Proceedings RootS25
3rd International Workshop on Soil-Vegetation-Atmosphere Interaction
Boldrin D, Cecconi M, Cotecchia F, Leung A, Pedone G, Perrini P, Romero E, Tagarelli V, Zdravković L (eds)
ISSN: 3107-300X

doi: doi.org/10.53243/RootS2025-45

pyrootmemo: An application to unify root reinforcement models

A. Yildiz[1], G.J. Meijer[2]

[1]*Methods for Model-based Development in Computational Engineering, RWTH Aachen University, Aachen, Germany*

[2]*Department of Architecture & Civil Engineering, University of Bath, Bath, United Kingdom*

ABSTRACT: Estimating vegetation effects on slope stability has been of academic and professional interest since late 1960s, motivated initially by understanding failures after clear cutting for timber harvesting and recently by designing nature-based solutions. A plethora of analytical, empirical, semi-empirical and numerical models can be found in the literature, using a wide variety of soil and root parameters, assumptions, coefficients, systems of units of measurement, and calculation methods. As a physics-based model using well-established constitutive laws or widely used numerical techniques is still largely lacking, many researchers and practitioners employ multiple methods from the literature and compare the results. We present herein an application to unify many root reinforcement models ranging from earlier models from the 1970s, based on force equilibrium, to latest extensions of fibre bundle models. pyrootmemo was developed entirely in Python, utilising the principles of object-oriented programming ensuring compatibility between different models. Modularity of the package enables users to adapt pyrootmemo to define material parameters and implement their models. Furthermore, automated metadata extraction yields reproducible model calculations ensuring transparency. Through providing an open-source and standardised application we aim to promote accessibility, clarity, accuracy and practical useability, thus facilitating future vegetation-based engineering.

Keywords: Root reinforcement; models; Python package

1 INTRODUCTION

Quantification of the effects of plant roots on the strength of soil has been of interest since the 1960s (Stokes and Yildiz, 2023). Many models to predict root reinforcement have since developed, for example (i) the Wu/Waldon model (WWM) by Wu et al. (1979); (ii) the reinforcement mobilisation model by Waldron (1977) and Waldron & Dakessian (1981), henceforth referred to as the 'Waldron' model; (iii) the fibre bundle model (FBM) by Pollen and Simon (2005) and generalised by Meijer (2021); (iv) the Root Bundle Model (RBMw) by Schwarz et al. (2013); or (v) the Dundee Root Analytical Model (DRAM) by Meijer et al. (2022), among many others. We identified four major challenges with the current state of model development:

1) Abundance of models: Many different models have been developed, with widely varying levels of complexity as well as the type and number of required input parameters. Currently, it is difficult to compare results from different models and determine which model performs best, and under what conditions (Cecconi et al., 2025).

2) Model implementation: Models range from simple analytical equations to more complex numerical algorithms. In cases where code is provided, these are written in different languages (e.g. C++, R, Python, etc.) with varying levels of documentation. This is limiting the accessibility and reuse of existing code.

3) Parameter definitions: models use different terminology which can be confusing. For example, in some models the tensile stiffness refers to a secant modulus while others use a tangent modulus. To avoid unnecessary errors when using these models, we must ensure clarity of definitions.

4) Parameter units: different models are commonly defined using different unit systems. More dangerously, some equations rely on a specific unit choice in order to yield the correct results. In other words, units can be 'hidden' in seemingly dimensionless model scalar coefficients. Such instances must be identified and eradicated to avoid unnecessary error.

To address these challenges, we present *pyrootmemo* (Yildiz & Meijer, 2025), an open-source Python package for root reinforcement predictions that unifies model terminology, streamlines the use of units, and facilities quick and transparent model selection, calculations and comparisons. Furthermore, sensitivity analysis and uncertainty quantification of models can be implemented easily thanks to the modular package structure. We welcome new contributions in the form of both implementations of existing and new models, as well as experimental data.

2 PACKAGE STRUCTURE

pyrootmemo is designed using Object-Oriented Programming (OOP) principles, making it easy to extend and customise to the assumptions, parameters and requirements of various models. Figure 1 illustrates the package structure consisting of modules, classes and subclasses. pyrootmemo consists of three main modules: *materials*, *models*, and *geometry*.

