baselines3` (SB3) [26] libraries, two open-source frameworks implementing several commonly used model-free deep RL algorithms. In particular, from SB3, we import the PPO algorithm [27]. For the mathematical modelling, we develop two models for the searcher shoot: (i) a model with the mechanics, but without the leaves (MM) and (ii) a model with both the mechanics and the leaves (MM). We perform all the simulations with the discount factor $\gamma = 0.99$ (see Section 5.1.1 in the Supplementary Material) and we train the models setting the number of episodes to 1 million.

In Figure 1, we plot the results of our simulations and we show the radius (Figure 1a) and the mass distributions (Figure 1b). We find that the radius decreases at each step, i.e. the agent chooses the actions leading to a smaller radius.

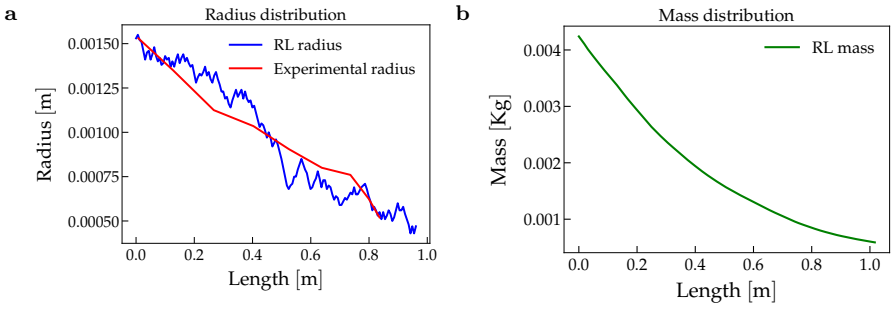
We compare our simulated scenario with five samples [28, 29]. As we can notice in Figure 1a, comparing the obtained radius distribution (in blue) with the experimental radius of sample S2 (in red), the relative error is 8.55% in the segment $[0, 0.868]$, which represents the length of the experimental sample. In the Supplementary Material, we include the simulation results and the comparison with all the other samples.

As the last case study, we consider the ML model, where, in addition to the mechanics features, we model also the mass of the leaves along the shoot. In Figure 2, we present the results of our simulations. Both the radius (Figure 2a) and the mass distributions (Figure 2b) are consistent with the results of the preceding model. Again, we compare the experimental radius (in red) of sample S2 with the one we acquire by RL, and, by computing the relative error, we find that the discrepancy between the two radii is 10.28% in the segment $[0, 0.868]$.



{fig: plot_MM}

Figure 1: Radius and mass distribution in the MM model. We show the radius and, consequently, the mass distribution obtained in the MM model after the training. Specifically, in Figure 1a, we compare the experimental radius (red) to the one obtained by our simulation (blue). Computing the relative error, we find that the discrepancy between the two radii is at most 8.55% in the segment $[0, 0.868]$. The value 0.868 represents the total length in meters of the shoot sample. In Figure 1b, we show the simulated mass. The mass decreases smoothly and is approximately zero at the tip.



{fig: plot_ML}

Figure 2: Radius and mass distribution in the ML model. We show the radius and, consequently, the mass distribution obtained in the ML model after the training. Specifically, in Figure 2a, we compare the experimental radius (red) to the one obtained by our simulation (blue). Computing the relative error, we find that the discrepancy between the two radii is at most 10.28% in the segment $[0, 0.868]$. The value 0.868 represents the total length in meters of the shoot sample. In Figure 2b, we show the simulated mass. The mass decreases smoothly and is approximately zero at the tip.

3 Discussion

{sec:discussion}

The relative error between the experimental radius and the simulated samples is less than 15% for all the samples and in both cases with or without leaves (see Tables 1-2). Such a small relative error suggests that the optimal policy successfully reproduces the radial profile of the samples. Going into more detail, in the (MM) case with generic c_2 and ψ_0, the error is between the 12.9% and the 16.8% (see Table 1), with the exception of sample S1. This error decreases between the 7.8% and the 9.8% when the constants are sample-specific. In particular, we observe that the coefficient of variation of c_2 is about 35% (see Table 3), while the coefficient of variation of ψ_0 is much greater since it is about 74%. This might imply that the environmental conditions have a relevant impact on the initial curvature of the shoot, while the stress threshold might be characteristic of the plant species. The difference between the length of the simulation and the experimental one, displayed in figure 1a can be explained by the approximation in the mass and density measures. Regarding the simulations with the leaves, i.e. the (ML) model, the relative error gets worse if compared to the (MM) model (see Table 2). This drop in the accuracy of the model can be due to the fact that the agent has no control over the distribution of the leaves. Moreover, the constants ψ_0 and c_2 are estimated on a model based on the optimisation of the mass of the main stem. Hence, effective implementation of the leaves would require a more accurate model; nevertheless, even in our approximation, we get an error between the 7.4% and the 14.1%. This result suggests that the aim of the plant during the self-supporting stage is the optimization of the main stem's mass. This is in line with the application of the behavioural ecology theoretical framework to plants (see [30] for a survey on the behavioural ecology of climbing plants). In other words, according to this theory, plants have the capability to place their organs in accordance with optimal economic models. For instance, depending on the external supports in the surrounding environment, the predation risk and the energetic stress, a shoot may delay leaf expansion or have short internodes. In our case, the plant has a limited amount of mass and can develop limited internal stress. A longer stem means, on one hand, a greater exploration capability, on the other hand, a greater risk, since the structure is more fragile. Our RL environment gives a quantitative answer to the trade-off that the plant has to face, in the specific case of the *C. guianense* in its natural habitat. However, the generality of the equations at the base makes this environment suitable for application to other plant species.

