RESolve – a 1D Richards Equation Model in Matlab

Lennon Ó Náraigh*

School of Mathematics and Statistics,
University College Dublin, Belfield, Dublin 4, Ireland

(Dated: November 28, 2025)

This report presents `RESolve`, a numerical method for the solution of the one-dimensional Richards equation in a saturated or unsaturated soil column. A sink term representing evapotranspiration is included. Different options for the boundary conditions (top and bottom) are available. The equation is solved numerically using MATLAB's `pdepe` solver. We present validation results confirming the correctness of our approach. We present an open-source test case, as well as a link to Github repository to promote the sustainable future use of `RESolve`.

I. GOVERNING EQUATION AND BOUNDARY CONDITIONS

In this report we use the 1D Richards Equation to model the flow of moisture in a soil column. The equation is given here as:

$$\frac{\partial\theta(h)}{\partial t} = \frac{\partial}{\partial z}\left[K(h)\left(\frac{\partial h}{\partial z}+1\right)\right] - \lambda(h,z,t), \qquad z \in (0,L), \qquad t > 0, \tag{1}$$

where

- $\theta(h)$ is the volumetric water content in notional units of $\mathrm{m}^3/\mathrm{m}^3$;
- h is the pressure head in units of m. The pressure head is negative in unsaturated conditions; $h = 0$ indicates saturation;
- $K(h)$ is the hydraulic conductivity, in units of $\mathrm{m}\cdot\mathrm{s}^{-1}$;
- λ is a sink term representing evapo-transpiration (ET), in units of s^{-1};
- z is the vertical coordinate, increasing upward with $z = 0$ at the bottom and $z = L$ at the soil surface.

This sign convention follows the Hydrus system. The same convention is used in the standard paper on the solution of the Richards equation by Dogan and Motz [1]. The gravity term therefore appears with a positive sign (as $K(h)$) in the flux expression. For the avoidance of doubt, the flux is:

$$f = K(h)\left(\frac{\partial h}{\partial z}+1\right). \tag{2}$$

A. Soil Hydraulic Functions: van Genuchten model

To close the Richards equation, constitutive relationships are required for the soil the unsaturated hydraulic conductivity, among other things. We use the van Genuchten–Mualem

* Corresponding author. Email: onaraigh@maths.ucd.ie

(VG) model for this purpose. We first of all introduce the effective saturation:

$$S_e(h) = \begin{cases} \dfrac{\theta(h) - \theta_r}{\theta_s - \theta_r}, & h < 0, \\ 1, & h \geq 0, \end{cases} \tag{3}$$

where θ_r and θ_s are the residual and saturated water contents. Here, θ_r and θ_s are limiting values corresponding to a residual saturation under dry conditions, and a saturated condition. Hence, $\theta_r \leq \theta \leq \theta_s$. Correspondingly, $0 \leq S_e \leq 1$. For negative heads corresponding to unsaturated soils, the VG model links h to S_e via the relation:

$$S_e(h) = \left[1 + (\alpha|h|)^n\right]^{-m}, \tag{4}$$

Equation (4) describes the so-called **water-retention curve**. Here, α is the air-entry pressure and is a parameter of the model. Also,

$$m = 1 - \tfrac{1}{n}, \qquad \alpha > 0, \quad n > 1, \tag{5}$$

and n is a further model parameter. The volumetric water content then follows as

$$\theta(h) = \theta_r + S_e(\theta_s - \theta_r). \tag{6}$$

Finally, the permeability is given by:

$$K(h) = K_s\, S_e^{\ell}\left[1 - \left(1 - S_e^{1/m}\right)^m\right]^2, \tag{7}$$

where K_s is the saturated hydraulic conductivity and ℓ is the pore-connectivity parameter (commonly $\ell = 0.5$).