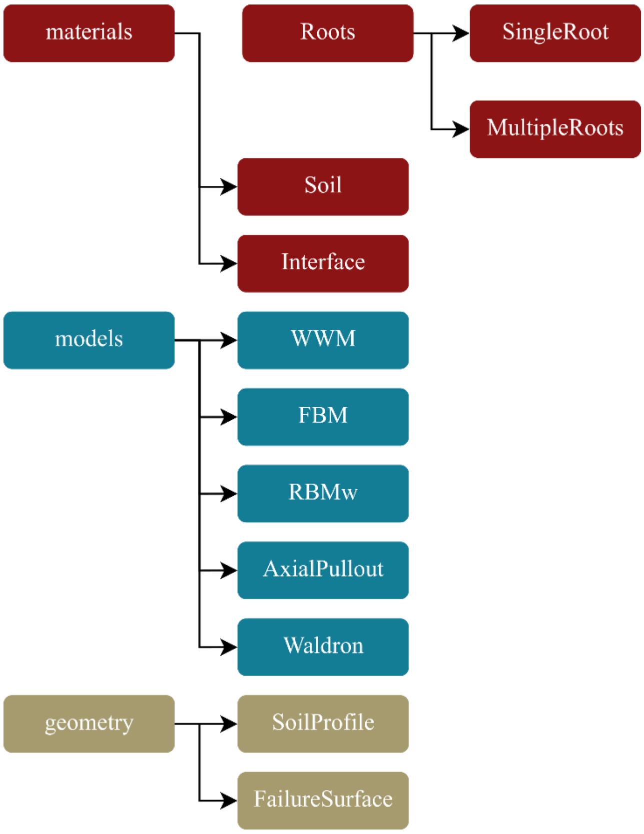

Figure 1. Package structure of pyrootmemo, illustrating modules, classes and subclasses

The *materials* module represents materials used in root reinforcement modelling, including roots, soils, and interfaces between them. *Roots*, as a parent class, has two child classes: *SingleRoot* and *MultipleRoots*, whereas *Soil* and *Interface* are only defined as parent classes. All materials are defined based on a controlled parameter list (which can be extended) so that the users can only enter the required information. For example, roots can be defined by their elastic modulus, diameter,

```
from pyrootmemo.materials import SingleRoot, MultipleRoots
from pyrootmemo.helpers import Parameter

alnus = MultipleRoots(species = 'alnus_incana',
                      diameter = Parameter([1, 2, 3], 'mm'),
                      tensile_strength = Parameter([40, 30, 20], 'MPa'),
                      elastic_modulus = Parameter([100, 100, 100], 'MPa'),
                      length = Parameter([50, 50, 50], 'cm'))

poa = SingleRoot(species = 'poa_pratensis',
                 diameter = Parameter(1, 'mm'),
                 tensile_strength = Parameter(30, 'MPa'),
                 elastic_modulus = Parameter(100, 'MPa'),
                 length = Parameter(40, 'cm'))
```

Figure 2. Creating MultipleRoots and SingleRoot objects using pyrootmemo

tensile strength, yield strength, plastic modulus, unload modulus, length and/or initial orientation (azimuth and elevation angle within a spherical coordinate system). All these parameters can be either entered as a numeric value for the *SingleRoot* class, or as a Python list for the *MultipleRoots* class. Figure 2 shows two examples of creating an individual root or a group of roots. Defining numerical values of parameters, e.g. diameter or tensile strength, requires the use of a helper class called *Parameter*. As many models implemented use different units, we utilised an existing Python package, *pint*, to efficiently handle unit conversions and dimensionality checks in the background (Grecco et al., 2012-2025). Once an object is created, each parameter becomes its attribute, thus any value can be called easily. For example, *poa.diameter* returns an object with a magnitude of 1.0 and a unit of mm, which is the diameter value of the Poa pratensis root object created in Figure 2.

```
import numpy as np
from pyrootmemo.models import Waldron
from pyrootmemo.materials import MultipleRoots, Interface, Soil
from pyrootmemo.geometry import SoilProfile, FailureSurface
from pyrootmemo.helpers import Parameter

soil = Soil(
    name = 'silty SAND',
    cohesion = Parameter(0.0, 'kPa'),
    friction_angle = Parameter(36.4, 'degrees')
)

soil_profile = SoilProfile(
    [soil], depth = Parameter([500.0], 'mm')
)

interface = Interface(
    shear_strength = Parameter(2.7, 'kPa')
)

failure_surface = FailureSurface(
    cross_sectional_area = Parameter(0.25 * np.pi * 110**2, 'mm^2'),
    depth = Parameter(250, 'mm'),
    shear_zone_thickness = Parameter(30, 'mm')
)

model = Waldron(
    alnus,
    interface,
    soil_profile,
    failure_surface,
    breakage = True,
    slipping = True
)

peak = model.calc_peak_reinforcement()
```