Building an RL environment, i.e., Searcher-Shoot, is a compelling strategy to understand complex systems biology behaviors, especially in the lack of data which are crucial in the application of Machine Learning. The RL approach we exploit in this work has led to enthusiastic results that make us even more hopeful for future developments. In particular, we plan to explore other specific biological behaviors, such as root competition. Indeed, different plant species may compete (or collaborate) to uptake soil's nutrients. In this framework, biomass production and soil occupation are suitable variables for an RL-based study. Moreover, in line with this work, we plan to build a more sophisticated

model, which can consider the curvature time development in addition to the optimal mass distribution. Indeed, plant movement is - partially - determined by the response to external signals. This response can be optimized to maximize (or minimize) a reward based on the signals themselves.

Sample	Relative Error %	Relative Error % with EP
S1	8.6%	13.6%
S2	-	8.5%
S3	16.8%	8.2%
S4	16.2%	7.8%
S5	12.9%	9.8%

{tab: comparison with samp

Table 1: The table reports the Relative Error (RE) with and without estimated parameters (EP). To show the strength of our results, we compute the Relative Errors (RE) by comparing our results with the experimental radii of the samples [28, 29]. To understand the influence of two crucial parameters, c_2 and ψ_0, which are the curvature threshold and the initial curvature, respectively, we simulate our MM model by using their estimated values and an average value.

Sample	Relative Error % with EP
S1	7.4%
S2	10.2%
S3	11.4%
S4	14.1%
S5	11.8%

{tab: comparison with samp

Table 2: The table reports the Relative Error (RE) with estimated parameters (EP), by considering mass leaves. To show the strength of our results, we compute the Relative Errors (RE) by comparing our results (ML model) with the experimental radii of the samples [28, 29]. To understand the influence of two crucial parameters, c_2 and ψ_0, which are the curvature threshold and the initial curvature, respectively, we simulate our ML model by using their estimated values and an average value.

Sample	ψ_0	c_2
S1	$-5.9e-12$	$8.7e-4$
S2	$-3.7e-12$	$3.6e-4$
S3	$-7.8e-12$	$4.8e-4$
S4	$-2.3e-11$	$6.1e-4$
S5	$-1.1e-11$	$4.9e-4$
Average	$-1e-11$	$5.6e-4$
Coeff. of Variation	73.61%	34.4%

{tab: fundamental par and

Table 3: The table reports the values of ψ_0, c_2, with average and coefficient of variation. The values of ψ_0 and c_2 are estimated utilizing the methods described in [31].

4 Methods

{sec:methods}

4.1 Notations for the mechanics of a planar elastic rod

Let e_1, e_2, e_3 a basis of orthonormal vectors for $\mathbb{R}^3$. We assume that the searcher shoot behaves like an inextensible and unshearable elastic rod [32] confined in the plane spanned by the vectors e_1 and e_2. The centerline of this rod lies on the curve $\Gamma \subset \text{span}\{e_1, e_2\}$. We parametrize the curve Γ with its arc length s, so that if the length of the curve is L, we have $s \in [0, L]$. We denote with $\underline{\Gamma}(s)$ the position in the plane of the point on Γ whose arc length is s. With this parametrisation, $\partial_s\Gamma(s)$ represents the normal tangent vector to Γ at the point $\underline{\Gamma}(s)$. We denote with $\theta(s)$ the angle between $\partial_s\Gamma(s)$ and the vector e_2 . Consequently, $\theta'(s)$ is the curvature of Γ at the point $\underline{\Gamma}(s)$.

We refer to Γ as the *current* configuration, which corresponds to the actual shape of the rod when subject to external physical forces. To study the effects of gravity acting on that rod, we need to consider the *intrinsic* configuration, denoted as $\hat{\Gamma}$, which corresponds to the geometric curve assumed by the rod when there are no external forces acting on it. Since the rod is inextensible, the arc length parameter of $\hat{\Gamma}$ and Γ is the same $s \in [0, L]$.

The difference between the curvature $\partial_s\hat{\theta}(s)$ of the intrinsic configuration and the curvature $\partial_s\theta(s)$ of the current configuration is proportional to the resultant moment (of force) $m(s)$ acting at the point $\underline{\Gamma}(s)$ (and directed along $\underline{e}_3$) through the Euler-Bernoulli formula:

$$m(s) = -E(s)I(s)(\partial_s\theta(s) - \partial_s\hat{\theta}(s)). \tag{1}$$ {eq:euler-bernoulli}

This relation holds because we are considering unshearable rods. With E we are denoting the Young's modulus, which expresses the stiffness of the material, and with I the second moment of area of the cross-section (with respect to e_3). We assume that the rod is in elastic equilibrium, that is, all the internal forces and moments are in balance with the external forces and moments. In this framework, considering equation (1) and the gravity force as the only external force acting on the rod, we can write the following differential equation (see for instance [33])

$$\partial_s(EI(\partial_s\hat{\theta} - \partial_s\theta))(s) = \sin(\theta(s))g\int_s^L \rho_3(s')A(s')ds'. \tag{2}$$ {eq:mech0}

In this equation, g represents the gravity acceleration constant, $\rho_3(s)$ the volume density of the shoot/rod at the point $\underline{\Gamma}(s)$ and $A(s)$ the cross-section area at that point.