In practice, it is better to solve the Richards Equation in the variable h. Therefore, Equation (1) is re-arranged. We have:

$$\frac{\partial \theta}{\partial t} = \frac{\partial \theta}{\partial h}\frac{\partial h}{\partial t}. \tag{8}$$

Here, $\partial h/\partial \theta$ is computed analytically from Equation (6). In practice, we identify the capacity $C(h)$:

$$C(h) = (\theta_s - \theta_r)\frac{\mathrm{d}S_e}{\mathrm{d}h} + \theta\rho g(c_r + c_w), \tag{9}$$

and we solve the following PDE in place of Equation (1):

$$C(h)\frac{\partial h}{\partial t} = \frac{\partial}{\partial z}\left[K(h)\left(\frac{\partial h}{\partial z} + 1\right)\right] - \lambda(h, z, t), \qquad z \in (0, L), \qquad t > 0, \tag{10}$$

Here, following Mathis et al. [2], $\rho g(c_r + c_w)$ is taken to be $9.81 \times 10^{-7}\,\mathrm{m}^{-1}$.

B. Sink Term for Evapotranspiration

The sink term $\lambda(h, z, t)$ accounts for water removal due to plant root uptake and/or soil surface evaporation. We follow the work by Mathias et al. [2] and use a Feddes-type reduction function, whereby

$$\lambda(z, t) = E_p(t)\phi_1(z)\phi_2(h) \tag{11}$$

Here, $E_p(t)$ is the potential evaporation, which is obtained from meteorological conditions. The term ϕ_1 is the root function, which describes the distribution of roots in the soil column. We use:

$$\phi_1(z) = \begin{cases} \frac{a}{L_r}\left[\frac{\mathrm{e}^{-a}-\mathrm{e}^{-az_M/L_r}}{(1+a)\mathrm{e}^{-a}-1}\right], & 0 \le z_M \le L_r, \\ 0, & z_M > L_r. \end{cases} \tag{12}$$

Here, we have momentarily used the Mathias convention for the vertical coordinate, with $z_M = L - z$, such that $z_M = 0$ corresponds to the surface. The parameters a (dimensionless) and L_r (units of m) characterize the root distribution. These will be provided on a case-by-case basis. Finally, ϕ_2 is the plant stress function, described by Feddes et al. [3]:

$$\phi_2(h) = \begin{cases} 0, & h \ge h_a, \\ 1, & h_a > h > h_d, \\ 1 - \frac{h-h_d}{h_w-h_d}, & h_d \ge h \ge h_w, \\ 0, & h < h_w. \end{cases} \tag{13}$$

Notice that $\phi_2(h)$ has a jump discontinuity at h_a. Again, the parameters h_a, h_d, and h_w will be provided on a case-by-case basis, as these depend on the crop type.

C. A note on units

The sink term λ has units of s^{-1}. The potential evaporation $E_p(t)$ has units of $\mathrm{m}\cdot\mathrm{s}^{-1}$. To make the units balance, it is clear that ϕ_1 has units of m^{-1}. This makes sense, as $\phi_1(z)$ is a true distribution, in the sense that $\phi_1(z) \ge 0$, and

$$\int_0^L \phi_1(z)\mathrm{d}z = 1. \tag{14}$$

D. Two notes of Caution!

Care should be taken that parameters are selected such that:

$$\theta(h_w) > \theta_r. \tag{15}$$

Otherwise, the ET function will be active when the medium is effectively dry. Not only is this unphysical, but it will cause code instability and potentially, blowup. Additionally, it should be noticed that ϕ_2 is a jump discontinuity at $h = h_a$, which may not be desirable from the numerical point of view. For a plot of $\phi_2(h)$, see Figure 1.

E. Boundary Conditions

Equation (1) is solved with a variety of different boundary conditions, depending on the context.

Bottom Boundary conditions: We look at two possibilities here:

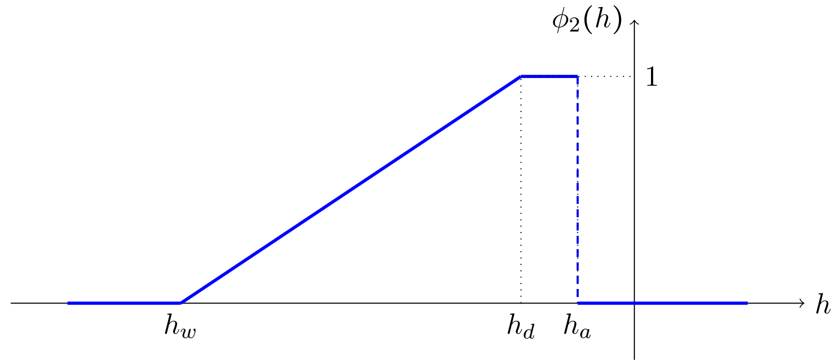

FIG. 1. Plot showing the plant stress function of Feddes et al. [3].