Figure 3. Calculating the peak reinforcement value using Waldron model

Models implemented so far (v0.1) in *pyrootmemo* includes: WWM, FBM, RBMw, AxialPullout and Waldron-type models. In addition to the models introduced in Section 1, The AxialPullout class models the relationship between root tensile forces and axial displacements, based on solving force equilibrium while accounting for root stiffness and root-soil interface resistance. These underpin direct shear models develop by Waldron (1977), Waldron & Dakessian (1981) and Meijer et al. (2022). Each model requires minimally *Soil* and *Roots* object in addition to model-specific requirements coded in the *geometry* module including the classes *SoilProfile* and *FailureSurface*.

Figure 3 depicts the calculation of the peak root reinforcement using a Waldron model. Once the necessary arguments are defined, e.g. root, soil, profile and failure surface objects, the code automatically checks if required attributes are defined. For instance, if a *SingleRoot* object is created without the tensile strength of the root, the package does not allow the creation of a *Waldron* model object. In addition to the calculation of the peak root reinforcement, reinforcement versus displacements plots can be generated automatically using the *plot* method of the model class.

Implementation of new models follow the same structure of the code presented in Figure 3. Every parameter in a model, e.g. root diameter or internal friction angle of soil, is fed into the calculations from the attributes of the corresponding class. Therefore, a new model would have its own methods implemented by an external user which only calls the attributes from the other objects. Due to the modular and flexible implementation of material classes, no custom modifications are required.

3 EXAMPLES

pyrootmemo values clarity of results presentation. The following section provides a synthetic but educational example for three (linear elastic) roots, see Table 1.

Table 1. Root properties used for model simulations

Diameter d_r [mm]	**Tensile strength** $t_{r,u}$ [MPa]	**Young's modulus** E_r [MPa]	**Length** L_r [mm]
3	30	200	500
4	20	180	500
5	10	160	500

These roots reinforce a soil failure surface with a cross-sectional area of $A_s = 0.1$ m^2 and a shear zone thickness of $h = 10$ mm. The soil angle of internal friction was set to $\phi' = 30°$, and the axial root-soil interface strength was chosen as $\tau_{i,u} = 10$ kPa.

Table 2. Model input parameter requirements

Parameter	**WWM**	**FBM**	**RBMw**	**WF**	**WFS**
Root					
d_r	x	x	x	x	x
$t_{r,u}$	x	x	x	x	x
E_r			x	x	x
L_r					x
Soil					
A_s	x	x	x	x	x
ϕ'	x			x	x
h				x	x
Interface (between roots and soil)					
$\tau_{i,u}$				x	x

Prediction of root mechanical reinforcement were made using the WWM, two FBMs with load sharing factor of $\beta_F = 0$ and 2 respectively, the RBMw (with an assumed Weibull shape parameter of $\kappa = 4$), and Waldron-type models incorporating tensile failure only (henceforth indicated using the acronym WF), or both tensile failure and root slippage (WFS). Each model required a unique set of root, soil and root-soil interface parameters, as indicated in Table 2.