We are now interested in computing the internal bending stresses. We know that the internal moment m is generated by the deflection from the intrinsic configuration $\hat{\Gamma}$. Indeed, this deflection generates an internal pressure called *stress*, that we denote with σ, and a deformation ε of each element of the rod, called *strain* Stresses and strains vary according to the position $\underline{\Gamma}(s)$ on the rod, and depend also on the position on the cross-section. Since the rod is

ushearable, the cross-section is always orthogonal to the tangent vector $\partial_s\Gamma$. To describe the position of a generic point on the cross-section at $\underline{\Gamma}(s)$, we name

$$\beta(s) = \frac{\partial_s^2\Gamma(s)}{|\partial_s^2\Gamma(s)|} \in \text{span}\{e_1, e_2\}$$

the normal vector, which is orthogonal to $\partial_s\Gamma(s)$, and we consider the binormal vector

$$\tau = \frac{\Gamma' \times \beta}{|\Gamma' \times \beta|},$$

Then, the cross-section at the point $\underline{\Gamma}(s)$ is a subset $C(s)$ of the plane $\text{span}\{\beta(s), \tau\}$ with the origin on the centerline. We define

$$C(s,z) = \{w \in \mathbb{R} \,:\, \underline{\Gamma}(s) + z\beta(s) + w\tau \in C(s)\}.$$

In this framework, the maximal bending stress at $\Gamma(s)$ results to be [34]

$$\begin{aligned} \sigma_m(s) &= \max\{|w| \,:\, C(s,w) \neq \emptyset\} \cdot \frac{m(s)}{I(s)} \\ &= \max\{|w| \,:\, C(s,w) \neq \emptyset\} \cdot E(s)|\partial_s\theta(s) - \partial_s\hat{\theta}(s)|. \end{aligned} \tag{3}$$

{eq:max_stress1}

4.2 Formulation of the models

We assume that the searcher shoot has a circular cross-section with radius r and that the Young's modulus E is constant all along the shoot. So, we have

$$\begin{aligned} A(s) &= \pi r^2(s) \\ I(s) &= \frac{\pi}{4} r^4(s) \\ r(s) &= \max\{|w| \,:\, C(s,w) \neq \emptyset\} \end{aligned}$$

and we name

$$u(s) = r^2(s),\ \psi(s) = u^2(s)(\partial_s\hat{\theta}(s) - \partial_s\theta(s)),\ \mu(s) = \int_s^L \pi\rho_3(s')u(s')ds'$$

So, equation (2) can be rewritten as

$$\begin{cases} \partial_s\psi(s) = c_1 \sin(\theta(s))\mu(s) \\ \partial_s\theta(s) = \partial_s\hat{\theta}(s) - \frac{\psi(s)}{u^2(s)} \\ \partial_s\mu = -\pi\rho_3(s)u(s) \end{cases} \tag{4}$$

{eq:mech1}

with

$$c_1 = \frac{4g}{\pi E}.$$

These equations hold for a.e. $s \in [0, L]$. At the boundary of this domain, we assume (i) to know the angle at the base of the shoot $\theta(0) = \theta_0$; (ii) that at the

tip of the shoot, there is not any external weight so that the intrinsic curvature equals the current curvature. Using the functions defined above, this means $\psi(L) = 0$ and $\mu(L) = 0$; (iii) the mass M of the whole shoot is known, so that $\mu(0) = M$.

The effectiveness of the self-sustaining behavior of the shoot can be quantitatively evaluated considering a threshold for the maximal internal stresses. In other words, we would like to find a distribution of the mass such that the maximal stress at each point of the shoot $\sigma_m(s)$ does not cross some fixed value $\bar{\sigma}$. Employing equation (3), this means that any solution of system (4) which satisfies the boundary conditions above discussed, has also to satisfy the condition

$$|\psi(s)| \leq c_2 u^{3/2}(s) \text{ for every } s \in [0, L] \tag{5}$$ {eq: curvature_condition}

with

$$c_2 = \frac{\bar{\sigma}}{E}.$$

We group all these considerations into the following problems.

Problem of shoot growth with Mechanics (MM). We want to find the maximal length L of the shoot for which there exists at each point a radius $r(s)$ such that the following system is satisfied

$$\begin{cases} \partial_s \psi(s) = c_1 \sin(\theta(s))\mu(s) \\ \partial_s \theta(s) = \partial_s \hat{\theta}(s) - \frac{\psi(s)}{u^2(s)} \\ \partial_s \mu = -\pi \rho_3(s) u(s) \\ \psi(L) = 0 \\ \theta(0) = \theta_0 \\ \mu(0) = M,\ \mu(L) = 0 \\ |\psi(s)| \leq c_2 u^{3/2}(s) \end{cases} . \tag{MM}$$ {eq: mechMM}