1. Free Drainage – this is a Neuman boundary condition, such that:
$$f = K(h), \qquad z = 0. \tag{16}$$
Since $f = K(h)\,(\partial_z h + 1)$, this amounts to $\partial_z h = 0$ at $z = 0$.
2. Saturation boundary condition – this is a Dirichlet condition, such that
$$h = h_{ref}, \qquad z = 0. \tag{17}$$
Here, h_{ref} is a constant reference level. To describe a saturated soil at $z = 0$, we take $h_{ref} = 0$.

Care should be taken here to define 'numerical saturation' slightly differently from $h = 0$ as Equation (10) may have trouble handling the extreme cases $\theta = \theta_s$ (corresponding to $h = 0$) and $\theta = \theta_r$ (corresponding to $h = -\infty$). Consequently, we introduce a 'numerical saturation level' h_{ref}, such that:
$$\theta(h_{ref}) = 0.999\theta_s. \tag{18}$$
Here, $\theta(h_{ref})$ is evaluated using Equation (6)

Top Boundary conditions: At the soil surface $z = L$, the standard approach is to use the rainfall rate as a flux boundary condition:
$$f(L,t) = f_{\mathrm{rain}}(t), \qquad \text{(Standard Top Boundary Condition)} \tag{19}$$
where $f_{\mathrm{rain}}(t)$ is the imposed rainfall rate, in units of $\mathrm{m}\cdot\mathrm{s}^{-1}$. In practice, this is obtained from meteorological data. The meteorological data is often in $\mathrm{mm}\cdot\mathrm{day}^{-1}$ and hence, care should be taken to input a rainfall flux in SI units into the PDE solver.

A more detailed and realistic approach takes account of the ponding phenomenon. We outline here:

- how this can be done in theory;
- how it can't be done in Matlab's `pdepe`;
- how to implement a pseudo-ponding condition in Matlab's `pdepe`.

To implement ponding logic at a time t, we would solve the Richards Equation with the standard boundary condition. Specifically, we would update the state of the system, going from time t to $t + \Delta t$. At the end of the update step, we would evaluate
$$f_{max}(t) = K(h_{ref})\left[\left(\frac{h_{ref} - h(L-\Delta z, t+\Delta t)}{\Delta z}\right)_{z=L} + 1\right]. \tag{20}$$

Here, $h(L-\Delta z, t+\Delta t)$ reefers to the updated head (at time $t+\Delta t$), one grid spacing (Δz) below the surface. If the rainfall exceeds this critical rate, we would reject the update step and would instead repeat the update step with a Dirichlet condition:

$$h(L,t) = h_{ref}. \tag{21}$$

Unfortunately, this logic cannot be implemented in `pdepe` because the boundary-condition function is separate from the PDE function, meaning that the interior point $h(L-\Delta z, t+\Delta t)$ is not available for the purpose of evaluating boundary conditions. Therefore, in `pdepe`, the closest thing to ponding logic that we can achieve is to set by hand a limiting flux f_{max} as a fixed parameter. The boundary condition then reads:

$$f(L,t) = f_{\mathrm{rain}}(t), \text{ if } f_{\mathrm{rain}}(t) < f_{max}, \tag{22a}$$
$$h(L,t) = h_{ref}, \text{ otherwise}. \tag{22b}$$

F. Initial Conditions

The initial pressure head distribution is specified as

$$h(z,0) = h_0(z), \tag{23}$$

where $h_0(z)$ may be uniform or vary with depth. In practice, we use $h_0(z) = -z + z_{wt}$, where z_{wt} is the initial level of the water-table. This follows the initial condition in Mathias et al. [2], but with the Hydrus convention whereby $z = L$ is the top of the domain. This can also be seen to be an equilibrium solution of the Richards Equation, $\partial_z h + 1 = 0$, albeit one that satisfies particular boundary conditions.