Table 3. Predicted peak root reinforcements

Model	**Reinforcement [kPa]**
WWM	6.31
FBM, $\beta_F = 0$	4.24
FBM, $\beta_F = 2$	3.93
RBMw	2.99
Waldron, breakage only	4.74
Waldron, breakage and slippage	2.16

These different sets of parameters had to be defined only once and could be reused for each model. Thus, only a few lines of code were required to obtain reinforcement predictions, highlighting the value of the object-oriented approach of *pyrootmemo*.

The peak root reinforcement predicted by the investigated models varies significantly, see Table 3. In addition, the rate and order in which roots mobilise their strength differs between models (Figures 4, 5 and 6).

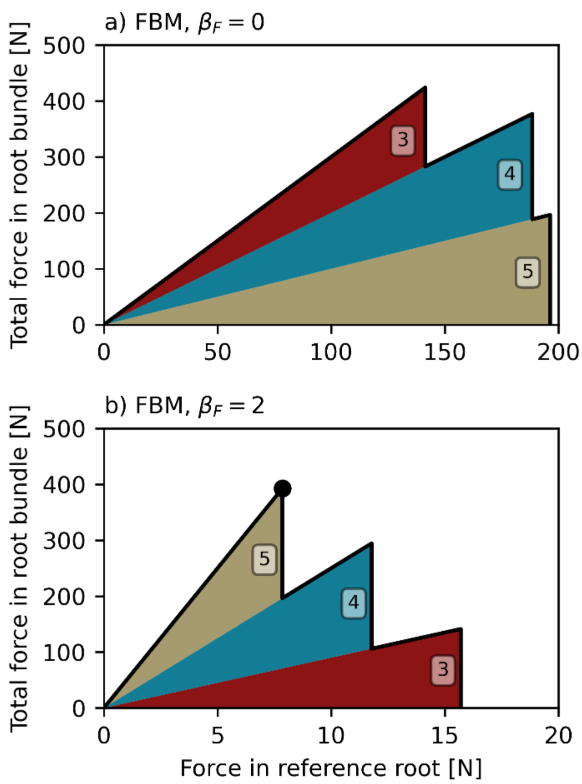

Figure 4. Mobilisation of root forces according to different FBMs. The 'reference root' is a hypothetical root with a diameter of 1 mm that is used to track the progressive nature of force mobilisation. Colours and labels indicate the contribution of individual roots and their diameters

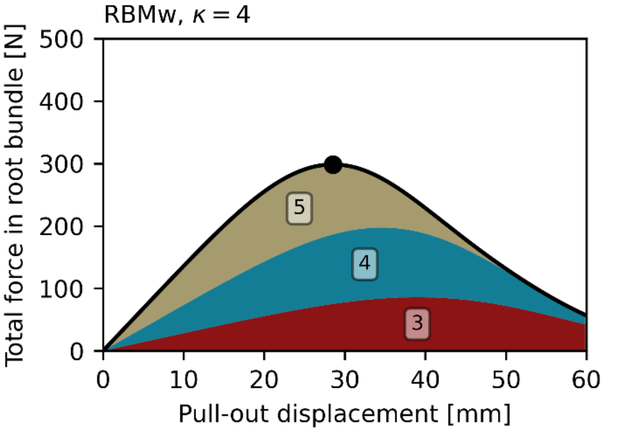

Figure 5. Mobilisation of root forces according to the RBMw model. Colours and labels indicate the contribution of individual roots and their diameters