Problem of shoot growth with Mechanics and Leaves (ML). In problem (MM) we consider just the weight of the main stem. However, in most of climbing plant species, a relevant part of the total biomass is due to the leaves. We assume that the leaves are not uniformly distributed along the shoot. On the contrary, we assume that they are located at intervals equally spaced. Moreover, we assume that the mass m_{lm} of a single leaf at the point $\underline{\Gamma}(s)$ depends just on the shoot radius $r(s)$ at $\underline{\Gamma}(s)$. Let d_{lm} the distance between two leaves locations and n_{lm} the (fixed) number of leaves at each location. Then, we name

$$s_i = i \times d_{\text{lm}} \text{ for } i = 1, ...N_{\text{lm}},$$

where N_{lm} is the total number of leaves locations. Now, we want to compute how the weight of the leaves affects the shoot at the point $\underline{\Gamma}(s)$. To achieve this, we subtract from the total leaves mass M_{lm} the mass of the leaves in the shoot portion between the base and $\underline{\Gamma}(s)$. We define

$$q_{\text{lm}}(s) = \left\lfloor \frac{s}{d_{\text{lm}}} \right\rfloor$$

and

$$\mathrm{RES}_{\mathrm{lm}}(s) = M_{\mathrm{lm}} - n_{\mathrm{lm}} \sum_{i=0}^{q_{\mathrm{lm}}(s)} m_{\mathrm{lm}}(r(s_i)),$$

where the operation $\lfloor x \rfloor$ is the greatest integer lower or equal than x. Then, to take the leaves into account, we compute the gravity force acting at the point $\underline{\Gamma}(s)$:

$$-g \left[\int_s^L \rho_3(s') A(s') ds' + \mathrm{RES}_{\mathrm{lm}}(s) \right] e_2.$$

This leads to another problem of length maximization. Like for the (MM) case, we want to find the maximal length L of the shoot for which there exists at each point a radius $r(s)$ such that

$$\begin{cases} \partial_s \psi(s) = c_1 \sin(\theta(s)) \left(\mu(s) + \mathrm{RES}_{\mathrm{lm}}(s)\right) \\ \partial_s \theta(s) = \partial_s \hat{\theta}(s) - \frac{\psi(s)}{u^2(s)} \\ \partial_s \mu = -\pi \rho_3(s) u(s) \\ \psi(L) = 0 \\ \theta(0) = \theta_0 \\ \mu(0) = M,\ \mu(L) = 0 \\ |\psi(s)| \leq c_2 u^{3/2}(s) \end{cases} . \qquad \text{(ML)} \quad \texttt{\{eq: mechML\}}$$

4.3 Models implementation and parameters

{SubSec: Model impl and Pa

Parameter	Description	Source	Model
g	Gravity acceleration constant	[28, 29]	MM, ML
E	Young's modulus	[28, 29]	MM, ML
M	Main stem freshmass	[28, 29]	MM, ML
c_2	Curvature threshold	[31]	MM, ML
ψ_0	Initial curvature	[31]	MM, ML
$(a_{\mathrm{vd}}, b_{\mathrm{vd}}, c_{\mathrm{vd}})$	Parameters for volume density fitting	[28, 29]	MM, ML
$(a_{\mathrm{lm}}, b_{\mathrm{lm}}, c_{\mathrm{lm}})$	Parameters for leaves mass fitting	[28, 29]	ML

{tab: parameters}

Table 4: The table reports the parameters of the model. In the case of the parameters resulting from the fitting procedure, we report the source of the data on which functions (6)-(7) are fitted. The column **Model** reports where we use the parameter: MM (Mechanical Model), and ML (Mechanical model with Leaves).

To begin with, we implement the System of equations (MM), where we consider *stress* and *strain* as factors responsible for its shaping, together with *gravity*, which acts as an external force on the plant's structure, affecting its curvature. Moreover, the material *density* is a function of the radius r, defined as follows:

$$\rho_3 = c_{\mathrm{vd}} + b_{\mathrm{vd}} \cdot r + a_{\mathrm{vd}} \cdot r^2. \qquad (6) \quad \texttt{\{eq: mat\_density\}}$$

We use Algorithm 1 to clarify the approach we implemented. The values of the constants c_2 and ψ_0 are estimated in two ways, leading to two groups of simulations. In the first group, c_2 and ψ_0 are the same for all the samples, while in the second group, they are estimated specifically for each sample by utilizing the method described in [31].

Algorithm 1 Algorithm of Shoot growth with mechanics {alg: 2}

Require: $M > 0$
  $r > 0$
  $(\psi, \theta, \mu) \leftarrow (\psi_0, \pi/2, M)$
  while $\mu \geq 0$ **and** $\psi \leq c_2 \times r^3$ **do**
    $r \leftarrow r$ **or** r_{S} **or** r_{L}
    $\rho \leftarrow c_{\mathrm{vd}} + b_{\mathrm{vd}} \cdot r + a_{\mathrm{vd}} \cdot r^2$
    $(\psi, \theta, \mu) \leftarrow$ Solve System (MM) with $u = r^2$, $\rho_3 \equiv \rho$,
      $\psi(0) = \psi$, $\theta(0) = \theta$, $\mu(0) = \mu$
    reward = 1
  end while
  reward = 0

Successively, we add the leaves' mass contribution, which affects the plant's weight remarkably. We implement the System of equations (ML), where we can notice that the effects of the leaves are visible on the curvature and, then, in the formulation of the equation of ψ. As for the material density, the mass of a single leaf depends on the radius r accordingly to the following relation:

$$m_{\mathrm{ml}} = c_{\mathrm{lm}} + b_{\mathrm{lm}} \cdot r + a_{\mathrm{lm}} \cdot r^2. \tag{7}$$ {eq: leaf_mass}

In Algorithm 2, we clarify how we implement our model in the RL context.