II. NUMERICAL SOLUTION USING PDEPE

Our solver `RESolve` does not directly solve the Richards Equation (10). Instead, we ue MATLAB's `pdepe`, which can solve PDEs in the general form

$$c(x,t,u,u_x)\, u_t = x^{-m_D} \frac{\partial}{\partial x}\left(x^{m_D} f(x,t,u,u_x)\right) + s(x,t,u,u_x), \tag{24}$$

where m_D specifies the dimensionality ($m_D = 0$ for the present 1D Cartesian geometry). To convert the generic form (24) to suit the Richards Equation, we identify:

$$x \equiv z, \qquad u \equiv h, \qquad m_D = 0. \tag{25}$$

We also identify:

$$\begin{aligned} c(z,t,u,u_z) &= (\theta_s - \theta_r)\frac{\mathrm{d}S_e}{\mathrm{d}h} + \theta\rho g(c_r + c_w), \\ f(z,t,u,u_z) &= K(u)\left(u_z - 1\right), \\ s(z,t,u,u_z) &= -\lambda(u,z,t). \end{aligned}$$

A. Implementation of Boundary Conditions in PDEPE

PDEPE expects boundary conditions in the form

$$p(z,t,u) + q(z,t)\,f(z,t,u,u_z) = 0.$$

We go through the various options now:

1. **Bottom, free drainage:** In the case of free drainage, we impose $f = K(h)$, hence $p + qf(h) = 0$, hence $q = 1$ and $p = -K(h)$.
2. **Bottom, saturated:** In the case of saturation, we impose $h = h_{ref}$, hence $q = 0$ and $p = u - h_{ref}$.
3. **Top, rainfall flux:** The rainfall flux condition is

$$f = q_{\mathrm{rain}}(t), \qquad z = L. \tag{26a}$$

Thus,

$$p = -q_{\mathrm{rain}}(t), \qquad q = 1. \tag{26b}$$

To implement the pseudoponding boundary condition (22), only a small modification to Equation (26) is required.

B. Spatial and Temporal Discretization

A uniform spatial mesh is used:

$$z_i = i\Delta z, \quad i = 0, \ldots, N.$$

The time integration uses MATLAB's built-in ODE solver (`ode15s`), controlled by user-specified `tspan`.

C. MATLAB Structure

The model is written in a single Matlab function `richards_pdepe.m`, which includes a main function setting up the spatial grid, time vector, soil hydraulic functions, rainfall input, and calls to `pdepe`. The main function contains subfunctions:

- a PDE function defining c, f, and s,
- a boundary condition function,
- an initial condition function.

Additional functions in separate files are included to give effect to the Van Genuchten model, to handle rainfall and evapotranspiraton inputs, and also, to perform postprocessing, as required.

III. VALIDATION

We validate our solver with respect to the results of Mathias (SM) et al. [2]. For the root distribution function and the plant-stress function, we use the parameters given in that paper and summarized here again in Table I. Furthermore, the meteorological data in that paper were kindly provided by SM upon request. Consequently, the daily rainfall data and the monthly potential evapotranspiration levels are available for the present validation purposes, and were obtained from the Theale gauging station in the Kennet valley, Berkshire, UK. The time series are from January 1961 to December 1997. We further use a soil column

TABLE I. Parameters Used in the Root Distribution and Plant Stress Functions

Parameter	**Value**	**Description**
L_r	$1\,\mathrm{m}$	Root zone depth
a	$2\,\mathrm{m}$	Shape parameter of the root distribution function
h_a	$-0.05\,\mathrm{m}$	Pressure head for onset of aeration stress
h_d	$-4\,\mathrm{m}$	Pressure head for reduced uptake due to dryness
h_w	$-150\,\mathrm{m}$	Wilting point pressure head

depth $L = 3\,\mathrm{m}$ and a constant initial condition $h_0(z) = -0.1\,\mathrm{m}$. We run the simulations from January 1st 1985, with a view to comparing `RESolve` and the model of SM within the window from January 1st 1988 to December 31st 1995. Results for the sandy soil (VG parameters in Table II) are shown in Figure 2. Here, we are plotting

$$\Theta(t) = \int_0^L \theta(t,z)\,\mathrm{d}z. \tag{27}$$

Results for the clay soil (VG parameters in Table II also) are shown in Figure 3. We emphasize that we have not solved both cases (sand and clay) using the standard top boundary condition – hence, no ponding logic is applied, even for the poorly drained clay. However, the agreement between the paper of Mathias et al. [2] and the present results is good (as confirmed by visual inspection). This validates the suitability of `pdepde` for the purpose of solving the 1D Richards Equation.