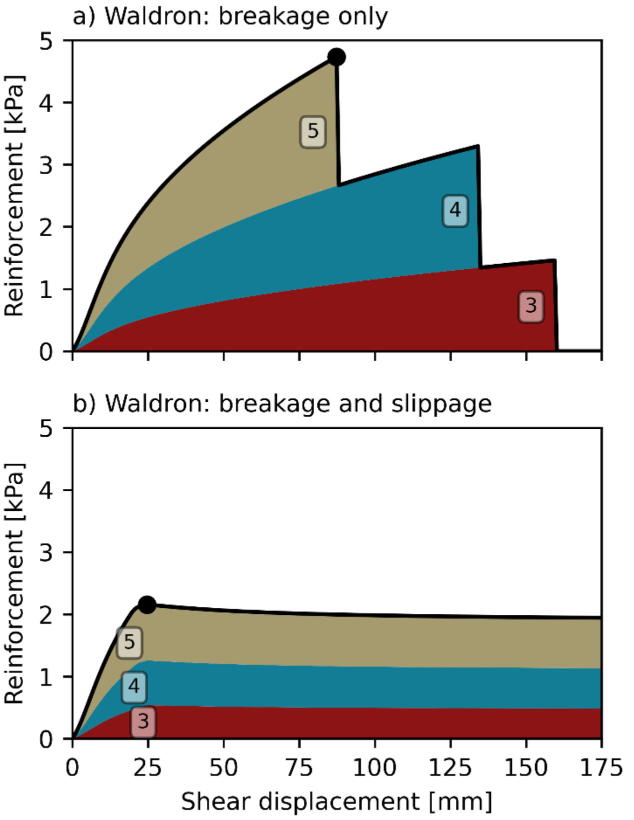

Figure 6. Mobilisation of root reinforcement with soil shear displacement according to Waldron-type models a) excluding and b) including root slippage. Colours and labels indicate the contribution of individual roots and their diameters

4 CONCLUSIONS

pyrootmemo provides a single platform to simplify using a range of root reinforcement models through standardisation of inputs, documentation and model outputs. It is a project still under development, and new versions will be released in the future. Future implementation plan includes i) adding further existing models (DRAM, further fibre bundle and root bundle models), ii) creating a graphical user interface, and iii) developing an integrated datahub for sharing open access root tensile strength data. We encourage readers to contribute to this open-source project through implementing new models, contributing experimental data and/or providing user feedback.

5 REFERENCES

Cecconi, M., Tagarelli, V., Cotecchia, F., Pane, V., Anselmucci, F., Bertolini, I., Biondi, G., Boldrin, D., Capobianco, V., Cardile, G., Cuomo, S., De Vita, P., Fraccica, A., Meijer, G., Pagano, L., Pirone, M., Schwarz, M., Tarantino, A., Vaunat, J., Yildiz, A. 2025. Soil-vegetation-atmosphere interaction for engineering applications: Recent multi-scale and multi-disciplinary insights, *Geomech. Energy Environ.* **43**, 100723. doi: 10.1016/j.gete.2025.100723

Grecco, H.E. 2012-2025. Pint: makes units easy. url: github.com/hgrecco/pint, Accessed on 2025-10-06.

Meijer, G.J. 2021. A generic form of fibre bundle models for root reinforcement of soil, *Plant and Soil* **468**, 45–65.

Meijer, G.J. Knappett, J.A. Bengough, A.G. Bull, D.J. Liang, T. Muir Wood, D. 2022. DRAM: A three-dimensional analytical model for the mobilisation of root reinforcement in direct shear conditions, *Ecological Engineering* **179**, 106621.

Pollen, N. Simon, A. 2005. Estimating the mechanical effects of riparian vegetation on stream bank stability using a fiber bundle model, *Water Resources Research* **41**, W07025.

Schwarz, M. Giadrossich, F. Cohen, D. 2013. Modeling root reinforcement using a root-failure Weibull survival function, *Hydrology and Earth Systems Science* **17**, 4367–4377.

Stokes, A. Yildiz, A. 2023. The reinforcement of soil by plant roots. *Encyclopedia of Soils in the Environment (Second Edition)*. doi: 10.1016/B978-0-12-822974-3.00226-3.

Waldron, L.J. 1977. Shear resistance of root-permeated homogeneous and stratified soil, *Soil Science Society of America Journal* **41**(5), 843–849.

Waldron, L.J. Dakessian, S. 1981. Soil reinforcement by roots – calculation of increased soil shear resistance from root properties, *Soil Science* **132**(6), 427–435.

Wu, T.H. McKinnell III, W.P. Swanston, D.N. 1977. Strength of tree roots and landslides on Prince of Wales Island, Alaska, *Canadian Geotechnical Journal* **16**(1), 19–33.

Yildiz, A. Meijer, G.J. 2025. Pyrootmemo. url: github.com/rootmemo/pyrootmemo, Accessed on 2025-10-06.