4.4 Reinforcement Learning Framework {SubSec: RL_framework}

We define the Searcher-Shoot environment as an MDP problem (see Section 5.1 in the Supplementary Material). Here, the agent is the plant's shoot, which aims to grow as long as possible and to reach this goal, it has to select the optimal configuration of diameters. Specifically, the agent learns how to complete the task (i.e., the mass distribution) in the highest number of steps, choosing radius values does not generate internal stresses over a fixed threshold. The fundamental elements of the framework are:

- **State and Observations.** At each step, the agent, in the current state, can observe the mass, the radius, and the curvature before the next move.
- **Actions.** In this framework, the action space is discrete. In the MM and ML models, at each time step, the agent can select one action among eleven options: it can leave the radius value unchanged or it can increase (or decrease) the radius of a certain quantity ($\pm 1 \cdot 10^{-5}$, $\pm 2 \cdot 10^{-5}$, $\pm 3 \cdot 10^{-5}$, $\pm 4 \cdot 10^{-5}$, $\pm 5 \cdot 10^{-5}$). Of course, this selection will influence the mass

Algorithm 2 Algorithm of Shoot growth with mechanics and leaves {alg: 3}

Require: $M > 0$, $M_{\text{lm}} \geq 0$, $d_{\text{lm}} > 0$, $d_{\text{lm}} > h > 0$
  $r > 0$
  $d_{\text{base}} \leftarrow 0$
  $(\psi, \theta, \mu) \leftarrow (\psi_0, \pi/2, M)$
  RES $\leftarrow M_{\text{lm}}$
  while $\mu \geq 0$ **and** $\psi \leq c_2 \times r^3$ **do**
    $r \leftarrow r$ **or** r_{S} **or** r_{L}
    $\rho \leftarrow c_{\text{vd}} + b_{\text{vd}} \cdot r + a_{\text{vd}} \cdot r^2$
    if $d_{\text{base}} \,\%\, d_{\text{lm}} < h$ **and** RES > 0 **then**
      RES $\leftarrow$ RES $- n_{\text{lm}} \cdot (c_{\text{lm}} + b_{\text{lm}} \cdot r + a_{\text{lm}} \cdot r^2)$
    end if
    $(\psi, \theta, \mu) \leftarrow$ Solve System (ML) with $u = r^2$, $\rho_3 \equiv \rho$, $\text{RES}_{\text{lm}} = \text{RES}$
      $\psi(0) = \psi$, $\theta(0) = \theta$, $\mu(0) = \mu$
    $d_{\text{base}} \leftarrow d_{\text{base}} + h$
    reward $= 1$
  end while
  reward $= 0$

distribution: intuitively, the larger the radius value, the larger the mass allocated in the next step. In the SM model, where the mechanic description is neglected, the agent can select one action among three possibilities: leave the radius unchanged, pick a random smaller radius, or pick a random larger radius than the previous one.

- **Reward.** Every time the agent moves to the next step, it will receive positive feedback equal to +1. Whether the move ends with the total mass equivalent to or less than 0, the reward is 0 in all the models (SM, MM, ML). In addition, in the MM and ML models, the reward is 0 if the condition on the curvature (5) is violated.

- **Episode and Reset.** The episode does not have a fixed term. Instead, it ends whether the mass becomes zero or negative (in all the models) or the picked radius causes the curvature to violate condition (5), as in Algorithm 1 and 2. As underlined in Section 4.3, this last condition is present only in the MM and ML models because we implement the mechanical features. Then, we set the system parameters and the observation space to their initial values.

5 Supplementary Material

5.1 Reinforcement Learning

{SubSec: RL}

Supervised, Unsupervised, and Reinforcement learnings are three paradigms of Machine Learning. While the first two approaches require examples and data (labeled or not) to extract information, RL models learn from interactions between an agent and an environment.

Specifically, the agent has an explicit goal, and to reach it, it can perform actions that influence the state of the environment, which has a set of immutable rules. The agent uses these interactions to adjust its behavior to complete its task. Beyond these, four sub-elements characterize the RL system [35]: policy, reward, value function and model. The meaning of these sub-elements is the following:

- The *policy* represents how the agent chooses the action based on the current state;
- The *reward* is the goal of the RL problem. It is a feedback signal defining the good and bad events for the agent;
- The *value function* is the total amount of reward an agent can expect to accumulate over the future, starting from a specific state. It helps the agent to understand the long-term consequences of actions;
- The *model* of the environment is an optional representation of the environment, which allows the planning of possible future situations before their experience.

We can express an RL problem using the mathematical formalism of the Markov Decision Process (MDP), used to study the control of sequential decisions that can influence states and future rewards. An MPD is a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma \rangle$ where $\mathcal{S}$ and $\mathcal{A}$ are the state and the action space, respectively; $\mathcal{R}$ is the reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$, representing the immediate reward; and, $\mathcal{P}$ is the transition function $\mathcal{P} : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$, and so the probability to move from a state to another having chosen an action. Finally, γ represents the discount factor, namely the chance for the agent to choose between an instant (short-sighted agent) and a future reward (farsighted agent).

Briefly, at each time step t, in a state $s_t \in \mathcal{S}$, an agent interacts with the environment and chooses an action $a_t \in \mathcal{A}$, which leads to a reward $r_{t+1} = \mathcal{R}(s_t, a_t)$ and a transition to a new state $s_{t+1} \in \mathcal{S}$. The probability to reach the state s_{t+1} is given by $\mathcal{P}(s_t, s_{t+1}, a_t)$.The choice of the action a_t relies on the policy adopted by the agent. Formally, a policy is a function $\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ which gives the probability of choosing an action $a \in \mathcal{A}$ knowing that the agent is in the state $s \in \mathcal{S}$. The goal is to maximize the total reward, learning a policy.