TABLE II. van Genuchten Parameters for Sandy and Clay Soils

Parameter	**Symbol**	**Sandy Soil**	**Clay Soil**
Residual water content	θ_r	0.0515	0.0961
Saturated water content	θ_s	0.3769	0.4616
VG parameter	α	3.321	2.711
VG parameter	n	2.503	1.149
Saturated hydraulic conductivity ($\mathrm{mm}\cdot\mathrm{day}^{-1}$)	K_s	3220	108.5
Pore-connectivity parameter	ℓ	−0.8653	−5.153

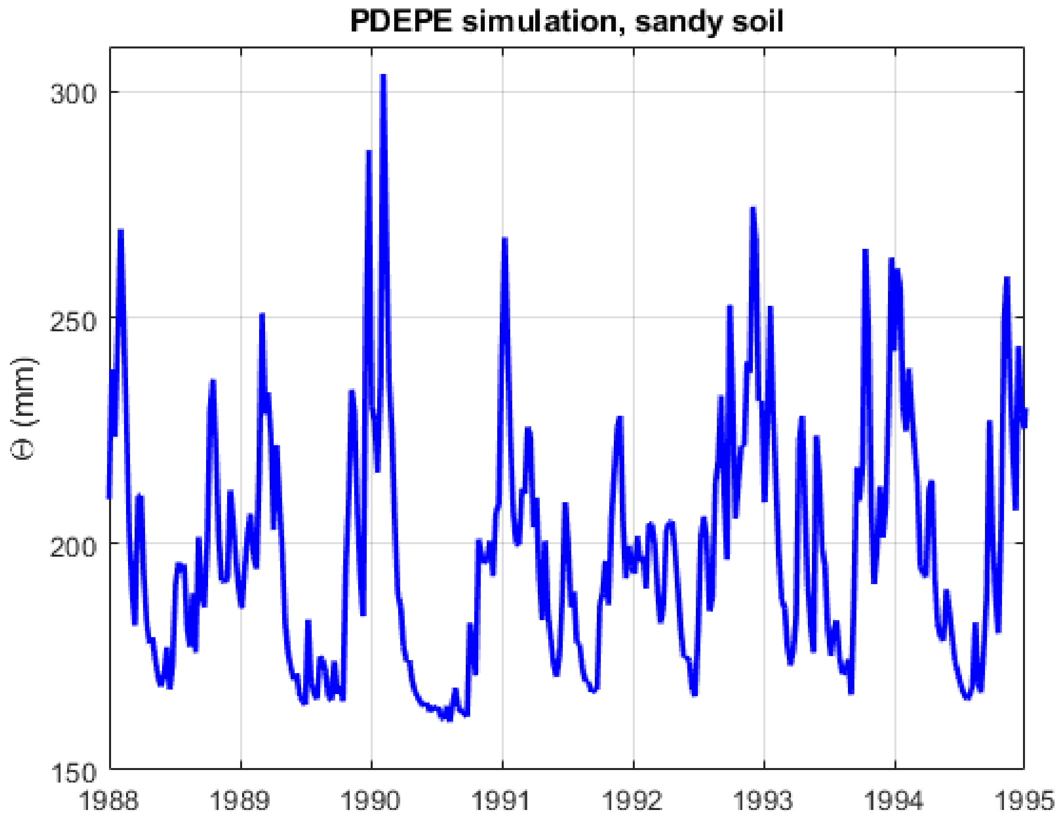

FIG. 2. Time series plot showing $\Theta(t)$ as computed using `RESolve` for a sandy soil. Results may be compared with Figure 2 in Reference [2].

A. Another note of caution!

We have not implemented a ponding condition in the present validation (or even a pseudo-ponding condition). This has not impacted the results greatly. However, this finding is not generalizable, and in future we will look at cases where implementing the ponding condition at the top produces results that differ greatly from those produced using the standard top boundary condition.