5.1.1 Proximal Policy Optimization

{SubSec: PPO Algorithm}

We can divide the RL algorithms into two main categories: *Model-Based*, in which a system uses a predictive model of the world to choose the best action (i.e., the algorithm exploits the knowledge of an MDP); *Model-free*, in which the agent learns a value function or a policy by interacting with the environment [36].

Having a model means relying on a function that predicts the future states and rewards, allowing the agent to plan by thinking ahead and explicitly deciding between its options. Instead, the absence of it means that the agent uses only the current state and its experience to learn.

In our work, we exploit the model-free approach in the form of the Proximal Policy Optimization (PPO) algorithm, introduced by Schulman et al. [27] in 2017. PPO is a policy gradient, on-policy algorithm, meaning that the algorithm is searching for an approximation of the best policy through a parameter θ, and each step for the upgrade of the policy π_θ relies on a sampling based on π_θ itself. There are two primary variants: PPO-Penalty and PPO-Clip, which we use.

In the PPO-Clip approach, the update of the parameter θ_k to θ_{k+1} relies on the maximisation of the following surrogate objective:

$$\theta_{k+1} = \arg\max_{\theta} \mathop{\mathbb{E}}_{(s,a)\sim\pi_{\theta_k}} [L(s, a, \theta_k, \theta)]$$

with

$$L(s, a, \theta_k, \theta) = \min\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s, a), g(\epsilon, A^{\pi_{\theta_k}}(s, a))\right),$$

and

$$g(\epsilon, A) = \begin{cases} (1+\epsilon)A & A \geq 0 \\ (1-\epsilon)A & A < 0. \end{cases}$$

$A^{\pi_{\theta_k}}$ is the advantage function related to the policy π_{θ_k},

$$\mathop{\mathbb{E}}_{(s,a)\sim\pi_{\theta_k}}$$

stands for the average with respect to (s, t). With this notation, the actions a are distributed according to the policy π_{θ_k} and the states s follow the stationary distribution of the Markov chain for the policy π_{θ_k}. The proof of the convergence of this method is in [37]. This approach tries to increase the probability of taking the best action without moving too far from the current policy, avoiding the system collapse (trust region approach [38]). Indeed, in L the hyperparameter ϵ represents how far the new policy can be from the old one. If A is positive, the picked action is better than the expectations, and it becomes more likely to choose it again. Otherwise, if A is negative, the picked action will be less called. The minimum and the function g limit the policy change by imposing the probability ratio to stay within an interval of amplitude 2ϵ of around 1.

5.2 Derivation of the model

In this section, we describe in detail the derivation of the model. Some excellent guidelines for elastic rods and material mechanics can be found in [32, 34].

Consider a planar elastic rod Γ, $\{\underline{d}_1, \underline{d}_2, \underline{d}_3\}$ subject to an external force F and an external moment L. The balance between the internal force n and the internal moment m with F and L is expressed by the following equations:

$$\begin{cases} \partial_s \underline{n} + \underline{f} = 0 \\ \partial_s \underline{m} + \partial_s \underline{\Gamma} \times \underline{n} + \underline{l} = 0 \end{cases} \tag{8}$$ {eq:balance}

where f and l represents respectively the external force F and the external moment L per unit of length. In other words, we write $f = \partial_s F$ and $l = \partial_s L$. Since we are just considering the gravity force, we employ the plane coordinates $\{\underline{e}_1, \underline{e}_2\}$ to recast the first equation of system (8):

$$\partial_s \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -g\rho_3(s)A(s) \end{bmatrix} = 0.$$

Here g is the gravity acceleration constant, $\rho_3(s)$ is the volume density of the elastic rod and $A(s)$ is the area of the cross-section of the rod at the point $\underline{\Gamma}(s)$. We assume that there are no internal forces acting at the tip of the rod. So, we get

$$n_1 \equiv 0\,,\; n_2(s) = -g \int_s^L \rho_3(s')A(s')ds'. \tag{9}$$ {eq:balance_force}

The internal moment m per unit of length must be balanced with the moment per unit of length generated by the internal force n (second equation of system (8) with $l = 0$). This gives the relation:

$$\partial_s m(s) + \sin(\theta(s))n_2(s) - \cos(\theta(s))n_1(s) = 0. \tag{10}$$ {eq:balance_moment}

The combination of equations (1), (9) and (10) give relation (2).

Now, we want to prove the relation (3). We assume that stress σ and strain ε are proportional:

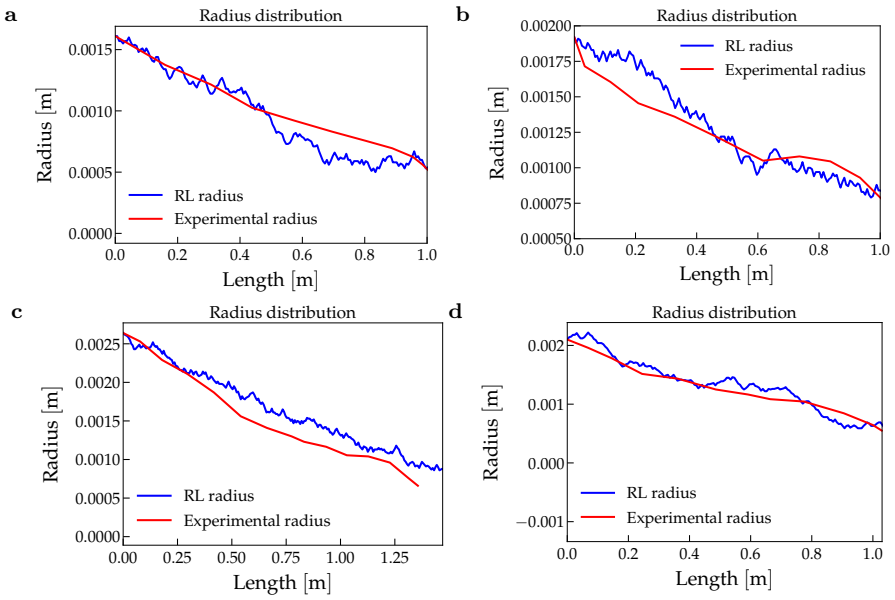
$$\sigma(s, z) = E(s)\varepsilon(s, z).$$

We recall that z represents the distance from the centreline along $\beta(s)$ on the cross-section $C(s)$. We also assume that the strain ε has the following form:

$$\varepsilon(s, z) = \alpha(s)z,$$

where $\alpha(s)$ is a proportionality constant which may vary along the rod. Since the stress $\sigma(s, z)$ is applied to the infinitesimal strip $\mathcal{L}(C(s, z))dz$, where $\mathcal{L}$ is the length (to be more precise, the Lebesgue measure) of $C(s, z)$, the internal moment acting on the cross-section $C(s)$ with respect to its centre is

$$\begin{aligned} m(s) &= \int_{\mathbb{R}} z\sigma(s, z)\mathcal{L}(C(s, z))dz \\ &= E(s)\alpha(s) \int_{C(s)} z^2 dzdw \\ &= E(s)\alpha(s)I(s). \end{aligned}$$



{fig: plots without leaves

Figure 3: Radius distribution comparison between MM model (with estimated parameters) and experimental radii. The subplots **a**, **b**, **c**, **d** show the samples S1, S3, S4, and S5, respectively.

This gives the relation (3), because we get

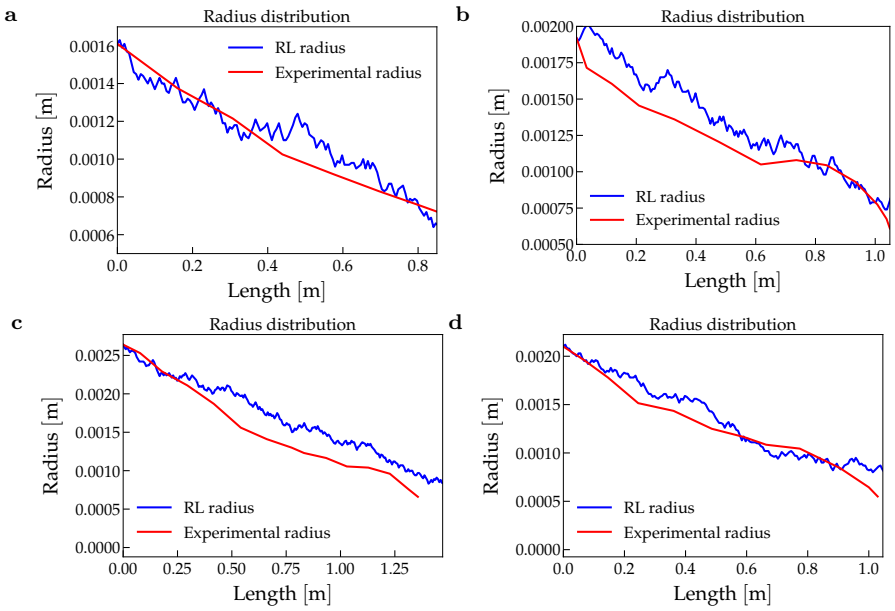
$$\frac{m(s)}{I(s)} z = \sigma(s, z).$$

Consequently, the maximal stress is at the edge of the cross-section, with $z = \max\{|y| : C(s, y) \neq \emptyset\}$.

5.3 Further Simulations and Tables

In Figure 3, we show the performance of our MM model w.r.t. the experimental radii of the samples, which appear in [28, 29].

In Figure 4, we show the performance of our ML model w.r.t. the experimental radii of the samples, which appear in [28, 29].



{fig: plots with leaves wi

Figure 4: Radius distribution comparison between ML model (with estimated parameters) and experimental radii. The subplots **a**, **b**, **c**, **d** show the samples S1, S3, S4, and S5, respectively.

References

[1] G Lawlor et al. *Plant physiology*. Springer Science & Business Media, 2012.

[2] Emanuela Del Dottore et al. "Circumnutations as a penetration strategy in a plant-root-inspired robot". In: *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2016, pp. 4722–4728.

[3] Mathieu Rivière, Julien Derr, and Stéphane Douady. "Motions of leaves and stems, from growth to potential use". In: *Physical biology* 14.5 (2017), p. 051001.

[4] George Greenhill. "Determination of the greatest height consistent with stability that a vertical pole or mast can be made, and the greatest height to which a tree of given proportions can grow". In: *Proceedings of the Cambridge Philosophical Society*. Vol. 4. 1881, p. 65.

[5] Thomas McMahon. "Size and shape in biology". In: *Science* 179.4079 (1973), pp. 1201–1204.

[6] Z Wei, S Mandre, and L Mahadevan. "The branch with the furthest reach". In: *Europhysics Letters* 97.1 (2012), p. 14005.