IV. OPEN-SOURCE TEST CASE

We present an open-source test case which may be used to validate other codes or familiarize a user with `RESolve`. The test case involves a two-year simulation with soil data appropriate for Johnstown Castle, Co. Wexford, Ireland, and provided in Tables III–IV. The VG parameters are obtained from the Irish Soil Map, by depth-averaging. Hence, the presented model is an effective single-layer model. We solve the equations using a soil column $L = 1.8\,\mathrm{m}$ with a saturated bottom boundary condition. We use the initial condition

$$h_0(z) = -z - 0.5\,\mathrm{m}. \tag{28}$$

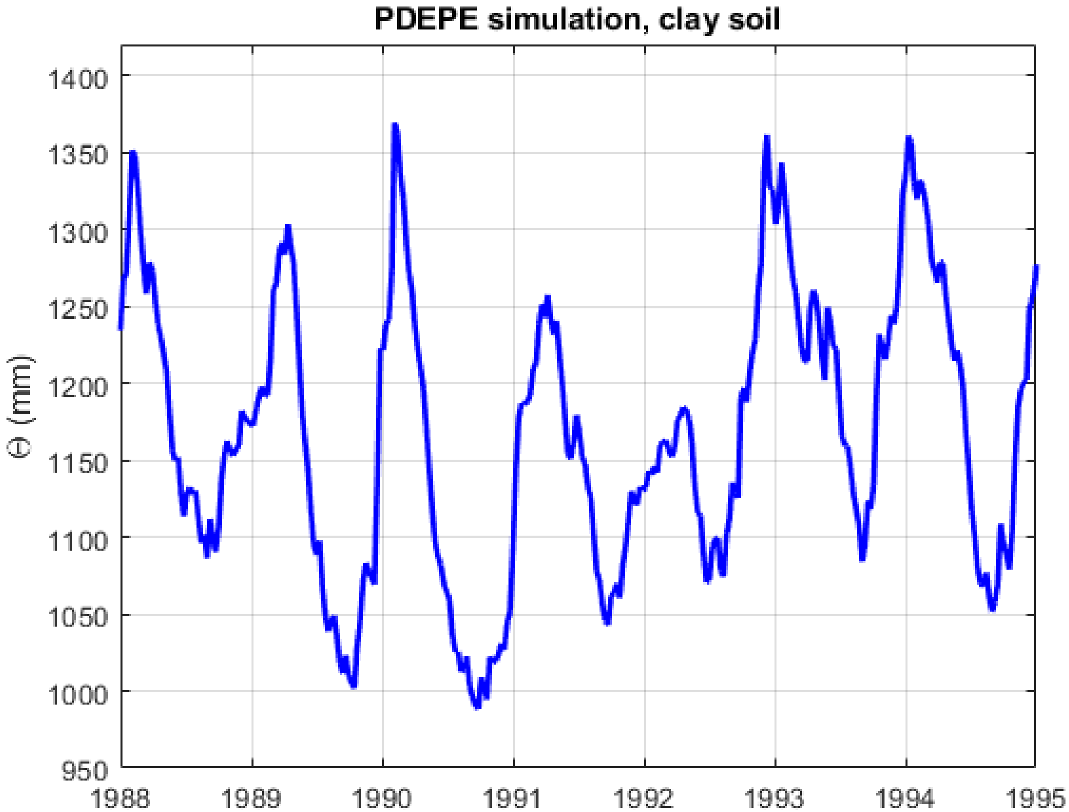

FIG. 3. Time series plot showing $\Theta(t)$ as computed using `RESolve` for a clay soil. Results may be compared with Figure 3 in Reference [2].