[7] Alberto Bressan and Qing Sun. "On the optimal shape of tree roots and branches". In: *Mathematical Models and Methods in Applied Sciences* 28.14 (2018), pp. 2763–2801.

[8] Fabio Tedone et al. “Optimal control of plant root tip dynamics in soil”. In: *Bioinspiration & Biomimetics* 15.5 (2020), p. 056006.

[9] R McNeill Alexander. *Optima for animals*. Princeton university press, 1996.

[10] Suzanne Lenhart and John T Workman. *Optimal control applied to biological models*. CRC press, 2007.

[11] Nick P Rowe and Thomas Speck. “Biomechanical characteristics of the ontogeny and growth habit of the tropical liana Condylocarpon guianense (Apocynaceae)”. In: *International Journal of Plant Sciences* 157.4 (1996), pp. 406–417.

[12] Nick Rowe, Sandrine Isnard, and Thomas Speck. “Diversity of mechanical architectures in climbing plants: an evolutionary perspective”. In: *Journal of Plant Growth Regulation* 23.2 (2004), pp. 108–128.

[13] Nick P Rowe and Thomas Speck. “Stem biomechanics, strength of attachment, and developmental plasticity of vines and lianas”. In: *Ecology of lianas* (2015), pp. 323–341.

[14] Pamela S Soltis et al. “Plants meet machines: Prospects in machine learning for plant biology”. In: *Applications in Plant Sciences* 8.6 (2020).

[15] Mohsen Hesami et al. “Machine learning: Its challenges and opportunities in plant system biology”. In: *Applied Microbiology and Biotechnology* 106.9-10 (2022), pp. 3507–3530.

[16] Anjun Ma et al. “Integrative methods and practical challenges for single-cell multi-omics”. In: *Trends in biotechnology* 38.9 (2020), pp. 1007–1022.

[17] Hock Chuan Yeo and Kumar Selvarajoo. “Machine learning alternative to systems biology should not solely depend on data”. In: *Briefings in Bioinformatics* 23.6 (2022), bbac436.

[18] Dhivya Elavarasan and PM Durairaj Vincent. “Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications”. In: *IEEE access* 8 (2020), pp. 86886–86901.

[19] Emre O Neftci and Bruno B Averbeck. “Reinforcement learning in artificial and biological systems”. In: *Nature Machine Intelligence* 1.3 (2019), pp. 133–143.

[20] Bruno Averbeck and John P O’Doherty. “Reinforcement-learning in fronto-striatal circuits”. In: *Neuropsychopharmacology* 47.1 (2022), pp. 147–162.

[21] Neythen J Treloar et al. “Deep reinforcement learning for optimal experimental design in biology”. In: *PLOS Computational Biology* 18.11 (2022), e1010695.

[22] Mohamed Helmy, Derek Smith, and Kumar Selvarajoo. “Systems biology approaches integrated with artificial intelligence for optimized metabolic engineering”. In: *Metabolic Engineering Communications* 11 (2020), e00149.

[23] Zhenpeng Zhou, Xiaocheng Li, and Richard N Zare. "Optimizing chemical reactions with deep reinforcement learning". In: *ACS central science* 3.12 (2017), pp. 1337–1344.

[24] Yasmeen Hitti et al. "Growspace: A Reinforcement Learning Environment for Plant Architecture". In: *Available at SSRN 4329504* ().

[25] Greg Brockman et al. *OpenAI Gym*. 2016. eprint: `arXiv:1606.01540`.

[26] Antonin Raffin et al. "Stable-baselines3: Reliable reinforcement learning implementations". In: *The Journal of Machine Learning Research* 22.1 (2021), pp. 12348–12355.

[27] John Schulman et al. "Proximal policy optimization algorithms". In: *arXiv preprint arXiv:1707.06347* (2017).

[28] Tom Hattermann et al. "Mind the gap: reach and mechanical diversity of searcher shoots in climbing plants". In: *Frontiers in Forests and Global Change* 5 (2022).

[29] Giacomo Vecchiato et al. "A 2D Model to describe the mechano-sensory behaviour of self-supporting shoots of climbing plants against gravity". In: *Submitted* ().

[30] Ernesto Gianoli. "The behavioural ecology of climbing plants". In: *AoB plants* 7 (2015).

[31] Giacomo Vecchiato, Michele Palladino, and Pierangelo Marcati. "An optimal control approach to the problem of the longest self-supporting structure". In: *In Preparation* ().

[32] Alain Goriely. *The mathematics and mechanics of biological growth*. Vol. 45. Springer, 2017.

[33] Daniele Agostinelli et al. "Nutations in growing plant shoots: The role of elastic deformations due to gravity loading". In: *Journal of the Mechanics and Physics of Solids* 136 (2020), p. 103702.

[34] Barry J Goodno and James M Gere. *Mechanics of materials*. Cengage learning, 2020.

[35] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[36] Tingwu Wang et al. "Benchmarking model-based reinforcement learning". In: *arXiv preprint arXiv:1907.02057* (2019).

[37] Markus Holzleitner et al. "Convergence proof for actor-critic methods applied to ppo and rudder". In: *Transactions on Large-Scale Data-and Knowledge-Centered Systems XLVIII: Special Issue In Memory of Univ. Prof. Dr. Roland Wagner*. Springer, 2021, pp. 105–130.

[38] Andrew R Conn, Nicholas IM Gould, and Philippe L Toint. *Trust region methods*. SIAM, 2000.