We simulate for four years. The first year is used for burn-in, so that after one year of simulation time, the model effectively forgets the initial condition, which may not be physical. We use synethic meteorologicacl data, with

$$q_{rainfall}(t) = q_{rf,max}\left\{\tfrac{1}{2}\left[\cos(2\pi t/T_{year}) + 1\right]\right\}, \tag{29a}$$

$$EP(t) = EP_{max}\left\{\tfrac{1}{2}\left[\cos(2\pi t/T_{year}) + 1\right]\right\}. \tag{29b}$$

As always, t is measured in seconds, and $T_{year} = 365 \times 24 \times 3600$. We also use $q_{rf,max} = 20\,\mathrm{mm}\cdot\mathrm{day}^{-1}$ and $EP_{max} = 10\,\mathrm{mm}\cdot\mathrm{day}^{-1}$. Sample results are $\Theta(t)$ are shown in Figure 4. We also show the space-time behaviour of $h(t,z)$ and $\theta(t,z)$ in Figure 5, showing the clear seasonality trends in the moisture levels in the soil.

V. CONCLUSION

Summarizing, we have introduced `RESolve`, a 1D Richards-equation mathematical model, to describe the moisture level in a saturated or unsaturated soil. The model is implemented by using the `pdepe` function in Matlab, meaning that the nitty-gritty of discretizing the

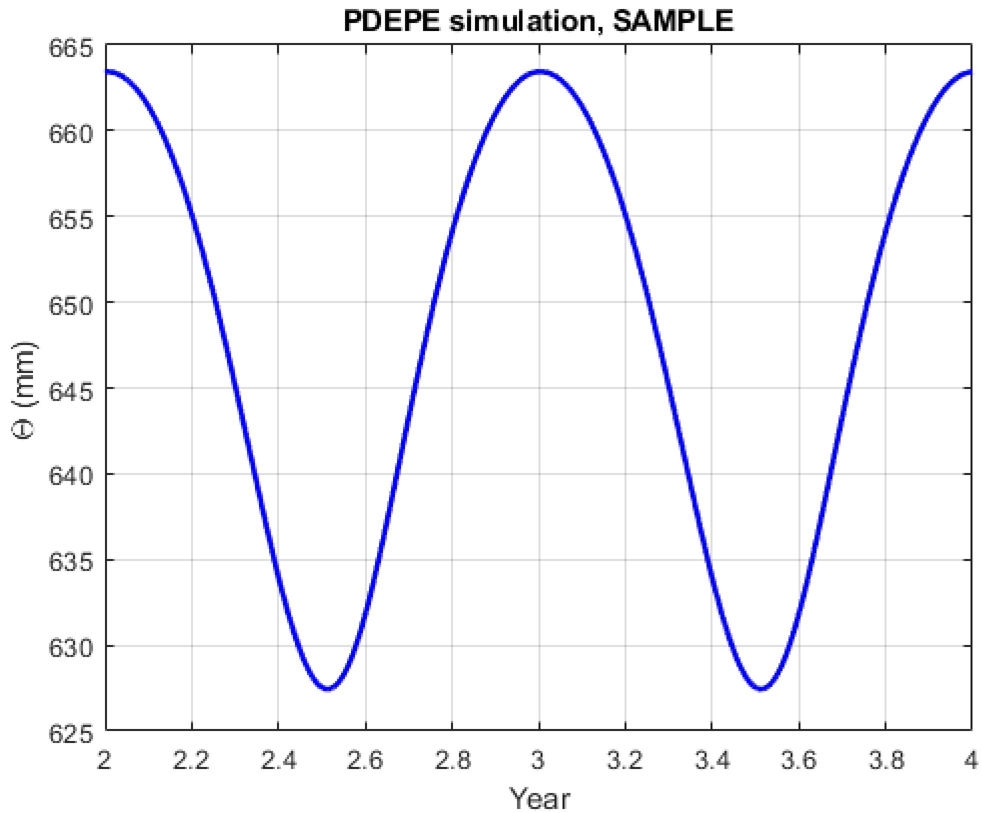

FIG. 4. Simulated time series of $\Theta(t)$ for Johnstown-castle-type soil

TABLE III. Parameters Used in the Root Distribution and Plant Stress Functions (Johnstown Castle)

Parameter	**Value**	**Description**
L_r	$0.25\,\mathrm{m}$	Root zone depth
a	$1.55\,\mathrm{m}$	Shape parameter of the root distribution function
h_a	$-0.25\,\mathrm{m}$	Pressure head for onset of aeration stress
h_d	$-3\,\mathrm{m}$	Pressure head for reduced uptake due to dryness
h_w	$-10\,\mathrm{m}$	Wilting point pressure head

TABLE IV. van Genuchten Parameters for Sandy and Clay Soils (Johnstown Castle)

Parameter	**Symbol**	**Value**
Residual water content	θ_r	0.077
Saturated water content	θ_s	0.396
VG parameter	α	0.894
VG parameter	n	1.424
Saturated hydraulic conductivity ($\mathrm{m \cdot s^{-1}}$)	K_s	$0.195/(24 \times 3600)$
Pore-connectivity parameter	ℓ	0.5

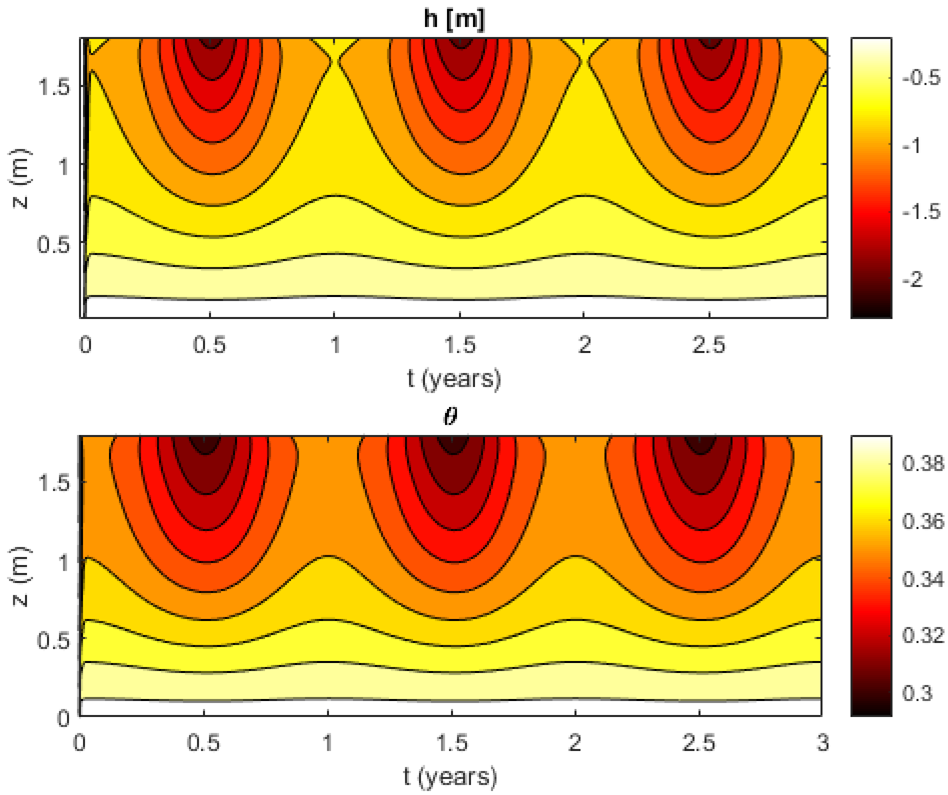

FIG. 5. Space-time plot of $h(t, z)$ and $\theta(t, z)$ for Johnstown-castle-type soil

PDE and handling the boundary conditions is left to an effective 'black box', enabling the user to focus on the model parametrization. A choice of different physically-relvant boundary conditions is available to the user. We have carefully validated our solver against the published results of Mathias *et al.* [2], for various soil types. We have presented a SAMPLE case, which we make freely available for validation and testing on the Github repository onaraighl/RESolve.

[1] Ahmet Dogan and Louis H Motz. Saturated-unsaturated 3d groundwater model. i: Development. *Journal of Hydrologic Engineering*, 10(6):492–504, 2005.

[2] Simon A Mathias, Todd H Skaggs, Simon A Quinn, Sorcha NC Egan, Lucy E Finch, and Corinne D Oldham. A soil moisture accounting-procedure with a Richards' equation-based soil texture-dependent parameterization. *Water resources research*, 51(1):506–523, 2015.

[3] Reinder A Feddes, Piotr Kowalik, Krystina Kolinska-Malinka, and Henryk Zaradny. Simulation of field water uptake by plants using a soil water dependent root extraction function. *Journal of Hydrology*, 31(1-2):13–26, 1976.